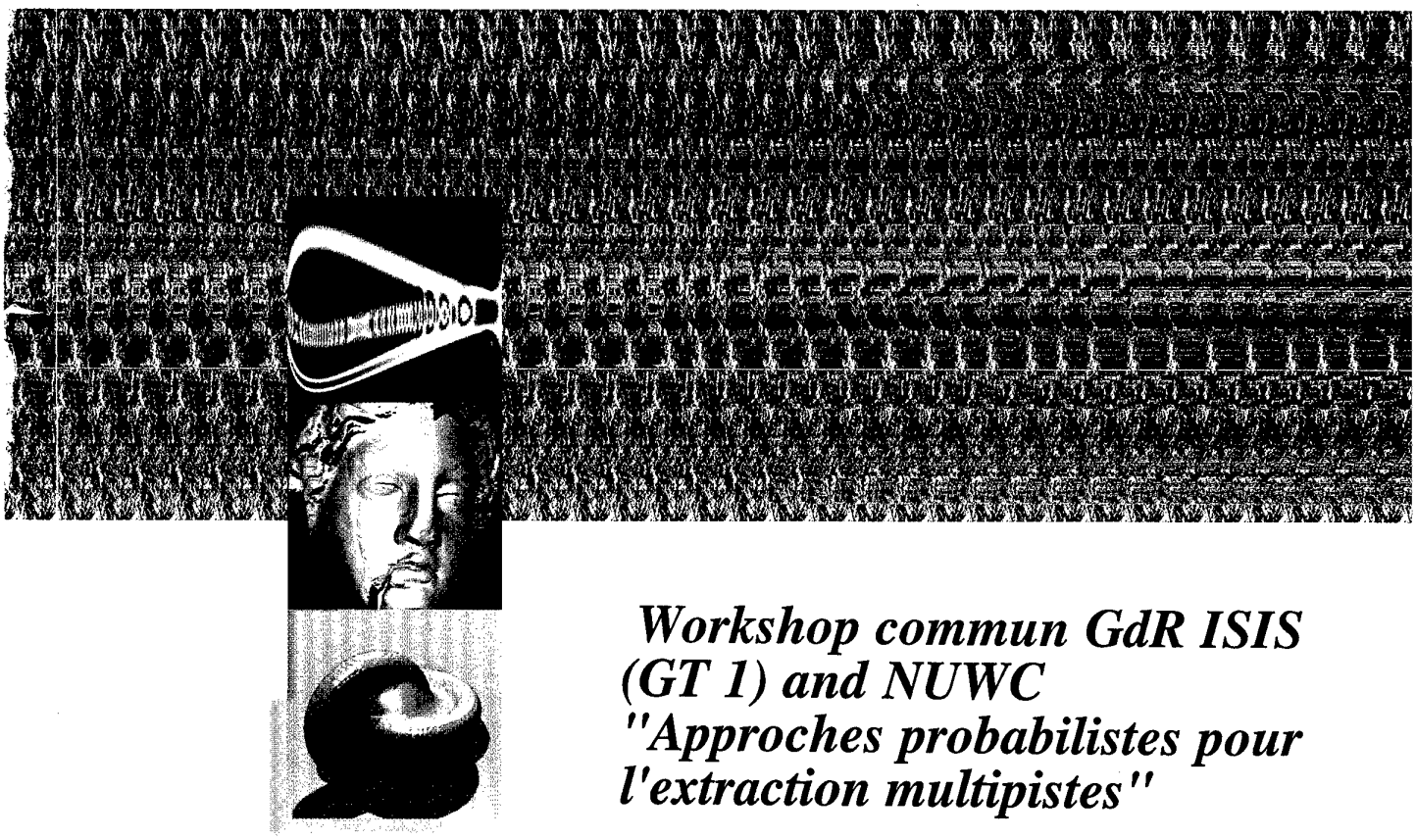
Un Club des partenaires

700 chercheurs

80 équipes

Information
Signal
Images
Vi**S**ion

isis



Workshop commun GdR ISIS (GT 1) and NUWC "Approches probabilistes pour l'extraction multipistes"

19990401 087

CNRS CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

isis

MINISTÈRE DE L'ÉDUCATION NATIONALE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
SECRÉTARIAT D'ÉTAT À LA RECHERCHE

http://www-isis.enst.fr
Établissement d'accueil : ENST Paris

GDR-PRC ISIS

Information Signal Image Vision

DTIC QUALITY INSPECTED 2

Workshop commun GdR ISIS (GT 1) and NUWC "Approches probabilistes pour l'extraction multipistes"

lundi, 9 novembre1998 - mardi, 10 novembre 1998

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	1999	Conference Proceedings 9 -10 Nov 1998

4. TITLE AND SUBTITLE
Probabilistic Methods for Multitarget Tracking

5. FUNDING NUMBERS
N00014-99-1-1015

6. AUTHOR(S)
Editor Dr. Jean-Pierre Lacdre

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Dr. Jean-Pierre Lacdre
IRISA/CNRS
Campus de Beaulieu
35042 Rennes cedex France

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Technical Director
Office of Naval Research International Field Office (ONRIFO)
PSC 802 Box 39
FPO AE 09499-0700

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release, distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
Compilation of abstracts from the conference entitled "Probabilistic Methods for Multitarget Tracking" held in Paris, France from 9 - 10 November 1998.

DTIC QUALITY INSPECTED 8

14. SUBJECT TERMS
Keywords: multitarget tracking, clutter, probabilistic, PMHT, radar, noise, tracking

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. 239.18
Designed using Perform Pro, WHS/DIOR, Oct 94

Le but de ce workshop est d'offrir un panorama relativement exhausif et prospectif de l'utilisation des méthodes d'estimation de paramètres de lois-mélanges dans le cadre de l'extraction multipistes. Ces méthodes ont été développées pour l'estimation conjointe des paramètres (états) des sources et des variables d'affectation des mesures aux pistes.

En effet, pour éviter les problèmes combinatoires, une idée fondamentale (initialement développée par Streit et Luginbuhl) consiste à considérer les variables d'affectation comme des variables aléatoires. Ce type d'approche présente un interêt certain d'une part du fait de sa grande souplesse (extensions aisées au cas multirécepteurs, au MAP, à diverses modélisations de sources et d'observation), et d'autre part du fait de la modération du volume de calcul requis.

Ainsi est-il possible de considérer dans son ensemble une séquence complète d'observations afin d'en extraire les pistes les plus vraisemblables. L'interêt devient évident en présence de "trous" de détection, de croisements difficiles, etc.

Streit et Luginbuhl (NUWC) ont développé la formalisation originale du problème (la méthode PMHT), dont des travaux précurseurs avaient été ceux d'Avitzour et de Salmond et Gordon. Depuis, ces travaux ont été analysés, développés par plusieurs équipes (e.g.~NUWC (USA), NPS (USA), CSSIP (Australie), DERA (UK), IRISA (France), U. Conn. (USA, etc.), dans des contextes variés (poursuite de cibles manoeuvrantes, pistage multicibles en radar, sonar, IR, vision). Ce workshop sera donc l'occasion de confronter les points de vues, d'échanger des idées. Le recueil des textes inclue 13 contributions qui apportent des éclairages différents et complémentaires pour la mise en oeuvre de ce type de méthodes. La diversité des approches, des applications illustre parfaitement la souplesse et la généralité du PMHT.

The aim of this workshop is to present a reasonably wide perspective of the use of mixture densities in the general context of multitarget tracking. These methods have been developed for jointly estimating the source (kinematic) parameters in the first hand and the assignment variables in the second one.

Indeed, to avoid combinatorial problems, a fundamental idea (at first, developed by Streit and Luginbuhl) is to model the assignment variables as random. Even if it corresponds to an approximation, this type of approach has definite advantages, essentially due to its versatility (e.g. see the very wide range of applications) and to reasonable requirements in terms of computation load.

Thus, it allow us to consider directly the whole spatio-temporal sequence of measurements (a batch). This is especially important for reallistic applications of multitarget tracking where numerous specific problems occur (crossings, non-detection,masking, spurious detections, etc.).

The original formulation of the problem (the PMHT method) has been developed by Streit and Luginbuhl (NUWC). Precursory contributions have been that of Avitzour in a first hand and Salmond and Gordon in the second. Since, the basic method has gained popularity and has been widely extended by numerous research teams (e.g. NUWC, Univ. of Connectict, NPS, DERA, IRISA, CSSIP), in varied contexts (radar, sonar, IR, vision, maneuvering sources, etc.). Thus rendering evident the vividity of this approach.

Workshop commun GdR ISIS (GT 1) and NUWC "Approches probabilistes pour l'extraction multipistes"

J. DEZERT et N. LI et X.R. LI. - Theoretical development of an Integrated JPDAF for multitarget tracking in clutter

H. GAUVRIT, C. JAUFFRET et J.P. LE CADRE - Combinatorial Optimization for Initialization of Probabilistic Approaches

M. GELGON, P. BOUTHEMY et J.P. LE CADRE - A PMHT-based approach to associating and estimating trajectories of multiple moving regions in a video analysis context

E. GIANNOPOULOS - Multi Sensor Track Association in Clutter Using

D.A. GRAY et M.L KRIEG - Recursive Least Squares and Kalman Filtering Approaches to Estimating and Tracking the Parameters of Mixture Models

O. GRONDIN et C. MUSSO - Strategy for Radar Pulse Allocation applied to Multitarget Tracking by Metaheuristics

S. PARIS, C. JAUFFRET et G. GOULET - Frequency Line Tracker in passive sonar system

J.P. LE CADRE et H. GAUVRIT - Approximations of the Cramér-Rao Bound for Multiple Target Motion Analysis

T. LUGINBUHL et P. WILLETT - Tracking A General, Frequency Moduladed Signal in Noise

D.VAN CAPPEL, P. ALINAT et J.M. PASSERIEUX - Frequency Line Detection and Tracking Using Multiple Hidden Markov Models

D.J. SALMOND et N.J. GORDON - Tracking in the Presence of Intermittent Spurious Objects and Clutter

R.L. STREIT et M.J. WALSH - Augmented Bearings-Only Target Motion Analysis with Propagation Loss Models

P. WILKETT, Y. RUAN et R. STREIT - A Variety of PMHTs

Theoretical development of an Integrated JPDAF for multitarget tracking in clutter

Jean Dezert
Onera
29 Avenue de la Division Leclerc
92320 Châtillon, France
E-mail: dezert@onera.fr

Ning Li, X.Rong Li
University of New Orleans
New Orleans, LA 70148
Phone: 504-280-7416, Fax: 504-280-3950
E-mail: nylee@uno.edu, xrlee@uno.edu

Abstract

An improved version of Integrated Probabilistic Data Association Filter (IPDAF) based on a new concept of probability of target perceivability was introduced recently in [13] for tracking a single target in clutter. In this paper, we extend the previous theoretical results to the multitarget tracking case to come up with a new integrated version of the Joint Probabilistic Data Association Filter called IJPDAF. Such algorithm provides a new basis of an integrated approach to multiple track initiation, confirmation, termination and maintenance.

1 Introduction

The purpose of multitarget tracking is to estimate the state of several targets based on a set of measurements provided by a sensor. For tracking in a clutter-free environment with perfect data association, targets are always assumed perceivable and measurements are assumed to be available, unique and to arise from a known target at every scan. In such ideal case, multitarget tracking will follow conventional recursive filtering. In practice however the perfect data association assumption is never fulfilled and conventional filtering techniques cannot be used because several measurements are available at every scan and the origin of measurements is uncertain. Moreover tracking targets in clutter involves tracks initiation, confirmation, maintenance and termination. Tracks initiation, confirmation and termination are basically decision problems whereas tracks maintenance is an estimation problem compounded with measurement uncertainty. This paper is only focused on tracks maintenance. Tracks confirmation and termination processes were discussed in [23, 13]. As already pointed out in previous works [8, 27, 19, 20, 13], a fundamental limitation of the Probabilistic Data Association Filter (PDAF) [4, 6] and Joint PDAF [3] is

the implicit strong assumption that targets are always perceivable. Of course in many real situations, this is not the case. To remove the implicit target perceivability assumption made by Bar-Shalom,Tse and Fortmann, a new formulation of IPDAF based on a new concept of target perceivability has been recently proposed for tracking a single target in clutter. We propose now to extend this formulation for the multitarget tracking applications. This new algorithm will be called Integrated JPDAF (IJPDAF) hereafter. A recent Viterbi Data Association (VDA) algorithm [17] has been developed with including target perceivability state within a multitarget tracker. However this method does not take fully into account crossing targets.

2 Brief Integrated PDAF review

At each time step, the sensor provides a set of candidate measurements to be associated with a target t (which may be potentially perceivable or unperceivable) by using a validation gate [4] around the predicted measurement $\hat{\mathbf{y}}^t(k|k-1)$ of the target. There are many different approaches to associate candidate measurements with predicted one. Here we succintly present the Integrated Probabilistic Data Association (IPDA) method [7, 8, 27, 20, 13] for tracking a single target in clutter. IPDAF takes into account the target perceivability presented in [24].

The set of m_k^t candidate measurements $\mathbf{y}_i^t(k)$, $i = 1, \ldots, m_k^t$ at time k is denoted $\mathbf{Y}^t(k)$. The set of all validated measurements for target t up to time k is denoted

$$\mathbf{Y}^{t,k} \triangleq \mathbf{Y}^t(k) \cup \mathbf{Y}^{t,k-1} \tag{1}$$

The corresponding innovations are for $i = 1, \ldots, m_k^t$

$$\tilde{\mathbf{y}}_i^t(k) \triangleq \mathbf{y}_i^t(k) - \hat{\mathbf{y}}^t(k|k-1) \tag{2}$$

If we consider only one target t (we don't care about some other existing targets in the environment) we can introduce the following integrated data association events

$$\begin{aligned}
\mathcal{E}_{-i}^t(k) &\triangleq \bar{O}_k^t \cap \theta_i^t(k) \qquad i = 1, \ldots, m_k^t \\
\mathcal{E}_0^t(k) &\triangleq \bar{O}_k^t \cap \theta_0^t(k) \\
\mathcal{E}_0^t(k) &\triangleq O_k^t \cap \theta_0^t(k) \\
\mathcal{E}_i^t(k) &\triangleq O_k^t \cap \theta_i^t(k) \qquad i = 1, \ldots, m_k^t
\end{aligned}$$

where O_k^t, $\bar{O}_k^t$, $\theta_i^t(k)$ $(i = 0, \ldots, m_k^t)$ correspond to the following exclusive and exhaustive events

$$\begin{aligned}
O_k^t &\triangleq \{\text{target } t \text{ is perceivable}\} \\
\bar{O}_k^t &\triangleq \{\text{target } t \text{ is unperceivable}\}
\end{aligned}$$

and

$$\theta_i^t(k) \triangleq \{y_i^t(k) \text{ comes from target } t\}$$
$$\theta_0^t(k) \triangleq \{\text{none of } y_i^t(k) \text{ comes from target } t\}$$

The IPDA approach [13] is built with the assumptions that the estimation errors have Gaussian densities at each step and the perceivability state can be modeled as a first-order homogeneous Markov-chain. It is also assumed that the target measurement is detected with probability P_d^t and the number of false measurements follows a given distribution μ_F. Moreover false measurements are assumed to be uniformly distributed in measurement space. In the IPDA filtering approach, when $m_k^t > 0$, the conditional mean estimate $\hat{\mathbf{x}}^t(k|k)$ is obtained by

$$\hat{\mathbf{x}}^t(k|k) = \hat{\mathbf{x}}^t(k|k-1) + \mathbf{K}^t(k)\tilde{\mathbf{y}}^t(k) \tag{3}$$

where the combined innovation $\tilde{\mathbf{y}}^t(k)$ is given by

$$\tilde{\mathbf{y}}^t(k) \triangleq \sum_{i=1}^{m_k^t} \beta_i^t \tilde{\mathbf{y}}_i^t(k) \tag{4}$$

where $\beta_i^t(k)$ $(i = \bar{0}, 0, 1 \ldots m_k^t)$ are the posterior integrated association probabilities defined as

$$\beta_i^t(k) \triangleq P\{\mathcal{E}_i^t(k)|\mathbf{Y}^{t,k}\} \tag{5}$$

Expressions for these probabilities can be found in [13]. Probabilities $\beta_{-i}^t(k) = P\{\mathcal{E}_{-i}^t(k)|\mathbf{Y}^{t,k}\}$ are all zeroes since $\mathcal{E}_{-i}^t(k)$ are all empty.

The update of the covariance equation is given by

$$\begin{aligned} \mathbf{P}^t(k|k) &= \beta_{\bar{0}}^t(k)\mathbf{P}^t(k|k-1) \\ &\quad +\beta_0^t(k)[\mathbf{I} + q_0^t\mathbf{K}^t(k)\mathbf{H}^t(k)]\mathbf{P}^t(k|k-1) \\ &\quad +(1-\beta_{\bar{0}}^t(k)-\beta_0^t(k))\mathbf{P}^{t,c}(k|k) + \tilde{\mathbf{P}}^t(k) \end{aligned} \tag{6}$$

with $\mathbf{P}^{t,c}(k|k) = [\mathbf{I} - \mathbf{K}^t(k)\mathbf{H}^t(k)]\mathbf{P}^t(k|k-1)$ and the semi-definite positive stochastic matrix $\tilde{\mathbf{P}}^t(k)$ has the same expression as in standard PDAF [4]

$$\tilde{\mathbf{P}}^t(k) = \mathbf{K}^t(k)\Big[\sum_{i=1}^{m_k^t} \beta_i^t \tilde{\mathbf{y}}_i^t(k)\tilde{\mathbf{y}}_i^{t'}(k) - \tilde{\mathbf{y}}^t(k)\tilde{\mathbf{y}}^{t'}(k)\Big]\mathbf{K}^{t'}(k) \tag{7}$$

q_0^t is a weighting factor given by [16, 21, 20]

$$q_0^t \triangleq \frac{P_d^t(P_g - P_{gg})}{1 - P_d^t P_g} \tag{8}$$

where the gate probability P_g and probability P_{gg} are given by

$$P_g \triangleq P\{\chi^2_{n_y} \leq \gamma\} \tag{9}$$

$$P_{gg} \triangleq P\{\chi^2_{n_y+2} \leq \gamma\} \tag{10}$$

When $m_k^t = 0$, updating equations of the IPDAF are

$$\hat{\mathbf{x}}^t(k|k) = \hat{\mathbf{x}}^t(k|k-1) \tag{11}$$

$$\mathbf{P}^t(k|k) = [\mathbf{I} + q_0^t P^{O^t}_{k|k-1,0} \mathbf{K}^t(k)\mathbf{H}^t(k)]\mathbf{P}^t(k|k-1) \tag{12}$$

where $P^{O^t}_{k|k-1,0}$ is the predicted probability of target perceivability. Its expression can be found in [13]. Prediction of the target state and measurement to time $k+1$ are computed as in the standard Kalman filter [4].

3 Multiple interfering targets

The equations above define the IPDA filter for a single target. Several targets could be handled with multiple copies of the IPDAF. However, with respect to any given target, measurements from interfering targets do not behave at all like the random clutter assumed above. Rather, the probability density of each candidate measurement must be computed based upon the densities of all targets that are close enough to interfere and upon the perceivability of each interfering target.

In order to account for this interdependence, consider now a cluster of targets (established tracks) numbered $t = 1, \ldots, T$ at a given time k. The set of m_k candidate measurements associated with this cluster is denoted

$$\mathbf{Y}(k) = \{\mathbf{Y}^1(k) \cup \ldots \mathbf{Y}^T(k)\} \tag{13}$$

Each measurement $\mathbf{y}_i(k)$ of such cluster belongs either to one perceivable target of the set $t = 1, \ldots, T$ or belongs to the set of false measurements, which will be indexed by $t = 0$ in the sequel.

Denoting $\hat{\mathbf{y}}^t(k|k-1)$ the predicted measurement for target t, the innovation corresponding to measurement i becomes

$$\tilde{\mathbf{y}}_i^t(k) \triangleq \mathbf{y}_i(k) - \hat{\mathbf{y}}^t(k|k-1) \qquad i = 1, \ldots, m_k \tag{14}$$

and the combined innovation becomes

$$\tilde{\mathbf{y}}^t(k) = \sum_{i=1}^{m_k} \beta_i^t(k)\tilde{\mathbf{y}}_i^t(k) \tag{15}$$

where $\beta_i^t(k)$ is the integrated posterior probability that measurement i originated from perceivable target t. $\beta_0^t(k)$ is the probability that none of measurements originated from perceivable target t and $\beta_0^t(k)$ is the probability that

target t is unperceivable at time k by the sensor. This is used in target t's state estimation equation to update the estimate $\hat{\mathbf{x}}^t(k|k)$.

In other words, the integrated Joint Probabilistic Data Association (IJPDA) and IPDA approaches utilize the same estimation equations; the difference is in the way the integrated association probabilities $\beta_i^t(k)$ will be computed. Wheras the IPDA algorithm computes $\beta_i^t(k)$, $i = \bar{0}, 0, \ldots, m_k$ separately for each target t, under the assumption that *all measurements not associated with target t are false* with taking into account the perceibability of the target, the IJPDA algorithm computes $\beta_i^t(k)$ *jointly* across the set of T targets and clutter. From the point of view of any target, this accounts for false measurements arising from both discrete interfering targets and random clutter. Derivation of thes probabilities are given in the next section.

4 Joint association probabilities

The key to the **standard JPDA algorithm** [3, 4] is based on the evaluation of the conditional probabilities of all the following feasible joint events :

$$\Theta(k) = \bigcap_{i=1}^{m_k} \Theta_i^{t_i}(k) \tag{16}$$

where $\Theta_i^{t_i}(k)$ is the event that the measurement i originated from origin t_i, $0 \leq t_i \leq T$. $t_i > 0$ is the index of the target to which measurement i is associated at time k. $t_i = 0$ means that measurement i is a false measurement. The feasible events are those joint events in which no more than one measurement originates from each target.

Actually, another better equivalent expression for a (classical) feasible joint association event $\Theta(k)$ is

$$\Theta(k) = \Big[\bigcap_{i=1}^{m_k} \mathcal{O}_i(k)\Big] \bigcap \Big[\bigcap_{t=1}^{T} \mathcal{P}_t(k) \equiv O_k^t\Big] \tag{17}$$

where $\mathcal{O}_i(k)$ represents the origin (clutter, target 1, ... or target T) of measurement i and $\mathcal{P}_t(k)$ is the perceivability state for target t. In the **standard JPDAF development**, all target are implicitly assumed to be perceivable (i.e. $P\{\mathcal{P}_t(k) = O_k^t\} \equiv 1$). This notation will make more sense for the definition of integrated feasible joint association events in the sequel. Thus, in the standard JPDAF, the previous expression reduces to

$$\Theta(k) = \Big[\bigcap_{i=1}^{m_k} \mathcal{O}_i(k)\Big] \tag{18}$$

The probabilities $\beta_i^t(k)$ that measurement i belongs to target t (implicitly assumed to be perceivable by the sensor) is obtained by summing over all feasible events $\Theta(k)$ for which this condition is true, that is

$$\beta_i^t(k) = \sum_{\Theta(k)} P\{\Theta|\mathbf{Y}^k\}\hat{\omega}_{it}(\Theta) \qquad i = 1, \ldots, m_k \tag{19}$$

$$\beta_0^t(k) = 1 - \sum_{i=1}^{m_k} \beta_i^t(k) \tag{20}$$

where time index k has been dropped for notation convenience and $\hat{\omega}_{it}(\Theta)$ is the corresponding component of the feasible association matrix

$$\hat{\Omega}(\Theta) = [\hat{\omega}_{it}(\Theta)] \tag{21}$$

This event matrix represents a feasible joint association event Θ whenever the following conditions are fulfilled

- any event matrix $\hat{\Omega}(\Theta)$ must be compatible with validation matrix Ω

$$\hat{\omega}_{it}(\Theta) = \begin{cases} 1 & \text{if } \Theta_i^t \in \Theta \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

- a measurement has only one origin

$$\sum_{t=0}^{T} \hat{\omega}_{it}(\Theta) = 1 \qquad \forall i \tag{23}$$

- at most one measurement arises from a target

$$\sum_{i=1}^{m_k} \hat{\omega}_{it}(\Theta) \leq 1 \qquad t = 1, \ldots, T \tag{24}$$

Example

Consider the following validation matrix for two targets

$$\Omega = \begin{array}{c|ccc} t & 0 & 1 & 2 \\ j & & & \\ 1 & 1 & 1 & 0 \\ 2 & 1 & 1 & 1 \\ 3 & 1 & 0 & 1 \end{array} \tag{25}$$

Then the set of feasible joint association matrices is

$$\hat{\Omega}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{\Omega}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{\Omega}_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_6 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_7 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \hat{\Omega}_8 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Generation of such feasible event matrices $\hat{\Omega}$ can be done from the initial validation matrix [6] Ω by using different kind of fast algorithms [14, 11, 29, 10].

Assuming the states of the targets conditioned on the past observations mutually independent, the posterior probability of a feasible joint association event is given by [6]

$$P\{\Theta|\mathbf{Y}^k\} = \frac{1}{c}\frac{\Phi(\Theta)!}{m_k!}\frac{\mu_F(\Phi(\Theta))}{V^{\Phi(\Theta)}}\prod_{i=1}^{m_k}[e_{t_i}(\mathbf{y}_i(k))]^{\tau_i(\Theta)}\prod_{t=1}^{T}[P_d^t]^{\delta_t(\Theta)}[1-P_d^t]^{1-\delta_t(\Theta)} \tag{26}$$

where c is a normalization constant and $\mu_F(\Phi)$ is the probability mass function (pmf) of the number of false measurements and V the volume of the surveillance region.

$e_{t_i}(\mathbf{y}_i(k)) \triangleq \mathcal{N}[\mathbf{y}_i(k); \hat{\mathbf{y}}^{t_i}(k|k-1), \mathbf{S}^{t_i}(k)]$ is the likelihood function of the measurement $\mathbf{y}_i(k)$ associated with target t_i. $\hat{\mathbf{y}}^{t_i}(k|k-1)$ is the predicted measurement for target t_i with associated innovation covariance $\mathbf{S}^{t_i}(k)$.

$\delta_t(\Theta)$, $\tau_i(\Theta)$ and $\Phi(\Theta)$ are respectively the target detection, measurement association and false measurement indicators of the event Θ_k under consideration. These indicators are defined as

$$\delta_t(\Theta) \triangleq \sum_{i=1}^{m_k}\hat{\omega}_{it}(\Theta) \leq 1 \qquad t = 1, \ldots, T \tag{27}$$

$$\tau_i(\Theta) \triangleq \sum_{t=1}^{T}\hat{\omega}_{it}(\Theta) \tag{28}$$

$$\Phi(\Theta) \triangleq \sum_{i=1}^{m_k}[1-\tau_i(\Theta)] \tag{29}$$

According to the model used for the pmf $\mu_F(\Phi)$ of the number of false measurements, two versions of JPDA have been proposed in [3, 4]

- **Parametric JPDA :** If we assume a Poisson pmf for $\mu_F(\Phi)$ which requires the spatial density λ of the false measurements,

$$\mu_F(\Phi) = \frac{(\lambda V)^{\Phi}}{\Phi!}e^{-\lambda V} \tag{30}$$

Thus the joint association probabilities are given by

$$P\{\Theta|\mathbf{Y}^k\} = \frac{1}{c}\prod_{i=1}^{m_k}[\lambda^{-1}e_{t_i}(\mathbf{z}_i(k))]^{\tau_i(\Theta)}\prod_{t=1}^{T}[P_d^t]^{\delta_t(\Theta)}[1-P_d^t]^{1-\delta_t(\Theta)} \quad (31)$$

where c is a new normalization constant.

- **Non parametric JPDA :** If we assume a diffuse prior pmf for $\mu_F(\Phi)$,

$$\mu_F(\Phi) = \epsilon \quad (32)$$

the joint association probabilities are now given by

$$P\{\Theta|\mathbf{Y}^k\} = \frac{\Phi!}{c}\prod_{i=1}^{m_k}[Ve_{t_i}(\mathbf{y}_i(k))]^{\tau_i(\Theta)}\prod_{t=1}^{T}[P_d^t]^{\delta_t(\Theta)}[1-P_d^t]^{1-\delta_t(\Theta)} \quad (33)$$

where c is a new normalization constant.

5 Integrated joint association probabilities

The derivation of the integrated joint posterior probabilities is based on the evaluation of the conditional probabilities of all the feasible integrated joint events which take into account the perceivability of targets involved in the data association process. To clarify this, we give first a simple example of the IJPDA process.

5.1 Example

Consider as previously the following validation matrix for a two targets case

$$\Omega = \begin{array}{c|ccc} & t=0 & 1 & 2 \\ j & & & \\ 1 & 1 & 1 & 0 \\ 2 & 1 & 1 & 1 \\ 3 & 1 & 0 & 1 \end{array} \quad (34)$$

The previous feasible event matrices $\hat{\Omega}(\Theta)$ must now been modified to take into account the perceivability or unperceivability of each target involved in a feasible joint data association event. This can be done by adding a row corresponding to a dummy measurement with indice $j = 0$. This row will describe the perceivability state of each target t. Every binary component $\hat{\omega}_{0t}$ $(t > 0)$ will characterize the perceivability of target t when $\hat{\omega}_{0t} = 1$ and unperceivability of t when $\hat{\omega}_{0t} = 0$. The dummy component $\hat{\omega}_{00}$ can take any arbitrary value. By convention we will always set $\hat{\omega}_{00} = 0$ in the following. Now, the set of integrated feasible event matrices $\hat{\Omega}^I$ can be obtained from the set of (unintegrated) feasible event matrices $\hat{\Omega}$ as follows :

$$\hat{\Omega}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \rightarrow \hat{\Omega}_1^I = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_2^I = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_3^I = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_4^I = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \hat{\Omega}_5^I = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \hat{\Omega}_6^I = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{\Omega}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \rightarrow \hat{\Omega}_7^I = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_8^I = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \hat{\Omega}_9^I = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{\Omega}_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \rightarrow \hat{\Omega}_{10}^I = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_{11}^I = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_6 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \rightarrow \hat{\Omega}_{12}^I = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \hat{\Omega}_{13}^I = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\hat{\Omega}_7 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \hat{\Omega}_{14}^I = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{\Omega}_8 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \rightarrow \hat{\Omega}_{15}^I = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

where the additional following **feasibility constraint** :

(C1) : Any detected target is necessarily perceivable

has been used for generating $\hat{\Omega}^I$ from $\hat{\Omega}$.

Every feasible integrated joint association matrix $\hat{\Omega}_j^I, j = 1 \ldots, 15$ charaterizes an **integrated joint association event** $\mathcal{E}_j$. Let $P(\mathcal{E}_j|\mathbf{Y}^k)$ be the integrated joint posterior probability of $\mathcal{E}_j$. Since events $\mathcal{E}_j$ are exhaustive and mutually exclusive, we always have

$$\sum_j P(\mathcal{E}_j|\mathbf{Y}^k) = 1 \tag{35}$$

When $P(\mathcal{E}_j|\mathbf{Y}^k)$ are known (see next section for details), the marginal integrated association probabilities $\beta_i^t(k), i = \bar{0}, 0, \ldots, m_k$ will be obtained from the integrated joint probabilities by summing over all the integrated joint events $\mathcal{E}_j$ in which the marginal event of interest occurs. In our example, the marginal probabilities for target $t = 1$ and $t = 2$ will be obtained as follows

Marginal integrated association probabilities for target t_1

$$\beta_{\bar{0}}^1(k) = P\{\bar{O}_k^1 \cap \theta_0^1(k)|\mathbf{Y}^k\} = P(\mathcal{E}_1|\mathbf{Y}^k) + P(\mathcal{E}_4|\mathbf{Y}^k) + P(\mathcal{E}_5|\mathbf{Y}^k) + P(\mathcal{E}_{11}|\mathbf{Y}^k)$$
$$\beta_0^1(k) = P\{O_k^1 \cap \theta_0^1(k)|\mathbf{Y}^k\} = P(\mathcal{E}_2|\mathbf{Y}^k) + P(\mathcal{E}_3|\mathbf{Y}^k) + P(\mathcal{E}_6|\mathbf{Y}^k) + P(\mathcal{E}_{10}|\mathbf{Y}^k)$$
$$\beta_1^1(k) = P\{O_k^1 \cap \theta_1^1(k)|\mathbf{Y}^k\} = P(\mathcal{E}_{12}|\mathbf{Y}^k) + P(\mathcal{E}_{13}|\mathbf{Y}^k) + P(\mathcal{E}_{14}|\mathbf{Y}^k) + P(\mathcal{E}_{15}|\mathbf{Y}^k)$$
$$\beta_2^1(k) = P\{O_k^1 \cap \theta_2^1(k)|\mathbf{Y}^k\} = P(\mathcal{E}_7|\mathbf{Y}^k) + P(\mathcal{E}_8|\mathbf{Y}^k) + P(\mathcal{E}_9|\mathbf{Y}^k)$$
$$\beta_3^1(k) = P\{O_k^1 \cap \theta_3^1(k)|\mathbf{Y}^k\} = 0$$

Marginal integrated association probabilities for target t_2

$$\beta_{\bar{0}}^2(k) = P\{\bar{O}_k^2 \cap \theta_0^2(k)|\mathbf{Y}^k\} = P(\mathcal{E}_1|\mathbf{Y}^k) + P(\mathcal{E}_2|\mathbf{Y}^k) + P(\mathcal{E}_8|\mathbf{Y}^k) + P(\mathcal{E}_{13}|\mathbf{Y}^k)$$
$$\beta_0^2(k) = P\{O_k^2 \cap \theta_0^2(k)|\mathbf{Y}^k\} = P(\mathcal{E}_3|\mathbf{Y}^k) + P(\mathcal{E}_4|\mathbf{Y}^k) + P(\mathcal{E}_7|\mathbf{Y}^k) + P(\mathcal{E}_{12}|\mathbf{Y}^k)$$
$$\beta_1^2(k) = P\{O_k^2 \cap \theta_1^2(k)|\mathbf{Y}^k\} = 0$$
$$\beta_2^2(k) = P\{O_k^2 \cap \theta_2^2(k)|\mathbf{Y}^k\} = P(\mathcal{E}_{10}|\mathbf{Y}^k) + P(\mathcal{E}_{11}|\mathbf{Y}^k) + P(\mathcal{E}_{15}|\mathbf{Y}^k)$$
$$\beta_3^2(k) = P\{O_k^2 \cap \theta_3^2(k)|\mathbf{Y}^k\} = P(\mathcal{E}_5|\mathbf{Y}^k) + P(\mathcal{E}_6|\mathbf{Y}^k) + P(\mathcal{E}_9|\mathbf{Y}^k) + P(\mathcal{E}_{14}|\mathbf{Y}^k)$$

One can easily check that

$$\sum_{i=\bar{0},0,1,\ldots,m_k} \beta_i^t(k) = 1 \qquad \forall t = 1, 2 \tag{36}$$

The state estimation equations will be exactly the same as in the IPDAF presented in the first section.

5.2 Derivation of integrated joint probabilities

We define an **integrated joint association event** $\mathcal{E}$ pertaining to the current time k as

$$\mathcal{E}(k) = \Big[\bigcap_{i=1}^{m_k} \mathcal{O}_i(k)\Big] \bigcap \Big[\bigcap_{t=1}^{T} \mathcal{P}_t(k)\Big] \tag{37}$$

where $\mathcal{O}_i(k)$ represents the source (clutter, target 1, ... or target T) of measurement i and $\mathcal{P}_t(k)$ represents the perceivability state for target t ($\mathcal{P}_t(k) = O_k^t$ when target t is perceivable or $\mathcal{P}_t(k) = \bar{O}_k^t$ otherwise). Such event characterizes both the origin of all measurements and the perceivability state of all targets.

Every integrated joint association event $\mathcal{E}(k)$ can be represented by an **integrated event matrix** of size $(m_k + 1) \times (T + 1)$

$$\hat{\Omega}^I(\mathcal{E}) = [\hat{\omega}_{it}(\mathcal{E})] \tag{38}$$

consisting of the units in the validation matrix Ω corresponding to the integrated association in $\mathcal{E}$, i.e. for $t = 0, 1, \ldots, T$ and $i = 1, \ldots, m_k$

$$\hat{\omega}_{it}(\mathcal{E}) = \begin{cases} 1 & \text{if } (\mathcal{O}_i(k) = t) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{39}$$

and for $t = 1, \ldots, T$

$$\hat{\omega}_{0t}(\mathcal{E}) = \begin{cases} 1 & \text{if } (\mathcal{P}_t(k) = O_k^t) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{40}$$

A **feasible integrated association event** is one satisfying the following constraints

(1) a measurement has only one origin, i.e.,

$$\sum_{t=0}^{T} \hat{\omega}_{it}(\mathcal{E}) = 1 \qquad \forall i \tag{41}$$

(2) at most one measurement originates from a (perceivable) target

$$\delta_t(\mathcal{E}) \triangleq \sum_{i=1}^{m_k} \hat{\omega}_{it}(\mathcal{E}) \leq 1 \qquad t = 1, \ldots, T \tag{42}$$

(3) any detected target is necessarily perceivable

$$\hat{\omega}_{0t}(\mathcal{E}) - \delta_t(\mathcal{E}) \geq 0 \qquad t = 1, \ldots, T \tag{43}$$

The binary variable $\delta_t(\mathcal{E})$ is called the **target detection indicator** since it indicates wether target t has been detected under $\mathcal{E}$. The **measurement association indicator** $\tau_i(\mathcal{E})$ and **false measurement indicator** $\Phi(\mathcal{E})$ are defined in the same way as in the JPDA approach, i.e.

$$\tau_i(\mathcal{E}) \triangleq \sum_{t=1}^{T} \hat{\omega}_{it}(\mathcal{E}) \tag{44}$$

$$\Phi(\mathcal{E}) \triangleq \sum_{i=1}^{m_k}[1-\tau_i(\mathcal{E})] \tag{45}$$

The binary component $\pi_t(\mathcal{E}) \triangleq \hat{\omega}_{0t}(\mathcal{E})$ is called the **target perceivability indicator** since it indicates whether the target is perceivable in the integrated joint event $\mathcal{E}$.

The generation of the integrated event matrices $\hat{\Omega}^I(\mathcal{E})$ can be obtained from every (non integrated) feasible event matrice $\hat{\Omega}(\Theta)$ by adding any row $i = 0$ which characterizes the feasible perceivability state for all targets. Hence from every given event matrix $\hat{\Omega}(\Theta)$, we will have to generate N_Θ integrated event matrices $\hat{\Omega}^I(\mathcal{E})$ where

$$N_\Theta = \prod_{t=1}^{T} 2^{1-\delta_t(\Theta)} \tag{46}$$

As in the classical JPDA approach, the evaluation of the integrated joint association event probabilities are obtained with Bayes' formula as follows

$$\begin{aligned} P\{\mathcal{E}|\mathbf{Y}^k\} &= P\{\mathcal{E}|\mathbf{Y}(k), m_k, \mathbf{Y}^{k-1}\} \\ &= \frac{1}{c} p[\mathbf{Y}(k)|\mathcal{E}, m_k, \mathbf{Y}^{k-1}] P\{\mathcal{E}|m_k\} \end{aligned}$$

where c is a normalization constant.

If we assume the states of targets given the past observations mutually independent, the **likelihood function of the measurements** $p[\mathbf{Y}(k)|\mathcal{E}, m_k, \mathbf{Y}^{k-1}]$ is exactly the same as the one derived in the stantard JPDA, i.e.

$$p[\mathbf{Y}(k)|\mathcal{E}, m_k, \mathbf{Y}^{k-1}] = V^{-\Phi(\mathcal{E})} \prod_{i=1}^{m_k} [e_{t_i}(\mathbf{y}_i(k))]^{\tau_i(\mathcal{E})} \tag{47}$$

where $e_{t_i}(\mathbf{y}_i(k)) \triangleq \mathcal{N}[\mathbf{y}_i(k); \hat{\mathbf{y}}^{t_i}(k|k-1), \mathbf{S}^{t_i}(k)]$ is the likelihood function of the measurement $\mathbf{y}_i(k)$ associated with target $t_i = \mathcal{O}_i(\mathcal{E})$. $\hat{\mathbf{y}}^{t_i}(k|k-1)$ is the predicted measurement for target t_i with associated innovation covariance $\mathbf{S}^{t_i}(k)$. V is the volume of the surveillance region.

The **prior probability** of an integrated joint association event is given now by

$$\begin{aligned} P\{\mathcal{E}|m_k\} = \frac{\Phi(\mathcal{E})!}{m_k!} \mu_F(\Phi(\mathcal{E})) \prod_{t=1}^{T} [P_d^t]^{\delta_t(\mathcal{E})} [1-P_d^t]^{1-\delta_t(\mathcal{E})} \\ \times \prod_{t=1}^{T} [P^{\mathcal{O}^t}_{k|k-1,m_k}]^{\pi_t(\mathcal{E})} [1-P^{\mathcal{O}^t}_{k|k-1,m_k}]^{1-\pi_t(\mathcal{E})} \end{aligned} \tag{48}$$

where $\mu_F(\Phi)$ is the probability mass function (pmf) of the number of false measurements. $P^{O^t}_{k|k-1,m_k}$ is the conditional predicted target perceivability which can be easily computed on line (see in [13] for details).

The posterior probability $P\{\mathcal{E}|\mathbf{Y}^k\}$ of an integrated joint association event is thus given by

$$P\{\mathcal{E}|\mathbf{Y}^k\} = \frac{1}{c}\frac{\Phi(\mathcal{E})!}{m_k!}\frac{\mu_F(\Phi(\mathcal{E}))}{V^{\Phi(\mathcal{E})}}\prod_{i=1}^{m_k}[e_{t_i}(\mathbf{z}_i(k))]^{\tau_i(\mathcal{E})} \times \prod_{t=1}^{T}[P_d^t]^{\delta_t(\mathcal{E})}[1-P_d^t]^{1-\delta_t(\mathcal{E})} \times \prod_{t=1}^{T}[P^{O^t}_{k|k-1,m_k}]^{\pi_t(\mathcal{E})}[1-P^{O^t}_{k|k-1,m_k}]^{1-\pi_t(\mathcal{E})} \quad (49)$$

Depending on the model used for the pmf $\mu_F(\Phi)$, two versions of IJPDA can be used

- **Parametric IJPDA :** If we assume a Poisson pmf for $\mu_F(\Phi)$ (which requires the spatial density λ of the false measurements), the integrated joint association probabilities become

$$P\{\mathcal{E}|\mathbf{Y}^k\} = \frac{1}{c}\prod_{i=1}^{m_k}[\lambda^{-1}e_{t_i}(\mathbf{z}_i(k))]^{\tau_i(\mathcal{E})} \times \prod_{t=1}^{T}[P_d^t]^{\delta_t(\mathcal{E})}[1-P_d^t]^{1-\delta_t(\mathcal{E})} \times \prod_{t=1}^{T}[P^{O^t}_{k|k-1,m_k}]^{\pi_t(\mathcal{E})}[1-P^{O^t}_{k|k-1,m_k}]^{1-\pi_t(\mathcal{E})} \quad (50)$$

  where c is a new normalization constant.

- **Non parametric IJPDA :** If we assume a diffuse prior pmf for $\mu_F(\Phi)$, the integrated joint association probabilities become

$$P\{\mathcal{E}|\mathbf{Y}^k\} = \frac{\Phi(\mathcal{E})!}{c}\prod_{i=1}^{m_k}[Ve_{t_i}(\mathbf{z}_i(k))]^{\tau_i(\mathcal{E})} \times \prod_{t=1}^{T}[P_d^t]^{\delta_t(\mathcal{E})}[1-P_d^t]^{1-\delta_t(\mathcal{E})} \times \prod_{t=1}^{T}[P^{O^t}_{k|k-1,m_k}]^{\pi_t(\mathcal{E})}[1-P^{O^t}_{k|k-1,m_k}]^{1-\pi_t(\mathcal{E})} \quad (51)$$

  where c is a new normalization constant.

If we assume the states of the targets given the past mutually independent, one needs the **integrated marginal association probabilities** which are obtained from the integrated joint probabilities by summing over all the integrated joint event $\mathcal{E}$ in which the integrated marginal event of interest occurs. Hence we will have for $t = 1, \ldots, T$

$$\beta_i^t(k) \triangleq P\{O_k^t \cap \theta_i^t(k)\} = \sum_{\mathcal{E}} P\{\mathcal{E}|\mathbf{Y}^k\}\hat{\omega}_{it}(\mathcal{E}) \tag{52}$$

$$\beta_0^t(k) \triangleq P\{O_k^t \cap \theta_0^t(k)\} = \sum_{\mathcal{E}} P\{\mathcal{E}|\mathbf{Y}^k\}[1 - \delta_t(\mathcal{E})]\pi_t(\mathcal{E}) \tag{53}$$

$$\beta_{\bar{0}}^t(k) \triangleq P\{\bar{O}_k^t \cap \theta_0^t(k)\} = \sum_{\mathcal{E}} P\{\mathcal{E}|\mathbf{Y}^k\}[1 - \delta_t(\mathcal{E})][1 - \pi_t(\mathcal{E})] \tag{54}$$

Once the integrated marginal probabilities $\beta_i^t(k)$ $(i = \bar{0}, 0, \ldots, m_k)$ are computed, the state estimation equations similar to those in the IPDAF can be used for track maintenance, termination and confirmation.

5.3 Simplification for IJPDAF

For integrated joint probabilities $P\{\mathcal{E}|\mathbf{Y}^k\}$ evaluations, a huge number of integrated event matrices has to be generated by IJPDAF comparatively to the standard JPDA approach. This could become a severe drawback for IJPDAF specially for heavy clutter/multitarget tracking applications. But even if this remark is perfectly true, we must however point out the fact that only integrated marginal probabilities $\beta_i^t(k)$ are required for track maintenance. The good news is that evaluation of $\beta_i^t(k)$ does not require the generation of all integrated event matrices $\hat{\Omega}^I(\mathcal{E})$ at all but only unintegrated event matrices $\hat{\Omega}(\Theta)$. Actually, it can be shown from (52)-(54) with elementary algebra that $\beta_i^t(k)$ $(i = \bar{0}, 0, 1, \ldots, m_k)$ can be finally expressed as

$$\beta_i^t(k) = \sum_{\Theta} P\{\Theta|\mathbf{Y}^k\} P_{k|k-1,m_k}^{O^t} \prod_{j \neq t} [P_{k|k-1,m_k}^{O^j}]^{\delta_j(\Theta)} \hat{\omega}_{it}(\Theta) \tag{55}$$

$$\beta_0^t(k) = \sum_{\Theta} P\{\Theta|\mathbf{Y}^k\} P_{k|k-1,m_k}^{O^t} \prod_{j \neq t} [P_{k|k-1,m_k}^{O^j}]^{\delta_j(\Theta)} [1 - \delta_t(\Theta)] \tag{56}$$

$$\beta_{\bar{0}}^t(k) = \sum_{\Theta} P\{\Theta|\mathbf{Y}^k\} [1 - P_{k|k-1,m_k}^{O^t}] \prod_{j \neq t} [P_{k|k-1,m_k}^{O^j}]^{\delta_j(\Theta)} [1 - \delta_t(\Theta)] \tag{57}$$

Hence the cost of computation involved in the IJPDAF is almost the same than the cost required within the standard JPDAF. Only the cost of computation of predicted target perceivability probability (which is not computationally greedy) must be add to the computation cost of the standard JPDAF. Moreover expressions (55)-(57) become fully consistent with those of standard JPDAF as soon as target perceivability probabilities tend towards unity.

5.4 Remark

In the preceding, targets' states given the past were assumed mutually independent. Actually, we could also consider the targets' state, given the past, as

correlated and perform a coupled estimation for the targets under consideration. This will yield the IJPDA coupled filter (IJPDACF). Details on this approach for JPDACF can be found in [6]. Furthermore amplitude information can also be included in the IJPDAF without any difficulty by the way already described in [18, 12, 20]. Indeed, if we set $P^{O^t}_{k|k-1,m_k} \equiv 1$ in (55)-(57), we get $\beta^t_0(k) \equiv 0$ and (19)-(20).

6 Conclusion

A new theoretical development of an integrated version of JPDAF has been given here. After a rigorous derivation of integrated joint association probabilities based on an enumeration of all feasible integrated event matrices, we have been able to express the integrated marginal association probabilities in a very simple form which requires only the generation of (unintegrated) event matrices as for the standard JPDAF. This important result shows that the cost of IJPDAF is comparable to JPDAF. Target state estimation is done by the IPDAF equations and track confirmation/termination can be performed using quite new procedures based on sequential probability ratio test (SPRT) or optimal design thresholdings. The IJPDAF formulation is fully consistent with the standard JPDAF when the perceivability probability of each target becomes one. Simulations results and tracking performance analysis of IJPDAF will be reported in forthcoming papers.

References

[1] M. Abramowitz and I.A. Stegun, "Handbook of Mathematical Functions," Dover publications, 1968.

[2] Y. Bar-Shalom and K. Birmiwal, *Variable Dimension Filter for Maneuvering Target Tracking*, IEEE Trans. AES,vol. AES-18, no. 5, Sept. 1982.

[3] T. Fortmann,Y. Bar-Shalom and M. Scheffe, *Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association*, IEEE Journal of Oceanic Engineering, vol. OE-8, no. 3, pp.173-184, July. 1983.

[4] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association.* New York: Academic Press, 1988.

[5] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles,Techniques, and Software.* Boston, MA: Artech House, 1993.

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

[7] S.B. Colegrove and J.K. Ayliffe, "An extension of Probabilistic Data Association to include Track Initiation and Termination," *Convention Digest*, 20th IREE International Convention, Melbourne, pp. 853-856, September 1985.

[8] S.B. Colegrove, A.W. Davis and J.K. Ayliffe, "Track Initiation and Nearest Neighbours Incorporated into Probabilistic Data Association," *Journal Electrical Electronics Engineering, Australia*,vol. 6, no. 3, pp. 191–198, September 1986.

[9] S.B. Colegrove and J.K. Ayliffe, "The Initiation and Maintenance of Target Tracks in a Non-Uniform Cluttered Environment," Defence Science and Technology Organization Technical Report NO ERL-0365-TR, November 1987.

[10] I.J. Cox and M.L. Miller, *On Finding Ranked Assignments with Application to Multitarget Tracking and Motion Correspondence*, IEEE Trans. AES,vol. AES-31, no. 1, pp. 486-489 January 1995.

[11] R. Danchick and G.E. Newman, *A Fast Method for Finding the Exact N-best Hypotheses for Multitarget Tracking*, IEEE Trans. AES,vol. AES-29, no. 2, pp. 555-560, April 1993.

[12] J. Dezert,"Autonomous navigation with uncertain reference points using the PDAF", in Multitarget-Multisensor Tracking : Applications and Advances, Vol 2, Y. Bar-Shalom Editor, Artech House,1992.

[13] J. Dezert, N. Li, X.R. Li, "A new formulation of IPDAF for tracking in clutter", To appear in next European Control Conference, ECC'99, Karlsruhe, Germany, September 1999.

[14] J.L. Fisher, D.P. Casasent, "Fast JPDA multitarget tracking algorithm", Applied Optics, Vol.28, No.2, pp. 371-376, 15 January 1989.

[15] Y. Guézengar, "Radar Tracking in Cluttered Environment applied to Adaptive Phased Array Radar", Ph.D. Dissertation (in French), Nantes University, France, November 16th, 1994.

[16] Y. Guézengar, "Nouvelle formulation des équations du filtre à association probabiliste de données", Revue Traitement du Signal, Vol 13, no 2, pp. 167-176, 1996.

[17] B. La Scala, G.W. Pulford, "Viterbi Data Association Tracking for Over-the-Horizon Radar," *Proc. International Radar Symposium - IRS98*, Vol. 3, pp. 1155-1164, Munich, Germany, September 15-17, 1998.

[18] D. Lerro and Y. Bar-Shalom, "Automated Tracking with Target Amplitude Information," *Proc. 1990 American Control Conference*, San Diego, CA, June 1990.

[19] N. Li, "Development, Analysis, and Design of Intelligent Probabilistic Data Association Filter for Target Tracking in Clutter," *M.S. Thesis*, University of New Orleans, 1997.

[20] X.R. Li and N. Li, "Intelligent PDAF : Refinement of IPDAF for tracking in clutter," *Proc. 29th Southeastern Symposium on Systeme Theory*, Cookeville, Tennessee, pp. 133–137, 1997.

[21] X.R. Li, "Tracking in Clutter with Strongest Neighbor Measurements : I - Theoretical Analysis, Submitted for Journal Publication, 1996.

[22] X.R. Li and N. Li, "Integrated Real-Time Estimation of Clutter Density for Tracking," *Proc. of SPIE Conference on Signal and Data Processing of Smal Targets*, Orlando, Florida, USA, April 1998.

[23] N. Li and X.R. Li, "Theoretical Design of Trackers for Tracking Probability Enhancement," *Proc. of SPIE Conference on Signal and Data Processing of Smal Targets*, Orlando, Florida, USA, April 1998.

[24] N. Li and X.R. Li, "Target Perceivability : An Integrated Approach to Tracker Analysis and Design," *Proc. of Fusion 98 International Conference*, Las Vegas, pp 174-181, July 1998.

[25] D. Musicki and R.J. Evans, "Tracking in clutter using Probabilistic Data Association," *Proc. Radar 92*, International Radar Conference, Brighton, UK, pp. 82–85, October 1992.

[26] D. Musicki, R.J. Evans and S. Stankovic "Integrated Probabilistic Data Association (IPDA)," *Proc. 31st IEEE Conf. Decision and Control*, Tucson, AZ, December 1992.

[27] D. Musicki, R.J. Evans and S. Stankovic "Integrated Probabilistic Data Association," *IEEE Trans. Automatic Control*,vol. AC-39, pp. 1237–1241, June 1994.

[28] A. Wald "Sequential Analysis," *John Wiley and Sons, Inc.*, New York 1947.

[29] B. Zhou and N.K. Bose, *Multitarget Tracking in Clutter : Fast Algorithms for Data Association*, IEEE Trans. AES,vol. AES-29, no. 2, pp. 352-363, April 1993.

Combinatorial Optimization for Initialization of Probabilistic Approaches*

H. Gauvrit[1], C. Jauffret[2] and J.P. Le Cadre[3]
1 LTSI, University of Rennes1, France,
2 GESSY, University of Toulon, France,
3 IRISA/CNRS, Rennes, France.
(e-mail: herve.gauvrit@univ-rennes1.fr)

Abstract

This paper is focussed on the initialization of the probabilistic approaches (the PMHT for the sequel) for multitarget multisensor tracking. Since data association represents the corner stone of any multitarget multisensor tracker, a method is proposed to provide an initialization which guarantees the probabilistic approaches to converge to the global solution. It is based on a solution to the data association problem. Such a solution is obtained by solving a combinatorial optimization problem. Since the dimension of the problem increases linearly with the number of scans and the number of measurements, even a sub-optimal solution can not be obtained for realistic problems. To avoid this drawback, the proposed algorithm solves a vectorial data association problem instead of a measurement to measurement association problem. With some hypotheses, this approach is shown to be quite promising.

Keywords: multitarget multisensor tracking, PMHT, EM algorithm, combinatorial optimization

1 Introduction

Numerous methods have been proposed to solve multitarget multisensor tracking problem. All of them need to face two different problems: data association and estimation of the kinematic parameters of the targets. These methods divide into probabilistic approaches where data association is included in the estimation problem and into combinatorial ones which aim to find a global solution to data association.

On the one hand, for the probabilistic approaches, the main difficulty relies upon the initialization of the algorithm. For example, only two sets of measurements are commonly used to initialize the JPDAF [1]. Moreover, since the JPDAF updates tracks based on the last set of measurements received and based on the predictive estimates of the targets, it may loose some tracks and may not accurately estimate the target in the crossing areas. On the other hand, in the PMHT [9, 10], the exhaustive enumeration of all the hypotheses of association is avoided, first, by introducing discrete variables to model data association and, finally, by making an hypothesis of independency about them. This means that a constraint on the association has been relaxed which leads to more numerous local optima and stationary points for the optimization problem. Thus, since few a priori information is available about the solution, the performance of the algorithm depends mainly on the initialization.

On the other hand, the combinatorial methods try to find a global solution to data association by solving a combinatorial optimization problem. These methods were more recently studied by Pattipati et al. [7] by introducing efficient methods for solving integer optimization problems. It has been shown that multitarget multisensor tracking can be formulated as a combinatorial optimization problem [2], and more precisely, as a N-dimensional assignment problem. Since such a problem is known to be NP-hard as soon as $N \geq 3$, only sub-optimal solutions may be obtained. Unfortunately, even sub-optimal solutions can not be obtained for high values of N (typically, $N \leq 10$).

As a consequence, since data association represents the corner stone of any multisensor multitarget tracking algorihtm, a sub-optimal solution to the combinatorial optimization may be a good initial point for globally optimizing a probabilistic criteria. In this paper, data association is focussed on the crossing areas when a track vanishes due to another track more powerful (passive sonar). Instead of measurement to measurement association, association of measurement vector will be considered in the sequel. It will be called vectorial data association. This new problem must answer the question to know whether the tracks after the crossing are the same than those before. We postulate that as soon as the tracks are well separated, data association consists only in the problem of tracking one target with false-alarms. All the vectorial measurements contain a possible track

*This work has been supported by Direct. Constr. Navales (DCN/Ing), France

embedded in false-alarms, before or after the crossing area. They correspond to a reduced information. The advantage of the method is to reduce the size, N, of the combinatorial problem by focussing on the crossing area (before/after). This is a rough approximation but realistic in order to obtain a sub-optimal solution which will lead to the global solution of the probabilistic approach. For the sequel, the PMHT is retained as the probabilistic approach.

The paper is divided as follow: probabilistic approaches are presented with their limitations in a first part, while a brief review of the combinatorial approaches is proposed in a second part. This approach is then extended to the vectorial case in a third part. The algorithm and the cost function are also described. Some experiments and a discussion will follow.

2 Probabilistic approaches

In the probabilistic approaches, different strategies are commonly applied whether data association is solved based on the last set of measurements received (e.g.: last scan) and the predictive estimates (JPDAF [1]) or based on all the measurements received (MHT [8], PMHT [9]). For the latter, data association results in an exhaustive enumeration of all the hypotheses of association since one measurement originates from one target at most. That is why such a constraint has been removed in PMHT. Multitarget multisensor tracking is then immersed in a mixture density problem where the mixture parameters are the probability that a measurement originates from a target. They are estimated jointly with the target parameters. This formulation of the problem proposed by Streit et al. [9] is very attractive since multitarget tracking as well as multisensor tracking are solved in the same way. For example, the general expression of the likelihood function for T scans and S sensors reduces to [3] :

$$p(\mathcal{Z}|\Phi) = \prod_{s=1}^{S} \prod_{t=1}^{T} \prod_{i_s=1}^{m_t^s} \left\{ \sum_{m=0}^{M} p(z_{i_s}^s|x_m)\ \pi_m(t) \right\}, \qquad (1)$$

where $\mathcal{Z}$ denotes the whole set of measurements, Φ is the parameter vector composed of the target parameters, X, and of the mixture parameters Π. The number of components of the mixture density is supposed known and equalled to M.

For the sequel, the discussion is restricted to the maximization of the likelihood function since in passive sonar, targets are generally supposed to move according to a MRU. Then, the problem reduces to a non-linear optimization problem. However, the problem is too difficult to maximize directly the log-likelihood. For this purpose, EM algorithm is applied [10]. The algorithm proceeds in two steps: the Expectation step (E-step) and the Maximization one (M-step). Numerical studies have demonstrated that the algorithm is very sensitive to the initialization [4]. For example, the algorithm may converge to a local solution. Moreover the algorithm may converge to stationary points of the likelihood. This problem is even more sensitive when solving spatio-temporal data association, that is a multisensor multitarget tracking problem. Different solutions have been proposed to overcome this drawback. One of them consists in maximizing a sequence of functions where the covariance of the measurements is artificially increased at the beginning, then, it is decreased such that the function is equalled to the likelihood function [5]. Nevertheless, no results regarding the convergence of the sequence are available even if the algorithm is more robust numerically. As a consequence, the problem of the initialization still remains. Basically, this problem comes from the lack of a priori information about data association.

3 The combinatorial optimization problem

The previous part has suggested the limits of the probabilistic approaches. Since the main difficulty of tracking multiple targets from multiple sensors relies upton data association, a global solution to this problem is investigated through the combinatorial nature of the problem. Since the aim of this paper is not to present the combinatorial methods, the issues of the formulation of the problem and the issues of the false-alarms will not be discussed here. For this purpose, see [7, 2] for example.

Recent works [2] have shown that the spatio-temporal data association problem arising from multisensor multitarget tracking may be formulated as a combinatorial optimization problem :

$$(\mathbf{P_N}) \qquad \Psi_N = \min_{\rho_{i_1 i_2 \cdots i_N}} \sum_{i_1=0}^{m_1} \cdots \sum_{i_N=0}^{m_N} c_{i_1 i_2 \cdots i_N}\ \rho_{i_1 i_2 \cdots i_N} \qquad (2)$$

subject to

$$\begin{cases} \displaystyle\sum_{i_2=0}^{m_2} \cdots \sum_{i_N=0}^{m_N} \rho_{i_1 i_2 \cdots i_N} & = 1,\ i_1 = 1, \ldots, m_1 \\ \vdots & \vdots \\ \displaystyle\sum_{i_1=0}^{m_1} \cdots \sum_{i_{N-1}=0}^{m_{N-1}} \rho_{i_1 i_2 \cdots i_N} & = 1,\ i_N = 1, \ldots, m_N \\ \rho_{i_1 i_2 \cdots i_N} \in \{0,1\} & \end{cases} \qquad (3)$$

where $c_{i_1 i_2 \cdots i_N} \in \mathbb{R}$ is the cost of the N-tuple $Z_{i_1 i_2 \cdots i_N}$, whereas the binary variable $\rho_{i_1 i_2 \cdots i_N}$ indicates whether the corresponding N-tuple belongs or not to the optimal partition of the measurements. m_n is the number of measurements in the n^{th} set. Moreover, the N sets of constraints are the mathematical expression of the association hypotheses. This problem is called in the litterature a N-dimensional assignment problem where $N = S \times T$ for T scans and S sensors. Thus, the number

of constraints increases linearly with N and the number of measurements inside each scan whereas the number of variables increases exponentially. Thus the problem becomes very rapidly intractable.

However, for small sizes (e.g. $N \leq 10$), sub-optimal solutions may be obtained by Lagrangean relaxation [7]. More precisely, $(N-2)$ sets of constraints are included in the objective function by introducing the Lagrangean vector, u:

$$(\mathbf{R_2}) \qquad \Psi^2(u) = \tag{4}$$
$$\min_{\rho_{i_1\cdots i_N}} \sum_{i_1=0}^{m_1} \cdots \sum_{i_N=0}^{m_N} (c_{i_1\cdots i_N} - u_{3,i_3} - \cdots$$
$$-u_{N,i_N})\,\rho_{i_1\cdots i_N} + \sum_{i_3=0}^{m_3} u_{3,i_3} + \cdots + \sum_{i_N=0}^{m_N} u_{N,i_N}$$

subject to

$$\begin{cases} \sum_{i_2=0}^{m_2} \cdots \sum_{i_N=0}^{m_N} \rho_{i_1\cdots i_N} & = 1,\ i_1 = 1\cdots m_1 \\ \sum_{i_1=0}^{m_1} \sum_{i_3=0}^{m_3} \cdots \sum_{i_N=0}^{m_N} \rho_{i_1\cdots i_N} & = 1,\ i_2 = 1\cdots m_2 \end{cases} \tag{5}$$

The relaxed problem is then solved by an efficient algorithm (for example, the auction algorithm) whereas the dual function is optimized by a sub-gradient method. Moreover, theoritical results assert that the dual-optimal solution verifies the following property :

$$\Psi^2(u) \leq \Psi^{2*} \triangleq \Psi^2(u^*) = \max_{u_3\cdots u_N} \Psi^2(u) \leq \Psi_N.$$

More precisely, this method provides the tightest bound on the solution for the given problem. This property represents the main interest of the method. It is necessary also to notice that the optimal solution of data association is not necessary required since the final aim is to provide an estimate of the target state vectors. In fact, two different solutions of data association, such that each one is near the optimal one, lead approximately to the same target estimates.

4 A vectorial combinatorial optimization problem

4.1 Motivations

In the previous part, the interest of the combinatorial method has been clearly exhibited. However, the dimension of the assignment problem represents the main limitation of this method. On the other hand, if data association is not thought as a measurement to measurement association but instead as a vectorial measurement association, it could be possible to solve biggest problems than those discussed in the previous part. Moreover, the attention would then be focussed on the real

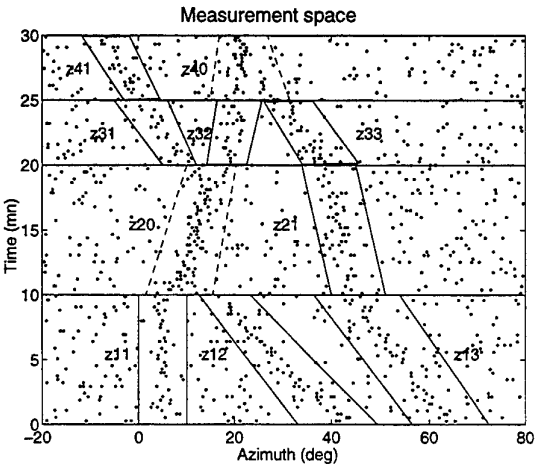


Figure 1: Example of cuts in the measurement space

data association problems that is when the tracks cross. This reasoning is based upon the fact that when tracks are well isolated, data association is not really a difficult problem since it reduces to a track embedded in false-alarms. On the other hand, when tracks cross, due to the different power of the targets, the less powerful track is absorbed by the other one. The problem is then to know whether tracks appearing after the crossing area are the continuation of existing tracks.

To illustrate this idea, consider for example the scenario presented on fig 1. Four areas clearly appear, it is possible to distinguish measurement vectors corresponding to isolated tracks $(\mathbf{z}_{11}, \mathbf{z}_{12}, \mathbf{z}_{13}, \mathbf{z}_{21}, \mathbf{z}_{31}, \mathbf{z}_{32}, \mathbf{z}_{33}, \mathbf{z}_{41})$ from measurement vectors corresponding to crossing areas $(\mathbf{z}_{20}, \mathbf{z}_{40})$. Suppose these measurement vectors have been obtained based on some criteria[1], the data association problem consists then in the assignment of these measurement vectors. In this example, the dimension of the assignment problem is reduced to four. The true association is then the following set of 4-tuples :

$$\{Z_{1020}, Z_{2011}, Z_{3130}\}$$
$$\text{where} \quad Z_{1020} = \{\mathbf{z}_{11}, \mathbf{z}_{20}, \mathbf{z}_{32}, \mathbf{z}_{40}\}$$
$$Z_{2011} = \{\mathbf{z}_{12}, \mathbf{z}_{20}, \mathbf{z}_{31}, \mathbf{z}_{41}\}$$
$$Z_{3130} = \{\mathbf{z}_{13}, \mathbf{z}_{21}, \mathbf{z}_{33}, \mathbf{z}_{40}\}$$

4.2 The cost function

In the combinatorial methods, dummy measurements are introduced to take into account false-alarms and missed detections [7]. In the context described above, dummy measurements represent now, measurement vectors of the crossing areas since the probability density function in these areas, is a mixture of various targets in addition to the false-alarms. This difficult mixture problem will be solved later by the probabilistic approach based

[1] track purity criteria for example [6]

on the initialization obtained by the combinatorial algorithm. Thus, the dummy measurement will not contribute to the cost function of a N-tuple $Z_{i_1 i_2 \cdots i_N}$ in the combinatorial problem. The cost $c_{i_1 \cdots i_N}$ of the N-tuple $Z_{i_1 i_2 \cdots i_N}$ includes only the contribution of the measurement vector of the isolated tracks. Since a measurement from such a vector originates whether from a target or a false-alarm, the likelihood function of $Z_{i_1 i_2 \cdots i_N}$ is defined by:

$$p(Z_{i_1 i_2 \cdots i_N}|x) = \prod_{\substack{n=1 \\ i_n \neq 0}}^{N} p(z_{n,i_n}|x)$$

$$= \prod_{\substack{n=1 \\ i_n \neq 0}}^{N} \prod_{t=t_n}^{T_n} \prod_{j=1}^{m^t_{n,i_n}} \sum_{m=0}^{1} \{\pi_m p(z_j(t)|x)\}$$

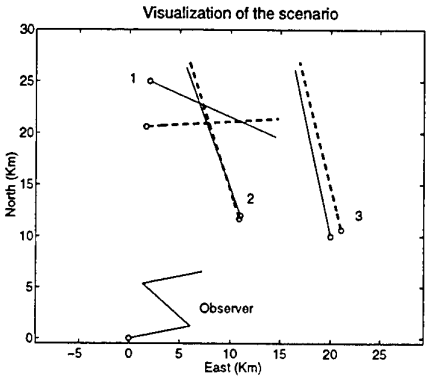
where t_n, T_n are respectively the beginning and the ending of the n^{th} area and m^t_{n,i_n} is the number of measurements at t in the vector. This expression of the likelihood is thus identical to the one defined in the probabilistic approach except that the number of components of the mixture is known for this calculation. A direct calculation based on the classical hypotheses of association [1] could be undertaken like in [5]. Since each N-tuple may contain a different number of measurements, a generalized likelihood ratio is defined as the cost function of the N-tuple $Z_{i_1 i_2 \cdots i_N}$:

$$c_{i_1 \cdots i_N} = -\log \left(\frac{\max_x p(Z_{i_1 i_2 \cdots i_N}|x)}{p(Z_{i_1 i_2 \cdots i_N}|x = \emptyset)} \right) \quad (6)$$

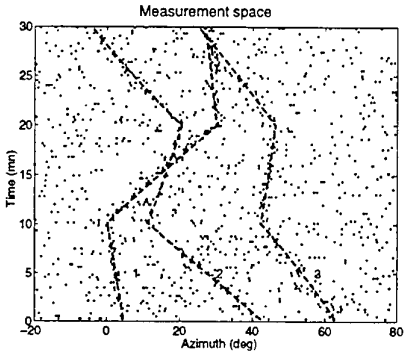
where the alternative represents the hypothesis that all the measurements of the N-tuple ($i_n \neq 0$) are false-alarms.

4.3 Results

The vectorial assignment problem has been studied in the passive sonar context. For this purpose, an hypothesis is added to guarantee the observability of the system: for each target, its track is supposed to contain isolated parts on at least two pieces of the target trajectories corresponding to a maneuver of the observer. In this simulation, isolated parts of the tracks are supposed to contain only the measurements coming from the target. This allows to simplify the probabilistic model used in the calculation of the costs. For the scenario of the example, the same cuts have been done in the measurement space, thus a 4-dimensional assignment problem has to be solved. On figure 2, the result of the solution obtained by the combinatorial agorithm is presented. This solution has been used to initialize the probabilistic algorithm (PMHT). There is no need now to increase the covariance of the measurement to make the algorithm less sensitive to the initialization. There is no need also



(a) Estimated and true trajectories



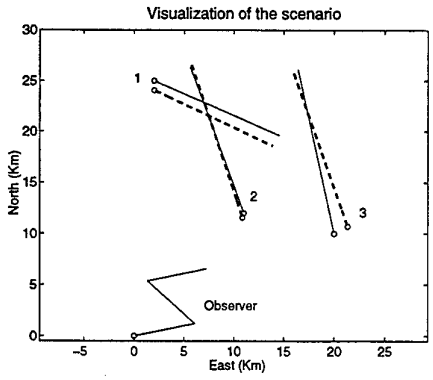
(b) Measurement space and estimated azimuths

Figure 2: Solution of the vectorial combinatorial optimization problem

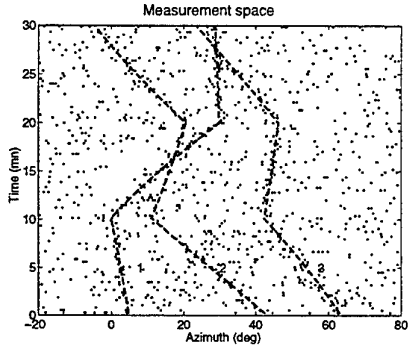
to know the number of components in the mixture since it corresponds to the number of N-tuples of the optimal solution plus one. Moreover, only ten iterations are necessary to estimate the parameters in the crossing areas. Figure 3 presents the results which exhibit that the probabilistic approach has no difficulty to converge to the global solution of the problem. The new solution corresponds to the maximum likelihood estimate of the problem.

5 Conclusion

A novel approach has been proposed to initialize the probabilistic approaches. Since data association is the basic problem in multitarget multisensor tracking, the method is based on a combinatorial optimization problem. However, combinatorial methods can not solve high dimensional assignment problem. To avoid this drawback, the proposed method focusses on the real data association problems by assigning measurement vectors



(a) Estimated and true trajectories



(b) Measurement space and estimated azimuths

Figure 3: Estimation obtained with a PMHT initialized with the combinatorial solution

instead of associating a measurement to a measurement. To this purpose, the measurement vectors have been supposed to be obtained based on some criteria. The optimal solution is then used as an initialization of a PMHT algorithm. The results exhibit clearly that this solution leads the probabilistic algorithm to converge to the global solution. Furthermore, the PMHT algorithm does not need to require hypotheses about the number of components of the mixture since it is obtained as a by-product of the combinatorial algorithm. Finally, some more work needs to be pursued in order to define criteria for obtaining the measurement vectors and in order to evaluate the performance of the algorithm.

References

[1] Y. Bar-Shalom, T. E. Fortmann, *Tracking and data association*, Academic Press, 1988.

[2] S. Deb, M. Yeddanapudi, K. Pattipati, Y. Bar-Shalom, *A generalized S-D assignment algorithm for multisensor-multitarget state estimation*, IEEE Transactions on Aerospace and Electronics Systems, Vol. 33, No. 2, April 1997.

[3] H. Gauvrit, C. Jauffret, J. P. Le Cadre, *A formulation of multitarget tracking as an incomplete data problem*, IEEE Transactions on Aerospace and Electronics Systems, Vol. 33, No. 4, October 1997.

[4] H. Gauvrit, *Extraction multi-pistes: approche probabiliste et approche combinatoire*, PhD thesis, University of Rennes 1, Nov. 1997.

[5] C. Jauffret, Y. Bar-Shalom, *Track formation with bearing and frequency measurements in the presence of false detections*, IEEE Transactions on Aerospace and Electronics Systems, Vol. 26, No. 6, pp. 999-1010, 1990.

[6] S. Mori, K. C. Chang, C. Y. Chong, K. P. Dunn, *Prediction of track purity and track accuracy in dense target environments*, IEEE Transactions on Automatic Control, Vol. 40, No. 5, May 1995.

[7] K. R. Pattipati, S. Deb, Y. Bar-Shalom, R. B. Washburn, *A new relaxation algorithm and passive sensor data association*, IEEE Transactions on Automatic Control, Vol. AC-37, No. 2, pp. 198-213, February 1992.

[8] D. B. Reid, *An Algorithm for Tracking Multiple Targets*, IEEE Transactions on Automatic Control, Vol. AC-24, No. 6, pp. 843-854, December 1979.

[9] R. L. Streit, T. E. Luginbuhl, *A Probabilistic Multi-Hypothesis Tracking Algorithm Without Enumeration and Pruning*, Proceedings of the Sixth Joint Service Data Fusion Symposium, pp. 1015-1024, Laurel, Maryland, 14-18 June 1993.

[10] R. L. Streit, T. E. Luginbuhl, *Maximum Likelihood Method for Probabilistic Multi-Hypothesis Tracking*, SPIE Int. Symposium, Signal and Data Processing of Small Targets 1994, SPIE Proceedings Vol. 2235, Orlando, FL, 5-7 April 1994.

A PMHT-based approach to associating and estimating trajectories of multiple moving regions in a video analysis context

Marc Gelgon[1], Patrick Bouthemy[1] and Jean-Pierre Le Cadre[2]
IRISA/INRIA[1] IRISA/CNRS[2]
Campus universitaire de Beaulieu
35042 Rennes cedex, France
e-mail : mgelgon@irisa.fr, bouthemy@irisa.fr, lecadre@irisa.fr
Tel : (33) 2.99.84.74.72 Fax : (33) 2.99.84.71.71

Abstract

In this paper, we address the issue of estimating the spatio-temporal trajectories of regions from a batch of detection maps. In the context of video content analysis, for instance for surveillance purposes, where moving objects are generally of interest, obtaining their trajectories is of much importance. Besides, if their tracking is disrupted by temporary misdetections, a partial trajectory association issue is combined to the trajectory estimation problem. We address this topic with a Probabilistic Multiple Hypothesis Tracking-based approach, with carries out data association and trajectories estimation as a mixture density parameter estimation using the EM algorithm. This technique is applied to state and measurement vectors including both geometric and kinematic information.

1 Problem statement

In this paper, we address the issue of estimating the spatio-temporal trajectories of regions from a batch of detection maps. In the context of video content analysis, for instance for surveillance purposes, detecting and tracking mobile objects in terms of 2D masks is an important task for a variety of applications (e.g. object recognition, trajectory analysis). Assuming detection maps are provided, the problem is two-fold : first, associate correctly the detected regions over time; second, estimated the trajectory (location and geometry) of each region.

We obtain a batch of detection maps using an approach based on [7], of which fig. 1a,b shows a example. This technique enables detection of mobile zones even in the case of a mobile camera. Spatially disconnected mobile zones are considered as distinct measurements. For each measurement, the following information is available : (1) a region mask, from which we extract a polygonal approximation; (2) a parametric (affine) model of the displacement (motion) between a pair of successive frames; (3) a symbolic region label. The detection algorithm is such that if a region is detected in two successive frames, the region label is maintained, thereby providing a reliable short-term temporal link (e.g. as shown in fig. 1a,b). Because an object may temporarily be static or totally occluded, there may be misdetections that break that temporal link (the region reappears bearing a new label).

A difficulty in estimating the silhouette in the image of the objects in each frame arises from the fact that the measured mask of a detected region may be affected e.g. by moving shadows, or partial occlusion. For instance, in the sequence shown in fig. 1a, the total occlusion (images 14 to 23) is preceded and followed by partial occlusions of the two moving elements.

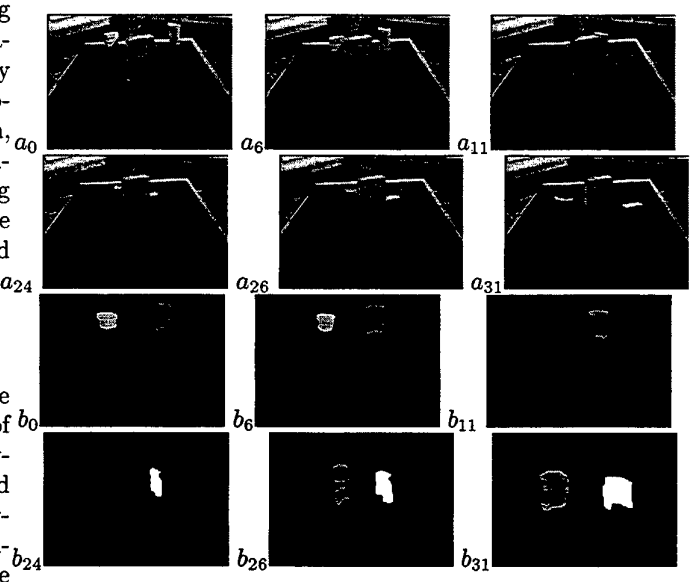


Figure 1: Original images (a) and resulting detection maps (b) at time t=0, t=6, t=11, t=24, t=26 and t=31

2 Problem formulation

A *measurement* in our problem is a detected region, with the information it carries. We shall call *partial track* a set of successive measurements linked over time by their label. One of our goals is to recover actual *tracks*, where a track is a set of measurements corresponding to a single physical moving element. These tracks are composed of one or several partial tracks. From each track has to be derived the *trajectory model* of the mobile element, assuming noisy measurements. The estimation of these trajectories is our second goal.

Let us denote $\mathcal{Z}$ the set of observed measurements in the batch $[0,\ldots,T]$ corresponding to the image sequence. This set and the set $Z(t)$ of the m_t measurements detected at time t decompose as follows :

$$\mathcal{Z} = \Big[Z(1),\ldots,Z(T)\Big] \quad (1)$$

$$Z(t) = \Big\{z_1(t),\ldots,z_{m_t}(t)\Big\} \quad (2)$$

Assume measurements originate from M' mobile elements in the scene. In practise, this number is *a priori* unknown, and is to be determined. We shall see how setting a number of trajectory models M greater than M' is initially sufficient. Once the M trajectory models are estimated, we shall infer redundant models, if any, and hence M'.

Each of the M trajectory models is described by a state vector at each time, and an evolution model related to this state vector. Let us denote $x_m(t)$ the state vector of model m at time t. We also define the set $X(t)$ of state vectors at a given time t and their set $\mathcal{X}$ over the batch, respectively as :

$$\mathcal{X} = \Big[X(1),\ldots,X(T)\Big] \quad (3)$$

$$X(t) = \Big\{x_1(t),\ldots,x_m(t)\Big\} \quad (4)$$

Each region is represented by a geometric (polygonal) model of its contour, and by its kinematics, measured as an affine inter-frame motion model. The state vector $x_m(t)$ and the measurement vector $z_j(t)$ are hence made up of two components :

$$x_m(t) = \Big[\ \mathcal{G}_m(t)\ ,\ \Theta_m(t)\ \Big]^T \quad (5)$$

$$z_j(t) = \Big[\ \tilde{\mathcal{G}}_j(t)\ ,\ \tilde{\Theta}_j(t)\ \Big]^T \quad (6)$$

where

- $\tilde{\mathcal{G}}_j(t) = \{\tilde{P}_j^1(t),\ldots,\tilde{P}_j^{\tilde{n}(t)}(t)\}$ is an ordered set of $\tilde{n(t)}$ vertices resulting from a polygonal approximation of the detected region. This approximation is carried out independently for each of the measurements.
- $\tilde{\Theta}_j(t) = [\tilde{a}_j^1(t)\ ,\ \ldots\ ,\ \tilde{a}_j^6(t)]^T$ is the measured parameter vector of the affine motion model.
- $\mathcal{G}_m(t) = \{P_m^1(t),\ldots,P_m^{n(t)}(t)\}$ and $\Theta_m(t) = [a_m^1(t)\ ,\ \ldots\ ,\ a_m^6(t)]^T$ are respectively the geometric ($n(t)$-vertices polygonal model) and kinematic model of the state vector.

We assume the evolution of these states can be modeled as linear and Markovian, with additive Gaussian white noise. Besides, we consider the measurement of this state is corrupted by Gaussian white noise, which covariance matrix is denoted R_m.We now define the evolution and measurement processes as equations for both the geometric and kinematic components.

- **Kinematic component**

The parameters of the motion model Θ_m are considered decorrelated and are estimated independently. A "constant velocity" evolution model is selected for these parameters (eqn. 7).

$$\begin{bmatrix} a \\ \dot{a} \end{bmatrix}_m^r (t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ \dot{a} \end{bmatrix}_m^r (t) + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix}_m^r (t) \quad (7)$$

where $[\epsilon_1,\epsilon_2]_m^{r\,T}$ is a Gaussian random vector, which covariance matrix Q_ϵ is expressed as :

$$Q_\epsilon = \sigma_\epsilon^2 \begin{bmatrix} \frac{1}{3} & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix} \quad (8)$$

The measurement equation is defined by considering a Gaussian measurement noise $\eta_m^r(t)$ of variance σ_η^2, on each of the motion parameters :

$$\tilde{a}_m^r(t) = a_m^r(t) + \eta_m^r(t) \quad (9)$$

Considering we have no prior knowledge on the kinematics of the moving object, no training set, and that no reliable estimation of the measurement uncertainty is available, σ_ϵ^2 and σ_η^2 are empirically user-set parameters. For real applications, dedicated estimation schemes could be added.

- **Geometric component**

As in [6], we track the geometric model through the set of its vertices. A temporal evolution model of each of these vertices exploits the affine motion model $\widehat{\Theta_{t+1}^t}$ estimated on the region at hand and filtered over time. This evolution is defined by equation 10 :

$$\begin{bmatrix} u \\ v \end{bmatrix}_m^r (t+1) = \begin{bmatrix} 1+a_2 & a_3 \\ a_4 & 1+a_5 \end{bmatrix}_m (t) \begin{bmatrix} u \\ v \end{bmatrix}_m^r (t) + \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}_m (t) + \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix}_m^r (t) \quad (10)$$

where $[\zeta_1,\zeta_2]_m^{r\,T}(t)$ is a Gaussian random vector, which covariance matrix Q_ζ is expressed as :

$$Q_\zeta = \sigma_\zeta^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (11)$$

The measurement $\tilde{P}_j^r(t) = [\tilde{u},\tilde{v}]_j^{r\,T}(t)$ of a vertex is modeled by relation 12 :

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix}_m^r (t) = \begin{bmatrix} u \\ v \end{bmatrix}_m^r (t) + \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}_m^r (t) \quad (12)$$

where measurement noises $\beta_1^r(t)$ and $\beta_2^r(t)$ are assumed to be Gaussian random vectors, of variance σ_β^2.

Again, we set σ_ζ^2 and σ_β^2 by trial-and-error.

Let us recall that our goal is to estimate the trajectories of the detected regions, and to associate partial tracks, if required. Trajectory and data association problem are known to be two tightly interwoven problems. Indeed, the association between partial tracks depends on the trajectories, which in turn must be computed from the whole set of measurements corresponding to a single physical element. The measurement-to-model assignment could be hard, as in MHT algorithms [1]. MHT techniques consist in enumerating possible assignments, evaluate the pertinence of the trajectories formed, and introduce criteria to prune the assignment hypothesis tree. Another classical tool for estimation/data association is JPDAF [1]. However, is it rather a track updating technique. PMHT (Probabilistic Multiple Hypothesis Tracking), which we employ here, offers a new alternative to these classical techniques. This statistical method consists in performing a MAP (Maximum A Posteriori) estimation of the models using Kalman filtering in the case of linear measurements and the EM algorithm for assigning, in a soft probabilitistic manner, measurements to models. Doing so, it avoids the NP-hard combinatorial issue. We refer the reader to [10, 3, 9] for in-depth coverage.

PMHT requires defining the following notations. We call K the set of measurement-to-model associations, which can be decomposed over time and measurements as follows :

$$K = \Big[K(1), \ldots, K(T)\Big] \tag{13}$$

$$K(t) = \Big\{k_1(t), \ldots, k_{m_t}(t)\Big\} \tag{14}$$

An association $k_j(t)$ takes values in $[1, \ldots, M]$, thereby indicating to which model measurement j is assigned.

Let us also introduce Π, the prior probability of models, which can also be decomposed over time and measurements as follows :

$$\Pi = \Big[\Pi(1), \ldots, \Pi(T)\Big] \tag{15}$$

$$\Pi(t) = \Big\{\pi_1(t), \ldots, \pi_M(t)\Big\} \tag{16}$$

Given a measurement at time t, $\pi_m(t)$ represents the priori probability of a measurement being assigned to model m, whatsoever this measurement may be.

In the statistical formulation defined in this paper, the sets $\mathcal{X}$, K and Π respectively contain continuous, random and continuous random variables. Let us consider the two following statistical assumptions on these variables.

- We assume a measurement is associated to one and one model only. The following consequence on assignment variables is inferred :

$$\sum_{m=1}^{M} p\Big(k_j(t) = m\Big) \;=\; \sum_{m=1}^{M} \pi_m(t) = 1 \tag{17}$$

- We assume that at most one measurement can originate from a mobile element at a time. This implies a dependence of assignment variables.

Classical multi-track extraction methods (JPDAF, MHT) are based on these two assumptions, whereas the approach we employ, namely PMHT, relies only on the first one. Consequently, we assume independence of the assignment variables, which enables the decomposition of the joint probability of $K(t)$ as described by equation 18. It is this very decomposition which avoids enumeration of measurement-to-track association hypothesis.

$$p(K(t)) \;=\; \prod_{j=1}^{m_t} p(k_j(t)) \tag{18}$$

3 Essential results of PMHT

The recall in this section the main theoretical aspects of PMHT that are used in our application.

The search for optimal assignments and states being two interlocking issues, Streit [10] proposed to include the data association problem in the estimation problem. Let us define vector $\Phi = (\mathcal{X}, \Pi)$. The $\{\pi_m\}_{m=1,\ldots,M}$ represent the laws of the discrete variables $k_j(t)$, and estimating Φ according to the *maximum a posteriori* criterion comes down to a joint estimation of assignments and states. The *a posteriori* distribution can be expressed by relation 19. Our goal is to find the parameter vector Φ which maximizes this probability.

$$p\Big(\mathcal{Z} \mid \mathcal{X}, \Pi\Big) = p\Big(X(0)\Big) \prod_{t=1}^{T} \underbrace{p\Big(Z(t) \mid X(t), \Pi(t)\Big)}_{\text{measurement likelihood}} \underbrace{p\Big(X(t) \mid X(t-1)\Big)}_{\text{prior state evolution}} \tag{19}$$

Gauvrit and Le Cadre [3] have shown that in the above expression, the measurement likelihood term can be expressed as a mixture density law, in which the parameters weighing the respective contributions of the elementary laws to the mixture as the prior probabilities of the models. The main result is the following :

$$\prod_{t=1}^{T} p\Big(Z(t) \mid X(t), \Pi(t)\Big) \;= \tag{20}$$

$$= \prod_{t=1}^{T} \prod_{j=1}^{m_t} \sum_{m=1}^{M} p\Big(z_j(t) \mid x_m(t)\Big)\pi_m(t) \tag{21}$$

An essential point is that, thanks to the independence assumption between assignment variables, writing

eqn. 20 as a mixture law (eqn. 21) is made possible. We refer the reader to [3] for further detail. Direct maximization of this law is however not feasible, since it is parameterized by the unknown weights $\pi_m(t)$.

Following the work of Redner and Walker [8], the EM algorithm [2] can be employed to estimate the parameters of such a mixture density, through an iterative procedure. Let us assume some initial parameters Φ^0 are available. At the $i+1^{th}$ iteration of the algorithm, in a first step ("E-Expectation" step), an approximation of the *a posteriori* distribution is computed, via its expectation, from measurements and current parameter values Φ^i. In a second step("M-Maximization" step), new parameter values Φ^{i+1} are estimated from the approximation that has just been determined. "E" and "M" steps are alternatively iterated until convergence. We shall specify an appropriate initial parameter configuration for our region trajectography problem.

Some radar and sonar systems are confronted to a problem similar to ours : spatial proximity or other criteria supply a short-term temporal link between measurement. Due to misdetections, the link is sometimes broken, leading to a "segment association problem". We adopt the method proposed by Giannopoulos et al. [4], and summarize below the main results. In this version of the PMHT technique, random association variables are defined between *partial tracks and models*, rather than between *individual measurements and models*. Let us denote $\mathcal{P}$ the set of q partial tracks and $K_l^{\mathcal{P}}$ the assignment of partial track $\mathcal{P}^l$. This assignment takes values in $[1,\ldots,M]$. $\mathcal{P}$ and the set $K^{\mathcal{P}}$ of assignments can be decomposed as follows :

$$\mathcal{P} = \left\{\mathcal{P}_1,\ldots,\mathcal{P}_q\right\} \tag{22}$$

$$K^{\mathcal{P}} = \left\{K_1^{\mathcal{P}},\ldots,K_q^{\mathcal{P}}\right\} \tag{23}$$

The expectation of the *a posteriori* distribution can be expressed as follows :

$$\begin{aligned} Q(\Phi \mid \Phi^i) &= \sum_{m=1}^{M}\sum_{\mathcal{P}_l\in\mathcal{P}} w_{\mathcal{P}_l,m}^{i+1}(t)\ ln[\pi_m(t)] \qquad (24)\\ &+ \sum_{m=0}^{M}\sum_{\mathcal{P}_l\in\mathcal{P}}\sum_{z_j\in\mathcal{P}_l} ln[p(z_j(t)\mid x_m(t))\ ]\ w_{\mathcal{P}_l,m}^{i+1}\\ &+ \sum_{m=1}^{M} ln[p(x_m(0))] + \sum_{m=1}^{M}\sum_{t=1}^{T} ln[p(x_m(t)\mid x_m(t-1))] \end{aligned}$$

where $w_{\mathcal{P}_l,m}^{i+1}$ is a weighing factor corresponding to the probability of assigning partial track $\mathcal{P}_l$ to model m, and is defined as in eqn. 25.

$$w_{\mathcal{P}_l,m}^{i+1} = \prod_{z_j\in\mathcal{P}_l}\left(\frac{\pi_m^i p(z_j\mid x_m(t))}{\sum_{m=1}^{M}\pi_m^i p(z_j\mid x_m(t))}\right) \tag{25}$$

The maximization of $Q(\Phi \mid \Phi^i)$ can be decomposed into two independent maximizations, first with regard to the parameters of the mixture (the $\pi_m(t)$'s), and second with regard to the states (i.e. the models) (the $x_m(t)$'s). Through these maximizations, one updates the estimated parameters $\Phi^i = (\Pi^i, X^i)$ at iteration $i+1$.

The first maximization problem has a simple analytic solution. For every t and m :

$$\pi_m^{i+1}(t) = \frac{1}{m_t}\sum_{j=1}^{m_t} w_{j,m}^{i+1}(t) \tag{26}$$

The second problem consists in the state estimation :

$$\begin{aligned} &(x_m(0),\ldots,x_m(T)) \in\\ &\quad \underset{X_m}{\text{argmax}}\left\{\sum_{\mathcal{P}_l\in\mathcal{P}}\sum_{z_j\in\mathcal{P}_l} ln(\ p(z_j(t)\mid x_m(t))\ )\ w_{j,m}^{i+1}(\right.\\ &\qquad + ln(p(x_m(0)))\\ &\qquad \left.+\sum_{t=1}^{T} ln(p(x_m(t)\mid x_m(t-1)))\right\} \end{aligned} \tag{27}$$

In the case of Markovian processes, it is more relevant to maximize the exponential of this expression :

$$p(x_m(0))\prod_{t=1}^{T}\left\{p(x_m(t)\mid x_m(t-1))\prod_{j=1}^{m_t} p(z_j(t)\mid x_m(t))^{w_{j,m}^{i+1}(t)}\right\} \tag{28}$$

Taking advantage of the Gaussian character of the measurement noise, this expression can be simplified by introducing a fictitious synthetic measurement $\tilde{z}_m(t)$ and its covariance matrix $\tilde{R}_m$, defined below (relations 32 and 33). $\mathcal{N}[\tilde{z}_m(t), x_m(t), \tilde{R}_m]$ denotes a Gaussian probability distribution of variable $\tilde{z}_m(t)$, parameterized by its mean $x_m(t)$ and covariance matrix $\tilde{R}_m$. At each instant t, we have :

$$\prod_{j=1}^{m_t} p(z_j(t)\mid x_m(t))^{w_{j,m}^{i+1}(t)} \tag{29}$$

$$\propto \prod_{j=1}^{m_t}\mathcal{N}[z_j(t), x_m(t), (w_{j,m}^{i+1}(t))^{-1}R_m] \tag{30}$$

$$\propto \mathcal{N}[\tilde{z}_m(t), x_m(t), \tilde{R}_m] \tag{31}$$

$$\tilde{z}_m(t) = \frac{1}{m_t\pi_m^{i+1}(t)}\sum_{j=1}^{m_t} w_{j,m}^{i+1}(t) z_j(t) \tag{32}$$

$$\tilde{R} = \frac{R_m}{m_t\pi_m^{i+1}(t)} \tag{33}$$

This transform leads to the classical expression (eqn. 34) of the *a posteriori* distribution of the state for a *single track*. Its well-known optimal solution is analytic and obtained by Kalman filtering with smoothing.

$$p(x_m(0))\prod_{t=1}^{T}\left\{p(x_m(t)\mid x_m(t-1))p(\tilde{z}_m(t)\mid x_m(t))\right\} \tag{34}$$

The practical resulting algorithm is particularly simple, since the estimation of $\mathcal{X}$ comes down to M independent estimations using Kalman filtering.

4 Algorithm initialization

We set the number of models M as the number of partial tracks found on the batch. The PMHT algorithm requires initializing states and prior probabilities of models. For the latter, we initially set for every instant t and for every model, $\pi_m^0(t) = 1/M$.

We exploit the partial tracks to build the M initial trajectories (initial states). Each model is initially assigned the measurements forming a partial track. Then, we estimate, independently, the M models related to these partial tracks. Figure 2 suggests this operation for three partial tracks. A prediction-only estimation mode is used whenever measurements are not available.

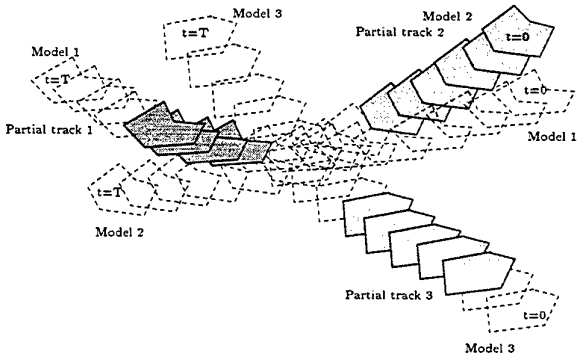


Figure 2: Building initial states, in the case of three partial trajectories (only the geometric component is shown here). Dotted lines represent temporal extents, when a prediction-only mode is employed.

For each trajectory model, a first estimation is carried out for the kinematics and geometric components. In practise, tracking the geometric model is confronted to the following problem. The vertices of the measured geometric model do not correspond to the same physical points over time, and the number of vertices may vary over time. Hence, before they can be used in a prediction/updating procedure with Kalman filters, it is necessary to transform the measured geometric model so that, for each frame of the sequence, a physical correspondence between the *predicted* and the *measured* geometric models. To do so, the two polygons are first spatially registered by minimizing a distance between them.

The geometric model $\mathcal{G}_m(t = 0)$ is initially defined as the measurement $\tilde{\mathcal{G}}_j(t = 0)$ assigned at iteration $i = 0$. Kalman filtering with smoothing is then performed over the extent of the batch. A procedure is included to adjust appropriately over time the number of vertices in the filtered geometric model.

We noticed that the reliability in "prediction-only" mode of the state is strongly dependent on the reliability of the last few measurements before the filter switches to this mode. Typically, these last few measurements can correspond to a progressive occlusion phase. This issue occurs both for progressive appearance and disappearance of a region. The geometric component is particularly affected, since the measured silhouette reveals only the visible part of the object. The purpose of this step is to discard such affected measurements. We carry out detection of occlusion and disoccusion phases as in [5], since this method as shown to be effective. In summary, it consists in detecting unexpected strong temporal variation of the area of the region. By unexpected, we mean that the predicted area at some time t accounts for the apparent divergent motion field (i.e. translation of the camera or of the object along the optical axis). In the innovation signal corresponding to these unexpected variations, we detect upward and downward jumps using Hinkley's test. Besides its simplicity, the interest of this test is two-fold. Because it is cumulative over time, it can detect (dis)occlusion phases with various speeds with the same test parameters. It also provides conveniently the time at which the (dis)occulsion phase starts (which may be a little earlier than the time at which it is detected).

Once the (dis)occulsion phases have been identified, if any, the corresponding measurements are discarded, and the states of all models are re-estimated over the batch.

From these initial states and priori model probabilities, the two steps of the EM algorithm are iterated (computation of the measurement-to-model assignment probabilities given the current states, derivation of priori probabilities of models and of the "synthetic" measurements $\tilde{z}_m(t)$, estimation of the states over the batch). The point is now how well a model initially associated to a particular partial track possibly fits one or several other partial tracks. If this is the case, measurement-to-model assignments evolve over iterations in such a way that these measurements are eventually evenly assigned to several models. If a model does not fit a partial track, then the states end up essentially unchanged. The convergence of the algorithm is established by examining the variations of the measurement-to-model assignment probabilities, and these latter also determine the decision of associating several partial tracks. The trajectories are simply the states at convergence.

5 Experimental results

Results for the "Breakfast" sequence are shown on figures 3 and 4. These results are obtained by setting $\sigma_\beta = 30$. Initial and final trajectories are respectively indicated on fig. 3a and 3b, with measurements. In this example, partial trajectories are associated by physically correct pairs. One can notice that a good initialization was provided by the algorithm. Below, fig. 4a and 4b respectively show the detected geometric models, and the geometric models at convergence, superimposed over the initial image for the first image of the sequence. The algorithm outputs geometric models, including the whole silhouette of the regions at instants when partial or total occlusion take place.

The practical use of the proposed technique is several fold. The understanding of the sequence content is im-

proved and a rich description of the content is provided : trajectories with the whole silhouette of objects and the motion are estimated at each time, including when measurements are either not available, or not reliable.

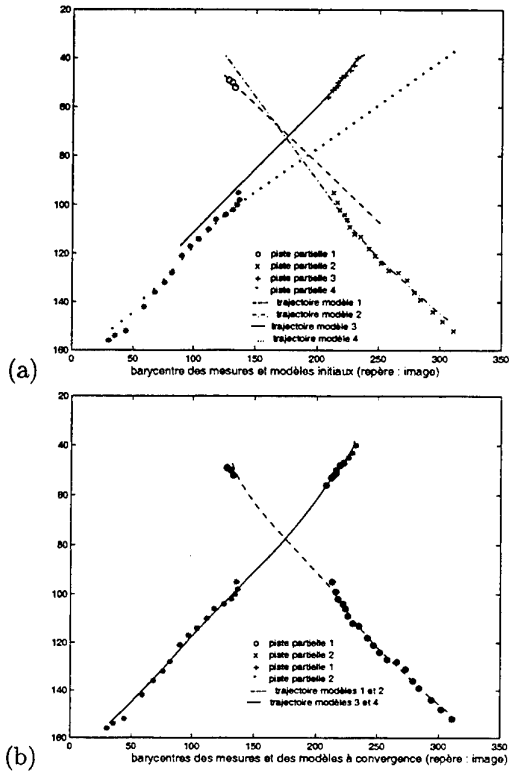


Figure 3: "Breakfast" sequence : measurements and initial trajectories (a) and trajectories at convergence (b). Only the gravity center of the geometric models is shown here.

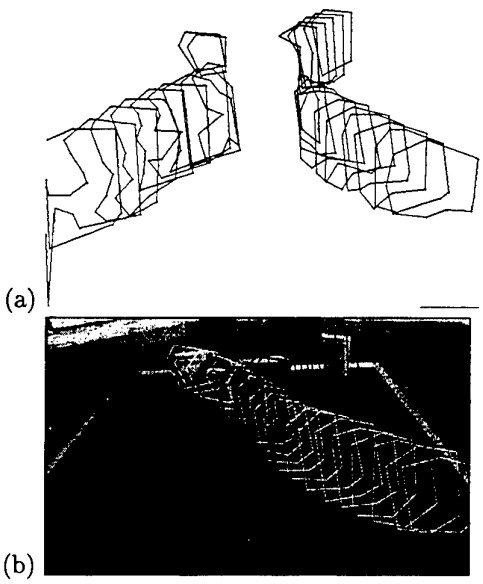


Figure 4: "Breakfast" sequence : measured geometric models (a), estimated geometric models at convergence, suuperimposed on the original image at $t = 0$ (b). On (a) and (b), for the sake of clarity, onyl one out of two geometric models are represented.

References

[1] Y. Bar-Shalom and X.R. Li. – *Estimation and Tracking : Principles, Techniques and Software.* – Artech House, Boston, MA., 1993.

[2] A.P. Dempster, N.M Laird, and D.B. Rubin. – Maximum likelihood from incomplete data via the EM algorithm. – *J. Royal Statistical Society Ser. B*, 39:1–38, 1977.

[3] H. Gauvrit, C. Jauffret, and J.P. Le Cadre. – A formulation of multitarget tracking as an incomplete data problem. – *IEEE Trans. on Aerospace and Electronic Systems*, 33(4), October 1997.

[4] E. Giannopoulos, R. Streit, and P. Swaszek. – Multitarget track segment bearing-only association and ranging. – In *31rst Asilomar conference on Signals, Systems and Computers*, Pacific Grove, USA, November 1997.

[5] F. Meyer and P. Bouthemy. – Exploiting the temporal coherence of motion for linking partial spatio-temporal trajectories. – In *Proc of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 746–747, New-York, June 1993.

[6] F. Meyer and P. Bouthemy. – Region-based tracking using affine motion models in long image sequences. – *CVGIP : Image Understanding*, 60(2):119–140, September 1994.

[7] J-M. Odobez and P. Bouthemy. – Separation of moving regions from background in an image sequence acquired with a mobile camera. – In H.H. Li, S. Sun, and H. Derin, editors, *Video Data Compression for Multimedia Computing*, pages 283–311. Kluwer Academic Publisher, 1997.

[8] R.A. Redner and H.F Walker. – Mixture densities, maximum likelihood and the EM algorithm. – *Society for Industrial and Applied Mathematics*, 26(2), 1984.

[9] R.L. Streit. – *Studies in Probabilistic Multi-Hypothesis Tracking and Related Topics*, volume SES 98-01. – Naval Underwater Warfare Center Division, février 1998.

[10] R.L. Streit and T.E. Luginbuhl. – A probabilistic multi-hypothesis tracking algorithm without enumeration and pruning. – In *Proc. of the 6th Joint Service Data Fusion Symposium*, pages 1015–1024. Laurel, Maryland, juin 1993.

Multi Sensor Track Association in Clutter Using PMHT

Evangelos Giannopoulos *
Naval Undersea Warfare Center
Division Newport
Newport, RI 02841

angelo@npri21.npt.nuwc.navy.mil

Abstract

In this paper we apply PMHT to to the association of track segments from multiple sensors in the presence of clutter. Track association arises in several situations where track level data is available. In a single sensor passive sonar system, tracks from multiple paths and multiple own ship legs need to be associated. In a distributed multi-sensor system, data from each sensor is processed independently and track segments are formed. The track segments are then sent to a centralized fusion system for association. Missed detections as well as false tracks are possible and are handled by the proposed approach. The state vector of the track segments from each sensor can be different. For example, one sensor can be tracking active sonar returns and the other can be tracking passive bearings-only data. The proposed algorithm is applied to the problem of multi-sensor multi-target target motion analysis and is demonstrated with simulated results. The basic algorithm is extended to include additional features in the tracks, when available.

1 Introduction

Track association arises in several situations where track level data is available. In a single sensor passive sonar system, tracks from multiple paths and multiple own ship legs need to be associated. In a distributed multi-sensor system, data from each sensor is processed independently and track

*This work was supported by Office of Naval Research

segments are formed. The track segments are then sent to a centralized fusion system for association. Missed detections as well as false tracks are possible and are handled by the proposed approach. The state vector of the track segments from each sensor can be different. For example, one sensor can be tracking active sonar returns and the other can be tracking passive bearings-only data.

One special case of track association is that of target localization from bearings-only observations in a multi-target and multipath environment. Although extensive work has been done for bearings-only localization algorithms for a single target, there are limited solutions in a multi-target environment. In a single-target environment the associations between segments are known by default, and segmented track bearing data is used by a standard estimation algorithm to localize the target. In a multi-target environment the thorny problem is that of segment association. Segments from multiple targets and from various own ship legs must be associated before each target can be localized. In a dense target and cluttered environment, the number of possible association combinations is large and enumeration is computationally expensive. The problem is exacerbated in a multipath environment.

This algorithm is based on the Probabilistic Multi-Hypothesis Tracking (PMHT) concept, a recently developed data fusion algorithm by Streit and Luginbuhl [1, 2]. Further studies and applications of this concept have been pursued by several authors in [3, 5, 6, 7, 8]. The original PMHT addresses the problem of origin uncertainty in multi-target tracking when multiple measurements are received without target labels, that is, the target of origin of a measurement is unknown. Unlike traditional methods that assign a single measurement to a single target model, it introduces the new and somewhat unconventional concept that every measurement is assigned to every target model with some probability. PMHT is derived by formulating a joint measurement–target assignment density, where the measurement-to-track assignments are modeled by a discrete but unobserved random variable. For details see [2].

The basic concept is extended in this paper to handle track segments, that is, groups of measurements that are believed to originate from a common source and have been previously associated by a standard single target tracking process such as a Kalman filter. A joint density of the measurements and target tracks is still formulated, but instead of using a unique assignment random variable for each measurement, one random variable is used for each target segment. The measurements are then conditionally independent, given the discrete track-segment-to-target-model random

variable. The joint density of the measurements is marginalized over the assignments, and the method of Expectation–Maximization (EM) is used to estimate a set of segment-to-target-model probabilities for each track segment and each target model. These track-segment-to-target-model probabilities are then distributed over each measurement in the segment and become the PMHT measurement-to-target-model probabilities. These are then used to create a 'synthetic' measurement for each target model, where a synthetic measurement is a probabilistically weighted combination of all the observed measurements.

The TSPMHT algorithm (as in the original PMHT) decouples the estimation problem into N (number of target models) independent target estimation problems because the synthetic measurements are conditioned within the iteration on known assignments. The procedure iterates between track-segment-to-target-model assignment estimation, synthetic measurement computation and target localization via a maximum likelihood estimator to converge to a solution. The solution for each target model within the iteration is the standard maximum likelihood (or least squares) solution for a single target using the synthetic measurements.

Under conditions of multipath the track-segment-to-target-model associations become the track-segment-to-path-to-target-model associations. The localization process is thus modified to include the multipath geometry. The TSPMHT algorithm is computationally practical because it avoids enumeration of segment-to-target assignments. The concepts are demonstrated with simulated data.

2 TSPMHT Derivation

The TSPMHT algorithm for track segments without clutter was first derived in [9] It is extended to include clutter in section 3. The derivation is done under the assumption that there is no process noise in the target model. This assumption is not limiting and with minor modifications all results are applicable to the case where process noise is included.

Let there be M targets each of which moves according to the standard discrete time linear model:

$$X_m(t+1) = F_m(t)X_m(t) + G_m(t)u_m(t) \tag{1}$$

for $t = 1, \ldots, T$. In (1) the subscript m is used to represent the trajectory of the m^{th} target. The matrices $F_m(t)$ and $G_m(t)$ are assumed to be known. We observe n_t measurements at time t and sensor s, normally one measurement

at each sensor for each target, and we use the variable r to indicate the r^{th} measurement at each time step. The measurements are based on the following standard observation model:

$$z_{s,r}(t) = H_s(X_m, t) + v_{s,r}(t) \tag{2}$$

The observation function H_s for each sensor s is assumed to be known and is different for each sensor. The added observation random sequence $v_{s,r}(t)$, is assumed to be zero mean, white and Gaussian. Let the $Z = \{z_{s,r}(t)\}$ be the set of all observed measurements, and let $\{Z_g\}_{g=1}^{G_s}$ be a given partitioning of the measurements into G_s track-segments or groups of group length n_g. It is important to note that the track segments can be of different lengths. Let $K = \{k_{sg}\}_{g=1}^{G_s}$ be a set of discrete random variables which are used to indicate an assignment for the g^{th} track-segment from the s^{th} sensor with $s = 1, \ldots, S$, to the m^{th} target model. All the measurements in the track-segment are assigned to that target model. The discrete random variable k_{sg} can take values in the range of $1, \ldots, M$. The assignments for each target-group to target-model are assumed to be independent and the measurements are assumed to be conditionally independent given k_{sg}.

Let $X = \{X_m\}$, $m = 1, \ldots, M$ be the parameter set for all the targets where $X_m = [xo_m, yo_m, vx_m, vy_m]^T$ represents the target initial position and velocity for the m^{th} target. Under the independence assumptions the joint density of the measurements and track-segment assignments parameterized by X is given by

$$P(Z, K) = \prod_{s=1}^{S} \prod_{g=1}^{G_s} p(k_{sg}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sg}; X). \tag{3}$$

The marginal density $P(Z)$ can now be computed by summing equation (3) over the discrete random variables K:

$$P(Z) = \prod_{s=1}^{S} \prod_{g=1}^{G_s} \sum_{k_{sg}=1}^{M} p(k_{sg}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sg}). \tag{4}$$

The conditional density is now easily obtained:

$$P(K|Z) = \prod_{s=1}^{S} \prod_{g=1}^{G_s} w_{s,k_{sg}} \tag{5}$$

The weights represent the track-segment to target-model probabilities and are given by

$$w_{s,k_{sg}} = \frac{p(k_{sg}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sg})}{\sum_{m'=1}^{M} p(k_{sm'}) \prod_{r=1}^{n_g} p(z_{s,r,g}|l)}. \tag{6}$$

Segmenting the data results in having as many weights as track-segments, thus reducing the number of parameters to be estimated. By using (3), (4) and (5) and straightforward manipulation we now obtain the Q function in the E step of the E-M algorithm for the target segments

$$Q(X^{n+1}; X^n) = \sum_{s=1}^{S} \sum_{g=1}^{G_s} \sum_{m=1}^{M} \log p(k_{sg} = m) w_{k_{sg}=m} + \sum_{s=1}^{S} \sum_{g=1}^{G_s} \sum_{r=1}^{n_g} \sum_{m=1}^{M} \log p(z_{s,g,r}|k_{sg} = m) w_{k_{sg}=m}. \tag{7}$$

By rearranging the terms of (7) and including the assumptions that the measurements have Gaussian additive noise we can now relate the track segment Q function to that of the multisensor PMHT Q function:

$$Q(X^{n+1}; X^n) = \sum_{s=1}^{S} \sum_{k_{g=1}}^{G_s} \log p(k_{sg}) w^n_{s,k_{sg}} + \sum_{s=1}^{S} \sum_{t=1}^{T} \sum_{r=1}^{n_t} \sum_{m=1}^{M} (\log N(z_{s,r,t}; H_s(X_m^{n+1}, t)), R_{s,m}) w^n_{s,g,r,m}. \tag{8}$$

where $N(\cdot;\cdot)$ denotes the obvious Gaussian pdf obtained via (2). Equation (8) at this point is of the same functional form as that of the multi sensor PMHT with the main difference being that the measurements are conditioned on the track-segments and are weighted by the track-segment probabilities. For the M-step of the EM process we proceed to take the gradient of the Q function with respect to X^{n+1}. This is easily shown to decouple into the estimation of M targets with synthetic–measurements formed for each target by combining all the measurements based on the computed weights on each individual sensor. See [1, 8] for details. The synthetic–measurements for each target model and the associated covariances are formed by

$$\tilde{z}_{s,t,m} = \frac{\sum_{r=1}^{n_t} \left(w^{n+1}_{s,g,r,m}\right) z_{s,t,r}}{\sum_{r=1}^{n_t} w^{n+1}_{s,g,r,m}}. \tag{9}$$

and

$$\tilde{R}_{s,m} = \frac{R_{s.m}}{\sum_{r=1}^{n_t} w_{s,g,r,m}^{n+1}} \tag{10}$$

It is important to note that synthetic–measurements are formed by combining measurements obtained with each sensor. TSPMHT can now be stated in algorithmic form:

1. Determine the initial values of the parameter set X. This is the initialization step and is critical to successful algorithm performance.
2. Calculate the weights for each sensor for all track-segment to target-model (and path in case of multipath) combinations at each time step.
3. Assign the track segment weights to each measurement that belongs to the track segment.
4. Calculate the synthetic measurement, $\tilde{z}_{s,t,m}$ and the synthetic measurement covariance matrix for each target- model and each sensor.
5. For each target-model apply the appropriate estimation algorithm (in our example the Levenberg–Marquardt nonlinear regression procedure is used).
6. Repeat steps 1-5 until convergence is reached.

3 False Tracks and Homothetic Targets

The basic TSPMHT, as is the case with PMHT, does not include a clutter model. It makes the basic assumption that every track segment comes from a target. Since the track-segment to target-model weights must sum to one all track segments, including those that are false, are assigned to a target with some (most likely) small probability. Clutter in this case refers to false track segments that are detected and generated by existing tracking algorithms. These track segments do not normally originate from the true targets but are mainly due to background noise in the observation space. For example, in the case of bearing only tracking distant shipping could cause real and false tracks to be generated that interfere with the targets of interest. The initiation of these segment is assumed to be Poisson distributed in the bearing time block that TSPMHT operates. Their length tends to be shorter than true target tracks and their bearing rate tends to be uniformly distributed within a lower and an upper bound region.

False track segment characteristics can be determined from the track filter characteristics, the maximum target dynamics and the processing system.

To handle false track segments we propose as in [4, 6] that a dummy target be used as the $M+1$ target model. The density of this "dummy" target-model is assumed to be uniform over the measurement space. Of course, as is the case with the pdf of each track segment g of length n_g, the probability density that a given track segment belongs to clutter depends on n_g, and based on the independence assumption previously stated is given by:

$$p(Z_{sg}|k_{sg} = M+1) = \left(\frac{1}{U_c}\right)^{n_g} \tag{11}$$

For the case of bearing observations , $U_c = 2\pi$. Although this model helps to account for false segments, it introduces a problem with the TSPMHT initialization. If the initial TSPMHT estimate is not close to the truth, the track-segments far away will be attracted to clutter and the TSPMHT solution will be drawn to a local maximum. To counter this effect, and to allow PMHT to look further than its current local estimate, Rago *et al.* [4] proposed that a homothetic target model be used. For each target model, additional modes are added that have the same trajectory (mean) as the main target model by a larger covariance matrix. The covariance matrix of each homothetic mode c is given by $\theta_c R$, where θ_c is a design parameter. This model is adapted here, is combined with uniform clutter model to allow TSPMHT to handle clutter and, at the same time, look for a more global solution.

The track-segment weights for target models $m = 1, ...M$, are now computed for each target model and homothetic mode. The index c is used to denote the homothetic mode associated with each target model.

$$w_{s,g,m,c} = \frac{p(k_{smc}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{smc})}{p(k_{s(M+1)}) \left(\frac{1}{U_c}\right)^{n_g} + \sum_{m'=1}^{M} \sum_{c=1}^{C} p(k_{sm'c'}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sm'c'})}. \tag{12}$$

while the weights that a given segment belongs to clutter, that is, $m = M+1$ are computed by

$$w_{s,g,M+1} = \frac{p(k_{s(M+1)}) \left(\frac{1}{U_c}\right)^{n_g}}{p(k_{s(M+1)}) \left(\frac{1}{U_c}\right)^{n_g} + \sum_{m'=1}^{M} \sum_{c'=1}^{C} p(k_{sm'c'}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sm'c'})}. \tag{13}$$

The synthetic measurements and covariances are only formed for the true target models $m = 1, ..., M$:

$$\tilde{z}_{s,t,m} = \frac{\sum_{c=1}^{C} \sum_{r=1}^{n_t} \left(w_{s,g,r,m,c}^{n+1}\right) z_{s,t,r}/\theta_c}{\sum_{c=1}^{C} \sum_{r=1}^{n_t} w_{s,g,r,m,c}^{n+1}/\theta_c}. \tag{14}$$

and

$$\tilde{R}_{s,m} = \frac{R_{s,m}}{\sum_{c=1}^{C} \sum_{r=1}^{n_t} w_{s,g,r,m,c}^{n+1}/\theta_c}. \tag{15}$$

The following *a priori* probabilities are recommended for use:

$$\pi_{m,c} = \begin{cases} \frac{1}{G*C} & m = 1, ..., M, c = 1, ..., C \\ 1 - \frac{1}{G*C} & m = M+1 \end{cases} \tag{16}$$

4 Extension to Multipath Propagation

TSPMHT is easily extended to handle track segment association and ranging in a multipath environment. The target model equations do not change; however, with each target model we now associate P paths. For this derivation we assume that the paths are discrete. The inclusion of paths affects both the weight computation and the synthetic measurement and synthetic variance formulation as shown below:

$$w_{s,k_{sg},r,m,q} = \frac{p(k_{sg}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sg}) \mid_{k_{sg}=(m,q)}}{\sum_{m'=1}^{M} \sum_{q'=1}^{P} p(k_{sg}) \prod_{r=1}^{n_g} p(z_{s,r,g}|k_{sg}) \mid_{k_{sg}=(m',q')}},$$

$$\tilde{z}_{s,t,m,q} = \frac{\sum_{r=1}^{n_t} \left(w_{s,g,r,m,q}^{n+1}\right) z_{s,t,r}}{\sum_{r=1}^{n_t} w_{s,g,r,m,q}^{n+1}}$$

$$\tilde{R}_{s,m,q} = \frac{R_{s,m,q}}{\sum_{r=1}^{n_t} w_{s,g,r,m,q}^{n+1}}.$$

The synthetic measurements are computed per path as well as per sensor and target model.

5 Feature Incorporation

When standard Kalman filters (or alpha-beta filters) track energy in the bearing-time space, they normally estimate bearing rate as well as bearing position. This information is useful and can be used to alleviate initialization problems as well as distinguish target tracks from clutter. In fact, if other features, such as frequency and frequency rate are available, they can also be included in TSPMHT as well as in the original PMHT. Features are incorporated as an extension of the measurement state vector and are used to estimate the track-segment to target-model weights. The required assumption is that the extended measurement vector be independent over time. The features can be correlated with the measurement observations as long the joint density is known.

Features are especially useful in the multi-sensor bearings-only problem, where bearing (and bearing rate) cannot be predicted without the correct target range. In this case it is possible to estimate parameters that are sensor independent, such as target course, and use these parameters to compute the TSPMHT weights. Of course, selecting the relative importance of the features and finding a joint density when they are jointly distributed requires further research. An alternative approach to using the actual measurements and features is to reduce track segments to a set of sufficient statistics, such as bearing, bearing rate, and target course in the TMA problem and use this feature set to compute the weights. It is then possible to modify the clutter model to include track segment length as a discriminator. Further investigation is recommended for the incorporation features. In this paper we use bearing rate in our simulation and we find it to be helpful in avoiding local maxima in the TSPMHT solution.

6 Simulation Results

The TSPMHT algorithm is demonstrated with a four target scenario in track clutter. The work is continuing and statistical results will be published at a later date in [10]. The scenario geometry and the Maximum Likelihood Estimation (MLE) solution for the case where the assignments are assumed

to be known are shown in Figure 1. Two sensors are originally positioned at $(0,0)$ and $(0,-1000)$ yards respectively in the X,Y plane and observe bearing measurements from the four targets as well as clutter. The sensors move due north in unison with ownship. There is one own ship turn at the middle of the scenario. It is assumed that the targets are not tracked during the ownship turn. The length of the scenario is 30 minutes.

It is assumed that a set of conventional filters (such as a Kalman or an alpha-beta filter) track the targets and report bearing and bearing rate information to a fusion center to be associated. For this simulation the targets are tracked for the entire length of each leg; however this is not required for TSPMHT to operate. In addition to the target tracks, the filters are also producing false tracks. False track initiation is assumed to be Poisson distributed in time with density λ_s, $s = 1, ..., S$. For this particular scenario 50 false tracks per sensor were generated. In general, λ_s, will be scenario dependent. False tracks are assumed to have both a bearing rate, $\dot{\beta}$, and a bearing acceleration, $\ddot{\beta}$. For this simulation $\dot{\beta}$ was uniformly distributed in the range of $[-3, ..., 3]$ (deg / min) and $\ddot{\beta}$ was uniformly distributed in the range of $[-0.4, ..., 0.4]$ (deg / min^2). Track length was modelled to have a Rayleigh density with parameter σ equal to 10% of the maximum track segment length. The additive bearing noise is assumed to be white Gaussian with $\sigma = 0.5^o$. The track segment outputs (measurements) as well as the initial target model bearings from sensor 1 and 2 are shown in Figure 2. The longer track segments belong to the target tracks.

The scenario described above was executed for the same random initialization near the true track, and different TSPMHT parameters settings. The results are summarized in Table 1. The parameters that were controlled were the maximum track segment length n_g, the homothetic model order and the inclusion of bearing rate in the weight computation. The final TSPMHT results were compared to the maximum likelihood solution root-mean-squared error (RMS) with perfect assignments. The ratio of $R = \frac{RMS_{target} - RMS_{MLE}}{RMS_{MLE}}$ was used as a way to determine performance. The results are given in Table 1. In several cases one or two of the tracks were lost. This is stated in the table and the numbers indicate which tracks were lost.

In general the TSPMHT algorithm converged to a reasonable solution when we used a first order homothetic model $c = 2, \theta_c = 4$. It performed even better when the track segment length was limited and the bearing rate was included to compute the weights. For all cases, when a homothetic

	Z	$c=1,\theta_1=1$	$c=2,\theta_2=4$	$c=1,\theta_4=16$
$max\ \ n_g=100$	β	Lost track (2)	2.5	6.0
	$\beta,\dot{\beta}$	Lost track (2,3)	5.1	6.7
$max\ \ n_g=15$	β	Lost Track (2)	3.1	0.2
	$\beta,\dot{\beta}$	0	-0.02	6.9

Table 1: Summary Results for Scenario

target model was used, the algorithm estimated three out of four solutions correctly. Work is continuing to understand the relationship between the parameters and how to best set them in the general case.

Sample results from a correct global solution are shown in Figures 3,4 for the target trajectories and final TSPMHT synthetic measurements respectively. Sample results from an example where TSPMHT did not achieve a global solution are shown in Figures 5,6. The key to this example is that the synthetic measurements for target 2 relied on clutter and were never drawn near the target track segments.

7 Summary

In this paper we presented TSPMHT. The PMHT ideas were extended to operate on track-segments instead of individual measurements. This allows the algorithm to exploit *a priori* knowledge of assignments over time. The concepts were demonstrated via simulations in the problem of multi-target multi-sensor TMA. TSPMHT does not rely on enumeration or pruning methods to associate track segments to target models. This is especially critical for dense target and cluttered environments. Instead, the association and estimation steps are performed jointly in an iterative process. The algorithm was extended to incorporate observed (or estimated) features in the track segments.

References

[1] R. L. Streit and T. E. Luginbuhl. *A Probabilistic Multi-Hypothesis Tracking Algorithm Without Enumeration and Pruning*, Proceedings

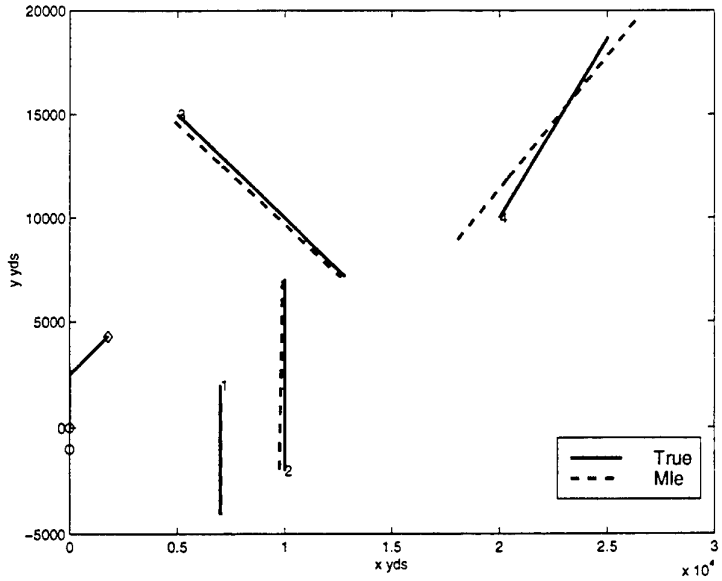


Figure 1: Scenario Geometry and MLE Solution

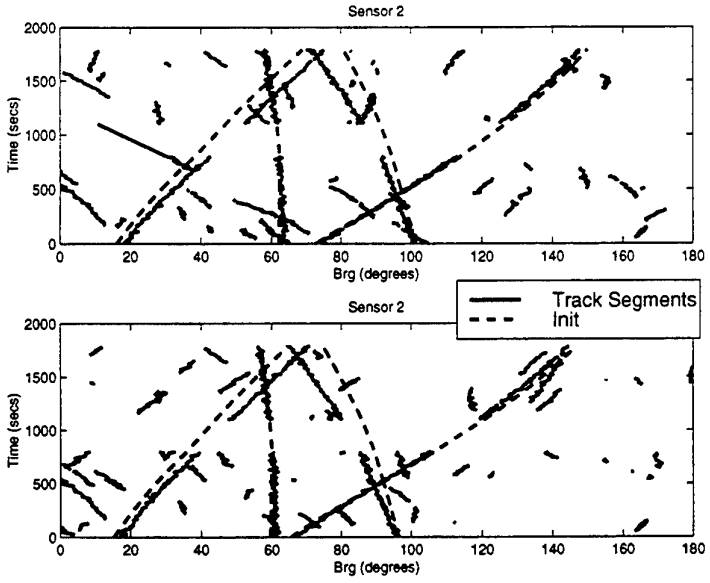


Figure 2: Track Segments and Initial Bearing Projection

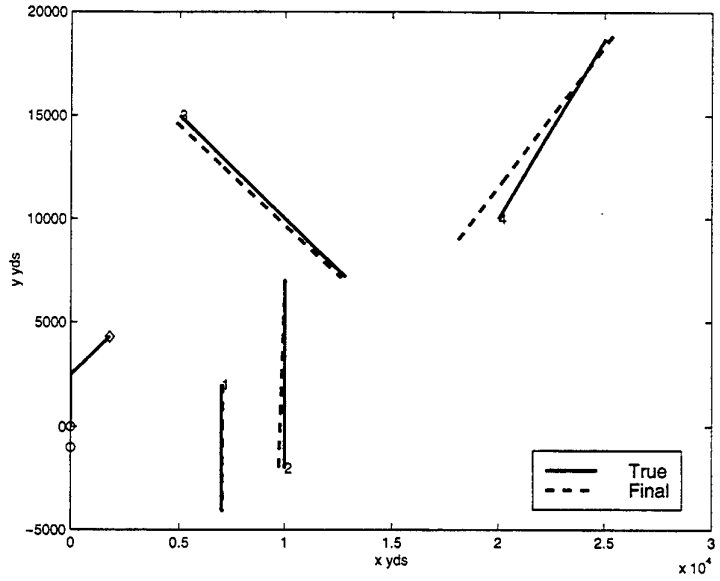


Figure 3: Final TSPMHT Global Solution

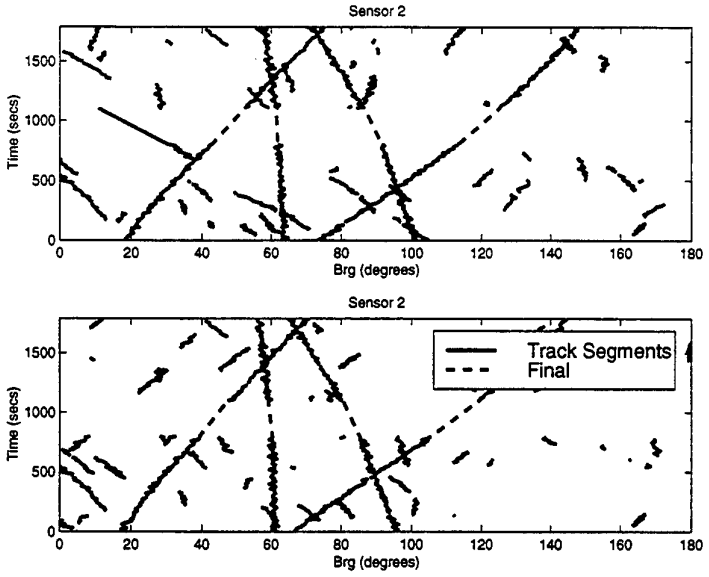


Figure 4: Final TSPMHT Synthetic Measurements for Global Solution

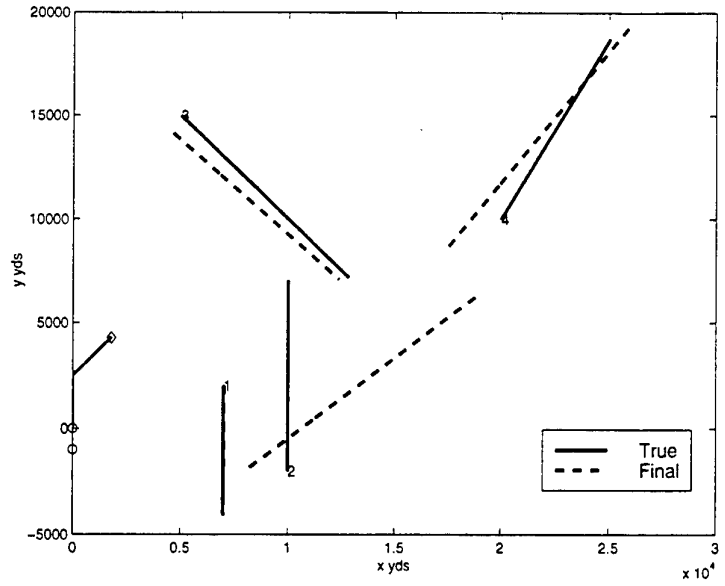


Figure 5: Final TSPMHT Local Solution

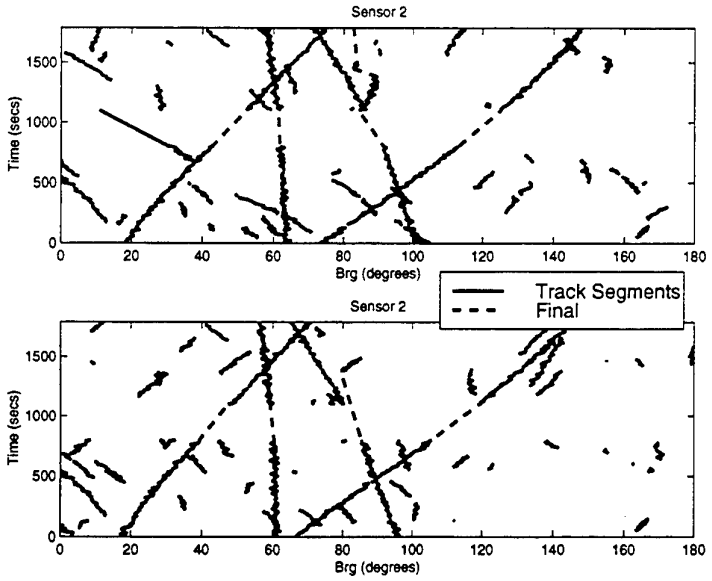


Figure 6: Final TSMPHT Local Solution Synthetic Measurements

of the Sixth Joint Service Data Fusion Symposium, Laurel, Maryland, 14—18 June 1993, 1015–1024.

[2] R. L. Streit and T. E. Luginbuhl. *Probabilistic Multi-Hypothesis Tracking*, Naval Undersea Warfare Center Technical Report 10,428, February 1995.

[3] C. Rago, P. K. Willett, and R. L. Streit. *Direct Data Fusion Using the PMHT*, Proceedings of the 1995 American Control Conference, Seattle WA, June 1995.

[4] C. Rago, P. K. Willett, and R. L. Streit. *A Comparison of the JPDAF and PMHT Tracking Algorithms*, Proceedings of the 1995 ICASSP, May 1995.

[5] D. T. Dunham and R. G. Hutchins. *Tracking Multiple Targets in a Cluttered Environment with a Probabilistic Multiple Hypothesis Tracking Algorithm*, SPIE Proceedings Volume 3080 of the Conference on Acquisition, Tracking and Pointing XI, SPIE Annual International Symposium on Aerospace, Orlando FL, 21-25 April 1997.

[6] H. Gauvrit, C. Jauffret and J.P. Le Cadre. *A General Formulation of Multitarget Tracking in the Viewpoint of an Incomplete Data Problem*, IEEE Transactions on AES, Vol 33, No 4., pg 1242-1257, October 1997.

[7] M. L. Krieg and D. A. Gray. *Multi-Sensor, Probabilistic Multihypothesis Tracking*, Proceedings of the First Australian Data Fusion Symposium, pg. 153–158, Adelaide, Australia, 21-22 November, 1996.

[8] E. H. Giannopoulos, R. L. Streit and P. F. Swaszek. *Probabilistic Multi-Hypothesis Tracking in a Multi-Sensor Multi-Target Environment*, Proceedings of the First Australian Data Fusion Symposium, pg. 184–189, Adelaide, Australia, 21-22 November, 1996.

[9] E. H. Giannopoulos, R. L. Streit and P. F. Swaszek. *Multi-Target Track Segment Bearings-Only Association and Ranging*, Thirty First Asilomar Conference of Signals, Systems and Computers, pg.1336-1340, Pacific Grove, CA, November, 1997.

[10] E. H. Giannopoulos. *Studies in Detection, Estimation and Data Fusion*, Ph.D. Thesis, in preparation.

Recursive Least Squares and Kalman Filtering Approaches to Estimating and Tracking the Parameters of Mixture Models

Douglas A. Gray[1,2] and Mark L. Krieg[1,3]

[1] CRC for Sensor Signal and Information Processing
[2] EEE Dept, University of Adelaide
[3] Defence Science and Technology Organisation, Australia

ABSTRACT

Recursive techniques for estimating and tracking the parameters of mixed linear systems are considered. For this class of problems, the measurement to model assignments must also be estimated at each instant in time. A recursive technique, termed recursive probabilistic least squares, for doing this is derived. This is then shown to lead in a natural way to a recursive Kalman filter.

1. Introduction

In many estimation and tracking problems the observed data is best represented by a mixtures model. For example, in tracking, the sensor measurement at a particular point in time may originate from any one of a number of possible targets but which particular one is unknown and may need to be estimated.

The EM algorithm may be used to estimate both the assignments and the model parameters, and a recent application of this to tracking multiple targets has been the PMHT algorithm of Streit and Luginbuhl [1]. An alternative to the use of the EM algorithm is to use a least squares approach; for example, a probabilistic least squares (PLS) has recently been proposed by Krieg and Gray [2],[3] for tracking problems. A similar batch probabilistic least squares approach when the states are time invariant has also been applied to deinterleaving radar pulses [4].

In this contribution a recursive form of the PLS technique is derived for a general time varying linear mixtures problem. The solution is a generalisation of that derived in [4], in which both the unknown states and the measurement associations are estimated. For each model, the solution for the state estimates is a generalisation of the usual recursive least squares solution - with the unknown associations being included in an intuitive manner. The unknown associations can be simply calculated at each time instant; their dependency on previous estimates being through the current state estimates.

This approach is then used to infer the general form of a Kalman filter for this class of problems and its relationship to the Kalman smoothers in [1] and [2] is discussed in Section 5.

2. Mixed Linear Models

Consider a mixture model, where, at time t_i, the measurements $\underline{z}(i)$ are generated by one of M linear models, i.e.,

$$\underline{z}(i) = \begin{cases} H^{(1)}(i)\underline{x}^{(1)} + \underline{w}^{(1)}(i) \\ H^{(2)}(i)\underline{x}^{(2)} + \underline{w}^{(2)}(i) \\ \vdots \\ H^{(M)}(i)\underline{x}^{(M)} + \underline{w}^{(M)}(i) \end{cases}$$

depending on which model generates the i-th data point. For each linear model, the time varying $H^{(\rho)}(i)$ are assumed known and the unknown but time invariant parameters (states), $\underline{x}^{(\rho)}$, are to be estimated. We can define a set of assignments, $\alpha^{(\rho)}(i)$, that are zero unless the measurement $\underline{z}(i)$ at time t_i is generated by the ρ-th model, in which case $\alpha^{(\rho)}(i) = 1$. This implies

$$\underline{z}(i) = H^{(\rho)}(i)\underline{x}^{(\rho)} + \underline{w}^{(\rho)}(i) = \sum_{v=1}^{M} \alpha^{(v)}(i)\{H^{(v)}(i)\underline{x}^{(v)} + \underline{w}^{(v)}(i)\}$$

where $\alpha^{(v)}(i) = \delta_{\rho v}$ and the $\underline{w}^{(\rho)}(i)$ represent independent noise processes, each with a noise covariance $R^{(\rho)}(i)$.

In general, the assignments are not known and must, along with the states, be estimated from the data. The problem of estimating both the unknown parameters $\underline{x}^{(\rho)}$ and the weights $\alpha^{(\rho)}(i)$ has been termed probabilistic least squares (PLS), and a batch algorithm for this was presented in [4].

3. Notation

To develop a recursive algorithm, we first consider a batch of measurements up to time t_k, and then, following [5], show how the optimum batch solution can be updated to provide the optimum batch solution at time t_{k+1}.

Define

$$\underline{z}_k = \begin{bmatrix} \underline{z}(1) \\ \underline{z}(2) \\ \vdots \\ \underline{z}(k) \end{bmatrix}, \qquad H_k^{(\rho)} = \begin{bmatrix} H^{(\rho)}(1) \\ H^{(\rho)}(2) \\ \vdots \\ H^{(\rho)}(k) \end{bmatrix} \qquad \text{and} \qquad \underline{\varepsilon}_k = \begin{bmatrix} \underline{z}_k - H_k^{(1)}\hat{\underline{x}}_k^{(1)} \\ \underline{z}_k - H_k^{(2)}\hat{\underline{x}}_k^{(2)} \\ \vdots \\ \underline{z}_k - H_k^{(M)}\hat{\underline{x}}_k^{(M)} \end{bmatrix}$$

where $\hat{\underline{x}}_k^{(\rho)}$ denotes an estimate of $\underline{x}^{(\rho)}$using all the data to time t_k, i.e., the $\underline{z}_k$.

The batch solution using all the data up to time t_{k+1} can be found by minimising

$$J_{k+1} = \sum_{i=1}^{k+1} \sum_{\rho=1}^{M} \{\alpha^{(\rho)}(i)\}^2 \{\underline{\varepsilon}^{(\rho)T}(i) R^{(\rho)-1}(i) \underline{\varepsilon}^{(\rho)}(i)\}$$

where

$$\underline{\varepsilon}^{(\rho)}(i) = \underline{z}(i) - H^{(\rho)}(i)\underline{x}^{(\rho)}.$$

subject to the constraint

$$\sum_{\nu=1}^{M} \alpha^{(\nu)}(i) = 1 \quad \forall \quad i.$$

The solution [4] is given by the pair of equations

$$\hat{\alpha}^{(\rho)}(i) = \frac{\left(\underline{\varepsilon}^{(\rho)T}(i) R^{(\rho)-1}(i) \underline{\varepsilon}^{(\rho)}(i)\right)^{-1}}{\sum_{\mu=1}^{M} \left(\underline{\varepsilon}^{(\mu)T}(i) R^{(\mu)-1}(i) \underline{\varepsilon}^{(\mu)}(i)\right)^{-1}} \tag{1}$$

and

$$\hat{\underline{x}}_{k+1}^{(\rho)} = \left[H_{k+1}^{(\rho)T} E_{k+1}^{(\rho)} H_{k+1}^{(\rho)}\right]^{-1} H_{k+1}^{(\rho)T} E_{k+1}^{(\rho)} \underline{z}_{k+1} \tag{2}$$

where

$$E_k^{(\rho)} = A_k^{(\rho)} R_k^{(\rho)-1} A_k^{(\rho)}$$

and where

$$A_k^{(\rho)} = blockdiagonal\{\alpha^{(\rho)}(1)I, \alpha^{(\rho)}(2)I, \cdots, \alpha^{(\rho)}(k)I\}$$

and

$$R_k^{(\rho)} = blockdiagonal\{R^{(\rho)}(1), R^{(\rho)}(2), \cdots, R^{(\rho)}(k)\}.$$

{Aside : The above formulation assumes that each component of the vector of measurements comes from the same model; this can trivially be relaxed to allow the more general case which is of some practical interest.}

4. Derivation of Recursive Probabilistic Least Squares Estimator.

Since the derivation is essentially a generalisation of that given in [5] we only sketch the approach.

Noting that $\underline{z}_{k+1} = \begin{bmatrix} \underline{z}_k \\ \underline{z}(k+1) \end{bmatrix}$ etc it follows that

$$H_{k+1}^{(\rho)T} E_{k+1}^{(\rho)} \underline{z}_{k+1} = H_k^{(\rho)T} E_k^{(\rho)} \underline{z}_k + \{\alpha^{(\rho)}(k+1)\}^2 H^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) \underline{z}(k+1)$$

Defining

$$P_{k+1}^{(\rho)-1} = H_{k+1}^{(\rho)T} E_{k+1}^{(\rho)} H_{k+1}^{(\rho)}$$

and

$$S_{k+1}^{(\rho)} = H^{(\rho)}(k+1) P_k^{(\rho)} H^{(\rho)T}(k+1) + \{\alpha^{(\rho)}(k+1)\}^{-2} R^{(\rho)}(k+1)$$

and

$$W_{k+1}^{(\rho)} = P_k^{(\rho)} H^{(\rho)T}(k+1) S_{k+1}^{(\rho)-1}$$

it follows that

$$\begin{aligned}
\underline{\hat{x}}_{k+1}^{(\rho)} &= P_{k+1}^{(\rho)} H_{k+1}^{(\rho)T} E_{k+1}^{(\rho)} \underline{z}_{k+1} \\
&= P_{k+1}^{(\rho)} \left\{ H_k^{(\rho)T} E_k^{(\rho)} \underline{z}_k + \{\alpha^{(\rho)}(k+1)\}^2 H^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) \underline{z}(k+1) \right\} \\
&= \underline{\hat{x}}_k^{(\rho)} + \{\alpha^{(\rho)}(k+1)\}^2 P_{k+1}^{(\rho)} H^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) \underline{z}(k+1) - W_{k+1}^{(\rho)} H^{(\rho)}(k+1) \underline{\hat{x}}_k^{(\rho)}
\end{aligned}$$

by virtue of the identity

$$P_{k+1}^{(\rho)} = (I - W_{k+1}^{(\rho)} H^{(\rho)}(k+1)) P_k^{(\rho)}.$$

After a further use of the matrix inversion lemma the above recursion for the state estimates can be shown to reduce to

$$\underline{\hat{x}}_{k+1}^{(\rho)} = \underline{\hat{x}}_k^{(\rho)} + W_{k+1}^{(\rho)} \{ \underline{z}(k+1) - H^{(\rho)}(k+1) \underline{\hat{x}}_k^{(\rho)} \} \tag{3}$$

Note that the $\alpha^{(\rho)}(k+1)$'s can be calculated directly as

$$\hat{\alpha}^{(\rho)}(k+1) = \frac{\left(\underline{\varepsilon}^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) \underline{\varepsilon}^{(\rho)}(k+1) \right)^{-1}}{\sum_{\mu=1}^{M} \left(\underline{\varepsilon}^{(\mu)T}(k+1) R^{(\mu)-1}(k+1) \underline{\varepsilon}^{(\mu)}(k+1) \right)^{-1}} \tag{4}$$

As is their wont, the equations are nonlinear but coupled in such a manner that allows, at each time instant, an iterative, but computationally demanding, solution.

The iteration at time instant t_{k+1} may be summarised as

(1) Initialise the parameter estimates, i.e., $\underline{\hat{x}}_{k+1}^{(\rho)}(0) = \underline{\hat{x}}_k^{(\rho)}$

(2) For $n = 1, 2, \ldots$ till convergence

(2.1) Form $\underline{\varepsilon}^{(\rho)}(k+1,n) = \underline{z}(k+1) - H^{(\rho)}(k+1) \underline{\hat{x}}_{k+1}^{(\rho)}(n-1)$.

(2.2) Calculate $\hat{\alpha}^{(\rho)}(k+1,n) = \dfrac{\left(\underline{\varepsilon}^{(\rho)T}(k+1,n) R^{(\rho)-1}(k+1) \underline{\varepsilon}^{(\rho)}(k+1,n) \right)^{-1}}{\sum_{\mu=1}^{M} \left(\underline{\varepsilon}^{(\mu)T}(k+1,n) R^{(\mu)-1}(k+1) \underline{\varepsilon}^{(\mu)}(k+1,n) \right)^{-1}}$.

(2.3) Update the covariances according to

$$P_{k+1}^{(\rho)-1}(n) = P_k^{(\rho)-1} + \{\hat{\alpha}^{(\rho)}(k+1,n)\}^2 H^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) H^{(\rho)}(k+1)$$

and

$$S_{k+1}^{(\rho)}(n) = H^{(\rho)}(k+1)P_k^{(\rho)}H^{(\rho)T}(k+1) + \{\hat{\alpha}^{(\rho)}(k+1,n)\}^{-2}R^{(\rho)}(k+1)$$

(2.4) Update the gain matrix according to

$$W_{k+1}^{(\rho)}(n) = P_k^{(\rho)}H^{(\rho)T}(k+1)S_{k+1}^{(\rho)^{-1}}(n)$$

(2.5) Update the state estimates according to

$$\underline{\hat{x}}_{k+1}^{(\rho)}(n) = \underline{\hat{x}}_k^{(\rho)} + W_{k+1}^{(\rho)}(n)\{\underline{z}(k+1) - H^{(\rho)}(k+1)\underline{\hat{x}}_k^{(\rho)}\}$$

(3) If "hard assignments" are required,

find $\nu = \arg\max_{\rho}\{\hat{\alpha}^{(\rho)}(k+1)\}$ and set $\hat{\alpha}^{(\rho)}(k+1) = \delta_{\rho\nu}$.

Note the time recursion can easily be modified to include a forgetting factor.

5. Kalman Predictor and Filtered Estimates

We present, without proof, a Kalman filter generalisation of the above recursion for the situation where the states are permitted to be time varying. Suppose, in addition to the measurement equation, we also have known dynamics for each of the states of the form

$$\underline{x}^{(\rho)}(k+1) = F^{(\rho)}(k)\underline{x}^{(\rho)}(k) + \underline{v}^{(\rho)}(k)$$

where

$$E\{\underline{v}^{(\rho)}(k)\underline{v}^{(\rho)T}(l)\} = Q^{(\rho)}(k)\delta_{kl}$$

The recursive solutions for the Kalman predictor, $\underline{\hat{x}}_{k+1/k}^{(\rho)}$, and the Kalman filtered output, $\underline{\hat{x}}_{k/k}^{(\rho)}$, are then given by

$$\underline{\hat{x}}_{k+1/k}^{(\rho)} = F^{(\rho)}(k)\underline{\hat{x}}_{k/k}^{(\rho)}$$

and

$$\underline{\hat{x}}_{k+1/k+1}^{(\rho)} = \underline{\hat{x}}_{k+1/k}^{(\rho)} + W_{k+1}^{(\rho)}\{\underline{z}(k+1) - H^{(\rho)}(k+1)\underline{\hat{x}}_{k+1/k}^{(\rho)}\}$$

where the Kalman gain, $W_{k+1}^{(\rho)}$, is given by

$$W_{k+1}^{(\rho)} = \{\alpha^{(\rho)}(k+1)\}^2 P_{k+1/k+1}^{(\rho)}H^{(\rho)T}(k+1)R^{(\rho)^{-1}}(k+1)$$

The recursion being finally completed by the the following update equations for the error covariances of the predicted and filtered states.

$$P_{k+1/k}^{(\rho)} = F^{(\rho)}(k) P_{k/k}^{(\rho)} F^{(\rho)T}(k) + Q^{(\rho)}(k)$$

and

$$P_{k+1/k+1}^{(\rho)\ -1} = P_{k+1/k}^{(\rho)\ -1} + \{\hat{\alpha}^{(\rho)}(k+1)\}^2 H^{(\rho)T}(k+1) R^{(\rho)-1}(k+1) H^{(\rho)}(k+1)$$

6. Summary

The problem of recursively estimating the parameters and assignments of a mixed process can be solved using probabilistic least squares techniques. Recursive least squares techniques or Kalman filters can be used depending on whether the states are assumed to be time varying or not. Combinatorial problems are avoided by using soft assignments.

7. References:

[1] R. Streit and T Luginbuhl "Probabilistic Multi-Hypothesis Tracking" *Technical Report* TR-10,428, NUWC Newport RI 1995.

[2] M.L. Krieg and D.A. Gray "Comparison of probabilistic least squares and probabilistic multi-hypothesis tracking algorithms for multi-sensor tracking" *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP-97)*, Vol. 1, Munich Germany, 515-518.

[3] M.L. Krieg and D.A. Gray "Comparison of pmh and pls trackers on real and simulated data" to appear *EURASIP Signal Processing.*

[4]. D. A. Gray, M.L. Krieg and M. R. Peake " Estimation of the Parameters of Mixed Processes by Least Squares Optimisation" *Proceedings of Fourth International Conference on Optimization; Techniques and Applications* (ICOTA 98) Vol 1, Perth Australia, July 1998, pp 891-898.

[5] Y. Bar-Shalom and T.E. Fortmann "Tracking and Data Association" Academic Press

STRATEGY FOR RADAR PULSE ALLOCATION APPLIED TO MULTITARGET TRACKING BY METAHEURISTICS

Workshop GT1-NUWC, ENST PARIS

Olivier GRONDIN Christian MUSSO

Office National d'Etudes et Recherches Aérospatiales (ONERA)
DTIM-MCT
29, avenue de la division Leclerc
92320 CHATILLON FRANCE

Email : grondin@onera.fr
Tel : 01.46.73.48.93

Email : musso@onera.fr
Tel : 01.46.73.49.33

ABSTRACT

This paper presents simulated annealing and genetic algorithms for solving a combinatorial optimization problem, applied to radar pulse management.
We have to manage a radar, working on a given area, during $[0,T]$. In this area, there are K detected targets. The goal is to find a radar pulse allocation strategy during the time $[0;T=N.\Delta t]$, in order to minimize a criterion taking into account the target state unaccuracy at the final time $T=N.\Delta t$, enhanced by the covariance matrices.
This is a complex optimization problem since it has an exponential complexity. Actually, nobody can exhib any algorithm able to find the optimal strategy, in a short computing time. Nevertheless, in a particular case with special added conditions, we have found the optimal solution by solving the well-known *Assignment Problem*. Dealing with the general case, we have implemented metaheuristics as simulated annealing and genetic algorithms, that search a strategy close to the optimal one in a short computing time.
An algorithm based on simulated annealing gives good solutions, for a small number of calculations; its performances are mainly due to a good choice of neighbourhood, described in this paper. Two genetic algorithms were also implemented but they did not give better solutions.

KEYWORDS

optimization, radar tracking, assignment problem, heuristic methods, genetic algorithm, simulated annealing

I. INTRODUCTION

In this paper, we consider the problem of optimal radar pulse allocation when the number of detected targets and the allocation time duration are fixed. The objective is to decide when a target has to be observed, in order to minimize a criterion taking into account the target state unaccuracy at the final time T of the allocation. We show that this problem is closed to a 0-1 integer programming problem, but the coefficients of the linear function to minimize, called cost function, are not explicit. Nevertheless, in the special case when each target is observed no more than once, we can exhib the cost function coefficients. We obtain a well-known integer programming formulation: the *Assignment Problem*. Dealing with the general case, we have implemented metaheuristics, as genetic algorithms and simulated annealing, that search a strategy close to the optimal one in a short computing time. The result comparison shows that our simulated-annealing based algorithm gives the best results.
The paper is organized as follows. In section II, the general allocation problem is formulated. In section III, an optimal solution is found in the special case when each target is observed no more than once, by resolving an *Assignment Problem*. Section IV and V deal with the general case. We search a solution close to the optimum with genetic algorithms, presented in section IV, and with a simulated annealing, presented in section V. In section VI, the comparison of our

experimental results shows the good behaviour of our simulated-annealing based algorithm. Finally, section VII summarises our results and discusses several ongoing research issues.

II. OPTIMAL RADAR PULSE ALLOCATION STRATEGY

A. problem formulation

We consider a radar, whose aim is to track K detected targets on a given area, during the time $[0,T]$. We assume that the target motion is uniform and that each track is initialized. The observations consist of the range r, the azimuth θ and the elevation ϕ. Moreover, the time between two measurements is fixed and equal to Δt. We also assume that during the time allocation $[0,T]$, the radar can perform N measurements and we do not take into account the delay due to the distance of the wave. The goal is to minimizea criterion taking into account the target state unaccuracy at the final time $T=N\ \Delta t$, i.e. after N measurements. The target state unaccuracy is underscored by the target covariance matrix. The allocation strategy is represented by a succession of N decisions. Each decision at time $t=i\Delta t$ ($i=1,2,\ldots,N$) is an integer value n in $[0,K]$ meaning that target n is observed at this time; $n=0$ means no measurement. Without loss of generality, we assume that $\Delta t=1$: a strategy is a sequence of $N=T$ decisions.

Because of the track initialization,we get two informations about targets: the first one, $X_{0,i}$, is the state vector of the target i, at time $t=0$. The second one, $P_{0,i}$, is the covariance matrix of the target i, at time $t=0$. More generally, the covariance matrix $P_{t,i}$ represents the variances and covariances of the target i state estimation error, at time t: position error and velocity error on the three axes x,y,z. $P_{t,i}$ is a (6,6) matrix.

The criterion to minimize, J, is the sum of the trace functional of the weighted target covariances:

$$J = \sum_{i=1}^{K} \mathrm{tr}\left(\alpha_i P_{T,i}\right) \tag{1}$$

In (1), α_i is a known positive or positive-semidefinite matrix of dimension (m,m) which weighs the relative importance of the various covariance matrix elements: for instance, it can emphasize a particular covariance matrix of a target that seems to be more threatening than the others.

Let us see now how we can calculate J, for a given strategy.

B. allocation strategy decoding

We assume that the target motion is uniform and that the acceleration is noised around zero. The state dynamic and measurement equations are assumed to be of the form:

$$X_{k+1,i} = FX_{k,i} + v_i \tag{2}$$

$$\omega_{k,i} = H\left(k, X_{k,i}\right) + w \tag{3}$$

where $X_{k,i}$ is the state vector of the target i at time t, v_i is a zero-mean white process noise with covariance matrix Q_i, w is a zero-mean white measurement noise with covariance matrix R, $\omega_{k,i}$ is the measurement vector and F is the system matrix.

We have:

$$X_{k,i} = \left[x_{k,i}\ \dot{x}_{k,i}\ y_{k,i}\ \dot{y}_{k,i}\ z_{k,i}\ \dot{z}_{k,i}\right]^T \tag{4}$$

$$F=\begin{bmatrix}1 & \Delta t & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & \Delta t & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & \Delta t\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix},\quad Q_i=\begin{bmatrix}\Gamma_{1i} & 0_2 & 0_2\\ 0_2 & \Gamma_{2i} & 0_2\\ 0_2 & 0_2 & \Gamma_{3i}\end{bmatrix}\quad \text{where}\quad \begin{cases}0_2=\begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}\\ \Gamma_{ji}=\begin{bmatrix}\Delta t^3/3 & \Delta t^2/2\\ \Delta t^2/2 & \Delta t\end{bmatrix}\sigma_{ji}^2\end{cases}$$

and σ_{ji}^2 is the variance of the component *j* of the acceleration noise vector on the target *i*.

$$\omega_{k,i}=\left[r_{k,i}\ \ \theta_{k,i}\ \ \varphi_{k,i}\right]^T$$

$$H\left(k,X_{k,i}\right)=\begin{bmatrix}\sqrt{X_{ki}(1)^2+X_{ki}(3)^2+X_{ki}(5)^2}\\ \arctan\left(X_{ki}(3)/X_{ki}(1)\right)\\ \arctan\left(X_{ki}(5)\Big/\sqrt{X_{ki}(1)^2+\left(X_{ki}(3)\right)^2}\right)\end{bmatrix}$$

$$R=\mathrm{diag}\left(\left[\sigma_r^2\ \ \sigma_\theta^2\ \ \sigma_\varphi^2\right]\right)$$

Remark: We assume that the measurement noises do not depend on the radar-target distance i.e. *R* does not depend on *i*. 'T' and 'E' denote transposition and expectation respectively.

It is well known that the Extended Kalman Filter (EKF) [1] provides a linear minimum mean-square error estimate of the state for the above dynamic system.
Defining $\hat{X}_{k|k-1,i}$, estimation of $X_{k,i}$ at time *k* given the set of measurements up to the included time *k*-1, by:

$$\hat{X}_{k|k-1,i}=\mathrm{E}\left(X_{k,i}\middle|\Omega_{k-1,i}\right)\ \text{where}\ \Omega_{k-1,i}=\left\{\omega_{j,i};\ j=1,2,\ldots,k-1\right\} \tag{5}$$

and the covariance matrix $P_{k|k-1,i}$, representing errors on the estimate $\hat{X}_{k|k-1,i}$ by:

$$P_{k|k-1,i}=\mathrm{E}\left(\left(X_{k,i}-\hat{X}_{k|k-1,i}\right)\left(X_{k,i}-\hat{X}_{k|k-1,i}\right)^T\middle|\Omega_{k-1,i}\right) \tag{6}$$

The covariance matrix $P_{k|k,i}$ (noted $P_{k,i}$ in the following) of the update state estimate at time *k*, given the set of measurements up to the included time *k* is given by:

$$P_{k,i}=P_{k|k,i}=P_{k|k-1,i}-P_{k|k-1,i}H_{k,i}^T S_{k,i}^{-1}H_{k,i}P_{k|k-1,i} \tag{7}$$

where:

$$P_{k|k-1,i}=FP_{k-1|k-1,i}F^T+Q_i=FP_{k-1,i}F^T+Q_i \tag{8}$$

$$H_{k,i}=\left[\nabla_X\left(H^T\left(k,X_{k,i}\right)\right)_{X_{k,i}=\hat{X}_{k|k-1,i}}\right]^T \tag{9}$$

$$S_{k,i} = H_{k,i} P_{k|k-1,i} H_{k,i}^T + R \tag{10}$$

From (7)-(10), $P_{k,i}$ is a function of $P_{k-1,i}$. If $P_{0,i}$ and $X_{0,i}$ are known for each target i, and if an allocation strategy s is given, we can calculate, thanks to equations (7)-(10), covariance matrices $P_{T,i}$ for each target. We can then evaluate $J(s) = \sum_{i=1}^{K} \mathrm{tr}(\alpha_i P_{T,i})$.

C. solution methodology

We want to find a strategy s^* such as $s^* = \underset{s \in S}{\mathrm{argmin}}(J(s))$ where S is the set of all feasible strategies - $|S| = (K+1)^T$ -.

Let $u_{k,i}$ be a bivalent integer variable such as :

$$u_{k,i} = \begin{cases} 1 \text{ if target } i \text{ is observed at time } k \\ 0 \text{ otherwise} \end{cases} \tag{11}$$

The radar cannot observe two targets at the same time. So, one constraint appears:

$$\sum_{i=1}^{K} u_{t,i} \leq 1 \quad \forall t \tag{12}$$

From (7)-(11) we can exhib $P_{k,i}$ function of $P_{k-1,i}$ and $u_{k,i}$:

$$P_{k,i} = P_{k|k-1,i} - P_{k|k-1,i} H_{k,i}^T S_{k,i}^{-1} H_{k,i} P_{k|k-1,i} u_{k,i} \tag{13}$$

From (13) we can develop J as:

$$J = \sum_{i=1}^{K} \mathrm{tr}\left(\alpha_i F^T P_{0,i} \left(F^T\right)^T\right) + \sum_{i=1}^{K}\sum_{t=1}^{T} \mathrm{tr}\left(\alpha_i F^{T-t} Q_i \left(F^{T-t}\right)^T\right) - \sum_{i=1}^{K}\sum_{t=1}^{T} A_{t,i} u_{t,i} \tag{14}$$

$$\text{where } A_{t,i} = \mathrm{tr}\left(\alpha_i F^{T-t} P_{t|t-1,i} H_{t,i}^T \left(H_{t,i} P_{t|t-1,i} H_{t,i}^T + R\right)^{-1} H_{t,i} P_{t|t-1,i} \left(F^{T-t}\right)^T\right) \tag{15}$$

We can see that $\sum_{i=1}^{K} \mathrm{tr}\left(\alpha_i F^T P_{0,i} \left(F^T\right)^T\right) + \sum_{i=1}^{K}\sum_{t=1}^{T} \mathrm{tr}\left(\alpha_i F^{T-t} Q_i \left(F^{T-t}\right)^T\right)$ is constant so it can be dropped from the minimization of J. Hence, to minimize J, the linear problem we have to solve is :

$$\max_{u_{t,i}} \bar{J} = \sum_{i=1}^{K}\sum_{t=1}^{T} A_{t,i} u_{t,i} \tag{16}$$

$$\sum_{i=1}^{K} u_{t,i} \leq 1 \quad \forall t \tag{17}$$

In order to find the optimal allocation strategy s^*, we have to search the sequence $\{u_{t,i} \text{ for } t = 1,...,T \text{ and } i = 1,...,K\}$ that maximizes $\bar{J}$. If we focus on the second index of the variables $u_{t,i}$=1, the allocation strategy, sequence of T decisions, appears. Unfortunately, we cannot exhib $A_{t,i}$ for all t and i since $A_{t,i}$ is a function of $P_{t-1,i}$, which is itself function of the

sequence $\{u_{t,i}=1 \ t=1,...T-1\}$. Nevertheless, for the special case when each target is observed no more than once, $P_{t-1,i}$ is a function of $P_{0,i}$ and we can explicit $A_{t,i}$ for all t and i.

III. OPTIMAL STRATEGY FOR A SPECIAL CASE

A. problem formulation

In this part, we assume that each target is observed no more than once, during $[0,T]$. This assumption is verified when the number of targets to track is bigger than the number of measurements the radar can perform and when the target noise processes are quite close; we choose here examples of scenarios where $K=2T$.
This assumption leads to a second constraint:

$$\sum_{t=1}^{T} u_{t,i} \leq 1 \ \forall i \tag{18}$$

Under this hypothesis, one measurement done for one target i at time τ implies that no measurement has been done during $[0,\tau-1]$ on this target. So we can explicit $P_{\tau|\tau-1,i}$ and $A_{\tau,i}$ is therfore known, for all τ value.

$$P_{\tau|\tau-1,i} = F^{\tau} P_{0,i} \left(F^{\tau}\right)^{T} + \sum_{m=1}^{\tau} F^{\tau-m} Q_i \left(F^{\tau-m}\right)^{T} \tag{19}$$

We can besides remark that $A_{t,i}>0$. In order to maximize (16), the constraint (17) becomes an equality constraint. The problem to solve is:

$$\begin{aligned} &\max_{u_{t,i}} \bar{J} = \sum_{i=1}^{K}\sum_{t=1}^{T} A_{t,i} u_{t,i} \\ &\sum_{i=1}^{K} u_{t,i} = 1 \ \forall t \quad \sum_{t=1}^{T} u_{t,i} \leq 1 \ \forall i \\ &u_{t,i} = 0 \text{ or } 1 \text{ for } i = 1,...,K \text{ and } t = 1,...,T \end{aligned} \tag{20}$$

where $A_{t,i}$ can be calculated from (15) and (19).

B. solution procedure

The problem in (20) is the well-known *Assignment Problem* extended to rectangular matrices (we remind that $T<K$). In 1973, Bourgeois and Lassalle extended Munkres'algorithm for the *Assignment Problem* to rectangular matrices (m,n) [2]; the complexity of this algorithm is $O(n^2m)$ when $m<n$. We have applied this algorithm to our problem in (20) and obtained the optimal solution of $\bar{J}$ and J. In the special case when T is small, we have noticed that the optimal strategy given by Bourgeois'algorithm was the same as the optimal one obtained by the all feasible strategy enumeration.
Dealing with the general case, when the number of measurements for a target is not fixed anymore, we are not able to solve $\bar{J}$. We have decided to develop heuristics , as genetic algorithms [3] and simulated annealing [4,5] to search a strategy close to the optimal one for a given calculation number.
The interest of this special case is that we are able to compare the heuristic method results with the optimal strategy.

IV. NEAR OPTIMAL STRATEGY BY GENETIC ALGORITHMS

A. generalities on genetic algorithms

Genetic Algorithms (GA) are intended to find a solution as close as possible to the global optimum of a given function, called objective function. One interest is that they often provide a solution close the optimum in a short computation time. Another interest is that they have given good results when applied to combinatorial optimization problems. We assume, without loss of generality, that we want to get a maximum of the objective function.
Let us introduce vocabulary proper to GA. A member of a population is a feasible solution of the objective function. Each member has several attributes, namely:

- genotype: solution encoding (solution x=> numeric string)
- phenotype: solution decoding (numeric string => solution x)
- fitness: it is a positive value given by an evaluation function, representing the member performance measure on the problem to solve. In many examples, the evaluation function chosen is the objective function.

A standard genetic algorithm, that normally yields good results in many practical problems, is composed of three operators, performed in the following order: Selection (or Reproduction), Crossover and Mutation.
Beginning with the population A of solutions and when these three operators are performed, the population B of new solutions is created. The operators can be applied to the population B, leading to the new population C and so on. The three-operator sequence is called a generation: a genetic algorithm is a sequence of a finite generation number.
How does a GA halt? Usually, the algorithm will stop when a fixed number of objective function calculations is reached. This number is equal to the number of generations multiplied by the number of members in a population.
We describe in the next section the different ways of choosing selection, crossover and mutation implementation in our algorithms.

B. description of the implemented genetic algorithms

First, our goal is to find the optimal strategy $\hat{S}$ that minimizes J. The objective function is J, a member of a population is an allocation strategy S. We want to minimize J; so, we cannot use the objective function as the evaluation function because strategies leading to the smallest values of J must have the biggest fitnesses. We consider the evaluation function F given by :

$$F(S) = \tilde{J} - J(S) \text{ where } \tilde{J} = \max_{S}\left(J(S)\right) \tag{21}$$

We can easily see that as $J(S)$ is positive for any strategy S, $F(S)$ is also positive for any S. J is maximum when no measurement is performed during $[0,T]$ so $\tilde{J}$ is given by $\tilde{J} = J(S_0)$, where S_0 is a sequence of T zeros. We can also verify that for a given strategy, the bigger the fitness will be, the smaller the J value will be. We also implement another evaluation function,the linear normalization: when the fitness is calculated as above for all the population members, we sort members by decreasing fitnesses. We create new fitnesses (by updating) that begin with a constant value and decreases linearly.
Two reproduction methods have been tested: selection by competition and roulette wheel selection. The first one consists in randomly choosing two members of the population A. The member with the greatest fitness value is copied in the population B with a given probability close to 1, otherwise, the other member is copied. The effect of the second selection, roulette wheel selection, is to return randomly a member of the population A. Although this selection procedure is non deterministic, each member's chance of being copied in B is directly proportional to its fitness.
Cross-over operator can be applied with a given probability i.e. in some case, children and parents will be the same. We have tested the two-point cross-over: when two members and

two crossing points in their genotype are chosen at random, it exchanges the integer string contained between the crossing points. New members are created in this way and called children.
Our mutation operator, applied with a slight probability, selects at random one decision in a given strategy and changes the decision value.
The population size is set to 30 members and 20 generations can occur. GA halts when 600 objective function values are computed.
In order to improve standard GA results, we implement the elitism module in the selection operator. This module simply carries forward the best member from the previous generation into the next.
Several GA are built from the combination of these operators. We present in section VI, two GA that give the best experimental results. More details about GA can be found in [3].

V. NEAR OPTIMAL STRATEGY BY SIMULATED ANNEALING

A. simulated annealing principles

Simulated Annealing (SA) is an algorithm developed over the past 20 years to tackle hard combinatorial optimization problems. It is one of the most popular because of its ease of use and of its asymptotic convergence to optimal solutions. It is extensively described in [5] but we remind some foundations about it.
SA is based on an analogy with statistical mechanics: a slow physical system cooling is simulated in order to attain minimumenergy states. We apply small disturbances on a current solution, and whereas the decreases of the objective function F are always accepted, we allow it to increase according to a probability. This probability, controlled by the « Temperature » parameter, allows the system to climb out of local minima. Applyingthis process successively, called « Metropolis procedure » or « stochastic relaxation », for every temperature in a sequence beginning with a great positive value and decreasing to zero, we simulate a slow cooling called « annealing ». The transition from one temperature to another is done when the system is « frozen » i.e. when the system has reached an « equilibrium at temperature θ ».
In order to implement the algorithm, we have to clarify some points: what is a « frozen system » and how to build the temperature sequence? Answers depend on the problem we have to solve and are given in the next paragraph where we describe our SA based algorithm.

B. description of the implemented simulated annealing algorithm

As seen before, a feasible solution of our problem i.e. a strategy, is an integer string of T decisions; one decision represents one element of the integer string. We can calculate for a given solution S, the criterion $J(S)$ and we have to find S' such as $J(S')$ is as small as possible. We present now how we have adapted SA to our problem.
Our system is « frozen at the temperature θ » if after disturbing N_n successive current solutions S_i, the resulting optimum strategy S_{min} has not changed. We will then diminish the temperature θ.
A current solution S_c is disturbed by choosing a solution S_d in the current solution neighbourhood. It is on your own to define effective neighbourhoods. A well-known one usually applied to combinatorial problems is the « *k-change* » neighbourhood procedure: given a solution S (sequence of integers for example), a solution neighbouring of S is obtained by replacing k elements of S. In our case, for a given strategy, applying a « *k-change* » procedure amounts to replacing k decisions in this strategy. We have tested a lot of *k-change* procedures for different k values and different ways of replacing such k decisions (random choices, permutations...). The procedure that give the best results is based on a *3-change* and called neighbourood N_3. It is built from the two-process succession: the first one is a permutation between two decisions chosen by drawing lots, the other is a substitution of a decision chosen at random by any integer value randomly chosen in the set of the feasible decisions.

The temperature sequence is such as : $\theta_n = r^n\theta_0$, where the cooling ratio r is contained in $]0,1[$.
This sequence is defined by an initial temperature $\theta_0>0$, a cooling ratio r and a final temperature θ_{fin}.
The objective function calculation number depends on the parameters, N_n, r, θ_0, θ_{fin}. In order to be fair to GA, we have chosen parameters such as the criterion calculation number is lower or equal to 600. Let us precise that for each run of a GA, the number of calculations is equal to 600 because the algorithm halts when this number is reached. In the case of SA, the criterion calculation number is not deterministic because it depends on the system equilibrium which depends on the neighbouring solutions. For SA, we consider the average number of criterion calculations on several runs.
One effective parameter set for our problem is $N_n = 5$; $r = 0,9$; $\theta_0 = 900$ and $\theta^*_{fin} = 0.1$ and leads to an average number of criterion calculations equal to 500 (average calculated on 20 runs of our algorithm).
All the elements are available to implement algorithms based on genetic algorithm and simulated annealing. In the next section, we present results we have obtained for two different scenarios.

VI. EXPERIMENTAL RESULTS

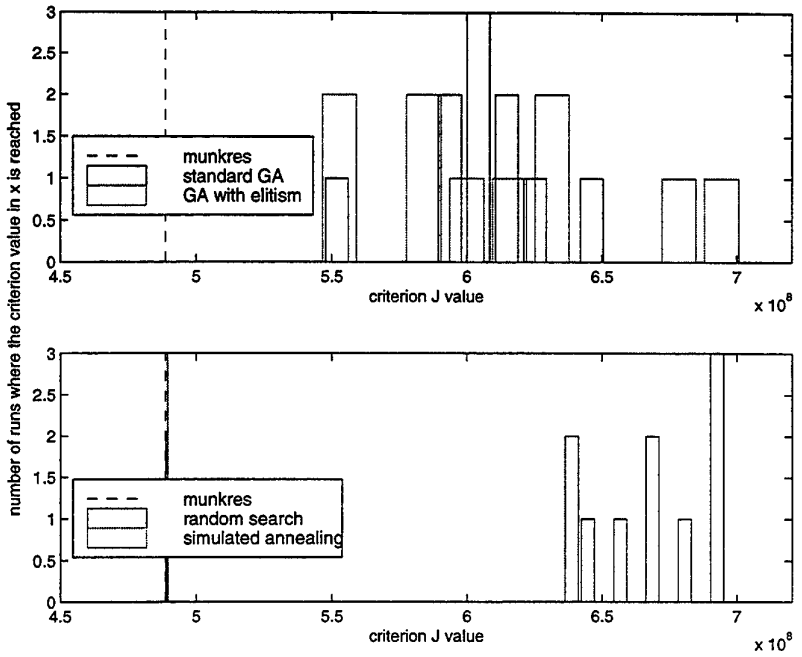
A. algorithms and results

We have tested four heuristic algorithms:
- standard GA: built as presented in section IV with the selection-by-competition module. Parameters are fixed by:
  crossover probability=0.8 and mutation probability=0.01
- GA with elitism: built as the standard GA but with the roulette-wheel-selection module and the elitism module where the two best solutions are kept without change from one population to another.
- Random Search: 600 allocation strategies are chosen at random. The algorithm gives in output the best one.
- Simulated Annealing: built as explained in section V, with neighbourhood N_3.

The results of these algorithms are compared to the optimal strategy given by the Bourgeois'algorithm, based on Munkres'algorithm, when it is possible. Random events occur in our heuristic algorithms. In order to test their robustness, we present results from 10 runs, as histograms. One histogram displays results given by 10 runs of one algorithm. One stick appears on the criterion value; its height shows how many times this value is reached. The more narrow the histogram will be, the more robust the corresponding algorithm will be.
Target state initialization follows the same model for all the presented scenarios. The target motion is uniform. Initial target positions x and y are chosen at random between -425 km and 425 km while z is chosen at random between 1 km and 10 km. The target velocities on x, y, z are initialized by choosing a random value between -800 m/s and 800 m/s. The standard deviations are chosen between 500 m and 6500 m for the target position and between 10 m/s and 100 m/s for the target velocity.
The measurement standard deviations are fixed to 1km for the range r and 2° for the azimuth θ and the elevation ϕ.

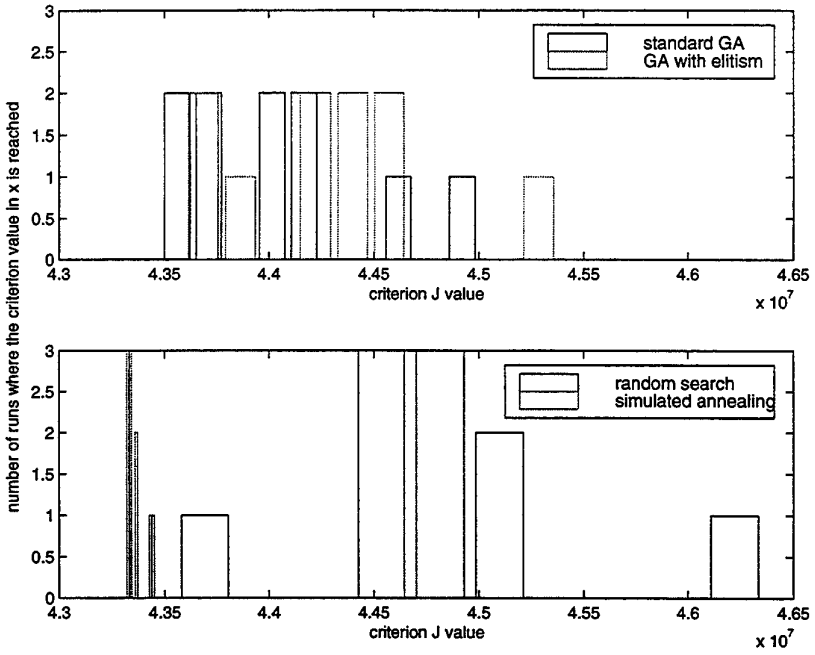
B. assumption « no more than one measurement on each target » is verified

We have to allocate 10 mesurements on 20 targets. It is a case where each target is observed either once or never. We cannote a very good behaviour of our SA based algorithm. The 10 runs give results very close to the optimal strategy. We can also remark that the random search gives the worst results and that those obtained by our genetic algorithms are quite disappointing.



C. assumption « no more than one measurement on each target » is no longer verified

We have to allocate 10 measurements on 8 targets. It is a case where a target can be observed more than once. We cannot have the optimal strategy and once again the best strategies are given by our simulated annealing algorithm. Genetic algorithms do not improve results and are less robust than simulated annealing. The worst strategy is obtained by a random search.



VII. CONCLUSION

Optimization of radar pulse allocation raises many difficulties: solving combinatorial optimization problems, searching an acceptable solution in a short computing time, and taking into account constraints due to the radar features. Nowadays, it is still impossible to find, in a short computing time, the optimum solution of such problems satisfying all these constraints. However, this allocation problem is very important to provide an accurate information on the targets, particularly in the military context. We have hence decided to search a solution close to the optimum via metaheuristic methods.

Heuristic method implementation is quite simple but difficulties appear when you have to choose parameter values. Bad GA results - slightly better than random search results - prove the difficulty to adapt our problem to GA. Moreover, how can we validate heuristics since we do not know the optimum? Fortunately, we have found in a special case the optimal allocation strategy and we have been able to compare it with the heuristic results and to compute gaps between the heuristic results and the optimum. These results show that for a given computing time, our SA based algorithm gives the best strategies and is very robust in comparison with genetic algorithms and random search. This is mainly due to the neighbourhood N_3, specially well adapted for our problem.

There are numerous extensions of this research. For instance, an optimal strategy search, taking into account radar features should reveal a real interest for business companies working on radar applications. Other metaheuristics, as tabu search, can be implementedand compared to SA.

Finally, we emphasize that heuristic methods can be applied to many other problems and that results depend on the quality of the method adaptation.

REFERENCES

[1] Y.BAR-SHALOM - T.E.FORTMANN- "Tracking and Data Association" - Mathematics in Science and Engineering - Volume 179 - 1988

[2] F.BOURGEOIS - J.C. LASSALLE - "An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices" - Communications of the ACM - Volume 14 n°12 - Décembre 1971

[3] D.GOLDBERG - "Genetic Algorithms in Search, Optimization and Machine Learning" - Editions ADISON WESLEY - 1989

[4] S.KIRKPATRICK C.GELATT- M.VECCHI- "Optimization by Simulated Annealing" - Science - Volume 220 n°4598 - May 1983 - pp.671-680

[5] E.H.L. AARTS - P.J.M. LAARHOVEN - "Simulated Annealing: Theory and Applications" - Kluwer Academic - 1987

Frequency Line Tracker in passive sonar system

S. Paris, C. Jauffret and G. Goullet
MS-GESSY, ISITV, Av G. Pompidou
BP 56, 83162 La Valette du Var Cedex, France
Tel: +33 04 94 14 24 58
e-mail:
(paris,jauffret)@isitv.univ-tln.fr

October 20, 1998

Abstract

This paper deals with the problem of Frequency Line Tracking (FLT) and more generally the problem of estimating narrow-band signal parameters in time-varying observation, encountered for example in passive sonar systems or speech recognition problems. We propose a new measurement model usable in frequency line tracker algorithms, based upon Maximum Likelihood estimator (ML) of sinusoids in noisy environment. The main purpose of this paper is to establish performance of frequency line tracker algorithms by Monte-Carlo simulation according to Hidden Markov Models (HMM) and our new approach.

We study the case of discontinuous frequency lines, by introducting a state detector and evaluate its performances.

1 Introduction

Detection tracking (frequency, bearing, ...) in passive sonar system gives the input of the so-called Target Motion Analysis [5]. For example, a lofargram is a time-frequency representation obtained by splitting the time signal into a subset of intervals and computing a so-called periodogram on each of them (a zero-padding technique can be used to improve the spectral analysis). Several problems are encountered in frequency line tracking : the first one is to determine the number of tracks (or lines) present in the lofargram, the second is the birth and death of track, encountered when the sonar goes in and out a convergence zone for a given target [3], and the third problem is the crossing track case. HMM frequency line tracker presented in [1, 12, 14] assumes to keep only the more energetic FFT cell corresponding to either a true detection or a false alarm. This approach needs the definition of two matrices (A, B) intervening in HMM $F \triangleq (\pi, A, B)$ [9]. All the terms of matrix B can be *a priori* computed, which is possible when, only one detection is retained. In case of several detections, such a *a priori* calculation cannot be planned because of the computation and the memory burden : only coefficients needed are evaluated on-line. A direct algorithm from HMM is the Forward-Backward procedure introduced first in FLT case in [12].

FLT problem has also been solved in terms of Dynamic Programming technique in [6, 10, 13]. In [6], several detections are retained after thresholding the periodogram of a single sinusoïd in the presence of additive noise; hence, there are at most one true detection and some false alarms. The Probabilistic Data Association (PDA) modeling coupled with a Viterbi algorithm has been employed. A third exploitation of the lofargram have been used by [10, 11, 13]. The Maximum *a posteriori* (MAP) estimator of a frequency state sequence is showed to be the argument of the periodogram under Gaussian hypotheses. Once again the Viterbi algorithm is used.

This paper is outlined as follows : Section 2 presents our frame work and the notation we will use. The next section describes the three algorithms for frequency line tracking and relation between HMM

and dynamic programming point of view for continuous frequency line. Discontinuous track case is presented in section 5.

2 Assumptions and Notations

We suppose first, we are facing the problem of tracking a frequency story $\{f(t), t \in \tau\}$ when the input signal is usually modeled by :

$$s(t) \triangleq a \sin[2\pi\Phi(t)] + \eta(t) \tag{1}$$

where $\eta(t)$ is a zero-mean Gaussian noise. The instantaneous frequency $f(t) \triangleq \frac{d\Phi(t)}{dt}$ is assumed to vary slowly with respect of the sampled frequency f_e. The total duration τ of the observed signal is split into K intervals of n samples. $\tau = [T_0, T_1[\cup [T_1, T_2[\cup ... \cup [T_{K-1}, T_K[$, each of them being of an equal length of $T \triangleq T_k - T_{k-1}, \forall k$. On each interval, n samples of $s(t)$ is sampled at $f_e \triangleq \frac{1}{\Delta t}$ (n is supposed even, $n = 2M$) with $n \triangleq f_e(T_k - T_{k-1})$. In the literature, $f(t)$ is assumed constant on each interval $[T_{k-1}, T_k[$, i.e. :

$\exists\, r_k \in \{0, ..., M-1\}$ such that : $f(t) = f_k \triangleq \frac{r_k}{T}$. So, we define :

$$s_k(i\Delta t) \triangleq s(i\Delta t + T_{k-1}) = a \sin(2\pi \frac{r_k}{n} i + \phi_k) + \eta(i\Delta t), \tag{2}$$

where ϕ_k is uniformly distributed in $[-\pi, \pi[$, and $\eta(i\Delta t)$ is a zero-mean Gaussian noise with standard deviation σ_η.

We compute the periodogram by :

$$P(k,l) \triangleq \left| \frac{1}{n} \sum_{i=0}^{n-1} s_k(i\Delta t) e^{-2j\pi \frac{li}{n}} \right|^2 \tag{3}$$

for $k = 1, ..., K$ and $l = 0, ..., M-1$. We recall the statistic of the periodogram, if $l \neq r_k$:

- $$P(k,l) \sim \chi^2_2 \tag{4}$$

  and :

- $$P(k, r_k) \sim \chi^2_2(\kappa = \frac{a^2}{2\sigma^2_\eta} n) \tag{5}$$

  where κ represents the SNR level after the "gain of FFT" (the factor n).
  There are three ways for exploiting a lofargram.

1. thresholded and binarised lofargram : Let's denote $m_k \leqslant M$ the number of detection retained after thesholding at time k, $Z_k \triangleq (z_1, ..., z_k)$ the measurement matrix from time 1 to time k with $z_k \triangleq (z_{k,1}, ..., z_{k,m_k})$, the measurement vector at time k, $z_{k,l} \in \{0, ..., M-1\}$.

2. thresholded lofargram : The number of detections at time k is equal to $m_k \leqslant M$, $Z_k \triangleq (z_1, ..., z_k)$ the measurement matrix from time 1 to time k with $z_k \triangleq (z_{k,1}, ..., z_{k,m_k})$, the measurement vector at time k and $z_{k,l} \triangleq (q_{k,l}, e_{k,l}), l = 1, ..., m_k$ a detection centered in the frequency cell $q_{k,l} \in \{0, ..., M-1\}$ with a "energy" equal to $P(k, q_{k,l})$.

3. non-thresholded lofargram : M is the total number of cells for each instant k. $Z_k \triangleq (z_1, ..., z_k)$ the measurement matrix since instant k with $z_k \triangleq (z_{k,1}, ..., z_{k,M})$, the measurement vector at time k and $z_{k,l} \triangleq (l, P(k,l)), l = 0, ..., M-1$.

3 Theoretical approach

3.1 Frequency line tracker algorithms

Three algorithms are generally used in frequency line tracking. Each of them assumes that the state sequence is modelised by a first order Markov chain also called a random-walk and defined by : $x_k \triangleq x_{k-1} + \epsilon_k$, with ϵ_k a zero-mean white noise with standard deviation d. The three algorithms compute *a posteriori* probability with the knowledging of $\Pr(x_k = \xi_k | x_{k-1} = \xi_{k-1})$ and $\Pr(z_k | x_k = \xi_k)$, two probabilities which are the common entries for the three procedures. For each approaches, frequency line tracking consists to give an estimation of the state sequence, $\widehat{X}_k \triangleq (\widehat{x}_1, ..., \widehat{x}_k)$ with $\widehat{r}_k \triangleq \widehat{x}_k$. Algorithms are written as follows :

1. Viterbi (VIT) algorithm : by computing the Maximum A Posteriori (MAP) likelihood track (see [6, 8, 10, 11, 13]) :

$$\Pr(X_k = \Xi_k | Z_k) = \frac{\Pr(z_k | x_k = \xi_k) \Pr(x_k = \xi_k | x_{k-1} = \xi_{k-1}) \Pr(X_{k-1} = \Xi_{k-1} | Z_{k-1})}{\Pr(z_k | Z_{k-1})}, \quad (6)$$

where $\Xi_k \triangleq (\xi_1, ..., \xi_k)$ represents a possible realization of X_k from time 1 to time k, $\xi_k \in \{0, ..., M-1\}$. The estimated track is defined by :
$\widehat{X}_{K|K} \triangleq (\widehat{x}_{1|K}, ..., \widehat{x}_{K|K}) \triangleq \arg \max_{X_K} \{\Pr(X_K | Z_K)\}$ (MAP).

2. Forward-Backward (F-B) algorithm by computing (see [1, 12, 14]) :

$$\Pr(x_k = \xi_k | Z_K) = \frac{\Pr(x_k = \xi_k, Z_k) \Pr(z_{k+1}, ..., z_K | x_k = \xi_k)}{\sum_{i=0}^{M-1} \Pr(x_k = i, Z_k) \Pr(z_{k+1}, ..., z_K | x_k = i)} \quad (7)$$

with the recursive expression :

$$\Pr(x_k = \xi_k, Z_k) = \Pr(z_k | x_k = \xi_k) \sum_{\xi_{k-1}=0}^{M-1} \Pr(x_{k-1} = \xi_{k-1}, Z_{k-1}) \Pr(x_k = \xi_k | x_{k-1} = \xi_{k-1})$$

and

$$\Pr(z_{k+1}, ..., z_K | x_k = \xi_k) = \sum_{\xi_{k+1}=0}^{M-1} \Pr(z_{k+1} | x_{k+1} = \xi_{k+1}) \Pr(x_{k+1} = \xi_{k+1} | x_k = \xi_k) \Pr(z_{k+2}, ..., z_K | x_k = \xi_k).$$

This is a local state estimator at time k with knowledge of all the available measurements. The estimated track is defined by :
$\widehat{X}_{k|K} \triangleq (\widehat{x}_{1|K}, ..., \widehat{x}_{k|K}), k = 1, ..., K$ and $\widehat{x}_{k|K} \triangleq E[x_k | Z_K]$ (MMSE).

3. Bayes-Markov (B-M) algorithm by computing (see [2, 4]) :

$$\Pr(x_k = \xi_k | Z_k) = \Pr(z_k | x_k = \xi_k) \Pr(x_k = \xi_k | Z_{k-1}) \quad (8)$$

with the recursive expression:

$$\Pr(x_k = \xi_k | Z_{k-1}) = \sum_{\xi_{k-1}=0}^{M-1} \Pr(x_{k-1} = \xi_{k-1}, Z_{k-1}) \Pr(x_k = \xi_k | x_{k-1} = \xi_{k-1}).$$

This approach is the forward step of the forward-backward procedure. The estimated track is defined by :
$\widehat{X}_{k|k} \triangleq (\widehat{x}_{1|k}, ..., \widehat{x}_{k|k}), k = 1, ..., K$ and $\widehat{x}_{k|k} \triangleq E[x_k | Z_k]$ (MMSE). It is compatible with a real time implementation. We precise in the next section our choice about probabilities used.

Remark 1 *All the probabilities will be approximated by their density, i.e.* $\Pr(\xi) \backsimeq (\frac{d\Pr}{d\xi})\Delta f$. *In the sequel, term* Δf *is dropped.*

Remark 2 *The Viterbi and Forward-Backward algorithms are batch procedure meanwhile the Bayes-Markov is a real-time approach.*

3.2 Modelization of frequency line tracker entries

Every frequency line tracker needs the computation of two probabilities. The first one $\Pr(x_k = \xi_k | x_{k-1} = \xi_{k-1})$, defines the probability of state transition between time $k-1$ and k. We define matrix A as follows $A \triangleq (a_{ij})_{i,j=1...M}$ with :

$$a_{ij} \triangleq \Pr(x_k = i-1 | x_{k-1} = j-1) = \frac{1}{\sqrt{2\pi}d} e^{\frac{-(i-j)^2}{2d^2}}. \tag{9}$$

Note that this matrix can be computed *a priori.*

The second entry defines the likelihood of a line of the lofargram, i.e. $\Pr(z_k | x_k = \xi_k)$. As mentionned in section 1, this probability depends on the choice of the lofargram exploitation :

a) thresholded and binarised lofargram :

The number (m_k) of retained measurements (after tresholding) is a random variable. Unlike the case treated in [1, 12, 14] where $m_k = 1$ and for which all the probabilities $\Pr(z_k = i | x_k = j)$ can be *a priori* computed and put in matrix $B = \Pr(z_k = i-1 | x_k = j-1)_{\substack{i=1,...,M \\ j=1,...,M}}$, we have to evaluate similar probabilities for a potentially tremendous number of cases. It is no way to compute *a priori* such a set of probabilities, unless they are the same. This is the way used by Bethel : in [2]; he proposes

$$\Pr(z_k | x_k = i) \propto \left\{ \begin{array}{c} \frac{Pd}{Pfa} \text{ if } P(k,i) > t, \text{ the detection thresholds} \\ \frac{1-Pd}{1-Pfa} \text{ if not.} \end{array} \right\} \text{(BET)}$$

Jauffret adopts the Probabilistic Data Association Model [6] and gets :

$$\Pr(z_k | x_k = i) \propto 1 - Pd + \frac{Pd}{\lambda} \sum_{j=1}^{m_k} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z_{k,j}-i)^2}{2\sigma^2}} \text{(PDA)} \tag{10}$$

where λ denotes the mean number of detections by volume unit, Pd the probability of detection and σ the standard deviation for detection. This last two models can be interpreted as a Track After Detect (TAD) approaches.

b) non-thresholded lofargram :

We present a new model based upon the ML estimator of a pure sinusoid in white Gaussian noise. Three assumptions are required for this new model :

1. Measurements are supposed conditionally independent in time.
2. The ML frequency estimator for $s(t)$ at each interval $[T_{k-1}\ T_k]$ is given by $\arg\max\{P(k,l)\}$ and its probability is proportionnal to $e^{\frac{MP(k,l)}{\sigma_\eta^2}}$.
3. The correct detections are Gaussian with a standard deviation equal to σ. Karan showed that, in case of linear FM signal, the distribution of $\widehat{f}_{ML}$ is symmetrical. Here, we choose the Gaussian approximation for convenience (see [7]).

We define at time k $H_l, l = 1, ..., M$ as " the frequency lines is in cell l at time k", i.e. $r_k = l$. Then $\Pr(z_k | x_k = \xi_k)$ can be written as follows :

$$\Pr(z_k | x_k = \xi_k) = c \sum_{l=1}^{M} \Pr(z_k | x_k = \xi_k, H_l) \Pr(H_l | x_k = \xi_k)$$

$$\Pr(z_k|x_k = \xi_k) \triangleq c \sum_{l=1}^{M} e^{\frac{MP(k,l)}{\sigma_\eta^2}} e^{-\frac{(\xi_k - l)^2}{2\sigma^2}} \text{ (TBD)} \tag{11}$$

where c denotes the *ad-hoc* normalization factor. As no thresholding stage is performed, this approach can be viewed as a Track Before Detect (TBD) one.

Remark 3 *In the sequel, we will denote for example by : VIT/TBD/MAP the combinaison between frequency line tracker algorithms (Viterbi : VIT, Forward-Backward F-B or Bayes-Markov : B-M), model (Track Before Detect : TBD, Probabilistic Data Association : PDA and Bethel model :BET) and the estimator used (MAP or MMSE).*

4 Continuous track problem

According to HMM convention, models needed for each algorithms, can be written in terms of matrix A and B and vector π (the initialization probability is chosen uniform on the state space) : $F \triangleq (\pi, A, B)$. In fact matrix B, where $m_k > 1$, is computed after receiving each measurement vector, but we still use HMM to describe input of frequency line trackers. Our SNR definition is written as follows :

$$SNR_{dB} \triangleq 10\log_{10}(\frac{a_k^2}{\sigma_\eta^2}) + 10\log_{10}(M) \tag{12}$$

The chosen criterium to establish performance of frequency line tacker is the computation of RMSE :

$$\sigma_{RMSE} \triangleq \frac{1}{N} \sum_{j=1}^{N} \left[\sqrt{\frac{1}{K} \sum_{k=1}^{K} (\widehat{x}_{k,j} - x_k)^2} \right] \tag{13}$$

where $\widehat{x}_{k,j}$ denotes the estimated state at time k of the j^{th} runs and x_k denotes the true state at time k, N being the number of Monte-Carlo runs.

In conclusion, HMM frequency line tracker algorithms can be used even with multiples detections per scan with a new formulation of the HMM in term of B matrix. In this case, we got the same entries formulation with dynamic programming point of view. The Viterbi algorithm perform the complete state sequence optimization and the Forward-Backward compute each individual state of the track.

4.1 Monte-Carlo simulations on continuous extracted tracks

We perform over different SNR's, $N = 1500$ frequency line tracking runs on a synthetic signal, with the three algorithms (6,7,8). Figure (1) shows an realization example of a lofargram where $M = 256$ and $K = 100$.

Figures (2) and (3) show entries of frequency line tracker with TBD (a) or TAD (b) approaches and the results of two frequency line tracking associated, i.e. VIT/TBD/MAP (a) and VIT/PDA/MAP (b). The SNR of input signal was 5 dB (a 512-point FFT) with a sampling frequency f_e equal to $2000Hz$. Matrix A is computed according to (9) with $d = \frac{6}{\sqrt{12}}$. For TBD approach, σ was chosen equal to $\frac{4}{\sqrt{12}}$. The TAD approach with the PDA modeling was initialized with following parameters : The threshold level is chosen to maximize the ratio $\frac{P_d}{P_{fa}}$ of χ^2 ROC curves and keeping Pd and $\lambda \propto Pfa$ associated with this threshold and $\sigma = \frac{4}{\sqrt{12}}$. The two extracted tracks in figure (3) with the same frequency line tracking algorithms are both of them pretty good, even if with the TBD modelization seems to be more efficient. Figure (4) show the computation of the RMSE for different SNR (see 13) and reveals two important points :

1. With model given (PDA, BET or TBD) Viterbi algorithms or F-B algorithms give one similar performances.

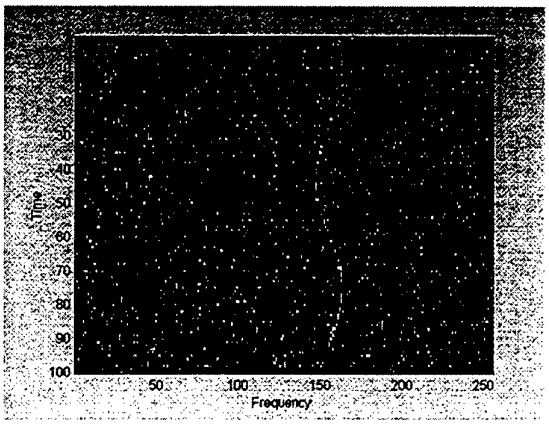


Figure 1: Example of synthetic lofargram with a 8 dB track

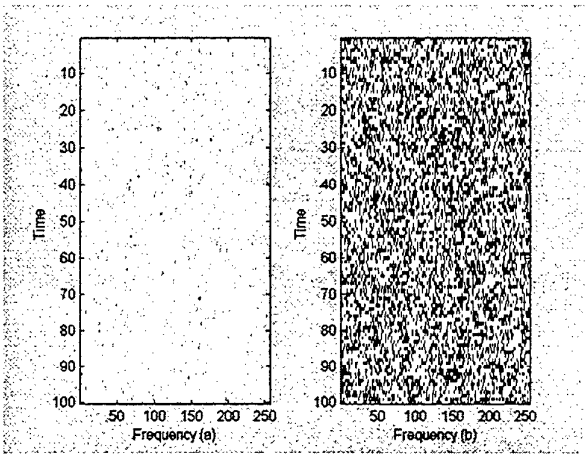


Figure 2: a) Input with TBD approach b) Input with TAD approach

2. With algorithm given (Viterbi, Bayes-Markov or Forward-Backward) TBD performs better than other.

5 Track validation

One of frequency line tracker assumption presented in section 3, is the existence of the track during the whole measurement recording. In fact, for different reasons, this is cannot be physically verified [3]. This problem, called in the sequel "discontinuous track problem", needs a state detector. We present here two tests based upon the Forward-Backward frequency line estimator.

5.1 A Neyman-Pearson state detector

First approach is to perform the likelihood ratio between signal-plus-noise hypothesis and noise-only hypothesis following Neyman-Pearson's lemma. Let's introduce a new state sequence $H_k \triangleq (h_1, ..., h_k)$ where each individual state h_k can take only two values, i.e. 0 or 1. We denote also Υ_k a complete realization of this new state sequence H_k.

We define the noise-only likelihood by : $\Pr(\widehat{x}_k | h_k = 0) \triangleq \frac{1}{M'}$. In that case, every frequency cell is

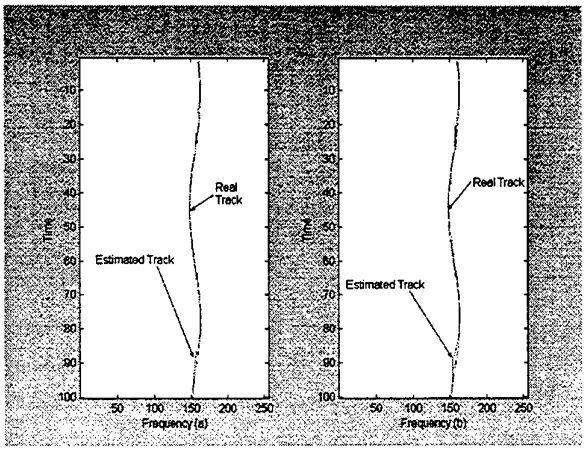


Figure 3: a) True & Extracted track with TBD approach b)True & Extracted track with TAD approach

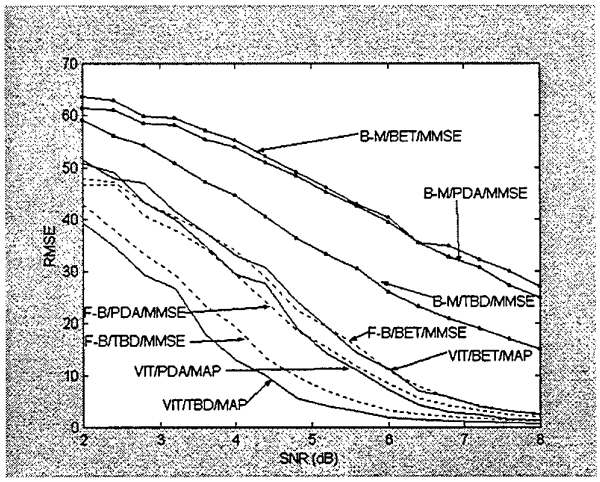


Figure 4: RMSE (number of cells) versus SNR for 9 FLT

a possible recipient of the estimated state in a bandwidth of M' cells. Parameter M' is proportional to the standard deviation d of the x_k state transition probability. In other hand, we also define the signal-plus-noise likelihood by : $\Pr(\widehat{x}_k|h_k = 1) \triangleq \Pr(x_k = \widehat{x}_k|Z_K)$, with the Forward-Backward *a posteriori* probability for the estimated state $\widehat{x}_k$. The Neyman-Pearson test (denote NPT) is equivalent to $M' \Pr(\widehat{x}_k|h_k = 1) \lessgtr t$, with t a *a priori* given threshold. Figure (5) shows two mean realizations and the $\pm$ two time the standard deviation intervals of confidence (over 1500 runs) of $\Pr(\widehat{x}_k|h_k = 1)$, for two discontinuous track (a 4dB and 15 dB frequency line) being present during time $[30 - 70]$.

The main problem of such detector is the "hole effect" for poor SNR, because the state detection is computed individually step by step. A second problem is the choice of the threshold t.

5.2 A Dynamic state detector

According with the dynamic programming approach for frequency line estimator, we propose a new state detector based upon a Forward-Backward algorithm. This state detector (denote FBT) is modeled by $F^d \triangleq (\pi^d, A^d, B^d)$ which are the common entries of the new state detector.

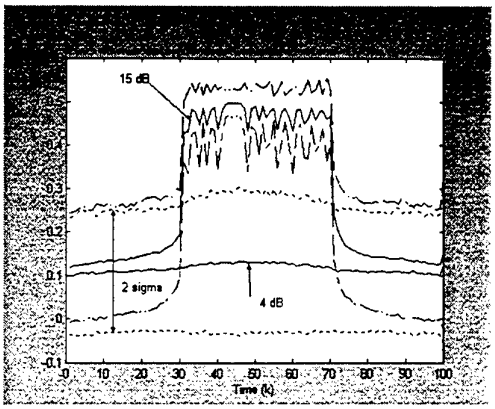


Figure 5: 2 means realizations of the signal-plus-noise likelihood and their intervals of confidence

For this goal, we must define the entries of this new state detector in terms of matrices A^d and B^d. Matrix A^d represents the state transition matrix (2×2) from time $k-1$ to time k and defined by : $A^d \triangleq (a_{ij}^d)_{i,j=0,1} = \Pr(h_k = \nu_i | h_{k-1} = \nu_j) = \begin{bmatrix} 1-q & q \\ q & 1-q \end{bmatrix}$ where q can be assimilated to a standard deviation of a noise process. The more q factor is important, the more sensible to the noise measurement the detector is. Matrix B^d $(K \times 2)$ must also be defined in terms of $b_{ij}^d \triangleq \Pr(\widehat{x}_k = i | h_k = \nu_j)$ where $\Pr(\widehat{x}_k | h_k = \nu_j)$ is the likelihood of the new measurement entry. Matrix B^d entries is modeled as follows :

$$\left\{ \begin{array}{c} \Pr(\widehat{x}_k | h_k = 1) \triangleq \Pr(x_k = \widehat{x}_k | Z_K) \\ \Pr(\widehat{x}_k | h_k = 0) \triangleq \frac{1}{M'} \end{array} \right\} \tag{14}$$

Then, we propose to compute recursively $\Pr(h_k = \nu_i | \widehat{X}_K)$ by using the Forward-Backward procedure :

$$\Pr(h_k = \nu_i | \widehat{X}_K) = \frac{\Pr(h_k = \nu_i, \widehat{X}_k) \Pr(\widehat{x}_{k+1}, ..., \widehat{x}_K | h_k = \nu_i)}{\sum_{i=0}^{1} \Pr(h_k = i, \widehat{X}_k) \Pr(\widehat{x}_{k+1}, ..., \widehat{x}_K | h_k = i)}. \tag{15}$$

Remark 4 *It is possible to use also a Viterbi Algorithm or a Bayes-Markov algorithms for the construction of such a test.*

Remark 5 *The use of Forward-Backward procedure is a little bit abusive : the "measurement" $\widehat{x}_k$ are not conditionnely independant.*

5.3 Monte-Carlo simulations on discontinuous extracted tracks

We are facing problem of extraction of discontinuous track as showed in figure (6), where for example, a 10 dB frequency line is present during instant $[30-70]$.

For the state detector performance analysis, we performs Monte-Carlo simulation with a discontinuous track. We compute $N = 1500$ runs for each SNR with a frequency line being present during time $[30-70]$. The performance analysis is done by computing RMSE of beginning of the track. The beginning of the track is the first state estimate in the H_k sequence higher than the given threshold.

For the FBT detector, the constant q is equal to $10e-5$, the threshold $t = 0.999$, $M' = 10$ and $\pi^d = [1 - 10e^{-5}, 10e^{-5}]$. These values has been chosen empirically. We denote the couple frequency line estimator - state detector by : F-B/TBD/MMSE-FBT where F-B/TBD/MMSE represents the Forward-Backward algorithm with TBD model for frequency line estimator and FBT, the new state detector based upon a Forward-Backward procedure.

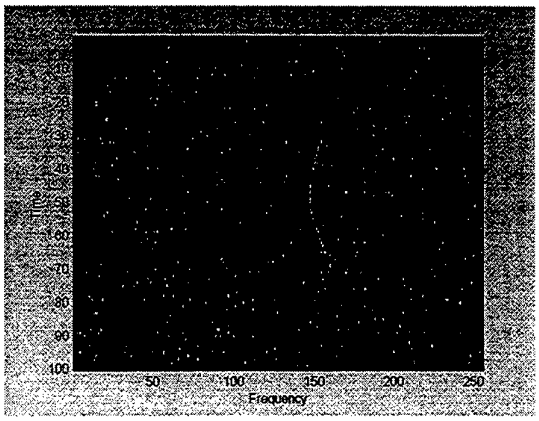


Figure 6: Discountinuous 10 dB frequency line

Figure (7) shows the RMSE statistic for the beginning of the track, i.e.

$$\sigma_{RMSE}^{beg} \triangleq \sqrt{\frac{1}{N}\sum_{j=1}^{N}[\widehat{x}_j^{beg} - x^{beg}]^2} \tag{16}$$

where $x^{beg} = 30$ for simulations.

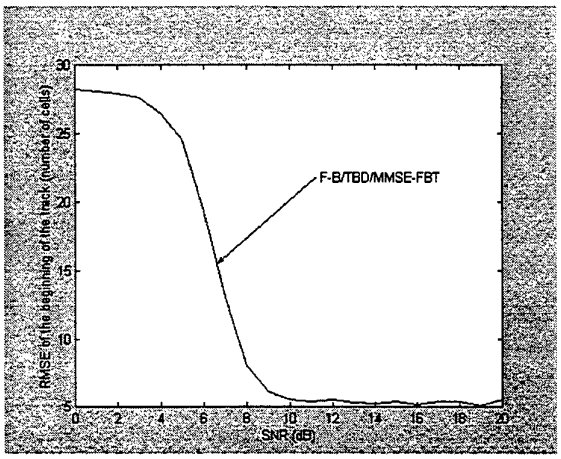


Figure 7: RMSE of the beginning of the track

Performances shows Forward-Backward state detector FBT offers correct results for SNR> 8 dB. The main goal of such dynamic state detector is to avoid the "hole effect" of NPT detector for lower SNR during the state detection sequence, by smoothing the Neyman Pearson test sequence.

6 Summary and Conclusions

In this paper, trackers for continuous and discontinuous frequency line and associated performances has been presented. The best continuous frequency line estimator is showed to be the combination between Viterbi algorithm and TBD likelihood definition. In this case, performances show good results for an SNR superior to 4 dB (after the FFT gain).

Discontinuous frequency line tracker is presented to be the association between a frequency sequence estimator and a state detector. To avoid the "hole effect" of the NPT detector, we use a dynamic state detector based on a Forward-Backward algorithm. In this case, the minimum acceptable SNR is about 8 dB (after FFT gain) for discontinuous frequency lines.

A further work will be to construct a multiple frequency discontinuous line tracker using these new results.

References

[1] R. Barrett and D. Holdsworth. Frequency tracking using hidden markov models with amplitude and phase information. *IEEE Transactions on signal processing, vol 41, no. 10*, pages 2965–2976, October 1993.

[2] R. Bethel and G. Paras. A pdf multitarget tracker. *IEEE Transactions on aerospace and electronic systems, vol. 30, no. 2*, pages 386–403, April 1994.

[3] W. S. Burdic. *Under acoustic systel analysis*, volume 1. Prentice Hall, 1984, pp 151.

[4] A. Jaffer and al. Improved detection and tracking of dynamic signals by bayes-markov techniques. *Proc. ICASSP'83, vol. 2*, pages 575–578.

[5] C. Jauffret and Y. Bar-Shalom. Track formation with bearing and frequency measurement in clutter. *IEEE Transactions on aerospace ans electronic systems, vol. 26*, pages 999–1010, November 1990.

[6] C. Jauffret and D. Bouchet. Frequency line tracking on a lofargram. *Asilomar*, 1996.

[7] M. Karan. Frequency tracking and hidden markov models. *Phd Thesis*, March 1995.

[8] R. Larson and J. Peschon. A dynamic programming approach to trajectory estimation. *IEEE Transactions on automatic control, vol. AC-11.*, pages 537–540, July 1966.

[9] L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine, vol. 3*, pages 4–16, January 1986.

[10] L. Scharf and H. Elliott. Aspects of dynamic programming in signal and image processing. *IEEE Transactions on automatic control, vol. AC-26*, pages 1018–1029, October 1981.

[11] R. Short and J. Toomey. Detection and estimation of frequency-random signals. *IEEE Transactions on information theory*, pages 940–946, 1982.

[12] R. Streit and R. Barrett. Frequency line tracking using hidden markov models. *IEEE Transactions on acoustics, speech, and signal processing, vol.38, no. 4*, pages 586–598, April 1990.

[13] J. Wolcin. Maximum a posteriori estimation of narrow-band signal parameters. *J. Acoust. Soc. Am, vol 68. no. 1*, pages 174–178, July 1980.

[14] X. Xie and R. Evans. Multiple target tracking and mutiple frequency line tracking using hidden markov models. *IEEE Transactions on signal processing, vol. 39*, pages 2659–2676, 1991.

Approximations of the Cramér-Rao Bound for Multiple Target Motion Analysis *

J.-P. Le Cadre , IRISA/CNRS , Campus de Beaulieu, 35 042, Rennes, France †,
H. Gauvrit , LTSI/Université de Rennes 1, Campus de Beaulieu, 35 042, Rennes,
and F. Trarieux, IRISA.

October 24, 1998

Abstract

This study takes place in the analysis of multiple target motion analysis (MTMA for the sequel), when the system state is not directly observed. The Cramér-Rao Lower Bound (CRLB) is a widely reference for assessing estimation performance. The lack of explicit bounds on the performance of MTMA remains an important issue in the tracking community. The problem is immersed in the general framework of the estimation of normal mixture parameters. Then, a general formulation of the CRLB is given. Our contribution is in the calculation of convenient approximations of the bounds relative to source kinematic parameters.

1 Introduction

This study takes place in the analysis of multiple target motion analysis (MTMA for the sequel), when the system state is not directly observed. A classical example is that of passive MTMA where measurements are only made of estimated bearings [1]. Such systems are basically used in passive sonar [1], infrared tracking or electronic warfare. The Cramér-Rao Lower Bound (CRLB) is a widely reference for assessing estimation performance. The CRLB is typically derived in terms of the likelihood function; i.e. for measurements distributions defined by non-random parameters, even if extensions to random parameters exist [2], [3].

The lack of explicit bounds on the performance of MTMA remains an important issue in the tracking community [4], [5],[6] and [7]. As a result, a great deal of attention has been devoted to measures of performance such track purity, correct assignment ratio [8], [9] etc. These methods are based on the discrete assignments of measurements to tracks and are thus not universally applicable. Their interest is, for a large part, due to the fact that numerous MTMA algorithms rely on "hard" association. In this meaning, this type of analysis is quite pertinent and sophisticated tools have thus been developed. However, there is a need for simple and (relatively) explicit formulations of the CRLB in the MTMA context. These bounds will be here developed in a general context which employs a probabilistic structure on the measurement to target association.

The difficulty of obtaining CRLB for MTMA is due to a need for performing association between measurements and tracks, and to incorporate this basic step in the CRLB calculation. In fact, when properly cast, a CRLB for the MTMA does exist, even if its evaluation may be difficult [10].This problem will be overcomed by means of a mixture modelling of the likelihood [11], [12]. It is then possible to immerse

*This work has been supported by DCN/Ingénierie/Sud, (Dir. Const. Navales), France
†e-mail: lecadre@irisa.fr

the problem in the general framework of the estimation of normal mixture parameters, for which an important statistical literature exists. Furthermore, this modelling has been widely used in the derivation of the Probabilistic Multiple Hypothesis Tracking (PMHT) developed by Streit and Luginbuhl [13].

This study clearly takes place in the general framework developed by Graham and Streit [4], which will be constantly used along this paper. It is also motivated by the development of MTMA methods that do not explicitely estimate measurements to target associations. Our contribution is in the calculation of convenient approximations of the bounds relative to source kinematic parameters. The paper is organized as follows. The mixture modelling of the observations is introduced in Section 2. General calculations of the CRLB (for this observation modelling) is presented in Section 3. Approximations of mixture terms are the object of Section 4. Calculations of explicit approximations of the CRLB for MTMA are detailed in Section 5. Numerical results illustrate their accuracy.

2 Measurements probability density functions and mixture density models :

Based on standard target independence assumptions, the measurement probability density function p conditioned on the vector states Ξ and the vector of measurement associations $\mathbf{A}$ is [14] :

$$p\left(z_{k,j}|\Xi_k, \mathbf{A}_k\right) = p\left(z_{k,j}|\mathbf{X}_{k,m}\right)_{/m=a_{k,j}} ,$$

where :

$$\begin{array}{l} z_{k,j} \text{ is the } j^{\text{-th}} \text{ measurement in the scan } k \ , \\ \Xi_k = (\mathbf{X}_{k,1}, \cdots, \mathbf{X}_{k,c_k}) \text{ vector of target states at time } k \ , \\ \mathbf{A}_k = (a_{k,1}, \cdots, a_{k,n_k}) \ , \ a_{k,j} \text{ target of origin for } z_{k,j} \ . \end{array} \tag{2.1}$$

Using assumptions of conditional independence for measurements within a scan results in ($\mathbf{Z}_k$: vector of measurements at time k) :

$$p\left(\mathbf{Z}_k|\Xi_k, \mathbf{A}_k\right) = \prod_{z_{k,j}} p\left(z_{k,j}|\mathbf{X}_{k,m}\right)_{/m=a_{k,j}} \ . \tag{2.2}$$

The combinatorial nature of the problem thus clearly appears. To maintain the computation load at a reasonable level, various methods have been considered like: pruning, gathering, gating, etc. However, for these approaches, the CRLB is an ill-defined bound on the estimation errors for the resulting point estimate.

A conceptually different approach has been developed by Graham and Streit [4]; in which the measurement procedure comprises two parts, an observation of the target of origin, followed by the traditional physical measurement. The likelihood function is then the joint likelihood of measurements and associations , i.e. $p\left(\mathbf{Z}, \mathbf{A}|\Xi, \mathbf{\Pi}\right) = p\left(\mathbf{Z}|\Xi, \mathbf{A}\right)\ p\left(\mathbf{A}|\mathbf{\Pi}\right)$, where $\mathbf{\Pi}$ consists of the additional describing the frequency function of the associations $\mathbf{A}$. Practically however, the data $\mathbf{A}$ are termed the missing data since it is not available.

The dependence in $\mathbf{A}$ can now be removed through marginalization [1] :

$$p\left(\mathbf{Z}|\Xi, \mathbf{\Pi}\right) = \sum_{\mathbf{A}\in\mathcal{A}} p\left(\mathbf{Z}, \mathbf{A}|\Xi, \mathbf{\Pi}\right) \ ,$$

yielding :

$$p\left(\mathbf{Z}|\Xi, \mathbf{\Pi}\right) = \prod_{k=1}^{T} \prod_{j} \left[\sum_{m=1}^{n_k} \pi_m \, p\left(z_{k,j}|\mathbf{X}_{k,m}\right)\right] \ ; \ 0 \leq \pi_m \leq 1 \quad \sum_m \pi_m = 1 \ , \tag{2.3}$$

[1] $\mathcal{A}$ domain of possible association vectors

which is typically referred to as (normal) mixture density. This *approximation* of the likelihood will be of constant use subsequently.

3 General calculations

For this section and the rest of the paper, we consider the following scenario : *two* sources move with a constant velocity vector. They are (partially) observed through a (passive) receiver (sonar, IR, radar). Measurements are generally bearings (sonar,IR), possibly frequency (sonar) or even range (radar). For the sake of simplicity, we shall restrict our attention to planar problems.
For deterministic motions, the source trajectories are defined by initial conditions i.e. a 4-dimensional vector whose components are x, y-position and x, y-velocity.

Associated with these deterministic models are their a priori probabilities π_1 and π_2 ($\pi_1 + \pi_2 = 1$). Let $\mathbf{X}_1$ and $\mathbf{X}_2$ the source state vectors and denote $q \triangleq \pi_1$, then the scenario parameters are represented by the following Φ-vector, $\Phi = (\mathbf{X}_1, \mathbf{X}_2, q)$. The batch data are denoted by Z. At each scan, 2 (or less) measurement-models are observed, each one with a probability q and $1-q$, i.e. :

$$z_j(k) = \begin{cases} \beta_1(\mathbf{X}_1,\, k) + w_1(k) & \text{if } z_j(k) \text{ originates from source 1,} \\ \beta_2(\mathbf{X}_2,\, k) + w_2(k) & \text{if } z_j(k) \text{ originates from source 2,} \end{cases} \tag{3.1}$$

In (3.1), w_1 and w_2 are the measurement noises. We assume them independent (from scan to scan), gaussian, with known and constant [2] variances (σ_1^2 and σ_2^2). The likelihood function then takes the following form :

$$\begin{aligned} p\,(Z|\Phi) &= \prod_{k=1}^{T} \prod_{j} p(z_j(k)|\Phi), \\ &= \prod_{k=1}^{T} \prod_{j} \left\{ q\, p_1(z_j(k)|\mathbf{X}_1) + (1-q)\, p_2(z_j(k)|\mathbf{X}_2) \right\} . \end{aligned} \tag{3.2}$$

We are now dealing with the calculation of the Fisher Information matrix (FIM for the rest). For the sake of simplicity, the following assumption is made (unity probability of detection for each source and no false alarm) :

$$c_k = n_k = 2\ , \forall k \in \{1, \cdots T\}$$

First, let us recall the classical expression of the FIM for the unique source case [3] :

$$\begin{aligned} \text{FIM} &= \mathbb{E}\left\{ \nabla_{\mathbf{X}_1} (\log p(Z|\mathbf{X}_1) \nabla^*_{\mathbf{X}_1} (\log p(Z|\mathbf{X}_1) \right\} , \\ &= \sum_{k=1}^{T} \frac{1}{\sigma^2}\, \mathbf{G}_1(k)\, \mathbf{G}_1^*(k) , \\ \text{where : } \mathbf{G}_1(k) &= \nabla_{\mathbf{X}_1}\, \beta_1(\mathbf{X}_1, k) , \\ &= \left(\frac{\cos\beta_1(k)}{r(k)}, -\frac{\sin\beta_1(k)}{r(k)}, k\frac{\cos\beta_1(k)}{r(k)}, -k\frac{\sin\beta_1(k)}{r(k)} \right)^* . \end{aligned} \tag{3.3}$$

This calculation [15], [16] may be easily extended to the mixture model (3.1 , 3.2), thus yielding :

[2] Along the measurement batch
[3] No assignation problem then exists

Proposition 1 *Let* FIM *the Fisher information matrix associated with the mixture model (3.1, 3.2), then :*

$$\text{FIM} = 2\sum_{k=1}^{T} I(k) \quad , where : I(k) = \begin{pmatrix} I_{11}(k) & I_{12}(k) & I_{13}(k) \\ I_{12}^*(k) & I_{22}(k) & I_{23}(k) \\ I_{13}^*(k) & I_{23}^*(k) & I_{33}(k) \end{pmatrix} ,$$

and :

$$I_{11}(k) = \frac{q^2 M_{2,0}(p_1,p_1,k)}{\sigma_1^2}\,\mathbf{G}_1(k)\mathbf{G}_1^*(k) \quad , I_{12}(k) = \frac{q(1-q)M_{1,1}(p_1,p_2,k)}{\sigma_1\sigma_2}\,\mathbf{G}_1(k)\mathbf{G}_2^*(k) ,$$

$$I_{22}(k) = \frac{(1-q)^2 M_{0,2}(p_2,p_2,k)}{\sigma_2^2}\,\mathbf{G}_2(k)\mathbf{G}_2^*(k) \quad , I_{13}(k) = \frac{-M_{1,0}(p_1,p_2,k)}{\sigma_1}\,\mathbf{G}_1(k) ,$$

$$I_{23}(k) = \frac{M_{0,1}(p_1,p_2,k)}{\sigma_2}\,\mathbf{G}_2(k) \quad , I_{33}(k) = \frac{1}{q(1-q)}\,(1 - M_{0,0}(p_1p_2,k) ,$$

$$M_{m,n}\,(p_i,p_j,k) = \int_{-\infty}^{\infty} \frac{p_i(z|\mathbf{X}_i)p_j(z|\mathbf{X}_j)}{p(z|\Phi)} \left(\frac{z-\beta_i(\mathbf{X}_i,k)}{\sigma_i}\right)^m \left(\frac{z-\beta_j(\mathbf{X}_j,k)}{\sigma_j}\right)^n dz .$$

Proof :
Consider, for instance, the calculation of I_{11} :

$$I_{11} = \mathbb{E}\left\{\nabla_{\mathbf{X}_1}(\log p(Z|\Phi)\nabla_{\mathbf{X}_1}^*(\log p(Z|\Phi)\right\} ,$$

where :

$$\nabla_{\mathbf{X}_1}\log p(Z|\Phi) = \sum_{k=1}^{T}\sum_{j=1}^{2} \nabla_{\mathbf{X}_1}\log\left\{q\,p_1(z_j(k)|\mathbf{X}_1) + (1-q)\,p_2(z_j(k)|\mathbf{X}_2)\right\} ,$$

$$= \sum_{k=1}^{T}\sum_{j=1}^{2} \frac{q\,p_1(z_j(k)|\mathbf{X}_1)}{\sigma_1^2\,p(z_j(k)|\Phi)}\,(z_j(k) - \beta_1(\mathbf{X}_1,k))\,\nabla_{\mathbf{X}_1}\beta_1(\mathbf{X}_1,k) . \qquad (3.4)$$

I_{11} is then obtained by calculating the expectation of the dyadic product of the term (3.4). The calculation is greatly simplified by the following remark : all the cross products yield null contributions. We then obtain :

$$I_{11} = \sum_{k=1}^{T}\sum_{j=1}^{2} \frac{q^2}{\sigma_1^2}\,\nabla_{\mathbf{X}_1}\beta_1(\mathbf{X}_1,k)\,\nabla_{\mathbf{X}_1}^*\beta_1(\mathbf{X}_1,k)$$
$$\times\,\mathbb{E}\left\{\frac{p_1^2(z_j(k)|\mathbf{X}_1)}{p^2(z_j(k)|\Phi)}\left(\frac{z_j(k)-\beta_1(\mathbf{X}_1,k)}{\sigma_1}\right)^2\right\} ,$$
$$= \sum_{k=1}^{T} m_k\frac{q^2}{\sigma_1^2}\,\nabla_{\mathbf{X}_1}\beta_1(\mathbf{X}_1,k)\,\nabla_{\mathbf{X}_1}^*\beta_1(\mathbf{X}_1,k)$$
$$\times\,2\,\mathbb{E}\left\{\frac{p_1^2(z|\mathbf{X}_1)}{p^2(z|\Phi)}\left(\frac{z-\beta_1(\mathbf{X}_1,k)}{\sigma_1}\right)^2\right\} . \qquad (3.5)$$

Denoting $\mathbf{G}_1(k)$ the gradient vector $\nabla_{\mathbf{X}_1}\beta(\mathbf{X}_1,k)$, expression (3.4) of I_{11} then follows. Calculation of I_{12} and I_{22} is quite similar.

It remains to calculate and approximate the scalar mixture terms $M_{m,n}$. Using an elementary transformation [17] ($y = \varepsilon(z-\bar{\mu})/\bar{\sigma}$, $\bar{\mu} \triangleq (\mu_1+\mu_2)/2$, $\bar{\sigma} \triangleq (\sigma_1\sigma_2)^{1/2}$, $\varepsilon = \pm 1$), the mixture terms $M_{m,n}(p_i,p_j,k)$ are considerably simplified. For the sequel, we shall adopt the very concise notations of

Behboodian [17] (i.e. : $d = |\mu_2 - \mu_1|/2\bar{\sigma}$, $r = \sigma_1/\sigma_2$, $d_1 = -d$, $d_2 = d$ and $r_1 = r$, $r_2 = 1/r$), yielding [4] [17] :

Lemma 1 *Let $M_{m,n}(p_i, p_j, k)$ the scalar mixture terms of Prop. 1, the following simplifications hold :*

$$\begin{aligned} M_{m,n}(p_i,p_j,k) &= \varepsilon^{m+n}\, r_i^{-m/2}\, r_j^{-n/2}\, G_{m,n}(g_i,g_j,k)\ , \\ &\quad where : G_{m,n}(g_i,g_j,k) = \int_{-\infty}^{\infty} (y-d_{i,k})^m\,(y-d_{j,k})^n\ (g_i(y)g_j(y))\,/g(y)\,dy\ , \\ &\quad and : d_k = \frac{|\beta_1(\mathbf{X}_1,k) - \beta_2(\mathbf{X}_2,k)|}{2\sqrt{\sigma_1\sigma_2}}\ , \\ &\quad g_i(y) = \frac{1}{\sqrt{2\pi r_i}} \exp\left(-\frac{1}{2r_i}(y-d_{i,k})^2\right)\ ; i = 1\,,2\ , \\ &\quad d_{2,k} = -d_{1,k} = d_k\ ,\ \varepsilon = 1\, for\, \mu_1 \le \mu_2\,, \varepsilon = -1\, for\, \mu_1 > \mu_2\ , \\ &\quad g(y) = q\,g_1(y) + (1-q)\,g_2(y)\ . \end{aligned} \tag{3.6}$$

Now, our analysis will be divided into two parts. At first, we shall examine approximations of the scalar mixture terms $M_{m,n}$. The second part consists in using these results for approximating the CRLB bounds relative to the kinematic parameters of the sources. Since this analysis is multidimensional, this part is essentially based on (linear) algebra.

4 Approximation of the mixture terms

We now restrict to close sources (i.e. $d_k \le 1$). First, for reasons we will present later, the case $r = 1$ is a special case, for which approximations of mixture terms $M_{m,n}(p_i, p_j, k)$ are particularly simple and easy to obtain. More precisely, considering a fourth-order expansion of the functions (3.6) $G_{m,n}(g_i, g_j, k)$, around 0 and relatively to d_k, the following approximations are obtained [16], for $r = 1$:

Result 1

$$\begin{aligned} &M_{0,0}(p_1,p_2,k) \approx 1 - 4\,q(1-q)\,d_k^2\ ,\ M_{1,1}(p_1,p_2,k) \approx 1 - 12\,q(1-q)\,d_k^2\ , \\ &M_{2,0}(p_1,p_1,k) \approx 1 - 4(3q-2)(1-q)\,d_k^2\ ,\ M_{0,2}(p_2,p_2,k) \approx 1 - 4q(1-3q)\,d_k^2\ , \\ &M_{1,0}(p_1,p_2,k) \approx -2q\,d_k + 8(3q-1)q(1-q)\ d_k^3\ , \\ &M_{0,1}(p_1,p_2,k) \approx 2(1-q)\ d_k + 8q(3q-2)(1-q)\ d_k^3\ . \end{aligned} \tag{4.1}$$

These approximations are quite accurate as far as $d_k \le 1$; which is, here, a convenient hypothesis. Rather surprisingly, the results obtained for the general case (i.e. $r \neq 1$) are fundamentally different. Considering r as a free parameter, the previous approach does not provide explicit results since there is no explicit expression of the integrals of the expansion of the terms $(g_i(y)\,g_j(y))/g(y)$. Then, analogously to [18], [17] a natural and rigorous approach consists in using series expansion of the functions $(g_i(y)\,g_j(y))/g(y)$. More precisely, we observe that :

$$\begin{aligned} &(g_i(y)\,g_j(y))/g(y) = \left(\frac{1}{\sqrt{2\pi r_i r_j/r}}\right)\ (h_i(y)\,h_j(y))/h(y)\ , \\ &\text{where :} \\ &h(y) = qh_1(y) + (1-q)\,r\,h_2(y) \text{ and: } h_i(y) = \exp\left[-(y-d_{i,k})^2/2\,r_i\right]\ ,\ i = 1,2\ . \end{aligned} \tag{4.2}$$

Now, it is easy to show that $qh_1(y)/(1-q)\,r\,h_2(y) < 1$ if y is in the interval $(-\infty, \alpha_1)$ or (α_2, ∞), with $\alpha_1 < \alpha_2$, and the converse (i.e. $(1-q)h_2(y)/qh_1(y) < 1$) if y is in the interval (α_1, α_2), where α_1 and α_2

[4] $\mu_1 \triangleq \beta_1(\mathbf{X}_1, k)$, $\mu_2 \triangleq \beta_2(\mathbf{X}_2, k)$

are the real roots of the following second order equation :

$$(1-r^2)\,y^2 + 2d_k(1+r^2)\,y + (d_k)^2(1-r^2) + 2r\log\left[\left(\frac{1-q}{q}\right)r\right] = 0\,, \tag{4.3}$$

if real roots exist. Using the method presented in [18], [17], the following expression of $G_{m,n}(g_i, g_j)$ is obtained :

$$G_{m,n}(g_i,g_j) = \left(\frac{1}{\sqrt{2\pi r_i r_j / r}}\right) \sum_{n=0}^{\infty} \left[\int_{-\infty}^{\alpha_1} H_n(y)\,dy + \int_{\alpha_1}^{\alpha_2} \bar{H}_n(y)\,dy + \int_{\alpha_2}^{\infty} H_n(y)\,dy\right]\,, \tag{4.4}$$

where the functions $H_n(y)$ and $\bar{H}_n(y)$ are straightforwardly deduced from above calculations and detailed in [17]. The computation of the integrals leads to deal with truncated moments of a normal distribution, which is already known. The advantage of this method lies in the fact that we approximate $G_{m,n}(g_i, g_j)$ by an alternate serie.

The above calculation is somewhat simplified if we assume that (4.3) has no real root, we then obtain [5] :

Result 2

$$\begin{aligned}
G_{0,0}(g_1,g_2,k) &= (1/(1-q))\ \textstyle\sum_{n=0}^{\infty}(-1)^n\, a_n^{-1/2} \exp\left[2d_k^2\,\frac{r}{a_n} n(n+1) - n\log\left(\frac{(1-q)r}{q}\right)\right]\,,\\
G_{2,0}(g_1,g_1,k) &= (1/(1-q))\ \textstyle\sum_{n=0}^{\infty}\left(-\frac{q}{(1-q)\,r}\right)^n a_{n+1}^{-3/2}\left[4d_k^2\,(1+n)^2\,\frac{r^3}{a_{n+1}}+1\right]\exp\left[2d_k^2\,(n+1)(n+2)\,r/a_{n+1}\right]\,,\\
G_{1,1}(g_1,g_2,k) &= (r/(1-q))\ \textstyle\sum_{n=0}^{\infty}\left(-\frac{q}{(1-q)\,r}\right)^n a_n^{-3/2}\left[4d_k^2\,n(n+1)\,\frac{r}{a_n}+1\right]\exp\left[2d_k^2\,n(n+1)\,r/a_n\right]\,,\\
G_{0,2}(g_2,g_2,k) &= (r/(1-q))\ \textstyle\sum_{n=0}^{\infty}\left(-\frac{q}{(1-q)\,r}\right)^n a_{n-1}^{-3/2}\left[r+4d_k^2\,\frac{n^2}{a_{n-1}}\right]\exp\left[2d_k^2\,(n)(n-1)\,r/a_{n-1}\right]\,,
\end{aligned} \tag{4.5}$$

where : $a_n = n(1-r^2)+1$.

An illustration of the accuracy of the above approximation is provided with fig. 1. The value of q is 0.1 ($d_k = 0.398$), and we compare (r varying from 0.1 to 1), the exact value of $M_{m,n}$ (see 3.6) with its approximation given by (4.5), for $n = 2$. The result is quite satisfactory, even for a value[6] of n as low as 2. However, when q approaches 1/2, greater values of n are needed for obtaining an accurate approximation.

A less rigorous but simpler approch consits in using a second order expansion of g_i and g_j, both with respect to d (around 0) and r (e.g. around 1). Calculations are performed by means of symbolic computation and yield :

$$\begin{aligned}
G_{0,2}(g_2,g_2) &\approx P_{0,2}(q,r) + d^2\,Q_{0,2}(q,r)\,,\\
G_{1,1}(g_1,g_2) &\approx P_{1,1}(q,r) + d^2\,Q_{1,2}(q,r)\,,\\
G_{2,0}(g_1,g_1) &\approx P_{2,0}(q,r) + d^2\,Q_{2,0}(q,r)\,.
\end{aligned} \tag{4.6}$$

The polynomials $P_{i,j}$ and $Q_{i,j}$ are detailed in Appendix B.

5 Approximations of the CRLB

5.1 Performance analysis for MTMA (reduced state vector)

We shall show now that it is possible to obtain explicit approximations of the bounds for the variance of estimated kinematic parameters. The two following ingredients are fundamental :

[5] Corresponding calculations are outlined in Appendix A

[6] n: expansion order

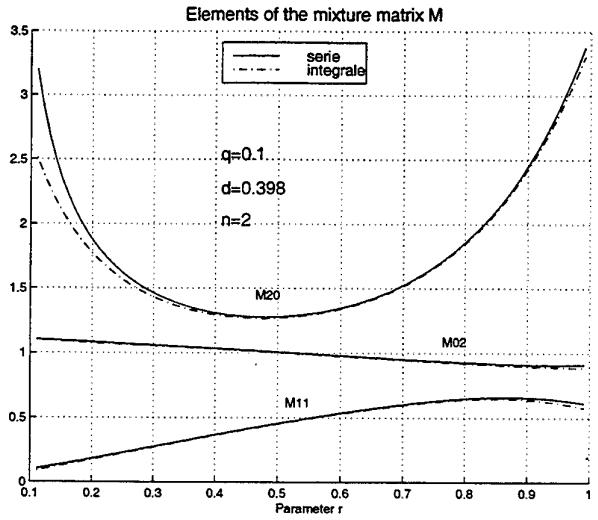


Figure 1: Order 2 approximation of $M_{m,n}$; q=0.1, n=2. Continuous line: order 2 approximation; dashed line: exact (integral) values

- kinematic parameters are modified polar coordinates (MPC) ,
- approximations of mixture terms (i.e. $:M_{m,n}$), given in section 4.

The fundamental role of MPC (β_0, $\dot{\beta}$, $\dot{r}/r$, $1/r$) in TMA has been put in evidence by Aidala and Hammel [19] and is now well recognized. Further, immersing the TMA problem in its natural *non-linear* framework leads to consider the Lie derivatives of the observation (i.e. the bearing), themselves spanned by the MPC [20]. We stress that the coordinate $(1/r)$ plays a particular role, since it is a "control" coordinate; so estimation of related component will be treated apart.

In order to facilitate the calculations, the following (partial) source state vectors will be considered throughout this section :

$$\mathbf{X}_1 = \left(\beta_1^0, \dot{\beta}_1, \ddot{\beta}_1\right)^* \ , \ \mathbf{X}_2 = \left(\beta_2^0, \dot{\beta}_2, \ddot{\beta}_2\right)^* \ . \tag{5.1}$$

In (5.1) β_i^0, $\dot{\beta}_i$, $\ddot{\beta}_i$ respectively denote the initial (i.e. at time 0) bearing, the bearing-rate and the time derivative of the bearing-rate of the i^{th} source. Also, we assume that the probability q is known. Further, note that the "usual" MPC have been slightly modified since we use $\ddot{\beta}$ in place of $\dot{r}/r$. This is quite justified since, in the absence of observer maneuver, we have $\ddot{\beta} = -2\,\dot{\beta}\,\dot{r}/r$ (see [20] for the general case). Thus, the following (quadratic) bearing model will be considered along this section :

$$\beta_i(k) = \beta_i(0) + k\,\dot{\beta}_i + \frac{k^2}{2}\,\ddot{\beta}_i \ .$$

Then, from (3.4), the FIM (relative to $\mathbf{X}_1$ and $\mathbf{X}_2$) takes the following form :

$$\begin{array}{l} \text{FIM} = \sum_{k=1}^{T} M_k \otimes \mathbf{G}_k \mathbf{G}_k^* \ , \\ \text{where :} \\ M_k = \begin{pmatrix} \frac{q^2}{\sigma_1^2}\, M_{2,0}(k) & \frac{q(1-q)}{\sigma_1\sigma_2}\, M_{1,1}(k) \\ \frac{q(1-q)}{\sigma_1\sigma_2}\, M_{1,1}(k) & \frac{(1-q)^2}{\sigma_2^2}\, M_{0,2}(k) \end{pmatrix} \ , \\ \mathbf{G}_k = (1, k, k^2/2)^* \ . \end{array} \tag{5.2}$$

Note that now the gradient vector $\mathbf{G}_k$ is *identical* for the two sources. This is due to the *coordinate choice* (i.e MPC).

It is quite reasonable to assume that the parameter d_k is sufficiently small (i.e. $d_k \leq 1$). A 3^{rd}-order expansion (w.r.t. d_k) of the components of the matrix M_k yields :

$$\begin{array}{l} M_k = M_0(k) + d_k^2\, M_1(k) \\ \text{where :} \\ M_0(k) = \left(\begin{array}{ll} \frac{q^2}{\sigma_1^2 r} \int_{-\infty}^{\infty} y^2 \frac{g_1^2(y)}{g(y)}\, dy & \frac{q(1-q)}{\sigma_1 \sigma_2} \int_{-\infty}^{\infty} y^2 \frac{g_1(y) g_2(y)}{g(y)}\, dy \\ \frac{q(1-q)}{\sigma_1 \sigma_2} \int_{-\infty}^{\infty} y^2 \frac{g_1(y) g_2(y)}{g(y)}\, dy & \frac{(1-q)^2 r}{\sigma_2^2} \int_{-\infty}^{\infty} y^2 \frac{g_2^2(y)}{g(y)}\, dy \end{array} \right) \end{array} \tag{5.3}$$

From the above expression and the Cauchy-Schwarz inequality, we deduce that M_0 is necessarily positive semi-definite. Calculation of CRLB will require convenient approximations of the mixture matrices M_0 and M_1. These approximations have been derived in section 4. In this section, it has been shown that the cases $r = 1$ and $r \neq 1$ must be considered separately since approximations are quite different. Indeed, algebraically, a major difference exists: the approximated matrix M_0 is rank-one when $r = 1$, while M_0 is full rank otherwise. The corresponding CRLB calculations will thus be considerably different.

5.1.1 The case $r = 1$:

For the case $r = 1$, the matrices M_0 and M_1 are straightforwardly deduced from (4.1), yielding :

$$M_0 = \left(\begin{array}{ll} q^2 & q(1-q) \\ q(1-q) & (1-q)^2 \end{array} \right), M_1 = \left(\begin{array}{ll} q^2(1-q)(3q-2) & 3q^2(1-q)^2 \\ 3q^2(1-q)^2 & q(1-q)^2(1-3q) \end{array} \right) \tag{5.4}$$

so, that :

$$\sigma^2\, \text{FIM} = M_0 \otimes \left(\sum_k \mathbf{G}_k\, \mathbf{G}_k^* \right) - M_1 \otimes \left(\sum_k 4\, d_k^2\, \mathbf{G}_k\, \mathbf{G}_k^* \right) . \tag{5.5}$$

It is now convenient to define the following matrices which will play a major role for the analysis :

$$\begin{array}{l} \mathcal{C}_0 \triangleq \sum_k \mathbf{G}_k\, \mathbf{G}_k^* \,,\; \mathcal{C}_1 \triangleq \sum_k \left(4\, d_k^2\, \mathbf{G}_k\, \mathbf{G}_k^*\right) \,, \\ A = M_0 \otimes \mathcal{C}_0 \,,\; B = M_1 \otimes \mathcal{C}_1 \,. \end{array} \tag{5.6}$$

Here, we note that since $\text{rank}(M_0) = 1$ ($M_0 = \mathbf{V}_0\, \mathbf{V}_0^*$) and $\mathcal{C}_0$ is invertible, the rank of A is 3. On another hand, the M_1 matrix is invertible, as well as $\mathcal{C}_1$, hence B is invertible. However, inversion of FIM must be considered with a certain care, since as $d_k^2 \ll 1$, the norm of B is (generally) quite smaller than the A one.
In fact, since A is rank deficient, we *cannot* use general formula for inverting the sum of invertible matrices. This difficulty requires to consider the eigensystem of A. Let us denote $\{\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3\}$ the eigenvectors of $\mathcal{C}_0$, as well as $\Delta = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ the (diagonal) matrix of the eigenvalues. Then, it is easily shown that the vectors $\{\mathbf{W}_1 = \mathbf{V}_0 \otimes \mathbf{V}_1 \,,\, \mathbf{W}_2 = \mathbf{V}_0 \otimes \mathbf{V}_2 \,,\, \mathbf{W}_3 = \mathbf{V}_0 \otimes \mathbf{V}_3\}$ are eigenvectors of A, $\{\lambda_i\}_{i=1}^3$ being the associated eigenvalues.
We then have :

$$\begin{array}{rcl} A & = & \mathcal{U}\, \Delta\, \mathcal{U}^* \quad , \mathcal{U} = \{\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3\} = \mathbf{V}_0 \otimes \mathcal{V} \,, \\ \text{where :} & & \mathcal{V} \triangleq \{\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3\} \,. \end{array}$$

The following inversion formula, valid for B invertible, is then instrumental [21] :

$$(B + \mathcal{U}\, \Delta\, \mathcal{U}^*)^{-1} = B^{-1} - B^{-1} \mathcal{U} \left(\Delta^{-1} + \mathcal{U}^* B^{-1} \mathcal{U} \right)^{-1} \mathcal{U}^* B^{-1} \,. \tag{5.7}$$

So, we have now to deal with the calculation of the various terms of the right member of (5.7).

Step 1)-Calculation of $B^{-1}\mathcal{U}$:

Since M_1 and C_1 are invertible, and invoking the classical results [23], i.e. :

$$\begin{array}{l}(A\otimes B)^{-1}=A^{-1}\otimes B^{-1}\,,\\ (A\otimes B)\,(C\otimes D)=AC\otimes BD\,,\end{array} \tag{5.8}$$

we obtain : $\left(\mathcal{V}\triangleq\{\mathbf{V}_1,\mathbf{V}_2,\mathbf{V}_3\}\right)$

$$B^{-1}\mathcal{U}=-(M_1^{-1}\otimes C_1^{-1})\,(\mathbf{V}_0\otimes\mathcal{V})\;=-(M_1^{-1}\mathbf{V}_0)\otimes(C_1^{-1}\mathcal{V})\,.$$

Of course a similar result holds for the conjugate term (i.e. $\mathcal{U}^*B^{-1}=-(\mathbf{V}_0^*M_1^{-1})\otimes(\mathcal{V}^*C_1^{-1})$).

Step 2)-Calculation of $\mathcal{U}^*B^{-1}\mathcal{U}$:

Using the previous result, we obtain :

$$\begin{aligned}\mathcal{U}^*B^{-1}\mathcal{U}&=-(\mathbf{V}_0^*\otimes\mathcal{V}^*)\left(M_1^{-1}\mathbf{V}_0\otimes C_1^{-1}\mathcal{V}\right)\,,\\ &=-(\mathbf{V}_0^*M_1^{-1}\mathbf{V}_0)\otimes\mathcal{V}^*C_1^{-1}\mathcal{V}\\ &=\alpha\,\mathcal{V}^*C_1^{-1}\mathcal{V}\,.\end{aligned} \tag{5.9}$$

In (5.9), α is simply a scalar ($\alpha=-\mathbf{V}_0^*M_1^{-1}\mathbf{V}_0$) factor of the 3×3-matrix $\mathcal{V}^*C_1^{-1}\mathcal{V}$. Value of α will be explicited later.

Step 3)-Calculation of $(\Delta^{-1}+\mathcal{U}^*B^{-1}\mathcal{U})^{-1}$:

From previous calculations, we deduce $(\Delta'^{-1}=\mathcal{V}\Delta^{-1}\mathcal{V}^*)$:

$$\begin{aligned}\Delta^{-1}+\mathcal{U}^*B^{-1}\mathcal{U}&=\Delta^{-1}+\alpha\,\mathcal{V}^*C_1^{-1}\mathcal{V}\,,\\ &=\mathcal{V}^*\left(\Delta'^{-1}+\alpha\,C_1^{-1}\right)\mathcal{V}\,.\end{aligned} \tag{5.10}$$

Now, the following implication holds true ($\mathcal{V}$ unitary matrix) :

$$C_0=\mathcal{V}\Delta\mathcal{V}^*\Rightarrow\Delta'^{-1}=\mathcal{V}\,\Delta^{-1}\mathcal{V}^*=C_0^{-1}\,.$$

so that :

$$\left(\Delta^{-1}+\mathcal{U}^*B^{-1}\mathcal{U}\right)^{-1}=\mathcal{V}^*\left(C_0^{-1}+\alpha\,C_1^{-1}\right)^{-1}\mathcal{V}\,. \tag{5.11}$$

Considering the preceding formula as well as the basic inversion formula (5.7), a last step is required, namely the calculation of the term $B^{-1}\mathcal{U}\mathcal{V}^*$ and of associated simplifications.

Step 4)-Calculation of $B^{-1}\mathcal{U}(\Delta^{-1}+\mathcal{U}^*B^{-1}\mathcal{U})^{-1}\mathcal{U}^*B^{-1}$:

Collecting previous results, we obtain :

$$B^{-1}\mathcal{U}\left(\Delta^{-1}+\mathcal{U}^*B^{-1}\mathcal{U}\right)^{-1}\mathcal{U}^*B^{-1}=\left[M_1^{-1}\mathbf{V}_0\otimes C_1^{-1}\mathcal{V}\right]\mathcal{V}^*\left(C_0^{-1}+\alpha\,C_1^{-1}\right)^{-1}\mathcal{V}\left[\mathbf{V}_0^*M_1^{-1}\otimes\mathcal{V}^*C_1^{-1}\right]\,. \tag{5.12}$$

A last simplification step is then :

$$\begin{aligned}\mathcal{V}\left[\mathbf{V}_0^* M_1^{-1} \otimes \mathcal{V}^* \mathcal{C}_1^{-1}\right] &= (1 \otimes \mathcal{V})\left(\mathbf{V}_0^* M_1^{-1} \otimes \mathcal{V}^* \mathcal{C}_1^{-1}\right), \\ &= (\mathbf{V}_0^* M_1^{-1}) \otimes (\mathcal{V}\mathcal{V}^* \mathcal{C}_1^{-1}), \\ &= (\mathbf{V}_0^* M_1^{-1}) \otimes \mathcal{C}_1^{-1}. \end{aligned} \tag{5.13}$$

The following result summarizes all the preceding calculations :

Proposition 2 *Under the section hypotheses, the* FIM *inverse takes the following form :*

$$\sigma^{-2}\,(\text{FIM})^{-1} = -M_1^{-1} \otimes \mathcal{C}_1^{-1} - \left(M_1^{-1}\mathbf{V}_0 \otimes \mathcal{C}_1^{-1}\right)\left(\mathcal{C}_0^{-1} + \alpha\,\mathcal{C}_1^{-1}\right)^{-1}\left(\mathbf{V}_0^* M_1^{-1} \otimes \mathcal{C}_1^{-1}\right).$$

It simply remains to calculate explicit expressions of elementary terms (i.e. α, $M_1^{-1}\mathbf{V}_0 \otimes \mathcal{C}_1^{-1}$). For $r = 1$, the following results are obtained :

$$\alpha = \frac{-1}{q(1-q)}, \tag{5.14}$$

$$M_1^{-1}\mathbf{V}_0 \otimes \mathcal{C}_1^{-1} = -\alpha \begin{pmatrix} \mathcal{C}_1^{-1} \\ \mathcal{C}_1^{-1} \end{pmatrix}. \tag{5.15}$$

Let us denote $\mathcal{P}$, the matrix $\left(\mathcal{C}_0^{-1} + \alpha\,\mathcal{C}_1^{-1}\right)^{-1}$; and $\text{FIM}^{-1}[1]$, $\text{FIM}^{-1}[2]$ the 3×3 diagonal block-matrices of FIM^{-1} corresponding to variance bounds for source 1 parameters, resp. source 2, we have :

Proposition 3 *For $r = 1$, the following approximation of the variance bounds holds* [7] *:*

$$\begin{aligned}\sigma^{-2}\,\text{FIM}^{-1}[1] &= \frac{-\alpha\,(1-3q)}{2q}\,\mathcal{C}_1^{-1} - \frac{1}{q^2(1-q)^2}\,\mathcal{C}_1^{-1}\mathcal{P}\mathcal{C}_1^{-1}, \\ \sigma^{-2}\,\text{FIM}^{-1}[2] &= \frac{-\alpha\,(2-3q)}{2(q-1)}\,\mathcal{C}_1^{-1} - \frac{1}{q^2(1-q)^2}\,\mathcal{C}_1^{-1}\mathcal{P}\mathcal{C}_1^{-1}. \end{aligned} \tag{5.16}$$

A further step of approximation may be considered for very close sequence of bearings. More precisely, if we assume that $d_k \ll 1$ then we can reasonably assume that (element-wise) $\mathcal{C}_0^{-1} \ll \mathcal{C}_1^{-1}$, so that $(\mathcal{P} \approx \alpha^{-1}\,\mathcal{C}_1)$:

$$\begin{aligned}\text{FIM}^{-1}[1] &\approx \frac{\sigma^2}{2\,q^2}\,\mathcal{C}_1^{-1} \\ \text{FIM}^{-1}[2] &\approx \frac{\sigma^2}{2\,(q-1)^2}\,\mathcal{C}_1^{-1}. \end{aligned} \tag{5.17}$$

The simplicity of (5.17) is rather striking, and the result appears quite natural. In particular, the sequence of "normalized distance" (i.e. $\{d_k\}_k$) plays the fundamental role. It also interesting to consider the diagonal terms of $\mathcal{C}_1^{-1}$ which approximate the CRLB for kinematic parameters and are themselves (roughly) approximated by considering a linear model of d_k (i.e. $d_k = k\,d$). Denoting c the value of the difference of initial bearings (i.e. $c \triangleq \beta_1^0 - \beta_2^0$), elementary calculations yield ($d = a(\frac{c}{T})$) :

$$\begin{aligned}\mathcal{C}_1^{-1}(1,1) &\approx \frac{15\,(105 + 280a + 276a^2 + 120a^3 + 20a^4)}{(175 + 525a + 615a^2 + 355a^3 + 105a^4 + 15a^5 + a^6)\;(c^2T)}, \\ \mathcal{C}_1^{-1}(2,2) &\approx \frac{60\,(560 + 1260a + 1005a^2 + 340a^3 + 48a^4)}{(175 + 525a + 615a^2 + 355a^3 + 105a^4 + 15a^5 + a^6)\;(c^2T^3)}, \\ \mathcal{C}_1^{-1}(3,3) &\approx \frac{6300\,(20 + 40a + 28a^2 + 8a^3 + a^4)}{(175 + 525a + 615a^2 + 355a^3 + 105a^4 + 15a^5 + a^6)\;(c^2T^5)}. \end{aligned} \tag{5.18}$$

[7] Note that if q is changed in $1-q$, then $\text{FIM}^{-1}[1]$ is changed in $\text{FIM}^{-1}[2]$

5.1.2 The case $r \neq 1$:

Then, the matrices M_0 and M_1 are *both* invertible. Thus, it is possible to apply the general inversion formula [21], valid for A and B invertible :

$$\begin{aligned} (A+B)^{-1} &= A^{-1} - A^{-1}\left(A^{-1}+B^{-1}\right)^{-1} A^{-1} . \\ A^{-1} &= M_0^{-1} \otimes \mathcal{C}_0^{-1} \quad , B^{-1} = M_1^{-1} \otimes \mathcal{C}_1^{-1} \end{aligned} \tag{5.19}$$

As previously, it may be reasonably assumed that (element-wise) $\mathcal{C}_0^{-1} \ll \mathcal{C}_1^{-1}$, so :

$$\left(A^{-1}+B^{-1}\right)^{-1} \approx B ,$$

therefore, using (5.8) :

$$\begin{aligned} \sigma^2 \, \text{FIM}^{-1} &\approx M_0^{-1} \otimes \mathcal{C}_0^{-1} - (M_0^{-1} \otimes \mathcal{C}_0^{-1})(M_1 \otimes \mathcal{C}_1)(M_0^{-1} \otimes \mathcal{C}_0^{-1}) , \\ &= M_0^{-1} \otimes \mathcal{C}_0^{-1} - \left(M_0^{-1} M_1 M_0^{-1}\right) \otimes \left(\mathcal{C}_0^{-1} \mathcal{C}_1 \mathcal{C}_0^{-1}\right) . \end{aligned} \tag{5.20}$$

In general (see sect. 4), the matrices M_0 and M_1 are relatively complicated. However, since the diagonal elements of $\mathcal{C}_0^{-1}$ play a major role it is interesting to approximate them. We obtain :

$$\begin{aligned} \mathcal{C}_0^{-1}(1,1) &\approx \frac{3(102\,T^3 + 51\,T^4)}{T\,(86\,T^3 + 65\,T^4)} \propto \frac{1}{T} , \\ \mathcal{C}_0^{-1}(2,2) &\approx \frac{12(30\,T + 16\,T^2)}{(216\,T^4 + 65\,T^5)} \propto \frac{1}{T^3} , \\ \mathcal{C}_0^{-1}(3,3) &\approx \frac{3645\,T}{(151\,T^5 + 65\,T^6)} \propto \frac{1}{T^5} . \end{aligned} \tag{5.21}$$

Even if these results have their own interest, it must be noted that very different values of σ_1 and σ_2 generally result in track coalescence [7]. The problem is then relevant of hypothesis testing.

5.2 Performance analysis for MTMA (complete state vector)

The previous analysis may be, rather easily, extended to the estimation of the complete source state vector. Again, the analysis becomes possible by considering MPC as the general framework. To avoid consideration of particular scenario we shall deal with a system constituted of two separated receivers [24]. The TMA problem then becomes completely observable. At first, kinematic relations will be considered, then allowing us to utilize the framework of the previous section.

The problem is defined as follows. Two (fixed) receivers are placed on the x-line, the first one at $(0,0)$ and the second at $(d,0)$. For both receivers, measurements are bearings-only (β_1 and β_2). Direct calculations yield :

$$\begin{aligned} \dot{\beta}_1 &= \frac{\det(\mathbf{v},\mathbf{r_1})}{r^2} = \frac{1}{r}\left(v_x \cos\beta_1 - v_y \sin\beta_1\right) , \\ \dot{\beta}_2 &= \frac{\det(\mathbf{v},\mathbf{r_2})}{r_2^2} = \frac{r\left(v_x \cos\beta_1 - v_y \sin\beta_1\right) - d v_y}{r^2 + 2dr \sin\beta_1 + d^2} , \\ &= \left(\dot{\beta}_1 - d\,v_y/r^2\right)\left(1 + 2\,d/r\,\sin\beta_1 + d^2/r^2\right)^{-1} . \end{aligned} \tag{5.22}$$

We then consider an expansion of $\beta_2, \dot{\beta}_2, \ddot{\beta}_2$, with respect to $\varepsilon \triangleq \frac{d}{r}$, around 0, yielding :

$$\beta_2^0 \approx \beta_1^0 - \frac{d}{r} ,$$

$$\begin{aligned}\dot{\beta}_2 &\approx \dot{\beta}_1\left(1-2\frac{d}{r}\sin\beta_1\right)\,,\\ \ddot{\beta}_2 &\approx \ddot{\beta}_1-\frac{d}{r}\left(2\,\dot{\beta}_1^2\,\cos\beta_1+3\ddot{\beta}_1\,\sin\beta_1\right)\,.\end{aligned}\tag{5.23}$$

For the two sources ($j=1,2$), the following modelling of $\beta_1^j(k)$, $j=1,2$ is considered :

$$\begin{aligned}\beta_1^j(k) &= \beta_1^j(0)+k\,\dot{\beta}_1^j+\frac{k^2}{2}\,\ddot{\beta}_1^j \quad k=1,\ldots,T\,,\\ \beta_2^j(k) &= \beta_1^j(0)+k\,\dot{\beta}_1^j+\frac{k^2}{2}\,\ddot{\beta}_1^j\\ &\quad-\left[d+2k\,d\,\dot{\beta}_1^j\sin\beta_1^j(0)+\frac{k^2d}{2}\,\left(2\,(\dot{\beta}_1^j)^2\,\cos\beta_1^j(0)+3\ddot{\beta}_1^j\sin\beta_1^j(0)\right)\right]\,\frac{1}{r}\,.\end{aligned}\tag{5.24}$$

Source trajectory itself is determined by the 4-dimensional state vector $\mathbf{X}$, $\mathbf{X}=\left(\beta_1^0,\dot{\beta}_1,\ddot{\beta}_1,\frac{1}{r}\right)^*$ and, for a source, the gradient vectors $\mathbf{G}_k^1$ and $\mathbf{G}_k^2$ associated with the measurements of receiver 1, respectively receiver 2 stand as follows :

$$\mathbf{G}_k^1=\begin{pmatrix}1\\ k\\ \frac{k^2}{2}\\ 0\end{pmatrix}\quad;\ \mathbf{G}_k^2=\begin{pmatrix}1-2\frac{kd}{r}\,\dot{\beta}_1\cos\beta_1-\frac{k^2d}{2r}\left(3\ddot{\beta}_1\,\cos\beta_1-2\dot{\beta}_1^2\,\sin\beta_1\right)\\ k\left(1-2\,\frac{d}{r}\,\sin\beta_1-k\frac{d}{r}\,\dot{\beta}_1\cos\beta_1\right)\\ \frac{k^2}{2}\left(1-3\,\frac{d}{r}\,\sin\beta_1\right)\\ -\left(d+2kd\,\dot{\beta}_1\,\sin\beta_1+\frac{k^2d}{2}\,(2\dot{\beta}_1^2\,\cos\beta_1+3\ddot{\beta}_1\,\sin\beta_1\,)\right)\end{pmatrix}\tag{5.25}$$

Since we are interested with close source trajectories, it is quite reasonable to assume that these gradients are *independent* of the source index. Now, let us denote FIM_1 and FIM_2 the Fisher information matrices associated with receiver 1 and 2, and FIM the global one. The measurements on receiver 1 and 2 being independent, we have :

$$\begin{aligned}\text{FIM} &= \text{FIM}_1+\text{FIM}_2\,,\\ &\quad\text{where :}\\ \text{FIM}_1 &= M_0\otimes\left(\sum_k\mathbf{G}_k^1\,\mathbf{G}_k^{1,*}\right)-M_1\otimes\left(\sum_k 4d_k^2\,\mathbf{G}_k^1\,\mathbf{G}_k^{1,*}\right)\,,\\ \text{FIM}_2 &= M_0\otimes\left(\sum_k\mathbf{G}_k^2\,\mathbf{G}_k^{2,*}\right)-M_1\otimes\left(\sum_k 4d_k^2\,\mathbf{G}_k^2\,\mathbf{G}_k^{2,*}\right)\,,\\ \mathbf{G}_k^2 &= \mathbf{G}_k^1+\mathbf{V}_k\,,\end{aligned}\tag{5.26}$$

so, that :

$$\text{FIM}=M_0\otimes\underbrace{\left(\sum_k\mathbf{G}_k^1\,\mathbf{G}_k^{1,*}+\sum_k\mathbf{G}_k^2\,\mathbf{G}_k^{2,*}\right)}_{\mathcal{C}_0}-M_1\otimes\underbrace{\left(\sum_k 4d_k^2\,\mathbf{G}_k^1\,\mathbf{G}_k^{1,*}+\sum_k 4d_k^2\,\mathbf{G}_k^2\,\mathbf{G}_k^{2,*}\right)}_{\mathcal{C}_1}\,.\tag{5.27}$$

We shall restrict our attention to the case $r=1$. Then, similarly to sect. 5.1.1 the matrix M_0 is also rank-one, so the calculation of FIM^{-1} is identical in its principle, yielding :

$$\sigma^{-2}\,\text{FIM}^{-1}=-M_1^{-1}\otimes\mathcal{C}_1^{-1}-\left(M_1^{-1}\mathbf{V}_0\otimes\mathcal{C}_1^{-1}\right)\left(\mathcal{C}_0^{-1}+\alpha\,\mathcal{C}_1^{-1}\right)^{-1}\left(\mathbf{V}_0^*M_1^{-1}\otimes\mathcal{C}_1^{-1}\right)\,.\tag{5.28}$$

In this case, further simplifications occur. For instance, eqs. (5.14) still hold true, yielding finally :

$$\text{FIM}^{-1}[1] \approx \frac{\sigma^2}{2\,q^2}\,\mathcal{C}_1^{-1}\,. \tag{5.29}$$

We are now concerned with the estimation of *both* the kinematic parameters and the probability q. The FIM is then a Hermitian 7×7 matrix, of the following form :

$$\text{FIM} = \left(\begin{array}{c|c} \text{FIM}_c & \frac{-1}{\sigma_1}\sum_k m_{10}(k)\,\mathbf{G}_k \\ & \frac{1}{\sigma_2}\sum_k m_{01}(k)\,\mathbf{G}_k \\ (\text{" "})^* & \beta \end{array} \right) \tag{5.30}$$

We consider again the case $r = 1$. Then, using the partitioned matrix inversion and the Woodbury lemmas [21],[22], we obtain :

$$\text{FIM}_{c,q}^{-1} \approx \text{FIM}_c^{-1}\,. \tag{5.31}$$

This result means that the CRLB is not significantly affected by the estimation of the parameter q. Calculations are detailed in the Appendix C.

6 Numerical results

First, we present the multiple source scenario, common for all the results of this section. Two (close) sources move with a constant velocity vector (rectilinear and uniform motion). Their trajectories are represented in the x, y-plane in fig. 2. The kinematic parameters ($r_x(0) = 15\,\text{km}, r_y(0) = 15\,\text{km}, v_x = 12\,m/s, v_y = 6\,m/s$) of target 1 are fixed (solid line), while that of target 2 take 15 different values corresponding ($r_x(0) = 22\,\text{km} \rightarrow 20,6\,\text{km}, r_y(0) = 18\,\text{km}, v_x = 11.5\,m/s, v_y = 5\,m/s$) to various initial positions (dashed lines). The receiver is fixed, at the origin. The (exact) observations (i.e. the bearings) associated with these scenarios are represented in fig. 2. The measurement noise is identical for the two sources and constant throughout the whole scenario ($\sigma(\hat{\beta}) = 1/8\,\text{rd}$). We note that the two targets have close bearings, so that the assumption $d_k \leq 1$ be satisfied.

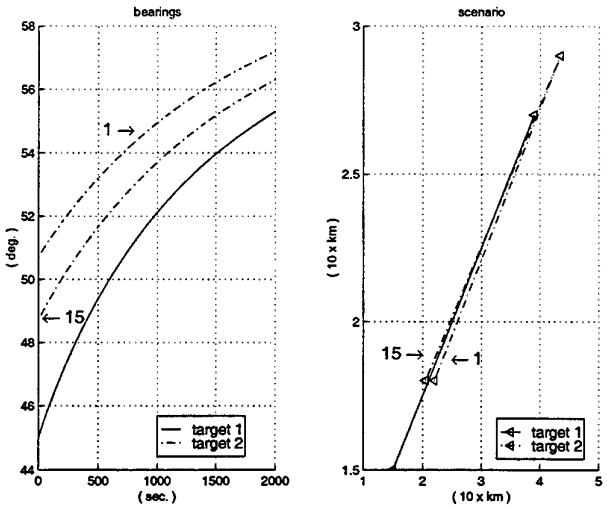


Figure 2: The multiple source scenario. Right: source trajectories; left: bearings versus time.

Accuracy of the approximations of the variance bounds is illustrated in fig. 3. The values of $\Delta\beta(0)$ (ranging from 5.7 to 3.7 deg.) correspond to the various initial positions of target 2. The solid lines represent the exact values of the lower bound relative to the estimation of $\beta_2(0)$ (left) resp. g_2 ($g =$

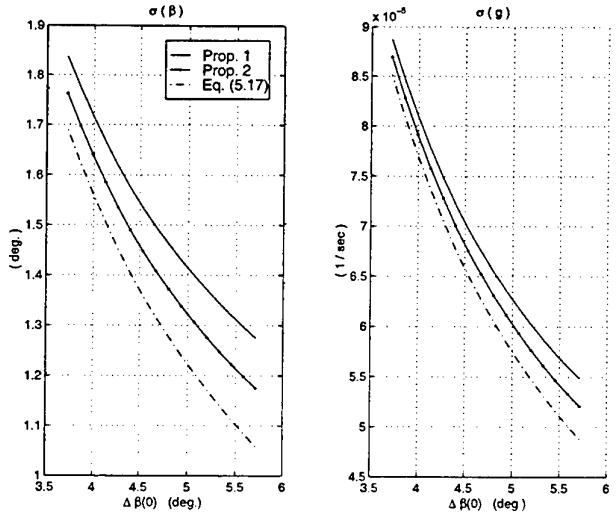


Figure 3: Accuracy of the CRLB approximations ($\sigma(\beta)$ and $\sigma(g)$).

$\dot{r}/r$), as given by Prop. 1. The continuous-dotted lines illustrate approximation given by Prop. 2 or 3; while the dashed line represents the simpler (and the more explicit approximation) given by (5.17). The approximation given by Prop. 2 or 3 performs satisfactorily; while the simpler one (5.17) is still quite acceptable. We note that, relatively to the initial measurement variance, the variance of $\hat{\beta}_2(0)$ is considerably reduced.

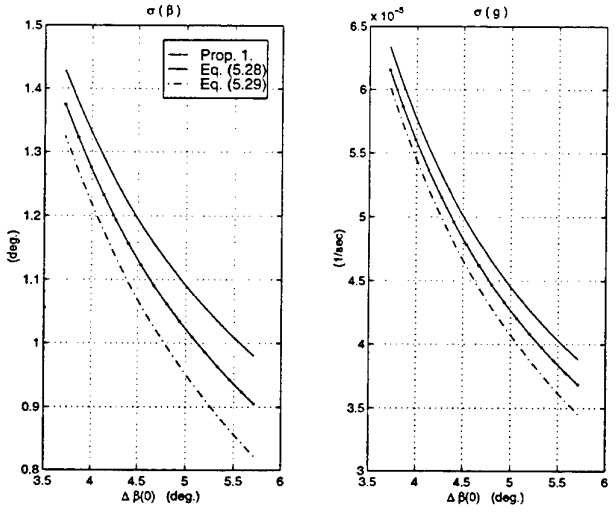


Figure 4: Accuracy of the CRLB approximations ($\sigma(\beta)$ and $\sigma(g)$, two receivers) .

We are now dealing with the estimation of the complete state vector. The source trajectories are unchanged; but, this time, two (fixed) receivers are considered (both on the x axis, separated by a distance of 2 km). In fig. 4, exact bounds (Prop. 1) are compared with approximations given by (5.28) resp. (5.29) (continuous-dotted for (5.28) and dashed for (5.29)). Then, this analysis is extended to the lower bound (see fig. 5) relative to the estimation of the "missing" coordinate (i.e. $1/r$). Again, the quality of the approximations is quite satisfactory. We note that, in comparison with the unique source case, the value of $\sigma(\hat{r})$ is very important.

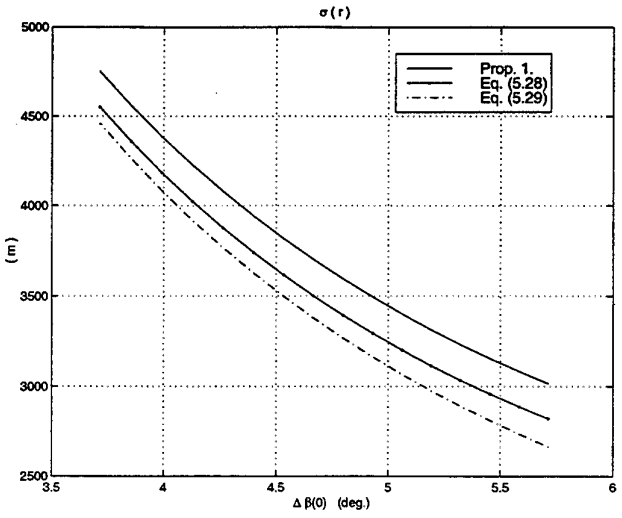


Figure 5: Accuracy of the CRLB approximations ($\sigma(r)$, two receivers)

7 Conclusion

Based on the use of modified polar coordinates and of convenient approximations of the mixture matrices, *explicit* approximations of the CRLB for MTMA have been developed. The pertinence of the *approximated* bounds has been illustrated by numerical comparisons.

A Appendix A :

This appendix deals with the calculation of $G_{0,0}(g_1, g_2, k)$. We consider the serie expansion presented in (4.4), but this time we assume that the second order equation (4.3) has no real root. We then have ($c_n = 1/\left((1-q)r\right)\,[(-q/(1-q)r)^n]$)

$$\begin{aligned} G_{0,0}(g_1,g_2) &= \sqrt{r/2\pi}\sum_{n=0}^{\infty}\int_{-\infty}^{\infty} H_n(y)\,dy\,, \\ &= \sqrt{r/2\pi}\sum_{n=0}^{\infty} c_n \int_{-\infty}^{\infty} h_1^{n+1}(y)\,h_2^{-n}(y)\,dy\,. \end{aligned} \tag{A.1}$$

Thus, the basic point is the calculation of the integral $\int_{-\infty}^{\infty} h_1^{n+1}(y)\,h_2^{-n}(y)\,dy$ (where: $h_i(y) = \exp\left[-(y-d_i)^2/\,2r_i\right]$, $i = 1, 2$), i.e. $\int_{-\infty}^{\infty} \exp\left[-(n+1)(y+d)^2/2r + n(y-d)^2\,r/2\right]$. Considering the following quadratic form factorization ($\alpha = n\,r - \frac{(n+1)}{r}$, $\beta = n\,r + \frac{(n+1)}{r}$) :

$$\begin{aligned} -(n+1)(y+d)^2/2r + n(y-d)^2\,r/2 &= \frac{1}{2}\left[\alpha\,y^2 - 2d\,\beta\,y + \alpha\,d^2\right]\,, \\ &= \frac{1}{2}\left[\alpha\left(y - d\frac{\beta}{\alpha}\right)^2 + d^2\left(\frac{\alpha^2-\beta^2}{\alpha}\right)\right]\,, \end{aligned} \tag{A.2}$$

we obtain :

$$\int_{-\infty}^{\infty} h_1^{n+1}(y)\,h_2^{-n}(y)\,dy = \exp\left[\frac{d^2}{2}\left(\frac{\alpha^2-\beta^2}{\alpha}\right)\right]\int_{-\infty}^{\infty}\exp\left[\frac{\alpha}{2}\left(y - d\frac{\beta}{\alpha}\right)^2\right]dy\,, \tag{A.3}$$

Under the hypothesis $r < 1$, we have $\alpha < 1$. Thus, the above integral exists. Making necessary variable changes, we have (integration of a normal density) :

$$\int_{-\infty}^{\infty}\exp\left[\frac{\alpha}{2}\left(y - d\frac{\beta}{\alpha}\right)^2\right]dy = \sqrt{2\,\pi}\left(\frac{r}{n(1-r^2)+1}\right)^{1/2}\,, \tag{A.4}$$

so that, finally (with : $a_n \triangleq n(1-r^2)+1$) :

$$G_{0,0}(g_1,g_2,k) = \frac{1}{(1-q)}\sum_{n=0}^{\infty}(-1)^n\,a_n^{-1/2}\,\exp\left[2d^2\frac{r}{a_n}\,n(n+1) - n\log\left(\frac{(1-q)r}{p}\right)\right]\,, \tag{A.5}$$

yielding (4.5).

B Appendix B

The polynomials $P_{i,j}$ and $Q_{i,j}$ (see (4.6)) are detailed below :

$$\begin{aligned} P_{0,2} &= 3(1-r) + 3q(1-r) + qr(r-1) + r^2 + 10q^2(r-1)^2\,, \\ P_{1,1} &= -7q + r + 16qr - 9qr^2 + 10q^2(r-1)^2\,, \\ P_{2,2} &= 7 - 15r + 9r^2 + q(-17 + 36r - 19r^2) + 10q^2(r-1)^2\,. \end{aligned} \tag{B.1}$$

and :

$$\begin{aligned} Q_{0,2} &= 468q^4(r-1)^2 - 12q^3(37 - 77r + 40r^2) + q^2(70 - 156r + 89r^2) - q\,, \\ Q_{1,1} &= 468q^4(r-1)^2 - 18q^3(49 - 100r + 51r^2) + 3q^2(158 - 332r + 175r^2) - 3q(20 - 44r + 25r^2)\,, \\ Q_{2,0} &= (q-1)\left[468q^3(r-1)^2 - 12q^2(71 - 145r + 74r^2) + q(436 - 916r + 483r^2) - 51 + 112r - 63r^2\right]\,. \end{aligned} \tag{B.2}$$

C Appendix C

Let us denote $\mathbf{H}$ the bordering vector :

$$\begin{aligned}\mathbf{H}^* &= \left(\frac{-1}{\sigma_1}\sum_k m_{10}(k)\,\mathbf{G}_k^*, \frac{1}{\sigma_2}\sum_k m_{01}(k)\,\mathbf{G}_k^*\right) = (\mathbf{m}\otimes\mathbf{G})^*\,,\\ &\text{where :}\\ \mathbf{G} &= \sum_k 2d_k\,\mathbf{G}_k \quad ,(r=1) \quad \mathbf{m} = \left(\frac{q}{\sigma_1}, \frac{(1-q)}{\sigma_2}\right)^*\,, \end{aligned} \tag{C.1}$$

and $\text{FIM}_{c,q}^{-1}$, the 6×6 block of FIM^{-1} relative to kinematic parameters. We consider again the case $r=1$. Using the partitioned matrix inversion and the Woodbury lemmas [21], we obtain :

$$\begin{aligned}\text{FIM}_{c,q}^{-1} &= \left(\text{FIM}_c - \beta^{-1}\,\mathbf{H}\mathbf{H}^*\right)^{-1}\,,\\ &= \text{FIM}_c^{-1} + \frac{\beta^{-1}}{1-\beta^{-1}\mathbf{H}^*\text{FIM}_c^{-1}\mathbf{H}}\,\text{FIM}_c^{-1}\mathbf{H}\mathbf{H}^*\text{FIM}_c^{-1}\,. \end{aligned} \tag{C.2}$$

We have now to deal with the calculation of the corrective terms $\mathbf{H}^*\text{FIM}_c^{-1}\mathbf{H}$ and $\text{FIM}_c^{-1}\mathbf{H}\mathbf{H}^*\text{FIM}_c^{-1}$.

- Calculation of $\mathbf{H}^*\text{FIM}_c^{-1}\mathbf{H}$ and $\text{FIM}_c^{-1}\mathbf{H}\mathbf{H}^*\text{FIM}_c^{-1}$:

Using Prop. 2, we obtain ($\mathbf{m} \triangleq (\frac{-1}{\sigma_1}m_{10}, \frac{1}{\sigma_2}m_{01})^*$ $\mathcal{P} \triangleq \left(\mathcal{C}_0^{-1} + \alpha\,\mathcal{C}^{-1}\right)^{-1}$) :

$$\text{FIM}_c^{-1}\mathbf{H} = -\left(M_1^{-1}\otimes\mathcal{C}^{-1}\right)(\mathbf{m}\otimes\mathbf{G}) - \left(M_1^{-1}\mathbf{V}_0\otimes\mathcal{C}^{-1}\right)\mathcal{P}\left(\mathbf{V}_0^*M_1^{-1}\otimes\mathcal{C}^{-1}\right)(\mathbf{m}\otimes\mathbf{G})\,,$$

so, that :

$$\mathbf{H}^*\text{FIM}_c^{-1}\mathbf{H} = -(\mathbf{m}^*M_1^{-1}\mathbf{m})\left(\mathbf{G}^*\mathcal{C}^{-1}\mathbf{G}\right) - \left(\mathbf{V}_0^*M_1^{-1}\mathbf{m}\right)^2(\mathcal{C}^{-1}\mathbf{G})^*\mathcal{P}(\mathcal{C}^{-1}\mathbf{G})\,. \tag{C.3}$$

In the case $r=1$, we have $\mathbf{m}=\mathbf{V}_0$, yielding ($\alpha = -\mathbf{V}_0^*M_1^{-1}\mathbf{V}_0$) :

$$\begin{aligned}\mathbf{H}^*\text{FIM}_c^{-1}\mathbf{H} &= \alpha\,\mathbf{G}^*\left(\mathcal{C}^{-1} - \alpha\,\mathcal{C}^{-1}\mathcal{P}\mathcal{C}^{-1}\right)\mathbf{G}\,,\\ &\approx 0\,. \end{aligned} \tag{C.4}$$

Similarly, we have :

$$\text{FIM}_c^{-1}\mathbf{H}\mathbf{H}^*\text{FIM}_c^{-1} \approx 0\,,$$

so that, finally :

$$\text{FIM}_{c,q}^{-1} \approx \text{FIM}_c^{-1}\,. \tag{C.5}$$

References

[1] S.C. NARDONE, A.G. LINDGREN and K.F. GONG, Fundamental Properties and Performance of Conventional Bearings-Only Target Motion Analysis. IEEE *Trans. on* AUT. CONTROL, vol. AC–29, no 9, September 1984, pp. 775–787.

[2] J. ZIV and M.ZAKAI, Some Lower Bonds on Signal Parameter Estimation. IEEE *Trans. on* INFORMATION THEORY, vol. IT–15, no 3, May 1984, pp. 775–787.

[3] B.Z. BOBROVSKI, E. MAYER-WOLF and M. ZAKAI, Some Classes of Global Cramér-Rao Bounds. The Annals of Statistics, vol. 16, no. 4, pp. 1421–1438, 1987.

[4] M.L. GRAHAM and R.L. STREIT, The Cramér-Rao Bounds for Multiple Target Tracking Algorithms. NUWC-NPT Technical Report 10,406, Naval Undersea Warfare Center, Newport, RI, Sept. 1994.

[5] L. PERLOVSKI, Cramér-Rao Bounds for the Estimation of Normal Mixtures. Pattern Recognition Letters 10, pp. 141–148, Sept. 1989.

[6] L. PERLOVSKI, Cramér-Rao Bounds for Tracking in Clutter and Tracking Multiple Objects. Pattern Recognition Letters 18, pp. 283–288, 1997.

[7] R.J. FITZGERALD, Track Biases and Coalescence with Probabilistic Data Association. IEEE *Trans. on* AEROSPACE AND ELECTR, SYSTEMS, vol. AES–21, no. 6, Nov. 1985, pp. 822–824.

[8] K.C. CHANG, S MORI and C.Y. CHONG, Performance Evaluation of Track Initiation in Dense Target Environments. IEEE *Trans. on* AEROSPACE AND ELECTR, SYSTEMS, vol. AES–30, no. 1, Jan. 1994, pp. 213–218.

[9] S MORI, K.C. CHANG and C.Y. CHONG, Performance Analysis of Optimal data Association with Applications to Multiple Target Tracking. In *Multitarget-Multisensor Tracking: Applications and advances*, vol. II, Boston, MA, Artech House, 1992, ch. 7.

[10] F.E. DAUM, Bounds on Performance for Multiple Target Tracking. IEEE *Trans. on* AUT. CONTROL, vol. AC–35, no. 4, April 1990, pp. 833–839.

[11] D.J. SALMOND, Mixture Reduction for Target Tracking in Clutter. Proc. Internat. Symp. Signal and Data Processsing of Small Targets , SPIE 1305, pp. 434–445, 1990.

[12] L.Y. PAO, Multisensor Multitarget Mixture Reduction Algorithms for Tracking, Journal of Guidance, Control and Dynamics, vol. 17, no. 6, pp. 1205–1211, Nov.-Dec. 1994.

[13] R.L. STREIT and T.E. LUGINBUHL, A Probabilistic Multi-Hypothesis Tracking Algorithm Without Enumeration and Pruning. In Proceedings od the Sixth Joint Service Data Fusion Symposium, Laurel, MD, June 14–18, 1993, pp. 1015–1024.

[14] E. GIANNOPOULOS, R.L. STREIT and P. SWASZECK, Probabilistic Multi-Hypothesis Tracking in a Multi-Sensor Multi-Target Environmemt. In Proceedings of the first Australian Data Fusion Symposium, Adelaide, Australia, 21–22 Nov. 1996, pp. 184–189.

[15] K. KASTELLA, A maximum Likelihood Estimator for Report-to-Track Association. Proc. Internat. Symp. Signal and Data Processsing of Small Targets , SPIE 1954, pp. 386–393, 1993.

[16] H. GAUVRIT, Extraction multi-pistes : approche probabiliste et approche combinatoire. Thèse de doctorat de l'université de Rennes 1, Nov. 1997.

[17] J. BEHBOODIAN, Information Matrix for a Mixture of Two Normal Distributions. Journal of Statistical Computation and Simulation, vol. 1, pp. 295–314, 1972.

[18] B.M. HILL, Information for Estimating the Proportions in Mixture of Exponential and Normal Distributions. Jour. Amer. Stat. Assoc. 58, pp. 918–932, 1963.

[19] V.J. AIDALA and S.E. HAMMEL, Utilization of Modified Polar Coordinates for Bearings-Only Tracking. IEEE *Trans. on* AUTOMATIC CONTROL, vol. AC–28, no. 3, March 1983, pp. 283–294.

[20] J.P. LE CADRE, On the Properties of estimability Criteria for Target Motion Analysis. IEE *Proc.* RADAR, SONAR, NAVIGATION, vol. 145, no. 2, April 1998, pp. 92–100.

[21] H.V. HENDERSON and S.R. SEARLE, On Deriving the Inverse of a Sum of Matrices. SIAM Revieew, vol. 33, no.1, Jan. 1981, pp. 53–59.

[22] P. LANCASTER and M. TISMENETSKY, *The Theory of Matrices* , 2^{-nd} edition, Academic Press, San Diego, 1985.

[23] D.S.G POLLOCK, Tensor Products and Matrix Differential Calculus. Linear Algebra and its Applications, vol. 67, 1985, pp. 169–193.

[24] O. TRÉMOIS and J.P. LE CADRE, Target Motion Analysis with Multiple Arrays: Performance Analysis. IEEE *Trans. on* AEROSPACE AND ELECT. SYSTEMS, vol. AES–32, no 3, July 1996, pp. 1030–1045.

Tracking A General, Frequency Modulated Signal In Noise *

Tod Luginbuhl
Naval Undersea Warfare Center
Newport, RI 02841-1708 USA
t.e.luginbuhl@ieee.org (EMAIL)

Peter Willett
University of Connecticut
Storrs, CT 06269 USA
willett@mailsrv.engr.uconn.edu (EMAIL)

Abstract

A general, frequency modulated (GFM) signal characterizes the vibrations produced by compressors, turbines, propellers, gears and other rotating machines in a dynamic environment. A GFM signal is defined as the composition of a real or complex, periodic or almost periodic function (the carrier) with a real, differentiable function (the modulation). This paper develops a frequency domain tracking algorithm for a GFM signal in noise using the expectation-maximization (EM) algorithm. The primary advantage of this approach is that the ratios (harmonic numbers) of the carrier function do not need to be known a priori. The tracking algorithm exploits knowledge of the noise spectrum so that a separate normalization procedure is not required. The noise spectrum is incorporated into the tracking algorithm in essentially the same way that a clutter or noise model is incorporated into the probabilistic multi-hypothesis tracking algorithm (PMHT). Consequently, the GFM signal tracking algorithm presented in this

*This work was sponsored by ONR and NUWC.

paper is a PMHT-style algorithm. A simulated example is presented to illustrate the algorithm's performance.

1 Introduction

A GFM signal is a frequency modulated (FM) periodic or almost periodic signal. If the carrier signal is only a single sinusoid, then the GFM signal is also a FM signal. Periodic or harmonic signals and almost period signals are important in many areas. In the remainder of this paper, the term periodic or harmonic signal is for both periodic and almost periodic signals. Measurement and analysis of periodic vibrations is a critical part of machine monitoring. This is particularly true of large, high speed compressors and turbines where bearing or coupling failure can result in catastrophic damage to equipment[8]. These signals are very useful for monitoring marine mammals and shipping traffic. Minimizing and suppressing harmonic signals (harmonic noise) is an important part of alternating current electrical power transmission and distribution. Harmonic signals also play a critical role in music reproduction and synthesis and in speech recognition and synthesis.

To analyze harmonic signals, it is necessary to measure the amplitude, phase and frequency of the signal's (approximate) Fourier series components. If the signal is truly periodic, then it is only necessary to estimate its fundamental frequency (one over the period), since all other component frequencies are integer multiples (harmonic numbers) of the fundamental frequency. Unfortunately, sensor bandwidth may make it impossible to observe the fundamental frequency of a harmonic signal. This is usually not an issue when using accelerometers to monitor machines because accelerometers can measure vibrations down to zero frequency [8]. However, sensor bandwidth is a necessary reality in some applications. When the fundamental frequency of a periodic signal can not be observed, it is necessary to measure the frequencies of the observed Fourier components because one can no longer find a unique set of integer multiples and an unobserved fundamental frequency to fit the data. The discussion in this paper assumes that that the relationships between the frequencies (harmonic numbers) comprising the signal are not known.

In many applications, the periodic signal of interest is also time varying. Hence, the parameters of signal of interest must be tracked through time as well. In this paper only the frequencies of the Fourier components are

assumed to vary with time, and the frequencies are assumed to vary in a FM manner. For machine analysis problems, a phase referencing sensor can be used to monitor the actual speed of rotation. While such information could be exploited by the methods developed in this paper, it is assumed that this kind of reference signal is not available. The approach taken is to estimate the center or average frequency of each Fourier series component and estimate the modulation separately. It will be shown in a later section that the modulation is common to all of the Fourier series components; hence information from each Fourier series component is used to estimate the modulation.

2 Previous Work

The problem of estimating the parameters of a periodic signal is not new to the literature. The methods developed by other authors fall into three categories: time domain methods, frequency domain methods and high order spectral techniques. Time domain methods are the most popular. The time domain methods break down into two groups. References [7], [18] and [4] assume that the signal is not time varying, while references [9] and [11] assume that the signal is time varying. The frequency domain approaches described in references [10] and [5] assume that the frequencies comprising the signal are time varying. The high order spectral techniques discussed in references [14], [1] and [20] are really designed to detect phase or frequency dependence between pairs of frequencies, but these methods can be used to estimate parameters as well. The high order spectral techniques assume the fequencies are not time varying.

With the exception of the high order spectral techniques, a major draw back to all the algorithms discussed in the previous paragraph is the algorithms need to know the relationships between the frequencies (i.e. the harmonic numbers). The high order spectral techniques are extremely sensitive to time variation which severly limits there utility. The objective of this paper is to develop an estimator that does not need to know the relationships between the frequency components of a periodic signal and that can track time variations in the frequencies. In the next section, a model of time varying periodic signals is presented that is a natural extension of FM. This model allows one estimate the parameters of a time varying periodic signal with out needing to know the relationship between the frequencies.

3 General, FM Processes

A natural way to generalize FM is to allow the carrier signal to be any periodic or almost periodic function. A further generalization of FM comes from thinking of the GFM signal as the composition of the carrier function and the modulation function. Hence, if $s(t)$ is the GFM signal produced by modulating the periodic function $h(t)$ with the message $m(t)$, then $s(t) = h(m(t))$. As a practical matter, the modulating signal can not be arbitrary. Since this is a generalization of FM, the modulating signal must be differentiable. In addition, the derivative of the modulation function with respect to time should be bounded above and bounded away from zero to insure the modulation function has finite bandwidth.

An example of a GFM is a harmonic series produced by some mechanical system:

$$h(t) = \sum_{l=1}^{L} A_l \exp\left[i2\pi f_l m(t) + \theta_l\right] \tag{1}$$

where $\{f_l\}$ are the frequencies of the harmonic signal, $\{\theta_l\}$ represents the initial phases and $m(t)$ is a modulation induced by loading changes on the mechanical system. If no integer relationship exists between the fequencies, then $h(t)$ is an almost periodic function. This expression suggests that it may be possible to estimate $\{f_l\}$ and $m(t)$ jointly. If this can be done, then it would not be necessary to know the relationship between the frequencies a priori. The frequency domain is a logical place to attempt this joint estimation problem provided tractable expressions can be obtained.

In many cases of practical interest, the modulation signal varies at a much slower rate than one over the (approximate) period of the carrier function. Hence, the modulation function will be approximately linear during one or more periods of the carrier. When this is true, approximating the modulation function with a continuous, piecewise linear function greatly simplifies analysis of a general, FM signal. In order to develop a piecewise linear approximation to the modulation function $m(t)$, let $J = [0, T]$ be an interval on the real line contained in the domain of m and divide J into Υ consecutive intervals $J_n = [t_n, t_{n+1}]$ for $n = 0, \ldots, \Upsilon - 1$. A continuous, piecewise linear approximation to $m(t)$ on J is given by

$$\tilde{m}_\Upsilon(t) = \sum_{n=0}^{\Upsilon-1} \left[\alpha(t_n)t + \beta(t_n)\right] r\left(\frac{t - t_n}{t_{n+1} - t_n}\right) \tag{2}$$

where the slope and intercept over J_n are given by

$$\alpha(t_n) = \frac{m(t_{n+1}) - m(t_n)}{t_{n+1} - t_n}, \tag{3}$$

and

$$\beta(t_n) = m(t_n) - \alpha(t_n)t_n \tag{4}$$

respectively. The function $r(t)$ is the "rectangular window" and is defined as

$$r(t) = \begin{cases} 1, & 0 \le t \le 1 \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The modulation signal can be approximated by $\tilde{m}_\Upsilon(t)$ to any degree of accuracy by making Υ sufficiently large. This approximation to the modulation function will be used to find an approximate expression for the discrete-time Fourier transform (DTFT) of a general, FM signal.

To develop an expression for the discrete-time Fourier transform (DTFT) of a GFM signal, assume that the modulation function is approximately linear over the intervals $\{[nKT_s, (n+1)KT_s]\}$ (i.e. the distance between the end points of each interval J_n is $t_{n+1} - t_n = KT_s$) where the integer $K > 1$. The sampling period T_s is choosen so that $s(t)$ can be perfectly reconstructed from the sequence of samples $\{s(t_n + jT_s)\}$. The DTFT of s over J_n is defined as

$$S_n(f) = \sum_{j=0}^{K-1} \sum_{l=1}^{L} A_l e^{i\phi_{ln}} \exp\left[i2\pi f_l \alpha(t_n)(t_n + jT_s)\right] \exp\left[-i2\pi jf\right]. \tag{6}$$

Using the identity for the finite sum over a geometric series and simplifying the expression yields

$$S_n(f) = \sum_{l=1}^{L} A_l \frac{\sin\left[K\pi(f_l T_s \alpha(t_n) - f)\right]}{\sin\left[\pi(f_l T_s \alpha(t_n) - f)\right]} e^{i\psi_{lk}(f)} \tag{7}$$

where

$$\psi_{lk}(f) = \theta_l + 2\pi f_l m(t_n) + \pi(K-1)(f_l T_s \alpha(t_n) - f). \tag{8}$$

The amplitude of $S_n(f)$ as a function of frequency depends on a set of constants $\{f_l\}$ and the slope of the modulation function at time t_n. If the frequencies $f_1, \ldots, f_L$ are well separated (i.e. more than $2/KT_s$ Hertz apart), then

$$|S_n(f)|^2 \approx \sum_{l=1}^{L} A_l^2 \frac{\sin^2\left[K\pi(f_l T_s \alpha(t_n) - k/K)\right]}{\sin^2\left[\pi(f_l T_s \alpha(t_n) - k/K)\right]} \tag{9}$$

provided the cross terms are negligible. Therefore, the magnitude squared of $S_n(f)$ consists of L trajectories where each trajectory is centered on a frequency f_l, and the path of the all of the trajectories is given by the sequence of slopes of the modulation function. This is a multitarget tracking problem where all the tracks have a common prior or model. An expectation-maximization (EM) estimation algorithm [2] inspired by the probabilistic multi-hypothesis tracking algorithm described in [16] will be developed in the remainder of this paper. Defining a statistical model for this estimation problem is the first step in this development.

4 The Statistical Model

Suppose a GFM signal $s(t)$ is contaminated with an independent, complex finite bandwidth, additive, zero mean noise $\eta(t)$ to form the observed signal $r(t)$. $r(t)$ is then sampled at a sufficiently high rate $1/T_s$ to be perfectly reconstructed. The observed sequence $\{r(t_n+jT_s)\}$ is transformed to the frequency domain by one or more K length discrete Fourier transforms (DFTs). The sequence of Υ DFTs partitions the sequence of samples into intervals $\{J_n\}$. The modulation $m(t)$ of the GFM from interval J_n to interval J_{n+1} is modelled as the output of a discrete time linear system driven by a deterministic control signal and a white Gaussian process noise vector.

The linear system producing $m(t_n)$ is defined as

$$a(t_{n+1}) = Aa(t_n) + b(t_n) + q(t_n) \tag{10}$$

and

$$m(t_n) = Ca(t_n) \tag{11}$$

where $a(t_n)$ is the real, M dimensional state variable, A is the state feedback matrix, $b(t_n)$ is the deterministic control vector, and $q(t_n)$ is the process noise. $q(t_n)$ is assumed to be a zero mean Gaussian vector with covariance Q such that $q(t_n)$ is statistically independent of $q(t_m)$ for $n \neq m$. For notational purposes, let $\mathcal{A}$ represent the sequence of state variables for the modulation process: $\mathcal{A} = \{a(t_n)\}_{n=0}^{\Upsilon-1}$. In the previous section it was shown that the slope of $m(t)$ appears in the expression for the DFT of a GFM when its modulation is approximately linear during the DFT interval. Let $\alpha(t_n)$ denote the slope of the modulation $m(t)$ over the DFT interval J_n; then,

$$\alpha(t_n) = \frac{m(t_{n+1}) - m(t_n)}{KT_s} = \frac{C}{KT_s}(a(t_{n+1}) - a(t_n)). \tag{12}$$

Since the linear state space model for the modulation is a Gauss-Markov process, the PDF of $a(t_{n+1})$ is

$$p(a(t_{n+1})|a(t_n)) = \frac{1}{(2\pi)^{M/2}|Q|^{1/2}} \exp\left[-\frac{1}{2}q^t(t_n)Q^{-1}q(t_n)\right] \tag{13}$$

because

$$q(t_n) = a(t_{n+1}) - Aa(t_n) - b(t_n). \tag{14}$$

The prior PDF of a_0 is given by

$$p(a(t_0)) = \frac{1}{(2\pi)^{M/2}|Q_0|^{1/2}} \exp\left[-\frac{1}{2}(a(t_0) - \bar{a})^t Q_0^{-1}(a(t_0) - \bar{a})\right]. \tag{15}$$

Hence, the PDF of $\mathcal{A}$ is equal to

$$P(\mathcal{A}) = p(a(t_0)) \prod_{n=0}^{\Upsilon-1} p(a(t_{n+1})|a(t_n)). \tag{16}$$

To develop an approximation to the spectrum corresponding to the time interval J_n, some notation for the DFT bins needs to be developed. Let $I = [0, 2\pi]$, and let $\{I_k\}$ partition I into K subintervals (these are the DFT bins) where $I_k = [d_k, d_{k+1}]$ and $d_k = 2\pi(k - 1/2)/K$ for $k = 0, \ldots, K-1$. The sequence of DFT bin endpoints are represented by $\mathcal{D} = \{d_k\}$, and the centers of the DFT bins are represented the sequence $\tilde{\mathcal{D}} = \{\tilde{d}_k\}$.

Let

$$\mathcal{S}_{rn} = \int_0^\pi \mathcal{S}_{rn}(\omega)\, d\omega \tag{17}$$

denote the total power in the power spectrum, $\mathcal{S}_{rn}(\omega)$, of $\{r(t_n + jT_s)\}$ during J_n, and let $\tilde{\mathcal{S}}_{rn}(\omega) = \mathcal{S}_{rn}(\omega)/\mathcal{S}_{rn}$ represent the normalized power spectrum. Then,

$$\tilde{\mathcal{S}}_{rn}(\omega) = \frac{\mathcal{S}_{rn}(\omega)}{\mathcal{S}_{rn}} = \frac{\mathcal{S}_{sn}(\omega) + \mathcal{S}_{\eta n}(\omega)}{\mathcal{S}_{sn} + \mathcal{S}_{\eta n}} \tag{18}$$

which is equivalent to $\tilde{\mathcal{S}}_{rn}(\omega) = \pi_s \tilde{\mathcal{S}}_{sn}(\omega) + \pi_\eta \tilde{\mathcal{S}}_{\eta n}(\omega)$ where $\pi_s = \mathcal{S}_{sn}/\mathcal{S}_{rn}$ and $\pi_\eta = \mathcal{S}_{\eta n}/\mathcal{S}_{rn}$ (assuming the noise and signal power are the same for all J_n). π_s and π_η are between zero and one and sum to one. Furthermore, the signal to noise ratio (SNR) is equal to π_s/π_η.

Since a normalized power spectrum is positive or zero on I and the total power of a normalized power spectrum equals 1, a normalized power spectrum is a PDF on I. Hence, $\tilde{\mathcal{S}}_{rn}(\omega)$ is a mixture PDF and may be approximated

by another mixture PDF on I [17]. The normalized power spectrum of the observe sequence over J_n may be approximated as $\tilde{\mathcal{S}}_{rn}(\omega) \approx \pi_s p(\omega|\alpha(t_n);\lambda) + \pi_\eta \tilde{\mathcal{S}}_\eta$ where

$$p(\omega|\alpha(t_n);\lambda) = \sum_{l=1}^{L} \pi_l p_l(\omega|\alpha(t_n);\lambda_l) \approx \tilde{\mathcal{S}}_{sn}(\omega) \tag{19}$$

is a mixture PDF approximating the unknown normalized GFM signal power spectrum. Observe that number of PDFs in the mixture PDF $p(\omega|\alpha(t_n);\lambda)$ need not equal the number of frequency lines in $\mathcal{S}_s(\omega)$. The number of PDFs in $p(\omega|\alpha(t_n);\lambda)$ only equals the number of frequency lines when the frequencies are widely separated, and when this is true, $\pi_l \approx A_l^2/\mathcal{S}_s$. The unknown parameters in this equation are the parameters contained in the set $\lambda = \{\pi_l, \lambda_l\}$. If the SNR is known then the total number of mixing proportions (component probabilities) is reduce by one. If the SNR is not known, then the signal to noise ratio can be estimated along with the normalized power spectrum of the signal. It is also possible to estimate the normalize noise power spectrum as well, but this is not pursued here. With these concepts in mind, a measurement sequence and a measurement PDF can be developed.

Let $\mathcal{R} = \{\mathcal{R}_n\}_{n=0}^{\Upsilon-1}$ represent the sequence of magnitude squared DFTs of length K obtained $r(t)$. Given the modulation sequence, the magnitude squared DFT $\mathcal{R}_n = \{R_{kn}\}$ at time t_n is conditionally independent of $\mathcal{R}_{n'}$ for all $n \neq n'$. Conceptually, each R_{kn} is equivalent to the sum of a set of independent random samples drawn from the part of $\tilde{\mathcal{S}}_{rn}(\omega)$ contained in interval I_k. If R_{kn} is quantized to N_{kn} (mapped to the positive integers), then this is approximately the same as observing a total of N_{kn} samples in I_k from $\tilde{\mathcal{S}}_{rn}(\omega)$. Related but different interpretations of the DFT of a signal appear in references [15] and [12].

To formalize the quantization of R_{kn} into N_{kn}, let $N_{kn} = \lfloor N_{max} R_{kn}/R_{max} \rfloor$ where N_{max} is some large integer and $R_{max} = \max_n \max_k R_{kn}$. For notational convienence, define $N_n = \sum_k N_{kn}$ and $N = \sum_n N_n$. The quantization of $\mathcal{R}$ into $\mathcal{N}$ is performed over the entire block of DFT data to retain relative scaling information, and N_{max} is choosen so that $N_{kn}/N_n \approx R_{kn}/R_n$. Given the sequence $\mathcal{A}$, the sequences $\mathcal{N}_n$ and $\mathcal{N}_{n'}$ are conditionally independent for $n \neq n'$. The PDF of $\mathcal{N}_n$ given $\mathcal{A}$ is the multinomial PDF

$$P(\mathcal{N}_n|\alpha(t_n));\lambda) = \left(\begin{array}{c} N_n \\ N_{0n} \ldots N_{K-1n} \end{array} \right) \prod_{k=1}^{B} P_{kn}(\lambda)^{N_{kn}} \tag{20}$$

where

$$P_{kn}(\lambda) = \int_{I_k} \tilde{S}_{rn}(\omega)\, d\omega. \tag{21}$$

and $\sum_k P_{kn} = 1$. The PDF of $\mathcal{N}$ conditioned on $\mathcal{A}$, is then

$$P(\mathcal{N}|\mathcal{A};\lambda) = \prod_{n=1}^{\Upsilon-1} P(\mathcal{N}_n|\alpha(t_n);\lambda). \tag{22}$$

The joint PDF of $\mathcal{N}$ and $\mathcal{A}$ is $P(\mathcal{N},\mathcal{A};\lambda) = P(\mathcal{N}|\mathcal{A};\lambda)P(\mathcal{A})$. This joint PDF represents the PDF of the observed, quantized DFT data.

Since,

$$P_{kn}(\lambda) \approx \sum_{l=1}^{L} \pi_l \int_{I_k} p_l(\omega|\alpha(t_n);\lambda_l)\, d\omega + \pi_\eta \int_{I_k} \tilde{S}_\eta(\omega)\, d\omega, \tag{23}$$

the parameter set $\lambda = \{\pi_\eta, \{\pi_l, \lambda_l\}\}$. In order to define the parameters λ_l, it is necessary to choose component PDFs for the mixture PDF $p(\omega|\alpha(t_n);\lambda)$. Let

$$p_l(\omega|\alpha(t_n);\lambda_l) = \frac{1}{\sigma_l\sqrt{2\pi}} \exp[-\frac{1}{2\sigma_l^2}(\omega - \alpha(t_n)\mu_l)^2]. \tag{24}$$

Then the parameter set $\lambda_l = \{\mu_l, \sigma_l^2\}$. for $l = 1, \ldots, L$.

5 The EM Method

To develop an EM algorithm, the complete and missing information must be identified and characterized. The complete data for the EM algorithm consists of continuous frequency samples and the indices of the mixture component PDF in $p(\omega|\alpha(t_n);\lambda)$ (in this discussion, it is convenient to think of $\tilde{S}_\eta(\omega)$ as component PDF $L+1$). Let Ω denote the set of continuous frequency samples. Ω consists of Υ sets of elements corresponding to each DFT: $\Omega = \{\Omega_n\}$. Given $\mathcal{A}$, Ω_n is conditionally independent of $\Omega_{n'}$ for all $n \neq n'$. The set of frequency measurements at each time t_n consists of the sets of samples in each of the K frequency bins: $\Omega_n = \{\{\omega_{jkn}\}\}$ where ω_{jkn} represents sample j from DFT bin k at time t_n. For each t_n, ω_{jkn} is independent of $\omega_{j'k'n}$ for $j \neq j'$ and $k \neq k'$, and the PDF of ω_{jkn} is $p(\omega|\alpha(t_n);\lambda)$. For each sample $\omega_{jkn} \in \Omega$, the corresponding index i_{jkn} of the component PDF in $p(\omega|\alpha(t_n);\lambda)$ is unknown. Let $\mathcal{I}$ denote this set of component indices, then

$\mathcal{I} = \{\mathcal{I}_n\}$ where $\mathcal{I}_n = i_{jkn}$. Since each i_{jkn} is additional information about each ω_{jkn}, $\mathcal{I}$ has the same statistical properties as Ω.

The complete data $\mathcal{Z}$ is equal to the union of Ω and $\mathcal{I}$. Therefore, $\mathcal{Z}$ has the same statistical properties as Ω and $\mathcal{I}$. The PDF of $\mathcal{Z}$ conditioned on $\mathcal{A}$ is given by

$$P(\mathcal{Z}|\mathcal{A};\lambda) = \prod_{n=0}^{\Upsilon-1}\left[\binom{N_n}{N_{01}\cdots N_{K-1n}}\prod_{k=0}^{K-1}\prod_{j=0}^{N_{kn}}\pi_l p_l(\omega_{jkn}|\alpha(t_n);\lambda_l)|_{l=i_{jkn}}\right] \quad (25)$$

where $\pi_{L+1} = \pi_\eta$, and $p_{L+1}(\omega_{jkn}|\alpha(t_n);\lambda_{L+1}) = \tilde{\mathcal{S}}_\eta(\omega_{jkn})$. If $\tilde{\mathcal{S}}_\eta(\omega)$ is known, then λ_{L+1} is empty. The joint PDF of $\mathcal{Z}$ and $\mathcal{A}$ is given by $P(\mathcal{Z},\mathcal{A};\lambda) = P(\mathcal{Z}|\mathcal{A};\lambda)P(\mathcal{A})$. To find the search function $\mathcal{Q}$ (usually called the auxiliary function in the literature) of the EM algorithm, the PDF of the missing information conditioned on the observed data must be derived.

The missing information in this problem is contained in the two sets Ω and $\mathcal{I}$. Ω is partially observed because $\mathcal{N}$ is known. The PDF of the missing data conditioned on the observed data is

$$P(\Omega\bigcup\mathcal{I}|\mathcal{N};\lambda) = \frac{P(\mathcal{Z},\mathcal{A};\lambda)}{P(\mathcal{N},\mathcal{A};\lambda)} \quad (26)$$

$$= \prod_{n=0}^{\Upsilon-1}\prod_{k=0}^{K-1}\prod_{j=0}^{N_{kn}}\left.\frac{\pi_l p_l(\omega_{jkn}|\alpha(t_n);\lambda_l)}{P_{kn}(\lambda)}\right|_{l=i_{jkn}}. \quad (27)$$

The search function can now be obtained from the PDFs $P(\mathcal{Z},\mathcal{A};\lambda)$ and $P(\Omega\bigcup\mathcal{I}|\mathcal{N};\lambda)$.

The search function of the EM algorithm is defined as

$$\mathcal{Q}(\lambda;\lambda') = \mathrm{E}\left\{\log[P(\mathcal{Z},\mathcal{A};\lambda)]P(\Omega\bigcup\mathcal{I}|\mathcal{N};\lambda')\right\} \quad (28)$$

where the expectation is taken over the missing information Ω and $\mathcal{I}$. Evaluating the expectation and simplifying the result yields

$$\mathcal{Q}(\lambda;\lambda') = \mathcal{Q}_0 + \mathcal{Q}(a(t_0)) + \sum_{n=0}^{\Upsilon-1}\left[\mathcal{Q}_n(\mathcal{A}) + \sum_{l=1}^{L+1}(\mathcal{Q}_{ln}(\pi_l) + \mathcal{Q}_{ln}(\lambda_l))\right] \quad (29)$$

where

$$\mathcal{Q}_0 = -\frac{1}{2}(\Upsilon+1)M\log[2\pi] + \sum_{n=0}^{\Upsilon-1}\log\binom{N_n}{N_{0n}\cdots N_{K-1n}}, \quad (30)$$

$$\mathcal{Q}(a(t_0)) = -\frac{1}{2}\log|Q_0| - \frac{1}{2}(a(t_0)-\bar{a})^t Q_0^{-1}(a(t_0)-\bar{a}), \quad (31)$$

$$\mathcal{Q}_n(\mathcal{A}) = -\frac{1}{2}\log|Q| - \frac{1}{2}q^t(t_n)Q^{-1}q(t_n), \quad (32)$$

$$q(t_n) = a(t_{n+1}) - Aa(t_n) - b(t_n), \quad (33)$$

$$Q_{ln}(\pi_l) = \sum_{k=0}^{K-1}\log(\pi_l)\frac{N_{kn}\pi_l'}{P_{kn}(\lambda')}\int_{I_k} p_l(\omega|\alpha'(t_n);\lambda_l')\,d\omega \quad (34)$$

and

$$\mathcal{Q}_{ln}(\lambda_l) = \sum_{k=0}^{K-1}\frac{N_{kn}\pi_l'}{P_{kn}(\lambda')}\int_{I_k}\log[p_l(\omega|\alpha(t_n);\lambda_l)]p_l(\omega|\alpha'(t_n);\lambda_l')\,d\omega. \quad (35)$$

The complete details of this derivation of the search function appear in [6].

The maximization step (M-step) of the EM algorithm maximizes the search function with respect to the parameters λ and the modulation state vector sequence $\mathcal{A}$. The derivation of the M-step is not given in this paper due to space limitations. The derivation is available in [6]. Instead, the resulting iterative estimation algorithm is summarized in the next section.

6 Estimation Algorithm Summary

Let $\lambda^{(i)}$ denote the parameter estimates after i iterations have been completed. The parameter estimates for iteration $i+1$ are obtained from $\lambda^{(i)}$. The mixing proportions (or component probabilities) are updated first because they are independent of the other parameters. The mixing proportions are updated by

$$\pi_l^{(i+1)} = \frac{1}{N}\sum_{n=0}^{\Upsilon-1}\sum_{k=0}^{K-1}\frac{N_{kn}\pi_l^{(i)}}{P_{kn}(\lambda^{(i)})}\int_{I_k} p_l(\omega|\alpha'(t_n);\lambda_l^{(i)})\,d\omega. \quad (36)$$

Updating the remaining parameters is more complicated, because the update equations for $\{\lambda_l\}$ and $\mathcal{A}$ are coupled.

Since $\{\lambda_l\}$ and $\mathcal{A}$ can not be solved for independently, there are two ways to proceed. One could simply implement a generalized EM algorithm [2] [19] by updating $\{\lambda_l^{(i)}\}$ using $\mathcal{A}^{(i)}$, and then update $\mathcal{A}^{(i)}$ using $\{\lambda_l^{(i+1)}\}$, or one could maximize the search function using a numerical optimization procedure. The generalized EM method is presented first.

Specifically, new estimates of the component means are computed using

$$\mu_l^{(i+1)} = \frac{\sum_{n=0}^{\Upsilon-1} \alpha^{(i)}(t_n) \sum_{k=0}^{K-1} \frac{N_{kn}}{P_{kn}(\lambda^{(i)})} \int_{I_k} \omega p_l(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega}{\sum_{n=0}^{\Upsilon-1} (\alpha^{(i)}(t_n))^2 \sum_{k=0}^{K-1} \frac{N_{kn}}{P_{kn}(\lambda^{(i)})} \int_{I_k} p(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega}. \quad (37)$$

Then, the component variances are reestimated using

$$\sigma_l^{(i+1)} = \frac{\sum_{n=0}^{\Upsilon-1} \sum_{k=0}^{K-1} \frac{N_{kn}}{P_{kn}(\lambda^{(i)})} \int_{I_k} (\omega - \alpha^{(i)}(t_n)\mu_l^{(i+1)})^2 p_l(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega}{\sum_{n=0}^{\Upsilon-1} \sum_{k=0}^{K-1} \frac{N_{kn}}{P_{kn}(\lambda^{(i)})} \int_{I_k} p_l(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega}. \quad (38)$$

The estimate of the modulation state vector sequence is obtained by solving a block tridiagonal system of linear equations akin to the Kalman smoothing filter. This system of equations is converted to a block upper triangular system of equations using Gaussian elimination (forward recursion). The estimates for the state vectors in $\mathcal{A}$ are then calculated using back substitution (backward recursion).

The foward recursion begins by using the new estimates of $\{\mu_l^{(i)}\}$ and $\{\sigma_l^{(i)}\}$ to compute the "measurment normalization coefficients"

$$x^{(i+1)}(t_n) = \sum_{l=1}^{L} \left(\frac{\mu_l^{(i+1)}}{\sigma_l^{(i+1)} K T_s} \right)^2 \sum_{k=0}^{K-1} \frac{N_{kn} \pi_l^{(i)}}{P_{kn}(\lambda^{(i)})} \int_{I_k} p_l(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega \quad (39)$$

and the "measurements"

$$y^{(i+1)}(t_n) = \sum_{l=1}^{L} \frac{\mu_l^{(i+1)}}{(\sigma_l^{(i+1)})^2 K T_s} \sum_{k=0}^{K-1} \frac{N_{kn} \pi_l^{(i)}}{P_{kn}(\lambda^{(i)})} \int_{I_k} \omega p_l(\omega|\alpha^{(i)}(t_n); \lambda_l^{(i)})\, d\omega \quad (40)$$

for $n = 0, \ldots, \Upsilon - 1$. From these variables, the off diagonal matrices $X_n^{(i+1)}$, the diagonal matrices $S_n^{(i+1)}$, the control matrices $U_n^{(i+1)}$ and the "measurement" matrices $Y_n^{(i+1)}$ are calculated for all valid n. The matrices X_n are obtained using the formula

$$X_n^{(i+1)} = Q^{-1}A + x^{(i+1)}(t_n) C^t C \quad (41)$$

for $n = 0, \ldots, \Upsilon - 1$. The matrices $S_n^{(i+1)}$ are calculated from

$$S_n^{(i+1)} = \begin{cases} Q_0^{-1} + A^t Q^{-1} A + x^{(i+1)}(t_n) C^t C & n = 0, \\ Q^{-1} + A^t Q^{-1} A + x x C_n^{(i+1)} - x S x_{n-1}^{(i+1)} & 0 < n < \Upsilon, \\ Q^{-1} + x^{(i+1)}(t_{n-1}) C^t C - x S x_{n-1}^{(i+1)} & n = \Upsilon. \end{cases} \quad (42)$$

where $xxC_n^{(i+1)} = (x^{(i+1)}(t_{n-1}) + x^{(i+1)}(t_n))C^tC$ and

$$xSx_n^{(i+1)} = X_n^{(i+1)}(S_n^{(i+1)})^{-1}(X_n^{(i+1)})^t. \tag{43}$$

The matrices $U_n^{(i+1)}$ are calculated from

$$U_n^{(i+1)} = \begin{cases} Q_0^{-1}\bar{a} - A^tQ^{-1}b(t_n) & n = 0, \\ Q^{-1}b(t_{n-1}) - A^tQ^{-1}b(t_n) + xSu_{n-1}^{(i+1)} & 0 < n < \Upsilon, \\ Q^{-1}b(t_{n-1}) + xSu_{n-1}^{(i+1)} & n = \Upsilon \end{cases} \tag{44}$$

where $xSu_n^{(i+1)} = X_n^{(i+1)}(S_n^{(i+1)})^{-1}U_n^{(i+1)}$. The matrices $Y_n^{(i+1)}$ are calculated using

$$Y_n^{(i+1)} = \begin{cases} C^ty^{(i+1)}(t_n) & n = 0, \\ C^t(y^{(i+1)}(t_n) - y^{(i+1)}(t_{n-1})) - xSy_{n-1}^{(i+1)} & 0 < n < \Upsilon, \\ -C^ty^{((i+1)}t_{n-1}) - xSy_{n-1}^{(i+1)}n = \Upsilon \end{cases} \tag{45}$$

where $xSy_n^{(i+1)} = X_n^{(i+1)}(S_n^{(i+1)})^{-1}Y_n^{(i+1)}$. This completes the forward recursion.

The backward recursion begins by solving for

$$a^{(i+1)}(t_\Upsilon) = -(S_\Upsilon^{(i+1)})^{-1}(Y_\Upsilon^{(i+1)} - U_\Upsilon^{(i+1)}). \tag{46}$$

The backward recursion

$$\begin{aligned} &a^{(i+1)}(t_n) = \\ &\quad -(S_n^{(i+1)})^{-1}(Y_n^{(i+1)} - U_n^{(i+1)}) + (S_n^{(i+1)})^{-1}(X_n^{(i+1)})^t a^{(i+1)}(t_{n+1}). \end{aligned} \tag{47}$$

yields the estimates for the remaining elements in $\mathcal{A}^{(i+1)}$. This completes an M-step of the generalized EM algorithm.

Because each M-step of the generalized EM scheme increases the search function, $P(\mathcal{N}, \mathcal{A}; \lambda^{(i+1)}) \geq P(\mathcal{N}, \mathcal{A}; \lambda^{(i)})$. The generalized EM iteration is repeated until

$$\frac{\log[P(\mathcal{N}, \mathcal{A}; \lambda^{(i+1)})] - \log[P(\mathcal{N}, \mathcal{A}; \lambda^{(i)})]}{1 + \log[P(\mathcal{N}, \mathcal{A}; \lambda^{(i+1)})]} < \epsilon \tag{48}$$

for some desired $\epsilon > 0$. However, since $\mathcal{Q}(\lambda^{i+1}; \lambda^i)$ is not maximized during each iteration, the maximum increase possible in $P(\mathcal{N}, \mathcal{A}; \lambda^{(i+1)})$ for each iteration is not achieved. This leads to the second approach.

To attain the maximum increase in $P(\mathcal{N}, \mathcal{A}; \lambda^{(i+1)})$, one could attempt to maximize $\mathcal{Q}(\lambda^{(i+1)}; \lambda^{(i)})$ during each iteration of the EM algorithm. A straightforward method to maximize $\mathcal{Q}(\lambda^{(i+1)}; \lambda^{(i)})$ is to iterate over the update equations for $\{\mu_l\}, \{\sigma_l\}$ and $\mathcal{A}$ until

$$\frac{\mathcal{Q}(\lambda^{(j+1,i)}; \lambda^{(j,i)}) - \mathcal{Q}(\lambda^{(j,i)}; \lambda^{(j-1,i)})}{1 + \mathcal{Q}(\lambda^{(j+1,i)}; \lambda^{(j,i)})} < \epsilon' \tag{49}$$

where the superscript (j, i) represents the double iteration (j represents the inner iteration to maximize $\mathcal{Q}$ and i represents the EM iteration). $\epsilon' > 0$ is some desired tolerance.

This second method was implemented and used to generate the examples, and it seems to be efficient in practice. One drawback to this double iteration scheme appears as the algorithm nears convergence. The reestimation formulas for $\{\lambda_l\}$ and $\mathcal{A}$ must be executed twice, and $\mathcal{Q}$ must be calculated twice to establish that $\mathcal{Q}$ has been maximized. Standard numerical optimization methods like the ones described in [13] or [3] are another possible way to maximize $\mathcal{Q}$ with respect to $\{\lambda_l\}$ and $\mathcal{A}$ at each EM stage. Use of these numerical methods was not pursued in this paper.

7 An Example

To illustrate the performance of the algorithm, 8193 samples of the GFM signal $s(t) = A\sum_{l=1}^{5} \exp[i2\pi\sqrt{l} f_o m(t)]$ were added to 8193 sampes of a zero mean, unit variance, complex white Gaussian noise sequence. Observe that this GFM signal's carrier function is almost periodic. The modulation function was choosen as

$$m(t) = t + \frac{\Delta}{2\pi f_o} \int_0^t f(x)\, dx \tag{50}$$

where $f_o = 0.1$, $f(t)$ was the output of a second order autoregressive (AR) filter with poles at $0.99e^{\pm 2\pi/1000}$ driven by zero mean real white Gaussian noise with variance 2.5×10^{-9}, and the modulation index Δ was equal to 2. 10240 samples were collected from the AR filter, and the first 2047 samples were dropped to eliminate the transient response. The remaining 8193 samples were used to generate 8193 samples from $s(t)$. Figure 1 shows the signal spectrogram obtained using 64 nonoverlapping 128 point DFTs. The SNR was set to approximately $-7dB$ where the SNR was defined as $10\log[5A^2]$ (i.e. $A = 1/5$). The resulting signal plus noise sequence was broken up into 64

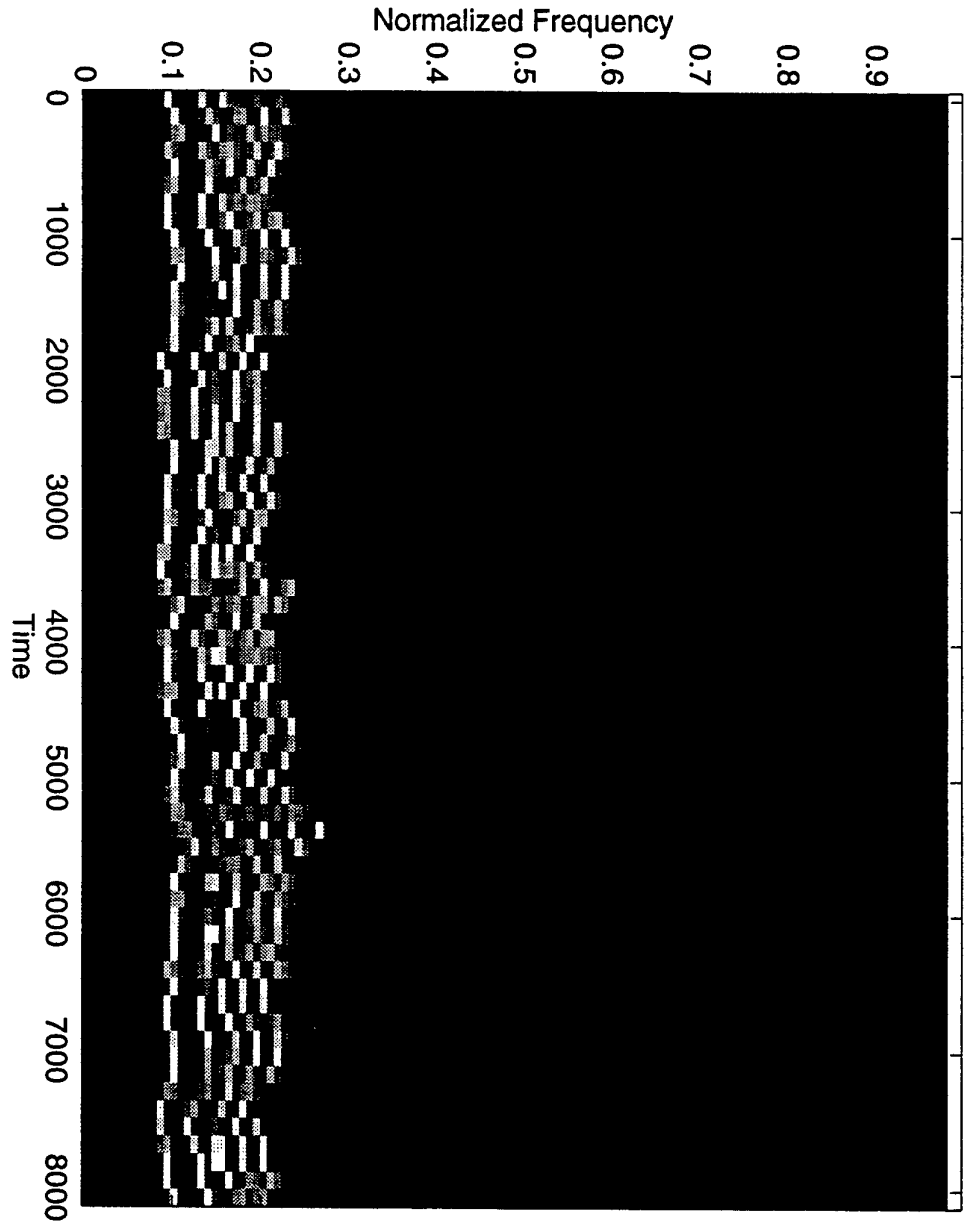


Figure 1: GFM signal spectrogram

consecutive nonoverlapping 128 point DFTs. Figure 2 shows the spectrogram of signal plus noise at $-7dB$ SNR. the variances of the

The EM estimation algorithm was initialized as follows. The mixing proportions of the mixture PDF $p(\omega|\alpha(t_n);\lambda_l)$ were set to $1/6$, and the means and variances of the five Gaussian component PDFs were set to $f_o\sqrt{l}$ and $(12 \times 128^2)^{-1}$ respectively. The state feedback matrix for the prior PDF of the modulation function was chosen as

$$A = \begin{bmatrix} 1 & 128 \\ 0 & 1 \end{bmatrix}, \tag{51}$$

and the control input sequence was set to $\{[128, 0]^t\}$. The covariance matrix of the process noise was

$$Q = \sigma_q^2 \begin{bmatrix} 128^3/3 & 128^2/2 \\ 128^2/2 & 128 \end{bmatrix} \tag{52}$$

where $\sigma_q^2 = 5.7220^{-6}$. The prior mean $\bar{a}$ was set equal to the zero vector, and the prior covariance matrix Q_o was set equal to Q. The initial estimate of the state vector was choosen as $a(t_n) = [m(t_n), 0]^t$. Lastly, the stopping tolerances ϵ and ϵ' were equal to 10^{-4}.

Figure 3 shows the spectrogram of estimated signal. Figure 3 is really a plot of the mixture PDF $p(2\pi f|\alpha(t_n);\lambda)$ where $f = k/4096$ for $k = 0, \ldots, 4095$ which includes the normalized noise power spectrum. The total and average mean square error between the modulation function and the estimate modulation function was 43.0952 and 0.6630 respectively. The total and average mean square error between the slope of the modulation function (over 128 point intervals) and the estimated slope was 0.1855 and 2.8995×10^{-3} respectively. The estimates of the frequencies of the GFM were 0.0996, 0.1408, 0.1733, 0.1989 and 0.2221. The amplitude estimates for each frequency of the GFM were 0.1806, 0.1953, 0.1847, 0.1766, 0.1891. The estimate of the signal to noise ratio was -7.6491 dB.

8 Summary

This paper described a new algorithm based on the EM method for estimating the parameters of a GFM signal in noise. The example demonstates that the algorithm seems to work quite well at low SNRs; however, multiple trials of the example need to be run to evaluate the algorithms performance.

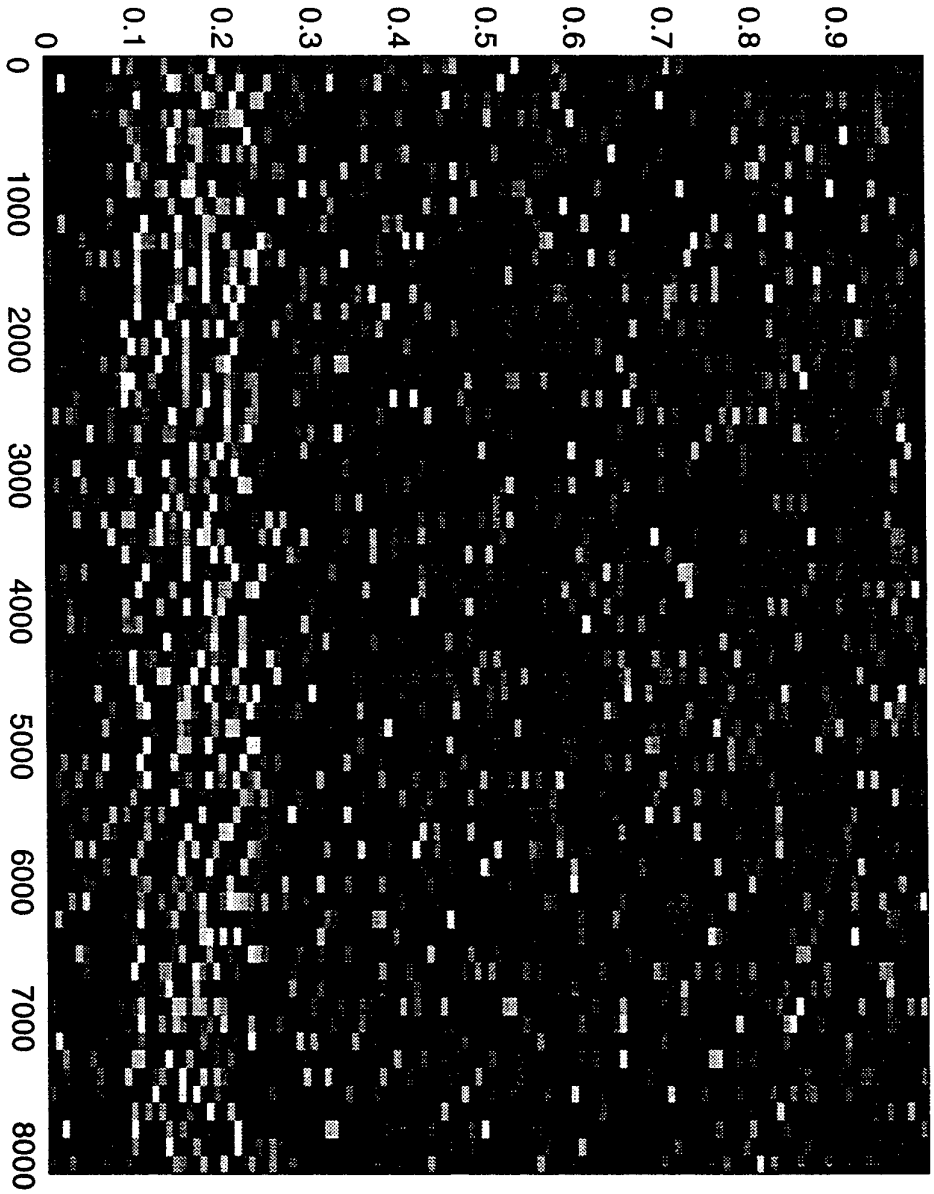


Figure 2: $-7dB$ GFM signal plus noise spectrogram

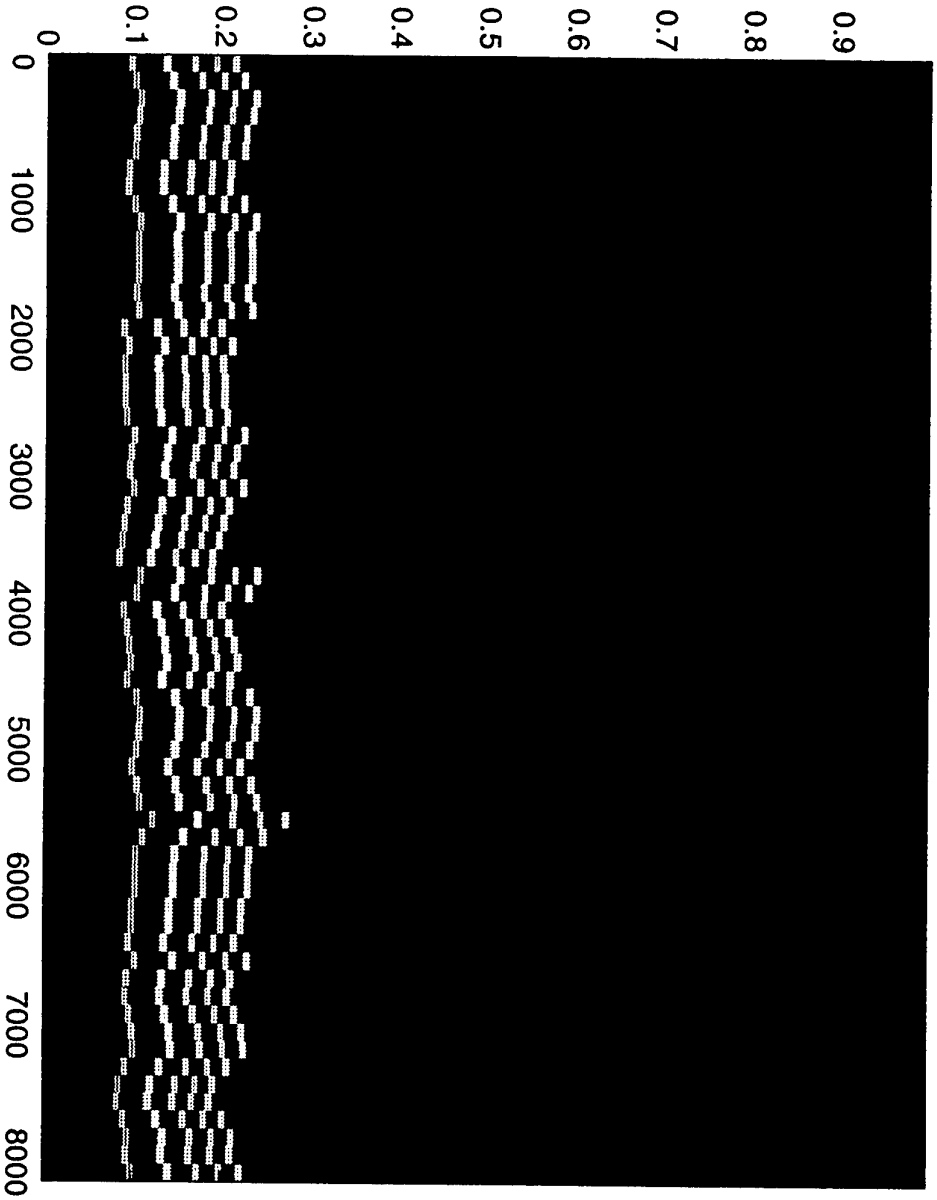


Figure 3: Estimated GFM signal spectrogram

References

[1] Kevin W. Baugh. Parametric phase coupling in the vibration spectra of rolling element bearings. Technical Report 92xxxx, Applied Research Laboratories, The University of Texas at Austin, 10000 Burnet Road, Austin, Texas 78713-8029, 1992.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society Series B*, 39(1):1–38, 1977. article includes discussion.

[3] Andrew Grace. *Optimization TOOLBOX For Use With MATLAB.* The Math Works, Incorporated, 24 Prime Park Way, Natick, Massachusetts, October 1994.

[4] Ben James, Brian D. O. Anderson, and Robert C. Williamson. Conditional mean and maximum likelihood approaches to multiharmonic frequency estimation. *IEEE Transactions on Signal Processing*, 42(6):1366–1375, June 1994.

[5] Tod E. Luginbuhl. Estimation of independent and proportional fm signals. Master's thesis, The University of Connecticut, 1991.

[6] Tod E. Luginbuhl. *Estimation of General, Discrete-Time FM Processes.* PhD thesis, University of Connecticut, 1998.

[7] D. R. A. McMahon and R. S. Barrett. ML estimation of the fundamental of a harmonic series. In *Proceedings of the IASTED International Symposium on Signal Processing and its Applications*, pages 333–336. IASTED, August 1987. University of Queensland.

[8] John S. Mitchell. *Introduction To Machinery Analysis And Monitoring.* Pennwell Publishing Company, second edition, 1993.

[9] Arye Nehorai and Boaz Porat. Adaptive comb filtering for harmonic signal enhancement. *IEEE Transactions On Acoustics, Speech, and Signal Processing*, ASSP-34(5):1124–1138, October 1986.

[10] Norman L. Owsley, Joseph J. Wolcin, and John F. Law. Multichannel maximum a posteriori signal parameter estimation. Technical Memorandum 801004, Naval Underwater Systems Center New London Laboratory, New London, CT 06320, January 1980.

[11] Philip J. Parker and Brian D. O. Anderson. Frequency tracking of non-sinusoidal periodic signals in noise. *Signal Processing*, 20:127–152, 1990.

[12] Leonid Perlovsky. Entropy, likelihood, einstein's model and spectrum estimation. Unpublished paper. Author is with Nichols Research, Wakefield, Massachusetts, USA, June 1996.

[13] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in FORTRAN: The Art of Scientific Computing.* Cambridge University Press, second edition, 1992.

[14] Takuso Sato, Kimio Sasaki, and Yoichi Nakamura. Real-time bispectral analysis of gear noise and its application to contactless diagnosis. *Journal of the Acoustical Society of America*, 62(2):382–387, August 1977.

[15] Roy L. Streit. Shot noise models of wideband targets in clutter. Unpublished paper. Author is with Naval Undersea Warfare Center, Newport, Rhode Island, USA, August 1996.

[16] Roy L. Streit and Tod E. Luginbuhl. Probabilistic mutli-hypothesis tracking. Technical Report 10,428, Naval Undersea Warfare Center, Newport, Rhode Island, February 1995.

[17] D. M. Titterington, A. F. M. Smith, and U. E. Makov. *Statistical Analysis Of Finite Mixture Distributions.* John Wiley & Sons, 1985.

[18] Langford B. White. An iterative method for exact maximum likelihood estimation of the parameters of a harmonic series. *IEEE Transactions on Automatic Control*, 38(2):367–370, February 1993.

[19] C. F. Jeff Wu. On the convergence properties of the em algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[20] Guotong Zhou and Georgios B. Giannakis. Retrieval of self-coupled harmonics. *IEEE Transactions On Signal Processing*, 43(5):1173–1186, May 1995.

Frequency Line Detection and Tracking Using Multiple Hidden Markov Models

D. Van Cappel P. Alinat J.M. Passerieux

Thomson Marconi Sonar SAS, 625 route des Dolines
BP157 06903 Sophia, FRANCE

Abstract

This paper addresses Automatic Detection and Tracking (ADT) of frequency lines in lofargrams. In case of low SNR's and highly variable line frequencies and amplitudes conventional sequential detection techniques (e.g. Nearest Neighbour or Probabilistic Data Association), with severely limited branching factors, are known to suffer from various difficulties. On the contrary algorithms based upon Hidden Markov Models (HMM) appear as a promising way to handle this problem since they allow, (1) to take into account as long as possible observed data blocks (batch processing), (2) to delay the decisions (knowledge of future), (3) to select adaptively between several frequency line variation models implemented in parallel.

This paper describes the implementation of such an HMM-based line ADT and provides some experimental results obtained on realistic complex simulated data. These results clearly outperform a classical recursive NNA ADT that had been previously implemented as reference.

I. Introduction

Among the various underwater sounds the narrow-band tones are of special interest because they are generally produced by man-made rotational machinery. On spectrograms (the so-called lofargrams) these tones result in spectral lines that can be isolated or included in a family of related lines. This paper addresses the problem of Automatic line Detection and Tracking (ADT) in spectrograms, each line being processed separately without taking into account that some lines may belong to a family.

This problem is difficult because generally the available SNR is rather low, the frequency and amplitude are randomly time-varying and the underwater propagation often induces some severe fading effects. Moreover the tracking process may also be disturbed by line crossings.

Spectrogram line ADT is part of the general Multiple-Target-Tracking problem (MTT) encountered in all surveillance systems such as radar, infrared and sonar. MTT has been studied for several decades [1]. Since the beginning the basic components of MTT are track initiation (and deletion), track prediction and association of new detected events selected by gating techniques. At the beginning, due to limited available computational capabilities, the solutions adopted for this problem were generally mono-hypothesis and recursive without knowledge of the future. In these classical solutions each track was updated on each scan by associating only, either the most likely event selected by Nearest Neighbour Association for example (NNA [1]), or a combination of events by the Probalistic Data Association method (PDA [2]). All the other possible hypotheses were not explored. Obviously better results can be achieved when using the multiple hypothesis tracking method (MHT), that is to say by assuming that at any given scan, several possible events are associated to each track. In this way later events (the future) are taken into account in order to choose the various possible associations at the present time. Nevertheless the classical MHT algorithm [3] is complex and its actual results are hampered by the difficulty of keeping the number of hypotheses reasonable.

In spectrogram line ADT progress has recently been accomplished by using the Hidden Markov Model ([4] to [7]). In the HMM-based line ADT the embedded line and measurement models have proved to be very effective and the Viterbi algorithm allows for taking into account the knowledge of the future more easily than the MHT method. A number of HMM line ADTs have already been described. For example, in [5] the influence of observation model dimension is studied: frequency, amplitude and phase. In [8] the computational burden associated with a full Viterbi approach is reduced by using the two thresholds of the sequential detection algorithm (Wald).

In fact the effectiveness of a line ADT depends, among other things, on three factors: firstly, the accuracy of the involved models (e.g. the state transition and observation probability distributions); secondly, the duration of the observation data block available to the algorithm; thirdly, the extent of the decision delay. Moreover line detection and line tracking remain basically two different functions which resort to different algorithms.

In this context our main study objectives encompass (1) clear understanding of the relations between the various choices of parameter values included in the Markov models and the three factors above, (2) development of HMM-based ADT algorithms taking into account the result of the first objective both for line initiation and line tracking, (3) rough comparison of the obtained results with those obtained by conventional techniques (NNA).

The paper is organized as follows. Section II presents the HMM theory applied to the line ADT problem. Section III describes the detection algorithm and Section IV the tracking algorithm. Conclusions about the obtained HMM-based line ADT are finally presented in Section V.

II. Application of HMM to line extraction

A. Lofar Lines

A lofargram is an energy time-frequency representation provided by successive Short Time Fourier Transform (STFT). On lofargrams the narrow band tones result in spectral lines embedded in background noise. Generally the frequency and amplitude of each line are more or less randomly time-dependent.
The frequency resolution of the STFT is supposed to be adapted to the maximum line variation rate so that the line bandwidth is smaller than the frequency resolution. The duration of each frequency line can range from a few scans to several hundred scans. Generally the SNR of the interesting lines is rather low. For a given SNR the more stable a line, the easier is the detection of the line.
At each scan t only the (spectral) maxima larger than a threshold η related to the noise level are taken into account. Let us name as events the selected maxima. Each event i is characterized by its amplitude a_i and by its frequency f_i. The set of all the events at time t constitutes the observation $\mathcal{O}_t$, i.e. the input of the ADT algorithm. As it will be seen further, for the HMM-based ADT this observation $\mathcal{O}_t$ is completely characterized by its likelihood (conditional probalibilty) $P(\mathcal{O}_t/X_t = x_t)$, $X_t = x_t$ where x stands for the frequency value of the line at scan t (state vector).
This conditional probability may be deduced from the probabilistic data description used in the PDA approach. For this purpose let us now assume that the observation at scan t $\mathcal{O}_t$ is composed of K events, $K = K_S + K_{fa}$, where K_S is the number of valid alarms (0 or 1, the probability of detection being $P_d = P(K_S = 1)$), and K_{fa} is the number of false alarms distributed as a Poisson variable with mean μ (depending on the threshold η used for the event detection). Let us also assume that (1) in the case of a false alarm, the frequency is uniformly distributed on the frequency range H of the STFT output and the amplitude is distributed according to a truncated chi-square variable, (2) for a valid alarm the frequency is gaussian with mean equal to the frequency value x_t of the line (and a known standard deviation σ), the amplitude is a non central truncated chi-square, $\mathcal{X}_2^2(\lambda)$, variable with $\lambda = \sqrt{\frac{2E}{N_0}}$ depending on the SNR. Assuming finally that the previous variables are mutually independent and randomly mixed, the likehood of the observed set of events $\mathcal{O}_t$ is given by [2]

$$P(\mathcal{O}_t/X_t = x) = C\ \Lambda(\mathcal{O}_t, X_t = x) \text{ with}$$

$$C = e^{-\mu}\frac{\mu^k}{k!}\left(\frac{1}{|H|}\right)^k\left(\frac{1}{P(\mathcal{X}_2^2 > \eta)}\right)^k \prod_{i=1}^{k} \mathcal{X}_2^2(a_i)$$

$$\Lambda = (1 - P_d) + \frac{|H|}{\mu}\frac{P(\mathcal{X}_2^2 > \eta)}{\sigma\sqrt{2\pi}} \sum_{i=1}^{k} \frac{\mathcal{X}_2^2(\lambda)(a_i)}{\mathcal{X}_2^2(a_i)} e^{-\frac{1}{2}\frac{(f_i - x)^2}{\sigma^2}}$$

This result will be used for line detection and tracking in Sections III and IV.

B. HMM definition and useful algorithms

An observed process $(\mathcal{O}_t)_t$ is an HMM if there exists a hidden process $(Q_t)_t$ taking discrete values in finite set Q such that:

- $(Q_t)_t$ is a Markov process,
- the value of $(\mathcal{O}_t)_t$ at scan t depends on the hidden process $(Q_t)_t$ only by its present value at scan t ,
- the past and the future values of the observed process are conditionally independent knowing the present Q_t of the hidden process.

In our frequency lines ADT the states are the possible values of the frequency lines, a spectral line therefore corresponds to a sequence of the hidden Markov process $(Q_t)_t$ and the observation, as defined in the previous section, consists in a set of measured couples (frequency, amplitude).
In this HMM context, for a given observation sequence $(\mathcal{O}_0, ..., \mathcal{O}_T)$ three basic algorithms [9] have proved to be useful:

- the Forward algorithm provides:
  $P(Q_T = q, \mathcal{O}_0, ..., \mathcal{O}_T)$ and $P(\mathcal{O}_0, ..., \mathcal{O}_T)$
- the Forward-Backward algorithm provides:
  $P(Q_u = q, \mathcal{O}_0, ..., \mathcal{O}_T) \quad u \leq T$
- the Viterbi algorithm provides the single MAP sequence $Q_0 = q_*^{(0)}, ...Q_T = q_*^{(T)}$ which maximizes the joint probability:
  $P(Q_0 = q^{(0)}, ..., Q_T = q^{(T)}, \mathcal{O}_0, ..., \mathcal{O}_T)$

C. HMM and continuous modeling

In the addressed problem the frequency values are actually continuous. Before applying HMM it is therefore necessary to quantify frequencies. In practice this task is acomplished by dividing the whole range H of STFT output into a large number M of adjacent frequency subbands. After this discretization the state i coresponds to the presence of the frequency f within the ith of these subbands.
In some domains, speech processing for example, the HMM modeling allows effective estimation of the model parameters from a sufficient numbers of samples. Such an estimation requires a high quality training data set. For the lofargram line ADT problem considered here we have preferred to deal with simple a priori analytic models from which it is easy to derive the HMM models.

For this purpose the (possibly non linear) dynamic and measurement equations of the continuous process, e.g. :

$$\begin{cases} X_{t+1} = f(X_t) + W_t \text{ with } P_{X_0} & \text{(dynamic equation)} \\ Y_t = g(X_t) + B_t & \text{(measurement equation)} \end{cases}$$

(where noises w_t and n_t are assumed to be zero-mean uncorrelated Gaussian variables with known standard deviations) are integrated over each of the M subbands in order to get state transitions probabilities a_{ij} and observations probabilities. Additional details about this integration in the reference [7].
In the case of spectral line ADT it has been noticed that the frequency line variations can generally be correctly modeled by one of the three models below:

- in many cases by a zero order dynamic equation such as $X_{t+1} = X_t + W_t$. Here W_t is a zero-mean Gaussian noise $\mathcal{N}(0, \sigma_W)$ and σ_W is chosen according to the line short terme stability,
- in some cases, by a zero order dynamic equation including a known deterministic slope: $X_{t+1} = X_t + \dot{x} + W_t$. Here this slope $\dot{x}$ is assumed slowly varying and its value is estimated from the past of the line,
- finally by a first order model given by:

$$\begin{cases} X_{t+1} = X_t + \dot{X}_t + W_t & \text{with } P_{X_0} \\ \dot{X}_{t+1} = \dot{X}_t + \dot{W}_t & \text{with } P_{\dot{X}_0} \end{cases}$$

  where $\dot{W}_t$ is a Gaussian noise $\mathcal{N}(0, \sigma_{\dot{W}})$ independent of W_t and where $\sigma_{\dot{W}}$ is chosen according to the long term line stability and σ_W according to the short term stability.

Each of these models allows a straightforward derivation of the state transition probabilities.

III. Line detection

This first stage aims at detecting the spectral lines. It is a simple binary decision test to be applied on each (f, t) bin of the spectrogram output in order to decide $\mathcal{H}_1$ "a line is present" or $\mathcal{H}_0$ "only background noise is present".
Let us now consider a block of $T+1$ scans of the spectrogram output and assume that the line to be detected is longer than $T+1$. Using expression of $\Lambda(\mathcal{O}_t, Q_t = q)$ given in Section II.A instead of the usual $P(\mathcal{O}_t/Q_t = q)$ and assuming that the background noise is time independent, the Forward-Backward algorithm provides :

$$\Lambda_{fb}(Q_u = q) = \frac{P(Q_u = q, \mathcal{O}_0, ..., \mathcal{O}_T/\mathcal{H}_1)}{P(\mathcal{O}_0, ..., \mathcal{O}_T, /\mathcal{H}_0)}$$

that is to say the likelihood ratio that a line is present at the point frequency f (corresponding to the state q) and time $t = u$.
In fact the position q is unknown in hypothesis $\mathcal{H}_1$ and a Generalized Likelihood Ratio Test (GLRT) has to be used:

$$\Lambda^g_{fb}(q) = \frac{\underset{q}{Max}\ P(Q_u = q, \mathcal{O}_0, ..., \mathcal{O}_T/\mathcal{H}_1)}{P(\mathcal{O}_0, ..., \mathcal{O}_T, /\mathcal{H}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma.$$

In real situations several lines may be simultaneously present and therefore a detection is considered for every q for which $\Lambda^g_{fb}(q) > \gamma$.
In real situations the standard deviation σ_W may also be very different from one line to another. Two zero-order models are therefore used in parallel: a model 1 with a small σ_W (for stable lines),a model 2 with a large σ_W (for unstable lines). Then the GLR test is performed with:

$$Max(\Lambda_{fb}(q/\text{model 1}), \Lambda_{fb}(q/\text{model 2})) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma.$$

In practice for each block, the above detection process is performed at the middle of the block center (i.e. at $u = \frac{T}{2}$), then the block is one-scan shifted in order to obtain the detection at the next time, and so on recursively.
It is also worth noticing that this processing provides an incoherent statistical integration of frequency varying lines similar to the classical so-called Automatic Line Integration (ALI) that provides an incoherent deterministic integration of frequency constant lines. Figure 1 to 4 demonstrate the effectivness of this bi-model Forward-Backward line detector on a simulated (but realistic) lofargram output. Here $T = 21$ scans and the decision delay is 10 scans.

IV. Line tracking

Once detected a line is tracked. The tracking is performed recursively by blocks of duration T, shifted by $\frac{T}{2}$(50 % overlapping): for each block the line is updated with the first half of the whole block duration MAP trajectory provided by the Viterbi algorithm.
Three line models have been implemented in parallel, (1) zero-order model with small s.d. σ_W (for stable lines), (2) zero order model with larger s.d. σ_W (for constant Doppler unstable lines), and (3) first order model with small σ_W (for stable lines with high Doppler rates).
Moreover the commutation between these three models is triggered by a Bayesian test (the LRs being given by the Forward algorithm) on the recent past of the line frequency variations.
Figures 5 and 6 demonstrate the effectiveness of this line ADT on the same simulated lofargram data as figure 1. Parameters for the tracking part of the algorithms have been set as follows: $T = 21$ scans, decision delay of 10 scans, σ_{W_1}=σ_{W_3}=0.1 Hz, σ_{W_2} =0.6 Hz and $\sigma_{\dot{W}_3}$=0.01 Hz/scan.
The results obtained by a standard NNA algorithm, for

the same lofargram data, are shown figure 7 for comparison purpose. For this last ADT algorithm the time constant for detection is limited to 3 scans, the association being performed at each scan with a delay of one scan (NNA 1-back-scan algorithm); moreover a single first-order dynamic model is used.

V. Conclusions

The problem of HMM-based line detection and tracking in spectrograms has been addressed. The obtained detection and tracking algorithms provide us with satisfying results both for simulated input data and for real data (not shown here). Compared to conventional NNA tracking algorithms the improvement is notable, mainly because HMM solution allows us to use long duration blocks (batch processing) and large decision delays.
Finally it should also be mentioned that the computational burden with HMM ADT's remains quite tractable: for the same input data, the increase of the computational power between the HMM solution (result figure 6) and the standard NNA solution (result figure 7) is only about 3 higher.

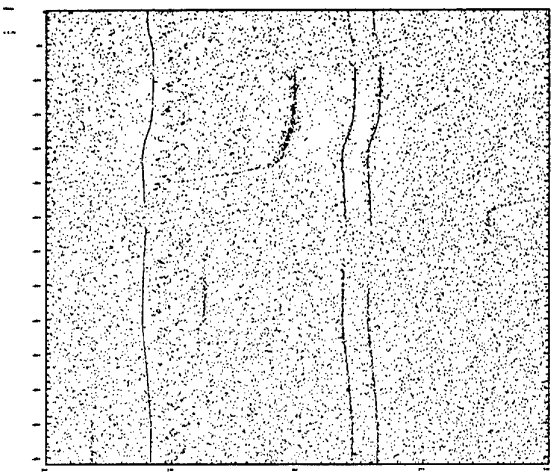


Figure 1: Detected events (Observation)

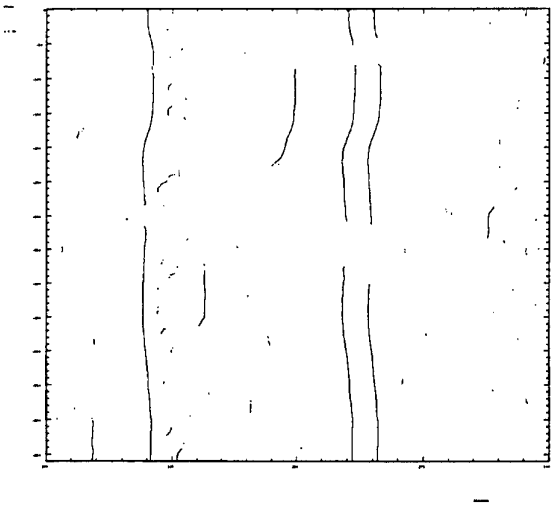


Figure 2: Output of forward-backward detection with zero order model, σ_W=0.1Hz

References

[1] **S.S. Blackman** "Multiple Target Tracking with Radar Applications", Artech House 1986

[2] **Y. Bar-Shalom** "Tracking Methods in a Multi Target Environment" IEEE Trans. on Automatic Control AC-23 (4) pp 618-626 (1978)

[3] **D.B. Reid** "An Algorithm for Tracking Multiple Targets" IEEE Trans. on Automatic Control AC-24 (6), pp 843-854 (1979)

[4] **R.L. Streit and R.F. Barett** "Frequency Line Tracking Using Hidden Markov Models" IEEE Trans. on Acoust., Speech and Signal Processing ASSP-28, pp 709-729 (1990)

[5] **R.F. Barret and D.A. Holdsworth** "Frequency Tracking Using Hidden Markov Models with Amplitude and Phase Information" IEEE Trans. on Signal Processing SP-41 (10), pp 2965-2976 (1993)

[6] **S. Sitbon** "Comparative study betweeen a new HMM tracker and conventional passive algorithms" UDT 1994, pp 441-445

[7] **S. Sitbon and H.M. Passerieux** "New Efficient Target Tracking Based Upon Hidden Markov Model and Probabilistic Data Association", Proc. 29th Asilomar Conf. on Signals,Systems and Computers, Nov. 1995

[8] **F.B. Shin and D.H. Kil** "Full Spectrum Signal Processing using a Classify-Before-Detect Paradigm" J. Acoust. Soc. of Am. 99(4), pp 2188-2197 (1996)

[9] **L.R.Rabiner** "A tutorial on Hiden Markov Models and Selected Applications in Speech Recognition", Proc. of the IEEE, Vol. 77, No. 2, Feb 1989,pp257-286

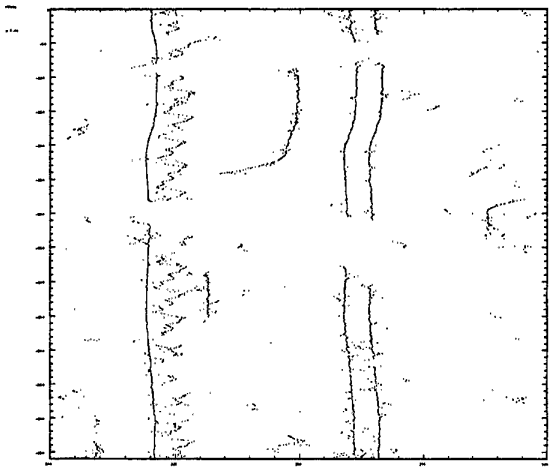


Figure 3: Output of forward-backward detection with zero order model, σ_W=0.6Hz

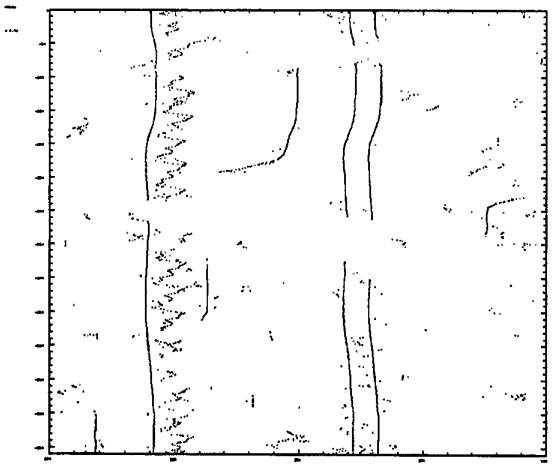


Figure 4: Output of forward-backward detection using 2 zero-order models

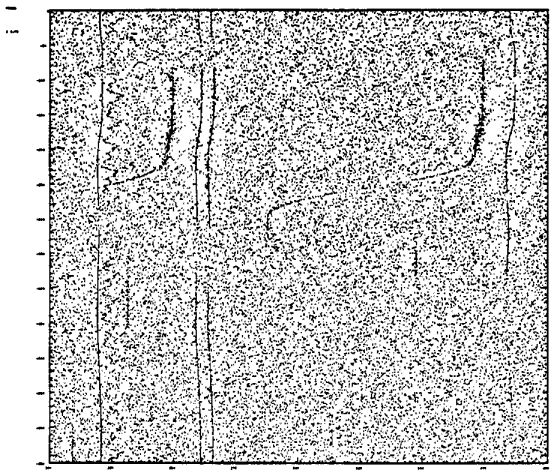


Figure 5: Detected events (Observation)

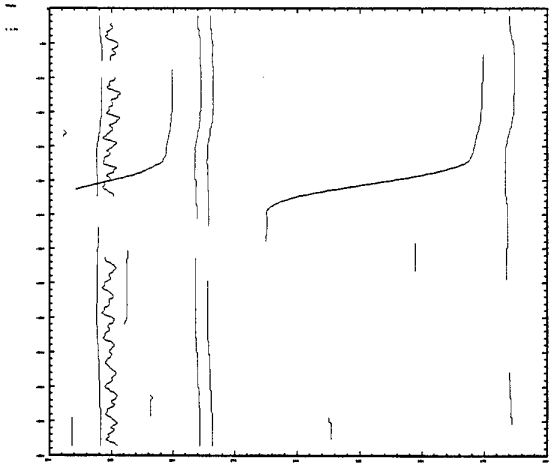


Figure 6: Output of multimodels HMM ADT

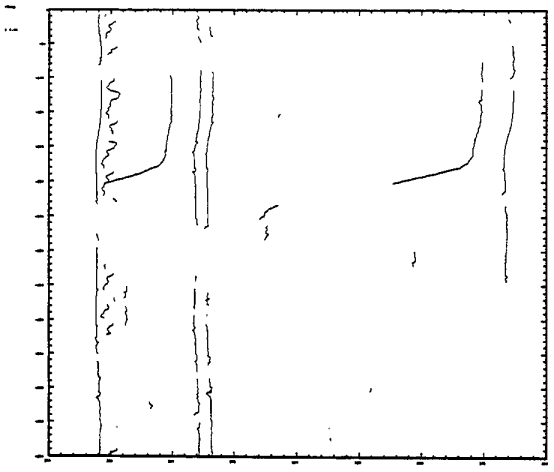


Figure 7: Output of standard NNA 1 backscan ADT

Tracking in the presence of intermittent spurious objects and clutter

D J Salmond* and N J Gordon†
Defence Evaluation and Research Agency, U.K.

ABSTRACT

The sampling based bootstrap filter is applied to the problem of maintaining track on a target in the presence of intermittent spurious objects. This problem is formulated in a multiple hypothesis framework and the bootstrap filter is applied to generate the posterior distribution of the state vector of the required target - i.e. to generate the target track. The bootstrap technique facilitates the integration of the available information in a near-optimal fashion without the need to explicitly store and manage hypotheses from previous time steps.

Keywords: Target tracking, Bayesian methods, bootstrap filter, SIR filter, particle filter, Monte Carlo methods

1. INTRODUCTION

There are many situations where it is required to maintain or initiate tracks on objects in the presence of spurious and potentially misleading measurements. This paper is concerned with the particular problem of maintaining track on a target in the presence of intermittent spurious objects. These spurious objects or sources are spawned in close proximity to the required target at random times and persist for random periods. Only one such source is present at any instant and its birth/death process is modelled by a Markov process. The problem is complicated by the presence of dense random clutter. Position and discrimination (signature) information is available on the objects and clutter from a scanning sensor. It is further assumed that the resolution of the sensor is finite, so when in close proximity, the target and spurious object may generate only a single common return. There is a high degree of measurement association uncertainty.

The usual (Bayesian) approach to problems of this type involves some form of multiple hypothesis filter. For optimality, it is necessary to take account of the history of every possible measurement association hypothesis. As measurements accumulate, this rapidly becomes infeasible, and some hypothesis pruning and/or merging scheme, such as Probabilistic Data Association (see Bar-Shalom and Li [1]) or some mixture reduction method (see Pao [2]), must be imposed. In particular, a successful application of the IMMJPDAF (Interacting Multiple Model Joint Probabilistic Data Association Filter) to the problem of tracking splitting targets with finite sensor resolution, which is clearly related to the example of this paper, is reported by Bar-Shalom, Chang and Blom [3]. Also a multiple hypothesis approach to tracking two closely spaced targets with finite sensor resolution is presented by Koch and van Keuk [4]. If system or measurement nonlinearities are present, further approximations (typically a version of the extended Kalman filter) must be employed.

©British Crown Copyright 1998/DERA
Published with the permission of the Controller of Her Britannic Majesty's Stationery Office.
*DERA, Ively Road, Farnborough, Hants, U.K.
†DERA, St Andrews Road, Malvern, Worcs, U.K.

In this paper, we adopt a different approach to implementing the formal Bayesian recursive filter. Instead of attempting to construct a functional approximation (almost invariably based on Gaussians) of the required pdf (probability density function), a large set of random samples are employed as an equivalent representation. This is the central tenet of the bootstrap filter proposed by Gordon et al [5] for nonlinear/non-Gaussian dynamic state estimation. It was noted by Avitzour [6] that this filter can be applied directly to problems involving measurement association uncertainty, provided that the measurement likelihood can be specified (also see Gordon [7]). The main advantage of this approach is that measurement association hypotheses from previous time steps do not need to be explicitly stored - they are contained implicitly in the sample set. In this case, suboptimality is due to the number of samples being finite, and as the sample size tends to infinity, the exact (optimal) result is obtained.

In the following sections, a statement of the above tracking and identification problem is given and outlines of the standard and hybrid versions of the bootstrap filter algorithm are supplied. An expression for the measurement likelihood necessary for application of the bootstrap filter is obtained and some simulation results are presented. This work is an extension of the results and analysis presented by Salmond et al [8].

2. GENERAL PROBLEM STATEMENT

2.1 Scenario

The dynamics of the primary target (T) are described by the following (known) discrete system model:

$$\underline{x}_{T\,k+1} = \underline{f}_{T\,k}\,(\,\underline{x}_{T\,k}\,,\,\underline{w}_{T\,k}\,) \tag{2.1}$$

where $\underline{x}_{T\,k}$ is the target state vector, $\underline{w}_{T\,k}$ is system driving noise, and $\underline{f}_{T\,k}$ describes the dynamics of the target.

At some random time step the target may spawn a secondary object D in the vicinity of T. Thereafter the secondary object moves independently of T according to the following dynamic model:

$$\underline{x}_{D\,k+1} = \underline{f}_{D\,k}\,(\,\underline{x}_{D\,k}\,,\,\underline{w}_{D\,k}\,) \tag{2.2}$$

where $\underline{x}_{D\,k}$ is the state vector of D. The initial distribution of $\underline{x}_{D\,k}$ at birth is a (known) function of $\underline{x}_{T\,k}$. The secondary object disappears after a random period and later (following another random period) another object D may be produced. The birth /death sequence of the object D is described by a Markov process. If $\gamma_k = 0$ indicates that D does not exist at time t_k and $\gamma_k = 1$ indicates that D is in existence, the transitions $0 \rightarrow 1$ and $1 \rightarrow 0$ depend only on the probabilities:

$$p_{01} = \Pr\{\,\gamma_k = 1 \mid \gamma_{k-1} = 0\,\} \quad \text{and} \quad p_{10} = \Pr\{\,\gamma_k = 0 \mid \gamma_{k-1} = 1\,\}\,. \tag{2.3}$$

Clearly, $\Pr\{\,\gamma_k = 1 \mid \gamma_{k-1} = 1\,\} = 1 - p_{10}$ and $\Pr\{\,\gamma_k = 0 \mid \gamma_{k-1} = 0\,\} = 1 - p_{01}$. If the time step Δt is constant, the average period between the death of one secondary object and the birth of another is $\Delta t/p_{01}$, while the average lifetime of D is $\Delta t/p_{10}$. Note that with this model only two objects may be present at any instant.

At k=0, it is assumed that only the primary target T is present ($\gamma_0 \equiv 0$). The prior distribution of $\underline{x}_{T0}$ is also assumed to be known.

2.2 Sensor model

At each time step k, N_k position measurements $\underline{z}_{ik}$ are received from a sensor whose position is precisely known. If only the primary target T is present (i.e. γ=0), the probability of detecting the target is P_{TD}. If the secondary object D is also present (i.e. γ=1), depending on its proximity to the primary target and the relative geometry, the sensor may be capable of resolving two objects or it may only be able to resolve a single composite object. If the two objects can be resolved, then the probability of receiving a measurement from T is P_{TD} and the probability of receiving a measurement from D is P_{DD} (and these are independent events). If the objects cannot be resolved, the probability of receiving a single composite measurement is P_{JD}. These probabilities may be functions of the appropriate T or D states. Additionally, other spurious or clutter measurements (independent of the two objects) may also be produced by the sensor. We assume that these clutter measurements are uniformly distributed over the measurement space and that they are not subject to resolution limitations (although this would only be significant in exceedingly dense clutter). The number of clutter measurements received at a given time step follows a Poisson distribution with mean m.

Associated with each position measurement $\underline{z}_{ik}$ is a classification flag or signature parameter c_{ik} which may provide an indication of the type of object from which the measurement originated (target, secondary object, composite or clutter), but gives no direct information on object position. c_{ik} could be a discrete output (for example, target, secondary object or clutter) or a continuous parameter such as a measure of target amplitude. The N_k measurements and classifications received at time t_k are denoted

$$Z'_k = \left\{ (\underline{z}_{1k}, c_{1k}), (\underline{z}_{2k}, c_{2k}), \ \dots \ , (\underline{z}_{N_k k}, c_{N_k k}) \right\} \ , \tag{2.4}$$

and the set of all data received up to and including time t_k is denoted

$$Z_k = \left\{ Z'_1, Z'_2, \dots, Z'_k \right\} \ . \tag{2.5}$$

It is assumed that the association between measurements and the objects is a priori unknown. An association hypothesis $\mathcal{H}$ defines a mapping λ from the subscripts of the measurements to their source (target (T), secondary (D), composite (J) or clutter(C)):

$$\lambda : \{1, 2, \dots, N_k\} \rightarrow \{T, D, J, C\} \ .$$

Given λ, the conditional pdfs of the measurements $\underline{z}_{ik}$ are denoted by

$$p_T(\underline{z}_{ik} \mid \underline{x}_{Tk}) \text{ if } \lambda(i) = T \ , \quad p_D(\underline{z}_{ik} \mid \underline{x}_{Dk}) \text{ if } \lambda(i) = D \ , \quad p_J(\underline{z}_{ik} \mid \underline{x}_{Tk}, \underline{x}_{Dk}) \text{ if } \lambda(i) = J \ ,$$

and

$$p_C(\underline{z}_{ik}) \quad \text{if } \lambda(i) = C \ .$$

The performance of the classifier is denoted similarly by $p_T(c_{ik} \mid \underline{x}_{Tk})$ for $\lambda(i) = T$, etc. Note that the classifier performance may be state dependent. It is assumed that these conditional distributions are known.

Although the specification of the classifier output is essentially identical in form to that of the measurements, it is convenient to make the distinction between $\underline{z}$ and c to emphasise that two quite different types of information are available, one of which is strongly indicative of object position while the other is primarily dependent on object type. However, if the classifier output is state dependent (albeit only weakly), then this also provides some information on the object state vector. The bootstrap filter (see

below) is able to exploit this.

2.3 Requirements

It is required to construct the current posterior pdf $p(\underline{x}_{Tk} \mid Z_k)$ of the target at each time step k.

3. MEASUREMENT/CLASSIFICATION LIKELIHOOD

3.1 State vector

For this problem it is convenient to define a system state vector

$$\underline{X}_k = (\underline{x}_{Tk} , \underline{x}_{Dk} , \gamma_k)$$

which evolves with time according to equations (2.1) to (2.3). Note that if $\gamma_k = 0$, $\underline{x}_{Dk}$ is redundant. In this case it is convenient to set $\underline{x}_{Dk} = \underline{x}_{Tk}$.

3.2 General form of likelihood

By careful consideration of the possible measurement-state associations, the likelihood of the state vector $\underline{X}$ given the data set Z' (dropping the time subscript k) may be shown to be (for m>0)

$$\begin{aligned} p(Z' \mid \underline{X}) \propto\ & (1-\gamma) f_T(\underline{x}_T \mid Z') \ + \ \gamma(1-P_{res}) f_J(\underline{x}_T, \underline{x}_D \mid Z') \\ & + \gamma P_{res} \left[f_T(\underline{x}_T \mid Z') f_D(\underline{x}_D \mid Z') - P_{TD} P_{DD} \sum_{j=1}^{N} \ell_T(\underline{x}_T \mid \underline{z}_j, c_j) \ell_D(\underline{x}_D \mid \underline{z}_j, c_j) \right] . \end{aligned} \tag{3.1}$$

Here, the likelihood ratio

$$\ell_T(\underline{x}_T \mid \underline{z}, c) = \frac{1}{m} \frac{p_T(\underline{z} \mid \underline{x}_T)}{p_C(\underline{z})} \frac{p_T(c \mid \underline{x}_T)}{p_C(c)} \ , \tag{3.2}$$

ℓ_D is similar and

$$\ell_J(\underline{x}_T, \underline{x}_D \mid \underline{z}, c) = \frac{1}{m} \frac{p_J(\underline{z} \mid \underline{x}_T, \underline{x}_D)}{p_C(\underline{z})} \frac{p_J(c \mid \underline{x}_T, \underline{x}_D)}{p_C(c)} \ . \tag{3.3}$$

The function $f_T(\underline{x}_T \mid Z')$, which may be interpreted as the likelihood of $\underline{x}_T$ for the case of a single target in clutter, is given by

$$f_T(\underline{x}_T \mid Z') = (1 - P_{TD}) + P_{TD} \sum_{j=1}^{N} \ell_T(\underline{x}_T \mid \underline{z}_j, c_j) \ , \tag{3.4}$$

$f_D(\underline{x}_D \mid Z')$ is similar and

$$f_J(\underline{x}_T, \underline{x}_D \mid Z') = (1 - P_{JD}) + P_{JD} \sum_{j=1}^{N} \ell_J(\underline{x}_T, \underline{x}_D \mid \underline{z}_j, c_j) \ . \tag{3.5}$$

Also P_{res} is the probability that T and D can be resolved (if γ=1) and this is a function of $\underline{x}_{Tk}$ and $\underline{x}_{Dk}$.

The likelihood (3.1) has three principal terms. The first of these (for γ=0) corresponds to cases when the secondary object is not present so that only measurements from the prime target or clutters are available. The second term (for γ=1 and P_{res}=0) represents the case where the secondary object is present but it is not resolved so that only measurements from the composite object or clutter are available. The third term (for γ=1 and P_{res}=1) corresponds to the case where again the secondary object is present but it can be resolved from the target. Various special cases follow directly from this expression, for example if $\gamma \equiv 0$ the likelihood for a single target in clutter is obtained. Also for $\gamma \equiv 1$ and $P_{TD}=P_{TD}=P_{res} \equiv 1$, we recover the special case presented by Salmond et al [8].

3.3 Solution for Gaussian measurements and uniform clutter

The above solution is valid for any form of (time independent) measurement error characteristic (including, for example, quantisation effects and skewed or truncated distributions). Likewise, any form of clutter distribution may be employed provided it is independent of the state $\underline{X}$. Moreover, the bootstrap technique (see below) may be directly applied to such distributions. However, consider the common assumptions of Gaussian measurement errors and uniformly distributed clutter:

$$p_T(\underline{z} \mid \underline{x}) = p_D(\underline{z} \mid \underline{x}) = \mathbb{N}(\underline{z}\,; \underline{h}(\underline{x}), R), \qquad p_J(\underline{z} \mid \underline{x}_T, \underline{x}_D) = \mathbb{N}(\underline{z}\,; \underline{h}_J(\underline{x}_T, \underline{x}_D), R_J),$$

and

$$p_C(\underline{z}) = 1/V.$$

Here, the measurements $\underline{z}$ are independent, zero mean Gaussian perturbations about a function of the state. In the above expressions, $\mathbb{N}$ indicated a Gaussian pdf. For the resolved objects the (possibly nonlinear) measurement function is $\underline{h}$ and the covariance of the Gaussian perturbation or error is R. For the composite return, the measurement function $\underline{h}_J$ depends on both $\underline{x}_T$ and $\underline{x}_D$. This function could depend on some centroid of $\underline{x}_T$ and $\underline{x}_D$ such as $(\underline{x}_T + \underline{x}_D)/2$. Also, the measurement error has covariance R_J which may be different to that of the resolved objects. For example, in the case of a radar sensor, the composite measurement might have a larger error due to glint type effects. The parameter V in the clutter distribution is the volume of the sensor surveillance region (which is assumed to "comfortably" encompass T and D). Note that the spatial density of clutter measurements in this case is $\mu = m/V$.

In this case, the likelihood $p(Z' \mid \underline{X})$ in (3.1) is specified by (for μ>0)

$$\ell_T(\underline{x}_T \mid \underline{z}, c) = \frac{1}{\sqrt{|2\pi R|}\ \mu} \frac{p_T(c \mid \underline{x}_T)}{p_C(c)} \exp\left(-\frac{1}{2} (\underline{z} - \underline{h}(\underline{x}_T))^T R^{-1} (\underline{z} - \underline{h}(\underline{x}_T)) \right), \tag{3.6}$$

and ℓ_D is similar. Also

$$f_T(\underline{x}_T \mid Z') = (1 - P_{TD}) + \frac{P_{TD}}{\sqrt{|2\pi R|}\,\mu} \sum_{j=1}^{N} \frac{p_T(c_j \mid \underline{x}_T)}{p_C(c_j)} \exp\left(-\frac{1}{2} (\underline{z}_j - \underline{h}(\underline{x}_T))^T R^{-1} (\underline{z}_j - \underline{h}(\underline{x}_T)) \right), \quad (3.7)$$

$f_D(\underline{x}_D \mid Z')$ is similar and

$$f_J(\underline{x}_T, \underline{x}_D \mid Z') = (1 - P_{JD}) + \frac{P_{JD}}{\sqrt{|2\pi R_j|}\,\mu} \sum_{j=1}^{N} \frac{p_J(c_j \mid \underline{x}_T, \underline{x}_D)}{p_C(c_j)} \exp\left(-\frac{1}{2} (\underline{z}_j - \underline{h}_J(\underline{x}_T, \underline{x}_D))^T R_j^{-1} (\underline{z}_j - \underline{h}_J(\underline{x}_T, \underline{x}_D)) \right). \quad (3.8)$$

4. OUTLINE OF THE BOOTSTRAP FILTER

4.1 Basic formulation

The bootstrap filter (which is also sometimes denoted the SIR - sampling importance resampling - filter) is described by Gordon et al [5],[9]. Briefly, suppose we have a set of independent random samples $\{\underline{X}_{k-1}(i) : i=1,..,N_S\}$ drawn from the pdf $p(\underline{X}_{k-1} \mid Z_{k-1})$. The bootstrap filter is an algorithm for propagating and updating these samples to obtain a set of values $\{ \underline{X}_k(i) : i=1,..,N_S \}$, which are approximately distributed as independent random samples from $p(\underline{X}_k \mid Z_k)$. Thus the filter is an approximate mechanisation (simulation) of the prediction and update stages of a recursive Bayesian filter.

Prediction: Each sample is passed through the system model to obtain samples from the prior at time step k to give $\underline{X}_k^*(i) = \underline{f}_{k-1}(\underline{X}_{k-1}(i), \underline{w}_{k-1}(i))$, where $\underline{w}_{k-1}(i)$ is a sample drawn from the pdf of the system noise $p(\underline{w}_{k-1})$.

Update: On receipt of the data set Z_k', evaluate the likelihood of each prior sample and so obtain a normalised weight for each sample:

$$q_{ki} = \frac{p(Z_k' \mid \underline{X}_k^*(i))}{\sum_{j=1}^{N_s} p(Z_k' \mid \underline{X}_k^*(j))} \quad (4.1)$$

Thus define a discrete distribution over $\{\underline{X}_k^*(i):i=1,..,N_S\}$, with probability mass q_{ki} associated with element $\underline{X}_k^*(i)$. Now resample (with replacement) N_S times from the discrete distribution to generate samples $\{\underline{X}_k(i):i=1,..,N_S\}$, so that for any j, $\Pr\{\underline{X}_k(j) = \underline{X}_k(i)\}=q_{ki}$.

The above steps of prediction and update form a single iteration of the recursive algorithm. To initiate the algorithm, N samples $\{\underline{X}_1^*(i):i=1,..,N_S\}$ are drawn from the known prior $p(\underline{X}_1)$. These samples feed directly into the update stage of the filter. The samples $\underline{X}_k(i)$ are approximately distributed as the required pdf $p(\underline{X}_k \mid Z_k)$.

It was pointed out by Avitzour [6] that this sample based filter could be directly applied to problems involving measurement association uncertainty - also see Gordon [7]. To apply the technique

it is necessary to construct the likelihood $p(Z'_k \mid \underline{X}_k)$ - see equation (4.1). This is available for the problem defined above (see equation (3.1)) and so the bootstrap filter may be directly applied.

The justification for the bootstrap filter is based on an asymptotic result. It is most difficult to prove any general result for a finite number of samples. Likewise it is most difficult to make any precise, provable statement on the crucial question of how many samples are required to give a satisfactory representation of the densities for filter operation. However, it is clear that the required number N_S depends on at least four factors: (i) the dimension of the state space, (ii) the typical "overlap" between the prior and the likelihood, (iii) the complexity of the state pdf, and (iv) the required number of time steps. If N_S is too small, the number of truly distinct values in the sample set may collapse, particularly if the system noise is low or non-existent. This is a major drawback of the bootstrap approach. The hybrid version of the bootstrap filter described below is intended to avoid this degeneracy - also see Gordon et al [5] and [9] where these practical issues are discussed further. Another recent analysis of sample based filters with other suggestions for maintaining a valid representation of the state distribution is given by Carpenter et al [10] and Pitt and Shephard [11].

4.2 Hybrid version of the bootstrap filter

The hybrid form of the bootstrap filter suggested by Gordon [7] is one approach to compensating for the finite size of the bootstrap sample set. The scheme was devised particularly for cases where the distribution of the state space may be multimodal and so is suited to problems involving measurement association uncertainty.

In most filtering problems it is known *a priori* that the state space parameters are continuous rather than discrete and so it is natural to consider smoothing the bootstrap sample set by fitting a continuous distribution at each time step. To retain the advantage of the sampling approach it is essential that the fitted distribution should be sufficiently versatile to follow the probability mass distribution of the samples - in particular, as noted above, the possibility of widely spaced modes should be accommodated. To meet this requirement, the hybrid filter employs a Gaussian mixture distribution to fit the weighted sample set $\{\underline{X}_k^*(i) : i=1,..,N_S\}$. Thus the modified update procedure is as follows:

> Obtain the discrete distribution over $\{\underline{X}_k^*(i) : i=1,..,N_S\}$ with probability mass q_{ki} associated with element $\underline{X}_k^*(i)$ using equation (4.1) as before. Then, instead of resampling directly, obtain a Gaussian mixture approximation to the posterior pdf at time step k:
>
> $$p(\underline{X}_k \mid Z_k) \approx \sum_{j=1}^{C(k)} \beta_{kj} \mathbb{N}(\underline{X}_k ; \hat{\underline{X}}_{kj}, P_{kj}) \quad , \tag{4.2}$$
>
> where $\mathbb{N}$ is a Gaussian pdf, C(k) is the number of mixture components required to approximate the weighted sample set, $\hat{\underline{X}}_{kj}$ is the mean of the j^{th} component, P_{kj} is the covariance of the j^{th} component and β_{kj} is the weight attached to the j^{th} component (clearly $\sum_{j=1}^{C(k)} \beta_{kj} = 1$). N_S samples $\{\underline{X}_k(i): i=1,..,N_S\}$ are then drawn from this mixture to complete the update stage.

By using a mixture approximation to the weighted samples, the gaps between the samples are effectively "filled in" in a way that is consistent with the disposition of the probability mass indicated by the relative location and importance of the samples.

The technique used to fit the mixture to the sample set is a form of kernel density estimation. Firstly a Gaussian kernel is placed around each sample to form a Gaussian mixture of N_S components: $\sum_{j=1}^{N_s} q_{kj}\, \mathbb{N}(\underline{X}_k ; \underline{X}_k^*(j), h\, P_k')$. Here, $P_k' = \sum_{j=1}^{N_s} q_{kj} (\underline{X}_k^*(j) - \underline{\hat{X}}_k)(\underline{X}_k^*(j) - \underline{\hat{X}}_k)^T$ is the overall covariance of the sample set, $\underline{\hat{X}}_k = \sum_{j=1}^{N_s} q_{kj} \underline{X}_k^*(j)$ is the overall mean and $h = 0.5\, N_S^{-2/d}$ is a multiplicative factor, d being the dimension of the state space. (The form of h is motivated by considering an equivalent rectangular grid of points over the state space.) This mixture is then "reduced" by merging groups of similar components in such a way that the overall mean and covariance of the mixture is unchanged. This is done using a technique described by Salmond [12] which attempts to achieve the desired reduction while minimising changes to the structure of the mixture. The number of components C(k) in the reduced mixture is less than or equal to some specified upper bound C_{max}. In the simulation below C_{max} has been set to 50. The parameters h and C_{max} control the smoothness of the fitted mixture and are essentially (problem dependent) tuning parameters. However, simulation experiments indicate that filter performance is not highly sensitive to these parameter values.

There are a number of potential advantages in the hybrid formulation of the bootstrap filter. The most obvious is that by smoothing the samples it should be possible to reduce the number of samples N_S relative to the standard filter and still maintain satisfactory performance. The acceptable reduction is likely to be problem dependent. Another advantage is that the Gaussian mixture approximation (4.2) provides a convenient functional form for the posterior pdf, from which it is easy to find marginal means and covariances or to plot marginal pdfs. Furthermore, this is a convenient form for the analysis of linear operations on the posterior (such as linear prediction). Also note that efficient routines exist for generating random samples from multivariate Gaussian distributions, so that it is simple to generate posterior samples via (4.2).

5. SIMULATION EXAMPLE

5.1 Scenario

Several simulation experiments have been performed for a scenario in the X-Y plane. Both objects T and D (if it exists) are assumed to obey the common second order, linear, Cartesian model. The X-coordinate for this model is

$$\left.\begin{aligned} x_{ik+1} &= x_{ik} + \Delta t\, \dot{x}_{ik} + (\Delta t^2/2) w_{ixk} \\ \dot{x}_{ik+1} &= \dot{x}_{ik} + \Delta t\, w_{ixk} \quad , \end{aligned}\right\} \tag{5.1}$$

for i = T and D. The Y coordinate is similar. The driving noise sequences w_{TXk}, w_{TYk}, w_{DXk} and w_{DYk} are independent Gaussian random processes of variance q. Note that since the models for T and D are the same in this example, trajectory characteristics cannot be used to distinguish between T and D. If the dynamics of T and D were known to be different and this could be modelled (as allowed in the general formulation). This would aid the discrimination process. The state vector for the system is

$$\underline{X}_k = \left(x_T, \dot{x}_T, y_T, \dot{y}_T, x_D, \dot{x}_D, y_D, \dot{y}_D, \gamma\right)_k \quad .$$

In the example below, the system driving noise q is set to 0.015^2 and the time step Δt=0.25. Also, the probability of D being born at time k+1 given that it does not exist at k is $p_{01} = \Pr\{ \gamma_{k+1} = 1 \mid \gamma_k = 0 \} = 0.1$, and the probability of it dying is $p_{10} = \Pr\{ \gamma_{k+1} = 0 \mid \gamma_k = 1 \} = 0.2$. Thus the average lifetime of a D object is $5\Delta t$, and the average interval between the death of one D object and the birth of another is $10\Delta t$.

Measurements of range and bearing ($\underline{z}_{ik} = (z_{irk}, z_{i\theta k})$) are taken from a sensor located at the origin. Thus for measurements originating from T or D, the nonlinear measurement function is defined by

$$z_{iRk} = r_{\lambda(i)k} + v_{iRk} \qquad \text{and} \qquad z_{i\theta k} = \theta_{\lambda(i)k} + v_{i\theta k} \quad , \tag{5.2}$$

where for resolved objects (λ(i)=T or D)

$$r_{\lambda(i)k} = \sqrt{x^2_{\lambda(i)k} + y^2_{\lambda(i)k}} \qquad \text{and} \qquad \theta_{\lambda(i)k} = \tan^{-1}\left(\frac{y_{\lambda(i)k}}{x_{\lambda(i)k}}\right), \tag{5.3}$$

and for the case of an unresolved composite measurement (λ(i)=J)

(5.4)

The measurement errors v_{irk} and $v_{i\theta k}$ are independent, zero mean, Gaussian processes of variance σ_r^2 and σ_θ^2, respectively (for λ(i)=T, D or J). T and D may be resolved provided they do not fall into the same range/bearing resolution cell (Δr, $\Delta\theta$), i.e.

$$P_{res}(\underline{x}_T, \underline{x}_D) = \begin{cases} 0 & \text{if } |r_T - r_D| < \Delta r \text{ and } |\theta_T - \theta_D| < \Delta\theta \\ 1 & \text{otherwise} \end{cases} \quad . \tag{5.5}$$

(Also see the resolution model suggested by Koch and van Keuk [4].) For the example results presented below:

$$\sigma_r = 0.01 \ , \ \Delta r = 0.05 \ , \ \sigma_\theta = 0.01 \text{ radians and } \Delta\theta = 0.05 \text{ radians.}$$

Note that the measurement error is substantially less that the sensor resolution, a common feature of radar systems. Also the detection probabilities for T, D and the composite return are all 0.99 (i.e. $P_{TD} = P_{DD} = P_{JD} = 0.99$).

Clutter measurements are uniformly distributed in range and bearing over the field of view of the sensor. In the simulation experiments, an acceptance gate in the measurement space is defined to reject any measurements that clearly originate from clutter and do not assist in the estimation of the state vector. This gate is defined by the maximum extent of the predicted T and D position samples plus four standard deviations of the measurement error. In the simulation experiment below, the clutter density is high: the average number of clutter returns in a $\Delta r \times \Delta\theta$ resolution cell is 0.25 and the average number of returns in a $\sigma_r \times \sigma_\theta$ cell is 0.01.

Associated with each measurement is a discrete classification flag which takes the values T, D or C. The classifier performance of the sensor against the target is a function of the target aspect presented. Assuming that the target's axis is directed along its velocity vector, the classification performance is a function of $\psi = \theta_{heading} - \theta_T$ (where $-180° < \psi < 180°$) where θ_T is the sightline angle between the sensor and the target with respect to the X-axis and

$$\theta_{heading} = \tan^{-1}(\dot{y}_T / \dot{x}_T)$$

is the target heading relative to the X-axis. The classification probabilities for the simulation example are given in the following table:

Origin of actual measurement	Classifier output		
	T	D	C
T for $10°<\|\psi\|<170°$	0.6	0.3	0.1
T for $0°<\|\psi\|<10°$ or $170°<\|\psi\|<180°$	0.45	0.45	0.1
D	0.3	0.6	0.1
J	0.45	0.45	0.1
C	0.15	0.15	0.7

Thus it is assumed that for D and clutter, performance is independent of aspect. Note that when the subtended target aspect is within 10° of the target's axis, the classifier output is equally likely to be T or D. At other aspects, performance is useful. Also the classifier has a 10% chance of mistaking the T, D or J for clutter, and correctly recognises clutter with a probability of 70%. The composite return J is equally likely to be classified as T or D - there is no classification J in this example. The state dependency of the target classification performance considerably complicates the estimation problem. The accuracy of the state information affects the degree to which the filter can rely on the classifications, while, conversely, it is possible to learn about the direction of the target velocity vector from the sequence of classifications. As already noted, the bootstrap filter is able to accommodate (and exploit) this state dependency.

At k=0, $\gamma \equiv 0$ so that only T is present. The prior distribution of the position of T is Gaussian with mean $(\bar{x}_{T0}, \bar{y}_{T0})$ and covariance

$$\begin{pmatrix} \sigma^2_{TX0} & 0 \\ 0 & \sigma^2_{TY0} \end{pmatrix} .$$

The prior distribution of the initial target velocity is defined in terms of direction and magnitude: the direction being uniformly distributed over $[0 , 2\pi)$ and the magnitude being uniformly distributed over $[0 , V_{T\,max}]$. In the example below,

$$\left.\begin{aligned} \bar{x}_{T0} = 0.1 \quad , \quad \bar{y}_{T0} = 1.9 \\ \sigma_{TX0} = \sigma_{TY0} = 0.25 \\ \text{and } V_{T\,max} = 0.05 \quad . \end{aligned}\right\}$$

The prior distribution of the position and velocity of each D at birth is assumed to Gaussian with mean $\underline{x}_T$ and a diagonal covariance matrix. In the example below, the standard deviation of the x and y positions is 0.03 ($= 3\sigma_r = 0.6\Delta r$) and the standard deviations of the x and y velocity components is 0.01 ($= 0.2V_{T\,max}$). Thus, D type objects are generated in close proximity and with similar velocities to the target (relative to the problem parameters).

5.2 Illustrative results

The results of a simulation experiment over 50 time steps are shown in figs 1-4 for both the standard and hybrid versions of the bootstrap filter. The simulated measurements and actual object paths are identical for the two filters so a direct comparison is possible. Furthermore, the measurement and classification statistics are perfectly matched to the filter parameters as are the dynamics of T and D. The standard filter is run with $N_S = 50000$ sample points while the hybrid version employs only $N_S = 5000$ samples. In the implementation of the hybrid filter, the samples for which γ=1 are clustered separately from those with γ=0. Thus the discrete nature of γ is maintained.

Diagrams

The two plots of fig 1 show the object paths, measurements and track for the standard bootstrap filter. In the upper plot, the actual path of the target is shown as a continuous line joining diamonds which indicate the target positions at each time step. The actual paths of the secondary D type objects are shown as continuous lines connecting crosses. There are two of these D type paths (alive at different times) in this example. The remaining individual symbols indicate the positions of those measurements that actually originate from the target. Those measurements correctly classified as type T are shown as diamonds, those incorrectly classified as type D are shown as crosses and those incorrectly classified as clutter are shown as asterisks. Those measurements which are the product of an unresolved T/D pair are surrounded by a square box. The position estimate (i.e. the mean of the posterior, $E[x_{Tk}, y_{Tk} \mid Z_k]$) from the tracking filter is shown as a dashed line. The lower plot of fig 1 shows all measurements accepted by the filter over the 50 time steps. The notation is as above and the track from the filter is repeated. Note that the lower plot encompasses the region shown in the upper diagram.

Fig 2 presents an analysis of the tracking problem, again for the standard bootstrap filter, as a function of time. The top left plot of fig 2 shows the type of measurements and associated classifications originating from T and D. At each time step, there is either a single measurement from the target T if D is not present, a single composite measurement J if the object is present but not resolved and two measurements T and D if D is present and the objects are resolved. Note that in this example, T and D are always detected. The classifications (T, D or C) associated with the measurements are indicated by different measurement symbols. The two periods when D is present (i.e. γ=1) can be clearly seen. The actual value of γ is also shown in the lower left plot of fig 2 together with the estimated value (continuous line) of γ from the filter, i.e $E[\gamma_k \mid Z_k]$. The top right plot shows the number of measurements that pass the filters acceptance test. This indicates the level of corrupting clutter. Finally the lower right plot indicates the filter estimation error in the Y coordinate. The continuous line shows the error in the mean: $E[y_{Tk} \mid Z_k] - y_{Tk}$. The dashed lines indicate the 98% error limits, i.e. 98% of the probability mass of the Y marginal posterior pdf lies within these limits.

Figs 3 and 4 show similar information for the hybrid version of the filter. However, since the plot of measurement types from T and D would be identical for both filters, this has been replaced by a plot showing the estimation error for the X coordinate of the hybrid filter.

Observations

Comparing the two sets of figures (1 and 2 with 3 and 4) it can be seen that the results from the two versions of the bootstrap filter are broadly similar. The main detailed differences are in the estimates of γ and in the track mean and confidence limits in the vicinity of the resolved D type object (about time steps 40 to 45). Also the standard version of the filter accepts rather more clutter measurements during the initial transient of the first few time steps.

The filters successfully accommodate the release of two D type objects (at different times) produced in close proximity to the target and moving with similar velocities. In the case of the first release at time step 20, the object only exists for 6 time steps and is never resolved. This has little effect on the filters although there is some indication of an increase in the estimate of γ (possibly due to a perceived jump in the target measurements when the decoy is launched and a change in the dynamics - which is effectively smoothed by the averaging of the two object paths). In the case of the second release at time step 35, the object gradually separates from the target and is resolved as a separate entity after 4 time steps. This presents a more difficult problem. The dynamics of D are quite similar to T. Furthermore, during the 7 time steps when T and D are resolved, the target is incorrectly classified as a D type object on 4 occasions while D is correctly classified on only 3 occasions. The filters correctly recognise this uncertainty as is shown by the increase in the interval between the 98% confidence limits which accommodate the separating T/D pair. This is particularly clear for the hybrid version (see fig 4). Note that soon after the two objects are resolved, the estimates of γ rise to above 0.8 (or 0.9 for the hybrid version) showing that the filters are confident that two objects are present. When D dies, the estimates of γ rapidly fall to about 0.2.

The number of clutter measurements accepted by the filters is generally between about 5 and 10 for each time step after the transient has decayed. Although this is quite high, the acceptance gate is generous so that these false returns are often fairly remote from the objects. Also, the classifier correctly recognises clutter on 80% of occasions. Nevertheless, the filters do successfully accommodate this level of disturbance. Note that, save for a single time step, the actual target position falls within the 98% confidence limits shown in figs 2 and 4.

In this example, the initial target position was $x_{T1} = 0$, $y_{Tk} = 2$, and the initial velocity was 0.03 in the positive X direction. For the observer at the origin, the target aspect the target aspect over the 50 time steps was always within the interval $10° < |\psi| < 170°$ (as is clear from fig 1) where the target classifications are informative. However, it should be noted that when the velocity estimate is very uncertain (particularly during the initial transient) the filter must allow for the possibility of the target classifications providing no discrimination against D type objects.

6. CONCLUDING REMARKS

The formal Bayesian solution to a complex nonlinear problem has been implemented using the sample based bootstrap filter. The test case included finite sensor resolution in the range-bearing domain, tracking in a Cartesian frame, interfering spurious objects and clutter, and state dependent classification information. Similar, good performance was obtained from the basic form of the bootstrap filter and from a hybrid version (the latter operating with only 10% of the samples required for the standard filter). The filters implicitly handle a large number of association hypotheses without explicit, complex hypothesis

construction and management.

The performance of the bootstrap method should be compared with other schemes including nearest neighbour, the IMMJPDAF [3] and an "explicit" multiple hypothesis approach. In particular, the trade-off between tracking performance, computation time and computer code complexity should be investigated. Furthermore, although the reported simulation example is representative of a number of other experiments, a full Monte Carlo study is required for proper assessment. Finally, the Bayesian model employed here allows only a single spurious object to exist at any one time. The model should be extended to allow for a number of such sources.

REFERENCES

1. Y. Bar-Shalom and X.R. Li, *Multitarget-multisensor tracking: principles and techniques*, Storrs, CT: YBS Publishing, 1995.
2. L. Pao, "Multisensor multitarget mixture reduction algorithms for tracking", *AIAA Journal of Guidance, Control and Dynamics*, Vol 17, No 6, pp. 1205-1211, 1994.
3. Y. Bar-Shalom, K.C. Chang and H.A.P. Blom, "Tracking splitting targets in clutter by using an Interacting Multiple Model Joint Probabilistic Data Association Filter", Chpater 4 of *Multitarget-Multisensor Tracking: Applications and Advances Volume II*, edited by Y. Bar-Shalom, Artech House, 1992.
4. W. Koch, G. van Keuk "Multiple hypothesis track maintenance with possibly unresolved measurements", *IEEE Transactions on Aerospace and Electronic Systems*, Vol 33, No 3, pp. 883-892, 1997.
5. N.J. Gordon, D.J. Salmond and A.F.M. Smith, "A novel approach to nonlinear/non-Gaussian Bayesian state estimation", *IEE Proceedings on Radar, Sonar and Navigation*, 140, 2, pp. 107-113, 1993.
6. D. Avitzour, "A stochastic simulation Bayesian approach to multitarget tracking", *IEE Proceedings on Radar, Sonar and Navigation*, Vol 142, No 2, pp. 41-44, 1995.
7. N.J. Gordon, "A hybrid bootstrap filter for target tracking in clutter", *IEEE Transactions on Aerospace and Electronic Systems*, Vol 33, No 1, pp. 353-358, 1997.
8. D.J. Salmond, D.Fisher and N.J. Gordon, "Tracking and identification for closely spaced objects in clutter", *Proceedings of the European Control Conference*, Brussels, 1997.
9. N.J. Gordon, D.J. Salmond and C.M. Ewing, "Bayesian state estimation for tracking and guidance using the bootstrap filter", *AIAA Journal of Guidance, Control and Dynamics*, Vol 18, No 6, pp. 1434-1443, 1995.
10. J. Carpenter, P. Clifford, P. Fearnhead, "An improved particle filter for non-linear problems", to appear in *IEE Proceedings on Radar, Sonar and Navigation.*
11. M.K. Pitt and N.Shephard, "Filtering via simulation:auxiliary particle filter", Working Paper from Nuffield College Oxford, U.K., 1997.
12. D.J. Salmond, "Mixture reduction algorithms for target tracking in clutter", *Signal and Data Processing of Small Targets*, edited by O Drummond, SPIE Volume 1305, 1990.

ACKNOWLEDGMENT

This research was sponsored by the U.K. MOD Corporate Research Programme, TG3.

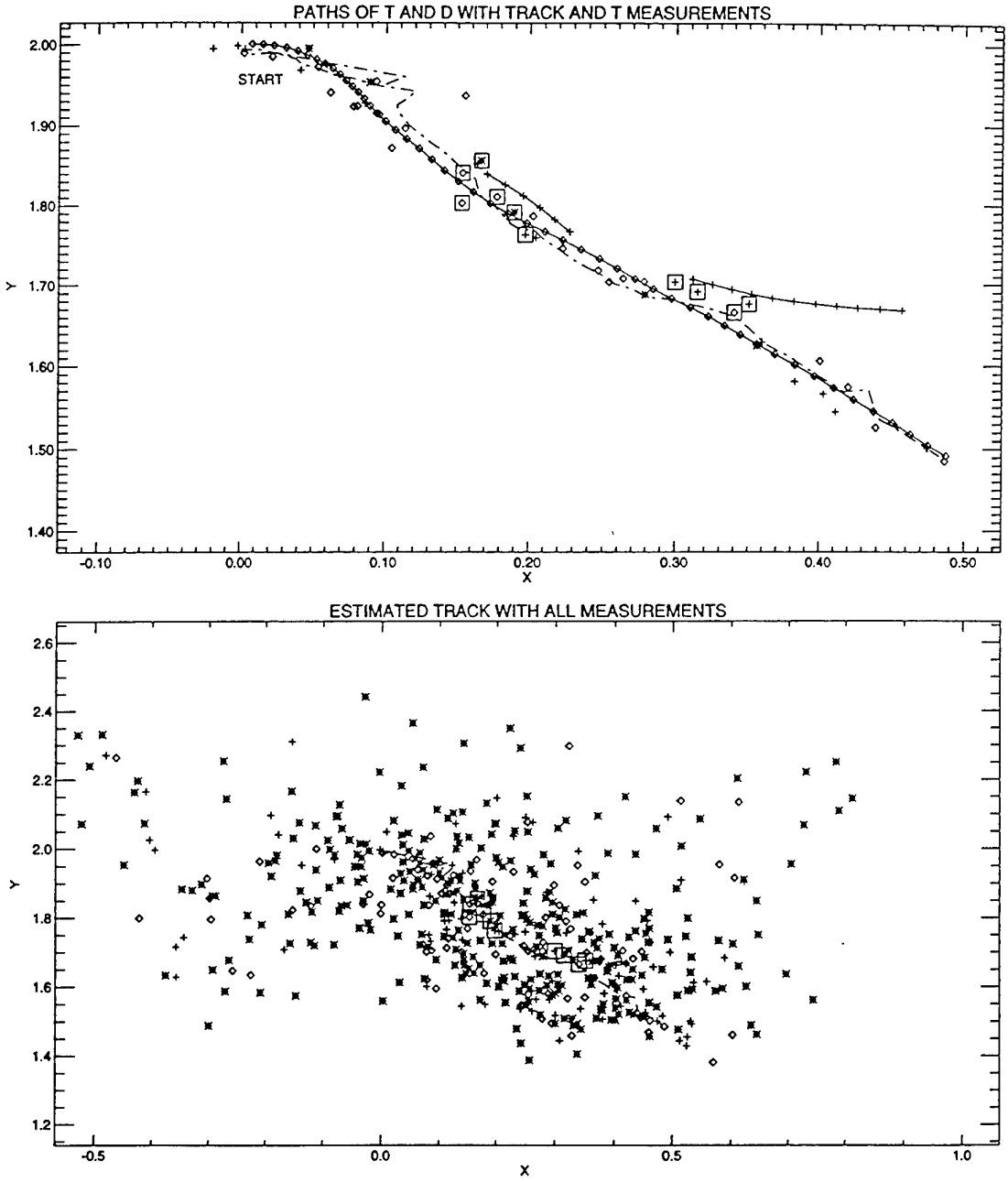


Fig 1: Object paths, measurements and track for standard version of the bootstrap filter

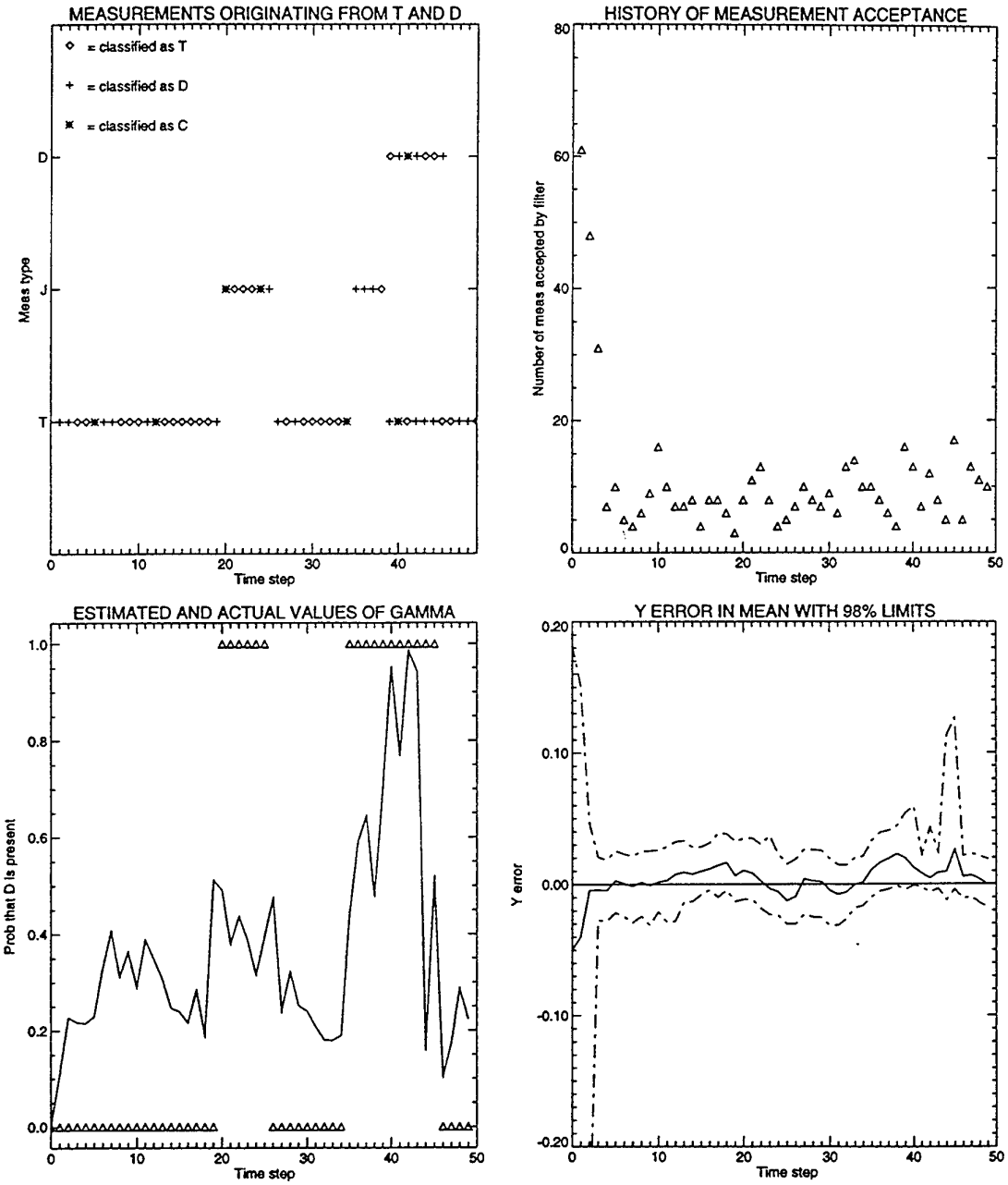


Fig 2: Performance analysis for standard version of bootstrap filter

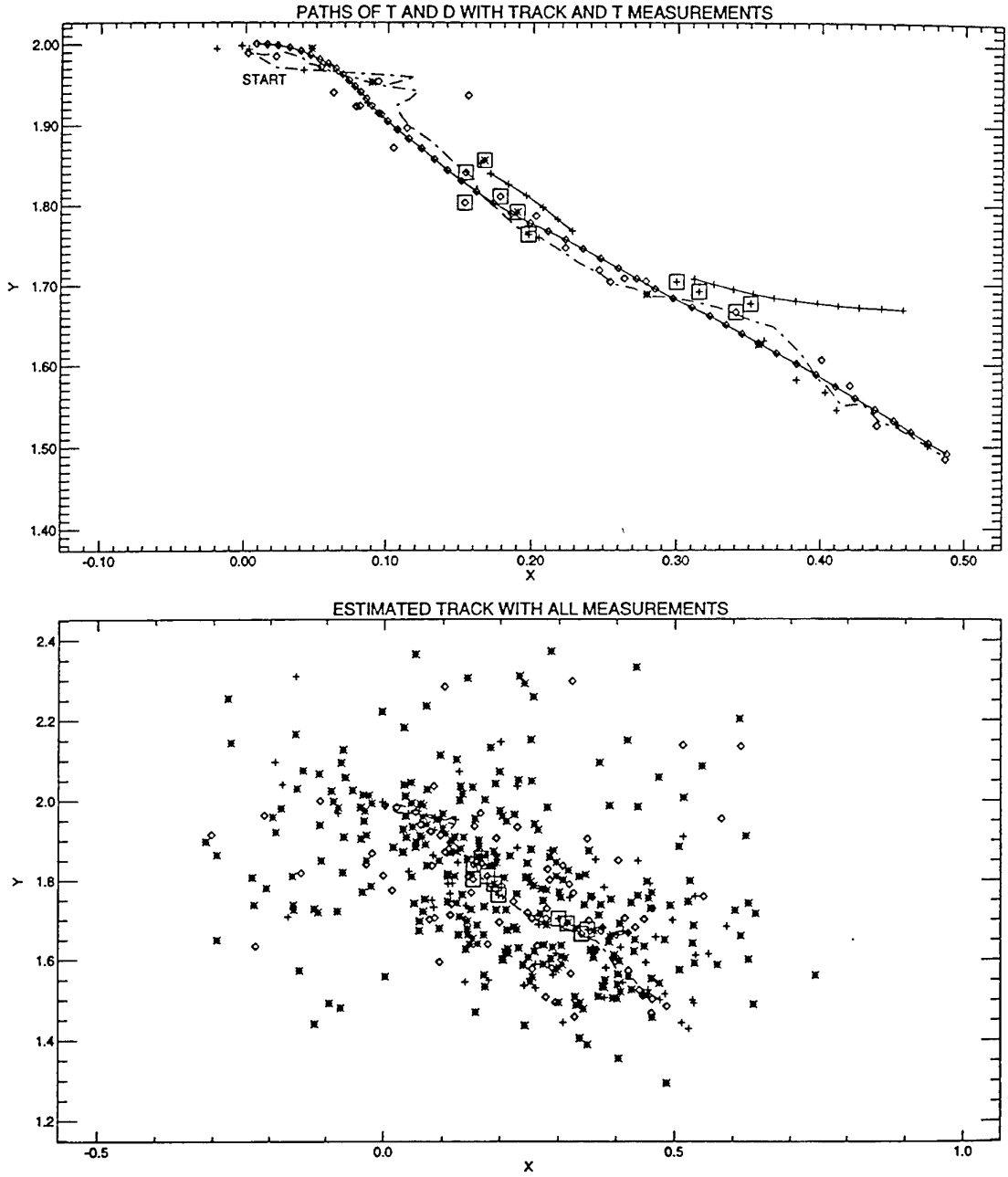


Fig 3: Object paths, measurements and track for hybrid bootstrap filter

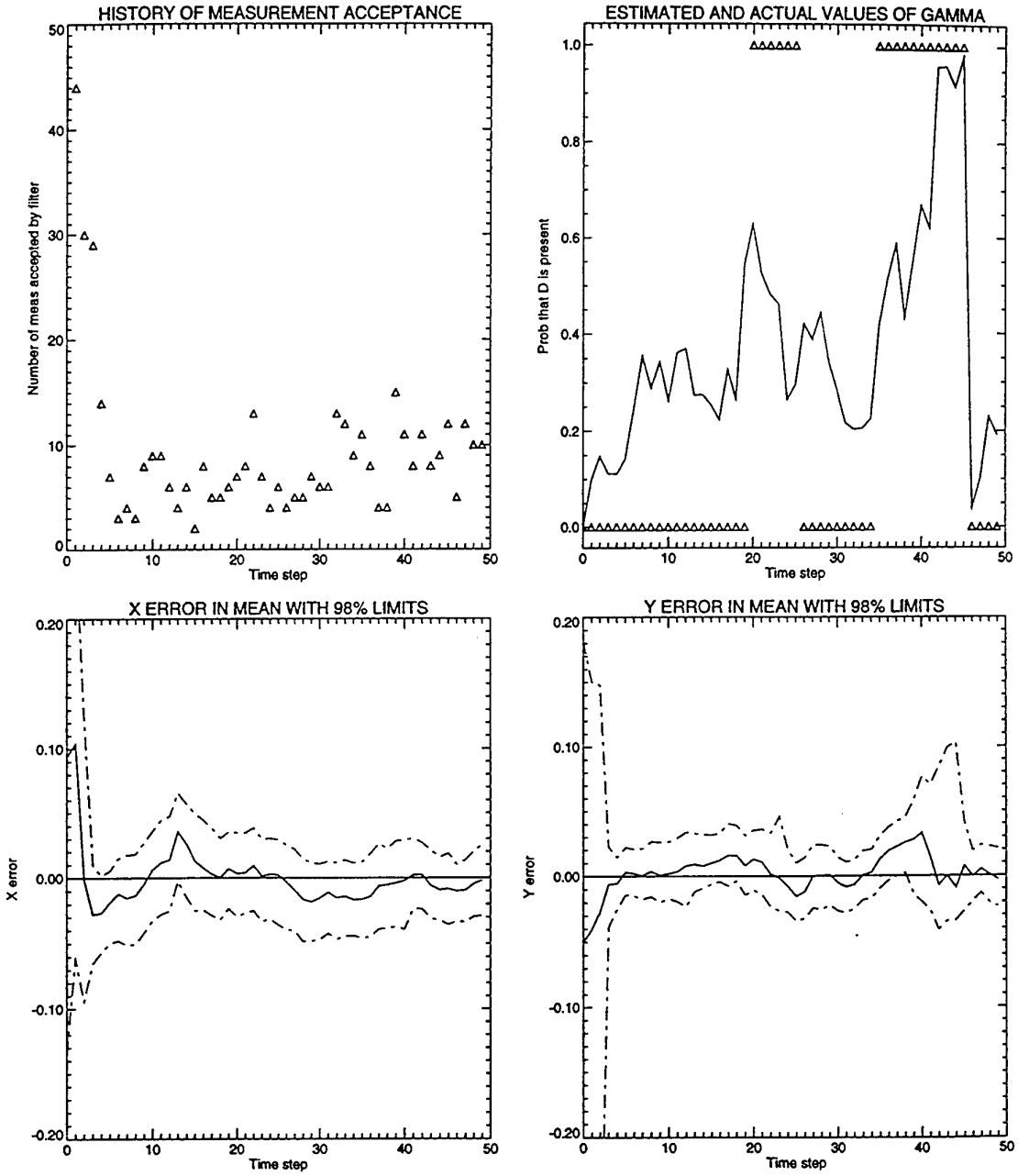


Fig 4: Performance analysis for hybrid bootstrap filter

Augmented Bearings-Only Target Motion Analysis With Propagation Loss Models*

Roy L. Streit
Naval Undersea Warfare Center
Code 2002, Building 1171/3
Newport, RI 02841
streitrl@csd.npt.nuwc.navy.mil

Michael J. Walsh
Naval Undersea Warfare Center
Code 2211, Building 1171/1
Newport, RI 02841
walshmj@csd.npt.nuwc.navy.mil

Abstract

This paper studies the augmented bearings-only target motion analysis problem, characterized by (1) bearing measurements augmented with received signal-to-noise ratio measurements, and (2) a combined acoustic propagation and sensor performance prediction model. A novel track-before-detect empirical maximum *a posteriori* (EMAP) approach is presented in which measurements are assumed independent prior to the detection decision. A limiting case of the EMAP formulation is the maximum likelihood formulation of the traditional bearings-only problem. By treating the range measurement corresponding to a bearing measurement as the missing information, the method of expectation-maximization is used to derive an algorithm for EMAP estimation. The EMAP algorithm is very simple — it is an iteratively reweighted linear least-squares algorithm, and does not require computing the gradient of the EMAP objective function. Moreover, the EMAP formulation enables a simple heuristic for compensating for model mismatch between the acoustic propagation prediction and the real world. The capacity to compensate for mismatch is important in applications that have insufficient data to produce accurate acoustic predictions. Triangulation and constant velocity target examples are presented.

1 Introduction

The fidelity of currently available ocean acoustic propagation and sensor performance prediction models is often not completely satisfactory, that is, accurate models of the environment may yield predictions that are mismatched to the real world because of the lack of adequate or current environmental data. Nonetheless, it may be possible with care to exploit such propagation models for target motion analysis (TMA) purposes. The main contributions of this paper are a novel mathematical treatment of the fidelity issue for passive sonar TMA, first reported in [1], and the derivation of an iteratively reweighted linear least squares estimation algorithm using the method of expectation-maximization (EM). It is a remarkable fact that this new algorithm does not require computing the gradient of the environmental prediction model in most cases of practical interest.

The idealized TMA problem studied in this paper is called the augmented azimuthal bearings-only TMA problem. Specifically, this is the TMA problem that arises when azimuthal bearing measurements are augmented with (*i*) received signal-to-noise ratio (SNR) measurements and (*ii*) a combined acoustic propagation and sensor (CAPS) signal processing model. This problem is physically unrealistic in that it avoids all discussion of multipath propagation, elevation angle measurements, etc.; however, the problem does have one salient merit — it pushes the fidelity of the acoustic model to the forefront. Thus, the augmented bearings-only TMA problem is well suited to the stated purpose of this paper. The methods proposed here are directly applicable to more realistic and important problems, e.g., TMA using cone angle measurements from linear arrays, but these problems are outside the scope of the present paper.

Clutter is a harsh reality for all TMA problems, that is, target measurements are obscured by other measurements unrelated to the target of interest. Clutter may be due to either random fluctuations in the noise background, or it may be due to interference, e.g., other targets. The difficult problems of correctly assigning measurements to target and of censoring (suppressing) clutter are critically important to successful performance of all TMA algorithms. Measurements of SNR yield improved target state

*This work was supported by the Office of Naval Research.

estimates in the presence of clutter, even in the absence of a detailed CAPS model, because measured SNR is informative for the assignment problem [2]. Investigations of how to incorporate CAPS models of uncertain fidelity into the TMA problem split naturally into two phases. The first phase addresses the use of a CAPS model in clutter-free environments. The critical question to answer in the first phase is "Can a CAPS model reliably improve TMA estimates when no clutter is present?" The second phase addresses using the model in the presence of clutter, so the appropriate question becomes "Can a CAPS model reliably contribute to the joint assignment and TMA estimation problem in cluttered environments?" Neither problem is trivial, and both are important in applications. This paper contributes to the clutter-free first phase of the investigation.

Two formulations of the augmented bearings-only TMA problem are discussed. The formulations use nearly indistinguishable statistical PDF's of the measured bearings, but they differ significantly in their statistical interpretation and in their use of the CAPS model. The first formulation is standard because it assumes measurements are statistically independent, conditioned on target state. Maximum likelihood (ML) target state estimates are defined using a specified target motion model and can be obtained numerically by gradient ascent procedures which require computing the derivatives of the CAPS model with respect to target state variables, or by search procedures (e.g., genetic algorithms) which perform extensive enumeration of possible target state variables.

The second formulation is novel in that it assumes the measurements determine a sequence of statistically independent pre-detection "empirical" PDF's on target state. Multiplying these densities gives the overall empirical density on target state. Empirical maximum *a posteriori* (EMAP) target state estimates are defined by evaluating the empirical density for a specified class of parametric target motion models. EMAP estimates are computed numerically — without taking the gradient of the CAPS model — by the new EM-based algorithm derived in this paper. In contrast to ML estimation, EMAP target estimation requires computing integrals of the CAPS model with respect to a parameterized target density, called herein the geometric kernel. It will be seen that adjusting the down-range variance parameter of the kernel is an intuitive method for compensating for mismatch between the CAPS model and the real world.

Section 2 develops notation and basic probabilistic structures used throughout the paper. Section 3 discusses the ML formulation of the augmented bearings-only TMA problem for constant velocity targets. The EMAP formulation for constant velocity targets is presented in Section 4. Derivation of the EMAP estimation algorithm is given in Section 5. The asymptotic equivalence of the EMAP and ML approaches to the augmented TMA problem is derived in Section 6. Examples are given in Section 7. Summary and concluding remarks are given in Section 8.

2 Notation and Definitions

The radiated sound field of a single point target of unknown source level impinges upon a sensor array, and the sensor signal processor generates an estimate of arrival angle and SNR at the sensor location. Arrival angles are estimates of azimuthal bearings, so the observable target coordinates lie in a horizontal plane. It is assumed that the CAPS model computes the probability density function (PDF) of the measured SNR at the sensor output as a function of signal arrival angle, sensor location, target location, and target source level. Although the functional form of the PDF of the measured SNR is important in the application, it is not important to the theoretical formulations discussed in this paper.

Let (x, y, z) denote target position at an arbitrary, but fixed, time. Azimuthal angles are measured in the x-y plane counter-clockwise from the positive x axis. Let $S(\gamma)$ denote target source level (within the sensor bandwidth) radiated in the azimuthal direction γ when the target bow points in a fixed reference direction, taken here to be the positive x-axis. Target source level also depends on elevation angle, but for simplicity only azimuthal variation is considered here. Let $Z = (\theta, \xi)$ denote the measured bearing θ and the measured SNR ξ of the target when sensor position is $X^o = (x^o, y^o, z^o)$. The target source level s radiating in the direction of the sensor depends on the aspect angle that the target presents to the sensor. If there are no ocean currents, so that the bow points in the direction of motion, simple geometrical considerations in the azimuthal plane gives the conditional density

$$p_{\mathbf{s}|\mathbf{xyzx^oy^oz^o}}(s|x, y, z, x^o, y^o, z^o) = \delta(s - S_{xyx^oy^o}), \tag{2.1}$$

where $\delta(\cdot)$ denotes the Dirac delta function, and

$$S_{xyx^oy^o} \equiv S\left(\tan^{-1}\left[\frac{y - y^o}{x - x^o}\right] - \tan^{-1}\left[\frac{y'}{x'}\right] - \pi\right), \tag{2.2}$$

where x' and y' are the derivatives of x and y with respect to time. The second term in the argument of (2.2) is the true target heading. The factor of π in (2.2) arises because radiated source level $S(\gamma)$ is referenced to the target, not the sensor. The density (2.1) is readily replaced by other PDF's to obtain fully stochastic models of source level; however, the deterministic form (2.1) is adequate for the purposes of this paper.

Both ML and EMAP formulations of the augmented bearings-only TMA problem employ a marginalization approach to incorporate aspect angle and to avoid estimating target depth. Let the likelihood function of the given CAPS model of bearing and SNR be denoted by

$$p_{\theta\xi|\mathbf{xyzsx^oy^oz^o}}(\theta,\xi|x,y,z,s,x^o,y^o,z^o).$$

Application of Bayes Theorem gives

$$\begin{aligned} p_{\theta\xi s|\mathbf{xyzx^oy^oz^o}}(\theta,\xi,s|x,y,z,x^o,y^o,z^o) &= p_{\theta\xi|\mathbf{xyzsx^oy^oz^o}}(\theta,\xi|x,y,z,s,x^o,y^o,z^o) \\ &\quad \times p_{\mathbf{s}|\mathbf{xyzx^oy^oz^o}}(s|x,y,z,x^o,y^o,z^o). \end{aligned} \tag{2.3}$$

Substituting (2.1) into (2.3) and integrating over all s gives

$$p_{\theta\xi|\mathbf{xyzx^oy^oz^o}}(\theta,\xi|x,y,z,x^o,y^o,z^o) \equiv p_{\theta\xi|\mathbf{xyzsx^oy^oz^o}}(\theta,\xi|x,y,z,S_{xyx^oy^o},x^o,y^o,z^o). \tag{2.4}$$

Marginalizing over target depth in the conditional PDF (2.4) gives

$$\begin{aligned} &\int_{z_{\min}}^{z_{\max}} p_{\theta\xi|\mathbf{xyzsx^oy^oz^o}}(\theta,\xi|x,y,z,S_{xyx^oy^o},x^o,y^o,z^o)p_{\mathbf{z}}(z)\,dz \equiv \\ &\qquad p_{\theta\xi|\mathbf{xysx^oy^oz^o}}(\theta,\xi|x,y,S_{xyx^oy^o},x^o,y^o,z^o), \end{aligned} \tag{2.5}$$

where the second term in the integrand denotes the *a priori* target depth PDF, which is zero outside the water column $[z_{\min}, z_{\max}]$ and is assumed known. If the target has a known fixed depth, then the *a priori* target depth density is simply the Dirac delta function located at the given target depth.

The marginal PDF (2.5) is called simply the CAPS density throughout the sequel. In general, the CAPS density is written

$$p_{\mathbf{z}|\mathbf{XX^o}}(Z|X,X^o) \equiv p_{\theta\xi|\mathbf{xysx^oy^oz^o}}(\theta,\xi|x,y,S_{xyx^oy^o},x^o,y^o,z^o), \tag{2.6}$$

where target state is defined by $X = (x, y, s)$. The absence of velocity in the notation for target state is not strictly correct because source level depends on aspect angle; however, this omission leads to no difficulties when used with deterministic aspect angle dependence (2.2) and constant velocity target motion models (see Equation (3.2)).

For later reference, the down-range marginal density of a PDF $f_{\mathbf{xy}}(x,y)$ with respect to the (one-dimensional) Cartesian variables x and y is defined by

$$f_\phi(\phi) \equiv \int_0^\infty f_{\mathbf{xy}}(\rho\cos\phi,\rho\sin\phi)\ \rho\ d\rho, \qquad -\pi < \phi \le \pi, \tag{2.7}$$

where (ρ,ϕ) are the polar coordinates of the target. The PDF $f_{\mathbf{xy}}(x,y)$ is said to be diffuse down-range if

$$f_{\mathbf{xy}}(\rho\cos\phi,\rho\sin\phi)\ \rho\ \propto \begin{cases} f_\phi(\phi), & \rho > 0, \\ 0, & \rho \le 0. \end{cases} \tag{2.8}$$

It follows from (2.8) that a diffuse down-range density is not necessarily diffuse in x and y.

3 Maximum Likelihood Formulation

Let $Z \equiv \{Z_n\}_{n=1}^N \equiv \{\theta_n,\xi_n\}_{n=1}^N$ denote a sequence of independent sensor measurements on the same target obtained at sensor locations $X^o \equiv \{X_n^o\}_{n=1}^N \equiv \{x_n^o,y_n^o,z_n^o\}_{n=1}^N$ and at times $\{t_n\}_{n=1}^N$. The measurements are not required to be identically distributed. Let $X \equiv \{X_n\}_{n=1}^N \equiv \{x_n,y_n,s_n\}_{n=1}^N$ denote the sequence of target states at the measurement times $\{t_n\}_{n=1}^N$. Without loss of generality, it is supposed that $\{t_n\}_{n=1}^N$ are listed in increasing order, that is, $t_n \le t_{n+1}$, $n = 1,\ldots,N-1$.

As is typical in TMA problems, the measurements Z are assumed independent, conditioned on target state. The conditional likelihood function of Z is then given by

$$\mathcal{L}_{\mathbf{Z}|\mathbf{X}\mathbf{X}^o}(Z\,|X,X^o) = \prod_{n=1}^{N} p_{\mathbf{Z}_n|\mathbf{X}_n\mathbf{X}_n^o}(Z_n\,|X_n,X_n^o). \tag{3.1}$$

A deterministic target motion model is specified, so that $x_n = x(t_n)$, $y_n = y(t_n)$, and $s_n = s(t_n)$. The standard target motion model for bearings-only TMA is constant velocity, so

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \frac{t_N - t}{t_N - t_1}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \frac{t - t_1}{t_N - t_1}\begin{bmatrix} x_N \\ y_N \end{bmatrix} \equiv \alpha(t)\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \beta(t)\begin{bmatrix} x_N \\ y_N \end{bmatrix}, \quad t_1 \le t \le t_N. \tag{3.2}$$

The model (3.2) uses end-point parameterization because the position parameters $\{x_1, y_1, x_N, y_N\}$ are dimensionally commensurate (a useful feature if Cramer-Rao lower bounds on estimation error are compared); however, the end-point model is mathematically equivalent to the more common position-velocity model. Substituting the velocity implicit in the end-point parameterization (3.2) into the aspect angle dependent source level function (2.2) gives

$$s(t_n) = s_n \equiv S(\gamma_n), \qquad 1 \le n \le N, \tag{3.3}$$

where the source level reference angle is given by

$$\gamma_n \equiv \tan^{-1}\left[\frac{y_n - y_n^o}{x_n - x_n^o}\right] - \tan^{-1}\left[\frac{y_N - y_1}{x_N - x_1}\right] - \pi. \tag{3.4}$$

Target state is therefore fully parameterized by $\lambda = \{x_1, y_1, x_N, y_N\}$, so that

$$X = X(\lambda) = \{x(\lambda), y(\lambda), s(\lambda)\}.$$

Hence, the target state estimate is determined from the ML parameter estimate

$$\widehat{\lambda}_{\mathrm{ML}} = \arg\max_{\lambda} \mathcal{L}_{\mathbf{Z}|\mathbf{X}\mathbf{X}^o}(Z\,|X(\lambda),X^o). \tag{3.5}$$

Taking the gradient of the conditional likelihood function with respect to λ and setting the result to zero gives the necessary conditions to be solved by appropriate numerical procedures.

A potentially serious difficulty with using the necessary ML conditions is that the gradient of the CAPS model is required with respect to the target parameter vector λ. Because the gradient is typically required at each iteration of a numerical procedure, ML estimates are often difficult and time consuming to compute. Also, the use of gradients may exacerbate any underlying CAPS model infidelity.

A maximum *a posteriori* (MAP) formulation can be obtained from the ML formulation by incorporating an appropriate *a priori* density on the end-point parameters of the constant velocity target model. Alternatively, a target process noise model can be included to compensate for target maneuvers. MAP estimators are not pursued further in this paper; however, the EMAP formulation presented in the next section is readily adapted to either of these approaches to MAP estimation.

Marginalizing (3.1) over the SNR measurements $Z_\xi \equiv \{\xi_n\}_{n=1}^N$ gives the (non-augmented) bearings-only TMA likelihood function

$$\mathcal{L}_{\mathbf{Z}_\theta|\mathbf{X}\mathbf{X}^o}(Z_\theta\,|X,X^o) = \prod_{n=1}^{N} p_{\theta_n|\mathbf{X}_n\mathbf{X}_n^o}(\theta_n\,|X_n,X_n^o), \tag{3.6}$$

where $Z_\theta \equiv \{\theta_n\}_{n=1}^N$. The traditional bearings-only TMA problem is obtained from (3.6) by assuming straight line propagation in the azimuthal plane and by omitting source level. In this special case, simple geometry and additive Gaussian noise assumptions give

$$p_{\theta_n|\mathbf{X}_n\mathbf{X}_n^o}(\theta_n\,|X_n,X_n^o) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp\left\{-\frac{1}{2\sigma_n^2}\left[\theta_n - \tan^{-1}\left(\frac{y_n - y_n^o}{x_n - x_n^o}\right)\right]^2\right\}, \tag{3.7}$$

where σ_n denotes the standard deviation of θ_n. Substituting (3.7) into (3.6) and taking the natural logarithm gives the usual σ_n^{-2}-weighted nonlinear least squares problem of traditional TMA.

4 Empirical MAP Formulation

Traditional TMA statistical models are post-detection models, that is, they assume *a priori* that the measurements Z belong to a common target having a specified parametric form (e.g., equations (3.2)–(3.4)). Post-detection tracking implies that measurements are independent if they are conditioned on the target. ML estimators thus answer the question "Given data generated from a target track, which parameterized track best fits the data?"

In contrast, the EMAP estimators proposed here differ fundamentally from traditional post-detection TMA because they are joint detection-estimation methods which seek to answer the alternative question "Does a target track of the specified parametric form fit the data?" A generalized likelihood ratio test (GLRT) in which track parameters are estimated and substituted into a likelihood ratio is the EMAP answer to the question; however, it is the estimated track — and not the GLRT detector — which is the object of interest in this paper.

The data $\{(Z_n, X_n^o)\}_{n=1}^N$ are assumed statistically independent because measurements are not specified *a priori* to belong to the same track. Independence implies that

$$p_{\mathbf{Z}\mathbf{X}^o}(Z, X^o) = \prod_{n=1}^{N} p_{\mathbf{Z}_n\mathbf{X}_n^o}(Z_n, X_n^o). \tag{4.1}$$

The data $\{(Z_n, X_n^o)\}_{n=1}^N$ contribute independent probability density assessments of "potential" target position and source level that are valid at the times at which the measurements are obtained. Let $X^\Omega \equiv \{X_n^\Omega\}_{n=1}^N \equiv \{x_n^\Omega, y_n^\Omega, s_n^\Omega\}_{n=1}^N$ denote so-called "empirical" random variables associated with potential locations and source levels. Empirical random variables are assumed independent when conditioned on their corresponding measurements and sensor locations; hence, the empirical target location PDF for the full data set is

$$p_{\mathbf{X}^\Omega|\mathbf{Z}\mathbf{X}^o}(X^\Omega|Z, X^o) = \prod_{n=1}^{N} p_{\mathbf{X}_n^\Omega|\mathbf{Z}_n\mathbf{X}_n^o}(X_n^\Omega|Z_n, X_n^o). \tag{4.2}$$

The empirical target likelihood function (4.2) is evaluated for specified parametric target motion models, once the conditional density of X_n^Ω is defined.

Full target state is not observable from a single bearing measurement. Consequently, the dummy random variable r_n is introduced to model the "missing" sensor range measurement corresponding to θ_n, and the density of the empirical variable X_n^Ω is expressed as a marginal density over r_n. Using Bayes Theorem, the marginal density is written in the form

$$p_{\mathbf{X}_n^\Omega|\mathbf{Z}_n\mathbf{X}_n^o}(X_n^\Omega|Z_n, X_n^o) = \int_0^\infty p_{\mathbf{x}_n^\Omega\mathbf{y}_n^\Omega|\mathbf{r}_n\mathbf{s}_n^\Omega\mathbf{Z}_n\mathbf{X}_n^o}(x_n^\Omega, y_n^\Omega|r_n, s_n^\Omega, Z_n, X_n^o)\, p_{\mathbf{r}_n\mathbf{s}_n^\Omega|\mathbf{Z}_n\mathbf{X}_n^o}(r_n, s_n^\Omega|Z_n, X_n^o)\, dr_n. \tag{4.3}$$

The statistical relationship between the missing range measurement r_n and the other random variables must be defined. Several applications of Bayes Theorem gives the identity

$$\begin{aligned} p_{\mathbf{r}_n\mathbf{s}_n^\Omega|\mathbf{Z}_n\mathbf{X}_n^o}(r_n, s_n^\Omega|Z_n, X_n^o) &= p_{\xi_n|\mathbf{r}_n\theta_n\mathbf{s}_n^\Omega\mathbf{X}_n^o}(\xi_n|r_n, \theta_n, s_n^\Omega, X_n^o) \\ &\quad \times \frac{p_{\mathbf{r}_n|\theta_n\mathbf{X}_n^o}(r_n|\theta_n, X_n^o)}{p_{\xi_n|\theta_n\mathbf{X}_n^o}(\xi_n|\theta_n, X_n^o)}\, p_{\mathbf{s}_n^\Omega|\mathbf{r}_n\theta_n\mathbf{X}_n^o}(s_n^\Omega|r_n, \theta_n, X_n^o). \end{aligned} \tag{4.4}$$

The last term in (4.4) is defined by (*cf.* Equation (2.1))

$$p_{\mathbf{s}_n^\Omega|\mathbf{r}_n\theta_n\mathbf{X}_n^o}(s_n^\Omega|r_n, \theta_n, X_n^o) = \delta(s_n^\Omega - S_{r_n\theta_n}), \tag{4.5}$$

where

$$S_{r_n\theta_n} \equiv S\left(\theta_n - \tan^{-1}\left[\frac{(r_n\sin\theta_n)'}{(r_n\cos\theta_n)'}\right] - \pi\right), \tag{4.6}$$

and where primes in (4.6) denote derivatives with respect to time. Substituting (4.4)–(4.6) into (4.3), marginalizing over empirical source level, and using Bayes Theorem in the purely data dependent de-

nominator in (4.4) gives

$$p_{\mathbf{x}_n^\Omega \mathbf{y}_n^\Omega | \mathbf{Z}_n \mathbf{X}_n^o}\left(x_n^\Omega, y_n^\Omega \,|Z_n, X_n^o\right) = \frac{p_{\theta_n|\mathbf{X}_n^o}(\theta_n|X_n^o)}{p_{\theta_n\xi_n|\mathbf{X}_n^o}(\theta_n,\xi_n|X_n^o)}$$
$$\int_0^\infty p_{\mathbf{x}_n^\Omega \mathbf{y}_n^\Omega | \mathbf{r}_n \mathbf{s}_n^\Omega \mathbf{Z}_n \mathbf{X}_n^o}\left(x_n^\Omega, y_n^\Omega \,|r_n, S_{r_n\theta_n}, Z_n, X_n^o\right)$$
$$\times p_{\xi_n|\mathbf{r}_n\theta_n\mathbf{s}_n^\Omega\mathbf{X}_n^o}\left(\xi_n\,|r_n,\theta_n,S_{r_n\theta_n},X_n^o\right)\; p_{\mathbf{r}_n|\theta_n\mathbf{X}_n^o}\left(r_n\,|\theta_n,X_n^o\right)\; dr_n. \qquad (4.7)$$

Substituting the integral (4.7) into (4.2) gives the overall empirical PDF as a product of integrals. If the measurement ξ_n is unavailable, it is easy to show that the appropriate integral is identical to (4.7), but with the middle term omitted; thus, data sets in which measured SNR is only intermittently available are easily accommodated.

Each term in the integrand of the integral representation (4.7) has a meaningful physical interpretation. The first term, called the geometric kernel, is a density on empirical target position, and it is conditioned on range, bearing, source level, measured SNR, and sensor position. For augmented bearings-only TMA problems, the geometric kernel is assumed to be a bivariate Gaussian whose mean vector and covariance matrix are determined by the conditioning variables. The kernel's mean vector is determined by range, bearing, and sensor location; the kernel's covariance matrix is a joint function of all the conditioning variables and is determined, in principle, via the CAPS model. Purely geometric considerations for the special case when the sensor lies at the origin) gives the kernel in the form

$$p_{\mathbf{x}_n^\Omega \mathbf{y}_n^\Omega | \mathbf{r}_n \mathbf{s}_n^\Omega \mathbf{Z}_n \mathbf{X}_n^o}\left(x_n^\Omega, y_n^\Omega \,|r_n, s_n^\Omega, Z_n, X_n^o\right)$$
$$= \frac{1}{2\pi\sigma_n\kappa_n r_n^2}\exp\left\{\frac{-1}{2r_n^2}\begin{bmatrix} x_n^\Omega - x_n^o - r_n\cos\theta_n \\ y_n^\Omega - y_n^o - r_n\sin\theta_n \end{bmatrix}' \Lambda(\theta_n,\sigma_n,\kappa_n)\begin{bmatrix} x_n^\Omega - x_n^o - r_n\cos\theta_n \\ y_n^\Omega - y_n^o - r_n\sin\theta_n \end{bmatrix}\right\} \qquad (4.8)$$

where $\Lambda(\theta_n,\sigma_n,\kappa_n)$ is the inverse covariance matrix (see Appendix A of [1]). Loosely speaking, the cross-range and down-range variances of the geometric kernel are determined by the sensor and the environment, respectively, together with square law azimuthal dispersion of empirical target location.

The standard deviation of the measured bearing θ_n is denoted by $\sigma_n \equiv \sigma(\theta_n,\xi_n,s_n,X_n^o)$, and it is determined by sensor signal processing considerations. The bearing variance $\sigma^2(\cdot)$ may be constant, but in general it depends parametrically on bearing because of beamwidth equalization issues (i.e., some beams may be narrower than others), on measured SNR and target source level because measurement variance typically depends on SNR, and on sensor location because the CAPS model may depend on absolute (not relative) sensor location.

The standard deviation of the missing range measurement r_n, denoted by $\kappa_n \equiv \kappa(\theta_n,\xi_n,s_n,X_n^o)$ in (4.8), is determined by the CAPS model. The down-range variance $\kappa^2(\cdot)$ may depend parametrically on bearing because the CAPS model may not be azimuthally symmetric, on measured SNR and source level because they influence down-range estimation error, and on sensor location because the CAPS model may be location sensitive.

The down-range variance κ_n^2 of the geometric kernel provides a new and potentially useful parameter that can be used to compensate for mismatch between the CAPS model and the real world. Using a down-range variance greater than κ_n^2 will decrease the CAPS model influence on TMA estimates and, conversely, using a smaller variance will increase its influence. This opens the possibility in practice of using what may be called a "monotone" simulated annealing scheme in which the down-range variance is initially made too large, and is then monotonically reduced in stages to the correct theoretical level implied by the CAPS model, namely κ_n^2. When CAPS modeling fidelity is considered a significant issue, it may be reduced only to the level thought reliable in the application. Such inflation/deflation procedures might also be employed whenever an explicit form for κ_n^2 is not readily obtained from the CAPS model.

The second density in the integrand of (4.7) is contributed by the CAPS model, which in general depends on the absolute location of the sensor in the ocean, i.e., azimuthal dependence of the propagation model can vary with sensor location. Simpler CAPS models may be used to reduce computational complexity if desired. If a deterministic CAPS model is available, it may be worthwhile — depending on the application — to use it as the mean of a Gaussian PDF with a suitably specified range and bearing dependent variance [3]. Such statistical models must be used with caution when received signal levels less than a certain threshold level cannot be reported in association with a bearing measurement. Truncation and renormalization of the Gaussian densities may be necessary to compensate for this effect.

The third density in the integrand of (4.7) specifies the sensor measurement window for the missing range measurement. It is derived by assuming that the *a priori* joint density of the measurement pair

(r_n, θ_n) corresponds to a uniformly distributed point over a feasible region $\mathcal{R}(X_n^o)$ of the x-y plane whose inner and outer radii are given by the radial functions $r_{\min}(\theta_n, X_n^o)$ and $r_{\max}(\theta_n, X_n^o)$, respectively. The outer radius may be interpreted as the maximum range at which signals of specified (maximum) source level are detected with specified probability P_d. Similarly, the inner radius may be interpreted as the near-field limit of the sensor, i.e., the minimum range at which the sensor's beamformer reliably estimates bearings. Thus, in polar coordinates, the joint density of (r_n, θ_n) is given by

$$p_{\mathbf{r}_n\theta_n|\mathbf{X}_n^o}(r_n, \theta_n|X_n^o) = \begin{cases} \mathcal{A}(X_n^o)\, r_n, & \text{for } (r_n, \theta_n) \in \mathcal{R}(X_n^o), \\ 0, & \text{otherwise,} \end{cases}$$

where $\mathcal{A}(X_n^o)$ is the reciprocal of the area of the feasible region $\mathcal{R}(X_n^o)$. Conditioning on bearing as well as sensor location gives, using Bayes Theorem,

$$p_{\mathbf{r}_n|\theta_n\mathbf{X}_n^o}(r_n|\theta_n, X_n^o) = \begin{cases} \varepsilon(\theta_n, X_n^o)\, r_n, & \text{for } r_{\min}(\theta_n, X_n^o) \le r_n \le r_{\max}(\theta_n, X_n^o), \\ 0, & \text{otherwise,} \end{cases} \tag{4.9}$$

where the normalization constant is $\varepsilon(\theta_n, X_n^o) = 2/(r_{\max}^2(\theta_n, X_n^o) - r_{\min}^2(\theta_n, X_n^o))$. The *a priori* bearing density $p_{\theta_n|\mathbf{X}_n^o}(\theta_n|X_n^o)$ is uniformly distributed if and only if the normalization constant $\varepsilon(\theta_n, X_n^o)$ is independent of θ_n.

The integral representation appropriate for the augmented bearings-only TMA problem is obtained from the general expression (4.7) by substituting the specific forms (4.8) and (4.9). The result is

$$p_{\mathbf{x}_n^\Omega\mathbf{y}_n^\Omega|\mathbf{Z}_n\mathbf{X}_n^o}(x_n^\Omega, y_n^\Omega|Z_n, X_n^o) = c_n(\theta_n, \xi_n, s_n, X_n^o) \int_{r_{\min}(\theta_n, X_n^o)}^{r_{\max}(\theta_n, X_n^o)} p_{\xi_n|\mathbf{r}_n\theta_n\mathbf{s}_n^\Omega\mathbf{X}_n^o}(\xi_n|r_n, \theta_n, s_n^\Omega, X_n^o)$$
$$\times \exp\left\{ \frac{-1}{2r_n^2} \begin{bmatrix} x_n^\Omega - x_n^o - r_n\cos\theta_n \\ y_n^\Omega - y_n^o - r_n\sin\theta_n \end{bmatrix}' \Lambda(\theta_n, \sigma_n, \kappa_n) \begin{bmatrix} x_n^\Omega - x_n^o - r_n\cos\theta_n \\ y_n^\Omega - y_n^o - r_n\sin\theta_n \end{bmatrix} \right\} \frac{dr_n}{r_n}, \tag{4.10}$$

where the proportionality factor is given by

$$c_n(\theta_n, \xi_n, s_n, X_n^o) = \frac{\varepsilon(\theta_n, X_n^o)}{2\pi\sigma(\theta_n, \xi_n, s_n, X_n^o)\kappa(\theta_n, \xi_n, s_n, X_n^o)} \frac{p_{\theta_n|\mathbf{X}_n^o}(\theta_n|X_n^o)}{p_{\theta_n\xi_n|\mathbf{X}_n^o}(\theta_n, \xi_n|X_n^o)}. \tag{4.11}$$

The integral representation (4.10) is fundamental to the formulation of the EMAP likelihood function. When the geometric kernel is independent of the CAPS model, the empirical density on the left hand side of (4.10) is the transform of the CAPS density; the kernel of this non-standard function transformation is the bivariate Gaussian density called herein the geometric kernel.

Appendix B of [1] shows that the down-range marginal density of the integral (4.10) is closely approximated by the Gaussian distribution, provided the bearing measurement standard deviation σ_n is small, say on the order of several degrees or less. This is an important analytical result because it confirms that bearing measurement variance is not a function of the CAPS model. The result also shows (with appropriate use of diffuse priors) that the traditional bearings-only TMA problem is recovered from (4.10) via down-range marginalization. The standard ML approach (see Section 3) to the augmented bearings-only TMA problem is obtained asymptotically as $\kappa \to 0$, as shown below in Section 6.

Let $\lambda = \{x_1, y_1, x_N, y_N\}$, just as in the ML formulation of Section 3. Substituting the target model parameterization (3.2)–(3.4) into the integral representation (4.10), and then substituting the result into the likelihood function (4.2) gives

$$p_{\mathbf{x}^\Omega\mathbf{y}^\Omega|\mathbf{Z}\mathbf{X}^o}(x(\lambda), y(\lambda)|Z, X^o) = \left\{ \prod_{n=1}^N c_n(\theta_n, \xi_n, S_{\lambda\theta_n}, X_n^o) \right\}$$
$$\times \left\{ \prod_{n=1}^N \int_{r_{\min}(\theta_n, X_n^o)}^{r_{\max}(\theta_n, X_n^o)} p_{\xi_n|\mathbf{r}_n\theta_n\mathbf{s}_n^\Omega\mathbf{X}_n^o}(\xi_n|r_n, \theta_n, S_{\lambda\theta_n}, X_n^o) \exp\left[\frac{-Q_n(r_n;\lambda)}{2r_n^2} \right] \frac{dr_n}{r_n} \right\}, \tag{4.12}$$

where the quadratic form $Q_n \equiv Q_n(r_n;\lambda)$ of the exponential term is

$$Q_n = \begin{bmatrix} \alpha(t_n)x_1 + \beta(t_n)x_N - x_n^o - r_n\cos\theta_n \\ \alpha(t_n)y_1 + \beta(t_n)y_N - y_n^o - r_n\sin\theta_n \end{bmatrix}' \Lambda(\theta_n, \sigma_n, \kappa_n) \begin{bmatrix} \alpha(t_n)x_1 + \beta(t_n)x_N - x_n^o - r_n\cos\theta_n \\ \alpha(t_n)y_1 + \beta(t_n)y_N - y_n^o - r_n\sin\theta_n \end{bmatrix}, \tag{4.13}$$

and source level as a function of aspect angle is, using (4.6) for constant velocity target motion,

$$S_{\lambda\theta_n} \equiv S\left(\theta_n - \tan^{-1}\left[\frac{y_N - y_1}{x_N - x_1}\right] - \pi\right), \tag{4.14}$$

and where the bearing and down-range standard deviations of the geometric kernel are given by

$$\begin{aligned} \sigma_n &\equiv \sigma(\theta_n, \xi_n, S_{\lambda\theta_n}, X_n^o), \\ \kappa_n &\equiv \kappa(\theta_n, \xi_n, S_{\lambda\theta_n}, X_n^o). \end{aligned} \tag{4.15}$$

The forms of $\sigma(\cdot)$ and $\kappa(\cdot)$ are derived from the sensor and CAPS models; their target range dependence in (4.15) arises via target parameterization.

The likelihood function (4.12) yields the EMAP parameter estimate

$$\widehat{\lambda}_{\mathrm{EMAP}} = \arg\max_{\lambda}\ p_{\mathbf{x}^\Omega \mathbf{y}^\Omega | \mathbf{Z}\mathbf{X}^\circ}(x(\lambda), y(\lambda) | Z, X^o). \tag{4.16}$$

The standard necessary conditions for the EMAP estimate (4.16) are found by setting the gradient of the posterior likelihood (4.12) with respect to the position parameters λ to zero. It is a remarkable fact that the EMAP necessary conditions do not require computing the gradient of the CAPS model if target source level is independent of aspect angle, that is, if $S(\gamma)$ is independent of γ, and if CAPS fidelity issues are addressed by the down-range variance inflation/deflation strategy discussed above.

5 Derivation and Statement of EMAP Algorithm

A general EMAP estimation algorithm is derived in this section using the method of EM. Familiarity with the method of EM is assumed in this section. General discussions of the method are widely available; for a general introduction, see [4, 5, 7]. For applications of the EM method specifically to Gaussian mixtures, see [6, 8, 9]. The EMAP algorithm is derived for constant velocity target motion; however, the derivation is very general and is easily extended to more general models.

The EMAP algorithm may be derived by discretizing the integrals in the likelihood function (4.12); however, discretization needlessly obscures the discussion. Instead, integrals are retained in the following derivation. The objective of the E-step is to define the so-called auxiliary function of the EM method and to simplify it if possible. The auxiliary function depends on two sets of target end-point parameter vectors, $\lambda' = \{x_1', y_1', x_N', y_N'\}$ and $\lambda = \{x_1, y_1, x_N, y_N\}$, where λ' is an initial (given) estimate and λ is arbitrary. The terms of the auxiliary function that are functions of λ are, using the quadratic form $Q_n(r_n; \lambda)$ defined by equation (4.13),

$$\Psi(\lambda) = -\frac{1}{2}\Psi_{\mathrm{MSE}}(\lambda) + \Psi_{\mathrm{CAPS}}(\lambda) - \Psi_{\mathrm{VAR}}(\lambda), \tag{5.1}$$

where

$$\Psi_{\mathrm{MSE}}(\lambda) = \sum_{n=1}^{N} \int_{r_{\min}(\theta_n, X_n^o)}^{r_{\max}(\theta_n, X_n^o)} w_{\theta_n \xi_n}(r_n; \lambda')\ Q_n(r_n; \lambda)\ \frac{dr_n}{r_n^2} \tag{5.2}$$

is a weighted mean squared error,

$$\Psi_{\mathrm{CAPS}}(\lambda) = \sum_{n=1}^{N} \int_{r_{\min}(\theta_n, X_n^o)}^{r_{\max}(\theta_n, X_n^o)} w_{\theta_n \xi_n}(r_n; \lambda')\ \log\left[p_{\xi_n | r_n \theta_n \mathbf{s}_n^\Omega \mathbf{X}_n^\circ}(\xi_n | r_n, \theta_n, S_{\lambda\theta_n}, X_n^o)\right]\ dr_n \tag{5.3}$$

is a contribution due to target source level dependence on aspect angle,

$$\Psi_{\mathrm{VAR}}(\lambda) = \sum_{n=1}^{N} \int_{r_{\min}(\theta_n, X_n^o)}^{r_{\max}(\theta_n, X_n^o)} w_{\theta_n \xi_n}(r_n; \lambda')\ \log\left[\sigma(\theta_n, \xi_n, S_{\lambda\theta_n}, X_n^o)\kappa(\theta_n, \xi_n, S_{\lambda\theta_n}, X_n^o)\right]\ dr_n \tag{5.4}$$

is a contribution due to variance dependence on target source level, and where the weights in (5.2)–(5.4) are given by the (Bayesian) ratio

$$w_{\theta_n\xi_n}(r_n;\lambda') = \frac{p_{\xi_n|r_n\theta_n s_n^\Omega \mathbf{X}_n^o}(\xi_n|r_n,\theta_n,S_{\lambda'\theta_n},X_n^o)\exp\left\{\frac{-Q_n(r_n;\lambda')}{2r_n^2}\right\}\frac{1}{r_n}}{\int\limits_{r_{\min(\theta_n,X_n^o)}}^{r_{\max(\theta_n,X_n^o)}} p_{\xi_n|r_n\theta_n s_n^\Omega \mathbf{X}_n^o}(\xi_n|r_n,\theta_n,S_{\lambda'\theta_n},X_n^o)\exp\left\{\frac{-Q_n(r_n;\lambda')}{2r_n^2}\right\}\frac{dr_n}{r_n}}. \tag{5.5}$$

Further details of this important but tedious step are straightforward, given familiarity with the application of EM to Gaussian mixtures, and are omitted.

The weights (5.5) are nonnegative, so it is evident from (5.2) that $\Psi_{\mathrm{MSE}}(\lambda)$ is a nonnegative quadratic function of λ. If target source level is independent of aspect angle, the term $\Psi_{\mathrm{CAPS}}(\lambda)$ is omitted from the auxiliary function (5.1) because it is independent of the parameter vector. Similarly, the term $\Psi_{\mathrm{VAR}}(\lambda)$ is omitted if the variances of the geometric kernel are independent of the CAPS model.

The objective of the M-step is to maximize the auxiliary function as a function of the parameter vector λ. Consider first the important special case where the auxiliary function comprises only the weighted squared error term, $\Psi_{\mathrm{MSE}}(\lambda)$. Let the time dependent matrix $H(t)$ be given by the 2×4 matrix

$$H(t) = \begin{bmatrix} \alpha(t) & \beta(t) & 0 & 0 \\ 0 & 0 & \alpha(t) & \beta(t) \end{bmatrix} \tag{5.6}$$

(*cf.* Equation (3.2)). The weights (5.5) depend on the initial parameter vector λ', but not on λ; hence, setting the gradient of $\Psi(\lambda)$ with respect to λ to zero and solving for λ gives the updated parameter estimate

$$\lambda^+ \equiv \arg\max_\lambda \Psi(\lambda) \equiv \arg\min_\lambda \Psi_{\mathrm{MSE}}(\lambda) = \left(\sum_{n=1}^N A_n(\theta_n,\xi_n)\right)^{-1}\left(\sum_{n=1}^N b_n(\theta_n,\xi_n)\right), \tag{5.7}$$

where $A_n(\theta_n,\xi_n)$ is the 4×4 matrix given by

$$A_n(\theta_n,\xi_n) \equiv H'(t_n)\Lambda(\theta_n,\sigma_n,\kappa_n)H(t_n)\int\limits_{r_{\min(\theta_n,X_n^o)}}^{r_{\max(\theta_n,X_n^o)}} w_{\theta_n\xi_n}(r_n;\lambda')\,\frac{dr_n}{r_n^2}, \tag{5.8}$$

and $b_n(\theta_n,\xi_n)$ is a vector of length 4 given by

$$b_n(\theta_n,\xi_n) \equiv H'(t_n)\Lambda(\theta_n,\sigma_n,\kappa_n)\int\limits_{r_{\min(\theta_n,X_n^o)}}^{r_{\max(\theta_n,X_n^o)}} w_{\theta_n\xi_n}(r_n;\lambda')\begin{bmatrix} x_n^o + r_n\cos\theta_n \\ y_n^o + r_n\sin\theta_n \end{bmatrix}\frac{dr_n}{r_n^2}. \tag{5.9}$$

It is evident that the matrices given in (5.8) have rank at most 2, so the matrix whose inverse is required in (5.7) attains full rank if only if the target is observable (in the statistical sense) from the measured data set. Observability questions are widely discussed in bearings-only TMA problems, but lie outside the intended scope of this paper.

The EMAP algorithm is an iteratively reweighted linear least squares algorithm for the special case when only the weighted squared error term need be retained. Explicitly, the algorithm for this special case takes the following recursive form: Let S_0 denote the given, or assumed, target source level, and let $\sigma_n \equiv \sigma(\theta_n,\xi_n,S_0,X_n^o)$ and $\kappa_n \equiv \kappa(\theta_n,\xi_n,S_0,X_n^o)$ for $1 \le n \le N$. Initialize the target end-point parameter

$$\lambda^{(0)} = \left(x_1^{(0)}, y_1^{(0)}, x_N^{(0)}, y_N^{(0)}\right),$$

and set $k = 0$. For $k \ge 0$, define the unnormalized weight function

$$\varpi_{\theta_n\xi_n}(r_n;\lambda^{(k)}) = p_{\xi_n|r_n\theta_n s_n^\Omega \mathbf{X}_n^o}(\xi_n|r_n,\theta_n,S_0,X_n^o)\exp\left\{\frac{-Q_n(r_n;\lambda^{(k)})}{2r_n^2}\right\}, \tag{5.10}$$

where the quadratic form is defined by (4.13). Compute the $3N$ one-dimensional integrals

$$d_n^{(k)}(\ell) = \int\limits_{r_{\min(\theta_n,X_n^o)}}^{r_{\max(\theta_n,X_n^o)}} \varpi_{\theta_n\xi_n}(r_n;\lambda^{(k)})\,\frac{dr_n}{r_n^\ell}, \qquad 1 \le n \le N, \quad \ell = 1,2,3. \tag{5.11}$$

Using the integrals (5.11), compute the 4×4 matrix

$$A^{(k)} \equiv \sum_{n=1}^{N} \frac{d_n^{(k)}(3)}{d_n^{(k)}(1)} H'(t_n)\Lambda(\theta_n, \sigma_n, \kappa_n)H(t_n) \tag{5.12}$$

and the length 4 vector

$$b^{(k)} \equiv \sum_{n=1}^{N} H'(t_n)\Lambda(\theta_n, \sigma_n, \kappa_n) \left(\frac{d_n^{(k)}(3)}{d_n^{(k)}(1)} \begin{bmatrix} x_n^o \\ y_n^o \end{bmatrix} + \frac{d_n^{(k)}(2)}{d_n^{(k)}(1)} \begin{bmatrix} \cos\theta_n \\ \sin\theta_n \end{bmatrix} \right). \tag{5.13}$$

Finally, compute the updated parameter vector

$$\lambda^{(k+1)} = \left(A^{(k)}\right)^{-1} b^{(k)}. \tag{5.14}$$

Linear least squares problems such as (5.7) are in practice best solved by reliable methods of numerical matrix analysis, instead of using the normal equations (5.14). Details are left to the reader.

The M-step cannot neglect the terms $\Psi_{\text{CAPS}}(\lambda)$ and $\Psi_{\text{VAR}}(\lambda)$ in the general case when target source level is dependent on aspect angle, or when the down-range and cross-range variances of the geometric kernel are dependent on the CAPS model. Solving for the updated parameter vector in the M-step requires solving the following problem:

$$\lambda^{+} \equiv \arg\max_{\lambda} \Psi(\lambda) \equiv \arg\min_{\lambda} \left\{ \Psi_{\text{MSE}}(\lambda) - 2\Psi_{\text{CAPS}}(\lambda) + 2\Psi_{\text{VAR}}(\lambda) \right\}. \tag{5.15}$$

The minimization problem (5.15) is evidently a nonlinear penalized least squares problem and must be solved by numerical methods such as Gauss-Newton. Unfortunately, evaluating the gradient of the objective function in (5.15) requires taking the gradient of the CAPS model, so the EMAP algorithm for the general case requires solving a difficult numerical problem at each step of the iteration, and it is thus comparable in computational complexity to algorithms based on the ML approach discussed in Section 3. Whether or not the penalized least squares formulation (5.15) can be exploited to advantage in the application is an open question.

6 ML Approach as the Asymptotic Limit of EMAP

Laplace's method for obtaining asymptotic expansions of integrals is used in this section to reveal the close relationship between the EMAP integral representation (4.10) and the standard Gaussian bearing error expression (3.7). Consider the generic Laplace-type integral

$$I(\lambda) = \int_a^b e^{-\lambda\phi(t)} f(t)\, dt, \tag{6.1}$$

where $\phi(t)$ is such that its absolute minimum on the interval $[a, b]$ occurs at the point $t = t_0$, where $a < t_0 < b$, $\phi'(t_0) = 0$, and $\phi''(t_0) > 0$. It is assumed that $f(t)$ has at least 2 and $\phi(t)$ has at least 4 continuous derivatives on $[a, b]$. The classical Laplace asymptotic expression, given by

$$I(\lambda) = \sqrt{\frac{2\pi}{\lambda\phi''(t_0)}}\, f(t_0)\, e^{-\lambda\phi(t_0)} + o\left(\frac{e^{-\lambda\phi(t_0)}}{\lambda^{\frac{3}{2}}}\right), \tag{6.2}$$

holds as $\lambda \to \infty$. It is derived rigorously using Watson's lemma in [10]. Expression (6.2) is merely the first term in an asymptotic expansion of the integral (6.1). The next term is not given in [10], but can be derived following their method (after fixing a minor numerical error in a coefficient). Omitting the tedious details, the result is

$$I(\lambda) = e^{-\lambda\phi(t_0)} \left\{ \sqrt{2\pi}\, f(t_0)\, \left(\lambda\phi''(t_0)\right)^{-\frac{1}{2}} + \sqrt{\pi}\, C\, \left(\lambda\phi''(t_0)\right)^{-\frac{3}{2}} \right\} + o\left(\frac{e^{-\lambda\phi(t_0)}}{\lambda^{\frac{5}{2}}}\right), \tag{6.3}$$

where the coefficient C is given by

$$C = f(t_0) \left[\frac{1}{2} \left(\frac{\phi'''(t_0)}{\phi''(t_0)} \right)^2 - \frac{1}{3} \frac{\phi^{\text{IV}}(t_0)}{\phi''(t_0)} \right] - f'(t_0) \frac{\phi'''(t_0)}{\phi''(t_0)} + f''(t_0). \tag{6.4}$$

These asymptotic results are applied to integrals of the form (4.10) and (5.11).

The particular integrals of interest in this paper take the form

$$J_\ell(\kappa) = \frac{c_n}{2\pi\,\sigma\kappa}\int_{r_1}^{r_2} p(r)\; r^{-\ell}\exp\left\{-\frac{a(r)}{\kappa^2}-\frac{b(r)}{\sigma^2}\right\}\,dr, \quad \ell = 1,2,3, \tag{6.5}$$

where

$$\begin{aligned} a(r) &= \frac{1}{2}\left[1-\frac{x\cos\theta+y\sin\theta}{r}\right]^2 \\ b(r) &= \frac{1}{2}\left[\frac{x\sin\theta-y\cos\theta}{r}\right]^2. \end{aligned} \tag{6.6}$$

The special case $J_1(\kappa)$ is equivalent to (4.10), as is seen by letting

$$\begin{aligned} r &= r_n, \quad r_1 = r_{\min}(\theta_n, X_n^o), \quad r_2 = r_{\max}(\theta_n, X_n^o), \\ x &= x_n^\Omega - x_n^o, \quad y = y_n^\Omega - y_n^o, \\ \kappa &= \kappa_n = \kappa(\theta_n,\xi_n,s_n,X_n^o), \quad \sigma = \sigma_n = \sigma(\theta_n,\xi_n,s_n,X_n^o), \\ c_n &= 2\pi\;\sigma(\theta_n,\xi_n,s_n,X_n^o)\;\kappa(\theta_n,\xi_n,s_n,X_n^o)\;c_n(\theta_n,\xi_n,s_n,X_n^o), \\ p(r) &= p_{\xi_n|r_n\theta_n s_n^\Omega X_n^o}(\xi_n|r_n,\theta_n,s_n^\Omega,X_n^o), \end{aligned}$$

and by algebraically manipulating the quadratic form. The asymptotic form of (6.5) is sought as $\kappa\to 0$; hence, κ^{-2} plays the role of λ and $a(r)$ plays the role of $\phi(t)$ in (6.1). The necessary condition for the minimum of $a(r)$ is that its derivative be zero. The unique root of $a'(r)=0$ is

$$r_0 = x\cos\theta + y\sin\theta. \tag{6.7}$$

It is assumed that r_0 is interior to the range of integration in (6.5). The second order condition is also satisfied, that is,

$$\left.\frac{d^2a(r)}{dr^2}\right|_{r=r_0} = \frac{1}{(x\cos\theta+y\sin\theta)^2} > 0. \tag{6.8}$$

The function

$$g(r) = \frac{c_n}{2\pi\sigma\kappa}\;p(r)\;r^{-\ell}\;\exp\left\{-\frac{b(r)}{\sigma^2}\right\}$$

plays the role of $f(t)$ in (6.1). Using (6.2) and the fact that $a(r_0)=0$ gives the asymptotic result

$$J_\ell(\kappa) = g(r_0)\sqrt{\frac{2\pi}{\kappa^{-2}a''(r_0)}}, \qquad \kappa\to 0. \tag{6.9}$$

Substituting (6.7) and (6.8) into (6.9) and simplifying gives

$$J_\ell(\kappa) = \frac{c_n}{\sqrt{2\pi}\sigma}\frac{p(x\cos\theta+y\sin\theta)}{|x\cos\theta+y\sin\theta|^{\ell-1}}\exp\left\{\frac{-1}{2\sigma^2}\left(\frac{x\sin\theta-y\cos\theta}{x\cos\theta+y\sin\theta}\right)^2\right\}, \quad \kappa\to 0. \tag{6.10}$$

Because $(x\cos\theta+y\sin\theta, x\sin\theta-y\cos\theta)$ are the coordinates of the point (x,y) after rotating the coordinate system by θ,

$$\frac{x\sin\theta-y\cos\theta}{x\cos\theta+y\sin\theta} = \tan\left(\tan^{-1}\left(\frac{y}{x}\right)-\theta\right) \cong \tan^{-1}\left(\frac{y}{x}\right)-\theta, \tag{6.11}$$

where the approximation in (6.11) is valid for small bearing measurement errors. Substituting (6.11) into (6.10) gives the approximation

$$J_\ell(\kappa) \cong \frac{c_n}{\sqrt{2\pi}\sigma}\frac{p(x\cos\theta+y\sin\theta)}{|x\cos\theta+y\sin\theta|^{\ell-1}}\exp\left\{\frac{-1}{2\sigma^2}\left(\theta-\tan^{-1}\left(\frac{y}{x}\right)\right)^2\right\}. \tag{6.12}$$

For $\ell = 1$, (6.12) is

$$J_1(\kappa) \cong \frac{c_n\ p(x\cos\theta + y\sin\theta)}{\sqrt{2\pi}\sigma} \exp\left\{\frac{-1}{2\sigma^2}\left(\theta - \tan^{-1}\left(\frac{y}{x}\right)\right)^2\right\}. \tag{6.13}$$

The result (6.13) is important because it shows that the classical ML formulation of the bearings-only TMA problem is recovered in the limit as $\kappa \to 0$.

Greater accuracy may be sought by using the next term in the asymptotic expansion; however, as is clear from (6.4) and the derivation above, evaluating the next term requires computing derivatives of the CAPS density function. The necessary derivatives may be computed for simple examples, but in general such derivative computations are probably of little practical use in this context.

Care must always be exercised in the use of asymptotic expansions to replace numerical integrals, and this application is no exception. Applying the result (6.12) to approximate the integrals (5.11) gives the asymptotic ratios

$$\frac{d_n^{(k)}(3)}{d_n^{(k)}(1)} = \frac{1}{|(x_n^\Omega - x_n^o)\cos\theta_n + (y_n^\Omega - y_n^o)\sin\theta_n|^2}, \quad \kappa \to 0,$$

and

$$\frac{d_n^{(k)}(2)}{d_n^{(k)}(1)} = \frac{1}{|(x_n^\Omega - x_n^o)\cos\theta_n + (y_n^\Omega - y_n^o)\sin\theta_n|}, \quad \kappa \to 0.$$

However, if these first order results are substituted into the EMAP algorithm in place of the numerical integrals, the dependence of the EMAP iterations on the CAPS model is lost entirely. The reason for this surprising result is that the EMAP approach is not valid at the limit point $\kappa^2 = 0$ because κ^2 is a variance and must remain strictly positive. An important implication of this observation is that, for sufficiently small nonzero values of κ^2, the EMAP algorithm will experience slow convergence rates. Numerical experience supports this observation. In those (highly atypical) applications in which the CAPS model is known to have such good fidelity that κ^2 must be chosen very small, it is necessary to use a succession of progressively smaller values of κ^2 to obtain satisfactory convergence rates in the EMAP algorithm. On the other hand, if a high fidelity CAPS model is known, the ML — not the EMAP — approach is the appropriate formulation of the problem.

Small measurement errors are typical in applications; however, if measurement errors are sufficiently large that the approximation (6.13) is inadequate, one may use the alternative density

$$J_1(\kappa) \cong \frac{c_n\ p(x\cos\theta + y\sin\theta)}{\sqrt{2\pi}\sigma} \exp\left\{\frac{-1}{2\sigma^2}\left(\frac{x\sin\theta - y\cos\theta}{x\cos\theta + y\sin\theta}\right)^2\right\} \tag{6.14}$$

in the traditional ML formulation. The model (6.14) is closely related to the pseudo-linear approximations used for traditional bearings-only TMA. The earliest references to approximations of this type are [11] and [12]; pseudo-linear methods are discussed extensively in [13], where an extensive bibliography is given.

The limit (6.13) is the product of two terms, a CAPS environmental prediction term depending on both measured SNR and bearing, and a geometric term depending on only measured bearing. In the fundamental ML equation (3.1), the same product is obtained by writing

$$p_{\mathbf{Z}_n|\mathbf{X}_n\mathbf{X}_n^o}(Z_n|X_n, X_n^o) = p_{\xi_n|\theta_n\mathbf{X}_n\mathbf{X}_n^o}(\xi_n|\theta_n, X_n, X_n^o)p_{\theta_n|\mathbf{X}_n\mathbf{X}_n^o}(\theta_n|X_n, X_n^o). \tag{6.15}$$

The factorization (6.15) follows directly from Bayes Theorem. The first term in (6.15) is independent of θ_n if the CAPS model is azimuthally symmetric.

7 Examples

7.1 Triangulation

To illustrate application of the EMAP estimation algorithm of Section 5, consider first the example in Figure 1, which shows ownship moving on a fixed course of 5° at a speed of 10 knots ($\approx$ 5.14 meters/second) toward a *fixed* target initially 10,000 yards (9144 meters) away at a bearing of 0° (recall

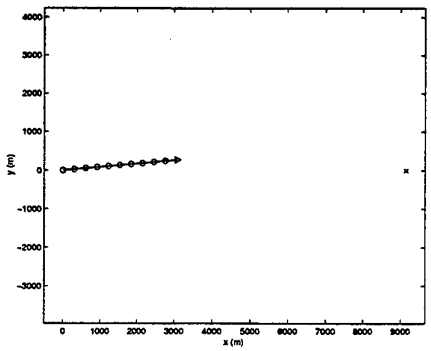


Figure 1: Classic triangulation example.

that all angles are measured counter-clockwise from the x axis). This is a classic triangulation problem in which the target parameters $\lambda = \{x, y\}$ are estimated from a series of azimuthal bearing measurements $Z_\theta = \{\theta_n\}_{n=1}^N$ taken at times $t = \{t_n\}_{n=1}^N$. In this example, ownship takes bearing and SNR measurements at 1 minute intervals on a 10 minute leg for a total of 11 measurements.

It is assumed that target source level is a *known* constant, independent of aspect angle, and that the target is an omnidirectional point source of relatively low frequency, so that propagation loss due to absorption is negligible. An infinite, homogeneous, isovelocity (1500 meters/second) ocean is assumed, with a constant, isotropic ambient noise field. These assumptions imply straight-line, direct path propagation.

Assuming additive, zero-mean, white Gaussian noise with known second order statistics, the stochastic model for the SNR measurements $Z_\xi = \{\xi_n\}_{n=1}^N$ is

$$\xi(r,t) = \bar{\xi}(r) + q(t), \tag{7.1}$$

where $\bar{\xi}(r)$ is the mean SNR from the propagation loss model and $q(t)$ is the noise process with variance η^2. The propagation loss model for this problem is given by the passive sonar equation, which relates received signal excess level SNR to target source level SL, propagation loss PL, ambient noise level NL, and sensor directivity index DI [14]. In decibels, the passive sonar equation is given by

$$SNR = SL - PL - NL + DI. \tag{7.2}$$

For this example, the notional target is assumed to have a source spectrum level (SSL) of 140 dB//1 μPa near 1 kHz, so that

$$SL \equiv SSL + 10\log\Delta f \tag{7.3}$$

for broadband detection, where Δf is the receiver bandwidth in Hertz. Propagation loss is due to spherical spreading, so the pressure field for a point source falls off as r, where r is range to the target. In decibels, this gives

$$PL = 20\log r. \tag{7.4}$$

Noise level is defined in terms of the noise spectrum level (NSL),

$$NL \equiv NSL + 10\log\Delta f. \tag{7.5}$$

Nominal curves for the ambient noise spectrum level in the open ocean are plotted in [14]. The value of 62 dB//1 μPa is used for NSL in this example.

The sensor model used for this example is a discrete line array of 25 elements with 1 meter spacing, using broadband detection in a 100 Hz band centered at 1 kHz. Simple geometrical considerations give a directivity index for this array of approximately 15 dB//1 μPa [15]. The cone angles measured by the line array are taken as azimuthal bearings in this example, with the additional assumption that the target and the sensor are at equal depths. The separation between sensor and ownship is assumed negligible and is ignored for this idealized problem.

Substituting (7.3)–(7.5) in (7.2) gives the mean SNR, in decibels, as

$$10\log\bar{\xi}(r) = SSL - 20\log r - NSL + DI. \tag{7.6}$$

Substituting (7.6) in (7.1) and writing the result as a probability density function gives

$$p_{\xi_n|r_n s_n^o}(\xi_n|r_n, SL) = \mathcal{N}(\xi_n|\bar{\xi}_n(r_n), \eta^2) \tag{7.7}$$

for the CAPS density of the n-th SNR measurement ξ_n, where $\mathcal{N}(\cdot|\mu, \sigma^2)$ denotes a Gaussian PDF with mean μ and variance σ^2 evaluated at an arbitrary point.

The EMAP algorithm stated in Section 5 is essentially unchanged for this reduced order triangulation problem. The target parameters to be estimated are the fixed locations $\lambda = \{x, y\}$. The quadratic terms $Q_n \equiv Q_n(r_n; \lambda)$ in (4.13) become

$$Q_n = \begin{bmatrix} x - x_n^o - r_n \cos\theta_n \\ y - y_n^o - r_n \sin\theta_n \end{bmatrix}' \Lambda(\theta_n, \sigma_n, \kappa_n) \begin{bmatrix} x - x_n^o - r_n \cos\theta_n \\ y - y_n^o - r_n \sin\theta_n \end{bmatrix}, \tag{7.8}$$

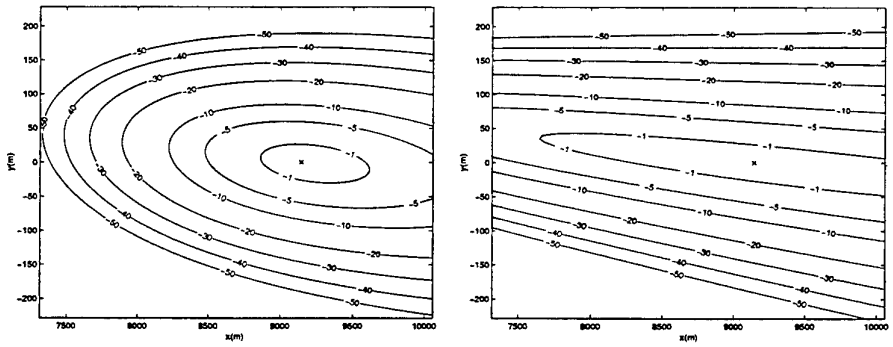
for the fixed target model. For triangulation, the matrix $H(t_n)$ defined in (5.6) reduces to the 2×2 identity matrix, the matrix (5.8) reduces to the 2×2 matrix

$$A_n(\theta_n, \xi_n) \equiv \Lambda(\theta_n, \sigma_n, \kappa_n) \int_{r_{\min(\theta_n, X_n^o)}}^{r_{\max(\theta_n, X_n^o)}} w_{\theta_n \xi_n}(r_n; \lambda') \frac{dr_n}{r_n^2}, \tag{7.9}$$

and the vector (5.9) reduces to the length 2 vector

$$b_n(\theta_n, \xi_n) \equiv \Lambda(\theta_n, \sigma_n, \kappa_n) \int_{r_{\min(\theta_n, X_n^o)}}^{r_{\max(\theta_n, X_n^o)}} w_{\theta_n \xi_n}(r_n; \lambda') \begin{bmatrix} x_n^o + r_n \cos\theta_n \\ y_n^o + r_n \sin\theta_n \end{bmatrix} \frac{dr_n}{r_n^2}. \tag{7.10}$$

Figure 2 shows plots of the likelihood function (4.12) for various values of the target location parameters λ, with and without the CAPS model. For each n, 79 sample ranges, from 500 to 20,000 yards



(a) With environmental model. (b) Without environmental model.

Figure 2: EMAP likelihood functions (dB//max) for triangulation example.

equispaced every 250 yards, were used for the numerical integration of (4.12). Both plots were generated with measurement standard deviations $\sigma = \sigma_0 r$ and $\kappa = \kappa_0 r$ (see Appendix A of [1]), with dimensionless standard deviations at $r = 1$ of $\sigma_0 = 0.0175$ and $\kappa_0 = 0.0873$ for each bearing measurement. The value of σ_0 corresponds to a bearing standard deviation of 1°. The values of σ and κ translate roughly to cross- and down-range standard deviations of 200 and 1000 yards, respectively, for a target 10,000 yards away. An SNR measurement standard deviation η equivalent to 2 dB was used for the plot in Figure 2(a). Comparison of the two plots reveals that the contours for the likelihood function with the CAPS model are much more compact; the augmented problem is much less diffuse in range. Similar plots (not included here) show that the contours generated with the environmental model approach those generated without the model as η is increased.

The scatter plot of Figure 3 shows the estimation results for 250 Monte Carlo runs of the EMAP algorithm. For each run, new measurements $Z = \{\theta_n, \xi_n\}_{n=1}^{11}$ were generated, and the location parameters were initialized for a target at 15,000 yards with a 45° bearing. The values $\sigma_0 = 0.0175$,

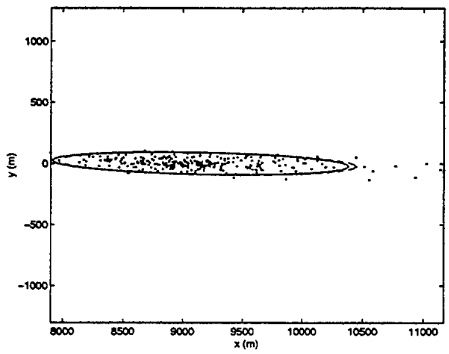


Figure 3: EMAP algorithm results for a 250 run Monte Carlo simulation. The EMAP and ML 90% containment ellipses are shown with solid and dashed lines, respectively.

$\kappa_0 = 0.0873$, and η equivalent to 2 dB were used for the measurement standard deviations. An increase in the log-likelihood function (4.12) of less than 1×10^{-12} was used as the stopping criterion for the EMAP algorithm. The average number of iterations for each run was approximately 145. Included in the plot are the 90% containment ellipses based on the sample covariances of the EMAP and ML estimates, where the ML estimates (not plotted) were computed with the CAPS model as described in Section 3. The two containment regions are almost indistinguishable. In fact, the EMAP estimates approach the ML estimates *exactly* as the down-range variance κ^2 is taken sufficiently small (see Section 6).

7.2 Constant Velocity Target

As a second example of application of the EMAP algorithm, consider the constant velocity target problem where the target parameters to be estimated are the end points of target motion, $\lambda = \{x_1, y_1, x_N, y_N\}$. Figure 4 shows ownship moving on a fixed course of 5° at a speed of 10 knots on the first leg, followed by a fixed course of 95° at a speed of 10 knots on the second leg after a 1 minute simulated maneuver. The target starts 10,000 yards away at a bearing of 0°, and moves with a constant speed of 5 knots on

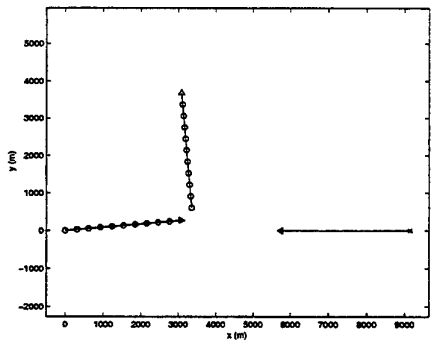
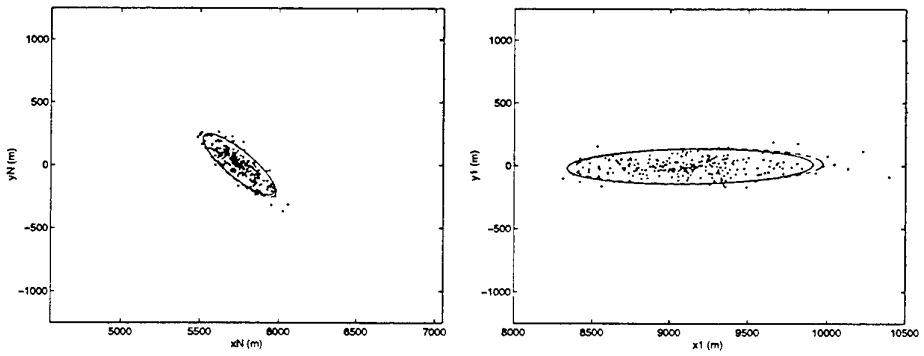


Figure 4: Constant velocity target example.

a 180° course. Ownship takes bearing and SNR measurements at 1 minute intervals on both 10 minute legs for a total of 22 measurements.

Figure 5 shows the results of a 250 run Monte Carlo simulation of the EMAP algorithm for this problem using the same environmental model, sample ranges, values for σ_0, κ_0, and η, and stopping criterion used for the triangulation example. For each run, new measurements $Z = \{\theta_n, \xi_n\}_{n=1}^{22}$ were generated, and the location parameters were initialized for a target moving from 15,000 yards away at a bearing of 45°, to a position 10,000 yards away at a 315° bearing, both with respect to ownship's initial position. The average number of iterations for each run was approximately 100. The 90% containment ellipses based on the sample covariances of the EMAP estimates and the ML estimates (not plotted) are shown about the true target end points. As with the triangulation example, the EMAP and ML containment regions are nearly equivalent.



(a) Final location (x_N, y_N).

(b) Initial location (x_1, y_1).

Figure 5: EMAP algorithm results for constant velocity target example. The EMAP and ML 90% containment ellipses are shown with solid and dashed lines, respectively.

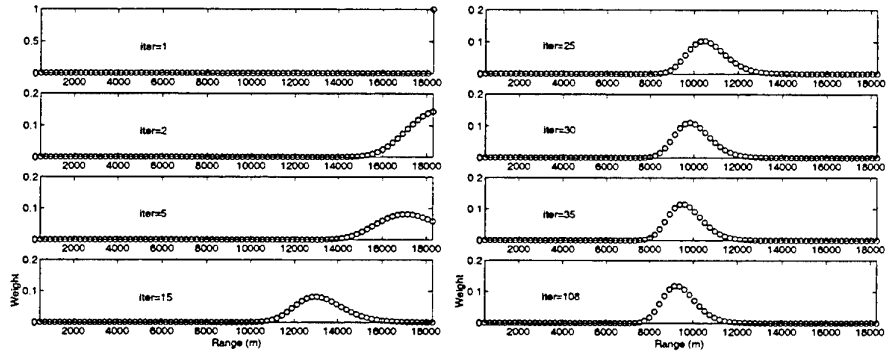


Figure 6: Down-range weights for first bearing at several iterations.

Figure 6 shows plots of the weights (5.5) for the 79 sample ranges down the first line of bearing at several iterations for a single run of the EMAP algorithm for the constant velocity target example. These weights describe the distribution of the components in the mixture for the first bearing. As the iterations increase, the peak of this distribution moves "like a log rolling under a carpet" closer to the component associated with the empirical target most likely to have generated the measurement. On the first iteration, the weight associated with the largest sampling range down the first line of bearing is equal to 1 due to the poor initialization. The remaining weights are all zero, as the sum of all the weights must add to 1 for each bearing. On the second iteration, the estimate of λ improves, and more mixture components contribute to the PDF of the target. The weight scales of the second and subsequent plots in Figure 6 are reduced to make the weight distributions more pronounced. Examination of the plot in Figure 6 for the final iteration of one run of the EMAP algorithm reveals that the standard deviation of the weight distribution is approximately 1000 yards, which is roughly the value of κ for the target range in this example. The down range variance specifies a window over which the CAPS model is averaged (in a weighted sense) or smeared. The EMAP algorithm samples the CAPS model, and approaches the ML algorithm discussed in Section 3, as κ^2 is made sufficiently small.

7.3 Comments

The above examples are generalized for *unknown* source level by treating source level as an unknown parameter, and estimating it between EM iterations. This extension of the EM method is referred to as the Generalized EM, or GEM, method, and is discussed in [7]. Source level is estimated in the GEM framework by conducting a one dimensional search on source level to increase the value of the auxiliary function between iterations. That is, source level is chosen to maximize the auxiliary function over its value at the current target positional estimates, rather than to maximize the auxiliary function over the whole parameters space simultaneously. It is shown in [7] that the likelihood function does not decrease

after an iteration of GEM, and that GEM converges if the likelihood function is bounded above. Though this approach to estimating source level presents no great difficulties, the EMAP algorithm is more than just a simple iteratively reweighted linear least squares algorithm in this case, and is not pursued further in this paper.

In the above examples, no attempt was made to speed up the EMAP algorithm convergence rate. The same "monotone" simulated annealing scheme suggested in Section 4 for compensating for model mismatch would speed up convergence at the beginning of the EM iterations, where initial estimates may be poor due to a bad initialization. Using non-equispaced sampling ranges and decreasing the number of ranges would also speed up convergence. Increasing the spacing down-range is justified, as the down-range standard deviation κ increases in proportion to range.

8 Summary and Concluding Remarks

Two formulations of the augmented bearings-only TMA problem have been presented. One formulation is based on maximum likelihood, and it is classical in that it is a post-detection tracking approach in which measurements are conditioned on the target state. The other formulation is novel in that it is based on an empirical MAP method in which measurements are unconditionally independent because they are pre-detection measurements; that is, measurements are assumed unconditionally independent until proven conditionally independent by a detection decision. The EMAP approach is analogous to a GLRT method for simultaneous detection and track estimation.

An algorithm for solving the augmented bearings-only TMA problem using the EMAP formulation is derived by the method of EM. The general EMAP algorithm is an iteratively reweighted linear least squares method provided (i) source level is independent of aspect angle and (ii) fidelity issues are treated by adjusting the down-range variance of the geometric kernel. Thus, the EMAP algorithm is a linear algorithm in most cases of practical interest. It is a remarkable corollary of the EMAP algorithm that the traditional, i.e., non-augmented, bearings-only TMA problem can be solved by an iteratively reweighted linear least squares algorithm. In the most general case, however, the EMAP algorithm is a nonlinear penalized least squares algorithm whose potential value in the application remains unexplored.

The empirical approach leads naturally to an integral representation of the measurement density in which possible lack of fidelity in the CAPS model is compensated by adjusting the down-range variance κ of the geometric kernel of the integrand. In effect, the proposed compensation averages the CAPS model prediction over a sliding Gaussian window whose size, both down-range and cross-range, increases linearly with the down-range direction because of cylindrical spreading. It is shown that in the limit, as the size of the averaging window goes to zero, the EMAP approach is equivalent to the standard ML approach, the preferred approach when CAPS predictions are reliable.

Triangulation and constant velocity target examples were presented to illustrate application of the EMAP algorithm. These examples assumed a known source level and direct path, straight-line propagation with propagation loss due to spherical spreading. The EMAP algorithm estimates were nearly equivalent to the ML algorithm estimates for a reasonably sized value of the down-range variance κ^2. The EMAP algorithm is generalized to unknown source level by using the Generalized EM method to estimate source level between EM iterations.

Different sensor types lead to different expressions for the geometric kernel and, hence, to different integral representations. For example, for linear arrays the angular measurement is conical angle, not azimuthal bearing; therefore, the geometric kernel generalizes to a trivariate Gaussian in (x, y, z) with one fixed variance corresponding to the cone angle measurement and two free variances that compensate CAPS prediction infidelities. In this case, for unbounded isovelocity ocean models, the integral of the representation is over the surface of a cone with vertex at the acoustic center of the array, axis along the array, and half angle equal to the measured conical angle, instead of a line integral over a ray as for simple azimuthal bearings. For bounded non-isovelocity ocean models, the rays comprising the locus of the cone are distorted by internal refraction and boundary reflections into a manifold whose detailed structure is determined by the acoustic model. Generalizing the geometric density term used in the EMAP formulation to other sensors with limited observability (in the statistical sense) would seem to present few intrinsic conceptual difficulties.

Acknowledgments

The authors thank Marcus Graham and John Baylog, both with the Naval Undersea Warfare Center, for their many stimulating conversations and comments.

References

[1] R. L. Streit, "Bearings-Only Target Motion Analysis With Propagation Loss Models of Uncertain Fidelity," Proceedings of the Third International Conference on Theoretical and Computational Acoustics, Newark, New Jersey, 14–18 July, 1997, to appear.

[2] D. Lerro and Y. Bar-Shalom, "Interacting Multiple Model Tracking With Target Amplitude Feature," IEEE Trans. on Aerospace and Electronic Systems, AES-29, 1993, pp. 494–509.

[3] J. G. Baylog, Personal communication, Naval Undersea Warfare Center, Newport, RI, 15 July 1996.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin,"Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp.1–38, 1977.

[5] C. F. J. Wu, "On the Convergence Properties of the EM Algorithm," Annals of Statistics, 11, 1983, pp. 95–103.

[6] R. A. Redner and H. F. Walker, "Mixture Densities, Maximum Likelihood, and the EM Algorithm," SIAM Review, 26(2), 1984, pp. 195–239.

[7] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, Wiley & Sons, 1997.

[8] D. M. Titterington, A. F. M. Smith, and U. E. Makov, *Statistical Analysis of Finite Mixture Distributions*, Wiley & Sons, New York, 1985.

[9] R. L. Streit and T. E. Luginbuhl, "Maximum Likelihood Training of Probabilistic Neural Networks," IEEE Trans. on Neural Networks, NN-5, 1994, pp. 764–783.

[10] N. Bleistein and R. A. Handelsman, *Asymptotic Expansion of Integrals*, Holt, Rinehart, and Winston, New York, 1975.

[11] R. G. Stansfield, "Statistical Theory of D.F. Fixing," *J. IEE, London*, pt. 3A, vol. 94, no. 15, 1947, pp. 762–770.

[12] C. J. Ancker, "Airborne Direction Finding – The Theory of Navigation Errors," *IRE Trans. on Aeronautical and Navigational Electronics*, 1958, pp. 199-210.

[13] A. Holtsberg, *A Statistical Analysis of Bearings-Only Tracking*, Doctoral Dissertation, Lund Institute of Technology, Dept. of Mathematical Statistics, 1992.

[14] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J. V. Sanders, *Fundamentals of Acoustics*, 3rd ed., Wiley & Sons, 1982.

[15] R. J. Urick, *Principles of Underwater Sound*, 3rd ed., McGraw-Hill, New York, 1983.

[16] A. Graham, *Kronecker Products and Matrix Calculus with Applications*, Wiley & Sons, New York, 1981.

A Variety of PMHTs

Peter Willett, Yanhua Ruan
University of Connecticut
Storrs, CT 06269

Roy Streit
Naval Undersea Warfare Center
Newport, RI 02841

Abstract

The PMHT is an algorithm of considerable beauty. In practice, its performance turns out to be at best similar to that of the PDAF in most cases; and since the implementation of the PDAF is less intense numerically the PMHT has been having a hard time finding acceptance. The task, therefore, is to "make the PMHT work".

The PMHT's problems its "nonadaptivity" to poor track quality, "narcissism" in the face of a string of apparently missed detections, and "over-hospitality" to clutter. In this paper we show modifications on the original basic PMHT which offer some improvement. We explore, among other things, the use of "homothetic" measurement models; the "end-point" and "start-point" PMHTs, which modify the PMHT assumptions such that the estimation goal is the track at only one point; maneuver-based PMHTs, including those with separate and joint homothetic measurement models; a modified PMHT whose measurement/target association model is more similar to that of the PDAF; PMHTs with eccentric and/or estimated measurement models. The above are improvements on the basic PMHT based on modifications of the underlying model. We also offer basic-PMHT implementations with aided convergence.

1 Introduction

Both the multi-hypothesis tracker (MHT) and probabilistic data association filter PDAF [1, 2] track imperfectly-detected targets in clutter via a *hard-association* model. That is, these algorithms enumerate the possible associations between measurements and target(s), and evaluate which is best. And because there are a great many such associations a full enumeration is computationally infeasible, hence for each algorithm the search is suboptimal.

The PMHT [3] makes modification to the measurement model. The PDAF and MHT assume, quite rightly, that a target can generate at most one measurement per scan; the PMHT sacrifices this constraint, and posits the measurement/target association process as independent across measurements. By doing so, it is able to render a *fully-optimal* (under a modified assumption) tracker. The associations become *soft* – in fact, governed by their posterior probabilities – and the integer-programming problem of target tracking is rendered continuous and amenable to an iterative "hill-climbing" method via the EM algorithm. The PMHT has much in common with the EM algorithm as applied to the estimation of parameters in a Gaussian mixture.

The PMHT is a very nice algorithm, in our opinion. It is probabilistically-sound (all the "bad" parts of the PMHT are out in the open in the original assumptions), and it is easily extensible. This extensibility has been exploited in a number of ways, such as those dealing with multiple targets [4], nonlinear models [5], and target-maneuver [6, 7]. It is very likely that this easy extensibility will be the aspect to the PMHT which provides it a wider audience and acceptance.

However, as of yet the PMHT has not managed to "beat" the simple PDAF in the game of lost-tracks. Therefore, in this paper, we propose to exploit the PMHT's easy extensibility to attempt its improvement. We develop a series of model-varied PMHTs and test them; since our goal is straightforward, we test in the most straightforward manner possible, by simulation in a linear Gaussian environment, with one target, imperfect detection, and clutter.

In the first section we discuss in some detail our perception of the PMHT's *problems*: its lack of adaptivity to a track being lost, its inability to sense an impending track loss, and its tendency to welcome several clutter measurements as a single detection of high accuracy. We then dwell briefly on a variety of different modifications on the PMHT: there are fifteen PMHT variations in all. We test them, and conclude. Many details are left out of this paper: for the interested reader, please see [8].

It will be seen, unfortunately, that no modification on the PMHT is uniformly superior to the simple PDAF; in certain situations no PMHT beats the PDAF. However, it will be seen that in very adverse tracking environments several PMHTs are much improved, offering, for example 50% lost tracks versus the PDAF's 80%.

2 PMHT Problems

The weakness of the PMHT is in its *dependence on a good track-initialization.* If initialized on the correct trajectory and allowed to iterate, the PMHT will perform very well indeed. However, reliance on some clairvoyant initialization is not a desirable property in a tracking algorithm, and while the PMHT is not incapable of recovering from a poor initialization, its tendency to be satisfied with a relatively minor likelihood-function local peak is a problem. There are a number of reasons why the PMHT is particularly prone to this, and while all are inter-related and in any case could be explained away as poor initialization, it is instructive to consider them separately that modifications can be made.

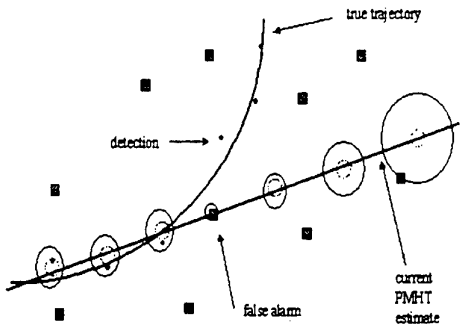


Figure 1: Illustration of the non-adaptivity of the basic PMHT: true detections are shown as o's, false alarms as □'s, the PMHT's internal estimate of its state error covariance as the larger ellipses, and the measurement error covariances which control the w's are shown as smaller dashed ellipses. Note both that the PMHT's state error covariances have shape controlled by the measurement covariances (both are circular in this plot); and that while the state error covariances increase in size, the w's are sought over a fixed-size volume.

2.1 Non-Adaptivity

Before examining the PMHT, let us consider the behavior of the PDAF. The PDAF forms posterior probabilities of each measurement being associated with the current track (called the β's in [1]) by computing the probability of each detection under an assumption of Gaussianity, with mean the predicted value of the measurement, and with covariance that of the innovations, usually denoted **S**. On an intuitive level, therefore, if the current track estimate from the PDAF is apparently poor, the resulting "large" **S** will force an adjustment in the posterior probabilities such that a wider field of view is encouraged; that is, when the track is looking poor, the PDAF looks further afield for a valid measurement. If this

action occurs soon enough and there *is* a valid un-missed measurement nearby, then a "rescue" of the track is effected; if not, then $\mathbf{S}$ grows without bound and the track is declared lost. With the PMHT, regardless of the quality of the current state estimate the posterior probabilities that measurements are track-generated is controlled by the measurement covariance $\mathbf{R}$. There is no adaptivity whatever, and the PMHT is to a large extent incapable of rescuing itself from a currently-poor track estimate. This is illustrated in figure 1.

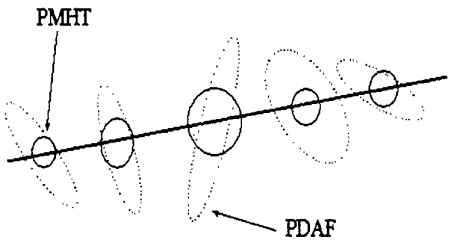


Figure 2: Illustration of a difference between the PDAF and PMHT: the former's state estimation covariances are adjusted in shape by the data, while the latter's are adapted only in size, not shape.

A further aspect to this non-adaptivity is through the estimation covariance *shape*. Specifically, the spread of its validated measurements enters the PDAF update as its "spread of the innovations" term, and the internal PDAF state estimate covariance (the P) adjusts its shape to reflect it. There is no mechanism at all for the PMHT to have other than spherical $\mathbf{P}$'s. This is illustrated in figure 2.

2.2 Narcissism

Further examination of figure 1 illustrates another PMHT problem: even a single false-alarm near an incorrect track estimate is satisfactory to the PMHT, and the lack of a reasonable number of validated detections does not faze it. This willingness to believe that its track is progressing normally in the face of overwhelming evidence to the contrary is certainly related to the "nonadaptivity" of the search for valid measurements discussed in the previous subsection.

2.3 Hospitality

In both the PDAF and PMHT the estimation covariance that would appear at scan t in the absence of any measurement at scan t is reduced by an amount corresponding to the perceived quality of the measurement(s) at scan t. Consider the case that there are several "valid" measurements. In the case of the PDAF the spread-of-the-innovations terms amounts to an admission of confusion by the data association step, and the perceived measurement quality is low. For the PMHT, however, we have $\tilde{\mathbf{R}} < \mathbf{R}$. The situation is presented schematically in figure 3. Intuition suggests that the PDAF behavior is in most tracking situations reasonable, while that of the PMHT is not, since the plurality of measurements hints that some must be clutter.

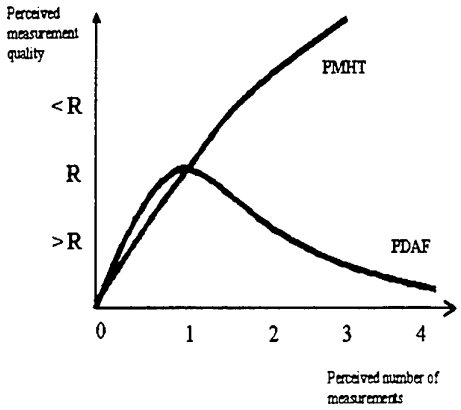


Figure 3: Illustration of the effect of multiple "valid" measurements on the PDAF and PMHT. As this number increases beyond unity, the PMHT perceives a high-quality "multiple" measurement (i.e. $\bar{\mathbf{R}} < \mathbf{R}$, a result which intuition suggests is incorrect.

3 Fixing the PMHT

3.1 The Homothetic PMHT

The homothetic PMHT, discussed previously in [10, 4], is a modification on the basic PMHT model such that measurements at scan t can come any Gaussian density having mean $\mathbf{x}(t)$ and variance $\{\kappa_p^2 \mathbf{R}\}_{p=1}^{P}$. Typical values used are $P = 2$, with $\kappa_1 = 1$ and $\kappa_2 = 3$. The "homothetic" nomenclature derives from "having the same mean"; that is, the model has been altered from having one target to P targets, and EM MAP estimation proceeds under the constraint that each of these models has the same track $\{\mathbf{x}(t)\}_{t=1}^{T}$.

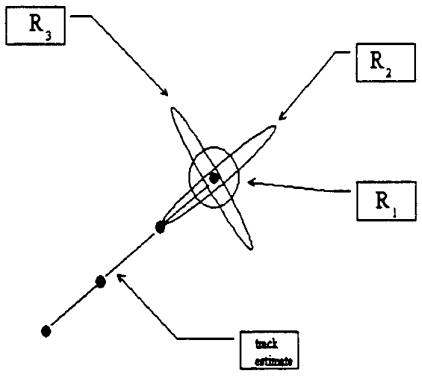


Figure 4: Illustration of the eccentric measurement covariance ellipses assumed under the "Spirograph" homothetic PMHT model.

3.2 The Spirograph PMHT

"Spirograph" refers to a children's toy, an aid to the drawing of flower-like designs of concentric ellipses which are rotated relative to each other. Essentially this is a more-involved homothetic PMHT. Whereas the homothetic PMHT assumes that measurements at scan t can come any Gaussian density having mean $\mathbf{x}(t)$ and variance $\{\kappa_p^2\mathbf{R}\}_{p=1}^P$, the variances in the case of the spirograph PMHT can be any of $\{\mathbf{R}_p\}_{p=1}^P$. As illustrated in figure 4 we use $P = 3$, with $\mathbf{R}_1 = \mathbf{R}$, and the other two eccentric ellipses with respectively major and minor axes lying in the direction of (currently-estimated) PMHT motion.

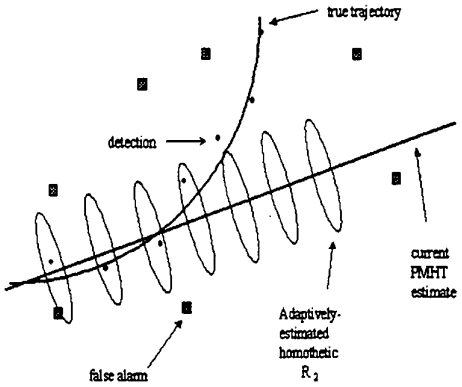


Figure 5: Illustration of the eccentric measurement covariance ellipses encouraged by a track estimate which is diverging from the true trajectory, under the adaptive homothetic model.

3.3 An Adaptive Homothetic PMHT

In this case we have $P = 2$, with the first ($p = 1$) homothetic model the standard ($\mathbf{R}$) one, and the second ($p = 2$) a measurement noise model which is estimated from the data in the batch.

3.4 The PMHT with a PDAF Measurement Model

There is a legitimate complaint that the PMHT measurement model, in which measurement/track associations are independent across all measurements, is unrealistic. For the case of a single target modification of the PMHT to the PDAF mesaurement mode is straightforward.

3.5 A Detection-Oriented PMHT

The basic PMHT's *narcissism* discussed earlier stems at least partly from its inability to realize that a string of missed detections is inappropriate. We therefore consider modifying the PMHT such that there must be at least one detection in P scans of data – or at least by making the event that all P scans of data contain nothing but false alarms infeasible.

3.6 The Endpoint PMHT

It is arguable that the PMHT's goal of track estimation – that is of estimation of $\{\mathbf{x}_s(t)\}_{t=1}^T$ – is incompatible with the true aims of target tracking. If a set of scans from $t = 1$ to $t = T$ is available, a reasonable assumption is that the most-recent data is in scan T, and estimation of the entire trajectory which led to $\mathbf{x}_s(T)$ is unnecessary. If this is accepted, then the basic PMHT model, in EM terms, can be modified such that we have (for a single target)

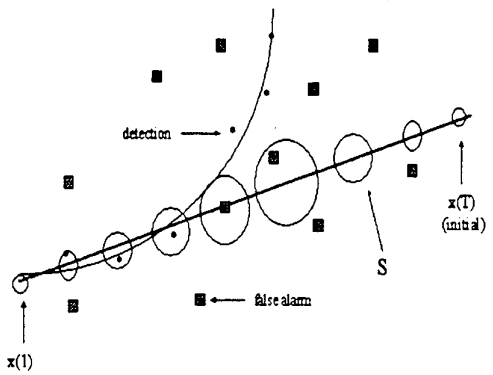


Figure 6: Illustration of the covariances used to calculate the w's under the endpoint PMHT. These are given as the *innovation* covariances (the $\mathbf{S}$'s) derived from a Kalman smoother with initial conditions $\mathbf{x}(1)$ and covariance $\mathbf{P}(1)$, and $\mathbf{x}(T)$ with covariance $\mathbf{0}$ - and *no measurements*. Note that these covariances wax and then wane as the scan number increases.

$\mathcal{X} = \mathbf{x}(T)$,

$\mathcal{Z} = \{\mathbf{z}_r(t)\}$, where $\mathbf{z}_r(t)$ is the r^{th} measurement vector at time t,

$\mathcal{K} = \left\{\{\{k_r(t)\}_{r=1}^{n_t}\}_{t=1}^{T}, \{\mathbf{x}(t)\}_{t=1}^{T-1}\right\}$, where $k_r(t)$ is the target from which the r^{th} measurement at time t arises.

Thus, the state *previous* to the current scan is moved from an estimation goal (i.e. $\mathcal{X}$) to nuisance variables (i.e. $\mathcal{K}$).

This may appear a very significant modification, but in fact the implementation of the endpoint PMHT is very similar to that of the basic PMHT. The main difference is that the *expectation* is now taken also over the now-unwanted intermediate trajectory variables.

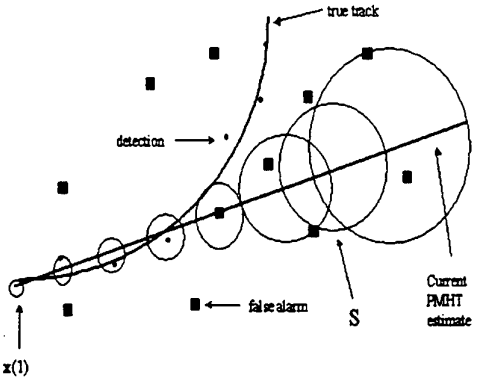


Figure 7: Illustration of the covariances used to calculate the w's under the startpoint PMHT. These are given as the *innovation* covariances (the $\mathbf{S}$'s) derived from a linear model with initial conditions $\mathbf{x}(1)$ and $\mathbf{P}(1)$. Note that, as opposed to figure 6, these grow monotonically as the scan number increases.

3.7 The Startpoint PMHT

This is a variation on the endpoint PMHT: we have

$\mathcal{X} = \mathbf{x}(1)$,

$\mathcal{Z} = \{\mathbf{z}_r(t)\}$, where $\mathbf{z}_r(t)$ is the r^{th} measurement vector at time t,

$\mathcal{K} = \left\{\{\{k_r(t)\}_{r=1}^{n_t}\}_{t=1}^{T}, \{\mathbf{x}(t)\}_{t=2}^{T}\right\}$, where $k_r(t)$ is the target from which the r^{th} measurement at time t arises.

The main difference, therefore, is that estimation is of the *start* value $\mathbf{x}(1)$ rather than the *end* value $\mathbf{x}(T)$. This has somewhat less intuitive appeal, but there are advantages. One is that, as illustrated in figure 7, is that the "look" regions become uniformly larger as t increases, which, while not precisely *adaptive*, is more in keeping with the true uncertainty in the data.

3.8 The Maneuvering PMHT

It turns out that the PMHT model is easily augmented to incorporate an underlying Markov maneuver process – naturally, this appears as part of the augmented "nuisance" EM variables $\mathcal{K}$, which are split into $\{k^1\}$, the original measurement/target association variables, and $\{k^2\}$, where $k^2(t) = l$ means that the (Markov) maneuver model is in state l at time t.

3.9 The Maneuvering/Homothetic PMHT

The maneuvering PMHT is again augmented to include a homothetic mesurement model. The random processes determining which of the homothetic measurement and maneuver process noises are active are independent.

3.10 The Joint Maneuver/Homothetic PMHT

The maneuver and homothetic processes, as above, are made joint. That is, either all measurements at scan t come from the low-variance measurement noise *and* the low-variance process noise is active between scan $t-1$ and t; ot both have high variance.

3.11 A Deflationary PMHT

It has been stated previously that the major problem with the basic PMHT is of *track-initialization*; that is, the likelihood surface over which the basic PMHT performs its maximization is often so "busy" with local maxima that convergence to a good trajectory is unlikely. Thus, a means to smoothen the likelihood surface is sought. An approach which has been commonly applied is to inflate the measurement covariance. It is hoped that if the initial EM iteration is performed with a large $\mathbf{R}$, and subsequent iterations use smaller and smaller (*deflating*) $\mathbf{R}$'s, then the eventual convergence to the global maximum will be encouraged. The idea is similar to that of simulated annealing, although "downhill" steps are not used.

3.12 A Homoscedastic PMHT

Whereas "homothetic" refers to measurement noise models having the same mean and difference covariances, "homoscedastic" corresponds to identical covariances and different means. If a set of P offsets $\{\delta\mathbf{x}_p(t)\}_{p=1}^{P}$ is specified, then we have

$$p(\mathcal{Z},\mathcal{X}) = p(\mathbf{x}(1))\prod_{t=2}^{T} p(\mathbf{x}(t)|\mathbf{x}(t-1))\prod_{t=1}^{T}\prod_{r=1}^{n_t}\left[\sum_{p=1}^{P}\pi_p\mathcal{N}\left\{\mathbf{z}_r(t);\mathbf{H}[\mathbf{x}(t)+\delta\mathbf{x}_p(t)],\mathbf{R}(t)\right\}\right] \quad (1)$$

This means that measurements can come from a variety of points offset from the true trajectory, and the intention is that an increased "look" region can be so obtained. To retain some degree of faith to the original model, we take $\delta \mathbf{x}_1(t) = \mathbf{0}$; the other offsets are generated as Gaussian with zero mean and covariance $\mathbf{P}(t)$.

4 Results

4.1 Model and Parameter Values

We use a two-dimensional kinematic model with position-only measurements. Some notes and parameter values:

- The inter-scan time is $\Delta t = 30$ seconds.
- We have $\sigma_m = 100$ meters.
- We explore varying process noises: $\sigma_p = .0001$, $\sigma_p = .001$, and $\sigma_p = .01$.
- We explore various probabilities of detection: $P_d \in \{50\%, 70\%, 90\%\}$.
- We explore various clutter densities, $\lambda \in \{10^{-5.5}, 10^{-6.5}\}$.
- We choose a track length $T = 30$ scans of data.

For the most part, targets begin their trajectories at scan $t = 0$ (explained shortly) with position coordinates (0,0) and velocity coordinates (5,5) meters per second, corresponding to 13.8 knots. We also examine the cases in which the initial velocities are (1,1) and (20,20) meters per second.

Some further details follow:

Time-Initialization. We use *two-point* time-initialization.

Declaration of Lost Tracks. We declare a track lost if

$$\left(\mathbf{x}_1^n(T) - \mathbf{x}_1^{true}(T)\right)^2 + \left(\mathbf{x}_3^n(T) - \mathbf{x}_3^{true}(T)\right)^2 > 2(4\sigma_m)^2 \tag{2}$$

in which the subscripts 1 and 3 refer to the two position coordinates.

RMSE. We compute the root mean square error in the obvious way, with two exceptions. First, we do not use tracks considered to have been lost. Second, we give only the RMSE for the final scan $t = T$ of data.

Clutter. For each scan a Poisson random variable is generated with parameter λV, in which λ is the clutter density and V denotes the (square) volume over which the false alarms are to be generated.

Monte Carlo Iterations. All simulations are on the basis of 200 Monte Carlo simulation runs. The exception to this is the PDAF, for which 500 runs are used.

Volume. This "volume" refers to the region V (two-dimensional, in our case) over which false-alarms are generated. For the basic PMHT and the PDAF, as long as V is both not too small and encloses the target, there is little concern. However, for some of the PMHT varieties, particularly those such as the homothetic and deflationary models, an inaptly-chosen region can cause significant problems. To avoid the "carry-along" phenomenon *we center the clutter at the origin* rather that at the true trajectory.

4.2 The Results

We test the various PMHT's described earlier, and also the PDAF, in:

- $\lambda = 10^{-5.5}$, initial speed 13.8 knots, $P_d \in \{50\%, 70\%, 90\%\}$, $\sigma_p \in \{.01, .001, .0001\}$, $\sigma_m = 100$ - pictured in figure 8 with $P_d = 90\%$ and $\sigma_p = .0001$;
- $\lambda = 10^{-6.5}$, initial speed 13.8 knots, $P_d \in \{50\%, 70\%, 90\%\}$, $\sigma_p \in \{.01, .001, .0001\}$, $\sigma_m = 100$ - pictured in figure 9 with $P_d = 90\%$ and $\sigma_p = .01$;

A number of other situations are discussed in [8] and are referred to in the sequel.

The results in terms of in-track percentage are given in respective tables 1, and 2, Further results in terms of RMSE for non-lost tracks are given in tables 3 and 4. We attempt to digest these in table 5. This lists the average rank (rank 1 is best, rank 16 is worst) for each of the sixteen algorithms tested, in each of the 63 situations in which the tests were run (in [8]), in terms of lost tracks and of RMSE for non-lost tracks.

We have the following comments:

PDAF. This tracker, unfortunately from our point of view, is almost consistently the best. Exceptions appear only to be in the case of *very* heavy clutter and maneuvering targets, situations in which certain of the PMHT algorithms perform better but not excitingly.

PMHT without sliding. This is the basic form of the PMHT, performed on batches of length 6 which overlap by exactly one sample, and tracks do not grow. Generally the performance of this is inferior to other PMHTs. However, it is seen that when tracking *is* working, the RMSE is low.

Sliding PMHT. This is the basic PMHT, performed on sliding batches of length 12, with a skip parameter of 3 scans. In most cases this is better than the basic PMHT with fixed batches; but other PMHTs have better performance. All PMHTs except the first use such a sliding window and batch length.

Homothetic PMHT. This simple modification improves the PMHT enormously in almost all cases. Naturally, due to the increased size of the assumed measurement noise, the RMSE is comparatively large.

Spirograph PMHT. It is somewhat surprising that this version of the PMHT is not promising, and appears to be worse than the sliding PMHT.

Adaptive/Homothetic PMHT. This modification shows very little promise. While its *rank* performance is reasonable, its performance in heavy clutter, in slow-moving, and in fast-moving target situations is abysmal.

PMHT with a PDAF Measurement Model. It was expected that there was little difference between the PMHT measurement model and that of the PDAF, and indeed that was found.

Detection-Oriented PMHTs. Particularly the second of these, in which the event that no detections arrived in $P = 3$ scans was made infeasible, was a strong performer. Its results were similar to those of the homothetic PMHT. This is a promising PMHT.

Endpoint PMHT. This PMHT is not promising. While its ranks are reasonable, its better performances tended to occur in situations in which the PDAF did much better. It is apparently strongly negatively affected both by a fast target and by a target with high process noise.

Startpoint PMHT. Particularly in terms of RMSE, but also to a lesser degree in lost-tracks, this is a promising PMHT.

Maneuver PMHT. Despite its rank statistics this PMHT does well, particularly in those situations in which the PDAF is poor.

Independent Homothetic/Maneuver PMHT. This PMHT is better than the previous, and is a particularly strong performer.

Joint Homothetic/Maneuver PMHT. The performance of this PMHT was disappointing. It is felt that some degree of tuning is possible, hence it will not be abandoned.

Deflationary PMHT. The performance of this convergence-aided PMHT is not uniformly good, but is intriguing. Of particular note is its strong performance for the slow-moving target, and correspondingly awful performance for the fast-moving target.

Homoscedastic PMHT. This convergence aid works well, but is perhaps slightly inferior to the deflationary PMHT. Improvement may be possible with some tuning.

5 Summary

The PMHT is a neat tracking algorithm, but has not shown its superiority to the simple PDAF. In this paper we have tried an extensive number of modifications to the PMHT, with the goal of finding those which are promising enough to pursue and "tune". The situation explored is of a single kinematic model in clutter, and is linear both plant and measurement. We have attempted to keep parameters in line with active sonar practice.

Before describing the modifications the basic PMHT's "problems" were discussed in some detail. These include the PMHT's "nonadaptivity", meaning that unlike the PDAF its look-region is not data-adaptive; the PMHT's "narcissism", meaning that the PMHT is unwilling to believe its track lost even in the face of seemingly convincing evidence to the contrary in terms of a long string of missed detections; and the PMHT's "hospitality" to a perceived plurality of detections, many of which can be clutter.

We make a number of observations:

- The PMHT is improved by a sliding batch.
- The PMHT is improved by a homothetic measurement model.
- Whatever problems the PMHT may have, they do not appear to have much to do with the measurement model: if a PDAF-like measurement model is employed there is little difference in performance.
- A version of the PMHT which makes the event that several scans in a row are all detection-free infeasible is a particularly strong performer, and worthy of further study.
- A version of the PMHT which "tracks" only beginning value of the batch appears to be a strong performer, not only in terms of lost tracks but also, somewhat surprisingly, in terms of RMSE.
- The use of a maneuver model, particularly a maneuver model with a homothetic measurement, seems to be a good idea even when the target does not maneuver. The ability of the track to execute a maneuver appears to encourage the PMHT to be less narcissistic.
- A PMHT model in which the maneuver process and homothetic model are probabilistically linked has not shown good performance. However, our experience is that this is a particularly hard PMHT to tune, and we will explore it further.
- Our attempts at automatic aiding of convergence of the PMHT to a global likelihood maximum were not successful.
- Several PMHTs are consistently better than the PDAF in terms of RMSE for tracks which are not lost.
- In extremely adverse situations certain PMHTs, especially those noted above and particularly the maneuvering PMHTs, are much better than the PDAF in terms of lost tracks.

- For the most part, the PDAF remains a better and more dependable tracker than the PDAF.

The last bullet is the most important, and is also the most disappointing. However, we are encouraged by certain of the PMHTs above, and shall continue attempting improvement.

Acknowledgement

This research has been supported by the Office of Naval Research through NUWC, Division Newport, under contract N66604-98-M-3735.

References

[1] Y. Bar-Shalom, X.R. Li, *Estimation and Tracking: Principles, Techniques and Software*, Artech House, Inc., 1993.

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

[3] R.L. Streit and T.E. Luginbuhl, "Probabilistic Multi-Hypothesis Tracking", *NUWC-NPT Technical Report 10,428*, February 1995.

[4] C. Rago, P. Willett, and R. Streit, "Direct Data Fusion Using the PMHT", *Proceedings of the 1995 American Control Conference*, June 1995.

[5] E. Giannopoulos, R. Streit, and P. Swaszek, "Probabilistic Multi-Hypothesis Tracking in a Multi-Sensor, Multi-Target Environment", em First Australian Data Fusion Symposium, November, 1996.

[6] A. Logothetis, V. Krishnamurthy, and J. Holst, "On Maneuvering Target Tracking via the PMHT", *Proceedings of the Conference on Decision and Control*, December 1997.

[7] Y. Ruan, P. Willett, and R. Streit, "The PMHT for Maneuvering Targets", *Proceedings of the 1998 SPIE Conference on Signal and Data Processing of Small Targets*, April 1998.

[8] P. Willett, Y. Ruan, and R. Streit, "A Variety of PMHTs" *University of Connecticut Technical Report TR 98-04*, October 1998.

[9] R. Hutchins, *private communication*, April 1998.

[10] C. Rago, P. Willett, and R. Streit, "A Comparison of the JPDAF and PMHT Tracking Algorithms", *Proceedings of the 1995 International Conference on Acoustics, Speech, and Signal Processing*, May 1995.

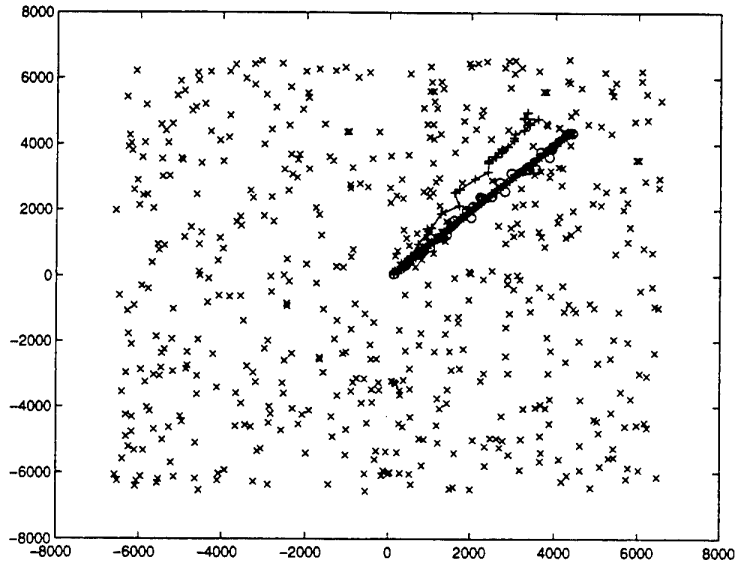


Figure 8: The true, PMHT and PDAF tracks with $\lambda = 10^{-5.5}$, $P_d = 90\%$, $\sigma_p = .0001$, $\sigma_m = 100$, and initial speed 13.8 knots. Detections are $\circ$'s, and the *final scan only* of clutter (i.e. $t = T$) is shown, with clutter returns denoted by $\times$.

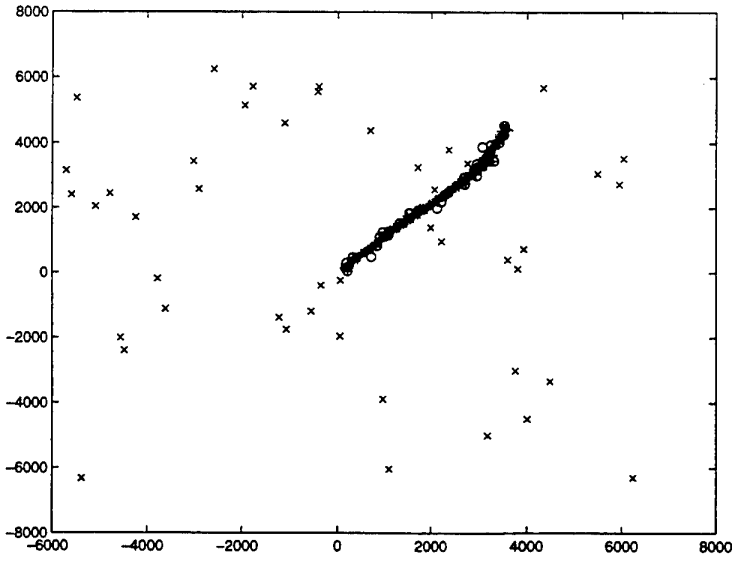


Figure 9: The true, PMHT and PDAF tracks with $\lambda = 10^{-6.5}$, $P_d = 90\%$, $\sigma_p = .01$, $\sigma_m = 100$, and initial speed 13.8 knots. Detections are $\circ$'s, and the *final scan only* of clutter (i.e. $t = T$) is shown, with clutter returns denoted by $\times$.

	$\sigma_p = .01$			$\sigma_p = .001$			$\sigma_p = .0001$		
	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$
PDAF	1	4	19	4	4	29	3	10	29
PMHT (no slide)	3	7	14	6	10	20	13	12	18
PMHT (sliding)	6	8	23	10	18	30	6	8	26
Homothetic PMHT	9	13	32	6	16	25	8	12	18
Spirograph PMHT	4	10	20	5	10	18	7	15	16
Adap./Hom. PMHT	0	0	2	0	2	2	1	1	3
PMHT (PDAF meas)	4	14	25	8	9	25	5	17	33
D.O. PMHT (P=1)	7	11	24	9	18	33	7	11	28
D.O. PMHT (P=3)	6	7	21	10	17	29	5	8	25
Endpoint PMHT	0	0	1	25	43	54	24	41	54
Startpoint PMHT	4	11	19	9	18	40	14	21	21
Maneuver PMHT	19	32	50	25	34	50	24	35	58
Man. PMHT (hom)	16	41	50	19	37	59	20	33	59
Man. PMHT (jnt)	2	7	13	2	10	24	4	12	9
Deflat. PMHT	6	16	37	15	14	34	7	18	28
Homosc. PMHT	11	31	25	13	23	34	11	21	30

Table 1: Illustration of the performance of the various tracking approaches, in terms of *in*-track percentage. The situation here is of initial speed 13.8 knots, clutter density $10^{-5.5}$ per square meter, and $\sigma_m = 100$ meters.

	$\sigma_p = .01$			$\sigma_p = .001$			$\sigma_p = .0001$		
	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$
PDAF	60	86	94	66	90	95	64	89	94
PMHT (no slide)	32	46	67	45	62	69	40	65	64
PMHT (sliding)	33	57	75	47	67	72	46	71	76
Homothetic PMHT	53	70	91	51	76	87	49	68	81
Spirograph PMHT	42	56	76	42	58	76	43	67	69
Adap./Hom. PMHT	6	23	42	15	39	62	19	31	49
PMHT (PDAF meas)	34	64	70	44	54	74	38	65	74
D.O. PMHT (P=1)	19	45	74	25	45	82	23	50	81
D.O. PMHT (P=3)	35	68	81	49	70	76	46	75	80
Endpoint PMHT	9	12	31	50	69	88	57	76	89
Startpoint PMHT	31	61	89	47	64	81	44	59	85
Maneuver PMHT	35	55	71	45	67	88	57	69	84
Man. PMHT (hom)	49	68	87	49	68	85	46	80	93
Man. PMHT (jnt)	40	68	91	43	73	89	40	71	86
Deflat. PMHT	40	61	85	44	72	90	52	74	85
Homosc. PMHT	43	71	88	54	72	82	37	67	81

Table 2: Illustration of the performance of the various tracking approaches, in terms of *in*-track percentage. The situation here is of initial speed 13.8 knots, clutter density $10^{-6.5}$ per square meter, and $\sigma_m = 100$ meters.

	$\sigma_p = .01$			$\sigma_p = .001$			$\sigma_p = .0001$		
	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$
PDAF	410	410	260	390	360	230	410	350	230
PMHT (no slide)	300	280	230	330	190	120	180	120	110
PMHT (sliding)	140	220	200	230	200	130	270	220	130
Homothetic PMHT	330	270	220	390	330	240	310	210	230
Spirograph PMHT	260	300	180	280	290	200	390	280	360
Adap./Hom. PMHT	—	—	350	—	230	320	290	520	280
PMHT (PDAF meas)	310	170	170	190	170	140	300	150	100
D.O. PMHT (P=1)	230	270	170	180	270	120	280	210	140
D.O. PMHT (P=3)	140	220	170	240	150	130	250	220	130
Endpoint PMHT	—	—	70	160	130	90	190	150	110
Startpoint PMHT	350	330	300	260	190	170	260	210	160
Maneuver PMHT	140	180	140	160	120	80	200	120	100
Man. PMHT (hom)	260	210	180	260	210	120	250	140	170
Man. PMHT (jnt)	200	260	250	290	340	350	340	340	240
Deflat. PMHT	280	180	170	250	170	90	260	230	140
Homosc. PMHT	220	240	170	240	140	150	350	170	130

Table 3: Illustration of the performance of the various tracking approaches, in terms of RMSE *at the final scan* for those tracks not considered lost. The situation here is of initial speed 13.8 knots, clutter density $10^{-5.5}$ per square meter, and $\sigma_m = 100$ meters.

	$\sigma_p = .01$			$\sigma_p = .001$			$\sigma_p = .0001$		
	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$	$P_d = .5$	$P_d = .7$	$P_d = .9$
PDAF	150	100	90	120	70	50	120	60	50
PMHT (no slide)	120	110	110	130	70	70	90	80	60
PMHT (sliding)	150	100	100	140	80	60	120	90	70
Homothetic PMHT	170	130	110	190	100	80	180	90	110
Spirograph PMHT	140	110	120	170	110	100	190	160	110
Adap./Hom. PMHT	330	330	320	370	390	310	360	370	360
PMHT (PDAF meas)	140	100	80	80	70	60	90	70	60
D.O. PMHT (P=1)	200	170	120	280	220	120	240	250	140
D.O. PMHT (P=3)	130	120	100	130	90	70	140	90	70
Endpoint PMHT	140	110	120	80	80	80	120	90	80
Startpoint PMHT	220	200	230	140	70	70	140	90	60
Maneuver PMHT	110	130	110	80	90	80	120	90	80
Man. PMHT (hom)	160	170	110	150	130	90	160	130	100
Man. PMHT (jnt)	200	170	140	190	160	120	230	180	110
Deflat. PMHT	90	100	90	100	60	60	120	70	60
Homosc. PMHT	110	110	100	90	70	60	130	70	60

Table 4: Illustration of the performance of the various tracking approaches, in terms of RMSE *at the final scan* for those tracks not considered lost. The situation here is of initial speed 13.8 knots, clutter density $10^{-6.5}$ per square meter, and $\sigma_m = 100$ meters.

	in-track	RMSE
PDAF	4.6	7.3
PMHT (no slide)	11.6	5.9
PMHT (sliding)	8.8	6.4
Homothetic PMHT	5.9	11.3
Spirograph PMHT	10.6	11.2
Adap./Hom. PMHT	7.8	7.7
PMHT (PDAF meas)	8.9	7.5
D.O. PMHT (P=1)	8.2	5.7
D.O. PMHT (P=3)	5.2	10.2
Endpoint PMHT	8.1	9.6
Startpoint PMHT	10.7	3.9
Maneuver PMHT	11.8	12.0
Man. PMHT (hom)	8.7	12.9
Man. PMHT (jnt)	13.7	15.3
Deflat. PMHT	5.3	4.1
Homosc. PMHT	5.9	5.0

Table 5: The average rank of each of the 16 algorithms in each of the 63 situations tested. The first column is in terms of in-track percentage, and the second in terms of RMSE.