

Joint Probabilistic Data Association for Autonomous Navigation

JEAN DEZERT, Member, IEEE
YAAKOV BAR-SHALOM, Fellow, IEEE
University of Connecticut

The problem dealt with here is that of estimating the kinematic state components of a vehicle in autonomous navigation using range and bearing measurements of landmarks detected in the field of view (FOV) of an on-board sensor. The estimates of the absolute position and velocity of the vehicle are provided by a new algorithm developed here and called joint probabilistic data association navigation filter (JPDAF). This algorithm, which is a new extension of the joint probabilistic data association filter (JPDAF) algorithm used in the multitarget tracking, evaluates, at each time step, all the possible associations of the landmark detections in the FOV with a set of stored navigation landmarks. This can be an efficient alternative to storing and correlating full area maps.

I. INTRODUCTION

We present a new application of the joint probabilistic data association (JPDA) method for solving autonomous navigation problems using landmark range and bearing measurements from an on-board sensor. The algorithm developed in the following is referred to as the joint probabilistic data association navigation filter (JPDAF). The probabilistic data association (PDA) and JPDA methods, which have been shown powerful for tracking single and multiple targets in clutter [2, 4, 5], have been recently found also useful for autonomous navigation problems [9]. However, only few investigations in this direction have been made up to now (see [7] for a general survey of commonly used autonomous navigation methods).

The basic autonomous navigation problem consists of estimating the kinematic components of the vehicle in navigation in order to guide it automatically toward some specific aimpoints chosen a priori on a given navigation map stored onboard. The update of the state estimate usually utilizes the information contained in the (relative) range and bearing measurements obtained by the sensor from landmarks detected in its field of view (FOV) during the mission and the information given by a stored navigation map. The navigation map is actually a register of landmarks which might possibly be flown over and which are referenced by their geographical coordinates.

The originality of the JPDAF algorithm lies in the simultaneous use of multiple landmark detections with appropriately quantified confidences (probabilities) to estimate the state of the vehicle. The ambiguities of association between the landmarks detected by the sensor and those referenced on the navigation map are taken into account by the JPDA method [2]. This new autonomous navigation algorithm has a simple real-time structure and a relatively low computational load and offers an alternative to the traditional methods using image correlation techniques which have high computational requirements. It may also offer advantages for mission planning and changes. The previous investigation of this problem was based only on the PDA method; it used only a single landmark detection in the FOV at a given time ([5, ch. 9]).

Section II describes the state and measurement models and formulates the autonomous navigation problem. Section III presents the solution of the navigation problem in the case where perfect data association is assumed. Section IV presents the JPDAF algorithm which solves the autonomous navigation problem for applications where there exist many true and false landmark detections at the same time in the FOV of the vehicle. Then, using a vehicle dynamic model and guidance law, Monte

Manuscript received May 20, 1992; revised October 20, 1992.

IEEE Log No. T-AES/29/4/10978.

This work was supported by the Office of Naval Research under Grant N00014-91-J-1950.

Dr. Dezert is sponsored by the European Space Agency and is visiting with the ESP Laboratory of the University of Connecticut.

Authors' current address: J. Dezert, Laboratory of Electronics, Signals, and Images, University of Orléans, Orléans, France; Y. Bar-Shalom, Department of Electrical and Systems Engineering, Room 312, U-157, University of Connecticut, Storrs, CT 06269-3157.

0018-9251/93/\$3.00 © 1993 IEEE

Carlo simulation results of the JPDANF are given in Section V to show the ability of the algorithm to guide automatically the vehicle toward a specific aimpoint on the map.

II. PROBLEM FORMULATION

The dynamics of the vehicle in autonomous navigation are modeled by the equation

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \Gamma(k)\mathbf{v}(k) \quad (1)$$

where $\mathbf{x}(k)$ is the n -dimensional state vector, $\mathbf{u}(k)$ is the known input vector evaluated from the guidance law, $\mathbf{v}(k)$ is process noise, assumed to be normally distributed with mean zero and known covariance $\mathbf{Q}(k)$, and the matrices $\mathbf{F}(k)$, $\mathbf{G}(k)$, and $\Gamma(k)$ are assumed to be known.

The range and bearing measurement relative to a specific landmark L_j detected at time k in the FOV of the sensor onboard is defined by (here “ $'$ ” denotes transposition)

$$\mathbf{z}_j(k) \triangleq [r(k, j), \zeta(k, j)]'. \quad (2)$$

The range measurement $r(k, j)$ between the position $(x(k), y(k))$ of the vehicle and the position $(x_j(k), y_j(k))$ of the detected landmark L_j is given by the nonlinear equation

$$r(k, j) = \{[x(k) - x_j(k)]^2 + [y(k) - y_j(k)]^2\}^{1/2} + \eta_r(k) \quad (3)$$

where $\eta_r(k)$ is a range noise assumed to be normally distributed with mean zero and known variance $\sigma_r^2(k)$. The bearing measurement $\zeta(k, j)$ which represents the angle between the line of sight (LOS) of the detected landmark L_j and a reference direction (chosen for convenience as the East direction in the sequel) is given by the nonlinear equation

$$\zeta(k, j) = \arctan \left[\frac{y(k) - y_j(k)}{x(k) - x_j(k)} \right] + \eta_\zeta(k) \quad (4)$$

where $\eta_\zeta(k)$ is a bearing noise assumed to be normally distributed with mean zero and known variance $\sigma_\zeta^2(k)$. Hence, the range and bearing measurement can be modeled as

$$\mathbf{z}_j(k) = \mathbf{h}[\mathbf{x}(k), \mathbf{L}_j(k)] + \eta(k) \quad (5)$$

where $\mathbf{L}_j(k)$ is the position $(x_j(k), y_j(k))$ of landmark L_j detected at time k , $\eta(k) = [\eta_r(k), \eta_\zeta(k)]'$ is the range-bearing noise vector assumed to be normally distributed with zero mean and known covariance matrix $\mathbf{R}_\eta(k) = \text{diag}(\sigma_r^2(k), \sigma_\zeta^2(k))$ and $\mathbf{h}[\mathbf{x}(k), \mathbf{L}_j(k)]$ is the nonlinear measurement vector function given by

$$\mathbf{h}[\mathbf{x}(k), \mathbf{L}_j(k)] \triangleq \begin{bmatrix} h_1[\mathbf{x}(k), \mathbf{L}_j(k)] \\ h_2[\mathbf{x}(k), \mathbf{L}_j(k)] \end{bmatrix} \quad (6)$$

with

$$h_1[\mathbf{x}(k), \mathbf{L}_j(k)] = [(x(k) - x_j(k))^2 + (y(k) - y_j(k))^2]^{1/2} \quad (7)$$

$$h_2[\mathbf{x}(k), \mathbf{L}_j(k)] = \arctan \left[\frac{y(k) - y_j(k)}{x(k) - x_j(k)} \right] \quad (8)$$

In the following, we assume the landmarks (which are usually chosen as specific points of the landscape) to be fixed during the navigation and the range and bearing noises to be stationary, that is to say

$$\mathbf{L}_j(k) = \mathbf{L}_j = (x_j, y_j) \quad (9)$$

$$\mathbf{R}_\eta(k) = \mathbf{R}_\eta = \text{diag}(\sigma_r^2, \sigma_\zeta^2). \quad (10)$$

The range-bearing measurement $\mathbf{z}_j(k)$ can be transformed into corresponding Cartesian measurement

$$\mathbf{y}_j(k) = \mathbf{f}[\mathbf{z}_j(k)] \quad (11)$$

where

$$\mathbf{f}[\mathbf{z}_j(k)] \triangleq \begin{bmatrix} r(k, j) \cos(\zeta(k, j)) \\ r(k, j) \sin(\zeta(k, j)) \end{bmatrix}. \quad (12)$$

Assuming the range and bearing noises to be small (i.e., $\sigma_\zeta \ll 1$ and $\sigma_r \ll r(k)$) and perfect knowledge of the position of the detected landmark L_j at time k , the Cartesian measurement equation (11) can be expressed by the linear observation equation

$$\mathbf{y}_j(k) = \mathbf{H}(k)\mathbf{x}_j(k) + \mathbf{L}_j(k) + \mathbf{w}_j(k) \quad (13)$$

where $\mathbf{H}(k)$ is the observation matrix chosen as

$$\mathbf{H}(k) \triangleq \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad (14)$$

if the kinematic state vector is chosen as $\mathbf{x}(k) = [x(k), \dot{x}(k), y(k), \dot{y}(k)]'$. The measurement noise vector $\mathbf{w}_j(k)$ is assumed normally distributed with mean zero and covariance matrix $\mathbf{R}(k, j)$ approximately given by [1].

$$\begin{aligned} \mathbf{R}(k, j) &= g[\mathbf{z}_j(k), \mathbf{R}_\eta] \\ &= \begin{bmatrix} r^2(k, j) \sin^2(\zeta(k, j)) \sigma_\zeta^2 + \cos^2(\zeta(k, j)) \sigma_r^2 \\ [\sigma_r^2 - r^2(k, j) \sigma_\zeta^2] \cos(\zeta(k, j)) \sin(\zeta(k, j)) \\ [\sigma_r^2 - r^2(k, j) \sigma_\zeta^2] \cos(\zeta(k, j)) \sin(\zeta(k, j)) \\ r^2(k, j) \cos^2(\zeta(k, j)) \sigma_\zeta^2 + \sin^2(\zeta(k, j)) \sigma_r^2 \end{bmatrix} \end{aligned} \quad (15)$$

In practical autonomous navigation applications, the information available at time k from the on-board sensor consists of n_k relative range and bearing measurements corresponding to the n_k landmarks detected in its FOV (whatever they are: referenced landmarks on the navigation map or some other unknown landmarks which may be considered as false alarms).

The set of range-bearing measurements available at time k is denoted

$$\mathbf{Z}(k) \triangleq \{z_j(k)\}_{j=1}^{n_k} \quad (16)$$

to which we can associate the set of Cartesian measurements $\mathbf{Y}(k)$ defined as

$$\mathbf{Y}(k) \triangleq \{\mathbf{f}[z_j(k)]\}_{j=1}^{n_k}. \quad (17)$$

We denote the accumulated Cartesian measurements at time k

$$\mathbf{Y}^k \triangleq \{\mathbf{Y}(i)\}_{i=1}^k = \{\mathbf{Y}^{k-1}, \mathbf{Y}(k)\}. \quad (18)$$

The general autonomous navigation problem consists of evaluating the best estimate (in the minimum variance sense) of the state of the vehicle given all the information available up to time k (i.e., the measurements and the navigation map). We are interested in the recursive computation of the conditional mean of the state, $\hat{\mathbf{x}}(k | k)$, and its associated covariance matrix, $\mathbf{P}(k | k)$,

$$\hat{\mathbf{x}}(k | k) = E[\mathbf{x}(k) | \mathbf{Y}^k] \quad (19)$$

$$\mathbf{P}(k | k) = E[(\mathbf{x}(k) - \hat{\mathbf{x}}(k | k))(\mathbf{x}(k) - \hat{\mathbf{x}}(k | k))' | \mathbf{Y}^k]. \quad (20)$$

III. AUTONOMOUS NAVIGATION FILTER USING THE JPDA APPROACH

We recall that in the case of perfect data association (known origin of each measurement), the solution of the autonomous navigation problem is given by the hierarchical Kalman filtering algorithm [6, 3]. Although the algorithm can also be implemented via sequential processing [3] (with certain caution due to the different measurement accuracies), the hierarchical structure is, however, natural for the situation of uncertain origin detections, as will be shown.

A. Fundamental Assumptions and Feasible Joint Events

Because of the uncertainty of the state vehicle and environment conditions, in practical situations we do not know exactly the origin of the measurement obtained by the on-board sensor. So, the previous autonomous navigation algorithm cannot be applied in realistic situations. Nevertheless, the hierarchical scheme outlined above is natural to combine the updated states obtained with separate JPDAFs, each using a cluster of detections, as discussed in the sequel. To solve the autonomous navigation problem we have to take into account all feasible joint association hypotheses concerning the origin of the detected landmarks. In the sequel, the following assumptions are made.

Assumption 1: Each range-bearing measurement is originated from only one landmark.

Assumption 2: Each referenced landmark can give rise to at most one measurement.

Assumption 3: The detected landmarks which may correspond to nonreferenced landmarks are uniformly distributed on the navigation map (in the Cartesian space).

Assumption 4: The conditional probability density function (pdf) of the state given the past measurements is assumed to be Gaussian.

We define the joint association events at the current time k

$$\theta_q(k) = \bigcap_{j=1}^{n_k} \theta_{jl}(k) \quad (21)$$

where $\theta_{jl}(k)$ ($j = 1, \dots, n_k$, $l = 0, \dots, N_k$) denote the individual association event that $z_j(k)$ (or equivalently $y_j(k)$ since there exist a one-to-one correspondence between $z_j(k)$ and $y_j(k)$) is from landmark L_l at time k ; n_k is the number of range and bearing measurements obtained by the sensor at time k (i.e., number of landmark detections); N_k is the number of the referenced landmarks of the navigation map which are supposed to be detected in the extended field of view (EFOV is augmented by the uncertainty in the vehicle position¹ at time k); l denotes the index of the referenced landmark to which measurement j is associated in the event under consideration; $l = 0$ corresponds to the particular hypothesis in which measurement j originated from an unknown landmark (a nonreferenced landmark) or a false alarm.

From the set of measurements $\mathbf{Y}(k)$ available at time k and the knowledge of the landmarks L_1, L_2, \dots, L_{N_k} supposed to be detected in the EFOV at time k , we define as in the JPDAF [2] the validation matrix

$$\Omega(k) \triangleq [\omega_{jl}] \quad j = 1, \dots, n_k; \quad l = 0, 1, \dots, N_k. \quad (22)$$

The binary elements ω_{jl} of Ω indicate if measurement j lies in the validation gate [2] of landmark l . The first column of Ω corresponding to index $l = 0$ (which stands for "no landmark") has all units since each measurement could have originated from a nonreferenced landmark or false alarm. Actually the validation matrix Ω can be considered as a snapshot representation of the validated measurements at the observation time as depicted in Fig. 1 for example. A typical validation matrix for seven predicted landmarks in the EFOV of the vehicle and ten measurements corresponding to the situation depicted in Fig. 1 looks

¹This will be typically evaluated from the knowledge of the predicted vehicle state and its covariance, the geometry of the FOV and the navigation map (see Appendix A for the EFOV derivation).

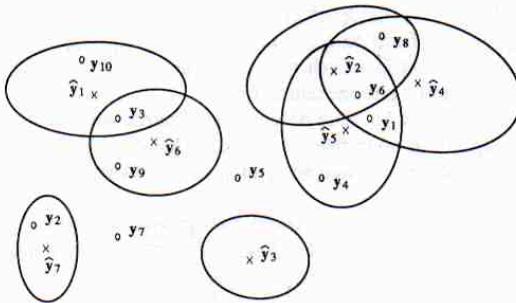


Fig. 1. Seven predicted landmarks with ten measurements.

as follows:

$$\Omega(k) = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

By using the JPDA logic that considers only the joint events made up of marginal events involving validated measurements [1], each joint feasible association event $\theta_q(k)$ can be represented by the following $n_k \times (N_k + 1)$ association matrix

$$\hat{\Omega}_q = [\hat{\omega}_{jl}(\theta_q(k))] \quad (23)$$

consisting of the units corresponding to the association in $\theta_q(k)$, that is

$$\hat{\omega}_{jl}(\theta_q(k)) = \begin{cases} 1 & \text{if } \theta_{jl}(k) \subset \theta_q(k) \\ 0 & \text{otherwise} \end{cases} \quad (24)$$

satisfying (according to Assumption 1)

$$\sum_{l=0}^{N_k} \hat{\omega}_{jl}(\theta_q(k)) = 1; \quad \text{for all } j \quad (25)$$

and (according to Assumption 2)

$$\delta_l(\theta_q) \triangleq \sum_{j=1}^{n_k} \hat{\omega}_{jl}(\theta_q(k)) \quad \text{for } l = 1, \dots, N_k. \quad (26)$$

The binary variable $\delta_l(\theta_q)$ is called the landmark detection indicator because it indicates whether the referenced landmark l has been detected or not under the joint association event θ_q . To indicate if measurement j is associated with a referenced landmark on the map in event θ_q , we define the binary measurement association indicator

$$\tau_j(\theta_q) \triangleq \sum_{l=1}^{N_k} \hat{\omega}_{jl}(\theta_q(k)) \quad \text{for } j = 1, \dots, n_k. \quad (27)$$

With this definition, the number of unassociated (false) measurements in event θ_q is then

$$\phi(\theta_q) \triangleq \sum_{j=1}^{n_k} [1 - \tau_j(\theta_q)]. \quad (28)$$

Thus, the generation of the association matrices $\hat{\Omega}_q$ corresponding to the feasible events θ_q can be done by scanning Ω and picking one unit per row and per column except for column $l = 0$, where the number of units is not restricted. The generation algorithm of association matrices $\hat{\Omega}_q$ and its Fortran implementation may be found in [8] and more recently in [10] for its fast implementation using the depth first search approach (DFS algorithm).

B. Clustering Approach to Reduce Number of Computations Involved

From a practical point of view, the number of association matrices $\hat{\Omega}_q$ to generate (and hence the number of joint association probabilities) can be very high, and the direct use of the global initial validation matrix for generating all association matrices is expensive. Actually, we can reduce the number of computations by using the clustering decomposition of the matrix $\Omega(k)$. Each cluster is used in a JPDA as a step in the hierarchical updating scheme of [6].

The clustering decomposition of any validation matrix involved in the JPDA logic first compresses the initial validation matrix $\Omega(k)$ by removing all its null rows (first element excluded) and columns, and then, decomposes the compressed matrix in a set of different validation submatrices associated to each independent cluster present in the validation configuration. A cluster is defined as a group of overlapping validation gates (interfering targets in multitarget tracking applications) which have some measurements in common. The decomposition algorithm consists of merging by the Boolean OR operator of all rows of the validation matrix (without the first column of $\Omega(k)$ corresponding to L_0) which have some units in common in a column of the clustering matrix $\Omega_c(k)$ of the configuration. Hence, each row of the clustering matrix characterizes an independent cluster present in the validation configuration. This efficient algorithm allows us to find all the independent clusters in a configuration in exactly N steps if N corresponds to the number of predicted landmarks in the EFOV (or the number of targets to track simultaneously in multitarget tracking applications). We give the Fortran implementation of the clustering decomposition algorithm (CDA) in Appendix B.

To illustrate the clustering decomposition method, consider the previous validation matrix corresponding to Fig. 1. We get, after the elimination of the null rows and columns of $\Omega(k)$, the compressed validation matrix

$$\Omega(k) = \begin{bmatrix} 0 & 1 & 2 & 4 & 5 & 6 & 7 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ 8 \\ 9 \\ 10 \end{array}$$

The first step of construction of the clustering matrix $\Omega_c(k)$ consists to set $\Omega_c(k)$ equal to $\Omega(k)$ (the column corresponding to L_0 has been removed here for convenience since it is never processed in the CDA) and to merge all the rows of $\Omega_c(k)$ having a unit in the column associated to landmark L_1 (here, the 3rd and the 8th rows). We get the following clustering matrix for step one of the clustering decomposition algorithm

$$\Omega_c(k) = \begin{bmatrix} 1 & 2 & 4 & 5 & 6 & 7 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

This operation repeats until all columns of $\Omega_c(k)$ have been processed gives the following sequence of matrices

$$\Omega_c(k) = \begin{bmatrix} 1 & 2 & 4 & 5 & 6 & 7 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\Omega_c(k) = \begin{bmatrix} 1 & 2 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\Omega_c(k) = \begin{bmatrix} 1 & 2 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

and, finally

$$\Omega_c(k) = \begin{bmatrix} 1 & 2 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{array}{l} \text{Cluster 1} \\ \text{Cluster 2.} \\ \text{Cluster 3} \end{array}$$

The first row of the clustering matrix tell us that the validation gates associated to landmarks L_2 , L_4 , and L_5 constitute the first cluster. Since there are three validations gates in this cluster, we say in the sequel that its size is equal to three. More generally, we define the size of a given cluster as the number of validation gates included in it. The second row of $\Omega_c(k)$ indicates that there is only one gate (the gate associated to predicted landmark L_7) which is the second cluster. Its size is then one. The third and last cluster for the validation configuration of Fig. 1 is composed of the validation gates associated to landmarks L_1 and L_6 .

C. Joint Probabilistic Data Association

From the clustering matrix of the validation configuration, we are able to evaluate separately and with a minimum cost of computation all the feasible joint association events within each cluster. The evaluation of the possible association events is based on the JPDA approach every time the size of a cluster is greater than 1. Otherwise the PDA approach is used for the evaluation of the posterior association probabilities. Across the clusters the overall estimate is obtained via the hierarchical scheme discussed earlier. We recall now the final expression of the association probabilities for the PDA and JPDA approach. The reader interested in the complete derivation of these probabilities can refer to [2].

For a cluster of size 1 associated to a predicted landmark l the posteriori association probabilities are given by the PDA approach as

$$\beta_{jl}(k) = \mathbb{P}\{\theta_{jl}(k) | \mathbf{Y}^k\} = \frac{e_{jl}}{b_l + \sum_{i=1}^{n_c(k)} e_{il}} \quad \text{for } j = 1, \dots, n_c(k) \quad (29)$$

$$\beta_{0l}(k) = \mathbb{P}\{\theta_{0l}(k) | \mathbf{Y}^k\} = \frac{b_l}{b_l + \sum_{i=1}^{n_c(k)} e_{il}} \quad (30)$$

where

$$e_{jl} = \exp\{-1/2[\mathbf{y}_j(k) - \hat{\mathbf{y}}(k | k-1, l)]' \times [\mathbf{S}(k, l)]^{-1} [\mathbf{y}_j(k) - \hat{\mathbf{y}}(k | k-1, l)]\} \quad (31)$$

$$b_l = \lambda |\mathbf{S}(k, l)|^{-1/2} (1 - \mathbb{P}_d(l)\mathbb{P}_g)/\mathbb{P}_d(l) \quad (32)$$

and λ represents the spatial density of unknown landmarks per unit area; $\mathbb{P}_d(l)$ is the probability of detection of the predicted landmark l associated to the cluster in consideration; \mathbb{P}_g is the chosen probability mass for the statistical gating of the measurements; $n_c(k)$ is the number of validated measurements for the cluster; $\mathbf{y}_j(k)$ $j = 1, \dots, n_c(k)$ are the validated Cartesian measurements relative to the cluster; $\hat{\mathbf{y}}(k | k-1, l)$ is the predicted Cartesian measurement of landmark l (i.e., the center of the validation

gate) and $\mathbf{S}(k, l)$ is the predicted covariance matrix of the measurement. The events $\theta_{jl}(k)$, $j = 0, 1, \dots, n_c(k)$, represent the elementary association events of the validation configuration for the cluster of size one under consideration relative to the predicted landmark l .

For a cluster of size N ($N > 1$) the marginal association probabilities corresponding to (29)–(30) are obtained from the joint association probabilities (evaluated by the JPDA approach) by summing over all the joint events in which the marginal event of interest occur. Using the definition of the association matrix $\hat{\Omega}_q$, this summation can be written as: (for $l = 1, \dots, N$)

$$\beta_{jl}(k) = \mathbf{P}\{\theta_{jl}(k) | \mathbf{Y}^k\} = \sum_{\theta_q} \mathbf{P}\{\theta_q(k) | \mathbf{Y}^k\} \hat{\omega}_{jl}(\theta_q(k)) \quad \text{for } j = 1, \dots, n_c(k) \quad (33)$$

$$\beta_{0l}(k) = \mathbf{P}\{\theta_{0l}(k) | \mathbf{Y}^k\} = 1 - \sum_{j=1}^{n_c(k)} \beta_{jl}(k) \quad (34)$$

where the joint association probabilities $\mathbf{P}\{\theta_q(k) | \mathbf{Y}^k\}$ are given in [2]

$$\begin{aligned} \mathbf{P}\{\theta_q(k) | \mathbf{Y}^k\} &= \frac{1}{c} \prod_{j=1}^{n_c(k)} \left\{ \frac{1}{\lambda} f_{l_j}[\mathbf{y}_j(k)] \right\}^{\tau_j(\theta_q)} \\ &\times \prod_{l=1}^N [\mathbf{P}_g \mathbf{P}_d(l)]^{\delta_l(\theta_q)} [1 - \mathbf{P}_g \mathbf{P}_d(l)]^{1-\delta_l(\theta_q)} \end{aligned} \quad (35)$$

with

$$\begin{aligned} c &= \sum_{\theta_q} \prod_{j=1}^{n_c(k)} \left\{ \frac{1}{\lambda} f_{l_j}[\mathbf{y}_j(k)] \right\}^{\tau_j(\theta_q)} \\ &\times \prod_{l=1}^N [\mathbf{P}_g \mathbf{P}_d(l)]^{\delta_l(\theta_q)} [1 - \mathbf{P}_g \mathbf{P}_d(l)]^{1-\delta_l(\theta_q)} \end{aligned} \quad (36)$$

$f_{l_j}[\mathbf{y}_j(k)] = N[\mathbf{y}_j(k); \hat{\mathbf{y}}(k | k-1, l_j), \mathbf{S}(k, l_j)]$ represents the Gaussian density with mean $\hat{\mathbf{y}}(k | k-1, l_j)$ and associated innovation covariance $\mathbf{S}(k, l_j)$ evaluated at the measurement point $\mathbf{y}_j(k)$; $\delta_l(\theta_q)$ and $\tau_j(\theta_q)$ are the detection and measurement association indicators, respectively, defined in (26) and (27); $\mathbf{P}_d(l)$ is the probability of detection of predicted landmark l in the cluster under consideration.

D. State Estimation

Using the total probability theorem with respect to all the feasible association events *in the cluster under consideration* of size N having $n_c(k)$ measurements, the local estimate of the state of the vehicle based on a particular predicted landmark l supposed to be detected in the EFOV is given by

$$\hat{\mathbf{x}}(k | k, l) = \hat{\mathbf{x}}(k | k-1) + \mathbf{K}(k, l) \tilde{\mathbf{y}}(k, l) \quad (37)$$

where $\tilde{\mathbf{y}}(k, l)$ is the combined innovation using all the validated measurements of the cluster under consideration. This combined innovation is given by

$$\tilde{\mathbf{y}}(k, l) \stackrel{\Delta}{=} \sum_{j=1}^{n_c(k)} \beta_{jl}(k) \tilde{\mathbf{y}}_j(k) \quad (38)$$

with

$$\tilde{\mathbf{y}}_j(k) \stackrel{\Delta}{=} \mathbf{y}_j(k) - \hat{\mathbf{y}}(k | k-1, l) \quad (39)$$

$$\hat{\mathbf{y}}(k | k-1, l) = \mathbf{H}(k) \hat{\mathbf{x}}(k | k-1) + \mathbf{L}_l(k). \quad (40)$$

The gain matrix $\mathbf{K}(k, l)$ is given by the standard Kalman filter equation

$$\mathbf{K}(k, l) = \mathbf{P}(k | k-1) \mathbf{H}'(k) [\mathbf{S}(k, l)]^{-1} \quad (41)$$

with

$$\mathbf{S}(k, l) = \mathbf{H}(k) \mathbf{P}(k | k-1) \mathbf{H}'(k) + \mathbf{R}(k, l) \quad (42)$$

$$\hat{\mathbf{x}}(k, l) = \mathbf{h}[\hat{\mathbf{x}}(k | k-1), \mathbf{L}_l(k)] \quad (43)$$

$$\mathbf{R}(k, l) = \mathbf{g}[\hat{\mathbf{x}}(k, l), \mathbf{R}_\eta]. \quad (44)$$

The covariance matrix associated with the estimate (37) is given by [2]

$$\begin{aligned} \mathbf{P}(k | k, l) &= \beta_{0l}(k) \mathbf{P}(k | k-1) \\ &+ [1 - \beta_{0l}(k)] \mathbf{P}_c(k | k, l) + \mathbf{P}^*(k, l) \end{aligned} \quad (45)$$

where $\mathbf{P}_c(k | k, l)$ is the covariance update if one has the correct measurement and $\mathbf{P}^*(k, l)$ is the stochastic increment matrix. These matrices are given by:

$$\mathbf{P}_c(k | k, l) = [\mathbf{I} - \mathbf{K}(k, l) \mathbf{H}(k)] \mathbf{P}(k | k-1) \quad (46)$$

$$\begin{aligned} \mathbf{P}^*(k, l) &= \mathbf{K}(k, l) \left[\sum_{j=1}^{n_c(k)} \beta_{jl}(k) \tilde{\mathbf{y}}_j(k) \tilde{\mathbf{y}}_j(k)' \right. \\ &\quad \left. - \tilde{\mathbf{y}}(k, l) \tilde{\mathbf{y}}(k, l)' \right] \mathbf{K}(k, l)' \end{aligned} \quad (47)$$

The *global updated estimate* of the state and its covariance are given by the *hierarchical fusion equations* [6] of all the *local estimates* $\hat{\mathbf{x}}(k | k, l)$ and $\mathbf{P}(k | k, l)$ given by (37) and (45), respectively, that is

$$\mathbf{P}(k | k) = \left[(1 - N_k) \mathbf{P}(k | k-1)^{-1} + \sum_l \mathbf{P}(k | k, l)^{-1} \right]^{-1} \quad (48)$$

$$\begin{aligned} \hat{\mathbf{x}}(k | k) &= \mathbf{P}(k | k) \left[(1 - N_k) \mathbf{P}(k | k-1)^{-1} \hat{\mathbf{x}}(k | k-1) \right. \\ &\quad \left. + \sum_l \mathbf{P}(k | k, l)^{-1} \hat{\mathbf{x}}(k | k, l) \right]. \end{aligned} \quad (49)$$

E. Summary of JPDANF Algorithm

We give now the summary of the algorithm for implementation of the JPDANF developed in the preceding.

Step 0: $\hat{\mathbf{x}}(k | k-1), \mathbf{P}(k | k-1)$ are known from the last step of the previous cycle.

Step 1: Search of the predicted landmarks $\mathbf{L}_l(k), l = 1, \dots, N_k$ in the EFOV (see Appendix A). If $N_k = 0$ then $\hat{\mathbf{x}}(k | k) = \hat{\mathbf{x}}(k | k-1), \mathbf{P}(k | k) = \mathbf{P}(k | k-1)$ and go to the guidance input computation step (step 11).

Step 2: If there is no detection in the FOV (i.e., $n_k = 0$) then $\hat{\mathbf{x}}(k | k) = \hat{\mathbf{x}}(k | k-1), \mathbf{P}(k | k) = \mathbf{P}(k | k-1)$ and go to the guidance input computation step (step 11).

Step 3: If there are detections $\mathbf{Z}(k) \triangleq \{z_j(k)\}_{j=1}^{n_k}$ in the FOV ($n_k > 0$), calculate the set of Cartesian measurements $\mathbf{Y}(k) \triangleq \{f[z_j(k)]\}_{j=1}^{n_k}$.

Step 4: Compute for the landmarks in the EFOV the predicted measurements $\hat{\mathbf{y}}(k, l)$ with their covariance matrix $\mathbf{S}(k, l)$ for $l = 1, \dots, N_k$ as follows

$$\hat{\mathbf{y}}(k | k-1, l) = \mathbf{H}(k) \hat{\mathbf{x}}(k | k-1) + \mathbf{L}_l(k) \quad (50)$$

$$\hat{\mathbf{z}}(k, l) = \mathbf{h}[\hat{\mathbf{x}}(k | k-1), \mathbf{L}_l(k)] \quad (51)$$

$$\mathbf{R}(k, l) = \mathbf{g}[\hat{\mathbf{z}}(k, l), \mathbf{R}_\eta] \quad (52)$$

$$\mathbf{S}(k, l) = \mathbf{H}(k) \mathbf{P}(k | k-1) \mathbf{H}'(k) + \mathbf{R}(k, l). \quad (53)$$

Step 5: Build the validation matrix $\Omega(k) = [\omega_{jl}]$ according the standard validation gating method [2] and compress it as indicated in subsection IIIB.

Step 6: Use the clustering decomposition of $\Omega(k)$ for getting the clustering matrix $\Omega_c(k)$ (as described in subsection IIIB) and then, for each cluster characterized by each row of $\Omega_c(k)$, do the three following steps.

Step 7: Compute the size of the cluster by summing all elements of its row in $\Omega_c(k)$.

Step 8: If the size of the cluster is 1, evaluate all the posterior association probabilities $\beta_{jl}(k)$ (29)–(30) for all the measurements validated in the gate of landmark \mathbf{L}_l associated to the cluster. Then compute the local estimate $\hat{\mathbf{x}}(k | k, l)$ and $\mathbf{P}(k | k, l)$ by (37) and (45).

Step 9: If the size of the cluster is greater than 1, build the validation matrix of this cluster, generate all the association matrices $\hat{\Omega}_q$ and compute the joint probabilities by the JPDA approach (35). Then compute all the marginal association probabilities $\beta_{jl}(k)$ by (33)–(34) and evaluate the local estimates $\hat{\mathbf{x}}(k | k, l)$ and $\mathbf{P}(k | k, l)$ by (37) and (45) for all landmarks l associated with the cluster.

Step 10: Compute the global state estimate and its covariance by the fusion equations (48)–(49).

Step 11: Evaluation of the guidance input $\mathbf{u}(k)$ based on the latest global estimate.

Step 12: Compute the predicted state and its covariance by the prediction equations (31)–(32) and increment index k and go back to the searching predicted landmarks step.

IV. SIMULATION RESULTS

Simulations of the JPDANF were done for a vehicle flying at low altitude modeled in the absence of guidance commands as a nearly constant velocity object in a plane with some process noise that models slight changes in the velocity. The vehicle travels with the nominal speed V_{nom} of 0.250 km/s. Its trajectory starts from point (20 km, 20 km) on the reference navigation map with an initial heading of zero degrees with respect to the East direction. The coordinates of the aimpoint to be reached by the vehicle are $x_a = 50$ km and $y_a = 50$ km. The dynamic equations of the vehicle, discretized over time intervals of length T is

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \Gamma(k)\mathbf{v}(k) \quad (54)$$

with

$$\mathbf{x}(k) = [x(k) \quad \dot{x}(k) \quad y(k) \quad \dot{y}(k)]' \quad (55)$$

$$\mathbf{F}(k) = \mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (56)$$

where $x(k)$ and $y(k)$ are the Cartesian position coordinates and $\mathbf{v}(k)$ is the process noise

$$\mathbf{v}(k) = [v_1(k) \quad v_2(k)]' \quad (57)$$

$$E[\mathbf{v}(k)] = 0 \quad \text{and} \quad E[\mathbf{v}(k)\mathbf{v}(j)'] = \mathbf{Q}\delta_{kj} \quad (58)$$

$$\Gamma(k) = \Gamma = \begin{bmatrix} T/2 & 0 \\ 1 & 0 \\ 0 & T/2 \\ 0 & 1 \end{bmatrix} \quad (59)$$

$$\mathbf{u}(k) = [\mathbf{u}_1(k) \quad \mathbf{u}_2(k)]' \quad (60)$$

$$\mathbf{G}(k) = \mathbf{G} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}. \quad (61)$$

The input vector $\mathbf{u}(k)$ is computed at each sample time in order to adjust the estimated speed of the vehicle to its nominal value and to guide the vehicle toward the chosen aimpoint. To take into account some aerodynamical constraints, a saturation factor f of 1.5 is used to force $\|\mathbf{u}(k)\| \leq fV_{\text{nom}}$. The FOV of the forward-looking sensor of the vehicle consists of an

angular sector having 60 deg for its aperture angle and 3 km for its maximum vision distance $d_{\max v}$. The area of the FOV is then about 4.71 km^2 . The range-bearing measurement of a landmark $L_j(k)$ detected in the FOV at time k is given by

$$z_j(k) = \mathbf{h}[\mathbf{x}(k), L_j(k)] + \eta(k) \quad (62)$$

where $\mathbf{h}[\mathbf{x}(k), L_j(k)]$ is given by (6) and where

$$\begin{aligned} E[\eta(k)] &= 0 & \text{and} & \quad E[\eta(k)\eta(j)'] = \mathbf{R}_\eta \delta_{kj}. \\ (63) \end{aligned}$$

The navigation end test consists of evaluating the time t_{\min} for the vehicle to be at the minimal distance of the aimpoint and then to compare t_{\min} with the sampling period T . The navigation is stopped at time k for which $t_{\min} \leq T$ since at this time, the vehicle is at the minimal distance of the aimpoint to reach; t_{\min} is analytically given by [9]

$$t_{\min} = \frac{\mathbf{d}(k)' \dot{\mathbf{d}}(k)}{\|\dot{\mathbf{d}}(k)\|^2} \quad (64)$$

where $\mathbf{d}(k)$ and $\dot{\mathbf{d}}(k)$ are defined as

$$\mathbf{d}(k) = [x_a - \hat{x}(k | k) \quad y_a - \hat{y}(k | k)]' \quad (65)$$

$$\dot{\mathbf{d}}(k) = [\dot{x}(k | k) \quad \dot{y}(k | k)]'. \quad (66)$$

For evaluation of the JPDANF, the following numerical values were used: $T = 1 \text{ s}$ for the sampling period; $\sqrt{Q_{11}} = \sqrt{Q_{22}} = 2 \cdot 10^{-4} \text{ km/s}$ for the process noise; $\sigma_r = 0.02 \text{ km}$ for the standard deviation of the range measurement noise; and $\sigma_\zeta = 0.003 \text{ rd}$ for the standard deviation of the bearing measurement noise.

The probability of detection of the true referenced landmarks in the FOV varies from $P_{d_{\min}} = 0.9$ (when the landmark is at the maximum distance $d_{\max v}$ of the FOV) to $P_{d_{\max}} = 1$ (when the landmark is at the minimum distance of the on-board sensor). In our simulations, the probability of detection of a landmark in the FOV which lies at the distance $r(k)$ of the sensor is approximated by the simple expression

$$P_d(r(k)) = \exp \left\{ \log(P_{d_{\min}}) \frac{r(k)}{d_{\max v}} \right\}. \quad (67)$$

This relationship gives in our simulations an increasing detection probability when $r(k)$ tends to zero (i.e., when the landmark is close to the sensor) and also

$$P_d(r(k)) = P_{d_{\min}} \quad \text{for } r(k) = d_{\max v} \quad (68)$$

$$P_d(r(k)) = P_{d_{\max}} = 1 \quad \text{for } r(k) = 0. \quad (69)$$

The probability P_g used for the validation gates set up in the JPDANF is chosen to 0.99. The model for the filter is based on the dynamic equations (54)–(61). The initial conditions of the vehicle are given by choosing the initial state estimate $\hat{\mathbf{x}}(0 | 0)$ and its covariance matrix $\mathbf{P}(0 | 0)$ and generating the real initial estimate

$\mathbf{x}(0)$ as

$$\mathbf{x}(0) \sim N(\hat{\mathbf{x}}(0 | 0), \mathbf{P}(0 | 0)). \quad (70)$$

The initial state estimate is chosen as

$$\hat{\mathbf{x}}(0 | 0) = [20 \text{ km} \quad 0.250 \text{ km/s} \quad 20 \text{ km} \quad 0 \text{ km/s}]$$

with the initial covariance matrix

$$\sqrt{\mathbf{P}(0 | 0)}$$

$$= \text{diag}[0.050 \text{ km} \quad 0.005 \text{ km/s} \quad 0.050 \text{ km} \quad 0.005 \text{ km/s}].$$

The autonomous navigation maps consist of a list of landmarks uniformly distributed on a square of 100 km by 100 km with some a priori chosen spatial density λ_l . The distribution of the false alarms (i.e., the landmarks not on the navigation map) is also assumed to be uniform in the FOV with spatial density λ_{fa} . A Monte Carlo simulation of 100 runs was performed for the evaluation of the JPDANF algorithm for different landmark densities λ_l and different false alarm densities λ_{fa} .

Table I provides the results of the autonomous navigation algorithm. The first line gives the different values chosen for the true landmark density λ_l (the number of referenced landmarks per km^2) which varies from 0.1 to 0.8. The second line corresponds to the different values of false alarm densities λ_{fa} (number of false alarms per km^2) which are from 1 to 60 times higher than the true landmark density λ_l . The quantities n_{td} and n_{fa} on the third and fourth lines indicate the average number of true and false detections in the FOV, respectively, at each sample time. The fifth line provides the rms values from 100 runs at the end of navigation in position (in meters) and in velocity (in meters/second), respectively. The last line of the table provides the percentage of navigation loss in each Monte Carlo simulation based on the following criterion. A navigation loss is declared if the final normalized estimation error is out of the 99.9% bounds of the Chi-square test for a four-dimensional random variable (the state vector being of dimension 4). In view of Table I, performance of JPDANF appears to be more sensitive to the number of true detections in the FOV than to the number of false detections since many of the false detections in the FOV are not validated by the gating and therefore are not processed at all. Table II provides the results of the autonomous navigation algorithm using the perfect data association under the same previous conditions.

V. CONCLUSIONS

A new autonomous navigation scheme based on the JPDA approach that processes landmark detections in the FOV of an on-board sensor has been developed. These detections—some true, some false—are associated to a set of stored landmarks and used to

TABLE I
Monte Carlo Results (100 Runs) With JPDANF

x_l	0.1				0.2				0.4				0.8			
λ_{fa}	1	2	4	6	1	2	4	6	1	2	4	6	1	2	4	6
n_{ld}	0.42	0.43	0.44	0.46	0.87	0.90	0.89	0.87	1.78	1.74	1.77	1.74	3.48	3.49	3.45	3.51
n_{fa}	4.73	9.45	18.8	28.2	4.70	9.42	18.8	28.2	4.71	9.44	18.8	28.3	4.72	9.43	18.8	28.2
x rms (m)	21.3	16.8	15.9	18.2	8.30	8.10	8.23	8.49	4.23	4.31	4.51	4.29	2.78	2.71	2.86	2.75
y rms (m)	20.9	17.5	15.5	19.6	8.70	7.75	8.17	8.17	4.22	4.38	4.35	4.38	2.79	2.86	2.80	2.69
vx rms (m/s)	0.91	0.87	0.91	0.90	0.72	0.70	0.70	0.72	0.58	0.58	0.60	0.59	0.51	0.51	0.51	0.51
vy rms (m/s)	0.90	0.89	0.91	0.92	0.72	0.69	0.72	0.71	0.58	0.59	0.59	0.59	0.51	0.51	0.51	0.50
% lost tracks	0	0	0	3	1	0	0	0	2	0	0	0	1	0	0	0

Note: Roman symbols here correspond to italic symbols in text.

TABLE II
Monte Carlo Results (100 Runs) With Perfect Data Association

x_l	0.1		0.2		0.4		0.8	
n_{ld}	0.45		0.87		1.78		3.54	
x rms (m)	16.02		7.42		4.31		2.70	
y rms (m)	16.02		7.45		4.31		2.71	
vx rms (m/s)	0.88		0.70		0.58		0.50	
vy rms (m/s)	0.88		0.70		0.58		0.50	
% lost tracks	0		0		0		0	

Note: Roman symbols here correspond to italic symbols in text.

update the state of the vehicle. The results obtained from Monte Carlo simulations prove the ability of this navigation filter to perform in very high false alarm environments. In the different environmental conditions tested in our simulations, the performance of the JPDANF is very close to that of the filter based on perfect data association (which of course can never be implemented in real applications). The very efficient cluster decomposition algorithm presented here for the purpose of the navigation problem, can also be used in many multitarget tracking applications.

APPENDIX A. SELECTION OF REFERENCED LANDMARKS BY THE EXTENDED FIELD OF VIEW APPROACH

Assuming the state $\mathbf{x}(k)$ of the vehicle normally distributed with mean $\hat{\mathbf{x}}(k | k - 1)$ and covariance

$P(k | k - 1)$, the probability to detect at time k any referenced landmark L of the map in the FOV of the vehicle is theoretically given by the general expression

$$P_d(k) = \frac{1}{P_g} \int_v N(\mathbf{x}(k); \hat{\mathbf{x}}(k | k - 1), P(k | k - 1)) P_d(L, \mathbf{x}(k)) d\mathbf{x}(k) \quad (71)$$

with

$$P_d(L, \mathbf{x}(k)) = \begin{cases} G(L, \mathbf{x}(k)) & \text{if } L \in \text{FOV}(\mathbf{x}(k)) \\ 0 & \text{otherwise} \end{cases} \quad (72)$$

where $G(L, \mathbf{x}(k))$ is a function depending on the sensor which characterizes the probability to detect a landmark L lying at the distance $\|L - \mathbf{x}\|$ of the vehicle (here the state $\mathbf{x}(k)$ is restricted to its position components). $\text{FOV}(\mathbf{x}(k))$ is the shape of

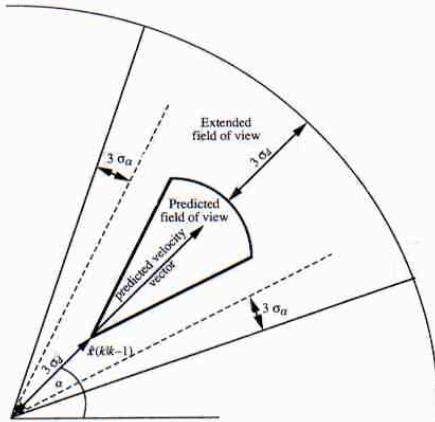


Fig. 2. Plotting the EFOV.

the FOV of the vehicle having the state $\mathbf{x}(k)$ and the integration volume V is defined as the region in the state space where the state will be found with some high probability P_g [2]. Since in view of (71), the computation of the multidimensional integral $\mathbf{P}_d(k)$ is impossible to evaluate without sophisticated numerical

integration methods, the selection of all the referenced landmarks which have some high probability to be detected in the FOV cannot be achieved. Another way based on an heuristic approach consists of selecting only referenced landmarks which lie in an EFOV computed from the predicted state $\hat{x}(k | k-1)$ and its covariance matrix $P(k | k-1)$ as depicted in the Fig. 2. The predicted heading angle α and the predicted velocity module v for time k are, respectively, given by

$$\alpha = \arctan \left[\frac{\hat{y}(k | k-1)}{\hat{x}(k | k-1)} \right] \quad (73)$$

$$\nu = \sqrt{\hat{x}(k | k-1)^2 + \hat{y}(k | k-1)^2} \quad (74)$$

and the values of σ_d and σ_α are given by

$$\sigma_d = \sqrt{\sigma_x^2 + \sigma_y^2} \quad (75)$$

$$\sigma_\alpha = \frac{\sigma_{\nu_y}}{\nu} \quad \text{if } \alpha = 0 \text{ or } \pm \pi r d \text{ or}$$

$$\sigma_\alpha = \frac{\sigma_{\nu_x}}{\nu} \quad \text{if } \sigma = \pm\pi/2rd \quad (76)$$

TABLE III

Subroutine cluster (omega, nr, nc, omegac, nrc, ncc)		
integer omega, omegac dimension omega(30,50), omegac(30,50) do 20 j=1,50 do 10 i=1,30 omegac(i,j) = omega(i,j) 10 continue 20 continue nr=nr ncc=ncc isum=0 do 40 j=1,ncc id=0 do 30 i=1,nrc id=id+omegac(i,j) 30 continue isum=isum+id 40 continue if(isum.eq.ncc)goto 150 do 140 j=1,ncc 50 idc=0 do 60 i=1,nrc idc=idc+omegac(i,j) 60 continue if (idc.eq.1) goto 140 if(idc.gt.1) then n=1 if(omegac(n,j).ne.1)then n=n+1 goto 70 end if	80 90 100 110 120 130 140 150	k=n+1 if(omegac(k,j).ne.1) then k=k+1 goto 80 end if do 90 m=1,ncc omegac(n,m)=omegac(n,m)+omegac(k,m) if (omegac(n,m).gt.1) omegac(n,m)=1 omegac(k,m)=0 continue nrc=nrc-1 do 110 i=k,nrc do 100 m=1,ncc omegac(i,m)=omegac(i+1,m) continue continue do 130 k=nrc+1,nr do 120 m=1,ncc omegac(k,m)=0 continue continue got to 50 end if continue return end

or otherwise by

$$\sigma_\alpha = \sqrt{\frac{1}{2\nu^2} \left[\sigma_{\nu_x}^2 + \sigma_{\nu_y}^2 - \frac{\sigma_{\nu_x \nu_y}}{\sin(\alpha)\cos(\alpha)} \right]} \quad (77)$$

where the quantities σ_x^2 , σ_y^2 , $\sigma_{\nu_x}^2$, $\sigma_{\nu_y}^2$, and $\sigma_{\nu_x \nu_y}$ are, respectively, given by the position and velocity elements of the covariance matrix $\mathbf{P}(k | k-1)$. After setting up the EFOV, all referenced landmarks of the navigation map included in the EFOV will be used in the JPDANF algorithm. The probability of detection of each validated landmark L entering in the computation of the association probabilities will be approximated by

$$\mathbf{P}_d(k) = \mathbf{G}(\mathbf{L}, \hat{\mathbf{x}}(k | k-1)). \quad (78)$$

APPENDIX B. CLUSTERING DECOMPOSITION ROUTINE

Input Arguments of the Routine: Ω is the initial compressed validation matrix $\Omega(k)$ without the column 0. The number of columns of $\Omega(k)$ is nc . Its maximum value is set to 50 in the routine. The number of rows of $\Omega(k)$ is nr . Its maximum value is set to 30 in the routine.

Output Arguments of the Routine: Ω_{egac} is the clustering decomposition matrix $\Omega_c(k)$. The number of columns of $\Omega_c(k)$ is ncc which is of course equal to nc . The number of rows of $\Omega_c(k)$ which corresponds to the number of independent clusters for the validation configuration is nrc . (See Table III.)

REFERENCES

- [1] Anderson, B. D. O., and Moore, J. B. (1979)
Optimal Filtering.
Englewood Cliffs, NJ: Prentice-Hall, 1979.
- [2] Bar-Shalom, Y., and Fortmann, T. E. (1988)
Tracking and Data Association.
New York: Academic Press, 1988.
- [3] Bar-Shalom, Y. (1989)
Multitarget-Multisensor Tracking.
Short course notes, 1989.
- [4] Bar-Shalom, Y. (Ed.) (1990)
Multitarget-Multisensor Tracking: Advanced Applications, Vol. I.
Dedham, MA: Artech House, 1990.
- [5] Bar-Shalom, Y. (Ed.) (1992)
Multitarget-Multisensor Tracking: Applications and Advances, Vol. 2.
Dedham, MA: Artech House, 1992.
- [6] Chong, C. Y. (1979)
Hierarchical estimation.
MIT/ONR C3 Workshop, Monterey, CA, 1979.
- [7] Cox, I. J., and Wilfong, G. T. (1990)
Autonomous Robots Vehicles.
New York: Springer-Verlag, 1990.
- [8] Dezert, J. (1988)
Poursuite multi-cibles mono-senseur: Analyse des principales approches développées dans le domaine.
Note Technique ONERA 1988-10, ONERA, France, 1988.
- [9] Dezert, J. (1990)
Vers un nouveau concept de navigation autonome d'engin.
Un lien entre la théorie de l'évidence et le filtrage à association probabiliste des données.
Thèse de Doctorat de l'Université de Paris-Sud, Centre d'Orsay, 27 Septembre 1990, France.
- [10] Zhou, B. (1992)
Multitarget tracking in clutter: algorithms for data association and state estimation.
Ph.D. dissertation, Dept. of Electrical and Computer Engineering, Pennsylvania State University, University Park, May 1992.

Jean C. Dezert (M'91) was born in l'Hay les Roses, France, in 1962. He received the electrical engineering degree from the Ecole Française de Radioélectricité, Electronique et Informatique, Paris, France in 1985, the D.E.A. in automatic control and signal processing from University Paris VII, in 1986, and the Ph.D. degree in automatic control and signal processing from the University of Paris XI, Orsay, France in 1990.



Between 1986 and 1990 he was associated with the systems department at ONERA, Châtillon, France, where he worked on radar tracking, identification, and autonomous navigation problems. Between May 1991 and May 1992 he held a postdoctoral research fellowship from the European Space Agency and worked on autonomous navigation at the ESP Laboratory in the Electrical and Systems Engineering Dept., of the University of Connecticut, Storrs. Since October 1992, he has been with the Laboratory of Electronics, Signals and Images (LESI) at the University of Orléans, Orléans, France, where he is also a Lecturer in automatic control.

Yaakov Bar-Shalom (S'63—M'66—SM'80—F'84) was born on May 11, 1941. He received the B.S. and M.S. degrees from the Technion, Israel Institute of Technology, in 1963 and 1967 and the Ph.D. degree from Princeton University, Princeton, NJ, in 1970, all in electrical engineering.

From 1970 to 1976 he was with Systems Control, Inc., Palo Alto, CA. Currently he is Professor of Electrical and Systems Engineering at the University of Connecticut. His research interests are in estimation theory and stochastic adaptive control. He has been elected Fellow of IEEE for "contributions to the theory of stochastic systems and of multitarget tracking." He has been consultant to numerous companies, and originated a series of Multitarget-Multisensor Tracking short courses offered via UCLA Extension, University of Maryland, at Government Laboratories, private companies and overseas. He has also developed the interactive software packages MULTIDAT for automatic track formation and tracking of maneuvering or splitting targets in clutter, PASSDAT for data association from multiple passive sensors, BEARDAT for target localization from bearing and frequency measurements in clutter and IMDAT for target image tracking using segmentation.

Dr. Bar-Shalom coauthored the monograph *Tracking and Data Association* (Academic Press, 1988), the graduate text *Estimation and Tracking: Principles, Techniques and Software* (Artech House, 1993) and edited the books *Multitarget-Multisensor Tracking: Applications and Advances* (Artech House, Vol. I 1990; Vol. II 1992). During 1976 and 1977 he served as Associate Editor of the IEEE Transactions on Automatic Control and from 1978 to 1981 as Associate Editor of Automatica. He was Program Chairman of the 1982 American Control Conference, General Chairman of the 1985 ACC, and Co-Chairman of the 1989 IEEE International Conference on Control and Applications. During 1983–1987 he served as Chairman of the Conference Activities Board of the IEEE Control Systems Society and during 1987–1989 was a member of the Board of Governors of the IEEE CSS. In 1987 he received the IEEE CSS Distinguished Member Award.

