Track Fusion in Multisensor-Multitarget Tracking

TRACK FUSION IN MULTISENSOR-MULTITARGET TRACKING

BY

DANIEL G. DANU

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2011) McMaster University
(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE: Track Fusion in Multisensor-Multitarget Tracking

AUTHOR: Daniel G. Danu

SUPERVISOR: Professor T. Kirubarajan

NUMBER OF PAGES: xix, 191

Abstract

Data fusion is the methodology of efficiently combining the relevant information from different sources. The goal is to achieve estimates and inferences with better confidence than those achievable by relying on a single source. Initial data fusion applications were predominantly in defense: target tracking, threat assessment and land mine detection. Nowadays, data fusion is applied to robotics (e.g., environment identification for navigation), medicine (e.g., medical diagnosis), geoscience (e.g., data integration from different sources) and industrial engineering (e.g., fault detection).

This thesis focuses on data fusion for distributed multisensor tracking systems. In these systems, each sensor can provide the information as measurements or local estimates, i.e., *tracks*. The purpose of this thesis is to advance the research in the fusion of local estimates for multisensor multitarget tracking systems, namely, *track fusion*. This study also proposes new methods for track-to-track association, which is an implicit subproblem of track fusion.

The first contribution is for the case where local sensors perform tracking using particle filters (Monte Carlo based methods). A method of associating tracks estimated through labeled particle clouds is developed and demonstrated with subsequent fusion. The cloud-to-cloud association cost is devised together with computation methods for the general and specialized cases. The cost introduced is proved to converge (with increasing clouds cardinality) toward the corresponding distance between the underlying distributions. In order to simulate the method introduced, a particle filter labeled at particle level was developed, based on the Probability Hypothesis Density (PHD) particle filter.

The second contribution is for the case where local sensors produce tracks using Kalman filter-type estimators, in the form of track state estimate and track state covariance matrix. For this case the association and fusion is improved in both terms of accuracy and identity, by introducing at each fusion time the prior information (both estimate and identity) from the previous fusion time.

The third contribution is for the case where local sensors produce track estimates under the form of MHT, therefore where each local sensor produces several hypotheses of estimates. A method to use the information from other sensors in propagating each sensor's internal hypotheses over time is developed.

A practical fusion method for real world local tracking sensors, i.e., asynchronous and with incomplete information available, is also developed in this thesis.

Acknowledgements

First, I would like to thank my supervisor, Dr. T. Kirubarajan for his support, guidance and enthusiasm throughout my time of study. His involvement at the top of the target tracking international community has introduced me to an environment of research excellence. It has provided me with an opportunity to meet inspiring world-class experts in the domain, including Dr. Yaakov Bar-Shalom. I am also grateful to Dr. T. Kirubarajan for my acceptance to the graduate program as a part-time student.

I am thankful to Dr. Abhijit Sinha for the numerous discussions on target tracking and his collaboration on my first papers, as well as Dr. Huimin Chen from University of New Orleans. I thank the other thesis committee members, Dr. Sorina Dumitrescu, Dr. Aleksandar Jeremic for their time and expert technical counsel.

Many people have helped me at different stages of this research, knowingly or unknowingly, directly or indirectly, not least the other members of the Estimation, Tracking and Fusion Laboratory at McMaster, researchers at DRDC Ottawa for providing real data for simulations, Dr. Warren Schappert and Brian Markle as inspiring scientists at my workplace, Thomas Lang from General Dynamics Canada, Cheryl Gies, for her continuous administrative support, Christina Dalima for the careful review of my articles and also the anonymous reviewers of my papers.

My thanks go back in time to my first professor of mathematics, Elvira Serban, who lightened up for me this wonderful world, as well as to Dr. Titu Andreescu, currently at the University of Dallas, who proved me the magic within, during several math camps and private time.

I express my gratitude to my parents for their love, support and continuous care for my education from early years. I thank the rest of my family, particularly my parents-in-law, for their love and encouragement as well.

I am especially grateful to my wife Michelle for all the love, support and sacrifices made for my doctoral pursuits. I also thank my three-years old daughter DeAnna for her coping with my time spent at home working on this thesis. Although she always gives me difficult time when asking the purpose of this research.

Notation and abbreviations

AIS - Automatic Identification System

C2C - Cloud-to-Cloud (referring to association or fusion)

GPS - Global Positioning System

FISST - Finite Set Statistics

IMM - Interacting Multiple Model

KF - Kalman filter

LMMSE - Linear Minimum Mean Square Error

MAP - Maximum a posteriori

MHT - Multiple Hypothesis Tracking

MMSE - Minimum Mean Square Error

MSE - Mean Square Error

OTH - Over-the-Horizon

PHD - Probability Hypothesis Density

RFS - Random Finite Set

T2T - Track-to-Track (referring to association or fusion)

pdf - Probability Density Function

$E[\cdot]$ - expectation

$\mathbf{F}$ - target state transition matrix

$\mathbf{H}$ - measurement matrix

$\mathbf{G}(k)$ - process noise gain matrix at sample time k

$\mathbf{P}(k)$ - target state estimate error covariance matrix at sample time k

$\mathbf{Q}(k)$ - process noise covariance matrix at sample time k

$\mathbf{R}(k)$ - measurement covariance error matrix at sample time k

Z^k - set of measurements from time 0 up to time k, $\{\mathbf{z}(0), \mathbf{z}(1), \ldots, \mathbf{z}(k)\}$

$p_{\mathbf{x}(k)|Z^k}(\mathbf{x})$ - conditional posterior pdf of target state

$p(\mathbf{x}(k)|Z^k)$ - conditional posterior pdf of target state (simplified notation)

$\mathbf{x}(k)$ - true target state vector at sample time k

$\hat{\mathbf{x}}(k)$ - target state estimate vector (track estimate) at sample time k

$\mathbf{w}(k)$ - measurement error vector at sample time k

$\mathbf{z}(k)$ - measurement vector at sample time k

$\boldsymbol{v}(k)$ - process noise vector at sample time k

$\xi^{(i)}$ - state vector of particle index i used in particle filter

Contents

List of Figures

Chapter 1

Introduction and Problem Statement

This section introduces the subject of this research thesis, namely, the problem of track fusion for distributed multisensor multitarget tracking applications. The *target tracking* problem is presented first with a description of the general track estimator for a single target. The more complex *multitarget tracking* problem is discussed in the following subsection. Specific local track estimator methods applicable to multitarget tracking and used within the framework of this research, namely, Kalman filter-type estimators, particle filters and the Multiple Hypothesis Tracking algorithm (MHT) are presented. The *multisensor multitarget tracking* is introduced next. Data fusion in multisensor multitarget tracking with its intrinsic problem of data association is detailed with emphasis on the distributed sensors tracking case.

1.1 Target Tracking

Given the *imperfect* and *incomplete* measurements of an unknown *target state*, the target tracking problem is to estimate the state of the target and its evolution over time. The estimation is performed recursively at discrete times t_0, t_1, ... t_k, ... Also, target tracking includes the evaluation of the precision of each target state estimate made.

At any given sample time t_k, the *target state* is represented by a vector whose components are *true* parameters of the target. These components may be target position, speed, and acceleration in three or less dimensions. For example, the target state of a ship can be represented as

$$\mathbf{x}(k) = \begin{bmatrix} x(k) & \dot{x}(k) & y(k) & \dot{y}(k) \end{bmatrix}^T, \tag{1.1}$$

where $(x(k), y(k))$ represents the target position in a 2D cartesian coordinate system xOy, and $\dot{x}(k)$, $\dot{y}(k)$ represent the corresponding velocities along the Ox and Oy axes. Equation (1.1) gives the *true* target state, unknown, i.e., to an observer on the shore. The target state evolves in time. i.e., as the ship in our example moves, changes course and speed. The target state is represented by a single point at a given time in the target state space $\mathcal{R}^{n_x}$ where n_x is the target state vector size. For bigger targets that are not small compared to the measurement resolution, i.e. a ship, the state of the center of the ship is tracked. In the above example the state has $n_x = 4$ and the state space is 4-dimensional.

The aim of target tracking is to provide the *track* of the target, $\hat{\mathbf{x}}(k)$, $k = 1, 2, \ldots$, which is the list of estimates of the target state at each sample time.

In the target tracking problem a sensor provides *measurements* of the target parameters at sample times t_k, $k = 0, 1, \ldots$. Each measurement is represented by a scalar (e.g. bearing-only) or vector, often of lesser size than the target state vector. For the example above, a fixed radar sensor on the shore is considered, that provides *measurements* containing positional information only [1]

$$\mathbf{z}(k) = \begin{bmatrix} z_x(k) & z_y(k) \end{bmatrix}^T, \tag{1.2}$$

where ($z_x(k)$ and $z_y(k)$) represent the target position measured by the sensor in the same cartesian coordinate system xOy.

The *imperfection* of measurements mentioned above is twofold. First, the measured state components are affected by errors due to the sensor limited precision, i.e.,

$$\begin{aligned} z_x(k) &= x(k) + w_x(k) \\ z_y(k) &= y(k) + w_y(k) \end{aligned} \tag{1.3}$$

where

$$\mathbf{w}(k) = \begin{bmatrix} w_x(k) & w_y(k) \end{bmatrix}^T \tag{1.4}$$

represents the *sensor error* vector with $w_x(k)$ and $w_y(k)$ the error components on the two cartesian directions Ox and Oy. These errors are time-dependent as they depend on the sensor-target geometry. Second, the sensor may have false detections and the whole measurement at a sample time may be a false one (i.e., false detection). The first type of error is captured by the sensor error probability density function (*pdf*), usually known for a given sensor and dependent on the target-sensor geometry. For

[1] A radar sensor usually measures range (r) and bearing (θ), however for the brevity of presentation the polar transform into Cartesian coordinates with the more intricate errors transform is skipped here.

a Gaussian and unbiased (zero-mean) error distribution this error is characterized solely by the *sensor error covariance*, denoted as $\mathbf{R}(k)$

$$\mathbf{R}(k) = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix}, \tag{1.5}$$

where σ_x^2, σ_y^2 and σ_{xy}^2 represent the variance of the errors on the Ox, Oy axes and respectively the covariance between them, here considered constant in time and space. The false measurements are modeled through the sensor *probability of false alarms* P_{FA} (Scharf, 1991).

The *incompleteness* of measurements mentioned above is also twofold. One aspect refers to the fact that the measurement vector most of the times has less state components than the target state components, as seen in (1.2). The other aspect is that at given sample times a target can go undetected (i.e. there are missed measurements or missed detections). The missed detections are modeled through the sensor *probability of detection* P_D (Scharf, 1991).

Measurement Equation

The relationship between the target state, sensor measurement and the sensor error is modeled in target tracking through the *measurement equation*, which for the example considered is

$$\mathbf{z}(k) = \mathbf{H}(k) \cdot \mathbf{x}(k) + \mathbf{w}(k), \tag{1.6}$$

where $\mathbf{H}(k)$ is the *measurement matrix* and $\mathbf{w}(k)$ is the measurement noise or measurement error in (1.4). For the sample target state and measurement errors given in

(1.1) and (1.4) the measurement matrix is constant over time and and has the form

$$\mathbf{H}(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{1.7}$$

Equations (1.6) and (1.7) describe a linear stochastic dependence between the sensor measurement and target space, however in the general case the *measurement equation* is nonlinear

$$\mathbf{z}(k) = \mathbf{h}(k,\, \mathbf{x}(k),\, \mathbf{w}(k)), \tag{1.8}$$

where $\mathbf{h}(\cdot)$ is a vector-valued, time-varying and nonlinear function and $\mathbf{w}(\cdot)$ gives the sensor measurement error, possibly dependent on the target state (Bar-Shalom *et al.*, 2001).

Dynamic Equation

The target state dynamics are modeled through the *dynamic equation* known also as *process plant* equation (Bar-Shalom *et al.*, 2001) or *target state transition* equation. This equation is based on the important assumption that the target state evolution in time is a *Markov process* (Bar-Shalom *et al.*, 2001). In such a process, the probability density function (*pdf*) of the future evolution of the state conditioned on the whole state history up to the time k_1 is equivalent to the *pdf* of the future evolution of the state conditioned only on the state at time k_1:

$$p\left(\mathbf{x}(k) \mid \mathbf{x}(\kappa), \kappa \leq k_1\right) = p\left(\mathbf{x}(k) \mid \mathbf{x}(k_1)\right), \quad \forall k > k_1. \tag{1.9}$$

Equation (1.9) can be summarized also as "The future is independent of the past if the present is known" (Bar-Shalom *et al.*, 2001). Based on this assumption, the target state transition from time k to time $k+1$ can be modeled through a *linear* stochastic process

$$\mathbf{x}(k+1) = \mathbf{F}(k) \cdot \mathbf{x}(k) + \boldsymbol{v}(k), \tag{1.10}$$

where $\mathbf{x}(k)$ is the true target state at k, $\mathbf{F}(k)$ is the *state transition matrix* (Bar-Shalom *et al.*, 2001) and $\boldsymbol{v}(k)$ is the target state model *process noise*. The process noise accounts for the uncertainties in the target motion that are not modeled through the $\mathbf{F}(k) \cdot \mathbf{x}(k)$ term. For the example taken with the ship target state (1.1), the state transition matrix is

$$\mathbf{F}(k) = \begin{bmatrix} 1 & \Delta k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta k \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{1.11}$$

where Δk is the time interval between sample times k and $k+1$. In this case any deviation of the ship motion from a constant velocity over the interval between the sample times k and $k+1$ is modeled through the target process noise $\boldsymbol{v}(k)$ and its chosen covariance matrix $\mathbf{Q}(k)$.

The process noise selection is based on the prior knowledge about the target motion (e.g., known motion type, maximum acceleration). In order to capture different possible motion types of a target (i.e., constant speed, coordinated turn, random accelerations), several target state transition models can be considered within a single estimator (i.e., with low process noise, with high process noise, with different matrix

$\mathbf{F}(k)$). The process noise is usually modeled as Discrete Continuous White Noise Acceleration (DCWNA) or Discrete White Noise Acceleration (DWNA) (Bar-Shalom *et al.*, 2001). The process noise (i.e., DCWNA or DWNA) is selected as uncorrelated (white) with zero-mean in order to preserve the Markov property (1.9) of the modeled target state. For the example with the ship, the Markov property states that given the state of the ship at a sample time k (i.e., position and velocity), the future evolution of the ship states (trajectory) has to be independent on the ship's states before time k. As the state captures the ship dynamics up to velocity, in order to have the state at time $k+1$ independent vs. the state at $k-1$ and previous ones, the possible ship accelerations at $k-1$ and k have to be independent. This is satisfied assumming a DWNA process noise model, which implies the sequence of accelerations between sample times to be white, and that over each interval between sample times the acceleration to be constant. More details on the process noise selection implications can be found in (Bar-Shalom *et al.*, 2001).

The more general *nonlinear* stochastic process model of the target state dynamic equation is

$$\mathbf{x}(k+1) = \mathbf{f}\left(k+1,\, \mathbf{x}(k),\, \boldsymbol{v}(k)\right), \tag{1.12}$$

where $\mathbf{f}(\cdot)$ is a time-varying nonlinear vector-valued function (Bar-Shalom *et al.*, 2001).

Observation. The target state in the example with the ship above takes a fixed value at each time k. However both dynamic equations (1.10), (1.12) model the target state as random processes through the added process noise, which is a random sequence (random process). Therefore in target tracking the unknown target state is modeled as a random process.

Assumptions

- In order to preserve the Markov property (1.9) for the target state, the process noise $\boldsymbol{v}(k)$ in both (1.10) and (1.12) is assumed zero-mean and white (independent samples in time).
- The measurement noise $\mathbf{w}(k)$ is assumed also to be zero-mean and white.
- The process noise and measurement noise are assumed to be mutually independent.

Optimal Bayesian Track Estimate

The end result of a single target tracking is the *track* of the target, which is the list of *optimal* estimates of the target state over time. This represents the trajectory of the target state estimated from the measurements. Each track entry $\hat{\mathbf{x}}(k)$ (or track estimate) at a given time k is conditioned on $Z^k = \{\mathbf{z}(0), \mathbf{z}(1), \ldots \mathbf{z}(k)\}$, the set of all past measurements up to time k. The optimality criterion used in this research is the one that minimizes the Mean Square Error (MSE) between the track estimate $\hat{\mathbf{x}}(k)$ and the true target state $\mathbf{x}(k)$. The estimate that minimizes the MSE is known in literature as the Minimum Mean Square Error (MMSE) one (Bar-Shalom *et al.*, 2001)

$$\hat{\mathbf{x}}^{\mathrm{MMSE}}(k) = \arg\min_{\hat{\mathbf{x}}} E\left[(\hat{\mathbf{x}}(k) - \mathbf{x}(k))^2 \mid Z^k\right]. \tag{1.13}$$

Equation (1.13) translates into (Bar-Shalom *et al.*, 2001)

$$\hat{\mathbf{x}}^{\mathrm{MMSE}}(k) = E\left[\mathbf{x}(k) \mid Z^k\right] \triangleq \int_{\mathcal{R}^{n_x}} \mathbf{x} \cdot p_{\mathbf{x}(k) \mid Z^k}(\mathbf{x})\, d\mathbf{x}, \tag{1.14}$$

where $p_{\mathbf{x}(k)\,|\,Z^k}(\mathbf{x})$ is the *conditional posterior pdf* of the target state taking values over the entire target state space $\mathbf{x} \in \mathcal{R}^{n_x}$, where n_x is the target state vector size. The index of the function is dropped for brevity and the notation $p(\mathbf{x}(k) \mid Z^k)$ is used further.

The *posterior conditional pdf* of the target state in (1.14) can be expressed using Bayes formula of conditional probabilities (Bayes, 1763; Bar-Shalom *et al.*, 2001; Papoulis and Pillai, 2001)

$$p(\mathbf{x}(k) \mid Z^k) = \frac{p(\mathbf{z}(k) \mid \mathbf{x}(k), Z^{k-1}) \cdot p(\mathbf{x}(k) \mid Z^{k-1})}{p(\mathbf{z}(k) \mid Z^{k-1})}. \tag{1.15}$$

The measurements error (noise) sequence is assumed white, and therefore $\mathbf{z}(k)$ conditioned on $\mathbf{x}(k)$ is independent of $\mathbf{z}(j), j \leq k-1$. Also as the process noise $\boldsymbol{v}(j)$, $j \leq k$ and measurement noise are independent sequences, the first factor in (1.15) becomes

$$p(\mathbf{z}(k) \mid \mathbf{x}(k), Z^{k-1}) = p(\mathbf{z}(k) \mid \mathbf{x}(k)), \tag{1.16}$$

the *measurement likelihood.* The second factor in the nominator can be written using the Chapman-Kolmogorov equation (Bar-Shalom *et al.*, 2001; Papoulis and Pillai, 2001)

$$p(\mathbf{x}(k) \mid Z^{k-1}) = \int p(\mathbf{x}(k) \mid \mathbf{x}(k-1), Z^{k-1}) \cdot p(\mathbf{x}(k-1) \mid \mathbf{Z}^{k-1}) d\mathbf{x}(k-1). \tag{1.17}$$

Using a rationale similar to the one used for (1.16)

$$p(\mathbf{x}(k) \mid Z^{k-1}) = \int p(\mathbf{x}(k) \mid \mathbf{x}(k-1)) \cdot p(\mathbf{x}(k-1) \mid \mathbf{Z}^{k-1}) d\mathbf{x}(k-1). \tag{1.18}$$

The first factor in the integral above is the *state transition pdf* and the second one is the target state conditional pdf at time $k-1$. Using (1.16) and (1.18), equation (1.15) can be written as

$$p(\mathbf{x}(k) \mid Z^k) = \frac{p(\mathbf{z}(k) \mid \mathbf{x}(k)) \cdot \int p(\mathbf{x}(k) \mid \mathbf{x}(k-1)) \cdot p(\mathbf{x}(k-1) \mid Z^{k-1}) d\mathbf{x}(k-1)}{p(\mathbf{z}(k) \mid Z^{k-1})}. \tag{1.19}$$

The denominator of (1.19) is a normalizing constant (constant over all times k) and using Chapman-Kolmogorov integral may be written as

$$\begin{aligned}
&p(\mathbf{z}(k) \mid Z^{k-1}) \\
&= \int p(\mathbf{z}(k) \mid \mathbf{x}(k)) \cdot p(\mathbf{x}(k) \mid Z^{k-1}) d\mathbf{x}(k) \\
&= \int p(\mathbf{z}(k) \mid \mathbf{x}(k)) \cdot \left[\int p(\mathbf{x}(k)) \mid p(\mathbf{x}(k-1)) \cdot p(\mathbf{x}(k-1) \mid Z^{k-1}) d\mathbf{x}(k-1)\right] d\mathbf{x}(k) \\
&= c.
\end{aligned} \tag{1.20}$$

From (1.20) and (1.19) results the *Bayesian recursion of the conditional target state pdf*

$$p(\mathbf{x}(k) \mid Z^k) = \frac{p(\mathbf{z}(k) \mid \mathbf{x}(k)) \cdot \int p(\mathbf{x}(k) \mid \mathbf{x}(k-1)) \cdot p(\mathbf{x}(k-1) \mid Z^{k-1}) d\mathbf{x}(k-1)}{c}. \tag{1.21}$$

All conditional *pdfs* that enter (1.21) are known:

- $p(\mathbf{x}(k) \mid \mathbf{x}(k-1))$ - the state transition *pdf* can be derived from the known $\mathbf{F}(k)$ and $p(\boldsymbol{v}(k))$ in (1.10) or $\mathbf{f}(\cdot)$ and $p(\boldsymbol{v}(k))$ in (1.12)

- $p(\mathbf{z}(k) \mid \mathbf{x}(k))$ - the measurement *likelihood* can be derived from the known $\mathbf{H}(k)$ and $p(\mathbf{w}(k))$ in (1.6) or $\mathbf{h}(\cdot)$ and $p(\mathbf{w}(k))$ in (1.8)

- $p(\mathbf{x}(k-1) \mid Z^{k-1})$ - the *pdf* of the state at $k-1$ is known from previous recursion

The estimate using (1.21) and (1.14) is unusable in practice because of the very high computational demand due to the double integration required on the target state space. Following subsections present three estimators that simplify the above integral.

1.1.1 Kalman Filter Estimator

The Kalman Filter (KF) estimator (Kalman, 1960) gives an analytic solution to the Bayesian recursive estimation in (1.21) under certain assumptions. The conditions on target dynamics and measurements for the KF to provide the *exact solution* for the MMSE track estimate are listed below.

- The initial target state (unknown) is a Gaussian distributed random variable.

- The target state dynamics can be modeled through a linear dynamic equation

  $$\mathbf{x}(k+1) = \mathbf{F}(k) \cdot \mathbf{x}(k) + \boldsymbol{v}(k), \tag{1.22}$$

  and the process noise $\boldsymbol{v}(k)$ is assumed additive, white, Gaussian distributed with zero-mean and known covariance, possibly time-varying $\mathbf{Q}(k)$.

- The measurements can be modeled through a linear measurement equation

  $$\mathbf{z}\left(k\right) = \mathbf{H}(k) \cdot \mathbf{x}\left(k\right) + \mathbf{w}\left(k\right), \tag{1.23}$$

  and the measurement noise $\mathbf{w}(k)$ is assumed additive white, Gaussian distributed with zero-mean and known covariance, possibly time-varying $\mathbf{R}(k)$.

- The measurement noise, process noise and initial target state are assumed mutually independent.

The above assumptions are known as the *Linear-Gaussian* assumptions and the resulting process is a *Gauss-Markov* process (Bar-Shalom *et al.*, 2001).

Some of the conditions above can be relaxed if extra processing is added to the basic KF equations. If the process noises are not Gaussian, however their covariances are available, the KF is the Linear Minimum Mean Square Error (LMMSE) estimator, as it is still the best linear estimator of the MSE estimate (Bar-Shalom *et al.*, 2001).

Kalman Filter Recursion Cycle

A sample cycle of the KF recursive estimation is shown below. The track estimate $\hat{\mathbf{x}}(k-1 \mid k-1)$ based on measurements up to time $k-1$, and its covariance matrix $\mathbf{P}(k-1 \mid k-1)$ are assumed known at time $k-1$. Upon the receipt of the measurement $\mathbf{z}(k)$ the corresponding estimates are computed for time k (Bar-Shalom *et al.*, 2001).

Prediction

Using the dynamic equation (1.22) compute the *predicted* state estimate at time k

$$\hat{\mathbf{x}}(k \mid k-1) = \mathbf{F}(k-1) \cdot \hat{\mathbf{x}}(k-1 \mid k-1) \tag{1.24}$$

and its covariance,

$$\mathbf{P}(k \mid k-1) = \mathbf{F}(k-1) \cdot \mathbf{P}(k-1 \mid k-1) \cdot \mathbf{F}(k-1)^T + \mathbf{Q}(k-1). \tag{1.25}$$

Using the measurement equation (1.23) compute the *predicted* measurement

$$\hat{\mathbf{z}}(k \mid k-1) = \mathbf{H}(k) \cdot \hat{\mathbf{x}}(k \mid k-1) \tag{1.26}$$

and its covariance

$$\mathbf{S}(k) = \mathbf{H}(k) \cdot \mathbf{P}(k \mid k-1) \cdot \mathbf{H}^T(k) + \mathbf{R}(k). \tag{1.27}$$

Update

Compute the filter gain matrix, which is the product of the cross-covariance between predicted state and predicted measurement $\mathbf{P}(k \mid k-1) \cdot \mathbf{H}(k)^T$ and the inverse of the predicted measurement covariance $\mathbf{S}(k)^{-1}$

$$\mathbf{W}(k) = \mathbf{P}(k \mid k-1) \cdot \mathbf{H}(k)^T \cdot \mathbf{S}(k)^{-1}. \tag{1.28}$$

Compute the *updated* state estimate

$$\hat{\mathbf{x}}(k \mid k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{W}(k)\boldsymbol{\nu}(k), \tag{1.29}$$

where

$$\boldsymbol{\nu}(k) = \mathbf{z}(k) - \hat{\mathbf{z}}(k|k-1) \tag{1.30}$$

is known as the *innovation* and its covariance is equal to the one computed for predicted measurements $\mathbf{S}(k)$. Compute the *updated* state estimate covariance matrix

$$\mathbf{P}(k \mid k) = \mathbf{P}(k \mid k-1) - \mathbf{W}(k) \cdot \mathbf{S}(k) \cdot \mathbf{W}(k)^T. \tag{1.31}$$

1.1.2 Interacting Multiple Model Estimator

The Interacting Multiple Model (IMM) is an adaptive estimator used in estimating the state of a target of which dynamic pattern changes over time. For example a target trajectory can be a combination of segments over which the target moves straight (e.g., with nearly constant velocity) and segments over which the target performs maneuvers (e.g., coordinated turns). Each one of these motion types can be modeled using the Linear-Gaussian model (Section 1.1.1), however two different dynamic equation models are needed. The IMM provides the mechanism to use several dynamic equation models, which models can differ in the transition matrix **F**, in the process noise, or both. Internally it estimates the target state using each of the dynamic models. The likelihood of each model is evaluated *interactively* based on the likelihood of the measurement conditioned on the model state estimate and of the transition probabilities between models. The weights of the state estimates produced by models are dynamically combined by considering the models as the states of a *Markov chain* process. If the Linear-Gaussian assumptions are satisfied for each mode, then the final estimate is the result of a *mixture of Gaussian pdfs.* More details on the IMM can be found in (Bar-Shalom *et al.*, 2001).

The KF can be used as internal estimator of each mode in the IMM, however other state estimators (e.g., the particle filter presented next) can be used as internal model estimators as well.

1.1.3 Particle Filter Estimator

For targets having nonlinear trajectories or being measured by nonlinear sensors (e.g. providing range and bearing or only bearing measurements) the KF assumptions are not met. The particle filter, introduced in (Gordon *et al.*, 1993), estimates the conditional pdf of the state by implementing directly the Bayesian recursion in (1.21) through Monte Carlo methods. The conditional *pdf* of the target state vector, $p(\mathbf{x} \mid Z^k)$, is approximated through a set of random *particles* $\{\xi_k^i \mid i = 1, \ldots, N\}$. Particles are vectors with the same dimension and components (e.g., position, velocity, etc) as the target state to be estimated. The set of random particles constitute a *random measure* of the *pdf.*

As in KF, the estimation cycle includes the two stages *prediction* and *update.* In the prediction step each particle in the set is propagated (moved) in the target state space using the dynamic equation (1.10) or (1.12). In the update step particles receive weights proportional to the received measurement likelihood function. The measurement likelihood function is evaluated for each of the particle using the measurement model (1.6) or (1.8). At the end of each recursion cycle a *resampling* step is added to adjust the number of particles and equalize their weights. A sample particle filter cycle is presented below, based on the bootstrap filter introduced in the seminal paper (Gordon *et al.*, 1993).

Prediction

Each particle $\xi_{k-1}^i, i = 1, \ldots, N$ available at sample time $k-1$ is predicted using the dynamic equation (1.12)

$$\tilde{\xi}_k^i = \mathrm{f}_k(\xi_{k-1}^i, \tilde{v}_{k-1}^i) \tag{1.32}$$

where $\tilde{v}_{k-1}^i$ is a sample drawn from the process noise *pdf*, $p(\boldsymbol{v}_{k-1})$.

Update

Upon the measurement $\mathbf{z}_k$ receipt at time k, the likelihood of each particle is used to compute the particle normalized *weight*

$$\tilde{w}_k^i = \frac{p(\mathbf{z}_k \mid \tilde{\xi}_k^i)}{\sum_{j=1}^{N} p(\mathbf{z}_k \mid \tilde{\xi}_k^j)} \tag{1.33}$$

Resample

N particles of equal weight $\{\xi_k^j \mid j = 1, \ldots, N\}$ are sampled from the weighted particles $\{(\tilde{\xi}_k^j, \tilde{w}_k^i) \mid j = 1, \ldots, N\}$.

The basic particle filter algorithm above is known as the *bootstrap filter*. There is a large variety of PFs known in target tracking, with various sampling (e.g. sampling from measurements distribution as well) and resampling schemes (Arulampalam *et al.*, 2002), (Doucet *et al.*, 2001).

1.2 Multitarget Tracking

In a multitarget tracking problem the sensor produces measurements of several targets present in the sensor field of view. Measurements from the targets at each sample time k [2] usually do not have identity, in the sense that it is unknown which target generated a given measurement. Also there are received measurements that do not pertain to any target, i.e., spurious, false alarms, created by background noise. This introduces the problem of *data association*, i.e., measurements to targets association,

[2] Measurements are considered synchronized here, however for a scanning sensor they can be scattered over the sensor scanning period, i.e., between k and $k+1$ (Bar-Shalom and Li, 1995).

that includes *gating*, *association*, *track initiation*, *confirmation* and *deletion* (Bar-Shalom and Li, 1995). There are two categories of solutions: non-Bayesian as the *2D Assignment* (Section 1.2.1) or Bayesian ones as the *Multiple Hypothesis Tracking* (Sections 1.2.3) or *Probability Hypothesis Density* (PHD) particle filter (Section 1.2.2).

Gating is the process of eliminating measurements from possible association to a target estimate (track). A sample gating is to allow a measurement to enter the association with a track if the measurement is within a given region around the predicted measurement (1.26) for that track. Within the Linear-Gaussian framework this could be done by verifying the measurement falls within the "confidence ellipsoid" of the predicted measurement. This is equivalent to verify the estimated innovation (1.30) is consistent with its estimated covariance **S** (1.27).

$$(\mathbf{z}(k) - \hat{\mathbf{z}}(k \mid k-1)) \cdot \mathbf{S}(k)^{-1} (\mathbf{z}(k) - \hat{\mathbf{z}}(k \mid k-1)) < g^2. \tag{1.34}$$

Here g is the threshold selected for the ellipsoid.

1.2.1 Data Association in Multitarget Tracking

Data association in multitarget single sensor tracking is the process of associating the newly received measurements to the existing (confirmed) tracks (target state estimates). We consider that at sample time $k-1$ there is a set of N_{k-1} confirmed tracks $\hat{\mathbf{x}}(k-1)^i, i = 1, \ldots, N_{k-1}$. At time k a set of M_k measurements is received, $\mathbf{z}(k)^j, j = 1, \ldots, M_k$. The data association problem is to find the best association between the measurements and tracks. This can be solved through a *hard* association or *2D Assignment* (Pattipati *et al.*, 2000) performed at every sample time k. The hard association means that at any given sample time k a measurement (or none) is associated to every track. The assignment is defined by the binary variable defined as

$$\chi_{ij} = \begin{cases} 1 & \text{if measurement } \mathbf{z}^j(k) \text{ is assigned to track } \hat{\mathbf{x}}^i(k) \\ 0 & \text{otherwise} \end{cases} \tag{1.35}$$

where

- χ_{i0} means that no measurement (or *dummy* measurement) was assigned to track i – implied *missed detection*
- χ_{0j} means that no target was assigned to measurement j – implied *false alarm* or *new target detection*

The optimal assignment for measurements to track association is solved through the minimization (Pattipati *et al.*, 2000)

$$\min_{\chi} \sum_{i=0}^{N_{k-1}} \sum_{j=0}^{M_k} \chi_{ij} \cdot c\left(\hat{\mathbf{x}}^i(k), \mathbf{z}^j(k)\right) \tag{1.36}$$

subject to

$$\sum_{j=0}^{M_k} \chi_{ij} = 1, \quad i = 1, \ldots, N_{k-1} \tag{1.37}$$

$$\sum_{i=0}^{N_{k-1}} \chi_{ij} = 1, \quad j = 1, \ldots, M_k. \tag{1.38}$$

The cost of each pairing is defined using the likelihood of the measurement conditioned on the track estimate, unless the pairing is eliminated by gating

$$\chi_{ij} = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ \infty & \text{for pairings eliminated by gating} \\ \log\left(p(\mathbf{z}^j(k) \mid \hat{\mathbf{x}}^i(k))\right) & \text{otherwise.} \end{cases} \tag{1.39}$$

1.2.2 Probability Hypothesis Density (PHD) Particle Filter

The Probability Hypothesis Density (PHD) filter is a Bayesian multitarget tracking estimator introduced in (Mahler, 2003). The PHD is based on the Random Finite Set (RFS) theory, point processes (Daley and Vere-Jones, 1988), and Finite Set STatistics (FISST) (Mahler, 2007). It estimates all the targets states at once, as a multitarget state (*metatarget*), however projected on the *single-target space.* It considers all the true targets states in the surveillance area at a given sample time as a single entity, the *finite* set of multitarget state vectors $X(k) = \{\mathbf{x}_1(k), \ldots, \mathbf{x}_N(k)\}$, where N is the number of targets. The multitarget space is defined as the combination of all subsets of the single-target space (Mahler, 2003, p.1157). The set of measurements received at sample time k is considered also a *finite* set $Z(k) = \{\mathbf{z}_1(k), \ldots, \mathbf{z}_M(k)\}$, where M is the number of measurements. The estimate of the multitarget state (metatarget) is modeled as a *random finite set* Ξ defined as a subset of the multitarget state space. The PHD filter does not estimate directly the random finite set Ξ, rather its *probability hypothesis density*, which is its *first* order *multitarget moment density*, $D_\Xi(\{\mathbf{x}\})$, where $\mathbf{x}$ is a single-target state vector. The nth order multitarget moment density of the random finite set Ξ is $D_\Xi(X)$, where $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and is the probability density that n of the targets in Ξ have states $x_1, \ldots, \mathbf{x}_N$. Therefore the PHD is the function whose integral over a region of the single target state gives the expected number of targets within the region. A sample PHD for three targets with in the 2-dimensional state space is shown in Fig.1.2.2. It can be seen that the target state estimates (tracks) cannot be identified from one another, as all targets are estimated in the same single-target space through a single function (i.e., the PHD function). Even though track estimates at a given time can be distinguished from one

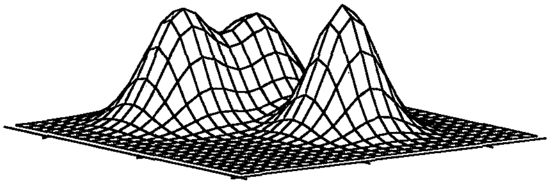


Figure 1.1: Sample PHD for three targets, estimated on the 2-dimensional single target space.

another through thresholding around the PHD function peaks, there is no intrinsic mechanism to propagate these identities over time. Two particle labeling schemes with the end result of identifying tracks was proposed in Appendix A. The purpose of particle labeling of the PHD filter is to allow the clouds estimated on different sensors to be fused, which is the research subject of Section 2. The propagation (prediction), update and resampling equations of the PHD filter are presented in Section A.2.1.

1.2.3 Multiple Hypothesis Tracking (MHT) Estimator

The Multiple Hypothesis Tracking (MHT) is a Bayesian tracking approach introduced in (Reid, 1979). MHT theoretically evaluates the probability of every combination (i.e., hypothesis) of measurements received over time as being generated by a target.

The *valid* hypotheses derived from sequences of measurements are evaluated and propagated over time, each of them generating a set of new hypotheses at every sample time k. The number of hypotheses grows continuously and therefore the implementation of an optimal MHT becomes unaffordable. A judicious selection of hypotheses becomes essential for an efficient and practical implementation. The *pruning* i.e., selection of hypotheses in a multisensor scenario is one of the contributions of this thesis and a more detailed presentation of the MHT is given in the corresponding Section 4.

1.3 Multisensor-Multitarget Tracking

With the ever increasing number of tracking systems and tracking system types (e.g., satellite radar, airborne radar, over-the-horizon radar, infrared, optical), the surveillance areas of various systems overlap. While sensor accuracy cannot be increased beyond a limit, the performance of target tracking algorithms can be greatly enhanced by employing multiple sensors with overlapping coverage regions. *Data fusion* is the combination of information from multiple sensors. The local sensor information entering the fusion process can be raw local measurements, locally associated measurements or local estimates (Danu *et al.*, 2007c).

1.3.1 Fusion of Measurements

In this type of fusion the local sensor transmits local measurements to the fusion center and the estimation is performed centralized. This method leads to theoretical optimal performance, but at the cost of increased communication, a system with a single point of failure that is not always feasible in real world due to legacy constraints.

For example, tracking systems might not have measurements readily available to be communicated outside the system, only final estimates.

1.3.2 Fusion of Local Estimates

While the subject of the current thesis is the track-to-track fusion, there are other type of estimates at which this type of fusion can be carried out:

- tracklet fusion
- fusion of associated measurements.

A performance evaluation of the distributed fusion algorithms above was performed by the author in (Danu *et al.*, 2007b).

1.4 Track Fusion Problem

The estimation fusion problem can be categorized as a class of problems in which estimates of a continuous parameter/state vector obtained by different sources are to be combined to obtain an overall estimate which, in general, has better accuracy.

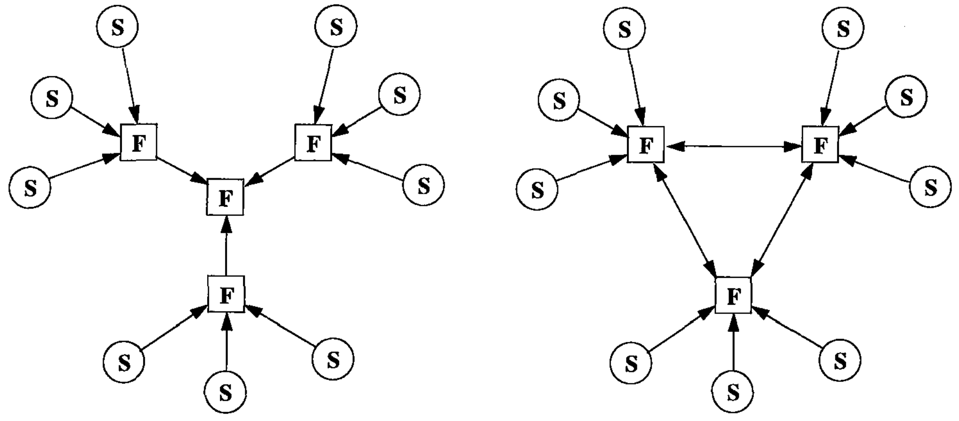
Some terms of the estimation fusion problem are defined as follows. The term *raw measurement* refers to the measurement from any sensor at the end of its signal processing chain. The term *processed measurement* refers to the data after some transform of the raw measurement to be used for estimation. One of the purposes in processing the raw measurements is to compress the data and save communication bandwidth (e.g., measurements may be locally associated first in order to eliminate the local false alarms ones). The term *local estimate* refers to any estimate that uses measurements from the local platform (i.e., the platform where the sensor is located) only. A local estimate may include data from a single sensor or multiple sensors, but all inputs must be from the local platform or the data processing unit. The fusion systems use three basic approaches of communication between a local platform and the fusion center, namely:

- sending raw measurements
- sending processed measurements, e.g., quantized measurements to satisfy the bandwidth constraint
- sending local posteriors, e.g., local estimates/covariances.

This research thesis focuses on the third approach since it is commonly used in the existing distributed tracking systems.

1.4.1 Estimation Fusion Architectures

Architectures for estimation fusion can be divided into two basic categories, namely: a hierarchial fusion architecture and a fully distributed fusion architecture (Bar-Shalom, 2006). Fig.1.2 shows examples of the two types of architectures where nodes marked as S denote sensors and nodes marked as F denote fusion centers. In the case of hierarchical architecture, local estimates obtained in the local fusion center are transmitted to the corresponding higher level fusion centers where these estimates are fused. On the other hand, in a fully distributed architecture, fusion centers do not have superior/subordinate relationship. Each local fusion center broadcasts its estimates to all other fusion centers, which, in turn, update their estimates by incorporating the new information. In the case of hierarchical architecture, the failure of a higher order fusion center makes all subordinate fusion centers unusable for fusion. A fully



(a) Hierarchical fusion architecture. (b) Distributed fusion architecture.

Figure 1.2: Sample centralized (hierarchical) and distributed fusion architectures.

distributed architecture does not have this limitation and hence it produces robust systems. However, it requires higher overall computation and it is unsuitable for a system that requires single global picture. The hierarchical fusion architecture can be further divided in terms of whether or not the higher level fusion centers transmit feedback to the lower level fusion centers. Although feedbacks can differ in terms of the information passed, in general, it contains fused estimates (Bar-Shalom and Blair, 2000). In this case the lower level fusion center can be simply re-initialized by using the estimates sent by the higher level fusion center.

1.4.2 The Correlation Among Local Estimation Errors

One of the major issues in estimation fusion is the cross-correlation among the local estimates (Bar-Shalom and Campo, 1986). This may not pose a severe problem in the static estimation fusion case since the only source of the cross-correlation is among the measurements from different sensors. This cross-correlation can be estimated and/or the raw measurements can be de-correlated. However, in a dynamic estimation fusion problem there are two more sources of cross-correlation, namely:

- common history of measurement errors
- common process noise.

The cross-correlation due to common history of measurement errors arises as the estimates communicated by a local fusion center at different times may use a common set of measurements. Hence, the same measurement errors arrive at the fusion center at different times. On the other hand, the errors in the state transition model corresponding to different local fusion centers are often cross-correlated, which causes the common process noise to be a part of the cross-correlation among estimates.

1.4.3 Track-to-Track Association

An important step in the track fusion process is the track-to-track association. The problem is similar to measurements to tracks association, however with increased complexity. First the dimension of the association problem can be higher. In measurement to track association we have one list of measurements and one list of tracks to associate. In track-to-track association we have as many list of tracks to associate as sensors participating in fusion. Further, as shown in the research presented in Section 3, the fused estimates could be used to improve the association and they enter as an extra list of tracks. Second, while the measurements errors and track estimate errors can be considered independent, this is not the case between tracks estimated at different local sensors. This correlation is the result of the common process noise used at local sensors and possibly due to feedback if applied.

1.5 Contribution

This thesis advances the research in the association and fusion of sensor estimates in distributed multisensor multitarget tracking. The first contribution is for the case where local sensors perform tracking using particle filters (Monte Carlo based methods). Given labeled particle clouds as local tracks estimates, a method of associating them is developed and is demonstrated with subsequent fusion. The cloud-to-cloud association cost is devised together with the computation method for the general case of non-Gaussian, non-resampled particle clouds and of different cardinalities. The method is also specialized for specific types of particle clouds. The cost introduced is proved to converge (with the clouds cardinality increase) toward the corresponding distance between the underlying distributions the particle clouds estimate. This first contribution is the subject of Chapter 2. In order to simulate the association method introduced, a particle filter labeled at particle level was developed, based on the Probability Hypothesis Density (PHD) (Mahler, 2003; Vo *et al.*, 2005) particle filter. The labeled particle filter introduced is detailed in Appendix A.

The second contribution is for the case where local sensors produce tracks using Kalman filter-type estimators, in the form of track state estimate and track state covariance matrix. For this case the association and fusion are improved in both terms of accuracy and identity, by introducing at each fusion time the prior information (both estimate and identity) from the previous fusion time. This is the subject of Chapter 3.

A third contribution is for the case where local sensors produce track estimates under the form of MHT, therefore where each local sensor produces several hypotheses of estimates. A method to use the information from other sensors in propagating each sensor's internal hypotheses over time is introduced in Chapter 4.

A practical fusion method for real world local tracking sensors, i.e., asynchronous and with incomplete information available, was developed and presented in Chapter 5. The sensor types are Over-the-Horizon (OTH) radar and Automatic Identification System (AIS) and data was received from DRDC Ottawa.

1.5.1 Publications Derived From Research

The following publications were derived from the research work:

Published Journal Article

- Sinha, A., Chen, H., Danu, D. G., Kirubarajan, T., and Farooq, M. (2008). Estimation and Decision Fusion: A Survey. In *Neurocomputing*, Vol. 71(13-15), pp. 2650-2656.

Journal Article Conditionally Accepted

- Danu, D. G., Kirubarajan, T., and Lang, T. (2010). Multisensor Particle Filter Cloud Fusion for Multitarget Tracking. Under 2nd revision *IEEE Transactions on Aerospace and Electronic Systems.*

Journal Articles in Preparation for Submission

- Danu, D. G. and Kirubarajan, T. Track-to-Track Association using Informative Prior Associations. To be submitted to *IEEE Transactions on Aerospace and Electronic Systems.*

- Danu, D. G., Chen, X., and Kirubarajan, T. Assignment-Based Particle Labeling for the PHD Particle Filter. To be submitted to *IEEE Transactions on Aerospace and Electronic Systems.*

- Danu, D. G., Chen, X., and Kirubarajan, T. Track Fusion with Feedback for Local Trackers Using MHT. To be submitted to *IEEE Transactions on Aerospace and Electronic Systems.*

Conference Papers

- Danu, D., Kirubarajan, T., and Lang, T. (2009). Wasserstein Distance for the Fusion of Multisensor Multitarget Particle Filter Clouds. In *Proceedings of the 12th International Conference on Information Fusion*, Seattle, USA.

- Danu, D. G., Lang, T., and Kirubarajan, T. (2009). Assignment-Based Particle Labeling for the PHD Particle Filter. In *SPIE Proceedings, Signal and Data Processing of Small Targets*, Vol. 7445, San Diego.

- Danu, D., Kirubarajan, T., Lang, T., and McDonald, M. (2008). Multisensor Particle Filter Cloud Fusion for Multitarget Tracking. In *Proceedings of the 11th International Conference on Information Fusion*, Cologne, Germany.

- Danu, D., Lang, T., and Kirubarajan, T. (2008). Track Fusion with Feedback for Local Trackers Using MHT. In *Proceedings of International SPIE Defense and Security Symposium*, Vol. 6969, Orlando, Florida.

- Danu, D., Sinha, A., and Kirubarajan, T. (2007). Track-to-Track Association Using Informative Prior Associations. In *SPIE Proceedings, Signal and Data Processing of Small Targets*, Vol. 6699, San Diego.

- Danu, D., Sinha, A., Kirubarajan, T., Farooq, M. and Brookes, D. (2007). Fusion of Over-the-Horizon Radar and Automatic Identification Systems for Overall Maritime Picture. In *Proceedings of the 10th International Conference on Information Fusion*, Québec.

- Danu, D., Sinha, A., Kirubarajan, T., Farooq, M., and Peters, D. (2007). Performance Evaluation of Multi-platform Distributed Data Fusion Methods for Multi-target Tracking. In *Proceedings of IEEE Aerospace Conference*, Big Sky, Montana.

- Sinha, A., Akselrod, D., Danu, D., Kirubarajan, T., Farooq, M. (2007). Network-Centric Multisensor-Multitarget Testbed with Results from Real Scenarios. *The First Northern Watch Conference and Exposition on Arctic C4ISR*, Halifax, Nova Scotia.

- Sinha, A., Chen, H., Danu, D. G., Kirubarajan, T., and M. Farooq. (2006). Estimation and Decision Fusion: A Survey. In *Proceedings of IEEE Conference on Engineering of Intelligent Systems.*

Chapter 2

Distributed Multisensor Particle Filter Cloud Fusion for Multitarget Tracking

This chapter introduces a novel method for the association and fusion of information from multiple particle filters in a distributed multisensor-multitarget scenario. Data considered herein for fusion consist of labeled tracks estimated through particle filter clouds. The cloud-to-cloud (C2C) association costs are computed using a novel distance defined at the particle cloud level, derived from the Wasserstein distance. Several association costs are presented, differing through the inner particle distance used within the C2C distance and through the type of particle clouds considered. These association costs use the whole estimated track information available through the particle clouds distributions, resulting in finer association than if only the estimates mean and sample covariance are used. Properties of the newly introduced distance are presented, including its closed form solution for the case of normally distributed clouds. A method of computing the distance for the general case is derived, based on linear programming interior point methods. Its convergence on particle filter clouds is proved theoretically when the Euclidean inner distance is used at the particle level and sample comparison with the classical track-to-track (T2T) test is given. Three methods of particle clouds fusion, namely, by direct combination of particles, by using the sample second order moments of the clouds, and by using the sample second order moments of clouds combined with their estimated sample cross-covariance, are presented. A simulation on a realistic distributed multisensor-multitarget scenario, with nonlinear target trajectories in clutter shows the improvement achieved by using more than one particle filter and the benefits of using the newly introduced association and fusion methods, compared with classical methods.

2.1 Introduction

Particle filters have been the subject of consistent research and improvements, with special emphasis on target tracking (Gordon *et al.*, 1993; Lin *et al.*, 2006). The essential work in (Gordon *et al.*, 1993) takes on a single target and, using the bootstrap particle filter, expands tracking performance beyond that of the classical Extended Kalman Filter (EKF) for nonlinear state dynamics and measurements. In (Arulampalam *et al.*, 2002) several versions of sampling and resampling developed in the meantime are presented, while in (Hue *et al.*, 2002) an extension of the particle filter to the multitarget case is introduced.

Multitarget particle filters were developed in (Clark and Bell, 2007; Hue *et al.*, 2002; Vermaak *et al.*, 2005) with the multitarget state being estimated through a single hypothesis with respect to the number of targets and measurement to targets association.

The fundamental work in (Mahler, 2003) on multitarget tracking based on random finite sets (RFSs) and finite set statistics (FISST) derives equations for sequential estimation of the first order multitarget moment or the probability hypothesis density (PHD). Based on the FISST and PHD approaches, new particle filters devised for multitarget scenarios, with the capability to handle target births, deaths and spawning were developed in (Vo *et al.*, 2005; Zajic and Mahler, 2003). The PHD particle filter was improved in (Danu *et al.*, 2009b; Lin *et al.*, 2006; Panta *et al.*, 2005)] with track identities, or track labels, use of which is made in the current work as input to the fusion process.

Within the Bayesian framework, particle filters that estimate the joint multitarget probability density (JMPD) in a multi-hypothesis approach were developed in (Kreucher *et al.*, 2005; Morelande *et al.*, 2007). One particle of these type of filters encodes a hypothesis about the entire multitarget state, including the number of targets and each target state. Special handling of the mixed labeling problem that occurs in these particle filters due to the coexistence of hypotheses that differ only in labels permutation is needed before estimations are extracted from these particle filters. The problem of closely spaced targets in a multitarget state is treated in (Blom and Bloem, 2007; Ekman *et al.*, 2007).

A particle filter fusion method applicable to a single target (i.e., when no association is needed) was introduced in (Coates, 2004). Fusion based on Gaussian mixture or Parzen approximations of particle clouds was presented in (Ong *et al.*, 2006, 2008). The information propagated from common observations due to the fusion with feedback method employed is handled through channel filters and particle division (Ong *et al.*, 2008).

An application of particle filters in a hierarchical data fusion system based on track fusion was discussed in (Lang and Dunne, 2008). Authors of (Lang and Dunne, 2008) have reported observations of instances of track-to-track (T2T) association errors due to the suboptimal approximation of particle clouds by their second-order statistics.

The purpose of this research was to introduce novel methods of association and fusion for track estimates provided under the form of labeled particle filter clouds. The methods are applicable to distributed multisensor-multitarget scenarios featuring sensors with overlapping surveillance areas and particle filters used as estimators. The fusion architecture is a decentralized one, where labeled particle clouds are obtained at local sensors, which in turn are made available to the fusion center. The fusion center could be one of the local estimators, in which case only the particle clouds of the other estimator(s) need to be communicated. Different novel ways of data association are introduced, based on the association costs computed at the particle cloud level, using a newly defined distance between clouds. The cloud-to-cloud (C2C) distance introduced is based on the Wasserstein distance and its value is defined through mapping at particle level between the clouds. This mapping links the discrete supports (particle locations) of the two underlying distributions estimated through the clouds. The transference plan (transportation matrix) established overcomes the problem of having to estimate the distance between two distributions with only partially overlapping (or not overlapping at all) supports, even when both pertain to the same true target. The data association methods differ in the type of particle clouds considered in computing the newly defined distance (e.g., before resampling, resampled, of equal or different cardinalities) and in the inner distance used at particles level within the aforementioned C2C distance. The C2C distance introduced here exploits in the computation of the association cost the entire information available through the clouds distributions. The usage of the whole information in the association cost is independent of the highest order moment characterizing the cloud distributions. This cost feature is paramount when associating particle clouds, whose distributions are characterized usually through moments of order higher than two. There are no approximations of the particle filter clouds, as the C2C distance is applied directly on the raw particles.

The association part of the method presented here is applicable to multitarget particle filters that capture the multitarget state in a single hypothesis (Clark and Bell, 2007; Hue *et al.*, 2002; Lin *et al.*, 2006), (Vermaak *et al.*, 2005; Vo *et al.*, 2005)]. The proposed association method preserves the local identities and, therefore, obtains at the fusion sensor track-valued estimates without the need of feedback at local sensors. Without feedback there are no common past observations contributing to the estimates being fused, and only the common process noise used in local dynamic models needs to be accounted for. The method is not directly applicable to multitarget particle filters whose output consists of several coexistent hypotheses (Kreucher *et al.*, 2005; Morelande *et al.*, 2007). More research is needed for the association of particles from such estimators in handling the coexistent hypotheses in the estimated joint multitarget probability density. The association method presented here is applicable to the particle filters handling closely spaced targets in (Blom and Bloem, 2007; Ekman *et al.*, 2007) if the closely spaced targets on each sensor are combined under the same cloud (label) for the duration they are closely spaced.

The C2C association cost is a distance measure, proved herein to be a metric, between any type of probability density functions (*pdf*) approximated through particle filters. A measure between Gaussian mixture *pdf*s was introduced in (Liu and Huang,

2000) but it was computed based on the distance between Gaussian distributions, which were used of equal and diagonal covariances in order to have it preserved as a metric [1].

For the fusion part, three methods are presented. They differ in the level of accounting for the dependence of the estimation errors, through the usage of sample covariance and cross-covariance of the particle clouds. The first method combines directly the particles of clouds to be fused and the second one considers in the fusion the sample covariance of the two clouds while the third one considers also the estimated sample cross-covariance of the two clouds.

The treatment of the estimators association and fusion at particle cloud level is a novelty, to the best of our knowledge. , as well as the application of the track-labeled PHD filter using resolution cells to a 2D scenario.

The chapter is structured as follows: the notion of labeled particle filter cloud and its equivalence to random probability measure are presented in Section 2.2. The method defined for data association performed at particle cloud level together with the costs introduced therein are presented in Section 2.3. Cost computation methods are presented in Section 2.4. Fusion methods for the associated particle filter clouds are derived in Section 2.5. Ground target tracking simulation and results are given in Section 2.6, and conclusions are drawn in Section 2.7.

[1] When estimating with a complex particle filter (e.g. PHD) that throws new particles around measurements, the closest a target particle cloud could get to the Gaussian distribution is being a pure mixture of at least two Gaussian distributions. This holds also for the non-realistic case in which the particle filter model, the sensor model and the true target trajectory, all follow the linear Gaussian model.

2.2 Particle Filter Clouds as Probability Measures

In the context of this research, a particle cloud is a cluster of particles of the same label representing the posterior probability distribution of a target state estimated by a particle filter [(Crisan, 2001), (Crisan and Doucet, 2002)]. Each particle in a cloud is defined by its state $\tilde{\xi}$, weight $\tilde{w}$, and label l. The states indicate particles' location in the estimating space (which could consist of position, velocity in one, two, or more dimensions) and the weights indicate the estimated probability of the target existence at the locations given by the states in the state space, while the label indicates the identity of the target. We use the notations $\Xi_s^{N,l_s} = \{\tilde{\xi}_s^i, \tilde{w}_s^i, l_s\}_{i=1}^N$ for a particle cloud of cardinality N and label l_s (with $l_s = 1, \ldots, L_s$) on sensor s (with $s = 1, \ldots, S$). Therefore, in the space defined by the particle state $\tilde{\xi} \in \mathbb{R}^{n_x}$, where n_x is the size of the target state $\mathbf{x} \in \mathbb{R}^{n_x}$, the mass distribution of a particle cloud Ξ^N is a probability mass function [(Bar-Shalom *et al.*, 2001)]

$$p_{\Xi^N}\left(\tilde{\xi}\right) = \sum_{i=1}^{N} \tilde{w}^i \delta_{\tilde{\xi}^i}\left(\tilde{\xi}\right), \tag{2.1}$$

where $\delta_{x_0}(\cdot)$ is the Dirac delta function concentrated at x_0 (Bar-Shalom *et al.*, 2001). Equation (2.1) is a Monte Carlo approximation of the posterior probability distribution of the target state on the support provided by the particle cloud Ξ^N [(Arulampalam *et al.*, 2002), (Crisan, 2001)]. Given the randomness of the particle sampling (and resampling) processes, the particle cloud is a random measure that approximates this posterior [(Crisan, 2001)]. If the same observation process $Y_{0:t} = \{\mathbf{y}_\tau : \mathbf{y}_\tau = \mathbf{h}(\mathbf{x}_\tau) + \mathbf{w}_\tau,\ \tau = 0, ..., t\}$ is used on different nodes (e.g., identical sensors, with same measurement model $\mathbf{h}(\cdot)$ and noise model $\mathbf{w}$ on all sensors), then the σ-algebra generated by the observation process on all nodes is the same. If identical dynamic models are used on all sensors for a given target, the posterior density of the state conditioned on the σ-algebra of measurements, $\pi_s^{Y_{0:t}}(\mathbf{x}_t)$ on sensor s, is identical on all sensors [(Crisan, 2001)]

$$\pi_1^{Y_{0:t}} = \cdots = \pi_S^{Y_{0:t}} \triangleq \pi\left(\mathbf{x}_t \mid \sigma\left(Y_{0:t}\right)\right). \tag{2.2}$$

For a sample realization of the observation process on sensor s, given by the event $\{Y_{0:t} = \mathbf{y}_{0:t}^s\}$, the above posterior densities become deterministic probability measures, and as measurements are different from sensor to sensor, the deterministic probability measures are also different

$$\pi_1^{y_{0:t}^1} \triangleq \pi\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}^1\right) \neq \cdots \neq \pi_S^{y_{0:t}^S} \triangleq \pi\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}^S\right). \tag{2.3}$$

The C2C association process therefore is performed between probability measures. It could be performed on either one of the measures, random or deterministic, in the former case associating identical measures on each sensor for a target, whereas in the latter associating for a given target, the closest probability measures on each sensor. The choice of associating the deterministic probability measures in (2.3) is followed in this research. Given the deterministic density of the target in (2.3) at one sensor, the particle filter cloud is a random probability measure that approximates it. In [(Crisan, 2001)] it was shown that under certain conditions that the particle filter must satisfy[2], the probability mass distribution in (2.1) is a sequence of random

[2] The transition kernel of the model is assumed Feller (i.e., its composition with any continuous bounded function is a bounded continuous function), and the likelihood function of the measurement

measures that converges weakly to the deterministic one. Using further the simplified notation Ξ^N for the probability mass distribution in (2.1), that is

$$\lim_{N\to\infty} \Xi^N \to \pi. \tag{2.4}$$

Therefore, the problem of multisensor multitarget particle clouds fusion is of discrete random probability measures estimated by the multiple sensors on the multitarget scenario. This process implies first the association of the particle clouds estimated on each sensor as representing the same target, followed by the combination of (estimates from) the associated clouds. The particle filter clouds association is treated in Sections 2.3 and 2.4, while their subsequent fusion is treated in Section 2.5. The fusion part is divided into two main stages, the first being the data association and second the estimation from combined associated data. Data to be fused here consists of labelled particles and their corresponding weights. Next the two stages are treated separately, in developing the fusion method for two sensors.

2.3 Association of Particle Filter Clouds

One main step of the data fusion process in distributed target tracking is track-to-track (T2T) association, in which tracks from different local estimates are correlated (or grouped) as pertaining to the same target. T2T association is a computationally intensive task, in most cases its complexity increasing exponentially with the number of sensors and targets. When fusing particle clouds, the problem is further complicated by having for each target a large number of representing particles. This

model is assumed bounded, continuous and strictly positive.

problem is greatly alleviated by labeling particles estimated as representing the same target at the sensor level, prior to association.

Using the notation introduced in Section 2.2 we denote by $\Xi_1^{l_1}$ a l_1-labeled cloud at sensor $s = 1$, and by $\Xi_2^{l_2}$ a l_2-labeled cloud at sensor $s = 2$, with labels $l_1 = 1..\hat{N}_1$ and $l_2 = 1..\hat{N}_2$. Here $\hat{N}_1$ and $\hat{N}_2$ are the number of targets, i.e. tracks approximated through particle clouds, estimated on sensor 1 and sensor 2, respectively. We assume that on both sensors labels l_1, l_2 are contiguous (in sequence) and start from 1 only for the purpose of simplifying the notation. We denote by $c\left(\Xi_1^{l_1}, \Xi_2^{l_2}\right)$ the cost of the hypothesis that clouds $\Xi_1^{l_1}$ and $\Xi_2^{l_2}$ correspond to the same target. Similarly to the case of T2T association, if the cloud association events among different cloud pairs are assumed independent, the most likely C2C association hypothesis can be found, within the 2D[3] assignment formulation, by solving the following constrained optimization

$$\min_{\chi} \sum_{l_1=0}^{L_1} \sum_{l_2=0}^{L_2} \chi_{l_1 l_2} \cdot c\left(\Xi_1^{l_1}, \Xi_2^{l_2}\right) \tag{2.5}$$

subject to

$$\sum_{l_1=0}^{L_1} \chi_{l_1 l_2} = 1, \quad l_2 = 1\,..\,L_2 \tag{2.6}$$

$$\sum_{l_2=0}^{L_2} \chi_{l_1 l_2} = 1, \quad l_1 = 1\,..\,L_1 \tag{2.7}$$

where $\chi_{l_1 l_2}$ is a binary variable. The labels l_1=0 and l_2=0 in (2.5)–(2.7) are used to denote the association with a dummy track. That is, for a cloud associated with one of label zero, no matching pair was found for it on the other sensor. The minimization

[3] The optimization problem generalizes to S-D assignment when the number of sensors S > 2 (Pattipati *et al.*, 2000).

in (2.5)–(2.7) can be solved using, for example, the auction algorithms [(Bertsekas and Castanon, 1993), (Jonker and Volgenant, 1987)]. It remains to be determined the evaluation of the C2C association cost, $c\left(\Xi_1^{l_1}, \Xi_2^{l_2}\right)$, which is treated in the next section. In computing the C2C association cost, two types of clouds are considered (from the particle resampling viewpoint): clouds of unresampled particles and clouds of resampled particles. A gating between clouds has to be used first, such that some costly multiassignments (defined next) are skipped.

As seen in Section 2.2, the particle filter clouds are random probability measures, therefore their association cost needs to be one that penalizes the distance and, implicitly, the dissimilarity between these measures.

2.3.1 Distance Between Probability Measures

There are several distances between probability measures described in the literature, however most of them require a common support for the two measures, i.e. a common space where both have non-zero values, in order to achieve a proper estimation. Some that do not requires this are listed below.

Levy-Prohorov Distance (Rachev and Ruschendorf, 1998)

for probabilities P_1 and P_2 it is defined as

$$d^{LP}(P_1, P_2) = \inf\{\varepsilon \mid \forall A \in \mathcal{B},\, P_1(B) \leq P_2(A^\varepsilon) + \varepsilon,\, P_2(A) \leq P_1(A^\varepsilon) + \varepsilon\}, \quad (2.8)$$

where A^ε is the ε-neighborhood of the set A of events, and $\mathcal{B}$ is the Borel σ-algebra of events on which P_1 and P_2 are defined. This distance is used mostly for theoretical expositions and is difficult to compute.

Haussdorff Distance (Hoffman and Mahler, 2004)

between two sets A and B (particle clouds can be seen as sets of particles) is defined as

$$d^H(A,B) = \max\{\inf\{\varepsilon > 0 \mid A^\varepsilon \supset B\}, \inf\{\varepsilon > 0 \mid B^\varepsilon \supset A\}\}. \tag{2.9}$$

In the Haussdorff distance the boundary particles have the most impact whereas their internal distribution and weight is not taken into account, making it poorly suited for characterizing the similarity of discrete distributions.

Wasserstein Distance (Vasershtein, 1969)

is defined between two probabilities P_1 and P_2 as

$$\ell_p(P_1,P_2) = \inf \sqrt[p]{\mathrm{E}\left[d(X,Y)^p\right]} \tag{2.10}$$

where X, Y are random variables distributed with probabilities P_1, respectively P_2 and d is a distance defined between X and Y. The infimum is taken over all joint distributions of X and Y, having marginals P_1 and P_2. In the literature this distance is also found under the names Kantorovich, Mallows, Earth Mover's Distance, Kantorovich-Rubinstein, depending on its order p or domain of applications. Vasershtein introduced the distance using $p = 1$ only.

2.3.2 Properties of the Wasserstein Distance

The Wasserstein distance can be used to assess the dissimilarity between two probability densities $f(\mathbf{x}_1)$, $g(\mathbf{x}_2)$ defined on the target state space $(\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^{n_x})$

$$d_p^W(f,g) = \inf_h \sqrt[p]{\int d(\mathbf{x}_1,\mathbf{x}_2)^p h(\mathbf{x}_1,\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2}, \tag{2.11}$$

where $d(\mathbf{x}_1, \mathbf{x}_2)$ is the (inner) distance between states (e.g. Euclidian, Mahalanobis) and $h(\mathbf{x}_1, \mathbf{x}_2)$ is the joint distribution of the states (whose marginals are the two densities $\int h(\mathbf{x}_1,\mathbf{x}_2)d\mathbf{x}_1 = g(\mathbf{x}_2)$ and $\int h(\mathbf{x}_1,\mathbf{x}_2)d\mathbf{x}_2 = f(\mathbf{x}_1)$) (Hoffman and Mahler, 2004).

In a mathematical formulation the Wasserstein distance defines the weak topology on the set of probability measures on a given space, and for dense probability measures on the given space, it defines a separable and complete metric if the inner distance $d(\cdot,\cdot)$ defines a separable and complete metric on the given space [(Villani, 2003)]. For example, this is the case if the Euclidean distance is used for $d(\cdot,\cdot)$ [(Carillo and Toscani, 2007)]. For distributions P_1, P_2 of two random variables (r.v.) $\mathbf{x}_1$, $\mathbf{x}_2 \in \mathbb{R}^{n_x}$ of means $\mathbf{m}_1$, $\mathbf{m}_2$ the Wasserstein distance between is always greater than the Euclidean distance between their means. From (Givens and Shortt, 1984):

$$d_2^W(P_1,P_2) = \sqrt{\|\mathbf{m}_1 - \mathbf{m}_2\|^2 + d_2^W(Q_1,Q_2)^2}, \tag{2.12}$$

where Q_1, Q_2 are the distributions of r.v. $\mathbf{x}_1 - \mathbf{m}_1$ and $\mathbf{x}_2 - \mathbf{m}_2$, respectively. Therefore, the Wasserstein distance between r.v. that have the same mean can be non-zero. This could be useful in distinguishing between probability distributions that pertain to targets compared to probability distributions that pertain to clutter or other false alarms, even if they are collocated, provided their distributions are sufficiently different.

Recently in [(Schuhmacher *et al.*, 2008)] a miss-distance between multiobject sets was introduced under the optimal subpattern assignment (OSPA) name for the purpose of multitarget filtering performance evaluation. For two sets, this distance penalizes both localization errors between the objects within the two sets and also the difference between the cardinalities of the two sets. We stress that this distance is not a candidate that serves our purpose, as the cardinalities of the clouds are not at all a criterion that affects the closeness of their underlying distributions. For example two identical distributions could be sampled with very different number of particles in the two clouds and the difference between their cardinalities should not be penalized by the C2C distance we are to define. The C2C distance was designed to measure solely the difference between the clouds underlying distributions.

2.3.3 Definition of C2C Association Cost

A particle cloud is a finite set of particles. Therefore the cost of associating two clouds can be formulated as the distance between the two sets (of M_1 and M_2 particles) defining the clouds. Also as seen in Section 2.2, the particle cloud is a random measure through a discrete number of points of the underlying deterministic distribution modeled by the particle filter for a given track. Therefore, based on the properties in Section 2.3.2 the Wasserstein distance is a good candidate in defining the C2C association cost. However, because we apply it between random measures, it cannot be defined directly and its properties are valid only under certain constraints.

Prior to computing the cost of associating two clouds

$$\Xi_s = \left\{\tilde{w}_s^i, \tilde{\xi}_s^i\right\}_{i=1}^{M_s}, \; s = 1,2 \tag{2.13}$$

on two sensors (labeled clouds, however, with label dropped here for notation simplification), both clouds are normalized such that their weights sum to one. This is done in order to re-interpret the particle cloud as a discretized representation of the target's probability density function in 2D-Cartesian space, of which integral is one

$$\int_{dA \subset R} p_s(x,y) \cdot dA(x,y) = 1 \tag{2.14}$$

where R is the region over which the density is estimated, $p_s(\cdot)$ is the density function of the state and $dA(\cdot,\cdot)$ is the elementary surface area for estimation in x-y plane, or otherwise is the elementary volume of the state space.

For the normalized weights we have

$$\sum_{i=1}^{M_s} \tilde{w}_s^i = \sum_{i=1}^{M_s} \frac{\tilde{w}_s^i}{\Delta A(\tilde{\xi}_s^i)} \Delta A(\tilde{\xi}_s^i) = \sum_{i=1}^{M_s} p_s(\tilde{\xi}_s^i) \cdot \Delta A(\tilde{\xi}_s^i) = 1 \tag{2.15}$$

where $\Delta A(\tilde{\xi}_s^i)$ is the area containing all points of which nearest neighbor particle is $\tilde{\xi}_s^i$, and such that the sum of these areas cover a contiguous region enclosing all particles of the cloud. From (2.15), upon normalization, each particle weight corresponds to the pdf of the cell surrounding the particle (of area $\Delta A(\tilde{\xi}_s^i)$ around $\tilde{\xi}_s^i$) multiplied by the area cell $\tilde{w}_s^i = p_s^i(\tilde{\xi}_s^i) \cdot \Delta A(\tilde{\xi}_s^i)$, representing the cell probability mass.

The inclusion of state velocity component into particle distance could be added by using for $d(\cdot,\cdot)$ the Mahalanobis distance, or by projecting the velocities into the position through multiplications by small time intervals $\delta\tau$.

However, as noted in [(Morelande *et al.*, 2007)] if just the measurement equation is nonlinear while the process equation is modeled through a linear Gaussian one, the particle filter can be implemented using particles containing position data, as the velocity can be derived from the position estimates using a Kalman filter-type estimator.

In [(Hoffman and Mahler, 2004)] the multitarget miss distance is introduced based on the Wasserstein distance as a means to assess the multitarget tracking performance (through providing a distance between the sets of estimated tracks and true targets). Here we use the adjustment of Wasserstein distance to finite sets and adapt it to sets of particles (clouds). The particles in a cloud can be weighted (2.1) and therefore the distance we introduce accounts for these weights as well. We use for functions f and g the sum of Dirac delta functions $\delta_{\Xi_1}(\tilde{\xi}_1)$ and $\delta_{\Xi_2}(\tilde{\xi}_2)$, defined on the surfaces $\tilde{\xi}_s = \{(x, y)\}$, covered by both sensors s = 1, 2 and with values on the discrete sets corresponding to the two particle clouds $\Xi_1 = \left\{\tilde{\xi}_1^1, \dots, \tilde{\xi}_1^{M_1}\right\}$, $\Xi_2 = \left\{\tilde{\xi}_2^1, \dots, \tilde{\xi}_2^{M_2}\right\}$

$$\delta_{\Xi_s}(\tilde{\xi}) = \sum_{i=1}^{M_s} \tilde{w}_s^i \cdot \delta_{\tilde{\xi}_s^i}(\tilde{\xi}_s), \quad s = 1,\, 2 \tag{2.16}$$

and $h(\cdot, \cdot)$ defined as

$$h(\mathbf{x}_1, \mathbf{x}_2) = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} w_{ij} \delta_{\tilde{\xi}_1^i}(\mathbf{x}_1) \delta_{\tilde{\xi}_2^j}(\mathbf{x}_2) \tag{2.17}$$

where w_{ij} are elements of the transportation matrix W known under this name from the linear programming special case of transportation problem. Introducing (2.16) and (2.17) in (2.11), we obtain the C2C association cost based on the Wasserstein distance as

$$d_p^W(\Xi_1, \Xi_2) = \inf_W \sqrt[p]{\sum_{i=1}^{M_1} \sum_{j=1}^{M_2} w_{ij} d(\tilde{\xi}_1^i, \tilde{\xi}_2^j)^p}, \tag{2.18}$$

with elements of matrix W satisfying the constraints

$$\sum_{i=1}^{M_1} w_{ij} = \tilde{w}_2^j, \quad j = 1 \mathinner{..} M_2, \tag{2.19}$$

$$\sum_{j=1}^{M_2} w_{ij} = \tilde{w}_1^i, \quad i = 1 \mathinner{..} M_1. \tag{2.20}$$

Constraints (2.19) and (2.20) assure that each particle in both clouds is fully assigned using its full weight to particles from the other cloud. The minimization problem (2.18)-(2.20) is a multiassignment problem in which each particle is assigned to several particles with different sub-weights, similar to the transportation problem, which is a special case of linear programming. Different methods for computing the C2C distance and association cost based on (2.18)-(2.20) are derived further, depending on the distance d used at particle level and on the type of particles used (i.e., resampled or before resampling).

2.3.4 Properties of C2C Distance (Association Cost)

In this subsection the convergence of the newly introduced distance between clouds is explored as well as the conditions under which it satisfies the properties of a metric. It is noted that the C2C distance as introduced in (2.18) does not penalize the difference between cardinalities of the two clouds M_1 and M_2, as this is not desired for the case of particle clouds. The C2C cost introduced penalizes differences in particles distribution whereas the difference in the number of particles between clouds is absorbed by allowing fractional particles association through the transportation matrix W, which has real values. This is a feature that makes the C2C distance different from the multitarget miss distance (Hoffman and Mahler, 2004), in which the difference between cardinalities needs to be taken into account.

Convergence of the Wasserstein C2C Distance

As shown in Section 2.2, a particle filter cloud is an (empirical) random measure of the deterministic posterior distribution of the target state obtained through the filter models. The requirements on the particle filter for its particle estimate to converge (as a sequence of random probability measures) toward the underlying deterministic posterior probability distribution of the target state were established in [(Crisan, 2001), (Crisan and Doucet, 2002)]. These requirements were briefly restated in Section 2.2 and here we assume they are met. In this subsection the convergence of the C2C distance, defined in (2.18)-(2.20) between particle clouds as random measures of the underlying deterministic posterior densities, toward the Wasserstein distance between these underlying densities is explored. The theoretical proof of convergence is given for the case when the Euclidean distance is used at particles level (as *inner distance*), while the convergence in other cases is also explored empirically in Section 2.6 as a function of:

- number of weighted particles
- resampling method and number of particles
- internal distance used (i.e. Mahalanobis)

Convergence of the Euclidean Wasserstein C2C Distance

We use the name Euclidean Wasserstein for the distance in (2.11), which applies to deterministic distributions, and C2C Euclidean Wasserstein for the distance defined in (2.18), which applies to particle filter clouds (random measures). In both cases we use the notation $d_2^{WE}(\cdot,\cdot)$, through which we imply the Euclidean distance is used for the inner distance $d(\cdot,\cdot)$ and the order is $p = 2$. It is known that the Wasserstein distance (2.11) is a metric that defines the weak topology on a space if this given space is endowed by the inner distance $d(\cdot,\cdot)$ with the complete separable metric property (Villani, 2009, Th.6.9). This means that the weak convergence to zero of the distance d_2^{WE} between a sequence of probability measures, π_k, and a given probability measure, π, is equivalent to the weak convergence of the sequence of measures toward the given measure:

$$\lim_{k\to\infty} d_2^{WE}(\pi_k,\pi) \rightharpoonup 0 \Longleftrightarrow \lim_{k\to\infty} \pi_k \rightharpoonup \pi. \tag{2.21}$$

We assume the particle filter satisfies the conditions necessary for its particle cloud estimate to converge toward the underlying deterministic posterior distribution estimated for the target state (2.4) established in (Crisan and Doucet, 2002) (e.g., satisfied for the bootstrap filter and multinomial resampling). Under these assumptions, the C2C Euclidean Wasserstein distance satisfies the convergence property defined through following theorem:

Theorem 1 (C2C Euclidean Wasserstein Distance Convergence). *Given two particle filter clouds, Ξ_1^N and Ξ_2^N, estimated by two distinct particle filters based on two different sequences of measurements $y_1^{0:t}$, $y_2^{0:t}$, the C2C Euclidean Wasserstein distance between them, $d_2^{WE}(\Xi_1^N,\Xi_2^N)$, converges weakly towards the Euclidean Wasserstein* *distance between the underlying posterior probabilities,* π_1, π_2*:*

$$\lim_{N\to\infty} d_2^{WE}(\Xi_1^N, \Xi_2^N) \rightharpoonup d_2^{WE}(\pi_1, \pi_2). \tag{2.22}$$

Three properties of the Euclidean Wasserstein distance (Villani, 2003) are listed below as they are used further in the reasoning.

P.1 For a sequence of random probability measures f_n that converges weakly to a deterministic probability measure f, the Euclidean Wasserstein distance between f_n and f converges weakly to zero and the reverse

$$\lim_{n\to\infty} d_2^{WE}(f_n, f) \rightharpoonup 0 \qquad \Longleftrightarrow \qquad \lim_{n\to\infty} f_n \rightharpoonup f. \tag{2.23}$$

P.2 (Weak lower semicontinuity) For two sequences of random probability measures f_n, g_n which converge weakly to two deterministic probability measures, $\lim_{n\to\infty} f_n \rightharpoonup f$ and $\lim_{n\to\infty} g_n \rightharpoonup g$, we have

$$d_2^{WE}(f, g) \leq \lim_{n\to\infty} \inf d_2^{WE}(f_n, g_n). \tag{2.24}$$

P.3 Triangle inequality

$$d_2^{WE}(f, g) \leq d_2^{WE}(f, h) + d_2^{WE}(h, g). \tag{2.25}$$

The sequences of the two particle filter clouds Ξ_1^n, Ξ_2^n of cardinalities n are assumed to converge weakly to the posterior distributions of the estimated target(s), π_1, and π_2, respectively, (by satisfying requirements in (Crisan and Doucet, 2002)), therefore based on P.1. we have

$$\lim_{n\to\infty} d_2^{WE}\left(\Xi_1^n, \pi_1\right) \rightharpoonup 0 \quad \text{and} \quad \lim_{n\to\infty} d_2^{WE}\left(\Xi_2^n, \pi_2\right) \rightharpoonup 0. \tag{2.26}$$

Using twice the triangle inequality (2.25) we get

$$d_2^{WE}\left(\Xi_1^n, \Xi_2^n\right) \leq d_2^{WE}\left(\Xi_1^n, \pi_1\right) + d_2^{WE}\left(\Xi_2^n, \pi_2\right) + d_2^{WE}\left(\pi_1, \pi_2\right) \tag{2.27}$$

therefore

$$d_2^{WE}\left(\Xi_1^n, \Xi_2^n\right) \leq \varepsilon_1(n) + \varepsilon_2(n) + d_2^{WE}\left(\pi_2, \pi_1\right) \tag{2.28}$$

with $\lim\limits_{n\to\infty} \varepsilon_1(n) \rightharpoonup 0$, $\lim\limits_{n\to\infty} \varepsilon_2(n) \rightharpoonup 0$. From (2.24) we have for the particle clouds Ξ_1^n, Ξ_2^n

$$\lim_{n\to\infty} \inf d_2^{WE}\left(\Xi_1^n, \Xi_2^n\right) \geq d_2^{WE}\left(\pi_1, \pi_2\right) \tag{2.29}$$

and combining (2.28) at the limit with (2.29) it results in

$$\lim_{n\to\infty} d_2^{WE}\left(\Xi_1^n, \Xi_2^n\right) \rightharpoonup d_2^{WE}\left(\pi_2, \pi_1\right) \tag{2.30}$$

Therefore, for particle filter clouds converging to the deterministic posterior distributions of the estimated targets, the Euclidean Wasserstein distance converges to the one between the deterministic conditional posterior distributions of the targets estimates.

Corollary 1. For two particle filter clouds Ξ_1^N, Ξ_2^N estimated by two (distinct) identical particle filters, based on the same sequence of measurements $y^{0:t}$, the C2C d_2^{WE} distance converges weakly to zero $\lim_{N\to\infty} d_2^{WE}\left(\Xi_1^N, \Xi_2^N\right) \rightharpoonup 0$.

Metric Properties

The C2C Euclidean Wasserstein distance satisfies the three properties of a metric, as being based on the Wasserstein distance, which is a metric itself if the inner distance $d(\cdot, \cdot)$ is a metric:

1. distance is zero for (and only for) identical measures

2. symmetry

3. triangular inequality

Based on Theorem 1 the metric properties above apply at the limit ($N \rightarrow \infty$) to the underlying deterministic (non-random) probability measures that the particle clouds estimate. Therefore the C2C Euclidean Wasserstein distance defined at the particle cloud level defines at the limit a metric on the underlying deterministic probability measures.

Comparison with the T2T Association Cost

In this subsection a comparison of the C2C association cost in (2.18)-(2.20) with the T2T association cost introduced in (Bar-Shalom and Li, 1995) is made for simple cases of Gaussian posterior distributions. First it is shown through a counterexample that the T2T test in (Bar-Shalom and Li, 1995) is not a metric on the space of estimated tracks posterior distributions and second the C2C association cost is compared with the T2T test on a Gaussian example.

Example 1. Given the three estimated tracks T_1, T_2, T_3, assumed in the 1-D space and with their posteriors normally distributed, therefore characterized by state and

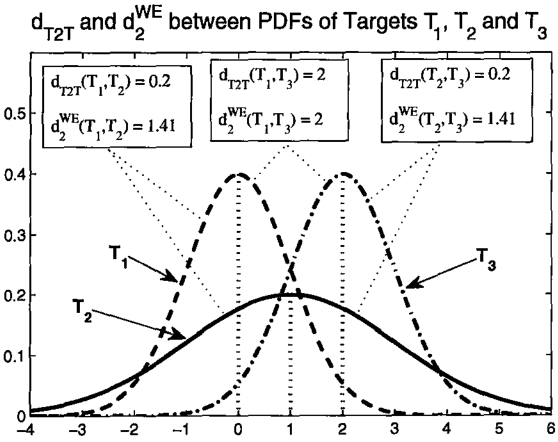


Figure 2.1: $T2T$ and d_2^{WE} distances between track estimates (posterior PDFs) of targets T_1, T_2, T_3.

variance $T_n(\hat{x}_n, \sigma_n^2)$, for $n = 1 \ldots 3$, as shown also in Fig.2.1

$$T_1(\hat{x}_1 = 0, \sigma_1^2 = 1), \qquad T_2(\hat{x}_2 = 1, \sigma_2^2 = 4), \qquad T_3(\hat{x}_3 = 2, \sigma_3^2 = 1), \tag{2.31}$$

we have the following T2T association costs between them, $d_{T2T} = \frac{(\hat{x}_1 - \hat{x}_2)^2}{P_1 + P_2}$, as defined in (Bar-Shalom and Li, 1995):

$$d_{T2T}(T_1, T_3) = \frac{(x_3 - x_1)^2}{\sigma_1^2 + \sigma_3^2} = \frac{(2-0)^2}{1+1} = 2, \tag{2.32}$$

$$d_{T2T}(T_1, T_2) = \frac{(x_2 - x_1)^2}{\sigma_1^2 + \sigma_2^2} = \frac{(1-0)^2}{1+4} = 0.2, \tag{2.33}$$

$$d_{T2T}(T_2, T_3) = \frac{(x_3 - x_2)^2}{\sigma_2^2 + \sigma_3^2} = \frac{(2-1)^2}{1+4} = 0.2. \tag{2.34}$$

Therefore from (32)-(34) we have

$$d_{T2T}(T_1, T_3) > d_{T2T}(T_1, T_2) + d_{T2T}(T_2, T_3) \tag{2.35}$$

which shows the T2T association test in (Bar-Shalom and Li, 1995) does not satisfy the triangle inequality and therefore is not a metric.

Example 2. In this 1-D example of associating two estimated tracks, i.e., $\hat{x}_1^1$, $\hat{x}_1^2$ from sensor $s = 1$ to the one track $\hat{x}_2^1$, estimated on sensor $s = 2$, the T2T association cost is shown to associate the estimated tracks in an unexpected manner. Given the estimated tracks posterior Gaussian distributions, characterized by $\hat{x}_1^1 = 2.5$, $\sigma_1^1 = 1.0$, $\hat{x}_1^2 = 3.0$, $\sigma_1^2 = 2.0$ and $\hat{x}_2^1 = 2.7$, $\sigma_2^1 = 1.0$, as shown in Fig.2.2, based on the T2T association test, $d_{T2T} = \frac{(\hat{x}_1 - \hat{x}_2)^2}{P_1 + P_2}$ (under the error independence assumption), we get the association

$$\begin{gathered} d_{T2T}(T_1^1, T_2^1) = \tfrac{0.2 \times 0.2}{1^2 + 1^2} = 0.02, \quad d_{T2T}(T_1^2, T_2^1) = \tfrac{0.3 \times 0.3}{1^2 + 2^2} = 0.018, \\ => T_1^2 \quad \text{associated to} \quad T_2^1. \end{gathered} \tag{2.36}$$

By using the $d_2^{WE}(\cdot, \cdot)$ for 1-D Gaussian distributions in (2.40) we get the contrary association

$$\begin{gathered} d_2^{WE}(T_1^1, T_2^1) = \sqrt{0.2^2 + 0^2} = 0.2, \quad d_2^{WE}(T_1^2, T_2^1) = \sqrt{0.3^2 + 1^2} = 1.044, \\ => T_1^1 \quad \text{associated to} \quad T_2^1, \end{gathered} \tag{2.37}$$

which seem more probable given the tracks $\hat{x}_1^1$ and $\hat{x}_2^1$ have closer means and have more alike distributions.

2.4 Computation of the C2C Association Cost

The calculation of the C2C association cost using the Wasserstein distance between particle clouds, as defined in (2.18)–(2.20) was carried out for the general case first in (Danu *et al.*, 2008a) through linear programming interior-point methods (Zhang,

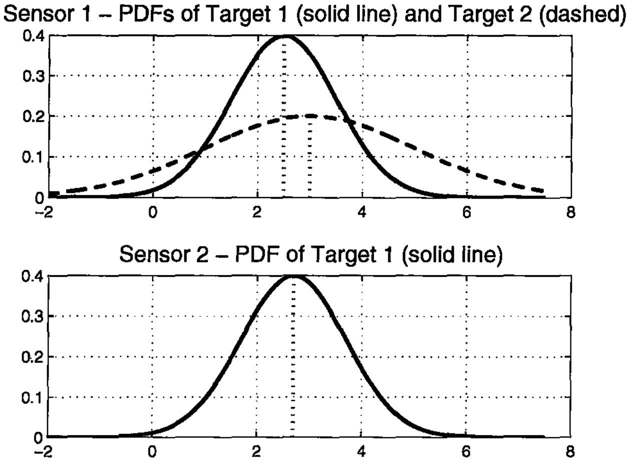


Figure 2.2: PDFs of estimated tracks on the two sensors.

1999). It is presented below, together with the formulas from (Givens and Shortt, 1984) applicable only to the special case of Gaussian distributed clouds and Euclidean distance used as inner particle distance. The computation in the general case becomes very intensive as the cardinalities of the two clouds get higher. Therefore, the implications of using in this computation a restricted number of highest weighted particles, or resampled clouds of lesser cardinalities are explored in Section 2.6. Also the resolution obtained in the Wasserstein distance by using the Euclidean or T2T distance at particle level is experimented.

2.4.1 C2C for Gaussian Particle Clouds Using Euclidean Wasserstein Distance

For the special case of particle clouds of Gaussian probability distributions, the Euclidean Wasserstein distance of order $p = 2$ has a closed form expression (Givens and Shortt, 1984), which for two probability distributions P_1, P_2 of random variables in

$\mathbb{R}^n$, with means $\mathbf{m}_1$, $\mathbf{m}_2$ and covariances $\mathbf{P}_1$, $\mathbf{P}_2$ in $\mathbb{R}^{n \times n}$, is

$$d_2^{WE}(P_1, P_2) = \sqrt{\|\mathbf{m}_1 - \mathbf{m}_2\|^2 + tr\mathbf{P}_1 + tr\mathbf{P}_2 - 2tr\left[\left(\sqrt{\mathbf{P}_1}\mathbf{P}_2\sqrt{\mathbf{P}_1}\right)^{1/2}\right]} \tag{2.38}$$

Based on (2.38) and the derivation of distance between zero mean distributions in (Givens and Shortt, 1984), we have for $n = 2$

$$d_2^{W}(P_1, P_2) = \sqrt{\|\mathbf{m}_1 - \mathbf{m}_2\|^2 + trP_1 + tr\mathbf{P}_2 - 2\left[tr(\mathbf{P}_1\mathbf{P}_2) + 2\sqrt{\det(\mathbf{P}_1\mathbf{P}_2)}\right]^{1/2}} \tag{2.39}$$

and for $n = 1$

$$d_2^{W}(P_1, P_2) = \sqrt{\|m_1 - m_2\|^2 + |\sigma_1 - \sigma_2|^2}. \tag{2.40}$$

2.4.2 C2C for Non-Gaussian Weighted Particle Clouds

The computation of the general Wasserstein distance (not necessarily using the Euclidean distance as its inner distance), as introduced in (2.18)–(2.20), for weighted particle clouds of an arbitrary distribution, can be carried out through interior point linear programming methods, as we showed in (Danu *et al.*, 2008a). The computation implies finding the minimum cost solution to the multiassignment of particles in one cloud to the particles in the other cloud. In order to translate (2.18)–(2.20) in a classical linear program

$$\min_{\mathbf{w}} \ \mathbf{d}^T\mathbf{w} \tag{2.41}$$

subject to

$$\begin{aligned} \mathbf{A}\mathbf{w} &= \mathbf{b}, \\ \mathbf{w} &\geq \mathbf{0}, \\ \mathbf{w} &\leq \mathbf{1} \end{aligned} \tag{2.42}$$

we use the following for $\mathbf{d} \in \mathbf{R}^{\mathbf{M_1 M_2}}$, $\mathbf{w} \in \mathbf{R}^{\mathbf{M_1 M_2}}$, $\mathbf{A} \in \mathbf{R}^{\mathbf{M_1+M_2, M_1 M_2}}$, $\mathbf{b} \in \mathbf{R}^{\mathbf{M_1+M_2}}$.

$$\mathbf{d} = [d_{11}\ d_{12}\ ...\ d_{1M_2}\ d_{21}\ d_{22}\ ...\ d_{2M_2}\ ...d_{M_1 1}\ d_{M_1 2}\ ...\ d_{M_1 M_2}] \tag{2.43}$$

represents the vector of distances $d_{ij} = d\left(\tilde{\xi}_1^i, \tilde{\xi}_2^j\right)$,

$$\mathbf{w} = [w_{11}\ w_{12} ... w_{1M_2}\ w_{21}\ w_{22} ... w_{2M_2} ... w_{M_1 1}\ w_{M_1 2} ... w_{M_1 M_2}]^T \tag{2.44}$$

is the content of transportation matrix W in vector form,

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 1 & 1 & \dots & 1 \\ 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \dots & 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 & \dots & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & \dots & 0 & 0 & \dots & 1 \end{bmatrix} \tag{2.45}$$

and $\mathbf{b} = [\tilde{\mathbf{w}}_1^1\ \tilde{\mathbf{w}}_2^1\ ...\ \tilde{\mathbf{w}}_{M_1}^1\ \tilde{\mathbf{w}}_1^2\ \tilde{\mathbf{w}}_2^2\ ...\ \tilde{\mathbf{w}}_{M_2}^2]^{\mathrm{T}}$ is the (stacked) vector of particle weights of both clouds.

Once the Wasserstein distance is formulated as a constrained minimization problem, a sample package solving the above minimization is LIPSOL (Linear-programming Interior Point SOLver) (Zhang, 1999).

2.4.3 C2C for Non-Gaussian Resampled Particle Clouds

Using clouds of labeled resampled particles at the particle filter level produces a fixed number of particles per cloud, all having same weight (i.e. 1/M for M particles per cloud). The constant weight of all particles results in the special form of $\delta_{\Xi_s}(\xi)$ as

$$\delta_{\Xi_s}(\xi) = \frac{1}{M_s}\sum_{i=1}^{M_s}\delta_{\xi_s^i}(\xi_s), \quad s = 1,\, 2. \tag{2.46}$$

In this case the transportation matrix W has the sums on all rows equal, and the same holds for columns

$$\sum_{i=1}^{M_1} w_{ij} = \frac{1}{M_2}, \quad \sum_{j=1}^{M_2} w_{ij} = \frac{1}{M_1}. \tag{2.47}$$

Equation (2.18) combined with (2.47) represent the constrained optimization to be solved in estimating the distance between the clouds (objective function). Two subcases arise in the computation of this cost: when $|\Xi_1| = |\Xi_2|$ (i.e. clouds are of equal cardinalities) and when $|\Xi_1| \neq |\Xi_2|$.

Resampled Particle Clouds of Equal Cardinalities

For equal cardinalities, $M_1 = M_2 = M$ and all particles in both clouds have identical weights, $1/M$. Consequently the C2C association cost can be simplified to the optimal objective function of a simple 2-D assignment (i.e., W to have elements in $\{0, 1\}$ only). The minimization (2.18) becomes

$$d_p^W(\Xi_1, \Xi_2) = \min_{\sigma} \sqrt[p]{\frac{1}{M}\sum_{i=1}^{M} d(\xi_i, \xi_{\sigma i})^p} \tag{2.48}$$

with $p = 1, 2$, where σi is a permutation of $i = 1, \ldots, M$. Equation (2.48) represents the 2-D assignment problem (without dummy association allowed, as all particles should be assigned), which can be solved by finding optimum σ permutation using auction algorithms (Bertsekas and Castanon, 1993; Jonker and Volgenant, 1987).

Resampled Particle Clouds of Different Cardinalities

For clouds of different cardinalities, $M_1 \neq M_2$, even though particles in each cloud have equal weights (i.e., $1/M_1$ for $s = 1$ and $1/M_2$ for $s = 2$), they are different between clouds. In order to reduce this problem to a similar 2-D problem as in 2.4.3, both clouds need to be reduced to the same number of particles and of equal weights (in order to allow one-to-one associations only and therefore W with elements in {0,1}). Using a method similar to the one in (Hoffman and Mahler, 2004), each particle $i = 1, .. M_1$ in the cloud Ξ_1 will be divided into $M_1^* = M_2/\gcd(M_1, M_2)$ new particles of the same state, where gcd stands for the greatest common divisor, thus obtaining the new equivalent cloud Ξ_1^{new} of $|\Xi_1^{new}| = M_1^* M_1 = M_1 M_2/\gcd(M_1, M_2)$ particles of weights $w = \gcd(M_1, M_2)/(M_1 M_2)$. The same resampling is applied to the particles in the cloud Ξ_2, in consequence obtaining the equivalent new cloud Ξ_2^{new} of $|\Xi_2^{new}| = M_2^* M_2 = M_1 M_2/\gcd(M_1, M_2)$ particles of same weight w. At this stage as Ξ_1^{new} and Ξ_2^{new} have the same number of particles of same weight, the cost can be computed by following the same 2-D association method in subsection 2.4.3. Another solution would be to apply directly the method in subsection 2.4.2, to allow fractional multiassignment between particles of both clouds (i.e. to allow W matrix elements to take values in the interval $[0, 1]$).

2.4.4 Discussion

Clouds Cardinalities Implications

The methods presented in Section 2.4 do not penalize the difference in the cardinalities of the clouds as long as the clouds cardinalities allow them to be estimates close to the underlying distributions. The C2C computation method introduced in subsection 2.4.2 prevents the difference in clouds cardinalities from significantly affecting the C2C association cost through the fractional multiassignment of particles. For any two clouds, independent of their cardinalities, no particle or part of a particle weight remains unassigned. The C2C computation method simplified in subsection 2.4.3 for resampled clouds of different cardinalities brings the clouds to the same cardinalities by splitting the particles in smaller weights using the *gcd*. This makes, implicitly, their internal particles of equal weights. The numerical example below illustrates how the initial difference between clouds cardinalities does not affect the C2C cost. For weighted particle clouds, the C2C distances between the simplified cloud

$$\Xi_1^3 = \{\{-1, 0.2\}, \{0, 0.6\}, \{1, 0.2\}\} \tag{2.49}$$

to the clouds

$$\Xi_2^6 = \{\{-1, 0.1\}, \{-1, 0.1\}, \{0, 0.3\}, \{0, 0.3\}, \{1, 0.1\}\{1, 0.1\}\} \tag{2.50}$$

or

$$\Xi_3^5 = \{\{-1, 0.2\}, \{0, 0.2\}, \{0, 0.2\}, \{0, 0.2\}\{1, 0.2\}\}, \tag{2.51}$$

all of different cardinalities, have the same value 0. For the distributions above, we consider now approximations with resampled particle clouds of different cardinalities, such as

$$\Xi_5^{res} = \{\{-1, 0.2\}, \{0, 0.2\}, \{0, 0.2\}, \{0, 0.2\}, \{1, 0.2\}\} \tag{2.52}$$

and

$$\begin{aligned} \Xi_{10}^{res} = \; & \{\{-1, 0.1\}, \{-1, 0.1\}, \{0, 0.1\}, \{0, 0.1\}, \{0, 0.1\}, \cdot \\ & \{0, 0.1\}, \{0, 0.1\}, \{0, 0.1\}, \{1, 0.1\}, \{1, 0.1\}\} \end{aligned} \tag{2.53}$$

The C2C distance between these resampled clouds is still 0 since the method in subsection 2.4.3 will bring both resampled clouds above to the same form Ξ_{10}^{res} of equal cardinalities and of equal weights.

The convergence in (2.30) was empirically explored using Monte Carlo simulations for the C2C computation on a restricted number N_0 of weighted (of the highest N_0 weights) particles in the cloud of cardinality $N \geq N_0$. Some of the results are presented in Section 2.6. The C2C computation on weighted particle clouds was found to be the method of choice due to the fractional multiassignment (finer than a 2-D) and the possibility to select the most significant N_0 particles based on their weights. The methods presented for resampled particle clouds use the 2-D assignment, which needs equal number of particles for a complete assignment, otherwise the unassigned particles would penalize the cost, as in (Hoffman and Mahler, 2004). One drawback of using the 2-D assignment on resampled clouds is that bringing both resampled clouds to the same cardinality (through their gcd as shown in 2.4.3) usually results in a very high number of particles.

Mahalanobis Distance Used as C2C Inner Distance

Instead of using the Euclidean distance as the *inner distance* in the C2C computation, the Mahalanobis distance, as defined below, could be used. This allows the usage of all particle state components (e.g. velocity as well, beside position) into the computation of the resulting C2C Mahalanobis Wasserstein distance. It implies the computation of the sample covariance of each cloud

$$\tilde{P} = \sum_{n=1}^{N} (\xi^n - \tilde{m}) (\xi^n - \tilde{m})', \tilde{m} = \sum_{n=1}^{N} \xi^n w^n \tag{2.54}$$

and based on it

$$d_{Mah}^{W} (\Xi_1^n, \Xi_2^n) = \min_{W} \sqrt{\sum_{i=1}^{M_1} \sum_{j=1}^{M_2} w_{ij} \left(\xi_{1,l_1}^i - \xi_{2,l_2}^j\right) \tilde{P}^{-1} \left(\xi_{1,l_1}^i - \xi_{2,l_2}^j\right)'}. \tag{2.55}$$

The Mahalanobis distance is defined between a random variable and a discrete value (with the random variable characterized by mean and covariance), whereas in (2.55) we are using the distance between two random variables (whose samples are the two sets of particles in the two clouds).

2.5 Cloud to Cloud Fusion

For two associated particle clouds, as corresponding to same target, Ξ_1 and Ξ_2, of unresampled particles with normalized weights (i.e., each cloud particles weights sum to one), three methods for computing the estimated fused state are presented below. *Method 1* – Derives the fused estimate as the direct combination of particles in both clouds. The method implicitly assumes the clouds have similar sample covariances.

$$\hat{x} = 0.5 \left(\sum_{i=1}^{M_1} \tilde{\xi}_1^i \tilde{w}_1^i + \sum_{j=1}^{M_2} \tilde{\xi}_2^j \tilde{w}_2^j \right). \tag{2.56}$$

Method 2 – Clouds are weighted by their sample covariances, following the information filter approach (Bar-Shalom and Li, 1995)

$$\hat{x} = \left(\mathbf{P}_1^{-1} + \mathbf{P}_2^{-1}\right)^{-1} \left(\mathbf{P}_1^{-1} \sum_{i=1}^{M_1} \tilde{\xi}_1^i \tilde{w}_1^i + \mathbf{P}_2^{-2} \sum_{j=1}^{M_2} \tilde{\xi}_2^j \tilde{w}_2^j \right), \tag{2.57}$$

where $\mathbf{P}_1$, $\mathbf{P}_2$, are sample covariances of clouds Ξ_1 and Ξ_2, computed through

$$\mathbf{P}_s = \sum_{i=1}^{M_s} \tilde{w}_s^i \left(\tilde{\xi}_s^i - \bar{\xi}_s\right) \left(\tilde{\xi}_s^i - \bar{\xi}_s\right)^T, \tag{2.58}$$

where $\bar{\xi}_s$ is the sample weighted mean. In this method independence of the estimation errors of both clouds is assumed, i.e., the cross-correlation due to common process noise is ignored.

Method 3 – Fusion is carried out making use of the estimated sample cross-covariance matrices between clouds, P_{12}, P_{21}. These are estimated based on the transportation matrix W between clouds, derived in the association step

$$\mathbf{P}_{12} = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} w_{ij} \left(\tilde{\xi}_1^i - \bar{\xi}_1\right) \left(\tilde{\xi}_2^j - \bar{\xi}_2\right)^T = \sum_{i=1}^{M_1} \left(\tilde{\xi}_1^i - \bar{\xi}_1\right) \sum_{j=1}^{M_2} \left(\tilde{\xi}_2^j - \bar{\xi}_2\right)^T w_{ij} \tag{2.59}$$

and used as

$$\mathbf{P} = \mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_{12} - \mathbf{P}_{21} \tag{2.60}$$

in the computation of fused estimate

$$\hat{x} = \left(\mathbf{P}_2 - \mathbf{P}_{21}\right)\mathbf{P}^{-1}\sum_{i=1}^{N_1}\tilde{\xi}_1^i\tilde{w}_1^i + \left(\mathbf{P}_1 - \mathbf{P}_{12}\right)\mathbf{P}^{-1}\sum_{j=1}^{N_2}\tilde{\xi}_2^j\tilde{w}_2^j. \tag{2.61}$$

2.6 Simulation

The architecture used in the simulation is a distributed one where fusion can happen at a rate lower than the sampling rates of the available sensors. The sampling of the sensors at the fusion times are supposed to be synchronous and no feedback (from the fused estimates to the local sensors estimates) is assumed.

2.6.1 PHD Particle Filter

The PHD definition is based on the theory of RFSs, whose statistics a.k.a. FISST define the targets in a scenario as an RFS (meta-target) and the set of observations as another RFS (meta-observation) (Mahler, 2003). The PHD was introduced in (Mahler, 2003) as the first order multitarget moment – the density function whose integral over a region is the expected number of targets in that region. Being defined in single-target state space, its value at a given point is the probability density function of target presence, therefore provides target localization. In the same work (Mahler, 2003) the author derives, within the Bayesian framework of FISST, the sequential estimation of the first order multitarget moment, also known as the PHD filter. The PHD filter has the ability to initiate tracks of newly born targets, spawning targets, as well as to terminate tracks for dead targets in a multitarget scenario of dynamic target cardinality, while also being able to account for clutter within the measurement set. A sequential Monte Carlo implementation of the PHD filter, known as the PHD particle filter, was derived in (Sidenbladh, 2003; Vo *et al.*, 2005; Zajic and Mahler, 2003).

2.6.2 Track Labeling

A PHD particle filter with the particle labeling capability was developed in (Lin *et al.*, 2006), based on the idea of running in parallel another estimator in order to preserve track identities (e.g. Kalman filter) and associate PHD peaks to current tracks identities and we proposed recently (Danu *et al.*, 2009b) a different labeling method. As a result, the PHD particles entering the fusion process, beside the state and weight, have the track label information, which groups them in labeled particle clouds. For example, after processing the measurements received at a given time k, the output of a given particle filter s (where $s = 1, \dots, S$, with S being the number of sensors) is the set of labeled particles corresponding to the estimated PHD

$$\hat{D}_{s,k|k} = \left\{ w_{k,s}^{l_s,i_l}, \xi_{k,s}^{l_s,i_l} \right\}, \; l_s = 1..\, \hat{N}_{s,k|k}, \; i_{l_s} = 1..\, L_{s,k}^{l_s}. \tag{2.62}$$

Here $\xi_{k,s}^{l_s,i_l}$ is a particle of track label l_s and index i, $w_{k,s}^{l_s,i_l}$ is its weight, $\hat{N}_{s,k|k}$ is the estimated number of targets, $l_s = 1..\, \hat{N}_{s,k|k}$ are all target labels (numbered here from 1 to $\hat{N}_{s,k|k}$), and $L_{s,k}^{l_s}$ is the number of particles in the cloud corresponding to track labeled l_s, all at time k and at sensor s. In further notation the time index k is dropped for simplification, as fusion is performed statically (at a common instance, thus association time being irrelevant for the presentation purpose).

One advantage of using labeled particles in the local particle filter sequential estimation is that new particles can be thrown into processing at any frame around every measurement (true or false) and no estimation of false measurements spatial distribution is needed for generating these new particles. The particles corresponding to false measurements are eliminated in the next frames, as they will not generate usually confirmed tracks, and particles of non-confirmed tracks are not propagated for more than two or three frames. (False tracks, when generated, are usually of small duration.) The advantage of using locally labeled particles for fusion is that the clustering at the fusion center is avoided and particles of tracks non-confirmed locally do not enter the fusion process, thus significantly decreasing the data association problem size and communication.

The simulation scenario contains two sensors $s = 1, 2$, each estimating through a labeled PHD filter the two targets with close, crossing and nonlinear trajectories, as shown in Fig.2.3. Over the second half of targets trajectories the combined sensor measurement errors are at times higher than the distance between the targets, therefore local estimation and association between sensor estimates are carried out under measurement origin uncertainty.

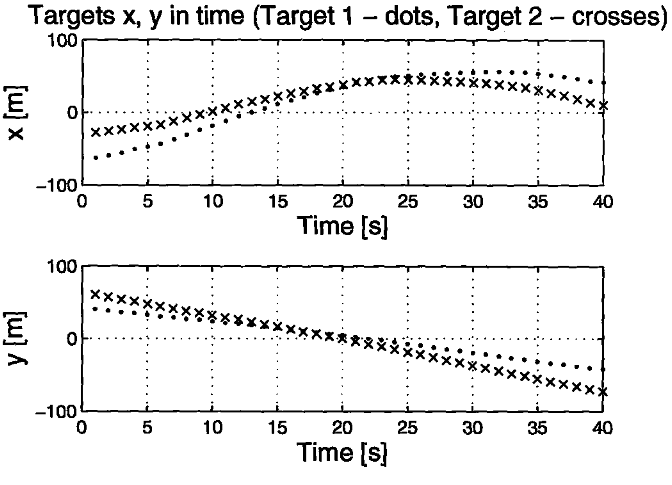


Figure 2.3: True targets trajectories in x and y over time.

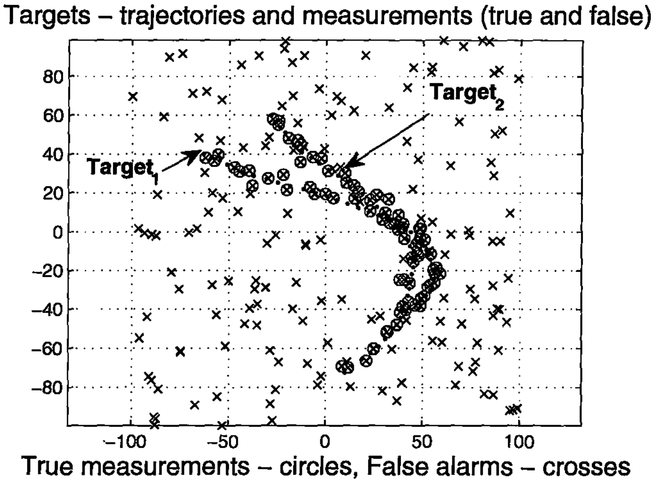


Figure 2.4: Targets measurements over a sample run.

Fig.2.4 shows the true measurements (circles) and false alarms (crosses) for a sample run of the simulation. The two local estimators used at the two sensors $s = 1, 2$, are PHD particle filters with track-valued (labeled) particle clouds. The equations and implementation of the track-labelled PHD particle filter follow the one in (Danu *et al.*, 2009a), here extended to two-dimensional tracking, briefly described below, with k denoting the time sample, and filter index s dropped. Particle states contain two-dimensional position and velocity in Cartesian coordinates, while measurements provide only position information.

Initialization

For $i = 1, \ldots, L_1$ sample position (x, y) of particles as $\tilde{\xi}^i_{1,(x,y)} \sim \mathcal{N}(Z_1, \sigma_Z)$, where Z_1 is the set of all initial measurements, and $\mathcal{N}$ stands for normal distribution. The velocity components of each particle are initialized uniformly distributed within maximum velocities allowed for a target $\tilde{\xi}^i_{1,(vx,vy)} \sim U(-v_{\max}, v_{\max})$, with $v_{max} = 6$ here. Initial weights are computed as $\tilde{w}^i_1 = \varsigma_1 / (L_1 \cdot n_1)$, where ς_1 is the estimated number of targets and n_1 is the number of initial measurements.

Prediction

For $i = 1, \ldots, L_{k-1}$ (confirmed particles at time $k-1$), sample $\tilde{\xi}_k^i \sim q\left(\cdot|\xi_{k-1}^i\right)$, where q is the proposal density, which translates into

$$\tilde{\xi}_k^i = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \xi_{k-1}^i + v_k, \tag{2.63}$$

with v_k denoting the process noise (with $\sigma_{x,y}^2 = 1.0$, $\sigma_{vx,vy}^2 = 0.25$) and the sampling time step $T = 1$s. Assuming no target spawning, weights are computed as $\tilde{w}_k^i = \tilde{w}_{k-1}^i \cdot e$, where $e = 0.95$ is the probability of target survival.

For $i = L_{k-1} + 1, \ldots, L_{k-1} + J_k$, newborn particles are sampled based on new measurements in the set Z_k. For this purpose Z_k is partitioned with respect to Υ_{k-1}, the set of tracks at time $k-1$, into

$$Z_{k,in} = \left\{ z_k^j \mid z_k^j \,\text{in validation region of a track in}\, \Upsilon_{k-1} \right\}$$

and $Z_{k,out} = \left\{ z_k^j \mid z_k^j \,\text{in no validation region of any track in}\, \Upsilon_{k-1} \right\}$. Particles position components are sampled for all measurements as $\tilde{\xi}_{k,(x,y)}^i \sim \mathcal{N}(Z_k, \sigma_z)$, where σ_z is the standard deviation of measurement position error. For measurements in $Z_{k,in}$ particles velocity components are sampled from $\tilde{\xi}_{k,(vx,vy)}^i \sim N\left(\hat{x}_{k-1}^n, \sigma_v\right)$, where $\hat{x}_{k-1}^n$ is the velocity estimate of track n at time $k-1$, in which region the measurement z was accepted. For measurements in $Z_{k,out}$ particles velocity components are sampled uniformly within $[-v_{x,max}, v_{x,max}]$ and $[-v_{y,max}, v_{y,max}]$. The weights of all newborn particles are computed as

$$\tilde{w}^i_{k|k-1} = \frac{b_k\left(\tilde{\xi}^i_k\right)}{J_k \cdot p_k\left(\tilde{\xi}^i_k|Z_k\right)} \tag{2.64}$$

where $b_k\left(\tilde{\xi}^i_k\right)$ is the intensity function of the point process (considered Poisson here) generating the target birth and $p_k\left(\tilde{\xi}^i_k|Z_k\right)$ the proposal density. By considering normal densities for b_k and p_k as in (Lin *et al.*, 2006), (Vo *et al.*, 2005), the weights are computed as I/J_k, where the constant intensity of the point process was taken as $I = 0.2$ (e.g. one target birth expected every 5th sampling time on average).

Update

For $i = 1, \ldots, L_k$, where $L_k = L_{k-1} + J_k$, the updated weights are computed as

$$\tilde{w}^i_k = \tilde{w}^i_{k|k-1} \cdot [1 - P_D + \sum_{z \in Z_{k,in}} \frac{\beta_1 P_D h(z|\xi^i_{k|k-1})}{\lambda \cdot c + \beta_1 C_k(z)} + \sum_{z \in Z_{k,out}} \frac{\beta_2 P_D h(z|\xi^i_{k|k-1})}{\lambda \cdot c + \beta_2 C_k(z)}] \tag{2.65}$$

where P_D is the probability of detection (considered unity here), h is the sensor measurement function, considered as normal with measurement error standard deviation 2.5 for each sensor in both x and y directions, $\lambda = 4$ is the average rate of clutter points per scan, c = 0.000025 is the uniform clutter density for the whole surveyed area, β_1=1.1, $\beta_2 = \beta_1/25$ are design parameters as in (Lin *et al.*, 2006) and

$$C_k(z) = \sum_{i=1}^{L_k} P_D h(z|\xi^i_{k|k-1}) \tilde{w}^i_{k|k-1}. \tag{2.66}$$

Track labels are obtained by using a shadow Kalman filter and the resolution cell technique (Lin *et al.*, 2006). The details of the particle labeling implementation for the 2-D PHD filter were derived and presented in (Danu *et al.*, 2009b). The unresampled particle clouds obtained at the local particle filters are considered for fusion, and their association costs are computed as introduced in (2.18)-(2.20), translated into a linear program using definitions in (2.43)–(2.45). The resulting linear program was solved using the LIPSOL package (Zhang, 1999).

2.6.3 Simulation Results

Average estimated errors and Root Mean Squared Errors (RMSE) (Bar-Shalom *et al.*, 2001) computed over time for 100 Monte Carlo runs are shown in Fig.2.5–2.6 and are summarized in Tables 2.1–2.2 for single filter and fused results. The comparison between the estimates RMSE obtained using the C2C distance and T2T test is captured also in Fig.2.5–2.6. The fusion Method1 was used in all simulation cases, therefore the results show the difference obtained through using different association costs. The C2C cost using Wasserstein distance of orders 1 and 2 on weighted and resampled clouds were used for comparison. The fused estimates MSE and RMSE over the MC runs are labeled in Fig.2.5–2.6 with Wd1 and Wd2 for C2C of orders 1 and 2, respectively, obtained on weighted clouds and with Wd1_res and Wd2_res for C2C of orders 1 and 2, respectively, obtained on resampled clouds.

Table 2.1: Position RMSE for single filters

	Target 1	Target 2
RMSE Sensor 1	3.5109	3.8981
RMSE Sensor 2	3.5624	3.8648

Table 2.2: Mean Position RMSE for Methods 1 and 2 of fusion using C2C cost and T2T test

	Target 1	Target 2
Method 1 - C2C with d_1^{WE}	2.59	2.75
Method 1 - C2C with d_2^{WE}	2.63	2.79
Method 2 - C2C with d_1^{WE}	2.64	2.91
Method 2 - C2C with d_2^{WE}	2.73	2.97
T2T test	2.96	3.23

2.6.4 C2C vs. Euclidean Distance for Gaussian Distributed Clouds

In order to assess the proposed association cost on clouds of different distributions, Monte Carlo simulations were performed for estimating the association cost of two clouds, normally distributed (corresponding to the unresampled case), with different known standard deviations. The number of particles considered was $M = 400$ on both clouds. The costs obtained in 100 runs using (2.18) are compared with the distribution of the distance between the clouds centers estimated on the same runs using the weighted sum of particles

$$d_{12} = \left\| \sum_{i=1}^{M} w_1^i \xi_1^i - \sum_{j=1}^{M} w_2^j \xi_2^j \right\| . \tag{2.67}$$

Two Gaussian particle clouds with particle states capturing the $2D$ position were considered, with distance between their centers of 2.594, one with constant standard deviation $\sigma_1 = 1.5$ and the second one with seven different standard deviations, smaller and higher compared with the one of the first one: $\sigma_2 = \{0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5\}$. The results obtained are summarized in Table 2.3, where it can be seen first that the C2C distance, computed on the random probability measures represented by the

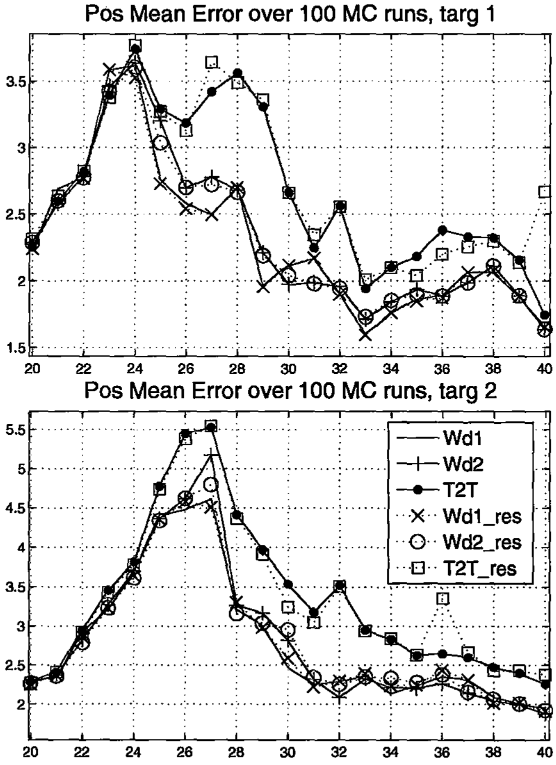


Figure 2.5: Position Mean Errors of fused estimates obtained using C2C distance and T2T test.

clouds follows closely the theoretical Wasserstein distance between the underlying Gaussian distributions, as given by (2.39). Compared with the sample Euclidean distance computed using (2.67), which is almost insensitive to the variance of the clouds, as only the estimate first order moment enters (2.67), the C2C cost computed based on (2.18) increases with the difference between the clouds second order moment (variance). The minimum C2C cost is obtained for equal variances of the clouds. While these simulations were using Gaussian clouds, however the cost in (2.18) is not making any assumption on normality, therefore differences in higher order moments of the clouds are expected to be penalized by the C2C cost as well for non-Gaussian clouds. Also the results in Table 2.3 show that a good C2C convergence is obtained

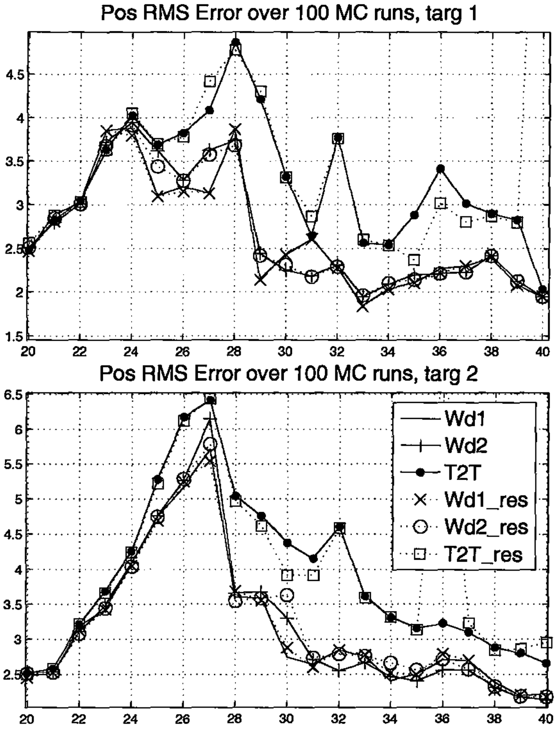


Figure 2.6: Position RMSE of fused estimates obtained using C2C distance and T2T test.

using 400 particles per cloud.

2.7 Conclusions

A method for the association and fusion of target posterior densities estimates obtained using particle filters was proposed. C2C association costs, computed at the particle level, depending on particles types were derived and the convergence of the cost on particle clouds was proved theoretically and tested through simulations. Several fusion methods were presented. Results obtained on a simulated multitarget scenario using the proposed association cost and fusion method were analyzed and

Table 2.3: C2C costs compared with sample Ed and theoretical T2T.

Cloud 1 std	$\sigma_1 = 1.5$						
Cloud 2 std	σ_2=0.5	σ_2=1.0	σ_2=1.5	σ_2=2.0	σ_2=2.5	σ_2=3.0	σ_2=3.5
Wd theor.	2.780	2.642	2.594	2.642	2.780	2.997	3.276
C2C mean	2.798	2.649	2.610	2.661	2.851	3.103	3.502
C2C std.	0.101	0.111	0.130	0.155	0.180	0.170	0.193
Ed mean	2.600	2.598	2.604	2.604	2.627	2.606	2.614
Ed std.	0.089	0.089	0.108	0.127	0.146	0.165	0.180
T2T theor.	2.692	2.071	1.496	1.077	0.792	0.598	0.464

showed improvement compared with the classical T2T test when applied on particle filter estimates. The fusion method introduced, at particle level, combines the whole information contained in the estimated targets posterior densities, not only the tracks states first order moments, therefore being applicable to highly non-Gaussian estimates and nonlinear estimators.

Chapter 3

Track-to-Track Fusion Using Prior Associations

In single-frame track-to-track association the identities of the fused tracks are not preserved over time. The fused track identities can be tagged based on a given sensor local tracks identities at each frame. Due to possible local tracks swapping (switching of a track from an estimated target to another estimated target, under measurement uncertainty conditions), this type of tagging based on a single sensor identities is not reliable. This chapter introduces a track-to-track association method that links the histories of fused tracks over time. The association method shows also a reduction of track swapping at the fusion center level in the presence of track swap at the local sensors level. The track association method uses the previous association hypotheses as priors in a multiple-hypothesis association chain. The fused tracks continuity is achieved through their prediction from one fusion time frame to the next. The predicted fused tracks of each selected association hypothesis at a given time frame participate in the tracks association of the next frame. The cross-correlation between local tracks at the association frame is considered in the association cost. The dependence between associations over consecutive frames is taken into account through the cross-correlation between the predicted fused tracks of previous frame to the time frame of current association. By taking into account the fused tracks identities when evaluating the fused tracks estimation errors (e.g. the errors of a fused track over all frames are computed with respect to the same true target), the procedure improves also the fused track performance estimation. The method and implementation proposed is intended to identify the histories of two or more tracks at the fusion center, and to improve the track-to-track association. Upon association, the track fusion method employed may use the previous fused track estimation, i.e. perform fusion with memory, or not, i.e. perform fusion without memory. Also the method can be used in conjunction with no, partial or full feedback from the fusion center to the local sensors. The feedback can be at the track estimation level, track identity level or both.

3.1 Introduction

Track-to-track association is an essential component of a distributed tracking system that performs track-to-track fusion. It generates and selects valid hypotheses of local tracks subsets as pertaining to the same targets. Its computational intensity increases exponentially with the number of sensors.

In the literature there are described several methods, referring both to the computation of the costs (likelihoods) of common origin of a subset of local tracks (Bar-Shalom *et al.*, 2007), (Bar-Shalom and Chen, 2006), (Kaplan and Blair, 2004), (You and Jingwei, 2006) as well as methods of selecting the best, or m-best, assignment hypotheses (Deb *et al.*, 1997), (Poore and Robertson-III, 1997), (Popp *et al.*, 2001). The cost of common origin of a subset of tracks is usually computed on a single frame basis (Bar-Shalom *et al.*, 2007), (Bar-Shalom and Chen, 2006), (Kaplan and Blair, 2004) and therefore the resulting fused track does not have a continuity foundation from one fusion frame to the next. At the local sensor level each track has a continuity over time frames through the track label and its dynamic model. However when the local sensors estimate under measurement uncertainty conditions local tracks may switch from following one target to following another target. This erroneous change of a local track to start estimating a different true target is termed *track swap.* This may prevent relying on the local tracks identities (IDs) at the fusion center from one association frame to the other. A method of computing track-to-track association costs over several local frames was proposed in (You and Jingwei, 2006), for two sensors, however relying solely on the local tracks identities, assuming no track swaps at the local tracker level and not taking into account the cross-correlation between local estimates over multiple frames (Bar-Shalom and Chen, 2008). The exact and approximate methods of computing the cross-correlation between local estimates over multiple frames was introduced in (Tian and Bar-Shalom, 2009). As shown therein, using multiple local frames in the association does not necessarily increase the power of test in selecting the association hypotheses. The sliding window over local tracks approach is valid only for the case that no track switch happens at the local trackers level for the windowing period, which cannot be guaranteed. The track-to-track association and further fused tracks identification over time becomes challenging when the local tracks of several targets are very close, within distances of the order of tracks uncertainties. This situation is usually linked to local measurement origin uncertainty that may generate sporadic local tracks swap. The track association method introduced in this chapter uses as prior at the current association frame the fused tracks and association obtained at the previous fusion frame. As tracks association is performed at each fusion frame, several cumulative association hypotheses are kept from the previous fusion frame as priors for assessing the current frame association hypotheses. The continuity of associations from one frame to the next is achieved through the predicted fused tracks estimates and their identities (labels). Each prior association hypothesis results in a different prior fused tracks set, of which predictions enter the current frame track-to-track association. An *(S+1)*-D association, with S equal to the number of sensors, is required in this case in the hypothesis selection. A limited number of distinct hypotheses are chosen to be used in the subsequent fusion step. The predicted fused tracks in the resulting assignment solutions are not used in the subsequent fusion, they are only used for linking the identity information from one fusion frame to the next, through weighting of the likelihoods of common origin for local tracks. The track-to-track association method relies on generating association hypotheses at each fusion frame and linking them through time in a fused track history. Therefore the method is intended to be used also in the identification of fused tracks histories over time. The cross-correlation between the local sensor tracks and also between predicted fused and local tracks is estimated with exact and approximate methods developed in (Chen *et al.*, 2003) and (Tian and Bar-Shalom, 2009). The cross-correlation between predicted fused tracks and current local tracks accounts for the dependency of estimation errors between fusion frames. The association scheme proposed can be combined with different fusion methods used in the subsequent fusion estimate step. Beside the classical one-scan fusion method introduced in (Bar-Shalom and Campo, 1986), several fusion methods were recently developed in (Tian and Bar-Shalom, 2009), (Tian and Bar-Shalom, 2010) that use or do not use previous frames and feedback in computing the fused tracks estimates at the current frame. As shown in (Tian and Bar-Shalom, 2009), the feedback usage in the fusion estimate step changes the cross-correlation present between between local track estimates and fused tracks and needs to be accounted for in the association step. The chapter is organized as follows: section 3.2 provides a description of the track-to-track association problem, section 3.3 details the usage of prior associations assignment, section 3.4 describes the proposed method of generating and assessing the hypotheses, as well as a sample implementation. Simulations and results are presented in section 3.5 and conclusions in section 3.6.

3.2 Problem Description

We consider $i = 1, \ldots, S$ local sensors tracking $j = 1, \ldots, N_t$ targets and use the notation $\hat{x}_i^{j_i(k)}(k)$ for the estimated local tracks states. The number of targets are assumed constant over time. The extension to a variable number of targets is straightforward once the target deaths and target births probabilities are considered in evaluating the association hypotheses costs. The sample time frames are represented by the index k and local sensors are considered synchronized for the brevity of the exposition. The identity index (label) of the local track that estimates a true target, i.e., j, at sensor i at time k is represented by the index $j_i(k)$. The notation with time-dependent index $j_i(k)$ is used as the ID of the local track that follows a true target at sensor i might change its value over time. This happens when the local track with ID j_i switches to estimating a different target at a subsequent fusion time. The track swap may happen when trackers run under measurement origin uncertainty (Bar-Shalom and Li, 1995). It is caused by measurement-to-track misassociation(s) used in the update of a track state estimate. The local track swap affects the track-to-track association part of track fusion. It also precludes the fusion center of labeling fused track identities based solely on a single sensor tracks identities. The problem of preserving the fused tracks identities over time and solution proposed is shown in Fig.3.1. In the example used, two sensors, $i = 1, 2$, are tracking two closely spaced targets $tg = 1, 2$ between the fusion time frames l and k. Fig.3.1-a shows the ideal case of no local track swap. Both sensors preserve the correct IDs of the local tracks over all sample times between l and k. Tracks $\hat{x}_1^1(\cdot)$ and $\hat{x}_1^2(\cdot)$ are the sensor 1 estimates of targets 1 and 2, respectively. Tracks $\hat{x}_2^1(\cdot)$ and $\hat{x}_2^2(\cdot)$ are sensors 2 estimates of targets 1 and 2, respectively. In this ideal case the identities of the associated and subsequently fused tracks $\hat{x}_F^1(\cdot)$, $\hat{x}_F^2(\cdot)$ can be maintained even with single-frame track association. The fused tracks identities are correctly inferred from the local IDs of any sensor.

In Fig.3.1-b a track swap occurs at sensor $s = 2$ between the fusion times l and k. The resulting fused track identities cannot be correctly preserved based on the IDs of any local sensor. For example, using labels of sensor 2, would indicate that $\hat{x}_F^1(l)$ and $\hat{x}_F^X(k)$ represent different targets, i.e. 1 respectively 2. The solution using the prior fused tracks in association is shown in Fig.3.1-c. If the predicted fused tracks $\hat{x}_F^1(k|l)$, $\hat{x}_F^2(k|l)$ are used in the track association, the weights for the correct label of associated tracks are increased. For the case presented in Fig.3.1-c the best association cost will result to be the one of tracks $[\hat{x}_1^1(k)\ \hat{x}_2^2(k)\ \hat{x}_F^1(k|l)]$ representing target 1 and $[\hat{x}_1^2(k|l)$ $\hat{x}_2^1(k|l)\ \hat{x}_F^2(k|l)]$ representing target 2. Based on a majority vote (Sinha *et al.*, 2008) the indices of resulting fused tracks at k will represent the correct ones of the true targets.

The details of the problem are presented next. At each fusion time k, the fusion center receives the local tracks estimates $\hat{x}_i^{j_i(k)}(k)$, $i = 1, \ldots, S$, with $j_i(k) = 1, \ldots, N_t$

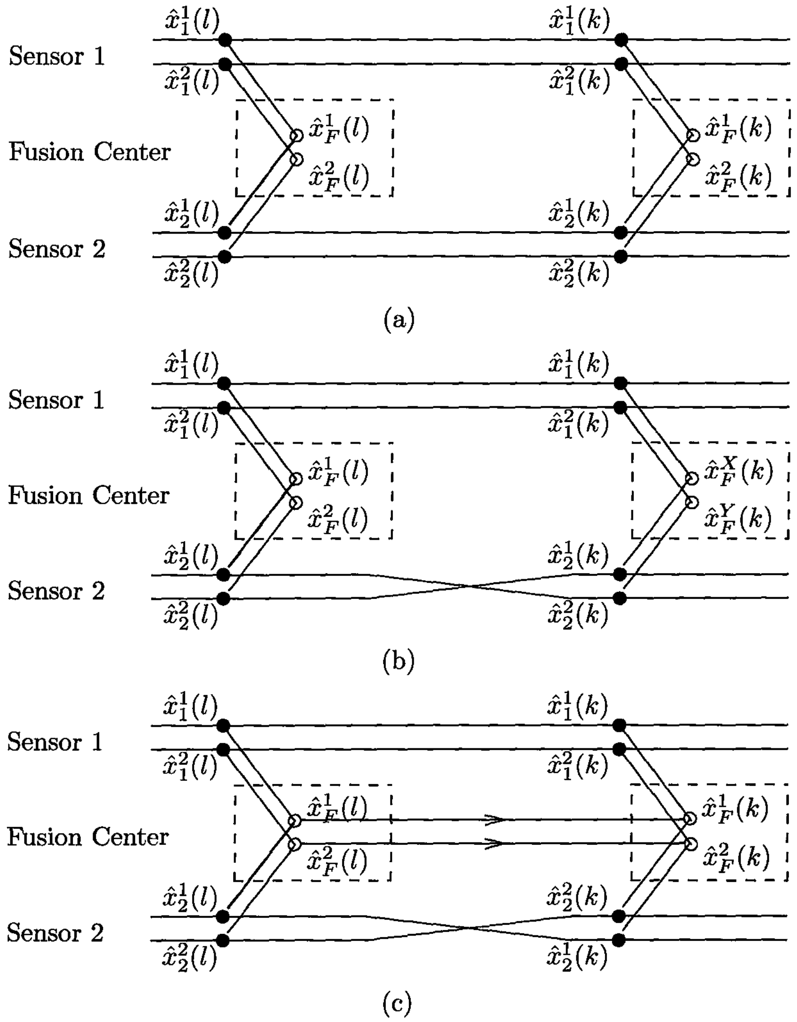


Figure 3.1: Track-to-track association and fused track identities in the presence/absence of local track swap for S=2 sensors tracking 2 targets. (a) Single frame track-to-track association (2-D) without local track swap and fused track identities preserved. (b) Single frame track-to-track association (2-D) with local track swap at sensor $s = 2$ and uncertain fused track identities. (c) Track-to-track association using prior (3-D) with fused track identities retrieved upon local swap at sensor $s = 2$.

and performs the track-to-track association prior to the fusion of the track estimates. We define the binary track-to-track assignment variable $\chi_{j_1,j_2,\ldots,j_S}$ representing the true/false common origin of the tracks at time frame k as (Pattipati *et al.*, 2000)

$$\chi_{j_1,j_2,\ldots,j_S} = \begin{cases} 1, & \text{if } j_1, j_2, \ldots, j_S \text{ denote the same target} \\ 0, & \text{otherwise} \end{cases} \tag{3.1}$$

The track-to-track association problem at a single fusion frame is the S-D generalized association problem captured through the constrained minimization (Deb *et al.*, 1997), (Pattipati *et al.*, 2000), (Poore and Robertson-III, 1997):

$$\min_{\chi} \sum_{j_1} \sum_{j_2} \cdots \sum_{j_S} c_{j_1,j_2,\ldots,j_S} \chi_{j_1,j_2,\ldots,j_S} \tag{3.2}$$

subject to

$$\sum_{j_1} \cdots \sum_{j_{i-1}} \sum_{j_{i+1}} \cdots \sum_{j_S} \chi_{j_1,j_2,\ldots,j_S} = 1, \quad i = 1, \ldots, S, \quad j_i = 1, \ldots N_t \tag{3.3}$$

where $c_{j_1,j_2,\ldots,j_S}$ is the cost of common origin for the local S-tuple. For a single frame association the cost is computed as

$$c_{j_1,j_2,\ldots,j_S} = -\ln \lambda_{j_1,j_2,\ldots,j_S} \tag{3.4}$$

where $\lambda_{j_1,j_2,\ldots,j_S}$ is the generalized likelihood ratio of common origin of tracks hypothesis $\mathcal{H}_{j_1,j_2,\ldots,j_S}$ (Bar-Shalom and Chen, 2006). The likelihood ratio is computed as the ratio of the common origin of tracks likelihood function (assumed joint normal density) to the diffuse *pdf* of track density, $1/\mathcal{V}$ (Bar-Shalom and Chen, 2006),

(Bar-Shalom *et al.*, 2001)

$$\lambda_{j_1,j_2,\ldots,j_S} = \mathcal{V}^{S-1}\mathcal{N}[\hat{\mathbf{x}}_{1,S}; 0, \mathbf{P}_{1,S}] \tag{3.5}$$

where

$$\hat{\mathbf{x}}_{1,S} = [\hat{x}_2^{j_2} - \hat{x}_1^{j_1},\ \hat{x}_3^{j_3} - \hat{x}_1^{j_1},\ \ldots\ \hat{x}_S^{j_S} - \hat{x}_1^{j_1}]' \tag{3.6}$$

and $\mathbf{P}_{1,S}$ is the covariance matrix of $\hat{\mathbf{x}}_{1,S}$, with block elements defined as (Bar-Shalom *et al.*, 2001)

$$(\mathbf{P}_{1,S})_{m-1,n-1} = E[(\hat{x}_m^{j_m} - \hat{x}_1^{j_1})(\hat{x}_n^{j_n} - \hat{x}_1^{j_1})], \quad m, n = 2, ..., S \tag{3.7}$$

The above procedure performs the track-to-track association separately from fusion frame to fusion frame. In (You and Jingwei, 2006)[1], for two sensors, the likelihood ratio of common origin of tracks (j_1, j_2) is developed over several frames. The sequential association is performed as

$$\lambda_{j_1j_2}(k) = \lambda_{j_1j_2}(l) + \hat{t}'_{j_1j_2}(k)\mathbf{P}^{-1}_{j_1j_2}(k)\hat{t}_{j_1j_2}(k) \tag{3.8}$$

with $\hat{t}_{j_1j_2}(k) = \hat{x}_1^{j_1}(k) - \hat{x}_2^{j_2}(k)$ and l being the previous fusion frame. The sequential (cumulative) likelihood ratio above was derived by assuming the conditional $\mathcal{H}_{j_1j_2}$ hypothesis in the computation of the joint probability density function of $\hat{t}^k_{j_1j_2} = \{\hat{t}_{j_1j_2}(n) : n = 1, 2, \ldots, k\}$. This implies the $\mathcal{H}_{j_1j_2}$ hypothesis holds true for the whole time interval for which the summation is done, which cannot be guaranteed for

[1] As shown in (Bar-Shalom and Chen, 2008) and (Tian and Bar-Shalom, 2009), in the paper (You and Jingwei, 2006) the cross-correlation between estimates over different time frames is ignored, which leads to erroneous results.

close tracks. Also the association costs, i.e. logarithms of hypotheses likelihoods, at different fusion frames are assumed independent, disregarding the cross-correlation present between different time frames. In the case of local trackers running under measurements uncertainty condition, the probability of a track swap between fusion times l and k at a local sensor can be expressed as the probability of local measurements misassociation for a given track within the time interval $(l, k]$.

The probability of a local track at a given fusion time k to represent the state estimate of a given true target $J \in [1 \dots N_t]$ is expressed in terms of tracks probabilities at previous fusion time l. It is the sum of the probability of the track representing the same target J at time l without local track swap in the interval $(l, k]$ and the sum of probabilities of the joint events of estimating a different target I at time l and making an irreversible swap (I to J) within the time interval $(l, k]$. This translates in the measurement association of the track with the measurement of target J at time k

$$\begin{aligned} P\left(\hat{\mathbf{x}}_{i,k}^{j_i(k)} \equiv \hat{\mathbf{x}}_{i,k}^{J}\right) = P\left(\hat{\mathbf{x}}_{i,l}^{j_i(l)} \equiv \hat{\mathbf{x}}_{i,l}^{J},\ \theta_k(\hat{\mathbf{x}}_{i,k-1}^{j_i(k-1)}, \mathbf{z}_{i,k}^{J})\right) + \\ \sum_{I=1\dots N_t, I \neq J} P\left(\hat{\mathbf{x}}_{i,k}^{j_i(l)} \equiv \hat{\mathbf{x}}_{i,l}^{I},\ \theta_k(\hat{\mathbf{x}}_{i,k-1}^{j_i(k-1)}, \mathbf{z}_{i,k}^{J})\right) \end{aligned} \tag{3.9}$$

where the event $\theta_k(\hat{\mathbf{x}}_{i,k-1}^{j_i(k-1)}, \mathbf{z}_{i,k}^{J})$ means the association of measurement $\mathbf{z}_{i,k}^{J}$ to the track $\hat{\mathbf{x}}_{i,k-1}^{j_i(k-1)}$ in its update. The $\equiv$ sign is used to show the labels of two tracks match. The above can be written only at track label level as

$$\begin{aligned} P\left(j_i(k) \equiv J\right) = P\left(j_i(l) \equiv J,\ \theta_k(j_i(k-1), \mathbf{z}_{i,k}^{J})\right) + \\ \sum_{I=1\dots N_t, I \neq J} P\left(j_i(l) \equiv I,\ \theta_k(\hat{\mathbf{x}}_{i,k-1}^{j_i(k-1)}, \mathbf{z}_{i,k}^{J})\right) \end{aligned} \tag{3.10}$$

If the local sensors provide above information for the fusion center, these probabilities could be taken into account in computing the association costs through a confusion matrix type information (Bar-Shalom *et al.*, 2005). However, as they are not available, a method using the track-to-track association information in the prior frames is proposed to mitigate the effect of local swap.

3.3 Utilization of Prior Association

In finding the best solutions for track-to-track association at a given fusion frame, the prior best association hypotheses (solutions) found at the previous frame are considered informative. The method introduces a multiple hypotheses association (MHA) approach, at each fusion frame generating sets of best association hypotheses and predicting their fused tracks for use in the generation of the best association hypotheses at the next frame. The association cost function of the various hypotheses can be implemented using local tracks oriented, or fused tracks oriented approaches. This classification is very close to the one used for single-sensor MHT (multiple-hypothesis tracking) methods in (Bar-Shalom *et al.*, 2007). We call fused tracks oriented approach the method of which costs are based on the likelihood scores of fused tracks. We call local tracks oriented approach the method of which costs are based on probabilities of local tracks association hypotheses. Using a similar approach as in (Reid, 1979) for single-sensor tracking, a local tracks oriented approach is pursued within the framework of a fusion center capable of inferring the number of targets. The MHA approach consists of using at fusion frame k the N-best solutions of the previous fusion frame, l, predicted to the current frame. N hypotheses of tracks association are generated at current frame k. The predictions of each set of fused tracks are used at the next fusion frame to generate another set of hypotheses, through association with local tracks estimates. The link between track-to-track association frames is carried out through the predictions of the fused tracks at a previous frame in the current frame association. Therefore for S sensors, at each frame it is performed an $(S+1)$-D association instead of an S-D one. The fused tracks are predicted (from fusion times l to k) when the local tracks are available (at time k). The process noise information (available from the local trackers) between frames l and k is used in the estimation of the likelihood ratios of common origin for the set of local tracks and predicted fused tracks. The storage of several hypotheses, with their corresponding fused tracks at every fusion time is done in the attempt to find the hypothesis that results in best costs along multiple fusion frames. A pruning of the resulting hypotheses is performed at each fusion step.

3.4 Method Description and Implementation

3.4.1 Theoretical method description

We consider two consecutive fusion frames taking place at local sample times l and k. At the fusion frame k we have available the set of N (best) cumulative hypotheses from previous fusion time l. These hypotheses are called cumulative as they are the result of sequential processing of all previous fusion frames. Each cumulative hypothesis Θ_p^l at time l is represented by the set of fused tracks resulting from the corresponding association hypothesis at time l. The set of N cumulative hypotheses available from time l is therefore $\Theta^l = \{\Theta_p^l\}$, $p = 1, \ldots, N$, where the index p is used to denote *prior*. These cumulative hypotheses are used as priors in the generation of the track-to-track association hypotheses set at current frame k, $\{\mathcal{H}_n(k)\}$, $n = 1, \ldots, M$. Here $\mathcal{H}_n(k)$ is a track-to-track association hypothesis selected at frame k, where the notation n

is used to denote *new* hypotheses. The recursion of the probability of a cumulative hypothesis can be written using Bayes rule, and an approach analogous to the one in (Reid, 1979) as

$$P\left(\Theta_n^l , \mathcal{H}_n(k) \mid \left[\hat{\mathbf{X}}_S(k)\right]\right) = \frac{1}{c} P\left(\left[\hat{\mathbf{X}}_S(k)\right] \mid \Theta_p^l , \mathcal{H}_n(k)\right) \cdot P\left(\mathcal{H}_n(k) \mid \Theta_p^l\right) \cdot P\left(\Theta_p^l\right), \tag{3.11}$$

where $\left[\hat{\mathbf{X}}_S(k)\right]$ is the set of local track estimates available at time k from all the S sensors and c is a constant used for normalization and equal to the sum of all right hand side (RHS) terms over all $p = 1, \ldots, N$ and $n = 1, \ldots, M$ hypotheses. A current cumulative hypothesis actually is the joint event of a prior cumulative hypothesis and a current association hypothesis, therefore (3.11) is the recursion for the cumulative hypothesis probability. The set of fused tracks estimates with their uncertainties (covariance matrices) at a current frame resulting from a given hypothesis is considered as a sufficient statistic for the hypothesis that generated them, therefore we can substitute them into (3.11) and obtain the recursion for the probability of a given set of fused tracks under a given hypothesis as

$$P\left(\Phi_n^k\right) = \frac{1}{c} P\left(\left[\hat{\mathbf{X}}_S(k)\right] \mid \Phi_p^l , \mathcal{H}_n(k)\right) \cdot P\left(\mathcal{H}_n(k) \mid \Phi_p^l\right) \cdot P\left(\Phi_p^l\right) \tag{3.12}$$

where Φ_n^k is the new set of fused tracks generated at frame k under the association hypothesis $\mathcal{H}_n(k)$ and resulting from the parent set of fused tracks Φ_p^l. The first probability in the RHS of (3.12) is the likelihood of the set of local tracks at frame k conditioned on the prior fusion hypothesis and the current association hypothesis. This can be estimated (for the purpose of hypotheses comparison) using $\Phi_p^{k|l}$, the set of the predicted states of the fused tracks in Φ_p^l (assuming these predictions are sufficient statistics for the fused tracks estimates) and it can be expanded in

$$P(\left[\hat{\mathbf{X}}_S(k)\right] \mid \Phi_p^{k|l}, \mathcal{H}_n(k)) = \frac{1}{c_1} P\left(\mathcal{H}_n(k) \mid \left[\hat{\mathbf{X}}_S(k), \Phi_p^{k|l}\right]\right) \cdot P\left([\hat{\mathbf{X}}_S(k)] \mid \Phi_p^{k|l}\right) \quad (3.13)$$

where the first is the likelihood of the association hypothesis $\mathcal{H}_n(k)$ given the local tracks and predicted fused tracks. Following (Bar-Shalom and Chen, 2006), the log-likelihood (ratio) of this event can be evaluated as the sum of the log-likelihood ratios of common origin of tracks with respect to each target at time k. In other words the log-likelihood of this event is evaluated as the maximum sum of the log-likelihood ratios of common origin of the tracks at l, over the subsets of local tracks in the current hypothesis. This is the negative of the sum of resulting costs of each of the branches corresponding to a given target in the resulting $(S+1)$-D assignment tree (S sets of local tracks from local sensors and one set of predicted fused tracks):

$$\min_{\chi_{j_1 j_2 \cdots j_S j_F}} \sum_{j_1} \sum_{j_2} \cdots \sum_{j_S} \sum_{j_F} c_{j_1 j_2 \ldots j_S j_F} \chi_{j_1 j_2 \ldots j_S j_F}(k) \quad (3.14)$$

subject to

$$\sum_{j_1} \cdots \sum_{j_{i-1}} \sum_{j_{i+1}} \cdots \sum_{j_S} \sum_{j_F} \chi_{j_1 j_2 \ldots j_S j_F}(k) = 1, \quad \forall\, i = 1..S, F, \; j_i \in 1...N_t, \; j_F \in 1...N_t \quad (3.15)$$

For each target $j = 1..N_t$ a cost (corresponding to the best minimum feasible solution) of common origin of local tracks and one fusion tracks will be found as the result of S-D association. The cost of the association hypothesis $\mathcal{H}_n(k)$ is the sum of the costs, as given by (3.14), for all targets, where the cost corresponding to a target is given by the negative log-likelihood of the common origin of a subset of tracks (Bar-Shalom and Chen, 2006)

$$c_{j_1 j_2 \cdots j_S j_F}(k) = -\ln \lambda_{j_1 j_2 \cdots j_S j_F}(k) \tag{3.16}$$

with

$$\lambda_{j_1 j_2 \cdots j_S j_F}(k) = \frac{\Lambda(\chi_{j_1 j_2 \cdots j_S j_F}(k))}{\mathcal{V}^{-(S+1)}} = \frac{\frac{1}{\mathcal{V}}\mathcal{N}(\hat{\mathbf{x}}_{1:S,F}(k); \mathbf{0}, \mathbf{P}_{1:S,F}(k))}{\mathcal{V}^{-(S+1)}} \tag{3.17}$$

and therefore

$$\lambda_{j_1 j_2 \cdots j_S j_F}(k) == \mathcal{V}^S \mathcal{N}\left(\hat{\mathbf{x}}_{1:S,F}(k); \mathbf{0}, \mathbf{P}_{1:S,F}\right) \tag{3.18}$$

The assumption of joint normal distribution for the set of local tracks and predicted fused track, all conditioned on the ones of a local sensor, i.e. first one, has been made above. The covariance matrix $\mathbf{P}_{1:S,F}(k)$ above is estimated as in (Bar-Shalom and Chen, 2006), (Chen *et al.*, 2003) and represents the covariance of the vector $\hat{\mathbf{x}}_{1:S,F}(k)$ below

$$\hat{\mathbf{x}}_{1;S,F}(k) = \left[\hat{\mathbf{x}}_2^{j_2}(k) - \hat{\mathbf{x}}_1^{j_1}(k); \quad , \ldots, \quad \hat{\mathbf{x}}_S^{j_S}(k) - \hat{\mathbf{x}}_1^{j_1}(k); \quad \hat{\mathbf{x}}_F^{j_F}(k|l) - \hat{\mathbf{x}}_1^{j_1}(k)\right]^T. \tag{3.19}$$

Here $\hat{\mathbf{x}}_i^{j_i}(k)$ is the track estimate of sensor i at frame k to be associated under hypothesis $\mathcal{H}_{j_1 j_2 \cdots j_S j F}(k)$ with predicted fused track $\hat{\mathbf{x}}_i^{j_F}(k \mid l)$. The $\hat{\mathbf{x}}_F^{j_F}(k \mid l)$ is the prediction of fused track $\hat{\mathbf{x}}_i^{j_F}(l)$ from frame l to k. The reference track $\hat{\mathbf{x}}_1^{j_1}(k)$ in (3.19) can be replaced by the predicted fused one, obtaining

$$\hat{\mathbf{x}}_{1:S;F}(k) = \left[\hat{\mathbf{x}}_1^{j_1}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l); \quad \hat{\mathbf{x}}_2^{j_2}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\ , \ldots, \quad \hat{\mathbf{x}}_S^{j_S}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\right]^T. \tag{3.20}$$

The computation of the covariance matrix $\mathbf{P}_{1:S;F}(k)$ of the stacked vector $\hat{\mathbf{x}}_{1:S;F}(k)$ is described next.

For fusion *without feedback* (Bar-Shalom, 2006) the diagonal elements of the covariance matrix of $\hat{\mathbf{x}}_{1:S;F}(k)$ are

$$\begin{aligned} p_{mm} &= Cov\left[\hat{\mathbf{x}}_m^{j_m}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\right] \\ &= E\left[(\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l)) \cdot (\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l))'\right] \\ &= \mathbf{P}_m^{j_m}(k) + \mathbf{P}_F^{j_F}(k|l) - \mathbf{P}_{m,F}^{j_m,j_F}(k;k|l) - \mathbf{P}_{F,m}^{j_F,j_m}(k|l;k)' \end{aligned} \tag{3.21}$$

with $m = 1, \ldots S$. The notation $\mathbf{P}_{m,n}^{j_m,j_n}(k;k|l)$ is used for the cross-covariance between $\tilde{\mathbf{x}}_m^{j_m}(k)$ and $\tilde{\mathbf{x}}_n^{j_n}(k|l)$. The non-diagonal elements of the covariance matrix above are

$$\begin{aligned} p_{mn} &= Cov\left[\hat{\mathbf{x}}_m^{j_m}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l), \hat{\mathbf{x}}_n^{j_n}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\right] \\ &= E\left[(\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l)) \cdot (\tilde{\mathbf{x}}_n^{j_n}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l))'\right] \\ &= \mathbf{P}_{m,n}^{j_m,j_n}(k) + \mathbf{P}_F^{j_F}(k|l) - \mathbf{P}_{m,F}^{j_m,j_F}(k;k|l) - \mathbf{P}_{F,n}^{j_F,j_n}(k|l;k) \end{aligned} \tag{3.22}$$

with $m = 1, \ldots S$, $n = 1, \ldots S$ and $m \neq n$.

For fusion *with feedback* (Bar-Shalom, 2006) all sensors $i = 1, \ldots, S$ local tracks $\hat{\mathbf{x}}_i^{j_i}(l)$, with $j_i, j_F \in [1, \ldots N_t]$ are replaced after fusion (at time l) with $\hat{\mathbf{x}}_F^{j_F}(l)$. For the hypothesis that the S tracks $\hat{\mathbf{x}}_i^{j_i}(l)$, $i = 1, \ldots, S$ pertain to the same true target, the diagonal elements of the covariance matrix of $\hat{\mathbf{x}}_{1:S;F}(k)$ are

$$\begin{aligned} p_{mm} &= Cov\left[\hat{\mathbf{x}}_m^{j_m}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\right] \\ &= E\left[(\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l)) \cdot (\tilde{\mathbf{x}}_n^{j_m}(k) - \tilde{\mathbf{x}}_F^{j_F}(k|l))'\right] \\ &= E\left[(\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_m^{j_m}(k|l)) \cdot (\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_m^{j_m}(k|l))'\right] \\ &= \mathbf{P}_m^{j_m}(k) + \mathbf{P}_m^{j_m}(k|l) - \mathbf{P}_m^{j_m}(k;k|l) - \mathbf{P}_m^{j_m}(k|l;k) \end{aligned} \tag{3.23}$$

with $m = 1, \dots S$. The non-diagonal elements of the covariance matrix above are

$$\begin{aligned} p_{mn} &= Cov\left[\hat{\mathbf{x}}_m^{j_m}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l), \hat{\mathbf{x}}_n^{j_n}(k) - \hat{\mathbf{x}}_F^{j_F}(k|l)\right] \\ &= E\left[(\tilde{\mathbf{x}}_m^{j_m}(k) - \tilde{\mathbf{x}}_m^{j_m}(k|l)) \cdot (\tilde{\mathbf{x}}_n^{j_n}(k) - \tilde{\mathbf{x}}_n^{j_n}(k|l))'\right] \\ &= \mathbf{P}_{m,n}^{j_m,j_n}(k) + \mathbf{P}_{m,n}^{j_m,j_n}(k|l) - \mathbf{P}_{m,n}^{j_m,j_n}(k|l;k) - \mathbf{P}_{m,n}^{j_m,j_n}(k;k|l) \end{aligned} \tag{3.24}$$

with $m = 1, \dots S$, $n = 1, \dots S$ and $m \neq n$.

For the linear gaussian case the exact or approximate recursions introduced in (Tian and Bar-Shalom, 2009) can be used. For the multimodal gaussian case, a similar method as in (Bar-Shalom and Chen, 2006), (Chen *et al.*, 2003) can be used for cross-covariances approximation. Both methods are described next.

Cross-Covariances computation using exact recursion

Assuming all targets have equal process noise covariances $Q(k)$ at each time k (i.e. they move in formation), the covariances in (3.21), (3.22) can be exactly computed based on the recursions in (Tian and Bar-Shalom, 2009).

The $\mathbf{P}_m^{j_m}(l|k)$ is computed using

$$\tilde{\mathbf{x}}_m^{j_m}(k|l) = W_0^e(k,l)\tilde{\mathbf{x}}_m^{j_m}(l|l) + \sum_{n=l+1}^{k} W_0^\nu(k, n-1)\nu(n-1) \tag{3.25}$$

where $m = 1, \dots, S, F$ and

$$W_0^e(k,l) = \prod_{n=0}^{k-l-1} F(k-n-1) \tag{3.26}$$

$$W_0^\nu(k, n-1) = -\prod_{p=0}^{k-n-1} F(k-p-1). \tag{3.27}$$

The $\mathbf{P}^{j_m,j_n}_{m,n}(k)$ is computed using (Tian and Bar-Shalom, 2009)

$$\mathbf{P}^{j_m,j_n}_{m,n}(k) = W^e_m(k,l)\mathbf{P}^{j_m,j_n}_{m,n}(l)W^e_n(k,l)' + \sum_{n=l+1}^{k} W^\nu_m(k,n-1)Q(n-1)W^\nu_m(k,n-1)' \tag{3.28}$$

where

$$W^e_s(k,l) = \prod_{n=0}^{k-l-1} [(I - K_s(k-n))H_s(k-n)]F(k-n-1) \tag{3.29}$$

$$W^\nu_s(k,n-1) = \{ \prod_{p=0}^{k-n-1} [(I - K_s(k-p))H_s(k-p)]\, F(k-p-1)\} \cdot [I - K_s(n)H_s(n)] \tag{3.30}$$

and

$$W^w_s(k,n) = \{ \prod_{p=0}^{k-n-1} [(I - K_s(k-p))H_s(k-p)]\, F(k-p-1)\} K_s(n). \tag{3.31}$$

The recursion of the cross-covariance between the current fusion frame k local estimates $\hat{x}^{j_s}_s(k)$ and the predicted fused estimates $\hat{x}^{j_F}_F(k|l)$, $s = 1, \dots, S$, $j_s, j_F = 1, \dots, N_t$ for the no feedback case results as (Bar-Shalom and Li, 1995)

$$\mathbf{P}_{F,i}(k|l;k) = W^l_0(k,l) \cdot \mathbf{P}_{F,1}(l) \cdot W^l_1(k,l) + \sum_{i=l+1}^{k} W^\nu_0(k,i-1) \cdot \mathbf{Q}(i-1) \cdot W^\nu_1(k,i-1)' \tag{3.32}$$

with

$$\mathbf{P}_{F,i}(l) = [I - K_{12}(l)] \cdot \mathbf{P}_1(l) + K_{12}(l) \cdot \mathbf{P}_{12}(l) \tag{3.33}$$

and

$$K_{12}(l) = [\mathbf{P}_1(l) - \mathbf{P}_F(l)] \cdot [\mathbf{P}_1(l) + \mathbf{P}_F(l) - \mathbf{P}_{1,F}(l) - \mathbf{P}_{1,F}(l)]^{-1}. \tag{3.34}$$

Cross-Covariances approximation based on steady-state

For nonlinear and non-gaussian assumptions the covariances in (3.21), (3.22) can be approximated using the method in (Chen *et al.*, 2003).

Next the the second probability in the RHS of (3.12) is evaluated. The conditional set of fused tracks at l, Φ_p^l, implicitly contains the corresponding association hypothesis that generated them at l, namely $\mathcal{H}_p(l)$. This contains the mappings of the local tracks identities to fused tracks identities. Therefore we write it as

$$P\left(\mathcal{H}_n(k) \mid \Phi_p^l\right) = P\left(\mathcal{H}_n(k) \mid \mathcal{H}_p(l), \Phi_p^l\right) = P\left([\mathcal{H}_n(k) \mid \mathcal{H}_p(l)] \mid \Phi_p(l)\right). \qquad (3.35)$$

The first equality in (3.35) can be written as the association $\mathcal{H}_p(l)$ is implicitly contained in the set of fused tracks Φ_p^l and does not introduce any additional condition. The second equality in (3.35) is based on the same fact that $P\left(\mathcal{H}_p(l)\right) = P\left(\mathcal{H}_p(l), \Phi_p(l)\right)$. The event $[\mathcal{H}_n(k) \mid \mathcal{H}_p(l)]$ in (3.35) represents a track swap event (hypotheses transition) at at least on local sensor. The mapping hypothesis $\mathcal{H}_p(l)$ at time frame l switches to the mapping hypothesis $\mathcal{H}_n(k)$. In an association hypothesis the fused tracks are taken as reference for the associated tracks identities. The swapping probability from one hypothesis to another is conditioned on the set of fused tracks configuration at l, Φ_p^l. This probability could be estimated at the local tracker level, based on the distribution and uncertainties of local tracks estimates between times l and k, using a formula based on (3.10), and communicated to the fusion center. However, this is not the case for the scenario treated here. For local trackers of equal probabilities and uniformly distributed fused tracks states at l, this probability would be the same for different swapping hypotheses. The third probability in (3.35) is the probability of the parent set of fused tracks. This is evaluated from the cost of the track-to-track association hypothesis that generated it at frame l. It is the probability of the set of fused tracks Φ_p^l resulting from the cumulative hypothesis at l, Θ_p^l.

Observation 1: For the same number of targets estimated by each local sensor at times l and k, and no swapping at any of the local sensors we have $P\left(\mathcal{H}_n(k) \mid \Phi_p^l\right) = P\left([\mathcal{H}_n(k) \mid \mathcal{H}_p(l)] \mid \Phi_p^l\right)$, and therefore

$$P\left([\mathcal{H}_n(k) \mid \mathcal{H}_p(l)] \mid \Phi_p^{k-1}\right) = \begin{cases} 1, \text{ for } \mathcal{H}_n(k) = \mathcal{H}_p(k-1) \\ 0, \text{ for } \mathcal{H}_n(k) \neq \mathcal{H}_p(k-1) \end{cases} \tag{3.36}$$

Observation 2: Next we emphasize the difference and show the improvement brought by our solution compared to the one in (You and Jingwei, 2006). We consider only two sensors and only one solution of hypotheses preserved at fusion time l, Φ^l. We take the logarithm in (3.12) and use the following.

For the first RHS probability we use the likelihood ratio of common origin of tracks as in (3.18). Here we drop the conditioning on (i.e. association with) the predicted fused tracks of previous frame in the first equality below, as

$$\begin{aligned} \ln P\left(\left[\hat{X}_S(k)\right] \mid \Phi^l, \mathcal{H}(k)\right) &= \ln P\left(\left[\hat{X}_S(k)\right] \mid \mathcal{H}(k)\right) = \ln\left(\mathcal{V} \cdot \mathcal{N}\left(\hat{\mathbf{x}}_{1;2}; \mathbf{0}, \mathbf{P}_{1;2}\right)\right) \\ &= c_2 + \left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right)^T \mathbf{P}_{1;2}(k)^{-1} \left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right). \end{aligned} \tag{3.37}$$

For the second RHS probability we use (3.36) under the assumptions that no track swap happens at the local trackers levels. This means the association hypothesis that generated Θ^l at frame l is also the valid one at frame k. In consequence $\mathcal{H}(k) \equiv \mathcal{H}(l)$ and this probability is equal to 1. The equivalence here means that association hypotheses group identically the track labels from local sensors sensors at both times

k and l.

$$\ln P\left(\mathcal{H}(k) \mid \Theta^l, NoTrackSwap, \mathcal{H}(k) \equiv \mathcal{H}(l)\right) = 0. \tag{3.38}$$

For the third RHS probability, we use the cumulative likelihood ratio of common origin of tracks at previous fusion time, $P(\Phi^l)$.

By combining (3.36), (3.38) in (3.12) a similar formula as the one obtained in (You and Jingwei, 2006) for the recursion of the logarithm of cumulative likelihood of the association hypothesis ($n = p$ dropped here, as $\mathcal{H}_n(k) \equiv \mathcal{H}_p(l)$ is implicitly assumed)

$$\ln P\left(\Phi^k\right) = c_2 + \left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right)^T \mathbf{P}_{1;2}^{-1}(k)\left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right) + \ln P\left(\Phi^l\right). \tag{3.39}$$

Equation (3.39) contains the recursion for the sum of likelihood ratios for all the pairs of tracks assumed of common origin in Φ^k. For a single pair of tracks (j_1, j_2) the log-likelihood of common origin, following (3.39) is obtained as in (You and Jingwei, 2006)

$$\lambda_{j_1 j_2}^k = c_3 + \left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right)^T \mathbf{P}_{1;2}^{-1}(k)\left(\hat{\mathbf{x}}_2(k) - \hat{\mathbf{x}}_1(k)\right) + \lambda_{j_1 j_2}^l = \lambda_{j_1 j_2}(k) + \lambda_{j_1 j_2}^l \tag{3.40}$$

with coefficients c_2 and c_3 resulting from the normalizing coefficient of the corresponding normal distributions. In the above simplifications, as in (You and Jingwei, 2006), implicitly, the assumptions of association hypothesis preservation over the cumulative summation period, as well as the non-swapping at local trackers level are made. The cross-correlation between the tracks at different fusion frames is not considered in (3.37) and resulting (3.40), which as shown in (Tian and Bar-Shalom, 2009) leads to erroneous results. In the method proposed here through (3.12) the conditioning on the previous frame fused tracks Φ_p^l is considered. The conditioning on the previous frame is present through the predicted fused tracks considered in the association cost (3.18) through (3.20) and its covariance matrix $\mathbf{P}_{m,n}^{j_m,j_n}(k;k|l)$ detailed in (3.21)–(3.22). The general recursion implemented through the method proposed here, accounts for swaps at the local trackers through considering several N hypotheses at each fusion frame. Also the cross-correlation between fusion frames is considered. The computation of the cost of an association hypothesis is, following (3.12)

$$\ln P\left(\Phi_n^k\right) = \ln \frac{1}{c} P\left(\left[\hat{X}_S(k)\right] |\Phi_p^l, \mathcal{H}_n(k)\right) + \ln P\left(\mathcal{H}_n(k)|\Phi_p^l\right) + \log P\left(\Phi_p^l\right) \qquad (3.41)$$

with intervening probabilities described above. The recursion (3.41) preserves, through the predicted fused tracks sets, the identity of the fused tracks over subsequent fusion times, even under an association hypothesis change (due to track swap at local sensor).

3.4.2 Proposed Implementation

The proposed implementation finds the best association hypotheses and resulting fused tracks using the recursion (3.41). The likelihood of hypotheses switching (hypotheses transition), namely the second in the RHS of (3.12) or (3.41) are indirectly accounted for by considering multiple hypotheses at each association frame. The continuity likelihood of a given set of fused tracks over several hypotheses is inferred from the evolution of the cumulative (current and previous) association costs for a set of best current hypotheses, selected at each fusion frame. Fig. 3.2 shows the block diagram of the method implementation. Only the fusion frames are shown in the figure through the fusion frame index K. At first fusion frame, i.e. $K-1$, there are N association hypotheses selected. These hypotheses become parent hypotheses

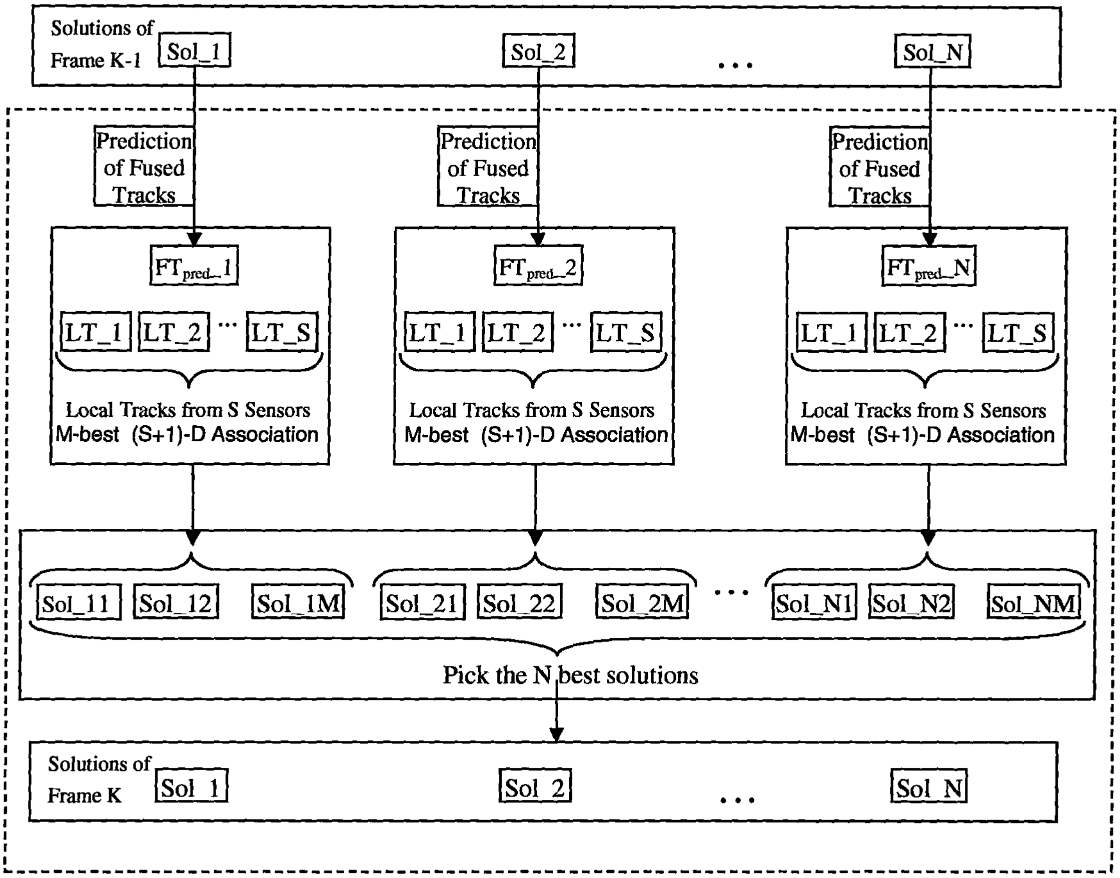

Solutions of Frame K-1
Sol_1
Sol_2
...
Sol_N
Prediction of Fused Tracks
Prediction of Fused Tracks
Prediction of Fused Tracks
FT_pred_1
FT_pred_2
FT_pred_N
LT_1 LT_2 ... LT_S
LT_1 LT_2 ... LT_S
LT_1 LT_2 ... LT_S
Local Tracks from S Sensors
M-best (S+1)-D Association
Local Tracks from S Sensors
M-best (S+1)-D Association
Local Tracks from S Sensors
M-best (S+1)-D Association
Sol_11 Sol_12 Sol_1M
Sol_21 Sol_22 Sol_2M
...
Sol_N1 Sol_N2 Sol_NM
Pick the N best solutions
Solutions of Frame K
Sol_1
Sol_2
...
Sol_N

Figure 3.2: Solutions flow for the multiframe multiple hypothesis track-to-track association.

in the multiple association hypotheses (MHA) tree. Each of these hypotheses generates a set of fused tracks. Each parent hypothesis selected at time $K-1$ enters the association at time K through its fused tracks, predicted for time K. For S sensors, a number of N problems of $(S+1)$-D association are generated at each frame. The best M solutions (hypotheses) of each of the $(S+1)$-D associations are kept. The best N association hypotheses are selected from the set of resulting N x M solutions. In order to avoid an exhaustive search through all possible hypotheses that might be generated (as required by an ideal MHA), two maximizations are used to prune the current set of solutions:

a) from the set of all possible association hypotheses continuation of each parent hypothesis, the best M hypotheses are kept (M-best S-D association)

b) from the set of M x N resulting hypotheses, the best non-duplicated N solutions are kept further. The duplicate hypotheses in the set of N-best hypotheses are removed without being replaced.

The continuity of the fused solutions over several frames and hypotheses transitions is achieved through the propagation of each of the N solutions of fused tracks at frame $k-1$ $\Phi_p^{k-1}, p = 1..N$ to one or more solutions at the frame k, $\Phi_p^k, n = 1..N$. A sample resulting tree of associations chain in time is shown in Fig.3.3. It can happen that several solutions at time $k-1$ generate identical association solutions at the next fusion time k and some of these identical solutions have their costs in the best N costs at k. In this case the places of the duplicates of a given solution are left empty in the current step, in order not to introduce too many confusing solutions.

The best association chain is retrieved starting backwards from the best hypothesis at the latest time (shown with thicker line). When two paths are available from one solution, the path of minimum cost is picked. The best solution for the sample

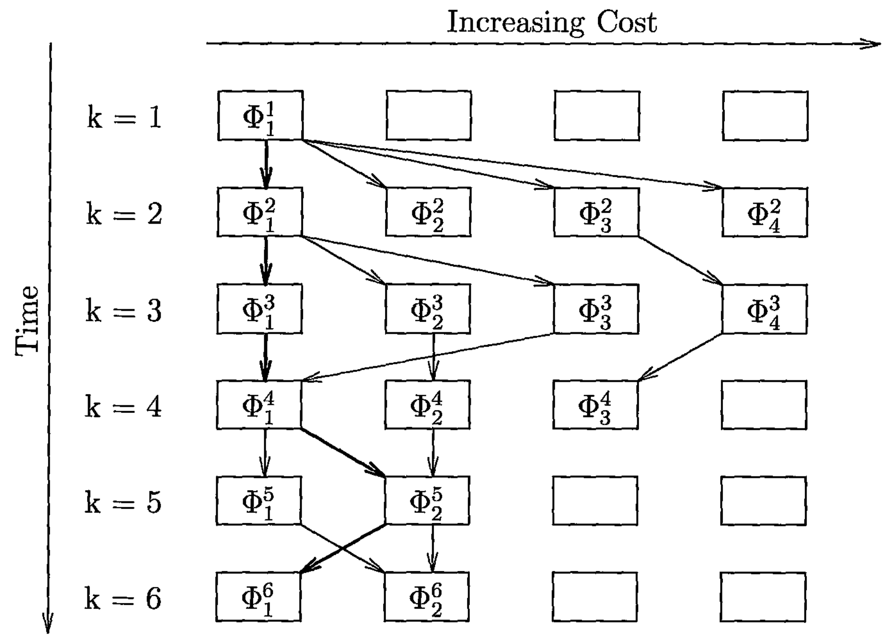


Figure 3.3: Sample of the resulting tree of fused tracks hypotheses, for k=6 association times and N=4.

tree is shown with thick line in Fig.3.3. From the same figure, it can be seen that the resulting transitions of association hypotheses are implicitly inferred in the best hypotheses propagation.

3.5 Simulation Results

In the simulation presented next there were used 3 sensors and 2 targets that evolve closely enough such that the local trackers run under measurement origin uncertainty, and therefore track swap happens at the local trackers level. A fixed number of targets and estimated tracks are simulated in the overlapping surveillance area of both sensors over the entire simulation time. The probability of detection is $P_D = 1$. For the m-best association the m=2 or m=3 best solutions are kept and the set of

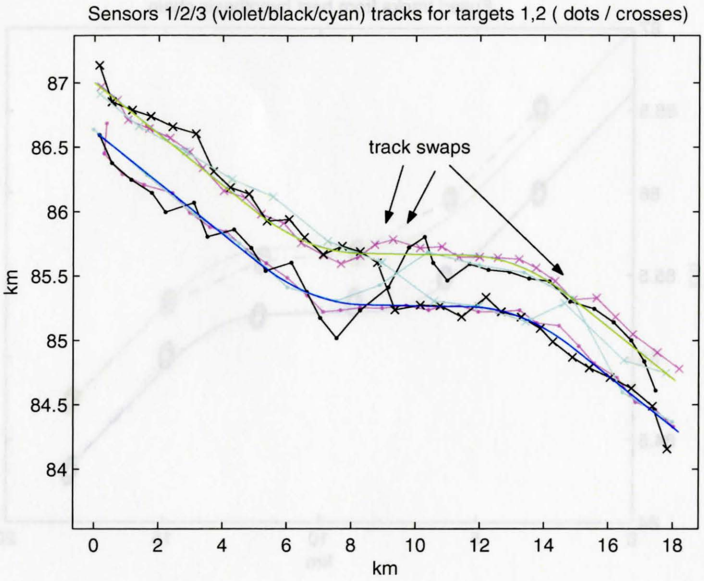


Figure 3.4: Local tracks showing track swaps during maneuver.

final fused solution preserved was set to N=2 and N=3. Therefore with respect to the association algorithms, we ran in parallel N m-best 4-D association problems and prune the solutions to preserve at most N hypotheses at each frame. Each local sensor is runnning an IMM estimator with two modes, of different process noise levels. For the IMM estimator, the approximate cross-covariance computation method is used. The positions of the sensors are $(0, 187)$ km, $(50, 0)$ km, while their bearing and range standard deviation errors are of 2 mrad and 50 m. Two sensors have the sample rate equal to 2 sec, and the third one 5 sec, with synchronous sampling and updates. A fusion is performed every 10 sec. The simulated targets have a velocity of 300 m/s and perform light coordinated turns of 1 deg/sec, as shown in Fig.3.4, or run straight trajectories. With these parameters, tracks swap happen at the local sensors, as shown in the same figure for a sample run of the coordinated turn scenario. Using several fusion hypotheses at each fusion time (i.e., N=3, M=3 in the sample run depicted

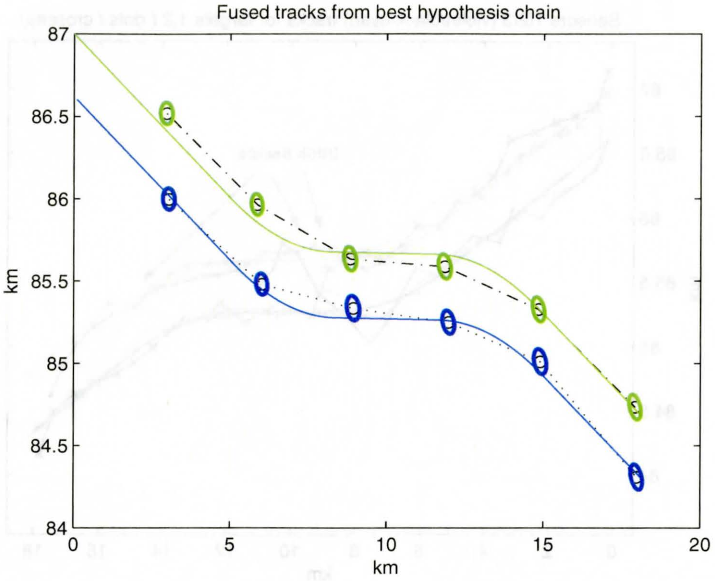


Figure 3.5: Fused tracks with track identities preserved. True target trajectories are pictured with continuous line. 2σ confidence ellipses of fused track estimates covariance matrices are shown.

Table 3.1: Results of Monte Carlo simulations.

Scenario Type	M	N	Correct Target Identification [%]
Nearly Constant Velocity	2	2	77.11
Nearly Constant Velocity	3	3	87.80
Coordinated Turn	2	2	67.34
Coordinated Turn	3	3	77.55

in Fig.3.4), the identity of the fused tracks in the best hypotheses chain is preserved during the simulation, as shown in Fig.3.5. Results of Monte Carlo simulations (100 runs) for the track identity preserved at the fusion center in the condition of local trackers randomly swapping targets, are shown in Table 3.1.

3.6 Conclusion

Various simulations showed that the identification of the targets in the case of local tracks swap is still possible at the fusion center. A dependency of the fused tracks identification results on the scenario was observed, therefore an on-line estimation of the results uncertainty is intended to be pursued as future work, as well as improved hypotheses generation, solutions pruning and search schemes. Additional information provided by the local trackers, as the estimation of track swap probability at local tracker between the fusion interval, as well as the process noise within the period, is intended to be used for improved hypotheses assessment.

Chapter 4

Track Fusion with Feedback for Local Trackers Using MHT

With current processing power, Multiple Hypothesis Tracking (MHT) becomes a feasible and powerful solution; however a good hypothesis pruning method is mandatory for efficient implementation. The availability of a continuously increasing number of tracking systems raises interest in combining information from these systems. The purpose of this work is to propose a method of information fusion for such trackers that use MHT locally with local information sent in the form of sensor global hypotheses and the fusion center combining them into fused global hypotheses. The information extracted from the best fused global hypotheses, in the form of ranking of received sensor global hypotheses, is sent back to local trackers, for optimized pruning. Details of the method, in terms of sensor global hypotheses generation, evaluation, pruning at local sensors, association and fusion of sensor global hypotheses at fusion center, and usage of the information received as feedback from the fusion center are presented.

4.1 Introduction

In a multitarget scenario with dynamic number of targets, using sensors with probability of detection less than unity, in the presence of false alarms and under measurement origin uncertainty, the multiple hypothesis tracking (MHT) has been shown to have a very good performance. Although the MHT computation increases exponentially in time, current computer power allows the use of MHT method at the local sensor level, with a proper local management of hypotheses, in order to keep their number within a reasonable limit. Therefore, the problem of hypotheses pruning becomes very important in decreasing the computation complexity and yet preserving tracking accuracy. This chapter presents a track-to-track association and fusion with feedback method to be utilized in the case when one or more local trackers use MHT. The novelty is the association and fusion of global hypotheses obtained by MHT estimators, as well as the usage of feedback from the fusion center at the local trackers in the hypothesis pruning process. The fusion center, having information on the best hypotheses available at several MHT trackers, contributes through feedback to improve the local pruning, using the global available information and forcing local trackers to promote from one fusion time to the next also the local hypotheses with best fusion results. A method for local hypotheses evaluation, pruning and selection for fusion is presented, as well as the selection of best fused hypotheses at the fusion center. The association (with feedback) method is suitable for data fusion in centralized or decentralized architectures. There is no necessity that all sensors run MHT locally, however the feedback received from fusion could be applied only to the MHT estimators, as it is in terms of hypothesis ranking. For a given target, its local tracks to be associated are considered with dependent errors; for different targets, their cross-correlated errors are assumed independent A simulation for evaluating the feedback scheme proposed is presented, using a scenario with multiple sensors and close targets, therefore having local measurements origin uncertainty.

4.2 MHT Tracking

While Bayesian target tracking approaches can be categorized as target-oriented (JPDAF) or measurement-oriented (MHT) (Bar-Shalom and Li, 1995), the MHT variants themselves could be categorized as following a hypothesis-oriented or track-oriented approach (Reid, 1979; Kurien, 1990). In the measurement-oriented approach, for each measurement, every target hypothesis (already confirmed through an existing track, newly detected, or none - i.e. false alarm) is considered for measurement to track association (Bar-Shalom and Li, 1995). A global hypothesis represents the result of a sequence of association of measurements from the beginning up to current time. In the target-oriented approach, for each confirmed target, all measurements are considered (Reid, 1979). As described in (Drummond, 2003), there are at least three accepted definitions for MHT. The first one recognizes it as the method that builds and propagates all hypotheses from the first measurement up to the current time. The second definition is any variant that propagates more than one hypothesis for subsequent processing and therefore approximates the suboptimal method introduced in (Reid, 1979). The third definition includes tracking methods based on multiple frame data association. Following this categorization, in this research the second definition is considered.

4.2.1 Hypothesis-oriented MHT

Following the hypothesis-oriented approach (Bar-Shalom and Li, 1995; Reid, 1979), each compatible sequence of measurements is considered to form a hypothesized track, of which statistics are estimated using a sequential estimator. A hypothesis, in the context of this method, is a set of tracks representing at a given time all the estimated targets in the surveillance area. Several hypotheses coexist, and are propagated at each measurement time. Also at most of measurement times each existing hypothesis generates several different offspring hypotheses, some other completely new hypothesis are generated, while few hypotheses might merge, if becoming close enough. Due to the continuously increasing number of hypotheses that would be generated in an optimal MHT, its implementation is most of the times unaffordable. Judicious pruning of hypotheses becomes an essential factor for a functional MHT implementation. The hypotheses are selected based on their estimated probabilities, first time introduced in (Reid, 1979). For $s = 1, \ldots, S$ sensors, we denote a cumulative association event (or hypothesis), labeled $l_s(k)$ at sensor s at a given time k by $\Theta_s^{k,l_s(k)} = \{X_s^{k,l_s}\}$, with $X_s^{k,l_s} = \{\hat{\mathbf{x}}_s^{l_s(k),i}(k)\}_{i=1}^{L_s(k)}$, the corresponding set of estimated tracks. The cumulative association event at time k, denoted by the upper index, is the result of previous cumulative association event and current association event $\theta_s^{l_s(k)}(k)$. Therefore we have $\Theta_s^{k,l_s(k)} = \{\Theta_s^{k,l_s(k-1)}, \theta_s^{k,l_s(k)}(k)\}$, as the current hypothesis is formed from the parent hypothesis and the current association event. A current association event, $\theta_s^{l_s(k)}(k)$, is the set of associated tracks, new formed tracks and false alarms at the current time k. The hypothesis probability, conditioned on the sequence of measurements up to time k, $\mathbf{Z}^{\mathbf{k}}$, can be therefore written (by applying Bayes rule) as (Bar-Shalom and Li, 1995), (Reid, 1979):

$$
\begin{aligned}
&P\left(\Theta_s^{k,l_s(k)}|Z^k\right) = P\left(\theta_s^{l_s(k)}(k), \Theta_s^{k-1,l_s(k-1)}\right) = P\left(\theta_s^{l_s(k)}(k), \Theta_s^{k-1,l_s(k-1)}|Z(k), Z^{k-1}\right) \\
&= \tfrac{1}{c}P\left(Z(k)|\theta_s^{l_s(k)}(k), \Theta_s^{k-1,l_s(k-1)}, Z^{k-1}\right) \cdot P\left(\theta_s^{l_s(k)}|\Theta_s k-1, l_s(k-1), Z^{k-1}\right) \cdot \\
&\quad P\left(\Theta_s^{k-1,l_s(k-1)}|Z^{k-1}\right)
\end{aligned} \tag{4.1}
$$

where $Z(k) = \{\mathbf{z_s^i(k)}\}_{\mathbf{i=1}}^{\mathbf{m_s(k)}}$ is the set of $m_s(k)$ measurements at time k at sensor s. The first term in lhs of (4.1) represents the likelihood of the current association event, the second represents the prior probability of current association event and last term represents the probability of cumulative association event at time $k-1$. Using Poisson distribution for the false alarms target births distributions, the recursion in (4.1) was reduced in (Bar-Shalom *et al.*, 2007) to

$$
\begin{aligned}
&P\left(\Theta_s^{k,l_s(k)}|Z^k\right) = \\
&\tfrac{1}{c'}\left(\lambda_\phi\right)^\phi\left(\lambda_\nu\right)^\nu \cdot \textstyle\prod_{j=1}^{m(k)}\{f_{\tau_j}\left[z_j(k)\right]\}^{\tau_j} \cdot \prod_t \left[[P_{Dt}(k)]^{\delta_t}\left[[1-P_{Dt}(k)]^{1-\delta_t}\right]\right. \cdot \\
&P\left(\Theta_s^{k-1,l_s(k-1)}|Z^{k-1}\right)
\end{aligned} \tag{4.2}
$$

where

c' contains all constant terms for all events

ϕ is the number of false alarms in the current association event

ν is the number of measurements assumed generated from newborn targets

λ_ϕ is the expected number of false measurements per unit volume in current frame

λ_ν is the expected number of measurements from new targets per unit volume in current frame

$m(k)$ is the number of measurements at time k

f_{tj} is the pdf of predicted measurement of track t_j to which measurement $z_j(k)$ is assigned under hypothesis considered (innovation pdf)

τ_j indicator function, equal to 1 for measurement $z_j(k)$ assigned to a track under hypothesis considered, 0 otherwise

δ_t indicator function, equal to 1 for target t detected (measurement assigned to track t) under hypothesis considered

Equation (4.2) is the basis of evaluating and ranking the local hypotheses in the pruning process and selection of hypotheses to be sent to the fusion center for the hypotheses oriented MHT.

4.2.2 Track-oriented MHT

In the track-oriented MHT, for each hypothesized target its track hypotheses are sequentially built into an independent tree, named track hypotheses tree, or target tree, with the root being the target birth, and a new generation of branches being added at each measurement frame (Kurien, 1990). All realistic (i.e. based on gating) track continuations brought by each measurement in a frame are considered to generate new branches, independently of the existence of other targets. In (Drummond, 2003) it is suggested also that different branches be generated by different target dynamics. A possible track for a target is represented by a succession of nodes in such a tree, from the root to a current leaf, and is named here a track hypothesis. Each target tree therefore contains all accepted track hypotheses. At a given time, the scenario

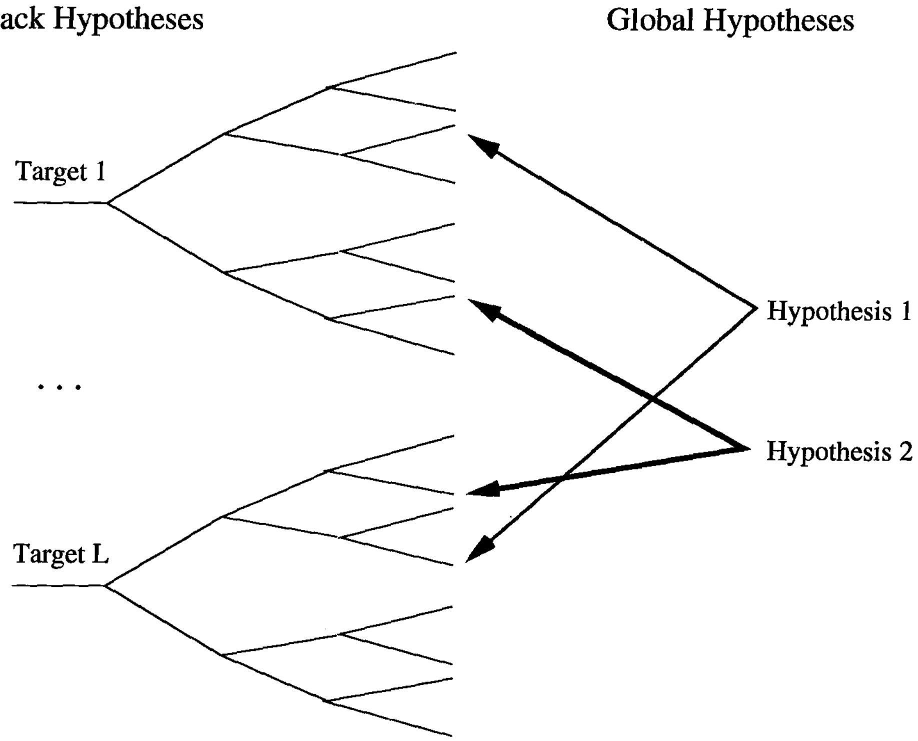


Figure 4.1: Track-Oriented MHT showing target trees of track hypotheses (L hypothesized targets) and sensor global hypotheses.

on the ground could be estimated (with more or less accuracy) through a combination of track hypotheses, by taking at most one track from each target tree. Such a combination represents a sensor global hypothesis. Therefore, following (Kurien, 1990), a sensor global hypothesis is represented through a list of pointers to the track hypotheses in different target trees.

As shown in (Bar-Shalom *et al.*, 2007), the track-oriented approach is equivalent to the hypothesis oriented approach under the assumption that the expected number of measurements from new targets per scan, per unit volume of the measurement space is zero (in terms of equations (4.1) and (4.2)), however in consequence all false alarms are considered as potential new targets. Being based on easier global hypotheses generation and management for implementation, as described further, the track-oriented MHT version was considered for the method presented here.

4.2.3 Sensor Global Hypotheses Generation

With each measurement frame, the current track hypotheses in a target tree are extended with a new set of branches, corresponding to different tracks continuations with different measurements. Using these new track hypotheses, new sensor global hypotheses are generated through feasible combinations of updated (new) track hypotheses. A *feasible* combination of track hypotheses implies that they do not share any common measurement in the past. To ensure that the newly generated global hypotheses are feasible, the component track hypotheses are chosen only from the offspring of track hypotheses previously grouped in a parent sensor global hypothesis. This is shown in Fig.4.2, where from the existing sensor global hypotheses at time k, $\Theta^{k,1}$ and $\Theta^{k,2}$, two other feasible global hypotheses, $\Theta^{k+1,1}$ and $\Theta^{k+1,2}$, are obtained for time $k+1$. The track hypotheses of each new global hypotheses, up to time k, are selected from the components of a previous global hypothesis at time k, therefore the feasibility up to time k is already ensured through this generation method. For example, from the target birth, up to time k, all track hypotheses of global hypotheses $\Theta^{k+1,1}$ are identical with the track hypotheses of $\Theta^{k,l}$. The hypothesis $\Theta^{k+1,1}$ is within the offspring set of $\Theta^{k,1}$, while $\Theta^{k+1,2}$ is within the offspring set of $\Theta^{k,2}$. The feasibility of global hypothesis from k to time $k+1$ implies that at time $k+1$ only, the track hypotheses do not have in common any new measurement, which is a classic constraint of the $2D$ assignment, commonly used in measurement to tracks association.

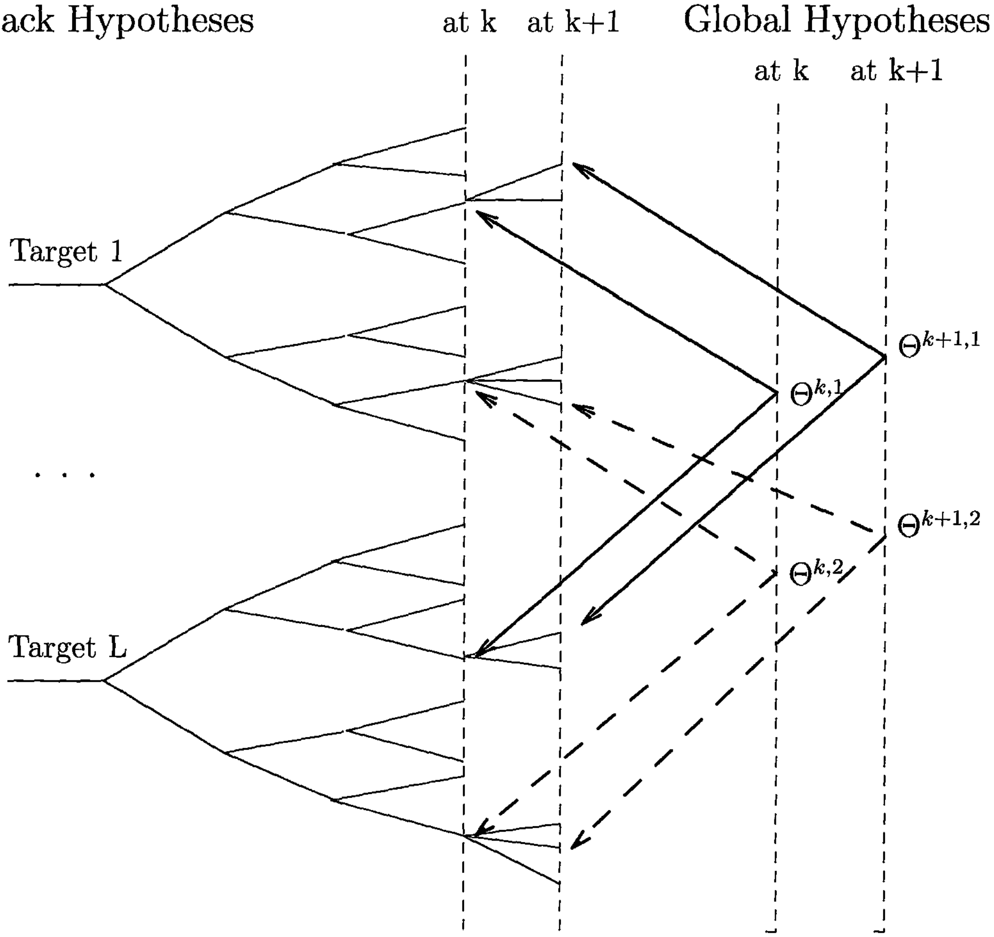


Figure 4.2: Sensor global hypotheses generation ensuring feasibility.

4.2.4 Hypotheses Management

In a MHT method, the number of track hypotheses grows exponentially in time, with each set of new measurements. The growth of hypotheses number has two main sources: multiple measurements per measurement frame (including false alarms) and multiple track hypotheses existence at each time for a given target. Two techniques, described further, gating and m-best hypotheses selection, are used to keep low both the number of track hypotheses and number of global hypotheses, respectively. Further the notation in (Kurien, 1990) is used and we denote by *screening* the procedure of not considering some alternatives at the time they occur, and by pruning the elimination of certain alternatives at a later time after they had been pursued. Screening of Track Hypotheses - Gating The gating technique (Bar-Shalom and Li, 1995) screens out measurements that are not valid association candidates for the continuation of a given track, therefore eliminating least probable track hypotheses. For each existing track hypothesis a validation region is computed around its resulting track estimate, i.e. $\hat{\mathbf{x}}^l(k-1)$, and only measurements that fall within it are considered as possible associations to and continuations of the given track. This region is computed assuming a normal distribution of measurements conditioned on the last track estimation (Bar-Shalom and Li, 1995)

$$p(z(k) \mid Z^{k-1}) = \mathcal{N}\left(z(k) \mid \hat{z}^l(k|Z^{k-1}), \mathbf{S}^l(k)\right). \tag{4.3}$$

Here $\hat{z}^l(k|Z^{k-1}) = \mathbf{H}(k)\hat{\mathbf{x}}^l(k|k-1)$ is the predicted measurement for track l based on the past information (built out of the set measurements up to time $k-1$ (i.e. Z^{k-1}). $\mathbf{S}^l(k)$ is the innovation covariance for track $\hat{\mathbf{x}}^l(k|k-1)$ (Bar-Shalom *et al.*, 2001). Based on (4.3), the validation region for track l at time k is defined as the region

$$\Re(l,k) = \{z(k) | \left(z(k) - \hat{z}^l(k|Z^{k-1})\right)^T \left(\mathbf{S}^l(k)\right)^{-1} \left(z(k) - \hat{z}^l(k|Z^{k-1})\right) \leq \gamma\} \tag{4.4}$$

with γ being the threshold selected depending on the desired accuracy and dimension of the measurement z.

Pruning of Sensor Global Hypotheses - m-best

What is proposed for pruning of sensor global hypotheses is, beside the selection of global hypotheses only as combination of offspring track hypotheses grouped in parent global hypotheses (detailed in subsection 4.2.3), the selection at each time of only the N-best (best in a likelihood sense) ones per sensor for further local processing. The selection is based on each sensor global hypothesis evaluation through its score, computed based on the likelihoods (or scores) of its component tracks hypotheses. The selection of sensor global hypotheses is detailed in the following subsection, while the evaluation of track hypotheses and their constraints is detailed in the subsection 4.2.5).

Pruning of Sensor Global Hypotheses Through m-Best 2D Assignment

Through the m-best 2D assignment method, the track hypotheses and sensor global hypotheses are pruned altogether at the global hypothesis level, by preserving from one time to the next at most N global hypotheses. This is implemented in two stages, as shown in Fig.4.3. First, for each of the N currently available global hypotheses at time k, the m-best track continuation hypotheses are found through m-best $2D$ assignment, ran between the list of track hypotheses at time $k-1$ and list of available measurements at time k. At the end of first stage there will be $N \times m$ hypotheses obtained. From this list of hypotheses, in the second stage, the best N hypotheses are selected, by ranking their likelihoods using equation (4.2). The $2D$ association can be implemented through auction algorithms, while the m-best solutions can be found by using Murty's algorithm (Popp *et al.*, 2001).

This method resolves (locks) at each time k at most N global hypotheses. An

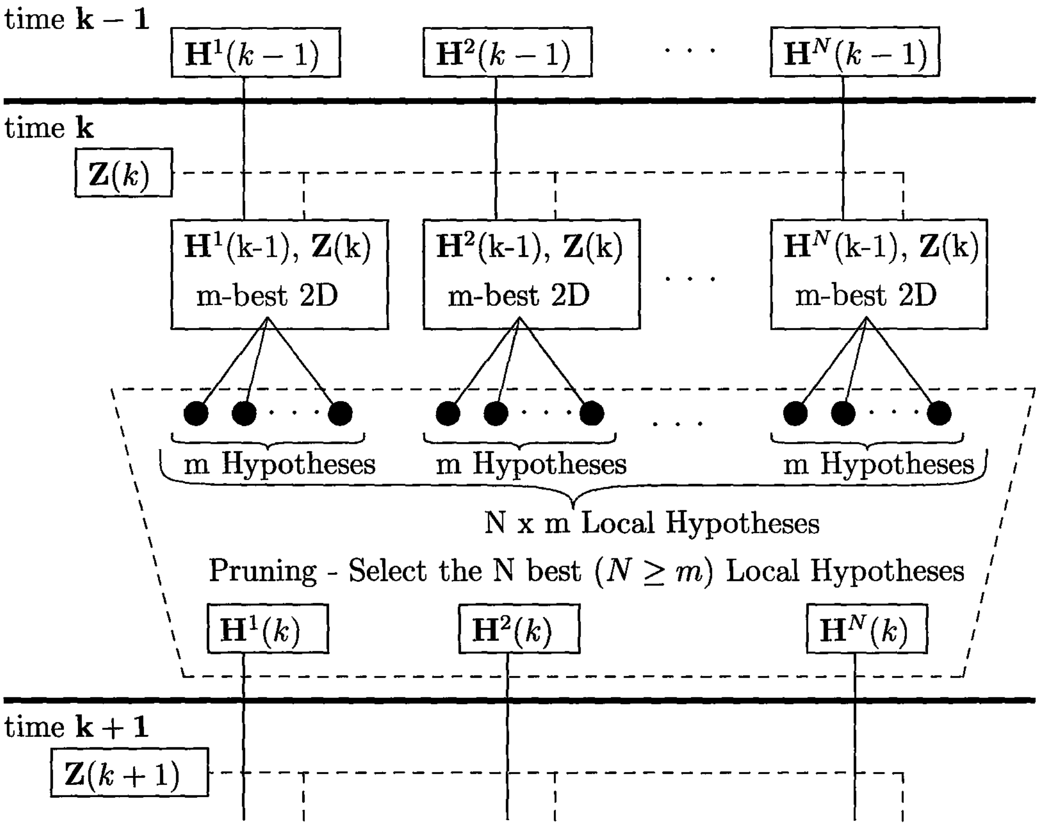


Figure 4.3: Pruning of global hypotheses and track hypotheses altogether through m-best 2D assignment.Note: the sensor index is dropped and a sensor global hypothesis labeled l, at time k is denoted by $H^l(k)$.

improvement of the above method could lock at each time k a number of N global hypotheses at time $k-n+2$, with $n > 2$. For this purpose, in the first stage, for each of the N currently locked global hypotheses at time $k-n+1$, an m-best $n-D$ association is ran between the list of tracks in the global hypothesis considered and the $n-1$ lists of measurements (from time $k-n+2$ to k). Using the m-best assignments found in each solution (out of m for each hypothesis), the tracks at time $k-n+2$ are locked to form a new global hypothesis. Therefore at the end of the first stage there will be $N \times m$ global hypotheses (tracks continuations from $k-n+2$

to k) to be retained from each initial N global hypotheses. In the second stage, from these $N \times m$ new global hypotheses, the best N are evaluated and selected. These N hypotheses will be resolved (locked) at time $k-n+2$. For fusion (if done at this time k), the set of tracks estimated at time k of each global hypothesis are considered and sent further. The $(n+1)-D$ association could be carried on through Lagrangian Relaxation techniques (Poore and Rijavec, 1993), while the m-best solutions can be found through variants of Murty's algorithm (Popp *et al.*, 2001). Two tracks in a sensor global hypothesis should not share at any time backwards (on any branch) a common measurement. This is avoided up to time $k-1$ through selecting in a new global hypothesis only of track hypotheses offspring of track hypotheses present in a single parent global hypothesis at time $k-1$. The $2D$ assignment of measurements to tracks, ran at time k and described through equations (4.5–4.7), ensures the unique assignment of a measurement at time k to a single track, through equation (4.7):

$$\min_{\chi} \sum_{l=0}^{L(k-1)} \sum_{i=0}^{m(k)} \chi_{i,j} \cdot \text{cost}\left(\hat{\mathbf{x}}^l(k|k-1), \mathbf{z}^i(k)\right) \tag{4.5}$$

subject to $\chi_{l,i}$ being a binary variable satisfying

$$\sum_{i=0}^{m(k)} \chi_{l,i} = 1, \qquad l = 1, 2, \ldots, L(k-1) \tag{4.6}$$

$$\sum_{l=0}^{L(k-1)} \chi_{l,i} = 1, \qquad l = 1, 2, \ldots, m(k) \tag{4.7}$$

with $\mathbf{z^0(k)}$ denoting a dummy measurement (i.e. the measurement to which a track not detected is associated) and $\mathbf{x^0(k-1)}$ denoting the dummy track (i.e. the track to which a false alarm is associated). To each dummy (track or measurement), several associations can be made, while to each non-dummy measurement or track, only a single track or measurement association can be made, respectively. Therefore finding the m-best solutions of (4.5–4.7) results in finding m feasible global sensor hypotheses continuations at k for each parent (existing) sensor global hypothesis at time $k-1$. For an n-D association, the constrained optimization to be solved is similar, and detailed in (Popp *et al.*, 2001).

4.2.5 Track Hypothesis and Sensor Global Hypothesis Scoring

As derived in (Bar-Shalom *et al.*, 2007) track hypotheses can be scored using cumulative log-likelihood ratios (in fact the negative of it). The likelihood of a track hypothesis can be computed as a product of likelihoods at all previous times, therefore cumulative, as innovations are white (Bar-Shalom *et al.*, 2001). The likelihood ratio, computed as the ratio between the likelihood and the false alarm density, is selected instead of pure likelihood as the former provides dimensionless score function, which allows comparison of track hypotheses of different lengths (different numbers of measurement associations) (Bar-Shalom *et al.*, 2007). Therefore, the likelihood ratio of a track hypothesis t continuation with measurement $\mathbf{z}_j(k)$ is

$$\Lambda_{tj} = \frac{f_t\left[\mathbf{z}_j(k)\right]}{\lambda_{FA}} \tag{4.8}$$

where $f_t\left[\mathbf{z}_j(k)\right]$ is the pdf of the innovation brought by measurement $\mathbf{z}_j(k)$ and λ_{FA} is the spatial density of false measurements (with measurements from new targets included in this density). The cumulative likelihood ratio of a track hypothesis t is computed as $\Lambda_t^k = \prod_n^k \Lambda_{tjn}(n)$ and its negative logarithm is used in the track hypothesis score computation (Bar-Shalom *et al.*, 2007)

$$\ell_t^k = -\ln \lambda_t^k = -\sum_{n=1}^{k} \ln \Lambda_{t,j(t,n)}(n) - \sum_{n=1}^{k} \ell_{t,j(t,n)}(n) \tag{4.9}$$

Having dimensionless track hypotheses scores, the score of a sensor global hypothesis can be computed by summing the component track hypotheses scores. This is justified by assuming independent probabilities of tracks corresponding to different targets and therefore the probability of a global hypothesis is the joint probability of the independent track hypotheses components. As the density of false alarms is equal for all track hypotheses in a given global hypotheses, the likelihood ratio can be used and the summation results from taking the log of the joint likelihood ratio. The track-oriented MHT was shown in (Bar-Shalom *et al.*, 2007) to be equivalent to the hypothesis-oriented MHT if the density of new targets is taken equal to zero and all false alarms are considered as potential new targets. Therefore, the probability of a track-oriented global hypothesis is equal to the probability of its counterpart hypothesis oriented MHT (with the density of new targets equal to zero). The score is obtained from the adjusted equation (4.2) to account for the density of new targets into the density of false alarms

$$P\left(\Theta_s^{k,l_s(k)}|Z^k\right) = \frac{1}{c}\prod_{j=1}^{m(k)}\left\{\frac{f_{tj}}{\lambda_{FA}}\right\}^{\tau_j}\prod\left[[P_{Dt}(k)]^{\delta_t}\left[1-P_{D_t}(k)\right]^{1-\delta_t}\right]\cdot P\left(\Theta_s^{k-1,l_s(k-1)}|Z^{k-1}\right) \tag{4.10}$$

with $P\left(\Theta_s^{k,l_s(k)}|Z^k\right)$ the probability of sensor s global hypothesis $l_s(k)$ at time k and other terms entering the equation described in subsection 4.2.1. For each sensor global hypothesis, the negative logarithm of its probability, computed following equation (4.10) is used as its score in the selection of m-best sensor global hypotheses (followed by the selection of the N-best sensor hypotheses out of the $m \times N$ obtained hypotheses).

4.3 MHT Fusion

At each time k, all sensors send to the fusion center their P-best ranked global hypotheses. The sensor global hypotheses association at the fusion center is described in the next subsection. The estimated fused tracks are obtained from the set of track hypotheses components of each sensor global hypothesis associated.

4.3.1 Sensor Global Hypotheses Association

A sensor global hypothesis, as previously described, consists of a set of track hypotheses. At the given fusion time k, these track hypotheses are represented by the corresponding set of track estimates at time k. For $S = 2$ sensors, each is considered participating with P_s hypotheses into the fusion process. The cost of associating any two hypotheses for these two sensors is introduced as the distance between their track hypotheses sets. The distance between such two sets (two global hypotheses) is evaluated as the cost of the best 2D track-to-track association of their track hypotheses (which are the elements of the two sets). This type of distance was proposed as the multitarget miss distance for multitarget tracking performance assessment in (Mahler, 2004). Here it is proposed as a measure of sensor global hypotheses closeness, as every sensor global hypothesis is a set (with possibly different number) of tracks. Therefore the cost of sensor global hypothesis to sensor global hypothesis association is computed as (Danu *et al.*, 2008b)

$$\mathrm{cost}\left(\Theta_1^{\mathrm{k},\mathrm{l}_1(\mathrm{k})},\Theta_2^{\mathrm{k},\mathrm{l}_2(\mathrm{k})}\right)=\min_{\chi}\sum_{\mathrm{l}_1(\mathrm{k})}\sum_{\mathrm{l}_2(\mathrm{k})}\chi_{\mathrm{l}_1(\mathrm{k}),\mathrm{l}_2(\mathrm{k})}\cdot\mathrm{cost}\left(\hat{\mathbf{x}}_1^{\mathrm{l}_1(\mathrm{k})}(\mathrm{k}),\hat{\mathbf{x}}_2^{\mathrm{l}_2(\mathrm{k})}(\mathrm{k})\right) \tag{4.11}$$

where $\chi_{l_1(k),l_2(k)}$ represents the binary assignment commonly used in the $2D$ assignment to impose the appropriate constraints between tracks selection in both hypotheses:

$$\sum_{l_1(k)=0}\chi_{l_1(k),l_2(k1)}=1, \qquad l_2=1,2,\ldots,L_2(k) \tag{4.12}$$

$$\sum_{l_2(k)=0}\chi_{l_1(k),l_2(k1)}=1, \qquad l_1=1,2,\ldots,L_1(k) \tag{4.13}$$

The track hypotheses labels $l_s(k)=0$, $s=1,2$ represent the dummy track (e.g. association of a track to an unexisting track in the other set). For $S>2$ sensors, following the same approach, the distance between S sensor global hypotheses is introduced as the cost of best $S-D$ assignment between their component S sets of track hypotheses:

$$\mathrm{cost}\left(\Theta_1^{k,l_1},\ldots,\Theta_S^{k,l_S}\right)=\min_{\chi_{1,\ldots,S}^{l_1,\ldots,l_S}}\sum_{l_1}\sum_{l_2}\chi_{1,\ldots,S}^{l_1,\ldots,l_S}\cdot\mathrm{cost}\left(\hat{\mathbf{x}}_1^{l_1},\ldots,\hat{\mathbf{x}}_S^{l_S}\right) \tag{4.14}$$

where in the tracks labels the time index k was dropped in the rhs. The cost of track-to-track association entering both equations (4.11) and (4.14) is computed using the general likelihood ratio for a group of tracks, introduced in (Bar-Shalom *et al.*, 2007). Having P_s sensor global hypotheses received at the fusion center from each sensor s, there are $A=\prod_{s=1}^{S}P_s$ possible associations of sensor global hypotheses. Once the costs of these associations are computed, as detailed above, through running A S-D associations, the best Q associations are chosen through ranking these A costs. The selection of best associations through ranking is possible because all sensor global hypotheses are considered feasible. A more constrained association at the fusion center could allow as feasible sensor global hypotheses associations only the offspring hypotheses of previously Q-best associated sensor hypotheses at the previous fusion time. These Q-best associated hypotheses are used also in the feedback, as detailed in section 4.3.3.

4.3.2 Sensor Global Hypotheses Fusion

In order to find the best fused estimate, the tracks hypotheses of the best associated sensor global hypotheses are combined, using an ML approach (Chen *et al.*, 2003).

4.3.3 Fusion Feedback Usage at Local Trackers

The feedback from fusion, sent back to a local MHT estimator (sensor), is the ranking obtained (Q-best) at the fusion center (FC) for the P-best sensor (local) global hypotheses that were sent by the corresponding sensor to the FC. Usage of this FC ranking of sensor global hypotheses is shown in Fig.4.4. The local MHT uses the N_1-best (out of the Q-best received) local hypotheses and locks (resolves them) for further local propagation. The remaining places (up to N, i.e. $N - N_1$) are filled by the best (locally scored) sensor global hypotheses, yet unused.

Sensor Global Hypotheses Compatibility

One restriction mentioned in (Chong *et al.*, 1990) that could be applied to limit the number of global hypotheses is to consider only combinations of sensor global hypotheses of which predecessors have also been associated. For example, if two associated hypotheses (at FC) are sent back to local sensors for further propagation, offspring hypotheses generated on one sensor from one of these hypotheses cannot be associated (in a next time frame, at FC) with offspring hypotheses generated on another sensor from other parent hypothesis. The rationale is because hypotheses represent mutually exclusive explanations of measurements origins for a given set of measurements. However, as local sensors are supposed to run under measurement origin uncertainty (which uncertainty basically generates the multiple hypotheses) a track swap might happen at any time at a local sensor, which results in measurements from one track to be used in another, therefore resulting in "hypothesis swap". Therefore, once two hypotheses of different fused hypotheses as parents happen to be selected in a best global hypothesis, this might be an indication of track swap at local sensor and can be used for track identity preservation at local and fusion centers.

4.4 Simulation

The simulation scenario for the method proposed uses $S = 3$ sensors and three closed targets, such that the local trackers run in measurement uncertainty conditions. Sample parameters used in the sensor global hypotheses generation, selection and fusion are:

$N = 3$ - at each new measurement frame, only three sensor global hypotheses are propagated

$m = 3$ - from each sensor global hypothesis, the best offspring hypotheses are found at each measurement frame NOTE: above parameters result in obtaining $N \times$

$m = 9$ hypotheses as intermediary hypotheses, out of which the 3-best are selected, as shown in Fig.4.2.

$P = 4$ - out of the $m \times N$ hypotheses obtained, the 4-best are communicated to the fusion center (FC)

$Q = 1$ - only the best sensor global hypothesis fused is indicated through feedback by the FC

$N1 = 1$ - the indicated hypothesis by the FC is forced for further propagation, independently of local ranking

$N2 = 2$ - two remaining places ($N_2 = N - N_1$) for hypotheses to be propagated further are selected through local scoring from the (already locally ranked) P-best local hypotheses

The true three targets trajectories, starting from the locations (0, 86.6)km, (0.4, 86.6)km and (0.8, 86.6km) and moving with velocities of 300m/s (undergoing two coordinated turns) are shown in Fig.4.5, sampled at $T = 2$ sec. The three sensors are positioned at S_1 (0, 187)km, S_2 (50, 0)km, S_3 (-50, 0)km.

Sensors resolutions are identical, with 2mrad in angle and 50m in range, therefore local trackers are running under measurement uncertainty conditions. Fusion is performed every 6 seconds. Sample hypotheses inheritance within the local MHT estimators at sensors 1 and 2 are shown in Fig.4.6. Comparison of local estimators errors with fused estimates errors are shown in Fig.4.7 and Fig.4.8 for all targets at sensors 2 and 3.

4.5 Conclusions

A method for fusing hypotheses obtained by local MHT estimators was proposed, including an association method for sensor global hypotheses through the usage of the multitarget miss distance. Usage of fusion results, obtained at local estimators through feedback in terms of local hypotheses ranking, was proposed in the selection of sensor global hypotheses to be propagated further. The feedback consists only in selection (ranking) of hypotheses obtained locally, therefore the cross-correlation of local estimates (represented by track hypotheses) over time is minimal. The method proposed does not require that all estimators (sensors) participating in the distributed fusion run MHT. For example estimators that do not run MHT still can send their current tracks to the fusion center where they are considered as a single track hypothesis.

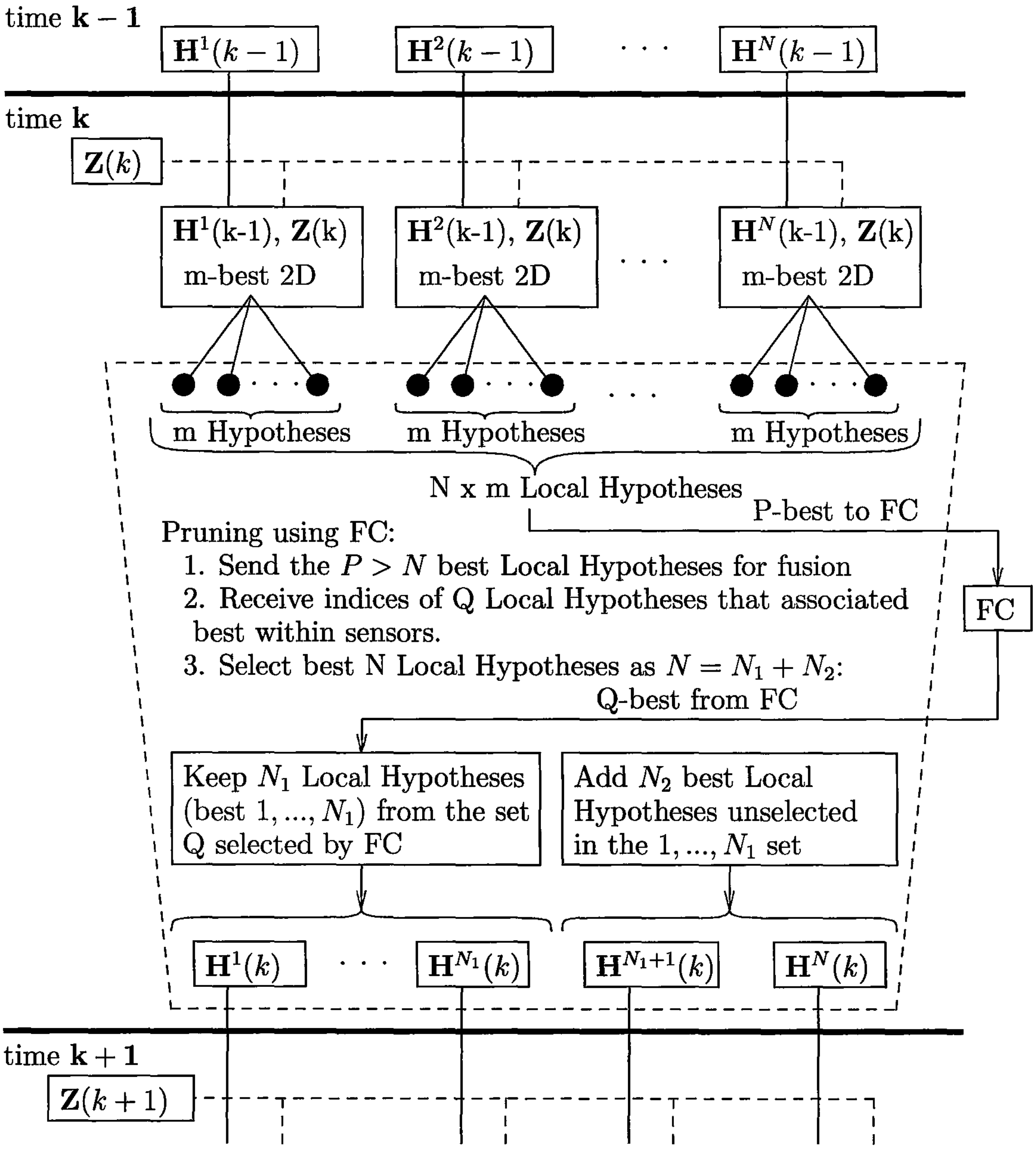

time $\mathbf{k}-\mathbf{1}$
$\mathbf{H}^1(k-1)$
$\mathbf{H}^2(k-1)$
...
$\mathbf{H}^N(k-1)$
time $\mathbf{k}$
$\mathbf{Z}(k)$
$\mathbf{H}^1$(k-1), $\mathbf{Z}$(k) m-best 2D
$\mathbf{H}^2$(k-1), $\mathbf{Z}$(k) m-best 2D
$\mathbf{H}^N$(k-1), $\mathbf{Z}$(k) m-best 2D
m Hypotheses
m Hypotheses
m Hypotheses
N x m Local Hypotheses
P-best to FC
Pruning using FC:
1. Send the $P > N$ best Local Hypotheses for fusion
2. Receive indices of Q Local Hypotheses that associated best within sensors.
3. Select best N Local Hypotheses as $N = N_1 + N_2$:
FC
Q-best from FC
Keep N_1 Local Hypotheses (best $1, ..., N_1$) from the set Q selected by FC
Add N_2 best Local Hypotheses unselected in the $1, ..., N_1$ set
$\mathbf{H}^1(k)$
...
$\mathbf{H}^{N_1}(k)$
$\mathbf{H}^{N_1+1}(k)$
$\mathbf{H}^N(k)$
time $\mathbf{k}+\mathbf{1}$
$\mathbf{Z}(k+1)$

Figure 4.4: Local MHT sensor global hypotheses generation and selection using feedback from the fusion center (FC).

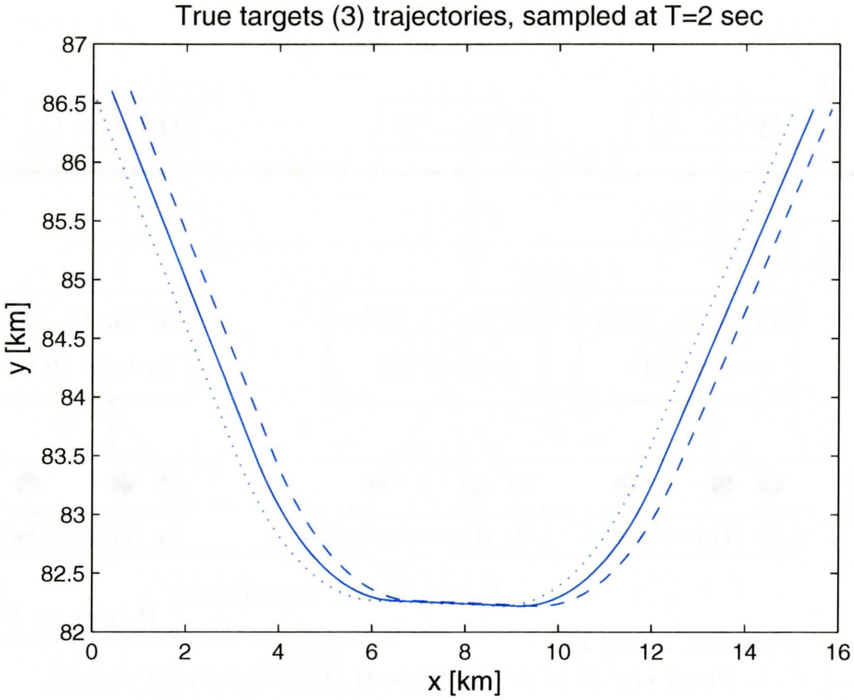
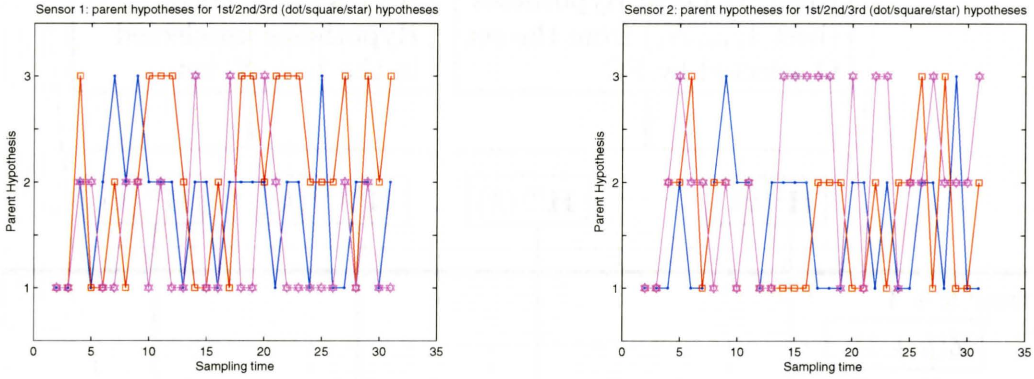


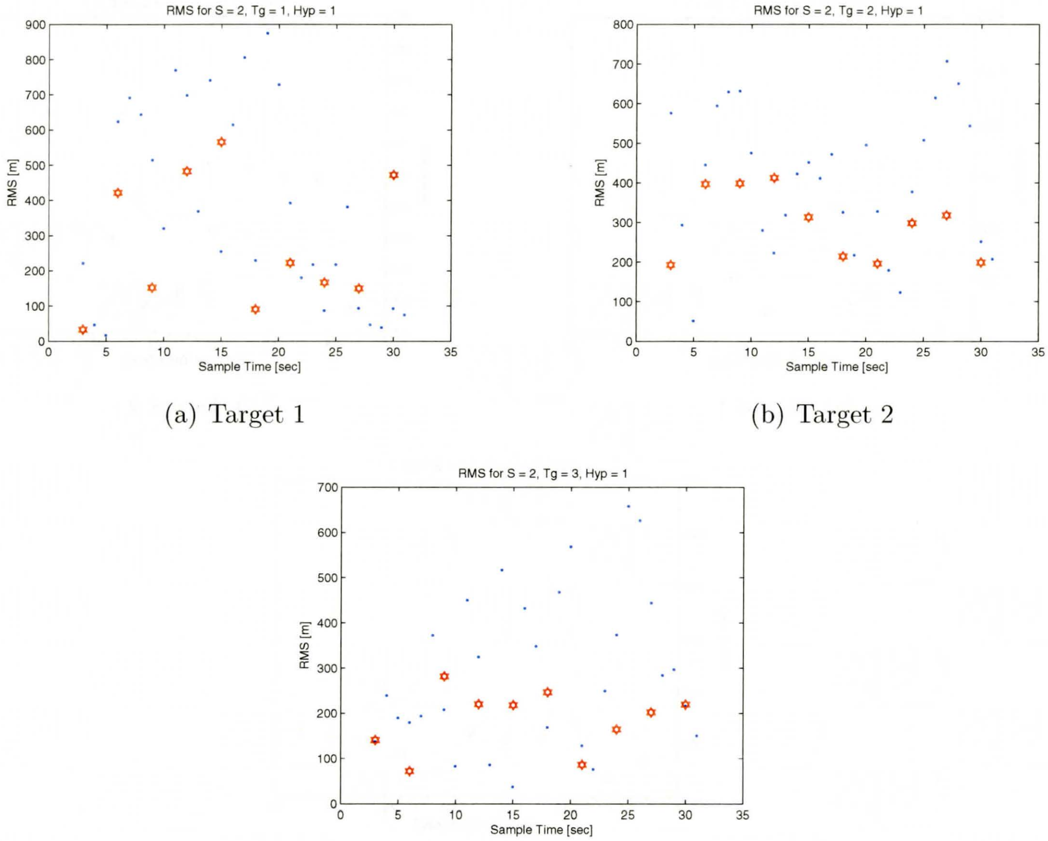
Figure 4.5: True targets trajectories.



(a) Sensor 1

(b) Sensor 2

Figure 4.6: Best ($N = 3$) hypotheses generation (inheritance) at sensors 1 and 2. With dots are 1st hypotheses, with squares are 2nd hypotheses and with stars are 3rd hypotheses at each sensor.

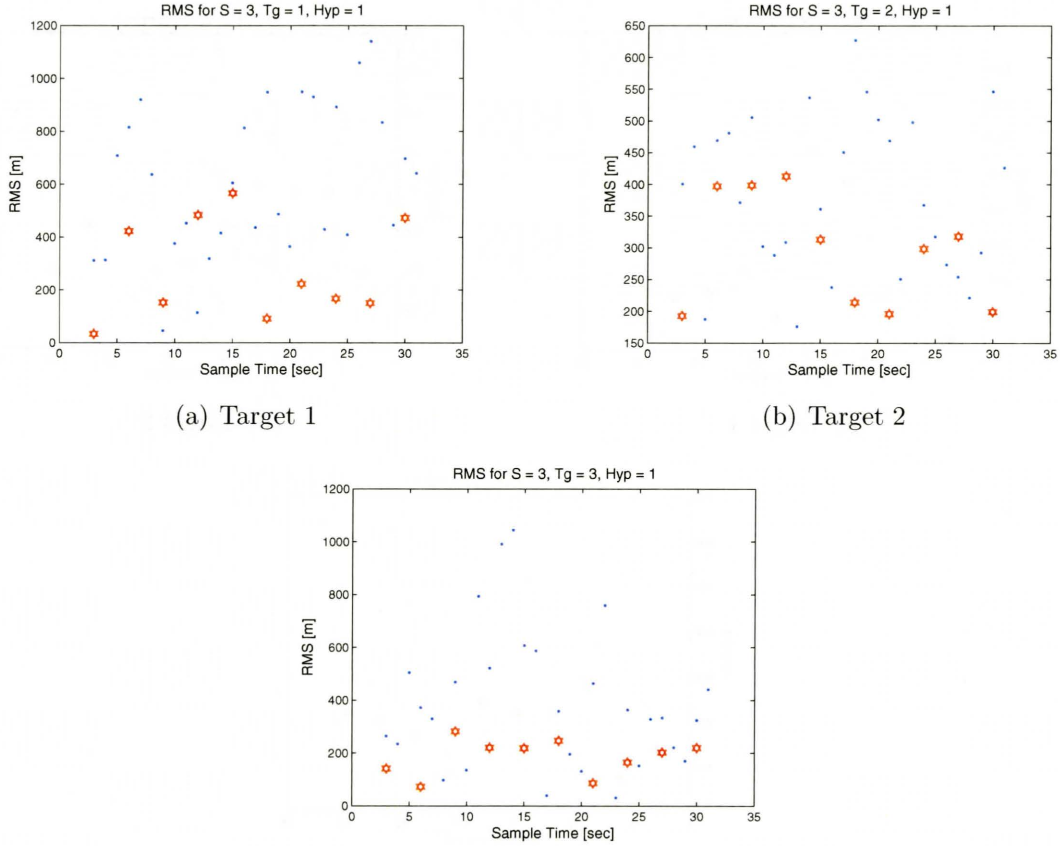


(a) Target 1

(b) Target 2

(c) Target 3

Figure 4.7: Estimate error for a sample ran, obtained at sensor 2 under 1st hypothesis and compared with fusion results (local estimate errors with dots, fused estimates errors with stars).



(a) Target 1

(b) Target 2

(c) Target 3

Figure 4.8: Estimate error for a sample ran, obtained at sensor 3 under 1st hypothesis and compared with fusion results (local estimate errors with dots, fused estimates errors with stars).

Chapter 5

Track Fusion with Asynchronous and Incomplete Data - OTH and AIS Estimators

5.1 Introduction

This chapter presents an application of track fusion to data collected from real-world tracking systems, courtesy of DRDC Ottawa. Track fusion is applied to estimates of Over-the-Horizon (OTH) Radar and Automatic Identification Systems (AIS) for the purpose of obtaining an improved overall maritime picture (Danu *et al.*, 2007a). Over-the-Horizon radar and Automatic Identification System are commonly used in the surveillance of maritime areas. This section introduces a method, which includes tracking and association algorithms, for fusing the information from these two types of systems. Data to be fused consists of *asynchronous* track estimates from the OTH system and measurements obtained from AIS. The data available at the fusion center, as output of real world systems, contained incomplete information, compared to theoretical tracking and fusion algorithms. A method to estimate the missing information in the input data is described. Results obtained using real data as well as simulated data are presented. This type of fusion provides overall pictures of maritime areas, with benefits for surveillance against military threats, as well as threats to exclusive economic zones.

Over-the-Horizon (OTH) radar systems and Automatic Identification System (AIS) are commonly used as stand-alone tracking systems in maritime surveillance. The fusion of the estimates provided by such independent systems, surveying within the same area, was desired for an improved overall maritime picture. The OTH radar covers a static (fixed) area, while the AIS, mounted on an aircraft, changes continuously its coverage area. A description of the OTH and AIS systems in presented in Section 5.2. The track estimates obtained from the real systems, OTH, AIS, provide incomplete information (e.g. incomplete covariance matrices), which cannot be used directly in fusion. Preprocessing of these estimates is described in Section 5.3. For track-to-track association, the M out of N method is used, based on a statistically derived cost, both detailed in Section 5.4. The fusion filter, based on the Kalman filter, is described in Section 5.5. Results achieved on real data, obtained courtesy of DRDC (Defense Research and Development Canada) Ottawa, as well as results obtained using simulated data, are presented in Section 5.6.

5.2 OTH and AIS Systems Description

5.2.1 Over-the-Horizon Radar System Data

The OTH radar system commonly measures the range, angle, and range rate of a detected target. Due to multiple-path reflections and clutter, the sensor has a high uncertainty in measurements. In addition, the sensor fails to detect a target when the target range rate is below the threshold of the motion target indicator (MTI) system. The multipath problem (several detections obtained for the same target) is already solved at the OTH tracker level, this work focuses on fusing the already estimated tracks. However, the environmental uncertainties, and multimode misidentification (Cameron *et al.*, 1996) reflect in the estimated OTH tracks through higher level of incertitude of the track estimate. Maneuvering targets and slowly moving targets may undergo bursts of missed detections. In analyzing and simulating the OTH data, the probability of detection, denoted as PD_{OTH}, the probability of a burst of missed detections, denoted as PM_{burst}, and the length of such burst, denoted as L_{burst}, are considered. The length of a burst of missed detections is equal to the equivalent number of detections that would cover the time interval. The coverage area of the OTH system is fixed; therefore targets entering the OTH surveillance area are consistently detected with the statistics above. The OTH system does not identify the targets through direct communication with them; this information is added through the fusion of the OTH tracks with the AIS data. Thus AIS-OTH fusion converts non-identified OTH tracks into a more informative maritime picture.

5.2.2 Automatic Identification System Data

An AIS commonly integrates a GPS (Global Positioning System) receiver with a standardized transceiver system and other navigational equipment on board ship. The AIS data is usually exchanged between nearby ships and vessel traffic systems, principally for identification of vessels at sea. This data consists of ID, position, course, speed and is used to resolve the problem of identifying ships when not in sight (e.g. at night, in fog, in radar blind arcs or shadows or at distance). The AIS information may become unavailable for certain intervals, e.g. the GPS could lose lock on the required number of satellites by being in the shadow of an obstruction (e.g. mountain, ship superstructure), as detailed in Section 5.2.3 and shown in Fig.5.1. In the real data used, the AIS data was acquired by an AIS receiver on an aircraft that circled the area for a few hours, therefore dynamically changing the coverage region. The precision of such a system is the precision of the GPS, therefore very high and the unique ID provides full target identification. However, ships not equipped with AIS systems, not transmitting AIS data, or out of the coverage area of the AIS receiver are not acquainted for by the AIS receiver. The OTH information adds the tracks not registered by the AIS receiver, as well as track estimates of the AIS registered tracks for periods when the AIS is locked in a blind area. OTH tracks may be considered as skin returns, while the reliability of AIS measurements, with complete target information (unique IDs), may be taken as beacon return (approach close to (Bar-Shalom and Li, 1995)).

5.2.3 Dynamics of the Coverage Area

Sample of the space-time coverage area dynamics of both AIS and OTH systems, estimated through their detected/tracked targets, over a period of more than nine hours is presented in Fig.5.1. During the observed interval the AIS receiver approaches the OTH surveillance area in the first hour, actively overlaps its coverage with the OTH coverage area for one hour, does not receive any information in the following hour, starts receiving again and overlaps its coverage with the OTH coverage area for the next five hours, and finally moves eastward (to the left) out of the OTH coverage region.

5.3 Sensor Data Preprocessing

5.3.1 OTH Data Available for Fusion

In a theoretical track-to-track fusion, the track state $\mathbf{x}(k)$ at time t_k usually includes position (x, y for a 2D estimate in maritime surface surveillance) and the respective velocities

$$\mathbf{x}(k) = [x(k)\, \dot{x}(k)\, y(k)\, \dot{y}(k)] \tag{5.1}$$

The corresponding state estimate covariance matrix $\mathbf{P}(k)$ obtained by the local tracker contains the covariance of each element of the state, as well as all cross-covariances between elements of the state.

The covariance matrix, as providing information on the certainty of the estimate, is essential in performing further fusion of the estimates. In the real-world OTH system considered herein, even though the estimated state is available with position and velocity components, $\mathbf{x}(k) = [x(k)\, v_x(k)\, y(k))\, v_y(k)]'$, due to the fact that from the

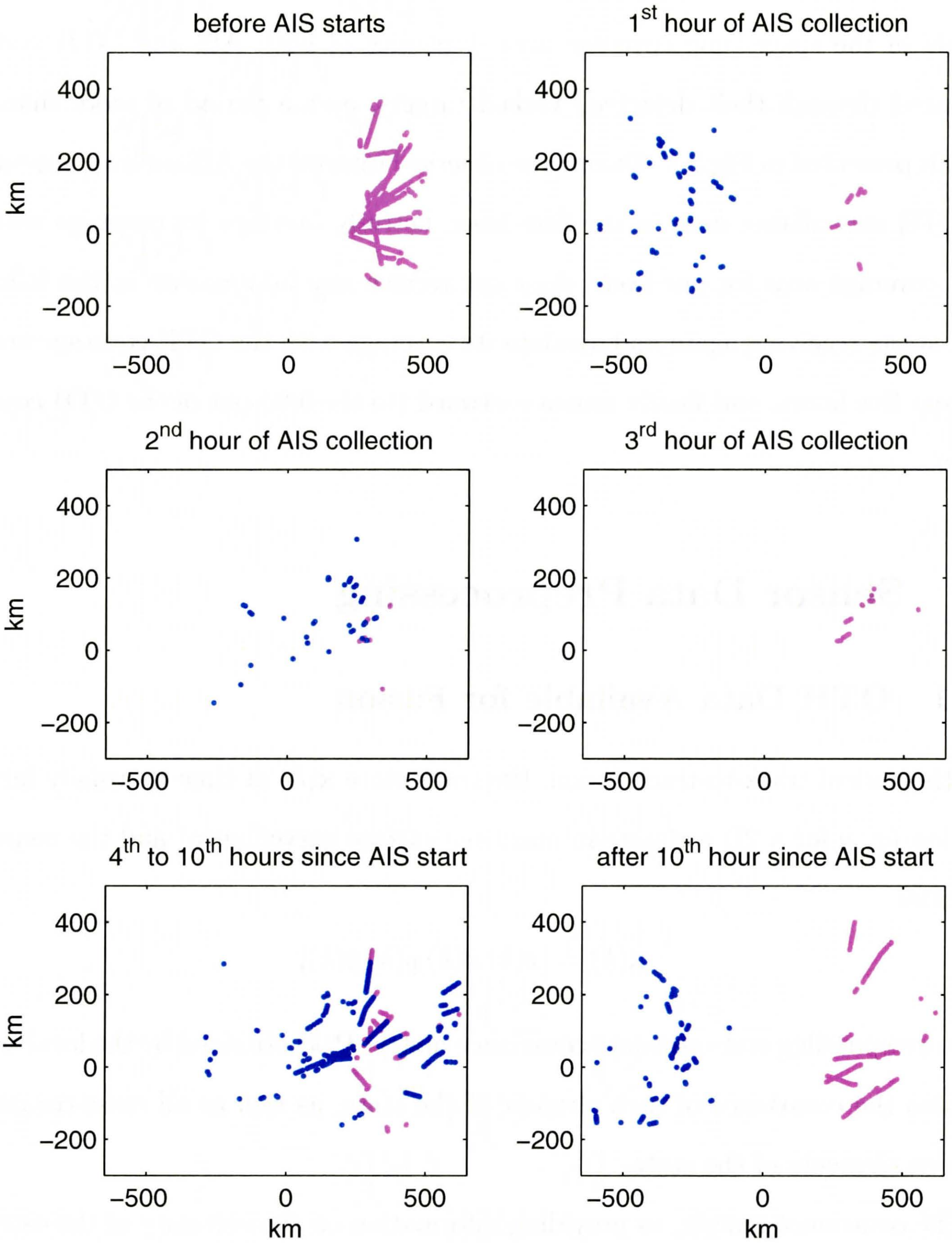


Figure 5.1: Dynamics of the coverage areas of AIS (blue tracks) and OTH (magenta tracks) real systems over a period of more than nine hours.

covariance matrix $\mathbf{P}(k)$ only the terms P_{11}, P_{33}, P_{13} and P_{31} are available, and therefore the uncertainty on the velocity estimates is not known, the velocity information cannot be used. This situation is common for tracking systems that output tracks for displaying purposes and were not designed with the purpose of further fusion, therefore having the covariance matrix partially dropped. A Kalman-type pre-filtering of the OTH estimated tracks with dropped information is performed prior to the fusion with the AIS estimation, described in Section 5.3.3.

5.3.2 AIS Data Available for Fusion

While the AIS system does not provide consistent detections over time (due to AIS locks in blind zones, as well as not receiving data from vessels not equipped with the system), the received data is characterized by accurate position information and confirmed identifier (ID). Whereas the covariance matrix is not provided directly, the terms corresponding to the variances of the position estimates are taken from the variance of the AIS measurement sensor. The cross-covariance between x and y coordinates is considered zero. In order to retrieve the required full state estimate and covariance matrix estimates needed for fusion, a pre-processing of the AIS estimates is performed, similar to the one applied to OTH estimates, and described in Section 5.3.3.

5.3.3 Processing of OTH Tracks and AIS Data Before Fusion

The pre-processing of OTH tracks and AIS data is applied for the purpose of completing the state information and the covariance matrix, up to the first order (velocities). The stages of track initialization and filtering are described below. For OTH, if the initial track data is obtained through a Kalman Filter estimator, the tracks errors are correlated over time (Bar-Shalom *et al.*, 2001). Therefore the direct application of a Kalman Filter on it would violate the assumption of independence on the input (measurements). The method described in (Bar-Shalom and Chen, 2008) can be applied for covariance reconstruction. For the AIS data, as being GPS type data, the Kalman Filter can be used. A pre-whitening filter may be applied on the data, if correlated in time. The track re-initialization for pre-filtering is performed using two data points. The partial covariance matrix available from the OTH (or AIS) system,

$$\mathbf{P}_{OTH}(k) = \begin{bmatrix} P_{xx}(k) & P_{xy}(k) \\ P_{xy}(k) & P_{xx}(k) \end{bmatrix} \tag{5.2}$$

is used to initialize what is the measurement noise in a filtered track:

$$\mathbf{R}_{preOTH}(k) = \mathbf{P}_{OTH}(k) \tag{5.3}$$

The partial state vector available from the OTH system

$$\mathbf{x}_{OTH}(k) = [\hat{x}(k|k) \quad \hat{y}(k|k)] \tag{5.4}$$

$$\mathbf{z}_{preOTH}(k) = \mathbf{x}_{OTH}(k) \tag{5.5}$$

and the pre-processed-OTH state and covariance matrix are computed using the information filter approach (Bar-Shalom *et al.*, 2001). The indices 0 and 1 correspond to the times t_0 and t_1, being the first two estimates of the track to be initialized.

$$\begin{aligned}&\mathbf{x}_{preOTH}(1) = \\ &\left(\left(\mathbf{HF}(0)^{-1}\right)^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)^{-1}\mathbf{HF}(0)^{-1}+\mathbf{H}^T\mathbf{R}(1)^{-1}\mathbf{H}\right)^{-1}\cdot \\ &\left(\left(\mathbf{HF}(0)^{-1}\right)^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)^{-1}\mathbf{z}(1)+\mathbf{H}^T\mathbf{R}(1)^{-1}\mathbf{z}(0)\right)\end{aligned} \tag{5.6}$$

and

$$\begin{aligned}&\mathbf{P}_{preOTH}(1) = \\ &\left(\left(\mathbf{HF}(0)^{-1}\right)^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)^{-1}\mathbf{HF}(0)^{-1}+\mathbf{H}^T\mathbf{R}(1)^{-1}\mathbf{H}\right)^{-1}\cdot \\ &\left(\left(\mathbf{HF}(0)^{-1}\right)^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)^{-1}\mathbf{HF}(0)^{-1}+\mathbf{H}^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)\mathbf{H}\right)\cdot \\ &\left(\left(\mathbf{HF}(0)^{-1}\right)^T\left(\mathbf{R}(0)+\mathbf{HQ}(0)\mathbf{H}^T\right)^{-1}\mathbf{HF}(0)^{-1}+\mathbf{H}^T\mathbf{R}(1)^{-1}\mathbf{H}\right)^{-1}\end{aligned} \tag{5.7}$$

where

$$\begin{aligned}\mathbf{z}(0) &= \mathbf{z}(0)_{preOTH}, \\ \mathbf{z}(1) &= \mathbf{z}(1)_{preOTH},\end{aligned} \tag{5.8}$$

$$\begin{aligned}\mathbf{R}(0) &= \mathbf{R}(0)_{preOTH}, \\ \mathbf{R}(1) &= \mathbf{R}(1)_{preOTH},\end{aligned} \tag{5.9}$$

$$\mathbf{F}(k) = \begin{bmatrix} 1 & T_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10}$$

with T_k being the time interval between two measurements, $T_k = t(k+1)-t(k)$. Using the information filter model for track initialization has the advantage of being able to start the pre-filtered track with non-informative prior for the covariance matrix, as the complete matrix is not available (Bar-Shalom *et al.*, 2001). The process noise covariance matrix is chosen modeled as DCWNA (Discrete Continuous White Noise Acceleration) (Bar-Shalom *et al.*, 2001):

$$\mathbf{Q}(k) = \mathbf{E}\left[\mathbf{G}_k \boldsymbol{v}_k \left(\mathbf{G}_k \boldsymbol{v}_k\right)^T\right], \tag{5.11}$$

resulting in

$$\mathbf{Q}(k) = \mathbf{E}\begin{bmatrix} 1/3T_k^3 & 1/2T_k^2 & 0 & 0 \\ 1/2T_k^2 & T_k & 0 & 0 \\ 0 & 0 & 1/3T_k^3 & 1/2T_k^2 \\ 0 & 0 & 1/2T_k^2 & T_k \end{bmatrix} \cdot \tilde{q}. \tag{5.12}$$

The dynamic state and measurement equations, at times $t(k)$ for $k > 1$ are

$$\mathbf{x}(k+1) = \mathbf{F}(k) \cdot \mathbf{x}(k) + \mathbf{G}(k) \cdot \boldsymbol{v}(k) \tag{5.13}$$

and

$$\mathbf{z}(k) = \mathbf{H}(k) \cdot \mathbf{x}(k) + \mathbf{w}(k), \tag{5.14}$$

respectively, for which the estimation of the complete state $\mathbf{x}(k)$ and covariance matrix $\mathbf{P}(k)$ uses the common Kalman filter (Bar-Shalom *et al.*, 2001). The noise coefficient $\tilde{q}$ is chosen such that the already available data in the state estimate and covariance matrix is not changed significantly by pre-filtering.

5.4 Tracks Association

The processing logic for a newly received AIS data is shown in Fig.5.2. A similar processing is applied at the receipt of an OTH track estimate. At the time an AIS data sample is received there are already formed OTH and AIS tracks, preprocessed as described in Section 5.3, as well as possible AIS data from which a track is not initialized yet (in AIS Initial Data Storage in Fig.5.2). The cost of association of the newly received incomplete AIS estimate with an existing OTH pre-filtered track is computed first as described in Section 5.4.1. If the association between the AIS and OTH track is not declared, then either the AIS data is filtered into an existing AIS track or the AIS data is used to initialize a new track with an existing single AIS measurement of same ID. If the association with an OTH track is declared, the AIS data is fused to the OTH track and further AIS data of same ID is filtered into the fused track.

The association is done in two steps. First each best cost found between tracks of different types is recorded for the given pair of tracks. If M past best costs are found recorded for a pair of AIS and OTH tracks within their list of last N associations, then the second step confirms the association and passes them for fusion. This method is recorded in literature as the M out of N association (Radar Corresponding Group, 2006).

5.4.1 Track Association Cost Definition

Below the cost computation is detailed for associating a newly received AIS data with an existing pre-filtered OTH track. The same procedure is used for a new sample of OTH data. The association cost of an input data to an already pre-filtered track is

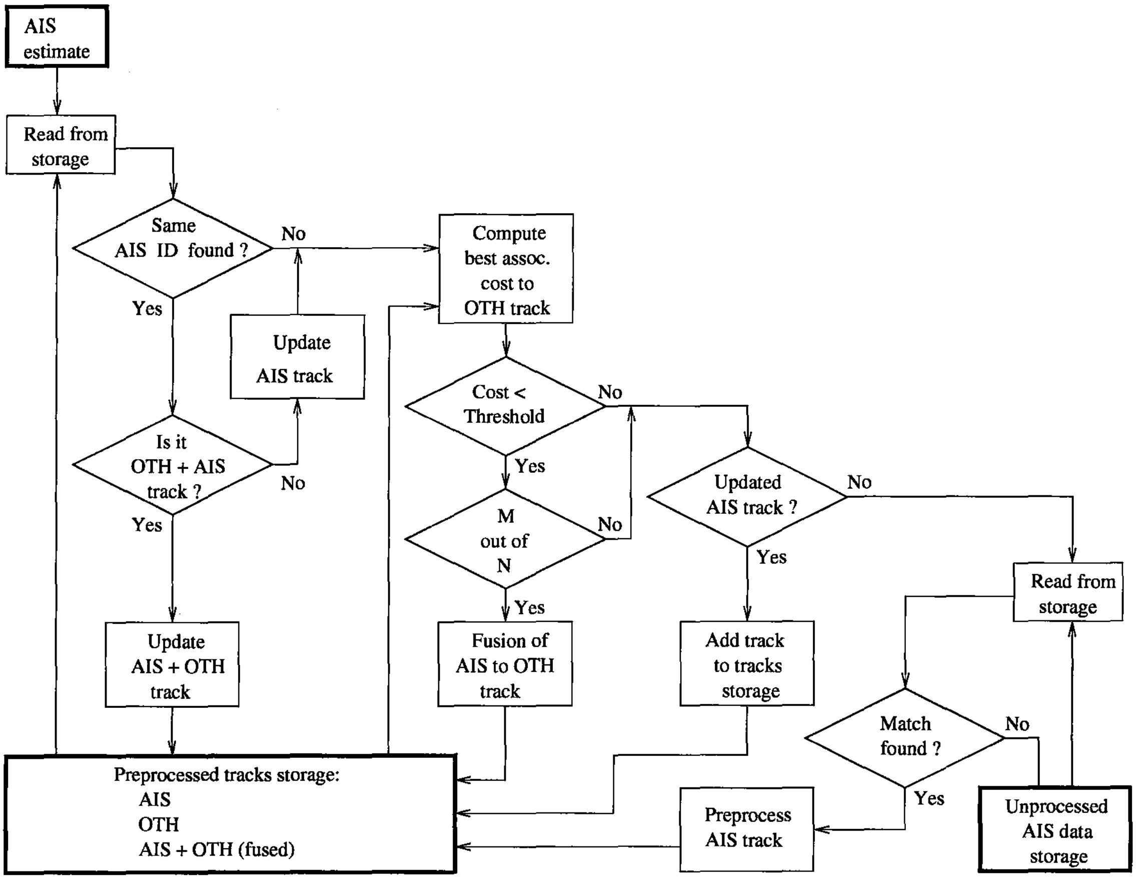


Figure 5.2: Sample association and fusion for AIS estimate.

computed as the negative logarithm of the likelihood of the new data and existing track being sampled from the same target. In order to compute the association cost of the newly received data $\mathbf{z}(k+1)$ at time $t(k+1)$, to the OTH track $\hat{\mathbf{x}}(k)$, last updated at time $t(k)$ (as AIS and OTH systems are asynchronous), the OTH track state needs to be predicted to the association time $t(k+1)$

$$\mathbf{x}(k+1|k) = \mathbf{F}(k) \cdot \mathbf{x}(k) \tag{5.15}$$

where $\mathbf{F}(k)$ is computed using (5.10) with $T_k = t(k+1) - t(k)$. The likelihood of the measurement $\mathbf{z}(k+1)$ being sampled from the same target as the predicted state estimate $\mathbf{x}(k+1|k)$ is the probability density function (pdf) of the measurement conditioned on the predicted state. The association cost is computed as the negative logarithm of this likelihood:

$$\text{cost} = -\ln\left(p\left(\mathbf{z}_{k+1}^{AIS} \,|\, \mathbf{H}\hat{\mathbf{x}}_{k+1|k}^{OTH}\right)\right) \tag{5.16}$$

Using the normal distribution for the conditional pdf above, (5.16) translates into

$$\text{cost} = -\ln\left(\mathcal{N}\left(\mathbf{z}_{k+1}^{AIS} \,|\, \mathbf{H}\hat{\mathbf{x}}_{k+1|k}^{OTH}, \mathbf{S}(k+1)\right)\right) \tag{5.17}$$

with

$$\mathcal{N}\left(\mathbf{z}(k+1)^{AIS} \mid \mathbf{H}\hat{\mathbf{x}}(k+1|k)^{OTH}, \mathbf{S}(k+1)^{OTH,AIS}\right) = \frac{1}{\sqrt{|2\pi \mathbf{S}^{AIS,OTH}(k+1)|}} \cdot \exp\left(\begin{array}{l} -\frac{1}{2} \cdot (\mathbf{z}^{AIS}(k+1) - \mathbf{H}\hat{\mathbf{x}}^{OTH}(k+1|k))^T \cdot \\ \qquad (\mathbf{S}(k+1)^{AIS,OTH})^{-1}(\mathbf{z}^{AIS}(k+1) - \mathbf{H}\hat{\mathbf{x}}^{OTH}(k+1|k)) \end{array} \right) \tag{5.18}$$

which is the normal pdf with mean equal to the predicted state and covariance matrix S. The covariance matrix in (5.16) denotes the covariance matrix of the innovation $\boldsymbol{\nu}(k+1) = \mathbf{z}^{AIS}(k+1) - \mathbf{H} \cdot \mathbf{x}^{OTH}(k+1|k)$, estimated by (Bar-Shalom *et al.*, 2001)

$$\mathbf{S}^{AIS,OTH}(k+1) = \mathbf{H}\mathbf{P}^{OTH}(k+1|k)\mathbf{H}^T + \mathbf{R}^{AIS}(k+1), \tag{5.19}$$

with the OTH predicted covariance matrix of the state

$$\mathbf{P}^{OTH}(k+1|k) = \mathbf{F}(k)\mathbf{P}^{OTH}(k \mid k)\mathbf{F}(k)^T + \mathbf{Q}^{OTH}(k), \tag{5.20}$$

Upper indices were added to indicate the source of the data, when data from both sources are present in the same equation. The position information (available) in the partial covariance matrix of the input data, similar to (5.2), this time for AIS, is used for $\mathbf{R}^{AIS}(k+1)$ in (5.19):

$$\mathbf{R}_{AIS}(k) = \begin{bmatrix} P_{AIS,xx}(k|k) & P_{AIS,xy}(k|k) \\ P_{AIS,xy}(k|k) & P_{AIS,yy}(k|k) \end{bmatrix}. \tag{5.21}$$

5.4.2 M out of N Association Method

For each input data of ID i, the cost detailed in section 5.4.1 is computed for the available tracks $j \in J$ (where J is the set of tracks of other type), as in 5.17

$$\text{cost}(i,j) = -\ln \mathcal{N}\left(\mathbf{z}^i(k+1) \mid \mathbf{H}\hat{\mathbf{x}}^j(k+1|k),\, \mathbf{S}^{i,j}(k+1)\right) \tag{5.22}$$

The track selected for the recorded association event to this input data $\mathbf{z}^i(k+1)$ is

$$j_{best} = \min_{j \in J} \mathrm{c}(i,j) \tag{5.23}$$

and the event is recorded to the track combination (i,j_{best}). Only the last N such best recorded combinations are stored for each track combination. Once a pair of tracks (i,j) is found having M combinations together out of the last N ones, the association is confirmed and the tracks will be fused. A gating, which restricts the number of tracks that enter the association, based on the input data position, currently available pre-filtered tracks positions, input (incomplete) covariance matrices and maximum velocities is applied first, for both x and y directions, is detailed below for x:

$$\left|z_x^i(k+1) - \hat{x}_x(k+1|k)\right| < (t_{k+1} - t_k)\mathbf{v}_{max} + \alpha\left(\sqrt{R_{11}^i} + \sqrt{P_{11}^j}\right) \tag{5.24}$$

where $\mathbf{v}_{max}$ is the maximum velocity allowed for a target and α is a constant chosen.

5.5 OTH Tracks to AIS Information Fusion

After the input data passes the M out of N association with an existing track, the input data is fused with the track of different type. The estimate of the fused track is computed using the update equation of the Kalman filtering.

5.5.1 Fused State Update Using Kalman Filter

Based on the new input data available $\mathbf{z}(k+1)$, equations of predicted track state in (5.15), predicted covariance matrix in (5.20) and innovation covariance matrix in (5.19), the updated state of the fused track is computed as

$$\hat{\mathbf{x}}^{i,j}(k+1) = \hat{\mathbf{x}}^{j}(k+1|k) + \mathbf{W}^{i,j}(k+1) \cdot \left(\mathbf{z}^{i}(k+1) - \mathbf{H}\mathbf{x}^{j}(k+1|k)\right) \tag{5.25}$$

where the combined gain matrix is computed as

$$\mathbf{W}^{i,j}(k+1) = \mathbf{P}^{i}(k+1|k) \cdot \mathbf{H}^{T} \cdot \left(\mathbf{S}^{i,j}(k+1)\right)^{-1} \tag{5.26}$$

The Kalman filter is optimal under the Gaussian-Markov assumption (white, Gaussian noises and initial state, and Markov process). The error of the input AIS and OTH estimates can be assumed as white. The independence of AIS and OTH errors is assumed based on their estimation from different sensors and estimators. The OTH estimator (tracker) uses a dynamic state equation with process noise to approximate the target state, while the AIS uses the (differential) GPS receiver in estimating the highly accurate position. Therefore, their error can be assumed as uncorrelated and, furthermore, an AIS input data and an OTH pre-filtered track can be assumed as having uncorrelated errors. However, as AIS and OTH pre-filtering are based on similar dynamic equations, a correlation of errors is present for an OTH input data and an AIS pre-filtered track. Based on the AIS higher precision, however, the correlation is considered to be insignificant. Therefore, the Kalman filter can be used with good performance for filtering one type of estimate with another.

5.6 Simulation

5.6.1 Simulated Data

The models of the simulated OTH and AIS tracks were developed such that they closely replicate real OTH and AIS tracks, provided courtesy of DRDC (Defense Research Development Canada) in Ottawa. The statistics of the OTH tracks are modeled through the normal probability of detection, $PD_{OTH} = 0.85$, the probability of a burst of missed detections, $PM_{burst} = 0.05$, and the Poisson distributed length of such burst, $L_{burst}(\lambda)$, with $\lambda = 8$, described in section 5.2.1. The length of a burst of missed detections is equal to the equivalent number of detections that would cover the time interval. Initial simulated measurements are considered in ground coordinates (i.e., after mapping the measurements in radar coordinates into ground coordinates). Combination of OTH tracks resulting from multipath propagation is considered already performed at the OTH tracker level, therefore each target is represented by only one track as input for fusion. The simulated track estimates for AIS and OTH systems are generated starting from OTH-type radar measurements in polar coordinates (range, angle and range rate)

$$\mathbf{z}_{polar} = [\ r \quad \theta \quad \dot{r}\ ]^T \tag{5.27}$$

where r is the target radial measurement, $\dot{r}$ is the target radial rate measurement and θ is the angular target position measurement. For the measurement covariance matrix

$$\mathbf{R}_{polar} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_{\dot{r}}^2 \end{bmatrix} \tag{5.28}$$

the standard deviations σ_r, σ_θ, and $\sigma_{\dot{r}}$ were set to 1000 m, 0.01 rad and 0.8 m/s, respectively. The standard polar-to-Cartesian translation with approximate terms based on linearization (Bar-Shalom *et al.*, 2001; Bar-Shalom and Li, 1995) is used

$$\begin{aligned} x(k) &= r(k) \cdot \cos(\theta(k)) \\ y(k) &= r(k) \cdot \sin(\theta(k)). \end{aligned} \tag{5.29}$$

The terms of the resulting covariance matrix in Cartesian coordinates are (Bar-Shalom *et al.*, 2001)

$$\begin{aligned} R_{xx}(k) &= r(k)^2 \cdot \sigma_\theta^2 \cdot \sin(\theta(k))^2 + \sigma_r^2 \cdot \cos\left(\theta(k)\right)^2 \\ R_{yy}(k) &= r(k)^2 \cdot \sigma_\theta^2 \cdot \cos(\theta(k))^2 + \sigma_r^2 \cdot \sin\left(\theta(k)\right)^2 \\ R_{xy}(k) &= (\sigma_r^2 - r(k)^2 \cdot \sigma_\theta^2) \cdot \sin\left(\theta(k)\right) \cdot \cos\left(\theta(k)\right). \end{aligned} \tag{5.30}$$

For the simulated sensors precision, the above standard conversion does not generate large bias errors (Bar-Shalom and Li, 1995). The dynamic model used to simulate the OTH tracks is based on the Kalman filter, with plant equation

$$\mathbf{x}(k+1) = \mathbf{F}(k) \cdot \mathbf{x}(k) + \mathbf{G}(k) \cdot \boldsymbol{v}(k) \tag{5.31}$$

and measurement equation

$$\mathbf{z}(k) = \mathbf{H}(k) \cdot \mathbf{x}(k) + \mathbf{w}(k). \tag{5.32}$$

where $\mathbf{x}_k$, $\mathbf{x}_{k+1}$ are the states (at discrete sampling times t_k, respectively t_{k+1}) of the track in Cartesian coordinates, $\mathbf{F}(k)$ is the transition matrix, $\mathbf{G}(k)$, $\boldsymbol{v}(k)$ represent the process noise, $\mathbf{H}$ is the measurement matrix and $\mathbf{w}(k)$ is the measurement noise. The resulting state vector is modeled as in (5.1) and the state transition matrix $\mathbf{F}(k)$ has the form in (5.10). The process noise is modeled using the DCWNA model in (5.11) and (5.12) with being the power spectral density of the process noise (Bar-Shalom *et al.*, 2001). The measurement vector $\mathbf{z}(k)$, with components transformed in Cartesian coordinates (5.29)-(5.30), the matrix $\mathbf{H}(k)$, and measurement noise vector $\mathbf{w}(k)$ are used in the forms (Sinha *et al.*, 2005)

$$\mathbf{z}(k) = \begin{bmatrix} \hat{x}(k) & \hat{y}(k) & \hat{\dot{r}}(k) \end{bmatrix}^T \tag{5.33}$$

$$\mathbf{H}(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \cos(\theta(k)) & 0 & \sin(\theta(k)) \end{bmatrix} \tag{5.34}$$

and

$$\mathbf{w}(k) = [\ w_x(k) \quad w_y(k) \quad w_{\dot{r}}(k)\ ]^T \tag{5.35}$$

respectively. Based on (5.33) and (5.35) the measurement noise covariance matrix has the form

$$\mathbf{R}(k) = \begin{bmatrix} R_{xx}(k) & R_{xy}(k) & 0 \\ R_{xy}(k) & R_{xx}(k) & 0 \\ 0 & 0 & \sigma_{\dot{r}}^2 \end{bmatrix}. \tag{5.36}$$

Each generated track is tagged with identifier (ID), which for the AIS completely identifies a target, while for OTH it does not (e.g. a single target for which track is

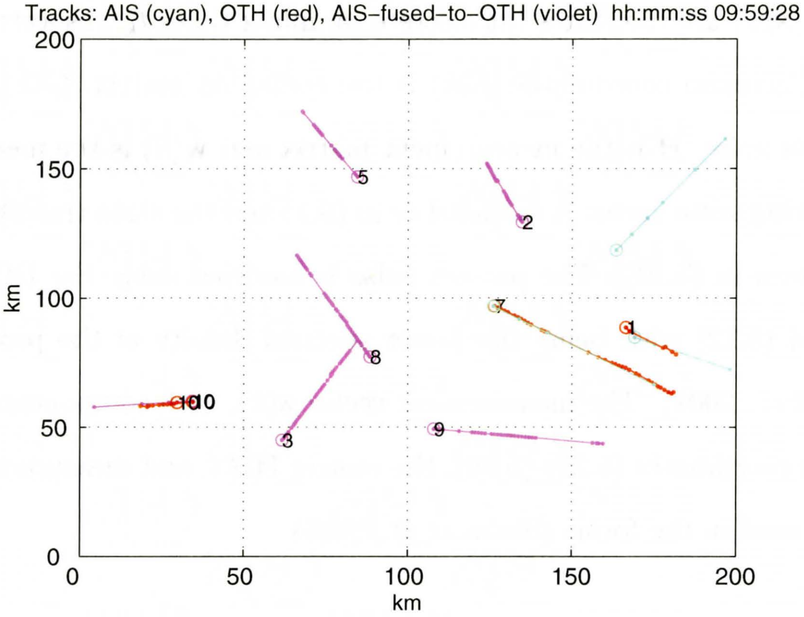


Figure 5.3: Simulated AIS (cyan), OTH (red) and fused (violet) tracks, during processing, over a time window interval.

lost and restarted might have different OTH IDs at different times). For the M out of N association method, the values $M = 3$ and $N = 4$ were used. Results obtained on simulated data, containing 10, respectively 50 targets, are presented in the next figures. In Fig.5.3 the tracks of 10 targets are shown during the processing, over a given window interval. While in Fig.5.4 the final results, with full tracks, over the whole simulated interval (around 20 hours) are displayed. The end of each track is marked with a circle, while the first fusion time of a pair of two tracks is marked with a star. For the simulation with 10 targets in Fig.5.3 and Fig. 5.4, all the AIS-OTH pairs from the same target are properly fused. For the simulation with 50 targets, shown in Fig.5.5 and Fig.5.6, there are 45 pairs of tracks properly fused. Some of the AIS tracks do not have enough measurements within the OTH area (3 out of the last 4 are required for fusion). All the fused tracks have the AIS-OTH pair properly

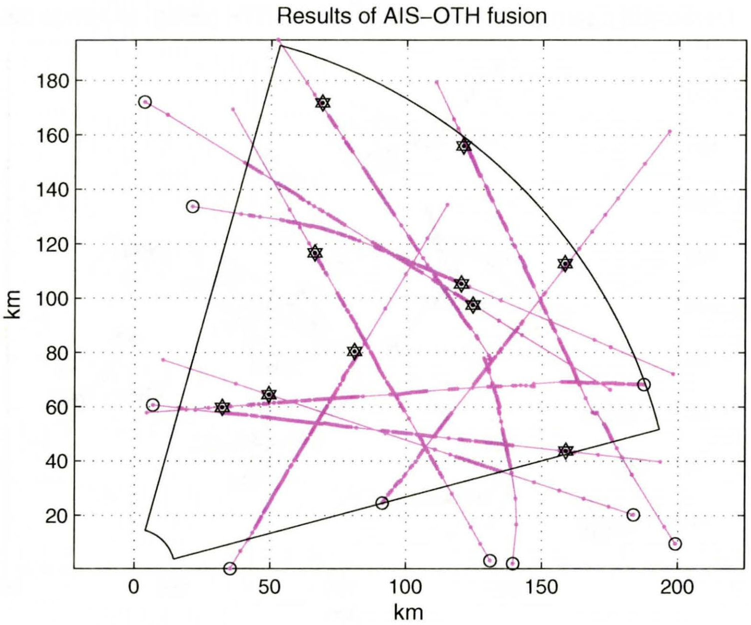


Figure 5.4: Results for 10 simulated targets within the OTH surveillance region. All tracks are fused, circles mark track ends and stars mark first fusion times.

matched (no swaps).

Real Data

Sample of the results obtained on real data with $M = 3$ and $N = 4$ in the association method, are presented next. In Fig.5.7 a snapshot of time is displayed, while in Fig. 5.8 the whole history of tracks, fused or not, is presented. All the overlapping AIS-OTH tracks are properly fused. For the OTH-AIS pair in Fig.5.8 not fused, and which seem to be from same target, the AIS receiver does not receive data while the target is tracked within the OTH surveillance region.

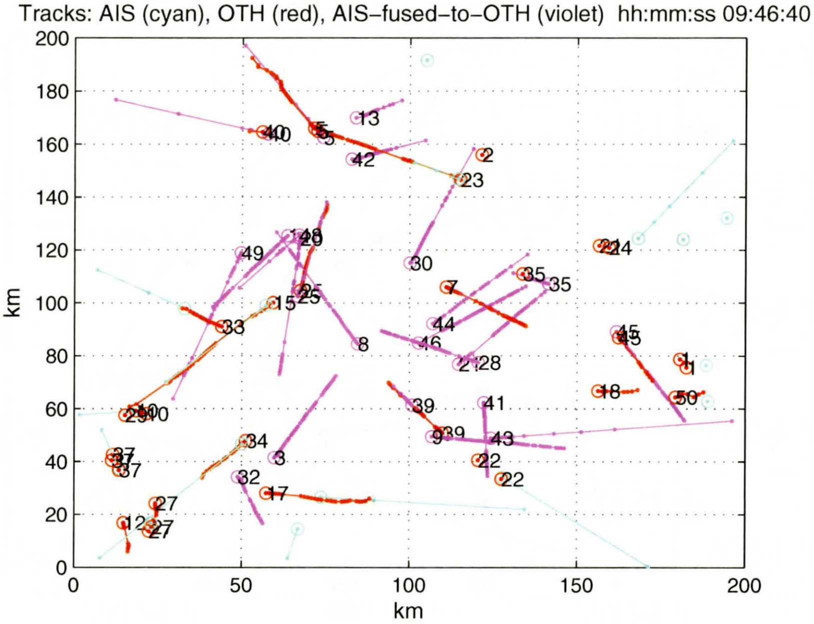


Figure 5.5: Simulated AIS (light blue), OTH (red) and fused (violet) tracks, during processing, over a time window interval.

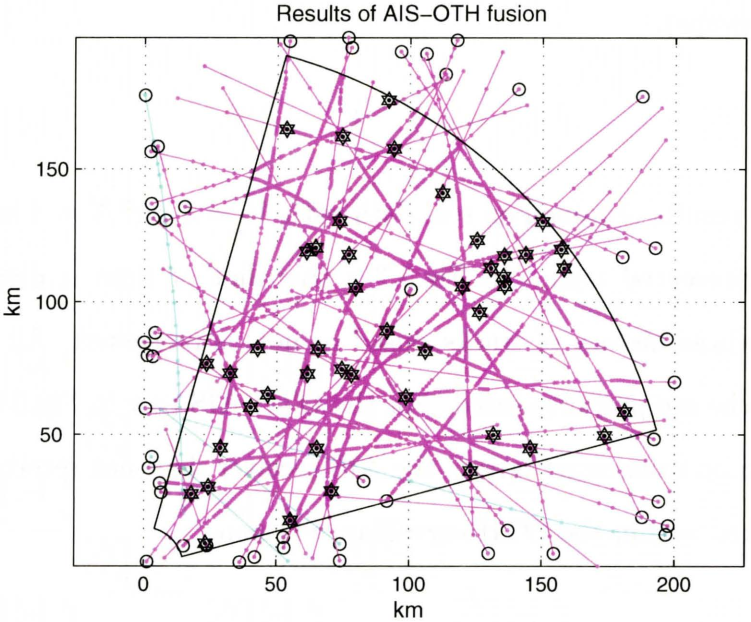


Figure 5.6: Final fusion results for 50 simulated targets within the surveillance region. Fused tracks are violet and non-fused AIS measurements are light blue.

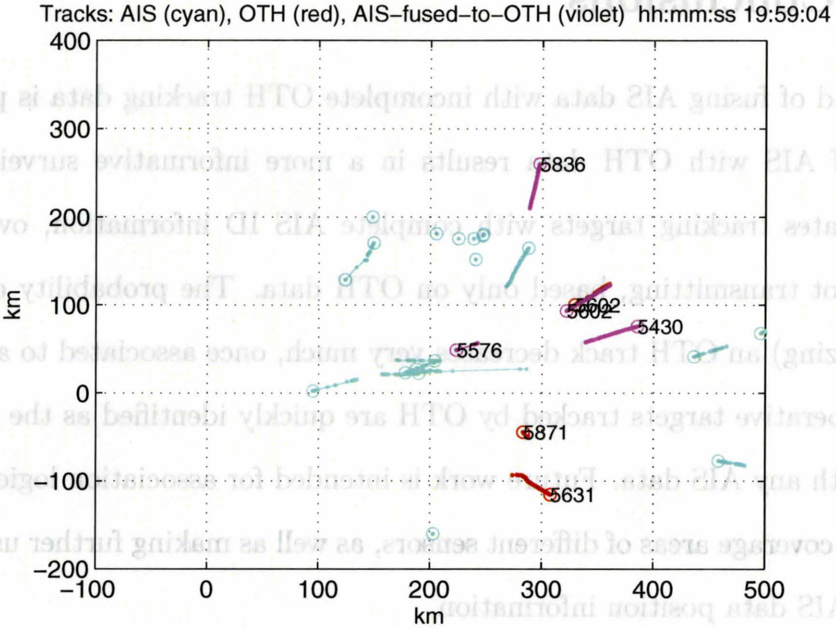


Figure 5.7: Real data AIS (blue), OTH (red) and fused (violet) tracks, during processing, over a time window interval.

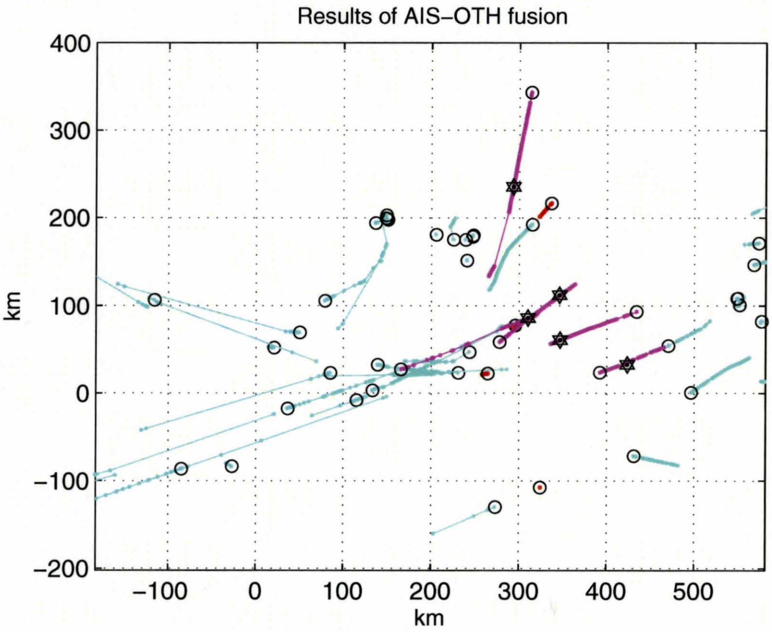


Figure 5.8: Results obtained with real data. Fused tracks are violet, OTH non-fused tracks are red and AIS non-fused are cyan. First fusion times are marked with stars.

5.7 Conclusions

A method of fusing AIS data with incomplete OTH tracking data is presented. The fusion of AIS with OTH data results in a more informative surveillance picture. It facilitates tracking targets with complete AIS ID information, over periods the AIS is not transmitting, based only on OTH data. The probability of loosing (and reinitializing) an OTH track decreases very much, once associated to an AIS ID. The non-cooperative targets tracked by OTH are quickly identified as the ones not being fused with any AIS data. Future work is intended for association logic at boundaries between coverage areas of different sensors, as well as making further use of the highly precise AIS data position information.

Chapter 6

Conclusion

This thesis introduces two novel methods of track fusion for multisensor tracking systems. The first method, presented in Chapter 2 associates and fuses local estimates provided under the form of particle clouds. The second method, presented in Chapter 3 introduces the usage of the prior information from the previous fusion time in the track-to-track association of the current time. This method was devised for local estimators providing tracks under the form of gaussian or gaussian mixture estimates, i.e. MMSE or LMMSE. The local tracks are given as mean estimate and covariance matrix or mean estimate and MSE matrix for the second method. This thesis also introduces in Chapter 4 a novel method of using the fused tracks in the pruning the local hypotheses in multisensor MHT tracking. A sample track fusion application to real data that uses asynchronous sensors, of different types, and with incomplete local covariance matrices is presented in Chapter 5. Novel particle labeling schemes for the PHD particle filter were introduced, presented in Appendix A, for the simulation and assessment of the first method of fusion.

The method for the association and fusion of target posterior densities estimates obtained using particle filters was proposed in Chapter 2. Cloud-to-cloud association costs, computed at the particle level, depending on particles types were derived and the convergence of the cost on particle clouds was proved theoretically and tested through simulations. Several applicable fusion methods were presented. Results obtained on a simulated multitarget scenario using the proposed association cost and fusion method were analyzed and showed improvement compared with the classical track-to-track test when applied on particle filter estimates. The fusion method introduced, at particle level, combines the whole information contained in the estimated targets posterior densities, not only the tracks states first order moments, therefore being applicable to highly non-Gaussian estimates and nonlinear estimators.

The novelty of the track fusion method introduced in Chapter 3 is the preservation of the fused tracks identities over time. It also shows possible the recovery of targets identities at the fusion center in the case of local tracks swap. This fusion method keeps and propagates identities of the estimated targets (tracks) at the fusion center, beside the ones at the local sensors. The identities of the fused tracks, as well as their estimates at the previous fusion frame are used in the track-to-track association of the current fusion frame.

A novel method for fusing track hypotheses obtained by local MHT estimators was introduced in Chapter 4, including an association method for sensor global hypotheses through the usage of the multitarget miss distance. Usage of fusion results, obtained at local estimators through feedback in terms of local hypotheses ranking, was proposed in the selection of sensor global hypotheses to be propagated further. The feedback consists only in selection (ranking) of hypotheses obtained locally, therefore the cross-correlation of local estimates (represented by track hypotheses) over time is minimal. The method proposed does not require that all estimators (sensors) participating in the distributed fusion run MHT. For example estimators that do not run MHT still can send their current tracks to the fusion center where they are considered as a single track hypothesis.

A method of fusing AIS data with incomplete OTH tracking data is presented in Chapter 5. The fusion of AIS with OTH data results in a more informative surveillance picture. It facilitates tracking targets equipped with AIS systems over periods the AIS is not transmitting, based only on the data received from the OTH sensor. The probability of loosing (and re-initializing) an OTH track decreases very much, once associated to an AIS track. The non-cooperative targets tracked by OTH are quickly identified as the ones not being fused with any AIS data.

Two methods of PHD estimate labeling at particle level for the PHD particle filter were proposed in Appendix A. The particle labeling obtains the estimated PHD mesh partitioned under the form of labeled particle clouds, each estimating the probability density function either of a confirmed track or of a measurement in the past two frames. Both methods are based on two 2D assignments, run within the PHD filter at the end of the update stage, therefore having costs more informative than the assignments run at the prediction step. The usage of both methods improves the sampling step, through better usage of particles around confirmed tracks and adds the extra track identity feature to the PHD filter.

6.0.1 Possible Future Work

The first method of fusion is applicable to particle filters that estimate the multitarget state through a single multitarget hypothesis. The cloud-to-cloud association cost and subsequent fusion could be extended in a future work to particle filters that estimate multiple multitarget hypotheses at a given sample time.

In the second method of fusion that uses the prior fused information, a dependency of the fused tracks identification performance on the scenario was observed. Therefore an on-line estimation of the target identification uncertainty could be pursued as future work. Additional information provided by the local trackers could be used for the target identification uncertainty and improved hypotheses assessment. This information could be the estimation of local track swap probability at each local tracker between the fusion intervals. Also the feedback implications could be explored.

Appendix A

Assignment-Based Particle Labeling for PHD Particle Filter

The probability hypothesis density (PHD) filter is an estimator that approximates, on a given scenario, the multitarget distribution through its first-order multitarget moment. This appendix presents two particles labeling algorithms for the PHD particle filter, through which the information on individual targets identity (otherwise hidden within the first-order multitarget moment) is revealed and propagated over time. By maintaining *all* particles labeled at any time, the individual target distribution estimates are obtained under the form of labeled particle clouds, within the estimated PHD. The partitioning of the PHD into distinct clouds, through labeling, provides over time information on confirmed tracks identity, tracks undergoing initiation or deletion at a given time frame, and clutter regions, otherwise not available in a regular PHD (or track-labeled PHD). Both algorithms imply particles tagging since their inception, in the measurements sampling step, and their re-tagging once they are merged into particle clouds of already confirmed tracks, or are merged for the purpose of initializing new tracks. Particles of a confirmed track cloud preserve their labels over time frames. Two data associations are involved in labels management; one assignment merges measurement clouds into particle clouds of already confirmed tracks, while the following 2D-assignment associates particle clouds corresponding to non-confirmed tracks over two frames, for track initiation. The algorithms are presented on a scenario containing two targets with close and crossing trajectories, with the particle labeled PHD filter tracking under measurement origin uncertainty due to observations variance and clutter.

A.1 Introduction

The PHD filter was designed for the unified tracking of multiple and varying number of targets in clutter, based on the random finite sets (RFS) theory and the finite set statistics (FISST), detailed in (Mahler, 2003). For the purpose of introducing the PHD filter, the multitarget state was modeled as a random finite set (RFS) and the PHD for a given multitarget distribution was defined as the density of which integral over any region of the individual-target state space gives the expected number of targets within the region integrated (Mahler, 2003, p.1154). From a point process theory perspective, the PHD is the *intensity function* of the multitarget posterior RFS (Vo *et al.*, 2005, p.1231), or in other words the density of the intensity measure (which intensity measure over a region gives the number of elements within). The integral of the PHD over the whole multitarget space gives the estimated number of targets (confirmed tracks), therefore it is not a classical probability density function. By estimating only the first-order moment of the multitarget state, the PHD is a good approximate sufficient statistic of the multitarget posterior only under the assumptions of small sensor covariance errors and small rate of false alarms (Mahler, 2003, p.1154), (Vo *et al.*, 2005, p.1232).

Subsequent works in (Vo *et al.*, 2005; Zajic and Mahler, 2003) defined its practical implementation using particle filter methods and imposed the filter as a common player on the target tracking community stage, being the subject of continuous research ever since. One direction of improvements is toward estimating and preserving targets identities over time, as within the PHD there is no identity or ordering for the elements of the multi-target state (Vo *et al.*, 2005, p.1226). In a particle filter implementation the PHD integral needs to be adjusted to the closest integer in order to obtain an estimate of the multitarget cardinality.

This appendix presents two methods of obtaining individual track-valued target distribution estimates, under the form of distinct labeled particle clouds, within the multitarget first-order moment estimated by the PHD filter. This is achieved through maintaining *all* particles defining the PHD labeled over time. The process implies particles labeling since their inception as particles sampled around measurements and their re-labeling once particle clouds around measurements are merged into particle clouds of confirmed tracks, or initialized into new ones. In the context of this research a *track* denotes a sequence of state estimates over time, linked through the hypothesis that they belong to the same target. A *track-valued estimate* defines here a state estimate, at a given time, that is tagged as belonging to a track. The labeling at particle level provides the full estimate of the individual target density that defines the track-valued estimate. The labeling methods were designed in order to obtain distinct particle clouds, for subsequent research on particle filter clouds fusion (Danu *et al.*, 2008a, 2009a).

Several approaches for obtaining track-valued estimates within the PHD were already introduced in the literature. In (Lin *et al.*, 2006) tracks are obtained and maintained through a Kalman filter (Bar-Shalom *et al.*, 2001) running in parallel with the PHD filter, and feeded with the PHD peaks as measurements. The PHD peaks are estimated by integrating over a fixed grid, that defines the resolution cell. In that approach labeling is applied at the track level, thus loosing the distribution information of particles that contributed toward obtaining each track-valued estimate [1]. In (Panta *et al.*, 2005) particles are labeled using clustering methods applied on the posterior PHD, and labeling a cluster with the most contributing particles of same label, which could lead to merging clusters of weak targets into stronger ones. Also it implies the condition of not having more than one target born at a time step. In (Panta *et al.*, 2007) there are presented two schemes of combining the PHD filter with the multitarget hypothesis tracking (MHT) method. In the first one clustering methods are used to extract PHD peaks to be used in subsequent MHT while in the second one PHD is used to filter clutter from the MHT input. The identity information is defined and used only at the MHT level, the PHD filter does not make use of target identities.

In the first method introduced in this appendix, following the idea in (Lin *et al.*, 2006), a resolution cell approach is used to estimate and extract the peaks of a track distribution from the PHD, which in this case is represented by labeled particles. On these estimates, a Kalman filter is run for track maintenance over time. Different from (Lin *et al.*, 2006), the labeling occurs at particle level and they are preserved for confirmed tracks over time. The KF here computes the covariances of estimates (predicted and updated) that are used in track initialization and associations. The track identity is preserved in our method through maintaining over time the particle

[1] The second moment of these are re-estimated through the Kalman filter covariance.

labels of the clouds confirmed as track-valued estimates. Different than in (Lin *et al.*, 2006), one *partitioning* and two data associations are run at different stages. All measurements are sampled with labeled particles, which labels are changed subsequently to either labels of confirmed tracks, labels of new tracks or they disappear over time through the resampling steps of the PHD filter. Two data associations are performed at each frame as part of the track maintenance process. First new measurements are assigned to confirmed tracks and next the unassociated measurements are associated between them over two frames. Both data associations are performed using the posteriors of the estimates, as measurements and tracks had already undergone the update step of the PHD filter at their association time. As shown in (Lin *et al.*, 2006) for track labeling, the cloud labeling improves the selection of importance distributions (e.g. different for particles around confirmed targets than for non-associated measurements).

In the second method the parallel Kalman filter is removed, as the clouds of confirmed tracks are propagated over time by preserving their labels from the previous frame. The peak extraction is based on integration over a given region (rectangular windowing) around all peaks of the clouds, and thus is not restricted to a rigid grid, as in the resolution cell approach. The track management is performed as in the first method, using this time sample covariances estimated on the labeled particle clouds. In both methods the states of either confirmed tracks or measurements that enter the assignments had already undergone the PHD update stage, therefore their uncertainty is smaller and assignment cost have higher resolution than the assignment cost computed on the priors, as in classical 2D assignments. Moreover, these *pseudo*measurements entering the association have velocity components. Both methods are presented for the two-dimensional case and are demonstrated on a scenario containing closed and intersecting targets, with measurements uncertainties

The PHD filter and its particle filter implementation is briefly described in Section A.2, the two assignment-based labeling methods of PHD particle filter are presented in Section A.3, the performance of the methods is evaluated on a simulated scenario in Section A.4, and conclusions are presented in Section A.5.

A.2 PHD Particle Filter

The PHD filter is described in subsection (A.2.1), as introduced in (Mahler, 2003). Its particle filter implementation is described in subsection (A.2.2), as introduced in (Vo *et al.*, 2005).

A.2.1 PHD Filter Equations

Denoting with $D_{k|k}(\mathbf{x})$ the estimated PHD at time $k-1$, its estimation at time k from the new measurements and its propagation from time $k-1$ is done in two stages, namely *prediction* and *update.*

Prediction: The prediction equation of the multitarget PHD from time k to $k-1$ is:

$$D_{k|k-1}(\mathbf{x}) = \gamma_k(\mathbf{x}) + \int \left(b_{k|k-1}(\mathbf{x}|\xi) + e_{k|k-1}(\xi)f_{k|k-1}(\mathbf{x}|\xi)\right) D_{k|k-1}(\xi)\lambda(d\xi) \tag{A.1}$$

where

Update: The update equation of the predicted PHD with measurements obtained at time k is

$$D_{k|k}(\mathbf{x}) = \left[(1 - P_D(\mathbf{x})) + \sum_{\mathbf{z}\in Z_k} \frac{P_D(\mathbf{x})g_k(\mathbf{z}|\mathbf{x})}{\kappa_k(\mathbf{z}) + \int P_D(\xi)g_k(\mathbf{z}|\xi)D_{k|k-1}(\xi)\lambda(d\xi)}\right] D_{k|k-1}(\mathbf{x}) \tag{A.2}$$

where

$P_D(\mathbf{x})$ is the probability of detection of an individual target with state $\mathbf{x}$

$g_k(\mathbf{z}|\mathbf{x})$ is the likelihood of individual targets

$\kappa(\mathbf{x})$ is the intensity function of the clutter

Even though the PHD is a statistic of the multitarget posterior, its propagation through both the prediction and update stages is done using statistics on the *individual target state.*

A.2.2 PHD Particle Filter Equations

This subsection briefly presents the PHD particle filter implementation established in (Vo *et al.*, 2005), from which both track labeling proposed in (Lin *et al.*, 2006) and the particle labeling proposed here are derived. At any particular time k, the PHD is estimated through a set of particles $\tilde{D}_k = \{(\xi_k^{(i)}, w_k^{(i)}) \mid i = 1, \ldots, L_k\}$, where $\xi_k^{(i)}$ and $w_k^{(i)}$ stands for the state and weight, respectively, of particle i at time k.

Prediction: From the PHD estimated at time $k-1$, $\tilde{D}_{k-1|k-1} = \{(\xi_{k-1}^{(i)}, w_{k-1}^{(i)}) \mid i = 1, \ldots, L_{k-1}\}$, its prediction for the time k is computed under the form of the new set of particles $\tilde{D}_{k|k-1} = \{\tilde{\xi}_k^{(i)}, \tilde{w}_k^{(i)} \mid i = 1, \ldots, L_k\}$:

- sample

$$\tilde{\xi}_k^{(i)} \sim \begin{cases} q_k(\xi|\xi_{k-1}^{(i)}, Z_k), & i = 1, \dots, L_{k-1} \\ p_k(\xi|Z_k), & i = L_{k-1}+1, \dots, L_{k-1}+J_k \end{cases}$$

- compute associated weights

$$\tilde{w}_{k|k-1}^{(i)} = \begin{cases} \dfrac{e_{k|k-1}(\xi_{k-1}^{(i)}) f_{k|k-1}(\tilde{\xi}_k^{(i)}|\xi_{k-1}^{(i)}) + b_{k|k-1}(\tilde{\xi}_k^{(i)}|\xi_{k-1}^{(i)})}{q_k(\tilde{\xi}_k^{(i)}|\xi_{k-1}^{(i)}, Z_k)} w_{k-1}^{(i)}, \\ \qquad\qquad\qquad\qquad\qquad i = 1, \dots, L_{k-1} \\ \dfrac{1}{J_k} \dfrac{\gamma(\tilde{\xi}_k^{(i)})}{p_k(\tilde{\xi}_k^{(i)}|Z_k)}, \qquad i = L_{k-1}+1, \dots, L_{k-1}+J_k \end{cases}$$

where

$e_{k|k-1}(\cdot)$, $f_{k|k-1}(\cdot \mid \cdot)$, $b_{k|k-1}(\cdot \mid \cdot)$, $\gamma(\cdot)$ are the same as defined in (A.1)

$q_k(\cdot \mid \cdot)$ is importance sampling density such that $q_k(\cdot \mid \cdot) > 0$ wherever $f_{k|k-1}(\cdot \mid \cdot) > 0$ and $b_{k|k-1}(\cdot \mid \cdot) > 0$

$p_k(\cdot \mid \cdot)$ is importance sampling density such that $p_k(\cdot \mid \cdot) > 0$ wherever $\gamma(\cdot) > 0$ or $g(\cdot \mid \cdot) > 0$

Update: The particle weights are updated as in equation (A.3) below

$$\tilde{w}_k^{(i)} = \left[\left(1 - P_D(\tilde{\xi}_k^{(i)})\right) + \sum\nolimits_{\mathbf{z}\in Z_k} \tfrac{P_D(\tilde{\xi}_k^{(i)}) g(\mathbf{z}|\tilde{\xi}_k^{(i)})}{\kappa_k(\mathbf{z}) + \sum_{j=1}^{L_{k-1}+J_k} P_D(\tilde{\xi}_k^{(j)}) g(\mathbf{z}|\tilde{\xi}_k^{(j)})}\right] \tilde{w}_{k|k-1}^{(i)}, \qquad (A.3)$$
$$i = 1, \dots L_{k-1} + J_k.$$

where $P_D(\cdot)$, $g(\cdot \mid \cdot)$ and $\kappa_k(\cdot)$ are the functions entering equation (A.2) and described there.

Resample: The resampling step implies the following operations:

- compute total mass $\hat{N}_{k|k} = \sum_{j=1}^{L_{k-1}+J_k} \tilde{w}_k^{(j)}$,
- resample $\left\{\tilde{\xi}_k^{(i)}, (\tilde{w}_k^{(i)}/\hat{N}_{k|k})\right\}_{i=1}^{L_{k-1}+J_k}$ to $\left\{\xi_k^{(i)}, (w_k^{(i)}/\hat{N}_{k|k})\right\}_{i=1}^{L_k}$, and
- rescale back by $\hat{N}_{k|k}$ and get $\left\{\xi_k^{(i)}, w_k^{(i)}\right\}_{i=1}^{L_k}$.

A.3 Assignment-based Labeled PHD Particle Filter

The PHD surface is defined on the individual target space as being the sum of the projections of each of the multitarget posterior dimensions (where the dimensions cardinality is the number of targets here) onto the single target space, through the first-order moment. The PHD is defined in the individual target space, while the multitarget posterior is defined in the multitarget space, therefore target identities and track estimate continuities over time frames are lost in the former.

Through particle labeling, the intention is to *(i)* preserve target identities and *(ii)* estimate single-target probability density functions within the resulting partitioned PHD, under the form of labeled particle clouds. Our approach is to keep all PHD particles labeled at any time k, and have as a result the PHD always partitioned into labeled particle clouds of labels $l_k = 1, \ldots, \Lambda_k$. Through the PHD filter stages presented in Section A.2, particle labels are preserved through prediction and resampling stages, and they are merged at the end of the update stage: measurement clouds into confirmed track clouds and measurement clouds (at k-1) with measurement clouds (at k) into new track clouds.

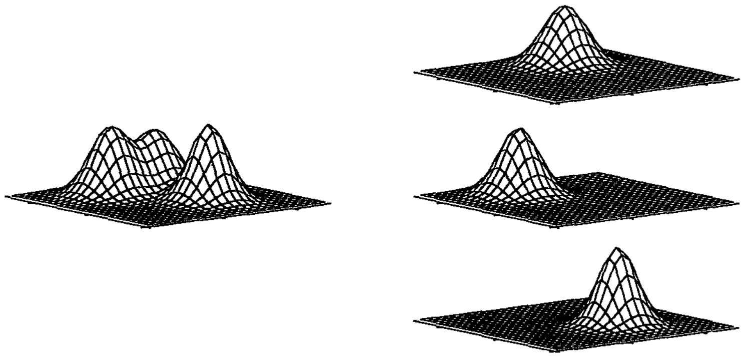
We stress out that particle labeling (with common label for each track) in the estimated PHD is basically the estimation (extraction) from the PHD surface of the probability density functions for each confirmed track, under the form of labeled particle clouds, with each track being defined in its own target state. Therefore having he PHD estimated at every time k through a set of *labeled particles*: $\tilde{D}_k = \{\xi_k^{(i)}, w_k^{(i)}, l_k^{(i)}\}$, with $i = 1, \dots, L_k$, basically moves the estimate from the PHD domain, estimated on the individual target space, $\mathbf{x} \in \{\{\mathbf{x}_1\}, \dots, \{\mathbf{x}_N\}\}$, back into the multitarget space $X = [\mathbf{x}_1, \dots, \mathbf{x}_N,]$, as shown in Figure A.1(a), A.1(b) and equations (A.4):

$$\begin{tikzcd}
\begin{array}{c}\text{multitarget}\\ \text{space}\end{array} \arrow[r] & \begin{array}{c}\text{individual}\\ \text{target space}\end{array} \arrow[r] & \begin{array}{c}\text{multitarget}\\ \text{space}\end{array}
\end{tikzcd}$$

$$f_{k|k}(X|Z^k) \xrightarrow{\text{PHD Filter estimates}} D_{k|k}(\mathbf{x}) \xrightarrow{\text{Particle Labeling in PHD}} \hat{f}_{k|k}(X|Z^k) \qquad \text{(A.4)}$$

$$\tilde{D}_k = \{\xi_k^{(i)}, w_k^{(i)}\} \longrightarrow \tilde{D}_k^{\{l_k\}} = \{\xi_k^{(i)}, w_k^{(i)}, l_k^{(i)}\}$$

In the following subsections we present the two assignment-based labeled PHD particle filter algorithms. The target states contain position and velocity as $\mathbf{x} = [x\ v_x\ y\ v_y]$ and measurements are received as $\mathbf{z} = [x\,y]$, therefore both state and observation are in the (x, y) two-dimensional cartesian space. The first method is close to the method presented in (Lin *et al.*, 2006), however one major difference is that the filter presented here implements the *labeling at the particle level*, thus obtaining *particle filter clouds* as estimates of *individual tracks.*



(a) Sample idealized PHD for three targets

(b) Particle clouds separated, through particle labeling

Figure A.1: Through labeling, pdf of different targets are distinguished and obtained basically in separate target spaces.

A.3.1 Method 1 - Particle Labeled PHD Filter Using Kalman Filter

Initialization

At time $k = 1$, given the set of n_1 observations $Z_1 = [\mathbf{z}_1^1, \dots, \mathbf{z}_1^{n_1}]$, each of them is sampled with m_1 particles. As usually only position information is given in measurements, the state components of particles are initialized separately in the sampling process, as in (Lin *et al.*, 2006). The initialization sequence includes particles sampling, weights computation, particles labeling and track list initialization, described next.

A. Particles Sampling

A total of $L_1 = m_1 n_1$ particles are initialized, of which

a. position components $\tilde{\xi}^{(i)}_{1,(x,y)}$ are sampled from the importance sampling density for measurements, $p(\xi|\mathbf{z}_1^n)$, selected based on sensor characteristics (measurement positions and uncertainties):

$$\tilde{\xi}^{(i)}_{1,(x,y)} = \tilde{\xi}^{(j+m_1(n-1))}_{1,(x,y)} \sim p(\xi|\mathbf{z}^n_1), \qquad i = 1,\dots,L_1, \quad j = 1,\dots,m_1, \quad n = 1,\dots,n_1, \tag{A.5}$$

b. velocity components $\tilde{\xi}^{(i)}_{1,(v_x,v_y)}$ are sampled from the joint uniform distribution within maximum velocities allowed for a target

$$\tilde{\xi}^{(i)}_{1,(v_x,v_y)} \sim U(\xi_{(v_x,v_y)} \mid [-v_{x\_max}, v_{x\_max}], [-v_{y\_max}, v_{y\_max}]), \quad i = 1,\dots,L_1. \tag{A.6}$$

B. Weights Computation

$$\tilde{w}^{(i)}_1 = \ell_1/(m_1 \cdot n_1), \quad i = 1,\dots,L_1, \tag{A.7}$$

where ℓ_1 is the initial estimated number of targets and n_1 is the number of measurements.

C. Particles Labeling

a. particles sampled from measurements $\mathbf{z}^n_1$, $n = 1,\dots,\ell_1$, guessed as from valid targets, are labeled as

$$l^{(i)}_1 = l^{(j+m_1(n-1))}_1 = l^c_1 = n, \quad i = 1,\dots,m_1\ell_1, \quad j = 1,\dots,m_1 \tag{A.8}$$

thus obtaining ℓ_1 labeled particle clouds of labels $l^c_1 = 1,\dots\ell_1$, where c stands for confirmed tracks

b. particles sampled from measurements $\mathbf{z}^n_1$, $n = \ell_1 + 1,\dots,n_1$, guessed as from candidate targets, are labeled as

$$l_1^{(i)} = l_1^{(j+m_1(n-1)+m_1\ell_1)} = l_1 = M_0{+}n, \quad i = m_1\ell_1{+}1, \ldots, L_1, \quad j = 1, \ldots, m_1 \tag{A.9}$$

where M_0 is a high number used to distinguish further between labels, i.e. between measurement labels (clouds of yet unconfirmed tracks) and confirmed track labels

D. Track List Initialization

The track list $\mathcal{T}_k$ at $k = 1$ is initialized with a tentative track for each observation, with their states capturing the first two moments, as needed by the parallel Kalman filter that will update them.

$$\mathcal{T}_1 = \hat{\mathcal{T}}_1^{1:\ell_1} = \left\{ \begin{bmatrix} Z_{1,x} \\ 0 \\ Z_{1,y} \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma^2_{Z_{1,x}} & 0 & 0 & 0 \\ 0 & v^2_{x\_max}/3 & 0 & 0 \\ 0 & 0 & \sigma^2_{Z_{1,y}} & 0 \\ 0 & 0 & 0 & v^2_{y\_max}/3 \end{bmatrix} \right\}. \tag{A.10}$$

Prediction

At each time $k \geq 2$, a new particle is sampled for each existing particle at $k-1$, using the importance sampling density for target transition distribution, $q(\cdot \mid \cdot)$. A number of J_k new particles are sampled also to account for the new set of n_k measurements, $Z_k = [\mathrm{z}_k^1, \ldots, \mathrm{z}_k^{n_k}]$, using the importance sampling density for measurements uncertainty, $p(\cdot \mid \cdot)$. The prediction sequence, including particles labeling, is given below.

A. Particles Sampling

a. From existing particles (at $k-1$):

$$\tilde{\xi}_k^{(i)} \sim q_k(\xi|\xi_{k-1}^{(i)}, Z_k), \quad i = 1, \ldots, L_{k-1}. \tag{A.11}$$

b. From current observations set Z_k:

First the set of measurements is partitioned into ones falling within the gate of a confirmed track in $\mathcal{T}_{k-1}$ and ones falling outside any track gate, $Z_k = \{Z_{k,in}, Z_{k,out}\}$. For this purpose, the confirmed tracks in track list $\mathcal{T}_{k-1} = \hat{T}_{k-1}^{1:\ell_{k-1}}$ are predicted to time k using Kalman filter prediction equations (Bar-Shalom *et al.*, 2001)

$$\mathcal{T}_{k-1} = \hat{T}_{k-1}^{1:\ell_{k-1}} = \{\hat{\mathbf{x}}_{k-1}^{l^c}, \mathbf{P}_{k-1}^{l^c}\}_{l^c=1}^{\ell_{k-1}} \xrightarrow{\text{KF Prediction}} \\ \{\hat{\mathbf{x}}_{k|k-1}^{l^c}, \mathbf{P}_{k|k-1}^{l^c}\}_{l^c=1}^{\ell_{k-1}} = \hat{T}_{k|k-1}^{1:\ell_{k-1}} = \mathcal{T}_{k|k-1} \tag{A.12}$$

Next either: *i)* a $2D$ association (Bertsekas and Castanon, 1993; Jonker and Volgenant, 1987), is performed between measurements and predicted confirmed tracks, or *ii)* measurements that fall within the gate of several tracks are duplicated and sampled several times, for each track gate. Measurements within $Z_{k,in}$ are sampled with $\alpha_1 s_k$ particles, while measurements in $Z_{k,out}$ are sampled with $\alpha_2 s_k$ particles, where α_1, α_2 are design parameters (Lin *et al.*, 2006) that account for probabilities of measurements.

$$\tilde{\xi}_k^{(i)} \sim \begin{cases} \tilde{\xi}_{k,\{x,y\}}^{(i)} \sim p(\xi_{\{x,y\}}|Z_{k,in}) \\ \tilde{\xi}_{k,\{v_x,v_y\}}^{(i)} \sim p(\xi|T_k^n) \end{cases}, i = L_{k-1}+1, \ldots, L_{k-1}+J_{k,in} \tag{A.13}$$

$$\tilde{\xi}_k^{(i)} \sim \begin{cases} \tilde{\xi}_{k,\{x,y\}}^{(i)} \sim p(\xi_{\{x,y\}}|Z_{k,out}) \\ \tilde{\xi}_{k,\{v_x,v_y\}}^{(i)} \sim U([-v_{x\_max}, v_{x\_max}], [-v_{y\_max}, v_{y\_max}]) \\ \qquad i = L_{k-1} + J_{k,in} + 1, \quad \ldots, L_{k-1} + J_k \end{cases} \tag{A.14}$$

where $J_{k,in} = \alpha_1 s_k n_{k,in}$, $J_{k,out} = \alpha_2 s_k n_{k,out}$, and $n_{k,in}$, $n_{k,out}$ are the cardinalities of $Z_{k,in}$ and $Z_{k,out}$, respectively.

B. Weights Computation

a. Weights for particles predicted from existing ones at $k-1$ are computed using

$$\tilde{w}_{k|k-1}^{(i)} = w_{k-1}^{(i)} e_{k|k-1}(\xi_{k-1}^{(i)}), \quad i = 1, \ldots, L_{k-1}. \tag{A.15}$$

b. Weights for newborn particles, sampled from measurements in $Z_{k,in}$, $Z_{k,out}$ are

$$\tilde{w}_{k|k-1}^{(i)} = \frac{1}{\alpha_1 s_k n_{k,in} + \alpha_2 s_k n_{k,out}} \cdot \frac{\gamma_k(\tilde{\xi}_k^{(i)})}{p_k(\tilde{\xi}_k^{(i)}|Z_k)}, \quad i = L_{k-1}+1, \ldots, L_{k-1} + J_k. \tag{A.16}$$

C. Particles Labeling

a. Persistent particles, propagated from $k-1$ through prediction, preserve their labels, therefore with values for confirmed tracks in $\{1, \ldots, \ell_{k-1}\}$ and for measurements in $\{M_{k-2}+1, \ldots, M_{k-1}\}$ (i.e. measurements clouds before $k-2$ and not merged yet into a confirmed or newly initiated track cloud are discarded at k).

$$\tilde{l}_k^{(i)} = l_{k-1}^{(i)}, \quad i = 1, \ldots, L_{k-1}. \tag{A.17}$$

b. Particles sampled from new measurements $\mathbf{z}_k^n \in Z_{in,k}$, where $n = 1, \ldots, n_{k,in}$ are labeled as

$$\tilde{l}_k^{(i)} = \tilde{l}_k^{(j+(n-1)\alpha_1 s_k)} = M_{k-1} + n \tag{A.18}$$

where $j = 1, \ldots, \alpha_1 s_k$, and therefore $i = L_{k-1} + 1, \ldots, L_{k-1} + J_{k,in}$.

c. Particles sampled from new measurements $\mathbf{z}_k^n \in Z_{out,k}$, where $n = 1, \ldots, n_{k,out}$ are labeled as

$$\tilde{l}_k^{(i)} = \tilde{l}_k^{(j+(n-1)\alpha_2 s_k + J_{k,in})} = M_{k-1} + n_{k,in} + n \tag{A.19}$$

where $j = 1, \ldots, \alpha_2 s_k$, and therefore $i = L_{k-1} + J_{k,in} + 1, \ldots, L_{k-1} + J_k$

Update

A. Weights Computation. All particles, $i = 1, \ldots L_{k-1} + J_k$, have their weights updated using (Lin *et al.*, 2006)

$$\begin{aligned}\tilde{w}_k^{(i)} = (\Big(1 - P_D(\tilde{\xi}_k^{(i)})\Big) &+ \sum_{\mathbf{z}\in Z_{k,in}} \frac{\beta_1 P_D(\tilde{\xi}_k^{(i)}) g(\mathbf{z}|\tilde{\xi}_k^{(i)})}{\kappa_k(\mathbf{z}) + \sum_{j=1}^{L_{k-1}+J_k} P_D(\tilde{\xi}_k^{(j)}) g(\mathbf{z}|\tilde{\xi}_k^{(j)})} \\ &+ \sum_{\mathbf{z}\in Z_{k,out}} \frac{\beta_2 P_D(\tilde{\xi}_k^{(i)}) g(\mathbf{z}|\tilde{\xi}_k^{(i)})}{\kappa_k(\mathbf{z}) + \sum_{j=1}^{L_{k-1}+J_k} P_D(\tilde{\xi}_k^{(j)}) g(\mathbf{z}|\tilde{\xi}_k^{(j)})}) \tilde{w}_{k|k-1}^{(i)}.\end{aligned} \tag{A.20}$$

B. Track List Update

a. The predicted tracks in $\mathcal{T}_{k|k-1}$ are updated as follows:

i) Find the peaks $\mathcal{P}_k = \{\pi_k^{i_c}\}_{i_c=1}^{P}$ of the PHD, using weights integration over the coarser resolution cell grid approach (Lin *et al.*, 2006), thus obtaining the PHD in a resolution cell (PHDRC) peaks.

ii) Label resolution cell peaks with the the most contributing cloud label $\mathcal{P}_k = \{\pi_k^{i_c}, l_k^{i_c}\}_{i_c=1}^{P}$. The shortcoming of allowing only an ID to lay within a resolution cell is circumvented in the next step.

iii) Update tracks in $\mathcal{T}_{k|k-1}$ with their $2D$-assigned PHDRC peaks obtained above, using Kalman filter update equation. Note that the pseudomeasurements entering the association here have also velocity components, as the component particles of had already undergone the PHD update and therefore had obtained this information (i.e. for peak $\pi_k^{i_c(l^c)}$ assigned to track $\hat{T}_{k|k-1}^{l^c}$):

$$\begin{aligned}\mathcal{T}_{k|k-1} = \hat{T}_{k|k-1}^{1:\ell_{k-1}} = \{\hat{\mathbf{x}}_{k|k-1}^{l^c}, \mathbf{P}_{k|k-1}^{l^c}, \pi_k^{i_c(l^c)}\}_{l^c=1}^{\ell_{k-1}} \xrightarrow{\text{KF Update}} \\ \{\hat{\mathbf{x}}_k^{l^c}, \mathbf{P}_k^{l^c}\}_{l^c=1}^{\ell_{k-1}} = \hat{T}_k^{1:\ell_{k-1}} = \mathcal{T}_k\end{aligned} \tag{A.21}$$

If no PHDRC peak is assigned to a track that had a measurement within its gate in $Z_{k,in}$, it is updated with that measurement directly. If the clouds of two tracks lie within the same resolution cell, the stronger target hides the other one in the resolution cell tagging, therefore the cloud of the weakest target does not participate in the assignment. It may result in having a track in the track list not updated, even if it had a valid measurement in its gate, therefore in Z_{in}. Such a track is updated separately, with its closest measurement in Z_{in} Another way to circumvent this is to compute the peaks of the resolution cell on each label separately. This is performed in the second algorithm however not on a resolution cell grid.

iv) If a track in $\mathcal{T}_{k|k-1}$ does not have assigned neither a PHDRC peak in $\mathcal{P}_k$, neither a measurement in $Z_{k,in}$, then its prediction is promoted directly into $\mathcal{T}_k$ and the track is marked as having a missed detection. All tracks found with two (or other design-established value of) consecutive missed detections are marked for deletion. The consequence is that their labels won't participate in the measurements partitioning process.

b. New tracks are added to $\mathcal{T}_k$ and their corresponding particle clouds labels change from measurement labels into track labels. Unassociated entries (RC peaks) in $\mathcal{P}_k$ are associated to unassociated entries (RC peaks) in $\mathcal{P}_{k-1}$ through $2D$-assignment and if valid pairs are found, new tracks are initialized using classic track initialization from measurements procedures (Bar-Shalom *et al.*, 2001).

c. Tracks are removed from $\mathcal{T}_k$ if they do not have associated peaks in $\mathcal{P}_{k-1}$ or no associated measurements for the past two frames.

C. Particles Labeling

a. Labels of any measurement particle cloud ($l_k > M_{k-1}$) that participated in a track update is changed into the respective track label, thus merging the two clouds.

b. Labels of newly confirmed tracks are given by incrementing the higher current confirmed track label, $l_k = \ell_{k-1} + 1, \dots, \ell_k$, with $\ell_k = \ell_{k-1} + N_{newtracks}$.

c. Labels of any pair of two measurement particle clouds that initialized a new track (one in $M_{k-2} < l_k(1) < M_{k-1}$, and other in $M_{k-1} < l_k(2) < M_k$) are changed into the new track label, therefore the two clouds merge.

At the end of the update step the labels of particle (clouds) are in the set

$$l_k^{(i)} \in \underbrace{\{1,\ldots,\ell_k\}}_{\text{confirmed tracks}} \cup \underbrace{\{M_{k-2}+1,\ldots,M_{k-1}\}}_{\text{measurements at } k\text{-}1} \cup \underbrace{\{M_{k-1}+1,\ldots,M_k\}}_{\text{measurements at } k} \tag{A.22}$$

Resampling

In the resampling step the resulting particles preserve the labels of particles from which they are obtained:

$$\{\xi_k^{(i)}, w_k^{(i)}, l_k^{(i)}\}_{i=1}^{L_k} \sim \{\tilde{\xi}_k^{(j)}, \tilde{w}_k^{(j)}, \tilde{l}_k^{(j)}\}_{j=1}^{L_{k-1}+J_k} \tag{A.23}$$

During resampling labels of clouds of low weights might be removed naturally (i.e. not resampled). Beside these labels which might disappear, at the end of the resampling stage the labels of unweighted particles (clouds) are preserved. They are in the set

$$l_k^{(i)} \in \underbrace{\{1,\ldots,\ell_k\}}_{\text{confirmed tracks}} \cup \underbrace{\{M_{k-1}+1,\ldots,M_k\}}_{\text{measurements at } k} \tag{A.24}$$

where the first set *includes* the labels of $\mathcal{T}_k$ and the second set *includes* the set of clouds build around unassociated measurements over the past two frames. Some tracks might have been deleted, therefore the set of labels in $\mathcal{T}_k$ might be smaller than $\{1,\ldots,\ell_k\}$. Also some measurements over past two frames might be already associated and had formed new tracks, therefore their set might be smaller than $\{M_{k-2}+1,\ldots,M_k\}$. Normally all other particles should disappear after resampling. If a particular cloud is still present and is in none of above sets, its particles will be removed, subject to the condition of its mass to be under a specified threshold v.

A.3.2 Method 2 - Particle Labeled PHD Filter Using Cloud Estimate

In this method the continuity of a track over time frames is preserved solely by the (common) label of the particles within its cloud. The resolution cell peaks are replaced in this algorithm with the mean estimates of the labeled particle clouds. These are estimated on both clouds of confirmed tracks and clouds of unassociated measurements.

Figure A.2 shows the flow of particles and their labeling in the PHD filter proposed.

Initialization

The initialization step is identical with the one presented in Section A.3.1, with the exception that the track list $\mathcal{T}_1$ is initialized directly with the labels $l_1^c = 1, \ldots \ell_1$ of particle clouds.

Prediction

A. Particles Sampling

The particles sampling from existing particles is as in Section A.3.1. The partition of measurements is done using the gating of particle clouds labeled $l_{k-1}^c = 1, \ldots \ell_{k-1}$, marked as for confirmed targets. Particle clouds with labels in $\mathcal{T}_{k-1}$ are predicted first and their gating is used next. The sampling from measurements, upon their partitioning, is performed as in Section A.3.1.

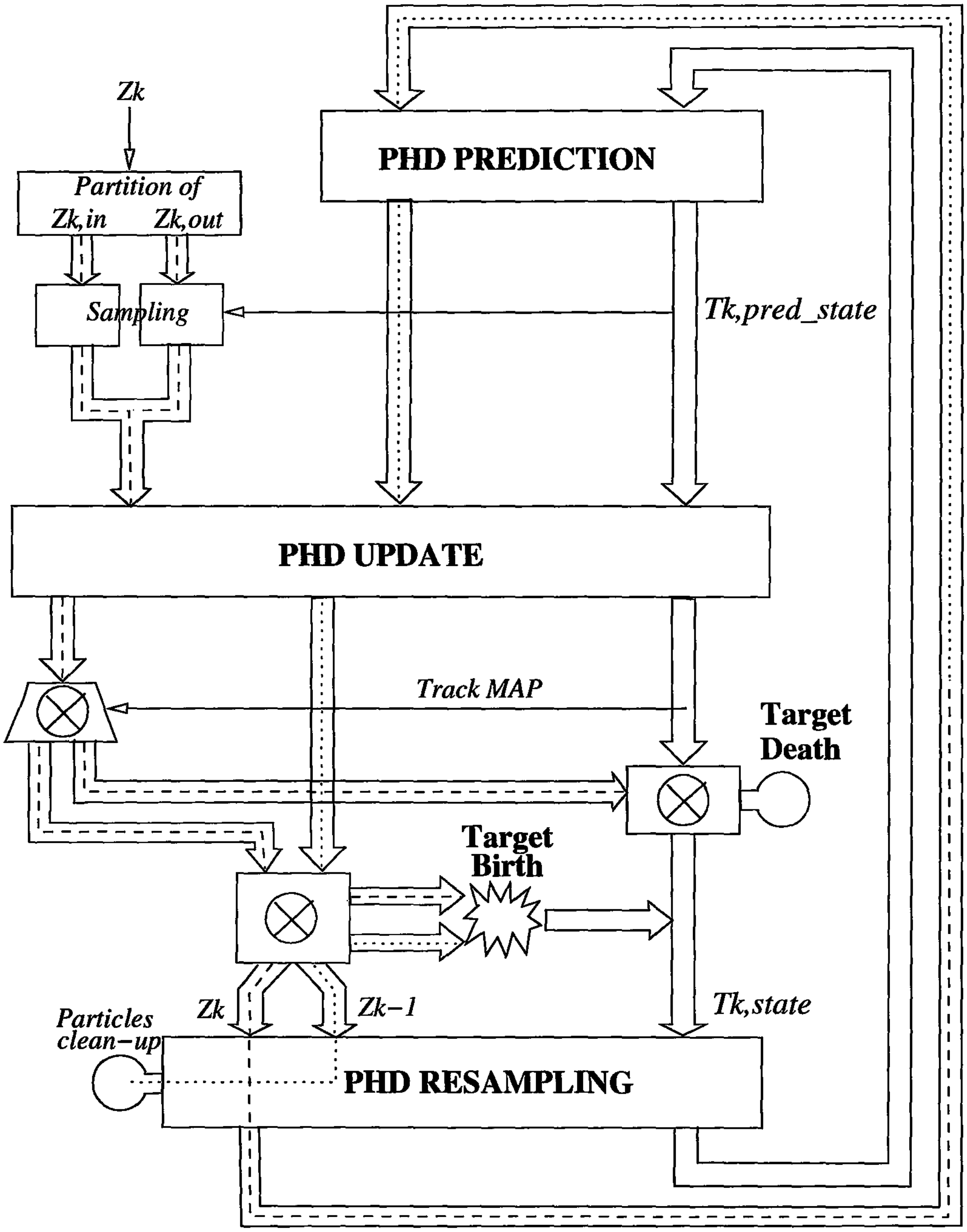


Figure A.2: Labels' flow in the particle labeled PHD for the filter cycle at time frame k. The arrows marked with discontinued line show the flow of measurements ingested at time k. The arrows marked with dotted line show the flow of measurements ingested at time $k-1$ and preserved at time k. The empty arrows show the flow of existing or newly formed tracks at time k

B. Weights Computation and *C. Particles Labeling* steps are identical to the ones in Section A.3.1

Update

A. Weights Computation

Is identical to the one in Section A.3.1

B. Track List Update

a. The cloud mean estimates and the sample covariance matrices, i.e., the matrix of the mean square error (MSE), of the confirmed tracks particle clouds, with labels in $1, \ldots, \ell_{k-1}$, are computed, as they are used in the next two 2D-assignments, $\mathcal{T}_k = \tilde{T}^{1:\ell_{k-1}}_{k|k-1} = \{\tilde{\mathbf{x}}^{l^c}_{k|k-1}, \tilde{\mathbf{P}}^{l^c}_{k|k-1}\}^{\ell_{k-1}}_{l^c=1}$

b. Compute the cloud estimates and the MSE matrix of measurements clouds, $\mathcal{M}_k$, sampled at k, and persistent measurements clouds $\mathcal{M}_{k|k-1}$, sampled at $k-1$, needed in the following two associations.

c. Combine confirmed tracks clouds in $\mathcal{T}_{k|k-1}$ with their 2D-assigned measurement clouds in $\mathcal{M}_k$, using Mahalanobis-distance based association costs.

d. Initialize new tracks through merging $\mathcal{M}_k$ measurement clouds with their 2D-assigned $\mathcal{M}_{k|k-1}$ measurement clouds, using Mahalanobis-based association costs. NOTE: The clouds of labels l_c of confirmed tracks have already their weights updated with the new measurements using the PHD filter equation (A.2), however their entity as clouds is considered still not updated entirely, as the particles sampled from measurements at k are not included in their clouds yet.

C. Particles Labeling

Here two $2D$-associations are performed for two purposes: *i)* to merge clouds of new measurements into clouds of confirmed tracks, and *ii)* to associate measurement clouds over current and previous frame in order to merge them into the cloud of a newly confirmed (and newly labeled) track. Both associations are performed *after* the measurements have been used in the PHD filter update equations, therefore are *not* affecting the PHD filter convergence for the current frame. Moreover, in *ii)*, the measurements to tracks (clouds) association, the updated cloud estimates of tracks are considered, and not the prior as in the case of usual association, therefore entering the association with lesser uncertainty. In *ii)*, the measurements to measurements association, the particle clouds of measurements at time $k-1$ and k were already updated *twice*, respectively *once*, before they enter the association. Therefore their association costs are more informative than the association costs of pure measurements. The two associations at particle cloud level are done using their estimates, however they could use directly the cloud-to-cloud association methods in (Danu *et al.*, 2009a). The steps are listed below:

a. The label of any measurement particle cloud in $\mathcal{M}_k$ that participated in a track update is changed into the respective track label.

b. Labels of newly confirmed tracks are given by incrementing the higher current confirmed track label, $l_k = \ell_{k-1} + 1, \ldots, \ell_k, \ell_k = \ell_{k-1} + N_{newtracks}$.

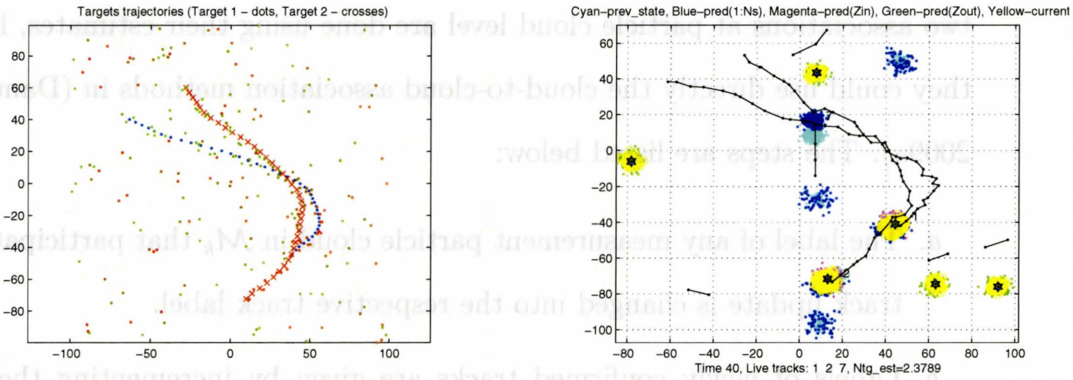
c. Labels of any pair of two measurement particle clouds that initialized a new track (one in $\mathcal{M}_{k|k-1}$, and other in $\mathcal{M}_k$ are changed into the new track label, as the two clouds merge.

Resampling

The resampling step is identical with the one presented in A.3.1

A.4 Simulation

We consider two targets with close trajectories, which also cross each other, observed in a $2-D$ cartesian space. The target state given as $\mathbf{x} = [x, v_x, y, v_y]$ and the measurement as $\mathbf{z} = [x, y]$. The observations variances are high enough compared with the targets closeness such that the tracking is performed under measurements origin uncertainty. The average rate of clutter points per scan is $r = 4$, therefore with $cz = 1/200^2$ and $\kappa_k = r \cdot cz$. The scenario and performance evaluation includes the track initiation stage for each target, over a period of 40 sec, with measurements at every second. Targets trajectories and sample measurements (including clutter) for a run are shown in Figure A.3.a.



(a) True targets trajectories and measurements.

(b) Track-valued estimates over one run.

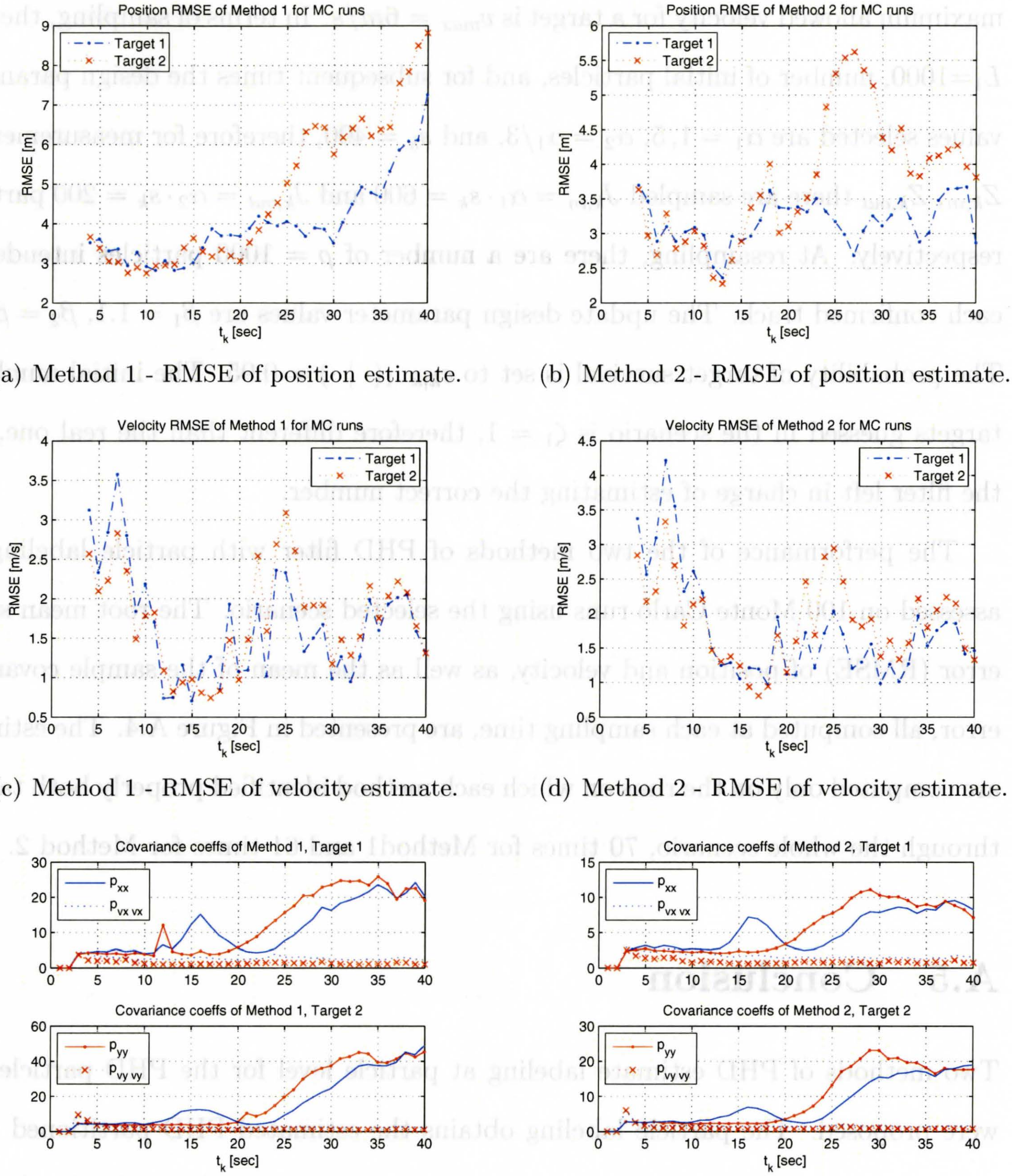
Figure A.3: True targets and track estimates using Method 1, Particle Labeled PHD using Kalman Filter.

The measurement noise error covariance is $P_w^2 = diag\{\sigma_x^2, \sigma_y^2\}$, with σ_x=2.5m/s and σ_y=2.5m/s. The particle filter process noise selected is $\sigma_v^2 = [1.0^2,\ 0.5^2]$ and the maximum allowed velocity for a target is $v_{max} = 6m/s$. In terms of sampling, there are L_1=1000, number of initial particles, and for subsequent times the design parameters values selected are $\alpha_1 = 1,5$, $\alpha_2 = \alpha_1/3$, and $s_k = 400$, therefore for measurements in $Z_{k,in}$, $Z_{k,out}$ there are sampled $J_{k,in} = \alpha_1 \cdot s_k = 600$ and $J_{k,out} = \alpha_2 \cdot s_k = 200$ particles, respectively. At resampling, there are a number of $\rho = 1000$ particles intended for each confirmed track. The update design parameter values are $\beta_1 = 1.1$, $\beta_2 = \beta_1/25$. The probability of target survival is set to $e_{k|k-1}(\cdot \mid \cdot) = 0.95$. The initial number of targets guessed in the scenario is $\zeta_1 = 1$, therefore different than the real one, with the filter left in charge of estimating the correct number.

The performance of the two methods of PHD filter with particle labeling was assessed on 100 Monte Carlo runs using the selected scenario. The root mean square error (RMSE) of position and velocity, as well as the mean of the sample covariance error, all computed at each sampling time, are presented in Figure A.4. The estimates are computed only on the runs on which each method identified properly both targets, through the whole scenario, 70 times for Method1 and 61 times for Method 2.

A.5 Conclusion

Two methods of PHD estimate labeling at particle level for the PHD particle filter were proposed. The particle labeling obtains the estimated PHD partitioned under the form of labeled particle clouds, each estimating the probability density function either of a confirmed track or of a measurement in the past two frames. Both methods are based on two 2D assignments, run within the PHD filter at the end of the update stage, therefore having costs more informative than the assignments run at



(a) Method 1 - RMSE of position estimate.

(b) Method 2 - RMSE of position estimate.

(c) Method 1 - RMSE of velocity estimate.

(d) Method 2 - RMSE of velocity estimate.

(e) Method 1 - average of error covariance terms.

(f) Method 1 - average of error covariance terms.

Figure A.4: Performance evaluation of Method 1 and Method 2 of particle labeled PHD particle filter.

the prediction step. The usage of both methods improves the sampling step, through better usage of particles around confirmed tracks. The methods were ran on a 2-D scenario in order to show that target identities are preserved for the case of closed and crossing trajectories, estimated with the PHD filter.

Bibliography

Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, **50**(2), 174–188.

Bar-Shalom, Y. (2006). On Hierarchical Tracking for the real world. *Transactions on Aerospace and Electronic Systems*, **42**(3), 846–850.

Bar-Shalom, Y. and Blair, W. D., editors (2000). *Multitarget-Multisensor Tracking: Applications and Advances, Vol.III.* Artech House, Norwood, MA.

Bar-Shalom, Y. and Campo, L. (1986). The effect of the common process noise on the two-sensor fused-track covariance. *IEEE Transactions on Aerospace and Electronic Systems*, **22**, 803–805.

Bar-Shalom, Y. and Chen, H. (2006). Multisensor track-to-track association for tracks with dependent errors. *Journal of Advances in Information Fusion*, **1**(1), 3–14.

Bar-Shalom, Y. and Chen, H. (2008). Covariance reconstruction for track fusion with legacy track sources. *Journal of Advances in Information Fusion*, **3**(2), 107–117.

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

Bar-Shalom, Y., Li, X. R., and Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software.* Wiley, New York.

Bar-Shalom, Y., Kirubarajan, T., and Gokberk, C. (2005). Tracking with classification-aided multiframe data association. *IEEE Transactions on Aerospace and Electronic Systems*, **41**(3), 868–878.

Bar-Shalom, Y., Blackman, S. S., and Fitzgerald, R. J. (2007). Dimensionless score function for multiple hypothesis tracking. *IEEE Trans. on Aerospace and Electronic Systems*, **43**(1), 392–400.

Bayes, T. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions*, **52**, 370–418.

Bertsekas, D. P. and Castanon, D. (1993). A forward/reverse auction algorithm for asymmetric assignment problem. *Technical Report, MIT*, **Lids-P-2159**.

Blom, H. and Bloem, E. (2007). Bayesian tracking of possibly unresolved maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, **43**(2), 612–627.

Cameron, A., Haberman, G., Mohandes, M., and Bogner, R. E. (1996). Modelling OTHR Tracks for Association and Fusion. In *Proceedings of IEEE National Radar Conference*, pages 100–105, Ann Arbor, Michigan.

Carillo, J. A. and Toscani, G. (2007). Contractive probability metrics and asymptotic behavior of dissipative kinetic equations. *Rivista di Matematica della Universita di Parma*, **6**, 75–198.

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

Chong, C. Y., Mori, S., and Chang, K. C. (1990). Distributed multitarget multisensor tracking. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Advanced Applications*, chapter 8, pages 247–295. Artech House.

Clark, D. E. and Bell, J. (2007). Multitarget state estimation and track continuity for the particle PHD filter. *IEEE Transactions on Aerospace and Electronic Systems*, **43**(4), 1441–1453.

Coates, M. (2004). Distributed particle filters for sensor networks. In *Third International Symposium on Information Processing in Sensor Networks*, pages 99–107.

Crisan, D. (2001). Particle filters – a theoretical perspective. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*, chapter 2, pages 17–42. Springer, New York.

Crisan, D. and Doucet, A. (2002). A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, **50**(3), 736–746.

Daley, D. J. and Vere-Jones, D. (1988). *An Introduction to the Theory of Point Processes*. Springer-Verlag, New York.

Danu, D., Sinha, A., Kirubarajan, T., Farooq, M., and Brookes, D. (2007a). Fusion of Over-the-Horizon Radar and Automatic Identification Systems for Overall Maritime Picture. In *Proceedings of the 10th International Conference on Information Fusion*, Québec.

Danu, D., Sinha, A., Kirubarajan, T., Farooq, M., and Peters, D. (2007b). Performance Evaluation of Multi-platform Distributed Data Fusion Methods for Multi-target Tracking. In *Proceedings of IEEE Aerospace Conference*, Big Sky, Montana.

Danu, D., Sinha, A., and Kirubarajan, T. (2007c). Track-to-Track Association Using Informative Prior Associations. In *SPIE Proceedings, Signal and Data Processing of Small Targets*, volume 6699, San Diego, CA.

Danu, D., Kirubarajan, T., Lang, T., and McDonald, M. (2008a). Multisensor particle filter cloud fusion for multitarget tracking. In *Proceedings of the 11th International Conference on Information Fusion*, pages 1191–1198, Cologne, Germany.

Danu, D., Lang, T., and Kirubarajan, T. (2008b). Track Fusion with Feedback for Local Trackers Using MHT. In *Proceedings of International SPIE Defense and Security Symposium*, volume 6969, Orlando, Florida.

Danu, D., Kirubarajan, T., and Lang, T. (2009a). Wasserstein distance for the fusion of multisensor, multitarget particle filter clouds. In *Proceedings of the 12th International Conference on Information Fusion*, pages 25–32, Seattle, WA.

Danu, D. G., Kirubarajan, T., and Lang, T. (2009b). Assignment-based particle labeling for PHD particle filter. In *Proceedings of 2009 SPIE Conference on Signal and Data Processing of Small Targets*, volume O. Drummond Ed., Vol. 7445, San Diego, CA.

Deb, S., Yeddanapudi, M., Pattipati, K., and Bar-Shalom, Y. (1997). A generalized S-D assignment algorithm for multisensor-multitarget state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, **33**(2).

Doucet, A., de Freitas, J. F. G., and Gordon, N. J., editors (2001). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York.

Drummond, O. E. (2003). Multiple-frame best-hypothesis target tracking with multiple sensors. In *Proc. SPIE*, volume 5204, pages 318–333.

Ekman, M., Sviestins, E., Sjoberg, L., Boers, Y., and Driessen, H. (2007). Particle filters for tracking closely spaced targets. In *Proceedings of the 10th International Conference on Information Fusion*, Quėbec, Canada.

Givens, C. R. and Shortt, R. M. (1984). A class of Wasserstein metrics for probability distributions. *The Michigan Mathematical Journal*, **31**(2), 231–240.

Gordon, N., Salmond, D., and Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F Radar, Sonar and Navigation*, **140**(2), 107–113.

Hoffman, J. R. and Mahler, R. P. S. (2004). Multitarget miss distance via optimal assignment. *IEEE Transactions on Systems, Man and Cybernetics, Part A, Systems and Humans*, **34**(5), 327–336.

Hue, C., Cadrex, J.-P. L., and Perez, P. (2002). Tracking multiple objects with particle filtering. *IEEE Transactions on Aerospace and Electronic Systems*, **38**(3), 791–812.

Jonker, R. and Volgenant, A. (1987). A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, **38**, 325–340.

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

Kaplan, L. and Blair, W. D. (2004). Assignment costs for multiple sensor track-to-track association. In *Proceedings of the xth International Conference on Information Fusion*, Stockholm, Sweden.

Kreucher, C., Kastella, K., and III, A. O. H. (2005). Multitarget tracking using the joint multitarget probability density. *IEEE Transactions on Aerospace and Electronic Systems*, **41**(4), 1396–1414.

Kurien, T. (1990). Issues in the design of practical multitarget tracking algorithms. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Advanced Applications*, chapter 3, pages 43–88. Artech House.

Lang, T. and Dunne, D. (2008). Applications of particle filters in a hierarchical data fusion system. In *Proceedings of the 11th International Conference on Information Fusion*, pages 1980–1986, Cologne, Germany.

Lin, L., Bar-Shalom, Y., and Kirubarajan, T. (2006). Track labeling and PHD filter for multitarget tracking. *IEEE Transactions on Aerospace and Electronic Systems*, **42**(3), 778–794.

Liu, Z. and Huang, Q. (2000). A new distance measure for probability distribution function of mixture type. In *Proceedings of ICASSP*, volume 1, pages 616–619, Istanbul, Turkey.

Mahler, R. (2004). Multitarget miss distance via optimal assignment. *IEEE Trans. on Systems Man. Cyb.*, **34**(3).

Mahler, R. P. S. (2003). Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, **39**(4), 1152–1178.

Mahler, R. P. S. (2007). *Statistical Multisource-Multitarget Information Fusion.* Artech House, Norwood, MA.

Morelande, M. R., Kreucher, C. M., and Kastella, K. (2007). A Bayesian approach to multiple target detection and tracking. *IEEE Transactions on Signal Processing*, **55**(5), 1589–1604.

Ong, L.-L., Upcroft, B., Ridley, M., Bailey, T., Sukkarieh, S., and Durrant-Whyte, H. (2006). Consistent methods for decentralized data fusion using particle filters. In *Proceedings of IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 85–91.

Ong, L.-L., Bailey, T., Durrant-Whyte, H., and Upcroft, B. (2008). Decentralized particle filtering for multiple target tracking in wireless sensor networks. In *Proceedings of the 11th International Conference on Information Fusion*, pages 342–349, Cologne, Germany.

Panta, K., Vo, B., and Singh, S. (2005). Improved probability hypothesis density (PHD) filter for multitarget tracking. In *Proceedings of the 3rd Conf. on Intelligent Sensing and Information Processing*, pages 213–218, Bangalore, India.

Panta, K., Vo, B.-N., and Singh, S. (2007). Novel data association schemes for the probability hypothesis density filter. *IEEE Transactions on Aerospace and Electronic Systems*, **43**, 556–570.

Papoulis, A. and Pillai, S. U. (2001). *Probability, Random Variables and Stochastic Processes.* McGraw-Hill, 4th edition.

Pattipati, K. R., Popp, R. L., and Kirubarajan, T. (2000). Survey of assignment techniques for multitarget tracking. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Advanced Applications, Vol. III*, chapter 2, pages 77–160. Artech House.

Poore, A. and Rijavec, N. (1993). A Lagrangian relaxation algorithm for multidimensional assignment problems arising from multitarget tracking. *SIAM Journal of Optimization*, **3**(3), 544–563.

Poore, A. and Robertson-III, A. J. (1997). A new Lagrangian Relaxation based algorithm for a class of multidimensional assignment problems. *Computation and Optimization and Applications*, **8**, 129–150.

Popp, R., Pattipati, K. R., and Bar-Shalom, Y. (2001). An m-best S-D assignment algorithm and parallelization with application to multitarget tracking. *IEEE Trans. on Aerospace and Electronic Systems*, **37**(1), 22–39.

Rachev, S. T. and Ruschendorf, L. (1998). *Mass Transportation Problems, Volume I: Theory.* Springer-Verlag, New York.

Radar Corresponding Group (2006). RCG-45 Impulse Noise Interference Effect on Binary Detection. Contributions, Radar Corresponding Group.

Reid, D. B. (1979). An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, **24**(6), 843–854.

Scharf, L. L. (1991). *Statistical Signal Processing.* Addison-Wesley, Reading, Mass.

Schuhmacher, D., Vo, B. T., and Vo, B.-N. (2008). A consistent metric for performance evaluation of multi-object filters. *IEEE Transactions on Signal Processing*, **56**(8), 3447–3457.

Sidenbladh, H. (2003). Multi-target particle filtering for the probability hypothesis density. In *Proceedings of the 6th International Conference on Information Fusion*, pages 800–806, Cairns, Australia.

Sinha, A., Kirubarajan, T., and Bar-Shalom, Y. (2005). Autonomous Surveillance by Multiple Cooperative UAVs. In *SPIE Proceedings*, volume 5913, pages 616–627.

Sinha, A., Chen, H., Danu, D. G., Kirubarajan, T., and Farooq, M. (2008). Estimation and Decision Fusion: A Survey. *Neurocomputing*, **71**(13:15), 2650–2656.

Tian, X. and Bar-Shalom, Y. (2009). Track-to-Track Fusion Configurations and Association in a Sliding Window. *Journal of Advances in Information Fusion*, **4**(2), 146–163.

Tian, X. and Bar-Shalom, Y. (2010). Algorithms for Asynchronous Track-to-Track Fusion. *Journal of Advances in Information Fusion*, **5**(2), 128–139.

Vasershtein, L. N. (1969). Markov processes over denumerable products of spaces describing large systems of automata. *Problems of Information Transmission*, **5**(3), 47–52.

Vermaak, J., Godsill, S. J., and Perez, P. (2005). Monte Carlo methods for multi-target tracking and data association. *IEEE Transactions on Aerospace and Electronic Systems*, **41**(1), 309–332.

Villani, C. (2003). *Topics in Optimal Transportation*, volume 58 of *Graduate Series in Mathematics*. American Mathematical Society, Providence, RI.

Villani, C. (2009). *Optimal Transport, Old and new*, volume 338 of *A series oj comprehensive studies in Mathematics*. Springer-Verlag, Berlin, Heidelberg.

Vo, B.-N., Singh, S., and Doucet, A. (2005). Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Transactions on Aerospace and Electronic Systems*, **41**(4), 1224–1245.

You, H. and Jingwei, Z. (2006). New Track Correlation Algorithms in a Multisensor Data Fusion System. *IEEE Transactions on Aerospace and Electronic Systems*, **42**(4), 1359–1371.

Zajic, T. and Mahler, R. (2003). A particle-systems implementation of the PHD multitarget tracking filter. In I. Kadar, editor, *Proceedings of 2003 SPIE Conference on Signal Processing, Sensor Fusion and Target Recognition XII*, volume 5096, pages 279–290, Orlando, FL.

Zhang, Y. (1999). User's guide to LIPSOL linear-programming interior point solvers v0.4. *Optimization Methods and Software*, **11**, 385–396.