B-Spline Based Multitarget Tracking

B-SPLINE BASED MULTITARGET TRACKING

BY

RAJIV SITHIRAVEL, M.ENG.

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 (2014) McMaster University
(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE:	B-Spline Based Multitarget Tracking
AUTHOR:	Rajiv Sithiravel B.Eng., (Electrical Engineering) Ryerson University, Toronto, Canada, 2006 M.Eng., (Electrical Engineering) University of Queensland, Brisbane, Australia, 2007 M.Eng., (Electrical Engineering) McMaster University, Hamilton, Canada, 2011
SUPERVISOR:	Dr. Thiagalingam Kirubarajan
NUMBER OF PAGES:	xxi, 199

To my family

Abstract

Multitarget tracking in the presence of false alarm is a difficult problem to consider. The objective of multitarget tracking is to estimate the number of targets and their states recursively from available observations. At any given time, targets can be born, die and spawn from already existing targets. Sensors can detect these targets with a defined threshold, where normally the observation is influenced by false alarm. Also if the targets are with low signal to noise ratio (SNR) then the targets may not be detected.

The Random Finite Set (RFS) filters can be used to solve such multitarget problem efficiently. Specially, one of the best and most widely used RFS based filter is the Probability Hypothesis Density (PHD) filter. The PHD filter approximates the posterior probability density function (PDF) by the first order moment only, where the targets SNR assumed to be much higher. The PHD filter supports targets die, born, spawn and missed-detection by using the well known implementations including Sequential Monte Carlo Probability Hypothesis Density (SMC-PHD) and Gaussian Mixture Probability Hypothesis Density (GM-PHD) methods. The SMC-PHD filter suffers from the well known degeneracy problems while GM-PHD filter may not be suitable for nonlinear and non-Gaussian target tracking problems.

It is desirable to have a filter that can provide continuous estimates for any distribution. This is the motivation for the use of B-Splines in this thesis. One of the main focus of the thesis is the B-Spline based PHD (SPHD) filters. The Spline is a well developed theory and been used in academia and industry for more than five decades. The B-Spline can represent any numerical, geometrical and statistical functions and models including the PDF and PHD. The SPHD filter can be applied to linear, nonlinear, Gaussian and non-Gaussian multitarget tracking applications. The SPHD continuity can be maintained by selecting splines with order of three or more, which avoids the degeneracy-related problem. Another important characteristic of the SPHD filter is that the SPHD can be locally controlled, which allow the manipulations of the SPHD and its natural tendency for handling the nonlinear problems. The SPHD filter can be further extended to support maneuvering multitarget tracking, where it can be an alternative to any available PHD filter implementations.

The PHD filter does not work well for very low observable (VLO) target tracking problems, where the targets SNR is normally very low. For very low SNR scenarios the PDF must be approximated by higher order moments. Therefore the PHD implementations may not be suitable for the problem considered in this thesis. One of the best estimator to use in VLO target tracking problem is the Maximum-Likelihood Probability Data Association (ML-PDA) algorithm. The standard ML-PDA algorithm is widely used in single target initialization or geolocation problems with high false alarm. The B-Spline is also used in the ML-PDA (SML-PDA) implementations. The SML-PDA algorithm has the capability to determine the global maximum of ML-PDA log-likelihood ratio with high efficiency in terms of state estimates and

low computational complexity. For fast passive track initialization, search and rescue operations the SML-PDA algorithm can be used more efficiently compared to the standard ML-PDA algorithm. Also the SML-PDA algorithm with the extension supports the multitarget tracking.

Acknowledgements

This thesis signifies a milestone in five years of my work at McMaster University. From the first day at McMaster I have felt at home and my experiences at McMaster has been nothing short of startling. This thesis is the result of many experiences I have encountered at McMaster University from number of remarkable individuals whom I wish to acknowledge.

First and foremost I wish to thank my advisor, professor Dr. T. Kirubarajan for offering me an opportunity to be one of his graduate student. He has been supportive since the day I began working on the "Performance Evaluator" for the testbed as a master student. His knowledge and guidance provided me unique opportunities. During my PHD studies, Kiruba contributed to a rewarding graduate school experience by giving me intellectual freedom in my work, supporting my attendance at conferences, engaging me in new ideas and demanding a high quality of work in all my endeavours. I am very fortunate to have crossed paths with him and I am very grateful for his guidance in this journey.

Next, I would like to thank my fellow lab-mates and collaborators. Notably, Dr. R. Tharmarasa and Dr. X. Chen who had worked with me on several different phase of the research. Undoubtedly, I benefited greatly from their keen scientific insight, knowledge of solving seemingly intractable practical difficulties and abilities to put complex ideas into simple terms. I also like to express my gratitude to various members of Defence Research and Development of Canada with whom I had the opportunity to work on number of projects and publications. Specifically, Dr. M. McDonald, Dr. S. Rajan and Dr. A. Yasotharan provided me a friendly and cooperative atmosphere during meetings and constructive feedback and insightful comments on my research work, publications and software developments.

Additionally, I would like to acknowledge the Department of Electrical and Computer Engineering at McMaster University, particularly Dr. T. R. Field, Dr. S. Kumar, Dr. J. P. Reilly, Dr. I. Bruce, Dr. A. Jeremic and Dr. C. H. Chen for their guidance and encouragement. Also, to Cherly Gies, for her extraordinary administrative support and vibrant enthusiasm.

Finally, I would like to thank my family for all their love and support. In particular to my wife, Thiba for her support and kindness during all these years and my son Hamish for his patience. I thank my sisters for their understanding and my parents for giving me a chance to live in this wonderful world.

Abbreviations and notation

ASMC	Auxiliary Sequential Monte Carlo
AP	Auxiliary Particle
AI	Amplitude Information
AOA	Angle of Arrival
DSS	Directed Subspace Search
EKF	Extended Kalman Filter
FDOA	Frequency Difference of Arrival
GM	Gaussian Mixture
GMP	Gaussian Mixture Particle
GS	Genetic Search
IMM	Interacting Multiple Model
IS	Importance Sampling
JPDAF	Joint Probability Data Association Filter
JML-PDA	Joint Maximum-Likelihood Probabilistic Data Association
JLLR	Joint Log Likelihood Ratio
KF	Kalman Filter
LLR	Log Likelihood Ratio
ML-PDA	Maximum Likelihood Probabilistic Data Association

MM	Multiple Model
MeMBer	Multitarget Multi-Bernoulli
MHT	Multiple Hypothesis Tracker
MPG	Multipass Grid
MCR	Monte Carlo Runs
NNSF	Nearest Neighbor Standard Filter
NEES	Normalized Estimation Error Squared
OSPA	Optimal Sub-pattern Assignment
PHD	Probability Hypothesis Density
PDF	Probability Density Function
RFS	Random Finite Sets
RPF	Regularized Particle Filter
RMSE	Root Mean Squared Error
SNR	Signal to Noise Ratio
SMC	Sequential Monte Carlo
SPHD	Spline Probability Hypothesis Density
SAS	Surveillance Aircraft System
TOA	Time of Arrival
TDOA	Time Difference of Arrival
TBD	Track Before Declare
USMC	Unscented Sequential Monte Carlo
UIF	Unscented Information Filter
UAVS	Unmanned Aerial Vehicle Systems
VLO	Very Low Observable

X_k	Multitarget system state
Z_k	Observation
$\mathbf{M}_k$	Mode number
$\mathbf{x}_{l,k}^{\mathbf{M}_k}$	l-th target state vector
$p_{k-1\|k-1}(X_{k-1}\|Z^{(k-1)})$	Multitarget prior density
$p_{k\|k-1}(X_k\|Z^{(k-1)})$	Multitarget predicted density
$p_{k\|k-1}(X_k\|X_{k-1})$	Multitarget transition density
$g_k(Z_k\|X_k)$	Multitarget likelihood density
$p_{k\|k}(X_k\|Z^{(k)})$	Multitarget posterior density
$D_{k-1\|k-1}(\mathbf{x}_{k-1}\|Z^{(k-1)})$	Prior PHD
$D_{k\|k-1}(\mathbf{x}_k\|Z^{(k-1)})$	Predicted PHD
$D_{\mathbf{c},k\|k-1}(\mathbf{x}_k)$	Predicted PHD of existing targets
$D_{\mathbf{s},k\|k-1}(\mathbf{x}_k)$	Predicted PHD of spawned targets
$D_{\mathbf{nb},k}(\mathbf{x}_k)$	Predicted newborn target PHD
$\hat{N}_{k\|k-1}$	Predicted expected number of targets
$D_{k\|k}(\mathbf{x}_k\|Z^{(k)})$	Posterior PHD
$\tilde{D}_{k\|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}\|Z^{(k-1)})$	Mode-dependent initial density
$D_{k-1\|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}\|Z^{(k-1)})$	Mode-dependent prior density
$\pi_{\mathfrak{p}\mathfrak{q}}$	Transition probability matrix
$D_{k\|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}\|Z^{(k-1)})$	Mode-dependent predicted PHD

$D_{k\|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}\|Z^{(k)})$	Mode-dependent updated PHD
$\mathbf{x}_r$	Multidimensional target system state at time r
$Z(i)$	Single data frame
$\mathbf{z}_j(i)$	Observation from i-th frame and j-th measurement
β_{ij}	Kinematic measurements
a_{ij}	Amplitude measurements
ψ_{ij}	Frequency measurements
$\mathbf{Z}$	Cumulative set of measurement
N_w	Number of frames
ρ_{ij}	Amplitude likelihood ratio
P_d	Probability of detection
P_{fa}	Probability of false alarm
$\mathbf{x}_s(i)$	Sensor-platform state
$\gamma_i(\mathbf{x}_r)$	Noise free doppler shifted frequency
U_β	Kinematic surveillance region's volume
U_γ	Frequency region's volume
$p\left(Z(i)\|\mathbf{x}_r\right)$	Individual likelihood functions
$\Lambda\left(\mathbf{Z}\|\mathbf{x}_r\right)$	Log-likelihood ratio
$\mathbf{x}_r^\ell$	ℓ-th target state
β_{ij}^ℓ	ℓ-th target originated kinematic measurement
ψ_{ij}^ℓ	ℓ-th target generated frequency measurement
$\gamma_i(\mathbf{x}_r^\ell)$	ℓ-th target originated Doppler shifted frequency

$\mathfrak{c}$	Velocity of sound in the medium
ρ_{ij}^{ℓ}	Amplitude likelihood ratio for the ℓ-th target
L_i	Target detection events
P_d^{ℓ}	Detection probability for the ℓ-th target
t_{em}	Unknown time of emission
$\mathbf{B}_{k-1\vert k-1}(\mathbf{x}_{k-1})$	Prior SPHD
$\mathbf{B}_{k\vert k-1}(\mathbf{x}_k\vert Z^{(k-1)})$	Predicted SPHD
$\mathbf{B}_{\mathbf{c},k\vert k-1}(\mathbf{x}_k)$	Predicted SPHD of existing targets
$\mathbf{B}_{\mathbf{s},k\vert k-1}(\mathbf{x}_k)$	Predicted SPHD of spawned targets
$\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k)$	Predicted newborn target SPHD
$\mathbf{B}_{k\vert k}(\mathbf{x}_k)$	Posterior SPHD
$\tilde{\mathbf{B}}_{k\vert k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent initial SPHD
$\mathbf{B}_{k\vert k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent predicted SPHD
$\mathbf{B}_{\mathbf{c},k\vert k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent predicted SPHD of existing targets
$\mathbf{B}_{\mathbf{s},k\vert k-1}(X_k, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent predicted SPHD of spawned targets
$\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent predicted SPHD of newborn targets
$\mathbf{B}_{k\vert k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$	Mode-dependent updated SPHD
$\mathcal{E}_s$	System state space
$\mathcal{E}_o$	Observation space

Contents

List of Figures

Chapter 1

Introduction and Problem Statement

1.1 Multitarget Tracking

Multitarget tracking is a dynamic state estimation problem [3,5], in which both the number of targets and the corresponding states are unknown and vary with time due to target appearance and disappearance at random times. Moreover, not all existing targets are detected by the sensors at every sampling time due to thresholding. In addition, the measurements received from sensors might have originated either from targets or from clutter and some may be false alarms. As a result, the observation set at each time step is a collection of indistinguishable partial observations, where only some of these observations are actually generated by the targets. The objective of multitarget tracking is to jointly estimate the number of targets and their states from measurements with uncertain origins.

A common method to solve the multitarget tracking problem is to use a data association technique [3] to assign a measurement to each track followed by the application of a single target filter [2,36,38] (e.g., a Kalman Filter (KF) for a linear system model or an Extended Kalman Filter (EKF) for a nonlinear system model [3]) to that track. There are several strategies in the literature to solve the data association problem. Some widely-used classical approaches are the Nearest Neighbor Standard Filter (NNSF) [2], Joint Probability Data Association Filter (JPDAF) [2] and the Multiple Hypothesis Tracker (MHT) [2].

The NNSF predicts the measurement for each target state and associates the prediction with the closest measurement. In contrast, the MHT keeps track of all possible association hypotheses over time. This is an NP-hard problem, since the number of associations grows exponentially over time, thus the MHT requires additional simplifications like pruning and merging to reduce the computational complexity. The JPDAF is a more appealing approach in that during each time update many hypotheses are merged to form a single track hypothesis following a validation process. A state estimate is then calculated for each remaining hypothesis and combined in proportion to the corresponding posterior hypothesis probabilities. The combination of multiple measurement-constrained estimates in the JPDAF can lead to problems such as track coalescence and latency in the initialization [41] of new tracks. Due to its combinatorial nature, data association in hypothesis enumeration techniques like MHT and JPDAF constitute 60%–90% of the computational load [47]. However, there are data association formulations that avoid explicit associations between measurements and tracks, including multitarget particle filters [18] and the Random Finite Sets (RFS) method [36,38,54].

1.2 Random Finite Sets Based Filtering

Since there is no ordering on the respective collections of target states and measurements at a certain time in association, they can be naturally represented as finite sets. This finite set property is also applied in the RFS formulation [36,38]. Modeling set-valued states and set-valued observations as RFS permits the Bayesian filtering framework [63] to estimate the multitarget states in the presence of clutter, missed detections and association uncertainty. Two most tractable alternatives to the optimal multiple target filter are the RFS based Probability Hypotheses Density (PHD) filter [36,38] and the Multitarget Multi-Bernoulli (MeMBer) filter [36,38,64].

The PHD filter is a recursion that propagates the first order statistical moment, or the intensity of the RFS of states over time [38]. The mathematical derivations of the PHD filter equations are based on probability generating functionals. The integral of the PHD in any region of the state space is the expected number of targets in that region. The PHD filter can track a time-varying number of targets and their PHD without the need for multitarget data association between measurements and tracks.

The Multitarget Multi-Bernoulli filter [36,71] is based on the assumption that every multitarget posterior is a multitarget multi-Bernoulli process. This approximation can be predicted forward recursively in time and updated with a set of measurements in a manner similar to the PHD filter recursion. The MeMBer filter can be extended into a full multitarget tracking algorithm with the inclusion of track labels to form a track table that can be recursively propagated to provide a full multitarget solution including track ID, existence likelihood and state uncertainty. Not only can the RFS based PHD filter and MeMBer filter successfully track non-maneuvering targets, they can also handle tracking maneuvering targets with two multiple model implementations, the Sequential Monte Carlo (SMC) method [18,36,45], and the Gaussian Mixture (GM) method [20,46,63,64,71]. Both the PHD filter and the MeMBer filter yield similar performances [36,54,65]. The main focus of this first part of the thesis is the PHD filter.

Two distinct PHD filter implementations are available in the literature: the Sequential Monte Carlo Probability Hypothesis Density (SMC-PHD) and the Gaussian Mixture Probability Hypothesis Density (GM-PHD) filters. The SMC-PHD filter implementation [65] consists of approximating the PHD by a set of weighted particles and does not need any further assumptions. The GM-PHD filter implementation [63] assumes that the PHD is a Gaussian mixture (GM). The design of the importance sampling (IS) function critically affects the filtering performance [18] of the SMC-PHD filter. One of the widely used methods is to approximate the IS function by the transition density [65]. Also, one can use the auxiliary particle approach to incorporate the measurement into the IS function as in the Auxiliary Sequential Monte Carlo Probability Hypothesis Density (ASMC-PHD) filter [69], and an improved version of the ASMC-PHD filter is the Auxiliary Particle Probability Hypothesis Density (AP-PHD) [4] filter. The Gaussian Mixture Unscented Sequential Monte Carlo Probability Hypothesis Density (GM-USMC-PHD) [73] filter uses the Gaussian mixture representation to approximate the IS and the predictive density functions via the Unscented Information Filter (UIF) [35]. The performance of particle based filters may be affected by degeneracy-related problems, which can be alleviated by using the Regularized Particle Filter (RPF) [19,44]. Combined GM-particle based implementations of the PHD filter were introduced in the Gaussian Mixture Particle Probability

Hypothesis Density (GMP-PHD) filter [15,72,74] and the Gaussian Mixture Sequential Monte Carlo Probability Hypothesis Density (GM-SMC-PHD) filter [50]. Note, GM-based methods may not be suitable for highly nonlinear non-Gaussian systems.

1.2.1 Original Contributions: SPHD and MM-SPHD Filters

The Spline Probability Hypothesis Density Filter

It is desirable to have a filter that can provide continuous estimates for any distribution. This provides the motivation behind the newly proposed Spline Probability Hypothesis Density (SPHD) filter. The proposed SPHD filter is developed based on B-Splines [17,53]. Any arbitrary geometrical, numerical or statistical curve can be represented by splines. This includes the multitarget probability density function or the probability hypothesis density in the PHD filter. The spline based multitarget PHD can be locally controlled [17] and such as permits manipulations on the PHD [11]. The spline based multitarget PHD is a polynomial of finite order and finding the spline based multitarget PHD at any point in the state space involves evaluation of a polynomial at that point [67]. The polynomial order of the multitarget PHD determines its continuity. The C^1 continuity [53] of the multitarget PHD can be maintained by selecting splines with order of three or more [53].

The B-Spline approach to multitarget tracking is an alternative to the SMC-PHD, the GM-PHD, the GMP-PHD, the GM-USMC-PHD, the GM-SMC-PHD and the AP-PHD filters. The SPHD filter offers continuous estimates of the PHD function of the state for any system model and avoids degeneracy by providing continuous estimates. The nonlinearity of the measurement model is naturally handled by the SPHD filter. The SPHD filter is not limited to Gaussian systems. The B-Spline method for single target tracking in a clean environment, i.e., no false alarms or missed detections, is presented in [11,25,33,48]. It is shown that not only does the B-Spline approach of single target tracking work well for one dimensional nonlinear non-Gaussian filtering problems, but also for multidimensional systems with the use of tensor products of splines [24,25,48].

Chapter 4 presents the SPHD filter derivations for a multitarget tracking problem. The multitarget multidimensional system state transition model of a SPHD filter is represented by tensor products of splines. The corresponding analytical SPHD prediction and posterior update equations are derived. A nonlinear non-Gaussian example is used to validate the performance of the proposed filter. The proposed SPHD filter interpolates the multitarget PHD function only over the region in which the multitarget density is significant, increasing the efficiency of the SPHD filter. The multitarget density outside the interval is assumed to be zero. Simulation results in Chapter 4 reveal that the SPHD filter works efficiently and increased measurement noise levels do not destabilize the SPHD tracker. The SPHD filter can maintain highly accurate tracks by taking advantage of dynamic knot movement, but at the expense of higher computational complexity, which makes it suitable for tracking few high-value targets under difficult conditions.

Spline Probability Hypothesis Density Filter for Nonlinear Maneuvering Target Tracking

To extend the application of the SPHD filter to maneuvering multitarget tracking problems, a multimodal version, called the Multiple Model (MM) Spline Probability Hypothesis Density (MM-SPHD) filter, is derived. The best choice for implementing multiple model estimation is the Interacting Multiple Model (IMM) [2] estimator. The IMM is a suboptimal filter that has been shown to be one of the most cost-effective state estimation schemes. Note, the standard IMM estimator assumes that the densities are Gaussian where the mode-dependent PHD filter is not.

Therefore, the integration of the IMM estimator into mode-dependent PHD filter algorithm is not preferred. Note, the IMM estimator uses only the first and the second order statistics to estimate the states, but for multiple-model multitarget state problems that cannot be approximated using the IMM estimator due to densities can be multi-modal when they represent multitarget states. Therefore, the multiple model implementation used here adopts a method similar to the one used in the MM-PHD [49] filter. This new MM-SPHD filter not only works well for tracking maneuvering targets, but also inherits all the capabilities of the SPHD filter [58].

Chapter 5 presents the MM-SPHD filter derivations with details on the estimation of maneuvering multitarget state and the extraction of corresponding individual target states. The multidimensional multitarget system state transition model of the MM-SPHD filter is represented by tensor products of splines. The corresponding analytical state prediction and posterior density equations are derived. A nonlinear example is used to validate the performance of the MM-SPHD filter vs. those of other multiple model PHD implementations. Simulation results reveal that the MM-SPHD filter works efficiently and increased measurement noise levels do not destabilize it whereas other MM implementation suffer at higher noise levels. The MM-SPHD filter can maintain highly accurate tracks by taking advantage of dynamic knot movement [58], but at the expense of higher computational complexity.

1.3 Passive Track Initialization for a Very Low Observable Target

In passive and active radar or sonar target tracking problems very low observable (VLO) targets are problematic [2], due to targets being masked by lofty background noises. Note, that the target with very low signal to noise ratio (SNR) is dim or very low observable. The target with low SNR can be detected by lowering the detection threshold with sufficient probability of detection P_d to track [2], but lower threshold level leads to higher probability of false alarm P_{fa}, results in high clutter. In addition, a high value of false alarm can be a problematic for data association in any standard tracking algorithm. The VLO target tracking problems in presence of clutter can be handel by using the well known track-before-declare (TBD) [2]. Where TBD either uses unthresholded sensor data or threshold data. TBD method uses remarkably lower threshold or no threshold where compared to other standard trackers and can be implemented by using recursive method or batch processing method [2,7]. Several frames of measurement data are used in TBD's batch processing method to extract or initialize target state. TBD's techniques include maximum likelihood-probabilistic data association (ML-PDA) algorithm [2]. Note, for low SNR and high false alarm target tracking problem the PHD filter under performs.

Generally, the single target tracking problem using azimuth and frequency measurements of many scans with high clutter or lower SNR scenario is a very ill conditioned problem [31]. The ML-PDA algorithm was introduced in [27] to solve such scenarios. The ML-PDA algorithm with the assumption of deterministic target motion uses no or low threshold level measurement data scans to determine track estimates [2]. Features such as the amplitude information (AI) as in [31] was also used to improve the performance of the ML-PDA algorithm. Where the amplitude information is integrated into the ML-PDA likelihood function [31]. By maximizing the log-likelihood ratio (LLR) the target state can be extracted from a batch of measurements. Sliding window implementation can be applied for real-time applications. Since the ML-PDA algorithm uses a very low or sometime no detection level thresholding, the complexity is higher compared to other standard trackers [3,2]. For initializing a single-target tracking with high clutter or low SNR the ML-PDA algorithm performs very well and standard ML-PDA algorithm cannot apply for multitarget tracking problems.

The ML-PDA algorithm has been used widely in a single target tracking by using variety of data sets i.e., optical [14] and active sonar [10]. Also the ML-PDA algorithm is used in multistatic tracking [70]. The ML-PDA recursive estimator as in [32] provides accurate initial estimates. The standard ML-PDA algorithm was further extended to a multitarget scenario as in [8,9,29]. Multiple targets can also be initialized in the existence of false alarms and missed detections as in [13]. Despite having the ability to handle VLO targets for target tracking, the ML-PDA algorithm's performance degraded by high-computational complexity and the lack of efficiency improvements [8]. The ML-PDA algorithm's track estimates have to satisfy the track validation process due to the fact that the outcome may come from no target or false alarm [2]. The main focus of this section is the initialization of VLO target using the ML-PDA estimator in target tracking.

The scalar value of the ML-PDA estimator's state estimates maximizes the highly non-convex ML-PDA LLR. The LLR function is made out of many local maxima and the region of LLR may have its maximum value close to insignificant gradient. The global maximum of LLR can be determined by three available methods i.e., Multipass Grid (MPG), Genetic Search (GS) and Directed Subspace Search (DSS) [8,10,27,31]. In terms of lower computational complexity the DSS is shown to performs the best [8]. Although the GS method is little used in tracking community [22] the GS method is used to find the maximum global of ML-PDA LLR in [8] show that the improvement in both low complexity as well as better performance than MPG. The DSS method easily identifies the areas in the parameter space that include the global maxima. This method is viable if the measurement space is a subspace of the parameter space [10].

1.3.1 Original Contributions: SML-PDA Algorithm

Spline Maximum-Likelihood Probabilistic Data Association Algorithm for Track Initialization

Any arbitrary geometrical, numerical or statistical shapes can be represented by B-Spline [12] including ML-PDA LLR. The B-Spline is used in many areas including target tracking [33,48,58,67], where the prior, predicted and posterior probability densities are represented by B-Splines using a finite set of knots. The B-Spline based filters [58,57] are applicable to any scenarios i.e., linear, non-linear, Gaussian and non-Gaussian problems. Using B-spline can model any probability density functions and probability hypotheses density [58] functions without any assumption regarding the system and measurement noises [25,58]. A spline approach is used to solve the nonlinear estimation problem of phase modulation in [11], B-splines using genetic algorithm for the optimization of a cost function is discussed in [23] and using monosplines to solve a nonlinear estimation problem is discussed in [67]. The B-Spline method for single target tracking in a clean environment, i.e., no false alarms or missed detections, is presented in [11,33,48]. The multidimensional target tracking problem can be solved as [24,25,48] by using tensor products of splines. The Spline filter can also track the multitarget state in conjunction with a data association algorithm as in [58] and an extension to [58] as in [57] supports the multiple model target tracking.

Considering all the capabilities of splines in Chapters 4 and 5, a new way of determining the global maximum of ML-PDA LLR by using B-Spline is implemented and is termed the Spline Maximum-Likelihood Probabilistic Data Association (SML-PDA) algorithm. The newly proposed algorithm represents the LLR in terms of B-Splines. The SML-PDA LLR can be of any distribution. Using a few knots SML-PDA LLR can be represented and the optimal knot selection, which is an iterative method. Iterative method can be used to create an optimal knot vector reverence to a given parameter. The global maximum of SML-PDA LLR can be easily determined by finding the knot element with a higher distribution. The SML-PDA LLR distribution can be linear, nonlinear, Gaussian or non-Gaussian. The tensor product of splines can be used to represent the single target multidimensional system state transition model. The analogous target originated likelihood function, likelihood function given that all the measurement are false detection and log-likelihood ratio are derived [2] in terms of B-Spline. Note, the AI integration to the ML-PDA LLR is also explained by the B-Splines formulation. The formulation also support the narrowband sonar scenario. From the parameter space a track is estimated using the LLR global maximum followed by the SML-PDA track validation [31]. The

SML-PDA algorithm can determine the global maximum of SML-PDA LLR with high efficiency and low complexity. For higher dimensional target state tracking problem (i.e., the sonar scenario) the SML-PDA algorithm performs well, but with a higher computational complexity. The proposed algorithm's performance is evaluated by using a nonlinear example both with and without AI for a bearing-only scenario and a narrowband sonar scenario. The SML-PDA algorithm is affected by the curse of dimensionality, which affects the computational complexity. But the performance in terms of target state initialization with low root means square error (RMSE) and high track acceptance is never compromised with increasing dimension of the target states.

1.4 Geolocating Very Low Observable Multitarget in High Clutter

During a reconnaissance mission in an enemy territory by the use of Unmanned Aerial Vehicle Systems (UAVS) or any Surveillance Aircraft System (SAS) the geolocation of Radars important [40]. Note, the reconnaissance is a military observation of a region to locate an enemy or ascertain strategic features i.e., emitters, missile batteries and submarine communication. On the other hand geolocation is also applied in civilian areas i.e., search and rescue missions of missing airlines/vessels, locating mobile phones [52] and used in intelligent transport and taxi systems [51]. In all of these cases the ascertained strategic features normally emit signals in presence of false alarm and the ascertained strategic features can be stationary or mobile.

There are several techniques for accurate positioning of the ascertain strategic feature including angle of arrival (AOA), time of arrival (TOA), time difference of arrival (TDOA) and frequency difference of arrival (FDOA) methods [26,39]. AOA is not suitable for airborne surveillance system. The maintenance of an array antenna and precise calibrations are required for AOA technique. The TOA technique is used in Global Positioning System (GPS) and requires the knowledge of time of emission. However, often in unfriendly environment the time of emission is unknown. This issue can be solved by using the TDOA technique as along as the synchronization between receivers can be maintained. If there is a relative motion between the emitter and receiver the FDOA technique performs well. All of these techniques do not work well in the presence of false alarm and/or very low signal to noise ratio (SNR). In military applications often the emitted-signal is very low observable (VLO) with high false alarm. Even in civilian cases i.e., search and rescue of an airplane's missing black-box in an ocean is a VLO geolocation problem.

Generally to initiate or prolong track in VLO single target tracking problem the standard Maximum-Likelihood Probabilistic Data Association (ML-PDA) algorithm [2] is widely used. The standard ML-PDA algorithm supports the initialization of multitarget in target tracking problems by the use of its multitarget extension the Joint Maximum-Likelihood Probabilistic Data Association (JML-PDA) [2, 9] algorithm. The JML-PDA algorithm extracted the joint track estimates from the batch of measurements by maximizing the joint log-likelihood ratio (JLLR) followed by the track validations. As the number of targets increase the performance of the JML-PDA algorithm degrades. Practical multitarget tracking application of JML-PDA algorithm for more than 3 targets may not be possible [9]. A combined ML-PDA and JML-PDA algorithms (MT-MLPDA filter) can be used for multitarget tracking as in [9]. The MT-MLPDA filter can handel track initiation, track maintenance/update and track termination. The MT-MLPDA filter performs well for linear, Gaussian multitarget tracking problems. The main focus here is the geolocation of VLO target using the JML-PDA algorithm in multitarget tracking. Since only the initialization/geolocation issues are considered, the affect of the number of targets is not a concern. Note, in search and rescue operation the computational complexity is not a problem, but the state estimate accuracy is an important factor. For example, consider a practical scenario i.e., geolocation of missing war-planes with sophisticated payload in a deep ocean with mountainous seabed. Here search operation may take weeks, months or years due to depth, the mountain-clutter, weather and under water current. But locating the position of the VLO signals is important due to environmental and security concern. So the JML-PDA algorithm can be applicable in certain practical multitarget geolocation problems.

1.4.1 Original Contributions: SJML-PDA Algorithm

Geolocating Multitarget in High Clutter Using the B-Spline Based Joint Maximum-Likelihood Probabilistic Data Association Algorithm

As with the application of B-Spline in ML-PDA algorithm a similar implementation is used here for the JML-PDA algorithm to improve performance. The proposed estimator is called the Spline Joint Maximum-Likelihood Probabilistic Data Association (SJML-PDA) algorithm. The SJML-PDA algorithm performs well for VLO multitarget tracking in high false alarm. The SJML-PDA algorithm initializes or geolocates the targets using the batch of measurements by maximizing the spline joint log-likelihood ratio (SJLLR). Note, the SJML-PDA LLR can be of any distribution. The optimal knot selection is used to represent the SJML-PDA LLR. The global maximum of SJML-PDA LLR can be easily determined by finding the knot element with a higher distribution peak. A tensor product of splines can be used to represent the multitarget multidimensional system state transition model. Note, the SJML-PDA LLR distribution can be linear, nonlinear, Gaussian or non-Gaussian. Here, formulations are derived based on a narrow-band sonar multitarget tracking problem. Formulation is also consider the AI information. The tracks are initialized or geolocated using the SJLLR global maximum and then track is validated [31] to identify the origin of the targets. The SJML-PDA algorithm is inherited all the capabilities of the SML-PDA algorithm. The proposed algorithm's performance is evaluated by using a nonlinear example both with and without AI for a narrowband sonar scenario. The curse of dimensionality is not a problem for geolocation since the time constrain is not consider in here. The performance is never compromised compared to the standard JML-PDA algorithm.

1.5 Related Publications

1.5.1 Journal Articles

1. Sithiravel, R., Tharmarasa, R., Rajan, S., and Kirubarajan, T., "Geolocating multitarget in high clutter using the B-spline based joint maximum-likelihood probabilistic data association algorithm", To be submitted to *IEEE Transactions on Signal Processing*, 2014.

2. Sithiravel, R., Tharmarasa, R., Rajan, S., and Kirubarajan, T., "Spline maximum-likelihood probabilistic data association algorithm for track initialization", To be submitted to *IEEE Transactions on Signal Processing*, 2014.

3. Sithiravel, R., Chen, X., McDonald, M., and Kirubarajan, T., "Multiple model spline probability hypothesis density filter", To be submitted to *IEEE Transactions on Aerospace and Electronic System*, 2014.

4. He, X., Sithiravel, R., Tharmarasa, R., Bhashyam, B., and Kirubarajan, T., "A spline filter for multidimensional nonlinear state estimation", *Signal Processing*, vol. 102, pp. 282–295, Sep. 2014.

5. Sithiravel, R., Chen, X., Tharmarasa, R., Balaji, B., and Kirubarajan, T., "The spline probability hypothesis density filter", *IEEE Transactions on Signal Processing*, vol. 61, no. 24, pp. 6188–6203, Dec. 2013.

1.5.2 Conference Proceedings

1. Sithiravel, R., Chen, X., McDonald, M., and Kirubarajan, T., "Spline probability hypothesis density filter for nonlinear maneuvering target tracking", *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pp. 1743–1750, Pacific Grove, CA, Nov. 2013.

2. Sithiravel, R., Tharmarasa, R., McDonald, M., Pelletier, M., and Kirubarajan, T., "The spline probability hypothesis density filter", *Proceedings of SPIE*, vol. 8392, pp. 83920E–83920E20, Baltimore, MD, Apr. 2012.

Chapter 2

Multitarget Tracking

2.1 Single Model Multitarget Tracking

The general state space model for a nonlinear multitarget dynamic system is of the form

$$X_k = f(X_{k-1}, \nu_k) \tag{2.1}$$

where X_k is the multitarget system state at time k, ν_k is the iid process noise sequence with known statistics and f is the nonlinear system transition function. The objective of multitarget tracking is to jointly estimate the number of targets and their states from measurement set

$$Z_k = h(X_k, \omega_k) \tag{2.2}$$

with uncertain origin, where Z_k is the observation set at time k while ω_k denotes the iid measurement noise with known statistics. The approach is to use an RFS to represent the Bayesian model for recursively estimating and updating the multitarget density based on measurements received at each time step. Let ϑ_k be the number of targets at time k, and suppose that at time $k-1$, ϑ_{k-1} system states $X_{k-1} = \{\mathbf{x}_{1,k-1}, \ldots, \mathbf{x}_{\vartheta_{k-1},k-1}\}$ are maintained in target state space $\mathcal{E}_s$. At the next time step k, some of these targets may die or new targets may be spawned. The surviving targets evolve to their new states, and new targets may appear independent of existing targets. This results in ϑ_k new states $X_k = \{\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{\vartheta_k,k}\}$. Note that the order in which the states are listed has no significance in the RFS multitarget model formulation. Let $Z^{(k)} : Z_0, Z_1, \ldots, Z_k \in \mathcal{E}_o$ be a sequence of measurement sets over time and η_k measurements $Z_k = \{\mathbf{z}_{1,k}, \ldots, \mathbf{z}_{\eta_k,k}\}$ are received at time k. Only some of these measurements are actually generated by targets. The dimensions of the state and measurement vectors X_k and Z_k are unknown and time-varying. The randomness in the set can be characterized by modeling the multitarget states and multitarget measurements as random finite sets Ξ_k and $\aleph_k$, respectively. Given the realization X_{k-1} of Ξ_{k-1} at time $k-1$, the multitarget states at time k can be modeled by the RFS as [38]

$$\Xi_k = S_k(X_{k-1}) \cup B_k(X_{k-1}) \cup \Gamma_k \qquad (2.3)$$

where $S_k(X_{k-1})$, $B_k(X_{k-1})$, and Γ_k denote the RFS of survived targets, spawned targets and newborn targets, respectively. Similarly, given a realization of X_k of $\aleph_k$, the multitarget measurement can be modeled by RFS as

$$\aleph_k \quad = \quad \Phi_k(X_k) \cup C_k \tag{2.4}$$

where, $\Phi_k(X_k)$ and C_k denote the RFS of measurements generated by X_k and due to clutter, respectively. Let $p_{k-1|k-1}(X_{k-1}|Z^{(k-1)})$ denote the multitarget priori density. Then the predicted and updated densities of the optimal multitarget Bayes filter recursion are given by

$$\begin{aligned} p_{k|k-1}(X_k|Z^{(k-1)}) \quad = \quad & \int p_{k|k-1}(X_k|X_{k-1}) \\ & \cdot p_{k-1|k-1}(X_{k-1}|Z^{(k-1)}) \\ & \cdot \mu_s(dX_{k-1}) \end{aligned} \tag{2.5}$$

and

$$p_{k|k}(X_k|Z^{(k)}) \quad = \quad \frac{p_k(Z_k|X_k)p_{k|k-1}(X_k|Z^{(k-1)})}{\int p_k(Z_k|X_k)p_{k|k-1}(X_k|Z^{(k-1)})\mu_s(dX_k)} \tag{2.6}$$

respectively, where $p_{k|k-1}(X_k|X_{k-1})$ is the system model multitarget Markov transition density, $p_k(Z_k|X_k)$ denotes the multitarget likelihood function and μ_s denotes an appropriate reference measure [65].

2.2 Multiple Model Multitarget Tracking

Maneuvering multitarget tracking algorithms face target motion model uncertainties and to overcome these uncertainties, most multitarget filters adopt multiple-model estimation techniques [30,49]. Thus, multiple-model estimators run each filter in their mode set using the same measurement assuming that the target state evolves according to one of r models in its mode set at any time and fuses the output of those filters to find an overall estimate [30].

Let ϑ_k be the number of targets at time k in multitarget state space $\mathcal{E}_s$. Then the multimodal multitarget state at time k can be written as

$$X_k \quad = \quad \{\mathbf{x}_{1,k}^{\mathbf{M}_k}, \ldots, \mathbf{x}_{\vartheta_k,k}^{\mathbf{M}_k}\} \in \mathcal{E}_s \tag{2.7}$$

where $\mathbf{x}_{l,k}^{\mathbf{M}_k}$ denotes the mode-dependent l-th target state vector at time k and $l \in \{1, \ldots, \vartheta_k\}$. Note that the order in which the multitarget states are listed has no significance in the Random Finite Set (RFS) multitarget model formulation. In the above, $\mathbf{M}_k \in \{1, \ldots, \mathsf{r}\}$ is the mode index parameter, where r is the number of possible models, and the mode index parameter is governed by an underlaying Markov process with mode transition probability

$$\pi_{\mathfrak{p}\mathfrak{q}} = P(\mathbf{M}_k = \mathfrak{q}|\mathbf{M}_{k-1} = \mathfrak{p}) \quad \mathfrak{p}, \mathfrak{q} = 1, 2, \ldots, \mathsf{r} \tag{2.8}$$

The mode transition probability $\pi_{\mathfrak{p}\mathfrak{q}}$ can be assumed time-invariant and independent of the multitarget state. The state of the l-th target is given by

$$\mathbf{x}_{l,k} \quad = \quad f_{k,\mathbf{M}_k}(\mathbf{x}_{l,k-1}, \nu_{k,\mathbf{M}_k}, \mathbf{M}_k) \tag{2.9}$$

where at time k $\mathbf{x}_{l,k}$ denotes the l-th target state, the mode-dependent i.i.d. process noise sequence is denoted as $\nu_{k,\mathbf{M}_k}$ and $f_{k,\mathbf{M}_k}(\cdot)$ is the mode-dependent nonlinear system transition function.

Let $Z^{(k)} = \{Z_0, Z_1, \ldots, Z_k\} \in \mathcal{E}_o$ be the cumulative sets of measurements from time 0 to time k and assume that η_k denotes the number of target-originated measurements at time k. Measurements also consist of observations generated by the false alarm process and assume ϖ_k denotes the number of false measurements at time k. Then the set of measurements at time k in observation space $\mathcal{E}_o$ is given by

$$Z_k \quad = \quad \{\mathbf{z}_{1,k}, \ldots, \mathbf{z}_{\eta_k,k}\} \bigcup \{\mathbf{c}_{1,k}, \ldots, \mathbf{c}_{\varpi_k,k}\} \in \mathcal{E}_o \tag{2.10}$$

where the l-th target-originated measurement is given by

$$\mathbf{z}_{l,k} \quad = \quad h_{k,\mathbf{M}_k}(\mathbf{x}_{l,k}, \omega_{k,\mathbf{M}_k}, \mathbf{M}_k) \tag{2.11}$$

and $\omega_{k,\mathbf{M}_k}$ denotes the mode-dependent i.i.d. measurement noise with known statistics and $h_{k,\mathbf{M}_k}$ is a mode-dependent nonlinear function. The false measurements $\mathbf{c}_{i,k}$ are assumed to be uniformly distributed and their number ϖ_k is Poisson-distributed. Let $P_{\mathfrak{d},k}$ denote the probability of detection, thus the probability of $Z_k(\mathbf{x}_{i,k}^{\mathbf{M}_k}) = \mathbf{z}_{i,k} = \emptyset$ (i.e., the i-th target is not detected) is $1 - P_{\mathfrak{d},k}$. The average number of measurement is $\gimel_k = P_{\mathfrak{d},k} \times \eta_k + \varpi_k^{'}$, where $\varpi_k^{'}$ is the average number of false alarms.

There are ϑ_k targets in state space $\mathcal{E}_s$ at time k and these targets can continue to exist, spawn new targets or terminate. In addition, new targets are born independently of already-existing targets. The number of targets and their states are unknown and, with maneuvering targets, the dynamic model of a target at any time is also unknown. That is, there are two unknown discrete random variables (i.e., number of targets and mode index of each target) and a continuous random variable (i.e., target state of each target) to be estimated at each time. From the observation space $\mathcal{E}_o$, $\beth_k$ measurements are received at time k. The origins of the measurements are not known, and thus the order in which they appear bears no significance. The measurements can also originate from clutter and false alarms.

At time k, the dynamic models of all targets and the dimension of the multitarget state X_k are unknown and time-varying. In the absence of model uncertainty, the randomness in the set can be characterized by modeling the multitarget states and multitarget measurements as random finite set Ξ_k and $\aleph_k$ respectively. Given the realization X_{k-1} of Ξ_{k-1} at time k-1, the multitarget states at time k can be modeled by the RFS as

$$\Xi_k \quad = \quad S_k(X_{k-1}) \cup B_k(X_{k-1}) \cup \Gamma_k \tag{2.12}$$

where $S_k(X_{k-1})$ denotes the surviving targets and $B_k(X_{k-1})$ denotes the spawned targets. In addition, Γ_k denotes the newborn targets and these newborn targets are born independently from the surviving targets. Similarly, given a realization of X_k of $\aleph_k$, the multitarget measurement can be modeled by the RFS as

$$\aleph_k \quad = \quad \Phi_k(X_k) \cup C_k \tag{2.13}$$

where $\Phi_k(X_k)$ denotes the RFS measurement generated by X_k and C_k denotes measurement generated by clutter.

Let $p_{k-1|k-1}(X_{k-1}|Z^{(k-1)})$ denote the multitarget prior density of the system dynamic model at time $k-1$. Then, the prediction and update steps of the optimal multitarget Bayes filter recursion are given by [2]

$$\begin{aligned} p_{k|k-1}(X_k|Z^{(k-1)}) &= \int p_{k|k-1}(X_k|X_{k-1}) \\ &\quad \cdot p_{k-1|k-1}(X_{k-1}|Z^{(k-1)}) \\ &\quad \cdot \mu_s(dX_{k-1}) \end{aligned} \tag{2.14}$$

$$p_{k|k}(X_k|Z^{(k)}) = \frac{g_k(Z_k|X_k)p_{k|k-1}(X_k|Z^{(k-1)})}{\int g_k(Z_k|X_k)p_{k|k-1}(X_k|Z^{(k-1)})\mu_s(dX_k)} \tag{2.15}$$

respectively, where $p_{k|k-1}(X_k|X_{k-1})$ is the multitarget dynamic model transition density, $g_k(Z_k|X_k)$ denotes the multitarget likelihood and μ_s takes the place of Lebesque measure. The posterior density $p_{k|k}(X_k|Z^{(k)})$ can be determined using (2.15). With model uncertainty states extraction from (2.15) becomes very difficult. Since multiple model formulation is only used in PHD filters here (not for ML-PDA or JML-PDA filters) the multimodal multitargets tracking formulation only described for the PHD filter implementation.

In order to get a multiple model PHD filter, in [37,49], it is assumed that there are r dynamic models and each target evolves according to one of them at time k. Furthermore, from time k to time $k+1$, the target can switch from model i to model j with probability $\pi(\mathbf{M}_{k+1} = j|\mathbf{M}_k = i)$. Then, a multimodel state space of the form

$$\mathfrak{X} = \biguplus_{i=1}^{\mathsf{r}} \mathfrak{X}_i \tag{2.16}$$

is considered for the target dynamic state evolution. The state vector in $\mathfrak{X}$ is $\mathbf{x}^e$ and, for $\mathbf{x}^e \in \mathfrak{X}_i$, it has the form $[\mathbf{x}^T\ i]^T$. Every target, no matter which dynamic model it follows, evolves in this hybrid space. The state vector for a single target is represented as $[\mathbf{x}^T\ \mathbf{M}]^T$ where $\mathbf{M} \in \{1, \ldots, \mathsf{r}\}$. Here, $\mathbf{x}$ is the kinematic state vector of that target. On the other hand, $\mathbf{M}$ is a variable with integer value and $\mathbf{M} = i$ denotes the random event that the target is evolving according to the i-th dynamic model at time k. It should be noted that the single target state space for different dynamic models may be different, so the dimension of $\mathbf{x}$ depends on the type of dynamic model represented by $\mathbf{M}$. With the above state space model, given the targets following dynamic model i, these targets can only stay in space $\mathfrak{X}_i$ $(i = 1, \cdots, \mathsf{r})$. For a real–valued function $f(\mathbf{x}^e)$ defined on $\mathfrak{X}$, its integration in a subregion $\mathfrak{S} \in \mathfrak{X}$ is given by

$$\int_{\mathfrak{S}} f(\mathbf{x}^e) d\mathbf{x}^e = \sum_{i=1}^{\mathsf{r}} \int_{\mathfrak{X}_i \bigcap \mathfrak{S}} f([\mathbf{x}\ i]^T) d\mathbf{x} \tag{2.17}$$

Correspondingly, for two real–valued functions f and g defined in space $\mathfrak{X}$, their inner product, $\langle f, g\rangle$, becomes

$$\langle f, g\rangle = \sum_{i=1}^{\mathsf{r}} \int_{\mathfrak{X}_i} f([\mathbf{x}\ i]^T) g([\mathbf{x}\ i]^T) d\mathbf{x} \tag{2.18}$$

In [37,49], the state transition for the k-th target is assumed to follow a jump Markov model:

$$f([\mathbf{x}_k, \mathbf{M}_k] | [\mathbf{x}_{k-1}, \mathbf{M}_{k-1}]) = \pi(\mathbf{M}_k | \mathbf{M}_{k-1}) f(\mathbf{x}_k | [\mathbf{x}_{k-1}, \mathbf{M}_{k-1}]) \tag{2.19}$$

In [37,49], except for the above jump Markov dynamic mode transition model, all the assumptions related to the target dynamic system and the measurement model are exactly same as those made in [36,63]. Thus, by substituting (2.17) and (2.18) into the PHD prediction–updating equations in [36,63], the equations for the MM–PHD prediction and update were obtained.

2.3 Introduction to PHD Filter

The Probability Hypothesis Density filter (PHD) is a first-order moment of a multitarget system with the assumption of higher targets signal to noise ratio. The first-order moment is not a vector as in the single target case, but rather it is a density, an information-theoretic best-fit approximation or a generalized fuzzy membership function [36]. The PHD is not a probability density function (PDF). At every instance in time the PHD filter estimates the number of targets directly from the data and it has the tendency to reject clutters. The PHD filter can be easily implemented by sequential Monte Carlo and Gaussian mixture approximation techniques. Note, it does not need a measurement to track association and it discloses unequivocal statistical models for missed detection, sensor field of view and false alarm. The PHD implementation supports major multitarget dynamics i.e., target appearance, target disappearance, spawning of new targets by prior targets.

2.3.1 PHD Filter for non-Maneuvering Targets

Here, we briefly review the PHD filter. The readers are referred to [36,38] for more details.

PHD Filter Initialization

The initialization of the multitarget PHD filter consists of choosing a multitarget prior PHD as

$$D_{0|0}(\mathbf{x}|Z^{(0)}) \quad = \quad N_0 S_0(\mathbf{x}) \tag{2.20}$$

where $S_0(\mathbf{x})$ is a probability density whose peaks correspond to a priori target positions, and N_0 is an initial estimate of the expected number of targets. The prior PHD denoted by $D_{0|0}(\mathbf{x}|Z^{(0)})$ as a density function defined on single target state $\mathbf{x} \in \mathcal{E}_s$ [38], and the expected N_0 is determined from the multitarget prior PHD $D_{0|0}(\cdot)$ as

$$\hat{N}_0 \quad = \quad \int_{\mathcal{E}_s} D_{0|0}(\mathbf{x}|Z^{(0)}) d\mathbf{x} \tag{2.21}$$

PHD Filter Predictor

The PHD filter assumes the standard multitarget motion model of Section 2.1 and that each target evolves and generates observations independently from one another. The motion of existing individual targets can be described by a single target Markov transition density $p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. The probability that a target with the state $\mathbf{x}_{k-1}$ at time step $k-1$ survives to time step k is $P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1})$. Spawning of new targets by existing targets can be described by $\beta_{\mathbf{s},k|k-1}(X_k|\mathbf{x}_{k-1})$, which denotes the likelihood that a group of new targets with state set X_k is spawned at time step k by a single target which state $\mathbf{x}_{k-1}$ at time step $k-1$, and its PHD is $\beta_{\mathbf{s},k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. Appearance of completely new targets is described by $\beta_{\mathbf{nb},k}(X_k)$, which is the likelihood that new targets with state set X_k will enter the scene at time step k, and its PHD is $\beta_{\mathbf{nb},k}(\mathbf{x}_k)$.

Assume that the prior PHD from scan $k-1$ is

$$D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) \quad \approx \quad D_{0|0}(\mathbf{x}|Z^{(0)}) \tag{2.22}$$

Then the predicted PHD equation can be written as

$$\begin{aligned} D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \quad &= \quad D_{\mathbf{c},k|k-1}(\mathbf{x}_k) \\ &\quad\; + D_{\mathbf{s},k|k-1}(\mathbf{x}_k) \\ &\quad\; + D_{\mathbf{nb},k}(\mathbf{x}_k) \end{aligned} \tag{2.23}$$

where the predicted PHD of existing targets is

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k) \quad &= \quad \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) \\ &\quad\; \cdot p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ &\quad\; \cdot D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) \\ &\quad\; \cdot d\mathbf{x}_{k-1} \end{aligned} \tag{2.24}$$

The predicted PHD of spawned targets can be expressed as

$$\begin{aligned} D_{\mathbf{s},k|k-1}(\mathbf{x}_k) \quad &= \quad \int \beta_{\mathbf{s},k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ &\quad\; \cdot D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) \\ &\quad\; \cdot d\mathbf{x}_{k-1} \end{aligned} \tag{2.25}$$

and, the predicted newborn target PHD

$$D_{\mathbf{nb},k}(\mathbf{x}_k) \;=\; \beta_{\mathbf{nb},k}(\mathbf{x}_k) \tag{2.26}$$

depends on the system model. The predicted expected number of targets can be determined as

$$\begin{aligned} \hat{N}_{k|k-1} &= \int D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})d\mathbf{x}_k \\ &= \hat{N}_{\mathbf{c},k|k-1} + \hat{N}_{\mathbf{s},k|k-1} + \hat{N}_{\mathbf{nb},k} \end{aligned} \tag{2.27}$$

and

$$\hat{N}_{\mathbf{c},k|k-1} \;=\; \int D_{\mathbf{c},k|k-1}(\mathbf{x}_k)d\mathbf{x}_k \tag{2.28}$$

$$\hat{N}_{\mathbf{s},k|k-1} \;=\; \int D_{\mathbf{s},k|k-1}(\mathbf{x}_k)d\mathbf{x}_k \tag{2.29}$$

$$\hat{N}_{\mathbf{nb},k} \;=\; \int D_{\mathbf{nb},k}(\mathbf{x}_k)d\mathbf{x}_k \tag{2.30}$$

where (2.28), (2.29) and (2.30) are the predicted expected number of existing targets, spawned targets, and newly-appearing targets, respectively.

PHD Filter Corrector

The predicted PHD can be corrected with the availability of measurement from observation space $Z_k \in Z^{(k)}$ at time step k to get the updated PHD with the assumption that no target generates more than one measurement. Each measurement is generated by no more than a single target and all measurements are conditionally independent of target state. The number of false alarms is Poisson distributed with an average rate of λ_k and the probability density of the spatial distribution of false alarms is $C_k(\mathbf{z}_k)$. The detection probability of a target with state $\mathbf{x}_k$ at time step k is $P_{\mathfrak{d},k}(\mathbf{x}_k)$ and the updated PHD at time step k is

$$\begin{aligned} D_{k|k}(\mathbf{x}_k|Z^{(k)}) &= (1-P_{\mathfrak{d},k}(\mathbf{x}_k))D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \\ &\quad + \sum_{\mathbf{z}_k \in Z_k} \frac{P_{\mathfrak{d},k}(\mathbf{x}_k)p_{k|k}(\mathbf{z}_k|\mathbf{x}_k)D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})}{\lambda_k C_k(\mathbf{z}_k) + \phi_k(\mathbf{z}_k|Z^{(k-1)})} \end{aligned} \tag{2.31}$$

where

$$\begin{aligned} \phi_k(\mathbf{z}_k|Z^{(k-1)}) &= \int P_{\mathfrak{d},k}(\mathbf{x}_k) \\ &\quad \cdot p_{k|k}(\mathbf{z}_k|\mathbf{x}_k) \\ &\quad \cdot D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \\ &\quad \cdot d\mathbf{x}_k \end{aligned} \tag{2.32}$$

and $p_{k|k}(\mathbf{z}_k|\mathbf{x}_k)$ denotes the single sensor-single target likelihood. The updated equation (2.31) is not lossless since approximations are made on the predicted multiple targets PHD to obtain the closed-form solution [38]. Finally, the updated expected number of targets can be determined as

$$\hat{N}_{k|k} = \int D_{k|k}(\mathbf{x}_k|Z^{(k)})d\mathbf{x}_k \tag{2.33}$$

2.3.2 PHD Filter for Maneuvering Targets

As described in [49], a recursive MM-PHD filter algorithm has three main stages. Those three stages are mixing, prediction and update.

MM-PHD mixing

Assume that the model-dependent prior density $D_{k-1|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}|Z^{(k-1)})$ is available at time $k-1$. Then using the total probability theorem the mode-dependent initial density can be defined as $\tilde{D}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)})$ can be determined as [49]

$$\begin{aligned} &\tilde{D}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) = \\ &\quad \sum_{\mathfrak{p}=1}^{\mathsf{r}} D_{k-1|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}|Z^{(k-1)})\pi_{\mathfrak{pq}} \qquad \mathfrak{q} = 1, \ldots, \mathsf{r} \end{aligned} \tag{2.34}$$

where r denotes number of filters and $\pi_{\mathfrak{pq}}$ denotes the Markovian model transition probability matrix. Note, the initial density fed to the PHD filter that is matched to motion model $\mathfrak{q}$. The target can spawn, die, or born and these are only considered in the prediction stage.

MM-PHD prediction

Assume that each target evolves and generates observations independently of one another. A target can continue to survive or disappear from the scene, can be spawned by already-existing targets, and also new targets can be born in the scene independently from the already-existing targets. The mode-dependent predicted density is determined as

$$\begin{aligned} D_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) &= D_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &+ D_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &+ D_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \qquad (2.35) \end{aligned}$$

where the mode-dependent predicted density of existing targets is expressed as follows:

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \\ &\cdot p_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \\ &\cdot \tilde{D}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) \\ &\cdot d\mathbf{x}_{k-1} \qquad (2.36) \end{aligned}$$

where $p_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p})$ denotes a mode-dependent single Markov transition density of the state of existing-targets and $P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$ denotes the mode-dependent survival probability of existing targets that accounts for the event that a target with state $\mathbf{x}_{k-1}$ at time step $k-1$ will survive at time step k. The mode-dependent predicted density of spawned targets can be expressed as

$$\begin{aligned} D_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \int \beta_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \\ &\cdot \tilde{D}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) \\ &\cdot d\mathbf{x}_{k-1} \qquad (2.37) \end{aligned}$$

where $\beta_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p})$ denotes the mode-dependent PHD of the new targets spawned by existing targets. The PHD of the mode-dependent likelihood function is $\beta_{\mathbf{s},k|k-1}(\mathbf{X}_k|\mathbf{M}_k = \mathfrak{q})$, which is the mode-dependent likelihood that a group of new targets with state set X_k will be spawned at time step k by a single target that had state $\mathbf{x}_{k-1}$ at time step $k-1$.

Appearance of completely new targets is also described by $\beta_{\mathbf{nb},k}(X_k, \mathbf{M}_k = \mathfrak{q})$, which is the mode-dependent likelihood that new targets with state set X_k will enter the scene at time step k and its PHD is $\beta_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$. The mode-dependent predicted newborn target density $D_{\mathbf{nb},k}(.)$ depends on the system model. The expected number of targets in the surveillance region can be determined by finding the area of the mode-dependent predicted PHD $D_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)})$.

MM-PHD update

The predicted density can be corrected with the available measurements $Z_k \in Z^{(k)}$ from observation space $\mathcal{E}_o$ at time step k to get the updated density with the assumption that no target generates more than one measurement. Each measurement is generated by no more than a single target and all measurements are conditionally independent of target state. The number of false alarm λ_k is Poisson distributed with spatial density $C_k(\mathbf{z}_k)$ with the assumption of standard multimodal multitarget measurement model from Section 2.2.

At time step k the detection probability of a target with state $\mathbf{x}_k$ is defined as $P_{\mathfrak{d}}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ and the mode-dependent updated PHD at scan k can be determined as (for $\mathsf{q} = 1, \ldots, \mathsf{r}$)

$$\begin{aligned} D_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k)}) \;\; &\cong \;\; (1 - P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}))D_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) \\ &+ \sum_{\mathbf{z}_k \in Z_k} \frac{\phi_k(\mathbf{z}_k|Z^{(k-1)}, \mathbf{M}_k = \mathfrak{q})}{\lambda_k C_k(\mathbf{z}_k) + \int \phi_k(\mathbf{z}_k|Z^{(k-1)}, \mathbf{M}_k = \mathfrak{q}) d\mathbf{x}_k} \end{aligned} \tag{2.38}$$

where the function $\phi_k(.)$ is given as

$$\begin{aligned} \phi_k(\mathbf{z}_k|Z^{(k-1)}, \mathbf{M}_k = \mathfrak{q}) \;\; &= \;\; P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &\cdot p_{k|k}(\mathbf{z}_k|\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &\cdot D_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) \end{aligned} \tag{2.39}$$

By finding the area of the mode-dependent updated PHD $D_{k|k}(.)$ one can determined the average number of targets as

$$\hat{N}_{k|k}^{\mathbf{M}_k = \mathfrak{q}} \;\; = \;\; \int D_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k)}) d\mathbf{x}_k \tag{2.40}$$

and the total number of estimated targets as

$$\hat{N}_{k|k} \;\; = \;\; \sum_{\mathfrak{q}=1}^{\mathsf{r}} \hat{N}_{k|k}^{\mathbf{M}_k = \mathfrak{q}} \tag{2.41}$$

2.4 Introduction to ML-PDA and JML-PDA Algorithms

The ML-PDA algorithm uses low threshold measurement data over a batch of measurement frames and computes track estimates using a sliding window with the assumption of deterministic target motion [2]. The ML-PDA algorithm was first introduced in [27] to estimate the motion parameters of a single target using bearing and frequency measurements over several scans in a high clutter or lower SNR scenario and it was later enhanced by incorporating measurement amplitude as a feature into the ML-PDA likelihood function [31]. The target state is obtained by maximizing the log-likelihood ratio (LLR) formulated from a batch of measurements. It is an effective approach to initialize tracks in high clutter environments, but it is restricted to a single-target tracking.

The standard ML-PDA algorithm supports the initialization/geolocation of multitarget in target tracking problems by the use of its multitarget extension the Joint Maximum-Likelihood Probabilistic Data Association (JML-PDA) [2, 9] algorithm. The JML-PDA algorithm extracted the joint track estimates from the batch of measurements by maximizing the joint log-likelihood ratio (JLLR) followed by the track validations. As the number of targets increase the performance of the JML-PDA algorithm degrades. In practical multitarget tracking application of JML-PDA algorithm for more than three targets may not be possible [9].

2.4.1 ML-PDA Algorithm

The standard ML-PDA algorithm's detail derivation is explained in this section. Note, the derivation supports a general scenario with features like amplitude information and frequency [31]. For further details readers can refer to [2].

At any given reference time r the multidimensional target system state can be defined as

$$\mathbf{x}_r \triangleq [x_r^1, \ldots, x_r^n]' \tag{2.42}$$

where $\mathbf{x}_r$ denotes the target kinematic state and n denotes the number of dimensions. At any time i the target evolves according to

$$\mathbf{x}(i) = f(\mathbf{x}_r, i), \tag{2.43}$$

without the influence of process noise. With the known detection probability P_d a single target is present in each data frame. Across the data frame the detections are independent.

In a single data frame at time i all the measurements can be defined as

$$Z(i) \triangleq \{\mathbf{z}_j(i)\}_{j=1}^{m_i} \tag{2.44}$$

where m_i denotes the measurement number. A true-target originated measurement is present per frame. The measurement vector $\mathbf{z}_j(i)$ from i-th frame and j-th measurement number can be defined as

$$\mathbf{z}_j(i) \triangleq [\beta_{ij} \;\; a_{ij}]' \tag{2.45}$$

where β_{ij} and a_{ij} are denoted the bearing and amplitude measurements, respectively [31].

The cumulative set of measurement during the entire period is given as

$$\mathbf{Z} \triangleq \{Z(i)\}_{i=1}^{N_w} \tag{2.46}$$

where N_w denotes the frame number.

The target originated measurement is defined as

$$\beta_{ij} = h(\mathbf{x}_r, \mathbf{x}_s(i)) + \omega_i, \tag{2.47}$$

where the sensor kinematic state is denoted by $\mathbf{x}_s(i)$ and ω_i is a zero-mean white Gaussian noise. Note that the measurement can also originate from false detection. Across the surveillance region with volume U_β the false alarm u_i is distributed uniformly as a random variable.

Further the ML-PDA implementation is needed some assumptions as described in [27] as follows:

1. Across the frames the number of false detection is independent and Poisson distributed respect to probability mass function $\mu_f(m)$, with spatial density λ.

2. The target and false alarm originated amplitudes are distributed as $p_1(a)$ and

$p_0(a)$, respectively. Note, $p_1(a)$ is affected by the signal to noise ratio (SNR) of the target. Generally, $p_1(a)$ is known or estimated.

3. Measurements conditioned on the target state received at different times are independent.

$$p[Z(i_1), Z(i_2)|\mathbf{x}_r] \quad = \quad p[Z(i_1)|\mathbf{x}_r]p[Z(i_2)|\mathbf{x}_r] \quad \forall \quad i_1 \neq i_2 \tag{2.48}$$

where $i_1, i_2 \in \{1, \ldots, N_w\}$

The assumptions in the ML-PDA algorithm are allowed to express the target-originated measurement's likelihood probability density function as

$$p(\beta_{ij}|\mathbf{x}_r) \quad = \quad \mathcal{N}\left(\beta_{ij}; h\left(\mathbf{x}_r, \mathbf{x}_s(i), i\right), \mathbf{R}\right). \tag{2.49}$$

where $\mathbf{R} = \mathbb{E}[\omega_i \omega_i^{'}]$ denotes the known covariance and sensor kinematic state is denoted by $\mathbf{x}_s(i)$.

The target states can be determined by maximizes the likelihood function, $p(\mathbf{Z}|\mathbf{x}_r)$. Further, an amplitude likelihood ratio can be included in the likelihood function.

The amplitude likelihood ratio is the ratio between the target and false alarm originated amplitudes likelihood functions and it can be described as

$$\rho_{ij} \quad = \quad \frac{p_1(a_{ij}|\xi)}{p_0(a_{ij}|\xi)}, \tag{2.50}$$

where $p_1(a_{ij}|\xi)$ is the amplitude probability density function (PDF) of those originated from the target that defined as

$$p_1(a_{ij}|\xi) \;=\; \frac{a_{ij}}{P_d(1+\mathrm{SNR})}\exp\left(-\frac{a_{ij}^2}{2(1+\mathrm{SNR})}\right), \qquad a_{ij} > \xi \tag{2.51}$$

and $p_0(a_{ij}|\xi)$ denotes the amplitude PDF of the validated measurements originated from false alarm that defined as

$$p_0(a_{ij}|\xi) \;=\; \frac{a_{ij}}{P_{fa}}\exp\left(-\frac{a_{ij}^2}{2}\right), \qquad a_{ij} > \xi \tag{2.52}$$

where a_{ij} and ξ are denoted the amplitude and the detector threshold in each measurement cell, respectively. The amplitude likelihood ratio is condition on $a_{ij} > \xi$. To declare a detection the value of ξ is selected depend on the SNR value and a suitable value of ξ is normally selected [31].

The probabilities P_d and P_{fa} are denoted the probabilities of measurements that originated from true-target and noise-only with conditioned on exceeding the threshold ξ.

For many applications $p_1(a_{ij}|\xi)$ and $p_0(a_{ij}|\xi)$ are considered Rayleigh distribution. For Rayleigh distributed measurements P_d and P_{fa} satisfy

$$P_d \;=\; \exp\left(-\frac{\xi^2}{2(1+\mathrm{SNR})}\right) \tag{2.53}$$

$$P_{fa} \;=\; \exp\left(-\frac{\xi^2}{2}\right) \tag{2.54}$$

For simplicity assume we have a 4-D scenario with target motion model including the frequency information then a 5-dimensional target state vector can be defined as

$$\mathbf{x}_r \triangleq [x_r^1 \; \dot{x}_r^1 \; x_r^2 \; \dot{x}_r^2 \; \gamma]' \tag{2.55}$$

where a constant unknown emitted frequency is denoted by γ, (x_r^1, x_r^2) are positions of the target and $\dot{x}_r^1$, $\dot{x}_r^2$ are velocities of the target.

Assume the state of the sensor-platform is at time i

$$\mathbf{x}_s(i) \triangleq [\xi_s(i) \; \dot{\xi}_s(i) \; \eta_s(i) \; \dot{\eta}_s(i)]' \tag{2.56}$$

where $\xi_s(i)$, $\eta_s(i)$ are positions of the sensor and $\dot{\xi}_s(i)$, $\dot{\eta}_s(i)$ are velocities of the sensor.

The noise free Doppler shifted frequency can be defined as

$$\gamma_i(\mathbf{x}_r) = \gamma \left[1 - \frac{v_\xi(i, \mathbf{x}_r) \sin \theta_i(\mathbf{x}_r) + v_\eta(i, \mathbf{x}_r) \cos \theta_i(\mathbf{x}_r)}{\mathfrak{c}}\right] \tag{2.57}$$

where $\gamma_i(\mathbf{x}_r)$ denotes the Doppler shifted frequency at the sensor at time i. The target and sensor-platform relative motions affect the $\gamma_i(\mathbf{x}_r)$. The relative velocity components are denoted as $v_\xi(i, \mathbf{x}_r)$, $v_\eta(i, \mathbf{x}_r)$. The velocity of sound in the medium is $\mathfrak{c}$. The angle $\theta_i(\mathbf{x}_r)$ is defined as

$$\theta_i(\mathbf{x}_r) = \arctan\left(\frac{\eta_s(i) - x_r^2}{\xi_s(i) - x_r^1}\right) \tag{2.58}$$

Using (2.55) and (2.56) the relative velocity components can be defined as

$$v_\xi(i, \mathbf{x}_r) = \dot{\xi}_s(i) - \dot{x}_r^1, \tag{2.59}$$

and

$$v_{\eta}(i, \mathbf{x}_r) \;\; = \;\; \dot{\eta}_s(i) - \dot{x}_r^2 \tag{2.60}$$

The noisy frequency measurement can be defined as

$$\psi_{ij} \;\; = \;\; \gamma_i(\mathbf{x}_r) + \nu_i, \tag{2.61}$$

where at time i, ψ_{ij} denote the frequency-generated measurement, $\nu_i \sim N(0, \sigma_{\gamma}^2)$

The measurement noise components ω_i and ν_i are conditionally independent. Noise-originated frequency measurements are assumed to be uniformly distributed in the entire surveillance region with volume U_{γ}.

The measurement vector $\mathbf{z}_j(i)$ from i-th frame and j-th measurement number can be defined to support the narrowband sonar case as

$$\mathbf{z}_j(i) \;\; \triangleq \;\; [\beta_{ij} \;\; a_{ij} \;\; \psi_{ij}]' \tag{2.62}$$

The bandwidth of the sonar is normally known $[\Omega_1, \Omega_2]$ and the measurements can lie anywhere within the range. If the measurement is only azimuth then the sonar operator can select a frequency subregion $[\Gamma_1, \Gamma_2]$. The frequency region can be defined as

$$U_{\gamma} \;\; = \;\; [\Gamma_1, \Gamma_2] \in [\Omega_1, \Omega_2] \tag{2.63}$$

A same method can be applied to find the frequency information for a 6-D scenario.

At time i using the total probability theorem as used in Probability Data Association (PDA) approach [2] the likelihood function $p\left(Z(i)|\mathbf{x}_r\right)$ can be determined as

$$\begin{aligned} p\left(Z(i)|\mathbf{x}_r\right) &= (1-P_d)\prod_{j=1}^{m_i} p(\mathbf{z}_j(i)|\text{“clutter”}) \\ &+ \frac{P_d}{m_i}\sum_{j=1}^{m_i} p(\mathbf{z}_j(i)|\mathbf{x}_r)\prod_{l\neq j} p(\mathbf{z}_l(i)|\text{“clutter”}) \\ &= (1-P_d)\prod_{j=1}^{m_i} p(\beta_{ij}, a_{ij}, \psi_{ij}|\text{“clutter”}) \\ &+ \frac{P_d}{m_i}\sum_{j=1}^{m_i} p(\beta_{ij}, a_{ij}, \psi_{ij}|\mathbf{x}_r)\prod_{l\neq j} p(\beta_{il}, a_{il}, \psi_{il}|\text{“clutter”}) \\ &= \frac{(1-P_d)}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} p_0(a_{ij}|\xi) \\ &+ \frac{P_d\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} p_0(a_{ij}|\xi)\sum_{j=1}^{m_i} p(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\rho_{ij} \end{aligned} \tag{2.64}$$

where $U = U_\beta U_\gamma$ and (2.64) denotes the weighted sum of all the likelihood functions corresponding to a certain measurement or no measurement respect to a true-target where rest of the measurements originated from the false alarm.

The observations are conditionally independent across the frame and the likelihood function of the entire set of measurements can be written in terms of individual likelihood functions [2] for a N_w frames as

$$p(\mathbf{Z}|\mathbf{x}_r) = \prod_{i=1}^{N_w} p\left(Z(i)|\mathbf{x}_r\right) \tag{2.65}$$

Note, the likelihood function conditioned on all measurements are false detection is defined as

$$\prod_{i=1}^{N_w}\left[\frac{1}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i}p_0(a_{ij}|\xi)\right]. \tag{2.66}$$

Dividing (2.65) by (2.66) and taking the logarithm of the resulting function, the log-likelihood ratio (LLR) can be determined as

$$\Lambda\left(\mathbf{Z}|\mathbf{x}_r\right) = \sum_{i=1}^{N_w}\ln\left[(1-P_d)+\frac{P_d}{\lambda}\sum_{j=1}^{m_i}\rho_{ij}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r)\right] \tag{2.67}$$

The target estimates, $\hat{\mathbf{x}}_r$ can be determined as

$$\hat{\mathbf{x}}_r = \arg\max_{\mathbf{x}_r}\Lambda\left(\mathbf{Z}|\mathbf{x}_r\right), \tag{2.68}$$

The ML-PDA algorithm's track estimates have to satisfy the track validations process due to the outcome may come from no target or false alarm [2].

2.4.2 JML-PDA Algorithm

The JML-PDA algorithm details of derivation is explained in this section. The derivations are based on a 5D dimensional narrowband sonar tracking. The kinematic measurements are angle-only and the amplitude information is also available.

The multidimensional multitarget system state can be expressed at reference time r as

$$\mathbf{x}_r \triangleq [\mathbf{x}_r^1,\ldots,\mathbf{x}_r^n]' \tag{2.69}$$

where $\mathbf{x}_r$ denotes the multitarget state, n denotes the number of targets.

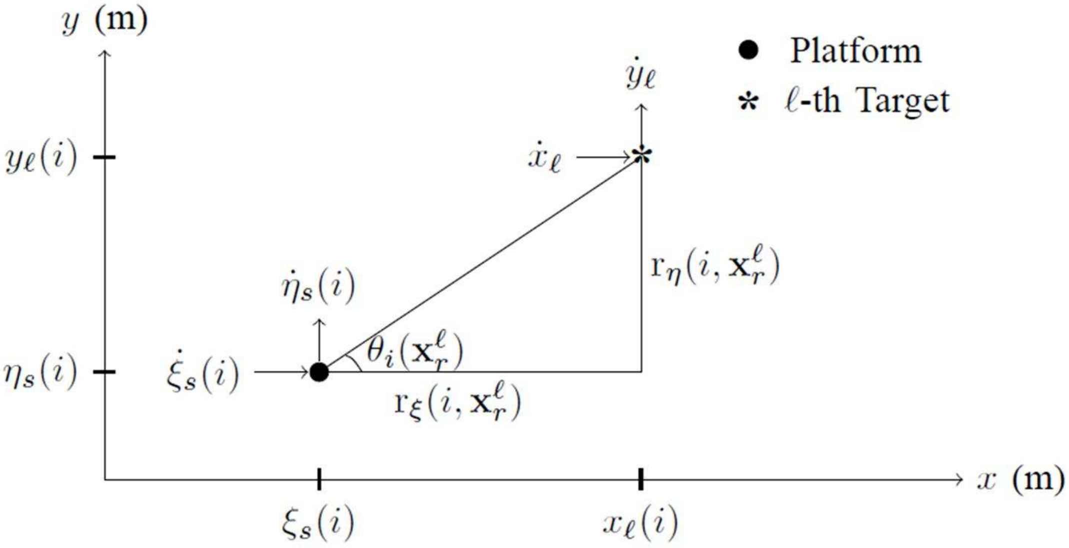

Figure 2.1: Motion of the platform and the ℓ-th target at time i

As shown in Figure (2.1) at time i a sensor on a platform moves along X-Y plane and $\mathbf{x}_r^\ell$ denotes the ℓ-th target state,

$$\mathbf{x}_r^\ell \triangleq [x_\ell(i) \;\; \dot{x}_\ell \;\; y_\ell(i) \;\; \dot{y}_\ell \;\; \gamma_\ell]' \tag{2.70}$$

where the length of the vector $\mathbf{x}_r^\ell$ is 5. The ℓ-th target's constant unknown emitted frequency is denoted by γ_ℓ, $(x_\ell(i), y_\ell(i))$ are positions of the ℓ-th target and the corresponding constant velocities in each direction are denoted by $\dot{x}_\ell$, $\dot{y}_\ell$.

The ℓ-th target evolves at any time i according to

$$\mathbf{x}_r^\ell(i) = f_\ell(\mathbf{x}_r^\ell, i) \tag{2.71}$$

where the process noise is ignored. Generally in JML-PDA algorithm the number of existing targets are known and assumed to be alive during the reconnaissance mission. Each target is detected with the probability P_d^{ℓ} and all the targets are present in each data frame. Across the data frame the detections are independent.

The observation at time i in a single data frame can be defined as

$$Z(i) \triangleq \{\mathbf{z}_j(i)\}_{j=1}^{m_i} \tag{2.72}$$

where the total measurment/observation is denoted by m_i. All the true-target originated measurements are present per frame. The j-th measurement number from the i-th frame can be expressed as

$$\mathbf{z}_j(i) \triangleq [\beta_{ij} \;\; \psi_{ij} \;\; a_{ij}]' \tag{2.73}$$

where β_{ij}, ψ_{ij} and a_{ij} are denoted the bearing, frequency and amplitude measurements, respectively [31]. As stated in [31] the envelope output of the detector is referred to amplitude.

The cumulative set of measurement during the entire period is given as

$$\mathbf{Z} \triangleq \{Z(i)\}_{i=1}^{N_w} \tag{2.74}$$

where N_w denotes the frame number.

Note the state of the sensor-platform at time i given as

$$\mathbf{x}_s(i) \triangleq [\xi_s(i) \; \dot{\xi}_s(i) \; \eta_s(i) \; \dot{\eta}_s(i)]' \tag{2.75}$$

where the sensor kinematic state is denoted by $\mathbf{x}_s(i)$, $(\xi_s(i), \eta_s(i))$ are positions of the sensor and $\dot{\xi}_s(i)$, $\dot{\eta}_s(i)$ are velocities of the sensor.

The ℓ-th target originated measurement at time/frame i is defined as

$$\beta_{ij}^{\ell} = h_{\ell}(\mathbf{x}_r^{\ell}, \mathbf{x}_s(i)) + \omega_i^{\ell}, \tag{2.76}$$

where $\beta_{ij} = \{\beta_{ij}^1, \ldots, \beta_{ij}^n\}$, ω_i^{ℓ} is a zero-mean white Gaussian noise. Note that the measurement can also originate from false detection. Across the surveillance region with volume U_β the false alarm u_i is distributed uniformly as a random variable.

The JML-PDA derivations use some the following assumptions [27]:

1. Across the frames the number of false detection is independent and Poisson distributed respect to probability mass function $\mu_f(m)$, with spatial density λ.

2. The multitarget and false alarm originated amplitudes are distributed as $p_1^{\ell}(a)$ (where $\ell = \{1, \ldots, n\}$) and $p_0(a)$, respectively. Note, $p_1^{\ell}(a)$ is affected by the signal to noise ratio (SNR) of the ℓ-th target. Generally, $p_1^{\ell}(a)$ is known or estimated.

3. Measurements conditioned on the ℓ-th target state received at different times are independent.

$$p[Z(i_1), Z(i_2)|\mathbf{x}_r^{\ell}] = p[Z(i_1)|\mathbf{x}_r^{\ell}]p[Z(i_2)|\mathbf{x}_r^{\ell}] \quad \forall \quad i_1 \neq i_2 \tag{2.77}$$

   where $\ell = \{1, \ldots, n\}$, $i_1, i_2 \in \{1, \ldots, N_w\}$

4. Total n number of previously confirmed targets exist.

5. Each target correspond to a certain measurement in each frame.

6. A certain measurement can not be associated to more than one target.
7. Target-originated measurement errors have the same distribution for each target.

The ℓ-th target-originated measurement's likelihood probability density function can be determined as

$$p(\beta_{ij}^{\ell}|\mathbf{x}_r^{\ell}) \;=\; \mathcal{N}\left(\beta_{ij}^{\ell}; h_{\ell}\left(\mathbf{x}_r^{\ell}, \mathbf{x}_s(i), i\right), \mathbf{R}\right). \tag{2.78}$$

where $\mathbf{R} = \mathbb{E}[\omega_i \omega_i^{'}]$ denotes the known covariance and sensor kinematic state is denoted by $\mathbf{x}_s(i)$.

The noisy ℓ-th target generated frequency measurement can be defined as

$$\psi_{ij}^{\ell} \;=\; \gamma_i(\mathbf{x}_r^{\ell}) + \nu_i^{\ell}, \tag{2.79}$$

where $\gamma_i(\mathbf{x}_r^{\ell})$ denotes the noise free Doppler shifted frequency, $\psi_{ij} = \{\psi_{ij}^{1}, \ldots, \psi_{ij}^{n}\}$ and $\nu_i^{\ell} \sim N(0, (\sigma_{\gamma}^{\ell})^2)$

The noise free ℓ-th target-originated Doppler shifted frequency can be defined as

$$\gamma_i(\mathbf{x}_r^{\ell}) \;=\; \gamma_{\ell}\left[1 - \frac{v_{\xi}(i, \mathbf{x}_r^{\ell}) \sin\theta_i(\mathbf{x}_r^{\ell}) + v_{\eta}(i, \mathbf{x}_r^{\ell}) \cos\theta_i(\mathbf{x}_r^{\ell})}{\mathfrak{c}}\right] \tag{2.80}$$

where the relative velocity components are denoted as $v_{\xi}(i, \mathbf{x}_r^{\ell})$, $v_{\eta}(i, \mathbf{x}_r^{\ell})$. The velocity of sound in the medium is $\mathfrak{c}$. The angle $\theta_i(\mathbf{x}_r^{\ell})$ is defined as

$$\begin{aligned} \theta_i(\mathbf{x}_r^{\ell}) \;&=\; \arctan\left(\frac{\mathrm{r}_{\eta}(i, \mathbf{x}_r^{\ell})}{\mathrm{r}_{\xi}(i, \mathbf{x}_r^{\ell})}\right) \\ &= h_{\ell}(\mathbf{x}_r^{\ell}, \mathbf{x}_s(i)) \end{aligned} \tag{2.81}$$

The target and sensor-platform relative motions affect the $\gamma_i(\mathbf{x}_r^\ell)$.

The measurement noise components ω_i^ℓ and ν_i^ℓ are conditionally independent. Noise-originated frequency measurements are assumed to be uniformly distributed in the entire surveillance region with volume U_γ.

The bandwidth of the sonar is normally known $[\Omega_1, \Omega_2]$ and the measurements can lie anywhere within the range. If the measurement is only azimuth then the sonar operator can select a frequency subregion $[\Gamma_1, \Gamma_2]$. The frequency region can be defined as

$$U_\gamma = [\Gamma_1, \Gamma_2] \in [\Omega_1, \Omega_2] \tag{2.82}$$

The SNR of each target can be different and therefore, the measurement amplitude likelihood ratio must be defined for each target. The amplitude likelihood ratio is the ratio between the target-originated amplitudes likelihood function and the false alarm originated amplitudes likelihood function.

The amplitude likelihood ratio for the ℓ-th target can be defined as

$$\rho_{ij}^\ell = \frac{p_1(a_{ij}^\ell|\xi, H_\ell)}{p_0(a_{ij}|\xi)}, \tag{2.83}$$

where $p_1(a_{ij}^\ell|\xi, H_\ell)$ is the amplitude probability density function (PDF) of those originated from the target, H_ℓ denotes the ℓ-th hypothesis, $p_0(a_{ij}|\xi)$ denotes the amplitude PDF of the validated measurements originated from false alarm, a_{ij} and ξ denote the amplitude and the detector threshold in each measurement cell, respectively. The amplitude likelihood ratio is conditioned on $a_{ij} > \xi$. A value of ξ is selected to declare a detection and ξ depends on SNR values [31]. The probabilities P_d^ℓ and P_{fa} are denoted the probabilities of measurements that originated from ℓ-th ($\ell = \{1, \ldots, n\}$) targets and noise with conditioned on exceeding the threshold ξ. For many applications $p_1^{\ell}(\cdot)$ and $p_0(\cdot)$ are considered Rayleigh distribution.

For simplicity assume $n = 3$ targets are present in a surveillance area with high false alarm. Using the measurements from a single frame of data the joint likelihood function can be determined by taking the weighted sum of eight events. The following refer to the possible target detection events and are expressed as

$$\begin{aligned}
L_i^0 &= p\left(Z(i)|\mathbf{x}_r, \text{no target detections}\right) && (2.84)\\
L_i^1 &= p\left(Z(i)|\mathbf{x}_r, \text{only target 1 is detected}\right) && (2.85)\\
L_i^2 &= p\left(Z(i)|\mathbf{x}_r, \text{only target 2 is detected}\right) && (2.86)\\
L_i^3 &= p\left(Z(i)|\mathbf{x}_r, \text{only target 3 is detected}\right) && (2.87)\\
L_i^{12} &= p\left(Z(i)|\mathbf{x}_r, \text{only targets 1,2 are detected}\right) && (2.88)\\
L_i^{13} &= p\left(Z(i)|\mathbf{x}_r, \text{only targets 1,3 are detected}\right) && (2.89)\\
L_i^{23} &= p\left(Z(i)|\mathbf{x}_r, \text{only targets 2,3 are detected}\right) && (2.90)\\
L_i^{123} &= p\left(Z(i)|\mathbf{x}_r, \text{only targets 1,2,3 are detected}\right) && (2.91)
\end{aligned}$$

Then the multitarget likelihood $p(Z(i)|\mathbf{x}_r)$ is determined as

$$\begin{aligned} p(Z(i)|\mathbf{x}_r) \;=\; & (1-P_d^1)(1-P_d^2)(1-P_d^3)L_i^0 \\ & + P_d^1(1-P_d^2)(1-P_d^3)L_i^1 \\ & + (1-P_d^1)P_d^2(1-P_d^3)L_i^2 \\ & + (1-P_d^1)(1-P_d^2)P_d^3L_i^3 \\ & + P_d^1P_d^2(1-P_d^3)L_i^{12} \\ & + P_d^1(1-P_d^2)P_d^3L_i^{13} \\ & + (1-P_d^1)P_d^2P_d^3L_i^{23} \\ & + P_d^1P_d^2P_d^3L_i^{123} \end{aligned} \tag{2.92}$$

where P_d^ℓ is the single frame detection probability for the ℓ-th target.

The possible target detections terms, L_i^ℓ are determined by associating a certain observation to each detected target and the rest of the observations are considered as false detection. The possible target detection terms are determined as

$$L_i^0 \;=\; \frac{\mu_f(m_i)}{U^{m_i}}\prod_{j=1}^{m_i} p_0(a_{ij}|\tau) \tag{2.93}$$

$$L_i^1 \;=\; \frac{\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} p_0(a_{ij}|\tau)\sum_{j=1}^{m_i} p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1)p_{ij}^1 \tag{2.94}$$

$$L_i^2 \;=\; \frac{\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} p_0(a_{ij}|\tau)\sum_{j=1}^{m_i} p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^2)p_{ij}^2 \tag{2.95}$$

$$L_i^3 = \frac{\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i}p_0(a_{ij}|\tau)\sum_{j=1}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^3)p_{ij}^3 \tag{2.96}$$

$$\begin{aligned} L_i^{12} = {} & \frac{\mu_f(m_i-2)}{U^{m_i-2}m_i(m_i-1)}\prod_{j=1}^{m_i}p_0(a_{ij}|\tau)\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1) \\ & \cdot p(\beta_{il},\psi_{il}|\mathbf{x}_r^2)p_{ij}^1p_{il}^2 \end{aligned} \tag{2.97}$$

$$\begin{aligned} L_i^{13} = {} & \frac{\mu_f(m_i-2)}{U^{m_i-2}m_i(m_i-1)}\prod_{j=1}^{m_i}p_0(a_{ij}|\tau)\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1) \\ & \cdot p(\beta_{il},\psi_{il}|\mathbf{x}_r^3)p_{ij}^1p_{il}^3 \end{aligned} \tag{2.98}$$

$$\begin{aligned} L_i^{23} = {} & \frac{\mu_f(m_i-2)}{U^{m_i-2}m_i(m_i-1)}\prod_{j=1}^{m_i}p_0(a_{ij}|\tau)\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^2) \\ & \cdot p(\beta_{il},\psi_{il}|\mathbf{x}_r^3)p_{ij}^2p_{il}^3 \end{aligned} \tag{2.99}$$

$$\begin{aligned} L_i^{123} = {} & \frac{\mu_f(m_i-3)}{U^{m_i-3}m_i(m_i-1)(m_i-2)}\prod_{j=1}^{m_i}p_0(a_{ij}|\tau)\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}\sum_{\substack{q=1\\q\neq j\\q\neq l}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1) \\ & \cdot p(\beta_{il},\psi_{il}|\mathbf{x}_r^2)p(\beta_{iq},\psi_{iq}|\mathbf{x}_r^3)p_{ij}^1p_{il}^2p_{iq}^3 \end{aligned} \tag{2.100}$$

where where $U = U_\beta U_\gamma$. The joint likelihood function of N_w frames can be determined as

$$p(\mathbf{Z}|\mathbf{x}_r) = \prod_{i=1}^{N_w}p(Z(i)|\mathbf{x}_r) \tag{2.101}$$

and the likelihood of all noise originated measurements is given as

$$\prod_{i=1}^{N_w}(1-P_d^1)(1-P_d^2)(1-P_d^3)L_i^0 \tag{2.102}$$

Then the joint log-likelihood ratio (JLLR) can be determined by dividing (2.101) by (2.102) and by taking the logarithmic of the result as

$$\begin{aligned}
\Psi(\mathbf{Z}|\mathbf{x}_r) \quad = \quad & \sum_{i=1}^{N_w}\ln[1+\frac{P_d^1}{\lambda(1-P_d^1)}\sum_{j=1}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1)p_{ij}^1 \\
& + \frac{P_d^2}{\lambda(1-P_d^2)}\sum_{j=1}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^2)p_{ij}^2 \\
& + \frac{P_d^3}{\lambda(1-P_d^3)}\sum_{j=1}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^3)p_{ij}^3 \\
& + \frac{P_d^1P_d^2}{\lambda^2(1-P_d^1)(1-P_d^2)}\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1)p(\beta_{il},\psi_{il}|\mathbf{x}_r^2)p_{ij}^1p_{il}^2 \\
& + \frac{P_d^1P_d^3}{\lambda^2(1-P_d^1)(1-P_d^3)}\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1)p(\beta_{il},\psi_{il}|\mathbf{x}_r^3)p_{ij}^1p_{il}^3 \\
& + \frac{P_d^2P_d^3}{\lambda^2(1-P_d^2)(1-P_d^3)}\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^2)p(\beta_{il},\psi_{il}|\mathbf{x}_r^3)p_{ij}^2p_{il}^3 \\
& + \frac{P_d^1P_d^2P_d^3}{\lambda^3(1-P_d^1)(1-P_d^2)(1-P_d^3)}\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}\sum_{\substack{q=1\\q\neq j\\q\neq l}}^{m_i}p(\beta_{ij},\psi_{ij}|\mathbf{x}_r^1) \\
& \cdot p(\beta_{il},\psi_{il}|\mathbf{x}_r^2)p(\beta_{iq},\psi_{iq}|\mathbf{x}_r^3)p_{ij}^1p_{il}^2p_{iq}^3]
\end{aligned} \tag{2.103}$$

where the expected number of false alarm per unit volume is denoted by λ.

The parameter estimate of $\hat{\mathbf{x}}_r$ can be determined as

$$\hat{\mathbf{x}}_r = \arg\max_{\mathbf{x}_r} \Psi\left(\mathbf{Z}|\mathbf{x}_r\right), \tag{2.104}$$

and (2.104) gives the track estimates of three targets in this case. The track validation must be done to determine whether targets originated from false alarm or true-target [2]. Note, this method can be applied to any number of targets with false alarm.

Chapter 3

B-splines

3.1 Introduction

In this section, a brief background on B-spline is provided and for further details readers can refer to [17,53]. Any arbitrary geometrical, numerical or statistical curve/surface/volume/hypersurface can be described by the B-spline representation [53]. The B-spline representation can be taken by two available methods: the Spline Interpolation Method (SIM) and the Spline Approximation Method (SAP) [12]. The SAP and the SIM can give a better representation for any given problem with any dimensions, but both methods are affected by the curse of dimensionality. The curse of dimensionality mostly influenced by the parameters of the problem and it is a challenging problem to solve with existing computer algorithms. The following subsections describe the theories and properties on B-spline for any dimension.

3.2 Univariate B-spline

Using B-Splines, any one-dimensional curve $\mathbf{C}(x)$ can be represented by

$$\mathbf{C}(x) = \sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x). \tag{3.1}$$

where $\mathbb{P}_i$ denotes the i-th control point and n_s denotes the total number of control points. The $B_{i,p,\mathbf{t}}(x)$ denotes the B-basis function of a certain variable x (e.g., multi-target state), which are piecewise polynomial functions forming a basis for the vector space of all piecewise polynomial functions of the desired degree and continuity. The order of the curve is denoted by p and the degree of the curve is $(p-1)$. Note that continuity is determined by the basis functions, hence the control points can be modified without altering the curve's continuity. The knot vector denoted by $\mathbf{t}$ is a non-decreasing sequence of real numbers, where $\mathbf{t} = \{t_1, \dots, t_\tau\}$, i.e., $t_i \leq t_{i+1}$, $i = 1, \dots, \tau$. The total number of knots is denoted by τ. Another important characteristic of the B-basis function is that of local support; this implies that each $B_{i,p,\mathbf{t}}(x)$ is non-zero only on a limited number of subintervals, not the entire domain, $[t_1, t_\tau]$. Since $\mathbb{P}_i$ is multiplied by $B_{i,p,\mathbf{t}}(x)$, moving $\mathbb{P}_i$ affects the curve only on the subintervals where $B_{i,p,\mathbf{t}}(x)$ is non-zero.

There are a number of ways to define the B-spline basis functions, e.g., divided-differences of truncated power functions, blossoming, or recurrence formula [12]. Generally, the recurrence formula is used since it is most suitable for computer implementation [53]. The i-th B-basis function of order p (or of degree $p-1$), is defined as

$$B_{i,1}(x) = \begin{cases} 1 \quad \text{if } \ t_i \leqslant x < t_{i+1}, \\ \\ 0 \qquad \text{otherwise.} \end{cases} \tag{3.2}$$

$$B_{i,p}(x) \ = \ \frac{(x - t_i)B_{i,p-1}(x)}{t_{i+p-1} - t_i} + \frac{(t_{i+p} - x)B_{i-1,p-1}(x)}{t_{i+p} - t_{i+1}} \tag{3.3}$$

The Control Polygon

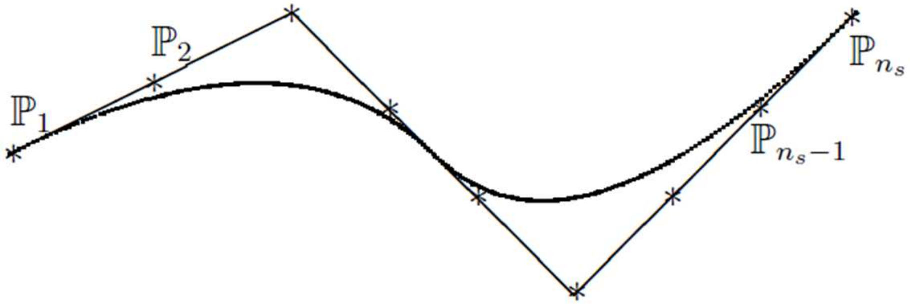

Figure 3.1: A B-spline curve for a given curve.

Figure (3.1) illustrates a B-spline curve for a given curve. The control points $\mathbb{P}_i$ in the B-spline curve define the vertices. The control polygon of a B-spline curve is the polygonal arc and also in here the control polygon is a piecewise linear B-spline curve. The polygonal arc formed by its control points, $\mathbb{P}_1, \mathbb{P}_2, \ldots, \mathbb{P}_{n_s}$. The distance between the control polygon and the curve increases as the order of the given curve increases. A higher order B-spline curve tends to smooth the control polygon and at the same time mimic its shape. As shown in Figure (3.2) the B-spline curve can be redefined by inserting knots into the curve to make the control polygons get closer to the given curve. If the refinement is infinite then the control polygon converges to the curve.

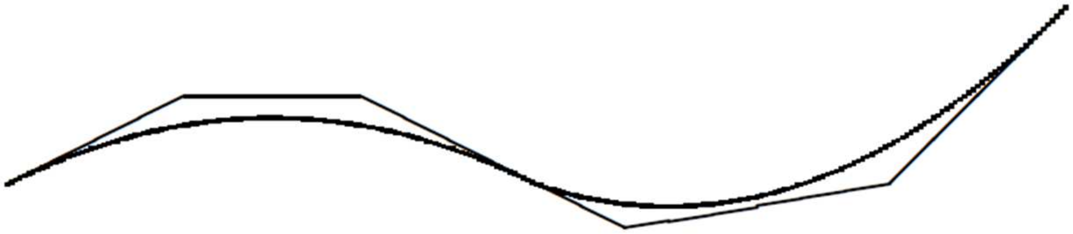

Figure 3.2: The B-spline curve with a redefined knot vector.

The Knot Vector

Moving a control point on a B-spline curve usually cause an effect on the curve. This effect can influence on a range of the curve (locally) or the whole curve (globally). Using the B-spline a curve locally controllable. The range of the curve is effected by the control point movement can be divided and they refer to as knots. The knots of a B-spline curve describe the following properties of the curve: the parametrization of the B-spline curve and the continuity at the joins between the adjacent polynomial segments of the B-spline curve. The chord-length parametrization is used for interpolation. The degree of continuity is determined by the number of equal knots. The curve is discontinuous if $n_{\mathbf{t}^u}$ consecutive internal knots are equal. Likewise if $n_{\mathbf{t}^u} - 1$ consecutive internal knots are equal, then the curve is continuous but not in general differentiable. As shown in Figure (3.3) a continuously differentiable curve with a discontinuity in the second derivative can be modeled using $n_{\mathbf{t}^u} - 2$ equal knots. For intersection algorithms, curves are usually expected to be continuously differentiable (C^1) [53].

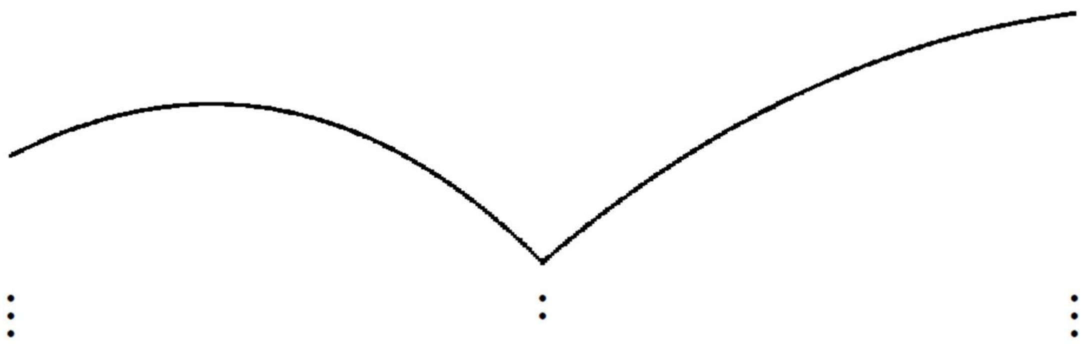

Figure 3.3: A quadratic B-spline curve with two equal internal knots.

3.3 Bivariate B-spline

A bivariate B-spline is defined as

$$\mathbf{S}(u,v) \;=\; \sum_{i=1}^{n_{s,1}} \sum_{j=1}^{n_{s,2}} \mathbb{P}_{i,j} B_{i,p_1,\mathbf{t}^u}(u) B_{j,p_2,\mathbf{t}^v}(v) \qquad (3.4)$$

where u and v denote two variables (or parameters) and $\mathbb{P}_{i,j}$ denotes the control points. The basis function in (3.4) represents a B-spline surface. The B-spline surface is a product of two B-basis functions of B-spline curves. The data of the bivariate B-spline must fulfill the following requirements: both knot vectors must be non-decreasing, the number of vertices must be greater than or equal to the order with respect to both parameters: $n_{s,1} \geq p_1$ and $n_{s,2} \geq p_2$ and there should be p_1 equal knots at the beginning and end of knot vector $\mathbf{t}^u$ and p_2 equal knots at the beginning and end of knot vector $\mathbf{t}^v$. The properties of the representation of a B-spline surface are similar to the properties of the representation of a B-spline curve. As shown in Figure (3.4) the control points $\mathbb{P}_{i,j}$ form a control net. The control net has similar properties to the control polygon of a B-spline curve, described in section 3.2. A B-spline surface has two knot vectors, one for each parameter. Figure (3.4) shows that *isocurves*, surface curves defined by fixing the value of one of the parameters. The surface is drawn using isocurves and the dimension is 3.

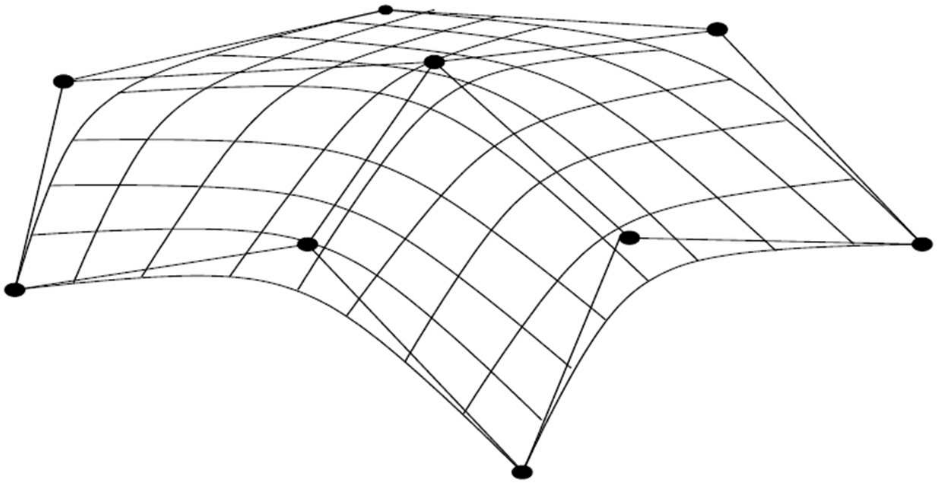

Figure 3.4: A B-spline surface and its control net.

A basis function of a B-spline surface is the product of two basis functions of two B-spline curves, $B_{i,p_1,\mathbf{t}^u}(u)B_{j,p_2,\mathbf{t}^v}(v)$. The B-basis support is the rectangle $[t_i^u, t_{i+p_1}^u] \times [t_j^v, t_{j+p_2}^v]$. If the basis functions in both directions are of degree one and all knots have multiplicity one, then the surface basis functions are pyramid-shaped [53]. For higher degrees, the surface basis functions are bell shaped.

3.4 Trivariate B-spline

The tensor-product B-Spline of three variables is called a trivariate B-spline (B-spline volume) and has the form

$$\mathbf{T}(u,v,w) = \sum_{i=1}^{n_{s,1}}\sum_{j=1}^{n_{s,2}}\sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{i,p_1,\mathbf{t}^u}(u) B_{j,p_2,\mathbf{t}^v}(v) B_{l,p_3,\mathbf{t}^w}(w) \tag{3.5}$$

with control points $\mathbb{P}_{i,j,l}$ and three variables (or parameters) u, v and w. The equation (3.5) shows that the B-basis function of a trivariate B-spline is a product of three basis functions of B-spline curves (B-splines). This is why a Trivariate B-spline is called a tensor-product trivariate/volume. The number of vertices with respect to the first, second and third parameters are denoted by $n_{s,1}$, $n_{s,2}$ and $n_{s,3}$. Can consider a same order for all the parameters and the order of the B-splines in here for the first, second and third parameters are denoted by p_1, p_2 and p_3.

The knot vector of the B-splines with respect first, second and third parameters are defined as $\mathbf{t}^u = (t_1^u, t_2^u, \ldots, t_{n_{s,1}+p_1}^u)$, $\mathbf{t}^v = (t_1^v, t_2^v, \ldots, t_{n_{s,2}+p_2}^v)$ and $\mathbf{t}^w = (t_1^w, t_2^w, \ldots, t_{n_{s,3}+p_3}^w)$. The control points of the B-spline volume, $\mathbb{P}_{d,i,j,l}$, where $d = 1, \ldots, \wp$, $i = 1, \ldots, n_{s,1}$, $j = 1, \ldots, n_{s,2}$, $l = 1, \ldots, n_{s,3}$. When the dimension of the underlying Euclidean space is $\wp = 3$, $\mathbb{P} = (x_{1,1,1}, y_{1,1,1}, z_{1,1,1}, \ldots, x_{n_{s,1},1,1}, y_{n_{s,1},1,1}, z_{n_{s,1},1,1}, \ldots, x_{n_{s,1},n_{s,2},1}, y_{n_{s,1},n_{s,2},1}, z_{n_{s,1},n_{s,2},1}, \ldots$ and $x_{n_{s,1},n_{s,2},n_{s,3}}, y_{n_{s,1},n_{s,2},n_{s,3}}, z_{n_{s,1},n_{s,2},n_{s,3}})$.

The data of the B-spline volume must fulfill the following requirements: three knot vectors must be non-decreasing, the number of vertices must be greater than or equal to the order with respect to both parameters: $n_{s,1} \geq p_1$, $n_{s,2} \geq p_2$ and $n_{s,3} \geq p_3$ and there should be p_1 equal knots at the beginning and end of knot vector $\mathbf{t}^u$, p_2 equal knots at the beginning and end of knot vector $\mathbf{t}^v$ and p_3 equal knots at the beginning and end of knot vector $\mathbf{t}^w$.

Evaluation and Rendering of Trivariate B-spline

Suppose it is necessary to evaluate $\mathbf{T}$ at $(\hat{u}, \hat{v}, \hat{w})$ as,

$$\mathbf{T}(\hat{u}, \hat{v}, \hat{w}) = \sum_{i=1}^{n_{s,1}} \sum_{j=1}^{n_{s,2}} \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(\hat{w}) B_{j,p_2,\mathbf{t}^v}(\hat{v}) B_{i,p_1,\mathbf{t}^u}(\hat{u}) \tag{3.6}$$

First consider evaluating the isoparametric surface with a fixed of w. For $i = 1, \ldots, n_{s,1}$, $j = 1, \ldots, n_{s,2}$, let

$$\gamma_{i,j} = \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(\hat{w}) \tag{3.7}$$

Then

$$\mathbf{T}(u, v, \hat{w}) = \sum_{i=1}^{n_{s,1}} \sum_{j=1}^{n_{s,2}} \gamma_{i,j} B_{j,p_2,\mathbf{t}^v}(\hat{v}) B_{i,p_1,\mathbf{t}^u}(\hat{u}) \tag{3.8}$$

is an isoparametric surface of $\mathbf{T}$, which is just a bivariate tensor product univariate B-spline. This can be evaluated with standard surface techniques, either using subdivision and refinement techniques or using the method of evaluating a surface as a series of curves. Now, consider evaluating an isocurve with fixed values of w and v. Then, for $i = 1, \ldots, n_{s,1}$

$$\begin{aligned}\sigma_i &= \sum_{j=1}^{n_{s,2}} \gamma_{i,j} B_{j,p_2,\mathbf{t}^v}(\hat{v}) \\ &= \sum_{j=1}^{n_{s,2}} \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(\hat{w}) B_{j,p_2,\mathbf{t}^v}(\hat{v}) \end{aligned} \tag{3.9}$$

and

$$\mathbf{T}(u, \hat{v}, \hat{w}) \;=\; \sum_{i=1}^{n_{s,1}} \sigma_i B_{i,p_1,\mathbf{t}^u}(\hat{u}) \tag{3.10}$$

Thus, the control points/coefficients σ_i of the isocurve in a u are actually points on a B-spline surface (in v and w). Finally,

$$\begin{aligned}\mathbf{T}(\hat{u}, \hat{v}, \hat{w}) &= \sum_{i=1}^{n_{s,1}} \sum_{j=1}^{n_{s,2}} \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(\hat{w}) B_{j,p_2,\mathbf{t}^v}(\hat{v}) B_{i,p_1,\mathbf{t}^u}(\hat{u}) \\ &= \sum_{i=1}^{n_{s,1}} \sum_{j=1}^{n_{s,2}} \gamma_{i,j} B_{j,p_2,\mathbf{t}^v}(\hat{v}) B_{i,p_1,\mathbf{t}^u}(\hat{u}) \\ &= \sum_{i=1}^{n_{s,1}} \sigma_i B_{i,p_1,\mathbf{t}^u}(\hat{u}) \end{aligned} \tag{3.11}$$

The roles of u, v, and w in the formulation above are interchangeable. More generally, define

$$\gamma_{i,j}(w) \;=\; \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(w), \tag{3.12}$$

and

$$\begin{aligned}\sigma_i(v, w) &= \sum_{j=1}^{n_{s,2}} \gamma_{i,j}(w) B_{j,p_2,\mathbf{t}^v}(v) \\ &= \sum_{j=1}^{n_{s,2}} \sum_{l=1}^{n_{s,3}} \mathbb{P}_{i,j,l} B_{l,p_3,\mathbf{t}^w}(w) B_{j,p_2,\mathbf{t}^v}(v). \end{aligned} \tag{3.13}$$

There are several ways to use the tensor-product nature of $\mathbf{T}(u, v, w)$ to evaluate it at $(\hat{u}, \hat{v}, \hat{w})$: the $n_{s,1} + 1$ isoparametric surfaces $\sigma_i(v, w)$ can be evaluated at $(\hat{v}, \hat{w})$, after which the resulting curve in u can be evaluated at $\hat{u}$, the $\sigma_i(\hat{v}, \hat{w})$ are the curve coefficients and alternatively, the $(n_{s,1}+1)(n_{s,2}+1)$ isoparametric curves, $\gamma_{i,j}(w)$ can be evaluated at w=$\hat{w}$. This yields an isoparametric surface, which can be evaluated at (u, v)= $\hat{u}, \hat{v}$.

3.5 Polyvariate B-spline

Unidimensional splines can be extended to multidimensional ones through the use of tensor product spline construction. A spline subspace $B_{i_j,p_j,t_j}(x_j)$ is defined for each dimension where x_j denotes the variable in the j-th dimension. Thus, the spline representation of a multidimensional function $\mathbf{C}(x_1, \ldots, x_m)$ is given as

$$\mathbf{C}(x_1, \ldots, x_m) = \sum_{i_1}^{n_s} \ldots \sum_{i_m}^{n_s} \mathbb{P}_{i_1,\ldots,i_m} B_{i_1,p_1,\mathbf{t}_1}(x_1) \ldots B_{i_m,p_m,\mathbf{t}_m}(x_m) \tag{3.14}$$

Similar to the unidimensional case, the construction of the above multidimensional spline polynomials can be done by solving a corresponding set of linear equations.

3.6 SIM vs. SAM

Using a set of control points $\{\mathbb{P}_1, \ldots, \mathbb{P}_{n_s}\} \in R^{\mathtt{d}}$, $\mathtt{d} = 2, 3$, can generate a piece-wise polynomial curve $\mathbf{C}$, by using B-spline approximation or interpolation as follows:

1. Interpolation: Interpolate through the given points (for a given parametrization $\mathbf{x} = (x_0, \ldots, x_{n_s})$), create $\mathbf{C}$ i.e.,

$$\mathbf{C}(x_i) = \mathbb{P}_i \tag{3.15}$$

2. Approximation: Find a curve $\mathbf{C}$ that will pass as close as possible to the given set of points. The distance is measured at the parameter values $\mathbf{x} = (x_0, \ldots, x_{n_s})$. The closest spline is defined by:

$$\min_{\mathbf{t},\mathbf{x}} D(\mathbf{t}, \mathbf{x}) \tag{3.16}$$

where

$$D(\mathbf{x}, \mathbf{t}) = \sum_{i=1}^{n_s} \|\mathbf{C}(x_i) - \mathbb{P}_i\|^2 \tag{3.17}$$

Two most popular curve approximation techniques are available in literature. One scheme is based on a deterministic approach using quadratic B-spline and the other scheme uses a genetic algorithm (GA) in its formulation where the B-spline can have any order [12], but here the B-splines interpolation has been used for our analysis.

Generally, for a given basic sequence of B-splines $\{B_{i,p,\mathbf{t}}\}_{i=1}^{n_s}$ and strictly increasing sequence of data series $\{x_j\}_{j=1}^{n_s}$, the B-spline interpolation function $\hat{c}(x)$ can be written as

$$\hat{c}(x) \quad = \quad \sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x) \tag{3.18}$$

where $\hat{c}(x)$ agrees with function $c(x)$ at all x_j if and only if

$$\sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x_j) \quad = \quad c(x_j), \qquad \text{for} \;\; j = 1, \ldots, n_s \tag{3.19}$$

In fact, (3.19) is a linear system of n_s equations with n_s unknown values of $\mathbb{P}_i$ and the i-th row and j-th column of the coefficient matrix equals $B_{i,p,\mathbf{t}}(x_j)$, which implies that the spline interpolation function can be found by solving a set of linear system equations [17]. The coefficient matrix can be verified for invertibility using Schoenberg-Whitney theorem [53].

3.7 Properties of B-splines

3.7.1 Local Control

Each segment is determined $d+1$ control points. If $t \in [t_r, t_{r+1})(d \leq r \leq \tau - d - 1)$, then

$$\mathbf{C}(x) \;=\; \sum_{i=r-d}^{r} \mathbb{P}_i B_{i,p,\mathbf{t}}(x), \tag{3.20}$$

Thus to evaluate $\mathbf{C}(x)$ it is sufficient to evaluate $B_{r-d,p,\mathbf{t}}(x), \ldots, B_{r,p,\mathbf{t}}(x)$

3.7.2 Convex Hull

If $t \in [t_r, t_{r+1})(d \leq r \leq \tau - d - 1)$, then

$$\mathbf{C}(x) \in \mathrm{CH}\{\mathbb{P}_{r-d}, \ldots, \mathbb{P}_r\}. \tag{3.21}$$

3.7.3 Invariance under Affine Transformation

Let $\mathbf{T}$ be an affine transformation. Then

$$\mathbf{T}\left(\sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x)\right) = \sum_{i=1}^{n_s} \mathbf{T}\left(\mathbb{P}_i\right) B_{i,p,\mathbf{t}}(x) \tag{3.22}$$

3.7.4 Differential of a B-spline

$$\mathfrak{D}(\sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x)) = (p-1)\sum_{i=1}^{n_s+1} \frac{\mathbb{P}_i - \mathbb{P}_{i-1}}{t_{i+p-1} - t_i} B_{i,p-1,\mathbf{t}}(x) \tag{3.23}$$

where $\mathfrak{D}(\cdot)$ denotes the differential operator, and both $\mathbb{P}_0$, and $\mathbb{P}_{n_s+1}$ are zeros.

3.7.5 Integral of a B-spline

The integral of a B-spline over the interval $[t_1, x]$ $(t_1 \leq x \leq t_s)$ is given by

$$\int_{t_1}^{x} \sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p}(c)dc = \sum_{i=1}^{s-1}(\sum_{j=1}^{i} \mathbb{P}_j (t_{j+p} - t_j)/p) B_{i,p+1}(x) \tag{3.24}$$

3.7.6 Positivity property

The B-splines $B_{i,p,\mathbf{t}}$ is made up of at most p nontrivial polynomial pieces and vanishes outside the interval $[t_i, ..., t_{i+p}]$ and is positive on the interior of that interval [17]. That is,

$$B_{i,p,\mathbf{t}}(x) > 0, \qquad t_i < x < t_{i+p} \tag{3.25}$$

while

$$B_{i,p,\mathbf{t}}(x) = 0 \quad \text{for} \quad t_i = t_{i+p} \tag{3.26}$$

With the positivity property of B-splines and having $\mathbb{P}_i \geq 0$ for all i, the spline representation of probability densities and likelihood functions all can be positive.

3.7.7 Knot insertion property

Additional knots can be inserted with the following knot insertion property [53]. If the knot sequence $\hat{\mathbf{t}}$ is obtained from the knot sequence $\mathbf{t}$ by the insertion of just one term, say x, then for any function $c \in \Psi$

$$\sum_i \mathbb{P}_i B_{i,p,\mathbf{t}}(x) \quad = c = \quad \sum_i \hat{\mathbb{P}}_i B_{i,p,\hat{\mathbf{t}}}(x) \tag{3.27}$$

with

$$\hat{\mathbb{P}}_i = (1 - \hat{w}_{i,p}(x))\mathbb{P}_{i-1} + \hat{w}_{i,p}(x)\mathbb{P}_i \quad \text{for all } i \tag{3.28}$$

where Ψ is the span of $B_{i,p,\mathbf{t}}(x)$ and $\hat{w}_{i,p}(x) = \frac{x-\hat{t}_i}{\hat{t}_{i+p-1}-\hat{t}_i}$.

3.7.8 Knot removal property

Knot removal is the reverse process of knot insertion. Let

$$c(x) \quad = \quad \sum_{i=1}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}}(x) \tag{3.29}$$

be defined on x, and let t_e be an interior knot of multiplicity s in $\mathbf{t}$; end knots are not removed. The size of the knot vector is $\tau = \|\mathbf{t}\|$, $e \in \{p, \dots, \tau - p\}$ and $1 \leq s \leq p$. Let $\mathbf{t}_r$ denote the knot vector obtained by removing t_e r times from $\mathbf{t}(e \leq r \leq s+e)$. Note that t_e is r times removable if $c(x)$ has a precise representation of the form [53]

$$c(x) = \sum_{i=1}^{n_s - r} \mathbb{Q}_i \hat{B}_{i,p,\mathbf{t}_r}(x) \tag{3.30}$$

where $\hat{B}_{i,p,\mathbf{t}_r}(x)$ are the basis functions on $\mathbf{t}_r$, that is equations (3.29) and (3.30) geometrically and parametrically represent the same curve. Hence, the knot t_e is r times removable if and only if the curve $c(x)$ is C^{p-s+r} continuous at $t_e s$. The new control points denoted by $\mathbb{Q}_i$ can be determined as described in [53]. Thus, the equations for removing t_e r-th time are

$$\mathbb{Q}_i = \begin{cases} \mathbb{Q}^r_{i=k} = \frac{\mathbb{Q}^{r-1}_k - (1-\alpha_k)\mathbb{Q}^r_{k-1}}{\alpha_k} & (e-p-r+1) \leq k \leq 0.5(2e-p-s-r), \\ \mathbb{Q}^r_{i=j} = \frac{\mathbb{Q}^{r-1}_j - \alpha_j \mathbb{Q}^r_{j+1}}{(1-\alpha_j)} & 0.5(2e-p-s+r+1) \leq j \leq (e-s+r-1). \end{cases} \tag{3.31}$$

with

$$\alpha_k = \frac{t - t_k}{t_{k+p+r} - t_k} \tag{3.32}$$

and

$$\alpha_j = \frac{t - t_{j-r+1}}{t_{j+p+1} - t_{j-r+1}} \tag{3.33}$$

3.7.9 Schoenberg-Whitney theorem

Let $\mathbf{t}$ be a knot vector, p and n be integers such that $n > p > 0$, and suppose x is strictly increasing with $n+1$ elements. Then matrix $L = B_{i,p,\mathbf{t}}(x_j)$ from (3.19) is invertible if and only if $B_{i,p,\mathbf{t}}(x_i) \neq 0$, $i = 0, \ldots, n$, i.e., if and only if $t_i < x_i < t_{i+p+1}$, for all i [53].

Chapter 4

The Spline Probability Hypothesis Density Filter

4.1 Introduction

The Probability Hypothesis Density (PHD) filter is a multitarget tracker that can alleviate the computational intractability of the optimal multitarget Bayes filter. The PHD filter recursively estimates the number of targets and their PHD from a set of observations and works well in scenarios with false alarms and missed detections. Two distinct PHD filter implementations are available in the literature: the Sequential Monte Carlo Probability Hypothesis Density (SMC-PHD) and the Gaussian Mixture Probability Hypothesis Density (GM-PHD) filters. While particle-based PHD implementations may suffer from degeneracy, GM-based methods may not be suitable for highly nonlinear non-Gaussian systems. This chapter proposes a B-Spline based Spline Probability Hypothesis Density (SPHD) filter, which has the capability to better approximate any arbitrary probability density function. The resulting algorithm can handle linear, non-linear, Gaussian, and non-Gaussian models. The SPHD filter can provide continuous estimates of the probability density function of the system state and it is immune to the degeneracy problem. The SPHD filter can maintain highly accurate tracks by taking advantage of dynamic knot movement, but at the expense of higher computational complexity, which makes it suitable for tracking a few high-value targets under difficult conditions. The SPHD filter derivations and simulations are provided in this chapter.

4.2 SPHD Filtering

This section describes the proposed SPHD filter derivation of the PHD filter. Consider a spatial Poisson process $X_k = \{\mathbf{x}_{i,k}\}_{i=1}^{\vartheta_k} \in \mathcal{E}_s$, where each $\mathbf{x}_k$ is a random target vector with survival probability $P_{\mathfrak{s},k}(\mathbf{x}_k)$ in a state space $\mathcal{E}_s$ and ϑ_k denotes the number of targets at a particular time k.

Denote the PHD and the SPHD of the multitarget state by $D_{k|k}(\mathbf{x}_k)$ and $\mathbf{B}_{k|k}(\mathbf{x}_k)$, respectively. Consider the following observation model with the detection probability $P_{\mathfrak{d},k}(\mathbf{x}_k)$ where each target in state space $\mathcal{E}_s$ generates a noisy observation in an observation space $\mathcal{E}_o$ through a kernel with density $p_k(\mathbf{z}_k|\mathbf{x}_k)$. This happens independently for each target in X_k.

Let $Z_k = \{\mathbf{z}_{i,k}\}_{i=1}^{\eta_k} \in Z^{(k)} \in \mathcal{E}_o$ denote the set of observations generated by targets in X_k and η_k denote the number of observations at time k. In addition to these detected targets, observations might be due to false alarms or clutter. False alarms are Poisson-distributed with an average rate of λ_k and the spatial density of false alarms is $C_k(\mathbf{z}_k)$.

4.2.1 Unidimensional SPHD filtering

To illustrate the Spline concept, we first derive a unidimensional SPHD and then generalize it to the multidimensional case.

Prediction

Assume that the prior multitarget SPHD of the system state at time $(k-1)$ is

$$D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) \approx \sum_{i}^{n_s} \mathbb{P}_i B_{i,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) \tag{4.1}$$

and for brevity (4.1) can be written as

$$\mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}) = \sum_{i} \mathbb{P}_i B_{i,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) \tag{4.2}$$

where $\mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1})$ denotes the multitarget prior SPHD, $\mathbf{t}_{k-1} = \{t_{1,k-1}, \ldots, t_{\tau,k-1}\}$ is the set of prior knots of the splines and $\mathbb{P}_i$ denotes the i-th control point of the prior SPHD. The number of knots is denoted by τ and n_s denotes the number of control points. The prior number of expected targets is the integral of the region of state space $\mathcal{E}_s$, which can be determined as

$$\hat{N}_{k-1|k-1} = \int_{\mathcal{E}_s} \mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}) d\mathbf{x}_{k-1} \tag{4.3}$$

The spline system model Markov transition density $p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$ is a two dimensional function determined using system model (2.1), and is given as

$$p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) = \sum_{r,y} \mathbb{P}_{r,y} B_{r,p,\mathbf{t}_k}(\mathbf{x}_k) B_{y,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) \tag{4.4}$$

where $\mathbf{t}_k = \{t_{1,k}, \ldots, t_{\tau,k}\}$ is the set of sequences of predicted knots. These predicted knots can be determined as [24]

$$t_{j,k} \;=\; f(t_{j,k-1}, 0) \qquad \text{for} \;\; j = 1, \ldots, \tau \tag{4.5}$$

where the $f(.,.)$ is the function of state transition as in (2.1) and this method of determining knots ensures that spline covers the region where the probability of density of system state is significant. The coefficients or control points of spline transition density $\mathbb{P}_{r,y}$ can be determined as described in [25].

A target may continue to exist, die or spawn or new targets may be born independently from existing ones. Thus considering these random events, the predicted SPHD can be determined using (2.23) [38] as

$$\begin{aligned} \mathbf{B}_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) &= \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k) + \mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k) + \mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k) \\ &= \mathbf{B}_{k|k-1}(\mathbf{x}_k) \end{aligned} \tag{4.6}$$

The predicted SPHD for the existing targets can be determined as [38]

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k) &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ &\quad \cdot D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) d\mathbf{x}_{k-1} \\ &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) \sum_{r,y} \mathbb{P}_{r,y} B_{r,p,\mathbf{t}_k}(\mathbf{x}_k) B_{y,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) \\ &\quad \cdot \sum_{i} \mathbb{P}_i B_{i,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) d\mathbf{x}_{k-1} \\ &= \sum_{r,y} \mathbb{P}_{r,y} B_{r,p,\mathbf{t}_k}(\mathbf{x}_k) \sum_{i} \mathbb{P}_i \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) \\ &\quad \cdot B_{y,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) B_{i,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) d\mathbf{x}_{k-1} \end{aligned} \tag{4.7}$$

where the third equality follows from the property that the order of summation and integration of splines is interchangeable [12]. Let

$$\begin{aligned} \xi_{i,y} &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) B_{y,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) \\ &\quad \cdot B_{i,p,\mathbf{t}_{k-1}}(\mathbf{x}_{k-1}) d\mathbf{x}_{k-1} \end{aligned} \tag{4.8}$$

which are integrals of polynomials. Then the predicted SPHD of the system state can be expressed as

$$D_{\mathbf{c},k|k-1}(\mathbf{x}_k) = \sum_r \sum_y \mathbb{P}_{r,y} B_{r,p,\mathbf{t}_k}(\mathbf{x}_k) \sum_i \mathbb{P}_i \xi_{i,y} \tag{4.9}$$

where the coefficient of the predicted SPHD is given as

$$\mathbb{P}_r = \sum_y \sum_i \mathbb{P}_{r,y} \mathbb{P}_i \xi_{i,y} \tag{4.10}$$

Assume $\psi^r_{y,i} = \mathbb{P}_{r,y}\mathbb{P}_i$ for any fixed r. Then a more concise expression for the coefficient can be written as $\mathbb{P}_r = tr(\psi^\tau \xi)$. Then the equation could be simplified as

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k) &= \sum_r \mathbb{P}_r B_{r,p,\mathbf{t}_k}(\mathbf{x}_k) \\ &= \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k) \end{aligned} \tag{4.11}$$

where $\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k)$ denotes the existing targets predicted SPHD. A similar approach can be applied to determine the spawned targets predicted SPHD $\mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k)$. The SPHD of new targets $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k)$ could be determined using proposition 2 from [69] where the posterior probability for an observed measurement that originates from a new target $P_k(Y_i)$ is determined as

$$\begin{aligned} P_k(Y_i) \quad &= \quad \frac{\int P_{\mathfrak{d},k}(\mathbf{x}_k)p(\mathbf{z}_i|\mathbf{x}_k)D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})d\mathbf{x}_k}{\lambda_k C_k(\mathbf{z}_i) + \int P_{\mathfrak{d},k}(\mathbf{x}_k)p(\mathbf{z}_i|\mathbf{x}_k)D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})d\mathbf{x}_k} \ , \\ &\quad\ \mathbf{z}_i \in Z_k \ , \ i = 1, \dots, \eta_k \end{aligned} \tag{4.12}$$

In the above, $D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \approx D_{\mathbf{c},k|k-1}(\mathbf{x}_k) + D_{\mathbf{s},k|k-1}(\mathbf{x}_k)$, Y_i denotes an i-th observed measurement that originates from a new target at time k and η_k denotes the total number of measurements at time k. Also $P_k(Y_i)$ values are determined for each measurement and it is compared with a tuning threshold probability ϵ. That is,

$$N_{\mathbf{nb},i} = \begin{cases} 1 \quad \text{if} \quad P_k(Y_i) \leqslant \epsilon, \\ \\ 0 \qquad \text{otherwise.} \end{cases} \tag{4.13}$$

A new target must be added if the number of newborn targets $N_{\mathbf{nb},i}$ is 1 for a specific measurement index i. This is achieved by adding a Gaussian distribution with mean $\mathbf{z}_i$ that originated from a new target and variance of measurement noise. Equation (4.12) can be written in terms of spline representation as

$$\begin{aligned} P_k(Y_i) \quad &= \quad \frac{\int P_{\mathfrak{d},k}(\mathbf{x}_k)\mathbf{B}_{l,k}(\mathbf{z}_i|\mathbf{x}_k)\mathbf{B}_{k|k-1}(\mathbf{x}_k)}{\mathbf{B}_{\lambda,k}(\mathbf{z}_i) + \int P_{\mathfrak{d},k}(\mathbf{x}_k)\mathbf{B}_{l,k}(\mathbf{z}_i|\mathbf{x}_k)\mathbf{B}_{k|k-1}(\mathbf{x}_k)} \ , \\ &\quad\ \mathbf{z}_i \in Z_k \ , \ i = 1, \dots, \eta_k \end{aligned} \tag{4.14}$$

where $\mathbf{B}_{k|k-1}(\mathbf{x}_k) \approx \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k) + \mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k)$ and $\mathbf{B}_{l,k}(\cdot)$ denotes the spline likelihood density and could be determined using the measurement model (2.2) over the interval $[t_{1,k}, t_{\tau,k}]$. The spline uniform clutter density is denoted by $\mathbf{B}_{\lambda,k}(\cdot)$. Then $P_k(Y_i)$ value is compared with the tuning threshold ϵ. Using (4.13) a newborn target SPHD can be added as

$$\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k) \;=\; \sum_{i=1}^{N_{\mathbf{nb}}} \mathbf{B}_{\mathbf{nb},k,i}(\mathbf{z}_i) \tag{4.15}$$

where $\mathbf{B}_{\mathbf{nb},k,i}$ is the SPHD of a newborn target with mean $\mathbf{z}_i$ and variance of measurement noise. The total number of newborn targets per scan is denoted by $N_{\mathbf{nb}}$ and $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k)$ denotes the cumulative sum of all the SPHD of newborn targets at scan k.

Update

Note that the SPHD filter provides the PHD estimates in a continuous space in state. These predicted SPHD $\mathbf{B}_{k|k-1}(\cdot)$ at any point over the interval $[t_{1,k}, t_{\tau,k}]$ can be determined using (4.6). Then, the interval where $\mathbf{B}_{k|k-1}(\cdot)$ is significant could be found. Using the measurement model equation (2.2), the value for the likelihood density function $\mathbf{B}_{l,k}(\mathbf{z}_k|\mathbf{x}_k)$ can be evaluated for the same interval. The updated posterior SPHD can be determined as [38]

$$\begin{aligned} D_{k|k}(\mathbf{x}_k|Z^{(k)}) &= (1 - P_{\mathfrak{d},k}(\mathbf{x}_k))D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \\ &\quad + \sum_{\mathbf{z}_k \in Z_k} \frac{P_{\mathfrak{d},k}(\mathbf{x}_k)p_{k|k}(\mathbf{z}_k|\mathbf{x}_k)D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})}{\lambda_k C_k(\mathbf{z}_k) + \phi_k(\mathbf{z}_k|Z^{(k-1)})} \end{aligned} \tag{4.16}$$

$$\begin{aligned} D_{k|k}(\mathbf{x}_k|Z^{(k)}) &= (1 - P_{\mathfrak{d},k}(\mathbf{x}_k))\mathbf{B}_{k|k-1}(\mathbf{x}_k) \\ &\quad + \sum_{\mathbf{z}_k \in Z_k} \frac{P_{\mathfrak{d},k}(\mathbf{x}_k)\mathbf{B}_{l,k}(\mathbf{z}_k|\mathbf{x}_k)\mathbf{B}_{k|k-1}(\mathbf{x}_k)}{\mathbf{B}_{\lambda,k}(\mathbf{z}_k) + \phi_k(\mathbf{z}_k|Z^{(k-1)})} \\ &= \sum_j \mathbb{P}_j B_{j,p,\mathbf{t}_k}(\mathbf{x}_k) \end{aligned} \tag{4.17}$$

where the $\phi_k(\mathbf{z}_k|Z^{(k-1)})$ can be evaluated as

$$\phi_k(\mathbf{z}_k|Z^{(k-1)}) \;=\; \int P_{\mathfrak{d},k}(\mathbf{x}_k)\mathbf{B}_{l,k}(\mathbf{z}_k|\mathbf{x}_k)\mathbf{B}_{k|k-1}(\mathbf{x}_k)d\mathbf{x}_k \tag{4.18}$$

Then the updated SPHD can be written

$$\mathbf{B}_{k|k}(\mathbf{x}_k) \;=\; \sum_j \mathbb{P}_j B_{j,p,\mathbf{t}_k}(\mathbf{x}_k) \tag{4.19}$$

where $\mathbf{t}_k$ denotes the set of posterior knots and (4.19) ensures that the multitarget posterior SPHD $\mathbf{B}_{k|k}(\mathbf{x}_k)$ is only evaluated over the interval where it is significant. Once the significant region is obtained, a simple way of selecting the knots for the posterior SPHD is to uniformly distribute the knots over this significant region [24]. The number of updated targets can be determined as

$$\hat{N}_{k|k} \;=\; \int_{\mathcal{E}_s} \mathbf{B}_{k|k}(\mathbf{x}_k)d\mathbf{x}_k \tag{4.20}$$

4.2.2 Multidimensional SPHD filtering

The tensor product transformation of splines can be used to approximate the SPHD [17] at higher dimensions. A polyvariate model is used here to derive the SPHD filter implementation for multidimensional multitarget state space models. Assume that a multidimensional multitarget system state at time k is denoted as $X_k = \{\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{\vartheta_k,k}\}$ where each of the targets has multidimensional state $\mathbf{x}_k = [\mathbf{x}_k^1, \ldots, \mathbf{x}_k^n]'$ and n denotes the number of dimensions.

Prediction

Assume that the multitarget multidimensional prior SPHD of the system state at time $(k-1)$ is

$$\begin{aligned} D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)}) &\approx \sum_{i_1}^{n_s}\cdots\sum_{i_n}^{n_s} \mathbb{P}_{i_1,\ldots,i_n} B_{i_1,p,\mathbf{t}^1_{k-1}}(\mathbf{x}^1_{k-1})\ldots B_{i_n,p,\mathbf{t}^n_{k-1}}(\mathbf{x}^n_{k-1}) \\ &= \mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}) \end{aligned} \tag{4.21}$$

where $\mathfrak{i} = i_1, \ldots, i_n$ and n denotes the number of dimensions. The number of knots for all the dimensions is the same (τ). The n dimensional knot $\mathbf{t}_{k-1} = \{\mathbf{t}^1_{k-1}, \ldots, \mathbf{t}^n_{k-1}\}$ is an $n \times \tau$ array. Each row vector of $\mathbf{t}_{k-1}$ consists of a set of prior knots $\mathbf{t}^l_{k-1} = \{t^l_{1,k-1}, \ldots, t^l_{\tau,k-1}\}$ where $l = 1, \ldots, n$. The n dimensional control point or coefficient matrix is denoted by $\mathbb{P}_{\mathfrak{i}}$ and n_s denotes the number of control points. The number of control points for all the dimensions is the same. Note that the number of knots must be greater than the number of control points. The system state transition density function $p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$ is a combination of two polyvariate functions with spline representation. That is,

$$\begin{aligned} p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) &= \sum_{j_1}\cdots\sum_{j_{2n}} \mathbb{P}_{j_1,\ldots,j_{2n}} B_{j_1,p,\mathbf{t}^1_k}(\mathbf{x}^1_k)\ldots B_{j_n,p,\mathbf{t}^n_k}(\mathbf{x}^n_k) \\ &\quad \cdot B_{j_{n+1},p,\mathbf{t}^1_{k-1}}(\mathbf{x}^1_{k-1})\ldots B_{j_{2n},p,\mathbf{t}^n_{k-1}}(\mathbf{x}^n_{k-1}) \end{aligned} \tag{4.22}$$

where $\mathfrak{j} = \{j_1, \ldots, j_{2n}\}$ and $\mathbf{t}_k$ denotes an $n \times \tau$ knot array at k and it consists of row vectors $\mathbf{t}^1_k, \ldots, \mathbf{t}^n_k$. Each row vector of $\mathbf{t}_k$ consists of a set of predicted knots $\mathbf{t}^l_k = \{t^l_{1,k}, \ldots, t^l_{\tau,k}\}$ where $l = 1, \ldots, n$. The predicted knot selection of a multidimensional system is much more challenging. A suboptimal but computationally efficient method is used here to find the predicted knots for the multidimensional spline. Assume that there are $\tau = \tau_l$ knots in $\mathbf{t}^l_{k-1}$. Then $\prod\limits_l \tau_l$ different sample vectors $\boldsymbol{\Psi}_{k-1,t}$ can be formed by selecting one knot $t^l_{e_l,k-1}$ from the set of knots in each dimension $\mathbf{t}^l_{k-1}$ and $e_l = 1, \ldots, \tau$. For example, the collection of all such sample vectors in the n-dimensional spline filter is $\boldsymbol{\Psi}_{k-1,t} = \{[t^l_{e_1,k-1} \ldots t^l_{e_n,k-1}]'\}_{e_1,\ldots,e_n}$ and the total number of $\boldsymbol{\Psi}_{k-1,t}$ is $\tau_1 \times \ldots \times \tau_n$. Then, the predicted sample knots $\boldsymbol{\Psi}_{k,t}$ are found by

$$\boldsymbol{\Psi}^{e_1,\ldots,e_n}_{k,t} \quad = \quad f_k(\boldsymbol{\Psi}^{e_1,\ldots,e_n}_{k-1,t}, 0) \tag{4.23}$$

where $f_k(\cdot,\cdot)$ is the transition function in (2.1). Then, to select the predicted knots in each dimension l, first project all sample vectors $\boldsymbol{\Psi}^{e_1,\ldots,e_n}_{k,t}$ into the axis of the l-th dimension [24,25]. In the n dimensional case, the projection will result in $\tau_1 \times \cdots \times \tau_n$ possibly overlapping points in each axis. Denote them as $\{t^l_{1',k}, t^l_{2',k}, \ldots, t^l_{(\tau_1\times\ldots\times\tau_n)',k}\}$ and assume that they are sorted in a non-decreasing order. Then, select the first predicted knot as $t^l_{1,k} = t^l_{1',k}$ and the last predicted knot as $t^l_{\tau_l,k} = t^l_{(\tau_1\times\ldots\times\tau_n)',k}$. The remaining $\tau_l - 2$ knots are selected for each dimension as $t^l_{i'',k} = t^l_{\phi(i''),k}$, where

$$\phi(i^{''}) \quad = \quad 1 + \lceil((\prod_l \tau_l) - 2)/(\tau_l - 2)\rceil \cdot (i^{''} - 1) \tag{4.24}$$

The selection of $2n$ dimensional control point $\mathbb{P}_\mathrm{j}$ is described next. For simplicity, assume that the multitarget multidimensional system equation (2.1) can be written as

$$\mathbf{x}_k \quad = \quad f_k(\mathbf{x}_{k-1}) + \nu_k \tag{4.25}$$

The l-th dimensional system noise is white Gaussian with variance $\sigma_{\nu,l}^2$. Then the coefficients or control points of spline transition density $\mathbb{P}_{j_1,\ldots,j_{2n}}$ can be determined as follows:

The l-th dimensional range of $\mathbf{x}_{k-1}^l$ at scan $k-1$ is selected as

$$\begin{aligned} \mathbf{x}_{\text{range},k-1|k-1}^l &= [\min(\mathbf{t}_{k-1}^l) \quad \min(\mathbf{t}_{k-1}^l) + \delta_{k-1|k-1}^l \\ &\quad \min(\mathbf{t}_{k-1}^l) + 2\delta_{k-1|k-1}^l \quad \ldots \quad \max(\mathbf{t}_{k-1}^l)] \end{aligned} \tag{4.26}$$

where $\mathbf{x}_{\text{range},k-1|k-1}^l$ and $\mathbf{t}_{k-1}^l$ denote the l-th dimensional range of $\mathbf{x}_{k-1|k-1}$ and l-th dimensional knot sequence vector, respectively. Also, the l-th dimensional $\delta_{k-1|k-1}^l$ is the step size and it can be determined as

$$\delta_{k-1|k-1}^l = \frac{\max(\mathbf{t}_{k-1}^l) - \min(\mathbf{t}_{k-1}^l)}{n_s - 1} \qquad l = 1, \ldots, n \tag{4.27}$$

where n_s denotes the number of control points.

Next, define two column vectors as

$$\mathbf{x}_{\text{range},k-1|k-1}^{\min} = \begin{bmatrix} \min(\mathbf{x}_{\text{range},k-1|k-1}^1) \\ \vdots \\ \min(\mathbf{x}_{\text{range},k-1|k-1}^n) \end{bmatrix} \tag{4.28}$$

and

$$\mathbf{x}_{\text{range},k-1|k-1}^{\max} = \begin{bmatrix} \max(\mathbf{x}_{\text{range},k-1|k-1}^1) \\ \vdots \\ \max(\mathbf{x}_{\text{range},k-1|k-1}^n) \end{bmatrix} \tag{4.29}$$

Then, using the state transition equation (4.25), (4.28) and (4.29), the n dimensional predicted state of $\mathbf{x}^{\min}_{\text{range},k|k-1}$ and $\mathbf{x}^{\max}_{\text{range},k|k-1}$ can be determined as follows:

$$\mathbf{x}^{\min}_{\text{range},k|k-1} = f_k(\mathbf{x}^{\min}_{\text{range},k-1|k-1}) \tag{4.30}$$

and

$$\mathbf{x}^{\max}_{\text{range},k|k-1} = f_k(\mathbf{x}^{\max}_{\text{range},k-1|k-1}) \tag{4.31}$$

Using (4.30) and (4.31), the l-th dimensional updated state of $\mathbf{x}^{l}_{\text{range},k|k}$ is given by

$$\begin{aligned}\mathbf{x}^{l}_{\text{range},k|k} = [&\mathbf{x}^{\min}_{\text{range},k|k-1}(l) - \sigma_{\nu,l} \quad \mathbf{x}^{\min}_{\text{range},k|k-1}(l) - \sigma_{\nu,l} + \delta^{l}_{k|k-1} \\ &\mathbf{x}^{\min}_{\text{range},k|k-1}(l) - \sigma_{\nu,l} + 2\delta^{l}_{k|k-1} \quad \ldots \quad \mathbf{x}^{\max}_{\text{range},k|k-1}(l) + \sigma_{\nu,l}]\end{aligned} \tag{4.32}$$

where the l-th dimensional step size can be determined as

$$\delta^{l}_{k|k-1} = \frac{\mathbf{x}^{\max}_{\text{range},k|k-1}(l) - \mathbf{x}^{\min}_{\text{range},k|k-1}(l) + 2\sigma_{\nu,l}}{n_s - 1} \tag{4.33}$$

The spline transition density $p(\mathbf{x}_k|\mathbf{x}_{k-1}) = p(\rho_1, \ldots, \rho_n, \lambda_1, \ldots, \lambda_n)$ can be determined as

$$\begin{aligned}
p(\rho_1, \ldots, \rho_n, \lambda_1, \ldots, \lambda_n) \;=\; & \prod_{\lambda_1=1}^{\ell_{\lambda_1,k}} \cdots \prod_{\lambda_n=1}^{\ell_{\lambda_n,k}} \prod_{\rho_1=1}^{\ell_{\rho_1,k-1}} \cdots \prod_{\rho_n=1}^{\ell_{\rho_n,k-1}} \\
& \cdot \frac{1}{\sqrt{2\pi\sigma_{\nu,1}^2}} e^{\frac{-\left(\mathbf{x}^1_{\text{range},k|k}(\lambda_1) - \left(\mathbf{x}^1_{\text{range},k-1|k-1}(\rho_1) + \cdots + \mathbf{x}^n_{\text{range},k-1|k-1}(\rho_n)\right)\right)^2}{2\sigma_{\nu,1}^2}} \\
& \cdot \frac{1}{\sqrt{2\pi\sigma_{\nu,2}^2}} e^{\frac{-\left(\mathbf{x}^2_{\text{range},k|k}(\lambda_2) - \left(\mathbf{x}^2_{\text{range},k-1|k-1}(\rho_2) + \cdots + \mathbf{x}^n_{\text{range},k-1|k-1}(\rho_n)\right)\right)^2}{2\sigma_{\nu,2}^2}} \\
& \cdots \frac{1}{\sqrt{2\pi\sigma_{\nu,n-1}^2}} e^{\frac{-\left(\mathbf{x}^{n-1}_{\text{range},k|k}(\lambda_{n-1}) - \left(\mathbf{x}^{n-1}_{\text{range},k-1|k-1}(\rho_{n-1}) + \mathbf{x}^n_{\text{range},k-1|k-1}(\rho_n)\right)\right)^2}{2\sigma_{\nu,n-1}^2}} \\
& \cdot \frac{1}{\sqrt{2\pi\sigma_{\nu,n}^2}} e^{\frac{-\left(\mathbf{x}^n_{\text{range},k|k}(\lambda_n) - \left(\mathbf{x}^n_{\text{range},k-1|k-1}(\rho_n)\right)\right)^2}{2\sigma_{\nu,n}^2}} \qquad (4.34)
\end{aligned}$$

where $\ell_{\rho_l,k-1}$ and $\ell_{\lambda_l,k}$ denote the lengths of the l-th dimensional $\mathbf{x}^l_{\text{range},k-1|k-1}$ and $\mathbf{x}^l_{\text{range},k|k}$, respectively, and $l = 1, \ldots, n$.

Then,

$$\begin{aligned}
p(\rho_1, \ldots, \rho_n, \lambda_1, \ldots, \lambda_n) \;=\; & \sum_{j_1} \cdots \sum_{j_{2n}} \mathbb{P}_{j_1,\ldots,j_{2n}} B_{j_1,\mathbf{t}_k^1}(\mathbf{x}_k^1) \ldots \\
& \cdot B_{j_n,\mathbf{t}_k^n}(\mathbf{x}_k^n) \\
& \cdot B_{j_{n+1},p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1) \ldots \\
& \cdot B_{j_{2n},p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n) \qquad (4.35)
\end{aligned}$$

From (4.35), where the blending functions, $p(\rho_1, \ldots, \rho_n, \lambda_1, \ldots, \lambda_n)$ are already known from (4.34), the values of $\mathbb{P}_{j_1,\ldots,j_{2n}}$ can be determined by solving a set of linear equations with the unknown parameters being the coefficients.

For a general nonlinear, non-Gaussian system, finding the predicted PHD is more challenging. For example, assume that the system follows (2.1), but with process noise being white and additive with density $p_{\nu_k}(\nu_k)$, which can be expressed as

$$p_{\nu_k}(\nu_k) \;=\; p_{\nu_k}(\mathbf{x}_k - f_k(\mathbf{x}_{k-1})) \tag{4.36}$$

Also consider the measurement model

$$\mathbf{z}_k \;=\; h_k(\mathbf{x}_k) + \omega_k \tag{4.37}$$

where the measurement noise ω_k is additive white with density $p_{\omega_k}(\omega_k)$. It is also independent of the process noise. Then, $p_{\omega_k}(\omega_k)$ can be written as

$$p_{\omega_k}(\omega_k) \;=\; p_{\omega_k}(\mathbf{z}_k - h_k(\mathbf{x}_k)) \tag{4.38}$$

In view of the additivity of the process noise in (2.5) [3], one has

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) \;=\; p_{\nu_k}(\mathbf{x}_k - f_k(\mathbf{x}_{k-1})) \tag{4.39}$$

Now substituting (4.39) in (4.22), the transition density of system state given in (4.22) becomes a nonlinear convolution. The n dimensional coefficients $\mathbb{P}_{j_1,\ldots,j_{2n}}$ for a nonlinear, non-Gaussian system model can be determined as in the nonlinear, Gaussian case in (4.35). Note that for the non-Gaussian case, (4.32) and (4.33) need to be modified accordingly.

The multidimensional multitarget predicted SPHD of a system state can be written as

$$\mathbf{B}_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \;=\; \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k)+\mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k)+\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k) \tag{4.40}$$

where the multidimensional predicted SPHD for existing targets can be determined as

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k) \;&=\; \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1})p_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ &\quad \cdot D_{k-1|k-1}(\mathbf{x}_{k-1}|Z^{(k-1)})d\mathbf{x}_{k-1} \\ &=\; \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1})\sum_{j_1}\ldots\sum_{j_{2n}}\mathbb{P}_{j_1,\ldots,j_{2n}} \\ &\quad \cdot B_{j_1,p,\mathbf{t}_k^1}(\mathbf{x}_k^1)\ldots B_{j_n,p,\mathbf{t}_k^n}(\mathbf{x}_k^n) \\ &\quad \cdot B_{j_{n+1},p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1)\ldots B_{j_{2n},p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n) \\ &\quad \cdot\sum_{i_1}\ldots\sum_{i_n}\mathbb{P}_{i_1,\ldots,i_n}B_{i_1,p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1)\ldots B_{i_n,p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n) \\ &\quad \cdot d\mathbf{x}_{k-1}^1\ldots d\mathbf{x}_{k-1}^n \end{aligned} \tag{4.41}$$

Define two $2n$ dimensional matrices W and C, and one n dimensional matrix ξ as follows:

$$\begin{aligned} W_{j_{n+1},\ldots,j_{2n},i_1,\ldots,i_n} \;&=\; \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}) \\ &\quad \cdot B_{j_{n+1},p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1)\ldots B_{j_{2n},p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n) \\ &\quad \cdot B_{i_1,p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1)\ldots B_{i_n,p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n) \\ &\quad \cdot d\mathbf{x}_{k-1}^1\ldots d\mathbf{x}_{k-1}^n \end{aligned} \tag{4.42}$$

Let $C_{\mathsf{j}} = \mathbb{P}_{\mathsf{j}}$ and $\xi_{\mathsf{i}} = \mathbb{P}_{\mathsf{i}}$. Using (4.41), (4.42) and (4.44) with additional manipulation, it can be shown that

$$\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k) = \sum_{j_1}\dots\sum_{j_n}\mathbb{P}_{j_1,\dots,j_n}B_{j_1,p,\mathbf{t}_k^1}(\mathbf{x}_k^1)\dots B_{j_n,p,\mathbf{t}_k^n}(\mathbf{x}_k^n) \tag{4.43}$$

where $\mathbb{P}_{j_1,\dots,j_n}$ is given by

$$\mathbb{P}_{j_1,\dots,j_n} = \sum_{i_1}\dots\sum_{i_n}\xi_{i_1,\dots,i_n}\sum_{j_{n+1}}\dots\sum_{j_{2n}}C_{j_1,\dots,j_{2n}}W_{j_{n+1},\dots,j_{2n},i_1,\dots,i_n} \tag{4.44}$$

A similar approach, as described for multidimensional state space case to find the predicted SPHD of existing targets, $\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k)$, can be applied to determine the predicted SPHD of spawned targets, $\mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k)$.

The multidimensional SPHD of predicted newborn targets, $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k)$, can be determined as follows. Using (4.12) for the i-th measurement vector, $\mathbf{z}_i$ the $P_k(Y_i)$ (scalar) value can be determined and then using (4.13) a newborn target can be added. If for the i-th measurement, $\mathbf{z}_i$ the $P_k(Y_i) = 1$ then each element of $\mathbf{z}_i$ can be considered as the mean of a newborn target state in their respective dimension with the variance of measurement noise. A newborn target can be added using Gaussian distribution with corresponding mean and variance from each state element of that newborn target.

The SPHD of all predicted newborn targets can be written as

$$\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k) = \sum_{i=1}^{N_{\mathbf{nb}}}\mathbf{B}_{\mathbf{nb},k,i}(\mathbf{z}_i) \tag{4.45}$$

where $\mathbf{B}_{\mathbf{nb},k,i}(\mathbf{z}_i)$ is the SPHD of a predicted newborn target. The total number of newborn targets per scan is denoted by $N_{\mathbf{nb}}$ and $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k)$ denotes the cumulative sum of the SPHD of all predicted newborn targets at scan k.

Update

Similar to the unidimensional case, the posterior SPHD of the multidimensional multitarget system state

$$\begin{aligned} D_{k|k}(\mathbf{x}_k|Z^{(k)}) &= (1 - P_{\mathfrak{d},k}(\mathbf{x}_k))D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)}) \\ &\quad + \sum_{\mathbf{z}_k \in Z_k} \frac{P_{\mathfrak{d},k}(\mathbf{x}_k)p_{k|k}(\mathbf{z}_k|\mathbf{x}_k)D_{k|k-1}(\mathbf{x}_k|Z^{(k-1)})}{\lambda_k C_k(\mathbf{z}_k) + \phi_k(\mathbf{z}_k|Z^{(k-1)})} \end{aligned} \tag{4.46}$$

can be found after evaluating its value over the region defined by the knots $\mathbf{t}_k$, i.e., the n dimensional region covered by the splines. Then the updated SPHD can be written as

$$\mathbf{B}_{k|k}(\mathbf{x}_k) = \sum_{\iota_1,\ldots,\iota_n} \mathbb{P}_{\iota_1,\ldots,\iota_n} B_{\iota_1,p,\mathbf{t}_k^1}(\mathbf{x}_k^1)\ldots B_{\iota_n,p,\mathbf{t}_k^n}(\mathbf{x}_k^n) \tag{4.47}$$

4.3 Simulation Results

In this section a nonlinear non-Gaussian multitarget tracking example is used to validate the effectiveness of the proposed SPHD filter. The selected example is a multidimensional one dealing with a practical bearing-only tracking problem. The problem of bearing-only tracking arises in many practical applications such as submarine tracking or airborne surveillance using a passive radar [42]. As shown in

Figure 4.1, a sensor is on an aircraft with

$$\begin{aligned} x_p(k) &= \bar{x}_p(k) + \Delta x_p(k) \qquad k = 0, 1, \ldots, 40 && (4.48) \\ y_p(k) &= \bar{y}_p(k) + \Delta y_p(k) \qquad k = 0, 1, \ldots, 40 && (4.49) \end{aligned}$$

where $\bar{x}_p(k)$ and $\bar{y}_p(k)$ are the average platform position coordinates, k is the scan index and the perturbations $\Delta x_p(k)$ and $\Delta y_p(k)$ are assumed to be mutually independent zero-mean Gaussian white noise sequences with variances $\sigma^2_{\Delta x_p} = 1$ and $\sigma^2_{\Delta y_p} = 1$, respectively. The average unperturbed platform motion is assumed to be horizontal with a constant velocity. Its coordinates are given by

$$\bar{x}_p(k) = 100k * T \quad \text{(m)} \tag{4.50}$$

$$\bar{y}_p(k) = 10000 \quad \text{(m)} \tag{4.51}$$

where the sampling time $T = 10$ (s). Targets move along the X-axis with

$$x_i(k+1) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_i(k) + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} \nu_i(k) \tag{4.52}$$

where the target state is

$$x_i(k) = \begin{bmatrix} x_i^1(k) \\ x_i^2(k) \end{bmatrix} \qquad i = 1, 2, 3 \tag{4.53}$$

and x_i^1 denotes the position (m) while x_i^2 denotes the velocity (m/s) of the i-th target. Note that there are three targets on the ground. This problem, which has been used for the comparison of nonlinear filtering algorithms before [3,34], represents a multitarget ground target tracking using an airborne passive radar in the presence of false alarms and missed detections.

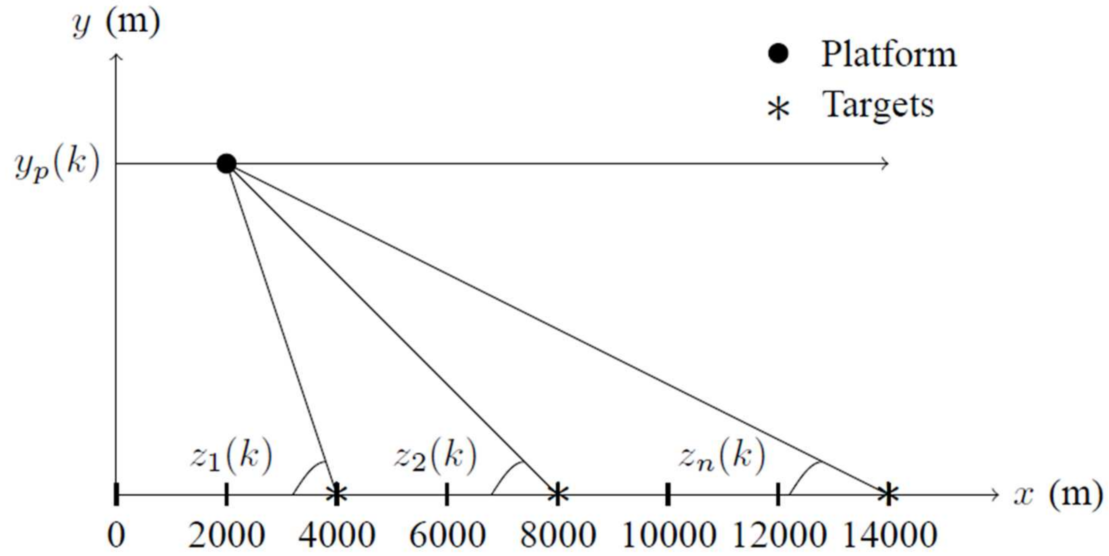

Figure 4.1: Motion of the platform and the three targets

These three targets, which have a probability of survival $P_{\mathfrak{s},k} = 0.98$, appear and disappear at specific times. The initial states are

$$[x_1(k)\ x_2(k)\ x_3(k)] = \begin{bmatrix} 400 & 6000 & 8000 \\ 20 & -20 & -25 \end{bmatrix} \tag{4.54}$$

and the start and end time steps are (1,40), (5,24) and (16,38), respectively. The process noise $\nu(k)$, which models the acceleration, is zero mean white Gaussian with standard deviation $\sigma_\nu = 0.01$ m/s^2. Each target is detected with probability $P_{\mathfrak{d},k} =$

0.98 and the target-originated measurements follow the observation model

$$z_i(k) = h[x_p(k), y_p(k), x_i^1(k)] + \omega(k) \qquad i = 1, 2, 3 \tag{4.55}$$

where

$$h[.] = \tan^{-1} \frac{y_p(k)}{x_i^1(k) - x_p(k)} \qquad i = 1, 2, 3 \tag{4.56}$$

is the angle between the X-axis and the line of sight from the sensor to the targets, and the sensor noise $\omega(k)$ is zero-mean white Gaussian with $\sigma_\omega = 2°$. The sensor noise is assumed independent of the sensor platform perturbations. The received measurements include clutter and false alarms. The clutter is modeled as uniform with average false alarm rate $\lambda_k = 10^{-4}$ $(\text{rad})^{-1}$ over the whole surveillance region $[0, \pi]$ rad.

For tracking multiple targets, an SPHD filter of order 3 is used with 20 knots for position and 10 knots for velocity. At scan $k = 0$, all measurements are used to initialize newborn targets as described in Section (4.2.2). The probability of target spawning is assumed to be zero and the probability of spontaneous target birth is 0.01.

Figures 4.2, 4.3 and 4.4 show the prior, predicted and posterior distributions of the SPHD filter at scan $k = 15$, respectively. As shown in these figures, the SPHD filter is capable of modeling any arbitrary density function of the system state without putting any specific constraint on system/measurement model. The two peaks correspond to the two targets at that time. Note that target 3 appears only at $k = 16$.

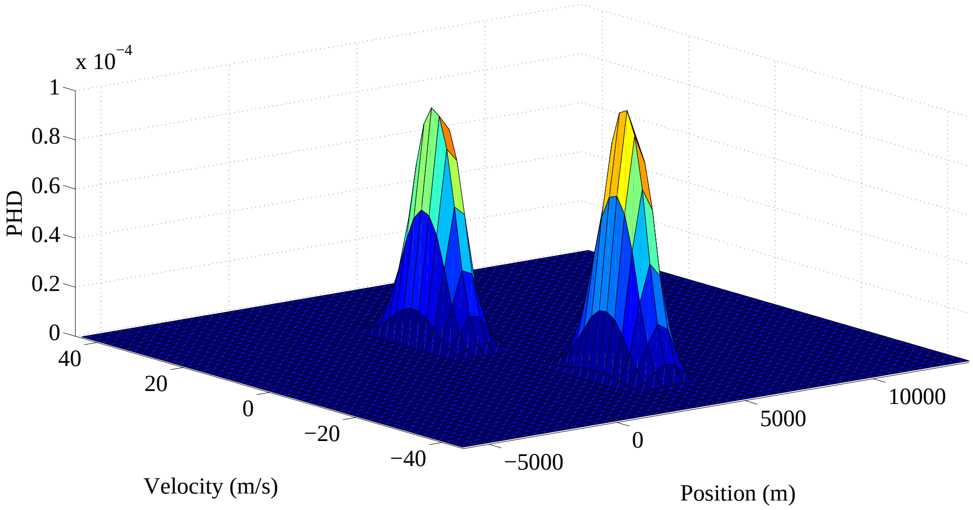

Figure 4.2: Prior PHD at $k = 15$ ($\sigma_\omega = 2°$).

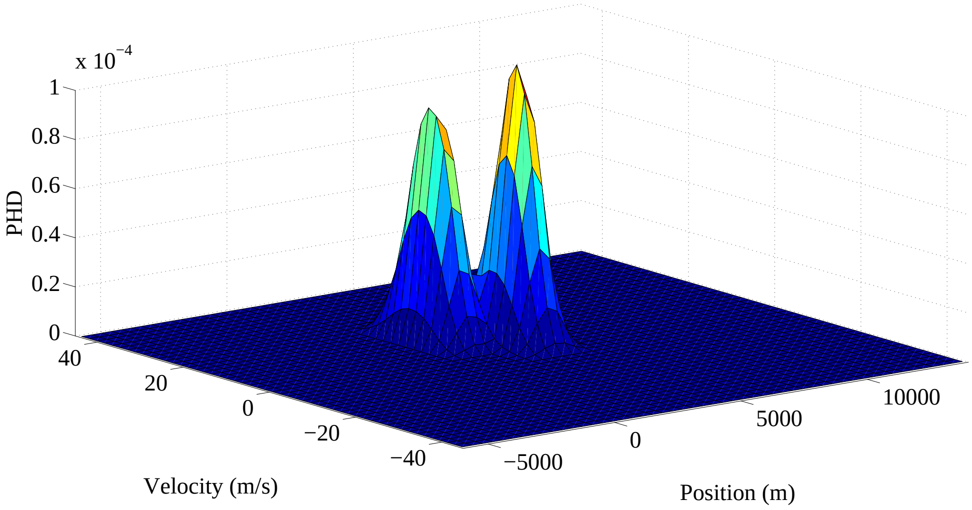

Figure 4.3: Predicted PHD at $k = 15$ ($\sigma_\omega = 2°$).

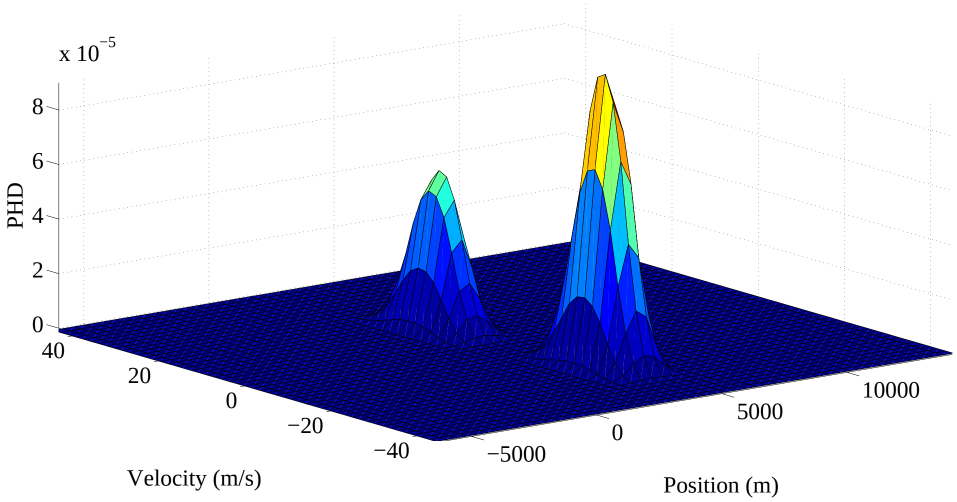

Figure 4.4: Posterior PHD at $k = 15$ ($\sigma_\omega = 2°$).

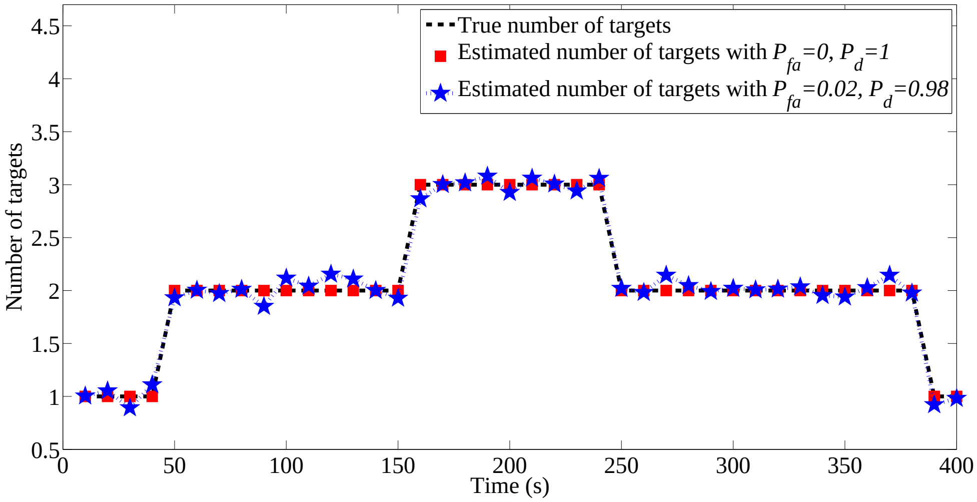

Figure 4.5: True vs. average of estimated number of targets from 1000 runs ($\sigma_\omega = 2°$).

The key feature of the SPHD filter is to find the number of targets at any time in any particular surveillance area. As shown in Figure 4.5, the SPHD filter accurately determines the number of targets for two different scenarios. In Figure 4.5, the P_{fa} denotes the probability of false alarms and P_d denotes the probability of detection.

The PHD filter does not provide a mechanism to get the target state estimates directly. One solution is to identify the local maxima of the SPHD surface. The K-means clustering algorithm [61] is used here for state extraction. An alternative is the expectation-maximization based peak extraction approach in [62]. The targets are associated to tracks using global nearest-neighbor assignment [47] based on the mean of each target cluster.

As shown in Figure 4.6, all three targets appear and disappear at various times during the surveillance interval. Also shown in Figure 4.6 are the average of the estimated trajectories. As shown in Figure 4.7, the SPHD filter estimated the velocities of all targets accurately. The mean velocity of newborn targets is selected randomly from a uniform distribution in the interval $[-40, 40]$ m/s and the standard deviation is assumed to be 0.10 m/s.

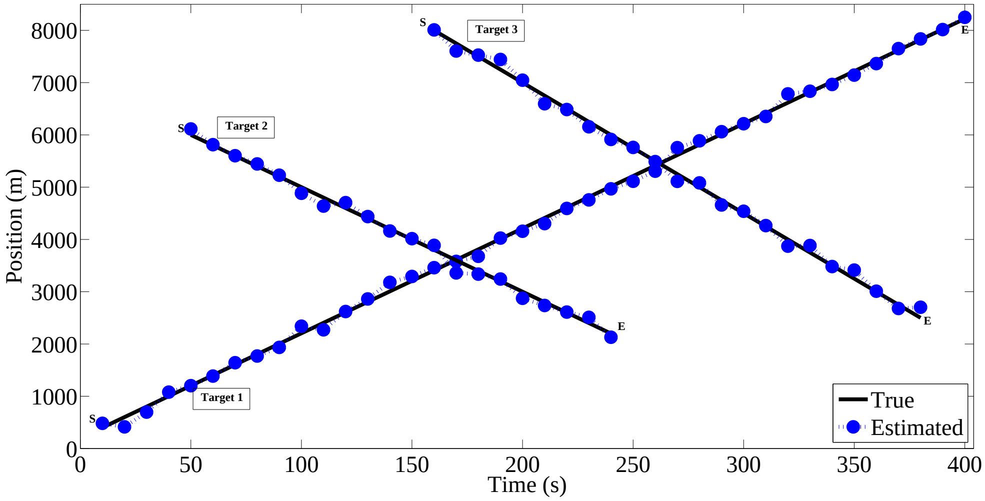

Figure 4.6: True position vs. average of estimated position from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

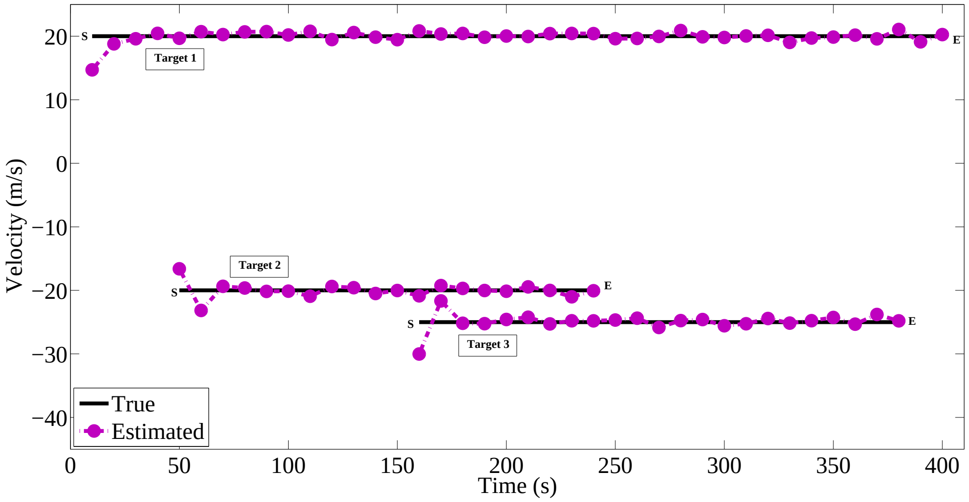

Figure 4.7: True velocity vs. average of estimated velocity from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

Next, the performance of the SPHD filter is analyzed with different numbers of knots. The number of targets is 6 and the initial states are

$$[x_1(k)\ x_2(k)\ x_3(k)\ x_4(k)\ x_5(k)\ x_6(k)] = \begin{bmatrix} 400 & 6000 & 9000 & 2000 & 5000 & 10000 \\ 20 & -20 & -25 & 40 & 25 & 0 \end{bmatrix} \tag{4.57}$$

while the start and end time steps are (1,40), (5,24), (1,38), (6,30), (15,40) and (1,40), respectively. As shown in Figures 4.8–4.10, the performance of the SPHD filter is not affected by the increase in the number of targets. As shown in Figure 4.8, the SPHD filter accurately determines the number of targets, and as shown in Figure 4.9, tracks corresponding to these 6 targets are correctly initialized and terminated to match the truth.

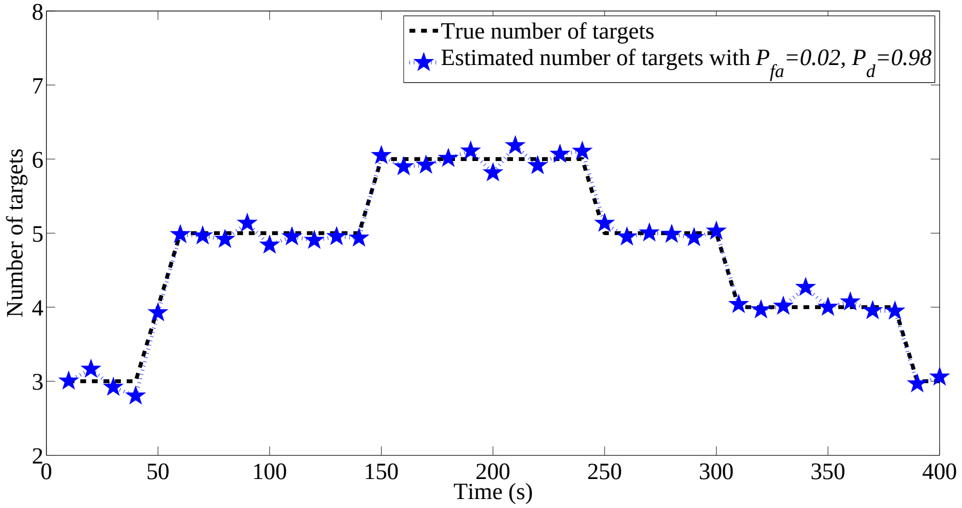

Figure 4.8: True vs. average of estimated number of targets from 1000 runs ($\sigma_\omega = 2°$).

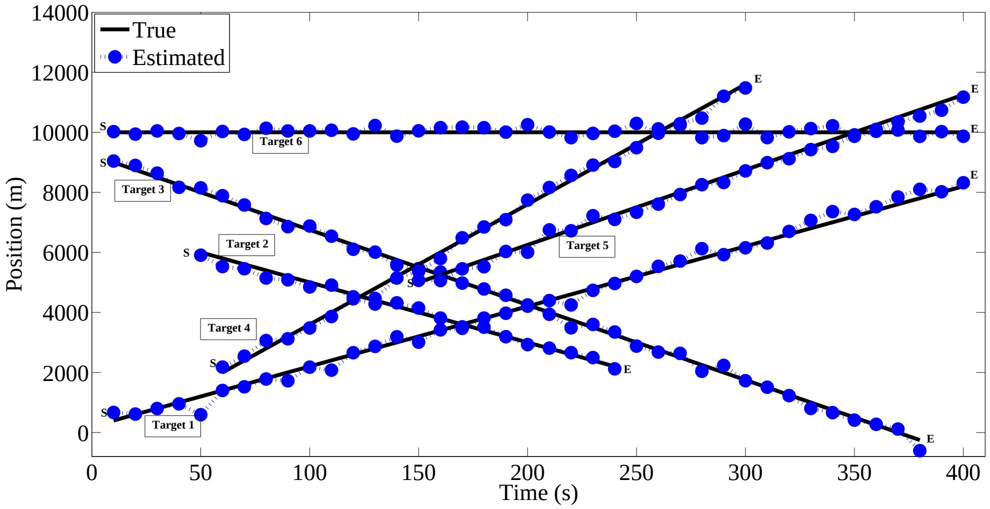

Figure 4.9: True position vs. average of estimated position from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

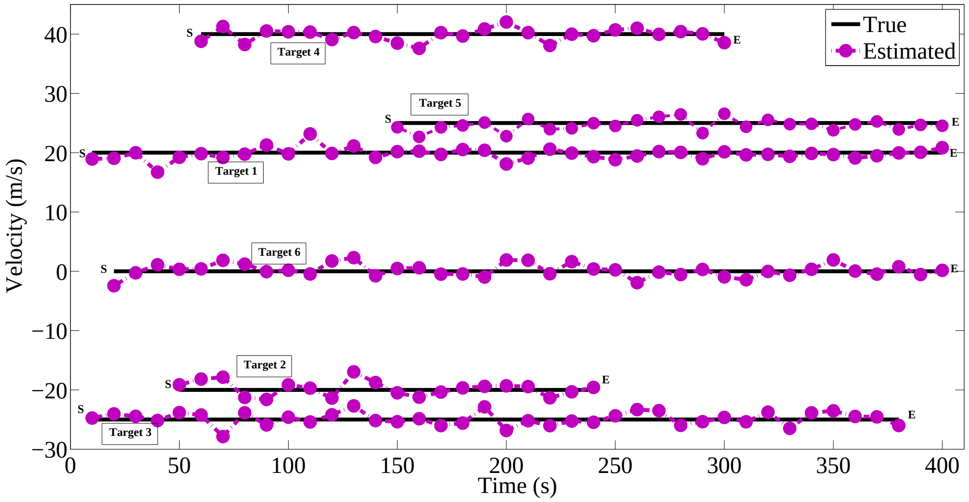

Figure 4.10: True velocity vs. average of estimated velocity from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

The selection of the number of knots depends on the application, especially on target separation. As shown in Table 4.1, increasing the number of knots affects the complexity of the SPHD filter, but results in better performance in terms of RMSE. Note that beyond $n_Pk = 50$ and $n_{Vk} = 20$, the performance of the SPHD filter does not improve much and, considering the computational time and the performance of the SPHD filter, one can note that the optimal knot selection is given by $n_Pk = 50$ and $n_{Vk} = 20$. Theoretically optimal selection of the number of knots is a topic for future research.

Number of knots	RMSE (m)	CPU time (sec)
$n_{Pk} = 10$, $n_{Vk} = 5$	594.05	5.700
$n_{Pk} = 20$, $n_{Vk} = 10$	340.02	7.800
$n_{Pk} = 40$, $n_{Vk} = 18$	310.58	10.190
$n_{Pk} = 50$, $n_{Vk} = 20$	260.87	13.475
$n_{Pk} = 60$, $n_{Vk} = 22$	259.10	15.190
$n_{Pk} = 70$, $n_{Vk} = 25$	258.59	16.090

Table 4.1: Average performance from 1000 runs vs. number of knots ($\sigma_\omega = 2°$), n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, number of targets = 6.

The complexity becomes a major problem for high dimensional multitarget tracking problems due to the use of the method from Section 5.2.1 to predict the knots. This step takes up 95% of the computation. As with other nonlinear filtering techniques the curse of dimensionality [66] is a challenge for the SPHD filter.

Next, the original example is analysed further with an increasing number of targets, but with fixed values for the number of position knots $n_{Pk} = 20$ and number of velocity knots $n_{Vk} = 10$. As shown in Table 4.2, increasing the number of targets for a given number of knots adversely affects the performance of the SPHD filter in terms of RMSE and computational time. Note that with optimal knot selection, which is beyond the scope of this Chapter, the SPHD filter performance can be improved but with higher complexity.

Number of targets	RMSE (m)	CPU time (sec)
5	345	7.775
10	596	8.500
15	739	9.125
20	1225	10.250
40	2253	12.350

Table 4.2: Average performance from 1000 runs vs. number of targets ($\sigma_\omega = 2°$), $n_{Pk} = 20$, $n_{Vk} = 10$ (fixed number of knots).

The position and velocity Root Mean Squared Error (RMSE), Normalized Estimation Error Squared (NEES) [3], CPU time, Optimal Sub-pattern Assignment (OSPA) [55], the number of false tracks [6] and track continuity [16] are used as performance metrics for the original example. The performance metric values are averaged over 1000 Monte Carlo runs.

The OSPA [55] metric measures the miss-distance between a set of true targets and a set of estimated tracks as a combination of localisation error and cardinality error [21]. Let $X = \{x_1, \cdots, x_{\acute{n}}\}$ and $Y = \{y_1, \cdots, y_{\acute{m}}\}$ be two finite sets. Then X denotes the set of true targets and Y denotes the set of estimated tracks. The OSPA metric is defined as

$$\bar{d}_{\acute{p}}^{(\acute{c})}(X,Y) = \begin{cases} 0 & \text{if } \acute{m} = \acute{n} = 0 \\ \Psi(X,Y) & \text{if } \acute{m} \leq \acute{n}. \\ d^{(\acute{c})}(X,Y) & \text{if } \acute{m} > \acute{n} \end{cases} \tag{4.58}$$

where

$$\Psi(X,Y) \triangleq \left(\frac{1}{\acute{n}} \left(\min_{\pi \in \Pi_{\acute{n}}} \sum_{i=1}^{\acute{m}} d^{(\acute{c})}(x_i, y_{\pi(i)}) + \acute{c}^{\acute{p}}(\acute{n} - \acute{m}) \right) \right)^{\frac{1}{\acute{p}}} \tag{4.59}$$

and the base distance between x and y is denoted by $d^{\acute{c}}(x,y) = \min{(\acute{c}, \|x-y\|)}$. Also, $\Pi_{\acute{n}}$ is the set of permutations with length $\acute{m}$ on the set of $\{1, \cdots, \acute{n}\}$ where $\acute{n} = \|X\|$ and $\acute{m} = \|Y\|$. In our simulations $\acute{p} = 10$ and $\acute{c} = 100$.

The performance of the SPHD filter is evaluated along with those of SMC-PHD, GM-PHD, GMP-PHD, GM-USMC-PHD, GM-SMC-PHD and AP-PHD filters.

The SMC-PHD filter uses the transition density to sample particles. Particles are initialized around the measurement [18] and 2500 particles are used per existing track and 50 particles are used for each new track. An estimate of the number of targets is determined by summing up the weights of the particles. The states are extracted using clustering [18].

The GM-PHD filter was implemented with the EKF, elimination threshold $T_p = 10^{-5}$, merging threshold $T_m = 4$m and maximum number of Gaussian terms = 200. An estimate of the number of targets is given by the sum of the weights of the mixture, and a Gaussian component is considered as target-originated when its weight is above 0.5.

The particle implementation of the GMP-PHD filter is based on the Generic Particle Filter (GPF) [1] and the resampling procedure is activated when the effective sample size [1, 18] N_{eff} is less than 80%. The GMP-PHD filter runs with 2500 particles using the predicted mixture components as proposal distributions. To mitigate the exponential growth of mixture components, at each time step the number of Gaussian components is limited to 100 components. The pruning and merging thresholds are the same as in the GM-PHD filter implementation.

In the GM-USMC-PHD filter implementation, the IS function is approximated in the form of a Gaussian mixture. The GM implementation of the GM-USMC-PHD filter is similar to the GM-PHD filter one, but with only 100 components. The number of samples per GM component is set to 2500. The newborn target initialization, resampling and state extraction steps follow [73]. The Unscented Information Filter (UIF) [3] is used to compute the mean and the covariance of Gaussian.

The GM implementation of the GM-SMC-PHD filter is similar to the GM-PHD filter one, but with only 100 components and the SMC implementation of the GM-SMC-PHD is similar to the SMC-PHD filter one. The estimation of the number of targets and state extraction are done as in [50].

The AP-PHD filter uses 2500 particles per existing track, while the number particles per new track is 100. The initialization of the new tracks is driven by the measurements. Each current measurement is associated with the corresponding highest bidder if the bid is greater than 0.4. The auxiliary importance sampling (AIS) [4] process starts with the selection of the measurements that are well described by the targets' states extracted from the estimated PHD and this is achieved using the Auction algorithm [4]. The state extraction is done as in [4].

The RMSE vs. CPU time for low measurement noise standard deviations (0.02° and 2°) with 1000 Monte Carlo runs for the SPHD filter, SMC-PHD filter, GM-PHD filter, GMP-PHD filter, GM-USMC-PHD filter, GM-SMC-PHD filter and the AP-PHD filter are presented in Figure 4.11.

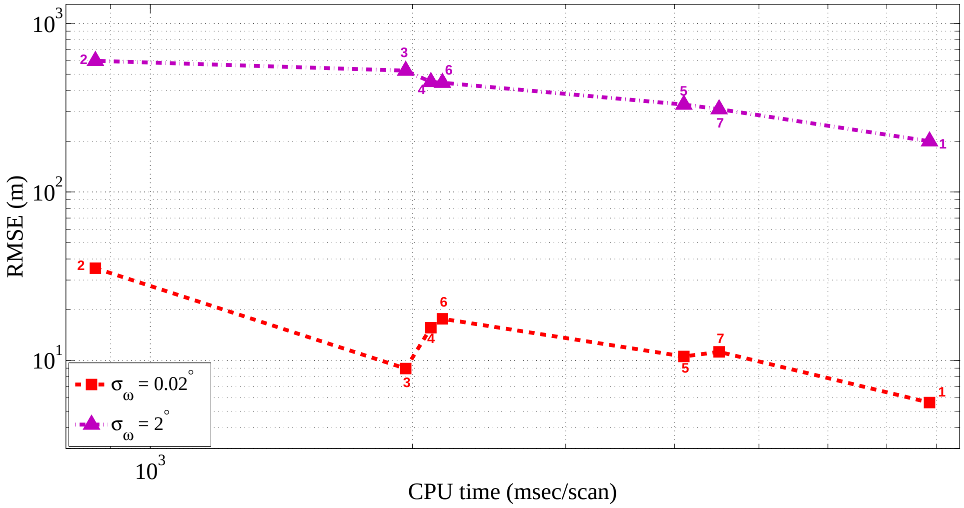

Figure 4.11: RMSE vs. CPU time from 1000 Monte Carlo runs, SPHD: 1, SMC-PHD: 2, GM-PHD: 3, GMP-PHD: 4, GM-USMC-PHD: 5, GM-SMC-PHD: 6, AP-PHD: 7 ($\sigma_\omega = 0.02°, 2°$).

As shown in Figure 4.11, for $\sigma_\omega = 2°$ the SPHD filter of order 3 with 10 velocity knots and 20 position knots provides more accurate results than the other filters, but with increased CPU time. The SMC-PHD filter has the lowest complexity, but with poor performance. The GM-USMC-PHD filter performance can be compared to that of the AP-PHD filter, but the latter has a higher complexity. The GM-SMC-PHD filter and the GMP-PHD filter have the same performance and the GM-PHD filter's performance is better than that of the SMC-PHD filter, but with a higher complexity.

Figure 4.12 reveals the consistency of the SPHD filter, SMC-PHD filter, GM-PHD filter, GMP-PHD filter, GM-USMC-PHD filter, GM-SMC-PHD filter, and the

AP-PHD filter, in terms of normalized estimation error squared (NEES) compared with the 95% confident-region of the $\mathcal{X}^2$ distribution [3] when the measurement noise standard deviation is $\sigma_\omega = 2°$.

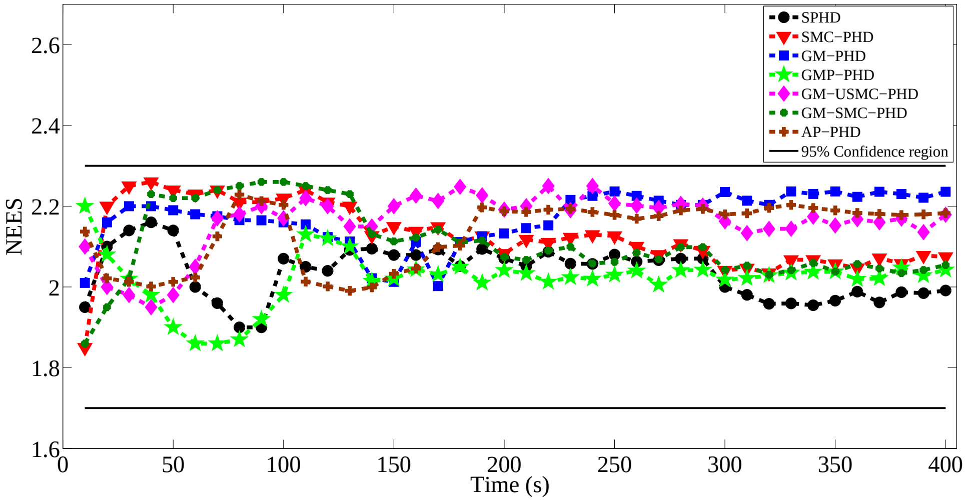

Figure 4.12: NEES comparison from 1000 Monte Carlo runs ($\sigma_\omega = 2°$).

To illustrate the degeneracy resistance capability of the SPHD filter, the standard deviation of the measurement noise is reduced to σ_ω=0.02°, but the parameters for the filters remain unchanged (with the correct measurement noise level). This scenario causes the variation of the particle weights to increase rapidly for the particle based PHD filters (i.e., they become degenerative). It can be observed from Figure 4.13 that all particle based filters are affected by the reduction in measurement noise, but the SPHD filter is able to provide efficient results with the same 10 velocity knots and 20 position knots. The GM-PHD filter performed better than the particle based filters at this low measurement noise level.

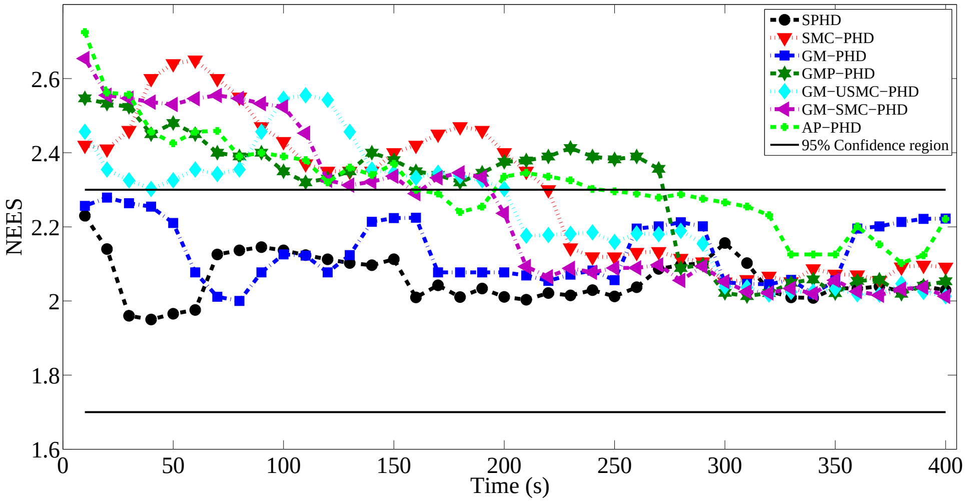

Figure 4.13: NEES comparison from 1000 Monte Carlo runs ($\sigma_\omega = 0.02°$).

Next, the advantage of the proposed SPHD filter over other methods at higher levels of measurement noise ($\sigma_\omega = 4°, 8°$) is illustrated. As shown in Figure 4.14, the SPHD filter outperforms the rest of the filters for higher measurement noise level. As shown in Figure 4.14, the AP-PHD filter performed better than the GM-USMC-PHD filter at higher measurement noise level, but the AP-PHD filter has a higher complexity. Though the process noise and measurement noise components are Gaussian, the updated posterior distribution is non-Gaussian because of the nonlinear measurement equation. As the nonlinearity increases with increasing measurement noise levels, GM-based PHD filters performed poorly. Note that the GMP-PHD, GM-USMC-PHD, and the GM-SMC-PHD filters can handle nonlinear system and measurement equations better with their particle filter implementation. But the GM approximations of the filters are based on the Gaussian assumption, which leads to poor performance at high measurement noise levels.

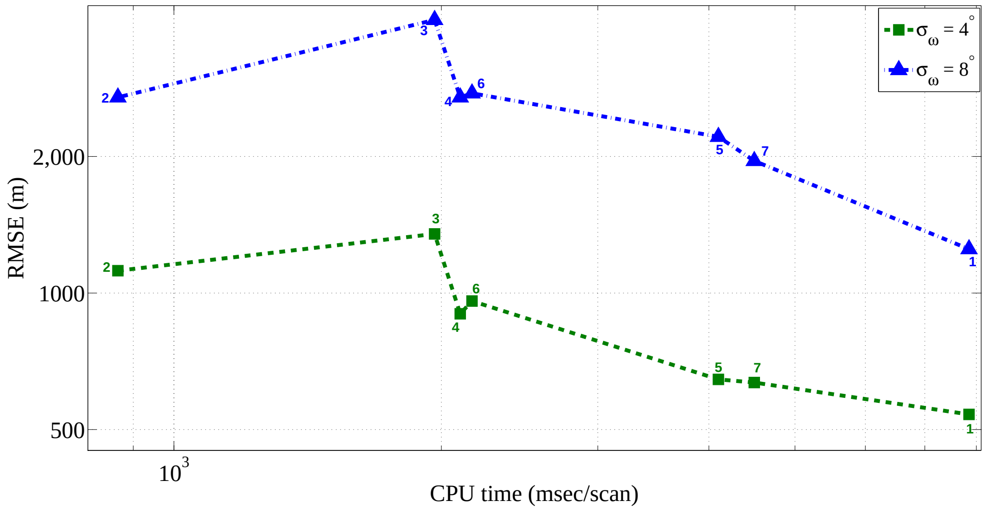

Figure 4.14: RMSE vs. CPU time from 1000 Monte Carlo runs, SPHD: 1, SMC-PHD: 2, GM-PHD: 3, GMP-PHD: 4, GM-USMC-PHD: 5, GM-SMC-PHD: 6, AP-PHD: 7 ($\sigma_\omega = 4°, 8°$).

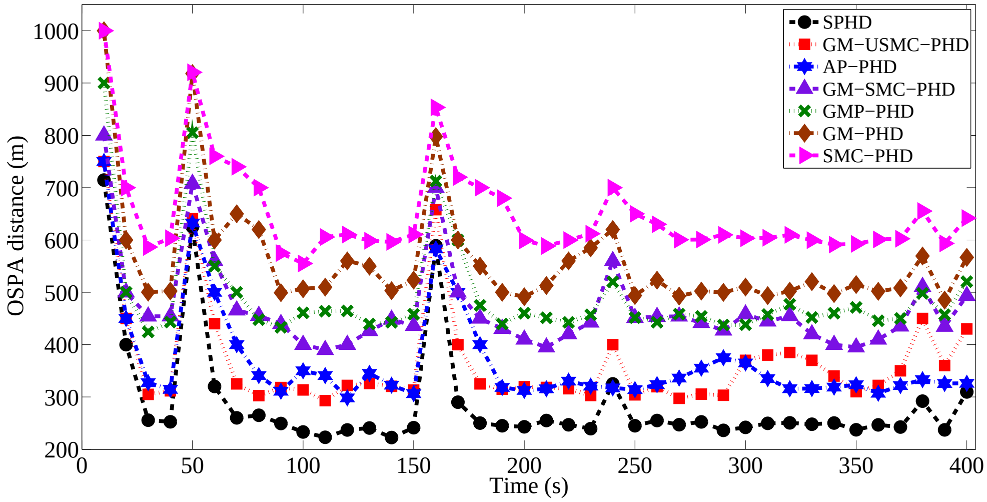

Figure 4.15: OSPA distance (m) averaged over 1000 Monte Carlo runs ($\sigma_\omega = 2.0°$, $\acute{c} = 10$, $\acute{p} = 100$).

Next, we present the OSPA metric [55], number of false tracks [6] and track continuity [16] to evaluate the performance of the SPHD filter relative to those of the other filters.

Filters	OSPA (m)	Number of false tracks	Track continuity (%)	CPU time (sec)
SPHD	240	0.0125	93.65	7.850
SMC-PHD	600	0.395	70.15	0.975
GM-PHD	505	0.205	76.69	1.965
GMP-PHD	450	0.178	80.25	2.165
GM-USMC-PHD	380	0.0926	88.25	4.120
GM-SMC-PHD	445	0.125	80.45	2.100
AP-PHD	340	0.0323	90.14	4.320

Table 4.3: Average performance metrics from 1000 Monte Carlo run ($\sigma_\omega = 2°$).

As shown in Table 4.3, the SPHD filter with a few knots performs the best of in terms of OSPA. The OSPA [55] metric measures the combination of both localization and cardinality distance. As shown in Figure 4.15, high values of OSPA distance occur when new targets are born around time index $k = 1, 5, 16$. Also, targets disappear with small OSPA peaks at time index $k = 24, 38, 40$.

As shown in Table 4.3, the average number of false tracks in the SPHD filter is lower than that of any other filter. The SPHD filter with a few knots has the lowest average number of false tracks due to the dynamic movement of its knots, where knots are dynamically (and automatically) moved to ensure that the SPHD filter covers the region where the posterior SPHD of system state is significant. The GM-PHD, GMP-PHD and GM-SMC-PHD filters have higher false track numbers due to the use of the EKF. The track continuity presented in Table 4.3 shows that the SPHD filter has the highest track continuity over all other filters, but the AP-PHD filter's performance is comparable to that of the SPHD filter in terms of track continuity.

The computational complexity of the SPHD filter is high. However, in critical surveillance problems where false tracks cannot be tolerated and high track continuity is essential (e.g., high-value ground target tracking, ballistic missile tracking), the spline filter is an appropriate option.

By moving the knots dynamically, the SPHD filter ensures it covers only the region where the posterior SPHD of the system state is significant so that the high computational efficiency of the SPHD filter is maintained at all times. Figures 4.16 and 4.17 from scan 4 illustrate the importance of knot selection. At scan 4 the true position and the estimated positions with and without the knot dynamic movement are calculated as 1001.4 m, 999.74 m and 945.5 m, respectively. As shown in Figure 4.17, with automatic knot movement, state extraction of well-separated as well as closely-spaced targets becomes more accurate. This is due to the concentration of knots in areas of high posterior SPHD.

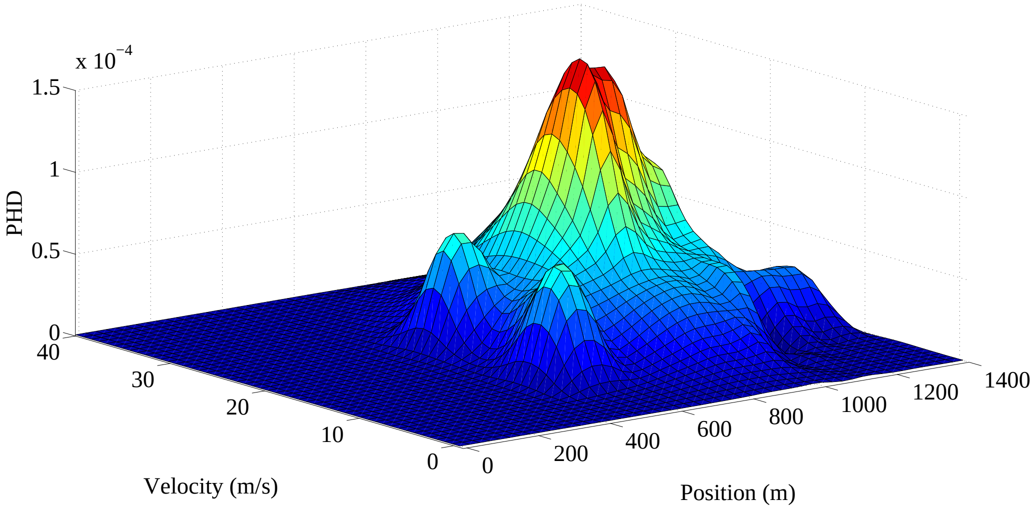

Figure 4.16: Posterior PHD (without dynamic knot movement) ($\sigma_\omega = 2.0°$).

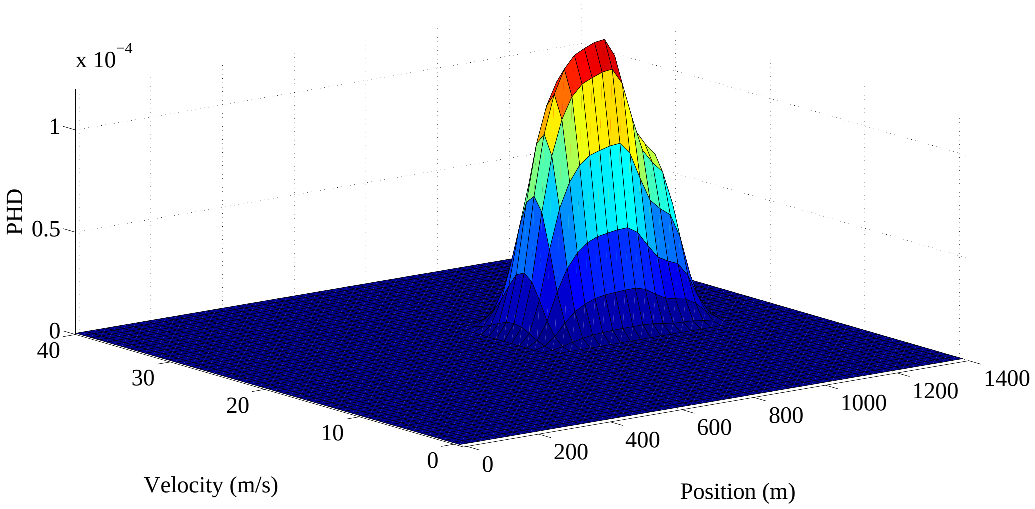

Figure 4.17: Posterior PHD (with dynamic knot movement) ($\sigma_\omega = 2.0°$).

The convergence of the SPHD filter depends on the number of knots used in a knot sequence to represent a variable as well as the distance between first and last knots. One of the most important conditions to ensure a convergent solution for tracking is to have the spline multitarget probability density equations to comply with the Schoenberg-Whitney Theorem in Section 3.7.9 and this can be achieved by selecting the number of knots to be greater than the number of control points. The number of knots can be selected depending on the region, desired state estimation accuracy and the number of targets. Note that increasing the number of knots always affects the complexity. Moreover, knots can be inserted into an existing knot vector by using the knot insertion property discussed in Section 3.7.7. The other indicator for convergence is determined by the distance between the first and the last knots, where the difference between these knots has to be greater than zero to have a convergent solution. Otherwise, the denominator of (3.3) becomes zero, which leads to non-convergent solutions. This is not an issue in tracking problems with a non-zero surveillance volume.

Chapter 5

Spline Probability Hypothesis Density Filter for Nonlinear Maneuvering Target Tracking

5.1 Introduction

The Probability Hypothesis Density (PHD) filter is an efficient algorithm for multitarget tracking in the presence of nonlinearities and/or non-Gaussian noise. The Sequential Monte Carlo (SMC) and Gaussian Mixture (GM) techniques are commonly used to implement the PHD filter. Recently, a new implementation of the PHD filter using B-splines with the capability to model any arbitrary density functions using only a few knots was proposed. The Spline PHD (SPHD) filter was found to be more robust than the SMC-PHD filter since it does not suffer from degeneracy and it was better than the GM-PHD implementation in terms of estimation accuracy, albeit with a higher computational complexity. In this chapter, we propose a Multiple Model (MM) extension to the SPHD filter to track multiple maneuvering targets. Simulation results are presented to demonstrate the effectiveness of the new filter.

5.2 MM-SPHD filtering

The proposed MM-SPHD filter implementation is based on the SPHD filter's extension to multiple model estimation. This section derives the MM-SPHD filter for the multidimensional multitarget state space models. Assume that a multidimensional multitarget system state at time k is denoted as $X_k = \{\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{\vartheta_k,k}\}$ where each target has multidimensional state $\mathbf{x}_k = [\mathbf{x}_k^1, \ldots, \mathbf{x}_k^n]'$ and n denotes the number of dimensions.

5.2.1 MM-SPHD mixing

The MM-SPHD filter derivations follow Section 2.3.2. Let the initial MM-SPHD be

$$\begin{aligned}\tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) &= \sum_{i_1}^{n_s} \ldots \sum_{i_n}^{n_s} \mathbb{P}_{i_1,\ldots,i_n} \\ &\cdot B_{i_1,p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\ &\cdot B_{i_n,p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_k = \mathfrak{q}) \qquad (5.1)\end{aligned}$$

where $\tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$ denotes the $\mathfrak{q}$-th mode-dependent initial multitarget multidimensional MM-SPHD and $\mathbf{M}_k \in \{1, \ldots, \mathsf{r}\}$ is the model index at time k, where r denotes the total number of models. The number of dimensions is denoted by n and $\mathfrak{i} = i_1, \ldots, i_n$. The number of knots for all dimensions is the same at τ.

The n dimensional knot $\mathbf{t}_{k-1} = \{\mathbf{t}^1_{k-1}, \ldots, \mathbf{t}^n_{k-1}\}$, is an $n \times \tau$ array. Each row vector of $\mathbf{t}_{k-1}$ consists of a set of prior knots $\mathbf{t}^l_{k-1} = \{t^l_{1,k-1}, \ldots, t^l_{\tau,k-1}\}$ where $l = 1, \ldots, n$. Assume that the l-th dimensional prior knot $\mathbf{t}^l_{k-1}$ represents a parametric curve in the range $[a, b]$ of order p (degree$= p-1$) with n_s input/control points. Then, the knot vector will have $\tau = n_s + p$ elements. The knot elements can be determined using the uniform knot selection method as [17]

$$\mathbf{t}^l_{k-1} \;=\; \begin{cases} t^l_{1,k-1}, \ldots, t^l_{p,k-1} = a \\ t^l_{i+p,k-1} = a + \frac{i(b-a)}{n_s+p-1} \quad \text{for} \quad i = 1, \ldots, (n_s - p) \\ t^l_{\tau-p,k-1}, \ldots, t^l_{\tau,k-1} = b \end{cases} \tag{5.2}$$

Alternatively, the knot elements can be created using average knot selection by averaging the parameter values $(x_i,\ i = 1, \ldots, (n_s - p))$ in their neighborhood given parametrization $\mathbf{x}^l_k = x_1, \ldots, x_{n_s}$ as [17]

$$\mathbf{t}^l_{k-1} \;=\; \begin{cases} t^l_{1,k-1}, \ldots, t^l_{p,k-1} = a \\ t^l_{i+p,k-1} = \frac{1}{p}\sum_{j=i}^{i+p-1} x_j \quad \text{for} \quad i = 1, \ldots, (n_s - p) \\ t^l_{\tau-p,k-1}, \ldots, t^l_{\tau,k-1} = b \end{cases} \tag{5.3}$$

or one can use the optimal knot selection, which is an iterative method proposed in [17]. This method does not consider the location of the input/control points for the optimization. The Matlab function *optknt* can be used to produce this knot vector.

The n dimensional control point set or coefficient matrix is denoted by $\mathbb{P}_{\mathrm{i}}$ and n_s denotes the number of control points. The number of control points for all dimensions is the same. Note that the number of knots must be greater than the number of control points.

The initial MM-SPHD $\tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$ matched to multitarget model $\mathfrak{q}$, is send to the MM-SPHD filter. The initial MM-SPHD $\tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$ can be calculated on the basis of the Markovian model transition probability matrix $\pi_{\mathfrak{pq}}$ and model-dependent multitarget multidimensional prior MM-SPHD $\mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p})$. That is,

$$\begin{aligned} \tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \quad &= \quad \sum_{\mathfrak{p}=1}^{\mathsf{r}} \mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \\ &\quad \cdot \pi_{\mathfrak{pq}} \qquad \mathfrak{q} \in \{1, \dots, \mathsf{r}\} \end{aligned} \tag{5.4}$$

where the prior MM-SPHD of the $\mathfrak{p}$-th dynamic system can be determined as

$$\begin{aligned} \mathbf{B}_{k-1|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \quad &= \quad \sum_{g_1}^{n_s} \dots \sum_{g_n}^{n_s} \mathbb{P}_{g_1,\dots,g_n} \\ &\quad \cdot B_{g_1,p,\mathbf{t}^1_{k-1}}(\mathbf{x}^1_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \dots \\ &\quad \cdot B_{g_n,p,\mathbf{t}^n_{k-1}}(\mathbf{x}^n_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \end{aligned} \tag{5.5}$$

and $\mathfrak{p} = \{1, \dots, \mathsf{r}\}$. For all r system models, the prior MM-SPHD $\mathbf{B}_{k-1|k-1}(\cdot)$ are summed together with scaling by the corresponding mode probability $\pi_{\mathfrak{pq}}$ to determine the initial MM-SPHD $\tilde{\mathbf{B}}_{k|k-1}(.)$ as in (5.4). The prior number of expected targets is the integral of $\tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q})$ over the region of state space $\mathcal{E}_s$ for the $\mathfrak{q}$-th model evaluated as

$$\hat{N}^{\mathfrak{q}}_{k-1|k-1} = \int_{\mathcal{E}_s} \tilde{\mathbf{B}}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) d\mathbf{x}_{k-1} \tag{5.6}$$

where $\mathfrak{q} = \{1, \ldots, \mathsf{r}\}$. Using (5.7), the prior number of expected targets can be determined for all the models. Then the overall prior number of expected targets can be determined as

$$\hat{N}_{k-1|k-1} = \sum_{\mathfrak{q}=1}^{\mathsf{r}} \hat{N}^{\mathfrak{q}}_{k-1|k-1} \tag{5.7}$$

The target can spawn, die or born and these events only considered at the prediction stage.

5.2.2 MM-SPHD prediction

The spline representation of the mode-dependent multitarget state transition density $p_{k|k-1}$ is a $2n$ dimensional function determined using system model (2.9) as

$$\begin{aligned} p_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q} | \mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \quad = \quad & \sum_{j_1} \cdots \sum_{j_{2n}} \mathbb{P}_{j_1, \ldots, j_{2n}} \\ & \cdot B_{j_1, \mathbf{t}_k^1}(\mathbf{x}_k^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\ & \cdot B_{j_n, \mathbf{t}_k^n}(\mathbf{x}_k^n, \mathbf{M}_k = \mathfrak{q}) \\ & \cdot B_{j_{n+1}, p, \mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_{k-1} = \mathfrak{p}) \ldots \\ & \cdot B_{j_{2n}, p, \mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_{k-1} = \mathfrak{p}) \end{aligned} \tag{5.8}$$

where $\mathfrak{q}, \mathfrak{p} \in \{1, \ldots, \mathsf{r}\}$, $\mathfrak{j} = \{j_1, \ldots, j_{2n}\}$ and $\mathbf{t}_k$ denotes an $n \times \tau$ knot array at time k and it consists of row vectors $\mathbf{t}_k^1, \ldots, \mathbf{t}_k^n$. Each row vector of $\mathbf{t}_k$ consists of a set of predicted knots $\mathbf{t}_k^l = \{t_{1,k}^l, \ldots, t_{\tau,k}^l\}$ where $l = 1, \ldots, n$.

The predicted knot selection of a multidimensional system is much more challenging. A suboptimal but computationally efficient method is used to find the predicted knots for the multidimensional spline. At time $k-1$, from (5.7) one can determine the mode-dependent prior number of targets and their states by using the K-means clustering algorithm [61]. An alternative is the Expectation-Maximization (EM) based peak extraction approach [62]. For example, assume that there are ϑ_{k-1} targets at $k-1$ and their states are

$$\mathbf{x}_{h_{k-1},k-1} \quad = \quad [\mathbf{x}^1_{h_{k-1},k-1}, \ldots, \mathbf{x}^n_{h_{k-1},k-1}]' \qquad h_{k-1} \in 1, 2, \cdots, \vartheta_{k-1} \tag{5.9}$$

Note that there are $\tau = \tau_l$ knot elements in the l-th dimension knot vector $\mathbf{t}^l_{k-1}$. Each dimension has τ number of elements in the corresponding knot vector. Next, we can find the l-th dimensional prior knot elements that approximate the l-th dimensional state parameters as

$$t^l_{\zeta_{h_{k-1}},k-1} \quad \approx \quad \mathbf{x}^l_{h_{k-1},k-1} \qquad h_{k-1} \in 1, 2, \cdots, \vartheta_{k-1} \tag{5.10}$$

where $\zeta_{h_{k-1}} \in (1, \ldots, \tau)$. Using the locations of $\zeta_1, \cdots, \zeta_{\vartheta_{k-1}}$ one can interactively shape the B-spline curve: knot elements can be added to $\mathbf{t}^l_{k-1}$ as in Section (3.7.7) to increase the number of control points that can be modified. When control points are moved, the level of continuity at the knots can increase or decrease. Knot removal as in Section (3.7.8) can be invoked in order to obtain the most compact representation of the curve. Knot removal can also be used to remove unnecessary knots. Note that the first and the last knot elements are never removed. This method can be used for all dimensions. Then, $\prod\limits_l \tau_l$ different sample vectors $\mathbf{\Psi}_{k-1,t}$ can be formed by selecting one knot $t^l_{e_l,k-1}$ from the set of knots in each dimension $\mathbf{t}^l_{k-1}$ and $e_l = 1, \ldots, \tau$. For example, the collection of all such sample vectors in the n-dimensional spline filter is

$\boldsymbol{\Psi}_{k-1,t} = \{[t^l_{e_1,k-1} \dots t^l_{e_n,k-1}]'\}_{e_1,\dots,e_n}$ and the total number of $\boldsymbol{\Psi}_{k-1,t}$ is $\tau_1 \times \dots \times \tau_n$. Then, the predicted sample knots $\boldsymbol{\Psi}_{k,t}$ are found by

$$\boldsymbol{\Psi}^{e_1,\dots,e_n}_{k,t} \quad = \quad f_{k,\mathbf{M}_k}(\boldsymbol{\Psi}^{e_1,\dots,e_n}_{k-1,t}, 0, \mathbf{M}_k = \mathfrak{q}) \tag{5.11}$$

where the $f_{k,\mathbf{M}_k}(\cdot,\cdot,\cdot)$ is the mode-dependent function of transition in (2.9). Then, to select the predicted knots in each dimension l, first project all the sample vectors $\boldsymbol{\Psi}^{e_1,\dots,e_n}_{k,t}$ into the axis of the l-th dimension. In the n dimensional case, the projection will result in $\tau_1 \times \cdots \times \tau_n$ possibly overlapping points in each axis. Denote them as

$$\{t^l_{1',k}, t^l_{2',k}, \dots, t^l_{(\tau_1 \times \dots \times \tau_n)',k}\} \tag{5.12}$$

and assume that they are sorted in a non-decreasing order.

Using (5.10) and known prior number of targets and their states, interactively shaped B-spline curve's l-th knot vector can be written as

$$\mathbf{t}^l_{k-1} = \{t^l_{1,k-1}, \dots, t^l_{\zeta_{h_{k-1}},k-1}, \dots, t^l_{\tau,k-1}\} \tag{5.13}$$

where $h_{k-1} \in 1, 2, \cdots, \vartheta_{k-1}$. The elements of knot vector $\mathbf{t}^l_{k-1}$ are in non-decreasing order and (5.13) can be written for n dimensional knot $\mathbf{t}_{k-1} = \{\mathbf{t}^1_{k-1}, \dots, \mathbf{t}^n_{k-1}\}$. Next, extract ϑ_{k-1} target states as

$$\mathfrak{x}_{l,k-1} \quad = \quad \mathfrak{x}_{h_{k-1},k-1} = [t^1_{\zeta_{h_{k-1}},k-1}, \dots, t^n_{\zeta_{h_{k-1}},k-1}] \tag{5.14}$$

where $l = h_{k-1} \in (1, 2, \cdots, \vartheta_{k-1})$ and the predicted states of the targets at time k as

$$\mathfrak{x}_{l,k} = f_{k,\mathbf{M}_k}(\mathfrak{x}_{l,k-1}, 0, \mathbf{M}_k) \quad l \in 1, 2, \ldots, \vartheta_{k-1} \tag{5.15}$$

Then, using (5.12) one can select the knots. The first and the last predicted knots are selected as $t^l_{1,k} = t^l_{1',k}$ and $t^l_{\tau_l,k} = t^l_{(\tau_1 \times \ldots \times \tau_n)',k}$, respectively. The rest of the knots can be selected close to the l-th dimensional predicted states $\mathfrak{x}_{l,k}$, where $l \in 1, 2, \ldots, \vartheta_{k-1}$. Knot removal can be invoked to remove unnecessary knots, but the first and the last knot elements are never removed. This method can apply for all the dimensions. Using this method of determining knots ensures that spline covers the region where the PHD of system state is significant. The mode-dependent coefficients or control points of spline transition density $\mathbb{P}_{j_1,\ldots,j_{2n}}$ can be determined as described in [58].

The mode-dependent spline predicted density can be calculated using (2.35) as

$$\begin{aligned} \mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &+ \mathbf{B}_{\mathbf{s},k|k-1}(X_k, \mathbf{M}_k = \mathfrak{q}) \\ &+ \mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \end{aligned} \tag{5.16}$$

The predicted MM-SPHD for the existing targets can be determined as [49]

$$\begin{aligned} D_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \\ &\cdot p_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|\mathbf{x}_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \\ &\cdot \tilde{D}_{k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}|Z^{(k-1)}) \\ &\cdot d\mathbf{x}_{k-1} \end{aligned} \tag{5.17}$$

and the spline predicted MM-SPHD for the existing targets as

$$\begin{aligned}
\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \sum_{j_1}^{n_s} \ldots \sum_{j_{2n}}^{n_s} \mathbb{P}_{j_1,\ldots,j_{2n}} \\
&\quad \cdot B_{j_1,p,\mathbf{t}_k^1}(\mathbf{x}_k^1, \mathbf{M}_k = \mathfrak{q}) \ldots B_{j_n,p,\mathbf{t}_k^n}(\mathbf{x}_k^n, \mathbf{M}_k = \mathfrak{q}) \\
&\quad \cdot B_{j_{n+1},p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_{k-1} = \mathfrak{p}) \ldots \\
&\quad \cdot B_{j_{2n},p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_{k-1} = \mathfrak{p}) \sum_{i_1}^{n_s} \ldots \sum_{i_n}^{n_s} \mathbb{P}_{i_1,\ldots,i_n} \\
&\quad \cdot B_{i_1,p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\
&\quad \cdot B_{i_n,p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_k = \mathfrak{q}) \; d\mathbf{x}_{k-1}^1 \ldots d\mathbf{x}_{k-1}^n \\
&= \sum_{j_1}^{n_s} \ldots \sum_{j_{2n}}^{n_s} \mathbb{P}_{j_1,\ldots,j_{2n}} B_{j_1,p,\mathbf{t}_k^1}(\mathbf{x}_k^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\
&\quad \cdot B_{j_n,p,\mathbf{t}_k^n}(\mathbf{x}_k^n, \mathbf{M}_k = \mathfrak{q}) \sum_{i_1}^{n_s} \ldots \sum_{i_n}^{n_s} \mathbb{P}_{i_1,\ldots,i_n} \\
&\quad \cdot \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \\
&\quad \cdot B_{j_{n+1},p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_{k-1} = \mathfrak{p}) \ldots \\
&\quad \cdot B_{j_{2n},p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_{k-1} = \mathfrak{p}) \\
&\quad \cdot B_{i_1,p,\mathbf{t}_{k-1}^1}(\mathbf{x}_{k-1}^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\
&\quad \cdot B_{i_n,p,\mathbf{t}_{k-1}^n}(\mathbf{x}_{k-1}^n, \mathbf{M}_k = \mathfrak{q}) \\
&\quad \cdot d\mathbf{x}_{k-1}^1 \ldots d\mathbf{x}_{k-1}^n \qquad (5.18)
\end{aligned}$$

where the third equality follows from the property that the order of summation and integration of splines is interchangeable [12].

Define two $2n$ dimensional matrices W and C, and one n dimensional matrix ξ as follows:

$$\begin{aligned}
W_{j_{n+1},\dots,j_{2n},i_1,\dots,i_n} &= \int P_{\mathfrak{s},k|k-1}(\mathbf{x}_{k-1}, \mathbf{M}_k = \mathfrak{q}) \\
&\quad \cdot B_{j_{n+1},p,\mathbf{t}^1_{k-1}}(\mathbf{x}^1_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \dots \\
&\quad \cdot B_{j_{2n},p,\mathbf{t}^n_{k-1}}(\mathbf{x}^n_{k-1}, \mathbf{M}_{k-1} = \mathfrak{p}) \\
&\quad \cdot B_{i_1,p,\mathbf{t}^1_{k-1}}(\mathbf{x}^1_{k-1}, \mathbf{M}_k = \mathfrak{q}) \dots \\
&\quad \cdot B_{i_n,p,\mathbf{t}^n_{k-1}}(\mathbf{x}^n_{k-1}, \mathbf{M}_k = \mathfrak{q}) \\
&\quad \cdot d\mathbf{x}^1_{k-1} \dots d\mathbf{x}^n_{k-1} \qquad (5.19)
\end{aligned}$$

Let $C_{\mathfrak{j}} = \mathbb{P}_{\mathfrak{j}}$ and $\xi_{\mathfrak{i}} = \mathbb{P}_{\mathfrak{i}}$. Using (5.17), (5.19) and (5.21) with additional manipulations, it can be shown that

$$\begin{aligned}
\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \sum_{j_1} \dots \sum_{j_n} \mathbb{P}_{j_1,\dots,j_n} \\
&\quad \cdot B_{j_1,p,\mathbf{t}^1_k}(\mathbf{x}^1_k, \mathbf{M}_k = \mathfrak{q}) \dots \\
&\quad \cdot B_{j_n,p,\mathbf{t}^n_k}(\mathbf{x}^n_k, \mathbf{M}_k = \mathfrak{q}) \qquad (5.20)
\end{aligned}$$

where $\mathbb{P}_{j_1,\dots,j_n}$ is given by

$$\mathbb{P}_{j_1,\dots,j_n} = \sum_{i_1} \dots \sum_{i_n} \xi_{i_1,\dots,i_n} \sum_{j_{n+1}} \dots \sum_{j_{2n}} C_{j_1,\dots,j_{2n}} W_{j_{n+1},\dots,j_{2n},i_1,\dots,i_n} \qquad (5.21)$$

where $\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ denotes the $\mathfrak{q}$-th mode-dependent existing targets' predicted SPHD.

A similar approach as described for $\mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ can be applied to determine the mode-dependent spawned targets' predicted SPHD $\mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$.

The mode-dependent SPHD of new targets $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ can be determined as follows [58].

First, the mode-dependent posterior probability for an observed measurement that originates from a new target $P_k(Y_i, \mathbf{M}_k = \mathfrak{q})$ is determined as

$$\begin{aligned} & P_k(Y_i, \mathbf{M}_k = \mathfrak{q}) = \\ & \frac{\mathbf{B}_{Y,k}(\mathbf{z}_i, \mathbf{M}_k = \mathfrak{q})}{\mathbf{B}_{\lambda,k}(\mathbf{z}_i) + \sum_{\mathfrak{q}=1}^{\mathsf{r}} \mathbf{B}_{Y,k}(\mathbf{z}_i, \mathbf{M}_k = \mathfrak{q})} \, , \qquad \mathbf{z}_i \in Z_k \ , \ i = 1, \ldots, \daleth_k \end{aligned} \tag{5.22}$$

and

$$\begin{aligned} \mathbf{B}_{Y,k}(\mathbf{z}_i, \mathbf{M}_k = \mathfrak{q}) \;\; = & \int P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ & \cdot \mathbf{B}_{l,k}(\mathbf{z}_i | \mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ & \cdot \mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ & \cdot d\mathbf{x}_k \end{aligned} \tag{5.23}$$

where Y_i denotes the i-th observed measurement that originates from a new target at time k and $\daleth_k$ denotes the total number of measurements at time k. In the above, $\mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \approx \mathbf{B}_{\mathbf{c},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) + \mathbf{B}_{\mathbf{s},k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ and $\mathbf{B}_{l,k}(\cdot)$ denotes the spline likelihood density and could be determined using the measurement model in (2.11). The spline uniform clutter density is denoted by $\mathbf{B}_{\lambda,k}(\cdot)$ and $P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ denotes the mode-dependent probability of detection.

The mode dependent $P_k(Y_i, \mathbf{M}_k = \mathfrak{q})$ values are determined for each measurement and it is compared with a tuning threshold probability ϵ. That is,

$$N_{\mathbf{nb},i} = \begin{cases} 1 \quad \text{if} \ \ P_k(Y_i, \mathbf{M}_k = \mathfrak{q}) \leqslant \epsilon, \\ 0 \qquad \text{otherwise.} \end{cases} \tag{5.24}$$

If the number of newborn targets $N_{\mathbf{nb},i}$ is 1 for a specific measurement index i, then a newborn target SPHD can be added as

$$\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \ \ = \ \sum_{i=1}^{N_{\mathbf{nb}}} \mathbf{B}_{\mathbf{nb},k,i}(\mathbf{z}_i) \tag{5.25}$$

where $\mathbf{B}_{\mathbf{nb},k,i}$ is the SPHD of a newborn target with mean $\mathbf{z}_i$ and variance of measurement noise. The total number of newborn targets per scan is denoted by $N_{\mathbf{nb}}$ and $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ denotes the cumulative sum of all the SPHD values of newborn targets at scan k. Overall, if for the i-th measurement, $\mathbf{z}_i$, $P_k(Y_i, \mathbf{M}_k = \mathfrak{q}) = 1$ then each element of $\mathbf{z}_i$ can be considered as the mean of a newborn target state in its respective dimension with the variance of measurement noise. A newborn target can be added using Gaussian distribution with corresponding mean and variance from each state element of that newborn target. The mode-dependent MM-SPHD of newborn targets, $\mathbf{B}_{\mathbf{nb},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$, depends on system model $\mathfrak{q}$. The expected number of targets can be determined by finding the area of mode-dependent MM-SPHD $\mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$.

5.2.3 MM-SPHD update

Note that the MM-SPHD filter provides the PHD estimates in a continuous space in state. These predicted MM-SPHD $\mathbf{B}_{k|k-1}(\cdot)$ at any point over the interval $[t_{1,k}, t_{\tau,k}]$

can be determined using (5.16). Then, the interval where $\mathbf{B}_{k|k-1}(\cdot)$ is significant could be found. Using the measurement model equation (2.11), the value for the likelihood density function $\mathbf{B}_{l,k}(\mathbf{z}_k|\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ can be evaluated for the same interval. The updated posterior MM-SPHD can be determined as [36] (for $\mathfrak{q} = 1, \ldots, \mathfrak{r}$)

$$\begin{aligned} D_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}|Z^{(k)}) &= (1 - P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}))\mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &\quad + \sum_{\mathbf{z_k} \in Z_k} \frac{\mathbf{B}_\phi(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})}{\mathbf{B}_\lambda(\mathbf{z_k}) + \int \mathbf{B}_\phi(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) d\mathbf{x}_k} \\ &= \sum_{\iota_1, \ldots, \iota_n} \mathbb{P}_{\iota_1, \ldots, \iota_n} B_{\iota_1, p, \mathbf{t}_k^1}(\mathbf{x}_k^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\ &\quad \cdot B_{\iota_n, p, \mathbf{t}_k^n}(\mathbf{x}_k^n, \mathbf{M}_k = \mathfrak{q}) \end{aligned} \tag{5.26}$$

where $\mathbf{B}_\phi(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ can be evaluated as follows:

$$\begin{aligned} \mathbf{B}_\phi(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= P_{\mathfrak{d},k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &\quad \cdot \mathbf{B}_{l,k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \\ &\quad \cdot \mathbf{B}_{k|k-1}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) \end{aligned} \tag{5.27}$$

Then, the updated MM-SPHD can be further simplified as

$$\begin{aligned} \mathbf{B}_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) &= \sum_{\iota_1, \ldots, \iota_n} \mathbb{P}_{\iota_1, \ldots, \iota_n} \\ &\quad \cdot B_{\iota_1, p, \mathbf{t}_k^1}(\mathbf{x}_k^1, \mathbf{M}_k = \mathfrak{q}) \ldots \\ &\quad \cdot B_{\iota_n, p, \mathbf{t}_k^n}(\mathbf{x}_k^n, \mathbf{M}_k = \mathfrak{q}) \end{aligned} \tag{5.28}$$

where $\mathbf{t}_k$ denotes the set of posterior knots and (5.28) ensures that the spline posterior density is only evaluated over the interval where it is significant. Once the significant region is obtained, a simple way of selecting the knots for the posterior intensity is to uniformly distribute the knots over this significant region [58]. The expected number of targets from model $\mathfrak{q}$ can be determined by taking the integral of mode dependent MM-SPHD updated equation $\mathbf{B}_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q})$ as

$$\hat{N}_{k|k}^{\mathfrak{q}} = \int_{\mathcal{E}_s} \mathbf{B}_{k|k}(\mathbf{x}_k, \mathbf{M}_k = \mathfrak{q}) d\mathbf{x}_k \tag{5.29}$$

The total number of targets can be determined as

$$\hat{N}_{k|k} = \sum_{\mathfrak{q}=1}^{\mathsf{r}} \hat{N}_{k|k}^{\mathfrak{q}} \tag{5.30}$$

Mode probability can be updated by first integrating a particular model's mode-dependent updated MM-SPHD. Then the result should be divided by the total expected/average number of targets [49].

5.3 Simulation Results

In this section, a nonlinear maneuvering multitarget tracking example is presented to validate the performance of the proposed MM-SPHD filter. The selected example is a multidimensional one dealing with the bearing-only ground target tracking problem, which arises in many practical applications such as submarine tracking or airborne surveillance using a passive radar [58]. Note that a standard radar tracking problem, where the range and azimuth measurements are available for tracking can be converted into a linear problem. Also, the bearing only tracking problem is inherently ill-conditioned [42, 60] and is better suited for comparing nonlinear target tracking algorithms.

As shown in Figure 5.1, the sensor is on an aircraft with

$$
\begin{aligned}
x_p(k) &= \bar{x}_p(k) + \Delta x_p(k) \qquad k = 0, 1, \ldots, 40 & (5.31)\\
y_p(k) &= \bar{y}_p(k) + \Delta y_p(k) \qquad k = 0, 1, \ldots, 40 & (5.32)
\end{aligned}
$$

where $x_p(k)$ and $y_p(k)$ are the x and y positions of the platform, respectively. The average platform position coordinates are denoted by $\bar{x}_p(k)$ and $\bar{y}_p(k)$, k is the time index and the perturbations $\Delta x_p(k)$ and $\Delta y_p(k)$ are assumed to be mutually independent zero-mean Gaussian white noise sequences with variances $\sigma^2_{\Delta x_p} = 1$ and $\sigma^2_{\Delta y_p} = 1$, respectively. Note that this problem has been used to compare nonlinear filtering tracking algorithms before [3, 24, 58]. The average unperturbed platform motion is assumed to be horizontal with a constant velocity. Its coordinates are given by

$$
\begin{aligned}
\bar{x}_p(k) &= 100k * T \qquad \text{(m)} & (5.33)\\
\bar{y}_p(k) &= 10000 \qquad \text{(m)} & (5.34)
\end{aligned}
$$

where the sampling time $T = 10$s. A system with three models is considered here to demonstrate the MM-SPHD. In the second and third models, a time-varying control term is added. The three system models are

$$
\mathbf{x}_i^1(k) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \mathbf{x}_i^1(k-1) + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} \nu_{1,k}, \tag{5.35}
$$

$$\mathbf{x}_i^2(k) \;=\; \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \mathbf{x}_i^2(k-1) + \begin{bmatrix} -T/2 \\ -T/500 \end{bmatrix} (k-1) + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} \nu_{2,k} \tag{5.36}$$

and

$$\mathbf{x}_i^3(k) \;=\; \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \mathbf{x}_i^3(k-1) + \begin{bmatrix} T/2 \\ T/500 \end{bmatrix} (k-1) + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} \nu_{3,k} \tag{5.37}$$

where the target state is

$$\mathbf{x}_i(k) = \begin{bmatrix} x_i^1(k) \\ x_i^2(k) \end{bmatrix} \qquad i = 1, 2, 3, 4, 5, 6 \tag{5.38}$$

and x_i^1 denotes the position in meters while x_i^2 denotes the velocity in m/s of the i-th target and $\nu_{1,k}$, $\nu_{2,k}$, and $\nu_{3,k}$ are all zero-mean white Gaussian random variables with standard deviation $\sigma_{\nu_{1,k}} = 0.05$ m/s^2, $\sigma_{\nu_{2,k}} = 0.08$ m/s^2 and $\sigma_{\nu_{3,k}} = 0.07$ m/s^2, respectively.

In this example, six maneuvering targets are traveling with initial states

$$[\mathbf{x}_1(k)\ \mathbf{x}_2(k)\ \mathbf{x}_3(k)\ \mathbf{x}_4(k)\ \mathbf{x}_5(k)\ \mathbf{x}_6(k)] = \begin{bmatrix} 1000 & -1000 & 17000 & -17000 & 10000 & -10000 \\ 40 & -40 & -50 & 50 & 50 & -50 \end{bmatrix} \tag{5.39}$$

and the start and end times of the six targets are (1,40), (1,40), (16,38), (16,38), (3,33) and (3,33), respectively.

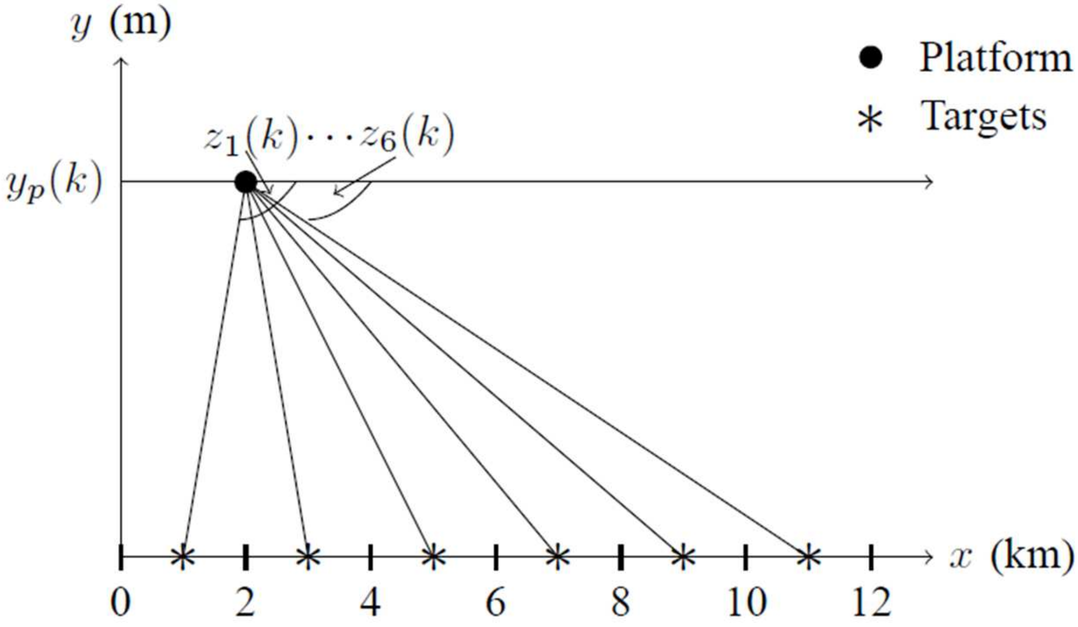

Figure 5.1: Motion of the platform and the six targets

Target 1 moves for the first 140s at a almost steady velocity with an early velocity of 40 m/s, then moves in a positive direction for the next 190s and, finally moves at a almost steady velocity for the last 60s. Target 2 moves for the first 140s at a almost steady velocity with an early velocity of −40 m/s, then moves in the negative direction for 190s and, finally moves at a almost steady velocity for the last 60s. Target 3 moves 90s in the negative direction with an early velocity of −50 m/s, then moves at a almost steady velocity for 50s and for the last 90s moves in a positive direction. Target 4 moves 40s in the positive direction with an early velocity of 50 m/s, then moves at a almost steady velocity for last 180s. Target 5 moves for the first 70s at a almost steady velocity with an early velocity of 50 m/s, then moves in a positive direction for the next 140s and, finally moves at a almost steady velocity for the last 90s. Target 6 moves for the first 70s at a almost steady velocity with an early velocity of -50 m/s, then moves in the negative direction for 140s and, finally moves at a almost steady velocity for the last 90s.

Targets move along the X-axis and these six targets, which have a probability of survival $P_{\mathfrak{s},k} = 0.98$, appear and disappear at specific times. The Markovian model transition probability matrices π_{pq} for the six targets are

$$\pi_{pq} = \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 2/5 & 3/5 & 0.0 \\ 2/5 & 0.0 & 3/5 \end{bmatrix}, \tag{5.40}$$

and the initial model probabilities for the models are 0.33.

Each target is detected with probability $P_{\mathfrak{d},k} = 0.95$ and the target-originated measurements follow the observation model

$$z_i(k) = h[x_p(k), y_p(k), x_i^1(k)] + \omega(k) \quad i = 1, 2, 3, 4, 5, 6 \tag{5.41}$$

where

$$h[.] = \tan^{-1}\frac{y_p(k)}{x_i^1(k) - x_p(k)} \quad i = 1, 2, 3, 4, 5, 6 \tag{5.42}$$

is the angle between the X-axis and the line of sight from the sensor to the targets. The sensor noise $\omega(k)$ is zero-mean white Gaussian with $\sigma_\omega = 2°$. The sensor noise is assumed independent of the sensor platform perturbations. The received measurements include false alarms. The clutter is modeled as uniformly distributed in the measurement space with average false alarm rate $\lambda_k = 10^{-4}$ $(\text{rad})^{-1}$ over the whole surveillance region $[0, \pi]$ rad.

For tracking multiple targets, an MM-SPHD filter of order 3 is used with 20 knots for position and 10 knots for velocity. At scan $k = 0$, all measurements are used to initialize newborn targets as described in Section 5.2.2. The probability of target spawning is assumed to be zero and the probability of spontaneous target birth is 0.01.

As shown in Figure 5.2, the MM-SPHD filter accurately determines the number of targets for two different scenarios. In Figure 5.2, the P_{fa} denotes the probability of false alarms and P_d denotes the probability of detection.

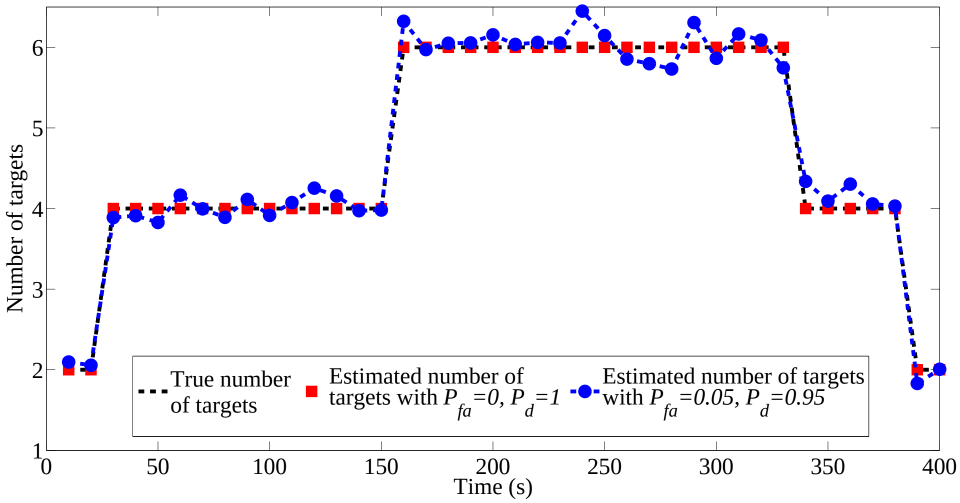

Figure 5.2: True vs. average of estimated number of targets from 1000 runs ($\sigma_\omega = 2°$).

The standard PHD filter does not have a closed solution and also cannot estimates target state directly. One solution is to identify the local maxima of the MM-SPHD surface. The K-means clustering algorithm [61] is used here for state extraction. An alternative is the Expectation-Maximization (EM) based peak extraction approach [62]. The targets are associated to tracks using global nearest-neighbor assignment [47] depend on the mean of each target-cluster. As shown in Figure 5.3, all six targets appear and disappear at various times during the surveillance interval. Also shown in Figure 5.3 are the averages of the estimated trajectories. As shown in Figure 5.4, the MM-SPHD filter estimated the velocities of all targets accurately. The mean velocity of newborn targets is selected randomly from a uniform distribution in the interval $[-100, 100]$ m/s and the standard deviation is assumed to be 0.4 m/s.

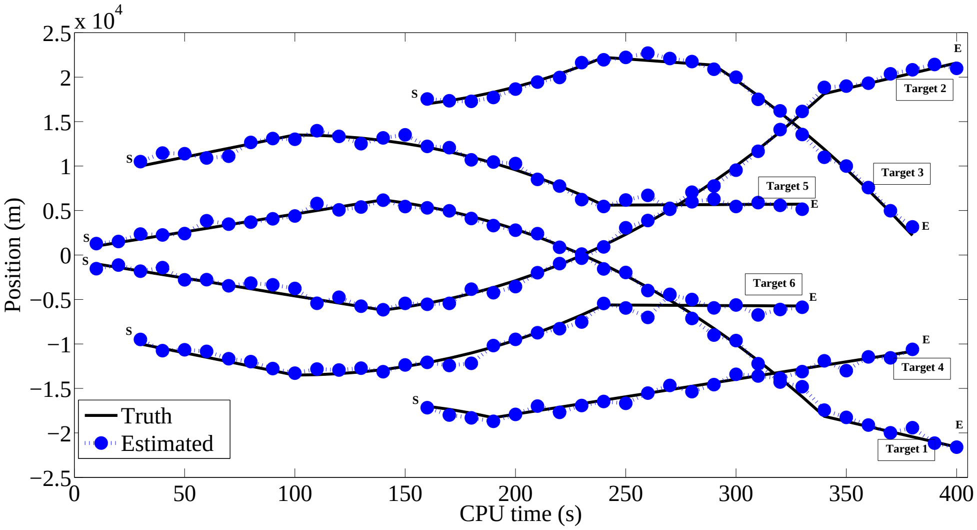

Figure 5.3: True position vs. average of estimated positions from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

Figure 5.5 clearly shows that the choice of knots has significant influence on the performance of the MM-SPHD filter. The selection of the number of knots depends on the application, especially on target separation. Increasing the number of knots affects the complexity of the MM-SPHD filter, but results in better performance in terms of RMSE. Note that with optimal knot selection, the MM-SPHD filter performance can be improved but with higher complexity.

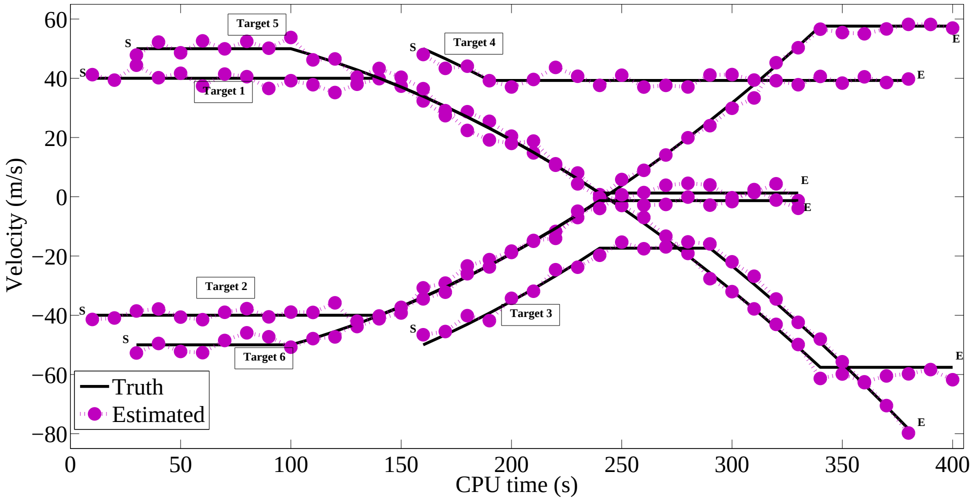

Figure 5.4: True velocity vs. average of estimated velocities from 1000 runs ($\sigma_\omega = 2°$, **S**: start, **E**: end).

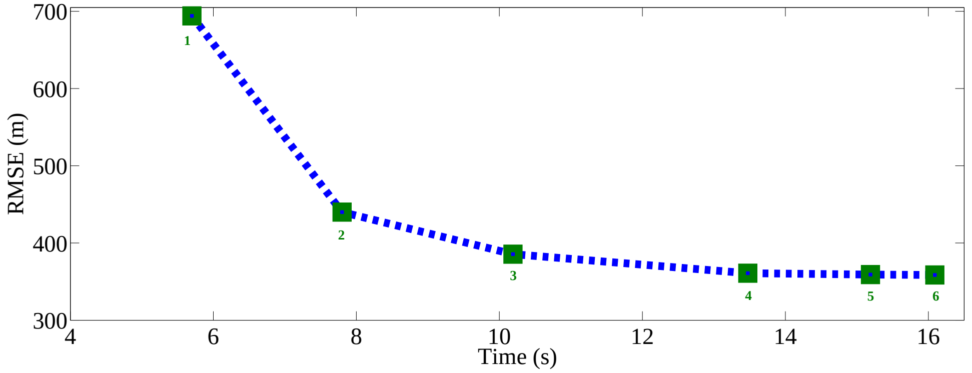

Figure 5.5: Average performance from 1000 runs vs. number of knots, $n_{Pk} = 10$, $n_{Vk} = 5$: 1, $n_{Pk} = 20$, $n_{Vk} = 10 : 2$, $n_{Pk} = 40$, $n_{Vk} = 18$: 3, $n_{Pk} = 50$, $n_{Vk} = 20$: 4, $n_{Pk} = 60$, $n_{Vk} = 22$: 5, $n_{Pk} = 70$, $n_{Vk} = 25$: 6 ($\sigma_\omega = 2°$), n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, number of targets = 6.

Next, the original example is analysed further with an increasing number of targets, but with fixed values for the number of position knots $n_{Pk} = 20$ and the number of velocity knots $n_{Vk} = 10$. As shown in Figure 5.6, increasing the number of targets for a given number of knots adversely affects the performance of the MM-SPHD filter in terms of RMSE and computational time.

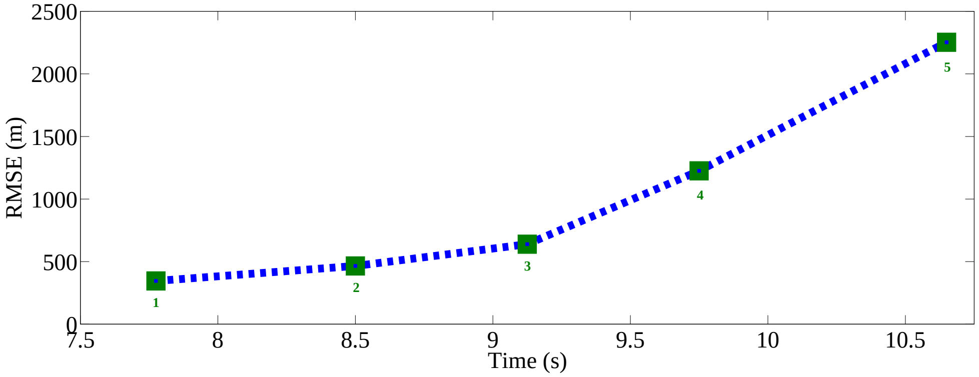

Figure 5.6: Average performance from 1000 runs vs. number of targets ($\sigma_\omega = 2°$), $n_{Pk} = 20$, $n_{Vk} = 10$, (fixed number of knots), $n_t = 5$: 1, $n_t = 10 : 2$, $n_t = 15$: 3, $n_t = 20$: 4, $n_{Vk} = 40$: 5, n_t: Number of targets.

The normalized estimation error squared (NEES) [3] and optimal subpattern assignment (OSPA) [55] are used as performance metrics for the example.

The OSPA [55] metric measures the miss-distance between a set of true targets and a set of estimated tracks as a combination of localisation error and cardinality error [21]. Let $X = \{x_1, \cdots, x_{\acute{n}}\}$ and $Y = \{y_1, \cdots, y_{\acute{m}}\}$ be two finite sets. Here, X denotes true finite set of targets and Y denotes the estimated finite set of tracks. The

OSPA metric is defined as

$$\bar{d}_{\acute{p}}^{(\acute{c})}(X,Y) = \begin{cases} 0 & \text{if } \acute{m} = \acute{n} = 0 \\ \Psi(X,Y) & \text{if } \acute{m} \leq \acute{n}. \\ d^{(\acute{c})}(X,Y) & \text{if } \acute{m} > \acute{n} \end{cases} \tag{5.43}$$

where

$$\Psi(X,Y) \triangleq \left(\frac{1}{\acute{n}} \left(\min_{\pi \in \Pi_{\acute{n}}} \sum_{i=1}^{\acute{m}} d^{(\acute{c})}(x_i, y_{\pi(i)}) + \acute{c}^{\acute{p}}(\acute{n} - \acute{m}) \right) \right)^{\frac{1}{\acute{p}}}$$

and the base distance between x and y denoted by $d^{\acute{c}}(x,y) = \min(\acute{c}, \|x-y\|)$, $\Pi_{\acute{n}}$ is the set of permutations with length $\acute{m}$ on the set of $\{1, \cdots, \acute{n}\}$ where $\acute{n} = \|X\|$ and $\acute{m} = \|Y\|$. In the simulations $\acute{p} = 10$ and $\acute{c} = 100$.

The MM-SPHD filter performance is evaluated along with those of multiple model based GM-USMC-PHD, the GM-SMC-PHD and the AP-PHD filters.

In the MM-GM-USMC-PHD filter implementation, the importance sampling function approximated in the form of a Gaussian mixture that is a sum of Gaussian components and the maximum number of Gaussian terms = 100. The number of samples per GM component or target is set to 2500. The newborn target initialization, resampling and state extraction follow [73]. Note that the GM-USMC-PHD filter does not need resampling because the GM process management for multitarget state extraction and component deletion enables the algorithm to have the same effect as resampling. The Unscented Information Filter (UIF) is the information form of the unscented Kalman filter (UKF) [3]. The UIF is used to compute the mean and the covariance of the Gaussian components.

The GM implementation of the MM-GM-SMC-PHD filter is with the EKF for filtering, pruning parameters of elimination threshold $T_p = 10^{-5}$, merging threshold $T_m = 4$m and maximum number of Gaussian terms 100. The SMC implementation of the MM-GM-SMC-PHD uses the transition density to sample particles. Particles are initialized around measurements [18] and 2500 particles are used per existing target and 50 particles are used for each newborn target. An estimate of the number of targets is determined by summing up all the weights of the particles. The estimation of the number of targets and their state extraction carried as in [50].

The MM-AP-PHD filter uses 2500 particles per existing target, while the number particles per newborn target is set to 100. The initialization of the newborn targets is driven by the measurements. The current measurements are associated with the highest bidder if the bid is at least equals 0.4. The Auxiliary Importance Sampling (AIS) [4] process starts with the selection of the measurements that are well described by the targets' states extracted from the estimated PHD and this is achieved using the Auction algorithm [4]. The state extraction is determined as in [4].

In order to facilitate a fair comparison, we ran all methods with the same multiple model strategy [49]. All PHD filters are initialized with Gaussian distribution with mean [1000 m, 40 m/s] and standard deviation $\sigma_{\nu_{1,k}} = 0.05$ m/s^2 representing target 1 and the constant-velocity model is used. Initially, assume in the desired surveillance region there is only one track exist that correspond to the target 1

The overall filter accuracy performance metric, the OSPA [55], is computed for each filter over 1000 Monte Carlo runs for measurement noise standard deviation levels σ_ω=2° and σ_ω=4°. The OSPA metric measures the combination of both localization and cardinality errors. The average OSPA values are plotted in Figures 5.7 and 5.8.

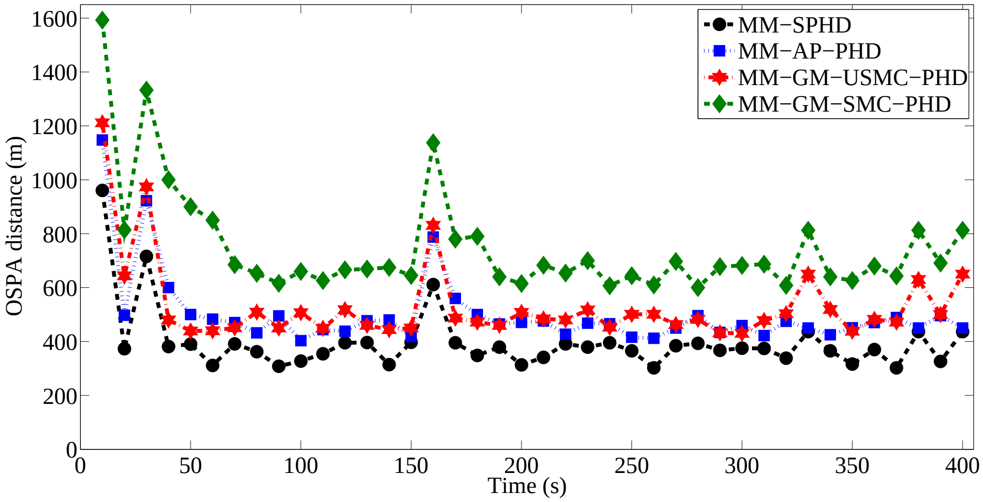

Figure 5.7: OSPA distance (m) averaged over 1000 Monte Carlo runs ($\sigma_\omega = 2°$, $\acute{c} = 10$, $\acute{p} = 100$).

The key observation is that the MM-SPHD filter with a few knots performed the best in terms of OSPA for both measurement noise levels. As shown in Figure 5.7 and 5.8, high values of OSPA distance occur when new targets are born around time indices $k = 1, 3, 16$. Also targets disappear with small OSPA peaks at time indices $k = 33, 38, 40$. As shown in Figure 5.8, as the nonlinearity increases with increasing measurement noise levels the GM-based MM-PHD filter perform poorly.

Figure 5.9 reveals the consistency of the MM-SPHD, MM-GM-USMC-PHD, MM-AP-PHD and the MM-GM-SMC-PHD filters in terms of normalized estimation error squared (NEES) compared with the 95% confident-region of the $\mathcal{X}^2$ distributions [3] when the measurement noise standard deviation is $\sigma_\omega = 2°$. To illustrate the degeneracy resistance capability of the proposed MM-SPHD filter, the standard deviation of the measurement noise is reduced to σ_ω=0.02°. The model parameters for the filters remain unchanged but with the correct measurement noise level. This scenario causes the particle-based PHD filters to become degenerative. It can be observed from Figure 5.10 that the MM-SPHD filter is able to provide efficient results with the same 10 velocity knots and 20 position knots. Note that using the Regularized Particle Filter (RPF) [18, 19] can avoid the degeneracy problem caused by sampling and resampling. However, the RPF has the disadvantage is that the samples are no longer guaranteed to asymptotically approximate the posterior [44].

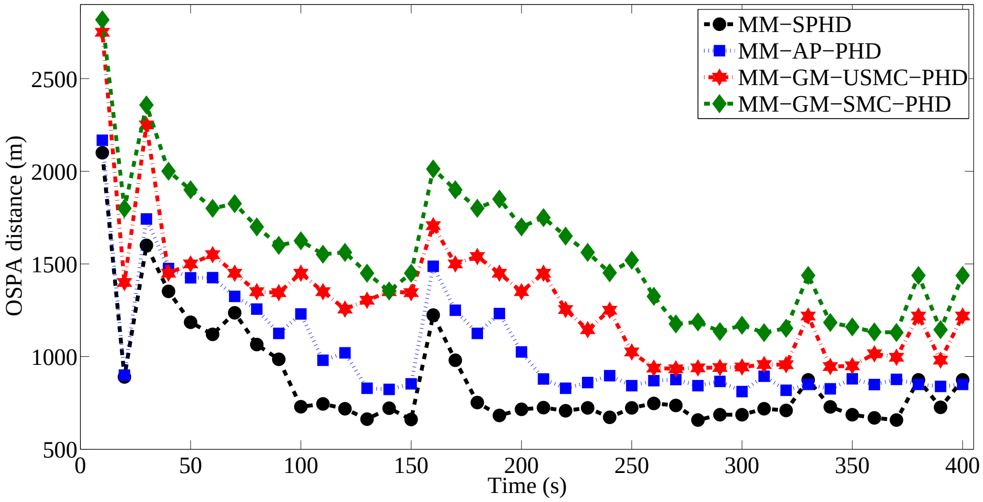

Figure 5.8: OSPA distance (m) averaged over 1000 Monte Carlo runs ($\sigma_\omega = 4°$, $\acute{c} = 10$, $\acute{p} = 100$).

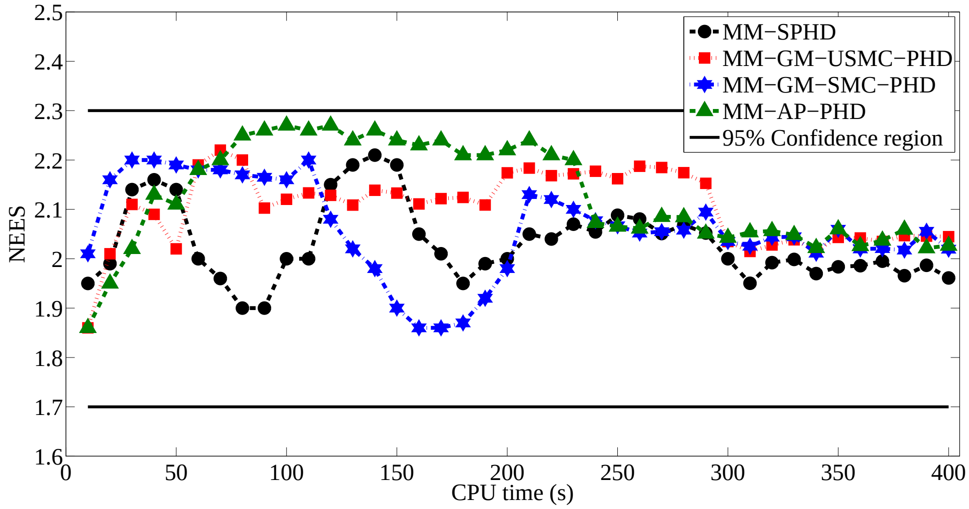

Figure 5.9: NEES comparison from 1000 Monte Carlo runs ($\sigma_\omega = 2°$).

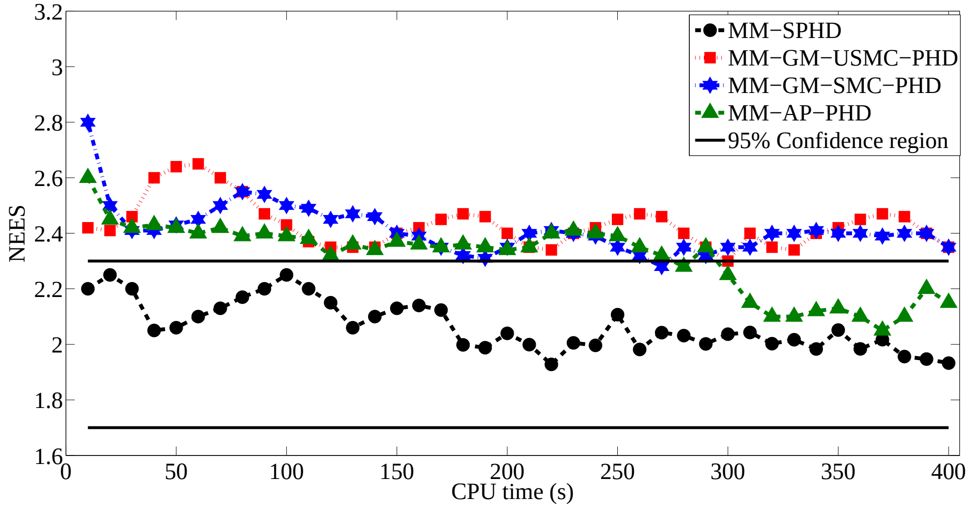

Figure 5.10: NEES comparison from 1000 Monte Carlo runs ($\sigma_\omega = 0.02°$).

Chapter 6

Spline Maximum-Likelihood Probabilistic Data Association Algorithm for Track Initialization

6.1 Introduction

The Maximum-Likelihood Probabilistic Data Association (ML-PDA) algorithm is commonly used in low-observable target tracking problems to initiate tracks. Initially, the ML-PDA algorithm was used in angle-only tracking problems, but subsequent extensions made it possible to work with amplitude information and frequency measurements, especially to track very low observable targets. For real-time applications, a window-based ML-PDA algorithm was also implemented. The performance of the ML-PDA algorithm is contingent upon finding of a global maximum in the log-likelihood ratio (LLR) followed by track validation. To improve performance of the ML-PDA algorithm in this paper, a B-Spline based Maximum-Likelihood Probabilistic Data Association (SML-PDA) algorithm is proposed. The newly proposed algorithm represents the LLR using a set of B-Splines and the SML-PDA LLR can have any arbitrary distribution. Using a few knots, SML-PDA LLR can be represented accurately and the suboptimal knot selection, which is an iterative method, can be used to create an optimal knot vector in order to find the unknown parameter vector. The global maximum of SML-PDA LLR can be determined by finding the knot element with the highest peak. The SML-PDA algorithm can determine the global maximum of ML-PDA LLR with highly accurate parameter estimates, but with low computational complexity for low dimensional problems. For higher dimensional target tracking problems, the SML-PDA algorithm performs well, albeit with a higher computational complexity, which make it suitable for tracking high-priority targets under difficult conditions.

6.2 SML-PDA Algorithm

The proposed SML-PHD algorithm's derivation are explained in this section. Assume there is a single target $\mathbf{x}_r = \{x_r^i\}_{i=1}^n \in \mathcal{E}_s$ already exist at a reference time r in the multidimensional state space $\mathcal{E}_s$ with survival probability $P_{\mathfrak{s}}(\mathbf{x}_r)$, where n denotes the state dimensions. The dynamic state of the target evolves as described in (2.43) respect to time. Across the data frame the detections are independent. A single target is always presence at all the data frames and it is detected with the detection probability P_d. Assume the subsequent measurement model with the P_d where the single target in state space $\mathcal{E}_s$ can stimulates observations. The measurement in the observation space $\mathcal{E}_o$ can also originated form the false alarms or clutter. In the surveillance region with volume U the false alarms are uniformly distributed. Across the frames the number of false detection is independent and Poisson distributed respect to probability mass function $\mu_f(m)$, with spatial density λ. Note, that the measurement also contain the amplitude features. As in [31] the envelope output of the detected is denoted as the amplitude.

Let the cumulative set of measurement $\mathbf{Z} \triangleq \{Z(i)\}_{i=1}^{N_w} \in \mathcal{E}_o$, where N_w denotes the frame number, $\mathcal{E}_o$ denotes the measurement space and $Z(i)$ denotes all the observations in a certain data-frame at time i defined as

$$Z(i) \quad \triangleq \quad \{\beta_{ij}, a_{ij}\}_{j=1}^{m_i} \in \mathcal{E}_o \tag{6.1}$$

where β_{ij} denotes the measurement from target-originated and from false detection and a_{ij} denotes the amplitude. In case the sensor is a narrowband sonar and the measurements contain frequency measurements then (6.1) can be modified as

$$Z(i) \quad \triangleq \quad \{\beta_{ij}, \psi_{ij}, a_{ij}\}_{j=1}^{m_i} \in \mathcal{E}_o \tag{6.2}$$

Note, the scenario involves with a single target, the total number of measurement per scan m_i equal to the sum of number of measurement originated from true target (once only) and the measurement originated from false detection, and ψ_{ij} denotes the frequency measurement.

The spline likelihood probability density function (SLPDF) of a target-originated measurement is defined as

$$\begin{aligned} p(\beta_{ij}, \psi_{ij}|\mathbf{x}_r) &\approx \sum_{k_1}^{n_s} \cdots \sum_{k_n}^{n_s} \mathbb{P}_{k_1,\ldots,k_n} B_{k_1,p,\mathbf{t}_i^1}(\beta_{ij}, \psi_{ij}|x_r^1) \ldots B_{k_n,p,\mathbf{t}_i^n}(\beta_{ij}, \psi_{ij}|x_r^n) \\ &= \sum_{k_1}^{n_s} \cdots \sum_{k_n}^{n_s} \mathbb{P}_{k_1,\ldots,k_n} B_{k_1,p,\mathbf{t}_i^1}(z_r^1) \ldots B_{k_n,p,\mathbf{t}_i^n}(z_r^n) \\ &= \mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r) \end{aligned} \tag{6.3}$$

where $i \in \{1, \ldots, N_w\}$, $j \in \{1, \ldots, m_i\}$, $\mathfrak{k} = k_1, \ldots, k_n$ and the number of dimensions denoted by n. The number of knot elements for all the dimensions is the same (τ) and p denotes the order of the splines. The n dimensional knot $\mathbf{t}_i = \{\mathbf{t}_i^1, \ldots, \mathbf{t}_i^n\}$ is an $n \times \tau$ array. Each row vector of $\mathbf{t}_i$ consists of a set of knot elements $\mathbf{t}_i^l = \{t_{1,i}^l, \ldots, t_{\tau,i}^l\}$ where $l = 1, \ldots, n$. The n dimensional control point or coefficient matrix is denoted by $\mathbb{P}_{\mathfrak{k}}$ and the number of control points denoted by n_s. For each dimensions the same number of control points are used. Note, that the number of knots must be greater than the number of control points. For brevity SLPDF of target originated measurement is denoted by $\mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)$. For simplicity, $\{z_r^k\}_{k=1}^n = \{(\beta_{ij}, \psi_{ij}|x_r^k)\}_{k=1}^n$ where z_r^l represent the l-th dimensional target-originated measurement. Note, with known surveillance region and measurement noise covariance (2.49) can be easily transformed into (6.3), but for the SML-PDA algorithm, Gaussian assumption of measurement noise is not needed. In the desired surveillance region the false measurements are distributed uniformly. For example, for bearing-only scenario $\beta_{ij} \sim U[\theta_1, \theta_2]$, where θ_1 and θ_2 are denote the angles and the frequency $\psi_{ij} \sim U[\Omega_1, \Omega_2]$, where Ω_1 and Ω_2 are frequencies. Using the tensor product of B-Spline can easily obtain the uniform distribution for multidimensional state problem [53].

The amplitude information can be obtained as in [31] for any scenario. Generally, a number of detections are made by the radar/sonar at every sampling time and must decide which measurements are likely to have originated from the target. The estimator operates in the presence of false alarm since more than one measurement can satisfy the validation criteria. As described in [31] the output of the envelope detector a is a Rayleigh distributed. The output PDF of envelope detector is $p_1(a_{ij})$ if originated from the true target or $p_0(a_{ij})$ if the output signal due to false alarm. Note, a suitable detection threshold ξ must be used for detection and it depends on the SNR as well as must satisfy the selection of P_d and P_{fa} values. The output likelihood PDF of the threshold detector are described in (2.51) and (2.52). The amplitude likelihood is incorporated into the likelihood function as ratio.

The amplitude likelihood ratio between the target and noise originated amplitude measurement likelihoods can be defined as

$$\rho_{ij} = \frac{p_1(a_{ij}|\xi)}{p_0(a_{ij}|\xi)}, \tag{6.4}$$

where the target originated amplitude likelihood PDF denoted by $p_1(a_{ij}|\xi)$ and $p_0(a_{ij}|\xi)$ denotes the amplitude likelihood PDF of the validated measurements originated from false alarm.

The spline amplitude likelihood probability density function (SAL-PDF) of those originated from the target can be defined as

$$\begin{aligned} p_1(a_{ij}|\xi) &= \frac{a_{ij}}{P_d(1+\mathrm{SNR})} \exp\left(-\frac{a_{ij}^2}{2(1+\mathrm{SNR})}\right), \quad a_{ij} > \xi \\ &\approx \sum_{\varsigma_1}^{n_s} \mathbb{P}_{\varsigma_1} B_{\varsigma_1, p, \mathbf{t}_i^{1'}}(a_{ij}|\xi) \\ &= \mathbf{B}_1(a_{ij}|\xi) \end{aligned} \tag{6.5}$$

where $i \in \{1, \ldots, N_w\}$, $j \in \{1, \ldots, m_i\}$, $\mathbf{B}_1(a_{ij}|\xi)$ denotes the SAL-PDF of those originated from the target, $\mathbf{t}_i^{1'} = \{t_{1,i}^{1'}, \ldots, t_{\tau,i}^{1'}\}$ is the set of knot elements of the splines and $\mathbb{P}_{\varsigma_1}$ denotes the ς_1-th control point of the SAL-PDF. The number of knots is denoted by τ and the number of control points are denoted by n_s.

The SAL-PDF of the validated measurements that originated from false alarm can be defined as

$$\begin{aligned} p_0(a_{ij}|\xi) &= \frac{a_{ij}}{P_{fa}} \exp\left(-\frac{a_{ij}^2}{2}\right), \quad a_{ij} > \xi \\ &\approx \sum_{\varsigma_0}^{n_s} \mathbb{P}_{\varsigma_0} B_{\varsigma_0, p, \mathbf{t}_i^{0'}}(a_{ij}|\xi) \\ &= \mathbf{B}_0(a_{ij}|\xi) \end{aligned} \tag{6.6}$$

where $\mathbf{B}_0(a_{ij}|\xi)$ denotes the SAL-PDF of those originated from false alarm, $\mathbf{t}_i^{0'} = \{t_{1,i}^{0'}, \ldots, t_{\tau,i}^{0'}\}$ is the set of knot elements of the splines and $\mathbb{P}_{\varsigma_0}$ denotes the ς_0-th control point of the SAL-PDF. The number of knots is denoted by τ and the number of control points are denoted by n_s . The B-splines [12,53] can be used to represent any distribution functions and in this case, the Rayleigh distributed functions $p_1(a_{ij}|\xi)$ and $p_0(a_{ij}|\xi)$ mean and variance values can be determined and using these values as described in [12,53] can select the knot vectors $\mathbf{t}_i^{1'}$ and $\mathbf{t}_i^{0'}$.

For further derivation purpose consider a mutually exclusive events as

$$\varphi_j(i) \triangleq \begin{cases} \{\text{measurement } \mathbf{z}_j(i) \text{ is from the target}\}, & j = 1, \ldots, m_i \\ \{\text{all the measurement are false}\}, & j = 0 \end{cases} \tag{6.7}$$

and the PDF of the measurement corresponding to (6.7) can be defined as

$$p\left(Z(i)|\varphi_j(i), \mathbf{x}_r\right) = \begin{cases} \frac{1}{U^{m_i-1}} p(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\rho_{ij} \prod_{l=1}^{m_i} p_0(a_{il}|\xi), & j = 1, \ldots, m_i \\ \frac{1}{U^{m_i}} \prod_{l=1}^{m_i} p_0(a_{il}|\xi), & j = 0 \end{cases} \tag{6.8}$$

where $U = U_\beta U_\gamma$ is the volume of the surveillance region.

At time i the spline likelihood function of the set of measurement can be determined by using the total probability theorem as

$$\begin{aligned} p\left(Z(i)|\mathbf{x}_r\right) &= \frac{(1-P_d)}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} p_0(a_{ij}|\xi) \\ &\quad + \frac{P_d\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} p_0(a_{ij}|\xi)\sum_{j=1}^{m_i} p(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\rho_{ij} \\ &= \frac{(1-P_d)}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} \mathbf{B}_0(a_{ij}|\xi) \\ &\quad + \frac{P_d\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} \mathbf{B}_0(a_{ij}|\xi)\sum_{j=1}^{m_i} \mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\frac{\mathbf{B}_1(a_{ij}|\xi)}{\mathbf{B}_0(a_{ij}|\xi)} \\ &= \mathbf{B}\left(Z(i)|\mathbf{x}_r\right) \end{aligned} \tag{6.9}$$

where $\mathbf{B}\left(Z(i)|\mathbf{x}_r\right)$ denotes the SLPDF of the set of measurement at time i. At time i across the frames the number of false detection is independent and Poisson distributed respect to probability mass function $\mu_f(.)$, with spatial density λ. The volume of the surveillance region is denoted by U.

Also the spline likelihood function given all the measurement are false detection can be defined at time i as

$$\begin{aligned} p\left(Z(i)|\varphi_0(i), \mathbf{x}_r\right) &= \frac{1}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} p_0(a_{ij}|\xi) \\ &= \frac{1}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} \mathbf{B}_0(a_{ij}|\xi) \\ &= \mathbf{B}\left(Z(i)|\varphi_0(i), \mathbf{x}_r\right) \qquad (6.10) \end{aligned}$$

Dividing (6.9) by (6.10) a dimensionless spline ML-PDA likelihood ratio (SML-PDA LR), $\mathbf{B}_\Phi\left(Z(i)|\mathbf{x}_r\right)$ can be determined at time i as

$$\begin{aligned} \mathbf{B}_\Phi\left(Z(i)|\mathbf{x}_r\right) &= \frac{\mathbf{B}\left(Z(i)|\mathbf{x}_r\right)}{\mathbf{B}\left(Z(i)|\varphi_0(i), \mathbf{x}_r\right)} \\ &= (1-P_d) + \frac{P_d}{\lambda}\sum_{j=1}^{m_i}\mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\frac{\mathbf{B}_1(a_{ij}|\xi)}{\mathbf{B}_0(a_{ij}|\xi)} \qquad (6.11) \end{aligned}$$

where the expected number of false alarm per unit volume is denoted by λ.

Note, the observations are conditionally independent across the frame. Therefore the spline likelihood function of the entire set of measurements can be written in terms of individual spline likelihood function [2]. The dimensional SML-PDA LR for the entire data can be defined as

$$\begin{aligned} \mathbf{B}_\Psi\left(\mathbf{Z}|\mathbf{x}_r\right) &= \prod_{i=1}^{N_w}\mathbf{B}_\Phi\left(Z(i)|\mathbf{x}_r\right) \\ &= \prod_{i=1}^{N_w}\left[(1-P_d) + \frac{P_d}{\lambda}\sum_{j=1}^{m_i}\mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r)\frac{\mathbf{B}_1(a_{ij}|\xi)}{\mathbf{B}_0(a_{ij}|\xi)}\right] \qquad (6.12) \end{aligned}$$

Alternatively (6.12) can be defined in terms of total log-likelihood ratio (LLR) as

$$\begin{aligned}\mathbf{B}_{\Lambda}\left(\mathbf{Z}|\mathbf{x}_r\right) &= \ln \mathbf{B}_{\Psi}\left(\mathbf{Z}|\mathbf{x}_r\right) \\ &= \sum_{i=1}^{N_w} \ln \left[(1-P_d) + \frac{P_d}{\lambda} \sum_{j=1}^{m_i} \mathbf{B}(\beta_{ij}, \psi_{ij}|\mathbf{x}_r) \frac{\mathbf{B}_1(a_{ij}|\xi)}{\mathbf{B}_0(a_{ij}|\xi)} \right] \qquad (6.13)\end{aligned}$$

where SML-PDA LLR $\mathbf{B}_{\Lambda}\left(\mathbf{Z}|\mathbf{x}_r\right)$ is a multidimensional spline PDF. Note, tensor product spline construction [12] can be used to build multidimensional splines PDF.

The target estimates $\hat{\mathbf{x}}_r$ from the SML-PDA algorithm can be determined as

$$\hat{\mathbf{x}}_r = \arg\max_{\mathbf{x}_r} \mathbf{B}_{\Lambda}\left(\mathbf{Z}|\mathbf{x}_r\right), \qquad (6.14)$$

The global maximum of SML-PDA LLR can be easily determined by finding the knot element with a higher distribution. The optimal knot selection can be done by using a Genetic Algorithm based on spline approximation method as described in [22,23] or using a few knots, SML-PDA LLR can be represented by using spline interpolation and the optimal knot selection, which is an iterative method, can be used to create a knot vector that is optimal respect to a given parametrization [58,57]. The different between the Spline Interpolation Method (SIM) and the Spline Approximation Method (SAM) is that SAM does not necessarily pass through all the control points, but must go through the first and last ones and SIM must pass through all $\mathbb{P}$ control points. This leads SIM to give a better characteristic information of any given curve/surface/volume/hypersurface. Thus the B-spline interpolation method has been adapted here to implement the SML-PDA algorithm.

The SML-PDA algorithm's track estimates have to satisfy the track validations process [31]. A track validation process is applied to see if the estimated target states come from true-target, no-target or false alarm [2]. Here the applied assumption is a target exist all the time, so not necessarily have to deal with no-target scenario, but the true-target may invisible due to high clutter or low SNR. During the validation process for an extracted state estimates the SML-PDA LLR distribution value is known and one can use this value as the tuning-parameter. The SML-PDA algorithm applied with the same measurement set number of times to see if the SML-PDA LLR distribution value go beyond the tuning-parameter. With the presence of the true-target the estimated values not differ much for all the trails, but the SML-PDA LLR values might be slightly differ. Note, the SML-PDA LLR formed by using an optimal knot sections, where one can have the exact capture of the SML-PDA LLR, so with the presence of the true-target between each trails the estimated states and the corresponding SML-PDA LLR values are always similar. If the state estimated result is generated from the false alarms mean for every trail/run the state estimate vary dramatically with different SML-PDA LLR. So this kind of testing used to validate the estimated tracks. If the outcome of track validation process comes from a false alarm then can repeat the state estimation process again by discarding the initial measurements frame by using a sliding window with desired size. Using the sliding widow this process can repeated until the track validation process outcome come from the true-target.

6.3 Simulation Results

The performance of the proposed SML-PDA algorithm is validated by using a practical problem. Here, using the SML-PDA algorithm a single target in high clutter or low SNR is initialized. The measurements in this scenario also contain features i.e., frequency and amplitude. Note, the selected problem is only deal with bearing-only ground target tracking problems. The selected scenario arises in many practical applications such as submarine tracking or airborne surveillance using a passive radar [29,31,58,57]. The sensor is a narrowband sonar and it sits on a moving platform. The desired surveillance region is selected to be a 30 km-by-30 km square region in which one target is placed and the origin is located at the (0,0) m. For this scenario assume a VLO target is present at all time.

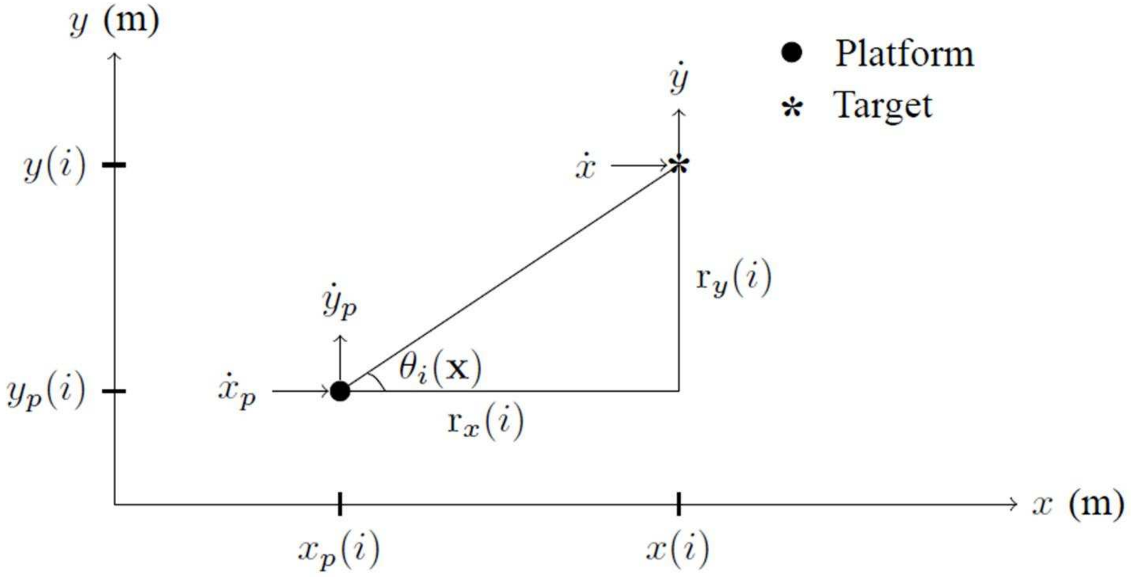

Figure 6.1: Motion of the platform and a target at time i

As show in Figure (6.1) at time i a sensor on a platform moves along X-Y plane and the state of the platform can be defined by

$$\mathbf{x}_p^{\mathtt{m}}(i) \;=\; [x_p(i) \;\; \dot{x}_p \;\; y_p(i) \;\; \dot{y}_p]' \qquad \mathtt{m} \in \{1,2\}, \tag{6.15}$$

where $\mathtt{m}$ denotes the mode of operations, the positions of the sensor platform are denoted as $x_p(i)$, $y_p(i)$ and the velocities of the sensor platform are denoted as $\dot{x}_p$, and $\dot{y}_p$. A sensor platform system with two models is considered here to demonstrate the SML-PDA algorithm.

The two system models of sensor platform are

$$\mathbf{x}_p^1(i+1) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_p^1(i) + \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \begin{bmatrix} \sigma_p^x \\ \sigma_p^y \end{bmatrix}, \qquad (6.16)$$

and

$$\begin{aligned} \mathbf{x}_p^2(i+1) &= \begin{bmatrix} 1 & \sin(w*T)/w & 0 & (\cos(w*T)-1)/w & 0 \\ 0 & \cos(w*T) & 0 & -\sin(w*T) & 0 \\ 0 & (1-\cos(w*T))/w & 1 & \sin(w*T)/w & 0 \\ 0 & \sin(w*T) & 0 & \cos(w*T) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_p^2(i) \\ &+ \begin{bmatrix} T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix} \begin{bmatrix} \sigma_p^x \\ \sigma_p^y \\ \sigma_p^w \end{bmatrix} \end{aligned} \qquad (6.17)$$

where (6.16) and (6.17) denote a constant velocity model (CVM) and a constant turn model (CTM). The sampling time is denoted as $T = 30$ (s) and the sensor platform process noises are zero-mean white Gaussian random variables with standard deviation $\sigma_p^x = 0.00005$ m/s^2, $\sigma_p^y = 0.00005$ m/s^2 and $\sigma_p^w = 0.00005$ rad/s, respectively. The angular rate is denoted by $w_{ar} = \pm 0.0200$ rad/s and σ_p^w denotes the standard deviation of turn rate noise variance.

For bearing-only estimation a target motion with SNR = 6 dB is defined as

$$\mathbf{x}(i) \;=\; \left[x(i) \;\; \dot{x} \;\; y(i) \;\; \dot{y} \;\; \gamma\right]', \tag{6.18}$$

where $\mathbf{x}(i)$ is a five dimensional vector, $x(i)$ and $y(i)$ are the positions of the target in the X and Y directions respect from the origin at the reference time i, respectively. The respective velocities $\dot{x}$, $\dot{y}$ and the frequency γ are assumed constants.

The target state motion, based on a constant velocity model is defined as

$$\mathbf{x}(i+1) \;=\; \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}(i) + \begin{bmatrix} T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_{\dot{\gamma}} \end{bmatrix}, \tag{6.19}$$

where target system process noises are zero-mean white Gaussian random variables with standard deviation $\sigma_x = 0.0$ m/s^2, $\sigma_y = 0.0$ m/s^2 and $\sigma_{\dot{\gamma}} = 0.00$ Hz, respectively.

As shown in Figure (6.1) the relative position and velocity components in the X and Y directions are denoted as $\mathrm{r}_x(i)$ and $\mathrm{r}_y(i)$, $v_x(i)$ and $v_y(i)$, respectively.

The target is detected with probability P_d and the target-originated measurements follow the observation model at time i

$$\beta_{ij} \quad = \quad h\left(\mathbf{x}(i), \mathbf{x}_p(i)\right) + \omega_i \qquad j \in \{1, \ldots, m_i\}, \tag{6.20}$$

where at time i

$$h[.] = \tan^{-1}\left(\frac{\mathrm{r}_y(i)}{\mathrm{r}_x(i)}\right) = \theta_i(\mathbf{x}) \tag{6.21}$$

is the true bearing between the target and the platform. The sensor noise ω_i is zero-mean white Gaussian with $\sigma_\omega^2 = 2.3°$. The target-originated frequency measurements follow the observation model as

$$\psi_{ij} \quad = \quad \gamma_i(\mathbf{x}) + \nu_i \qquad j \in \{1, \ldots, m_i\}, \tag{6.22}$$

where ν_i is a zero-mean white Gaussian noise with known variance $\sigma_\gamma^2 = 0.0054$ Hz and $\gamma_i(\mathbf{x})$ denotes the noise free Doppler shifted frequency at the sensor due to the relative motion between the target and sensor platform at time time i.

The noise free Doppler shifted frequency can be defined as

$$\gamma_i(\mathbf{x}) \quad = \quad \gamma\left[1 - \frac{v_x(i)\sin\theta_i(\mathbf{x}) + v_y(i)\cos\theta_i(\mathbf{x})}{\mathfrak{c}}\right] \tag{6.23}$$

where the relative velocity components are denoted as $v_x(i)$, $v_y(i)$ and $\mathfrak{c}$ denotes the velocity of sound in the medium. The received measurements also include false alarms. The sensor noise is assumed independent of the sensor platform noise. The noise-free amplitude as well as the false alarm-originated amplitude measurements are generated by using the Rayleigh distribution and SNR values.

The SML-PDA algorithm's performance is evaluated with and without frequency information. First the SML-PDA algorithm is implemented for a general bearing-only problem with amplitude information to track a target moving at a constant velocity. The sensor's two dimensional resolution cell has area $A = \left(3\sqrt{2.3}\right)^2 \approx 20$. The return signal is processed by a quadrature receiver with a Constant False Alarm Rate (CFAR) setting for $P_{fa} = 0.1$ per cell, which yields a false alarm spatial density of $\lambda_i = P_{fa}/A = 0.0048$ (degree)$^{-1}$. The clutter is modeled as uniformly distributed in the measurement space with average false alarm rate λ_i over the whole surveillance region $[-\pi, \pi]$ rad.

Next the SML-PDA algorithm performance is evaluated for narrowband sonar scenario. For narrowband case, the signal processor was assumed to consist of the frequency band [500 Hz 1000 Hz] with a 2048-point fast Fourier transform (FFT).

The frequency cell size defined as

$$C_\gamma = 500/2048 = 0.25 \text{ Hz} \ , \tag{6.24}$$

and for azimuth measurements the sonar is assumed to have 34 equal beams resulting in an azimuth cell with size

$$C_\theta = 180°/34 = 5.30°, \tag{6.25}$$

where C_θ denotes the azimuth cell size. Since the noises are distributed uniformly in the cell, the frequency and azimuth measurement standard deviation can be determined as

$$\sigma_\theta = 5.30/\sqrt{12} \approx 1.52^\circ \tag{6.26}$$

$$\sigma_\gamma = 0.25/\sqrt{12} \approx 0.072 \text{ Hz} \tag{6.27}$$

Note the uniform factor $\sqrt{12}$ is correspond to the worst case [31] and $\sigma_\theta^2 \approx 2.3^\circ$, which is the same as used in bearing-only scenario. The SNR in a cell is taken as 6 dB and $P_d \approx 0.70$. Then using (2.51) and (2.52), the detection threshold is determined to be $\xi = 2.10$ and the $P_{fa} = 0.11$. The expected number of false alarms per unit volume can be determined using

$$P_{fa} \quad = \quad \lambda C_\theta C_\gamma, \tag{6.28}$$

and substituting P_{fa}, C_θ and C_γ can obtain the $\lambda = 0.083$/deg.Hz. Note the surveillance region of azimuth and frequency measurements are

$$U_\theta = [-\pi, \pi] \tag{6.29}$$

$$U_\gamma = [747 \text{ Hz}, 753 \text{ Hz}] \tag{6.30}$$

Finally the SML-PDA estimator performance is evaluated for both bearing-only and narrowband sonar case with and without the amplitude information.

The trajectories of the targets and the sensor platform are shown in Figure (6.2). As shown in Figure (6.2) the scenario represents a single ground target tracking problem using an Air Craft Carrier based sensor where a target is moving on the ground. Note, for bearing-only target tracking problem the sensor platform has to perform maneuvering to get a visible observation of the target, but for narrowband sonar scenario maneuvering is not needed.

The sensor with initial sates at i is selected as

$$\mathbf{x}_p^1(i=1) \quad = \quad [-7000\ (\mathrm{m})\ \ 15\ (\mathrm{m/s})\ \ -7000\ (\mathrm{m})\ \ 20\ (\mathrm{m/s})]' \tag{6.31}$$

and a CVM is used as the initial model. This scenario has 14 legs and the first 5 time-index the CVM is used followed by next 3 time-index CTM is used. Note, a same pattern followed for next 56 time-index between CVM and CTM, but CTM with angular rate $w_{ar} = \pm 0.0200$ rad/s. A sliding window, with a 20 batch of measurement frames is used to compute the target estimates.

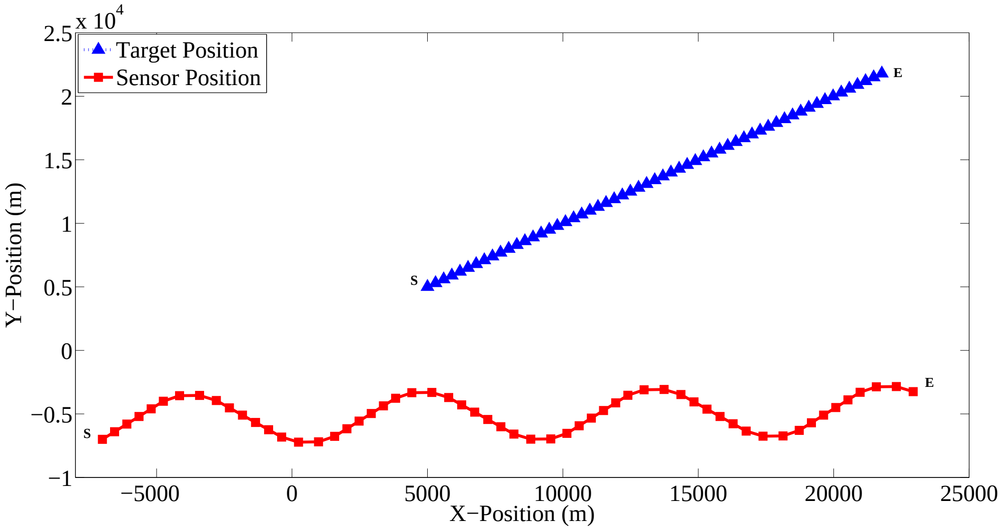

Figure 6.2: Trajectories of true-target and true-platform, **S**: start, **E**: end.

The target is traveling with initial states

$$\mathbf{x}(i=1) \quad = \quad [5000\ (\mathrm{m})\ \ 10\ (\mathrm{m/s})\ \ 5000\ (\mathrm{m})\ \ 10\ (\mathrm{m/s})\ \ 750\ (\mathrm{Hz})]' \qquad (6.32)$$

where the emitted frequency is 750 (Hz). As shown in Figure (6.3) the target-originated and noise-only measurements can be distinguished from a single Monte Carlo run (MCR). However, the index of the target-originated measurement is not known to the estimator. Figures (6.4) and (6.5) show a set of amplitude measurements and frequency measurement, respectively. Note total number of scans/frame in a MCR is 56 and the sampling time is 30 (s).

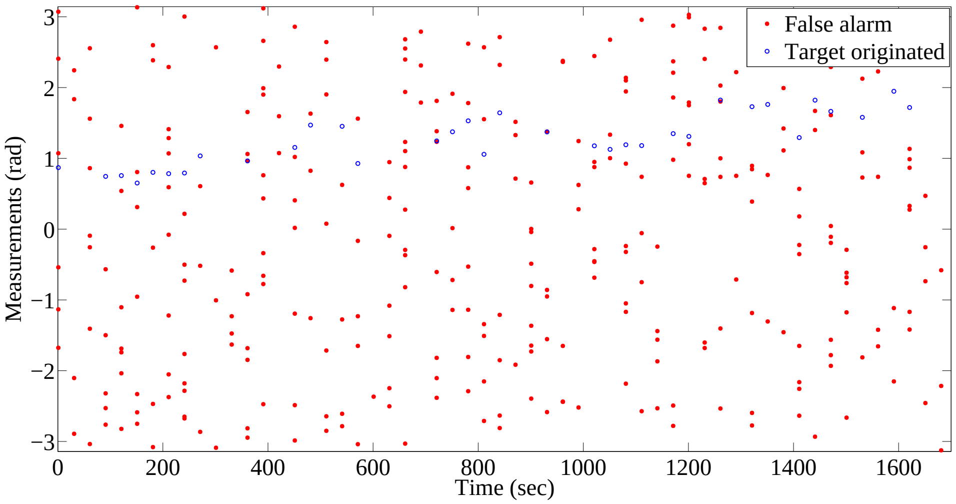

Figure 6.3: Target-originated and false alarm measurements, Scans = 56, $T = 30$ (s)

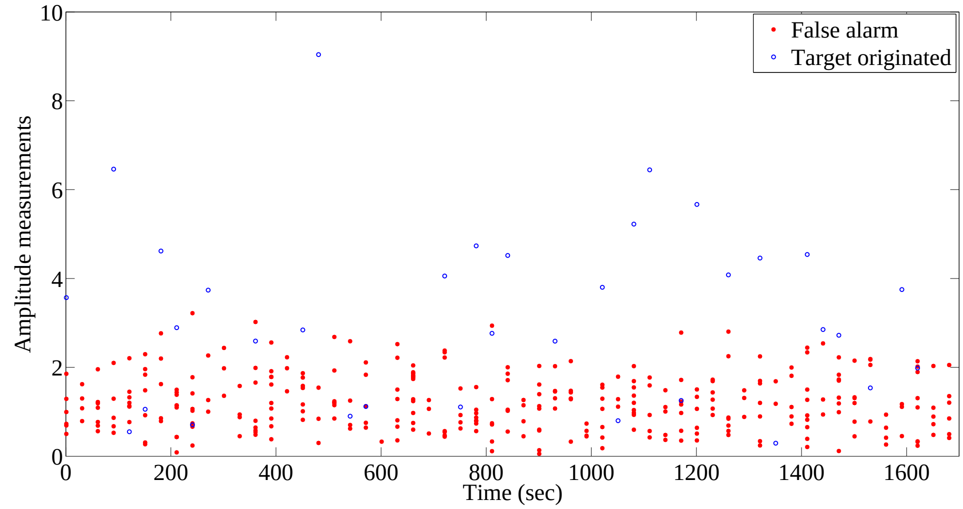

Figure 6.4: Amplitude measurements, Scans = 56, $T = 30$ (s)

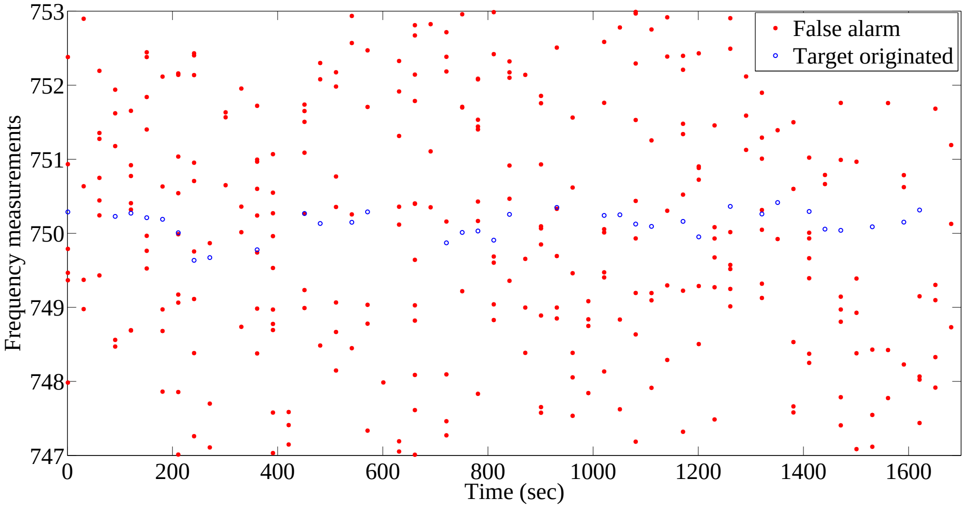

Figure 6.5: Frequency measurements, Scans = 56, $T = 30$ (s)

For a single target estimates a SML-PDA estimator of order 3 is used with 20 knots for positions, 6 knots for velocities and 6 knots for frequency (for narrowband sonar case). The target initial estimate is average over 100 MCR. The spline log likelihood ratio (SLLR) has many local maximum and a global maximum. The system must have the property of observability [2] and it is achieved by manoeuvering the sensor platform as shown in Figure (6.2). Note, that the SLLR contains many local maximum and a its a highly non-convex function. The spline interpolation method as described in Section (3.2) is used to represent the LLR. The knot elements can be determined using uniform knot selection method as [17]. The optimal knot selection, which is an iterative method proposed in [17], can be used to creates a knot vector that is optimal respect to a given parametrization. This method does not consider the location of the input/control points for the optimization. The Matlab function *optknt* can be used to produce this knot vector. Also interactively can shape the B-spline LLR by adding knot elements as in Section (3.7.7) which can increase the number of control points. When control points are moved, the level of continuity at the knots can change (increase or decrease); hence after modification is completed knot removal as in Section (3.7.8) can be invoked in order to obtain the most compact representation of the SLLR. Knot removal can also be used to remove unnecessary knots and note that the first and the last knot elements are not removed. This method can be used for all the dimensions.

The SML-PDA estimator can estimate the state by identifying the global maxima of the SLLR surface. The K-means clustering algorithm as used in [58] can be used for state extraction. An alternative is the expectation-maximization based peak extraction approach in [62] or using a convex optimization method. Note, SLLR is a non-convex and using a optimization technique is a challenging problem. Since spline interpolation is used here to represent the LLR the convex optimization comes with a high complexity, but this problem can be solved by using the spline approximation method i.e., Genetic Algorithm as in [23]. Note, spline interpolation method is only considered here. The global maximum of SLLR or the state estimation can be easily determined by finding the knot element with a higher distribution. This can be determined by using the matlab algorithm from the Table (6.1) for a 2-D scenario. Using the algorithm in Table (6.1) can determine the state easily and the accuracy of the state estimate depends on the number of knot selection. On the Table (6.1) the X(loc) and Y(loc) denote the estimated position and velocity. State estimate validation as described in [2] is performed to validate the true target present.

```
M=fnplt(SLLR);
X = M{1};
Y = M{2};
Z = M{3};
maxV = max(max(Z));
loc = find(Z == maxV);
x = [X(loc)  Y(loc)]';
```

Table 6.1: State estimation algorithm for 2-D scenario

The initial state estimate is averaged over 100 MCR and the knot selection influences the performance of the SML-PDA algorithm. As shown in Figure (6.6) for bearing-only scenario as the knot selections are increased for positions and velocities then the root means square error (RMSE) is reduced, but with a higher complexity. Considering the time of complexity and the RMSE of the SML-PDA algorithm can choose $n_{Pk} = 25$, $n_{Vk} = 7$ as the sub-optimal knot selections for positions and velocities in X and Y directions. Note, as shown in Figure (6.6) with amplitude information

(AI) the SML-PDA algorithm is performed well in terms of RMSE, but with slightly higher complexity.

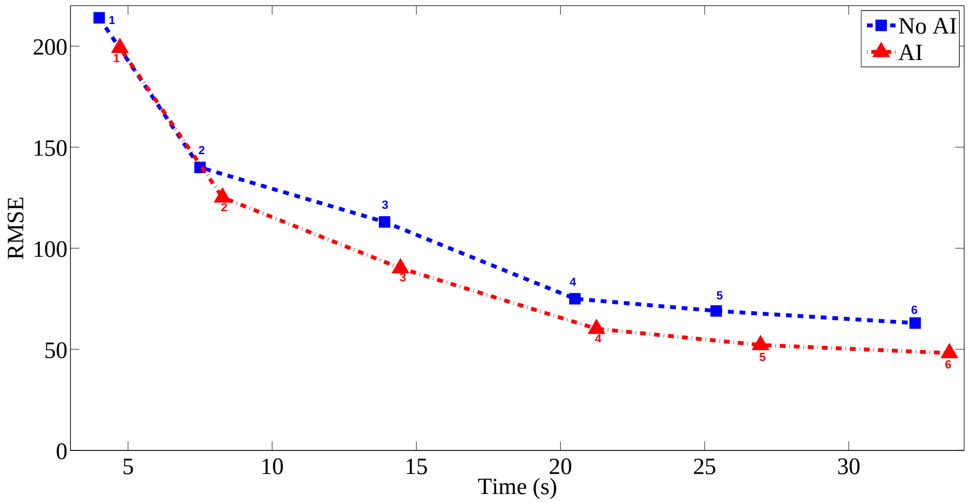

Figure 6.6: Average performance from 100 runs vs. number of knots for bearing-only case, $n_{Pk} = 10$, $n_{Vk} = 4$: 1, $n_{Pk} = 15$, $n_{Vk} = 5$: 2, $n_{Pk} = 20$, $n_{Vk} = 6$: 3, $n_{Pk} = 25$, $n_{Vk} = 7$: 4, $n_{Pk} = 30$, $n_{Vk} = 8$: 5, $n_{Pk} = 35$, $n_{Vk} = 9$: 6, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots.

Next the performance of the SML-PDA algorithm in terms of knot selection is evaluated for the narrowband sonar case. As shown in Figure (6.7) the performance of SML-PDA algorithm in terms of RMSE is improved by increasing the knot selection, but with higher complexity. As shown in Figures (6.6) and (6.7) the performance of SML-PDA algorithm for the bearing-only scenario with AI and the narrowband sonar scenario with out AI have similar outcomes, but with dissimilar complexities. In sonar scenario the SML-PDA algorithm is performed well with AI. The optimal knot selection can be above $n_{Pk} = 25$, $n_{Vk} = 7$, and $n_{fk} = 7$ if there is no complexity issues. Note, in the sonar scenario the SML-PDA algorithm's complexity much higher compare to the bearing-only case due to the sonar case's dimension of the state space increased to 5 compared to bearing-only case (four dimensions). As shown in Figures (6.6) and (6.7) the SML-PDA algorithm performance is affected by the dimensions of the state space.

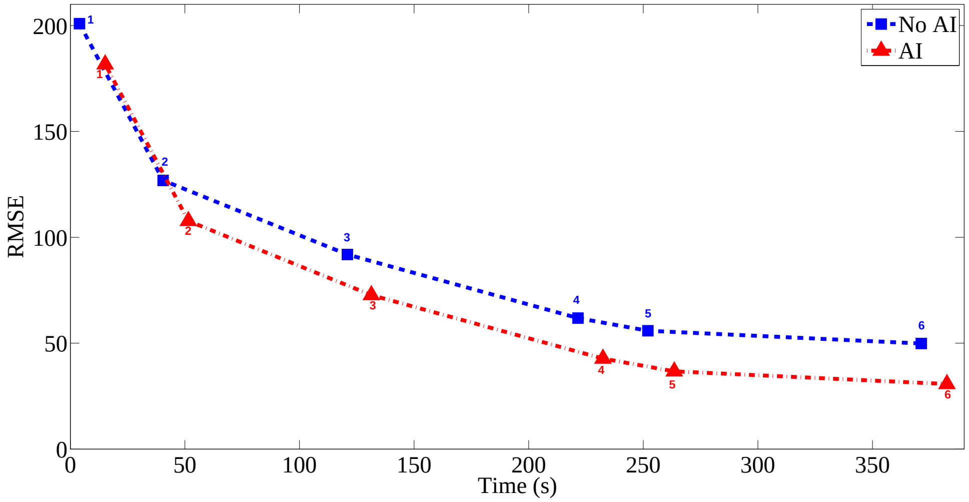

Figure 6.7: Average performance from 100 runs vs. number of knots for narrowband sonar case, $n_{Pk} = 10$, $n_{Vk} = 4$, $n_{fk} = 4$: 1, $n_{Pk} = 15$, $n_{Vk} = 5$, $n_{fk} = 5 : 2$, $n_{Pk} = 20$, $n_{Vk} = 6$, $n_{fk} = 6$: 3, $n_{Pk} = 25$, $n_{Vk} = 7$, $n_{fk} = 7$: 4, $n_{Pk} = 30$, $n_{Vk} = 8$, $n_{fk} = 8$: 5, $n_{Pk} = 35$, $n_{Vk} = 9$, $n_{fk} = 9$: 6, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, n_{fk}: number of frequency knots.

The proposed SML-PDA algorithm's performance is compared with a regular ML-PDA algorithm. The parameter value of ML-PDA's LLR is determined by using the Directed Subspace Search algorithm and followed by a validation process as in [8]. The scalar value of ML-PDA LLR is calculated for set of grid points and the maximum LLR value is determined by using optimization algorithm [8]. Here, LLR's global maximum is a converged parameters.

As shown in the Tables (6.2), (6.3), (6.4) and (6.5) for different SNR values the

RMSE values and the initial target state values were calculated. Clearly, the SML-PDA algorithm performed well in terms of initializing the target state with lower RMSE values. Note, DSS method of ML-PDA estimator has to overcome the non-convex optimization problem and reaching a optimal solution is a challenging, which leads to poor performance. Also, the performance of ML-PDA algorithm degraded by the initial point selection. The SML-PDA algorithm does not use optimization. The global maximum is determined by finding the knot element with a high LLR value respect to a dimension and this can be applied to all the dimensions. The performance efficiency of the SML-PDA algorithm is mostly related to the optimal knot selections. Note, as the SNR increased (ξ fixed) P_d is also increased, which lead both the SML-PDA and the ML-PDA algorithms to perform well. The performance of both algorithms are compared against the Cramer-Rao lower bound (CRLB) [2] for all the scenarios. As shown in Tables (6.2), (6.3), (6.4) and (6.5) the SML-PDA algorithm performance for all the scenarios can be compared to the CRLB results.

Filters	SNR	RMSE	CRLB	Estimated Initial State	CPU Time
	(dB)	(m)	(m)	[m m/s m m/s]	(sec)
SML-PDA	3	540	530	[4625 8.52 5330 8.62]	9.52
	6	461	449	[5300 8.55 4760 11.25]	9.56
	9	345	339	[4795 11.25 4790 10.77]	8.56
	12	245	241	[5180 8.80 5170 10.51]	8.56
ML-PDA	3	605	530	[4562 7.42 4552 7.10]	11.82
	6	520	449	[5400 12.35 4612 8.25]	12.52
	9	401	339	[4692 11.88 4689 11.52]	10.59
	12	302	241	[4780 11.53 5230 10.85]	13.01

Table 6.2: Average performance metrics from 100 Monte Carlo run, Bearing-only scenario without AI, $n_{Pk} = 20$, $n_{Vk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots.

Filters	SNR	RMSE	CRLB	Estimated Initial State	CPU Time
	(dB)	(m)	(m)	[m m/s m m/s]	(sec)
SML-PDA	3	490	482	[5290 9.02 5260 10.85]	14.70
	6	400	394	[4780 10.75 4790 9.22]	14.59
	9	305	297	[5155 9.55 4825 10.57]	14.56
	12	205	194	[5100 10.11 5150 9.77]	14.65
ML-PDA	3	550	482	[4600 12.05 4602 7.90]	18.44
	6	470	394	[5350 8.12 5292 9.05]	20.35
	9	350	297	[5275 10.83 5229 10.92]	19.41
	12	275	194	[5180 10.75 4830 9.23]	18.22

Table 6.3: Average performance metrics from 100 Monte Carlo run, Bearing-only scenario with AI, $n_{Pk} = 20$, $n_{Vk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots.

As shown in Tables (6.2) and (6.3) for bearing-only scenario the SML-PDA algorithm is performed well in terms of RMSE and the computational complexity compare to the ML-PDA algorithm. As the SNR value increased both algorithms performed well and with amplitude information both algorithms performance efficiencies improved as shown in Table (6.3). For the narrowband sonar case, as shown in Tables (6.4) and (6.5) the SML-PDA algorithm performed well in terms of RMSE with and without AI, but with higher complexity compare to the ML-PDA algorithm. The performance of both algorithms for bearing-only with AI scenario and narrowband sonar case without AI scenario are the same in terms of RMSE, but with dissimilar complexities. Using the SML-PDA algorithm as shown in Tables (6.2), (6.3), (6.4) and (6.5) can initialize a target with high efficiency for the bearing-only and the narrowband sonar target tracking scenarios. Both scenarios are affected by high clutter or low observable problems. As shown in Tables (6.4) and (6.5) the SML-PDA algorithm is affected by the curse of dimensionality and compare to bearing-only scenario with four dimensions, the narrowband sonar scenario has one more dimension. Increased state dimensions do not affect the target state initialization or RMSE values, but the computational time. Search and rescue operations normally involve in a longer period, but once the victim/object/target is identified the states of the victim/object/target has to be initialized with high accuracy. The SML-PDA algorithm can be used for this kind of scenarios. Not only the SML-PDA algorithm is affected by the curse of dimensionality, but also by the optimal knot selection process. Knot adding and removing as in Sections (3.7.7) and (3.7.8) are used to get the most compact representation of the SLLR, which lead to high complexity.

Filters	SNR (dB)	RMSE (m)	CRLB (m)	Estimated Initial State [m m/s m m/s Hz]	CPU Time (sec)
SML-PDA	3	487	485	[5289 9.05 4740 9.15 760.27]	122.25
	6	402	395	[5210 10.76 5201 10.72 759.87]	121.35
	9	300	297	[5154 9.57 5160 10.59 741.52]	120.56
	12	200	196	[4900 10.04 5150 9.78 742.4]	122.25
ML-PDA	3	549	485	[4603 8.05 5402 12.90 765.45]	30.87
	6	475	395	[4652 8.72 4722 11.23 760.54]	29.58
	9	345	297	[5274 9.73 5230 9.06 740.15]	29.74
	12	274	196	[4820 9.72 5160 9.73 742.52]	28.85

Table 6.4: Average performance metrics from 100 Monte Carlo run, Narrowband Sonar scenario without AI, $n_{Pk} = 20$, $n_{Vk} = 6$, $n_{fk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, n_{fk}: number of frequency knots.

Filters	SNR (dB)	RMSE (m)	CRLB (m)	Estimated Initial State [m m/s m m/s Hz]	CPU Time (sec)
SML-PDA	3	450	435	[4760 9.35 4765 9.22 756.52]	135.55
	6	370	354	[4801 9.52 5175 10.75 753.25]	138.45
	9	275	274	[5100 9.75 5100 10.67 747.45]	137.45
	12	185	182	[5080 9.727 4910 10.55 748.25]	138.66
ML-PDA	3	505	435	[4680 11.65 5354 8.25 756.45]	34.22
	6	410	354	[4710 11.12 4750 9.32 754.25]	34.00
	9	300	274	[5250 10.53 5145 9.55 746.48]	33.25
	12	215	182	[5160 10.40 4865 9.63 747.25]	34.02

Table 6.5: Average performance metrics from 100 Monte Carlo run, Narrowband Sonar scenario with AI, $n_{Pk} = 20$, $n_{Vk} = 6$, $n_{fk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, n_{fk}: number of frequency knots.

Track acceptance is evaluated base on [31] for the ML-PDA algorithm and for the SML-PDA algorithm the evaluation technique is followed as described in Section (6.2). As shown in Figure (6.8) as the SNR increased both algorithms are performed well in terms of track acceptance. The ideal value of the track acceptance is 100 due to only 100 Monte Carlo runs were used. The SML-PDA algorithm is performed well as illustrated in the Figure (6.8) for both bearing-only and narrowband sonar scenarios. Note, the SML-PDA algorithm is used an optimal knot section to capture the exact replica of the SLLR, which leads the SML-PDA algorithm to perform well in terms of track acceptance, but with a higher computational complexity. A similar track acceptance outcomes were observed for the bearing-only case with amplitude information and the the narrowband sonar case without the amplitude information.

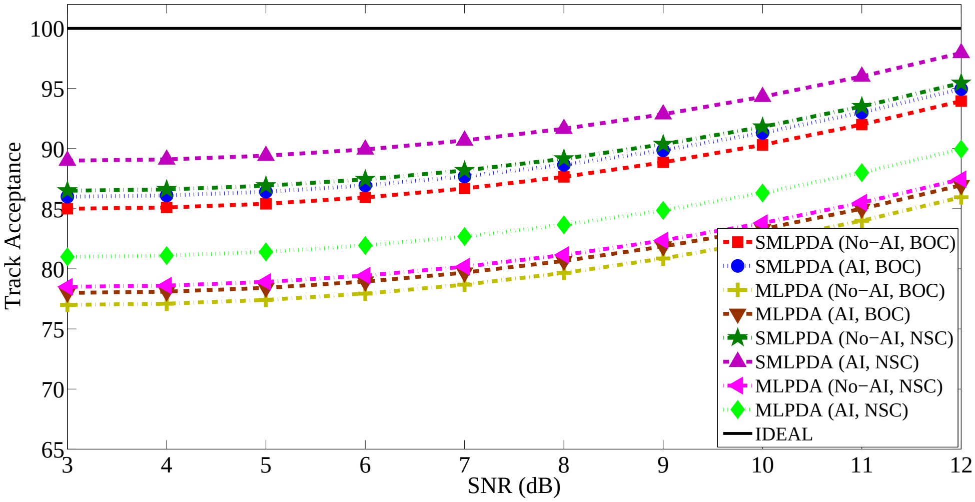

Figure 6.8: Track Acceptance vs. SNR, AI: Amplitude information, BOS: Bearing-only scenario, NSC: Narrowband sonar scenario, MCR=100.

Chapter 7

Geolocating Multitarget in High Clutter Using the B-Spline Based Joint Maximum-Likelihood Probabilistic Data Association Algorithm

7.1 Introduction

The geolocation problem can be solved by well known techniques such as angle of arrival (AOA), time of arrival (TOA), time difference of arrival (TDOA) and frequency difference of arrival (FDOA). However, the performances of these techniques are degraded with false alarm. For the false alarm or low signal to noise ratio (SNR) scenario the Maximum-Likelihood Probabilistic Data Association (ML-PDA) algorithm can be used to geolocate targets. Commonly the ML-PDA algorithm is used in very low observable (VLO) target tracking problems. Recently proposed B-Spline based Maximum-Likelihood Probabilistic Data Association (SML-PDA) algorithm performed well in passive track initilization/geolocation problems for a single target scenario with very high false alarm. Note, the SML-PDA algorithm is applicable only to a single target tracking problem. In this paper geolocate multitarget in very low observable problems a B-Spline based Joint Maximum-Likelihood Probabilistic Data Association (SJML-PDA) algorithm is proposed. The formulation supports a five dimensional passive narrowband target geolocation problems with amplitude information. The target geolocation is determined by finding the global maximum in the joint log-likelihood ratio (JLLR) followed by the track validation process. Simulation results are presented to demonstrate the effectiveness of the new algorithm.

7.2 SJML-PDA Algorithm

The proposed SJML-PDA algorithm's derivation details are described in this section. Formulations are based on a 5D narrowband sonar scenario. The kinematic, frequency and amplitude measurements are available. Kinematic measurements are angle-only measurements and as in [31], the envelope output of the detector is the amplitude. Consider a multitarget geolocation problem in a surveillance region with n targets with low SNR. The multitarget $\mathbf{x}_r \triangleq [\mathbf{x}_r^1, \ldots, \mathbf{x}_r^n]' \in \mathcal{E}_s$ already present at the reference time r in the multidimensional state space $\mathcal{E}_s$. The ℓ-th target state can be defined as $\mathbf{x}_r^\ell \triangleq [x_\ell(i) \quad \dot{x}_\ell \quad y_\ell(i) \quad \dot{y}_\ell \quad \gamma_\ell]'$, $\|\mathbf{x}_r^\ell\| = \eta = 5$ and γ_ℓ, $(x_\ell(i), y_\ell(i))$ and $(\dot{x}_\ell, \dot{y}_\ell)$ are the ℓ-th target's constant unknown emitted frequency, positions and constant velocities, respectively. The multitarget dynamic states evolve according to (2.71). All of the n targets are present during the surveillance in all of the frames and ℓ-th target detected with the probability P_d^ℓ. Across the data frame the detections are independent. The subsequent measurement model with the P_d^ℓ can stimulate observations from multitarget state space $\mathcal{E}_s$. The measurement in the observation space $\mathcal{E}_o$ can also originate from the false alarms or clutter. The false alarms are uniformly distributed in the surveillance region with the volume U. Across the frames the number of false detection is independent and Poisson distributed respect to probability mass function $\mu_f(m)$, with spatial density λ.

The angle-only cumulative measurements $\mathbf{Z}$ along with the measurements from frequency and amplitude information (AI) from N_w frames can be expressed as

$$\mathbf{Z} \triangleq \{Z(i)\}_{i=1}^{N_w} \in \mathcal{E}_o \tag{7.1}$$

where $\mathcal{E}_o$ denotes the measurement space and $Z(i)$ denotes all the observations in a single data-frame at time i defined as

$$Z(i) \quad \triangleq \quad \{\beta_{ij}, \psi_{ij}, a_{ij}\}_{j=1}^{m_i} \in \mathcal{E}_o \tag{7.2}$$

where β_{ij}, ψ_{ij} and a_{ij} denote the bearing, frequency and amplitude measurements from target-originated and noise-only originated, m_i denotes the total number of measurement per scan.

The spline likelihood probability density function (SLPDF) of a ℓ-th target-originated measurement is defined as

$$\begin{aligned} p(\beta^{\ell}_{ij}, \psi^{\ell}_{ij}|\mathbf{x}^{\ell}_{r}) &\approx \sum_{k_1}^{n_s}\cdots\sum_{k_\eta}^{n_s} \mathbb{P}_{k_1,\ldots,k_\eta} B_{k_1,p,\mathbf{t}_i^{\ell,1}}(\beta^{\ell}_{ij}, \psi^{\ell}_{ij}|\mathbf{x}^{\ell}_{r}(1))\ldots B_{k_\eta,p,\mathbf{t}_i^{\ell,\eta}}(\beta^{\ell}_{ij}, \psi^{\ell}_{ij}|\mathbf{x}^{\ell}_{r}(\eta)) \\ &= \mathbf{B}(\beta^{\ell}_{ij}, \psi^{\ell}_{ij}|\mathbf{x}^{\ell}_{r}) \end{aligned} \tag{7.3}$$

where ℓ-th target state dimension is denoted by η, $\ell = \{1,\ldots,n\}$, $i \in \{1,\ldots,N_w\}$, $j \in \{1,\ldots,m_i\}$ and $\mathfrak{k} = k_1,\ldots,k_\eta$. The number of knot elements for all the dimensions is the same (τ) and the p denotes the spline order. The ℓ-th target η dimensional knot $\mathbf{t}^{\ell}_{i} = \{\mathbf{t}^{\ell,1}_{i},\ldots,\mathbf{t}^{\ell,\eta}_{i}\}$ is an $n \times \tau$ array. Each row vector of $\mathbf{t}^{\ell}_{i}$ consists of a set of knot elements $\mathbf{t}^{\ell,l}_{i} = \{t^{\ell,l}_{1,i},\ldots,t^{\ell,l}_{\tau,i}\}$ where $l = 1,\ldots,\eta$. The η dimensional control point or coefficient matrix is denoted by $\mathbb{P}_{\mathfrak{k}}$ and the number of control points is denoted by n_s. For each dimension the same number of control points are used. Note that the number of knots must be greater than the number of control points. For brevity SLPDF of ℓ-th target-originated measurement is denoted by $\mathbf{B}(\beta^{\ell}_{ij}, \psi^{\ell}_{ij}|\mathbf{x}^{\ell}_{r})$. The false measurements are distributed uniformly in the desired surveillance region. For the bearing-only measurement scenario $\beta_{ij} \sim U_\beta[\theta_1, \theta_2]$, where θ_1 and θ_2 are denote the angles and for frequency measurement $\psi_{ij} \sim U_\gamma[\Omega_1, \Omega_2]$, where Ω_1 and Ω_2 are frequencies. Using the tensor product of B-Spline can obtain the uniform distribution for multidimensional state problem [53].

The amplitude information can be obtained as in [31] for any scenario. Generally, a number of detections are made by the radar/sonar at every sampling time and must decide which measurements are likely to have originated from the target. The estimator operates in the presence of false alarm since more than one measurement can satisfy the validation criteria. As described in [31] the output of the envelope detector a is a Rayleigh distributed. The output PDF of envelope detector is $p_1(a_{ij}^{\ell})$ if originated from the true ℓ-th target or $p_0(a_{ij})$ if the output signal originated from noise-only. Note, a suitable detection threshold ξ must be used for detection and depends on the SNR as well as must satisfy the selection of P_d^{ℓ} and P_{fa} values. The amplitude likelihood is incorporated into the likelihood function as ratio.

The amplitude likelihood ratio for the ℓ-th target can be defined as

$$\rho_{ij}^{\ell} = \frac{p_1(a_{ij}^{\ell}|\xi, H_\ell)}{p_0(a_{ij}|\xi)} \tag{7.4}$$

where $p_1(a_{ij}^{\ell}|\xi, H_\ell)$ is the amplitude probability density function (PDF) that originated from the ℓ-th target, H_ℓ denotes the ℓ-th hypothesis, $p_0(a_{ij}|\xi)$ denotes the amplitude PDF of the validated measurements originated from false alarm, a_{ij} and ξ denote the amplitude and the detector threshold in each measurement cell, respectively.

The spline amplitude likelihood probability density function (SAL-PDF) of those originated from the target can be defined as

$$\begin{aligned} p_1(a_{ij}^{\ell}|\xi, H_\ell) &= \frac{a_{ij}^{\ell}}{P_d^{\ell}(1+\mathrm{SNR}^{\ell})} \exp\left(-\frac{(a_{ij}^{\ell})^2}{2(1+\mathrm{SNR}^{\ell})}\right), \qquad a_{ij}^{\ell} > \xi \\ &\approx \sum_{\varsigma_1}^{n_s} \mathbb{P}_{\varsigma_1} B_{\varsigma_1,p,\mathbf{t}_i^{\ell,1'}}(a_{ij}^{\ell}|\xi) \\ &= \mathbf{B}_1(a_{ij}^{\ell}|\xi) \end{aligned} \tag{7.5}$$

where $i \in \{1,\ldots,N_w\}$, $j \in \{1,\ldots,m_i\}$, $\mathbf{B}_1(a_{ij}^{\ell}|\xi)$ denotes the SAL-PDF of those originated from the target, $\mathbf{t}_i^{\ell,1'} = \{t_{1,i}^{\ell,1'},\ldots,t_{\tau,i}^{\ell,1'}\}$ is the set of knot elements of the splines and $\mathbb{P}_{\varsigma_1}$ denotes the ς_1-th control point of the SAL-PDF. The number of knots is denoted by τ and the number of control points are denoted by n_s.

The SAL-PDF of the validated measurements that originated from false alarm can be defined as

$$\begin{aligned} p_0(a_{ij}|\xi) &= \frac{a_{ij}}{P_{fa}} \exp\left(-\frac{a_{ij}^2}{2}\right), \quad a_{ij} > \xi \\ &\approx \sum_{\varsigma_0}^{n_s} \mathbb{P}_{\varsigma_0} B_{\varsigma_0, p, \mathbf{t}_i^{0'}}(a_{ij}|\xi) \\ &= \mathbf{B}_0(a_{ij}|\xi) \end{aligned} \tag{7.6}$$

where $\mathbf{B}_0(a_{ij}|\xi)$ denotes the SAL-PDF of those originated from false alarm, $\mathbf{t}_i^{0'} = \{t_{1,i}^{0'}, \ldots, t_{\tau,i}^{0'}\}$ is the set of knot elements of the splines and $\mathbb{P}_{\varsigma_0}$ denotes the ς_0-th control point of the SAL-PDF. The number of knots is denoted by τ and the number of control points are denoted by n_s . The B-splines [12,53] can be used to represent any distribution functions and in this case, the Rayleigh distributed functions $p_1(a_{ij}^\ell|\xi, H_\ell)$ and $p_0(a_{ij}|\xi)$ mean and variance values can be determined and using these values as described in [12,53] can select the knot vectors $\mathbf{t}_i^{\ell,1'}$ and $\mathbf{t}_i^{0'}$.

The spline amplitude likelihood ratio for the ℓ-th target can be defined as

$$\mathbf{B}_{ij}^\ell = \frac{\mathbf{B}_1(a_{ij}^\ell|\xi)}{\mathbf{B}_0(a_{ij}|\xi)} \tag{7.7}$$

The multitarget likelihood $p(Z(i)|\mathbf{x}_r)$ described in (2.92) can be written for a general n targets case as

$$p(Z(i)|\mathbf{x}_r) \;=\; \left(\prod_{e=1}^{n}(1-P_d^e)\right) L_i^0 + \sum_{q=1}^{n}\left(\sum_{k=1}^{\begin{pmatrix} n \\ q \end{pmatrix}} P_d^{g_q(k)} \left(\prod_{\substack{e=1 \\ e\neq g_q(k)}}^{n} (1-P_d^e)\right) L_i^{g_q(k)}\right) \tag{7.8}$$

where n denotes the number of target and using $\begin{pmatrix} n \\ q \end{pmatrix} = \frac{n!}{(n-q)!q!}$ all the combinatorial sets of targets can be determined. Note sets are independent and exclusive. The likelihood where all the measurements are false alarm is given by $\left(\prod_{e=1}^{n}(1-P_d^e)\right) L_i^0$ and $g_q(\cdot)$ is dependent of q where q denotes the number of available true-target. For e.g., if the total target is 3,

$$\begin{aligned} g_{(q=1)} &= [\{1\},\{2\},\{3\}] \\ g_{(q=2)} &= [\{1,2\},\{1,3\},\{2,3\}] \\ g_{(q=3)} &= [\{1,2,3\}] \end{aligned} \tag{7.9}$$

and for $g_{(q=2)}(k=1) = \{1,2\}$, which indicates targets 1 and 2 are detected. Note $\|g_{(q=2)}\| = 3$ and $\|g_{(q=2)}(k=1)\| = 2$. The detected probability is determined as

$$P_d^{g_{(q=2)}(k=1)} \;=\; P_d^{\{1,2\}} \;=\; P_d^1 P_d^2 \tag{7.10}$$

The possible target detections terms, $L_i^{g_q(k)}$ are determined by associating a certain observation to each detected target and the rest of the observations are considered as false detection. The possible target detections terms are determined as

$$L_i^{g_q(k)} = \frac{\mu_f(m_i - \varpi_k)}{U^{m_i-\varpi_k}\prod_{v=1}^{\varpi_k}(m_i + \varpi_k - v)} \prod_{j=1}^{m_i} p_0(a_{ij}|\xi) \overbrace{\sum_{j_1=1}^{m_i} \cdots \sum_{\left(\begin{array}{c} j_{\varpi_k=1} \\ j_{\varpi_k} \neq j_1 \\ \vdots \\ j_{\varpi_k} \neq j_{\varpi_k-1} \end{array}\right)}^{m_i}}^{\varpi_k}$$

$$\cdot \prod_{v \in g_q(k)}^{\varpi_k} p\left(\beta_{ij_v}, \psi_{ij_v}|\mathbf{x}_r^v\right)\rho_{ij_v}^v \tag{7.11}$$

where $\varpi_k = \|g_q(k)\|$ and $k \neq 0$. If the measurement is only from false detections ($\|g_q(k)\| = 0$) then the L_i^0 can be defined as

$$L_i^0 = \frac{\mu_f(m_i)}{U^{m_i}} \prod_{j=1}^{m_i} p_0(a_{ij}|\tau) \tag{7.12}$$

The joint likelihood function of N_w frames can be determined as

$$p(\mathbf{Z}|\mathbf{x}_r) = \prod_{i=1}^{N_w} p(Z(i)|\mathbf{x}_r) \tag{7.13}$$

and the likelihood of all noise-only originated measurements is given as

$$\prod_{i=1}^{N_w} \left(\prod_{e=1}^{n} (1 - P_d^e) \right) L_i^0 \tag{7.14}$$

Then the spline joint log-likelihood ratio (SJLLR) can be determined by dividing (7.13) by (7.14) and taking the logarithmic of the result as

$$
\begin{aligned}
&\Psi(\mathbf{Z}|\mathbf{x}_r) = \\
&\sum_{i=1}^{N_w} \ln \left[1 + \sum_{q=1}^{n} \left(\sum_{k=1}^{\left(\begin{array}{c} n \\ q \end{array}\right)} \left(\frac{P_d^{g_q(k)}}{\lambda^q P_{nd}^{g_q(k)}} \overbrace{\sum_{j_1=1}^{m_i} \cdots \sum_{\left(\begin{array}{c} j_{\varpi_k=1} \\ j_{\varpi_k} \neq j_1 \\ \vdots \\ j_{\varpi_k} \neq j_{\varpi_k-1} \end{array}\right)}^{m_i}}^{\varpi_k} \prod_{v \in g_q(k)}^{\varpi_k} p\left(\beta_{ij_v}, \psi_{ij_v}|\mathbf{x}_r^v\right) \rho_{ij_v}^v \right)\right)\right] \\
&= \sum_{i=1}^{N_w} \ln \left[1 + \sum_{q=1}^{n} \left(\sum_{k=1}^{\left(\begin{array}{c} n \\ q \end{array}\right)} \left(\frac{P_d^{g_q(k)}}{\lambda^q P_{nd}^{g_q(k)}} \overbrace{\sum_{j_1=1}^{m_i} \cdots \sum_{\left(\begin{array}{c} j_{\varpi_k=1} \\ j_{\varpi_k} \neq j_1 \\ \vdots \\ j_{\varpi_k} \neq j_{\varpi_k-1} \end{array}\right)}^{m_i}}^{\varpi_k} \prod_{v \in g_q(k)}^{\varpi_k} \mathbf{B}(\beta_{ij_v}, \psi_{ij_v}|\mathbf{x}_r^v)\mathbf{B}_{ij_v}^v \right)\right)\right] \\
&= \mathbf{B}_\Lambda\left(\mathbf{Z}|\mathbf{x}_r\right) \qquad (7.15)
\end{aligned}
$$

where $P_{nd}^{g_q(k)}$ can be determined using (7.10), $P_{nd} = (1 - P_d)$ and $\varpi_k = \|g_q(k)\|$. The SJML-PDA LLR $\mathbf{B}_\Lambda\left(\mathbf{Z}|\mathbf{x}_r\right)$ is a multidimensional spline PDF. Note, tensor product spline construction [12] can be used to build multidimensional splines PDF.

The target estimates $\hat{\mathbf{x}}_r$ from the SJML-PDA algorithm can be determined as

$$\hat{\mathbf{x}}_r \quad = \quad \arg\max_{\mathbf{x}_r} \mathbf{B}_{\Lambda}\left(\mathbf{Z}|\mathbf{x}_r\right), \tag{7.16}$$

The multitarget states can be determined by using any optimization algorithm. Note, (7.16) is a nonlinear nonconvex problem. As in [59] same method can be used to extracted the individual targets sates. The SJML-PDA LLR has many local maximums and only the first n number of local maximums (including the global max) with higher SJML-PDA LLR are taken into consideration. Since the number of targets is already known we can extract the first n highest states from the SJML-PDA LLR by finding the knot with the higher distribution then removing the knot from that locations and looking for the next one. This can be repeated n times. The SJML-PDA algorithm's track estimates have to satisfy the track validations process [31]. A track validation process is applied to see if each estimated target state comes from true-target, no-target or false alarm [2]. Here, same method used in [59] is applied to validate each target status. However, all the targets must hence originated from the true-target for track-validation to be successful. If the track validation unsuccessful then discard the current measurement frame and keep the next frame as initial frame by using a desired size moving window. The same routine can be applied until all the estimated-targets originated from the true-targets.

7.3 Simulation Results

In this section, the proposed SJML-PDA algorithm is used for a practical multitarget geolocation problem to verify its performance compared to the standard JML-PDA algorithm. The selected scenario contains four targets in high clutter or low SNR. The measurements in this scenario also contain amplitude information (AI) and deal with bearing-only underwater multitarget geolocation problems. The selected scenario arises in many practical applications such as submarine tracking or airborne surveillance using a passive radar [29,31,58,57]. The four targets are assumed to be four submarines and the sensor is a narrowband sonar which sits on a moving underwater drone. The movement of the drone ensures the observability of the targets. All the submarines and the drone are in the same plane and the surveillance region is selected to be a 100 km-by-100 km square region in which four VLO targets are present at all time.

The drone and the targets are moving along X-Y plane and the unknown ℓ-th target state $\mathbf{x}_r^\ell$ is denoted as

$$\mathbf{x}_r^\ell \triangleq [x_\ell(i) \;\; \dot{x}_\ell \;\; y_\ell(i) \;\; \dot{y}_\ell \;\; \gamma_\ell]' \tag{7.17}$$

where $(x_\ell(i), y_\ell(i))$, $(\dot{x}_\ell, \dot{y}_\ell)$ and γ_ℓ denote the positions, constant velocities and constant frequency of the ℓ-th target, respectively. Based on a constant velocity model the ℓ-th target motion is defined as

$$\mathbf{x}_r^\ell(i+1) = \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_r^\ell(i) + \begin{bmatrix} T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sigma_x^\ell \\ \sigma_y^\ell \\ \sigma_{\dot{\gamma}}^\ell \end{bmatrix} \tag{7.18}$$

where the ℓ-th target system process noises are zero-mean white Gaussian random variables with standard deviation (s.d) $\sigma_x^\ell = \sigma_x$, $\sigma_y^\ell = \sigma_y$ and $\sigma_{\dot{\gamma}}^\ell = \sigma_{\dot{\gamma}}$, respectively, where $\ell = \{1, 2, 3, 4\}$.

At time i the known sensor kinematic state $\mathbf{x}_s(i)$ given as

$$\mathbf{x}_s(i) \triangleq [\xi_s(i)\ \dot{\xi}_s(i)\ \eta_s(i)\ \dot{\eta}_s(i)]' \tag{7.19}$$

where $(\xi_s(i), \eta_s(i))$ and $(\dot{\xi}_s(i), \dot{\eta}_s(i))$ denote the position and velocities of the sensor.

Table 7.1: Simulation parameters

Parameters	Values
P_{fa}	0.100
ξ (detection threshold)	2.1460
SNR	5 dB, 6 dB, 7 dB, 8 dB
P_d^1, P_d^2, P_d^3, P_d^4	0.5751, 0.6299, 0.6818, 0.7298
σ_w (kinematic measurement noise s.d)	1.5°
σ_γ (frequency measurement noise s.d)	0.0735 Hz
$\sigma_x, \sigma_y, \sigma_{\dot{\gamma}}$ (process noise s.d)	$1.0000e^{-08}$ m/s^2, $1.0000e^{-08}$ m/s^2, $1.0000e^{-08}$ Hz
T (sampling time)	30 s
Targets start time	1 s, 1 s, 1 s, 1 s
$\mathbf{x}_r^1(1)$ (Target 1 initial state)	-2000 m, 30 m/s, 5000 m, 10 m/s, 750 Hz
$\mathbf{x}_r^2(1)$ (Target 2 initial state)	20000 m, -10 m/s, 15000 m, 10 m/s, 755 Hz
$\mathbf{x}_r^3(1)$ (Target 3 initial state)	-1000 m, 20 m/s, 11000 m, 10 m/s, 745Hz
$\mathbf{x}_r^4(1)$ (Target 4 initial state)	10000 m, -15 m/s, 10000 m, -10 m/s, 748Hz
$\mathbf{x}_s$ (initial sensor state)	-7000 m, 10 m/s, -7000 m, 10 m/s
Total time	1680 s
Bearing range U_β	$[-\pi\ \pi]$ rad
Frequency range U_γ	[740 760] Hz
Number of Monte Carlo runs	100
N_w (number of frames)	10

All of the target's initial states and their corresponding initial times are known as in Table (7.1). Each target is detected with the probability of detection P_d^ℓ as in Table (7.1), where $\ell = \{1, 2, 3, 4\}$ and the probability of false alarm $P_{fa} = 0.100$. Using the probability of false alarm as described in [31] the threshold value $\xi = 2.1460$ is determined. As stated in the Table (7.1) with known target-depended SNR (dB) values and the calculated ξ corresponding P_d^ℓ values are calculated.

The ℓ-th target-originated measurements follow the observation model at time i

$$\beta_{ij}^\ell = h_\ell(\mathbf{x}_r^\ell, \mathbf{x}_s(i)) + \omega_i^\ell, \tag{7.20}$$

where ω_i^ℓ is a zero-mean white Gaussian noise with kinematic noise standard deviation $\sigma_\omega^\ell = \sigma_\omega$ degree, where $\ell = \{1, 2, 3, 4\}$ and $h_\ell(\cdot)$ can be determined as

$$\begin{aligned} h_\ell(\mathbf{x}_r^\ell, \mathbf{x}_s(i)) &= \arctan\left(\frac{\mathrm{r}_\eta(i, \mathbf{x}_r^\ell)}{\mathrm{r}_\xi(i, \mathbf{x}_r^\ell)}\right) \\ &= \theta_i(\mathbf{x}_r^\ell) \end{aligned} \tag{7.21}$$

where $\theta_i(\mathbf{x}_r^\ell)$ is the angle as defined in Figure (2.1), which is the true bearing between the target and the platform and $\mathrm{r}_\eta(i, \mathbf{x}_r^\ell), \mathrm{r}_\xi(i, \mathbf{x}_r^\ell)$ are the relative positions. False alarm generated measurement is uniformly distributed in the surveillance region with volume U_β. For each scan 10 number of false alarms N_{fa} are received, with clutter density $\lambda = N_{fa}/(U_\beta U_\gamma) = 0.08/\text{rad.Hz}$.

The ℓ-th target-originated frequency measurements follow the observation model as

$$\psi_{ij}^\ell = \gamma_i(\mathbf{x}_r^\ell) + \nu_i^\ell, \tag{7.22}$$

where $\gamma_i(\mathbf{x}_r^\ell)$ denotes the noise free Doppler shifted frequency defined in (2.80) and

$\nu_i^\ell \sim N(0, (\sigma_\gamma^\ell)^2)$ where ν_i^ℓ is a zero-mean white Gaussian noise with known s.d $\sigma_\gamma^\ell = \sigma_\gamma$ Hz. The frequency originated measurements is uniformly distributed in the surveillance region U_γ. The noise free amplitude as well as the false alarm-originated amplitude measurements are generated by using the Rayleigh distribution and SNR values.

Third order of B-Spline is applied for the SJML-PDA implementation to maintain the continuity of the SJML-PDA LLR and used 35 knots for positions, 6 knots for velocities and 6 knots for frequency. The target initial estimate is average over 100 MCR. The SJML-PDA algorithm performance is evaluated for a narrowband sonar case with and without the amplitude information.

As shown in Figures (7.1), (7.2) and (7.3) the multitarget-originated and noise-only generated bearing, frequency and amplitude measurements can be distinguished from a single Monte Carlo run (MCR). However, the index of the multitarget-originated measurements are not known to the estimator. Note, total number of scans/frame in a MCR is 56 and the sampling time is 30 (s).

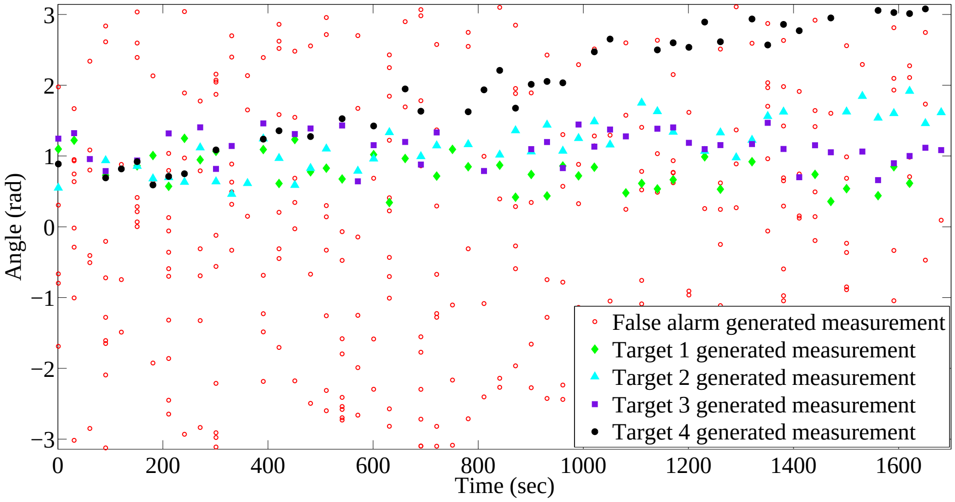

Figure 7.1: Target-originated and false alarm measurements, Scans = 56, $T = 30$ (s)

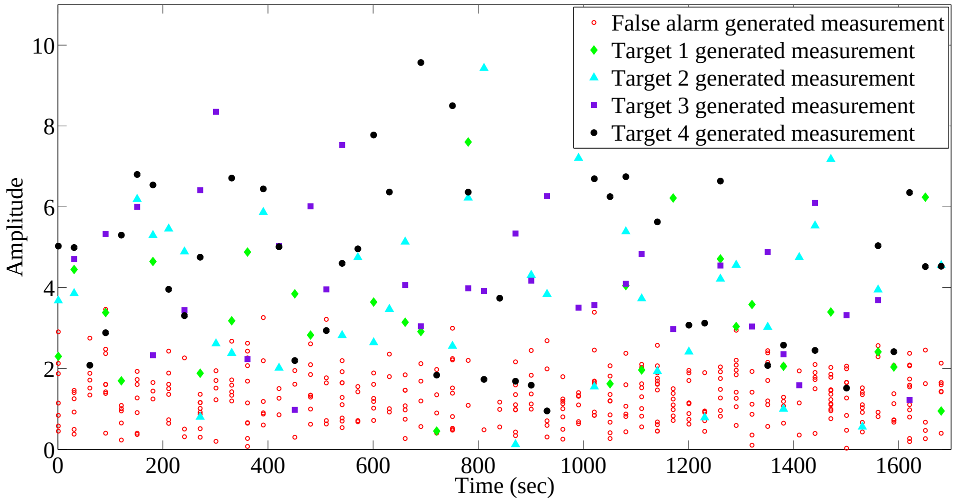

Figure 7.2: Amplitude measurements, Scans = 56, $T = 30$ (s)

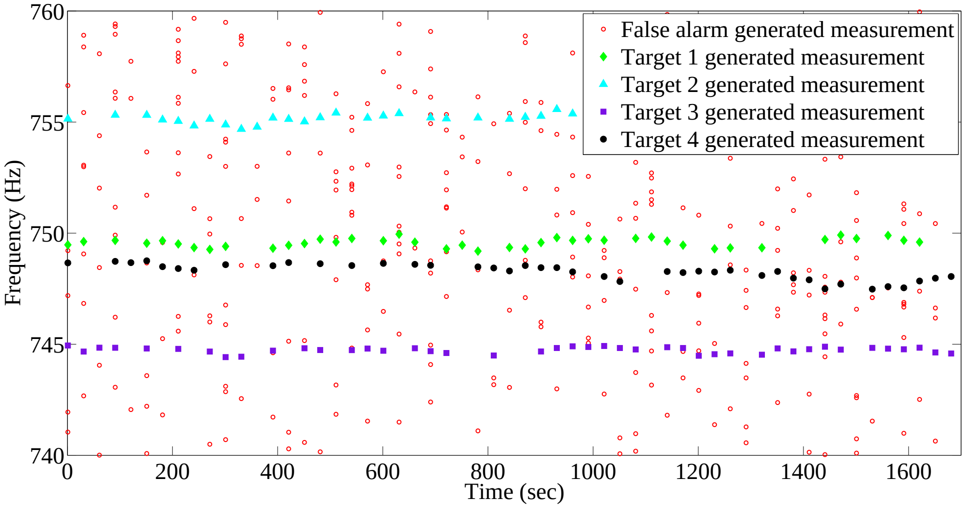

Figure 7.3: Frequency measurements, Scans = 56, $T = 30$ (s)

Knots can be added and removed as described in Sections (3.7.7) and (3.7.8) to produce the exact replica of the multitarget spline joint log-likelihood ratio (SJLLR). The knot elements can be determined by using the uniform knot selection method [17]. Adding new knots will increase the control points and the movement of the control points eventually affect the level of the continuity at the knot. However, can invoke the knot removal to keep the continuity while get best SJLLR representation. The multitarget states are extracted from the SJLLR. Using the known number of targets (in this case 4) the first 4 maximum values of SJLLR can be determine. The knot locations on these maximum values are the states of each targets. Note, that the knot vectors with first 4 higher SJLLR values are the multitarget states [59]. Each extracted target has to go through a track validation process to determine if the each estimated target states originates from true-target, no-target or false alarm [2]. A similar track validation method as described in [59] is used here to validate the origin of each extracted target. If any one of the targets is determined not to have originated from the true-target then the validation processes is considered failed. The geolocation process can be repeated by discarding the initial measurement frame and by taking the second measurement frame as initial frame by using a sliding window frames with the desired size. This process must be repeated until all the extracted targets are determined not to have originated from noise. The geolocated states in Figure (7.4) is averaged over 100 Monte Carlo runs.

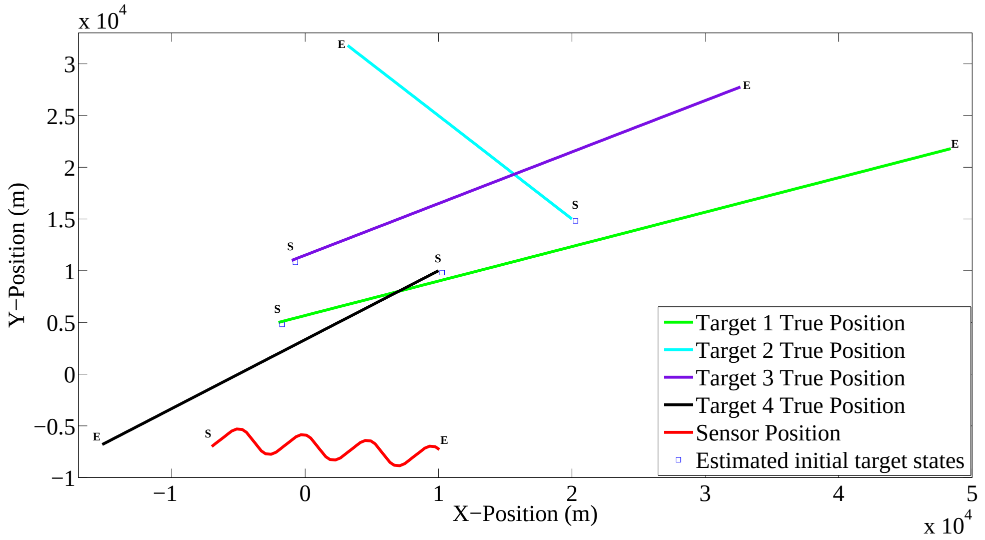

Figure 7.4: Trajectories of true-target, true-platform and estimated initial target states **S**: start, **E**: end.

The performance of the SJML-PDA algorithm is compared with a regular JML-PDA algorithm. The state extraction and track validation of the JML-PDA algorithm follow as in [8, 9]. As shown in Tables (7.2) and (7.3) the position RMSE values and the geolocated target state for each target values were calculated. Similar to the SML-PDA algorithm [59] the SJML-PDA algorithm performed well in terms of geolocating the target state with lower RMSE values. Regular JML-PDA algorithm suffer by non-convex optimization problem. Reaching an optimal solution is hard for the regular

JML-PDA algorithm due to the imperfect selection of the initial search point. The SJML-PDA algorithm does not use optimization, but can use the optimization as described in [23]. From the first 4 knot elements with higher SJLLR values are determined with respect to each dimension. These knot elements are correspond to the state of each target. The performance efficiency of the SJML-PDA algorithm is mostly related to the optimal knot selections. Normally, in Search and Rescue mission (e.g., searching a black-box) time is not a constrain, and can select the knots depending on the scenario, area of the surveillance regions and the desired initial state estimate accuracy. The performance of both algorithms are compared against the Cramer-Rao lower bound (CRLB) [2] for all the scenarios. As shown in Tables (7.2) and (7.3) the SJML-PDA algorithm performance for all the scenarios can be compared to the CRLB results.

Filters	RMSE	CRLB	Estimated Initial State
	(m)	(m)	[m m/s m m/s Hz]
SJML-PDA	510	470	$[-2112.52\ \ 35.45\ \ 5765.25\ \ 8.26\ \ 756]$ $[18965.25\ \ -12.35\ \ 16150.45\ \ 12.35\ \ 761]$ $[-916\ \ 23.75\ \ 10008.79\ \ 12.5\ \ 740]$ $[11070.45\ \ -13.75\ \ 9800.20\ \ -8.95\ \ 743.25]$
JML-PDA	670	470	$[-1790.25\ \ 38.45\ \ 3987.45\ \ 14.25\ \ 760]$ $[22350.25\ \ -14.10\ \ 17256.25\ \ 7.53\ \ 765]$ $[-1630.48\ \ 17.25\ \ 12253.88\ \ 14.36\ \ 751]$ $[8900.10\ \ -17.25\ \ 11545.65\ \ -13.45\ \ 754.12]$

Table 7.2: Average performance metrics from 100 Monte Carlo run, Narrowband Sonar scenario without AI, $n_{Pk} = 35$, $n_{Vk} = 6$, $n_{fk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, n_{fk}: number of frequency knots.

Filters	RMSE (m)	CRLB (m)	Estimated Initial State [m m/s m m/s Hz]
SJML-PDA	390	360	$[-1945.74\ \ 32.02\ \ 5100.10\ \ 9.45\ \ 748.42]$ $[19627.32\ \ -10.89\ \ 15650\ \ 9.60\ \ 756.11]$ $[-988\ \ 21\ \ 11107\ \ 11.01\ \ 743]$ $[10120\ \ -14.90\ \ 10251.22\ \ -9.85\ \ 749.25]$
JML-PDA	450	360	$[-2110\ \ 33\ \ 5100.10\ \ 11.24\ \ 746]$ $[19000\ \ -12\ \ 14001.33\ \ 9.01\ \ 757]$ $[-1100\ \ 24.23\ \ 12125.11\ \ 13.05\ \ 740]$ $[9175.12\ \ -13.88\ \ 10822.45\ \ -9.05\ \ 750]$

Table 7.3: Average performance metrics from 100 Monte Carlo run, Narrowband Sonar scenario with AI, $n_{Pk} = 35$, $n_{Vk} = 6$, $n_{fk} = 6$, n_{Pk}: number of position knots, n_{Vk}: number of velocity knots, n_{fk}: number of frequency knots.

The performance of track validation is illustrated in Figure (7.5) in terms of track acceptance. For simplicity SNR values for all the targets are considered the same and P_d^ℓ follow the same. Track acceptance is evaluated based on the techniques described in [8] for the JML-PDA algorithm and in [59] for the SJML-PDA algorithm. As shown in Figure (7.5) as the SNR increased both algorithms performed well in terms of track acceptance. The ideal value of the track acceptance is 400 since only 100 Monte Carlo runs were used with 4 targets present. Note, the SJML-PDA algorithm is used an optimal knot selection to capture the exact replica of the SJLLR, which leads the SJML-PDA algorithm to perform well. Using the AI both algorithms are performed well.

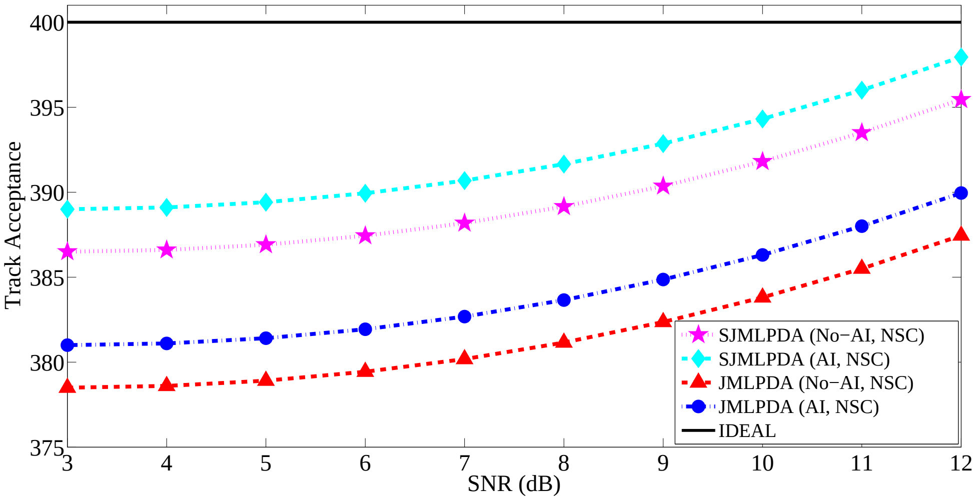

Figure 7.5: Track Acceptance vs. SNR, AI: Amplitude information, NSC: Narrowband sonar scenario, MCR=100.

Chapter 8

Conclusion

In this first part of the thesis, a Spline Probability Hypothesis Density (SPHD) filter implementation was developed by using the B-Splines. The proposed filter is an alternative to the Sequential Monte Carlo, Gaussian Mixture, Gaussian Mixture Particle, Gaussian Mixture Unscented Sequential Monte Carlo, Gaussian Mixture Sequential Monte Carlo, and the Auxiliary Particle implementations of the PHD filter. As shown in Chapter 4, the resulting algorithm can handle linear, non-linear, Gaussian, and non-Gaussian models. The SPHD filter can provide continuous estimates of the probability density function of the system state and it is immune to the degeneracy problem. The SPHD filter can maintain highly accurate tracks by taking advantage of dynamic knot movement, but at the expense of higher computational complexity, which makes it suitable for tracking a few high-value targets under difficult conditions. The SPHD filter performs well with a few knots and provides continuous state estimates for any system, which leads to non-degenerative results.

An extension of SPHD filter supports maneuvering target tracking as described in Chapter 5. The MM-SPHD filter implementation was presented as an alternative to the Sequential Monte Carlo and the Gaussian Mixture MM-PHD filters for maneuvering target tracking problems. The resulting algorithm can handle linear, non-linear, Gaussian and non-Gaussian models. The MM-SPHD filter inherits all of the performance from the SPHD filter. This new filter, which yields accurate results albeit with a higher computational load, is useful in tracking high-value maneuvering targets (e.g., missiles, submarines, ground targets) in the presence of nonlinearity or non-Gaussianity.

B-Splines technique is also applied to very low observable tracking problem as described in Chapters 6 and 7. In chapter 6, a Spline Maximum-Likelihood Probabilistic Data Association (SML-PDA) algorithm implementation was developed for track initialization. The proposed algorithm is an alternative to the standard ML-PDA algorithm. The global maximum of SML-PDA LLR can be determined by finding the knot element with a highest peak. The SML-PDA algorithm can determine the global maximum of ML-PDA LLR with high efficiency in terms of state estimates and low computational complexity, which is good for fast passive track initialization. For higher dimensional target tracking problems the SML-PDA algorithm performs well, albeit with a higher computational complexity, which makes it suitable for tracking a target in search and rescue operations under difficult conditions. As like the precursor of ML-PDA algorithm the newly proposed SML-PDA algorithm performs well with amplitude information.

Chapter 7 addressed the use of B-spline based Joint Maximum-Likelihood Probabilistic Data Association (SJML-PDA) algorithm in a very low observable (VLO) multitarget geolocation problem without the time constraint. The SJML-PDA algorithm is a multitarget extension of the SML-PDA algorithm and it inherits all the capabilities of the SML-PDA algorithm. The proposed SJML-PDA algorithm can be used in search and rescue mission in a very harsh environment i.e., geolocating a flight black-box, mines and submarines in oceans around the world. The multitarget geolocation is determined by finding the global maximum in the joint log-likelihood ratio followed by the track validation process. The multitarget SJML-PDA LLR can be any distribution and can be created by using the tensor product of splines. Using the optimal knot selection the SJML-PDA algorithm can geolocate in very low SNR or high false alarm. As illustrated from the simulations results in Chapter 7 can be observed that the SJML-PDA algorithm works very well with the amplitude information.

Appendix A

Basic Geolocation

The geolocation of a single target in a clean environment by using the TDOA technique is used here to explain the geolocation problem. The TDOA method is used for simplicity. As shown Figure (A.1) this is a multidimensional multisensor geolocation problem with one emitter/source in the surveillance region. The emitter with unknown location $X = [x, y, z]'$ geolocated by the N number of sensors with known locations $X_i = [x_i, y_i, z_i]'$. At t_i the emitted signal arrived at sensor i is given by

$$t_i = t_{em} + \frac{\|X - X_i\|}{\mathfrak{c}} + \nu_i \tag{A.1}$$

where t_i, t_{em} denote the TOA and unknown time of emission, respectively. The speed of the emitted signal in a specific medium is denoted by $\mathfrak{c}$ and ν_i is the TOA measurement noise. The measurement noise is a Gaussian process with zero mean and variance σ_i^2 and independent from other sensors. Note in (A.1) $\|X - X_i\|$ denotes the difference of the location of emitter and the i-th sensor's two-norm.

The TOA can be determined using (A.1) for all N sensors. From (A.1) there are

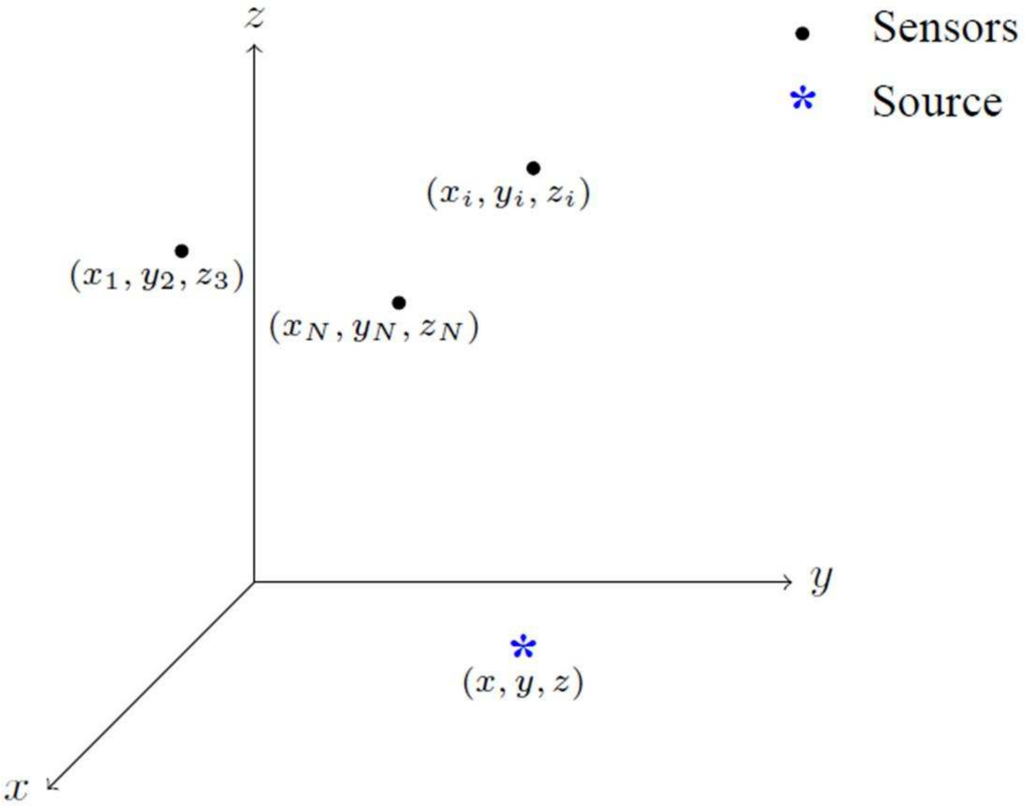

Figure A.1: Geolocation of an emitter by multisensor with no false alarm

four unknowns (t_{em}, x, y, z) to determined. The time of emission t_{em} can be eliminated by using the time difference of arrival (TDOA) technique between two sensors as

$$\begin{aligned}\Delta t_{ij} &= t_i - t_j \\ &= \left(t_{em} + \frac{\|X - X_i\|}{\mathfrak{c}} + \nu_i\right) - \left(t_{em} + \frac{\|X - X_j\|}{\mathfrak{c}} + \nu_j\right) \\ &= \left(\frac{\|X - X_i\|}{\mathfrak{c}} - \frac{\|X - X_j\|}{\mathfrak{c}}\right) + (\nu_i - \nu_j) \\ &= h_{ij}(X) + \nu_{ij} \end{aligned} \tag{A.2}$$

where i and j denote the sensors and the unknowns are (x, y, z) only. For simplicity (A.2) can be written by considering a reference (e.g., sensor 1) then (A.2) can be written as

$$\begin{aligned}\Delta t_{i1} &= t_i - t_1 \\ &= \left(t_{em} + \frac{\|X - X_i\|}{\mathfrak{c}} + \nu_i\right) - \left(t_{em} + \frac{\|X - X_1\|}{\mathfrak{c}} + \nu_1\right) \\ &= \left(\frac{\|X - X_i\|}{\mathfrak{c}} - \frac{\|X - X_1\|}{\mathfrak{c}}\right) + (\nu_i - \nu_1) \\ &= h_i(X) + \nu_{i1} \end{aligned} \tag{A.3}$$

Considering N sensors $(N-1)$ TDOA equations can be written as

$$\begin{bmatrix} \Delta t_{21} \\ \Delta t_{31} \\ \vdots \\ \Delta t_{N1} \end{bmatrix} = \begin{bmatrix} h_2(X) \\ h_3(X) \\ \vdots \\ h_N(X) \end{bmatrix} + \begin{bmatrix} \nu_{21} \\ \nu_{31} \\ \vdots \\ \nu_{N1} \end{bmatrix} \tag{A.4}$$

and (A.4) can be simplified as

$$\Delta t \quad = \quad H(X) + \nu \tag{A.5}$$

Then the emitter location $\hat{X} = (x, y, z)$ can be determined by

$$\hat{X} \quad = \quad \arg\min_{X}(\Delta t - H(X))^{'} Q^{-1}(\Delta t - H(X)) \tag{A.6}$$

where $Q = \mathbb{E}[\nu\nu^{'}]$ denotes the measurement noise covariance matrix. Estimating the $\hat{X}$ is a problem due to (A.6) is a nonlinear nonconvex problem. The TDOA method also can be applied to multitarget geolocating problem as in [43, 68]. Generally using two separate methods the multitarget states can be determined. First the TDOA measurements obtained for each target then each emitter is geolocated by solving (A.6) as described earlier.

As shown in the Figure (A.2) with false alarm the original geolocation of a single-emitter problem becomes more complex. At time t_i each sensor receives measurement from true-emitter and from false alarm. The false alarm generated measurement at t_i is $[x_{c,k}, y_{c,k}, z_{c,k}]_{k=1}^{M_{t_i}}$. The total number of false alarm generated measurement is denoted by M_{t_i}. Using the well known conventional AOA, TOA, TDOA and FDOA methods is difficult to estimate the state of the emitter with false alarm. It becomes a complex problem for geolocating VLO multitarget.

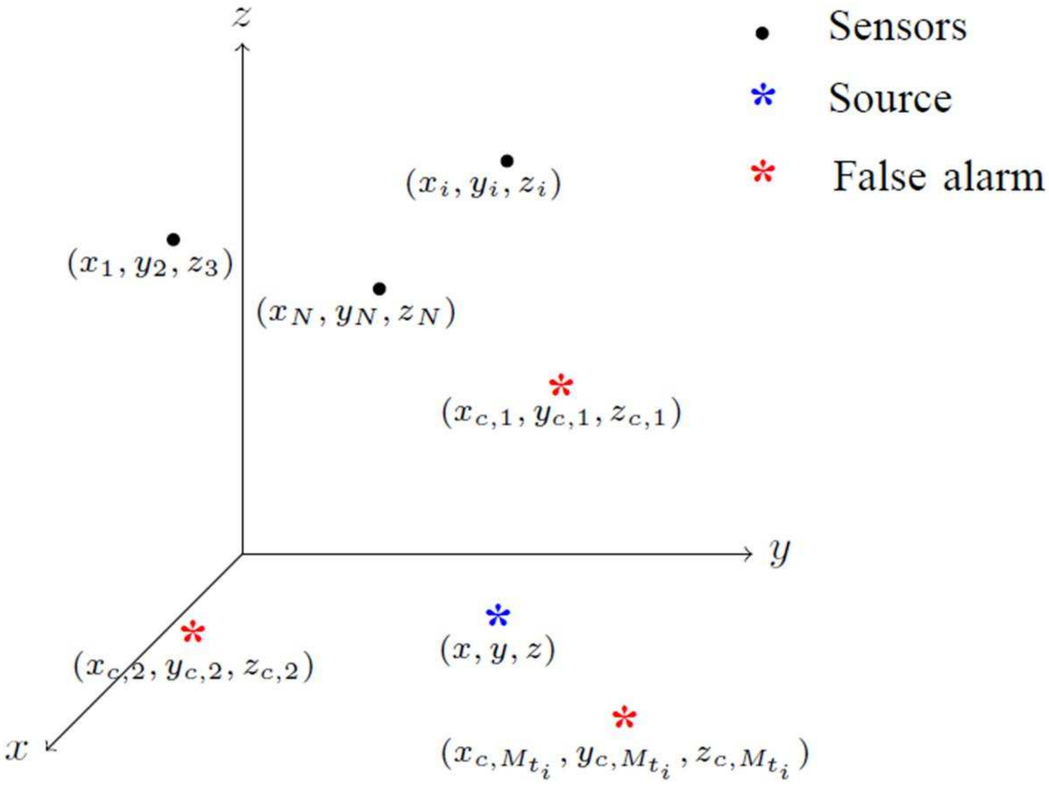
z
● Sensors
* Source
* False alarm
(x_i, y_i, z_i)
(x_1, y_2, z_3)
(x_N, y_N, z_N)
$(x_{c,1}, y_{c,1}, z_{c,1})$
y
$(x_{c,2}, y_{c,2}, z_{c,2})$
(x, y, z)
$(x_{c,M_{t_i}}, y_{c,M_{t_i}}, z_{c,M_{t_i}})$
x

Figure A.2: Geolocation of an emitter by multisensor with false alarm

Bibliography

[1] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking", *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[2] Bar-Shalom, Y., Willett, P. K., and Tian, X., *Tracking and Data Fusion*, YBS Publishing, 2011.

[3] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation*, John Wiley, New York, 2001.

[4] Baser, E., and Efe, M., "A novel auxiliary particle PHD filter", *Proceedings of the International Conference on Information Fusion*, pp. 165–172, July 2012.

[5] Blackman, S. S., "Multiple hypothesis tracking for multiple target tracking", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, pp. 5–18, Jan. 2004.

[6] Blair, W. D., Watson, G. A., Kirubarajan, T., and Bar-Shalom, Y., "Benchmark for radar allocation and tracking in ECM", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 1097–1114, Oct. 1998.

[7] Blackman, S., *Multiple Target Tracking with Radar Applications*, Artech House, MA, 1986.

[8] Blanding, W., Willett, P., and Bar-Shalom, Y., "MLPDA: advances and a new multitarget approach", *EURASIP journal on Advances in Signal Processing*, vol. 2008, pp. 1–13, Mar 2007.

[9] Blanding, W., Willett, P., and Bar-Shalom, Y., "Multiple target tracking using maximum likelihood probabilitic data association", *Proceedings of the IEEE Aerospace Conference*, pp. 1–12, Big Sky, MT, Mar 2007.

[10] Blanding, W., Willett, P., and Bar-Shalom, Y., "Directed subsapce search ML-PDA with application to active sonar tracking", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, pp. 201–216, Jan 2008.

[11] Bucy, S. R., and Youssef, M. H., "Nonlinear filter representation via spline functions", *Proceedings of the 5th Symposium on Nonlinear Estimation Theory and its Applications*, pp. 51–60, Sep. 1974.

[12] Buss, R. S., *A Mathematical Introduction with OpenGL*, Cambridge University Press, 2003.

[13] Chen, H., Bar-Shalom, Y., Pattipati, K., R., and Kirubarajan, T., "MDL approach for multiple low observable track initiation", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, pp. 862–882, July 2003.

[14] Chummum, Y., Bar-Shalom, Y., and Kirubarajan, T., "Adaptive early detection ML-PDA estimator for LO targets with EO sensors", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, pp. 694–707, Apr 2002.

[15] Clark, D., Vo, B.-T., and Vo, B.-N., "Gaussian particle implementations of probability hypothesis density filters", *Proceedings of the IEEE Aerospace Conference*, pp. 1–11, Big Sky, Montana, Mar. 2007.

[16] Coman, C. I., and Kreitmair, T., "Evaluation of the tracking process in ground surveillance applications", *Proceedings of the European Radar Conference*, pp. 553–556, Oct. 2009.

[17] De Boor, C., *A Practical Guide to Splines*, Springer-Verlag, New York, 2001.

[18] Doucet, A., De Freitas, N., and Gordon, N., *Sequential Monte Carlo Methods in Practice*, Springer, New York, 2001.

[19] El-Fallah, A., Zatezalo, A., Mahler, R. P. S., Mehra, R. K., and Alford, M., "Regularized multitarget particle filter for sensor management", *Proceedings of the SPIE Signal Processing, Sensor Fusion, and Target Recognition XV*, vol. 6235, pp. 1–11, Apr. 2006.

[20] Georgescu, R., Willett, P., and Schoenecker, S., "GM-CPHD and ML-PDA applied to the metron multi-static sonar dataset", *Proceedings of 13th Conference on Information Fusion*, pp. 1–8, July 2010.

[21] Gorji, A. A., Tharmarasa, R., and Kirubarajan, T., "Performance measures for multiple target tracking problems", *Proceedings of the 14th International Conference on Information Fusion*, pp. 1–8, July 2011.

[22] Goldberg, D., E., *Genetic Algorithums in Search, Optimization, and Machine Learning*, Addison-Wesley, Boston, Mass, USA, 1989.

[23] Goldenthal, R., and Bercovier, M., “Spline curve approximation and design by optimal control over the konts using genetic algorithums”, *Internationl Congress on Evolutionary Methods for Design*, Nov 2003.

[24] He, X., Tharmarasa, R., Kocherry, D. L., Bhashyam, B., and Kirubarajan, T., “A spline filter for multidimensional nonlinear state estimation”, *Proceedings of the International Conference on Information Fusion*, pp. 1–8, July 2011.

[25] He, X., Sithiravel, R., Tharmarasa, R., Bhashyam, B., and Kirubarajan, T., “A spline filter for multidimensional nonlinear state estimation”, Accepted to *Signal Processing*, 2014.

[26] Ho, K. C., and Chan, Y. T., “Geoloation of a known altitude object from TDOA and FDOA measurements”, *IEEE Transactions on Signal Proessing*, pp. 770–783, July 1997.

[27] Jauffret, C., and Bar-Shalom, Y., “Track formation with bearing and frequency measurements in clutter”, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 6, pp. 999–1010, Nov 1990.

[28] Jia, B., Xin, M., Cheng, Y., “Sparse-grid quadrature nonlinear filtering”, *Signal Processing*, pp. 327–341, Feb. 2012.

[29] Jiang, X., Harishan, K., Tharmarasa, R., Kirubarajan, T., and Thayaparan, T., “Integrated track initialization and maintenance in heavy clutter using probabilistic data association”, *Signal Processing*, vol. 94, no. 2014, pp. 241–250, July 2013.

[30] Li, X. R., and Jilkov, V. P., “Survey of maneuvering target tracking. Part V.

Multiple-model methods", *IEEE Transactions on Aerospace and Electronic Systems*,vol. 41, pp. 1255–1321, Oct. 2005.

[31] Kirubarajan, T., and Bar-Shalom, Y., "Low observable target motion analysis using amplitude information", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1367–1384, Oct 1996.

[32] Kirubarajan, T., Bar-Shalom, Y., and Lerro, D., "Bearing-only tracking of maneuvering targets using a batch-recursive estimator", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 3, pp. 770–780, July 2001.

[33] Kocherry, D. L., Tharmarasa, R., Lang, T., and Kirubarajan, T., "Spline filter for target tracking", *Proceedings of SPIE*, vol. 7698, pp. 1–8, Apr. 2010.

[34] Lin, X., Kirubarajan, T., Bar-Shalom, Y., and Maskell, S., "Comparison of EKF, pseudomeasurement and particle filters for a bearing-only target tracking problem", *Proceedings of SPIE*, Orlando, FL, Apr. 2002.

[35] Lee, D., "Nonlinear estimation and multiple sensor fusion using unscented information filter", *Proceeding of IEEE Signal Processing Letters*, pp. 861–864, 2008.

[36] Mahler, R. P. S., *Statistical Multisource-Multitarget Information Fusion*, Artech House, Boston, 2007.

[37] Mahler, R. P. S., "On multitarget jump-Markov filters", *Proceedings of International Conference on Information Fusion*, Singapore, July. 2012.

[38] Mahler, R. P. S., "Statistics 102 for Multisource-Multitarget Detection and Tracking", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 7, pp. 376–389, June 2013.

[39] Ma, C., Klukas, R., and Lachapelle, G., "An enhaned two-step least squares approach for TDOA/AOA wireless location", *In Proeeding of the IEEE International Conference on Communiations*, vol. 02, pp. 987–991, May 2003.

[40] Merrill, I. S., *Radar Handbook*, McGraw Hill, New York, 2008.

[41] Musicki, D., and Evans, R., "Clutter map information for data association and track initialization", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, pp. 387–398, Apr. 2004.

[42] Nardone, S. C., Lindgren, A. G., and Gong, K. F., "Fundamental properties and performance of conventional bearing-only target motion analysis", *IEEE Transactions on Automatic Control*, vol. 29, pp. 775–787, Sep. 1984.

[43] Ng, L. C., and Bar-Shalom, Y., "Multisensor multitarget time delay vector estimation", *IEEE Transactions on Acousti, Speech, and Signal Processing*, pp. 669–678, Aug. 1986.

[44] Oudjane, N., and Musso, C., "Progressive correction for regularized particle filters", *Proceedings of the Third International Conference on Information Fusion*, vol. 2, pp. 10–17, July 2000.

[45] Ouyang, C., Ji, H., and Li, C., "Improved multi-target multi-Bernoulli filter", *IET Transactions on Radar, Sonar and Navigation*, vol. 6, pp. 458–464, July 2012.

[46] Pasha, S. A., Vo, B.-N., Tuan, H. D., and Ma, W.-K., "A Gaussian mixture PHD filter for jump Markov system models", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, pp. 919–936, July 2009.

[47] Popp, R. L., Kirubarajan, T., and Pattipati, K. R., Chapter 2 *in Multitarget/Multisensor Tracking: Applications and Advances III*, (Bar-Shalom, Y. and Blair, W. D., eds.), Artech House, 2000.

[48] Punithakumar, K., McDonald, M., and Kirubarajan, T., "Spline filter for multidimensional nonlinear/non-Gaussian Bayesian tracking", *Proceeding of SPIE*, vol. 6969, pp. 1–8, May. 2008.

[49] Punithakumar, K., Kirubarajan, T., and Sinha, A., "Multiple-model probability hypothesis density filter for tracking maneuvering targets", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, pp. 87–98, Jan. 2008.

[50] Petetin, Y., and Desbouvries, F., "A mixed GM/SMC implementation of the probability hypothesis density filter", *Proceedings of the International Conference on Information Science, Signal Processing and their Applications*, pp. 425–430, July 2012.

[51] Rappaport, T. S., Reed, J. H., and Woerner, B. D., "Position location using wireless communiations on highways of the future", *IEEE Communications Magazine*, pp. 33–41, Oct. 1996.

[52] Reed, J. H., Krizman, K. J., Woerner, B. D., and Rappaport, T. S., "An overview of the challenges and progress in meeting the E-911 requirement for location service", *IEEE Communications Magazine*, pp. 30–37, Apr. 1998.

[53] Rogers, F. D., *An Introduction to NURBS*, Morgan Kaufmann Publishers, California, 2001.

[54] Streit, R. L., *Poisson Point Processes: Imaging, Tracking, and Sensing*, Springer, New York, 2010.

[55] Schuhmacher, D., Vo, B. -T., and Vo, B. -N., "A consistent metric for performance evaluation of multi-object filters", *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[56] Sithiravel, R., Tharmarasa, R., McDonald, M., Pelletier, M., Kirubarajan, T., "The Spline Probability Hypothesis Density Filter", *Proceedings of SPIE, Signal Processing, Sensor Fusion, and Target Recognition XXI*, vol. 8392, Baltimore, MA, May 2012.

[57] Sithiravel, R., Chen, X., McDonald, M., and Kirubarajan, T., "Spline Probability Hypothesis Density Filter for Nonlinear Maneuvering Target Tracking", *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pp. 1743–1750, Pacific Grove, CA, Nov. 2013.

[58] Sithiravel, R., Chen, X., Tharmarasa, R., Balaji, B., and Kirubarajan, T., "The spline probability hypothesis density filter", *IEEE Transactions on Signal Processing*, vol. 61, no. 24, pp. 6188–6203, Dec. 2013.

[59] Sithiravel, R., Tharmarasa, R., Sreeraman, R., and Kirubarajan, T., "Spline Maximum-Likelihood Probabilistic Data Association Filter for Passive Track Initialization", *To be submitted IEEE Transactions on Signal Processing.*

[60] Straka, O., Dunik, J., and Simandl, M., "Performance evaluation of local state estimation methods in bearings-only tracking problems", *Proceedings of International Conference on Information Fusion*, pp. 1–8, July 2011.

[61] Tang, X., Wei, P., "Multitarget state extraction for the particle probability hypothesis density filter", *IET Radar, Sonar and Navigation*, vol. 5, pp. 877–883, Oct. 2011.

[62] Tobias, M., and Lanterman, A. D., "A probability hypothesis density-based multitarget tracker using multiple bistatic range and velocity measurement", *Proceeding of the Southeastern Symposium on System Theory*, pp. 205–209, Mar. 2004.

[63] Vo, B.-N., and Ma, W.-K., "The Gaussian mixture probability hypothesis density filter", *IEEE Transactions on Signal Processing*, vol. 54, pp. 4091–4104, Nov. 2006.

[64] Vo, B.-T., Vo, B.-N., and Cantoni, A., "The cardinality balanced multi-target multi-Bernoulli filter and its implementations", *IEEE Transactions on Signal Processing*, vol. 57, pp. 409–423, Feb. 2009.

[65] Vo, B.-N., Singh, S., and Doucet, A., "Random finite sets and sequential Monte Carlo methods in multi-target tracking", *Proceedings of the international Radar Conference*, pp. 486–491, Sep. 2003.

[66] Vermaak, J., Godsill, S. J., and Perez, P., "Monte Carlo filtering for multi target tracking and data association", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, pp. 309–332, Jan. 2005.

[67] Wang, A. H., and Klein, R. L., "Implementation of non-linear estimators using monospline", *Proceeding of IEEE Conferecne on Decision and Control including the 15th Symposium on Adaptive Processes*, vol. 15, pp. 1305–1307, Dec. 1976.

[68] Wax, M., and Kailath, T., "Optimum localization of multiple sources by passive arrays", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, pp. 1210–1217, Oct. 1983.

[69] Whiteley, N., Singh, S., and Godsill, S., "Auxiliary particle implementation of probability hypothesis density filter", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, pp. 1437–1454, July 2010.

[70] Willett, P., Coraluppi, S., and Blanding, W., "Comparison of soft and hard assignment ML trackers on multistatic data", *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, Mar 2006.

[71] Yin, J. J., Zhang, J. Q., Hu, B., and Lu, Q. Y., "The polynomial predictive Gaussian mixture MeMBer filter", *Proceeding of IEEE Sensor Array and Multichannel Signal Processing Workshop*, pp. 233–236, Oct. 2010.

[72] Yin, J., Zhang, J., Zhao, J., "The Gaussian particle multi-target multi-Bernoulli filter", *Proceeding of International Conference on Advanced Computer Control*, vol. 4, pp. 556–560, Mar. 2010.

[73] Yoon, J., Kim, D., and Yoon, K., "Gaussian mixture importance sampling function for unscented SMC-PHD filter", *Signal Processing*, vol. 93, no. 9, pp. 2664–2670, Sep. 2013.

[74] Zhang, J., Ji, H., and Ouyang, C., "A new Gaussian mixture particle CPHD filter for multitarget tracking", *Proceeding of International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 1–4, Dec. 2010.