

EFFECTS OF THE RECEIVER CHARACTERISTICS, VARIATION OF TARGET
INTENSITY AND NONLINEARITY ON PROBABILITY HYPOTHESIS
DENSITY FILTER COMPARED TO DATA ASSOCIATION TECHNIQUES

by

Kemal ÖKSÜZ

B.S., System Engineering, Land Forces Academy, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2016

ACKNOWLEDGEMENTS

Firstly I would like to thank to my supervisor Assoc. Prof. Ali Taylan Cemgil who has the leading role on my academic attempt and progress. He was a member of oral interview committee when I decided to embark on a different career by applying to computer engineering master program of Boğaziçi University in 2013. Since then, with his encouraging and constructive attitude, he has made me have self confidence during my education and research. Also he is the person that drove me into multiobject tracking research area, thereby allowing me to integrate my army career with my research career.

My superiors, Col. Nurettin Saral and Lt.Col. Mehmet Erol, inconvertibly have a role for completing my education since they allowed me to attend the courses even though I have a job with full of responsibility in Turkish Armed Forces. I owe them an appreciation for believing in me that I could cope with my duty and education concurrently. It would not be possible to complete this thesis without their support.

One of my best friends, Burak Ergöçmen, a pilot instructor in TAF, shared me some information on patterns of aircrafts and helicopters. He was very patient in spite of my insistent calls and questions. I thank him for helping me throughout my research.

Of course I reserved the last words to my family. My wife Seda, my father Hasan, my mother Tülay and my brother Kubilay always tolerated my excuses to study whenever they offered me to do something together. As being totally novice to the computer engineering field and working in the weekdays, I will never forget my remedial year when I could just meet them in the dinners for a short time. Thus, I believe that I am very lucky to have such a family.

Lastly, I thank to the Scientific and Technological Research Council of Turkey (TÜBİTAK). This thesis has been supported by TÜBİTAK MS scholarship(2210-A).

ABSTRACT

EFFECTS OF THE RECEIVER CHARACTERISTICS, VARIATION OF TARGET INTENSITY AND NONLINEARITY ON PROBABILITY HYPOTHESIS DENSITY FILTER COMPARED TO DATA ASSOCIATION TECHNIQUES

Target tracking algorithms adopted in modern radars are designed such that they can track multitarget by considering target births and target deaths. These algorithms are derived by integrating the data association techniques into the single target filters. Recently, target tracking methods exploiting random finite sets have been emerged as an alternative to the data association techniques. Unlike data association methods, random finite set based techniques do not perform tracking based on the targets but instead propagate a target intensity function covering the entire state space in time and thereby decrease the dimension of the state space. In this thesis, firstly on a linear scenario we investigate the effects of receiver characteristics and variation of target intensity on the performance of PHD filter that is a random finite set based filter. The parameters we consider for receiver characteristics are detection probability and false alarm intensity; for variation of target intensity we investigate the effect of target birth and death probabilities. We also provide a linear regression model representing effects of these parameters. As a tracking performance metric we use OSPA distance. At each step, we compare our results with a data association method, global nearest neighbor technique, in order to identify the advantages and disadvantages of the both of the methods. Secondly we investigate the effect of the nonlinear models on both of the methods. By fixing the parameters to the values that results in equal average OSPA distances of both techniques in the linear case, we include nonlinear models in order to identify which technique is effected by nonlinearity more.

ÖZET

ALICI ÖZELLİKLERİ, HEDEF YOĞUNLUĞUNDAKİ DEĞİŞİM VE NONLINEERLİĞİN OLASILIK HİPOTEZ YOĞUNLUĞU FİLTRESİ ÜZERİNDE VERİ İLİŞKİLENDİRME TEKNİKLERİNE KIYASLA ETKİLERİ

Modern radarlarda kullanılan hedef takip algoritmaları, hedef doğum ve ölümlerini dikkate alarak birden çok hedefi aynı anda takip edebilecek şekilde dizayn edilmektedir. Bu algoritmalar, tek hedef takibi için kullanılan filtrelere veri ilişkilendirme tekniklerinin entegre edilmesiyle geliştirilmiştir. Son dönemde ise veri ilişkilendirme temelli metodlara alternatif olarak rassal sonlu kümeleri kullanan hedef takip algoritmaları ortaya çıkmıştır. Veri ilişkilendirme tekniklerinden farklı olarak, rassal sonlu küme temelli teknikleri hedeflere dayalı bir takip gerçekleştirmezler, onun yerine bütün durum uzayını örten bir hedef yoğunluk fonksiyonunu zaman içerisinde yayarlar ve bu şekilde durum uzayının boyutunu azaltırlar. Bu tezde, öncelikle lineer bir senaryo üzerinde alıcı özellikleri ve hedef yoğunluğundaki değişimin bir rassal sonlu küme temelli teknik olan OHY Filtresinin performansı üzerindeki etkisini araştırıyoruz. Alıcı özelliği parametreleri olarak tespit olasılığı ve parazit oranını; hedef yoğunluğundaki değişim için ise hedef doğum ve ölüm olasılıklarını kullanıyoruz. Bu parametrelerin etkilerini açıklayan bir lineer regresyon modeli de sunuyoruz. Performans ölçütü olarak ise OSPA mesafesini kullanıyoruz. Her bir adımda sonuçlarımızı bir veri ilişkilendirme tekniği ile, en yakın global komşu tekniği ile, her iki tekniğin avantaj ve dezavantajlarını belirlemek için kıyaslıyoruz. İkinci olarak ise lineer olmayan modellerin her iki teknik üzerindeki etkisini araştırıyoruz. Lineer durumdaki ortalama OSPA mesafesini her iki teknik için eşit yapan değerlere sabitleyerek, hangi tekniğin lineer olmayan modellerden daha çok etkilendiğini belirlemek amacıyla modele nonlineerliği ilave ediyoruz.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ÖZET	v
LIST OF FIGURES	ix
LIST OF TABLES	xii
LIST OF SYMBOLS	xiii
LIST OF ACRONYMS/ABBREVIATIONS	xvii
1. INTRODUCTION	1
2. BASICS OF THE RADAR	5
2.1. Major Elements of a Radar	5
2.2. EM Waves	6
2.2.1. Radar Frequencies	7
2.2.2. Interaction with Matter	8
2.3. Basic Radar Measurements and Detection	8
2.3.1. Basic Radar Measurements	8
2.3.1.1. Azimuthal and Elevation Angle	10
2.3.1.2. Range Using the Speed of Light	10
2.3.1.3. Range Rate Using Doppler Frequency Shift	10
2.3.1.4. Polarization	10
2.3.2. Noise and Detection	11
2.4. Basic Radar Functions	12
3. CONVENTIONAL MODELING OF TRACKING PROBLEM AND OPTIMAL BAYESIAN FILTER	14
3.1. Single Target Modeling	15
3.2. A Simple Multitarget Environment Modeling	15
3.3. Multitarget Modeling: Missed Detection, False Alarm, Target Birth, Death and Spawning	17
3.4. Optimal Bayesian Filter	18
4. SINGLE TARGET TRACKING TECHNIQUES	20

4.1.	Kalman Filter	21
4.2.	Extended Kalman Filter	24
4.3.	Unscented Kalman Filter	26
4.3.1.	Unscented Transformation	27
4.3.2.	UKF Algorithm	27
4.4.	Sequential Monte Carlo in Target Tracking	29
4.4.1.	Importance Sampling	30
4.4.2.	Resampling	31
4.4.3.	Sequential Importance Sampling	32
5.	EXTENDING TO MULTITARGET ENVIRONMENT: DATA ASSOCIATION	38
5.1.	Transition and Measurement Models	38
5.2.	Gating	39
5.3.	A Data Association Algorithm: Global Nearest Neighbor Technique . .	41
6.	MULTITARGET TRACKING VIA PROPAGATING FIRST ORDER STATISTICAL MOMENT:PHD FILTER	46
6.1.	Modeling Tracking Using Random Finite Sets	47
6.1.1.	Probability Distribution of a RFS	49
6.1.2.	Transition Model	50
6.1.3.	Measurement Model	50
6.2.	Finite Set Statistics(FISST)	51
6.2.1.	Construct the Multitarget Densities and Likelihood Functions .	52
6.2.2.	Construct the Multitarget Optimal Bayes Filter	54
6.2.3.	Convert the Bayes Filter into Probability Generating Functional Form	55
6.3.	The PHD	57
6.4.	PHD Filter	59
6.4.1.	Initialization	60
6.4.2.	Prediction	60
6.4.3.	Update	62
6.4.4.	State Estimation	63
6.5.	Evaluation of the PHD Filter	64

7. IMPLEMENTATION STRATEGIES OF THE PHD FILTER	66
7.1. Particle Filter Based Approaches	66
7.1.1. SMCPHD Filter	67
7.1.1.1. Initialization	67
7.1.1.2. Prediction	67
7.1.1.3. Update	68
7.1.1.4. State Estimation	68
7.1.1.5. The Algorithm	68
7.1.2. Improved SMCPHD Filter	70
7.1.2.1. Improved SMCPHD Formulation of the PHD	70
7.1.2.2. State and Error Estimation	74
7.2. Gaussian Mixture Based Approaches	75
7.2.1. GMPHD Filter	75
7.2.1.1. Prediction	77
7.2.1.2. Update	79
7.2.1.3. Pruning for the GMPHD	80
7.2.2. EKFPHD Filter	80
7.2.3. UKFPHD Filter	80
8. EXPERIMENTS AND RESULTS	85
8.1. OSPA Distance	85
8.2. Linear Model	86
8.2.1. Realization of the Filters	86
8.2.2. Effects of the Parameters	89
8.2.2.1. Receiver Characteristic Parameters: λ_{FA} and P_D	89
8.2.2.2. Target Intensity Parameters: P_B and P_S	90
8.2.3. Linear Regression Analysis	92
8.3. Nonlinear Model	96
8.3.1. Realization of the Filters	97
8.3.2. Effect of the Nonlinearity	100
9. CONCLUSION AND FUTURE WORK	103
REFERENCES	105

LIST OF FIGURES

Figure 2.1.	Basic Elements of a Radar	6
Figure 2.2.	Radar Frequencies and Electromagnetic Spectrum	7
Figure 2.3.	Noise and Detection Example	11
Figure 3.1.	Bayesian Network of Single Target Model	16
Figure 3.2.	Bayesian Network of Basic Multitarget Model	16
Figure 3.3.	Bayesian Network of Realistic Multitarget Model	17
Figure 4.1.	Tracking the Attitude of a Helicopter by Kalman Filter	23
Figure 4.2.	Erroneous Estimation Example	24
Figure 4.3.	Principle of the Unscented Transformation	27
Figure 4.4.	Tracking an Aircraft by Particle Filter	35
Figure 4.5.	Particle Filter Algorithm	37
Figure 5.1.	Data Association Process	40
Figure 5.2.	Gating Step Example	41
Figure 5.3.	Matrix Representation of Generated Hypotheses	43

Figure 5.4.	Tree Representation of Generated Hypotheses	44
Figure 6.1.	Illustration of Possible Scenarios of an RFS	48
Figure 6.2.	An Illustration of Probability Hypothesis Density	59
Figure 6.3.	Effect of the Closer Targets on PHD	65
Figure 7.1.	SMCPHD Filter Algorithm	69
Figure 7.2.	Improved SMCPHD Filter Algorithm	76
Figure 7.3.	GMPHD Filter Algorithm	81
Figure 7.4.	GMPHD Filter State Estimation	82
Figure 7.5.	Pruning for GMPHD Filter	82
Figure 7.6.	EKFPHD Filter Algorithm	83
Figure 7.7.	UKFPHD Filter Algorithm	84
Figure 8.1.	The Measurements Generated According to the Model in Table 8.2	88
Figure 8.2.	KFGNN Filtering of the Measurements in Figure 8.1	88
Figure 8.3.	GMPHD Filtering of the Measurements in Figure 8.1	89
Figure 8.4.	OSPA Distance Comparison for Filters	90
Figure 8.5.	OSPA Distance Comparison for Varying λ_{FA}	91

Figure 8.6.	OSPA Distance Comparison for Varying P_D	91
Figure 8.7.	OSPA Distance Comparison for Varying P_B	92
Figure 8.8.	OSPA Distance Comparison for Varying P_S	93
Figure 8.9.	The Measurements Generated According to the Model in Table 8.4	97
Figure 8.10.	EKFPHD Filtering of the Measurements in Figure 8.9	98
Figure 8.11.	UKFPHD Filtering of the Measurements in Figure 8.9	98
Figure 8.12.	ISMCPHD Filtering of the Measurements in Figure 8.9	99
Figure 8.13.	EKFGNN Filtering of the Measurements in Figure 8.9	99
Figure 8.14.	OSPA Distance Comparison of Realization Techniques for Nonlinear Implementations	100
Figure 8.15.	Effect of Using More Particles on ISMCPHD	101
Figure 8.16.	Effect of Nonlinearity on the Filters	102

LIST OF TABLES

Table 2.1.	IEEE Standart for Radar Frequency Letter-Band Nomenclature . . .	9
Table 4.1.	Hovering Helicopter Model	22
Table 4.2.	Particle Filter Toy Example Model	34
Table 8.1.	Domain of Each Parameter for Linear Model Experiments	86
Table 8.2.	State Transition and Measurement Models of the Linear Scenario .	87
Table 8.3.	The Results of the Linear Model Experiments	95
Table 8.4.	State Transition and Measurement Models of the Nonlinear Scenario	96

LIST OF SYMBOLS

A	State transition matrix
A_t	State transition matrix (changing in time) at t
$b(x_t)$	PHD of target birth
$b(x_t x_{t-1})$	PHD of target spawning
C	Measurement process matrix
C_t	Measurement process matrix (changing in time) at t
c	Speed of light
$D_{t t}[x_t Y_{1:t}]$	Updated PHD at t
$D_{t-1 t}[x_t Y_{1:t-1}]$	Predicted PHD at t
D_θ^2	Global association distance of an hypothesis
$\bar{e}_{p,card}^{(c)}(X, Y)$	OSPA cardinality error of sets X and Y with parameters p, c
$\bar{e}_{p,loc}^{(c)}(X, Y)$	OSPA localization error of sets X and Y with parameters p, c
$fa(\cdot)$	Spatial probability distribution of false alarm process
f_d	Doppler frequency
$f_\Xi(X)$	Probability distribution of an RFS Ξ
$f_t(\cdot)$	State transition function at t
G	Gating region size
$G_x(S)$	Probability generating function of $p(x)$
$G_{t t}[h Y_{1:t}]$	Updated probability generating functional of multitarget filtering density
$h_t(\cdot)$	Measurement process function at t
h^X	Power functional of function $h(x)$
I_t	Total number of particles at t for ISMCPHD
J_t	Number of new particles generated at t for SMCPHD
$J_t^{(b)}$	Total number of new born particles generated at t
$J_t^{(b')}$	Total number of spawning particles generated at t
K_t	Kalman Gain matrix at t
L_t	Number of particles that approximates PHD at t

$m^{(b,i)}$	Mean of i th Gaussian component of target birth PHD
$m^{(b',i)}$	Mean of i th Gaussian component of target spawning PHD
M_t	Total number of measurements at t
N_t	Total number of targets at t
\hat{N}_t	Expected number of targets at t
\bar{N}_t	Smoothed expected number of targets at t
n	Dimension of state space
n_t	Switch variable of directed graphical model for labeling the hypothesis at t
p_B	Probability of Birth
p_D	Probability of Detection
p_{FA}	Probability of False Alarm
p_S	Probability of Survival of a Target for the next time step
$P^{(b,i)}$	Covariance matrix of i th Gaussian component of target birth PHD
$P^{(b',i)}$	Covariance matrix of i th Gaussian component of target spawning PHD
$P_{t t-1}$	Predicted covariance matrix of KF at t
$P_{t t}$	Updated covariance matrix of KF at t
P_t^y	Innovation covariance matrix of UKF at t
P_t^{xy}	Cross covariance matrix of UKF at t
$p(x_t y_{1:t})$	Filtering density at t
Q	Covariance matrix of state transition noise
$q_t(x)$	Proposal distribution of IS
$r_t(x_t^{(i)} Y_{1:t})$	Proposal for target birth in ISMC PHD
R	Covariance matrix of measurement process error
$q_t(x_t^{(i)} x_{t-1}^{(i)}, Y_{1:t})$	Proposal for spawning and state transition in ISMC PHD
$s_B(x_t)$	Spatial distribution of target birth at t
$s_{FA}(x_t)$	Spatial distribution of false alarm process at t
$s_S(x_t)$	Spatial distribution of target spawning at t
\bar{T}	Window size of PHD state estimator
$T(x_t^i)$	RFS of surviving target

v_r	Radial velocity of the target
v_t	Measurement error at t
$v_{t t-1,b}$	Predicted expected number of target birth for ISMCPHD
w^i	Weight of i th sigma point in UKF
$w^{(i)}$	Normalized weight of i th particle
w_t	State transition noise at t
$w^{(b,i)}$	Weight of i th Gaussian component of target birth PHD
$w^{(b',i)}$	Weight of i th Gaussian component of target spawning PHD
$w_{t t-1,b}^{(i)}$	Predicted weight of i th new born target particle for ISMCPHD
$w_{t t-1,s}^{(i)}$	Predicted weight of i th surviving target particle for ISMCPHD
$w_{t t,b}^{(i)}$	Updated weight of i th new born target particle for ISMCPHD
$w_{t t,s}^{(i)}$	Updated weight of i th surviving target particle for ISMCPHD
$w_{t t,s}^{i,j}$	Weight of i th particle of j th replica particle set for ISMCPHD
$W(x)$	Weight distribution of IS
$x^{(i)}$	i th particle of PF
x_t	Target state at t
$x_{t t-1}$	Predicted target state at t
$x_{t t}$	Updated target state at t
$x_{t t-1,b}^{(i)}$	Predicted state of i th new born target particle for ISMCPHD
$x_{t t-1,s}^{(i)}$	Predicted state of i th surviving target particle for ISMCPHD
x_t^i	State of i th target at t
$\hat{x}^{(i)}$	State of i th particle in SMC
X_t	Target state set at t
y_t	Measurement state at t
y_t^i	State of i th measurement at t
Y_t	Measurement state set at t
Z_t	Normalization constant at t
α	First parameter of $\alpha - \beta$ filter
β	Second parameter of $\alpha - \beta$ filter

β_t	label of target states determining target birth in ISMCPHD at t
$\beta_{T(x)}$	Belief mass function of RFS $T(x)$
$\delta_{x'}(x)$	Dirac delta function concentrated at x'
θ	A Measurement to Track Association Hypothesis
Θ	RFS of false alarm and clutter
λ	Wavelength of the EM Wave
λ_B	Parameter of Poisson distribution of target birth
λ_{FA}	Parameter of Poisson distribution of false alarm
λ_S	Parameter of Poisson distribution of target spawning
$\mu_{t t-1}$	Predicted mean at t
$\mu_{t t}$	Updated mean at t
ν_t	Target velocity at t for $\alpha - \beta$ filter
Ξ_t	Target state RFS at t
$\pi(x)$	Target distribution of IS
ρ	Number of particles associated per target for SMCPHD
Σ_t	Measurement state RFS at t
$\Upsilon(x_t^i)$	RFS of missed detection x_t^i
$\phi(x)$	Potential of $\pi(x)$ in IS
φ	RFS of target birth
$\varphi(x_t^i)$	RFS of spawned target x_t^i

LIST OF ACRONYMS/ABBREVIATIONS

ADT	Automatic Detection and Tracking
CHT	Composite Hypothesis Tracking
EKF	Extended Kalman Filter
EKFGNN	Global Nearest Neighbor Integrated Extended Kalman Filter
EKFPHD	Extended Kalman Gaussian Mixture Probability Hypothesis Density Filter
EM	Electromagnetic
FISST	Finite Set Statistics
GM	Gaussian Mixture
GNN	Global Nearest Neighbour
HMM	Hidden Markov Model
IEEE	The Institute of Electrical and Electronics Engineers
IF	Intermediate Frequency
i.i.d.	Independent and Identically Distributed
IS	Importance Sampling
ISMCPHD	Improved Sequential Monte Carlo Probability Hypothesis Density Filter
ITU	International Telecommunications Union
KF	Kalman Filter
KFGNN	Global Nearest Neighbor Integrated Kalman Filter
MC	Monte Carlo
MeMBeR	Multi Bernoulli Filter
OTH	Over the Horizon
PDAF	Probabilistic Data Association Filter
PHD	Probability Hypothesis Density
RF	Radio Frequency
RFS	Random Finite Set
ROC	Receiver Operating Characteristics
SMC	Sequential Monte Carlo

SMCPHD	Sequential Monte Carlo Probability Hypothesis Density Filter
SNR	Signal-to-Noise Ratio
TAF	Turkish Armed Forces
UKF	Unscented Kalman Filter
UKFPHD	Unscented Kalman Gaussian Mixture Probability Hypothesis Density Filter

1. INTRODUCTION

Since the first radar patent [1] issued to Christian Hulsmeier for his experiment on detection of radio waves reflected from ships and the proposal that is to investigate the feasibility of death rays to disable an aircraft leded Sir Robert Watson-Watt [2] to the detection of an aircraft using radio waves; the radar systems have evolved significantly not only in military applications but also in civilian cases. Especially, after the computer science has started to effect all of the fields and get the things automated; the radar technologies have evolved such that they have the Automatic Detection and Tracking (ADT) characteristic for performing their tracking function. Since then radar tracking have become a proliferous research area spanning from $\alpha - \beta$ tracking filters [3] devised in 1957 to random finite set based approaches that have began to be spotlighted in the beginning of this millennium. Dating back to $\alpha - \beta$ tracking filters, the conventional radar tracking methods including single target tracking methods such as Kalman filter [4,5] and data association techniques like multi hypothesis tracking [6] are well studied, thus any combination of this methods has been implemented for radar tracking. Kalman filter examines the tracking problem from Bayesian point of view by assuming the model complies with linearity and normally distributed. This technique is among the most prevalent techniques used for radar tracking and a few variations of the technique exist.

Since the Kalman filter based approaches assumes the transition and measurement models to be Gaussian, in some cases, i.e. model with heavy tailed densities, the filter may not perform well. Hence Monte Carlo techniques, particularly Sequential Monte Carlo [7–9] (aka particle filter), are also commonly used for radar tracking scenarios since these techniques have the power of dealing with integrals that are not mathematically tractable. Especially the rapid improvements in the hardware of the computer systems enable the tracking engineers to integrate particle filter based approaches to the radar processing unit, since the significant drawback of the Monte Carlo methods is their dependency on high processing time. However, a radar system that is an example of a real time system needs to carry out the estimation in

an online manner.

Unless combined with data association techniques, Kalman filter and SMC based approaches can be solely implemented in single target environments. But this is not appropriate for the real world problems since air traffic is getting more complex in time involving not only the conventional targets such as aircraft, helicopter etc, but also unconventional flying objects like UAV and quad-copter. Since it is not feasible to allocate a radar per target, somehow we need to extend these filters to the multitarget environment. Technically speaking, why single target approaches fail in multitarget environment is their lack of methodology for associating the measurements with the tracks. More precisely, after completing one scan a radar gathers the measurement data some of which are originated from a target and the rest are not. Hence the radar has a set of targets for the previous time interval and a set of measurements for now but single target tracking methods do not offer a association solution between these two sets. That's why we employ data association (also called measurement to track association) techniques [10]. After successful association between measurements and the targets, multiple number of single tracking filters can concurrently process the data.

Unlike these approaches that we label as conventional approaches, recently random finite set based approaches for target tracking have drawn great interest. The peculiarity of these approaches is that they omit the data association problem, thereby increasing the response time of the algorithm by simplifying the problem. From a mathematical point of view these techniques are derived from the point process theory [11] but in order to enable the tracking engineers use the approaches without delving into the complex mathematical formulation, its own terminology and statistics called finite set statistics [10, 12] have ensued. Various filtering methods using this point of view have emerged in the last decade such as Probability Hypothesis Density (PHD) filter [13], Cardinalized PHD filter [14] and Multi Bernoulli filter [10]. Specifically the PHD filter is the correspondent technique of $\alpha - \beta$ filters with respect to both their intuition and to be the initiatory of their research spaces. The PHD filter propagates the first order statistical moment, namely the PHD function of the filtering density. Even though the technique was introduced

theoretically without any implementation method initially, researches of the past decade resulted in mainly two implementation techniques including particle filter based approaches [15, 16] and gaussian mixture based [17, 18] approaches.

In this thesis we aim to identify the effects of receiver characteristics, variation in the target number and nonlinearity on the PHD filter. The parameters that we consider for receiver characteristics are detection probability and false alarm intensity, on the other hand for variation in the target intensity we investigate the effect of variation of birth and death probabilities of the target. We also represent a linear regression analysis that shows the effects of the parameters. At each step we depict the performance comparison with a data association technique, specifically global nearest neighbor technique. Then, in order to measure the effect of the nonlinearity, we fix the parameters to the values where in the linear case we have equal performance and measure how much the performance deteriorates for each nonlinear PHD filter implementation. PHD filter implementation technique that is used for linear case is Gaussian Mixture PHD filter and for nonlinear case we use Extended Kalman PHD filter, Unscented Kalman PHD filter and Improved Sequential Monte Carlo PHD filter. For the data association part, we use Kalman filter global nearest neighbor (GNN) in the linear case and Extended Kalman filter GNN for the nonlinear case. Moreover, we also aim at making up the thesis like a tutorial for someone interested in this area be familiarize with the basic notions of multitarget tracking. That's why, we include some toy examples when we describe the multitarget filters. The structure of the thesis is as follows: In Chapter 2 we represent the basics of a radar in order to familiarize the reader with some radar concepts and its data processing mechanism. Then in Chapter 3 we formulate the tracking problem, depict the problem in three different scenerios along with their directed graphical models and provide the optimal solution of the problem by explaining the optimal Bayesian filter. Afterwards we begin to offer the solution techniques for the problem. In Chapter 4 we represent the techniques that are used for single target tracking that are Kalman filter based approaches and particle filter. After explaining these single target tracking methods, we extend them to multitarget tracking environments by depicting the data association problem and a solution technique,

particularly global nearest neighbor in Chapter 5. This ends the conventional techniques of the multitarget tracking and we begin to represent the PHD filter in Chapter 6. In Chapter 7 we provide the implementation strategies for the PHD filter. Then we represent experiments that investigate the effects of receiver characteristics, variation in target intensity and nonlinearity on PHD filter in Chapter 8 and conclude the thesis in Chapter 9.

2. BASICS OF THE RADAR

The word Radar, abbreviation of Radio detection and ranging, reflects the purpose of the initial efforts inspired by developing a system that can only detect the target and estimate its range. Then gradually today's modern radar systems have become complex digital systems that not only detect targets and determine range but also track, identify, image, and classify targets while suppressing strong unwanted interference and countermeasures. Modern systems apply these major radar functions in an expanding range of applications, from the traditional military and civilian tracking of aircraft and vehicles to two and three-dimensional mapping, collision avoidance, earth resources monitoring, and many others. Rather than interfering directly with tracking filters, we think that having basic knowledge on radars will help the reader understand the role of these tracking methods. Thus, by avoiding the complex details of the subject, we provide a brief introduction on radars consisting of its major elements, EM waves, what a radar measures and how it detects and lastly its basic functions.

2.1. Major Elements of a Radar

Although the elements of the radars can vary with respect to application, an elementary form of radar comprises a transmitter, a receiver, an antenna and a signal processor. You may see these elements along with a simple transmission and reception process in Figure 2.1. Generally these 4 subsystems have the following functions in a radar system:

- Transmitter: It is responsible for generating the EM waves.
- Antenna: It sends and receives the EM waves between transmitter and the atmosphere.
- Receiver: It amplifies the received signal, converts the RF signal to an intermediate frequency (IF) and before sending it to the signal processor, convert the analog signal into a digital form. Receiver Protector Switch is

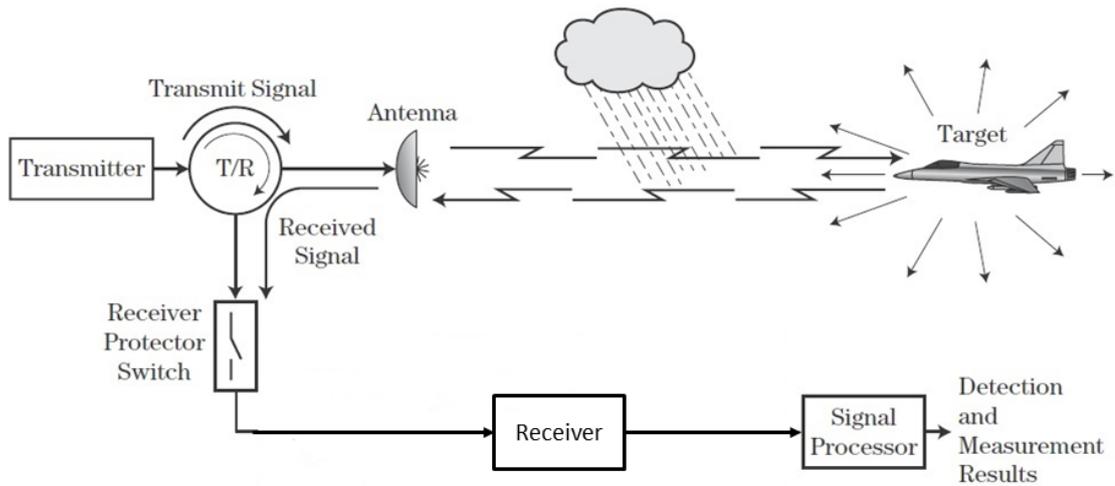


Figure 2.1. Basic radar elements involving in transmission and reception process
(adapted from [19])

responsible for protecting the sensitive receiver from the high-power transmit signal while they can be attached to the antenna simultaneously.

- **Signal Processor:** It is the subsystem where the tracking algorithms are executed. The input, provided by the receiver, is processed to infer the relevant data and decide what to do next. This is the subsystem where tracking algorithms are implemented.

2.2. EM Waves

EM waves are nothing but a type of light, hence they act according to the laws of light. In this section, in order to provide general overview about the type of the EM waves utilized in radars, the space in the electromagnetic spectrum that can be used by radars and possible interactions of these waves through its journey from transmitter to receiver is explained.

2.2.1. Radar Frequencies

Conventional radars generally have been operated at frequencies extending from about 220 MHz to 35 GHz. These are not necessarily the limits, since radars can be, and have been, operated at frequencies outside either end of this range. Skywave HF over-the-horizon (OTH) radar might be at frequencies as low as 4 or 5 MHz, and groundwave HF radars as low as 2 MHz. At the other end of the spectrum, millimeter radars have operated at 94 GHz. Laser radars operate at even higher frequencies. The place of radar frequencies in the electromagnetic spectrum is shown in Figure 2.2 [20].

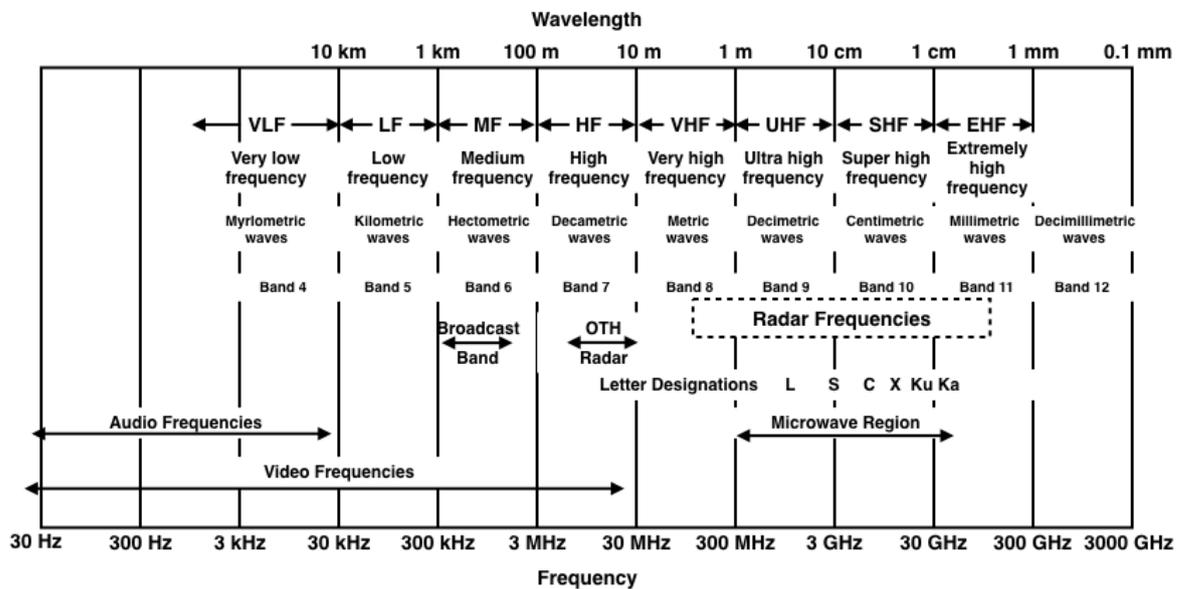


Figure 2.2. Radar Frequencies and Electromagnetic Spectrum [20].

The letter designation in Figure 2.2 represents different radar frequency bands. Although it was initially employed in order to provide secrecy for the military applications, this convention is still prevalent due to the necessity for classification. To illustrate, HF band for radar is utilized for detecting targets at long ranges (2000 nmi) by taking the advantage of the refraction of HF energy by the ionosphere. On the otherhand, radars at X band are convenient in size, so this facilitates mobility while curtailing the range. For more information about the peculiarities of the radar bands one may see [21]. Furthermore, since the usage of the electromagnetic spectrum is excessive, it is divided in small intervals and each interval is allocated for

a specific purpose by International Telecommunications Union (ITU). Thus, for a specific letter-band only a portion of its interval is available in practice. Table 2.1 represents letter-band nomenclature standard adopted by IEEE [22] and specific portions of the electromagnetic spectrum for radiolocation (radar) use for Region 1 that includes Turkey [23].

2.2.2. Interaction with Matter

During their journey, from transmitting to receiving, EM waves can be exposed to some internal and external factors that may effect their characteristics. These factors consist of diffraction occurring due to antenna; attenuation, refraction, depolarization occurring due to atmosphere and lastly reflection due to target. We do not delve into the details of these factors due to the scope of our research, but one must be aware of these factors while modeling the measurements, that are nothing but incoming EM waves.

2.3. Basic Radar Measurements and Detection

In this section, we introduce possible types of measurements that a radar can gather and the detection phenomenon including some definitions.

2.3.1. Basic Radar Measurements

A radar must get some inputs constantly so as to provide its operator with the desired output that is the location of the target in general case. Modern radars can observe the following target parameters simultaneously:

- (i) Azimuthal Angle
- (ii) Elevation Angle
- (iii) Range Using the Speed of Light
- (iv) Range Rate Using Doppler Frequency Shift
- (v) Polarization

Table 2.1. IEEE Standart for Radar Frequency Letter-Band Nomenclature.

Band	Frequency	Specific Frequency Ranges for Radar Based on ITU Frequency Assignments for Region 1
HF	3-30 MHz	4.438-4.488 MHz 5.25-5.275 MHz 9.305-9.355 MHz 13.45-13.55 MHz 16.1-16.2 MHz 24.45-24.6 MHz 26.2-26.35 MHz
VHF	30-300 MHz	39-39.5 MHz 42.0-42.5 MHz
UHF	300-1000 MHz	420-450 MHz 890-942 MHz
L	1000-2000 MHz	1215-1400 MHz
S	2000-4000 MHz	2300-2500 MHz 2700-3600 MHz
C	4000-8000 MHz	5250-5850 MHz
X	8000-12000 MHz	8400-10680 MHz
K_u	12-18 GHz	13.4-14.0 GHz 15.4-17.7 GHz
K	18-27 GHz	24.05-24.25 GHz
K_a	27-40 GHz	33.4-37.0 GHz
V	40-75 GHz	59.0-64.0 GHz
W	75-110 GHz	76.0-77.5 GHz 78.0-81.0 GHz 92.0-100.0 GHz
mm	110-300 GHz	136.0-155.5 GHz 231.5-235.0 GHz 238.0-248.0 GHz

2.3.1.1. Azimuthal and Elevation Angle. These are the angles that are determined by the pointing angle of the antenna main beam at the time of detection.

2.3.1.2. Range Using the Speed of Light. Since EM waves propagate at the speed of light, it is enough to detect the time interval between the transmitting and receiving back the wave. Equation 2.1 represents the range where c is the speed of light in meters per second ($c \approx 3 \times 10^8$ m/s) and ΔT is the elapsed time.

$$R = \frac{c\Delta T}{2} \quad (2.1)$$

2.3.1.3. Range Rate Using Doppler Frequency Shift. If there is relative motion between the radar and the target, then the frequency of the EM wave reflected from the target and received by the radar will be different from the frequency of the wave transmitted from the radar. This is called the Doppler effect and analogously the difference between the frequency of the received wave and that of the transmitted wave is called the Doppler frequency shift. Equation 2.2 represents an approximation for the Doppler frequency shift, f_d , that holds as long as $v_r \ll c$, where v_r is the radial component of the target's velocity vector toward the radar and λ is the wavelength of the transmitted EM wave. Range Rate, the measurement obtained by the radar, is defined as the negative of the radial velocity.

$$f_d \approx \frac{2v_r}{\lambda} \quad (2.2)$$

2.3.1.4. Polarization. By using the change in the polarization of the EM wave that reflects from the target, a radar may have some information about the geometrical shape of that target. Moreover, this information may help the radar or the operator to discriminate between a target and unwanted reflected wave [19].

2.3.2. Noise and Detection

All objects in the universe with a temperature above absolute zero radiates EM waves called thermal noise. The receiver also has its own internal thermal noise owing to randomly moving electrons while searching occurs. Assuming that a detection occurs, there will be observed voltage sum in the receiver that consists of these noises and the target signal power. If the target signal power is much greater than the noise power, then applying a detection threshold will detect only the targets while ignoring the thermal noise. To illustrate basically, in Figure 2.3 you may see 300 samples two (134th and 283rd) of which have higher frequency than the threshold, thus makes them labeled as targets.

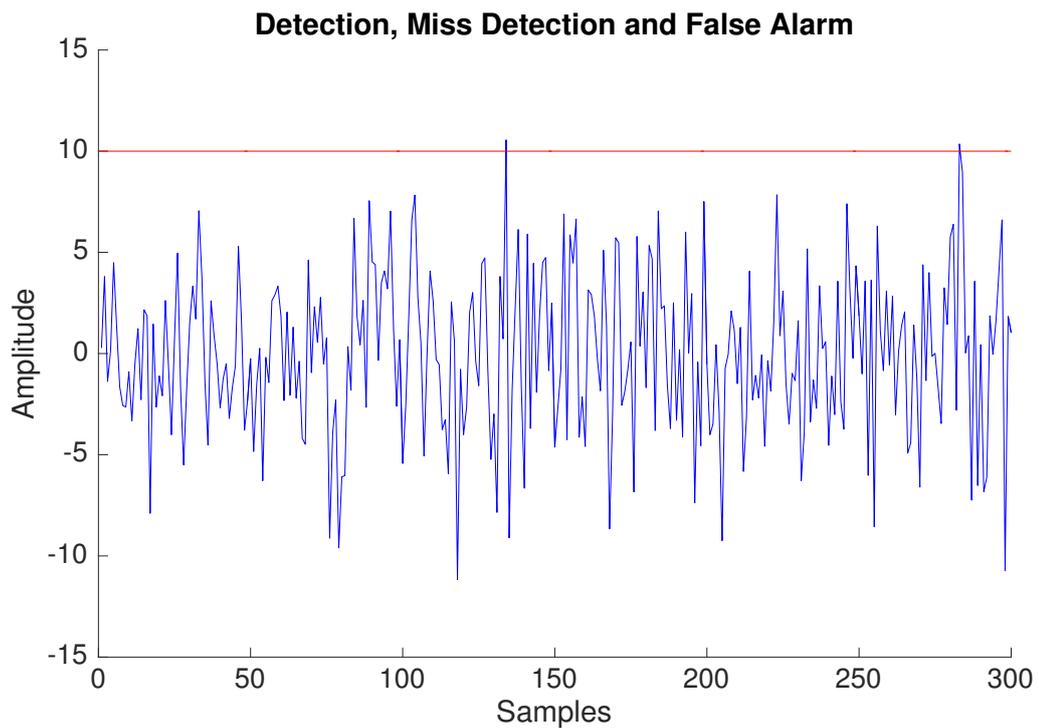


Figure 2.3. Noise and Detection Example: The figure represents 300 samples of echoes obtained in the receiver. The blue line is the samples and red dashed line is the threshold for detection.

However, since the noise in the environment is random, it is possible that the noise signal exceeds the threshold even if no target is present. This is an example of false

alarm. Additionally, owing to this random nature it is also possible that there exists a target but the sum of target signal power and the noise does not exceed the threshold. This is called a missed detection. In Figure 2.3 183rd detection may be regarded as false alarm. The randomness in the process results in specifying the notions in a probabilistic manner. Namely p_D and p_{FA} are called probability of detection and probability of false alarm respectively and we call them as receiver characteristics throughout the thesis (You may see them as Receiver Operating Characteristics(ROC) in some textbooks). By changing the threshold, p_D and p_{FA} can be adjusted. For example increasing the threshold will increase p_D but unfortunately it will also increase p_{FA} that is an undesirable result. To increase p_D while at the same time lowering p_{FA} , the target signal power must be increased relative to the noise power. The ratio of the target signal power to noise power is referred as the signal-to-noise ratio. For the detection theory Bayesian and Neyman Pearson view points exist to determine a good threshold and [24] is a detailed reference on the detection process including how to model signals, determine the threshold and how the threshold effects p_D and p_{FA} etc.

2.4. Basic Radar Functions

While there are hundreds of different types of radars in use, basic functions [19] of the radars can be listed as:

- (i) Searching/Detecting: Almost all radars have to search a given volume and detect targets without a priori information regarding the targets' presence or position. A radar searches a given volume by pointing its antenna in a succession of beam positions that collectively cover the volume of interest. A mechanically scanned antenna moves through the volume continuously. At each position, one or more pulses are transmitted, and the received data are examined to detect any targets. The antenna is then steered to the next beam position, and the process is repeated. This procedure is continued until the entire search volume has been tested, at which point the cycle is repeated.
- (ii) Tracking: Once a target is detected in a given search volume, a measurement is made of the target state, that is, its position in range, azimuth angle, and

elevation angle, and, often, its radial component of velocity. Tracking radars measure target states as a function of time. Individual position measurements are then combined and smoothed to estimate a target track. Individual measurements are invariably contaminated by measurement noise and other error sources. An improved estimate of the target position over time is obtained by track filtering, which combines multiple measurements with a model of the target dynamics to smooth the measurements. This function is the correspondence of the tracking filters that are described in this thesis. Also it must be noted that the optimum radar configurations for tracking and searching are different. Consequently, these search and track functions may be performed by two different radars. This is common in situations where radar weight and volume are not severely limited. When radar weight and volume are limited, as in airborne operations, the search and track functions must be performed by one radar that must then compromise between optimizing search and track functions.

- (iii) Imaging: In radar, imaging is a general term that refers to several methods for obtaining detailed information on discrete targets or broad-area scenes. One application of imaging is to identify the target in order to choose the required response accurately. One technique is to measure a one-dimensional high-range-resolution "image" or two dimensional range/cross-range image of the target. Then by analyzing this image to make an identification decision.

3. CONVENTIONAL MODELING OF TRACKING PROBLEM AND OPTIMAL BAYESIAN FILTER

Tracking is the processing of measurements obtained from a target in order to maintain an estimate of its current state, which typically consist of:

- Kinematic components: position, velocity, acceleration, turn rate, etc.
- Feature components: radiated signal strength, spectral characteristics, radar cross-section, target classification, etc.
- Constant or slowly varying parameters: aerodynamic parameters, etc. [25]

In other words, informally we can address the tracking problem as inferring the current state of the target by getting use of the measurements. We can formalize the problem as follows: Consider the evolution of the state sequence $\{x_t, t \in \mathbb{N}\}$ of a target given by

$$x_t = f_t(x_{t-1}, w_{t-1}) \quad (3.1)$$

where $f_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_x}$ is a possibly non-linear function of the state x_{t-1} , $\{w_{t-1}, t \in \mathbb{N}\}$ is an i.i.d. process noise sequence, n_x, n_w are dimensions of state and process noise vectors, respectively, and \mathbb{N} is the set of natural numbers. The object of tracking is to recursively estimate x_t from measurements

$$y_t = h_t(x_t, v_t) \quad (3.2)$$

where $h_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$ is a possibly non-linear function, $\{v_t, t \in \mathbb{N}\}$ is an i.i.d. measurement noise sequence, n_y, n_v are dimensions of measurement and measurement noise vectors, respectively. In particular, we seek filtered estimates of x_t based on the set of all available measurements $y_{1:t} = \{y_t, t = 1, \dots, t\}$ up to time t [9].

It is also beneficial to emphasize the structure of the model associated with the problem. Since the tracking phenomenon is an instance of dynamical time series modeling, Hidden Markov Model [26–28] is a suitable model, in other words different tracking models are just the different variations of HMM. In addition, the density that we are to infer using HMM is defined as the filtering density, $p(x_t|y_{1:t})$, and this type of problem is called the filtering problem [27]. That’s why in the literature, the methods that are devised to solve the tracking problem are called the filters, e.g. Kalman filter, PHD filter, etc.

In the rest of this section we construct graphical models for the problem in gradually complex stages, respectively for a single target environment, for a multitarget environment without target birth, death, spawning, missed detection and false alarm, lastly for a complete realistic multitarget environment. We believe that modeling the problem in this gradual manner help the reader to understand the problem and modeling the complex environment used in this thesis. Subsequently we derive an optimal filter by exploiting the HMM structure.

3.1. Single Target Modeling

In this model there is only one target in the environment and for each time interval t , we have exactly one measurement that is obtained by the radar from the target. For simplicity assume that both the state and the measurement space is one dimensional and there is only one radar collecting measurements. State and measurements at time t are represented by x_t and y_t respectively. For single target motion and measurement models we have the directed graphical model [27] represented in Figure 3.1.

3.2. A Simple Multitarget Environment Modeling

In this model we have N targets and the radar collects exactly one measurement for each, thus we have N measurement at each time interval t . However, for each time step after collecting the measurements the radar does not have an idea about which measurement belongs to which target. In Figure 3.2 x_t^i and y_t^j represents i th target

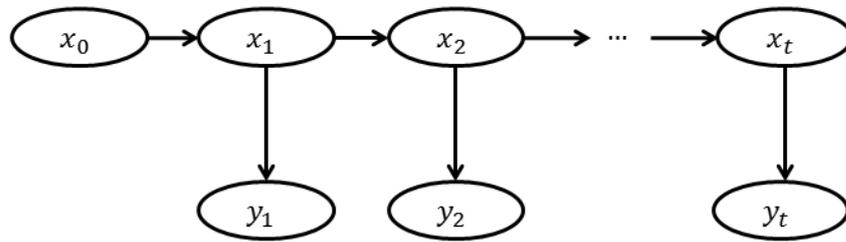


Figure 3.1. Bayesian Network of Single Target Model.

state and j th measurement at time t . Additionally there is a switch variable, n_t for each time interval to infer which of the measurement is the true one. For each different configuration of measurement to state data associations, n_t is set to a different value in order to find the configuration that maximizes the likelihood. For N targets at time t , there are $N!$ different configurations and for each value of n_t we must investigate the the likelihood in order to learn the bayesian network. We will introduce basic data association methodology [29, 30] in Chapter 5.

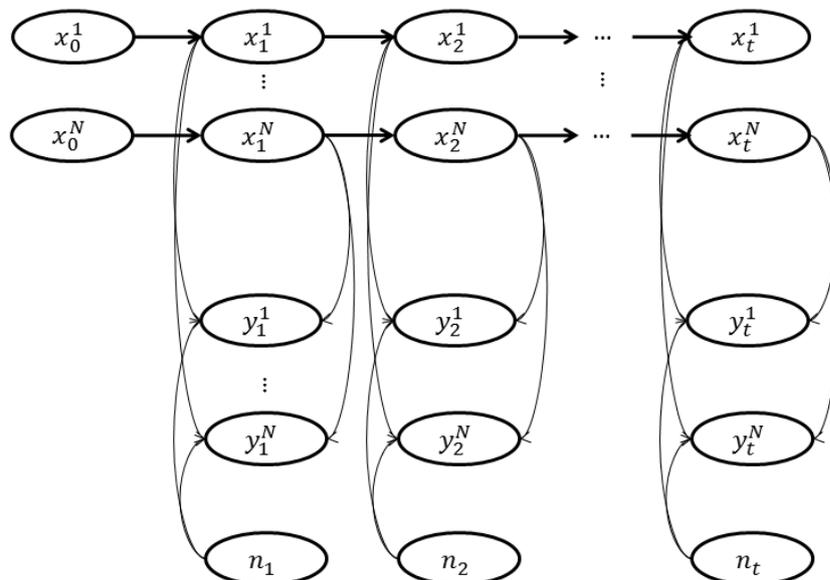


Figure 3.2. Bayesian Network of Basic Multitarget Model.

3.3. Multitarget Modeling: Missed Detection, False Alarm, Target Birth, Death and Spawning

In this section, we introduce a common and realistic model along with some necessary definitions frequently used in multitarget modeling. Initially we represent the directed graphical model in Figure 3.3. As can be seen from this figure, major difference from Figure 3.2 is the number of targets and measurements in the environment are also dynamic. Namely at each time interval a new target may be born, or an existing target may spawn into new targets or die resulting in a possible variation of the target number for each time interval, denoted by N_t in Figure 3.3. For the measurements, the radar may not detect all the targets for that time interval (Missed Detection) or may signal a detection even though a target does not exist (False Alarm), thus in this model number of measurements is not necessarily equal to the number of targets (see Subsection 2.3.2). Furthermore, possible number of configurations for a switch variable of time t , n_t , also increases since we must investigate not only the possibility of belonging to a specific target for a measurement but also the probability of being a false alarm and a new born target.

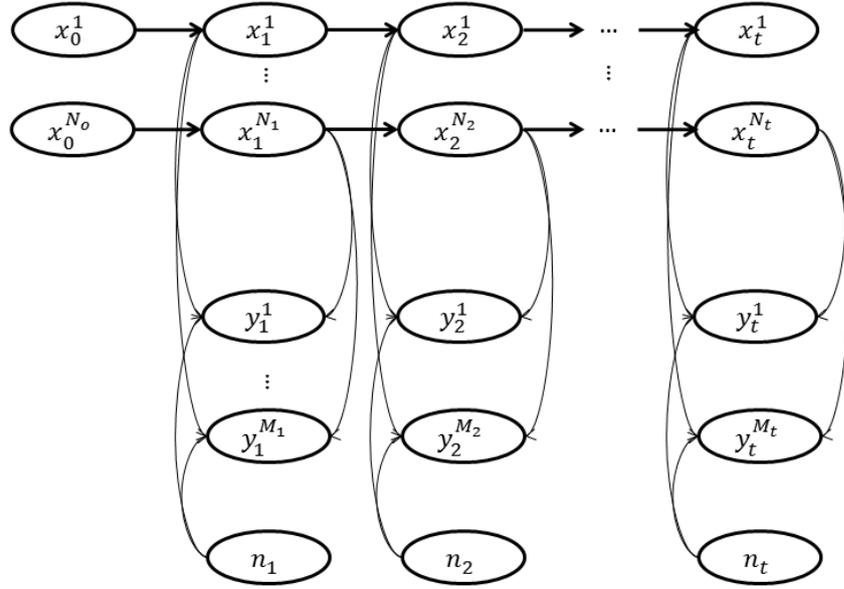


Figure 3.3. Bayesian Network of Realistic Multitarget Model.

Based on these explanations, we list some definitions and symbolization for multitarget tracking as follows:

- p_B : Probability of birth of new target
- p_S : Probability of spawning of an existing target into new targets
- p_{FA} : Probability of being false alarm for a measurement
- p_D : Probability of detection a target for the radar

Also note that these probabilities can also be adjusted in the model such that they may fluctuate over time. In the next section we will derive an optimal filter as a solution of this problem.

3.4. Optimal Bayesian Filter

The purpose of Bayesian inference is to provide a mathematical machinery that can be used for modeling and thereby learning a system, in which the uncertainties of the system are taken into account and the decisions are made according to the inferred knowledge from the model. The tools of this machinery are the probability distributions and the rules of probability calculus. [31] For the problem that we have defined up to now, we use these tools and derive a filter called optimal Bayesian filter [31, 32] that aims to compute the filtering density, $p(x_t|y_{1:t})$. Sequential state estimation by using Bayesian perspective was first mentioned in [33] and now it has been investigated by many researchers for a wide range of applications.

Assume that prior distribution, $p(x_0)$, measurement and target motion model, measurements up to time t , $y_{1:t}$, are known, states follow a first order Markov process and the observations are independent of the given states, we must derive $p(x_t|y_{1:t})$ from $p(x_{t-1}|y_{1:t-1})$.

- (i) Initialization: We initialize the filter by setting the prior, $p(x_0)$, as the initial filtering density.
- (ii) Prediction: Assuming that we know the filtering density of previous step,

$p(x_{t-1}|y_{1:t-1})$, the joint distribution of x_t, x_{t-1} can be computed by using Markov property as

$$\begin{aligned} p(x_t, x_{t-1}|y_{1:t-1}) &= p(x_t|x_{t-1}, y_{1:t-1})p(x_{t-1}|y_{1:t-1}) \\ &= p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1}) \end{aligned}$$

Integrating over x_{t-1} gives Chapman-Kolmogorov equation;

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \quad (3.3)$$

This is the prediction step of the filter.

- (iii) Update: Since we have $p(x_t|y_{1:t-1})$ and the measurement model, we can compute the filtering density by using Bayes' Rule.

$$p(x_t|y_{1:t}) = \frac{1}{Z_t}p(y_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1})$$

Since the measurements are conditionally independent, $y_{1:t-1}$ of $p(y_t|x_t, y_{1:t-1})$ can be dropped:

$$p(x_t|y_{1:t}) = \frac{1}{Z_t}p(y_t|x_t)p(x_t|y_{1:t-1}) \quad (3.4)$$

where the normalization constant, $Z_t = \int p(y_t|x_t)p(x_t|y_{1:t-1})dx_t$.

In the following chapter we devise two filters based on optimal Bayesian filter, Kalman filter and particle filter.

4. SINGLE TARGET TRACKING TECHNIQUES

The traditional method of obtaining tracks with a surveillance radar has been to have an operator manually mark with grease pencil on the face of the cathode ray tube the location of the target on each scan. The simplicity of such a procedure is offset by the poor accuracy of the track. It was found that the accuracy of track can be improved by using a computer to determine the trajectory from inputs supplied by an operator. Again, it came that a human operator cannot update target tracks at a rate greater than about once per two seconds and its effectiveness of detecting new targets decreases after about half an hour of operation [34]. Then with the advent of Automatic Detection and Targeting, operations in the tracking process (such as the target detection, track initiation, track association, track update, track smoothing or filtering and track termination) have begun to be carried out by the computers. After the handover to the computers, one of the first tracking algorithms presented by Sklansky in 1957 is α - β filter [3, 20, 21, 35]. While α - β filter is improved to α - β - γ filter [35] and α - β - γ - δ filter [35] in time, the initial version computes the smoothed position of the target by following equations:

$$x_{t|t-1} = x_{t-1|t-1} + \nu_{t-1}\Delta T \quad (4.1)$$

$$x_{t|t} = x_{t|t-1} + \alpha(y_t - x_{t|t-1}) \quad (4.2)$$

$$\nu_t = \nu_{t-1} + \frac{\beta}{\Delta T}(y_t - x_{t|t-1}) \quad (4.3)$$

where $x_{t|t-1}$ is the predicted position, $x_{t|t}$ is the updated position and ΔT is the time interval. The selection of the α and β parameters is a design trade off. The simplicity and the compactness of the filter is unequivocal. First we predict the position according to the law of motion, then we inject the measurement and proceed recursively.

In addition to α - β filter, one of the first tracking filters named after Rudolph E. Kalman, who offered a recursive solution to the linear data processing problem in 1960, is Kalman filter [36]. Since then, Kalman filter has become a popular tool in a wide

range of applications, thus there are numerous tutorials and books on Kalman filter including modified versions and its implementation on a particular field [5, 35, 37–40]. Particularly [25], [41] and [10] handle the topic from a radar tracking perspective. In this chapter, we present Kalman filter along with two extensions for the nonlinear environments of this filter, Extended Kalman filter and Unscented Kalman filter and then we provide an another approximation strategy known as particle filter.

4.1. Kalman Filter

Kalman filter is derived from optimal Bayesian filter by imposing linearity and Gaussian assumptions. Firstly to represent the model to be dealt with, we go through the following stochastic process based on Figure 3.1.

$$\begin{aligned}
 x_t &= Ax_{t-1} + w_{t-1} & (4.4) \\
 y_t &= Cx_t + v_t \\
 w_t &\sim \mathcal{N}(w_t; 0, Q) \\
 v_t &\sim \mathcal{N}(v_t; 0, R) \\
 x_0 &\sim \mathcal{N}(x_0; \mu_0, P_0)
 \end{aligned}$$

Note that this model is a special case of the general model that is described in Chapter 3 by imposing the following assumptions.

- the noise parameters, w_t and v_t , are statistically independent and drawn from $\mathcal{N}(0, Q)$, $\mathcal{N}(0, R)$ respectively such that Q and R are known.
- State transition and measurement models are normally distributed with known matrices A , C that defines linear functions.
- x_0 is drawn from Gaussian with known parameters.

Holding these conditions Kalman filter recursion cycle consists of following equations:

- (i) Initialization: Set $\mu_{0|0}$ and $P_{0|0}$ from prior, $\mathcal{N}(\mu_0, P_0)$

(ii) Prediction:

$$\mu_{t|t-1} = A\mu_{t-1|t-1} \quad (4.5)$$

$$P_{t|t-1} = AP_{t-1|t-1}A^T + Q \quad (4.6)$$

(iii) Update:

$$\mu_{t|t} = \mu_{t|t-1} + K_t(y_t - C\mu_{t|t-1}) \quad (4.7)$$

$$P_{t|t} = (\mathbf{I} - K_tC)P_{t|t-1} \quad (4.8)$$

where K_t matrix is Kalman gain such that $K_t = P_{t|t-1}C^T(CP_{t|t-1}C^T + R)^{-1}$; $\mu_{t|t-1}$, $\mu_{t|t}$ are predicted and updated mean values and $P_{t|t-1}$, $P_{t|t}$ are predicted and updated covariance matrices for the corresponding Gaussian distribution respectively. As seen the filter propagates the first and second order moments of the Gaussian distribution and by these sufficient statistics online estimation is processed.

On a toy example of estimating the altitude of a hovering helicopter (or more informally estimating a constant number), we explain the notion of filtering as follows. The model is described in Table 4.1.

Table 4.1. Hovering Helicopter Model.

Transition Model	Measurement Model
$x_t = x_{t-1} + w_{t-1}$	$y_t = x_t + v_t$
$x_0 \sim \mathcal{N}(x_0; 1000, 10)$	$v_t \sim \mathcal{N}(v_t; 0, R = 10)$
$w_t \sim \mathcal{N}(w_t; 0, Q = 0.0001)$	

In Figure 4.1 you may see tracking altitude of a hovering helicopter by Kalman filter for 150 seconds. When more measurements are collected, estimated mean is

corrected and also variance decreases. Stabilizing the filter by decreasing the variance constitutes the desirable property of a filter. When we set R of the Kalman filter to 10 that is the value in the model, the variance asymptotically converges to 0 and the filter rapidly have approximate values for the mean although initialized from 995.

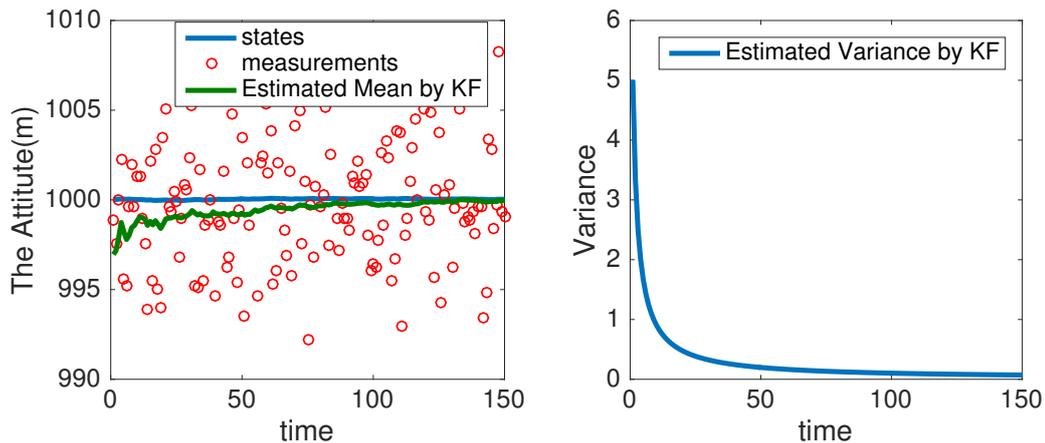


Figure 4.1. Tracking the Attitude of a Helicopter by Kalman Filter: Mean estimations are on the left and covariance estimations are on the right.

Next we investigate what if we determine R parameter of the model wrongly on two examples. In Figure 4.2, while we keep the measurements the same, we have changed the value of R by a 0.01 and 100 factors respectively in order to show the effect of the parameters on the estimation. In the upper pair of the subfigures of Figure 4.2, you may see that the filter responds excessively to each measurement since the filter is told that the variance of the measurements are very low. In other words it believes the measurements too much due to their dummy exactness. This is followed by good variance estimates of the filter but unstable mean (position) estimates. In the lower pair of the figures, the filter does not respond to the measurements since the filter recognizes the measurements as unbelievable due to their variance of 1000. Despite 150 measurements, both the filter can not reach to the attitude of the helicopter and the variance is still high and does not seem to converge.

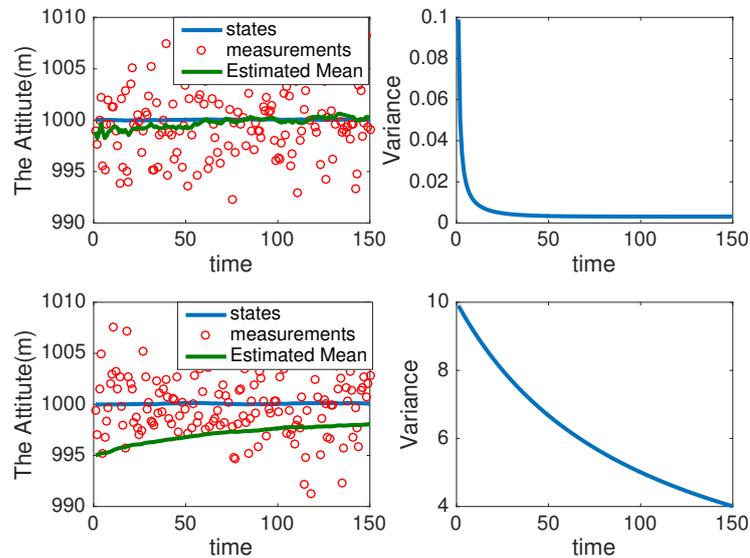


Figure 4.2. Erroneous Estimation Example: Tracking the attitude of a hovering helicopter with wrong model parameters, $R = 0.1$ for the upper pair of figures and $R = 1000$ for the lower pair of figures.

4.2. Extended Kalman Filter

Many natural dynamical models are nonlinear in nature. While there are a lot of variations of the Kalman filter for solving the non-linearity problem such as Gaussian sum filter, Point-mass approach, spline function and Fourier series expansion, extended Kalman filter is the one that is the most prevalent among engineers. We know that there are modern air surveillance systems in Turkish Armed Forces performing tracking via a kind of extended Kalman filter (EKF). Nonlinearity prevents $P(x_i|y_1, \dots, y_{t-1})$ and $P(x_i|y_1, \dots, y_t)$ to be Gaussian and may cause them not to have any particular probability distribution, so any form of sufficient statistics can not be propagated in time. However Extended Kalman filter linearizes the model locally by exploiting Taylor series expansion¹ of the nonlinear functions $f_t(\cdot)$ and $h_t(\cdot)$ of Equation 3.1 and Equation 3.2 respectively. Taylor series expansion of the function $f_t(x_{t-1}, 0)$ and $h_t(x_t, 0)$ about

¹Taylor Series Expansion of a real function $f(x)$ about a point $x = a$ is given by
$$f(x) = f(a) + f'(a)(x - a) + \frac{f^{(2)}(a)}{2!}(x - a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x - a)^n + \dots$$

$x_{t-1|t-1}$ is (Note that we excluded the noise term by setting it to 0):

$$f_t(x_{t-1}, 0) = f_t(x_{t-1|t-1}, 0) + f'_t(x_{t-1|t-1}, 0)(x_{t-1} - x_{t-1|t-1}) + \dots \quad (4.9)$$

$$h_t(x_t, 0) = h_t(x_{t|t-1}, 0) + h'_t(x_{t|t-1}, 0)(x_t - x_{t|t-1}) + \dots \quad (4.10)$$

Thus by omitting the terms after the second term, EKF performs the approximation to the point and this is called local linearization. After incorporating the white noises of the transition and measurement models, the model looks like those in the Kalman filter:

$$x_t \approx f_t(x_{t-1|t-1}, 0) + A_{t-1}(x_{t-1} - x_{t-1|t-1}) + w_{t-1} \quad (4.11)$$

$$y_t \approx h_t(x_{t|t-1}, 0) + C_t(x_t - x_{t|t-1}) + v_t \quad (4.12)$$

$$w_t \sim \mathcal{N}(w_t; 0, Q) \quad (4.13)$$

$$v_t \sim \mathcal{N}(v_t; 0, R) \quad (4.14)$$

$$x_0 \sim \mathcal{N}(x_0; \mu_0, P_0) \quad (4.15)$$

such that A_{t-1} and C_t Jacobian matrices defined as:

$$A_{t-1} = \left. \frac{\partial f_t(x, 0)}{\partial x} \right|_{x=x_{t-1|t-1}}, C_t = \left. \frac{\partial h_t(x, 0)}{\partial x} \right|_{x=x_{t|t-1}} \quad (4.16)$$

Resulting EKF recursion equations are as follows:

(i) Initialization: Set $\mu_{0|0}$ and $P_{0|0}$ from prior, $\mathcal{N}(\mu_0, P_0)$

(ii) Prediction:

$$\mu_{t|t-1} = f_t(\mu_{t-1|t-1}, 0) \quad (4.17)$$

$$P_{t|t-1} = A_{t-1}P_{t-1|t-1}A_{t-1}^T + Q \quad (4.18)$$

(iii) Update:

$$\mu_{t|t} = \mu_{t|t-1} + K_t(y_t - h(\mu_{t|t-1}, 0)) \quad (4.19)$$

$$P_{t|t} = (\mathbf{I} - K_t C_t) P_{t|t-1} \quad (4.20)$$

$$(4.21)$$

Before passing to another variation of the Kalman filter, we refer to three more points here. Firstly, one may take into account any number of the terms from the Taylor expansion even though we considered only two terms. Clearly this will make the linearization approximation more precise but result in expensive computation. Secondly we may also allow the noises be dynamically change in the process, however, we have assumed the noises do not change in time. Finally, we list the shortcomings of EKF as linearization can produce highly unstable filters if the assumptions of local linearity is violated and derivation of the Jacobian matrices are nontrivial [42]. Thus in the next section we introduce another approximation methodology for the nonlinear model, Unscented Kalman filter, that can resolve these limitations.

4.3. Unscented Kalman Filter

Linearization process of the EKF assumes that all second and higher order terms of Taylor expansion are negligible, but sometimes these higher order terms, effecting the posterior significantly, decrease the performance of the EKF filter. Polar to cartesian coordinate transformation in [43] is an example of this this situation. Unscented Kalman filter (UKF) is inspired from the intuition that it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function or transformation [44]. Hence, unlike EKF, UKF aims to approximate the posterior by using a set of transformed samples that are called sigma points. Although at first glance the technique may seem similar to particle filter, it is different owing to the nature of sampling which is not random for UKF. The samples to be transformed are well defined and exhibit certain characteristics. From a broad perspective the algorithm proceeds as follows: after initialization we select sigma

points, then prediction step transforms the sigma points by using the nonlinear function (unscented transformation) and the update step are performed respectively. Following we define unscented transformation before passing on to the recursion equations.

4.3.1. Unscented Transformation

Unscented transformation is the process of applying a nonlinear function to the sigma points that have x_{t-1} mean and P_{t-1} variance in order to compute the statistics of the posterior, particularly x_t and P_t , by using the transformed points. In the case of non-additive transition and measurement noise, the unscented transformation scheme is applied to the augmented state [45]:

$$x_t^{aug} = \begin{bmatrix} x_t^T & w_{t-1}^T & v_t^T \end{bmatrix}^T \quad (4.22)$$

You may see the depiction of the principle of the unscented transformation in Figure 4.3.

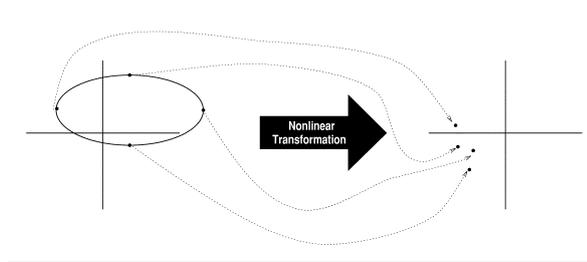


Figure 4.3. Principle of the Unscented Transformation [43].

4.3.2. UKF Algorithm

UKF algorithm consists of the following steps:

- (i) Initialization: Set $\mu_{0|0}$ and $P_{0|0}$ from prior, $\mathcal{N}(\mu_0, P_0)$

- (ii) Selection of Sigma Points: We need a total of $2n + 1$ sigma points and their associated weights where n is the dimension of the state space. There are some methods for selecting the sigma points such as a general method called scaled unscented transformation [46]. Setting w^0 such that $-1 < w^0 < 1$ and $x_{t-1}^0 = \mu_{t-1|t-1}$ and satisfying the condition $\sum_{i=0}^{2n} w^i = 1$ for the weights, following procedure is an instance of the unscented transformation.

$$x_{t-1}^i = x_{t-1}^0 + \left(\sqrt{\frac{n}{w^0} P_{t-1|t-1}} \right)_i, i = 1, \dots, n \quad (4.23)$$

$$x_{t-1}^{i+n} = x_{t-1}^0 - \left(\sqrt{\frac{n}{w^0} P_{t-1|t-1}} \right)_i, i = 1, \dots, n \quad (4.24)$$

$$w^i = \frac{1 - w^0}{2n}, i = 1, \dots, 2n \quad (4.25)$$

such that $\left(\sqrt{\frac{n}{w^0} P_{t-1|t-1}} \right)_i$ is the i th row or column of the matrix square root of $\frac{n}{w^0} P_{t-1|t-1}$. Determining the w^0 effects the points to be accumulated or scattered with respect to the origin. For example if $w^0 > 0$, sigma points move further from the origin.

- (iii) Prediction: In this part sigma points are instantiated through the transition model by using the unscented transformation to calculate predicted mean and covariance. The procedure is as follows:

$$x_t^i = f(x_{t-1}^i, 0), i = 0, 1, \dots, 2n \quad (4.26)$$

$$\mu_{t|t-1} = \sum_{i=0}^{2n} w^i x_t^i \quad (4.27)$$

$$P_{t|t-1} = \sum_{i=0}^{2n} w^i (x_t^i - \mu_{t|t-1})(x_t^i - \mu_{t|t-1})^T + Q \quad (4.28)$$

- (iv) Update: In this part x_t^i 's are instantiated through the measurement model. By using these points we calculate innovation covariance matrix and cross covariance matrix, P_t^y, P_t^{xy} respectively. Then by using Kalman filter equations,

measurement update is performed.

$$y_t^i = h(x_t^i, 0), i = 0, 1, \dots, 2n \quad (4.29)$$

$$\hat{y}_t = \sum_{i=0}^{2n} w^i y_t^i \quad (4.30)$$

$$P_t^y = \sum_{i=0}^{2n} w^i (y_t^i - \hat{y}_t)(y_t^i - \hat{y}_t)^T + R \quad (4.31)$$

$$P_t^{xy} = \sum_{i=0}^{2n} w^i (x_t^i - \mu_{t|t-1})(y_t^i - \hat{y}_t)^T \quad (4.32)$$

$$K_t = P^{xy}(P_t^y)^{-1} \quad (4.33)$$

$$\mu_{t|t} = \mu_{t|t-1} + K_t(z_t - \hat{y}_t) \quad (4.34)$$

$$P_{t|t} = P_{t|t-1} - K_t P_t^y K_t^T \quad (4.35)$$

We assumed that the noise is additive and constant but it may vary in time and may not be additive. We can incorporate time dependent noise into the model by adding a subscript to the noise parameters, Q and R . You may also find the derivation and general equations of the filter for the augmented state vector, x_t^{aug} , that is constructed when the noise is nonadditive in [43].

Until now we considered a limited set of solutions for single target environment, since Kalman filtering based techniques imposes Gaussian assumption to the model. In the next section we represent how we drop this assumption by accepting an approximation using Sequential Monte Carlo techniques.

4.4. Sequential Monte Carlo in Target Tracking

In a non Gaussian and non linear environment, optimal algorithms tend to fail due to the complexities of the multidimensional integrals. Thus as an alternative, approximation methods are prevalent to find a solution to these problems. However, these approaches, particularly Monte Carlo Methods, provide only approximate and suboptimal solutions. Monte Carlo Methods are random sampling approaches that

solve difficult numerical integration problems. Since their first use in 1949 by Metropolis and Ulam [7], Monte Carlo methods have been explored to address many intractable problems including target tracking under a class of MC Methods called particle filter. Particle filter was first introduced formally to Bayesian target tracking in 1993 [8] as a combination of Monte Carlo sampling and Bayesian statistics. The particle filter approximates the posterior by a set of random samples with associated weights. As the number of the samples becomes very large, computed distribution gets closer to the posterior distribution. Particle filter gets use of an online extension of Importance Sampling and Resampling methods to compute the filtering density, $p(x_t|y_{1:t})$. Thus in this section we introduce the topics that the Particle filter logic based upon.

4.4.1. Importance Sampling

Consider a target distribution $\pi(x) = \phi(x)/Z$ where the non-negative function $\phi(x)$ is known, but the overall normalization constant Z is assumed to be computationally intractable. The main idea of importance sampling (IS) is to estimate expectations $E_\pi(\varphi(x))$ by using weighted samples from a tractable IS distribution $q(x)$. Note that unlike other MC methods, our goal is not directly generating samples but estimating expectations via weighted samples. More specifically, we can write the normalization constant as

$$Z = \int \phi(x)dx = \int \frac{\phi(x)}{q(x)}q(x)dx = \int W(x)q(x)dx = E_q(W(x))$$

where $W(x) \equiv \phi(x)/q(x)$ is called weight function. Therefore we have

$$E_\pi(\varphi(x)) = \frac{1}{Z} \int \varphi(x) \frac{\phi(x)}{q(x)} q(x) = \frac{E_q(\varphi(x)W(x))}{E_q(W(x))}$$

A Monte Carlo estimate of $E_\pi(\varphi(x))$ is then given by

$$\mu_N = \frac{\sum_{i=1}^N W^{(i)}\varphi(x^{(i)})/N}{\sum_{i=1}^N W^{(i)}/N} \quad (4.36)$$

where $W^{(i)} \equiv W(x^{(i)})$ and the 'particles' are sampled from $q(x)$. Using normalized weights $w^{(i)} \equiv W^{(i)} / \sum_{i'=1}^N W^{(i')}$ we can write the approximation as

$$\mu_N = \sum_{i=1}^N w^{(i)} \varphi(x^{(i)}) \quad (4.37)$$

In other words, instead of sampling from the target density $p(x)$, we sample from a different tractable distribution $q(x)$ and correct by reweighing the samples accordingly.

4.4.2. Resampling

A practical issue is that, unless the IS sampling density $q(x)$ is close to the target density $\pi(x)$, the normalized weights will typically have mass in only a single component. This can be partially addressed using resampling. Resampling is a randomized pruning algorithm where we discard particles with low weight and increase the number of particles with high weight. As an input to the resampling, we have the weighted particles that are generated by Importance Sampling. Following are the two common techniques for resampling.

Multinomial resampling equivalent to the inversion method when the target is a discrete distribution. Here, we view the weighted sample set $x^{(i)}$ as a discrete distribution with categories $i = 1 \dots N$ where the probability of the i th category is given by the normalized weight $w^{(i)}$. To sample M times independently from this target, we generate $u^{(j)} \sim U(0, 1)$ for $j = 1 \dots M$ and we obtain an unweighted set of particles by evaluating the generalized inverse at each $u^{(j)}$.

Systematic resampling is quite similar to multinomial resampling, only the generation of $u^{(j)}$ is not entirely random but systematic. To generate M samples we only select $u^{(1)}$ uniformly random from the interval $[0, 1/M]$ and set $u^{(j)} = u^{(1)} + (j - 1)/M$. Therefore, each $u^{(j)}$ is located on a uniform grid with a random initial shift.

4.4.3. Sequential Importance Sampling

We now apply importance sampling to the tracking problem. The goal is to draw samples from the posterior

$$p(x_{1:t}|y_{1:t}) = \underbrace{p(y_{1:t}|x_{1:t})p(x_{1:t})}_{\phi(x_{1:t})} / \underbrace{p(y_{1:t})}_{Z_t} \quad (4.38)$$

where we assume that the normalization term Z_t is intractable. An importance sampling approach uses at each time t an importance distribution $q_t(x_{1:t})$, from that we draw samples $x_{1:t}^{(i)}$ with corresponding importance weights.

$$W_t^{(i)} = \frac{\phi(x_{1:t}^{(i)})}{q_t(x_{1:t}^{(i)})} \quad (4.39)$$

The key idea in particle filter is the sequential construction of the IS distribution q and the recursive calculation of the importance weights. Without loss of generality, we may write

$$q_t(x_{1:t}) = q_t(x_t|x_{1:t-1})q_t(x_{1:t-1}) \quad (4.40)$$

In particle filtering, one chooses an IS proposal q that only updates the current x_t and leaves previous samples unaffected. This is achieved using

$$q_t(x_{1:t}) = q_t(x_t|x_{1:t-1}, y_{1:t})q_{t-1}(x_{1:t-1}) \quad (4.41)$$

As we are free to chose the IS proposal q fairly arbitrarily, we can also construct the proposal on the fly conditioned on the observations seen so far:

$$q_t(x_{1:t}|y_{1:t}) = q_t(x_t|x_{1:t-1}, y_{1:t})q_{t-1}(x_{1:t-1}|y_{1:t-1}) \quad (4.42)$$

we will not include $y_{1:t}$ in the notation but it should be understood that the proposal q can be constructed at the observations so far.

Due to the sequential nature of the tracking problem and q , the weight function $W_t(x_{1:t})$ admits a recursive computation

$$\begin{aligned} W_t(x_{1:t}) &= \frac{\phi(x_{1:t})}{q_t(x_{1:t})} = \frac{p(y_t|x_t)p(x_t|x_{t-1}) \prod_{i=1}^{t-1} p(y_i|x_i)p(x_i|x_{i-1})}{q(x_t|x_{1:t-1}) \prod_{i=1}^{t-1} q(x_i|x_{1:i-1})} \\ &= \underbrace{\frac{p(y_t|x_t)p(x_t|x_{t-1})}{q(x_t|x_{1:t-1})}}_{v_t} W_{t-1}(x_{1:t-1}) \end{aligned} \quad (4.43)$$

where v_t is called the incremental weight. Particle filtering algorithms differ in their choices for $q_t(x_t|x_{1:t-1})$. The optimal choice (in terms of reducing the variance of weights) is the one step filtering distribution

$$q_t(x_t|x_{1:t-1}) = p_t(x_t|x_{t-1}, y_t) \quad (4.44)$$

However, sampling from this filtering density is often difficult in practice, and simpler methods are required. The popular bootstrap filter uses the model transition density as the proposal

$$q_t(x_t|x_{1:t-1}) = p_t(x_t|x_{t-1}) \quad (4.45)$$

for that the incremental weight is $v_t = p(y_t|x_t)$. For the bootstrap filter, the IS distribution does not make any use of the recent observation and therefore has the tendency to lose track of the high-mass regions of the posterior. Indeed, it can be shown that the variance of the importance weights for the bootstrap filter increases in an unbounded fashion so that the state estimates are not reliable. In practice, therefore, after a few time steps the particle set typically loses track of the exact posterior mode. A crucial extra step to make the algorithm work is resampling that prunes branches with low weights and keeps the particle set located in high probability regions. It can be shown that although the particles become dependent due to resampling, the estimations are still consistent and converge to the true values as the number of particles increases to infinity [47].

In order to represent the internal dynamics of the PF algorithm, we illustrate it by a toy example that is modeled according to Table 4.2. Data is predefined and

Table 4.2. Particle Filter Toy Example Model.

Transition Model	Measurement Model
$x_t = Ax_{t-1} + w_{t-1}$ where $A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	Target in range of i th radar; $y_t(i) = d_t(i) + v_t$ else $y_t(i) = \emptyset$ where $d_t(i) = \ \text{RadarLoc}(i) - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} x_t\ $ $y_t(i)$ is the i th element of measurement vector y_t
$w_t \sim N(w_t; 0, 0.01)$	$v_t \sim N(v_t; 0, 0.1)$
$p(x_0 = (0, -0.2, 0, -0.5)') = 1$	

generated in advance according to transition model in order to enable the target pass through the areas where the radars are located and the prior information is known. In other words, the system knows where the target will come. The location of the radars, the trajectory of the target and the expectation of the trajectory inferred by PF Algorithm along with ten particles' per each time interval during 1000 seconds can be observed in Figure 4.4.

From a general point of view the intuitive recursion of the algorithm is as follows. After initialization, the particles are propagated according to the transition model (This corresponds to the prediction step) and importance weights are evaluated according to the observation model (This corresponds to the update step) due to bootstrap filter choice for filtering. Then new weights of the particles are assigned in a recursive manner. Finally the algorithm proceeds with resampling. In the example there are three zones with radar located and between them two zones that are radar free. In the first radar covered zone that has radar1, radar2 and radar3; since we have three range measurements, the probability concentrates on specific locations. In addition, whenever the particles moving according to the proposal, they tend to scatter. However owing to resampling, the scattered particles with less weights are discarded, hence the particles gathered such that their likelihoods are greater. Basically the role of the resampling

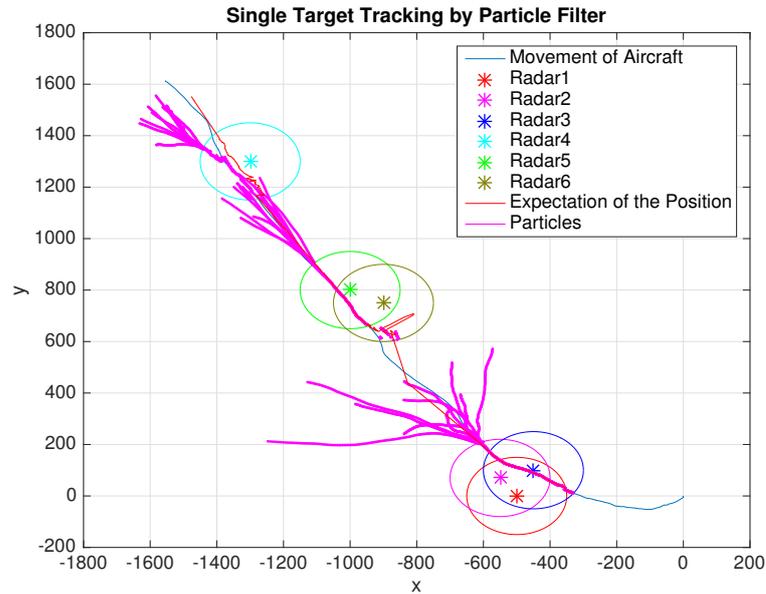


Figure 4.4. Tracking an Aircraft by Particle Filter: The figure represents the behavior of the particles when the target is detected and not detected. At each time step 10 particles are represented in the figure.

to prevent the domination of a single particle by increasing the possibility of giving offsprings of particles with greater weights. After the target leaves the zone of the radars and come to radar free zone, the particles tends to scatter since we do not have any measurement, hence the algorithm assumes the uniform importance weights. The particles just makes a random walk in the space according to the proposal. However, you can observe the gathering of the particles as soon as a new measurement data is available which is also apparent in the second radar zone. Finally for the last radar zone that has only a single range only radar we observe that estimated trajectory does not fit the trajectory of the target although the projected ten particles are in the right track. The reason is simply as follows: This is a range only radar, so it generates particles that reflects the model. However a single range only radar is not enough for detecting the position correctly. The particle filter algorithm is represented in Figure 4.5.

In this chapter we considered Kalman filter and SMC based approaches for a single target noncluttered environment. As a result of this we have just one target and measurement, thus we know in advance that which measurement belongs to which target. However, all variations of these filters can be used also for multitarget tracking including many real life applications such as the one that we gave as an example from the Turkish Armed Forces in the beginning of the this chapter. The requirement in a multitarget environment is measurement to track association (also called data association) methodology. As long as this association is performed, these filters can propagate either larger covariance matrices and mean vectors or larger particle sets, hence can perform multitarget tracking. In the next chapter we consider this problem and refer to a method as a solution to data association problem from a Bayesian perspective.

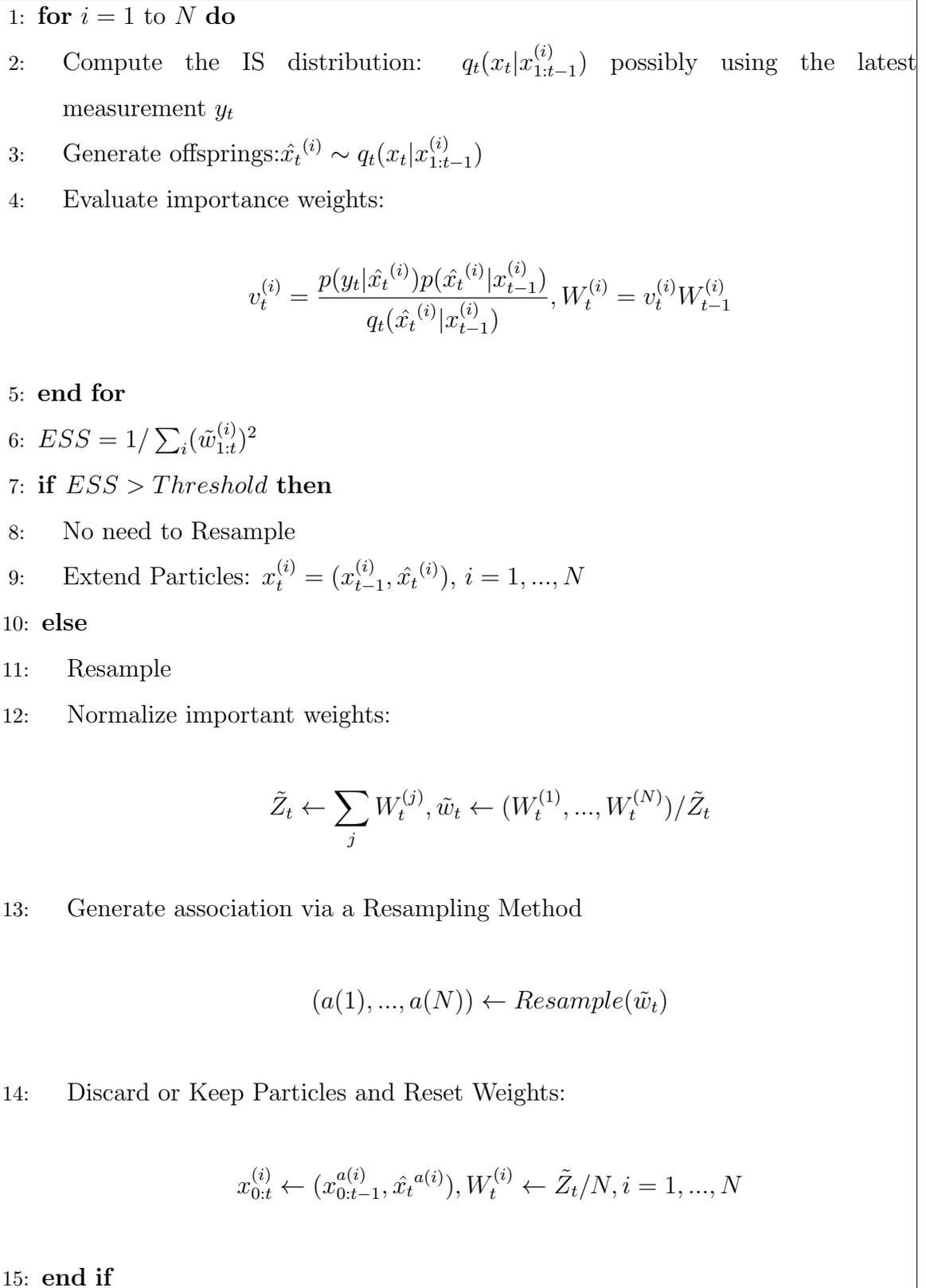


Figure 4.5. Particle Filter Algorithm

5. EXTENDING TO MULTITARGET ENVIRONMENT: DATA ASSOCIATION

In Chapter 4 we provided the basic infrastructure of the target tracking including the filtering phenomenon, linear, nonlinear models and various model transition distributions. Since allocating a radar for each target is not doable in the real world, tracking algorithms need to be designed such that the radars can maintain the information of several targets simultaneously. For example a single AN/MPQ-64 (Sentinel) radar can track more than 50 targets simultaneously and can prioritize them to enable the operator override the system manually to examine targets of particular interest [48]. In this chapter we present the basic notions for multitarget tracking. Throughout this chapter we get use of the correspondent model for multitarget environment, that is, the one presented in Figure 3.3. Initially we explain transition and measurement models, then we introduce the essential steps of multitarget tracking and finally we represent global nearest neighbor data association technique.

5.1. Transition and Measurement Models

In this section we explain formally the assumptions of conventional transition and measurement models respectively in multitarget environments. Conventional multitarget transition model is based on following assumptions [10]:

- The motions of the targets are statistically independent.
- New targets appear in the scene uniformly distributed and with Poisson-time arrivals.
- p_S is the survival probability of a target from time t to $t + 1$.

Similarly, conventional multitarget measurement model is based on following assumptions [10]:

- A single sensor observes a scene involving an unknown number of unknown targets.
- No target generated measurement is generated by more than a single target.
- A single target generates either a single measurement with probability p_D or no measurement with probability $1 - p_D$.
- The false alarm process is Poisson distributed in time and uniformly distributed in space.
- Target generated measurements are conditionally independent of state.
- The false alarm process and target measurement process are statistically independent.

As we described above we may have no measurement or more than one measurement for each existing track and at the end of each step we do not know explicitly which track ensues which measurement. The only information that we have is a list of measurements that are gathered now and the tracks that are estimated in the previous time stamp. The process of choosing the best measurement for each track is called data association. Independent of the algorithm that is preferred, the process of data association is depicted in Figure 5.1. The intuition of the process is that after we collect the measurements we predict the next position of each target via the prediction equations of the filter that we use, in the gating step we limit the possible number of associations, then we carry out the association by using a data association algorithm and lastly we just update all the predicted tracks by using the correspondent measurement.

Since the prediction and the update parts are the parallel processed single target tracking filters, in this part we investigate the gating and measurement-track association steps.

5.2. Gating

As a result of the combinatorial explosion in the number of possible measurement-track association hypotheses, we cannot consider all measurement to

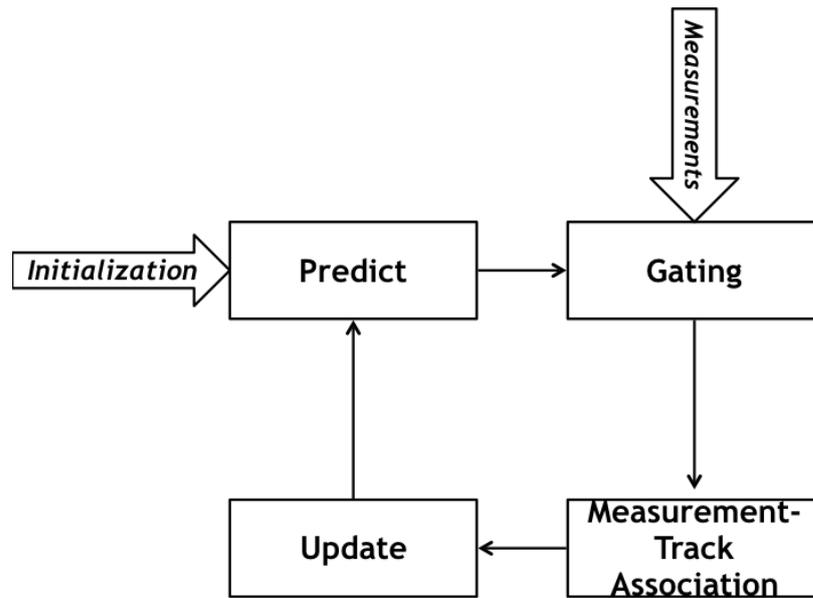


Figure 5.1. Data Association Process: In addition to the conventional recursive steps of a filter, gating and measurement to track association steps are incorporated in the process.

track combinations. Hence, we need a method of decreasing the number of hypothesis to a manageable amount. This method is called gating (a.k.a. validation of candidates step). A measurement in the gate, while not guaranteed to have originated from the target the gate pertains to, is a valid association candidate [25]. If there is more than one measurement in the gate, this leads to an association uncertainty and can be resolved by a measurement to data association technique described in next section. An example of this problem is depicted in Figure 5.2. In the figure there are four measurements collected by the radar (circles) and two predicted tracks (crosses). If we exclude gating, we have a total of 840 possible association hypotheses but by gating we decrease it to 62 hypotheses since we label some associations as invalid. For example we can not consider associating the measurement y_4 with any of the tracks. It may be either a false alarm or a new track.

There are a lot of techniques for implementing the gates. Owing to its simplicity rectangle gates [49] are common in the tracking process. Furthermore, more complex

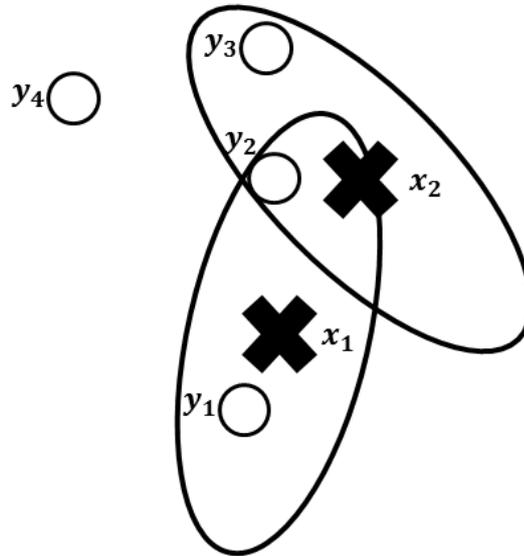


Figure 5.2. Gating Step Example: Measurements are represented by circles and the predicted tracks are represented by crosses.

gates can be used such as the one derived by Kalman equations for Gaussian systems. Formally the procedure is as follows: Assuming that the radar provides M possible target each having a measurement vector y_m . Each hypothesis has tracks x_t^i with covariance P_t^i . If the measurement y_m is inside the gating region size of G then;

$$(y_m - Cx_t^i)^T B^{-1} (y_m - Cx_t^i) \leq G \quad (5.1)$$

such that $B = R + CP_t^i C^T$. However, this operation involves matrix inversion which is an expensive operation compared to the rectangle gates.

5.3. A Data Association Algorithm: Global Nearest Neighbor Technique

After having only the valid measurements for the correspondent track, we need to have the best measurement (or depending on the technique, a logical combination of these measurement) for each existing predicted track. Several tracking algorithms have been proposed performing the data association ranging from the simpler nearest

neighbor filter to the complex multi hypothesis trackers such as Multi Hypotheses Tracking, Probabilistic Data Association and Joint Probabilistic Data Association. In this section we represent the technique that we use in our experiment, global nearest neighbor technique [10]. Generally for a data association algorithm, there are four essential steps to be included: hypothesis generation step, hypothesis evaluation step, track maintenance step and determining the best hypothesis step, and this process is the method how we examine global nearest neighbor technique.

- (i) Hypothesis Generation: A hypothesis, denoted by θ , is a consistent measurement to track association that can formally be defined as $\theta : \{1, 2, \dots, M_t\} \rightarrow \{0, 1, 2, \dots, N_{t-1}, b_t^1, \dots, b_t^n\}$ such that M_t is the number of measurements collected at time t and N_{t-1} is the number of targets that are detected for time $t-1$ with 0 represents false alarms and b_t^i represents new tracks. After each collection of measurements, all of the feasible hypotheses need to be generated by the algorithm in order to monitor the search space consisting of these hypotheses. For example consider Figure 5.2, then a feasible hypothesis can be stated as y_1 and y_2 are associated with x_1 and x_2 respectively, y_3 is a new track so associated with x_3 and y_4 is a false alarm. Another valid but less probable hypothesis is to label all measurements with new tracks. You may see all of the hypotheses that can be generated for the corresponding time interval of Figure 5.2 in Figures 5.3 and 5.4 that represents a matrix representation and a tree representation of the hypotheses respectively.
- (ii) Hypothesis Evaluation: As can be inferred from the name of the technique, we try to assign the nearest neighbor to the predicted track positions, $x_{t|t-1}$. But as in Figure 5.2 there may be some measurements that results in conflicts. Thus, purpose of the GNN association is to minimize the global association distance for each time interval. We define the global association distance for an association hypothesis θ as:

$$D_\theta^2 \triangleq \sum_{i=1}^n d(x_{t|t-1}^i, y_t^j) \quad (5.2)$$

0	0	0	0
1	0	0	0
3	0	0	0
0	1	0	0
1	1	0	0
3	1	0	0
0	2	0	0
1	2	0	0
3	2	0	0
0	4	0	0
1	4	0	0
3	4	0	0
0	0	2	0
1	0	2	0
3	0	2	0
0	1	2	0
1	1	2	0
3	1	2	0

0	2	2	0
1	2	2	0
3	2	2	0
0	4	2	0
1	4	2	0
3	4	2	0
0	0	5	0
1	0	5	0
3	0	5	0
0	1	5	0
1	1	5	0
3	1	5	0
0	2	5	0
1	2	5	0
3	2	5	0
0	4	5	0
1	4	5	0
3	4	5	0

0	0	0	6
1	0	0	6
3	0	0	6
0	1	0	6
1	1	0	6
3	1	0	6
0	2	0	6
1	2	0	6
3	2	0	6
0	4	0	6
1	4	0	6
3	4	0	6
0	0	2	6
1	0	2	6
3	0	2	6
0	1	2	6
1	1	2	6
3	1	2	6

0	2	2	6
1	2	2	6
3	2	2	6
0	4	2	6
1	4	2	6
3	4	2	6
0	0	5	6
1	0	5	6
3	0	5	6
0	1	5	6
1	1	5	6
3	1	5	6
0	2	5	6
1	2	5	6
3	2	5	6
0	4	5	6
1	4	5	6
3	4	5	6

Figure 5.3. Matrix Representation of Generated Hypotheses: Each row corresponds to an hypothesis. Red rows are inconsistent rows, i.e. one track is assigned by two measurements. Thus they must be deleted from the matrix.

such that according to θ association hypothesis y_t^j is associated with $x_{t|t-1}^i$. Now we derive $d(x_{t|t-1}^i, y_t^j)$ that is not simply an Euclidean distance but a measure that takes into account the uncertainties as follows. The total likelihood that y_t^j is generated by $x_{t|t-1}^i$ is

$$p(y_t^j | x_{t|t-1}^i) = \int p(y_t^j | x) p(x | x_{t|t-1}^i) dx \quad (5.3)$$

We can derive the posterior using Bayes Theorem by assuming $p(y_t^j)$ is uniform since there is no prior information about the collected measurements.

$$p(x_{t|t-1}^i | y_t^j) = \frac{p(y_t^j | x_{t|t-1}^i)}{\sum_{k=1}^{M_t} p(y_t^k | x_{t|t-1}^i)} \quad (5.4)$$

So we can define the $d(x_{t|t-1}^i, y_t^j) \triangleq -\log(p(x_{t|t-1}^i | y_t^j))$

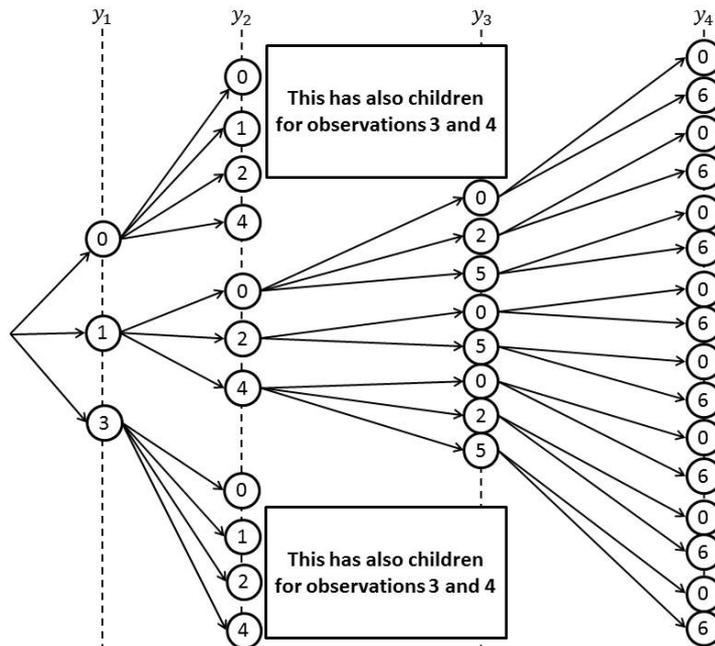


Figure 5.4. Tree Representation of Generated Hypotheses: From root to any leaf, each path corresponds to a hypothesis. In order to be clear, sub-trees generated by first and third children of y_2 are not depicted.

- (iii) Determining the Best Hypothesis: After determining the distance between the tracks and the hypothesis we need to find a solution to the optimization problem. However a straightforward implementation of an algorithm to solve this problem can be an exhaustive search in the association space by considering all possible associations but this is not a efficient solution for a real time system. Thus there are various optimization algorithms designed for minimization including Hungarian Algorithm [4] and JVC Algorithm [50].
- (iv) Track Maintenance: Track maintenance resolves track initialization and track deletion since the multitarget model involves target births and target deaths. From GNN perspective, track initialization and track deletion are carried out by some rules that are application dependent. To illustrate what is meant by a rule, track initialization is processed if four measurements for a target are collected in last six time intervals. Also we can exemplify the application dependence by a ballistic missile tracker in which rather than waiting it is better to label a measurement by a new track, so this four consecutive measurements rule may be

decreased to two consecutive measurements.

The GNN approach, which only considers the single most likely hypothesis for track update and new track initiation, only works well in the case of widely spaced targets, accurate measurements, and few false alarms in the track gates. For example, from results given in [49], even if the true target return is present, a single uniformly distributed false alarm in a three dimensional radar measurement space (typically range and two angles) reduces the probability of correct association to about 0.85. Thus, in about one out of six track update attempts a false alarm will be chosen rather than the correct target return. For the more usual case of multiple closely spaced targets and where missed true target detections occur, the probability of false track update is much worse. Experience indicates that often a single false update may lead to track loss and two consecutive false updates will usually lead to track loss [51].

Until now we described the single target tracking techniques and extend them by including data association methods. Dating back from the first radar technologies, nearly all the radars developed get use of the combination of these two techniques. In the next chapter we begin to view the problem from a different and new emerging perspective based on random finite sets.

6. MULTITARGET TRACKING VIA PROPAGATING FIRST ORDER STATISTICAL MOMENT:PHD FILTER

In Chapter 4 we represented how Kalman filter performs recursive and online estimation as a close form solution of the optimal Bayesian filter of Section 3.4. Rather than propagating the full distribution, what Kalman filter does is to propagate the first and second moments² of the filtering density, that is the mean $x_{t|t}$ and $P_{t|t}$ respectively. If it is assumed that SNR ratio is high enough then the higher order moment-tensors can be neglected, thus results in $x_{t|t}$ and $P_{t|t}$ are sufficient statistics and that is the case in Kalman filter. Now assume that SNR is so high that only first statistical moment, that is $x_{t|t}$ particularly, can represent the filtering density. Even though, it may seem inconvenient for the real world applications at first glance, an example of this kind of filter is $\alpha - \beta$ filters (also called constant gain Kalman filter) represented in the introduction of Chapter 4. The criteria that we defined is the increase in the Signal to Noise Ratio that results in covariance matrices (higher order statistical moments in general) to dwarf and if the covariance is small then unimodal distributions are all look similar. In other words the likelihood is so highly accumulated around some $x_{t|t}$ that any distribution derived is similar to each other. Hence under some conditions dropping higher order statistical moments, thereby approximating the filtering density is convenient and results in less computation from an implementation point of view. Probability Hypothesis Density, abbreviated as PHD, is an analogy of the single target case for the multitarget case. It is also first order moment of the multitarget filtering density, but it is not a fixed number like $x_{t|t}$ but a distribution over the entire target state space. The concept relies on Random Finite Sets (RFS) and Finite Set Statistics (FISST) as represented in the Mahler's original work in [13]. We introduce the PHD filter from point of view in [13], however, the RFS and FISST approaches are mathematically based on non-homogenous poisson point processes [11]. Singh et.al. addressed the problem and the filter by using the pure point process theory in [52]. We represent the radar

²n-th moment of a real valued continuous function $f(x)$ of a real variable about a value c is defined as: $\mu_n = \int_{-\infty}^{\infty} (x - c)^n f(x) \partial x$

tracking model via RFS' in Section 6.1, then we introduce finite set statistics that includes the methodology for deriving the PHD in Section 6.2. In Section 6.3, we define what is and is not the probability hypothesis density and finally derive the prediction and update recursions in Section 6.4.

6.1. Modeling Tracking Using Random Finite Sets

As can be inferred from the term, a Random Finite Set (aka Point Process) is a generalization of a random variable to sets. To be more precise, the size of a random variable is fixed, but the size of a random finite set can be varying. This property makes the random finite sets suitable for transition and measurement models of tracking. To illustrate the variation of the sizes of the set, you may see the Figure 6.1. While at time t there are two targets in the scene, just three of many possible scenarios for time $t + 1$ are depicted in the figure. In the upper scenario, there is no either a target birth or a target death, thus set size is still same with different state values of x_1 and x_2 . In the middle figure, you may see x_1 is dead and in the lower x_3 is a new target appearing in the scene. Therefore by using a function over the entire state space along with the transition and measurement models, we aim to find out the target intensity function of $t + 1$.

PHD filter models the uncertainty in the system using a multitarget set Ξ

$$\Xi = \{x_t^1, x_t^1, \dots, x_t^{N_t}\} \quad (6.1)$$

where N_t is the number of targets and the elements of Ξ are random instantiations from the single target state space. Then we can define the multitarget state space as all the finite subsets of single target state space. Similarly we can define the multitarget measurement space by combining all single target measurement spaces. Then any multitarget measurement must be a subset of this combined space. Next, we represent the probability distribution of a RFS, then the transition and the measurement models that are used in RFS.

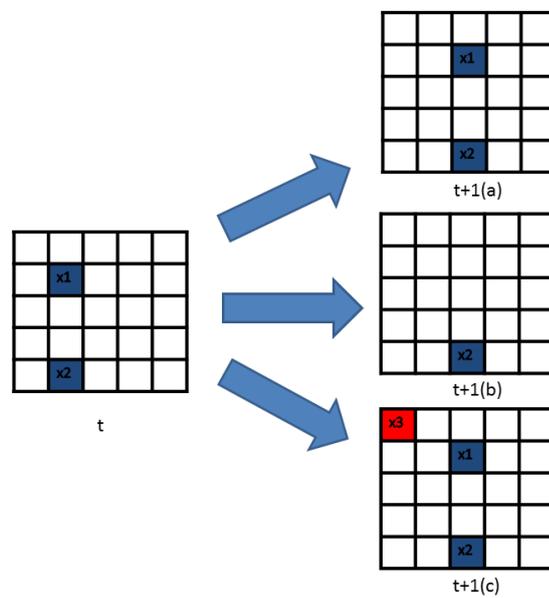


Figure 6.1. Illustration of Possible Scenarios of an RFS: The figure illustrates possible scenarios of an RFS for the next time step. As seen, in subfigures (b) and (c), the set size changes. An intensity function must be able to capture this information by using the state transition model. x_1 , x_2 and x_3 are the track identities and the state space is 5×5 two dimensional space.

6.1.1. Probability Distribution of a RFS

Due to the nature of the random variable, an RFS also has a probability distribution that also take the number of the events (targets/measurements in this case) in account. It can be stated as follows:

$$f_{\Xi}(X) = \begin{cases} f_{\Xi}(\emptyset) & \text{if } X = \emptyset \\ f_{\Xi}(\{x_1\}) & \text{if } X = \{x_1\} \\ f_{\Xi}(\{x_1, x_2\}) & \text{if } X = \{x_1, x_2\}, |X| = 2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{cases}$$

Also by comparing a set and a vector in which the order is important, we can state a term of the distribution explicitly as $p(\Xi) = n!p(X_t)$ where Ξ represent the random set and X_t is the corresponding vector. Furthermore if $\int f_{\Xi}(X)\delta X = 1$, then it is a multitarget density function where we define the set integral in region S as follows:

$$\int_S f_{\Xi}(X)\delta X = f_{\Xi}(\emptyset) + \sum_{n \geq 1}^{\infty} \frac{1}{n!} \int_S f_{\Xi}(\{x_1, \dots, x_n\}) dx_1 \dots dx_n \quad (6.2)$$

Even though summation of the sets are undefined, thus we need to define what is meant by $f_{\Xi}(\{x_1, \dots, x_n\})$. Since it is the unordered structure of a vector we can define $f_{\Xi}(\{x_1, \dots, x_n\}) = n!f_X(x_1, \dots, x_n)$ To illustrate simply, $f_{\Xi}(\{x_1, x_2\})=2! \times f_X(x_1, x_2)$ where f_X is the corresponding multivariate probability distribution. It is clear since in the set the order is not important, thus both (x_1, x_2) and (x_2, x_1) vector serves for the $\{x_1, x_2\}$ set.

Lastly in this section we define the cardinality distribution of a RFS that is the distribution for the number of events (e.g. number of targets). In other words the

probability that there are n elements in RFS Ξ is

$$p_{\Xi}(n) = \int_{|X|=n} f_{\Xi}(X) \delta X \quad (6.3)$$

$$= \frac{1}{n!} \int f_{\Xi}(\{x_1, \dots, x_n\}) dx_1 \dots dx_n \quad (6.4)$$

6.1.2. Transition Model

If we have an RFS at time $t - 1$ for target states as follows:

$$\Xi_{t-1} = \{x_{t-1}^1, x_{t-1}^2, \dots, x_{t-1}^{N_{t-1}}\} \quad (6.5)$$

then at time t we have the following RFS.

$$\Xi_t = \underbrace{T(x_{t-1}^1) \cup \dots \cup T(x_{t-1}^{N_{t-1}})}_{\text{Survive or Die}} \cup \underbrace{\varphi(x_{t-1}^1) \cup \dots \cup \varphi(x_{t-1}^{N_{t-1}})}_{\text{Spawn}} \cup \underbrace{\varphi}_{\text{New born}} \quad (6.6)$$

where

$$T(x_{t-1}^i) = \begin{cases} \emptyset & \text{if die with } (1-p_S) \\ \{f_t(x_{t-1}^i, w_{t-1})\} & \text{if survives with } p_S \end{cases}$$

$\varphi(x_{t-1}^i)$ is the RFS defining the set of spawned target that is modeled as Poisson. Lastly φ is the RFS of the new born targets at time t that is also modeled as Poisson. Also it is assumed that $T(x_{t-1}^i)$, $\varphi(x_{t-1}^i)$ and φ are independent.

6.1.3. Measurement Model

If we have a RFS at time t for target states as follows:

$$\Xi_t = \{x_t^1, x_t^1, \dots, x_t^{N_t}\} \quad (6.7)$$

then at time t we have the following RFS with respect to the measurements.

$$\Sigma_t = \underbrace{\Upsilon(x_t^1) \cup \dots \cup \Upsilon(x_t^{N_t})}_{\text{Detected or Missed Detection}} \cup \underbrace{\Theta}_{\text{False Alarm and Clutter}} \quad (6.8)$$

where

$$\Upsilon(x_t^i) = \begin{cases} \emptyset & \text{if not detected with } (1-p_D) \\ \{h_t(x_t^i, w_t)\} & \text{if detected with } p_D \end{cases}$$

Θ is the RFS of the false alarms and clutters at time t that is modeled as Poisson distributed.

6.2. Finite Set Statistics(FISST)

Finite Set Statistics (FISST) is a methodology devised for dealing with the multitarget tracking without delving into the abstractions and complexities of Point Process theory [11] that also models stochastic multiobject problems. In other words, finite set statistics is a stripped-down version of point process theory aiming to avoid all possible abstractions. For example missed detections and dying targets could be represented as thinning and marking of a nonhomogeneous poisson point process, but from a FISST point of view these are defined more engineering friendly without losing the essence of the theory. Mahler established the approach while defining the PHD filter in [13] and then by relating the topic with statistical behavior of random variables (he presented a tutorial for this also in [53]), he made up a tutorial that we believe the briefest possible one in [12]. Furthermore a detailed information on the approach can be found in [10].

In order derive the appropriate PHD filter, main methodology that can be tracked is as follows:

- (i) Construct RFS motion and measurement model
- (ii) Using multitarget calculus, convert these models into multitarget densities and

likelihood functions

- (iii) From these construct the optimal approach: a multitarget Bayes filter
- (iv) Convert the Bayes filter into probability generating functional form
- (v) Use simplifying approximations and multitarget calculus to derive PHD filters for the application.

Apart from the motion and measurement models that we discussed in the previous section, the upper road map makes up the topics that we represent in this section.

6.2.1. Construct the Multitarget Densities and Likelihood Functions

In order to derive the multitarget density we have two steps, converting the model into a belief mass function, then converting the belief mass function to densities. Belief mass function, that is analogous to cumulative distribution functions of random variables, $\beta_{T(x_{t-1}^i)}(S)$, where S is a closed set variable, of $T(x_{t-1}^i)$ is defined as follows:

$$\beta_{T(x_{t-1}^i)}(S) = p(T(x_{t-1}^i) \subseteq S) \quad (6.9)$$

$$= p(T(x_{t-1}^i) = \emptyset) + p(T(x_{t-1}^i) \neq \emptyset, T(x_{t-1}^i) \subseteq S) \quad (6.10)$$

$$= 1 - p_S(x_{t-1}^i) + p_S(x_{t-1}^i) \int_S f_t(x_{t-1}, w_{t-1}) dx_t \quad (6.11)$$

Also for birth and spawning are Poisson as stated in the previous section. Thus the belief mass function of a Poisson distributed set variable, say B , with expected number of λ_B is:

$$\beta_\varphi(S) = \exp(-\lambda_B + \lambda_B \int_S s(x) dx) \quad (6.12)$$

where $s(x)$ is the spatial distribution of new appearances. Lastly due to the independence of the terms, belief mass function of the random set is:

$$\beta_{\Xi_t}(S|X_t) = \beta_{T(x_{t-1}^1)}(S) \dots \beta_{T(x_t^{N_t})}(S) \cdot \beta_{\varphi(x_{t-1}^1)}(S) \dots \beta_{\varphi(x_t^{N_t})}(S) \cdot \beta_\varphi(S) \quad (6.13)$$

Now the belief mass functions are to be converted into multitarget density and multitarget likelihood function as in Equation 6.14 and equation 6.15 respectively. But notice that the derivatives and the integrals are set derivatives and set integrals (see chapter 11 of [10] for differential rules of multiobject derivatives).

$$p(X_t|X_{t-1}) = \left[\frac{\delta}{\delta X_t} \beta_{\Xi_{t|t-1}}(S|X_{t-1}) \right]_{S=\emptyset} \quad (6.14)$$

$$p(Y_t|X_t) = \left[\frac{\delta}{\delta Y_t} \beta_{\Sigma_t}(T|X_t) \right]_{T=\emptyset} \quad (6.15)$$

Following on a simple scenario we clarify the methodology for deriving multitarget density and likelihood function. Assume that there is no target in the scene and we want to derive the multitarget likelihood function. So $X_t = \emptyset$ but there may be arbitrary number of measurements only consisting of false alarms. Thus the measurement model is $\Sigma_t = \Theta$. Firstly we must construct the belief mass function of the model.

$$\beta_{\Sigma_t}(T|X_t = \emptyset) = p(\Theta \subseteq T|\emptyset) \quad (6.16)$$

$$= \sum_{m=0}^{\infty} p(\Theta \subseteq T, |\Theta| = m|\emptyset) \quad (6.17)$$

$$= \sum_{m=0}^{\infty} p(|\Theta| = m|\emptyset) p(\Theta \subseteq T|\emptyset, |\Theta| = m) \quad (6.18)$$

$$= e^{-\lambda_{FA}} \sum_{m=0}^{\infty} \frac{\lambda_{FA}^m}{m!} p(\{\theta_1, \dots, \theta_{N_{FA}}\} \in T) \quad (6.19)$$

$$= e^{-\lambda_{FA}} \sum_{m=0}^{\infty} \frac{\lambda_{FA}^m}{m!} p(\theta \in T)^m \quad (6.20)$$

$$= e^{-\lambda_{FA}} \sum_{m=0}^{\infty} \frac{\lambda_{FA}^m}{m!} p_c(T)^m \quad (6.21)$$

$$= e^{-\lambda_{FA}} e^{\lambda_{FA} p_c(T)} \quad (6.22)$$

$$= e^{\lambda_{FA} p_c(T) - \lambda_{FA}} \quad (6.23)$$

Second we need to derive the multitarget likelihood function:

$$p(Y_t|X_t) = \frac{\delta\beta_{\Sigma_t}}{\delta Y_t}(T|X_t = \emptyset) \quad (6.24)$$

$$= \frac{\delta}{\delta Y_t} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.25)$$

$$= \frac{\delta^m}{\delta y_m \dots \delta y_1} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.26)$$

$$= \frac{\delta^{m-1}}{\delta y_m \dots \delta y_2} \frac{\delta}{\delta y_1} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.27)$$

$$= \frac{\delta^{m-1}}{\delta y_m \dots \delta y_2} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \frac{\delta}{\delta y_1} (\lambda_{FAPc}(T) - \lambda_{FA}) \quad (6.28)$$

$$= \lambda_{FAC}(y_1) \frac{\delta^{m-2}}{\delta y_m \dots \delta y_3} \frac{\delta}{\delta y_2} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.29)$$

$$= \lambda_{FA}^2 c(y_1) c(y_2) \frac{\delta^{m-2}}{\delta y_m \dots \delta y_3} e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.30)$$

$$= \dots = \lambda_{FA}^m \cdot c(y_1) \dots c(y_m) \cdot e^{\lambda_{FAPc}(T) - \lambda_{FA}} \quad (6.31)$$

Throughout the derivation of the multitarget likelihood function, we get use of the multitarget calculus, specifically derivative rules. In this case chain rule, linear rule and sum rule are used. One may see [10] for the list of rules that can be incorporated in the derivation of such densities. Finally, as in equation 6.15 we set $T = \emptyset$ and get the multitarget likelihood function as follows:

$$p(Y_t|X_t) = \frac{\delta\beta_{\Sigma_t}}{\delta Y_t}(T = \emptyset|X_t = \emptyset) = e^{-\lambda_{FA}} \lambda_{FA}^m c(y_1) \dots c(y_m) \quad (6.32)$$

6.2.2. Construct the Multitarget Optimal Bayes Filter

The equations are the generalized version of the optimal Bayes filter represented in Section 3.4.

(i) Prediction Step:

$$p(X_t|Y_{1:t-1}) = \int p(X_t|X_{t-1})p(X_{t-1}|Y_{1:t-1})dX_{t-1} \quad (6.33)$$

(ii) Update Step:

$$p(X_t|Y_{1:t}) = \frac{1}{Z_t} p(Y_t|X_t)p(X_t|Y_{1:t-1}) \quad (6.34)$$

where the normalization constant, $Z_t = \int p(Y_t|X_t)p(X_t|Y_{1:t-1})dX_t$

By applying the multitarget density and the likelihood function into the equations, we construct the optimal filter for multitarget tracking. Although we derived all the terms required for the optimal multitarget Bayes filter, the integrals defined are set integrals and they are intractable. Thus, we get use of a transformation that is called probability generating functionals.

6.2.3. Convert the Bayes Filter into Probability Generating Functional Form

In this section we derive the concept of probability generating functional in a gradual manner. Initially to define what a probability generating function [54], it is a power series representation of the probability distribution of a discrete random variable and it is formally defined as:

$$G_x(s) = p(x = 0) + p(x = 1)s^1 + p(x = 2)s^2 + \dots = \sum_{i=0}^{\infty} p(x = i)s^i \quad (6.35)$$

where x is the random variable, $p(x)$ is the probability distribution of x . As seen, s is the free variable of the function $G_x(s)$ and for all values of s , the function converges. To illustrate, $G_x(0) = 0$ and $G_x(1) = 1$. Probability generating function is one of the transformation method that is used in order to simplify the operations such as Fast Fourier Transform in physics. An example of what probability generating functions catalyze is to enable us to calculate the expectation (or first statistical moment) easily

by using the first derivative of the function at $s = 1$ since

$$G_x(s) = p(x = 0) + p(x = 1)s^1 + p(x = 2)s^2 + \dots \quad (6.36)$$

$$G'_x(s) = 1 \cdot p(x = 1)s^0 + 2 \cdot p(x = 2)s^1 + \dots \quad (6.37)$$

and equation 6.37 is the definition of the first statistical moment of a discrete variable.

Another issue that must be clarified as a preliminary is that briefly a functional is simply the function of the functions. A functional is a function from a vector space into its underlying scalar field, or a set of functions of the real numbers. In other words, it is a function that takes a vector as its input argument, and returns a scalar. [55]. Indeed in the multitarget case, since the process does not count on the random variables but instead the vectors and the sets, probability generating functions are not sufficient to describe the process. Thus by combining these two notions, probability generating functionals [11] are able to represent the characteristics of the process.

Let $f_{t|t}(X_t|Y_{1:t})$ be the multitarget filtering density at time t . Then the probability functional is defined by

$$G_{t|t}[h|Y_{1:t}] = \int f_{t|t}(X_t|Y_{1:t}) h^X \delta X \quad (6.38)$$

such that the power functional of the function $0 \leq h(x) \leq 1$ is defined by

$$h^X = \begin{cases} 1 & \text{if } x = \emptyset \\ \prod_{x \in X} h(x) & \text{otherwise} \end{cases}$$

When we compare the Equation 6.35 and Equation 6.38, the similarity stands out. Furthermore, the convenience of estimating the expectation of a random variable holds with respect to probability generating functionals as well. Since the expectation of a multitarget density function is the PHD, we can find PHD as follows by using

probability generating functional.

$$D_{t|t}(x|Y_{1:t}) = \left[\frac{\delta G_{t|t}}{\delta h} [h] \right]_{h=1} \quad (6.39)$$

As a consequence, for the PHD filter rather than propagating the full joint posterior, we propagate the first moment of the joint posterior by firstly converting the recursion into a probability generating functional form and then find out the PHD by setting h to be 1.

6.3. The PHD

Before deriving the recursion of the PHD filter, in this section we illustrate the notion of Probability Hypothesis Density by giving examples and explaining its properties. It is shown that probability hypothesis density is the first order statistical moment of a point process, a.k.a. random finite set, in [13, 56, 57] and we give the essential information of PHD by associating it with the conventional expected value definition. The first-order moment of filtered multitarget density, $f_{t|t}(X_t|Y_{1:t})$, is conventionally stated as:

$$\hat{X}_{t|t} = \int X \cdot f_{t|t}(X_t|Y_{1:t}) \delta X \quad (6.40)$$

However, since the basic operations such as addition is undefined on random finite sets are not defined, we adopt the following transformation for X .

$$\delta_{X_t} \triangleq \begin{cases} 0 & \text{if } X_t = \emptyset \\ \sum_{b \in X_t} \delta_b(x_t) & \text{otherwise} \end{cases}$$

where $\delta_b(x)$ is the Dirac Delta function concentrated at b . Hence, after the transformation, the first-order statistical moment, that is Probability Hypothesis density (or intensity function in point process terminology), $D_{t|t}$ has the following

form.

$$D_{t|t}(x_t|Y_{1:t}) = \int \delta_{X_t}(x_t) \cdot f_{t|t}(X_t|Y_{1:t}) \delta X \quad (6.41)$$

Since the set integral of Equation 6.41 can be expanded by using Equation 6.2

$$D_{t|t}(x_t|Y_{1:t}) = 0 + \int \delta_{x_t^1}(x_t) f_{t|t}(x_t^1|Y_{1:t}) dx_t^1 + \quad (6.42)$$

$$\int \int [\delta_{x_t^1}(x_t) + \delta_{x_t^2}(x_t)] f_{t|t}(x_t^1, x_t^2|Y_{1:t}) dx_t^1 dx_t^2 + \dots + \quad (6.43)$$

$$\underbrace{\int \dots \int}_{N_t \text{ times}} [\delta_{x_t^1}(x_t) + \dots + \delta_{x_t^{N_t}}(x_t)] f_{t|t}(x_t^1 \dots x_t^{N_t}|Y_{1:t}) dx_t^1 \dots dx_t^{N_t} \quad (6.44)$$

For example, assume that we know that there are two targets, x_t^1, x_t^2 , in the environment, then the form that the density have is

$$D_{t|t}(x_t|Y_{1:t}) = \int \int [\delta_{x_t^1}(x_t) + \delta_{x_t^2}(x_t)] f_{t|t}(x_t^1, x_t^2|Y_{1:t}) dx_t^1 dx_t^2 \quad (6.45)$$

$$= \int \int \delta_{x_t^1}(x_t) f_{t|t}(x_t^1, x_t^2|Y_{1:t}) dx_t^1 dx_t^2 + \int \int \delta_{x_t^2}(x_t) f_{t|t}(x_t^1, x_t^2|Y_{1:t}) dx_t^1 dx_t^2 \quad (6.46)$$

$$= \int f_{t|t}(x_t^1 = x_t, x_t^2|Y_{1:t}) dx_t^2 + \int f_{t|t}(x_t^1, x_t^2 = x_t|Y_{1:t}) dx_t^1 \quad (6.47)$$

$$= f_{t|t}(x_t^1 = x_t|Y_{1:t}) + f_{t|t}(x_t^2 = x_t|Y_{1:t}) \quad (6.48)$$

In general if there are N_t targets in the environment, then

$$D_{t|t}(x_t|Y_{1:t}) = \sum_{n=1}^{N_t} f_{t|t}(x_t^n = x_t) \quad (6.49)$$

It can easily inferred from the Equation 6.49

$$\hat{N}_t = \int D_{t|t}(x_t|Y_{1:t}) x_t \quad (6.50)$$

such that \hat{N}_t is the expected number of targets for time t . In other words, probability hypothesis density is not a probability distribution that sums up to 1 but a function whose peaks are the possible states of targets. To concretize the PHD, we present the illustration in Figure 6.2. In the figure the PHD function of a one dimensional state space is represented by blue solid line. In order to extract the target states from the function, we must first determine the number of objects which is the integration of the PHD itself. For the particular case the integration produces 4, thereby using the counting property of the PHD, we project the four peaks of the function to the state space that are represented by the red dashed lines in the figure. Indeed we deal with the state estimation problem in detail in the next section that explains the filter equations.

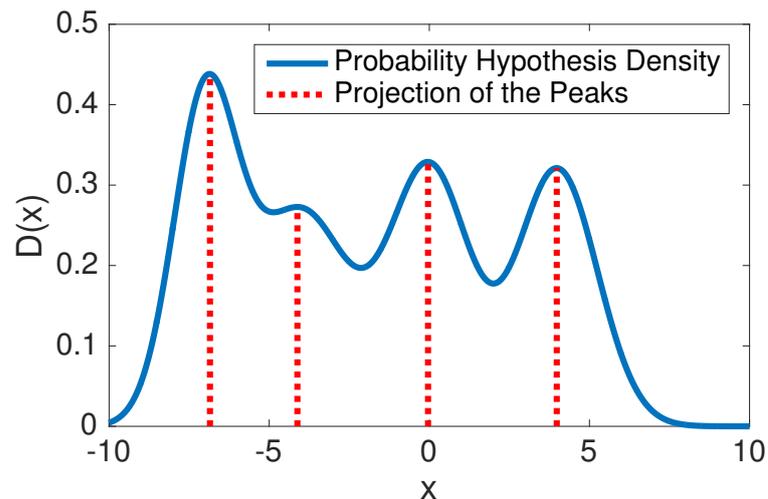


Figure 6.2. An Illustration of the Probability Hypothesis Density on One Dimensional State Space.

6.4. PHD Filter

We adopt the same methodology that is used in previous chapters when we described the behavior of a filter. To be more precise, PHD filter consists of four steps as usual: Initialization, prediction, update and finally state estimation.

6.4.1. Initialization

In order to initialize the filter, it is enough to determine a priori PHD that encapsulates the priori knowledge about the multitarget environment. Specifically, this knowledge consists of both the number of targets and their distributions on the state space. That is, assuming that \hat{N}_0 is the expected number of targets apriori and $s_0(x)$ is the probability density with maxima corresponding the target states.

$$D_{0|0}(x|Y_0) = D_{0|0}(x) = \hat{N}_0 s_0(x) \quad (6.51)$$

Similar to the general idea represented in Equation 6.51, the sum of Gaussians can be a initialization method (we also get use of the sum of Gaussians in Figure 6.2), since each Gaussian has a density summing up to 1, thus the integration of these Gaussians will be the number of Gaussians in the summation. The method is formulated as follows such that μ_0^i and $P_{0|0}^i$ corresponds to the means and covariances of the particular addends respectively.

$$D_{0|0}(x) = \mathcal{N}(x, \mu_0^1, P_{0|0}^1) + \dots + \mathcal{N}(x, \mu_0^{\hat{N}_0}, P_{0|0}^{\hat{N}_0}) \quad (6.52)$$

6.4.2. Prediction

We briefly present the methodology employed to derive the prediction equation of the PHD filter and describe the prediction recursion equation in this section. We assume that for the prediction step we have the updated PHD of time $t - 1$, $D(x_{t-1}|Y_{1:t-1})$ and want to construct the predicted PHD of time t , $D(x_t|Y_{1:t-1})$. First by denoting the resulting prediction density by $p(X_t|Y_{1:t-1})$, we can express it in

terms of its probability generating functional by using Equation 6.38 as follows:

$$G_{t|t-1}[h] = \int h^{X_t} p(X_t|Y_{1:t-1}) \delta X_t \quad (6.53)$$

$$= \int h^{X_t} \left(\int p(X_t|X_{t-1}) p(X_{t-1}|Y_{1:t-1}) dX_{t-1} \right) \delta X_t \quad (6.54)$$

$$= \int \left(\int h^{X_t} p(X_t|X_{t-1}) dX_t \right) p(X_{t-1}|Y_{1:t-1}) \delta X_{t-1} \quad (6.55)$$

$$= \int G_{t+1|t}[h|X_{t-1}] p(X_{t-1}|Y_{1:t-1}) \delta X_{t-1} \quad (6.56)$$

So for the prediction step, we can recursively proceed in terms of probability generating functionals and by using Equation 6.39, we can find the first statistical moment, the PHD, of the prediction density as follows:

$$D(x_t|Y_{1:t-1}) = \left[\frac{\delta G_{t|t-1}[h]}{\delta x_t} \right]_{h=1} \quad (6.57)$$

Consequently we have the following prediction equation:

$$D(x_t|Y_{1:t-1}) = \underbrace{b(x_t)}_{\text{birth targets}} + \int \left(\underbrace{p_S \cdot p(x_t|x_{t-1})}_{\text{surviving targets}} + \underbrace{b(x_t|x_{t-1})}_{\text{spawning targets}} \right) \cdot D(x_{t-1}|Y_{1:t-1}) dx_{t-1} \quad (6.58)$$

where

- $b(x_t)$ is the PHD of the target birth. Since target birth is assumed to be a Poisson process

$$b(x_t) = \lambda_B s_B(x_t) \quad (6.59)$$

such that $s_B(x_t)$ is the spatial distribution of the target births and λ_B is the expected value of the Poisson process. Both of the quantities may be adjusted to vary in time. On the other hand fixed values can be also considered according to the requirements of the application.

- $b(x_t|x_{t-1})$ is the PHD of the target spawning. Since the spawning process is

analogous to the birth process in the structure, by adopting the variables pertaining to the spawning, λ_S and $s_S(x_t)$ respectively, the expressions referred are valid also for spawning targets.

- p_S is the survival probability of a target. In the equation we referred p_S as fixed. However, it may vary both with respect to time and/or location.
- $p(x_t|x_{t-1})$ is the transition model of the target.
- $D(x_{t-1}|Y_{1:t-1})$ is the updated PHD of time $t - 1$.

6.4.3. Update

Before constructing the update equation of time t , $D(x_t|Y_{1:t})$, we assume that we have the predicted PHD of time t , $D(x_t|Y_{1:t-1})$. Now we consider first the fraction part of the Bayes update of Equation 6.34 and express it in terms of probability generating functionals by using Equation 6.38.

$$F[g, h] = \int g^{Y_t} p(Y_t|X_t) \delta Y_t \int h^{X_t} p(X_t|Y_{1:t-1}) \delta X_t \quad (6.60)$$

$$= \int h^{X_t} \int g^{Y_t} p(Y_t|X_t) p(X_t|Y_{1:t-1}) \delta X_t \delta Y_t \quad (6.61)$$

$$= \int h^{X_t} G[g|X_t] p(X_t|Y_{1:t-1}) \delta X_t \quad (6.62)$$

$$\text{where } G[g|X_t] = \int g^{Y_t} p(Y_t|X_t) \quad (6.63)$$

the PHD can be expressed as:

$$D(x_t|Y_{1:t}) = \frac{1}{Z_t} \frac{\delta y_t^{M_t+1} F}{\delta y_t^{M_t} \dots \delta y_t^1 \delta x} [0, 1] \quad (6.64)$$

where the denominator of Equation 6.34 and also Equation 6.64 is

$$Z_t = \frac{\delta^{M_t} F}{\delta y_t^{M_t} \dots \delta y_t^1} [0, 1] \quad (6.65)$$

Consequently by using the methodology expressed above the following update PHD equation is derived.

$$D(x_t|Y_{1:t}) \approx \underbrace{[1 - p_D]D(x_t|Y_{1:t-1})}_{\text{No Detection}(\Sigma = \emptyset)} + \underbrace{\sum_{i=1}^{M_t} \frac{p_D p(y_t^i|x_t) D(x_t|Y_{1:t-1})}{fa(y_t^i) + \int p_D p(y_t^i|x_t) D(x_t|Y_{1:t-1}) dx_t}}_{\text{Detection Cases}} \quad (6.66)$$

where

- $fa(y_t^i)$ is the likelihood of y_t^i with respect to the probability distribution of false alarm.
- p_D is the detection probability of the radar for a target. In the Equation we determined it as fixed, however it may vary over time and/or location.
- $p(y_t^i|x_t)$ is the measurement likelihood of y_t^i .
- $D(x_t|Y_{1:t-1})$ is the predicted PHD of time t .

Also you may notice for Equation 6.66 that unlike the prediction step it is an approximation. We discuss this condition in Section 6.5 while evaluating characteristics of the PHD filter.

6.4.4. State Estimation

As depicted in Figure 6.2, for a specific time t , initially we evaluate the integral of the PHD function and find out the expected number of targets, \hat{N}_t ; then it is enough to find \hat{N}_t maxima points of the function. However in the implementation the value of \hat{N}_t may be unstable and sensitive to the false alarms and missed detections. Thus, if the target number of the application is known not to be very dynamic, then one can employ averaging the target number for a time window, say \bar{T} , as formulated below.

$$\bar{N}_t = \frac{\sum_{i=1}^{\bar{T}} \hat{N}_{t-i+1}}{\bar{T}} \quad (6.67)$$

6.5. Evaluation of the PHD Filter

For an approximate first order moment filter these three properties can be regarded as good properties.

- (i) Information loss from distributions to the moment should be minimized.
- (ii) The prediction step should be lossless, that is the predicted distribution $p_{t|t-1}(X)$ should result in the moment $D_{t|t-1}(x)$.
- (iii) The update step should be lossless, that is the updated distribution $p_{t|t}(X)$ should result in the moment $D_{t|t}(x)$.

In [13], it is shown that PHD filter satisfies the first two good properties. However this is not the case for the third one. As stated in update part of the previous section it is an approximation since in order to go through the derivation, $p(X_t|Y_{1:t-1})$ must be assumed to be Poisson. However, if sensor covariances and SNR is high enough such that it causes false alarm rates not to be large, then the measurements will accumulate around the target states and moment matching operation of the equation will perform well.

Another shortcoming of the filter appears when the targets are closely located. The scene is depicted in Figure 6.3. In the figure, the upper left one is the example depicted in Figure 6.2 and for three consecutive rows we adjusted the mean of the Gaussian densities such that this operation results in accumulation of the targets. Then gradually it is observed that the peaks of the filter disappears and even though we know the expected number of targets (that is also not so stable as discussed in previous section), that is four specifically, we are not able to extract the target states from the PHD function.

Another disadvantage of the filter is the instability of \hat{N} . This problem and a possible solution is issued in the state estimation part of the previous section, so we do not require to recite it again. [58] represents this instability and also stimulated Mahler to develop cardinalized PHD filter [14]. Along with the PHD, CPHD filter propagates

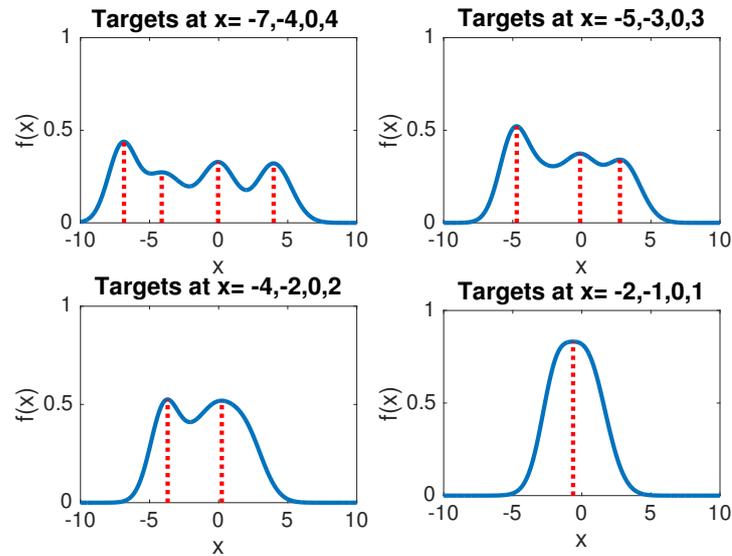


Figure 6.3. Effect of the Closer Targets on PHD: Blue solid lines refer to the corresponding PHD function for each scenario and red dashed lines are the projection of the peaks of the PHD function onto the one dimensional state space.

the cardinality distribution (see Equation 6.3) of the filtering density in time, so it resembles to Kalman filters that propagates the second order statistical moment as well but cardinality distribution is not the exact covariance density. The covariance density function is defined in [59] but the filter equations can not be extracted due to the intractability of the function.

On the other hand, omitting data association significantly makes the PHD filter attractive. Moreover the computational demand is $O(N_t M_t)$ for a time interval and that makes the filter computationally implementable. It also admits the standard multitarget measurement and transition model dynamics including target birth and death, spawning, missed detection and false alarm.

In the next chapter we consider the implementation strategies of the PHD filter that we represented from a theoretical framework in this chapter.

7. IMPLEMENTATION STRATEGIES OF THE PHD FILTER

As represented in Equations 6.58 and 6.66, PHD filter recursion involves integrals that are hard to deal with. However, there has been devised an approximation strategy and a closed form solution for these integrals. We cover these two implementation methods in the chronological order that they were developed. The first method is to get use of particle filter (See Section 4.4) to approximate the integrals and the second one is based on Gaussian assumptions and called Gaussian Mixture PHD.

7.1. Particle Filter Based Approaches

Since the equations have intractable integrals, immediately one may resort to Monte Carlo methods that allow to approximate these integrals by means of different sampling strategies. For this specific case, it is the Sequential Monte Carlo method that can be applied recursively to the equations in an online manner. As a result three independent algorithms were proposed in 2003. Mahler proposed an algorithm that does not include target birth and death in [60]. The algorithm in [61] does not include false alarm property of the model and was not a general algorithm but considered a special case for ground target tracking. The algorithm that includes all aspects of the transition and measurement models was first developed by Vo in [15] and that is further improved in [16]. In our thesis we consider Vo's improved algorithm and like the other subjects in our thesis, we adopt a simple to harder approach. Thus next two sections provide the basic algorithm and the improved version respectively. It must also need to be mentioned that there are other implementations of the filter based on particle filter such as the extension of auxiliary particle filter to the PHD filter [62, 63].

7.1.1. SMCPHD Filter

There is an important characteristic of this algorithm that also motivated us choosing it. Firstly, the particles are in an adaptive manner such that the algorithm prevents the system to be inefficient. Note that the particle number is the bottleneck of the particle filter methods with respect to performance. Furthermore it is an extension of the standard particle filter algorithm that is discussed in Section 4.4. As a result when we omit the target birth and death along with assuming only one target without clutter then it reduces to the standard particle filter algorithm. We handle the algorithm in the prediction and update steps first, finally we represent the pseudocode.

7.1.1.1. Initialization. To initialize the filter, we need to have a particle approximation. Thus we can approximate the prior intensity (PHD) function by means of importance sampling.

7.1.1.2. Prediction. Assuming at $t-1$ we have the set of particles and their associated weights, $\{x_{t-1|t-1}^{(i)}, w_{t-1|t-1}^{(i)}\}_{i=1}^{L_{t-1}}$, that approximates $D(x_{t-1}|Y_{1:t-1})$, we want to constitute the predicted PHD approximation. By substituting the particles into the prediction equation of PHD filter (Equation 6.58) we have the following;

$$D(x_t|Y_{1:t-1}) = b(x_t) + \int [p_S \cdot p(x_t|x_{t-1}) + b(x_t|x_{t-1})] \cdot D(x_{t-1}|Y_{1:t-1}) dx_{t-1} \quad (7.1)$$

$$\approx b(x_t) + \sum_{i=1}^{L_{t-1}} [p_S \cdot p(x_t|x_{t-1}^{(i)}) + b(x_t|x_{t-1}^{(i)})] w_{t-1|t-1}^{(i)} \quad (7.2)$$

Thus if we split the approximation from the addition operation and apply two independent particle filter applications by using two proposals for target birth part and the rest; we have $D(x_{t-1}|Y_{1:t-1})$ approximation. Assuming that $q_t(x_t^{(i)}|x_{t-1}^{(i)}, Y_{1:t})$ and $r_t(x_t^{(i)}|Y_{1:t})$ are the proposals corresponding to the right hand side part of equation and left hand side of the equation respectively, the weights are estimated as

follows:

$$w_{t|t-1}^{(i)} = \begin{cases} \frac{[p_S \cdot p(x_t | x_{t-1}^{(i)}) + b(x_t | x_{t-1}^{(i)})] w_{t-1|t-1}^{(i)}}{q_t(x_t^{(i)} | x_{t-1}^{(i)}, Y_{1:t})} & , i=1, \dots, L_{t-1} \\ \frac{b(x_t^{(i)})}{J_t r_t(x_t^{(i)} | Y_{1:t})} & , i=L_{t-1}+1, \dots, L_{t-1} + J_t \end{cases}$$

7.1.1.3. Update. Assuming at t we have the set of particles and their associated weights, $\{x_{t|t-1}^{(i)}, w_{t|t-1}^{(i)}\}_{i=1}^{L_{t-1}+J_t}$, that approximates $D(x_t | Y_{1:t-1})$, we want to constitute the updated PHD approximation. Applying the update results in

$$D(x_t | Y_{1:t}) \approx \sum_{i=1}^{L_{t-1}+J_t} w_t^{(i)} \delta_{x_t^{(i)}}(x) \quad (7.3)$$

where

$$w_{t|t}^{(i)} = [1 - p_D] w_{t|t-1}^{(i)} + \sum_{j=1}^{M_t} \frac{p_{DP}(y_t^j | x_t^{(i)}) w_{t|t-1}^{(i)}}{f a(y_t^j) + \sum_{k=1}^{L_{t-1}+J_t} p_{DP}(y_t^j | x_t^{(k)}) w_{t|t-1}^{(k)}} \quad (7.4)$$

7.1.1.4. State Estimation. Target state estimates are extracted from the approximation of the intensity function by applying clustering techniques such as k-means clustering.

7.1.1.5. The Algorithm. If we do not limit the increasing particle number somehow, then in every step the particle number increase by J_t . In order to achieve that we first estimate the expected number of targets for time t by $\hat{N}_t = \sum_{i=1}^{L_{t-1}+J_t} w_{t|t}^{(i)}$, then if we want to allocate ρ particles per time, we perform the resampling such that we have $\rho \hat{N}_t$ particles are generated. Thus by means of resampling we adaptively allocate the particles to the targets. The algorithm is represented in Figure 7.1.

1: **for** $i = 1$ to $L_{t-1} + J_t$ **do**

2: Compute the IS distributions: $q_t(x_t^{(i)}|x_{t-1}^{(i)}, Y_{1:t}), r_t(x_t^{(i)}|Y_{1:t})$

3: Generate offsprings:

$$\hat{x}_t^{(i)} \sim q_t(x_t^{(i)}|x_{t-1}^{(i)}, Y_{1:t}) \text{ for } i = 1, \dots, L_{t-1}$$

$$\hat{x}_t^{(i)} \sim r_t(x_t^{(i)}|Y_{1:t}) \text{ for } i=L_{t-1}, \dots, L_{t-1} + J_t$$

4: Evaluate importance weights:

$$w_{t|t-1}^{(i)} = \begin{cases} \frac{[p_S \cdot p(x_t|x_{t-1}^{(i)}) + b(x_t|x_{t-1}^{(i)})]w_{t-1|t-1}^{(i)}}{q_t(x_t^{(i)}|x_{t-1}^{(i)}, Y_{1:t})} & , i = 1, \dots, L_{t-1} \\ \frac{b(x_t^{(i)})}{J_t r_t(x_t^{(i)}|Y_{1:t})} & , i = L_{t-1}, \dots, L_{t-1} + J_t \end{cases}$$

5: Compute new weights in the update step:

$$w_{t|t}^{(i)} = [1 - p_D]w_{t|t-1}^{(i)} + \sum_{j=1}^{M_t} \frac{p_D p(y_t^j|x_t^{(i)})w_{t|t-1}^{(i)}}{fa(y_t^j) + \sum_{k=1}^{L_{t-1}+J_t} p_D p(y_t^j|x_t^{(k)})w_{t|t-1}^{(k)}}$$

6: Resampling Step

7: Compute the Expected Number of Particles: $\hat{N}_t = \sum_{i=1}^{L_{t-1}+J_t} w_{t|t}^{(i)}$

8: Generate L_t particles via a Resampling Method

$$\left\{ \frac{w_{t|t}^{(i)}}{\hat{N}_t}, \tilde{x}_t^{(i)} \right\}_{i=1}^{L_t} \leftarrow \text{Resample} \left(\left\{ \frac{w_{t|t}^{(i)}}{\hat{N}_t}, x_t^{(i)} \right\}_{i=1}^{L_{t-1}+J_t} \right)$$

9: Multiply the weights by \hat{N}_t to get $\left\{ w_{t|t}^{(i)}, x_t^{(i)} \right\}_{i=1}^{L_t}$

10: **end for**

Figure 7.1. SMCPHD Filter Algorithm.

7.1.2. Improved SMCPHD Filter

The algorithm that is represented in Figure 7.1 has drawbacks and needs to be improved. Firstly target birth may take place anywhere and this results in generating particles that are spreaded all over the state space. Since the bottleneck of a particle filter is the number of particles propagated and we are obliged to have a good approximation to have good estimates; the algorithm gets in trouble with respect to efficiency. A possible solution is rather than generating new born targets over all target space, we can confine these particles to be born near the measurements, namely the regions with high measurement likelihood. In comparison to the SMCPHD algorithm, the results established in [16] are efficient with the same number of particles. The second drawback is the ad hoc clustering approaches of SMCPHD Algorithm that are used for target state estimation. In the same paper, a new approach is also represented for target state and error estimation. In this section we firstly represent the adapted PHD recursion formulation of Improved SMCPHD filter, then represent how the target states are extracted.

7.1.2.1. Improved SMCPHD Formulation of the PHD. Since the new technique rely on improving the efficiency of target births by allowing them to be generated around the measurements, then we must somehow distinguish the survived targets and new born targets. This is performed by using an augmented state vector defined as

$$x_t = \begin{bmatrix} x'_t & \beta_t \end{bmatrix}^T \quad (7.5)$$

where x'_t is the conventional state matrix and β_t is the label of the target that distinguishes a newborn target from a survived target such that

$$\beta_t = \begin{cases} 0 & , \text{ for a survived target} \\ 1 & , \text{ for a newborn target} \end{cases}$$

Now by adapting this convention we modify the PHD prediction and update equations, Equation 6.58 and Equation 6.66 respectively. For the prediction step we need to rephrase the birth PHD and state transition model since the p_S does not depend on β_t . The birth PHD can be restated as follows:

$$b(x_t) = b(x'_t, \beta_t) = \begin{cases} b(x'_t) & , \beta_t = 1 \\ 0 & , \beta_t = 0 \end{cases}$$

The transition model can be derived as follows:

$$p(x_t|x_{t-1}) = p(x'_t, \beta_t|x'_{t-1}, \beta_{t-1}) \quad (7.6)$$

$$= p(x'_t|\beta_t, x'_{t-1}, \beta_{t-1})p(\beta_t|x'_{t-1}, \beta_{t-1}) \quad (7.7)$$

$$= p(x'_t|\beta_t, x'_{t-1}, \beta_{t-1})p(\beta_t|\beta_{t-1}) \quad (7.8)$$

$$= p(x'_t|\beta_t, x'_{t-1})p(\beta_t|\beta_{t-1}) \quad (7.9)$$

$$= p(x'_t|x'_{t-1})p(\beta_t|\beta_{t-1}) \quad (7.10)$$

$$(7.11)$$

such that

$$p(\beta_t|\beta_{t-1}) = \begin{cases} 0 & , \beta_t = 1 \\ 1 & , \beta_t = 0 \end{cases}$$

Assuming no spawning and so dropping $b(x_t|x_{t-1})$ we have the following prediction equation

$$D(x'_t, \beta_t|Y_{1:t-1}) = b(x'_t, \beta_t) + \sum_{\beta_{t-1}=0}^1 \int p_S \cdot p(x'_t, \beta_t|x'_{t-1}, \beta_{t-1}) \cdot D(x'_{t-1}, \beta_{t-1}|Y_{1:t-1}) dx_{t-1} \quad (7.12)$$

Thus,

$$D(x'_t, \beta_t | Y_{1:t-1}) = \begin{cases} b(x'_t) & , \beta_t = 1 \\ \int p_S \cdot p(x'_t, 0 | x'_{t-1}, 0) \cdot D(x'_{t-1}, 0 | Y_{1:t-1}) dx_{t-1} + \\ \int p_S \cdot p(x'_t, 0 | x'_{t-1}, 1) \cdot D(x'_{t-1}, 1 | Y_{1:t-1}) dx_{t-1} & , \beta_t = 0 \end{cases}$$

$$D(x'_t, \beta_t | Y_{1:t-1}) = \begin{cases} b(x'_t) & , \beta_t = 1 \\ \int p_S \cdot p(x'_t | x'_{t-1}) \cdot D(x'_{t-1}, 0 | Y_{1:t-1}) dx_{t-1} + \\ \int p_S \cdot p(x'_t | x'_{t-1}) \cdot D(x'_{t-1}, 1 | Y_{1:t-1}) dx_{t-1} & , \beta_t = 0 \end{cases}$$

Similar operations are applied to the update equation. We consider the measurement model and detection probability. We assume that new targets are always detected, so the detection probability is:

$$p_D = \begin{cases} 1 & , \beta_t = 1 \\ p_D & , \beta_t = 0 \end{cases}$$

The measurement model does not rely on β_t :

$$p(y_t | x_t) = p(y_t | x'_t, \beta_t) = p(y_t | x'_t) \quad (7.13)$$

When we adapt the above convention, the PHD update equation has the following form:

$$D(x'_t, \beta_t | Y_{1:t}) \approx [1 - p_D] D(x'_t, \beta_t | Y_{1:t-1}) + \sum_{i=1}^{M_t} \frac{p_D p(y_t^i | x'_t, \beta_t) D(x'_t, \beta_t | Y_{1:t-1})}{f a(y_t^i) + \sum_{\beta_t=0}^1 \int p_D p(y_t^i | x'_t, \beta_t) D(x'_t, \beta_t | Y_{1:t-1}) dx_t} \quad (7.14)$$

If $\beta_t = 0$;

$$D(x'_t, 0|Y_{1:t}) \approx [1 - p_D]D(x'_t, 0|Y_{1:t-1}) + \sum_{i=1}^{M_t} \frac{p_D p(y_t^i|x'_t) D(x'_t, 0|Y_{1:t-1})}{f a(y_t^i) + \int p_D p(y_t^i|x'_t) D(x'_t, 0|Y_{1:t-1}) dx_t + \int p_D p(y_t^i|x'_t) D(x'_t, 1|Y_{1:t-1}) dx_t} \quad (7.15)$$

If $\beta_t = 1$;

$$D(x'_t, 1|Y_{1:t}) \approx \sum_{i=1}^{M_t} \frac{p_D p(y_t^i|x'_t) D(x'_t, 1|Y_{1:t-1})}{f a(y_t^i) + \int p_D p(y_t^i|x'_t) D(x'_t, 0|Y_{1:t-1}) dx_t + \int p_D p(y_t^i|x'_t) D(x'_t, 1|Y_{1:t-1}) dx_t} \quad (7.16)$$

Now for the prediction and update equations of improved SMCPHD, the integrals needs to be approximated by the weighted particles. But firstly we consider the method of generating particles of the birth targets. For j th measurement of time t , denoted as y_t^j , i th of total ρ newborn particles, denoted as $x_{t|t-1,b}^{(i)}$, is generated such that y_t^j is a random sample from $p(y_t^j|x_{t|t-1,b}^{(i)})b(x_{t|t-1,b}^{(i)})$ and the weights of particles of the birth targets are uniform:

$$w_{t|t-1,b}^{(i)} = \frac{v_{t|t-1,b}}{\rho M_t} \quad (7.17)$$

such that $v_{t|t-1,b}$ is the expected number of newborn particles from $t-1$ to t as follows:

$$v_{t|t-1,b} = \sum_{j=1}^{M_t} \sum_{i=1}^{\rho} p(y_t^j|x_{t|t-1,b}^{(i+(j-1)\rho)})b(x_{t|t-1,b}^{(i+(j-1)\rho)}) \quad (7.18)$$

For the prediction step to estimate the particles and weights of the surviving targets, respectively denoted as $x_{t|t-1,s}^{(i)}$, $w_{t|t-1,s}^{(i)}$, it is a classical particle filtering example, namely we just sample from a proposal, say $p(x_t|x_{t-1})$. Assuming that I_{t-1} is the total number of particles for time $t-1$, For the update step we update the weights of

surviving and newborn targets as follows:

$$w_{t|t,s}^{(i)} = [1 - p_D]w_{t|t-1,s}^{(i)} + \sum_{j=1}^{M_t} \frac{p_{DP}(y_t^j | x_{t|t-1,s}^{(i)})w_{t|t-1,s}^{(i)}}{fa(y_t^j) + \sum_{k=1}^{\rho M_t} w_{t|t-1,b}^{(k)} + \sum_{k=1}^{I_{t-1}} p_{DP}(y_t^j | x_{t|t-1,s}^{(k)})w_{t|t-1,s}^{(k)}} \quad (7.19)$$

$$w_{t|t,b}^{(i)} = \sum_{j=1}^{M_t} \frac{w_{t|t-1,b}^{(i)}}{fa(y_t^j) + \sum_{k=1}^{\rho M_t} w_{t|t-1,b}^{(k)} + \sum_{k=1}^{I_{t-1}} p_{DP}(y_t^j | x_{t|t-1,s}^{(k)})w_{t|t-1,s}^{(k)}} \quad (7.20)$$

7.1.2.2. State and Error Estimation. The examination of Equation 7.15 gives the intuition that the update state is the creation of $M_t + 1$ replicas of the particle set $\{x_{t|t-1}^i; i = 1, \dots, I_{t-1}\}$ since the summation of $M_t + 1$ terms is the same as the union of replicated particles. Then denoting the weights of the corresponding replica of the j th ($j \in \{0, 1, \dots, M_t\}$) particle set by $\{w_{t|t,p}^{i,j}; i = 1, \dots, I_{t-1}\}$, then the weights can be estimated by the following:

If $j = 0$;

$$w_{t|t,s}^{i,j} = [1 - p_D]w_{t|t-1,s}^{(i)} \quad (7.21)$$

Else if $j = 1, \dots, M_t$

$$w_{t|t,s}^{i,j} = \frac{p_{DP}(y_t^j | x_{t|t-1,s}^{(i)})w_{t|t-1,s}^{(i)}}{fa(y_t^j) + \sum_{k=1}^{\rho M_t} w_{t|t-1,b}^{(k)} + \sum_{k=1}^{I_{t-1}} p_{DP}(y_t^j | x_{t|t-1,s}^{(k)})w_{t|t-1,s}^{(k)}} \quad (7.22)$$

Now we define $W_{t,s}^j = \sum_{i=1}^{I_{t-1}} w_{t|t,s}^{i,j}$, that is the total weight assigned to a particle set replica $j = 1, \dots, M_t$. If y_t^j is a measurement that results in non-zero likelihood for some particles, it is possibly originated from a target, thus results in high $W_{t,s}^j$, otherwise it will tend to zero. For each particle set replica the state estimate, \hat{x}_t^j and its covariance

matrix \mathbf{P}_t^j , that is a value of error, can be computed as

$$\hat{x}_t^j = \sum_{i=1}^{I_{t-1}} w_{t|t,s}^{i,j} x_{t|t-1,s}^i \quad (7.23)$$

$$\mathbf{P}_t^j = \sum_{i=1}^{I_{t-1}} w_{t|t,s}^{i,j} (x_{t|t-1,s}^i - \hat{x}_t^j)(x_{t|t-1,s}^i - \hat{x}_t^j)^T \quad (7.24)$$

The algorithm is represented Figure 7.2.

7.2. Gaussian Mixture Based Approaches

SMCPHD provides an approximation to the PHD function without posing any restrictions. On the other hand, Gaussian Mixture PHD provides a closed form solution for the PHD function by assuming the linear Gaussian transition and measurement models. For the first time Vo and Ma showed in [17] that if the initial function is a gaussian mixture, then the posterior PHD is also a gaussian mixture. Furthermore the idea is extended to the nonlinear case (but still Gaussian) in [18] by adopting the approaches of EKF and UKF. We represent these approaches in this section respectively. Since we represented EKF and UKF in Chapter 4, rather than reexplaining we just refer to the PHD implementation strategies of these subjects.

7.2.1. GMPHD Filter

Linear Gaussian model was discussed in Section 4.1, we represented the model particularly in Equation 4.4. For that specific model we impose the following assumptions.

- (i) Initial PHD function is a Gaussian Mixture.
- (ii) PHDs of target birth and target spawning are mixture of Gaussians as represented

```

1: INPUT:
2: 1. Particle set of surviving targets:  $\{w_{t-1,s}^{(i)}, x_{t-1,s}^{(i)}\}_{i=1}^{L_{t-1}}$ 
3: 2. Particle set of newborn targets:  $\{w_{t-1,b}^{(i)}, x_{t-1,b}^{(i)}\}_{i=1}^{S_{t-1}}$ 
4: 3. Measurement Set  $Z_t = \{z_t^1, \dots, z_t^{M_t}\}$ 
5: Union New Born Targets and Surviving Targets:
    $\{w_{t-1}^{(i)}, x_{t-1}^{(i)}\}_{i=1}^{I_{t-1}} = \{w_{t-1,s}^{(i)}, x_{t-1,s}^{(i)}\}_{i=1}^{L_{t-1}} \cup \{w_{t-1,b}^{(i)}, x_{t-1,b}^{(i)}\}_{i=1}^{S_{t-1}}$ 
6: for  $i = 1$  to  $I_{t-1}$  do
7:   Draw  $x_{t|t-1,s}^{(i)} \sim p(x|x_{t-1}^{(i)})$ 
8:   Compute Weights  $w_{t|t-1,s}^{(i)} = p_S w_{t-1}^{(i)}$ 
9:   Compute Weights  $w_{t|t,s}^{(i)}$  according to Equation 7.19.
10:  for  $j = 1$  to  $M_t$  do
11:    Compute Weights  $w_{t|t,s}^{(i),j}$  according to Equation 7.22
12:  end for
13: end for
14: for  $j = 1$  to  $M_t$  do
15:   if  $W_{t,s}^j = \sum_{i=1}^{I_{t-1}} w_{t|t,s}^{(i),j} > \tau$  then
16:    Compute  $\hat{x}_t^j, \mathbf{P}_t^j$  according to Equations 7.23, 7.24
17:   end if
18: end for
19: Estimate Cardinality:  $v_t^s = \sum_{i=1}^{I_{t-1}} w_{t|t,s}^{(i)}$ ;  $\hat{n}_t = \text{ROUND}(v_t^s)$ 
20: Estimate Number of Particles to Resample:  $L_t = \hat{n}_t \cdot \rho$ 
21: Resample  $L_t$  times from  $\{w_{t|t,s}^{(i)}/v_t^s, x_{t|t-1,s}^{(i)}\}_{i=1}^{I_{t-1}}$  to obtain  $\{w_{t,s}^{(i)}, x_{t,s}^{(i)}\}_{i=1}^{L_t}$ 
   with  $w_{t,s}^{(i)} = v_t^s/L_t$ 
22: for  $j = 1$  to  $M_t$  do
23:   for  $k = 1$  to  $\rho$  do
24:      $i = k + (j - 1) \rho$ 
25:     Draw  $x_{k|k-1,b}^{(i)} \sim b(x|y_t^j)$ 
26:     Compute Weights:  $w_{t|t-1,b}^{(i)} = v_{t|t-1}^b / (\rho M_t)$ 
27:     Compute Weights:  $w_{t|t,b}^{(i)}$  according to Equation 7.20.
28:   end for
29: end for
30: Estimate Number of Particles to Resample:  $S_t = \rho M_t$ 
31: Resample  $S_t$  times from  $\{w_{t|t,b}^{(i)} / \sum_{i=1}^{S_t} w_{t|t-1,b}^{(i)}, x_{t|t-1,b}^{(i)}\}_{i=1}^{S_t}$  to obtain  $\{w_{t,b}^{(i)}, x_{t,b}^{(i)}\}_{i=1}^{S_t}$ 

```

Figure 7.2. Improved SMCPHD Filter Algorithm.

respectively.

$$b(x_t) = \sum_{i=1}^{J_t^{(b)}} w_t^{(b,i)} \mathcal{N}(x_t; m_t^{(b,i)}, P_t^{(b,i)}) \quad (7.25)$$

$$b(x_t|x_{t-1}) = \sum_{i=1}^{J_t^{(b')}} w_t^{(b',i)} \mathcal{N}(x_t; x_{t-1} + m_t^{(b',i)}, P_t^{(b',i)}) \quad (7.26)$$

where $m_t^{(b,i)}, P_t^{(b,i)}$ are the mean and covariances of the i th Gaussian component of the target birth PHD and $x_{t-1} + m_t^{(b',i)}, P_t^{(b',i)}$ are the mean and covariances of the i th Gaussian component of the target spawning PHD.

- (iii) The survival and detection probabilities, p_D, p_S , does not vary with respect to a target or time. They are constant.

Following we represent the recursive equations of GMPHD.

7.2.1.1. Prediction. Assume that we have:

$D(x_{t-1}|Y_{1:t-1}) = \sum_{i=1}^{J_{t-1}} w_t^{(i)} \mathcal{N}(x_{t-1}; m_{t-1}^{(i)}, P_{t-1}^{(i)})$ and want to construct $D(x_t|Y_{1:t-1})$.

$$D(x_t|Y_{1:t-1}) = b(x_t) + \int (p_S \cdot p(x_t|x_{t-1}) + b(x_t|x_{t-1})) \cdot D(x_{t-1}|Y_{1:t-1}) dx_{t-1} \quad (7.27)$$

$$\begin{aligned} &= \sum_{i=1}^{J_t^{(b)}} w_t^{(b,i)} \mathcal{N}(x_t; m_t^{(b,i)}, P_t^{(b,i)}) \\ &+ \int \left(p_S \cdot \mathcal{N}(x_t; Ax_{t-1}, Q) \cdot \sum_{i=1}^{J_t^{(b')}} w_t^{(b',i)} \mathcal{N}(x_t; x_{t-1} + m_t^{(b',i)}, P_t^{(b',i)}) \right) \\ &\cdot \sum_{i=1}^{J_{t-1}} w_t^{(i)} \mathcal{N}(x_{t-1}; m_{t-1}^{(i)}, P_{t-1}^{(i)}) dx_{t-1} \end{aligned} \quad (7.28)$$

Theorem 7.1. *Given A, d, Q, m and P of appropriate dimensions, with Q and P*

positive definite.

$$\int \mathcal{N}(x_t; Ax_{t-1} + d, Q) \mathcal{N}(x_{t-1}; m, P) = \mathcal{N}(x_t; Am + d, Q + APA^T) \quad (7.29)$$

Now applying the Theorem 7.1 to the integral of the derived expression and making some simplifications we have the following expression for the prediction equation.

$$D(x_t|Y_{1:t-1}) = b(x_t) + D^{(\alpha)}(x_t|Y_{1:t-1}) + D^{(\beta)}(x_t|Y_{1:t-1}) \quad (7.30)$$

$$D^{(\alpha)}(x_t|Y_{1:t-1}) = p_S \sum_{i=1}^{J_{t-1}} w_{t-1}^{(i)} \mathcal{N}(x_t; m_{t|t-1}^{(i)}, P_{t|t-1}^{(i)}) \quad (7.31)$$

$$m_{t|t-1}^{(i)} = Am_{t-1}^{(i)} \quad (7.32)$$

$$P_{t|t-1}^{(i)} = Q + F_{t-1} P_{t-1}^{(i)} F_{t-1}^T \quad (7.33)$$

$$D^{(\beta)}(x_t|Y_{1:t-1}) = \sum_{i=1}^{J_{t-1}} \sum_{j=1}^{J_t^{(b')}} w_{t-1}^{(i)} w_t^{(b',i)} \mathcal{N}(x_t; m_{t|t-1}^{(i,j)}, P_{t|t-1}^{(i,j)}) \quad (7.34)$$

$$m_{t|t-1}^{(i,j)} = m_{t-1}^{(i)} + m_t^{(b',j)} \quad (7.35)$$

$$P_{t|t-1}^{(i,j)} = P_{t-1}^{(i)} + P_t^{(b',j)} \quad (7.36)$$

7.2.1.2. Update. Assume that we have:

$D(x_t|Y_{1:t-1}) = \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^{(i)} \mathcal{N}(x_t; m_{t|t-1}^{(i)}, P_{t|t-1}^{(i)})$ and want to construct $D(x_t|Y_{1:t})$.

$$D(x_t|Y_{1:t}) \approx [1 - p_D]D(x_t|Y_{1:t-1}) + \sum_{i=1}^{M_t} \frac{p_D p(y_t^i|x_t) D(x_t|Y_{1:t-1})}{fa(y_t^i) + \int p_D p(y_t^i|x_t) D(x_t|Y_{1:t-1}) dx_t} \quad (7.37)$$

$$\approx [1 - p_D] \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^{(i)} \mathcal{N}(x_t; m_{t|t-1}^{(i)}, P_{t|t-1}^{(i)}) \quad (7.38)$$

$$+ \sum_{j=1}^{M_t} \frac{p_D \mathcal{N}(x_t; Cx_t, R) \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^{(i)} \mathcal{N}(x_t; m_{t|t-1}^{(i)}, P_{t|t-1}^{(i)})}{fa(y_t^j) + \int p_D \mathcal{N}(x_t; Cx_t, R) \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^{(i)} \mathcal{N}(x_t; m_{t|t-1}^{(i)}, P_{t|t-1}^{(i)}) dx_t} \quad (7.39)$$

Theorem 7.2. *Given C , R , m and P of appropriate dimensions, with P and R positive definite.*

$$\mathcal{N}(z; Cx, R) \mathcal{N}(x; m, P) = q(z) \mathcal{N}(x; \bar{m}, \bar{P}) \quad (7.40)$$

Now applying the Theorem 7.1 to the integrals and Theorem 7.2 to the appropriate product of Gaussians we have the following expression for the update equation.

$$D(x_t|Y_{1:t}) \approx (1 - p_D)D(x_t|Y_{1:t-1}) + \sum_{j=1}^{M_t} D(x_t|y_t^j) \quad (7.41)$$

where

$$D(x_t|y_t^j) = \sum_{i=1}^{J_{t|t-1}} \frac{p_D w_{t|t-1}^{(i)} q_t^{(i)}(y_t^j) \mathcal{N}(x; m_t^{(i)}(y_t^j), P_t^{(i)})}{fa(y_t^j) + p_D \sum_{k=1}^{J_{t|t-1}} w_{t|t-1}^{(k)} q_t^{(k)}(y_t^j)} \quad (7.42)$$

$$q_t^{(k)}(y_t^j) = \mathcal{N}(y_t^j; Cm_{t|t-1}^{(k)}, R + CP_{t|t-1}^{(k)}C^T) \quad (7.43)$$

$$m_t^{(i)}(y_t^j) = m_{t|t-1}^{(i)} + K_t^{(i)}(y_t^j - Cm_{t|t-1}^{(i)}) \quad (7.44)$$

$$P_t^{(i)} = [I - K_t^{(i)}C]P_{t|t-1}^{(i)} \quad (7.45)$$

$$K_t^{(i)} = P_{t|t-1}^{(i)}C^T(CP_{t|t-1}^{(i)}C^T + R)^{-1} \quad (7.46)$$

The algorithm and the state estimation process are represented in Figure 7.3 and Figure 7.4 respectively.

7.2.1.3. Pruning for the GMPHD. It can be observed from Equations 7.30 and 7.41 that at some time t , there exists $(J_{t-1}(1 + J_t^{(b')}) + J_t^{(b)})(1 + |Y_t|)$ Gaussian components. As when t increases, the number of these components increases; so we need a method of dealing with the problem. Deleting the components with lower peaks is a way of handling the problem. Moreover we can also approximate the two near located peaks of Gaussians by one more strong Gaussian. The pruning algorithm including the truncating and merging skills is represented in Figure 7.5.

7.2.2. EKFPHD Filter

We represented Extended Kalman filter to the nonlinear cases by using Taylor series expansion in Section 4.2. The intuition is to linearize locally the transition and the measurement model such that Kalman recursions can be applied and thereby approximate the posterior by a Gaussian. We can also apply the intuition to the GMPHD filter and adjust the filter with respect to the nonlinear scenarios. The algorithm is represented in Figure 7.6 in which we call it EKFPHD filter.

7.2.3. UKFPHD Filter

Similar to the EKFPHD filter that is derived from the conventional Kalman filter by applying the EKF formulation to it, UKFPHD is analogously derived by means of UKF represented in Section 4.3. Basically, unlike the linearization of EKF, the idea of UKF is to approximate the posterior using the unscented transformation of some weighted points. Since the prediction and update equations of PHD consist of Gaussians, we can approximate independent peaks of the equations by applying UKF and EKF techniques. The algorithm is represented in Figure 7.7.

```

1: INPUT:
2: 1.Components of Gaussian Mixture  $\{w_{t-1}^{(i)}, m_{t-1}^{(i)}, P_{t-1}^{(i)}\}_{i=1}^{J_{t-1}}$ 
3: 2.a measurement set  $Z_t$ 
4: Step 1:Prediction for target birth
5:  $i = 0$ 
6: for  $j = 1$  to  $J_t^{(b)}$  do
7:    $i = i + 1, w_{t|t-1}^{(i)} = w_t^{(b,j)}, m_{t|t-1}^{(i)} = m_t^{(b,j)}, P_{t|t-1}^{(i)} = P_t^{(b,j)}$ 
8: end for
9: for  $j = 1$  to  $J_t^{(b')}$  do
10:   for  $k = 1$  to  $J_{t-1}$  do
11:      $i = i + 1, w_{t|t-1}^{(i)} = w_t^{(k)} w_t^{(b',j)}, m_{t|t-1}^{(i)} = m_{t-1}^{(b',j)} + m_{t-1}^{(k)}, P_{t|t-1}^{(i)} = P_t^{(b',j)} + P_{t-1}^{(k)}$ 
12:   end for
13: end for
14: Step 2:Prediction for existing targets
15: for  $j = 1$  to  $J_{t-1}$  do
16:    $i = i + 1, w_{t|t-1}^{(i)} = p_S w_{t-1}^{(j)}, m_{t|t-1}^{(i)} = A m_{t-1}^{(j)}, P_{t|t-1}^{(i)} = Q + A P_{t-1}^{(j)} A^T$ 
17: end for
18:  $J_{t|t-1} = i$ 
19: Step 3:Construction of PHD Update Components
20: for  $j = 1$  to  $J_{t|t-1}$  do
21:    $\eta_{t|t-1}^{(j)} = C m_{t|t-1}^{(j)}, S_t^{(j)} = R + C P_{t|t-1}^{(j)} C^T, K_t^{(j)} = P_{t|t-1}^{(j)} C^T [S_t^{(j)}]^{-1}, P_{t|t}^{(j)} = [I - K_t^{(j)} C] P_{t|t-1}^{(j)}$ 
22: end for
23: Step 4:Update
24: for  $j = 1$  to  $J_{t|t-1}$  do
25:    $w_t^{(j)} = (1 - p_D) w_{t|t-1}^{(j)}, m_t^{(j)} = m_{t|t-1}^{(j)}, P_t^{(j)} = P_{t|t-1}^{(j)}$ 
26: end for
27:  $k = 0$ 
28: for  $\forall y \in Y_t$  do
29:    $k = k + 1$ 
30:   for  $j = 1$  to  $J_{t|t-1}$  do
31:      $w_t^{(k J_{t|t-1} + j)} = p_D w_{t|t-1}^{(j)} \mathcal{N}(y; \eta_{t|t-1}^{(j)}, S_t^{(j)})$ 
32:      $m_t^{(k J_{t|t-1} + j)} = m_{t|t-1}^{(j)} + K_t^{(j)} (y - \eta_{t|t-1}^{(j)}), P_t^{(k J_{t|t-1} + j)} = P_{t|t}^{(j)}$ 
33:   end for
34:    $w_t^{(k J_{t|t-1} + j)} = \frac{w_t^{(k J_{t|t-1} + j)}}{f a(y) + \sum_{i=1}^{J_{t|t-1}} w_t^{(k J_{t|t-1} + i)}}, \text{ for } j=1, \dots, J_{t|t-1}$ 
35: end for
36:  $J_t = k J_{t|t-1} + J_{t|t-1}$ 
37: OUTPUT:  $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$ 

```

Figure 7.3. GMPHD Filter Algorithm.

```

1: INPUT:  $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$ 
2: Set  $\hat{X}_t = \emptyset$ 
3: for  $i = 1$  to  $J_t$  do
4:   if  $w_t^{(i)} > 0,5$  then
5:     for  $j = 1$  to  $ROUND(w_t^{(i)})$  do
6:        $\hat{X}_t = [\hat{X}_t, w_t^{(i)}]$ 
7:     end for
8:   end if
9: end for
10: OUTPUT:  $\hat{X}_t$  as the multitarget state estimate

```

Figure 7.4. GMPHD Filter State Estimation.

```

1: INPUT:
2: 1.Components of Gaussian Mixture  $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$ 
3: 2.a truncation threshold  $\tau$ 
4: 3.a merging threshold  $M$ 
5: 4.maximum allowable number of Gaussian terms  $J_{max}$ 
6:  $r = 0$ 
7:  $I = \{i = 1, \dots, J_t | w_t^{(i)} > \tau\}$ 
8: repeat
9:    $r = r + 1$ 
10:   $j = \operatorname{argmax}_{i \in I} w_t^{(i)}$ 
11:   $L = \{i \in I | (m_t^{(i)} - m_t^{(j)})^T (P_t^{(i)})^{-1} (m_t^{(i)} - m_t^{(j)}) \leq M\}$ 
12:   $\tilde{w}_t^{(r)} = \sum_{i \in L} w_t^{(i)}$ 
13:   $\tilde{m}_t^{(r)} = \frac{1}{\tilde{w}_t^{(r)}} \sum_{i \in L} w_t^{(i)} x_t^{(i)}$ 
14:   $\tilde{P}_t^{(r)} = \frac{1}{\tilde{w}_t^{(r)}} \sum_{i \in L} w_t^{(i)} (P_t^{(i)} + (\tilde{m}_t^{(r)} - m_t^{(i)})(\tilde{m}_t^{(r)} - m_t^{(i)})^T)$ 
15:   $I = I \setminus L$ 
16: until  $I = \emptyset$ 
17: if  $r > J_{max}$  then replace  $\{\tilde{w}_t^{(i)}, \tilde{m}_t^{(i)}, \tilde{P}_t^{(i)}\}_{i=1}^r$  by those of the  $J_{max}$  Gaussians with largest weights.
18: OUTPUT:  $\{\tilde{w}_t^{(i)}, \tilde{m}_t^{(i)}, \tilde{P}_t^{(i)}\}_{i=1}^r$  as pruned Gaussian components.

```

Figure 7.5. Pruning for GMPHD Filter.

1: **INPUT:**

2: 1.Components of Gaussian Mixture $\{w_{t-1}^{(i)}, m_{t-1}^{(i)}, P_{t-1}^{(i)}\}_{i=1}^{J_{t-1}}$

3: 2.a measurement set Z_t

4: **Step 1:Prediction for target birth**

5: Follow Step 1 of GMPHD Algorithm

6: **Step 2:Prediction for existing targets**

7: **for** $j = 1$ to J_{t-1} **do**

8: $i = i + 1, w_{t|t-1}^{(i)} = p_S w_{t-1}^{(j)}, m_{t|t-1}^{(i)} = f_t(m_{t-1}^{(j)}, 0)$
 $P_{t|t-1}^{(i)} = G_{t-1} Q G_{t-1}^T + A_{t-1} P_{t-1}^{(j)} A_{t-1}^T$ where

$$A_{t-1} = \left. \frac{\partial f_t(x_{t-1}, 0)}{\partial x_{t-1}} \right|_{x_{t-1}=m_{t-1}^{(j)}}, G_{t-1} = \left. \frac{\partial f_t(m_{t-1}^{(j)}, w_{t-1})}{\partial w_{t-1}} \right|_{w_{t-1}=0}$$

9: **end for**

10: $J_{t|t-1} = i$

11: **Step 3:Construction of PHD Update Components**

12: **for** $j = 1$ to $J_{t|t-1}$ **do**

13: $\eta_{t|t-1}^{(j)} = h_t(m_{t|t-1}^{(j)}, 0), S_t^{(j)} = U_t R U_t^T + C_t P_{t|t-1}^{(j)} C_t^T$
 $K_t^{(j)} = P_{t|t-1}^{(j)} C_t^T [S_t^{(j)}]^{-1}, P_{t|t}^{(j)} = [I - K_t^{(j)} C_t] P_{t|t-1}^{(j)}$ where

$$C_t = \left. \frac{\partial h_t(x_t, 0)}{\partial x_t} \right|_{x_t=m_{t|t-1}^{(j)}}, U_t = \left. \frac{\partial h_t(m_{t|t-1}^{(j)}, v_t)}{\partial v_t} \right|_{v_t=0}$$

14: **end for**

15: **Step 4:Update**

16: Follow Step 4 of GMPHD Algorithm to obtain $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$

17: **OUTPUT:** $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$

Figure 7.6. EKFPHD Filter Algorithm.

1: INPUT:
2: 1.Components of Gaussian Mixture $\{w_{t-1}^{(i)}, m_{t-1}^{(i)}, P_{t-1}^{(i)}\}_{i=1}^{J_{t-1}}$
3: 2.a measurement set Z_t
4: Step 1:Construction of target birth components
5: Follow Step 1 of GMPHD Algorithm
6: for $j = 1$ to i do
7: Set $\mu = \begin{bmatrix} m_{t t-1}^{(j)} \\ 0 \end{bmatrix}$, $C = \begin{bmatrix} P_{t t-1}^{(j)} & 0 \\ 0 & R \end{bmatrix}$ (We denote $x_{t-1}^0, P_{t-1 t-1}$ of unscented transformation in Section 4.3 by μ, C)
8: Use unscented transformation with mean μ and covariance C to generate a set of sigma points and weights, denoted by $\{z_t^{(k)}, u^{(k)}\}_{k=0}^L$
9: Partition $z_t^{(k)} = [(x_{t t-1}^{(k)})^T, (v_t^{(k)})^T]^T$, for $k = 0, 1, \dots, L$
10: Compute $y_{t t-1}^{(k)} = h_t(x_{t t-1}^{(k)}, v_t^{(k)})$, for $k = 0, 1, \dots, L$ $\eta_{t t-1}^{(j)} = \sum_{k=0}^L u^{(k)} y_{t t-1}^{(k)}$, $S_t^{(j)} = \sum_{k=0}^L u^{(k)} (y_{t t-1}^{(k)} - \eta_{t t-1}^{(j)})(y_{t t-1}^{(k)} - \eta_{t t-1}^{(j)})^T$ $G_t^{(j)} = \sum_{k=0}^L u^{(k)} (x_{t t-1}^{(k)} - m_{t t-1}^{(j)})(x_{t t-1}^{(k)} - \eta_{t t-1}^{(j)})^T$, $K_t^{(j)} = G_t^{(j)} [S_t^{(j)}]^{-1}$ $P_{t t}^{(j)} = P_{t t-1}^{(j)} - G_t^{(j)} [S_t^{(j)}]^{-1} [G_t^{(j)}]^T$
11: end for
12: Step 2:Construction of existing target components
13: for $j = 1$ to J_{t-1} do
14: $i = i + 1$
15: $w_{t t-1}^{(i)} = p_S w_{t-1}^{(j)}$
16: Set $\mu = \begin{bmatrix} m_{t-1}^{(i)} \\ 0 \\ 0 \end{bmatrix}$, $C = \begin{bmatrix} P_{t-1}^{(i)} & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{bmatrix}$
17: Use unscented transformation with mean μ and covariance C to generate a set of sigma points and weights, denoted by $\{z_t^{(k)}, u^{(k)}\}_{k=0}^L$
18: Partition $z_t^{(k)} = [(x_{t-1}^{(k)})^T, (w_{t-1}^{(k)})^T, (v_t^{(k)})^T]^T$, for $k = 0, 1, \dots, L$
19: Compute $x_{t t-1}^{(k)} = f_t(x_{t-1}^{(k)}, w_{t-1}^{(k)})$, for $k = 0, 1, \dots, L$ $y_{t t-1}^{(k)} = h_t(x_{t t-1}^{(k)}, v_t^{(k)})$, for $k = 0, 1, \dots, L$ $m_{t t-1}^{(i)} = \sum_{k=0}^L u^{(k)} x_{t t-1}^{(k)}$, $P_{t t-1}^{(i)} = \sum_{k=0}^L u^{(k)} (x_{t t-1}^{(k)} - m_{t t-1}^{(i)})(x_{t t-1}^{(k)} - m_{t t-1}^{(i)})^T$ $\eta_{t t-1}^{(i)} = \sum_{k=0}^L u^{(k)} y_{t t-1}^{(k)}$, $S_t^{(i)} = \sum_{k=0}^L u^{(k)} (y_{t t-1}^{(k)} - \eta_{t t-1}^{(i)})(y_{t t-1}^{(k)} - \eta_{t t-1}^{(i)})^T$ $G_t^{(i)} = \sum_{k=0}^L u^{(k)} (x_{t t-1}^{(k)} - m_{t t-1}^{(i)})(x_{t t-1}^{(k)} - \eta_{t t-1}^{(i)})^T$, $K_t^{(i)} = G_t^{(i)} [S_t^{(i)}]^{-1}$ $P_{t t}^{(i)} = P_{t t-1}^{(i)} - G_t^{(i)} [S_t^{(i)}]^{-1} [G_t^{(i)}]^T$
20: end for
21: $J_{t t-1} = i$
22: Step 3:Update
23: Follow Step 4 of GMPHD Algorithm to obtain $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$
24: OUTPUT: $\{w_t^{(i)}, m_t^{(i)}, P_t^{(i)}\}_{i=1}^{J_t}$

Figure 7.7. UKFPHD Filter Algorithm.

8. EXPERIMENTS AND RESULTS

In this chapter we represent the performance of the PHD filter under varying receiver characteristics, target intensity and linearity conditions compared to a data association technique, specifically global nearest neighbor data association. As a performance metric we use OSPA Distance we define in Section 8.1. Then we represent our experimental evaluation in two sections categorized according to the linearity of the measurement model. Section 8.2 represents the performance of the PHD filter for linear case and it also investigates how PHD filter is effected from the variations in the receiver characteristic and target number variation parameters. In this section we compare GMPHD with the KFGNN as a data association technique. Then the following section represents the nonlinear case in which we investigate the effect of the nonlinearity to both of the techniques. For the nonlinear case we extend the comparison to EKFPHD, UKFPHD and ISMCPHD as implementation techniques of the nonlinear PHD filters, on the other hand we use EKFGNN as a data association method for comparison purposes.

8.1. OSPA Distance

OSPA distance [64] is a Wasserstein distance based metric consisting of two parts, localization and cardinality errors. Informal definition can be expressed as the error per target and formally can be defined as:

$$\bar{e}_{p,loc}^{(c)}(X, Y) = \left(\frac{1}{n} \min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p \right)^{1/p} \quad (8.1)$$

$$\bar{e}_{p,card}^{(c)}(X, Y) = \left(\frac{c^p(n-m)}{n} \right)^{1/p} \quad (8.2)$$

if $m \leq n$, and $\bar{e}_{p,loc}^{(c)}(X, Y) = \bar{e}_{p,loc}^{(c)}(Y, X)$, $\bar{e}_{p,card}^{(c)}(X, Y) = \bar{e}_{p,card}^{(c)}(Y, X)$ else. The notation is as follows: $d^{(c)}(x, y) = \min(c, d(x, y))$ is the distance between x and y cut off at $c > 0$, $d(x, y)$ is an arbitrary norm. Π_k is the set of permutations on $\{1, 2, \dots, k\}$ for

any $k \in N = \{1, 2, \dots\}$. $X = x_1, \dots, x_m$ and $Y = y_1, \dots, y_n$ are state and estimated state set. Here c and p are the parameters of the OSPA Metric. By increasing c , we look for a correspondence in a larger space, along with increasing the penalty of miss, p determines the corresponding norm. Throughout our experiments we use $p = 2$, $c = 5$ in the linear case and $c = 20$ in the nonlinear case.

8.2. Linear Model

In this section we try to find out how target number variation and receiver characteristics effect KFGNN compared to GMPHD. Table 8.1 represents the domain of each parameter and underlined values are their default values. Namely, if we want to find out the effect of parameter x , then we declare other three parameters as constants and set them their default values. In each experiment we use scenarios consisting of 100 time intervals, then we get the average OSPA distance. Moreover, in order to sweep the effects of the extremum scenarios, we run 30 Monte Carlo simulations and average the averaged OSPA distance value again. Lastly, we assumed that initially there are two targets present that are being tracked by the filter.

Table 8.1. Domain of Each Parameter for Linear Model Experiments.

Parameter				
λ_{FA}	10	<u>30</u>	50	70
P_b	0.05	<u>0.2</u>	0.4	0.6
P_s	0.8	0.9	<u>0.95</u>	1
P_D	0.6	0.8	<u>0.95</u>	1

8.2.1. Realization of the Filters

In this section we show an instance of the realization of both of the filter with the default values of the parameters. Table 8.2 represents model of the linear environment. In order to represent the model we use the default values in the model. However the

parameters expressed above are not fixed throughout the experiments. By using the

Table 8.2. State Transition and Measurement Models of the Linear Scenario.

Transition Model	Measurement Model
$x_t^i = Ax_{t-1}^i + w_{t-1}$,	$y_t^i = \begin{cases} Cx_t^i + v_t & , \text{ if detection occurs} \\ \emptyset & , \text{ if no detection} \end{cases}$
$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
$w_{t-1} \sim \mathcal{N}(0, Q), Q = \begin{pmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{pmatrix}$	$v_t \sim \mathcal{N}(0, R), R = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix}$
$x_0^i = [X_0, V_0, X_0, V_0]^T$	$ c_t \sim PO(\lambda_{FA}), \lambda_{FA} = 30$
$X_0 \sim U(-500, 500), V_0 \sim U(-5, 5)$	$c_t^i \sim [U(-500, 500), U(-500, 500)]^T$
$p_b = 0.2, p_S = 0.95$	$p_D = 0.95$
$x_t = \bigcup_{k=1}^{ \text{tracks} } x_t^k$	$y_t = \bigcup_{k=1}^{ \text{obs} } y_t^k \bigcup_{k=1}^{ \text{c}_t } c_t^k$

expressed model, all the measurements generated are superimposed in Figure 8.1. That is, both of the filters are applied to these measurements.

Results of the KFGNN filter and GMPHD filter are depicted in Figure 8.2 and Figure 8.3 respectively. By just glancing at the resulting figures we can infer that KFGNN has some latency for deciding about a target births and target deaths compared to GMPHD. On the other hand GMPHD can decide some irrelevant measurements as targets lasting just for one time interval. This is just because both of the filters have totally different target birth strategy. That is, while KFGNN looks for two measurements in four time intervals, GMPHD propagates a mixture of gaussians and decides target if corresponding Gaussian peak exceeds threshold. Figure 8.4 represents the resulting localization, cardinality and total OSPA distance

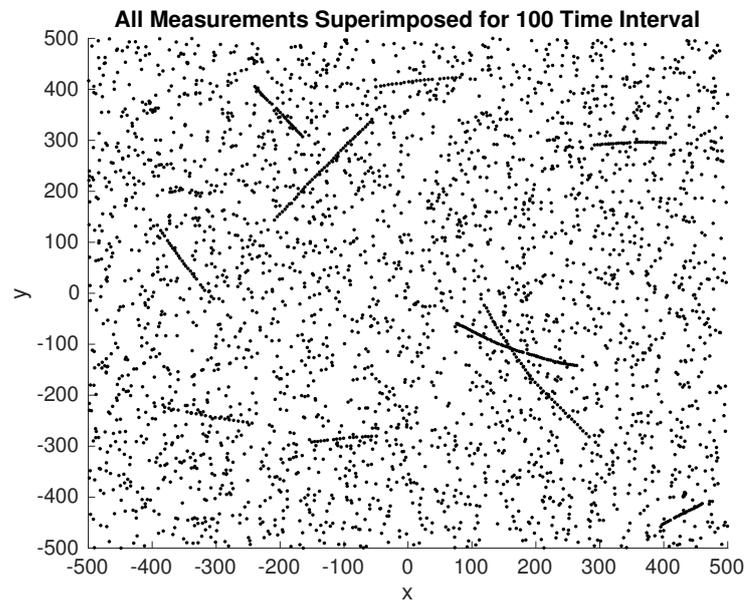


Figure 8.1. The Measurements Generated According to the Model in Table 8.2.

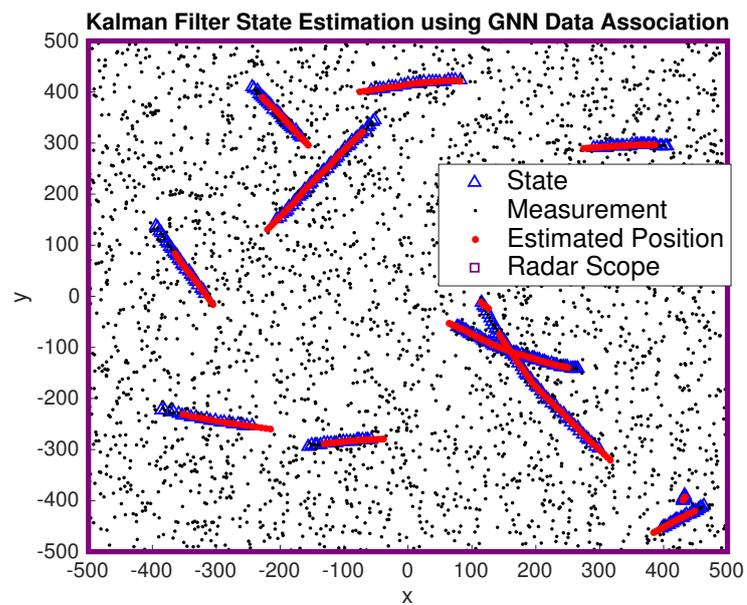


Figure 8.2. KFGNN Filtering of the Measurements in Figure 8.1.

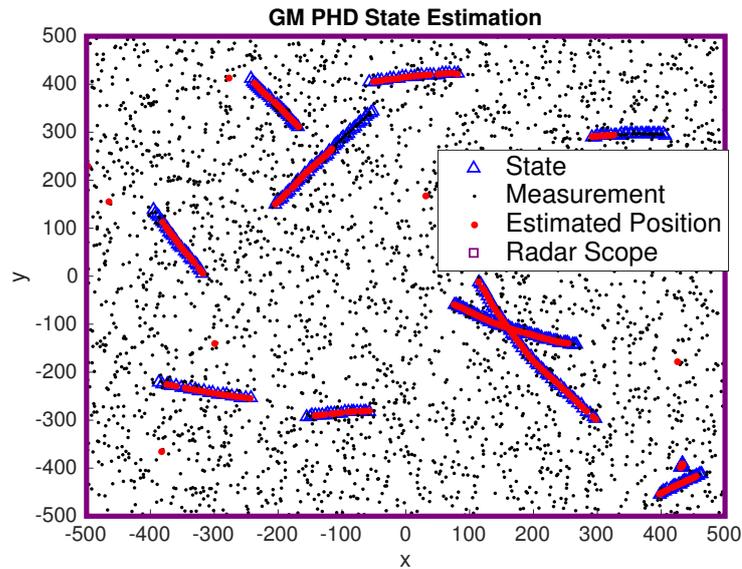


Figure 8.3. GMPHD Filtering of the Measurements in Figure 8.1.

comparison over time. With these default values of the parameters GMPHD performs better than KFGNN by an OSPA distance of 1.82 compared to 2.35. In the following section we see how OSPA distance is effected by changes in parameters.

8.2.2. Effects of the Parameters

In this section we see how OSPA distance changes when environmental and sensor parameters are changed and try to decide in which circumstances we should prefer which technique. As told before the parameters to be examined are λ_{FA} , P_b , P_s and P_D .

8.2.2.1. Receiver Characteristic Parameters: λ_{FA} and P_D . As described in Chapter 6, PHD is an intensity function and may be effected by every measurements even if they are clutters. However, data association techniques can discard most of the false alarms by using gating strategies. In other words data association techniques just care about the measurements that are in the gating regions and the number of measurements outside the gating region does not effect the result. Thus, whereas

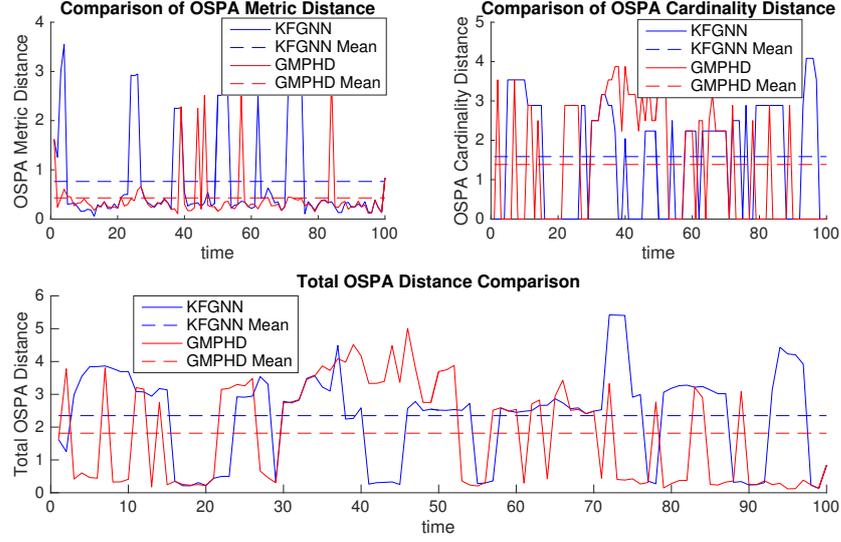


Figure 8.4. OSPA Distance Comparison for Filters Over Time: Upper pair of figures represent the localization and cardinality components of OSPA distance and the below one is the total OSPA distance.

changes in false alarm intensity does not effect the data association techniques significantly, the performance of the GMPHD filter degrades as shown in Figure 8.5.

Second parameter that we investigate in this subsection is the detection probability of the radar. As expected, the performance of both of the filters improves as P_D increases, but in Figure 8.6 we notice that better detection ratios contributes GMPHD filter more than KFGNN filter.

8.2.2.2. Target Intensity Parameters: P_B and P_S . In this section we investigate how a variation in the target intensity effects the performance of the filters. Basically, target number variation is effected by two parameters: P_B and P_S . Higher P_B values and lower P_S values will result in more variation on the target number in time. Figures 8.7 and 8.8 depict the behaviors of the filters under different P_B and P_S values. Naturally each existing target take some time for both of the filters to be labelled as a new target, but we see in Figure 8.7 that PHD filter is more capable of rapid detection of new targets.

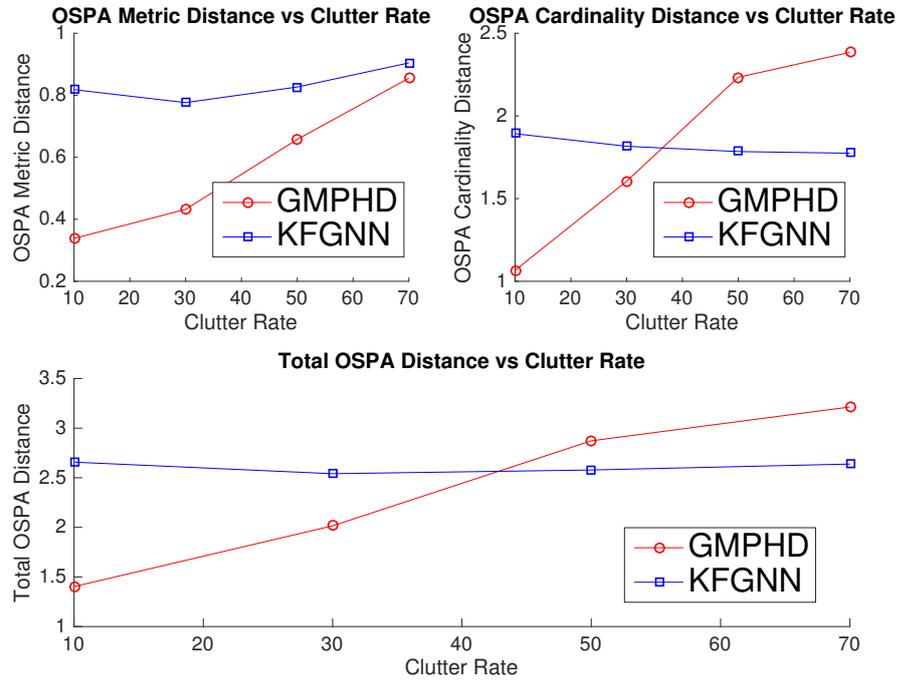


Figure 8.5. OSPA Distance Comparison for Varying λ_{FA} .

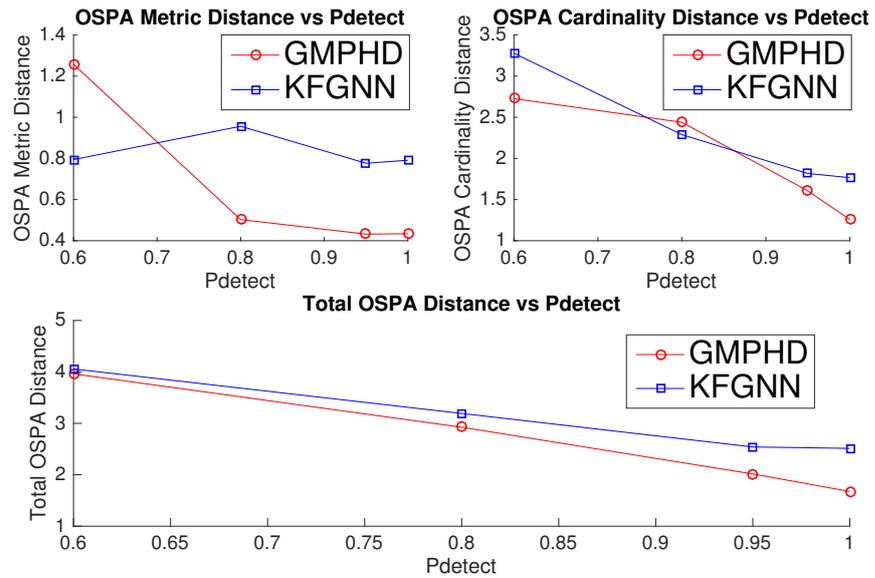


Figure 8.6. OSPA Distance Comparison for Varying P_D .

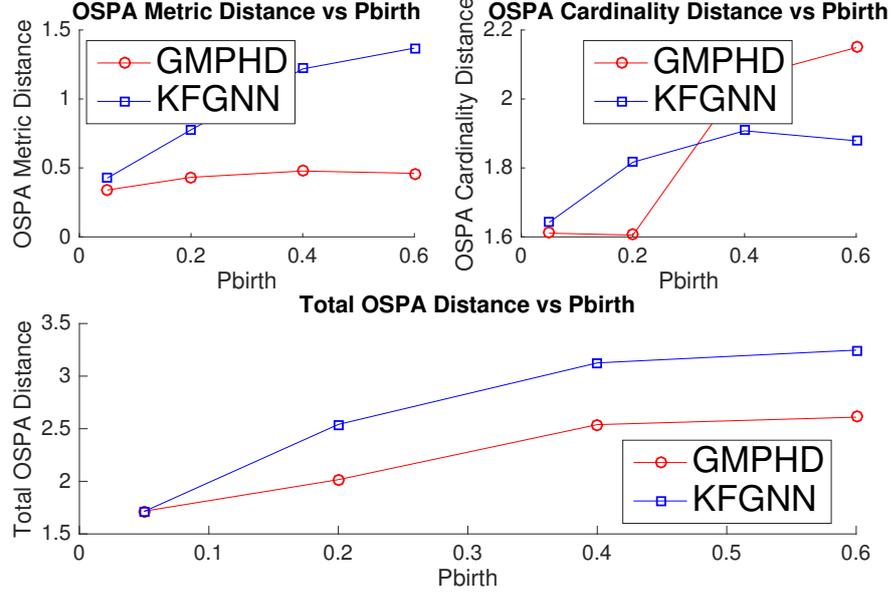


Figure 8.7. OSPA Distance Comparison for Varying P_B .

Moreover in Figure 8.8, it is depicted that if the targets in the scene more tend to disappear (which means that lower P_S), then the performance of KFGNN degrades. On the other hand, PHD filter is not significantly effected by P_S parameter. As a result of these two paramater investigation, we can say that the variation of the target number effects KFGNN filter more than GMPHD filter.

8.2.3. Linear Regression Analysis

In previous section, we analyzed the effects of the four parameters by fixing the values of all other parameters. In this section we are trying to understand whether the results that we derived in previous section are valid if all parameters are free (of course each parameter is free only within its limits). Thus we construct the following linear regression model.

$$\bar{e} = x_1.P_B + x_2.P_S + x_3.P_D + x_4.\lambda_{FA} + x_5 \quad (8.3)$$

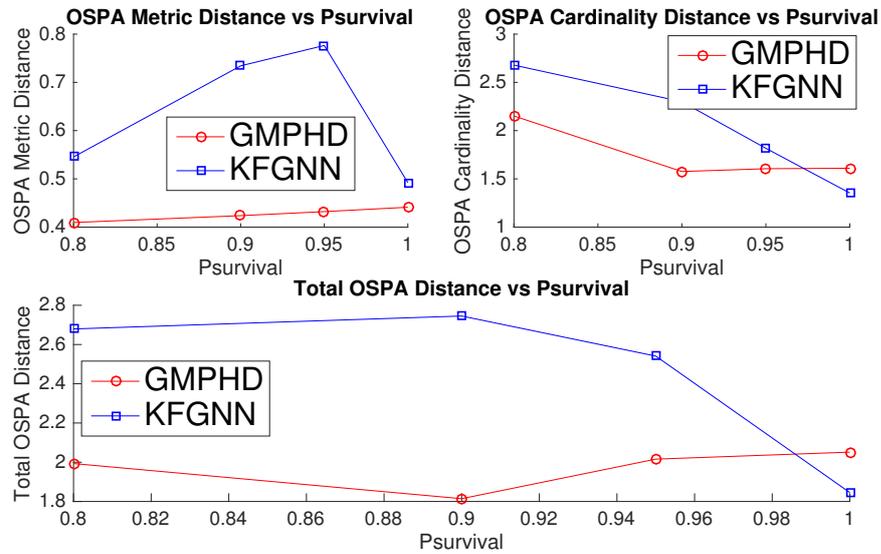


Figure 8.8. OSPA Distance Comparison for Varying P_S .

In order to find out the regression coefficients, we simulate both of the filters with every combination of the values depicted in Table 8.1 (in total $4^4 = 256$ four tuples) and we noted the resulting OSPA distance for the corresponding technique in vector \vec{e} where for each technique \vec{e} is 256×1 vector. Then the problem has the form of solving the following equation by using minimum least squares technique.

$$\vec{e} = \begin{bmatrix} 0.05 & 0.80 & 0.60 & 10 & 1 \\ 0.05 & 0.80 & 0.60 & 30 & 1 \\ 0.05 & 0.80 & 0.60 & 50 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.60 & 1 & 1 & 30 & 1 \\ 0.60 & 1 & 1 & 50 & 1 \\ 0.60 & 1 & 1 & 70 & 1 \end{bmatrix}_{256 \times 5} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

If \vec{x}_{GNN} and \vec{x}_{PHD} are the vectors holding the regression coefficients, resulting values are as follows:

$$\vec{x}_{GNN} = \begin{bmatrix} 2.6228 \\ -3.2747 \\ -4.2097 \\ -0.0002 \\ 8.7823 \end{bmatrix}, \vec{x}_{PHD} = \begin{bmatrix} 0.8113 \\ -0.1328 \\ -5.1112 \\ 0.0223 \\ 6.2050 \end{bmatrix} \quad (8.4)$$

Then we can deduce the following results:

- For P_B ; Since $2.6228 > 0.8113 > 0$, both of the techniques degrades, if P_b increases; but KFGNN degrades more rapidly than GMPHD.
- For P_S ; Since $-3.2747 < -0.1328 < 0$, both of the techniques improves, if P_s increases. On the other hand, the improvement of KFGNN is significant but for GMPHD the improvement is very slight since the value is close to 0. So we can say that GMPHD filter is not effected significantly from P_s .
- For P_D ; Since $-5.1112 < -4.2097 < 0$, both of the techniques improves, if P_D increases; but GMPHD improves more rapidly than KFGNN.
- For λ_{FA} ; Since $0.0223 > -0.0002 \approx 0$, GMPHD degrades when λ_{FA} increases but KFGNN is not effected by the clutter rate parameter. Even if the value 0.0223 seems to close 0, note that the domain of λ_{FA} is not $[0, 1]$ like other parameters, its domain is 10, 30, 50, 70, thus 0.0223 is a sufficiently large value to effect the performance of the filter.

These deductions from the regression analysis match with the results that we depicted on Figures 8.5, 8.6, 8.7 and 8.8 literally.

Since we know that least squares estimation is an approximation with some error, we also provide the 2 - norm of the 256×1 error vectors, err_i , $i = GNN$ or PHD , in

order to show the quality of the estimation as follows:

$$err_i = \vec{e}_i - \begin{bmatrix} 0.05 & 0.80 & 0.60 & 10 & 1 \\ 0.05 & 0.80 & 0.60 & 30 & 1 \\ 0.05 & 0.80 & 0.60 & 50 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.60 & 1 & 1 & 30 & 1 \\ 0.60 & 1 & 1 & 50 & 1 \\ 0.60 & 1 & 1 & 70 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}_i \quad (8.5)$$

$$\|err_{GNN}\| = 5.3975 \quad (8.6)$$

$$\|err_{PHD}\| = 4.4307 \quad (8.7)$$

$$(8.8)$$

$2 - norm$ of the error vectors encapsulates all the error of 256 distinct simulations. So error per simulation is 0.0173 and 0.0211 for GMPHD and KFGNN respectively. These values can be interpreted as small errors and the model is said to be verified by considering these small errors.

Before delving into the nonlinear model case, we provide a table that encapsulates entire results that we derived for the linear case in Table 8.3.

Table 8.3. The Results of the Linear Model Experiments.

	Parameters	KFGNN	GMPHD	Result
ROC Parameters	λ_{FA} Increase	No Effect	Degrade Significantly	Worse receivers degrade PHD filter more rapidly
	P_D Decrease	Degrade	Degrade more	
Target Intensity Parameters	P_B Increase	Degrade Significantly	Slightly Degrade	Variation in target intensity degrades GNN filter more rapidly
	P_S Decrease	Degrade Significantly	No Effect	

8.3. Nonlinear Model

In this section we investigate how nonlinearity effects the performance of the nonlinear extensions of the PHD filter and GNN data association techniques. Rather than analyzing the effect of the parameters, we fix the parameters to values which we have equal OSPA distance showing that their linear expected performance is equal too. The transition and measurement models are represented in Table 8.4. Different from the linear case, the measurement model estimates the noisy distance, that is Euclidean norm, and noisy azimuthal angle, that is a function of arctangent. Moreover, since these values need to be calculated depending upon a radar, we included a radar at $(0, 0)$. In Section 8.3.1, we represent the realization of each technique on the same data set and depict the effect of the nonlinearity by comparing the average OSPA distances of each technique. Similar to the linear case we run 30 Monte Carlo simulations, scenarios consist of 100 time intervals and assumed that initially there are five targets present that are being tracked by the filter.

Table 8.4. State Transition and Measurement Models of the Nonlinear Scenario.

Transition Model	Measurement Model
$x_t^i = Ax_{t-1}^i + w_{t-1}$,	$y_t^i = \begin{cases} h(x_t^i) + v_t & , \text{ if detection occurs} \\ \emptyset & , \text{ if no detection} \end{cases}$
$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$h(x_t^i) = \begin{pmatrix} \left\ \begin{bmatrix} x_{t,1}^i & x_{t,3}^i \end{bmatrix}^T - RadarLoc \right\ \\ \arctan\left(\frac{x_{t,3}^i - RadarLoc_2}{x_{t,1}^i - RadarLoc_1}\right) \end{pmatrix}$
$w_{t-1} \sim \mathcal{N}(0, Q), Q = \begin{pmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{pmatrix}$	$v_t \sim \mathcal{N}(0, R), R = \begin{pmatrix} 10 & 0 \\ 0 & \frac{\pi^2}{180} \end{pmatrix}, RadarLoc = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$
$x_0^i = [X_0, V_0, X_0, V_0]^T$	$ c_t \sim PO(\lambda_{FA}), \lambda_{FA} = 42$
$X_0 \sim U(-500, 500), V_0 \sim U(-5, 5)$	$c_t^i \sim \begin{pmatrix} U(0, RadarRange) \\ U(0, 2\pi) \end{pmatrix}, RadarRange = 1000$
$p_b = 0.2, p_S = 0.95$	$p_D = 0.95$
$x_t = \bigcup_{k=1}^{ \text{tracks} } x_t^k$	$y_t = \bigcup_{k=1}^{ \text{obs} } y_t^k \bigcup_{k=1}^{ \text{c}_t } c_t^k$

8.3.1. Realization of the Filters

The filters that we take into account as a RFS based technique are EKFPHD, UKFPHD and ISMCPHD, on the other hand as a data association method we use EKFGNN for comparison purposes. You may see the superimposed measurements generated according to the measurement model of Table 8.4 in Figure 8.9. Results of the EKFPHD, UKFPHD, ISMCPHD filter and EKFGNN filter are depicted in Figure 8.10, Figure 8.11, Figure 8.12 and Figure 8.13 respectively. For ISMCPHD, we use $\eta = 800$ and $\rho = 200$ as the number of particles per persisting target and number of particles for investigating a measurement if it ensues a birth in our experiments. Also the the threshold of the RFS based techniques for labeling a target is set to 0.95. In order to depict the performance of each implementation technique, we provide Figure 8.14. For this specific experiment we observe that nonlinearity effects the Kalman filter based implementation methods more than Data Association techniques. However ISMCPHD performs better than other Kalman filter based methods and even as good as EKFGNN.

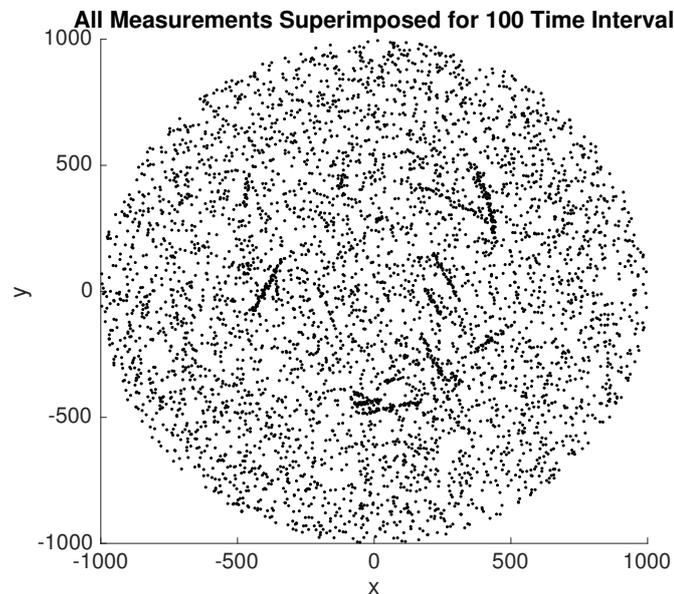


Figure 8.9. The Measurements Generated According to the Model in Table 8.4.

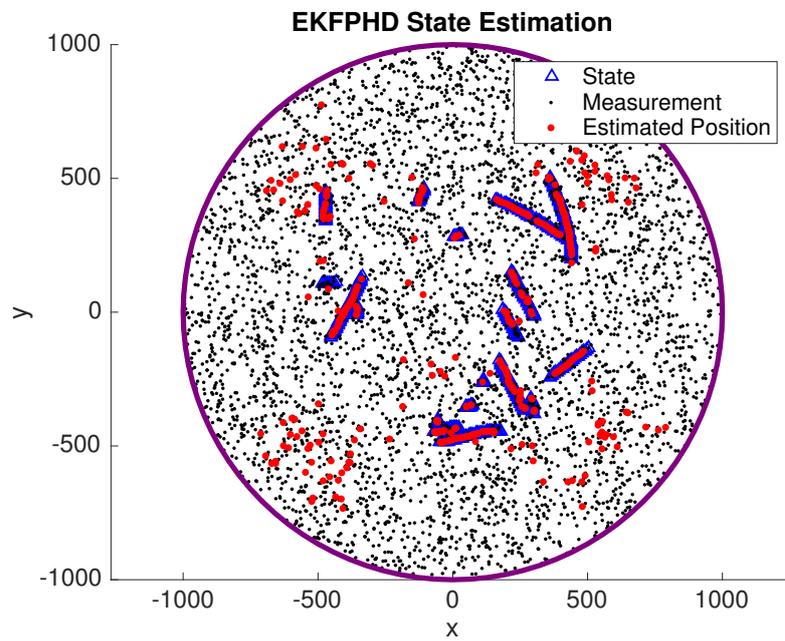


Figure 8.10. EKFPHD Filtering of the Measurements in Figure 8.9.

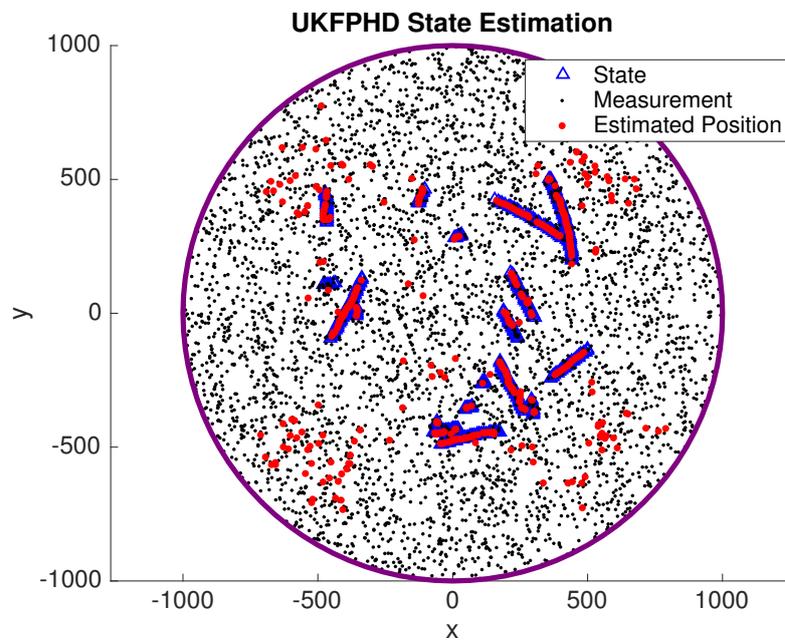


Figure 8.11. UKFPHD Filtering of the Measurements in Figure 8.9.

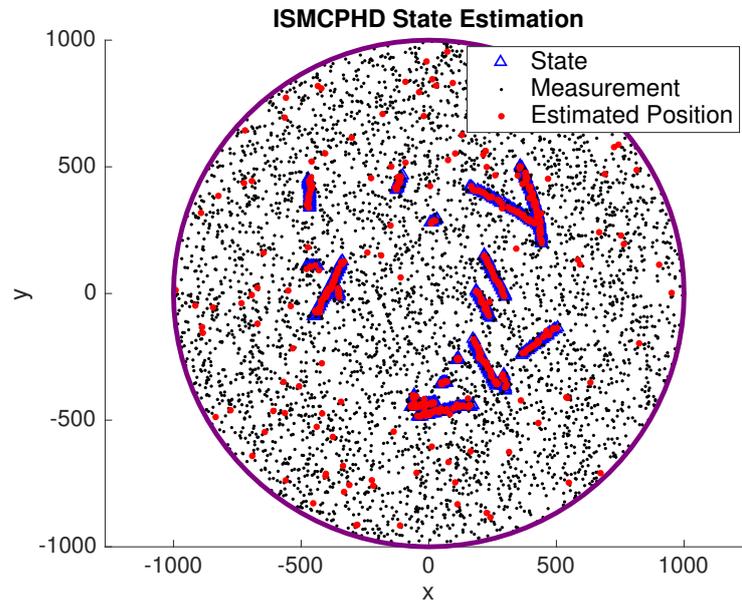


Figure 8.12. ISMCPHD Filtering of the Measurements in Figure 8.9.

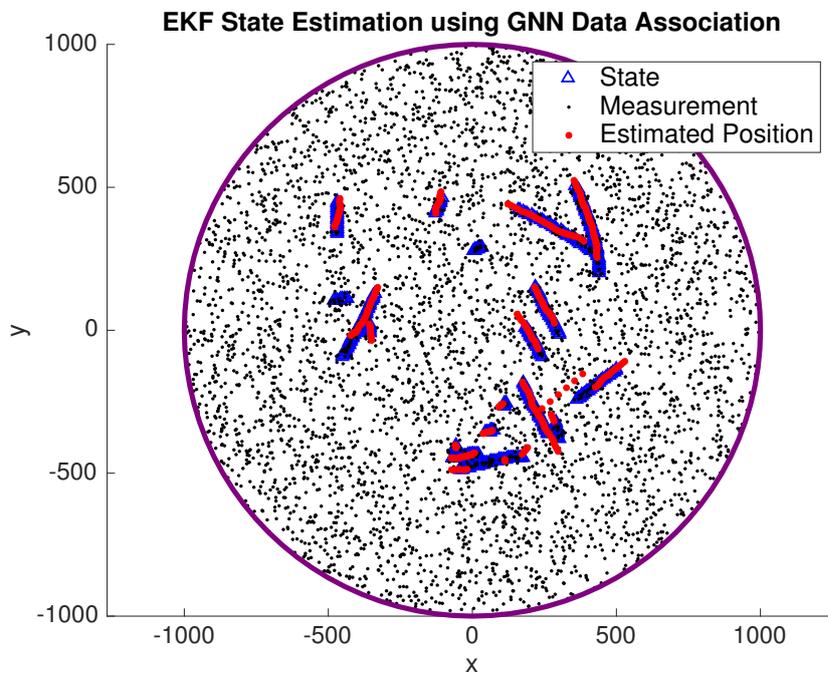


Figure 8.13. EKFGNN Filtering of the Measurements in Figure 8.9.

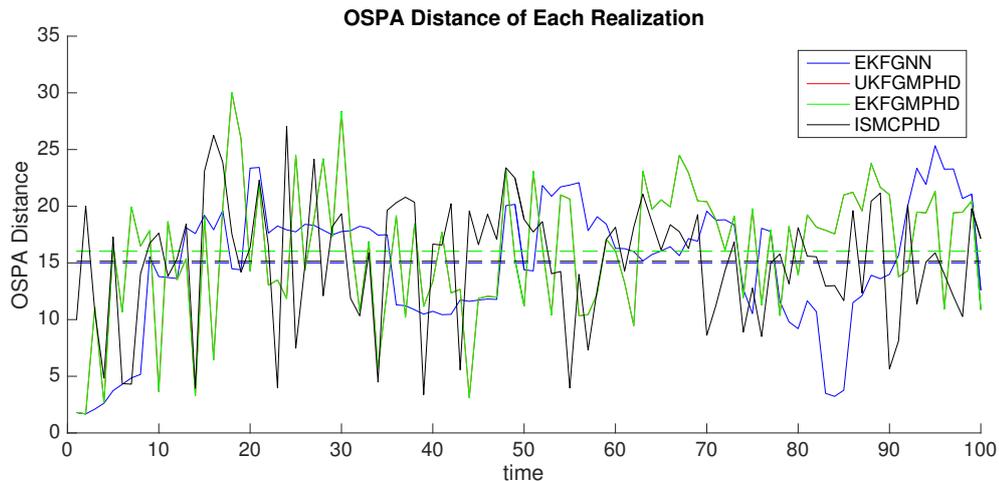


Figure 8.14. OSPA Distance Comparison of Realization Techniques for Nonlinear Implementations Over Time: Dashed lines represent the average of 100 OSPA distance values.

There is one more point we address in this section about ISMCPHD. It is known that as a result of central limit theorem and law of large numbers, the more particles generated, the better the performance of ISMCPHD will be. So, let's consider setting the $\rho = 500$ and $\eta = 2000$ to improve the performance of ISMCPHD. The resulting OSPA distance compared to GNN is depicted in Figure 8.15. Compared to 15, OSPA distance of ISMCPHD decreases to 14.1 when we use more particles and this makes it the most precise algorithm among all. Therefore we can conclude that if we have enough computation that allows us to propagate the particles in real time, ISMCPHD performs better than all the other realization techniques and data association techniques.

8.3.2. Effect of the Nonlinearity

While the previous section describe each of realization technique, in this section we provide a general comparative analysis with 30 MC simulations of each technique. In Figure 8.16 we see that all of the RFS based techniques performs worse than EKFGNN. However from the previous section we know that SMC based implementation of PHD filter can be improved by allocating more particles. Lastly we also observe that SMC based and KF based implementation methods have similar

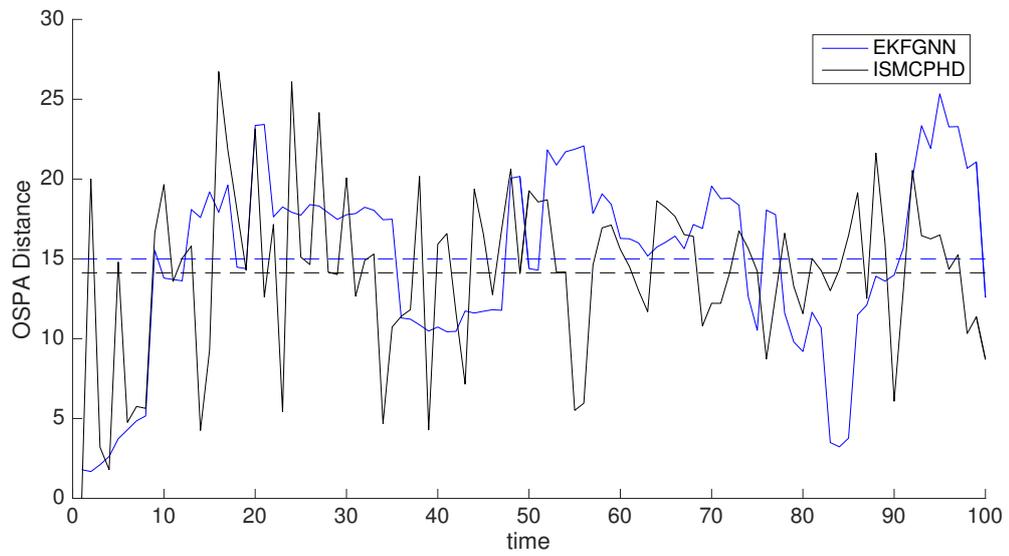


Figure 8.15. Effect of Using More Particles on ISMCPHD: Using more particles on ISMCPHD improves the performance by decrementing OSPA distance.

patterns on the same measurement set, on the other hand the EKFGNN can be distinguished easily.

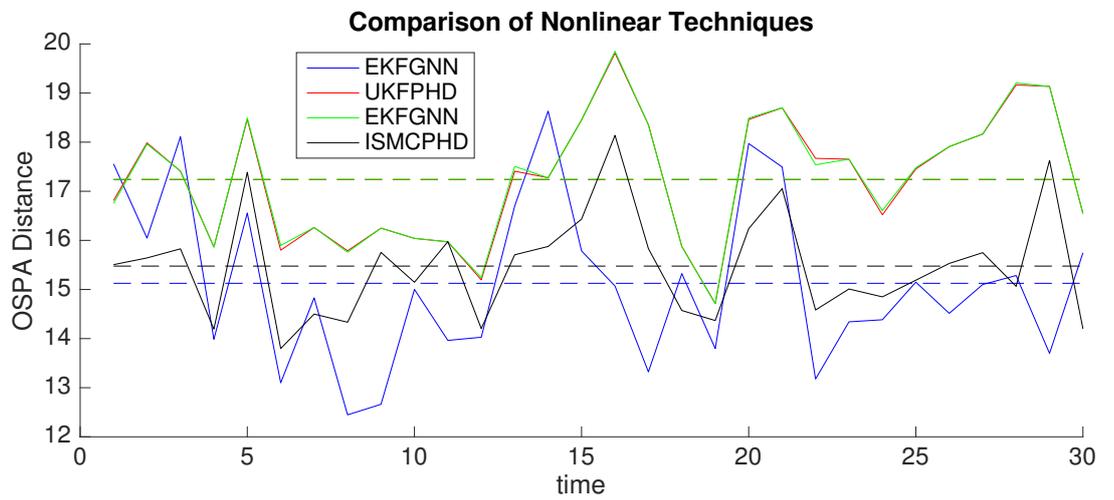


Figure 8.16. Effect of Nonlinearity on the Filters.

9. CONCLUSION AND FUTURE WORK

In this thesis, after we explained the basic notions of target tracking and data association based multitarget tracking methods, we provided an RFS based method, probability hypothesis density filter, by considering its all of the implementation techniques. Then we performed experiments in order to find out how PHD filter is effected from the receiver characteristics, variation in target number and nonlinearity compared to a data association technique, particularly global nearest neighbor. The result that we have is if signal to noise ratio is high enough, namely clutter rate is low and detection probability is high, PHD filter has better performance. On the other hand the performance of the PHD filter degrades significantly when clutter rate increases but GNN is not effected from this parameter. For the variation of target number, we observed that PHD filter has better performance and is not effected significantly compared to GNN technique. Lastly, for the nonlinearity analysis we observed that nonlinearity effects KF based realization techniques better, while ISMCPHD method may result in better performance than GNN technique if we have such a high computational power that can propagate a huge number of particles in real time.

For further research topics, although the total OSPA distance is suitable for multitarget tracking, cardinality and localization error components may be improved by including the extracted states that are out of the cut distance into the cardinality error, but not in localization error component. Therefore, localization error can represent the precision of the filter and cardinality error may be a metric for measuring the false alarm output rate of the filter. Secondly, as we observed in the experiments, RFS based realization methods have high false alarm rate that are scattered all around the radar scope but they can detect almost every target in the scene. Thus if we suggest a scheme to sweep out the scattered false alarms by imposing a gating strategy, then they tend to have better performances. But if we impose a gating strategy then we again need some rules for target birth and target death as in data association techniques. Luckily, RFS based techniques can decide target deaths inherently by considering the decrease

in the PHD function. On the other hand, for target births we can use the particle approximation strategy of the ISMCPHD. Since we just need particles for detecting the target births, then the computational power we need will not be so high as the original ISMCPHD. So, in order to sweep the false alarms we will try to combine the three different methodology. Thirdly, nowadays the most popular RFS based methodology has become the Multi-Bernoulli (MeMBer) filter that can approximate directly to multitarget filtering density, not the first moment as in the PHD filter case. Therefore, we plan to lead our research into MeMBer filter.

REFERENCES

1. *British Patent issued to Christian Hülsmeier entitled Hertzian-wave Projecting and Receiving Apparatus Adapted to Indicate or Give Warning of the Presence of a Metallic Body, Such as a Ship or a Train, in the Line of Projection of Such Waves*, September 1904.
2. Watson-Watt, S. R., *Three Steps to Victory*, Odhams Press, Ltd., London, 1957.
3. Sklansky, J., “Optimizing the Dynamic Parameter of a Track while-Scan System”, *RCA Rev.*, Vol. 18, pp. 163–185, 1957.
4. Kuhn, H. W., “The Hungarian Method for the assignment problem”, *Naval Research Logistics Quarterly*, Vol. 2, pp. 83–97, 1955.
5. Welch, G. and G. Bishop, *An Introduction to Kalman Filter*, ACM, Inc, 2001.
6. Reid, D. B., “An Algorithm for Tracking Multiple Targets”, *IEEE Trans. on Automatic Control*, Vol. 26, No. 6, pp. 843–854, December 1979.
7. Metropolis, N. and S. Ulam, “The Monte Carlo Method”, *J. Am. Statistical Association*, Vol. 44, No. 247, pp. 335–341, 1949.
8. Gordon, N. J., D. J. Salmond and A. F. M. Smith, “Novel approach to non-linear/non-Gaussian Bayesian state estimation”, *Proc. Inst. Elect. Eng.*, Vol. 140, No. 2, pp. 107–113, 1993.
9. Arulampalam, M. S., S. Maskell, N. Gordon and T. Clapp, “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking”, *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, pp. 174–188, 2002.
10. Mahler, R. P., *Statistical Multisource Multitarget Information Fusion*, Artech

House, 2007.

11. Daley, D. and D. Vera-Jones, *An Introduction to the Theory of Point Processes*, Springer, 1988.
12. Mahler, R. P., ““Statistics 102” for Multisource-Multitarget Detection and Tracking”, *IEEE Journal on Selected Topics in Signal Processing*, Vol. 7, No. 3, pp. 376–389, June 2013.
13. Mahler, R. P., “Multitarget Bayes Filtering via First-Order Multitarget Moments”, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 39, No. 4, pp. 1152–1178, October 2003.
14. Mahler, R. P., “PHD filters of higher order in target number”, *Aerospace and Electronic Systems, IEEE Transactions on*, Vol. 43, No. 4, pp. 1523–1543, October 2007.
15. Vo, B.-N. and S. Singh, “Sequential Monte Carlo Implementation of the PHD Filter for Multi-target Tracking”, *In Proceedings of the Sixth International Conference on Information Fusion*, pp. 792–799, 2003.
16. Ristic, B., D. Clark and B.-N. Vo, “Improved SMC implementation of the PHD filter”, *Information Fusion (FUSION), 2010 13th Conference on*, pp. 1–8, July 2010.
17. Vo, B.-N. and W.-K. Ma, “A closed-form solution for the probability hypothesis density filter”, *Information Fusion, 2005 8th International Conference on*, Vol. 2, 2005.
18. Vo, B.-N. and W.-K. Ma, “The Gaussian Mixture Probability Hypothesis Density Filter”, *Signal Processing, IEEE Transactions on*, Vol. 54, No. 11, pp. 4091–4104, Nov 2006.

19. Mark A. Richards, W. A. H., James A. Scheer, *Principles of Modern Radar: Basic Principles*, Scitech Publishing, Inc., 2010.
20. Skolnik, M. I., *Introduction to Radar Systems*, McGraw-Hill Inc., 1981.
21. Skolnik, M. I. (Editor), *Radar Handbook*, McGraw-Hill Companies, third edn., 2008.
22. “IEEE Standard Letter Designations for Radar-Frequency Bands”, *IEEE Std 521-2002 (Revision of IEEE Std 521-1984)*, pp. 1–3, 2003.
23. Electronic Communications Committee within the Conference of Postal and Telecommunications Administrations, *The European Table of Frequency Allocations and Applications in the Frequency Range 8,3 kHz to 3000 GHz (ECA Table)*, May 2015.
24. Trees, H., *Detection Estimation and Modulation Theory Part 1*, John Wiley Sons, Inc, 2001.
25. Bar-Shalom, Y. and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, 1995.
26. Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer Science Business Media, LLC, 2006.
27. Barber, D., *Bayesian Reasoning and Machine Learning*, 2013.
28. Ghahramani, Z., “An Introduction to Hidden Markov Models and Bayesian Networks”, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 15, No. 1, pp. 9–42, 2002.
29. Gad, A., M. Farooq, J. Serdula and D. Peters, “Multitarget Tracking in a Multisensor Multiplatform Environment”, *The 7th international Conference on Information Fusion*, 2004.

30. Bar-Shalom, Y. and T. Fortmann, *Tracking and Data Association*, Academic Press, 1988.
31. Sarkka, S., *Bayesian Filtering and Smoothing*, Cambridge University Press, 2013.
32. Chen, Z., “Bayesian filtering: From Kalman filters to Particle Filters, and Beyond”, *Statistics*, Vol. 182, 2003.
33. Ho, Y. C. and R. C. K. Lee, “A Bayesian approach to problems in stochastic estimation and control”, *IEEE Trans. Automat. Contr.*, Vol. 9, pp. 333–339, 1964.
34. Singh, J. P., “Evolution of the Radar Target Tracking Algorithms: A Move Towards Knowledge Based Multi-Sensor Adaptive Processing”, *Computational Advances in Multi-Sensor Adaptive Processing*, 2005.
35. C.K.Chui and G.Chen, *Kalman Filtering with Real Time Applications*, Springer-Verlag Berlin Heidelberg, 2009.
36. Kalman, R. E., “A New Approach to Linear Filtering and Prediction Problems”, *Transactions of the ASME–Journal of Basic Engineering*, Vol. 82, pp. 35–45, 1960.
37. Pizzinga, A., *Restricted Kalman Filtering: Theory, Methods, and Application*, Springer, 2012.
38. Anderson, B. D. and J. B. Moore, *Optimal Filtering*, Prentice Hall, 1979.
39. Bar-Shalom, Y., X. R. Li and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley and Sons, Inc., 2001.
40. Maybeck, P. S., *Stochastic Models, Estimation and Control*, Marcel Dekker, 1979.
41. Ramachandra, K., *Kalman Filtering Techniques for Radar Tracking*, Marcel Dekker, 2000.

42. Julier, S. J. and J. K. Uhlmann, “A New Extension of the Kalman filter to Nonlinear Systems”, *SPIE Proceedings, Signal Processing, Sensor Fusion, and Target Recognition*, Vol. 3068, 1997.
43. Julier, S. J. and J. K. Uhlmann, “Unscented Filtering and Nonlinear Estimation”, *Proceedings of the IEEE*, Vol. 92, No. 3, 2004.
44. Uhlmann, J. K., “Simultaneous map building and localization for real time applications”, transfer thesis, Univ. Oxford, 1994.
45. Terejanu, G., *Unscented Kalman Filter Tutorial*, Web-tutorial, Department of Computer Science and Engineering, University at Buffalo, Buffalo, New York, 2008.
46. Julier, S. J., “Scaled Unscented Transformation”, *American Control Conference*, Vol. 6, No. 4555–4559, 2002.
47. Cemgil, A. T., *A Tutorial Introduction to Monte Carlo methods, Markov Chain Monte Carlo and Particle Filtering*, Elsevier Major Reference Works, 2012.
48. “AN/MPQ-64 Sentinel”, <https://www.militaryperiscope.com/mdb-smpl/weapons/sensors/grdradar/w0006331.shtml>, accessed at June 2016.
49. Blackman, S. and R. Popoli, *Design and Analysis of Modern Tracking Algorithms*, 1999.
50. Rummond, O., D. Castanon and M. Bellovin, “Comparison of 2-D Assignment Algorithms for Sparse, Rectangular, Floating Point, Cost Matrices”, *Journal of the SDI Panels on Tracking*, Vol. 4, pp. 4–81–4–97, 1990.
51. Blackman, S., “Multiple Hypothesis Tracking For Multiple Target Tracking”, *Aerospace and Electronic Systems*, Vol. 19, No. 1, 2004.
52. Singh, S. S., B. Vo, A. Baddeley and S. Zuyev, “Filters For Spatial Point Processes”, *SIAM Journal on Control and Optimization*, Vol. 48, p. 2275–2295,

2009.

53. Mahler, R. P., “Statistics 101” for Multisource, Multitarget Data Fusion”, *IEEE AE Systems Magazine*, Vol. 19, No. 1, pp. 53–64, January 2004.
54. Grimmett, G. and D. Welsh, *Probability: An Introduction*, Oxford University Press, 2 edn., 2014.
55. “Functional (mathematics)”, [https://en.wikipedia.org/wiki/Functional_\(mathematics\)](https://en.wikipedia.org/wiki/Functional_(mathematics)), accessed at June 2016.
56. Mahler, R. P., “A Theoretical Foundation for the Stein-Winter Probability Hypothesis Density (PHD) Multitarget Tracking Approach”, *In the Proceedings of the 2000 MSS National Symposium on Sensor and Data Fusion*, Vol. 1, pp. 99–117, June 2000.
57. Goodman, I., R. P. Mahler and H. Nguyen, *Mathematics of Data Fusion*, Boston: Kluwer Academic Publishers, 1997.
58. Erdinc, O., P. Willett and Y. Bar-Shalom, “Probability Hypothesis Density Filter for Multitarget Multisensor Tracking”, *7th International Conference on Information Fusion*, pp. 146–153, 2005.
59. Mahler, R. P., “In Proceedings of the Workshop on Estimation, Tracking, and Fusion: A Tribute to Y. Bar-Shalom”, *7th International Conference on Information Fusion*, pp. 134–166, 2001.
60. Zajic, T. and R. Mahler, “A Particle Systems Implementation of the PHD Multitarget Tracking Filter”, *Signal Processing, Sensor Fusion and Target Recognition*, Vol. 5096, pp. 291–299, 2003.
61. Sidenbladh, H., “Multi-target particle filtering for the Probability Hypothesis Density”, *Proc. Int’l Conf. on Information Fusion*, pp. 800–806, 2003.

62. Whiteley, N., S. Singh and S. Godsill, “Auxiliary Particle Implementation of Probability Hypothesis Density Filter”, *Aerospace and Electronic Systems, IEEE Transactions on*, Vol. 46, No. 3, pp. 1437–1454, July 2010.
63. Baser, E. and M. Efe, “A novel auxiliary particle PHD filter”, *Information Fusion (FUSION), 2012 15th International Conference on*, pp. 165–172, July 2012.
64. Schuhmacher, D., B. Vo and B.-N. Vo, “A Consistent Metric for Performance Evaluation of Multi-Object Filters”, *IEEE Transactions on Signal Processing*, Vol. 56, No. 8, pp. 3447–3457, 2008.