RANGE PARAMETERIZED BEARINGS-ONLY TRACKING USING PARTICLE FILTER

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ALİ ERKİN ARSLAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2012

Approval of the thesis:

RANGE PARAMETERIZED BEARINGS-ONLY TRACKING USING PARTICLE FILTER

submitted by **ALİ ERKİN ARSLAN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Department of Electrical and Electronics Engineering**, **Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Mübeccel Demirekler _____
Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Erol Kocaoğlan _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Mübeccel Demirekler _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Kemal Leblebicioğlu _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Ömer Morgül _____
Electrical and Electronics Engineering Dept., Bilkent University

Prof. Dr. Mustafa Kuzuoğlu _____
Electrical and Electronics Engineering Dept., METU

Date: **14.09.2012**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Ali Erkin Arslan

Signature :

ABSTRACT

RANGE PARAMETERIZED BEARINGS-ONLY TRACKING USING PARTICLE FILTER

Arslan, Ali Erkin

Ph.D., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Mübeccel Demirekler

September 2012, 136 pages

In this study, accurate target tracking for bearings-only tracking problem is investigated. A new tracking filter for this nonlinear problem is designed where both range parameterization and Rao-Blackwellized (marginalized) particle filtering techniques are used in a Gaussian mixture formulation to track both constant velocity and maneuvering targets. The idea of using target turn rate in the state equation in such a way that marginalization is possible is elaborated. Addition to nonlinear nature, unobservability is a major problem of bearings-only tracking. Observer trajectory generation to increase the observability of the bearings-only tracking problem is studied. Novel formulation of observability measures based on mutual information between the state and the measurement sequences are derived for the problem. These measures are used as objective functions to improve observability. Based on the results obtained better understanding of the required observer trajectory for accurate bearings-only target tracking is developed.

Keywords: bearings-only tracking, nonlinear systems, passive ranging, target motion analysis, range parameterization, particle filters, observability, observer trajectory.

ÖZ

PARÇACIK FİLTRESİ KULLANILARAK MESAFE PARAMETRELEŞTİRMELİ AÇIYA DAYALI İZLEME

Arslan, Ali Erkin

Doktora, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Mübeccel Demirekler

Eylül 2012, 136 sayfa

Bu çalışmada açıya dayalı hedef izleme problemi için hassas hedef izleme yöntemleri araştırılmıştır. Doğrusal olmayan bu problem için mesafe parametreleştirilmesi ve marjinalize parçacık filtrelemesinin birleştirilerek Gauss dağılımı karışımında kullanıldığı, hem sabit hızlı hem de manevra yapan hedefleri izleyebilen yeni bir filtre tasarlanmıştır. Bu amaçla hedef dönüş oranı, durum denklemine marjinalizasyonu mümkün kılacak şekilde eklenmiştir. Açıya dayalı hedef izleme probleminin doğrusal olmamasının yanı sıra gözlenebilirlik sorunu da vardır. Gözlemcinin gezingesini, açıya dayalı hedef izlemede gözlenebilirliği artıracak şekilde düzenleme üzerine çalışılmıştır. Bu problem için, durum vektörü ve ölçüm dizilerinin karşılıklı bilgi miktarı kullanılarak yeni gözlenebilirlik ölçütleri geliştirilmiştir. Bu ölçütler gözlenebilirliği eniyileme için amaç fonksiyonu olarak kullanılmışlardır. Elde edilen sonuçlar ile açıya dayalı hedef izleme hassasiyetini artırmak için gereken gözlemci gezingesi hakkında daha gelişmiş bir görüş oluşturulmuştur.

Anahtar Kelimeler: açıya dayalı izleme, doğrusal olmayan sistemler, edilgen mesafe ölçme, hedef hareket analizi, mesafe parametreleştirilmesi, parçacık filtreleri, gözlenebilirlik, gözlemci gezingesi.

ACKNOWLEDGMENTS

Foremost, I am grateful to my advisor to Prof. Dr. Mübeccel Demirekler for her guidance, encouragement, and kindness, without which the completion of this work would not have been possible.

Also, I would like to thank my mother, my sister and my father who were concerned about me through all the stages of this thesis. I would like to express my gratitude to my colleagues, my managers and my company for their understanding and concern. I thank my friends Örsan Aytekin, Eren Balcı and Ergin Yalçın for their support and friendship. My sincere thanks also go to Furuzan Atay and Emre Tuna for their friendship and proof-reading.

Finally, I need to thank Caner Eler for his extraordinary commentary about life, culture and almost everything under the guise of the TDF broadcast which put me in a good mood while I was heavily involved with the thesis.

TABLE OF CONTENTS

ABSTRACT iv
ÖZ v
ACKNOWLEDGMENTS vi
TABLE OF CONTENTS vii
LIST OF TABLES x
LIST OF FIGURES xi
LIST OF ABBREVIATIONS xiii
CHAPTERS
1. INTRODUCTION AND LITERATURE SURVEY 1
1.1 Motivation of the Study 5
1.2 A Short Summary of the Thesis 5
1.3 Contributions of the Study 7
1.4 Outline of the Thesis 8
2. NONLINEAR FILTERING 10
2.1 Optimal Bayesian Solution 10
2.2 Extended Kalman Filter 11
2.3 Gaussian Sum Filter 13
2.4 Particle Filter 16
2.4.1 Particle Filter Derivation and Sequential Importance Resampling Filter 16
2.4.2 Rao-Blackwellized (Marginalized) Particle Filters 21
3. BEARINGS-ONLY TRACKING 25
3.1 Formulation for the BOT Problem 25
3.2 Unobservability in Bearings-only Tracking 29
3.3 Models of Maneuvering Targets 32
3.4 Range Parameterized EKF 33
3.4.1 Range Parameterization Procedure 35
3.5 IMM for BOT of Maneuvering Targets 37
3.5.1 Review for Interactive Multiple Model Based BOT 37
3.5.2 IMM Formulation 38

4. MUTUAL INFORMATION FOR STOCHASTIC SYSTEMS 42
4.1 Entropy 42
4.2 Mutual Information 43
4.3 Mutual Information between the State and the Observation Sequences for Linear Gaussian Systems 43
4.4 The Mutual Information between the State and the Measurement Sequences for Nonlinear Stochastic Systems 46
5. RANGE PARAMETERIZED INTERACTING MULTIPLE MODEL GAUSSIAN MIXTURE FILTER 48
5.1 The Problem and the Solution Methodology 48
5.2 Implementation 51
5.3 One Cycle of the Filter 54
5.4 Simulations and Comments 59
6. RANGE PARAMETERIZED COORDINATED TURN RATE MARGINALIZED PARTICLE FILTER 62
6.1 Problem Statement 62
6.2 Tracking Model 63
6.3 One Cycle of the Filter 69
6.4 Simulations 71
6.5 Results and Comments 72
7. OPTIMIZATION OF OBSERVER MOTION BASED ON MUTUAL INFORMATION 76
7.1 One Step Optimization of BOT using Linear System Mutual Information Formulation 76
7.2 The Mutual Information as a Function of the Input for Recursive Formulation 84
7.3 The Gradient of the Measure for Acceleration Input Model 87
7.4 The Gradient of the Measure for Constant Speed Ownship Model 94
7.5 Finite Horizon Input Optimization Based on the Mutual Information for BOT 99
8. IMPLEMENTATION AND PERFORMANCE COMPARISON OF OBSERVER MANEUVER PLANNING APPROACHES BASED ON THE MUTUAL INFORMATION FOR BOT 104
8.1 Ownship Motion Planning for BOT Using Mutual Information Formulation Derived for Linear Systems 106
8.1.1 Algorithm Type 1 106
8.1.2 Algorithm Type 2 111
8.1.3 Algorithm Type 3 113
8.2 Input Optimization Using the Gradient of the Measure M_k 114

8.2.1 Algorithm Type 4 114
8.2.2 Algorithm Type 5 116
8.3 Finite Horizon Input Optimization Based on the Mutual Information for BOT .. 119
8.4 Evaluation of the Results 121
9. CONCLUSIONS 129
REFERENCES 132
VITA 136

LIST OF TABLES

TABLES

Table 1- Monte Carlo runs and the final position errors for Type 2, Type 3 and 124
Table 2- Monte Carlo runs and the final position errors for Type 5. 124

LIST OF FIGURES

FIGURES

Figure 1- 2-D Bearings-only tracking. 25
Figure 2- Unobservable cases: First case: Stationary observer and constant velocity 31
Figure 3- Observable cases: First case: Stationary target and constant velocity observer..... 31
Figure 4- Range interval is divided into sub-intervals. Their uncertainty ellipses define a... 33
Figure 5- The pdf of a Gaussian Mixture Density for range parameterization with 3. 34
Figure 6- Flow of IMM algorithm for a 3 model system. 39
Figure 7- IMM structure of each mixture component..... 50
Figure 8- Gaussian mixture components at time k-1 and range parameterization at time k.. 50
Figure 9- Flow diagram for the RPIMMGM filter..... 53
Figure 10- Comparison of position trajectory estimate (red line) accuracy in one of the..... 59
Figure 11- Average position error of 100 Monte Carlo runs (RPIMMGM left, standard 60
Figure 12- RMS velocity error in north direction for 100 Monte Carlo runs (RPIMMGM . 60
Figure 13- RMS velocity error in east direction for 100 Monte Carlo runs (RPIMMGM 60
Figure 14- Same bearing angle measurements (θ_1 at time t_1 and θ_2 at time t_2) can originate. 63
Figure 15- Trajectories corresponding to different Monte Carlo runs. RPCTMPF is on the. 72
Figure 16- Average position errors for Monte Carlo runs. RPCTMPF is on the left and..... 72
Figure 17- Turn rate particle distribution during tracking for one run of the Monte Carlo... 74
Figure 18- Target position (x), Ownship (*), estimated target position (+) in meters. 107
Figure 19- Zoomed view of Figure 18. Target and ownship are so close to each other as.. 107
Figure 20- The change of the information measure M_k during the track. High values are.. 108
Figure 21- The total measure value reaches up to 150 at the final step. This is the..... 108
Figure 22- Average position error. It is seen that tracking error is extremely low for this.. 109
Figure 23- Ownship accelearion input in x-direction for unconstrained case..... 109
Figure 24- Ownship accelearion input in y-direction for unconstrained case..... 110
Figure 25- Divergence of the track estimation from true target position when EKF is used111
Figure 26- Tracking estimate and the ownship motion for Type 2 algorithm with the 112
Figure 27- Type 2 limited acceleration of 0.02 m/s^2. It is observed that as the input..... 113
Figure 28- Type 4 limited acceleration of 0.005 m/s^2. Target position (x), Ownship (*).... 114
Figure 29- The change of the information measure M_k during the track for Type 4 with ... 115
Figure 30- The increase of the total measure during the track for Type 4 for limited..... 115
Figure 31- The mean position error for Type 4 with limited acceleration of 0.005 m/s^2.... 116
Figure 32- Tracking for Type 5 with a speed constraint of 10 m/s for the ownship..... 117
Figure 33- Tracking for Type 5 with a speed constraint of 10 m/s for ownship when the.. 117
Figure 34- The change of the information measure M_k during the track for Type5 with 118
Figure 35- The total measure value during the track for Type 5 with limited speed of..... 118
Figure 36- The mean position error for Type 5 with limited with limited speed of 10 m/s. 119
Figure 37- A finite horizon optimization example (Type 6) of window legth of 20 steps.. 120

Figure 38- The average position error for Type 6. Since the optimization is performed..... 121
Figure 39- Average position error during tracking for MC simulation 1 where the input... 124
Figure 40- Average position error during track for MC simulation 2 where input limit is.. 125
Figure 41- Average position error during track for MC simulation 3 where input limit is.. 125
Figure 42- Average position error during track for MC simulation 4 where input limit is.. 125
Figure 43 Average position error during track for MC simulation 5 where input limit is... 126
Figure 44- Average position error during track for MC simulation 6 where input limit is.. 126
Figure 45- Average position error during track for MC simulation 7 where ownship 126
Figure 46- Average position error during track for MC simulation 8 where ownship 127
Figure 47- Average position error during track for MC for Type 6 where ownship moves 127
Figure 48- Average position error during track for MC simulation for Type 6..... 128

LIST OF ABBREVIATIONS

BOT	Bearings-Only Tracking.
EKF	Extended Kalman Filter
FIM	Fisher Information Matrix
GM(F)	Gaussian Mixture (Filter)
GS(F)	Gaussian Sum (Filter)
IMM	Interacting Multiple Model Filter
IRST	Infrared Search and Track
KF	Kalman Filter
MMSE	Minimum Mean Square Error
pdf	Probability Density Function.
PF	Particle Filter
RP	Range Parameterization
RPCTMPF	Range Parameterized Coordianted Turn Rate Marginalized Particle Filter
RPIMMGM	Range Parameterized Interacting Multiple Model Gaussian Mixture Filter
SMC	Sequantial Monte Carlo
TMA	Target Motion Analysis
IMM-EKF	Interacting Multiple Model Extended Kalman Filter
MMPF	Multiple Model Particle Filter
UKF	Unscented Kalman Filter
CT	Coordinated Turn
CA	Constant Acceleration
CV	Constant Velocity

CHAPTER 1

INTRODUCTION AND LITERATURE SURVEY

Bearings-only Tracking (BOT) is a passive tracking problem where the target kinematics (position and velocity) is estimated based on the bearing angles of the target line-of-sight relative to the observer. Angle measurement is a passive technique developed either for sonar [1, 2] or electro-optic tracking [10] systems. The passive nature of the measurement is a major advantage for military applications.

In the literature BOT is also named as Target Motion Analysis (TMA), Passive Ranging or Angle-only Tracking. Due to the lack of other range data about the target, the tracking system is supposed to use only the noisy angle measurements and the observer's known kinematic information based on the observer's inherent navigation sensors. Observer is also referred as ownship or owncraft throughout the thesis without any change in its meaning.

The lack of sufficient data usually causes to tracker divergence. Tracking geometry may improve the tracker's performance. If the target has a higher order motion than the observer, range information becomes unobservable [11-14]. For a constant velocity target, the ownship should have maneuvers at least after a period of constant velocity motion (which sometimes referred as legs) to outmaneuver the target [12]. Due to the inherent unobservability of the problem, tracker initialization has a vital importance on the tracking performance. References [3-8, 41] investigate the observability and the ownship maneuvers to minimize the range uncertainty.

Another drawback of the BOT scenario is the nonlinearity of the tracking system. Therefore nonlinear filtering which is usually an approximation of the optimal

Bayesian recursion is necessary. The nonlinearity prevents the use of mature and explicit tracking techniques based on Kalman Filters [11]. Nonlinearity also limits the analysis and Cramer-Rao lower bound calculations [31, 40]. The nonlinear nature of the BOT problem requires application of the nonlinear filtering techniques. Therefore research on the nonlinear filtering techniques is strongly related to the research on the BOT problem [13, 17].

To estimate the state of a dynamical system is the aim of an estimation filter. Estimation of the state means 'finding the conditional pdf of the state given the measurements'. Below we define the state estimation problem for the linear Gaussian case and also for the nonlinear case and compare them.

The discrete time state space formulation of a linear Gaussian dynamical system is

$$\xi_{k+1} = A_k \xi_k + v_k \tag{1.1}$$

$$z_k = C_k \xi_k + w_k \tag{1.2}$$

where A_k is the state transition and C_k is the measurement matrices respectively. Equation (1.1) is the state equation and (1.2) is the measurement equation. Process noise sequence $\{v_k\}$ is an independent and identically distributed (i.i.d) sequence so is the measurement noise sequence $\{w_k\}$. Furthermore these sequences are independent of each other and also independent from the initial state ξ_0. We also assume that all of these random variables are Gaussian. For this case all conditional pdf's of the state or the output also become Gaussian. State estimation reduces to the computation of the mean and the covariance matrix of the conditional pdf of the state. Kalman Filtering gives the optimal state estimate in the mean square sense.

The state pdf becomes non-Gaussian when the state equation and/or the measurement equation is nonlinear or any of the random variables is non-Gaussian. In this case getting the conditional pdf of the current state becomes more difficult and an explicit analytical formulation for the recursive optimal computation of the state pdf is very difficult if not impossible. For the general nonlinear and/or non-Gaussian problems

the system is represented as given in (1.3) and (1.4). Note that for the BOT problem only the measurement equation is nonlinear:

$$x_{k+1} = a_k(x_k, v_k) \tag{1.3}$$

$$z_k = c_k(x_k, w_k)\ . \tag{1.4}$$

When the system is nonlinear but the state space is discrete and contains finite number of states, grid based methods give the optimal recursive solution for state estimation but with high computational cost [17]. When this is not the case suboptimal algorithms like the Extended Kalman Filter (EKF), approximate grid based methods and Particle Filters (PF) are applied [17].

The non-maneuvering target is comparatively less problematic where [1, 2] investigated this problem for sonar application. A reasonable approach for this nonlinear problem is to use the Extended Kalman Filter (EKF). Using Range Parameterization (RP) as in [15, 16], a multiple model (MM) Gaussian mixture (GM) filter formulation is also possible. Another approach is to use Monte Carlo based methods like Particle Filtering (PF) [17, 18].

For the maneuvering targets, observability problem is more traumatic because for this case not only the range but also the target motion model is uncertain. IMM-EKF, MMPF, and Jump Markov System PF have been applied to solve the maneuvering target state estimation problem [10, 39]. Different coordinate systems like Modified Polar Coordinates are also investigated to cope with observability problem [9, 38].

Most of the studies on the optimal observer motion are based on the assumption that the target has a constant velocity without any noise terms. The only noise is the one with the bearing angle measurements [28, 32]. In addition to these assumptions, previous studies have constrained the observer's motion to finite number of legs, and optimized the leg structure [33, 34]. These approaches use optimization methodologies for increasing the accuracy of the estimation by minimizing a cost function based on the Fisher Information Matrix (FIM).

Existing studies basically show that for any observer-target scenario, the observer motion must be selected such that [28]:

1. The observer moves towards the target.
2. The observer maneuvers in order to maximize the bearing angle rate during tracking.

Attempts are made to model the target motion as a finite state Markov chain [29]. In [29], the measurements are discretized and the optimal observer motion is obtained only for a limited case.

Deterministic systems are considered either observable or not based on the Gramian matrix [30]. This notion of complete observability is not suitable for BOT problem where the target has stochastic motion based on the plant noise. Furthermore, bearing measurement has also a noise term usually modeled as Gaussian. It is for sure that a "degree of observability" is a more realistic measure for the stochastic problems, as it is the case for our problem.

In [37] the mutual information between the last state and the measurement sequence is proposed as an observability measure for continuous stochastic systems. The mutual information is the entropy difference between a priori and a posteriori distributions. This measure is a degree of observability instead of a yes/no answer. They also propose a second moment approximation for the nonlinear and non-Gaussian processes where [35] also refers to this approach to find the observer path by maximizing the mutual information. However neither the details of the formulation nor results are available.

In [27] the mutual information between the state and the observation sequences is given in a recursive form for nonlinear systems. It defines the observability measure considering not only the last state but the whole state and the measurement sequences.

1.1 Motivation of the Study

The thesis focuses on increasing the tracking performance of the BOT problem. The problem is interesting for more than one reason. First of all it is a nonlinear estimation problem. Although general methods like EKF or PF exist for the state estimation problem of nonlinear systems they should always be adapted to the particular system under investigation.

Another and even more interesting property of the BOT is its inherent unobservability problem. Observability of stochastic systems is not a mature concept that a consensus is obtained on. A measure instead of a yes/no answer to the observability seems to be much more suitable for stochastic systems. For the BOT problem it is well known that the observability in some sense is possible if the observer makes some maneuvers relative to the target. A high motivation here is to find the best observer path to increase some observability measure.

One final but important motivation is the practical importance of the problem. Passive tracking is attractive especially in military applications because of its ability of "seeing without being seen". Most of the sonar applications requires passive tracking as well as electro-optic systems. Passive tracking is a hot subject also for the radar systems.

1.2 A Short Summary of the Thesis

The subject of the thesis is to improve the accuracy of the bearings-only tracking. This is possible when the nonlinear filtering methods are adapted according to the nature of BOT problem. Additionally, the observer motion during tracking is an important factor for efficient tracking so must be optimized to further increase the tracking performance. These two are the major goals that are examined in the thesis.

Range parameterization (RP) is a technique used for bearings-only tracking problems of non-maneuvering targets. In this study an Interacting Multiple Model (IMM) based approach is introduced with a motivation of widening the use of the range parameterized filter for maneuvering targets as explained in Chapter 5.

The filter given in Chapter 6 is inspired by the idea of using the turn rate value as a state component. The basic solution method is to use particle filtering. However it is observed that the new modeling makes the use of Rao-Blackwellized particle filter possible so that only the turn rate component of the state is handled by particles which reduces the computational complexity of particle filtering.

The mutual information is considered as a measure of the level of observability for nonlinear stochastic systems in [27]. The BOT has an intrinsic observability problem which deteriorates the target tracking accuracy independent of the type of the tracker used. For the BOT problem as the observability increases, the range uncertainty decreases and as a result the tracking accuracy increases. This is the motivating factor to investigate and utilize the mutual information to confront with the observability problem. Since unobservability is the main issue for the BOT problem, we use the mutual information as the objective function that must be maximized for tracking accuracy (Chapter 7). Observability, i.e., the mutual information is increased by the selection of the observer path.

Another motivating factor of using the mutual information as a measure is its stochastic nature. Although there are rules of thumb for the ownship maneuver these are mostly the results obtained for the deterministic state case. The mutual information is a measure that directly relates the measurements and the states in which both are considered as stochastic processes.

The recently proposed representation of the mutual information between the state and the observation sequences and its recursive formulation for nonlinear systems given in [27] inspired the derivation of the methods of Chapter 7. This recursion permits the computation of the observability measures of nonlinear systems. It is observed that the recursive formula derived for observability has two possible interesting uses. One is the one step and the other is the N step finite horizon optimization of the input for the BOT problem.

1.3 Contributions of the Study

Two nonlinear tracking methods are proposed to fulfill the first goal of designing BOT filter algorithms. Both of them use range parameterization (RP). The first one uses Interacting Multiple Model (IMM) in a Gaussian Mixture (GM) formulation. The second one is a totally new approach using particle filtering where a marginalized (Rao-Blackwellized) particle filter (MPF) is applied. The target turn rate is the nonlinear component of MPF. Also GM and range parameterization (RP) is inherent in this filter. The first filter is computationally simple and gives satisfactory results for maneuvering targets. The second one is computationally more expensive but its ability to marginalize state components for applicability of marginalized particle filtering is its appealing advantage.

The observer motion optimization to increase target observability is another issue of this thesis. Mutual information between the target and the measurement sequences is taken as the objective function that should be maximized. Unlike the work that exists in the general literature, the target model of our study is not limited to a zero noise constant velocity case. The target motion model used in this study is the noisy constant velocity model, which is much more realistic. The recursive mutual information formulation of [27] is adapted to BOT input optimization problem. A simplified measure is obtained using the recursive formulation.

Our first attempt of using the observability measure for optimal and/or sub-optimal observer path planning is to optimize the incremental step in the recursive form of the observability measure. We formulate the gradient of the incremental value of the mutual information with respect to the input and use it for one-step maximization of the mutual information both for the constrained and the unconstrained input cases. In this context two different input models are developed and analyzed.

Secondly, the recursive formulation of the mutual information is used for an N step horizon Dynamic Programming method for the optimal input selection problem. This aim is achieved by discretizing the input. The appealing effect of the N step optimization is its better performance compared to one step optimization

Additionally, as an alternative formulation, the mutual information between the state sequence and the measurement sequence for linear systems is investigated. This formula is used for one step maximization of the mutual information for the nonlinear BOT. To do this an approximation for the measurement function of BOT is obtained by computing the Jacobian of the measurement function. A set of algorithms using this idea are developed and their performances are evaluated through simulations. This formulation gives a chance to analyze the motion behavior of the observer to increase the observability of a single step of the BOT problem.

1.4 Outline of the Thesis

The general outline of the thesis is as follows:

Chapter 1 is an introduction for the problem. The related literature is also discussed in this Chapter.

Chapter 2 is a general discussion of mostly used nonlinear filtering techniques. Namely the Extended Kalman Filtering (EKF), the Gaussian Sum/Mixture Filtering (GSF/GMF), and the Particle Filtering (PF) are presented in this Chapter. These filters are the ones used in the thesis to solve the BOT problem.

Detailed description of the BOT problem is given in Chapter 3. Problem formulation, the unobservability phenomenon, different target models for BOT, IMM filters and Range Parameterization (RP) are discussed in this chapter.

Chapter 4 is devoted to a brief explanation of the mutual information between the state and the measurement sequences for stochastic systems. It gives a theoretical background for the ideas and approaches developed in Chapter 7 and 8.

The first algorithm developed, Range-Parameterized Interacting Multiple Model Gaussian Mixture Filter (RPIMMGM), is explained in Chapter 5. The algorithm is developed for maneuvering targets and is based on IMM, RP and GM filtering. Algorithm formulation and the simulation results are given in the same chapter. It should be noted that although we have developed the algorithm for BOT problem,

we have realized that for a similar application a similar formulation had already been developed independently in [25].

The Range Parameterized Coordinated Turn Rate Marginalized Particle Filter (RPCTMPF) which is a novel bearings-only tracking filter is described in Chapter 6. It is a more advanced algorithm compared to RPIMMGM. It is developed for maneuvering targets. In addition to RP and GM, a Rao-Blacwellized PF is used to observe the target maneuver more accurately. Algorithm formulation and the simulation results are given in the same Chapter.

In Chapter 7, the idea of using the mutual information between the measurements and the states for the optimization of the observer motion for BOT and the algorithms developed using this idea are explained in detail. The derivations and the implementation alternatives are presented.

Chapter 8 is for the implementation details and simulation results of the theory given in Chapter 7. Comments and performance comparisons for the implemented algorithms are given in this Chapter.

Chapter 9 is the conclusion for the thesis. Contributions of the study and the results obtained are discussed in this Chapter.

CHAPTER 2

NONLINEAR FILTERING

In this chapter we start by the Optimal Bayesian Solution for non linear state estimation problem. A closed form solution cannot be provided by the Optimal Bayesian Solution. The major approximate filtering approaches for the state estimation of the nonlinear systems are briefly discussed in this Chapter.

2.1 Optimal Bayesian Solution

The nonlinear behavior of the target and the observer is given below as a nonlinear stochastic state space representation:

$$\xi_{k+1} = a_k(\xi_k, v_k) \tag{2.1}$$

$$z_k = c_k(\xi_k, w_k)\,. \tag{2.2}$$

The state sequence of the target $\{\xi_k, k \in N\}$ has dynamic equation (2.1) where $a_k : R^{n_\xi} x R^{n_v} \to R^{n_\xi}$ can be a nonlinear function. Similarly, $c_k : R^{n_\xi} x R^{n_w} \to R^{n_z}$ is possibly a nonlinear function. $\{v_k, k \in N\}$ is an independent identically distributed (i.i.d.) state (process) noise sequence. $\{w_k, k \in N\}$ is an i.i.d. measurement noise sequence. We also assume that $\{w_k, k \in N\}$, $\{v_k, k \in N\}$ and ξ_0 are independent. n_ξ, n_z, n_v and n_w are dimensions of the state, the measurement, the process noise and the measurement noise vectors, respectively.

Tracking is the estimation of the target state recursively for the current time *k* using the measurements $Z^k = \{z_i, i = 1:k\}$.

The pdf of the current state, $p(\xi_k \mid Z^k)$, based on Chapman-Kolmogorov equation is given below both for prediction and filtering.

The prediction equation is

$$p(\xi_{k+1} \mid Z^k) = \int p(\xi_{k+1} \mid \xi_k) p(\xi_k \mid Z^k) d\xi_k. \tag{2.3}$$

Update is applied as below when the new measurement is available

$$p(\xi_{k+1} \mid Z^{k+1}) = \frac{p(z_{k+1} \mid \xi_{k+1}) p(\xi_{k+1} \mid Z^k)}{p(z_{k+1} \mid Z^k)} \tag{2.4}$$

where the denominator is nothing but a normalizing constant given by

$$p(z_{k+1} \mid Z^k) = \int p(z_{k+1} \mid \xi_{k+1}) p(\xi_{k+1} \mid Z^k) d\xi_{k+1}. \tag{2.5}$$

(2.3) and (2.4) form the optimal Bayesian solution. It is not possible to apply this recursion for the analytical evaluation of the posterior density for almost all nonlinear systems. Bayesian solution does not provide a closed form solution for the recursion. Kalman Filter (KF) is the solution of the above recursive expressions for the linear Gaussian systems. For others the techniques like Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF) and Particle Filter (PF) are approximations to the optimal Bayesian recursion.

2.2 Extended Kalman Filter

EKF uses local linearization and Kalman filtering for nonlinear tracking systems. Before the explanation of the EKF, KF equations are given for completeness.

Assume a linear Gaussian system as

$$\xi_{k+1} = A_k \xi_k + v_k \tag{2.6}$$

$$z_k = C_k \xi_k + w_k \tag{2.7}$$

where $\{v_k, k \in N\}$, $\{w_k, k \in N\}$ and ξ_0 are mutually independent zero-mean Gaussian with covariance matrices Q_k, R_k and P_o respectively.

Under these assumptions both $p(\xi_k \mid Z^k)$ and $p(\xi_{k+1} \mid Z^{k+1})$ are Gaussian and the KF recursively calculates the conditional mean, $\xi_{k+1|k+1} = E\{\xi_{k+1} \mid Z^{k+1}\}$, and the covariance matrix, $P_{k+1|k+1} = \mathrm{cov}(\xi_{k+1} \mid Z^{k+1})$, of the state which completely describes the distribution of the Gaussian posterior.

The Kalman Filter equations are as follows

$$\xi_{k+1|k} = A_k \xi_{k|k} \tag{2.8}$$

$$P_{k+1|k} = Q_k + A_k P_{k|k} A_k^T \tag{2.9}$$

$$\xi_{k+1|k+1} = \xi_{k+1|k} + K_{k+1}(z_{k+1} - C_{k+1}\xi_{k+1|k}) \tag{2.10}$$

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} S_{k+1} K_{k+1}^T \tag{2.11}$$

where K_k, the Kalman Gain and S_k, the innovation covariance matrix are given below:

$$S_k = C_k P_{k|k-1} C_k^T + R_k \tag{2.12}$$

$$K_k = P_{k|k-1} C_k^T S_k^{-1}. \tag{2.13}$$

For the general nonlinear case Extended Kalman Filter (EKF) applies a first order linearization to the nonlinear functions. EKF assumes that the process and the measurement noises are Gaussian and the distribution of the conditional state is approximated as Gaussian at each step. The EKF algorithm is given for the nonlinear equations of (2.1) and (2.2) for additive noise case:

$$\xi_{k+1} = a_k(\xi_k) + v_k \tag{2.14}$$

$$z_k = c_k(x_k) + w_k. \tag{2.15}$$

For this case applying local linearization to the functions a_k and c_k the Jacobians at $\xi_{k|k}$ and $\xi_{k+1|k}$, are obtained as the matrices A_k and C_{k+1}.

$$A_k \equiv \left[\nabla_{\xi_k} a_k^T(\xi_k)\right]^T|_{\xi_k=\xi_{k|k}} \tag{2.16}$$

$$C_{k+1} \equiv \left[\nabla_{\xi_{k+1}} c_{k+1}^T(\xi_{k+1})\right]^T|_{\xi_{k+1}=\xi_{k+1|k}} . \tag{2.17}$$

The EKF equations are given as

$$\xi_{k+1|k} = a_k(\xi_{k|k}) \tag{2.18}$$

$$P_{k+1|k} = Q_k + A_k P_{k|k} A_k^T \tag{2.19}$$

$$\xi_{k+1|k+1} = \xi_{k+1|k} + K_{k+1}(z_{k+1} - c_{k+1}(\xi_{k+1|k})) \tag{2.20}$$

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} S_{k+1} K_{k+1}^T \tag{2.21}$$

with

$$S_k = C_k P_{k|k-1} C_k^T + R_k \tag{2.22}$$

$$K_k = P_{k|k-1} C_k^T S_k^{-1}. \tag{2.23}$$

EKF filter estimation is a Gaussian approximation of potentially skewed pdf of the posterior. Therefore if nonlinearity is high, poor filter performance and even divergence is possible.

2.3 Gaussian Sum Filter

If the system is not Linear-Gaussian, Gaussian Sum Filter (GSF) is an alternative approach to EKF and Particle Filter (PF). It removes the error introduced by the Gaussian assumption in EKF and it is computationally simpler than PF. The Gaussian sum representation is formed as the convex combination of the output of

several EKFs operating in parallel. Based on [21] Gaussian Sum Filtering can be briefly described as below:

Let *n*-dimensional state vector ξ_k of the system evolve as

$$\xi_{k+1} = a_k(\xi_k) + v_k \qquad k = 0,1,...,N-1 \tag{2.24}$$

with the measurement equation

$$z_k = c_k(\xi_k) + w_k \qquad k = 0,1,...,N\,. \tag{2.25}$$

Suppose that the prior distribution can be represented as

$$p(\xi_k \mid z^{k-1}) = \sum_{i=1}^{N_{GM}^{'}} \alpha_{ki}^{'} N(\xi_k; m_{ki}, P_{ki}^{'}) \tag{2.26}$$

where m_{ki} is the mean, $N_{GM}^{'}$ is the number of Gaussians for prior, $\alpha_{ki}^{'}$ is the probability weight and $P_{ki}^{'}$ is the prior covariance for the i^{th} Gaussian at time *k*. Let

$$C_{ki} \equiv \frac{\partial}{\partial \xi_k} c_k(\xi_k)\bigg|_{\xi_k = m_{ki}} \tag{2.27}$$

then the measurement update for Gaussian Sum Filter is

$$p(\xi_k \mid z^k) = \sum_{i=1}^{N_{GM}^{'}} \alpha_{ki} N(\xi_k; \mu_{ki}, P_{ki}) \tag{2.28}$$

where

$$\mu_{ki} = m_{ki} + K_{ki}(z_k - c_k(m_{ki})) \tag{2.29}$$

$$P_{ki} = P_{ki}^{'} - K_{ki} C_{ki} P_{ki}^{'} \tag{2.30}$$

$$K_{ki} = P_{ki}^{'} C_{ki}^{T} [C_{ki} P_{ki}^{'} C_{ki}^{T} + R_k]^{-1} \tag{2.31}$$

$$\alpha_{ki} = \frac{\alpha_{ki}^{'}\beta_{ki}}{\sum_{j=1}^{N_{GM}^{'}}\alpha_{kj}^{'}\beta_{kj}} \tag{2.32}$$

$$\beta_{ki} \equiv N(z_k; c_k(m_{ki}), C_{ki}P_{ki}^{'}C_{ki}^{T} + R_k) \tag{2.33}$$

If there is a significant plant noise that cannot be modeled as Gaussian it may be reasonable to represent it as a Gaussian mixture:

$$p(v_k) = \sum_{l=1}^{q_k}\gamma_k N(v_k; \omega_{kl}, Q_{kl}) \tag{2.34}$$

with this assumption the time update is

$$p(\xi_{k+1} \mid z^k) = \sum_{i=1}^{N_{GM,k+1}^{'}}\alpha_{(k+1)i}^{'}N(\xi_{k+1}; m_{(k+1)i}^{'}, P_{(k+1)i}^{'}) \tag{2.35}$$

where

$$N_{GM,(k+1)}^{'} = N_{Gm,k}^{'}q_k \tag{2.36}$$

$$m_{(k+1)i}^{'} = a_k(\mu_{kj}) + \omega_{kl} \tag{2.37}$$

$$P_{(k+1)i}^{'} = A_{(k)i}P_{kj}A_{(k)i}^{T} + Q_{kl} \tag{2.38}$$

$$\alpha_{(k+1)i}^{'} = \alpha_{kj}\gamma_{kl} \tag{2.39}$$

$$A_{(k)i} \equiv \frac{\partial}{\partial\xi_k}a_k(\xi_k)\Big|_{\xi_k=\mu_{ki}} . \tag{2.40}$$

The growth in the number of terms should be reduced in order to use the filter in a practical application. Removing the terms with small weights is advised but care must be taken to apply such a policy. Merging of similar components is also possible.

2.4 Particle Filter

In the case of non-Gaussianity in process and/or measurement noises or a high nonlinearity in the state or observation equation, state estimation is very difficult. Extended Kalman Filters, Gaussian Sum approximations and grid-based filters are possible but poor solutions. Except the last one these approximations are mostly unsatisfactory. Grid based filters are based on deterministic numerical integration and are powerful although they are difficult to implement and computationally expensive [22].

Sequential Monte Carlo Methods (SMC) are simulation based approaches that can be used to estimate the posterior distributions. Compared to grid based methods, SMC methods are easy to implement, flexible, and applicable to many general cases [17, 22]. "*Bootstrap filters, condensation, particle filters, Monte Carlo filters, interacting particle approximations and survival of the fittest"* are the names for the methods based on SMC methods. The main reason for the increasing amount of usage of Monte Carlo integration methods is due to its applicability to nonlinear and non-Gaussian systems with its proven convergence properties. Furthermore constraints for the state components of a real system can be handled easily which could be difficult even for a linear filtering problem with classical linear methods.

2.4.1 Particle Filter Derivation and Sequential Importance Resampling Filter

To define the particle filtering approach firstly the problem is stated as follows

$$\xi_{k+1} = a(\xi_k, v_k) \tag{2.41}$$

$$z_k = c(\xi_k, w_k) \tag{2.42}$$

where $\{v_k, k \in N\}$ and $\{w_k, k \in N\}$ satisfy the usual independency assumptions.

The state vector is desired to be estimated using the observations, $Z^k = \{z_m\}_{m=1}^k$. To do this, both the prior, $p(\xi_k \mid Z^{k-1})$, and the posterior, $p(\xi_k \mid Z^k)$, distributions need to be estimated. Chapman-Kolmogorov equation gives the result as

$$p(\xi_k \mid Z^{k-1}) = \int p(\xi_k \mid \xi_{k-1}) p(\xi_{k-1} \mid Z^{k-1}) d\xi_{k-1} \tag{2.43}$$

$$p(\xi_k \mid Z^k) = \frac{p(z_k \mid \xi_k) p(\xi_k \mid Z^{k-1})}{p(z_k \mid Z^{k-1})} \tag{2.44}$$

where

$$p(z_k \mid Z^{k-1}) = \int p(z_k \mid \xi_k) p(\xi_k \mid Z^{k-1}) d\xi_k \tag{2.45}$$

The posterior pdf, $p(\xi_k \mid Z^k)$ is characterized by $\{\xi_k^i, n_k^i\}_{i=1}^{N_s}$ where $\{\xi_k^i\}_{i=1}^{N_s}$ is the set of support points (particles), $\{n_k^i\}_{i=1}^{N_s}$ is the associated weights and N_s is the number of particles, where

$$\sum_{i=1}^{N_s} n_k^i = 1. \tag{2.46}$$

The posterior density can be approximated using Dirac delta function, δ, as

$$p(\xi_k \mid Z^k) \approx \sum_{i=1}^{N_s} n_k^i \delta(\xi_k - \xi_k^i). \tag{2.47}$$

This is a discrete weighted approximation of the posterior. Furthermore the weights are chosen using the principles of importance sampling [17, 22]. Importance sampling is based on the principle that if drawing samples from *p(x)* is difficult another density *q(x)* may be used alternatively to draw the samples and the weights are assigned as

$$n_i \propto \frac{p(\xi^i)}{q(\xi^i)}. \tag{2.48}$$

q(x) is named as the importance density. The selection of *q(x)* depends on the problem but a practical and mostly used choice is given in this section. First of all to get a sequential method it is assumed that the importance density satisfies

$$q(\Xi^k \mid Z^k) = q(\xi_k \mid \Xi^{k-1}, Z^k) q(\Xi^{k-1} \mid Z^{k-1}) . \tag{2.49}$$

where $\Xi^k = \{\xi_m\}_{m=1}^k$. So samples $\Xi^{k,i} \sim q(\Xi^k \mid Z^k)$ can be obtained by augmenting each of the existing samples $\Xi^{k-1,i} \sim q(\Xi^{k-1} \mid Z^{k-1})$ with the new state samples $\xi_k^i \sim q(\xi_k \mid \Xi^{k-1}, Z^k)$. The weight update equation can be derived as follows

$$p(\Xi^k \mid Z^k) = \frac{p(z_k \mid \Xi^k, Z^{k-1}) p(\Xi^k \mid Z^{k-1})}{p(z_k \mid Z^{k-1})}$$

$$= \frac{p(z_k \mid \Xi^k, Z^{k-1}) p(\xi_k \mid \Xi^{k-1}, Z^{k-1})}{p(z_k \mid Z^{k-1})} p(\Xi^{k-1} \mid Z^{k-1})$$

$$\propto p(z_k \mid \xi_k) p(\xi_k \mid \xi_{k-1}) p(\Xi^{k-1} \mid Z^{k-1}) \tag{2.50}$$

and the weight update becomes

$$n_k^i \propto \frac{p(z_k \mid \xi_k^i) p(\xi_k^i \mid \xi_{k-1}^i) p(\Xi^{k-1,i} \mid Z^{k-1})}{q(\xi_k \mid \Xi^{k-1}, Z^k) q(\Xi^{k-1,i} \mid Z^{k-1})} = n_{k-1}^i \frac{p(z_k \mid \xi_k^i) p(\xi_k^i \mid \xi_{k-1}^i)}{q(\xi_k^i \mid \Xi^{k-1,i}, Z^k)}$$

$$n_k^i \propto n_{k-1}^i \frac{p(z_k \mid \xi_k^i) p(\xi_k^i \mid \xi_{k-1}^i)}{q(\xi_k^i \mid \Xi^{k-1,i}, z_k)}. \tag{2.51}$$

If $q(\xi_k^i \mid \Xi^{k-1,i}, z_k) = q(\xi_k^i \mid \xi_{k-1}^i, z_k)$ then update equation can be simplified further as

$$n_k^i \propto n_{k-1}^i \frac{p(z_k \mid \xi_k^i) p(\xi_k^i \mid \xi_{k-1}^i)}{q(\xi_k^i \mid \xi_{k-1}^i, z_k)}. \tag{2.52}$$

Finally Sequential Importance Sampling (SIS) algorithm is written as

- At time '*k-1*' sample sequences and weights are as

$$\{\xi_0^i, \xi_1^i, \ldots, \xi_{k-1}^i\}_{i=1}^{N_s}, \ \{n_{k-1}^i\}_{i=1}^{N_s}.$$

- At time *'k'* :
  - Define N_s number of pdf's : $q(\xi_k \mid \xi_{k-1}^i, z_k)$

- Draw one sample from each pdf and extend the sample sequences to include the last set of samples as $\{\xi_0^i, \xi_1^i, \cdots \xi_k^i\}_{i=1}^{N_s}$
- Update the weights using

$$n_k^i \propto n_{k-1}^i \frac{p(z_k \mid \xi_k^i) p(\xi_k^i \mid \xi_{k-1}^i)}{q(\xi_k^i \mid \xi_{k-1}^i, z_k)}$$

- Resultant pdf is represented as:

$$p(\xi_k \mid Z^k) \approx \sum_{i=1}^{N_s} n_k^i \delta(\xi_k - \xi_k^i)$$

An unavoidable drawback of the algorithm is that after a few iterations all particle weights except few tend to 0. This problem is called "*the degeneracy problem"*. A good choice of the importance density and resampling help to overcome this problem [22]. Optimal choice for importance function is

$$\begin{aligned} q(\xi_k \mid \xi_{k-1}^i, z_k)_{opt} &= p(\xi_k \mid \xi_{k-1}^i, z_k) \\ &= \frac{p(z_k \mid \xi_k, \xi_{k-1}^i) p(\xi_k \mid \xi_{k-1}^i)}{p(z_k \mid \xi_{k-1}^i)}. \end{aligned} \tag{2.53}$$

And the weights for optimal importance density are

$$n_k^i \propto n_{k-1}^i p(z_k \mid \xi_{k-1}^i) \tag{2.54}$$

with

$$p(z_k \mid \xi_{k-1}^i) = \int p(z_k \mid \xi_k) p(\xi_k \mid \xi_{k-1}^i) d\xi_k . \tag{2.55}$$

Using the optimal importance density is not always possible. It requires to sample from $p(\xi_k \mid \xi_{k-1}^i, z_k)$ and evaluate the integral (2.55) over the new state. If the importance density is taken as

$$q(\xi_k \mid \xi_{k-1}^i, z_k) = p(\xi_k \mid \xi_{k-1}^i). \tag{2.56}$$

Then the weight update simplifies to

$$n_k^i \propto n_{k-1}^i p(z_k \mid \xi_k^i). \tag{2.57}$$

The selection of importance density is important for the filter performance and worth attention.

As noted before there should also be a resampling phase to prevent degeneracy. Resampling removes particles with less weights and populates the particles which had higher weights. After resampling all new particles have the equal weight *1/N*$_s$. By this way a new set, $\{\xi_k^{i*}\}_{i=1}^{N_s}$, is generated by drawing N_s samples from the distribution

$$p(\xi_k \mid Z^k) \approx \sum_{i=1}^{N_s} n_k^i \delta(\xi_k - \xi_k^i). \tag{2.58}$$

The probability of the a new selected particle with an index i^* being the j^{th} one of the previous set is proportional to its weight as

$$\Pr(\xi_k^{i*} = \xi_k^j) = n_k^j. \tag{2.59}$$

Although resampling reduces degeneracy it introduces another problem known as s*ample impoverishment* where particles with high weights will be sampled many times resulting to a loss of diversity. Another point to mention is that resampling phase limits the parallel implementation for particles because they are combined at each resampling phase.

In practice there are various particle filters each of which can be regarded as a special case of SIS. Selection of the importance density and resampling methodology differentiate these algorithms. Sequential Importance Resampling filter (SIR) is the particle filter that uses the prior density for importance density as follows

$$q(\xi_k^i \mid \xi_{k-1}^i, z_k) = p(\xi_k^i \mid \xi_{k-1}^i) \tag{2.60}$$

and resampling is visited at every time index. SIR method does have the advantage that the particle weights are easily evaluated and that the importance density can be easily sampled.

2.4.2 Rao-Blackwellized (Marginalized) Particle Filters

One of the drawbacks of particle filtering is its computational cost. Computational complexity of PF increases with the size of the state dimension. One solution to this problem is to marginalize out the states appearing linearly in the dynamics. If there is a linear sub-structure available in the general model, marginalization can be utilized in order to obtain better estimates and possibly reduce the computational complexity [17, 24].

Basic idea of Rao-Blackwellization is to partition the state vector as

$$\xi_k = \begin{bmatrix} \xi_k^l \\ \xi_k^n \end{bmatrix} \tag{2.61}$$

where ξ_k^l denotes the part of the state with conditionally linear dynamics and ξ_k^n denotes the nonlinear part of the state. Using Bayes' theorem we can marginalize out the linear part of the state from the state-space model and estimate them using the Kalman filter.

The nonlinear state variables are estimated using the particle filter. This technique is called Rao-Blackwellization or marginalizaiton. Rao-Blackwellization can be applied for diagonal, triangular and generalized model.

The generalized model for the marginalized particle filter is

$$\begin{aligned} \xi_{k+1}^n &= f_k^n\left(\xi_k^n\right) + A_k^n\left(\xi_k^n\right)\xi_k^l + G_k^n\left(\xi_k^n\right)v_k^n \\ \xi_{k+1}^l &= f_k^l\left(\xi_k^n\right) + A_k^l\left(\xi_k^n\right)\xi_k^l + G_k^l\left(\xi_k^n\right)v_k^l \\ z_k &= h_k\left(\xi_k^n\right) + C_k\left(\xi_k^n\right)\xi_k^l + w_k \\ vk &= \begin{bmatrix} v_k^l \\ v_k^n \end{bmatrix} \sim N\left(0, Q_k\right) \quad Q_k = \begin{bmatrix} Q_k^l & Q_k^{\ln} \\ (Q_k^{\ln})^T & Q_k^n \end{bmatrix} \\ w_k &\sim N(0, R_k), \quad \xi_0^l \sim N(\bar{x}_0, \bar{P}_0). \end{aligned} \tag{2.62}$$

However our problem is in the form of the "diagonal form" so the details of the general form are not given here. The problem formulated in the thesis is in the diagonal form given below:

$$\begin{aligned} \xi_{k+1}^{n} &= f_k^{n}(\xi_k^{n}) + v_k^{n} \\ \xi_{k+1}^{l} &= A_k^{l}(\xi_k^{n})\xi_k^{l} + v_k^{l} \\ z_k &= h_k(\xi_k^{n}) + C_k(\xi_k^{n})\xi_k^{l} + w_k \end{aligned} \tag{2.63}$$

where

$$v_k = \begin{bmatrix} v_k^{l} \\ v_k^{n} \end{bmatrix} \sim N(0, Q_k), \quad Q_k = \begin{bmatrix} Q_k^{l} & 0 \\ 0 & Q_k^{n} \end{bmatrix} \tag{2.64}$$

$$w_k \sim N(0, R_k), \quad \xi_0^{l} \sim N(\bar{x}_0, \bar{P}_0).$$

The density ξ_0^{n} can be arbitrary but has to be known. Aim is to find $p(\xi_k | Z_k)$ using the Bayes' theorem:

$$p(\xi_k^{l}, \Xi^{n,k} | Z^{k}) = p(\xi_k^{l} | \Xi^{n,k}, Z^{k})\, p(\Xi_k^{n} | Z^{k}) \tag{2.65}$$

where $\Xi^{n,k} = \{\xi_j^{n}\}_{j=0}^{k}$ and $Z^{k} = \{z_j\}_{j=0}^{k}$.

First part is computed using the Kalman Filter for $p(\xi_k^{l} | \Xi^{n,k}, Z^{k})$ as:

$$p(\xi_k^{l} | \Xi_k^{n}, Z^{k}) = N(\xi_k^{l}; \hat{\xi}_{k|k}^{l}, P_{k|k}) \tag{2.66a}$$

$$p(\xi_{k+1}^{l} | \Xi^{n,k+1}, Z^{k}) = N(\xi_{k+1}^{l}; \hat{\xi}_{k+1|k}^{l}, P_{k+1|k}) \tag{2.66b}$$

where

$$\hat{\xi}_{k|k}^{l} = \hat{\xi}_{k|k-1}^{l} + K_k(z_k - h_k - C_k \hat{\xi}_{k|k-1}^{l}) \tag{2.67a}$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_{k|}^{T} \tag{2.67b}$$

$$S_k = C_k P_{k|k-1} C_k^{T} + R_k \tag{2.67c}$$

$$K_k = P_{k|k-1} C_k^T S_k^{-1} \tag{2.67d}$$

and

$$\hat{\xi}_{k+1|k}^l = A_k^l \hat{\xi}_{k|k}^l \tag{2.68a}$$

$$P_{k+1|k} = A_k^l P_{k|k} (A_k^l)^T + Q_k^l . \tag{2.68b}$$

The recursions are initiated with $\hat{\xi}_0^l = \bar{\xi}_0$ and $P_{0|-1} = \bar{P}_0$. Note that h_k, A_k, and C_k are functions of ξ_k^n. Since ξ_k^n is represented by particles, for each particle we have the associated linear system.

Second part is the Particle Filter for calculation of $p(\Xi_k^n | Z_k)$:

$$p(\Xi^{n,k} | Z^k) = \frac{p(z_k | \Xi^{n,k}, Z^{k-1}) \, p(\xi_k^n | \Xi^{n,k-1}, Z^{k-1})}{p(z_k | Z^{k-1})} \, p(\Xi^{n,k-1} | Z^{k-1}) \tag{2.69}$$

where an approximation of $p(\Xi^{n,k-1} | Z^{k-1})$ is provided by the previous iteration and

$$p(z_k | \Xi^{n,k}, Z^{k-1}) = N(z_k; h_k + C_k \hat{\xi}_{k|k-1}^l, C_k P_{k|k-1} C_k^T + R_k) \tag{2.70a}$$

$$p(\xi_{k+1}^n | \Xi^{n,k}, Z^k) = N(\xi_{k+1}^n; f_k^n, Q_k^n) \tag{2.70b}$$

are required for particle filter weight update and time update calculations. Hence for each particle a corresponding linear system is formed.

Below is the algorithm for the diagonal model where N stands for the number of particles:

1. Initialization: For $i=1,\ldots,N$ initialize $\xi_{0|-1}^{n,(i)} \sim p(\xi_0^n)$ particles and set $\{\xi_{0|-1}^{l,(i)}, P_{0|-1}^{(i)}\} = \{\bar{\xi}_0^l, \bar{P}_0\}$,

2. For $i=1,\ldots,N$ evaluate the importance weights $q_k^{(i)} = p(z_k \left| \Xi^{n,k,(i)}, Z^{k-1} \right.)$ and normalize $\tilde{q}_k^{(i)} = \dfrac{q_k^{(i)}}{\sum_{j=1}^{N} q_k^{(j)}}$,

3. Particle Filter measurement update: Resample with replacement N particles according to,

$$\Pr(\xi_{k|k}^{n,(i)} = \xi_{k|k-1}^{n,(j)}) = \tilde{q}_k^{(j)},$$

4. Particle filter time update and Kalman Filter

a-) Apply Kalman Filter measurement update, using (2.67),

b-) Particle Filter time update: For $i=1,\ldots,N$ predict new particles according to

$$\xi_{k+1|k}^{n,(i)} \sim p(\xi_{k+1|k}^{n} \left| \Xi^{n,k,(i)}, Z^{k} \right.)$$

c-) Apply Kalman Filter time update, using (2.68),

5. Set $k=k+1$ and iterate from step 2.

The marginalized particle filter is an effective tool to facilitate the implementation of particle filters for real applications having large state vectors.

CHAPTER 3

BEARINGS-ONLY TRACKING

This Chapter is for the BOT problem description. The formulation, observability and target models for BOT is discussed. Techniques used like Range Parameterization and Interacting Multiple Model filtering for BOT are explained.

3.1 Formulation for the BOT Problem

In this study we have concentrated on the 2-D problem. 3-D problem has similar characteristics but it has more complex representation. Unobservability problem is inherent also for this case although two angle measurements are available.

Figure 1 describes the BOT problem where owncraft/ownship has a motion with 2 legs. Conventionally bearing angle is measured in clockwise direction relative to the north.

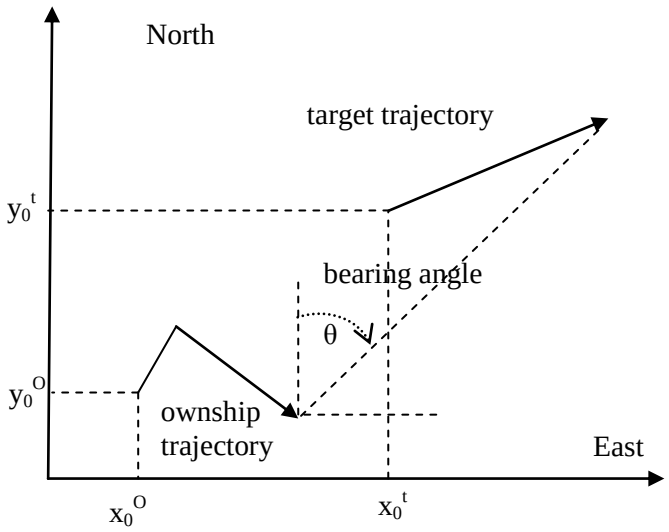


Figure 1- 2-D Bearings-only tracking.

Assuming the target has almost constant velocity motion we begin formulating the non-maneuvering target case by defining the target and ownship state vectors. Target's state vector is composed of position and velocity components in 2-D as

$$\xi_k^t = \left[x_k^t \, y_k^t \, \dot{x}_k^t \, \dot{y}_k^t\right]^T . \tag{3.1}$$

Similarly the ownship has

$$\xi_k^o = \left[x_k^o \, y_k^o \, \dot{x}_k^o \, \dot{y}_k^o\right]^T \tag{3.2}$$

as its state vector. For the non-maneuvering case it is possible to use relative state vector:

$$\xi_k = \xi_k^t - \xi_k^o = \left[x_k \;\; y_k \;\; \dot{x}_k \;\; \dot{y}_k\right]^T . \tag{3.3}$$

For this formulation the state equation takes the form

$$\xi_{k+1} = A\xi_k + G_k v_k - U_{k,k+1} \tag{3.4}$$

where

$$A_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; G_k = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix} \tag{3.5}$$

T being the sampling interval, v_k is i.i.d process noise vector with $v_k \sim N(0, Q)$, and

$$U_{k,k+1} = \begin{bmatrix} u_{k_1} \\ u_{k_2} \\ u_{k_3} \\ u_{k_4} \end{bmatrix} = \begin{bmatrix} x_{k+1}^o - x_k^o - T\dot{x}_k^o \\ y_{k+1}^o - y_k^o - T\dot{y}_k^o \\ \dot{x}_{k+1}^o - \dot{x}_k^o \\ \dot{y}_{k+1}^o - \dot{y}_k^o \end{bmatrix} \tag{3.6}$$

is the known input vector which is nonzero if there are legs or other type of acceleration on the observer trajectory. Observer position and velocity are assumed to be known by the ownship's navigation system. The measurement is the angle

between the north and the line-of-sight of the target relative to the ownship and is modeled as

$$z_k = c(\xi_k) + Hw_k, \tag{3.7}$$

where w_k is zero mean Gaussian i.i.d noise of the measurement. *H* is 1, and the true bearing angle is

$$c(\xi_k) = \arctan(\frac{x_k}{y_k}). \tag{3.8}$$

The relative motion of the target with respect to the observer is a constant velocity motion model described by (3.4) with the input that is related to the ownship maneuver.

The input model (3.6) models a case where we can freely select the input. Position and the velocity of the ownship can have jumps at all time steps depending on the input selection. This is clearly not realistic. A more realistic modeling is choosing the acceleration of the ownship freely in both directions. The formulation corresponding to this case is given below.

Similar to (3.3) the relative state vector is used in the free acceleration case as

$$\xi_k = \xi_k^t - \xi_k^o = \left[x_k \;\; y_k \;\; \dot{x}_k \;\; \dot{y}_k\right]^T. \tag{3.9}$$

The corresponding state equation is

$$\xi_{k+1} = A\xi_k + Gv_k - Gu_k^T \tag{3.10}$$

where

$$A_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, G_k = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix} \tag{3.11}$$

T being the sampling interval, $\{v_k\}$ is an i.i.d process noise vector with $v_k \sim N(0,Q)$, and

$$u_k = \begin{bmatrix} u_{k_1} \\ u_{k_2} \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \end{bmatrix} \tag{3.12}$$

is the known input vector that corresponds to the known observer accelerations. The measurement equation is the same as (3.7):

$$z_k = c(\xi_k) + Hw_k \ . \tag{3.13}$$

Optimal ownship trajectory generation is generally a constrained optimization problem because of the ownship maneuver limitations. Using an unconstrained acceleration is obviously not possible for real problems. Even constrained acceleration vector as proposed by [35] has some drawbacks: the ownship may eventually reach to some large speeds that make the control of the direction more difficult afterwards. Also only limiting the acceleration does not limit the ownship velocity which is again not always realistic.

Another possible model for the ownship motion is a constant speed with free heading motion model. This approach lets the ownship have a constant speed (maximum possible reachable speed [32]) which is a more realistic assumption for some cases. In Chapter 7 we use this model in the optimization algorithm to select the observer motion. Using this model, optimization algorithm deals with the ownship direction only, rather than trying to tune the ownship velocity and direction at the same time by controlling the two dimensional ownship acceleration vector.

For this model we define the ownship's state as its position and the speed,

$$\xi^o = \left[x^o y^o s\right]^T. \tag{3.14}$$

The input of the system is the heading of the ownship with respect to the north denoted by $u_k \equiv \varphi_k$ at time *k*.

The ownship motion is modeled as

$$\xi^o_{k+1} = L(\varphi_k)\xi^o_k \tag{3.15}$$

where *T* is the sampling period and

$$L(\varphi_k) = \begin{bmatrix} 1 & 0 & T\cos\varphi_k \\ 0 & 1 & T\sin\varphi_k \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.16}$$

We assume that the heading angle, φ_k, may change at the beginning of each sampling interval and is constant during the sampling period *T*. Heading can change between [-π,π] or may be constrained depending on the problem. The ownship's state equation together with target's state equation forms the system equations. The measurement is still a function of the relative difference between the target and ownship as in (3.7).

As mentioned in the first chapter the tracking problem is nonlinear because of the measurement equation. For tracking maneuvering targets maneuver models like Coordinated Turn (CT) is taken into account and model switching is used. Details of the CT models and Interacting Multiple Model (IMM) tracking will be given later. Next section is devoted to a more detailed description of the unobservability problem.

3.2 Unobservability in Bearings-only Tracking

In bearings-only tracking the only measurement is the direction of arrival of a signal from the target to the observer. The kinematic information of the observer is also known. The aim is to find the location of the target by using only the bearing angle. This aim is achieved when the observer makes an accelerated motion. In the

literature it is stated that "*the observer has to have a higher order motion than the target"* for a proper tracking. It can also be stated as *"the observer trajectory has to have one more nonzero derivative than the target trajectory"* [11]. A simple observation reveals this fact as explained below.

Considering a 2-D scenario where the observer and the target has constant velocity and the true bearing angle to the target at time *t* is

$$\begin{aligned}\arctan\frac{x^t(t)-x^o(t)}{y^t(t)-y^o(t)} &= \arctan\frac{x^t(0)+\dot{x}^t t-x^o(0)-\dot{x}^o t}{y^t(0)+\dot{y}^t t-y^o(0)-\dot{y}^o t}\\ &= \arctan\frac{x^t(0)-x^o(0)+(\dot{x}^t-\dot{x}^o)t}{y^t(0)-y^o(0)+(\dot{y}^t-\dot{y}^o)t}\\ &= \arctan\frac{(x^t(0)-x^o(0))k+(\dot{x}^t-\dot{x}^o)kt}{(y^t(0)-y^o(0))k+(\dot{y}^t-\dot{y}^o)kt}\end{aligned} \tag{3.17}$$

for $\forall k \neq 0$. Hence same bearing angles are measured if the relative position and the velocity is multiplied by any nonzero constant k. Thus the angle measurements cannot give a unique solution violating the full and unique recovery of the initial state. This would not be the case if the observer has a higher order term (acceleration) for some period of time during tracking. This can be satisfied for example when the ownship has piecewise constant velocity which yields legs.

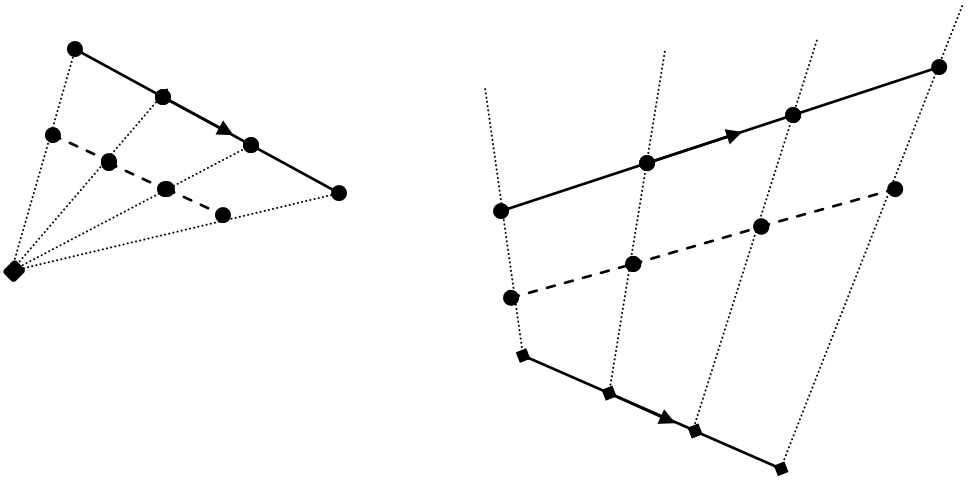


Figure 2- Unobservable cases: First case: Stationary observer and constant velocity target. Second case: Constant velocity observer and constant velocity target. Dashed paths are example alternative constant velocity target paths that can generate same line-of-sight measurement (Target is denoted by circles and observer is denoted by squares.).

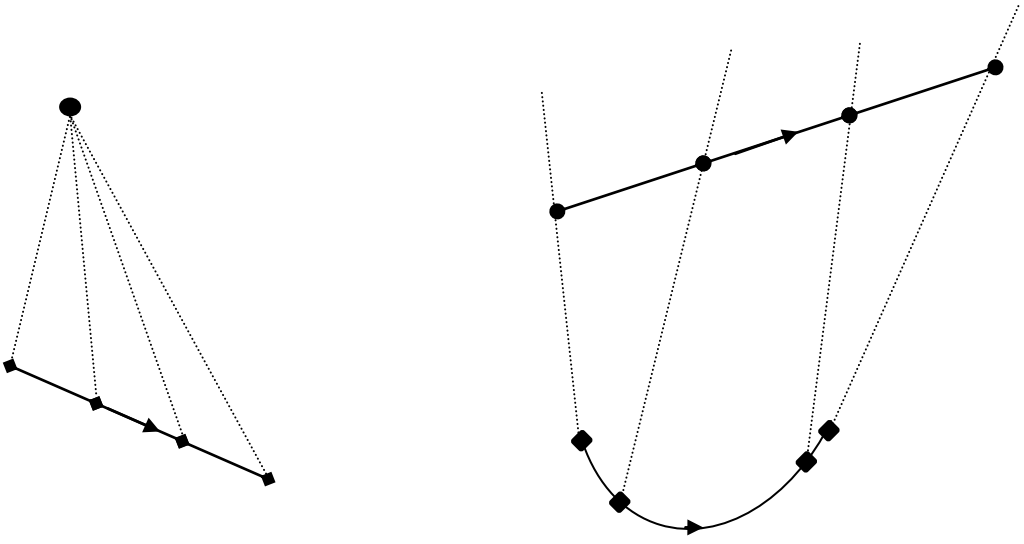


Figure 3- Observable cases: First case: Stationary target and constant velocity observer. Second case: Constant velocity target and accelerating observer (Target is denoted by circles and observer is denoted by squares.).

More detailed analysis of observability for BOT can be found in [3- 8].

3.3 Models of Maneuvering Targets

The constant velocity target model for the BOT problem for non-maneuvering case is already introduced. In this model acceleration of the target is considered as white noise. For the maneuvering targets, higher order models like the constant acceleration (CA) model are sometimes more appropriate [10]. In CA the time derivative of the acceleration is considered as white noise rather than the acceleration.

Another important maneuvering target model is the coordinated turn (CT) model. Coordinated Turn Model assumes that the target is moving with a constant speed and turning with a turn rate, ω. The derivative of the acceleration vector *a* can be written in terms of turn rate, ω, velocity vector *v*, and a white noise input *w* as follows

$$\dot{\mathbf{a}} = -\omega^2 \mathbf{v} + \mathbf{w} . \tag{3.18}$$

The turn rate is a cross-product expression

$$\omega = \frac{|\mathbf{v}\,\mathbf{x}\,\mathbf{a}|}{v^2} . \tag{3.19}$$

Using these two equations the two dimensional modeling of the target motion having a coordinated turn is possible. Details are not presented here however the discrete-time state transition matrix for a coordinated turn with a turn rate of ω is given as

$$A(\omega_k) = \begin{bmatrix} 1 & \dfrac{\sin \omega_k T}{\omega_k T}.T & 0 & \dfrac{-(1-\cos \omega_k T)}{\omega_k T}.T \\ 0 & \cos \omega_k T & 0 & -\sin \omega_k T \\ 0 & \dfrac{(1-\cos \omega_k T)}{\omega_k T}.T & 1 & \dfrac{\sin \omega_k T}{\omega_k T}.T \\ 0 & \sin \omega_k T & 0 & \cos \omega_k T \end{bmatrix} \tag{3.20}$$

where *T* is the sampling period. To track maneuvering targets using a constant velocity model and two coordinated turn rate models is common [10]. An IMM algorithm can be implemented with the three models. One of the CT models is used for positive turns where the second is used for the negative turns.

3.4 Range Parameterized EKF

Bearing measurement does not provide information for the range of the target. In the literature the range ambiguity of BOT is handled by a Multiple Model (MM) approach. The idea is using a mixture of Gaussian representation instead of a single Gaussian in modeling the initial range uncertainty so that the initial range uncertainty can be represented better. In this representation each mixture component can be considered as a hypothesis. These hypotheses are generated with probabilities that are proportional to the range intervals they occupy. This approach results in an almost uniform probability distribution over the range interval. Such a parameterization is applied in the Cartesian Coordinates in [15]. Combined state estimate is obtained as weighted sums of the state estimates and covariance matrices generated by the hypotheses. This is a Gaussian Mixture (GM) filter where each component describes a different initialization. It has a fixed number of sub-filters. Each hypothesis produces a mixture component of GM and is iterated by an EKF.

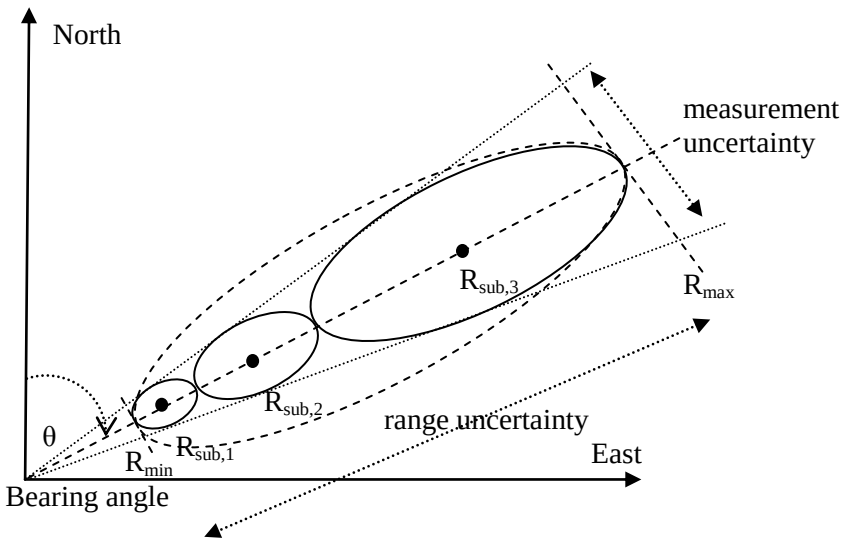


Figure 4- Range interval is divided into sub-intervals. Their uncertainty ellipses define a mixture nearly uniform comparing to a single Gaussian (dashed ellipse).

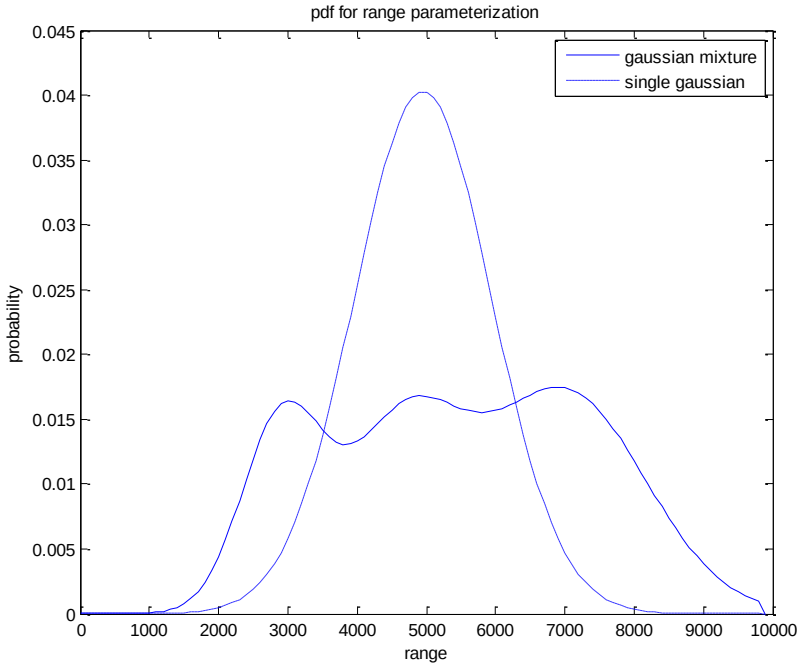


Figure 5- The pdf of a Gaussian Mixture Density for range parameterization with 3 intervals can approximate uniform range distribution (solid line) better than the pdf of a single Gaussian (dashed line).

Reference [15] summarizes the general qualities of the range parameterization approach as:

a) "*The initial position uncertainty representation is better adapted to the true sector shape compared to the single model elliptical shape.*"

b) "*The initial range probability distribution can be made almost uniform or any desired distribution.*"

c) "*The nonlinear effects are the same for all models.*"

d) "*Numerical problems that come from the transformation of ellipses, are decreased as the eccentricity decreases with the number of hypotheses.*"

e) "*The Cartesian coordinate representation gives simpler state and covariance calculations.*"

Based on [15], Gaussian Mixture Measurements-Integrated Track Splitting (GMM-ITS) filter is developed in [16] where both non-Gaussian measurement likelihood and non-Gaussian track state are approximated by Gaussian mixtures. In this study

the (unmeasured) range is assumed to belong to one of the sub-range intervals and treated as pseudo measurements of the range for each bearing measurement. Each bearing measurement generates the pseudo measurements by range parameterization. Each pseudo track is updated by the pseudo measurements, generating a new set of pseudo tracks at time *k*. Therefore number of pseudo tracks increase exponentially. These pseudo tracks form the Gaussian mixture.

In [16], by using the Gaussian mixture representation for both measurements and track estimates, the need for nonlinear estimation (Extended Kalman filters, Unscented Kalman filters or Particle Filters) is eliminated. As seen on Gaussian mixture type filters, component pruning scheme needs to be implemented to control the number of track components. This allows a trade-off between the number of track components and the track quality.

3.4.1 Range Parameterization Procedure

Assuming that the target can initially be in a range determined by the interval [Rmin, Rmax], and has a uniform distribution, range parameterization is done as follows [15].

Let $L_{sub,i,}$ and $R_{sub,i}$ be the length and the mean range of the sub-interval *i* at the initial time and N_H is the number of sub-intervals. The range interval [R_{min} R_{max}] is divided to N_H sub intervals such that the interval length to its mean range ratio for each interval is constant, *c*:

$$\frac{L_{sub,i}}{R_{sub,i}} = c \qquad i = 1,....., N \,. \tag{3.21}$$

Since *c* is constant the eccentricity of position uncertainty ellipses are same for each sub interval. (3.21) can be satisfied if sub-intervals are determined by geometric progression. Assuming ρ is the common ratio it can be obtained as

$$R_{\max} = R_{\min} \rho^{N_H} , \tag{3.22}$$

that is

$$\rho = \left(\frac{R_{\max}}{R_{\min}} \right)^{\frac{1}{N_H}}. \tag{3.23}$$

This value is used for the geometric progression to find each sub interval R_i using the relation

$$\rho = \left(\frac{R_{i+1}}{R_i} \right) \quad i = 1...N_H \tag{3.24}$$

starting from $R_1 = R_{\min}$.

The length and the mean range of each sub interval are

$$L_{sub,i} = R_{i+1} - R_i \tag{3.25}$$

$$R_{sub,i} = 0.5(R_{i+1} + R_i)\,. \tag{3.26}$$

Each sub interval is represented by an ellipse whose covariance is used to obtain the covariance matrix (P_i) of the sub measurement *i*. This value depends on the bearing angle measurement θ and its standard deviation σ_θ as

$$P_i = T \begin{bmatrix} (L_{sub,i}/2)^2 & 0 \\ 0 & (R_{sub,i}\sigma_\theta) \end{bmatrix} T^T \tag{3.27}$$

where

$$T = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}. \tag{3.28}$$

Denoting γ_i the probability that sub interval *i* contains the target, γ_i is assumed either proportional to the length [15] or the area of the ellipse [16] for the sub interval *i*.

3.5 IMM for BOT of Maneuvering Targets

3.5.1 Review for Interactive Multiple Model Based BOT

Maneuvering target tracking is itself a hard problem even when the states are completely observable. The maneuver detection and modeling is important and Interactive Multiple Model (IMM) is an efficient method considering its low computation requirements. In a BOT scenario when the target starts maneuvering, the tracking model becomes unobservable. Even when the own-craft is maneuvering this may not be sufficient for observability of a target having a coordinated turn (CT) with high turn rate. Therefore model selection is very important for accurate target motion analysis in an IMM based filter.

In reference [10], an IMM filter with 2 models is introduced for BOT. The author claims that 2-Model and 3-Model IMM's give similar performance in radar applications. Based on this experience, 2-Model IMM is applied to a BOT (an Infra-red Search and Track (IRST)) problem in [10]. Modified Spherical Coordinate filter (MSC), which uses an alternative state representation based on spherical coordinates is used for non-maneuvering target and a constant acceleration (CA) filter is used for the maneuvering part of the trajectory. The CA filter is composed of two decoupled filters for azimuth and elevation where angle, angle rate and angle acceleration are used as state variables in each of them. In [10] IMM filtering works well because the state estimates of less likely models are updated with the states of higher probability models and therefore less likely models does not develop large errors and overly optimistic covariance matrices.

In this study we use 'coordinated turn' (CT) models in an IMM framework. A coordinated turn model with known angular velocity is linear in Cartesian coordinates which is the great advantage of it. A very common multi model approach that uses CT models is to use models that reflect the system behavior at every instance of time, i.e. the system is assumed to be described by only one of the models where models are different from each other only by the angular rate of the CT. According to linear CT approach turn rate, ω_T, must be known or at least a good

estimate of ω_T must be available. However it is possible that a target makes turns with a wide range of possible ω_T values. This makes the known turn rate CT approach impractical. The solution is to use IMM. A turn with a small angular velocity can be obtained as a mixture of a turn with a large angular velocity and a CV (zero angular velocity) motion. Since the models interact in the mixing part of the IMM and the estimates are the merged estimates of the sub-models IMM is well suited to uncertain and/or time varying systems. By choosing a large value of the angular velocity, $\omega_{T,max}$, all turns with smaller turn rates can be regarded as combinations of a CT model with $\omega_{T,max}$ and a CV model. This idea is applied in [14]. It uses three models in the IMM. One CV, one CT with positive angular velocity, ω_{Tmax}, and one with negative angular velocity, $-\omega_{Tmax}$, to cover all target dynamics in this range.

The usage of IMM in the first proposed algorithm in this study is given in Chapter 5. Next section is a brief review of IMM formulation.

3.5.2 IMM Formulation

Assuming currently there are N_H hypotheses, IMM applies a mixing for the next filter cycle. This keeps hypothesis number fixed to N_H. This special hypothesis mixing is appealing and it keeps the IMM filter computationally simple but still accurate. Below is the diagram for IMM algorithm for a three model system.

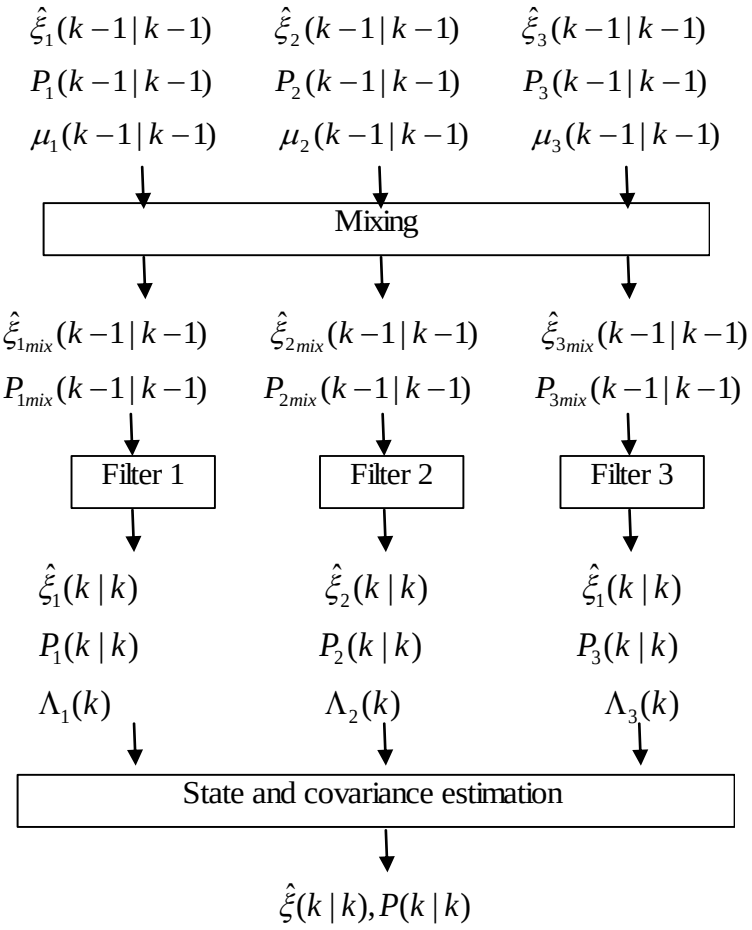


Figure 6- Flow of IMM algorithm for a 3 model system.

In Figure 6:

$\hat{\xi}_i(k-1|k-1)$ is the state estimate at *k-1* for filter *i*,

$P_i(k-1|k-1)$ is the covariance matrix at *k-1* for filter *i*,

$\mu_i(k-1)$ is the probability that the target is in model *i* after the measurement at *k-1*,

$\mu_{ij}(k-1)$ is the probability that the target was in model *i* at *k-1* given the target will make a transition to model *j*,

T_{ij} is the transition probability to state *j* given the target is at state *i*,

$$\mu_{ij}(k-1) = \frac{T_{ij}\mu_i(k-1)}{c_j(k-1)} \tag{3.29}$$

where

$$c_j(k-1) = \sum_{i=1}^{N_H} T_{ij}\mu_i(k-1) \tag{3.30}$$

is the mixing probability. The mixing is done as

$$\hat{\xi}_{j,mix}(k-1\,|\,k-1) = \sum_{i=1}^{N_H} \mu_{ij}(k-1)\hat{\xi}_i(k-1\,|\,k-1), \tag{3.31}$$

and

$$P_{j,mix}(k-1\,|\,k-1) = \tag{3.32}$$

$$\sum_{i=1}^{N_H} \mu_{ij}(k-1)\begin{pmatrix} P_i(k-1\,|\,k-1) + \\ [\hat{\xi}_i(k-1\,|\,k-1) - \hat{\xi}_{j,mix}(k-1\,|\,k-1)][\hat{\xi}_i(k-1\,|\,k-1) - \hat{\xi}_{j,mix}(k-1\,|\,k-1)]^T \end{pmatrix}$$

The mixed means and their associated covariance matrices, $\left(\hat{\xi}_{j,mix}(k-1\,|\,k-1), P_{j,mix}(k-1\,|\,k-1)\right)$, are used in the j^{th} hypothesis' matched Kalman filter equation to re-estimate the posterior density of j^{th} hypothesis at time k. The posterior can be represented by $\mathcal{N}[\xi_j(k); \hat{\xi}_j(k\,|\,k), P_j(k\,|\,k)]$. Probability of the j^{th} hypothesis is updated using the likelihood function again obtained by using the j^{th} Kalman Filter innovation covariance and measurement prediction as

$$\Lambda_j(k) = \mathcal{N}[z(k); \hat{z}_j(k\,|\,k\text{-}1), S_j(k)]. \tag{3.33}$$

Finally the model probabilities are

$$\mu_j(k) = \frac{1}{c}\Lambda_j(k)c_j(k-1) \tag{3.34}$$

where c is the normalization constant and model probabilities are used for getting the overall filter output as

$$\hat{\xi}(k \mid k) = \sum_{i=1}^{N_H} \mu_i(k) \hat{\xi}_i(k \mid k) \tag{3.35}$$

$$P(k \mid k) = \sum_{i=1}^{N_H} \mu_i(k) \begin{pmatrix} P_i(k \mid k) + \\ [\hat{\xi}_i(k \mid k) - \hat{\xi}(k \mid k)][\hat{\xi}_i(k \mid k) - \hat{\xi}(k \mid k)]^T \end{pmatrix}. \tag{3.36}$$

CHAPTER 4

MUTUAL INFORMATION FOR STOCHASTIC SYSTEMS

In this chapter a brief theoretical introduction is given for the entropy and the mutual information for stochastic systems.

4.1 Entropy

Entropy is commonly used as a measure of uncertainty for a random variable. For a discrete random variable it is defined as

$$H(X) = -\sum_{x \in S} p(x) \log p(x) \tag{4.1}$$

where *p(x)* is the probability mass function of the random variable *X*. For a continuous random variable, the differential entropy is defined by

$$h(X) = -\int_{S} f(x) \log f(x) dx. \tag{4.2}$$

where *S* is the support set and *f(x)* is the probability density function of the random variable *X*. The conditional differential entropy is given by

$$h(X \mid Y) = -\int\int_{S} f(x, y) \log f(x \mid y) dx dy = h(X, Y) - h(Y). \tag{4.3}$$

It is observed that the conditional entropy is the difference between the differential entropy of the joint random variables *X* and *Y* and the differential entropy of *Y*.

4.2 Mutual Information

"*Mutual information is a measure of the amount of information that one random variable contains about another random variable*" [42]. For the continuous random variables *X* and *Y* it is defined as

$$I(X,Y) = \int\int f(x,y) \log \frac{f(x,y)}{f(x)\,f(y)}\, dxdy \tag{4.4}$$

where *f(x,y)* is the joint probability density function of the random variables *X* and *Y*.

The relationship between the differential entropy and the mutual information is given by

$$I(X,Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X) = H(X) + H(Y) - H(X,Y). \tag{4.5}$$

It is remarkable that the measure is zero only if *X* and *Y* are independent. Also it is observed that the mutual information is the decrease in the entropy of one random variable when the other is known.

4.3 Mutual Information between the State and the Observation Sequences for Linear Gaussian Systems

Mutual information as an observability measure of a linear Gaussian system is summarized below [27]. Assume that the following linear, time invariant and Gaussian system is represented in the state space form:

$$\xi_{k+1} = A\xi_k + Gv_k \tag{4.6}$$

$$z_k = C\xi_k + Hw_k \tag{4.7}$$

where $\{v_k, k \in N\}$ and $\{w_k, k \in N\}$ are i.i.d. sequences. Also ξ_o, $\{v_k, k \in N\}$ and $\{w_k, k \in N\}$ are independent and

$$\xi_o \sim N(\xi_o; \bar{\xi}_0, \Sigma_0),\ v \sim N(v;0,Q),\ w \sim N(w;0,R). \tag{4.8}$$

We define the state and the measurement sequences as $\Xi^k = (\xi_0\ \xi_1 ... \xi_k)^T$ and $Z^k = (z_0\ z_1 ... z_k)^T$. For this system the observability measure between the state sequence, Ξ^k and the measurement sequence Z^k based on the mutual information is [36],

$$I(\Xi^k, Z^k) = \frac{1}{2}\log\frac{\left|\Sigma_{\Xi^k}\right\|\Sigma_{Z^k}\right|}{\left|\Sigma_{[\Xi^k, Z^k]}\right|} \tag{4.9}$$

or

$$I(\Xi^k, Z^k) = \frac{1}{2}\log\frac{\left|\Sigma_{\Xi^k}\right|}{\left|\Sigma_{\Xi^k|Z^k}\right|} \tag{4.10}$$

where |. | is the determinant.

It is also possible to reformulate the mutual information in terms of the system parameters, as follows [27],

$$I(\Xi^k, Z^k) = \frac{1}{2}\log\frac{\left|\Sigma_0\right\|GQG^T\right|^k}{\left|\Sigma_{\Xi^k} - \Sigma_{\Xi^k Z^k}\Sigma_{Z^k}^{-1}\Sigma_{Z^K\Xi^K}\right|}. \tag{4.11}$$

Another form of this equation, in terms of the basic system parameters is

$$I(\Xi^k, Z^k) = \frac{1}{2}\log\frac{\left|C_k A_k \Sigma_0 A_k^T C_k^T + C_k G_k Q_k G_k^T C_k^T + H_k R_k H_k^T\right|}{\left|H_k R_k H_k^T\right|} \tag{4.12}$$

where

$$C_k = \begin{bmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C \end{bmatrix}, \quad H_k = \begin{bmatrix} H & 0 & \cdots & 0 \\ 0 & H & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & H \end{bmatrix}, \tag{4.13}$$

$$G_k = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ G & 0 & \cdots & 0 \\ AG & & \cdots & 0 \\ \vdots & & \cdots & \vdots \quad \vdots \\ A^{k-1}G & A^{k-2}G & \cdots & G \end{bmatrix}, \quad Q_k = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q \end{bmatrix}, \tag{4.14}$$

$$R_k = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix}. \tag{4.15}$$

The observability measure defined as the mutual information between the output and the state has some interesting properties [27]:

i. The observability measure is not zero even when there are some unobservable states. A reason for this result is that the observability measure is the information gained about the whole state sequence not only about the value of the state vector at time 0; and the measurements provide some information about the state sequence. Another reason is that the initial state is somehow known since its pdf is known.

ii. If the determinant of the covariance matrix of the process noise, *Q*, increases, the observability measure increases. Larger *Q* values mean more uncertainty in the state that makes the measurements more valuable.

iii. If the determinant of the initial covariance matrix Σ_0 increases, the observability measure increases.

iv. If there is no process noise and no initial state uncertainty the observability measure is zero. This is reasonable since for this case the state sequence is known exactly and there is no need for the measurements.

v. If the determinant of the matrix $C^T C$ increases, then the lower bound of the observability measure increases. Large $C^T C$ makes the measurements more valuable.

vi. If the determinant of the covariance matrix of the measurement noise, *R*, increases, the lower bound of the observability measure decreases.

The properties given above show that the mutual information is a measure of the importance of a measurement for the estimation of the state. If state uncertainty is large, then the measurement is more valuable and the measure has a higher value. Facts (vi) says that if the measurement noise increases, then the observability measure decreases, which is logical considering the meaning of observability.

4.4 The Mutual Information between the State and the Measurement Sequences for Nonlinear Stochastic Systems

The mutual information between the state and the measurement sequences can be used as an observability measure for nonlinear stochastic systems [27]. The recursive form of the measure is derived in [27]. The obtained recursive form can be calculated by using particle filtering methods. Here we briefly review the method.

Given a nonlinear discrete time system as

$$\xi_{k+1} = a(\xi_k, v_k) \tag{4.16}$$

$$z_k = c(\xi_k, w_k) \tag{4.17}$$

where $\{v_k, k \in N\}$ and $\{w_k, k \in N\}$ are i.i.d. sequences, ξ_o, $\{v_k, k \in N\}$ and $\{w_k, k \in N\}$ are independent. State and measurement sequences are denoted by

$$\Xi^k = (\xi_0\ \xi_1 ... \xi_k)^T\ ,\ Z^k = (z_0\ z_1 ... z_k)^T\ . \tag{4.18}$$

For this system the mutual information as the observability measure between the state sequence Ξ_k, and the measurement sequence Z^k is

$$I(\Xi^k, Z^k) = \int\int f(\Xi^k, Z^k) \log \frac{f(\Xi^k, Z^k)}{f(\Xi^k) f(Z^k)} d\Xi^k dZ^k \tag{4.19}$$

where $f(\Xi^k)$ and $f(Z^k)$ are the probability density functions of Ξ^k and Z^k and $f(\Xi^k, Z^k)$ is their joint probability density function. The following theorem that will be applied to the ownship trajectory optimization of the BOT problem is given without proof.

Theorem 1 [27]: The observability measure can be recursively calculated as

$$I(\Xi^k, Z^k) = I(\Xi^{k-1}, Z^{k-1}) - h(z_k \mid \xi_k) + h(z_k \mid Z^{k-1}) . \tag{4.20}$$

When a deterministic input is applied to the system (4.16) becomes (4.21). It may be more appropriate to show its effect in the probability density functions to emphasize the dependency of the mutual information to the deterministic input as given in (4.22):

$$\xi_{k+1} = a(\xi_k, w_k, u_k) \tag{4.21}$$

$$I(\Xi^k, Z^k) = \int\int f_{U^k}(\Xi^k, Z^k) \log \frac{f_{U^k}(\Xi^k, Z^k)}{f_{U^k}(\Xi^k) f_{Uk}(Z^k)} d\Xi^k dZ^k \ . \tag{4.22}$$

This observability measure can be recursively calculated as

$$I_{U^{k-1}}(\Xi^k, Z^k) = I_{U^{k-2}}(\Xi^{k-1}, Z^{k-1}) - h(z_k \mid \xi_k) + h_{U^{k-1}}(z_k \mid Z^{k-1}) . \tag{4.23}$$

Equation (4.23) indicates the dependence of the terms on the input sequence that is applied to the system. Here U^k represents the applied pattern of the input sequence up to time *k*. This relation of the mutual information with the input plays a key role for observer trajectory generation to increase the mutual information which is discussed in Chapter 7.

CHAPTER 5

RANGE PARAMETERIZED INTERACTING MULTIPLE MODEL GAUSSIAN MIXTURE FILTER

5.1 The Problem and the Solution Methodology

Here we examine two dimensional bearings-only problem where only the bearing measurement θ_{mes} is available with a measurement error standard deviation of σ_θ. There is no range measurement. Target may be maneuvering or having a nearly constant velocity motion.

It was mentioned that target range observability is not always possible for bearings-only tracking problem. Furthermore the problem is a nonlinear tracking problem which is emerging from nonlinear bearing angle measurements. Unknown target initial position can only be constrained in a wide interval range. So the initial target state can better be modeled by mixture of Gaussians. Reference [15] formulates a fixed sized Gaussian Mixture filtering model developed to cope with initial uncertainty and system non linearity using range parameterization at initialization. Study in [16] extends this by applying range parameterization at each step during tracking.

The increased accuracy gained by these approaches can be extended to maneuvering target case which is the aim of this chapter. The filters of [15] and [16] are developed for constant velocity targets that limit their applicability to real life problems. Especially airborne targets are agile and have maneuvers which can be modeled by coordinated turns. Maneuvering target tracking requires additional maneuvering models and a model switching methodology. Computational efficiency of the IMM

filtering and the Gaussian Mixture filtering with mixture reduction techniques make the proposed filter computationally efficient for the bearings-only tracking problem.

The philosophy behind the method proposed here is to model each Gaussian mixture component of range parameterized Gaussian Mixture filter previously developed in [16] as an IMM sub-filter. By this way each sub-filter of the Gaussian Mixture is able to anticipate maneuvers and adapt in an IMM fashion. The IMM filter proposed here is a three model filter. These models are constant velocity (CV), coordinated turn positive (CTP), and coordinated turn negative (CTN). The mixing phase of IMM allows each model to update itself based on current target motion (either constant velocity or coordinated turn) so that the mismatching filters will not deteriorate the performance. The filter will be updated with the best known current state and when the target starts a maneuver its matching model becomes dominant. Note that all three models have the same state vector when coordinated turn models with constant turn rate are selected. This makes mixing phase of the IMM simple and straightforward.

The number of regions that partitions the range is denoted by N_L which is a fixed number. Their values generate pseudo range measurements at each time step. We assume that the probability of detection is one and there are no false measurements. With this assumption the number of measurements becomes N_L at each time step. We use Gaussian mixture representation of the state. So if there are N_M mixture components, each having three hypotheses there will be $N_M x3$ hypotheses and N_M IMM filters. This structure is different from the structure given in [15], where it has only N_M filters. At each time step a single angle measurement produces N_L pseudo measurements due to the pseudo ranges. That increases the number of IMM filters to $N_M x N_L$. This exponential increase is controlled by pruning less likely filters.

Figure **7** and Figure 8 illustrate the structure of the system and show the IMM filter, the Gaussian mixture components and range parameterization all together. The mean of each Gaussian mixture component is the combined mean of the hypothesis' means weighted by their probability, μ_{IMM}, which is found by the IMM iteration. Combined

covariance matrix of the Gaussian mixture component is also calculated where a spread of the mean term is included.

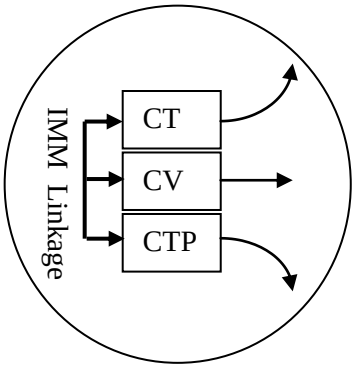
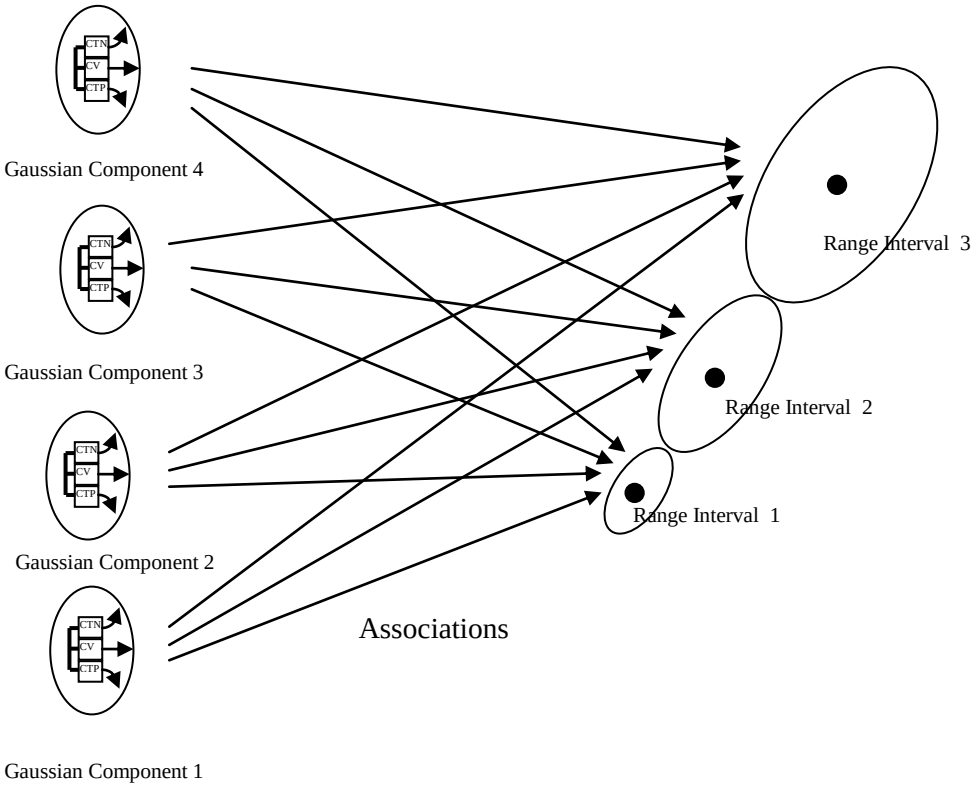


Figure 7- IMM structure of each mixture component



Gaussian Mixture at time k-1.

Range Parameterization at time k based on the new bearings-angle measurement.

Figure 8- Gaussian mixture components at time *k-1* and range parameterization at time *k* generates possible associations. Each association is regarded as a new Gaussian mixture component.

5.2 Implementation

Nomenclature

$N_M(k)$ is the number of Gaussian mixture components at time k,

$m=\{1,...,N_M\}$ is the index for the mixture components,

$N_I=3$ is the number of models for IMM filters in each mixture component,

$i=\{1,2,3\}$ is the index for the models in the IMM filters where 1, 2 and 3 stands for Constant Velocity (CV), Coordinated Turn Positive (CTP) and Coordinated Turn Negative (CTN), respectively.

$N_L(k)$ is the number of sub-intervals resulted by range parameterization after the angle measurement at time k, (Center of the each subinterval represents a pseudo measurement in Cartesian Coordinate system.),

$l=\{1,...,N_L\}$ is the index for the sub-intervals,

$z_l(k)$ is the pseudo measurement (in Cartesian Coordinates obtained via range parameterization) for the l^{th} sub-interval at time k, and γ_l is the associated probability of l^{th} pseudo measurement to be true obtained via the range parameterization procedure (The probability is proportional to the interval length.),

$R_l(k)$ is the pseudo measurement covariance matrix for the l^{th} sub-interval at time k, determined via the range parameterization procedure,

$\mu^i_{IMM,m}(k)$ is the i^{th} IMM model probability for m^{th} mixture component at time k,

$\mu_{GM,m}(k)$ is the m^{th} mixture component probability at time k,

$\hat{\xi}^i_m(k)$ is the state estimate of the i^{th} IMM model of the m^{th} mixture component at time k,

$P^i_m(k)$ is the covariance matrix for the i^{th} IMM model of the m^{th} mixture component at time k,

$\hat{z}_m^i(k)$ is the measurement prediction at time k for the i^{th} IMM model of the m^{th} mixture component,

$S_{m,l}^i(k)$ is the measurement innovation covariance matrix at time k of the i^{th} IMM model of the m^{th} mixture component for the l^{th} measurement sub-interval,

$\hat{\xi}_m(k)$ is the state estimate of the m^{th} mixture component for time k,

$P_m(k)$ is the covariance matrix for the m^{th} mixture component for time k,

$S_{m,l}(k)$ is the measurement innovation covariance matrix for time k of the m^{th} mixture component associated with l^{th} measurement interval,

$\hat{\xi}_{m,l}^i(k+1|k+1)$ is the state update of the i^{th} IMM model of the m^{th} mixture component at time $k+1$ using the l^{th} measurement interval at time $k+1$,

$P_{m,l}^i(k+1|k+1)$ is the covariance matrix update of the i^{th} IMM model of the m^{th} mixture component at time $k+1$ using the l^{th} measurement interval at time $k+1$,

$\theta_{mes}(k)$ is the angle measurement at time k,

$\theta_{mes}^k = \{\theta_{mes}(1), \theta_{mes}(2)..., \theta_{mes}(k)\}$ is the set of angle measurements up to time k,

$Q_m^i(k)$ is the system noise covariance matrix for the i^{th} IMM model of m^{th} mixture component.

The algorithm consists of prediction and filtering steps as usual. All these steps are done in IMM formulation. Measurement likelihoods based on the IMM filters is used for mixture probability updates of Gaussian mixtures. Also IMM outputs are used for the update of the mixture components which is not a standard application of the IMM filter. These properties permit us to use mixture of Gaussians with IMM formulation in bearings-only tracking problem. A very similar approach to the one described here is also observed in reference [25] for a multi-sensor framework.

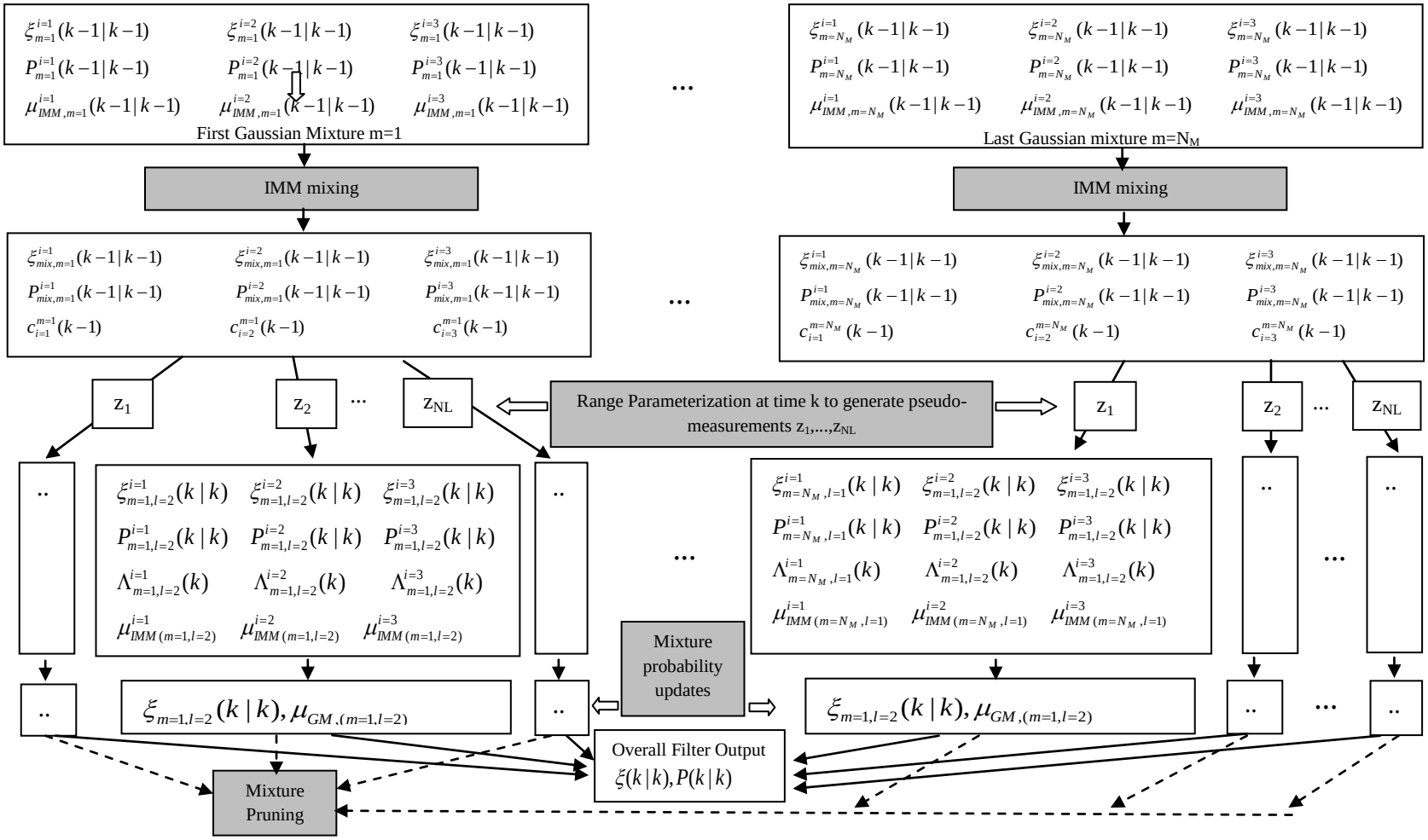


Figure 9-Flow diagram for the RPIMMGM filter

5.3 One Cycle of the Filter

Figure 9 is a diagram for the flow of the filter algorithm. The steps of the filter are given as:

The Prediction Step

A posteriori pdf for target position estimate at time *k-1* can be given as

$$p(\hat{\xi}(k-1) \mid \theta_{mes}^{k-1}) = \sum_{m=1}^{N_M(k-1)} \mu_{GM,m}(k-1) p(\hat{\xi}(k-1) \mid m, \theta_{mes}^{k-1}) \tag{5.1}$$

and the pdf of the state prediction can be written as

$$p(\hat{\xi}(k) \mid \theta_{mes}^{k-1}) = \int_{\hat{\xi}(k-1)} p(\hat{\xi}(k) \mid \hat{\xi}(k-1)) p(\hat{\xi}(k-1) \mid \theta_{mes}^{k-1})) d\hat{\xi}(k-1). \tag{5.2}$$

Using (5.1) and (5.2) we get

$$p(\hat{\xi}(k) \mid \theta_{mes}^{k-1}) = \sum_{m=1}^{M_{k-1}} \mu_{GM,m}(k-1) p(\hat{\xi}(k) \mid m, \theta_{mes}^{k-1}) \tag{5.3}$$

where

$$p(\hat{\xi}(k) \mid m, \theta_{mes}^{k-1}) = \sum_{i=1}^{3} \mu_{IMM,m}^{i}(k-1)\, p(\hat{\xi}(k) \mid m, i, \theta_{mes}^{k-1}) \tag{5.4}$$

is a combination of model pdf's, $p(\hat{\xi}(k) \mid m, i, \theta_{mes}^{k-1})$. The IMM mixing probabilities are calculated as

$$\mu_{ij}^{m}(k-1) = \frac{T_{ij}\mu_{IMM,m}^{i}(k-1)}{c_m^j(k-1)} \tag{5.5}$$

where $\mu_{ij}^{m}(k-1)$ is the probability that the target is in model *i* at *k-1* given the target will make a transition to model *j* for mixture component *m*. T_{ij} is the transition probability to model *j* given the target is at model *i*. Here,

$$c_j^m(k-1) = \sum_{i=1}^{N_H} T_{ij}\mu_{IMM,m}^{i}(k-1). \tag{5.6}$$

The IMM mixing is done as in (3.31):

$$\hat{\xi}^{j}_{m,mix}(k-1\,|\,k-1) = \sum_{i=1}^{N_H} \mu_{ij}^{m}(k-1)\hat{\xi}^{i}_{m}(k-1\,|\,k-1) \tag{5.7}$$

and

$$P^{j}_{m,mix}(k-1\,|\,k-1) = \tag{5.8}$$

$$\sum_{i=1}^{N_H} \mu_{ij}^{m}(k-1)\begin{pmatrix} P^{i}_{m}(k-1\,|\,k-1) + \\ [\hat{\xi}^{i}_{m}(k-1\,|\,k-1) - \hat{\xi}^{j}_{m,mix}(k-1\,|\,k-1)][\hat{\xi}^{i}_{m}(k-1\,|\,k-1) - \hat{\xi}^{j}_{m,mix}(k-1\,|\,k-1)]^{T} \end{pmatrix}$$

$\hat{\xi}^{j}_{m,mix}$ and $P^{j}_{m,mix}$ are used in model matched Kalman Filter equations of IMM. *A* being the state transition matrix,

$$\hat{\xi}^{i}_{m}(k\,|\,k-1) = A(k-1)\hat{\xi}^{i}_{m,mix}(k-1\,|\,k-1) \tag{5.9}$$

$$P^{i}_{m}(k\,|\,k-1) = A(k-1)P^{i}_{m,mix}(k-1\,|\,k-1)A(k-1)^{T} + Q^{i}_{m}(k-1). \tag{5.10}$$

The track prediction equation can be written as

$$p(\hat{\xi}(k)\,|\,m,i,\theta^{k-1}_{mes}) = \mathcal{N}\left(\hat{\xi}(k);\hat{\xi}^{i}_{m}(k\,|\,k-1),P^{i}_{m}(k\,|\,k-1)\right). \tag{5.11}$$

The Measurement Update Step

The update of IMM filter of each mixture component is done again over the IMM models for each parameterized measurement z_l using KF equations since the measurement is not the bearing angle but a position in Cartesian Coordinates:

$$\hat{\xi}^{i}_{m,l}(k\,|\,k) = \hat{\xi}^{i}_{m}(k\,|\,k-1) + K^{i}_{m,l}(k)\left(z_l(k) - H\hat{\xi}^{i}_{m}(k\,|\,k-1)\right) \tag{5.12}$$

$$P^{i}_{m,l}(k\,|\,k) = P^{i}_{m}(k\,|\,k-1) - K^{i}_{m,l}(k)S^{i}_{m,l}(k)K^{i\,T}_{m,l}(k)\,. \tag{5.13}$$

Given that *C* is the measurement equation matrix,

$$S^{i}_{m,l}(k) = CP^{i}_{m}(k \mid k-1)C^{T} + R_{l}(k) \tag{5.14}$$

$$K^{i}_{m,l}(k) = P^{i}_{m}(k \mid k-1)C^{T} S^{i}_{m,l}{}^{-1}(k) \tag{5.15}$$

and R_l is the measurement covariance matrix determined via Range Parameterization procedure.

IMM output

Each mixture component is summarized by the mean and the covariance matrix of the associated IMM filter's output. To find this, first IMM model probability updates, $\mu^{i}_{IMM,(m,l)}(k)$, are found as given in (3.34). $\mu^{i}_{IMM(m,l)}(k)$ is the IMM model probability for m^{th} mixture component, l^{th} measurement and i^{th} IMM model.

To do this, first likelihoods are calculated based on

$$\Lambda^{j}_{m}(z_{l}(k)) = \mathcal{N}[z_{l}(k); C\hat{\xi}^{j}_{m}(k \mid k-1), S^{j}_{m,l}(k)]. \tag{5.16}$$

Then,

$$\mu^{j}_{IMM(m,l)}(k) = \frac{1}{c}\Lambda^{j}_{m}(z_{l}(k))c^{m}_{j}(k-1) \tag{5.17}$$

where c is the normalizing constant.

State estimates and covariance matrices for mixture components are calculated as given in (3.35) and (3.36):

$$\hat{\xi}_{m,l}(k \mid k) = \sum_{i=1}^{3} \mu^{i}_{IMM(m,l)}(k)\hat{\xi}^{i}_{m,l}(k \mid k) \tag{5.18}$$

$$P_{m,l}(k \mid k) = \sum_{i=1}^{3} \mu^{i}_{IMM(m,l)}(k) \begin{pmatrix} P^{i}_{m,l}(k \mid k) + \\ [\hat{\xi}^{i}_{m,l}(k \mid k) - \hat{\xi}_{m,l}(k \mid k)][\hat{\xi}^{i}_{m,l}(k \mid k) - \hat{\xi}_{m,l}(k \mid k)]^{T} \end{pmatrix} \tag{5.19}$$

where

$$\mu^{i}_{IMM\,(m,l)}(k) = p(i \mid \theta^{k}_{mes}, m, l). \tag{5.20}$$

After the measurement update we have a total number of $N_M(k)=N_M(k\text{-}1)xN_L(k)$ Gaussian mixture components.

Mixture Probability Update

Denoting Gaussian *{m,l}* with a new index *m'* we have

$$p(\xi(k) \mid \theta(k)) = \sum_{m'=1}^{N_M(k)} \mu_{GM,m'}(k)\mathcal{N}\left(\xi(k); \hat{\xi}_{m'}(k \mid k), P_{m'}(k \mid k)\right). \tag{5.21}$$

Finally it is required to update mixture probabilities, $\mu_{GM,m'}(k)$, for Gaussian Mixture representation. These values are used for overall filter output generation and for mixture reduction, i.e. less likely mixture components are removed until the total number of components are equal to the maximum number of components allowed. The maximum number is determined as a performance parameter at the start of the tracking. After the removal of the less likely ones likelihood of the surviving components are renormalized. We have

$$\mu_{GM,m'}(k) = p(m' \mid \theta^{k}_{mes}) = p(m, l \mid \theta^{k}_{mes})$$

$$= \frac{p(\theta_{mes}(k) \mid m, l, \theta^{k-1}_{mes})\, p(l \mid m, \theta^{k-1}_{mes})\, p(m \mid \theta^{k-1}_{mes})}{p(\theta(k) \mid \theta^{k-1}_{mes})}. \tag{5.22}$$

We can simplify the irrelevant terms like

$$p(l \mid m, \theta^{k-1}_{mes}) = p(l) = \gamma_l. \tag{5.23}$$

Then,

$$\mu_{GM,m'}(k) = p(m, l \mid \theta^{k}_{mes}) = \frac{p(\theta_{mes}(k) \mid m, l, \theta^{k-1}_{mes})\gamma_l \mu_{GM,m}(k-1)}{p(\theta_{mes}(k) \mid \theta^{k-1}_{mes})} \tag{5.24}$$

where

$$p(\theta_{mes}(k) \mid m,l,\theta_{mes}^{k-1}) = p(z_l(k) \mid m,\theta_{mes}^{k-1}) = \mathcal{N}(z(k);\hat{z}_m(k),S_{m,l}(k)) \tag{5.25}$$

$$\hat{z}_m(k) = \sum_{i=1}^{3} \mu^i_{IMM,m}(k)\hat{z}^i_m(k)\,. \tag{5.26}$$

In (5.26) ,

$$\hat{z}^i_m = C\hat{\xi}^i_m(k \mid k-1) \tag{5.27}$$

is the measurement prediction for i^{th} IMM model of m^{th} component. In this expression $\hat{z}_m(k)$ stands for the measurement prediction of the m^{th} Gaussian which is composed of the predictions, $\hat{z}^i_m(k)$, of sub-models of the inherent IMM filter, based on the probability, $\mu^i_{IMM,m}$. Similarly,

$$S_{m,l}(k) = \sum_{i=1}^{3} \mu^i_{IMM,m}(k)\begin{pmatrix} S^i_{m,l}(k) + \\ [\hat{z}^i_m(k) - \hat{z}_m(k)][\hat{z}^i_m(k) - \hat{z}_m(k)]^T \end{pmatrix}. \tag{5.28}$$

Overall filter output

We use mixture probabilities for generating overall RPIMMGM filter output as a combination of the mixture components based on their probabilities:

$$\hat{\xi}_{RPIMMGM}(k \mid k) = \sum_{i=1}^{N_M(k)} \mu_{GM,m'}(k)\hat{\xi}_{m'}(k \mid k), \tag{5.29}$$

$$P_{RPIMMGM}(k \mid k) = \tag{5.30}$$

$$\sum_{i=1}^{N_M(k)} \mu_{M,m'}\begin{pmatrix} P_{m'}(k \mid k) + \\ [\hat{\xi}_{m'}(k \mid k) - \hat{\xi}_{RPIMMGM}(k \mid k)][\hat{\xi}_{m'}(k \mid k) - \hat{\xi}_{RPIMMGM}(k \mid k)]^T \end{pmatrix}.$$

Since every measurement interval creates pseudo measurements in Cartesian coordinates, prediction and update cycles can be done in a standard Kalman filter fashion for the models of IMM.

5.4 Simulations and Comments

A 2-D BOT is generated where ownship makes constant velocity motion with multiple legs. Monte Carlo simulations are performed for a scenario where angle measurement errors are randomly generated for each run. In the experiments comparisons are made with a standard IMM filter. The proposed RPIMMGM BOT tracker has much smaller average error for position and velocity states. Both trackers have 4 state components representing position and velocities in 2-D.

For the simulation 100 Monte Carlo runs are performed. Number of range intervals are 6 for range parameterization. Maximum 12 Gaussian mixture is allowed during the tracking. For the IMM turn models 3 mrad/sec positive and negative turns are used together with the (constant velocity) no turn model. Target is located 9 km north and 10.5 km east of the ownship having speeds of 7.7 m/s in each direction at the initialization. Target start to a positive coordinated turn of 3 mrad/sec at the 30^{th} step of the track. Model transition probabilities are 0.9 for staying on the same mode and 0.05 for transitioning any one of the remaining modes.

The simulation results are shown in Figure 10 to Figure 13. Figure 10 shows the trajectories of the target and ownship as well as one of the state estimate trajectory of 100 Monte Carlo runs.

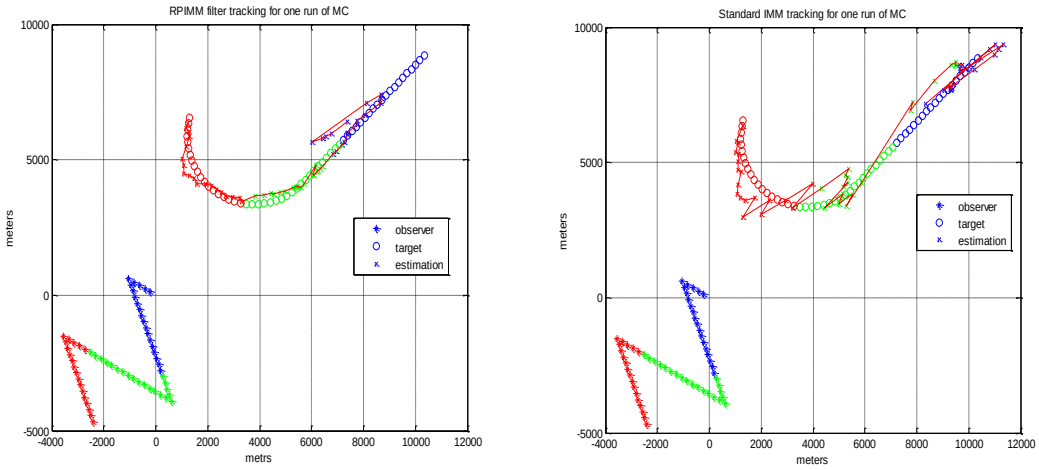


Figure 10- Comparison of position trajectory estimate (red line) accuracy in one of the Monte Carlo simulations. RPIMMGM left, standard IMM right (First 20 steps are blue, next 20 steps are green, last 20 steps are red both for the target and the observer).

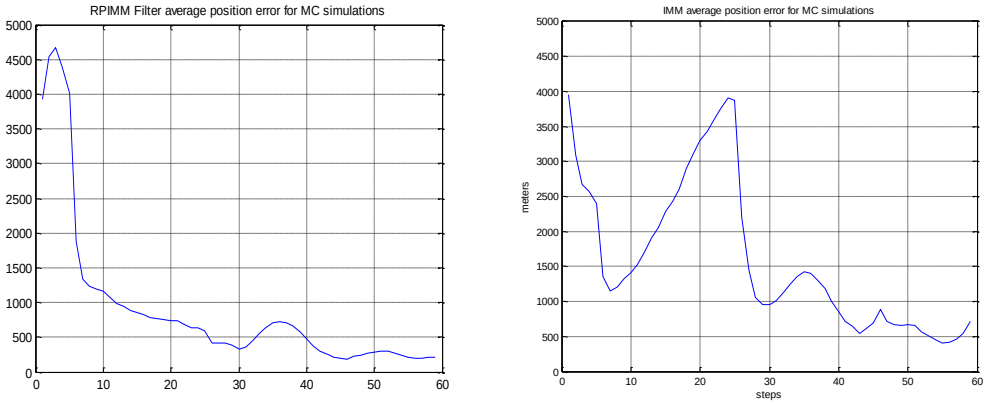


Figure 11- Average position error of 100 Monte Carlo runs (RPIMMGM left, standard IMM right)

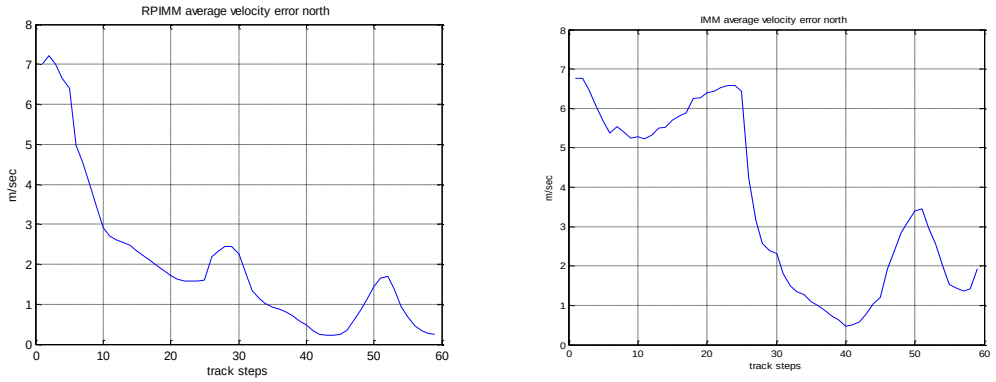


Figure 12- RMS velocity error in north direction for 100 Monte Carlo runs (RPIMMGM left, standard IMM right)

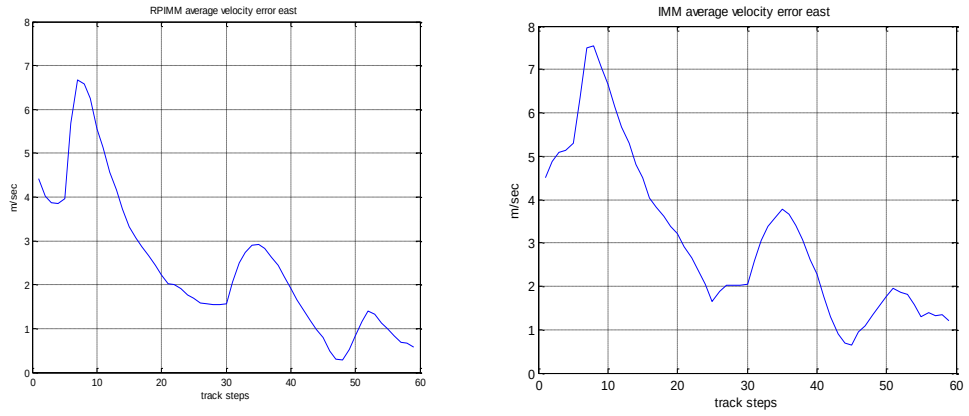


Figure 13- RMS velocity error in east direction for 100 Monte Carlo runs (RPIMMGM left, standard IMM right)

Figure 11 reveals several phenomena. If the position error is investigated it is observed that for both filters position error starts to increase again after 30^{th} step. This is because of the start of the target maneuver. Ownship maneuvers are at 5^{th} , 25^{th}, and 45^{th} steps. Especially for the IMM filter the position error gradually increases between 8^{th} to 25^{th} steps and suddenly decreases because of the observability obtained by the ownship maneuver at 25^{th} step. Also for the RPIMMGM filter a similar behavior is seen after the 5^{th} step. RPIMMGM position and velocity errors are clearly less than the IMM filter errors. This shows that using RP target estimate is better adapted to the target maneuvers.

When the position error starts to decrease by the ownship maneuver, velocity error increases for some time interval. This is because the filter closes the gap between the estimation and the true target position which is seen as a change in the speed components during the corrective move of the filter output trajectory.

It should also be stated that range parameterization have to be applied such that it covers the estimated target position and its reasonable neighborhood. If it is applied to a very large interval, the resulting pseudo measurements can mislead the tracker which increase estimation error for RPIMMGM.

RPIMMGM filter has higher computational complexity compared to IMM, however the performance increase gained is believed to be more valuable for most of the real problems.

CHAPTER 6

RANGE PARAMETERIZED COORDINATED TURN RATE MARGINALIZED PARTICLE FILTER

6.1 Problem Statement

In this section a novel algorithm is proposed for tracking maneuvering targets in bearings-only framework. Track accuracy loss based on the difficulties like range unobservability and target model switching based on maneuver is tried to be minimized by using a marginalized particle filter.

In this study, it is observed that even there is a range ambiguity, in other words even possible target trajectories have different slant ranges relative to the observer, same bearing measurements are observed for targets having the same turn rate value, ω, as seen in Figure 14. Hence, knowing that the target can make coordinated turns with a variable unknown but constrained turn rate, the turn rate can be modeled as an additional state variable similar to a bulk motion component of the state vector for a group of targets. In the succeeding section it will be demonstrated that based on the developed model, the turn rate component can also be marginalized with a Rao-blackwellized particle filter.

The algorithm developed in this Chapter is based on the Range Parameterized Gaussian Mixture filtering [15, 16] in which there are multiple sub-tracks that are obtained by the pseudo range measurements. Each sub-track has a different range value. We can use the estimated turn rate value similar to bulk turn rate parameter used for tracking of group targets [23, 13] and come up with an augmented filtering model.

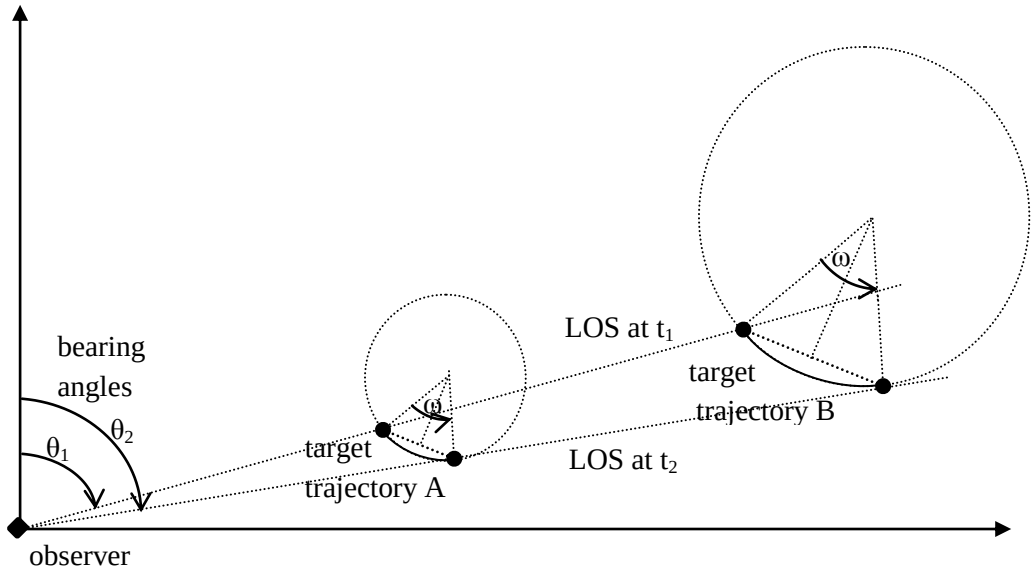


Figure 14- Same bearing angle measurements *(θ_1* at time *t_1* and *θ_2* at time *t_2*) can originate from targets having the same coordinated turn value, *ω*, with different ranges. Hence the range ambiguity is still present. Target trajectory can be either A or B.

The advantage gained by this fact is the applicability of a particle filter having only a single (nonlinear) state component since after the Rao-Blacwellization system can be reduced to this form. This is the motivating factor of the filter developed in this Chapter, the Range Parameterized Coordinated Turn Rate Marginalized Particle Filter (RPCTMPF). The details will be given in the following sections starting by the model description.

6.2 Tracking Model

The target is assumed to have a model given by

$$\xi_{k+1} = A(\omega_k)x_k + v_{k,x} \tag{6.1}$$

where ω_k is the current turn rate at time *k*, and the coordinated turn relation is given by

$$A(\omega_k) = \begin{bmatrix} 1 & \frac{\sin \omega_k T}{\omega_k T}.T & 0 & \frac{-(1-\cos \omega_k T)}{\omega_k T}.T \\ 0 & \cos \omega_k T & 0 & -\sin \omega_k T \\ 0 & \frac{(1-\cos \omega_k T)}{\omega_k T}.T & 1 & \frac{\sin \omega_k T}{\omega_k T}.T \\ 0 & \sin \omega_k T & 0 & \cos \omega_k T \end{bmatrix} \tag{6.2}$$

T, being the sampling interval.

The state vector is

$$\xi_k = \begin{bmatrix} x_{k,1} & x_{k,2} & x_{k,3} & x_{k,4} \end{bmatrix}^T \tag{6.3}$$

here $x_{k,1}$ and $x_{k,3}$ are target x and y positions and $x_{k,2}$ and $x_{k,4}$ are target velocities on x and y directions, respectively, with respect to an inertial frame.

The turn rate ω_k is modeled as:

$$\omega_k = \omega_{k-1} + n_{k,\omega} \tag{6.4}$$

where $n_{k,\omega}$ is distributed normally. It must also be stated that ω_o and $n_{k,\omega}$ need not necessarily be normal distributions. If a better initial distribution is available, like the uniform distribution covering the maximum turn rate, it can also be used. This relaxation is due to the fact that a Rao-Blackwellized Particle Filter based on the marginalization of the diagonal model given in (2.63) is used during tracking where the turn rate is considered as the nonlinear part of the state vector.

The implemented filter is a Gaussian mixture filter due to range parameterization and track splitting [15, 16]. Hence a Gaussian mixture representation of the state is obtained at each iteration with sub-tracks being the mixture components:

$$p(\xi_{k-1} \mid Z^{k-1}) = \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) p(\xi_{k-1} \mid c, Z^{k-1}) \tag{6.5}$$

where $\gamma_{k-1}(c)$ is the probability that 'sub-track c is true given the measurements up to *k-1*. $N_{k-1,c}$ is the total number of track components at *k-1* and $p(\xi_{k-1} \mid c, z^{k-1})$ is the state pdf given that the component *c* is true.

This Gaussian Mixture representation results from range parameterization of the bearing measurement at initialization as well as during the tracking period. Similar to RPIMMGM filter, bearing measurement is converted to pseudo measurements on the line-of-sight determined by the measured bearing angle. Multiple ranges are determined based on geometric progression (Section 3.4). The multiple pseudo measurements result in multiple sub-tracks. In the filter proposed here range parameterization is carried out at each new measurement so that a better Gaussian Mixture approximation is obtained.

Based on the target model (2.1) each track component c is described as

$$\xi_{k+1}^{c} = A(\omega_k)\xi_k^c + v_{k,\xi}^c \tag{6.6}$$

where $v_{k,\xi}^c$ has identical distribution for all *c*.

At time *k* we have the sub-measurement set Z_k:

$$\mathrm{Z_k} = \left\{\mathrm{z_{k,g}}; \mathrm{g} = 1,..., \mathrm{N_{k,g}}\right\} \tag{6.7}$$

and

$$\mathrm{Z^k} = \left\{\mathrm{Z_k}; \mathrm{t} = 0,..., \mathrm{k}\right\} \tag{6.8}$$

where $N_{k,g}$ is the number of sub-range measurements at *k* after range parameterization. We have the measurement model as

$$z_{k,g} = H\xi_k^c + w_{k,z} \tag{6.9}$$

where index *g* is for the g^{th} sub-measurement and *c* is for the c^{th} sub-track. Also *H* is given as

$$\mathrm{H}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}. \tag{6.10}$$

At time k we have $N_{k,g}$ measurements and $N_{k-1,c}$ components that are present for each particle. Therefore, for each particle i, $N_{k,g}\times N_{k-1,c}$ sub-filters are to be run concurrently to update the system. The filter estimates the stacked vector defined as

$$\mathrm{s}_{\mathrm{k}}^{\mathrm{T}}=[\xi_{\mathrm{k}}^{\mathrm{T}} \quad \omega_{\mathrm{k}}] \tag{6.11}$$

(6.4), (6.6), (6.9) and (6.11) describes system and the filter model completely. Once ω_k is estimated the system equation (6.6) becomes linear. Measurement equation (6.9) is linear based on the parameterized range measurements, however ω_k is used in (6.6) for the calculation of the A matrix as a nonlinear term. This model is a reduced one of general form given in equations (2.62) which separates the state vector into linear, ξ_k^l, and nonlinear, ξ_k^n, sub components. The general model is repeated here for convenience.

$$x_{k+1}^n = f_k^n\left(x_k^n\right)+A_k^n\left(x_k^n\right)x_k^l+G_k^n\left(x_k^n\right)v_k^n \tag{6.12a}$$

$$x_{k+1}^l = f_k^l\left(x_k^n\right)+A_k^l\left(x_k^n\right)x_k^l+G_k^l\left(x_k^n\right)v_k^l \tag{6.12b}$$

$$y_k = h_k\left(x_k^n\right)+C_k\left(x_k^n\right)x_k^l+w_k \tag{6.12c}$$

Note that for our problem

$x_k^n=\omega_k, \quad x_k^l=\xi_k^c, \quad f_k^n(x_k^n)=\omega_k, \quad A_k^n(x_k^n)=0, \quad G_k^n(x_k^n)=1, \quad f_k^l(x_k^n)=0,$ $A_k^l(x_k^n)=A(\omega_k), G_k^n(x_k^n)=1.$

According to the above formulation the structure has “diagonal type” as given in equation (2.63). For the details of Rao-Blackwellized particle filter the reader may refer to Chapter 2.

For our problem the marginalization can be applied as

$$p(\xi_k, \Omega^k \mid Z^k) = \underbrace{p(\xi_k, \mid \Omega^k, Z^k)}_{KF} \underbrace{p(\Omega^k \mid Z^k)}_{PF} \tag{6.13}$$

where the first part of the equation (6.13) is carried out by KF iterations and the second part is estimated with a PF. Here

$$\Omega^k = \{\omega_i; i = 0, \ldots, k\}. \tag{6.14}$$

The KF iterations are applied based on linear equations (6.6) and (6.9) for each particle *i* as

$$\hat{\xi}_{k|k,c,g}(i) = \hat{\xi}_{k|k-1,c,g}(i) + K_{k,c,g}(i)\left\{z_{k,g} - H\hat{\xi}_{k|k-1,c,g}(i)\right\} \tag{6.15}$$

$$P_{k|k,c,g}(i) = P_{k|k-1,c}(i) - K_{k,c,g}(i) S_{k,c,g}(i) K^T_{k,c,g}(i) \tag{6.16}$$

$$S_{k,c,g}(i) = H P_{k|k-1,c}(i) H^T + R_{k,g} \tag{6.17}$$

$$K_{k,c,g}(i) = P_{k|k-1,c}(i) H^T S^{-1}_{k,c,g}(i) \tag{6.18}$$

And

$$\hat{\xi}_{k|k-1,c}(i) = A(\omega_{k-1}) \hat{\xi}_{k-1|k-1,c}(i) \tag{6.19}$$

$$P_{k|k-1,c}(i) = A(\omega_{k-1}) P_{k-1|k-1,c}(i) A(\omega_{k-1})^T + Q_{k,n_x}\ . \tag{6.20}$$

A Sequential Importance Resampling Particle Filter (SIRPF) is used in this problem. Therefore $p(\omega_k \mid \Omega^{k-1}, Z^{k-1})$ is to be used as importance function for an SIR type particle filter. Then for the particle weight updates the likelihood function $p(Z_k \mid \Omega^k, Z^{k-1})$ is required as described in SIRPF. Samples are drawn using equation (6.4).

Range parameterization approximates measurement with a probabilistic distribution function as

$$p(Z_k \mid \Omega^k, \xi_k(i)) \approx \sum_{g=1}^{N_{k,g}} \alpha_{k,g} N(z_{k,g}; H\xi_k(i), R_{k,g}) \tag{6.21}$$

where

$$Z_k = \{z_{k,g}; g = 1, \ldots, N_{k,g}\} \tag{6.22}$$

is the set of pseudo measurements after range parameterization. For a particle *i*, the turn rate is described as $\omega_k(i)$ at time *k*, and $\alpha_{k,g}$ is the probability of g^{th} pseudo-measurement is true as defined in range parameterization method. Prediction for the sub-state ξ_k of the complete state (6.11) is a combination of Gaussians:

$$\begin{aligned} p(\xi_k(i) \mid \Omega^k, Z^{k-1}) &= \int_{\xi_{k-1}(i)} p(\xi_k(i) \mid \xi_{k-1}(i), \Omega^k)\, p(\xi_{k-1}(i) \mid \Omega^{k-1}, Z^{k-1}) d\xi_{k-1}(i) \\ &= \int_{\xi_{k-1}(i)} p(\xi_k(i) \mid \xi_{k-1}(i), \Omega^k) \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c)\, p(\xi_{k-1}(i) \mid c, \Omega^{k-1}, Z^{k-1}) d\xi_{k-1}(i) \\ &= \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) \int_{\xi_{k-1}(i)} p(\xi_k(i) \mid \xi_{k-1}(i), \Omega^k)\, p(\xi_{k-1}(i) \mid c, \Omega^{k-1}, Z^{k-1}) d\xi_{k-1}(i) \\ &= \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) p(\xi_k(i) \mid c, \Omega^k, Z^{k-1}). \end{aligned} \tag{6.23}$$

For the likelihood below,

$$p(Z_k \mid \Omega^k, Z^{k-1}, i) = \int_{\xi_k(i)} p(Z_k \mid \Omega^k, \xi_k(i))\, p(\xi_k(i) \mid \Omega^k, Z^{k-1}) d\xi_k(i) \tag{6.24}$$

using (6.21) and (6.23) we get,

$$p(Z_k \mid \Omega^k, Z^{k-1}, i) = \int_{\xi_k(i)} \sum_{g=1}^{N_{k,g}} \alpha_{k,g}\, p(z_{k,g} \mid \Omega^k, \xi_k(i), g) \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) p(\xi_k(i) \mid c, \Omega^k, Z^{k-1}) d\xi_k(i)$$

$$= \sum_{g=1}^{N_{k,g}} \alpha_{k,g} \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) \int_{\xi_k(i)} p(z_{k,g} \mid \Omega^k, \xi_k(i), g)\, p(\xi_k(i) \mid c, \Omega^k, Z^{k-1}) d\xi_k(i)$$

$$= \sum_{g=1}^{N_{k,g}} \alpha_{k,g} \sum_{c=1}^{N_{k-1,c}} \gamma_{k-1}(c) p(z_{k,g} \mid \Omega^k, c, Z^{k-1}) . \tag{6.25}$$

Then for each particle *i*, the likelihood function (6.25) should be calculated where

$$\mu_{k,c,g}(i) = p(z_{k,g} \mid \Omega^k, c, Z^{k-1}) = N(z_{k,g}; \hat{\xi}_{k|k-1,c}(i), S_{k,c,g}(i)) \tag{6.26}$$

This modeling method of the problem facilitates the application of a particle filter. By using marginalization heavy computation required for a complete state particle filter is removed, but we are still able to get the advantage of using a particle filter to increase filter accuracy. The cost of the increase in the number of Kalman Filters by a factor of the number of particles is to be mentioned which is the case when Rao-Blackwellization (marginalization) is used. Also it must be emphasized that the augmentation of the turn rate parameter leads to a bearings-only target tracker capable of tracking maneuvering targets.

6.3 One Cycle of the Filter

Each cycle of the algorithm for the filter can be described as an update:

$$\left[\left\{ \omega_k(i), n_k(i), \left\{ \gamma_k^c(i), \xi_{k|k}^c(i), P_{k|k}^c(i) \right\}_{c=1}^{N_{k,c}(i)} \right\}_{i=1}^{N_P} \right] = RPCTMPF \quad \left[\left\{ \omega_{k-1}(i), n_{k-1}(i), \left\{ \gamma_{k-1}^c(i), \xi_{k-1|k-1}^c(i), P_{k-1|k-1}^c(i) \right\}_{c=1}^{N_{k-1,c}(i)} \right\}_{i=1}^{N_P}, Z_k \right]$$

The details of the cycle is given as:

Particle Filter Time Update

- Sample turn rate particles $\omega_k(i) \sim \mathcal{N}(\omega_k(i); \omega_{k-1}(i), Q_\omega)$ for $i=1,...,N_p$ where N_p is the number of particles.

Kalman Filter Time Update

- $\hat{\xi}_{k|k-1,c}(i) = A(\omega_{k-1}(i)) \hat{\xi}_{k-1|k-1,c}(i)$
- $P_{k|k-1,c}(i) = A(\omega_{k-1}(i)) P_{k-1|k-1,c}(i) A(\omega_{k-1}(i))^T + Q_\xi$

  for $i=1,...,N_p$, for $c=1,...,N_{k-1,c}(i)$ where $N_{k-1,c}(i)$ is the number of track components at time $k-1$ for i^{th} particle .

Particle Filter Measurement Update

- Calculate the likelihood function

$$\mu_{k,c,g}(i) = p(z_{k,g} \mid \hat{\xi}_{k|k-1,c}(i), Z^{k-1})$$

for $i=1,...,N_p$, $c=1,...,N_{k-1,c(i)}$, $g=1,...,N_{k,g}$, where $N_{k,g}$ is the number of range partitions at time k.

- Calculate normalized particle weights

$$n_k(i) = \frac{\sum_c \sum_g \mu_{k,c,g}(i)\gamma_{k-1,c}(i)\alpha_{k,g}}{\sum_i \sum_c \sum_g \mu_{k,c,g}(i)\gamma_{k-1,c}(i)\alpha_{k,g}}$$

where $\alpha_{k,g}$ is the probability of g^{th} pseudo-measurement is true as defined in range parameterization.

- Resample using updated particle weights.

Kalman Filter Measurement Update

- Use the Kalman Filter measurement update equations as

$$\hat{\xi}_{k|k,c,g}(i) = \hat{\xi}_{k|k-1,c,g}(i) + K_{k,c,g}(i)\left\{z_{k,g} - H\hat{\xi}_{k|k-1,c,g}(i)\right\}$$

$$P_{k|k,c,g}(i) = P_{k|k-1,c}(i) - K_{k,c,g}(i)S_{k,c,g}(i)K^T_{k,c,g}(i)$$

Mixture Probability Calculation

- For each particle *i* there is a Gaussian mixture representation where each mixture component has a probability of being true. This probability values are updated:

$$\gamma_{k,c,g}(i) = \frac{\mu_{k,c,g}(i)\gamma_{k-1,c}(i)\alpha_{k,g}}{\sum_c \sum_g \mu_{k,c,g}(i)\gamma_{k-1,c}(i)\alpha_{k,g}}$$

- The new track components $\{c, g\}$ are indexed as $\{c^+\}$.

Mixture Reduction

- Based on the mixture probability values, for Gaussian Mixture representation of each particle, mixture reduction is applied to reduce the number of components. Mixture reduction is done by removing less likely ones. Reduced set of components are re-indexed as *c* for the next iteration.

6.4 Simulations

A 2-D BOT scenario is generated where own-craft makes legged constant velocity motion similar to the one in previous Chapter. The accuracy of the tracker depends on several factors. Ownship maneuvers increase estimation performance whereas model switching of the target severely affects the performance because of the nature of the bearings-only measurement. Therefore it is easy to detect a coordinated turn maneuver of the target if the ownship happens to make one or preferably more than one maneuver especially right after the target maneuver. However there is no correlation between these two events and tracking error can increase during the tracking if target maneuvers are either large or more frequent than the ownship maneuvers.

A difficult scenario is generated and Monte Carlo simulations are performed. The new algorithm, namely Range Parameterized Coordinated Turn Rate Marginalized Particle Filter (RPCTMPF) and the one described in previous Chapter, the Range-Parameterized Interacting Multiple Model Gaussian Mixture Filter (RPIMMGM) are compared. RPIMMGM was already compared to an IMM filter and better tracker performance was observed compared to the IMM filter. The aim here is using the Rao- Blackwellization to have a superseding filter compared to RPIMMGM filter.

The scenario is such that the target have a constant velocity motion during the start, up to 30^{th} step, then it starts to make a positive coordinated turn with 1.5 mrad/s and keeps turning for 30 steps. For this simulation, the number of sub ranges for range parameterization is taken as 6. Maximum 12 Gaussian mixture components are allowed. Less likely components are removed to keep 12 components. 500 particles are used for the marginalized particle filter. At initialization, range parameterization is done between 10 km to 15 km. During the tracking, range parameterization is

applied for the range interval of 8 km around the estimated range. Target is located 9 km north and 10.5 km east of the ownship at initial time. Target has velocity components of 7.7 m/s in each direction at initialization. At the initialization of the filter, both speed components of the state estimate are taken as 0.

The RPIMMGM filter is also initialized the same way as the RPCTMPF. All tracker parameters are set equal to the RPCTMPF to be able to get results for performance comparison.

6.5 Results and Comments

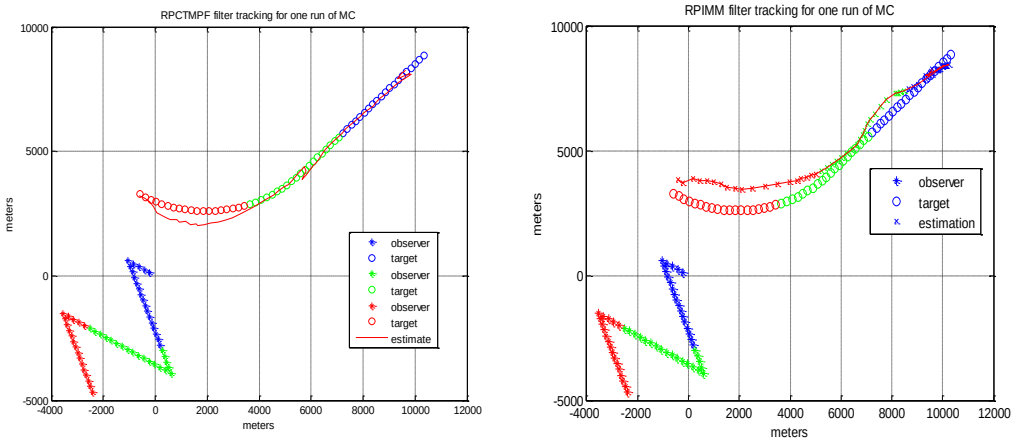


Figure 15- Trajectories corresponding to different Monte Carlo runs. RPCTMPF is on the left and RPIMMGM is on the right.

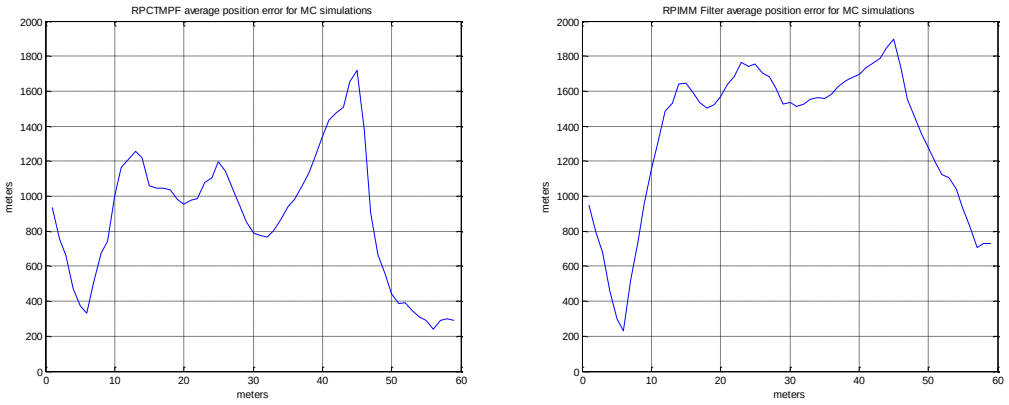


Figure 16- Average position errors for Monte Carlo runs. RPCTMPF is on the left and RPIMMGM is on the right. Clearly before the maneuver at 30th step position error is less than 1200 m for RPCTMPF where for RPIMMGM this is around 1600 m. Also after the

maneuver RPCTMPF adapts better so that the error goes to 300 m where for RPIMMGM this is approximately 700 m.

Figure 15 shows tracking trajectories for one run of RPCTMPF and RPIMMGM filter. Single trial results are not so informative therefore the average position error results given in Figure 16 can be investigated. For both filter after the maneuver at 30^{th} step, error is increased although the error for RPCTMPF is not as much as the other one. The sharp decrease in error after 45^{th} step is because of the observer maneuver at that instance. The final average position error for RPCTMPF reduces up to 300 m whereas for the RPIMMGM the final average error is over 700 m. The overall performance of the RPCTMPF is clearly better than RPIMMGM. The average error is also much less than RPIMMGM for the first 30 step of the tracking. For this period RPCTMPF has an error less than 1200 m whereas RPIMMGM has an error around 1600 m.

The better estimation results indicate that using particle filtering for turn rate estimation resulted in better maneuver adaptation and motion modeling.

For RPCTMPF adaptation to the maneuver is possible only if the turn rate particles can adapt to the maneuver. This can be seen from their distribution during tracking (Figure 17). During the simulations it is observed that the turn rate noise model in (6.4) should be adjusted such that the explained adaptation of particles are possible. The pure Gaussian noise model is modified as a summation of a Gaussian and a uniform density. The uniform part has the weight 0.1. This relaxes the particles which represent the turn rate value. It has an effect of increasing the tail probability for the Gaussian distribution to favor maneuver possibility.

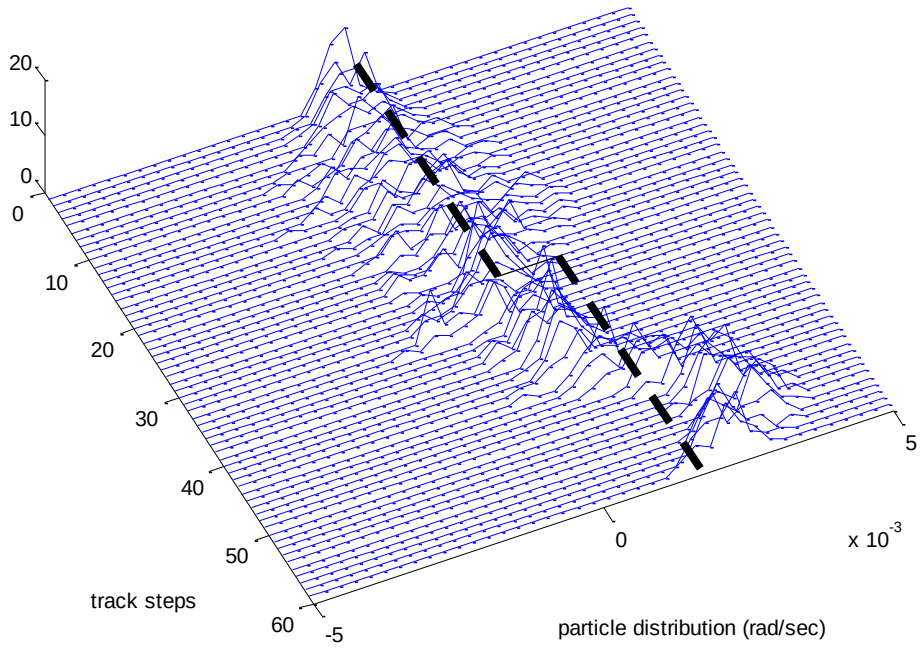


Figure 17- Turn rate particle distribution during tracking for one run of the Monte Carlo simulation given in Figure 15. As the target makes a positive turn of 1.5 mrad after 30th step, the adaptation of the particles to the new maneuver is visible. The dashed lines represent the true turn rate values.

RPCTMPF algorithm utilizes particle filtering. To decrease computational cost of the particle filter, Rao-Blackwellization is applied thanks to the model developed here where turn rate is augmented as a bulk turn motion. In other words, Rao-Blackwellization is possible by modeling target motions as coordinated turn (which is true almost for all real vehicles) and by using the fact explained in Figure 14. The particle filter estimates the turn rate. For each particle we have a Gaussian mixture representation of the target state. This representation is based on the nature of the range parameterization which represents range measurement as a Gaussian mixture of sub-measurements. Gaussian mixture based tracking is a good way of coping with nonlinearity.

Another factor increasing the overall tracking performance is to dynamically adapt range limits for range parameterization that we apply in the simulations. As long as the tracker accurately estimates the target position, this value can be used in pseudo measurement generation of the range parameterization. This way the tracker is not fed by the unrealistic wide range interval of pseudo measurements. This results in a better state estimate and maneuver adaptation. In the simulations performed, range parameterization is applied around the estimated range for the actual tracking step with an interval of 8 km.

CHAPTER 7

OPTIMIZATION OF OBSERVER MOTION BASED ON MUTUAL INFORMATION

This chapter is devoted to different methods of forming and solving the optimization problem where the objective function is based on the mutual information. In the next section we use the approximate mutual information for a time horizon of length one. The approximation is based on the linearization of the measurement equation and the method uses the mutual information formulation derived for the linear systems. In sections 7.3 and 7.4 nonlinear recursive form of the mutual information is used for a time horizon of length one again. Finally in section 7.5 a dynamic programming approach is developed for an arbitrary time horizon however with some restrictions on the input.

7.1 One Step Optimization of BOT using Linear System Mutual Information Formulation

Equation (4.12) defines the mutual information between the measurement and the state sequences in terms of the system parameters. When the sequence depth is one the components of equations (4.13), (4.14) and (4.15) reduce to the ones given in (4.6) and (4.7) as

$$I(\xi_k, z_k) = \frac{1}{2}\log\frac{\left|CA\Sigma_0 A^T C^T + CGQG^T C^T + HRH^T\right|}{\left|HRH^T\right|}. \tag{7.1}$$

Here for BOT problem H and R are scalars where without loss of generality H=1, so

$$HRH^T = R. \tag{7.2}$$

Σ_0 is the initial covariance matrix at *k=0*.

The state vector for BOT is repeated here,

$$\xi_k = \begin{bmatrix} y_k & x_k & v_{y,k} & v_{x,k} \end{bmatrix}^T = \begin{bmatrix} y_k & x_k & \dot{y}_k & \dot{x}_k \end{bmatrix}. \tag{7.3}$$

Having the position components as the first two components and the velocity components as the last two of the state vector, simplifies the notation by defining

$$p_k = \begin{bmatrix} y_k & x_k \end{bmatrix}^T \tag{7.4}$$

and

$$\kappa_k = \begin{bmatrix} v_{y,k} & v_{x,k} \end{bmatrix}^T. \tag{7.5}$$

In the remaining part of this section the derivations are done for the model that uses accelerations of the ownship in both directions as the input vector as given in (3.12). The state equation for this case is repeated here

$$\xi_{k+1} = A\xi_k + G_k v_k - G_k u_k = \begin{bmatrix} y_k & x_k & v_{y,k} & v_{x,k} \end{bmatrix}^T. \tag{7.6}$$

Here $\{v_k\}$ is the 2 dimensional i.i.d process noise vector with $v_k \sim N(v_k; 0, Q)$, appearing as the acceleration of the target in each dimension. Let *T* be the sampling interval, then

$$A_k = \begin{bmatrix} I_2 & TI_2 \\ 0 & I_2 \end{bmatrix} = A \tag{7.7}$$

with

$$G_k = \begin{bmatrix} \frac{T^2}{2} I_2 \\ TI_2 \end{bmatrix} = G \tag{7.8}$$

where

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{7.9}$$

Then state equation can be rewritten as

$$\begin{bmatrix} p_k \\ v_k \end{bmatrix} = A_{k-1} \begin{bmatrix} p_{k-1} \\ \kappa_{k-1} \end{bmatrix} + G_{k-1} v_{k-1} - G_{k-1} u_{k-1} . \tag{7.10}$$

The position components can be written explicitly as

$$p_k = p_{k-1} + T I_2 \kappa_{k-1} + \frac{T^2}{2} I_2 v_{k-1} - \frac{T^2}{2} I_2 u_{k-1} . \tag{7.11}$$

The nonlinear measurement of BOT is repeated here as

$$y_k = c(\xi_k) + H w_k \tag{7.12}$$

$$c(\xi_k) = \arctan(\frac{x_k}{y_k}) \tag{7.13}$$

where $w_k \sim N(w_k; 0, R)$ is the random noise in the measurement. We define C_k as the local linearization of the nonlinear function $c(\xi_k)$, which is the Jacobian of it at the point $\xi_{k|k-1} = E(\xi_k \mid [z_1, z_2, \ldots z_{k-1}])$. That is,

$$C_k^{\ T}(\xi_{k|k-1}) = \frac{\partial h(\xi)}{\partial \xi}\Big|_{\xi_{k|k-1}} = \begin{bmatrix} \frac{\partial h(\xi_k)}{\partial y_k} \\ \frac{\partial h(\xi_k)}{\partial x_k} \\ \frac{\partial h(\xi_k)}{\partial v_{y,k}} \\ \frac{\partial h(\xi_k)}{\partial v_{x,k}} \end{bmatrix}\Big|_{\xi_{k|k-1}} = \begin{bmatrix} \frac{-x_{k|k-1}}{x_{k|k-1}^2 + y_{k|k-1}^2} \\ \frac{y_{k|k-1}}{x_{k|k-1}^2 + y_{k|k-1}^2} \\ 0 \\ 0 \end{bmatrix} \tag{7.14}$$

Let $Z = [z_1\ z_2, \ldots, z_k\ ]$ and $\Xi = [\xi_1\ \xi_2, \ldots, \xi_k\ ]$ denote the measurement and the state sequences. To maximize $I(\Xi, Z)$ the term inside the log function of (7.1) should be maximized. This is obviously the same as the maximization of the original function

since $\log(.)$ is a strictly increasing function. The first term of the numerator is calculated as

$$C_k(\xi_{k|k-1})A = \frac{1}{x_{k|k-1}^2 + y_{k|k-1}^2}\begin{bmatrix} -x_{k|k-1} & y_{k|k-1} & 0 & 0 \end{bmatrix}\begin{bmatrix} I_2 & TI_2 \\ 0 & I_2 \end{bmatrix} \tag{7.15}$$

$$= \frac{1}{n_k}\begin{bmatrix} -x_{k|k-1} & y_{k|k-1} & -Tx_{k|k-1} & Ty_{k|k-1} \end{bmatrix}$$

where,

$$n_k \equiv x_{k|k-1}^2 + y_{k|k-1}^2 \tag{7.16}$$

So,

$$C_k(\xi_{k|k-1})A\Sigma_0 A^T C_k(\xi_{k|k-1})^T = \tag{7.17}$$

$$\frac{1}{n_k^2}\begin{bmatrix} -x_{k|k-1} & y_{k|k-1} & -Tx_{k|k-1} & Ty_{k|k-1} \end{bmatrix}\begin{bmatrix} \Sigma_{0,1} & \Sigma_{0,2} \\ \Sigma_{0,2}^T & \Sigma_{0,3} \end{bmatrix}\begin{bmatrix} -x_{k|k-1} & y_{k|k-1} & -Tx_{k|k-1} & Ty_{k|k-1} \end{bmatrix}^T$$

$$= \frac{1}{n_k^2}\left[\bar{c}_k \Sigma_{0,1} \bar{c}_k^{\,T} + T\bar{c}_k(\Sigma_{0,2} + \Sigma_{0,2}^T)\bar{c}_k^{\,T} + T^2 \bar{c}_k \Sigma_{0,3} \bar{c}_k^{\,T}\right] \tag{7.18}$$

where

$$\bar{c}_k \equiv \begin{bmatrix} -x_{k|k-1} & y_{k|k-1} \end{bmatrix} \tag{7.19}$$

and $\Sigma_{0,1}, \Sigma_{0,2}, \Sigma_{0,2}^T \Sigma_{0,3}$ are the 2x2 subcomponents of Σ_0 .

For the second term of the numerator of (7.1):

$$GQG^T = \begin{bmatrix} T^2 I_2 / 2 \\ TI \end{bmatrix} Q \begin{bmatrix} T^2 I_2 / 2 & TI \end{bmatrix} = \begin{bmatrix} \frac{T^4}{4}Q & \frac{T^3}{2}Q \\ \frac{T^3}{2}Q & T^2 Q \end{bmatrix} \tag{7.20}$$

$$C_k(\xi_{k|k-1})GQG^T C_k(\xi_{k|k-1})^T = \frac{1}{n_k^2} C_k(\xi_{k|k-1}) \begin{bmatrix} \frac{T^4}{4}Q & \frac{T^3}{2}Q \\ \frac{T^3}{2}Q & T^2 Q \end{bmatrix} C_k(\xi_{k|k-1})^T \tag{7.21}$$

$$= \frac{1}{n_k^2}\frac{T^4}{4} C_k(\xi_{k|k-1}) \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} C_k(\xi_{k|k-1})^T$$

$$= \frac{1}{n_k^2}\frac{T^4}{4} \bar{c}_k Q \bar{c}_k{}^T$$

The sum of the two terms (7.18) and (7.21) are,

$$= \frac{1}{n_k^2} \left[\bar{c}_k(\xi_k) \underbrace{\left(\Sigma_{0,1} + T(\Sigma_{0,2} + \Sigma_{0,2}^T) + T^2 \Sigma_{0,3} + \frac{T^4}{4} Q \right)}_{K} \bar{c}_k(\xi_k)^T \right] \tag{7.22}$$

$$\equiv \frac{1}{n_k^2} \bar{c}_k(\xi_k) K \bar{c}_k(\xi_k)^T$$

Finally adding the third term, the numerator of (7.1) is computed as

$$\frac{1}{n_k^2} \bar{c}_k(\xi_k) K \bar{c}_k(\xi_k)^T + R \tag{7.23}$$

The denominator and the second term of the numerator of (7.1) is R and it is independent of the relative position given by x_k and y_k. The maximization of the mutual information of equation (7.1) turns to

$$\max \quad \frac{1}{n_k^2} \bar{c}_k(\xi_k) K \bar{c}_k(\xi_k)^T \ . \tag{7.24}$$

If there were no constraints the trivial solution of the problem is to select $x_{k|k-1} = y_{k|k-1} = 0$, since the degree of the denominator, $n_k^2 \equiv (x_{k|k-1}^2 + y_{k|k-1}^2)^2$, is higher than the degree of the numerator. This means that the relative position of the target is zero, i.e., the optimum position of the ownship is the same as the position of the target. However for most of the cases this is not possible. Note that,

$$p_k = p_{k-1} + T\kappa_{k-1} - \frac{T^2}{2}u_{k-1} \tag{7.25}$$

So for p_k=0

$$u_{k-1} = \frac{2}{T^2}\left[p_{k-1} + T\kappa_{k-1}\right] \tag{7.26}$$

which may not be possible to satisfy with a bounded u_{k-1}. Note that the constraint that we have selected is $u_{k-1}^T u_{k-1} < \bar{\gamma}$. We can write the maximization problem in terms of u_{k-1} as follows. First note that

$$\bar{c}_k^T = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} p_k \equiv F p_k \tag{7.27}$$

and,

$$n_k = \bar{c}_k \bar{c}_k^T = p_k^T F^T F p_k = p_k^T p_k \tag{7.28}$$

$$\begin{aligned} n_k &= p_k^T p_k \\ &= \left[p_{k-1} + T\kappa_{k-1} - \frac{T^2}{2} I_2 u_{k-1} \right]^T \left[p_{k-1} + T\kappa_{k-1} - \frac{T^2}{2} I_2 u_{k-1} \right] \\ &= (a_k - \lambda_k)^T (a_k - \lambda_k) \end{aligned} \tag{7.29}$$

where

$$a_k = p_{k-1} + T\kappa_{k-1} \tag{7.30}$$

and

$$\lambda_k \equiv \frac{T^2}{2} I_2 u_{k-1} \,. \tag{7.31}$$

With the above definition the constraint takes the form $\lambda_k^T \lambda_k < \gamma$.

For some γ the objective function becomes:

$$\bar{c}_k(x_k) K \bar{c}_k(x_k)^T = \left[a - \lambda\right]^T F^T K F \left[a - \lambda\right] \tag{7.32}$$

And the reward function of (7.24) becomes

$$\max \quad \frac{1}{n_k^2}[a-\lambda]^T F^T KF[a-\lambda] \tag{7.33}$$

Since $F^T KF$ is a positive definite matrix there is an orthonormal matrix P, such that, $P^{-1}=P^T$ and $P^{-1}F^TKFP=D$ where D is diagonal. Therefore we can write the optimization problem as:

$$\max \quad \frac{(s-b)^T D(s-b)}{\left[(s-b)^T(s-b)\right]^2} \tag{7.34}$$

Where s=$P^{-1}\lambda$ and $b=P^{-1}a$, so that

$$\begin{aligned}(a-\lambda)^T F^T KF(a-\lambda) &= (Pb-Ps)^T F^T KF(Pb-Ps) \\ &= (s-b)^T P^T F^T KFP(s-b) = (s-b)^T D(s-b)\end{aligned} \tag{7.35}$$

and

$$\begin{aligned}(a-\lambda)^T(a-\lambda) &= (\lambda-a)^T PP^{-1}(\lambda-a) \\ &= (P^{-1}\lambda-P^{-1}a)^T(P^{-1}\lambda-P^{-1}a) = (s-b)^T(s-b)\end{aligned} \tag{7.36}$$

Let

$$D=\begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} \tag{7.37}$$

The maximization problem reduces to:

$$\max_{w_1,w_2} J(s_1,s_2) = \frac{\sum_{i=1}^{2} d_i(s_i-b_i)^2}{\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^2} \tag{7.38}$$

with a constraint defined for the components of s

$$s_1^2+s_2^2 \le \gamma_s \ . \tag{7.39}$$

The gradient of the reward function (7.38) is

$$\nabla J = \begin{bmatrix} \partial J / \partial s_1 \\ \partial J / \partial s_2 \end{bmatrix} \tag{7.40}$$

$$= \begin{bmatrix} \dfrac{2d_1(s_1-b_1)\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^2 - 2\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]2(s_1-b_1)\sum_{i=1}^{2}d_i(s_i-b_i)^2}{\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^4} \\ \dfrac{2d_2(s_2-b_2)\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^2 - 2\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]2(s_2-b_2)\sum_{i=1}^{2}d_i(s_i-b_i)^2}{\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^4} \end{bmatrix}$$

$$= \frac{2}{\left[\sum_{i=1}^{2}(s_i-b_i)^2\right]^3} \begin{bmatrix} (s_1-b_1)\left(d_1\left[\sum_{i=1}^{2}(s_i-b_i)^2\right] - 2\left[\sum_{i=1}^{2}d_i(s_i-b_i)^2\right]\right) \\ (s_2-b_2)\left(d_2\left[\sum_{i=1}^{2}(s_i-b_i)^2\right] - 2\left[\sum_{i=1}^{2}d_i(s_i-b_i)^2\right]\right) \end{bmatrix}$$

If there were no constraints then $s_1=b_1$ and $s_2=b_2$ makes the gradient zero, hence maximizing the mutual information. If $s=b$ then $\lambda=a$, meaning that the input should take the ownship to the target's current predicted position as stated before. The limitation of the input of the ownship generates the constraint:

$$s_1^2 + s_2^2 \le \gamma_s \ . \tag{7.41}$$

To solve the constrained problem we define the Lagrangian as

$$L(s_1, s_2, \lambda_{lag}) \equiv J(s_1, s_2) + \lambda_{lag}(s_1^2 + s_2^2 - \gamma_s) \ . \tag{7.42}$$

The gradient of the Lagrangian is the sum of the gradients of the objective function and the constraint.

The gradient of the constraint is

$$\nabla\lambda(s_1^2 + s_2^2 - 1) = 2\lambda_{lag}\begin{bmatrix} s_1 \\ s_2 \end{bmatrix}. \tag{7.43}$$

If we add (7.43) to (7.40) we get the gradient of the Lagrangian, $\nabla\ L(s_1, s_2, \lambda_{lag})$,

$$= \frac{1}{\left[\sum_{i=1}^{2}(s_i - b_i)^2\right]^3}\begin{bmatrix}(s_1 - b_1)\left[d_1\sum_{i=1}^{2}(s_i - b_i)^2 - 2\sum_{i=1}^{2}d_i(s_i - b_i)^2\right]\\(s_2 - b_2)\left[d_2\sum_{i=1}^{2}(s_i - b_i)^2 - 2\sum_{i=1}^{2}d_i(s_i - b_i)^2\right]\end{bmatrix} + 2\lambda_{lag}\begin{bmatrix}s_1\\s_2\end{bmatrix}. \quad (7.44)$$

The solution of the constrained optimization problem can be done in several different ways. Here we use the heuristic approach given below.

1. For the case where $\lambda_{lag}=0$, let $s_1 = b_1$ and $s_2 = b_2$. This is the optimal solution if

$$s_1^2 + s_2^2 \le \gamma_s \, . \quad (7.45)$$

2. For the case $\lambda_{lag} \neq 0$,

i-) Select $\lambda_{lag} > 0$ to obtain an iterative solution that maximizes the Lagrangian for the given λ_{lag} . Compute the value of the objective function.

ii-) Increase the value of λ_{lag} and repeat the procedure until the constraint is satisfied.

7.2 The Mutual Information as a Function of the Input for Recursive Formulation

In the previous section we have found the optimal input for a time horizon of depth one. During the optimization of the objective function we have used the linearized system. In this subsection our aim is to solve the same optimization problem by using directly the nonlinear model of the system. The mutual information between the state sequence, Ξ^k, and the measurement sequence, Z^k, was described in Section 4.4 as a recursive relation

$$I_{U^{k-1}}(\Xi^k, Z^k) = I_{U^{k-2}}(\Xi^{k-1}, Z^{k-1}) - h(z_k \mid \xi_k) + h_{U^{k-1}}(z_k \mid Z^{k-1}) \quad (7.46)$$

The availability of the recursive formulation of the mutual information and its significant value as a measure of the level of observability is the basic motivation of this section. The idea is to elaborate the recursive formulation to find a practical method to optimize inputs based on this criterion.

First, this measure is further elaborated by analyzing its gradient which opened a way to interesting practical applications especially for the case of one step optimization.

A simplified form of the recursive formulation is obtained to be used in finding the optimal input for the BOT problem. Based on this result a finite horizon optimization is formulated.

For the BOT problem the input can affect the ownship motion either as an acceleration vector or as a heading angle. The recursive formulation of the mutual information $I_{U^{k-1}}(\Xi^k, Z^k)$ as defined in equation (7.46) is composed of the mutual information at *k-1*, $I_{U^{k-2}}(\Xi^{k-1}, Z^{k-1})$, and the two extra terms.

$I_{U^{k-2}}(\Xi^{k-1}, Z^{k-1})$ is determined by the input sequence, U^{k-2}, and the corresponding measurement sequence Z^{k-1}. With the application of the input u_{k-1}, system evolves to step k and z_k is observed. Note that, $h(z_k \mid \xi_k)$ is independent of the control input sequence, U^k, and have no role on the maximization of the mutual information.

It is also observed that the term,

$$h_{U^{k-1}}(z_k \mid Z^{k-1}) = h(z_k \mid Z^{k-1}, U^{k-2}, u_{k-1}) \tag{7.47}$$

is dependent on u_{k-1}. The information content of $\{Z^{k-1}, U^{k-2}\}$ can be interchanged by the posterior pdf of the state at time *k-1* obtained by a particle filter. Maximization of the above defined term is possible by the proper selection of u_{k-1}.

The recursive formulation and the explanation above shows that other than the constant term, the mutual information value changes based on the last term of the recursive form, that is $h_{U^{k-1}}(z_k \mid Z^{k-1})$. This term is the only term that has u_{k-1} dependency. Therefore, the recursive maximization of $I_{U^{k-1}}(\Xi^k, Z^k)$ at time k is achieved by the maximization of $h_{U^{k-1}}(z_k \mid Z^{k-1})$ with respect to u_{k-1}. That is,

$$u_{k-1}^{opt} = \arg\max_{u_{k-1}} I_{U^{k-2}, u_{k-1}}(\Xi^k, Z^k) \tag{7.48}$$

$$= \arg\max_{u_{k-1}} I_{U^{k-2}}(\Xi^{k-1}, Z^{k-1}) - h(z_k \mid \xi_k) + h_{U^{k-2},u_{k-1}}(z_k \mid Z^{k-1})$$

$$= \arg\max_{u_{k-1}} h_{U^{k-2},u_{k-1}}(z_k \mid Z^{k-1}) .$$

The term that is maximized is defined as M_{k-1}:

$$M_{k-1} \equiv h_{U^{k-1}}\left(z_k \mid Z^{k-1}\right) = h_{U^{k-2},u_{k-1}}(z_k \mid Z^{k-1}) . \tag{7.49}$$

Now we have a simplified measure that can be used for maximization of mutual information and so the observability by proper selection of u_{k-1}. This measure obtained is used frequently in the rest of the thesis and has an important role for the optimization algorithms developed.

The mutual information (7.46) can be maximized for *k* steps by maximizing M^k,

$$M^{N_H} \equiv \sum_{i=1}^{N_H} M_{i-1} . \tag{7.50}$$

It is now reasonable to investigate if it is possible to obtain M_{k-1} at any instance of BOT in a practical way. M_{k-1} can be written as follows

$$M_{k-1} = h_{U^{k-1}}(z_k \mid Z^{k-1}) =$$

$$= -\int f_{U^{k-1}}(z_k, Z^{k-1}, U^{k-1}) \log f_{U^{k-1}}(z_k \mid Z^{k-1}) dz_k dZ^{k-1}$$

$$= -\int f_{U^{k-1}}(Z^{k-1}) \int f_{U^{k-1}}(z_k \mid Z^{k-1}) \log f_{U^{k-1}}(z_k \mid Z^{k-1}) dz_k dZ^{k-1} . \tag{7.51}$$

The outer integral is independent of the inner one and sums up to 1. This further simplifies the measure M_{k-1} to

$$M_{k-1} = -\int f_{U^{k-1}}(z_k \mid Z^{k-1}) \log f_{U^{k-1}}(z_k \mid Z^{k-1}) dz_k . \tag{7.52}$$

The information gained by the measurement sequence Z^{k-1} having an input sequence of U^{k-1} is statistically equivalent to the prior state estimation by the particle filter at time *k*, that uses Z^{k-1} as measurements and U^{k-1} as inputs. This can be shown as,

$$f_{U^{k-1}}(z_k \mid Z^{k-1}) = \int f_{U^{k-1}}(z_k, \xi_k \mid Z^{k-1}) d\xi_k \tag{7.53}$$

$$= \int f_{U^{k-1}}(z_k \mid \xi_k, Z^{k-1})\, f_{U^{k-1}}(\xi_k \mid Z^{k-1}) d\xi_k$$

$$f_{U^{k-1}}(z_k \mid Z^{k-1}) \cong f_{U^{k-1}}(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}) = \sum_i d^i_{k|k-1} f_{U^{k-1}}(z_k \mid \xi^i_{k|k-1}, d^i_{k|k-1}) \tag{7.54}$$

where $\{x^i_{k|k-1}, d^i_{k|k-1}\}$ $i = 1...N$ are the set of particles and corresponding weights, d, where N is the number of particles.

Then using the particle filter approximation of the prior distribution,

$$M_{k-1} = h(z_k \mid Z^{k-1}, u^{k-1}) \cong -\int f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right) \log f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right) dz_k. \tag{7.55}$$

Equation (7.55) shows that having the posterior pdf at time *k-1*, based on $\{U^{k-1}, Z^{k-1}\}$ the recursive calculation of the mutual information is possible. Using the particle representation of the prior at time *k*, the value for the measure M_{k-1} can be calculated at time *k* using the last input u_{k-1}. However optimal selection of the input is another issue and is handled in the next section. Here we must say that it is not easy if not impossible to find an explicit expression of the input in terms of the information state. So in the next section first we will show the computation for the gradient of the incremental reward, M_{k-1}. The second issue that we deal is the computation of the optimal input for a finite time interval that is based on the predicted target position. This solution is of dynamic programming type.

7.3 The Gradient of the Measure for Acceleration Input Model

In this section the aim is to maximize the simplified observability measure M_k at time *k*, by proper selection of the input u_{k-1} . At the maximum point the derivative of M_k with respect to u_k is expected to be 0. Using (7.55) we evaluate the derivative as given below:

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{\partial}{\partial u_{k-1}} h(z_k \mid Z^{k-1})$$

$$\cong \frac{\partial}{\partial u_{k-1}}\left(-\int f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)\log f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)dz_k\right)$$

$$= -\int \frac{\partial}{\partial u_{k-1}}\left(f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)\log f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)\right)dz_k$$

$$= -\int \frac{\partial}{\partial u_{k-1}}\left(f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)\right)\left[1+\log f\left(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\}\right)\right]dy_k \ . \tag{7.56}$$

In the equation above, the given information Z^{k-1} is replaced by the information state which is the conditional pdf of the state of the system. The information state is represented by a set of particles with equal weights of *1/N*. We approximate $f(z_k \mid \{\xi^i_{k|k-1}, d^i_{k|k-1}\})$ as a sum of conditional Gaussian pdf's each conditioned on one particle. With this representation we write

$$\frac{\partial M_{k-1}}{\partial u_{k-1}} = -\int \frac{\partial}{\partial u_{k-1}}\left(\frac{1}{N}\sum_{i=1}^{N} f\left(z_k \mid \xi^i_{k|k-1}, d^i_{k|k-1}\right)\right)\left[1+\log \sum_{i=1}^{N}\frac{1}{N} f\left(z_k \mid \xi^i_{k|k-1}, d^i_{k|k-1}\right)\right]dz_k \tag{7.57}$$

The first term on the right is the partial derivative

$$\frac{\partial}{\partial u_{k-1}}\left(\frac{1}{N}\sum_{i=1}^{N} f\left(z_k \mid \xi^i_{k|k-1}, d^i_{k|k-1}\right)\right) \tag{7.58}$$

where state equation is repeated here (3.10)

$$\xi^i_{k|k-1} = A\xi^i_{k-1|k-1} + G_k v^i_k - G_k u_{k-1} \ . \tag{7.59}$$

Define m^i_k as the expected bearing angle for the particle *i* at time *k*, using (3.8) and (3.13),

$$m^i_k \equiv c(\xi^i_k) = \arctan\left(\frac{x^i_{k|k-1}}{y^i_{k|k-1}}\right). \tag{7.60}$$

In terms of state components at *k-1* it can be expressed as

$$m_k^i = \arctan\left(\frac{x_{k-1|k-1}^i + T\dot{x}_{k-1|k-1}^i - (T^2/2)u(2)_{k-1} + (T^2/2)v(2)_{k-1}^i}{y_{k-1|k-1}^i + T\dot{y}_{k-1|k-1}^i - (T^2/2)u(1)_{k-1} + (T^2/2)v(1)_{k-1}^i}\right). \tag{7.61}$$

Denoting the numerator and denominator with a_k^i and b_k^i respectively, we get,

$$m_k^i \equiv \arctan\frac{a_k^i}{b_k^i}. \tag{7.62}$$

Now it is possible to change the notation for (7.58) using sum of Gaussians as follows

$$\frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N}\frac{1}{N} f\left(z_k \mid \xi_{k|k-1}^i, d_{k|k-1}^i\right)\right) = \frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N}\frac{1}{N} N\left(z_k; m_k^i, \sigma_w^2\right)\right). \tag{7.63}$$

Then,

$$\frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N}\frac{1}{N} N\left(m_k^i, \sigma_w^2\right)\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial u_{k-1}} N\left(z_k; m_k^i, \sigma_w^2\right) \tag{7.64}$$

where σ_w^2 is the measurement noise covariance of the system.

Here it is assumed that the particles are resampled and having equal weights as

$$d_{k|k-1}^i = \frac{1}{N} \quad for \quad i = 1,..., N. \tag{7.65}$$

We can reformulate the derivative as

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial u_{k-1}} N\left(z_k; m_k^i, \sigma_w^2\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{\partial m_k^i}{\partial u_{k-1}}\frac{\partial}{\partial m_k^i} N\left(z_k; m_k^i, \sigma_w^2\right) \tag{7.66}$$

Using the components of the input vector u_{k-1}, the first term of (7.66) can be written as,

$$\frac{\partial m_k^i}{\partial u_{k-1}} = \begin{bmatrix} \frac{\partial}{\partial u(1)_{k-1}} m_k^i \\ \frac{\partial}{\partial u(2)_{k-1}} m_k^i \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial u(1)_{k-1}} \arctan \frac{a_k^i}{b_k^i} \\ \frac{\partial}{\partial u(2)_{k-1}} \arctan \frac{a_k^i}{b_k^i} \end{bmatrix} \tag{7.67}$$

$$= \frac{1}{1+\left(\frac{a_k^i}{b_k^i}\right)^2} \begin{bmatrix} \frac{\partial}{\partial u(1)_{k-1}} \frac{a_k^i}{b_k^i} \\ \frac{\partial}{\partial u(2)_{k-1}} \frac{a_k^i}{b_k^i} \end{bmatrix} \tag{7.68}$$

$$= \frac{(b_k^i)^2}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} \frac{b_k^i \frac{\partial a_k^i}{\partial u(1)_{k-1}} - a_k^i \frac{\partial b_k^i}{\partial u(1)_{k-1}}}{(b_k^i)^2} \\ \frac{b_k^i \frac{\partial a_k^i}{\partial u(2)_{k-1}} - a_k^i \frac{\partial b_k^i}{\partial u(2)_{k-1}}}{(b_k^i)^2} \end{bmatrix}. \tag{7.69}$$

Using the values of a_k^i and b_k^i given in (7.61) and (7.62), it can be simplified as

$$\frac{\partial m_k^i}{\partial u_{k-1}} = \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} -a_k^i \frac{\partial}{\partial u(1)_{k-1}} b_k^i \\ b_k^i \frac{\partial}{\partial u(2)_{k-1}} a_k^i \end{bmatrix} = \frac{T^2/2}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \tag{7.70}$$

The second term of (7.66) can be written as,

$$\begin{aligned} \frac{\partial}{\partial m_k^i} N\left(z_k; m_k^i, \sigma_w^2\right) &= \frac{\partial}{\partial m_k^i} \left(\frac{1}{\sqrt{2\pi\sigma_w^2}} \exp\left(-(z_k - m_k^i)^2/(2\sigma_w^2)\right) \right) \\ &= \frac{(z_k - m_k^i)/\sigma_w^2}{\sqrt{2\pi\sigma_w^2}} \exp\left(-(z_k - m_k^i)^2/(2\sigma_w^2)\right) \\ &= \frac{(z_k - m_k^i)}{\sigma_w^2} N\left(z_k; m_k^i, \sigma_w^2\right) \end{aligned} \tag{7.71}$$

Using (7.63) and (7.64) we have

$$\frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N}\frac{1}{N}\,f\left(z_k \mid x_{k|k-1}^{i}, d_{k|k-1}^{i}\right)\right)=\frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial u_{k-1}}\,N\left(z_k; m_k^i, \sigma_v^2\right) \tag{7.72}$$

This is reformulated in (7.73) by using (7.70) and (7.71),

$$\frac{\partial}{\partial u_{k-1}}\sum_{i=1}^{N}\frac{1}{N}\,f\left(z_k \mid x_{k|k-1}^{i}, d_{k|k-1}^{i}\right)=\frac{1}{N}\sum_{i=1}^{N}\frac{T^2/2}{(a_k^i)^2+(b_k^i)^2}\begin{bmatrix}a_k^i\\-b_k^i\end{bmatrix}\frac{(y_k-m_k^i)}{\sigma_w^2}\,N\left(z_k; m_k^i, \sigma_w^2\right) \tag{7.73}$$

It is important to note that we have replaced the derivative with a summation where calculation of the measurement likelihood, $N\left(z_k; m_k^i, \sigma_w^2\right)$, for each particle is required. The aim is to calculate (7.56). Using (7.73) in (7.56) it follows that

$$\frac{\partial}{\partial u_{k-1}}M_{k-1}=\frac{\partial}{\partial u_{k-1}}h(z_k \mid Z^{k-1})$$

$$\cong -\int\frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N}f\left(z_k \mid x_{k|k-1}^{i}, d_{k|k-1}^{i}\right)\right)\left[1+\log\sum_{i=1}^{N}\frac{1}{N}\,f\left(z_k \mid x_{k|k-1}^{i}, d_{k|k-1}^{i}\right)\right]dz_k$$

$$-\int\left[\frac{1}{N}\sum_{i=1}^{N}\frac{T^2/2}{(a_k^i)^2+(b_k^i)^2}\begin{bmatrix}a_k^i\\-b_k^i\end{bmatrix}\frac{(z_k-m_k^i)}{\sigma_w^2}\,N\left(z_k; m_k^i, \sigma_w^2\right)\right] \tag{7.74}$$

$$\left[1+\log\sum_{i=1}^{N}\frac{1}{N}\,f\left(z_k \mid \xi_{k|k-1}^{i}, d_{k|k-1}^{i}\right)\right]dy_k$$

$$=-\frac{T^2/2}{N}\sum_{i=1}^{N}\frac{1}{(a_k^i)^2+(b_k^i)^2}\begin{bmatrix}a_k^i\\-b_k^i\end{bmatrix} \tag{7.75}$$

$$\int\left[\frac{(z_k-m_k^i)}{\sigma_w^2}\,N\left(z_k; m_k^i, \sigma_w^2\right)\right]\left[1+\log\frac{1}{N}\sum_{j=1}^{N}N\left(z_k; m_k^j, \sigma_w^2\right)\right]dz_k$$

$$=-\frac{T^2}{2N\sigma_w^2}\sum_{i=1}^{N}\frac{1}{(a_k^i)^2+(b_k^i)^2}\begin{bmatrix}a_k^i\\-b_k^i\end{bmatrix} \tag{7.76}$$

$$\int\left[(z_k-m_k^i)N\left(z_k; m_k^i, \sigma_w^2\right)\right]\left[\log\frac{1}{N}\sum_{j=1}^{N}N\left(z_k; m_k^j, \sigma_w^2\right)\right]dz_k$$

$$-\frac{T^2}{2N\sigma_w^2}\sum_{i=1}^{N}\frac{1}{(a_k^i)^2+(b_k^i)^2}\begin{bmatrix}a_k^i\\-b_k^i\end{bmatrix}\int(z_k-m_k^i)N\left(z_k; m_k^i, \sigma_w^2\right)dz_k$$

The second term of (7.76) has the first moment of the Gaussian distribution that is equal to 0. Then the equation simplifies to,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = -\frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log \frac{1}{N} \sum_{j=1}^{N} N\left(z_k; m_k^j, \sigma_w^2\right) \right] dz_k. \tag{7.77}$$

At this point, in order to deal with the logarithm of the Gaussian sum of (7.77), we approximate the sum with a single Gaussian using the mean of the bearing angle expectations of particles, m_k^j , for $j=1,..,N$, i.e.,

$$\frac{1}{N} \sum_{j=1}^{N} N\left(z_k; m_k^j, \sigma_w^2\right) \cong N\left(z_k; m_k, \sigma_{wm}^2\right). \tag{7.78}$$

Using the fact that the first moment of the Gaussian is 0, more simplification is possible:

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} \cong -\frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log N\left(z_k; m_k, \sigma_{wm}^2\right) \right] dz_k \tag{7.79}$$

$$= -\frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log \frac{1}{\sqrt{2\pi\sigma_w^2}} - \frac{(z_k - m_k)^2}{2\sigma_w^2} \right] dz_k \tag{7.80}$$

finally,

$$= \frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int N\left(z_k; m_k^i, \sigma_w^2\right) (z_k - m_k^i) \left[\frac{(z_k - m_k)^2}{2\sigma_w^2} \right] dz_k \tag{7.81}$$

is obtained. Equation (7.81) can further be simplified using the trick of adding and subtracting m_k^i inside the integral,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int N\left(z_k; m_k^i, \sigma_w^2\right)(z_k - m_k^i) \left[\frac{(z_k - m_k^i + m_k^i - m_k)^2}{2\sigma_w^2} \right] dz_k \tag{7.82}$$

so that,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \int N\left(z_k; m_k^i, \sigma_w^2\right) \left[\frac{(z_k - m_k^i)^3 + (z_k - m_k^i)(m_k^i - m_k)^2 + 2(z_k - m_k^i)^2 (m_k^i - m_k)}{2\sigma_w^2} \right] dz_k. \tag{7.83}$$

Again replacing the odd moments of Gaussian distribution with 0, (7.83) becomes

$$= \frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} \frac{(m_k^i - m_k)}{\sigma_w^2} \int N\left(z_k; m_k^i, \sigma_w^2\right) \left[(z_k - m_k^i)^2 \right] dz_k \tag{7.84}$$

where the integral is nothing but the second moment of the Gaussian distribution resulting to the final simplification as

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{T^2}{2N\sigma_w^2} \sum_{i=1}^{N} \frac{1}{(a_k^i)^2 + (b_k^i)^2} \begin{bmatrix} a_k^i \\ -b_k^i \end{bmatrix} (m_k^i - m_k). \tag{7.85}$$

Equation (7.85) reveals that it is possible to obtain an approximation of the gradient of the measure that we have defined by simply using the particles. This formulation gets rid of any numerical method of gradient calculation as long as the particles are present. This derivation is important for the gradient based optimization that we have used in the thesis.

7.4 The Gradient of the Measure for Constant Speed Ownship Model

For the case where ownship has constant speed but a variable heading value the derivative of the measure M_k with respect to heading angle input is derived here. Again the target state is the position and velocity in the *x* and the *y* directions,

$$\xi^t = \left[x^t \, y^t \, \dot{x}^t \, \dot{y}^t\right]^T \tag{7.86}$$

where the target has constant velocity model with a noise term as acceleration,

$$\xi_{k+1}^t = A\xi_k^t + G_k v_k{}^T \tag{7.87}$$

with

$$A_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad G_k = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix}. \tag{7.88}$$

The formulation used in this section has the target, ξ_k^t and the ownship ξ_k^o state representations separated instead of the relative state, ξ_k . The bearing angle is still the measurement of the relative position of the target,

$$z_k = c(\xi_k) + w_k = c(\xi_k^t, \xi_k^o) + w_k \tag{7.89}$$

where

$$c(\xi_k^t, \xi_k^o) = \arctan\left(\frac{x_k^t - x_k^o}{y_k^t(1) - y_k^o(1)}\right) = \arctan\left(\frac{x_k}{y_k}\right) \tag{7.90}$$

In (7.90) x_k and y_k stands for the coordinates of the relative target position. Similar to the previous section, m_k^i is the expected bearing angle for the particle *i*. Using ownship motion equation (3.15) we have

$$m_k^i \equiv h(\xi_k^i) = \arctan\left(\frac{x_{k|k-1}^i}{y_{k|k-1}^i}\right) \tag{7.91}$$

$$= \arctan\frac{{x^t}_{k-1|k-1}^i + T{\dot{x}^t}_{k-1|k-1}^i + (T^2/2)v(2)_{k-1}^i - x_{k-1}^o - Ts\sin\varphi_{k-1}}{{y^t}_{k-1|k-1}^i + T{\dot{y}^t}_{k-1|k-1}^i + (T^2/2)v(1)_{k-1}^i - y_{k-1}^o - Ts\cos\varphi_{k-1}} \equiv \arctan\frac{a_k^i}{b_k^i}$$

Here it is assumed that particles are resampled and having equal weights as

$$d_{k|k-1}^i = \frac{1}{N} \quad for \quad i = 1,..., N. \tag{7.92}$$

Then similar to the equation (7.66) of previous section we have,

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial u_{k-1}}N\left(z_k; m_k^i, \sigma_w^2\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{\partial m_k^i}{\partial u_{k-1}}\frac{\partial}{\partial m_k^i}N\left(z_k; m_k^i, \sigma_w^2\right) \quad . \tag{7.93}$$

Since the input is changed to heading angle of the ownship, equation (7.67) of the previous section becomes,

$$\frac{\partial m_k^i}{\partial u_{k-1}} = \left[\frac{\partial}{\partial \varphi_{k-1}} m_k^i\right] = \left[\frac{\partial}{\partial \varphi_{k-1}}\arctan\frac{a_k^i}{b_k^i}\right] \tag{7.94}$$

$$= \frac{1}{1+\left(\frac{a_k^i}{b_k^i}\right)^2}\left[\frac{\partial}{\partial \varphi_{k-1}}\frac{a_k^i}{b_k^i}\right] = \frac{(b_k^i)^2}{(a_k^i)^2 + (b_k^i)^2}\left[\frac{b_k^i\frac{\partial a_k^i}{\partial \varphi_{k-1}} - a_k^i\frac{\partial b_k^i}{\partial \varphi_{k-1}}}{(b_k^i)^2}\right] \tag{7.95}$$

which simplifies to,

$$\frac{\partial m_k^i}{\partial u_{k-1}} = \frac{1}{(a_k^i)^2 + (b_k^i)^2}\left[-Ts\cos\varphi_{k-1}b_k^i - Ts\sin\varphi_{k-1}a_k^i\right]$$

$$= \frac{-Ts}{(a_k^i)^2 + (b_k^i)^2}\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]. \tag{7.96}$$

The second term of (7.93) is the same as (7.71),

$$\begin{aligned}\frac{\partial}{\partial m_k^i} N\left(z_k; m_k^i, \sigma_w^2\right) &= \frac{\partial}{\partial m_k^i}\left(\frac{1}{\sqrt{2\pi\sigma_w^2}} \exp\left(-(z_k - m_k^i)^2/(2\sigma_w^2)\right)\right) \\ &= \frac{(z_k - m_k^i)/\sigma_w^2}{\sqrt{2\pi\sigma_w^2}} \exp\left(-(z_k - m_k^i)^2/(2\sigma_w^2)\right) \\ &= \frac{(z_k - m_k^i)}{\sigma_w^2} N\left(z_k; m_k^i, \sigma_w^2\right). \end{aligned} \tag{7.97}$$

Using (7.96) and (7.97) in (7.93), one obtains:

$$\frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N} \frac{1}{N} f\left(z_k \mid \xi_{k|k-1}^i, d_{k|k-1}^i\right)\right) = \frac{1}{N}\sum_{i=1}^{N} \frac{\partial m_k^i}{\partial u_{k-1}} \frac{\partial}{\partial m_k^i} N\left(z_k; m_k^i, \sigma_w^2\right)$$

$$= \frac{1}{N}\sum_{i=1}^{N} \frac{-Ts}{(a_k^i)^2 + (b_k^i)^2}\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right] \frac{(z_k - m_k^i)}{\sigma_w^2} N\left(z_k; m_k^i, \sigma_w^2\right). \tag{7.98}$$

It is desired to calculate

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{\partial}{\partial u_{k-1}} h(z_k \mid Z^{k-1})$$

$$\cong -\int \frac{\partial}{\partial u_{k-1}}\left(\sum_{i=1}^{N} \frac{1}{N} f\left(z_k \mid x_{k|k-1}^i, d_{k|k-1}^i\right)\right)\left[1 + \log \sum_{i=1}^{N} \frac{1}{N} f\left(z_k \mid x_{k|k-1}^i, d_{k|k-1}^i\right)\right] dz_k. \tag{7.99}$$

Using (7.98) in (7.99)

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = -\int \left[\frac{1}{N} \sum_{i=1}^{N} \frac{-Ts\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \frac{(z_k - m_k^i)}{\sigma_w^2} N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[1 + \log \sum_{i=1}^{N} \frac{1}{N} f\left(z_k \mid \xi_{k|k-1}^i, d_{k|k-1}^i\right) \right] dz_k \tag{7.100}$$

is obtained. That is,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{Ts}{N} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int \left[\frac{(y_k - m_k^i)}{\sigma_w^2} N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[1 + \log \frac{1}{N} \sum_{j=1}^{N} N\left(z_k; m_k^j, \sigma_w^2\right) \right] dz_k \tag{7.101}$$

$$= \frac{Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log \frac{1}{N} \sum_{j=1}^{N} N\left(z_k; m_k^j, \sigma_w^2\right) \right] dz_k \tag{7.102}$$

$$+ \frac{Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int (z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) dz_k$$

The second term on the right has the first moment of the Gaussian distribution that equals to 0. Then the equation simplifies to,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log \frac{1}{N} \sum_{j=1}^{N} N\left(z_k; m_k^j, \sigma_w^2\right) \right] dz_k \tag{7.103}$$

The approximation at (7.78) is applied again, then,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} \cong \frac{Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right) \right] \left[\log N\left(z_k; m_k, \sigma_w^2\right) \right] dz_k \tag{7.104}$$

and

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} \cong \frac{Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \tag{7.105}$$

$$\int \left[(z_k - m_k^i) N\left(z_k; m_k^i, \sigma_w^2\right)\right] \left[\log \frac{1}{\sqrt{2\pi\sigma_w^2}} - \frac{(z_k - m_k)^2}{2\sigma_w^2}\right] dz_k \ .$$

Again the first moment of the Gaussian is removed for simplification, resulting to

$$= \frac{-Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \int N\left(z_k; m_k^i, \sigma_w^2\right)(z_k - m_k^i) \left[\frac{(z_k - m_k)^2}{2\sigma_w^2}\right] dz_k \tag{7.106}$$

Similar to the previous section,

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{-Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \tag{7.107}$$

$$\int N\left(z_k; m_k^i, \sigma_w^2\right)(z_k - m_k^i) \frac{(z_k - m_k^i + m_k^i - m_k)^2}{2\sigma_w^2} dz_k$$

$$= \frac{-Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \tag{7.108}$$

$$\int N\left(z_k; m_k^i, \sigma_w^2\right) \frac{(z_k - m_k^i)^3 + (z_k - m_k^i)(m_k^i - m_k)^2 + 2(z_k - m_k^i)^2 (m_k^i - m_k)}{2\sigma_w^2} dz_k$$

is found. Finally,

$$= \frac{-2Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} \frac{(m_k^i - m_k)}{\sigma_w^2} \int N\left(z_k; m_k^i, \sigma_w^2\right)(z_k - m_k^i)^2 dz_k \tag{7.109}$$

is obtained. Since the second moment of the Gaussian is nothing but the covariance, this further simplifies the equation to

$$\frac{\partial}{\partial u_{k-1}} M_{k-1} = \frac{-2Ts}{N\sigma_w^2} \sum_{i=1}^{N} \frac{\left[b_k^i \cos\varphi_{k-1} + a_k^i \sin\varphi_{k-1}\right]}{(a_k^i)^2 + (b_k^i)^2} (m_k^i - m_k) \ . \tag{7.110}$$

For the ownship motion structure characterized with the heading angle we come up with a simplified value for the gradient of the measure. The gradient is turned to a summation using the particles.

7.5 Finite Horizon Input Optimization Based on the Mutual Information for BOT

The aim of this section is to investigate if it is possible to optimize the decision of the input sequence in the finite horizon BOT problem using M_{k-1} for $k=1,....N_H$. Here N_H stands for the horizon. Optimization is in the sense of maximization of mutual information between the state sequence and the measurement sequence, $I(\Xi^k, Z^k)$. In Section 7.2 the recursive formulation of $I(\Xi^k, Z^k)$ is elaborated and it is seen that the maximization of this value is achieved by maximizing M^k of equation (7.50) through the horizon by the selection of a proper sequence of the input.

The total reward function to be maximized for the horizon N_H is defined as follows.

$$M^{N_H} = \sum_{k=1}^{N_H} M_{k-1} = \sum_{k=1}^{N_H} h_{U^{k-1}}(z_k \mid Z^{k-1}) = \sum_{i=1}^{N_H} h_{U^{k-2},u_{k-1}}(z_k \mid Z^{k-1}) \tag{7.111}$$

The problem is to find the optimal control input sequence $U^{k-1^*} = \{u_0, u_1, ..., u_{k-1}\}$ to maximize (7.111).

At time *k-1*, having an input sequence of U^{k-2}, a posterior state representation is available denoted by $\xi_{k-1|k-1}$. Then the prior at *k*, $\xi_{k|k-1}$ can be determined based on the selected input, u_{k-1}. M_{k-1} can be considered as instantaneous reward, and it is a function of the prior $\xi_{k|k-1}$ at *k*, which is determined by the posterior at *k-1*, $\xi_{k-1|k-1}$. It is a measure of the increase of the mutual information depending on u_{k-1}. It is important to emphasize that the instantaneous reward is a function of state representation and the input at *k-1*. For the BOT problem, which is formed as a Markov process, a policy *g*, for determination of the input selection can be dependent on the particle filter state representation, that is $u_k = g_k(x_{k|k})$. A dynamic

programming approach is possible if we can use M_{k-1} as the instantaneous reward and determine inputs based on its value.

Various models can be selected for the input function as explained before. Input can be chosen as the acceleration term of the ownship motion with some threshold. It is also possible to keep the ownship speed as a predetermined constant and let the direction be the variable to be optimized. Actually this approach suggests the use of the neighboring positions of the ownship as optimization variables. A quantization of these positions is possible and is applied in this study to simplify the optimization procedure. Quantization of the positions corresponds to the quantization of the heading angle of the ownship as well as the speed.

The above formulation of the problem requires the optimization of the ownship's trajectory for a time horizon of N_H steps. We assume that initial time is $k=0$ and the ownship is at the origin initially. After N_H^{th} step, the actual time is $N_H.T$, where T is the period of each tracking step. The ownship can be still at origin, (0,0) if input sequence is $U^{N_H-1}=\{u_0,\ldots,u_{NH-1}\}=\{0,..,0\}$). One other alternative for the position of the ownship is to be at the extreme west $(0, N_H.T.v_{own})$ using the input sequence $U^{N_H-1}=\{1,\ldots,1\}$ where 1 denotes the direction west with the constant ownship speed, v_{own}. The examples can be increased but it is seen that the points where ownship can be situated after N_H time steps is a function of N_H and are determined as a set of possible ownship position grid nodes spanned over the 2-D space.

For an arbitrary initial position we define any feasible ownship position indexed by g at $k=N_H$ as

$$\xi_g^o(N_H) \in F_{N_H} \tag{7.112}$$

and

$$F_{N_H} = sp(\xi^0(0), N_H)$$

where F_{N_H} is the set of feasible ownship positions at $k=N_H$ and *sp* represent the function to span the space to get the feasible positions (grid nodes) for ownship at $k=N_H$, considering the initial ownship position is $\xi^0(0)$.

Having defined feasible positions for the ownship at $k=N_H$, as $\xi_g^o(N_H) \in F_{N_H}$, it is also possible to get the feasible ownship positions at $k=N_H-1$ as

$$\xi_g^o(N_H - 1) \in F_{N_H - 1} \tag{7.113}$$

where

$F_{N_H-1} = sp(\xi^0(0), N_H - 1)$.

The feasible associations for the ownship between time instances $k=N_H-1$ and $k=N_H$ can be determined. First we define the function $s_{k-1,k}(\xi_g^o(k-1), \xi_g^o(k))$ as a neighbor indicator as follows:

If a feasible ownship position $\xi_g^o(k-1)$ denoted using the index *g* at time *k-1* is in the neighborhood of a feasible position $\xi_g^o(k)$ at time *k* then,

$$s_{k-1,k}(\xi_g^o(k-1), \xi_g^o(k) = 1 \tag{7.114}$$

and they are neighbor grid nodes, else,

$$s_{k-1,k}(\xi_g^o(k-1), \xi_g^o(k) = 0 \tag{7.115}$$

where $\xi_g^o(k) \in F_k = sp(\xi^o(0), k) \quad for\ g = 1,.2,..N_g$.

The *s* function determines the feasible motions for each time step, that is the motions of the ownship to its neighboring grid nodes.

Here N_g is the number of feasible grid positions at time step *k*, in other words number of the members of the set F_k. 'Neighborhood' can be defined according to the quantization of the speed and the heading angles of the ownship. For $k=N_H$, after determining the associations, it is possible to evaluate the terminal rewards as,

$$M_{k-1}(\xi_g^o(k)) = h_{U^{k-2},u_{k-1}}(z_k \mid Z^{k-1}) = h(z_{k|o}(k) \mid \xi_{k|0}^o(k)) \tag{7.116}$$

Calculation of instantaneous reward requires calculation of $\xi_{k|0}(k)$ and $z_{k|o}(k)$. These are the target's *k* step predicted state and *k* step bearing angle predictions, respectively. Particles are evolved *k* steps to get the predictions for state and bearing angles.

The terminal rewards are taken as the rewards-to-go for the last step:

$$A_{N_H-1}(\xi_g^o(N_H)) \equiv M_{N_H-1}(\xi_g^o(k)) = h(z_{N_H|0}(k) \mid \xi_{N_H|0}^o(k)) \tag{7.117}$$

where

$$\xi_g^o(N_H) \in F_{N_H} \ . \tag{7.118}$$

Reward-to-go, for any feasible input for ownship state at step *k* is given by

$$A_{k-2}(\xi_g^o(k-1)) \equiv \max(A_{k-1}(\xi_g^o(k)) + M_{k-2}(\xi_g^o(k-1))). \tag{7.119}$$

Given the displacement of the ownship is feasable,

$s_{k-1,k}(\xi_g^o(k-1), \xi_g^o(k)) = 1.$

The associated input u_{k-2} is

$$u_{k-2}^* = \arg\max_{u_{k-2}}(A_{k-1}(\xi_g^o(k)) + M_{k-2}(\xi_g^o(k-1))) \tag{7.120}$$

given $s_{k-1,k}(\xi_g^o(k-1), \xi_g^o(k)) = 1.$

The input u_{k-2} that maximizes equation (7.120) is selected. Depending on this optimal input u_{k-2}^*, the observer jumps to a new grid node. For this node, the value $A_{k-1}(\xi_g^o(k)$ that maximizes $A_{k-2}(\xi_g^o(k-1))$ shows the next state $\xi_g^o(k)$, and the associated input for displacement from *k-1* to *k*. The associated input is u_{k-1}^* , which is the optimal one that brings $\xi_g^o(k-1)$ to $\xi_g^o(k)$.

A reward of 0 is given for the unfeasible associations to remove that paths during optimization. Each feasible association $(s_{k-1,k}(\xi_g^o(k-1),\xi_g^o(k))=1)$, determines a possible input, $u_{k-1}^i(\xi_g^o(k-1),\xi_g^o(k))$.

Back propagation goes up to the initial time where the ownship is at the origin:

$$A^*(\xi_g^o(0)) = \max(A_0(\xi_g^o(1))). \tag{7.121}$$

The maximum reward $A^*(\xi_g^o(0))$ and the associating input sequence $U^k = \{u_0^*, u_1^*, ..., u_{N_H-1}^*\}$ is obtained. Typically only u_0^* is used, and the system is evolved to *k=1*. Estimation filter updates the next state with the next observation. Then the horizon is reduced to N_H*-1* and the programming is repeated with the updated state and reduced horizon. Another approach would be to always keep the horizon fixed and to solve the problem to get the first optimal input. This requires more computation time than the first approach.

CHAPTER 8

IMPLEMENTATION AND PERFORMANCE COMPARISON OF OBSERVER MANEUVER PLANNING APPROACHES BASED ON THE MUTUAL INFORMATION FOR BOT

In the previous Chapter, we have investigated the mutual information between the state and the measurement sequences and derived some practical formulas to be used in the BOT problem to improve the ownship trajectory that increases the overall tracking accuracy.

In Section 7.1 we obtained an approximate formulation for one step maximization of the mutual information between the state vector and the nonlinear measurement using the formulation derived for the linear systems. In Section 1 of this Chapter we implement this approach. The implementation is done for the unconstrained and constrained input cases. Furthermore we have used two optimization methods, one is the brute force method of computing the value of the objective function on a grid and the other is based on the gradient ascent approach.

In Section 7.2 we investigated the recursive form of the mutual information derived for the nonlinear systems [27] and a simplified measure, M_k is obtained. The simplified measure can be applied to any system that contains an input. This measure is elaborated and an approximation for its gradient is proposed in Sections 7.3 and 7.4. Since BOT is the tracking of a nonlinear system particle filter implementation is almost unavoidable. The derived approximate formulation uses the particle values where the gradient computation reduces to a simple summation in this framework. The problem is solved both for the acceleration and the heading angle input cases.

The time horizon is selected as one and the optimization problem is solved by the steepest ascent method. The results are given in Section 2 of this Chapter.

Single step optimization, although simpler to solve, is obviously a sub optimal solution. The theory related with the mutual information is rich enough to consider arbitrarily large time horizons however it is not easy to find a computationally feasible solution to the constrained optimization problem. Some other approximations can be done to overcome the problem. In Section 7.5 we investigated the possibility of using the recursive formulation of the mutual information derived for the nonlinear systems in [27] for a finite horizon dynamic programming problem. Still depending on the simplified measure, M_k, a finite horizon optimization problem where the input position is constrained on grids on a 2-D space is introduced. Section 3 of this Chapter is for the implementation of this approach.

20 Monte Carlo simulations are done for each implementation. It is known that the initial estimations of the state and ownship positions affect the tracking performance. To make the simulations realistic an error is introduced for the initial target position and velocity estimation. The initial estimated position of the target is given as 100 m and 1000 m deviated from the true position, in y (north) and x (east) directions, respectively. The ownship is located at the origin with no initial velocity. The initial position of the target is around 500 m north and 2000 m east of the ownship, with an initial velocity of 8 m/s north and 10 m/s east. Also speed variables of the filter are initialized with a bias of -1m/s in both the north and the east directions. Target, observer and estimation trajectories are color coded to easily understand their locations with respect to the others for the 100 steps simulation. Every succeeding 10 steps have same color starting from red, and repeated as patterns of red, green and blue.

8.1 Ownship Motion Planning for BOT Using Mutual Information Formulation Derived for Linear Systems

We have conducted 3 different experiments using the formulas derived for the linear systems. In this formulation mutual information between the state and the measurement is represented in terms of some system matrices and related covariance matrices. The simplified objective function derived in (7.24) of Chapter 7 is used. The optimization is done for one step.

8.1.1 Algorithm Type 1

The basic idea of this algorithm is to find the input vector, that is, a 2-dimensional acceleration term of the ownship that maximizes the one step reward with no constraints on the input. The system evolves with the input found at step *k*, and at step *k+1* this procedure is repeated. Optimization method is the simple computation of the objective function on a 2-dimensional grid. Although the global maximum is not found exactly, because of the sufficient grid resolution of the acceleration input, satisfactory results are obtained (Figure 18).

The results are in accordance with the expectations of Section 7.1. With the unconstrained input, ownship tends to arrive to the target's predicted position in one step using the input acceleration appropriately. As a result the relative target position tends to go to 0 in each direction. The high mutual information measure, and the high tracking accuracy are observed in Figures 18-22.

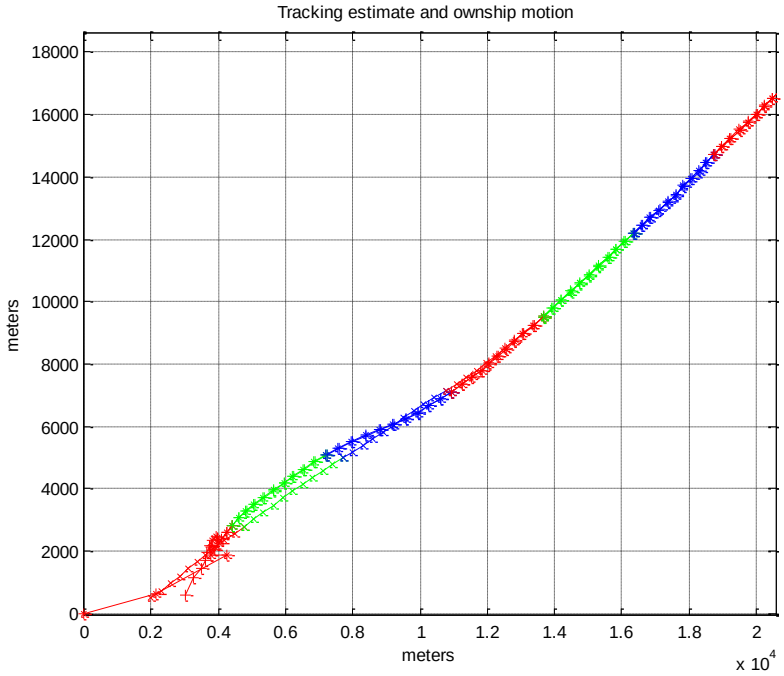


Figure 18- Target position (x), Ownship (*), estimated target position (+) in meters. Ownship has a tendency to minimize the relative distance. The ownship and the target are close to each other and the target position estimation accuracy is high.

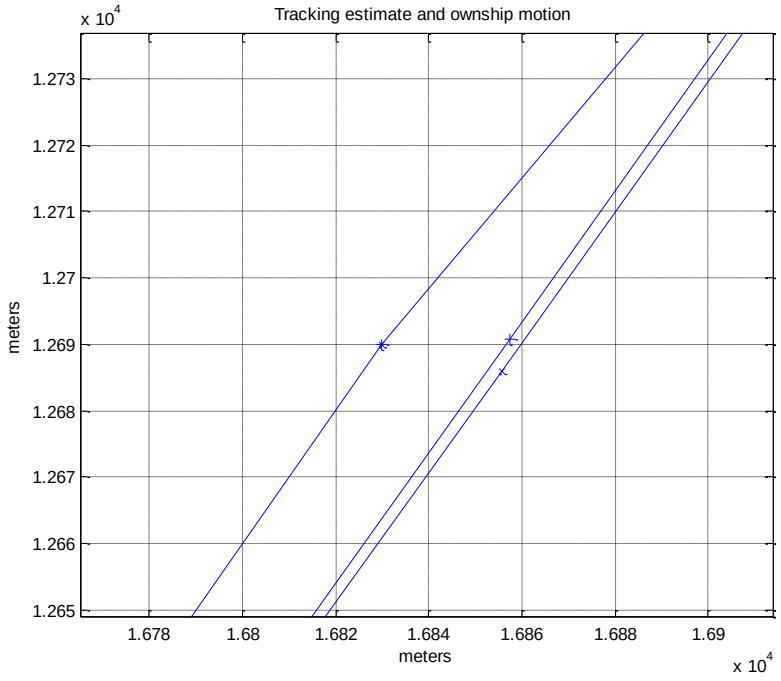


Figure 19- Zoomed view of Figure 18. Target and ownship are so close to each other as expected. The estimation is also very close to the target position.

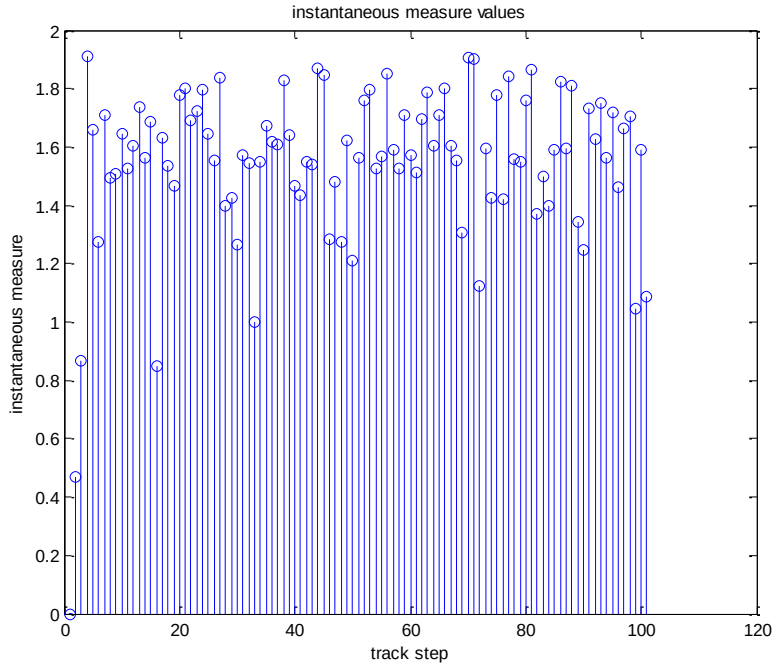


Figure 20- The change of the information measure M_k during tracking. High values are indication of high instantaneous observability measure.

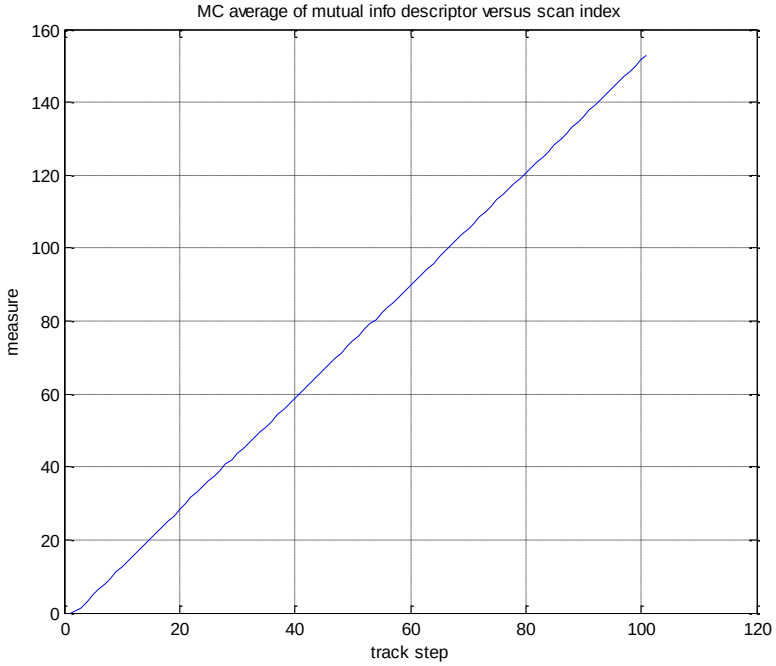


Figure 21- The total measure value reaches up to 150 at the final step. This is the addition of instantaneous values of the previous Figure to show the total measure value during tracking.

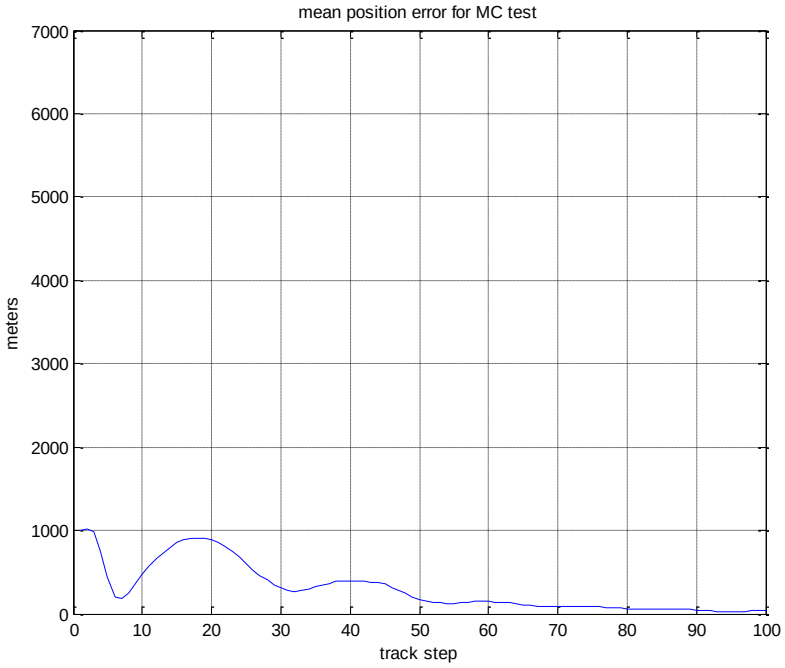


Figure 22- Average position error. It is seen that tracking error is extremely low for the unconstrained case.

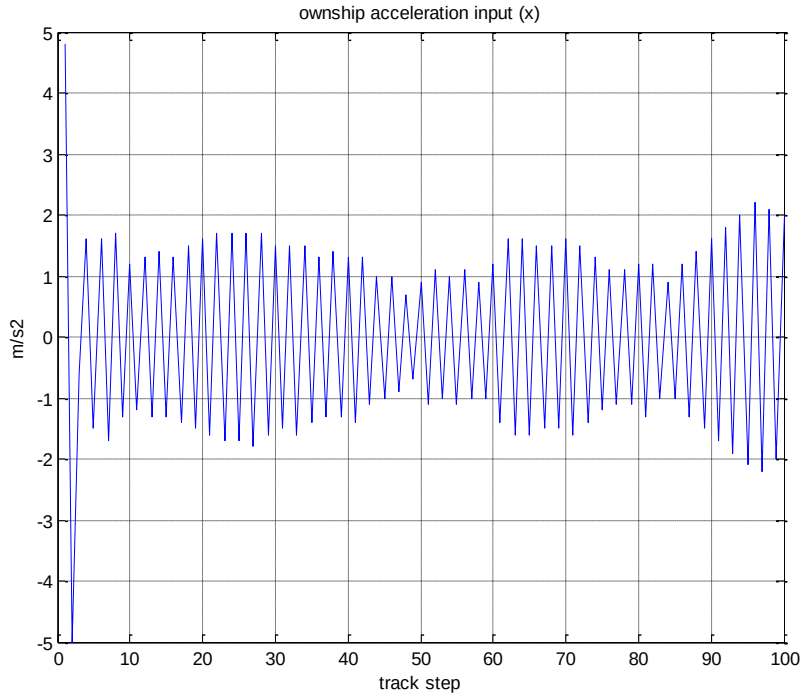


Figure 23- Ownship accelearion input in x-direction for unconstrained case.

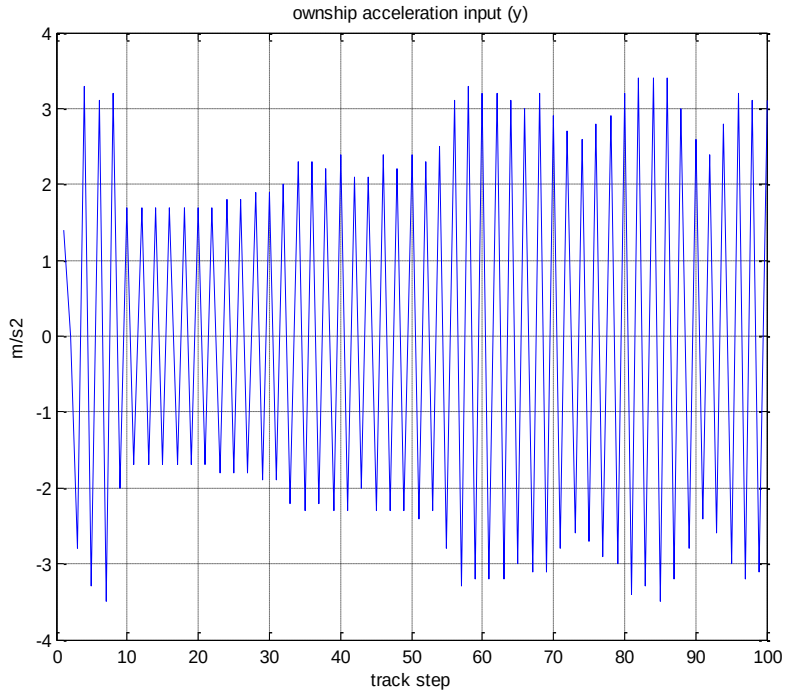


Figure 24- Ownship accelearion input in y-direction for unconstrained case.

One important note is that the Jacobian of the nonlinear measurement equation is used to calculate the objective function. However the filter used is a particle filter that uses the nonlinear nature of the measurements. The Jacobian used in defining the objective function is already computed and used in the EKF formulation. Therefore it looks practical to use EKF in tracking for computational simplicity. However, especially in the unconstrained input case, since the ownship tends to close the gap to the target, nonlinearity of the measurement increases and sooner or later the EKF filter diverges. As a consequence, the calculated reward which is an approximation after linearization is also severely affected. Therefore particle filter is used in the simulations. Figure 25 shows the case of divergence if EKF is preferred.

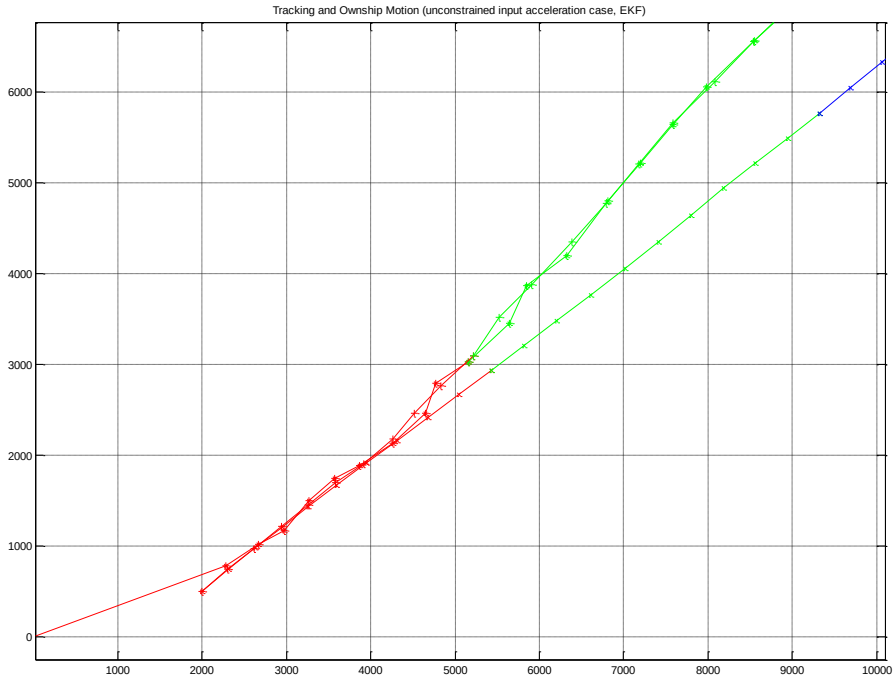


Figure 25- Divergence of the track estimation from true target position when EKF is used. Tracker assumes the target is close to the observer but neither the estimate nor the observer is close to the target. Target position (x), Ownship (*), estimated target position (+), position unit is meters.

8.1.2 Algorithm Type 2

This algorithm is similar to Type 1 algorithm so that at each step the input vector is selected as the one that gives the highest reward value for one time step. The constraint introduced to the input is the additional property of Type 2. The constraint is on the norm of the input vector. The norm of the input vector, i.e. the norm of the acceleration of the ownship is limited. The optimization method is same as the previous one, i.e., it is the brute force method of computing the value of the objective function at each grid point.

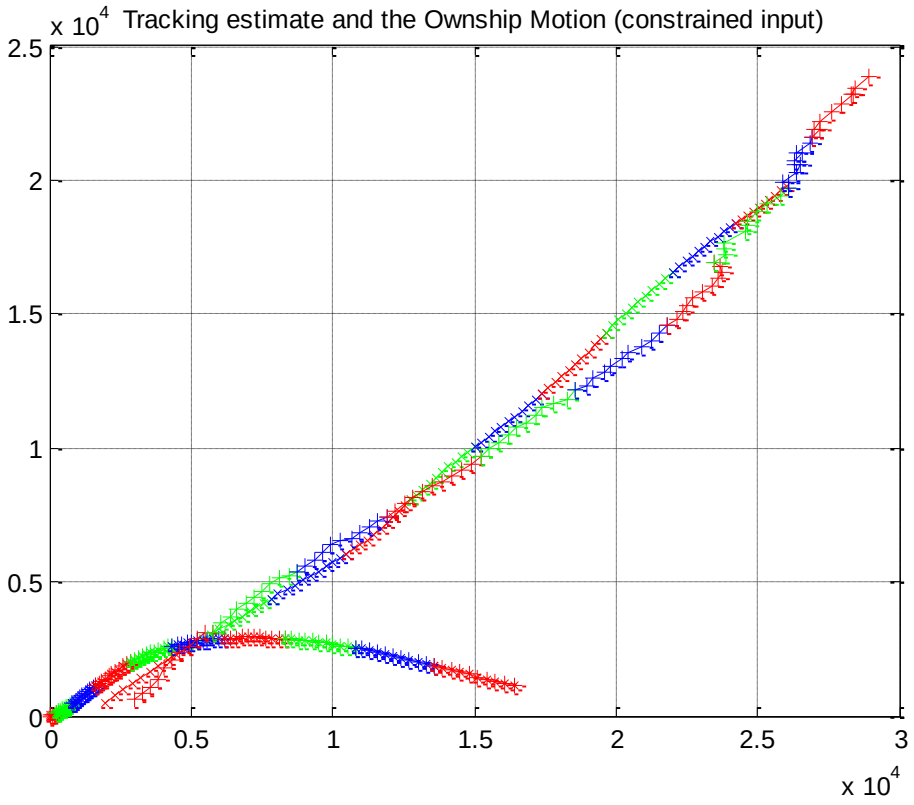


Figure 26- Tracking estimate and the ownship motion for Type 2 algorithm with the limited acceleration of 0.005 m/s^2. Target position (x), Ownship (*), estimated target position (+) in meters.

As seen from Figure 26 when the input acceleration is very limited the tracking accuracy decreases. Due to limited acceleration and the one step optimization ownship cannot reach the target.

On the other hand when the input acceleration limit is increased the tracking performance increases and observer keeps a trajectory almost aligned with the target path up to the final steps as seen in Figure 27. This is because the one step optimization algorithm is able to direct the ownship towards the target if the input limit is high enough.

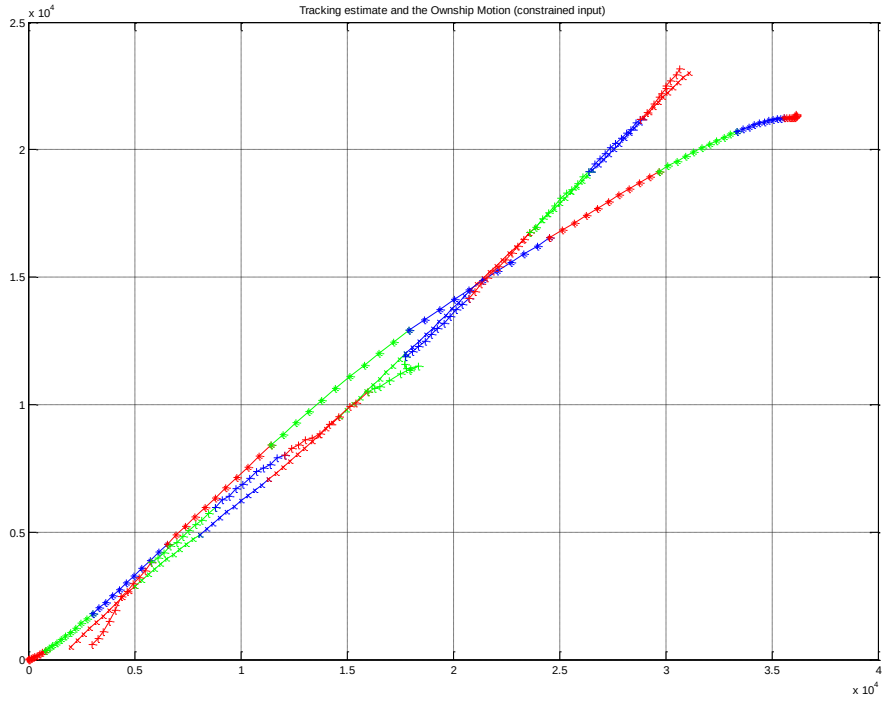


Figure 27- Type 2 algorithm using the limited acceleration of 0.02 m/s^2. It is observed that as the input limit is increased observer comes more closer to the target and the overall tracking accuracy increases. Target position (x), ownship (*), estimated target position (+) in meters.

For most of the real scenarios the ownship is not able or allowed to be very close to the target. This property is only indirectly satisfied in this application. Monte Carlo result of 20 run is given in Table 1.

8.1.3 Algorithm Type 3

This algorithm is different from Type 2 only at the optimization method. New version uses the gradient of the Lagrangian derived in (7.44) of Chapter 7 and it is a gradient ascent based constrained optimization method. Monte Carlo result of 20 run is given in Table 1.

8.2 Input Optimization Using the Gradient of the Measure M_k

The two algorithms given below are one step optimization algorithms like Type 1-3. However we use the simplified measure M_k as the objective function in Type 4 and Type 5.

8.2.1 Algorithm Type 4

In the algorithm Type 4, instead of the gradient calculated in Type 3 described above, the approximate gradient formulation (7.85) of Chapter 7 is used. The input is again the acceleration with a limit as in Type 2 and Type 3. One of the Monte Carlo run results is given as an example in the following figures. Monte Carlo result of 20 run is given in Table 1.

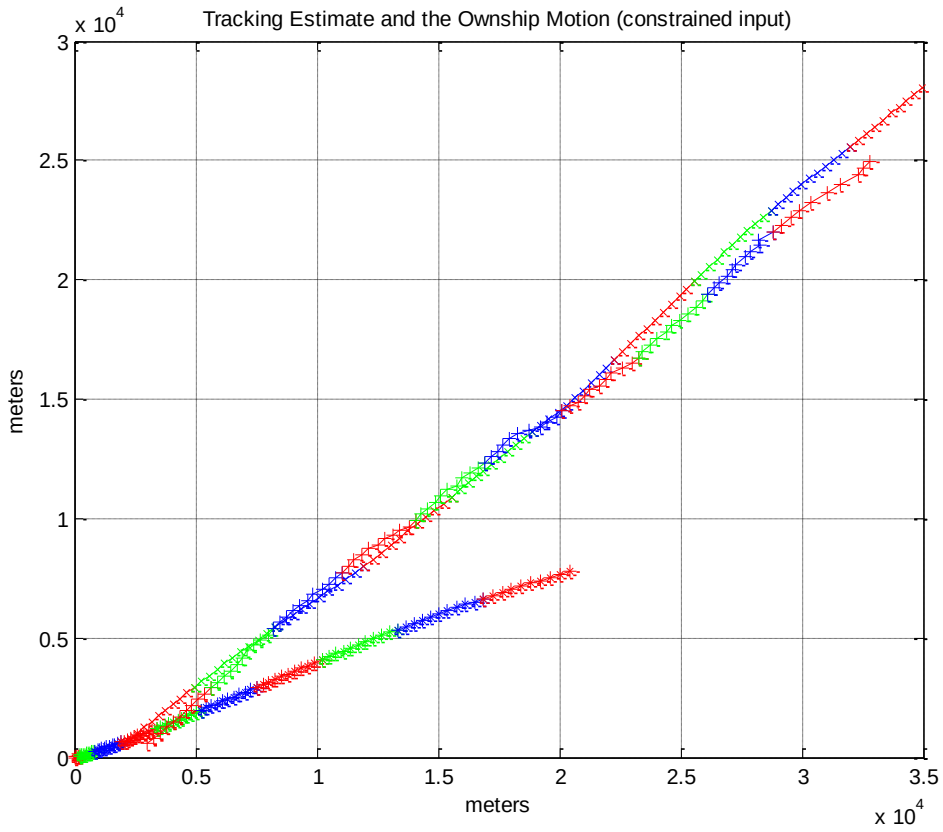


Figure 28- Type 4 limited acceleration of 0.005 m/s^2. Target position (x), Ownship (*), estimated target position (+) in meters.

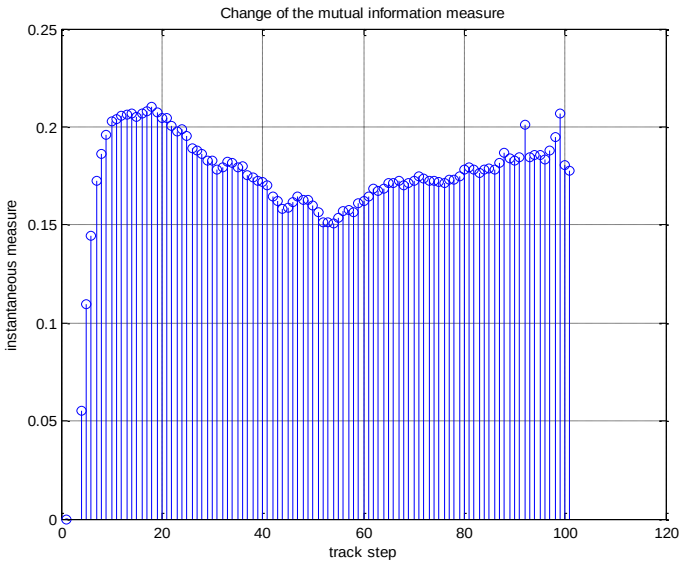


Figure 29- The change of the information measure M_k during tracking for Type 4 with limited acceleration of 0.005 m/s^2.

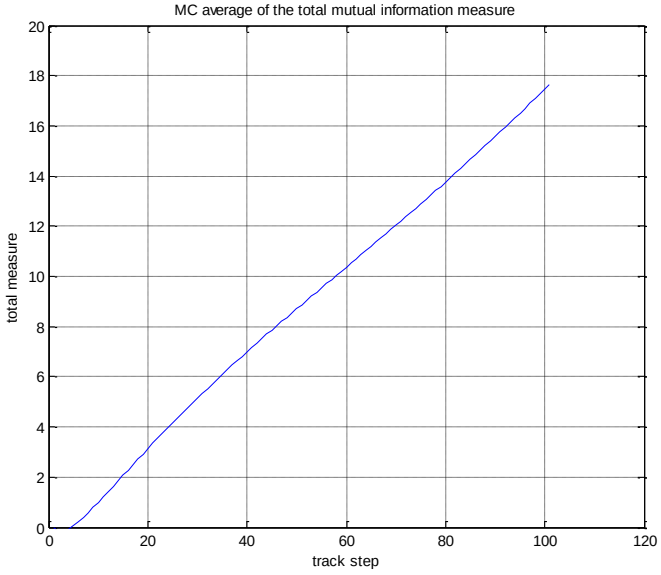


Figure 30- The total measure during the tracking of algorithm Type 4 for limited acceleration of 0.005 m/s^2.

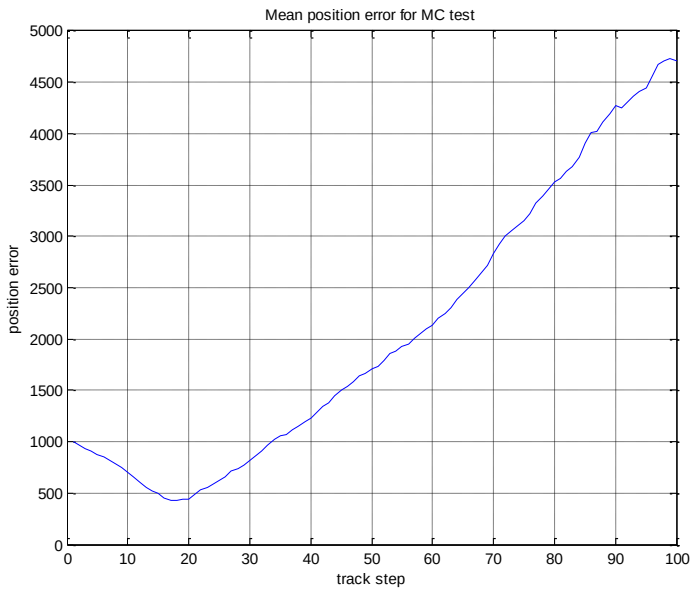


Figure 31- The mean position error for Type 4 with limited acceleration of 0.005 m/s^2.

8.2.2 Algorithm Type 5

This algorithm uses constant speed as input instead of an acceleration input. The gradient of M_k is calculated as given in (7.110) of Chapter 7. The algorithm may be implemented such that, when the range increases above a given threshold due to the relatively high target velocity compared to the ownship velocity, the constant ownship speed is doubled. This modification increases the performance if the observer is far behind the target and is able to speed up. Figure 32 and Figure 33 show the effect of the modification in speed. The speed switching is not used in the simulations to make a fair performance comparison with the other algorithms.

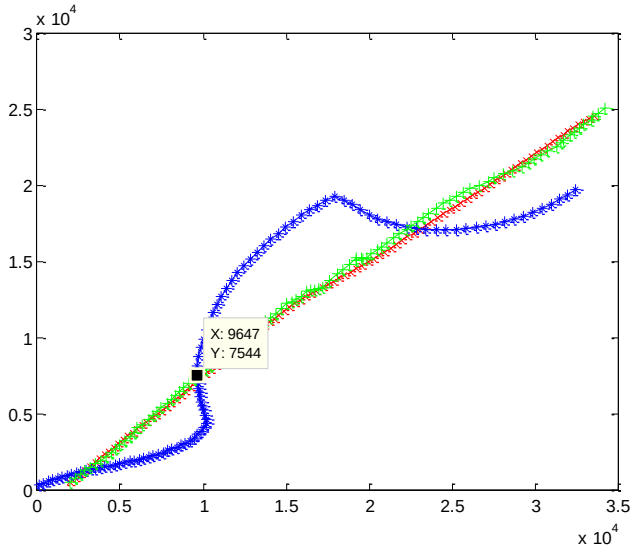


Figure 32- Tracking for algorithm Type 5 with a speed constraint of 10 m/s for the ownship. The speed is doubled if relative range is more than a threshold. This helps the observer to redirect itself its heading towards the target. Target position (x, red), Ownship (*, blue), estimated target position (+, green) in meters.

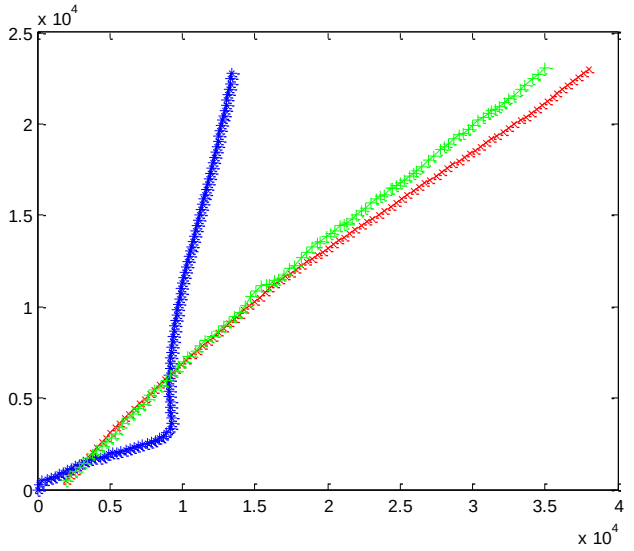


Figure 33- Tracking for algorithm Type 5 with a speed constraint of 10 m/s for the ownship when the speed is not doubled. Due to the limited speed the observer is not able to redirect itself towards the target as the relative range increases. Target position (x, red), Ownship (*, blue), estimated target position (+, green) in meters.

Figure 34, Figure 35, Figure 36 are the corresponding measure and error values for the tracking in Figure 32.

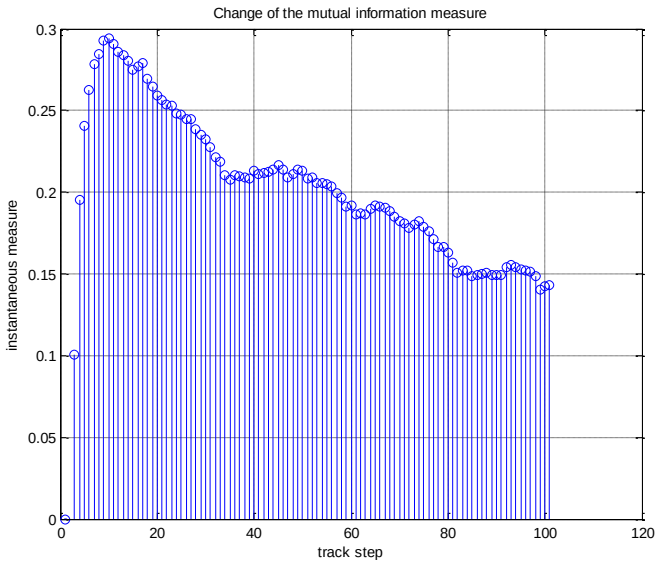


Figure 34- The change of the information measure M_k during tracking for Type 5 with limited speed of 10 m/s.

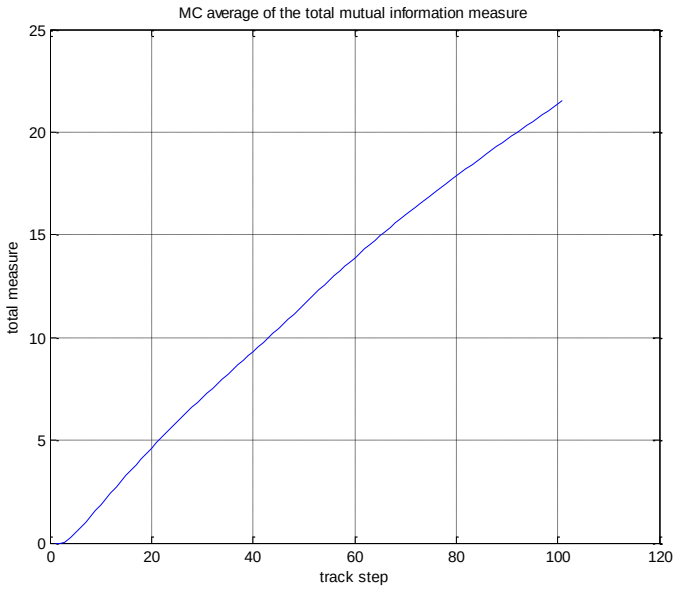


Figure 35- The total measure value during the track for Type 5 with limited speed of 10 m/s.

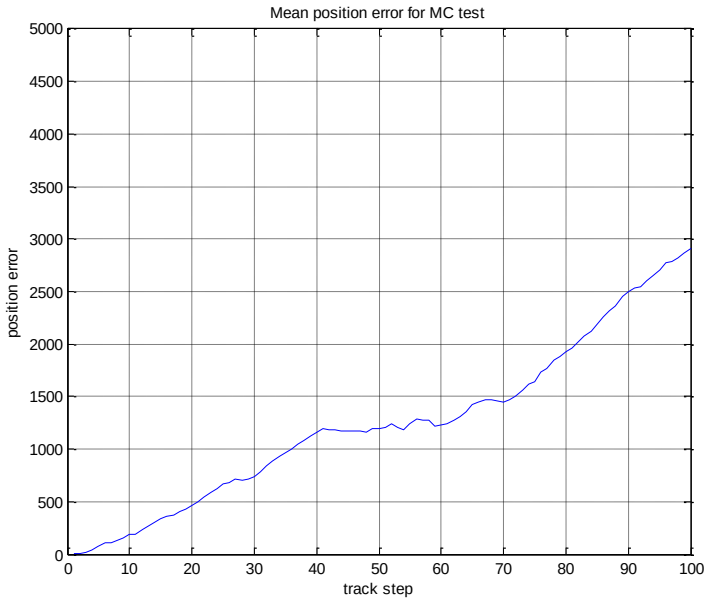


Figure 36- The mean position error for Type 5 with limited with limited speed of 10 m/s.

All of the above algorithms optimize the one step value of the mutual information. Even for the limited acceleration the ownship may have extremely large velocities which are not realistic. The high speed of the ownship also causes large deviations between the ownship trajectory and the target trajectory. Once this occurs it becomes very difficult for the ownship to turn into the direction of the target to obtain a better state estimation using the limited acceleration. This is a result of using time horizon of the optimization process as one, in which the future positions of the target are not taken into account. Note that this situation does not occur for the unconstrained input case since the optimization problem reduces into a decoupled form.

8.3 Finite Horizon Input Optimization Based on the Mutual Information for BOT

The experiments done in this section are based on the idea given in Section 5 of Chapter 7 that uses a finite time horizon that need not be equal to one. The advantage of such a method is that the mutual information is tried to be maximized considering the whole time horizon. The other methods introduced so far in this thesis, aim to increase the change in the mutual information at each step locally without

considering the whole trajectory. Even though this is acceptable in some practical cases, optimization using a finite horizon has much better performance in the expense of increased computation. Since the trajectory of the target for the horizon is predicted at the initial time it is necessary to limit the horizon length to a value that the trajectory has almost constant velocity.

To reduce the computation and create an infrastructure to implement such an optimization, input is assumed to be such that the ownship can only be on the grid nodes spanning the space. The assumption that we have made is that at each step the motion can be from one grid to a neighboring grid. The grid positions must be arranged according to the estimated speed of the target as well as the capability of the ownship. One more comment may be on the capability of the ownship for making large turns in a small time. Such a constraint is not implemented in this study. The finite horizon input optimization algorithm is referred as algorithm Type 6.

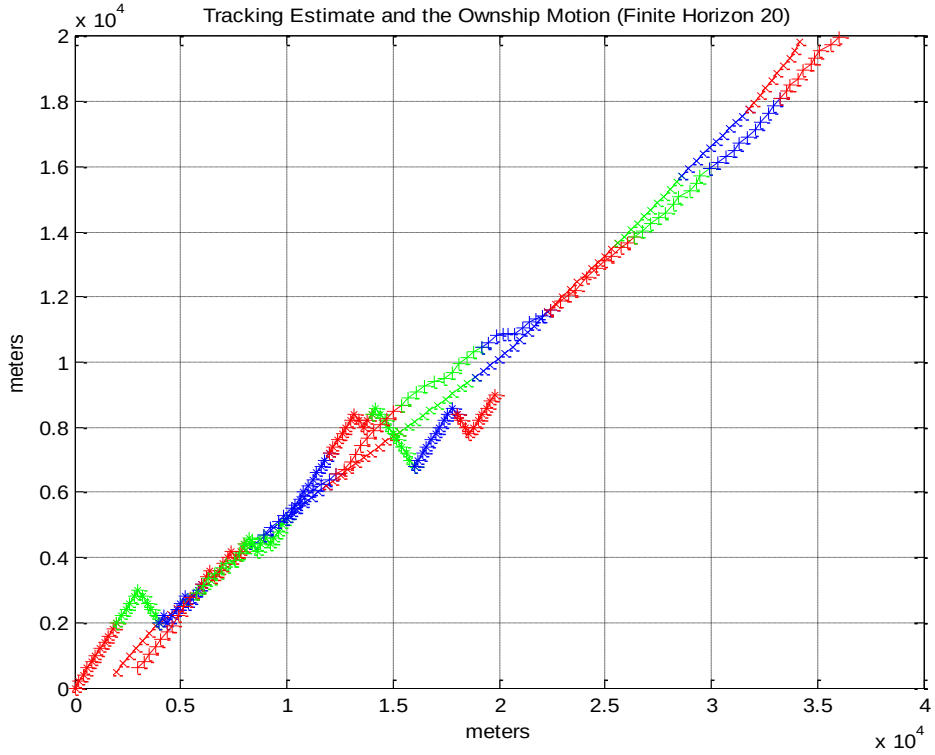


Figure 37- A finite horizon optimization example using algorithm Type 6 with horizon length of 20 steps. As the horizon is increased tracking error reduces and it is seen that the observer travels along the target direction to decrease the gap. Target position(x), Ownship(*), estimated target position(+) in meters.

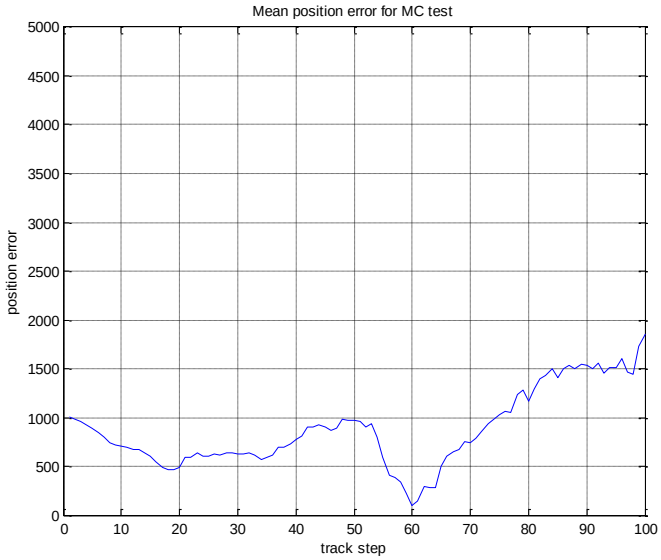


Figure 38- The average position error for Type 6. Since the optimization is performed for a horizon rather than one step the tracking error is low.

8.4 Evaluation of the Results

To evaluate the developed algorithms Monte Carlo simulations are performed. For the algorithms that are used for one step optimization for the constrained input case (Type 2 , Type 3, Type 4), various limit values are applied during the simulations. For Type 5 we assume that the optimization is an unconstrained one and the input is not the acceleration but the direction of the ownship that moves with a constant speed. To be fair for the two different input types speed of the ownship for Type 5 is selected such that the ownship has similar displacement characteristics with the other types.

It must be emphasized that the aim is to increase the tracking accuracy. Therefore the Monte Carlo average of the position error in meters for the complete track sequence should be investigated. Related average position error graphs of different trackers are given for the cases described in Table 1 and Table 2.

But the tables show the average position error only at the final time for simplicity. The final value of the mutual information measure is also given. Although only the final value of the information measure and the position error does not show the whole tracking performance, it is informative to some extent. A high final value of the measure is an indication of better tracking. State error can be defined as the norm of the difference of the estimated and the actual state vectors that includes the velocities as well as the positions. Only the position part can also be used as "error" so tracking accuracy can be compared using the position error.

Considering Table 1, simulations 1 and 5 has the Type 2 algorithm with acceleration limits of 0.003 m/s2 and 0.05 m/s2. It is observed that with the increased limit, the tracking error is reduced and the final measure is increased. This can also be observed for Type 3 in simulations 2 and 3. These results are the expected results since for the higher acceleration limits the observer has the chance of getting closer to the target. Furthermore both results indicate that there is a high correlation between the observability measure and the error, i.e., the measure increases when the tracker accuracy increases. The unlimited input as given by Type 1 has the most accurate tracking results and higher mutual information measure (Figure 21-22).

It is seen that for the unconstrained case, the ownship has to move to the predicted position of the target to increase the reward function and hence the mutual information. The experiments done for this case show almost perfect tracking that justify the use of the proposed objective function. This interesting result is consistent with the existing results in the literature derived for deterministic constant velocity target motion: Range should be minimized and the bearing angle rate should be kept high [32]. Note that coming close to the target also increases the bearing angle rate. Also note that for the unconstrained case the one step and the N step optimizations give the same result since objective function is the summation of decoupled one step optimization problems.

Type 2 is the gradient free version of Type 3 but still involves the approximation used by the linearization of the mutual information. The reward function, which is

derived using the linear form of the mutual information have to be evaluated in almost all feasible input values. This is achieved by constructing a grid and evaluating the objective function on the grid nodes. This is computationally intensive and performance is limited by the grid resolution. When simulations 1 and 2 are compared it is seen that Type 3 has better performance than the Type 2. Type 3 is the alternative gradient approach. This can be explained by the limited grid resolution of Type 2.

Type 4 does not use the linear form of the mutual information. Here we assume that the system is nonlinear and no linearization is applied to the objective function. To solve the problem, recursive integral form of the objective function and its derivative is used. The particle filter approach allows the use of the particles to approximate the gradient. Simulations 3 and 4 are for the limit of $0.02 m/s^2$ acceleration for Type 3 and 4. Type 4 has a better performance when compared to Type 3. The linearization utilized for calculating the gradient of the mutual information function can be considered as the factor limiting performance for Type 3 compared to Type 4.

Different from others, Type 5 uses the ownship direction as the input and utilizes a constant speed. Therefore direct comparison is not possible with the previous types. Table 2 shows Monte Carlo simulations and related graphs are given in succeeding figures. It is clearly observed that as the speed variable increases the position error decreases and the mutual information measure increases.

Type 6 is the dynamic programming approach. For a total track sequence of 100 steps, the required computation is quite high because the length of the horizon exponentially increases the computation time. To reduce the computation time, horizon is taken as a smaller period than the whole tracking duration of 100 steps. Horizon is chosen as 5 steps in our first simulations. Hence, for the total tracking of 100 steps, the optimization of the input is repeated 20 times for the horizon of 5 steps. The grid size taken for the simulations is 200 m in both directions.

For a 5 step horizon case the final position error (at the time step 100) is 3908 meters (Figure 47). When the horizon is increased to 20 steps the position error is reduced to 1858 meters (Figure 48). This supports the claim we stated as that the estimation

accuracy provided by the finite horizon programming should be better as long as a sufficiently long horizon interval is selected. Also even though there cannot be a direct comparison due to different input constraint type of the models of the other algorithms, the position errors observed by this method are smaller than the previous one step optimization algorithms as expected.

Table 1- Monte Carlo runs and the final position errors for Type 2, Type 3 and Type 4 algorithms.

Monte Carlo Simulation Index	Algorithm Type	Acceleration limit (m/s^2)	Final Position Error (m)	Final Mutual Information
1	Type 2	0.003	5817,00	17,50
2	Type 3	0.003	4862,00	17,74
3	Type 3	0.02	2660,00	22,35
4	Type 4	0.02	1624,00	22,69
5	Type 2	0.05	2176,00	20,54
6	Type 4	0.05	1556,00	22,64

Table 2- Monte Carlo runs and the final position errors for Type 5.

Monte Carlo Simulation Index	Algorithm Type	speed	Final Position Error	Final Mutual Information
7	Type 5	8 m/s	4207	19,86
8	Type 5	10 m/s	2903	22,06

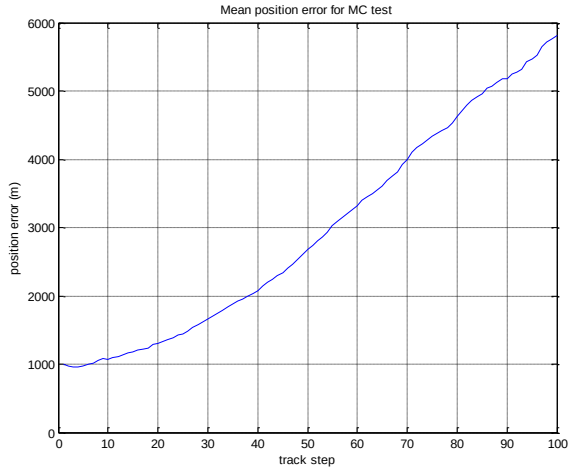


Figure 39- Average position error during tracking for MC simulation 1 where the input limit is 0.003 m/s^2 and the algorithm used is Type 2.

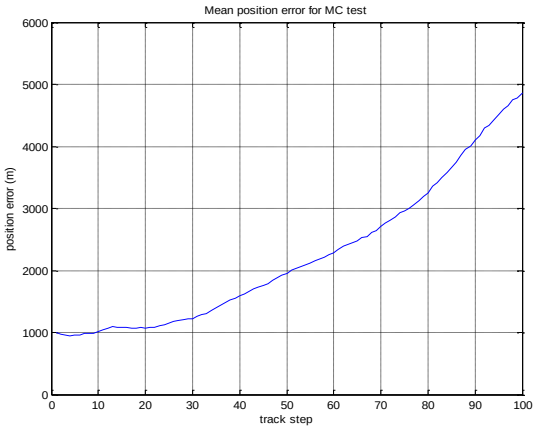


Figure 40- Average position error during tracking for MC simulation 2 where input limit is 0.003 m/s^2 and the algorithm used is Type 3.

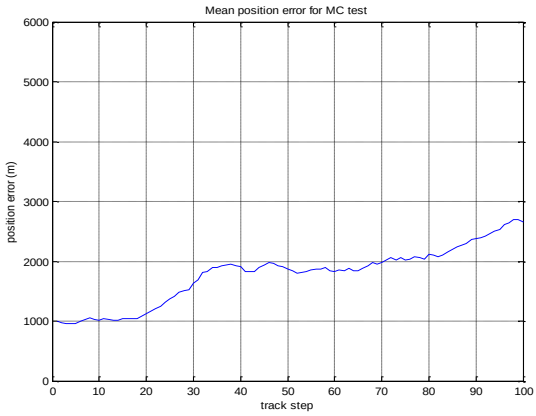


Figure 41- Average position error during tracking for MC simulation 3 where input limit is 0.02 m/s^2 and the algorithm used is Type 3.

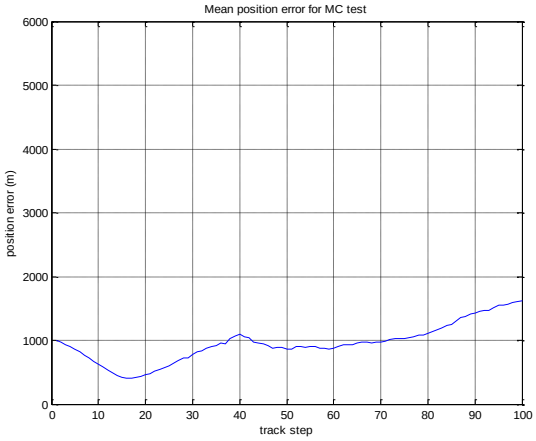


Figure 42- Average position error during tracking for MC simulation 4 where input limit is 0.02 m/s^2 and the algorithm used is Type 4.

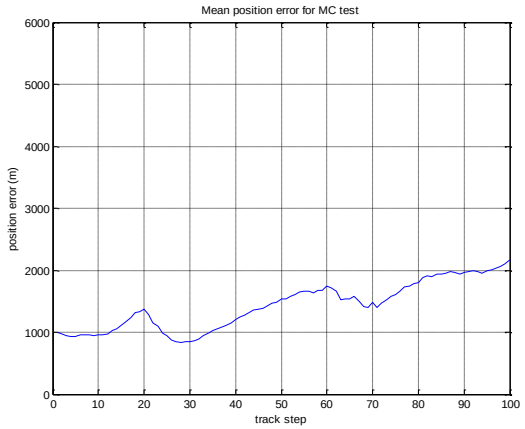


Figure 43- Average position error during tracking for MC simulation 5 where input limit is 0.05 m/s^2 and the algorithm used is Type 2.

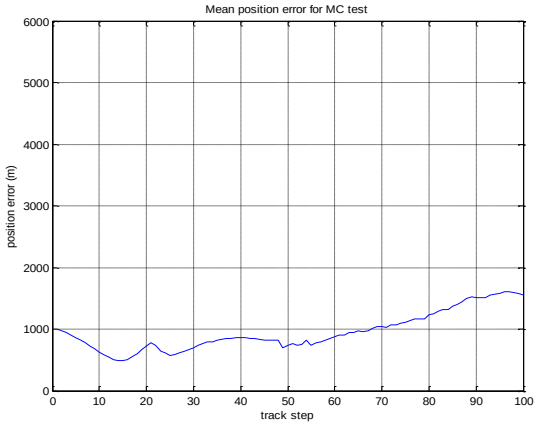


Figure 44- Average position error during tracking for MC simulation 6 where input limit is 0.05 m/s^2 and the algorithm used is Type 4.

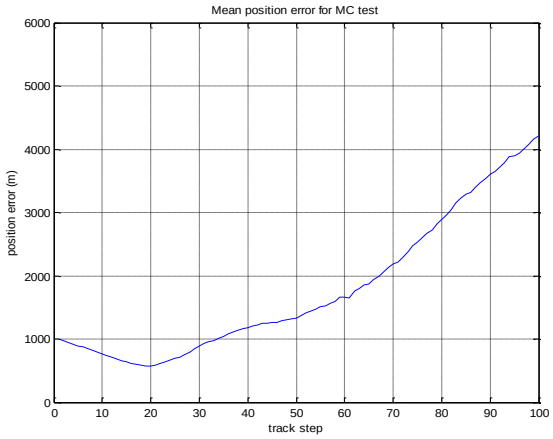


Figure 45- Average position error during tracking for MC simulation 7 where ownship speed is 8 m/s and the algorithm used is Type 5.

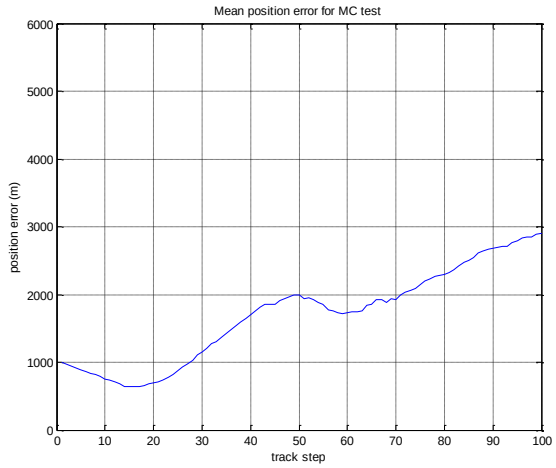


Figure 46- Average position error during tracking for MC simulation 8 where ownship speed is 10 m/s and the algorithm used is Type 5.

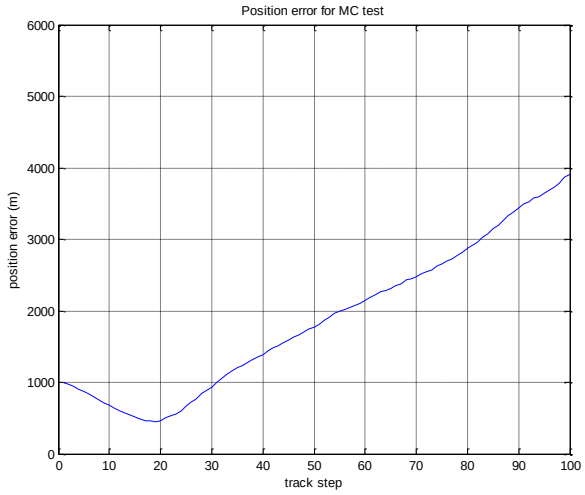


Figure 47- Average position error during tracking for MC simulation of algorithm Type 6 where ownship moves in a grid space with 200 meters intervals. Finite horizon optimal path programming is done for a horizon of 5 repeatedly.

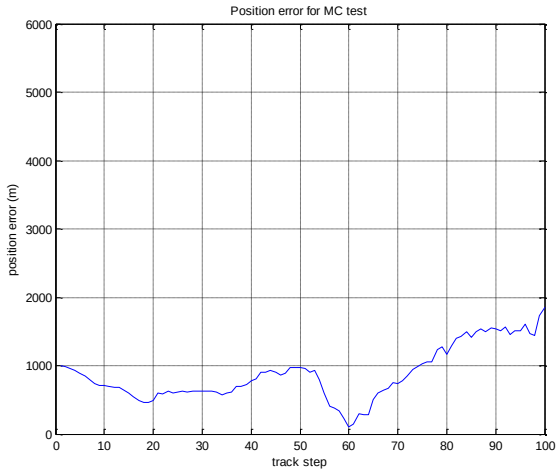


Figure 48- Average position error during tracking for MC simulation of algorithm Type 6 where ownship moves in a grid space with 200 meters intervals. Finite horizon optimal path programming is done for a horizon of 20 repeatedly.

Two comments can be made for algorithm Type 6. If the horizon is set to a very high value, target motion prediction may be very poor for the far future which deteriorates the performance of the method and the expected performance increase cannot be observed.

Secondly since the ownship is constrained with the grid positions, ideal input displacement can never be possible. So the algorithm is still a suboptimal approach. Increasing grid resolution may solve this problem with the expense of exponentially increased computational cost.

CHAPTER 9

CONCLUSIONS

Bearings-only tracking is a problem with a wide application area. It is not only practically valuable but also theoretically challenging. The theoretical challenge is to estimate the states of a system which are not “observable”. Earlier studies show that model selection becomes a more important issue for BOT problem compared to some other “observable” systems. Furthermore observability of stochastic systems is not a well defined subject that a consensus is established.

Another important subject that we have studied is the optimal input design that is the optimal maneuver of the observer. This problem is still open. We have used a new observability measure as an objective function that must be maximized to solve the optimal observer trajectory problem.

Some methods are proposed in this thesis and some algorithms are developed to track a target by only a bearing measurement for the 2-D case. Beside the algorithms that we have generated we also gained some valuable information about the inherent structure of the problem. We believe that the study on optimal trajectory generation of the observer sheds some light on the nature of the problem.

Another challenge that we have encountered is to model the maneuvering target motion that can be solvable by a reasonable computational complexity using particle filters.

The first study given in Chapter 5 is about using efficient models to decrease the tracking error after the maneuver of the observer. The target is assumed to be a maneuvering one. The method generated is the Range Parameterized Interacting

Multiple Model Gaussian Mixture Filter (RPIMMGM). It is shown that this filter improves the tracking performance compared to the RP based filter. It also provides the capability of maneuvering target tracking. The increase in the tracking accuracy is shown by comparing it with an IMM based BOT tracker.

Second algorithm is also developed for maneuvering targets. The Range Parameterized Coordinated Turn Rate Marginalized Particle Filter (RPCTMPF) has the ability of using a particle filter in a Rao-Blackwellized fashion where nonlinear component of the state is the one dimensional turn rate. This choice decreases the computational need compared to a standard particle filter of similar performance. In this model turn rate is assumed to be non-Gaussian. An improvement is gained by non-Gaussian modeling of the turn rate. The improvement is especially visible when a constant velocity target starts to a coordinated turn maneuver. Another effective improvement is obtained by adapting the maximum and minimum range limits for the range parameterization during tracking. Better adaptation of the filter to the maneuver and better position accuracy is observed after these modifications.

The contribution of the second algorithm is the idea of using turn rate as a bulk parameter for each sub-track of a maneuvering target. In this thesis, this idea empowers to the marginalization of the state during state estimation. The range adaptation during tracking is another novel part of the algorithm which has a great effect on the performance.

The principle aim of the thesis is to improve the BOT accuracy. Another aspect of increasing the tracking accuracy is to improve the ownship motion. We have attacked the observer trajectory generation problem as an optimization problem where the objective function is an information theoretic observability measure based on the mutual information between the state and the measurement sequences. Although there are various studies in the literature that aims to increase observability of BOT, use of information theoretical approach is very limited.

The definition and the theory generated in [36] is applied to the BOT problem. In [36] observability is defined as the mutual information between the last state and the input for linear systems. The application of this theory is by no means a trivial one.

Especially the constraints on the input make the problem difficult. In this study first the mutual information, formulated for the linear case [36] is taken into consideration. Using linearization of the BOT equations, the linear formulation is reformulated resulting in an objective function which has to be maximized. The one step optimization problem is solved for the ownship motion. The ownship maneuver characteristic is investigated using this derivation. The experiments done for this case show almost perfect tracking that justify the use of the proposed objective function. This interesting result is consistent with the existing results in the literature derived for deterministic constant velocity target motion.

After the unconstrained problem, constrained input case is examined. For the constrained input case, a one step optimization for the ownship maneuver is described using the gradient of the objective function derived for the BOT problem.

One recent study on the observability problem for both linear and nonlinear stochastic systems proposes a new measure [27]. This is the mutual information between the state and the observation sequences and is formulated recursively for the nonlinear systems. Another contribution done in the thesis can be stated as obtaining of a simplified measure using this representation of mutual information for nonlinear systems. This measure has direct dependency on the ownship maneuver. The maneuver selection to maximize the measure is handled by two approaches.

When the gradient of the simplified measure is examined it is seen that further simplifications are possible and a compact formulation for the gradient is obtained in this study. To do this, particles of the particle filter of the active tracker is used. The particle filter is already deployed for the state estimation of the BOT problem.

Finally, the simplified measure is proposed as a reward function for a finite horizon problem. A finite horizon optimization problem is formulated and solved for the ownship motion, using the technique of dynamic programming. The finite horizon formulation which is another contribution of the thesis improves the optimization compared to one step optimization alternatives.

REFERENCES

[1] J. C. Hassab, "Underwater Signal and Data Processing", Boca Raton, FL: CRC Press, 1989

[2] S. C. Nardone, A.G. Lindgren, and K. F. Gong, "Fundamental Properties and Performance of Conventional Bearings-only Target Motion Analysis", IEEE Trans. Automatic Control, vol. 29, no. 9, pp. 775-787, 1984

[3] C. Jauffret, D. Pillon, "Observability in Passive Target Motion Analysis", IEEE Trans. Aerospace and Electronic Systems, vol.32 pp. 1290-1300, 1996

[4] P. Shar, X. R. Li, "A Practical Approach to Observability of Bearings-only Target Tracking," Proc. SPIE, vol.3809 pp. 514-520, 1999

[5] T. L. Song, "Observability of Target Tracking With Bearings-only Measurements," IEEE Trans. Aerospace and Electronic Systems vol.32 pp.1468-1471, 1996

[6] E. Fogel, M. Gavish, "Nth-order Dynamics Target Observability from Angle Measurements," IEEE Trans. Aerospace and Electronic Systems, vol. AES-24, pp. 305-308, 1988

[7] T. L. Song, "Practical Guidance for Homing Missiles With Bearings-Only Measurements", IEEE Trans. Aerospace and Electronic Systems, 32 pp. 434-443, 1996

[8] W. Grossman, "Bearings-only Tracking: A Hybrid Coordinate System Approach", AIAA Journal of Guidance, Control, and Dynamics, 17, 3 pp. 451-457, 1994

[9] V. J. Aidala, S. E. Hammel, "Utilization of Modified Polar Coordinates for Bearings-only Tracking", IEEE Transactions on Automatic Control, AC-28, 3 pp. 283-294, 1983

[10] S. S. Blackman, S. H. Roszkowski, "Application of IMM Filtering to Passive Ranging", Proc. SPIE, vol.3809 pp. 270-281, 1999

[11] Y.Bar Shalom and X. Li. Estimation and Tracking: Principles, Techniques, and Software, Artech House, 1993

[12] S.S. Blackman and R. Popoli, Design and Analysis of Modern Tracking Systems, Artech House, 1999

[13] B. Ristic, A. Arulampalam, N. Gordon, Beyond the Kalman Filter: Particle Filters for Tracking Applications, Artech House, 2004

[14] Tina Erlandsson, "Angle-Only Target Tracking", Thesis, Linkoping, 2007

[15] T.R. Kronhamn, Bearings-only target motion analysis based on multihypothesis Kalman filter and adaptive ownership motion control. IEE Proceedings: Radar, Sonar and Navigation, 145(4), pp. 247-252, 1998

[16] D. Musicki, Bearings-only single-sensor target tracking using Gaussian mixtures. Automatica 45 (elsevier) 2009

[17] M. S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking" Signal Processing, IEEE Transactions on Signal Processing, Volume 50, Issue 2, pp. 174–188, 2002

[18] B. Ristic, A. Arulampalam, N. Gordon, "Tracking a maneuvering target using angle-only measurements: algorithms and performance", Signal Processing, 83 pp. 1223-1238, 2003

[19] J. K. Uhlmann, "Simultaneous map building and localization for real time applications," Transfer Thesis, Univ. Oxford, Oxford, U.K.,1994.

[20] S. J. Julier, J. K. Uhlmann, "Unscented filtering and nonlinear estimation", Proceedings of IEEE, vol:92 no:3 March, 2004

[21] D.L. Alspach, H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations", IEEE Transactions on Automatic Control, AC-17, August, 1972

[22] Arnaud Doucet, Nando de Freitas, Neil Gordon "Sequential Monte Carlo Methods in Practice", Statistics for Engineering and Information Science, Springer, 2001

[23] M. R. Morelande, S. Challa, "An Algortihm for Tracking Group Targets", in Proc. Workshop on Multiple Hypothesis Tracking: A Tribute to S. Blackman, (San Diego, CA) May 2003

[24] T. Schön, F. Gustafsson, "Marginalized Particle Filters for Mixed Linear/Nonlinear State-Space Models", IEEE Trans. On Signal Proc., vol. 53 no.7, pp. 2279-2289 July 2005

[25] D. Musicki, "Bearings only muti-sensor maneuvering target tracking", ScienceDirect, System and Control Letters, vol. 57, pp. 216-221, 2008

[26] Y.B. Shalom, X. Li. Multitarget-Multisensor Tracking: Principles and Techniques YBS 1995

[27] Y.Subasi, "Quantitave Measures of Observability for Stochastic Systems", Thesis, Middle East Technical University, 2012

[28] J. M. Passerieux, D. Cappel, "Optimal observer maneuver for bearings-only tracking", IEEE Trans. Aerospace and Electronic Systems, vol.34 no.3 pp. 777-788, 1998

[29] J. P. Le Cadre, D. Tremois, "Optimization of the observer motion using dynamic programming", IRISA/CNRS RENNES , 1995

[30] R. E. Kalman, "On the General Theory of Control Systems", Proc. 1st Int. Cong. Of IFAC, Moscow 481-492, 1960

[31] T. Brehard, J.-P. Le Cadre, "Closed-Form Posterior Cramer-Rao Bounds for Bearings-Only Tracking", IEEE Transactions on Aerospace and Electronic Systems Vol. 42, NO. 4, 2006

[32] G. J., Olsder, "On the optimal maneuvering during bearings-only tracking.", In Proceedings of 23rd Conference on Decision and Control, Las Vegas, NV, pp. 935-940. 1984

[33] B. J., MacCabe, "Accuracy and tactical implications of bearings-only ranging algorithms.", Operation Research, 33, 1 pp. 95-109, 1985

[34] J. A., Fawcett, "Effect of course maneuver on bearings-only range estimation.", IEEE Transactions on Acoustic, Speech and Signal Processing, 36, 8, 1988).

[35] A. Logothetis, A. Isaksson, R. J. Evans, "An Information Theoretic Approach to Observer Path Design for Bearings-Only Tracking", Proceedings of the 36th Conference on Decision & Control San Diego, California USA, 1997

[36] T. Huang, B. Chen "Information evolutions in the linear stochastic control systems.", Proceddings of the 7th World Congress on Intelligent Control and Automation, Chongqing,, China, 3384-3387, 2008

[37] R. R. Mohler, C. S. Hwang, "Nonlinear data observability and information", Journal of The Franklin Institute, vol. 325, no 4, pp. 443-464, 1988

[38] J. Xu1, X. Feng, S. Ren, and J. Zeng, "Comparison of SUT H∞ − and UKF in Cartesian and Modified Polar Coordinates for bearings-only tracking", 0-933957-39-8, 2011 MTS

[39] T. Brehard, J.-P. Le Cadre "Hierarchical Particle Filter for Bearings-Only Tracking", IEEE Transactions on Aerospace and Electronic Systems Vol. 43, NO. 4, 2007

[40] P. Tichavsky, C. Muravchik, A. Nehorai, “Posterior Cramer-Rao bounds for discrete-time nonlinear filtering.”, IEEE Transactions on Signal Processing, 46, 5, pp. 1386-1396, 1998

[41] C. Jauffret, D. Pillon, A-C. Pignol, “Bearings-only Maneuvering Target Motion Anlaysis from a Nonmaneuvering Platform”, IEEE Transactions on Aerospace and Electronic Systems Vol. 46, NO. 4, 2010

[42] T. M. Cover, J. A. Thomas, Elements of Information Theory, 2nd Edition, A John Wiley & Sons, Inc. New Jersey, USA, 2006

VITA

PERSONAL INFORMATION

Surname, Name: Arslan, Ali Erkin
Nationality: Turkish (TC)
Date and Place of Birth: 04 December 1977, Elazığ
Marital Status: Single
Phone: +90 533 231 80 66
email: erkinarslan@yahoo.com

EDUCATION

Degree	Institution	Year of Graduation
MS	METU Electrical and Electronics Engineering	2003
BS	METU Electrical and Electronics Engineering	2000
High School	Bursa Science High School, Bursa	1995

WORK EXPERIENCE

Year	Place	Enrollment
2000-2012	ASELSAN, Microelect., Guidance & Electro-Optics Div.	Lead Engineer

FOREIGN LANGUAGES

Advanced English
Fluent French

PUBLICATIONS

Arslan, A. E., Demirekler M. "Visual Tracking with Group Motion Approach", Conference on Computer Vision and Pattern Recognition Workshops, Vol. 8, 2004

HOBBIES

Cycling, Tennis, RC Helicopters