Imperial College London

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

Tracking with Bearing and Range measurements

Author:

Haotian Wu

Supervisor:

Richard Vinter

Submitted in partial fulfillment of the requirements for the MSc degree in Master of Research of Imperial College London

September 2018

Abstract

This thesis focuses on a class of nonlinear filter for the tracking problems with noisy bearing and range measurements. And the aim of the project is to find some efficient and robust algorithms for the tracking problems with bearing and range measurements.

An unified 'Analytic filter' is extended in this thesis for the nonlinear tracking problems with noisy bearing and range measurements. In this thesis, we apply the filter into three typical classes of the tracking problems: Bearing-Only tracking problems, Rang-Only tracking problems and Bearing-Range Mixture tracking problems.

Two typical themes of this 'moment matching' filter this thesis introduced: changing noise model into 'noise before nonlinearity' and approximating the conditional densities. The noise model is changed into the 'noise before nonlinearity' in this filter. The new noise model provides the possibility to compute the density of the 'augmented vector' directly. And the approximation uses 'moment matching' method, using a Gaussian variable to approximate the state. In this way, we can get the recursive formulation of the estimation.

This filter is based on the success of the 'Shifted Rayleigh filter', and we extended it into the bearing-range mixture cases, and it works very well in some scenarios. This filter converges quickly and works well in some common scenarios with noisy bearing and range measurements. Compared with other nonlinear filters, the lower computation demand and its robust convergence are two main advantages of this filter. The computation in this filter is trigonometric and matrix computation, which is easy to compute. And this filter calculates the conditional density by direct and accurate way make the results very accurate and robust.

Acknowledgments

My sincere gratitude goes to Professor Richard Vinter. It is his encouragement and kindness that help me complete the postgraduate project. Professor Vinter recommended lots of useful resource of the filter to me. Relating books and papers, regular meetings and valuable suggestions from him all make really big difference. And I am really honored that I can do my postgraduate project under his supervision and guidance. Really grateful and lucky to have chance to further my education on the topics which I am interested in.

This thesis is based on the success of the Shifted Rayleigh filter and Mixture Gaussian filter and both of them are mainly designed by the Professor Vinter. At first, the aim of the project is only to try the bearing-only tracking, and Professor Vinter gives me the chance to further this topic. Really thanks for trusting my competence and giving me the chance to try extending this analytic filter into new area. And really grateful to acknowledge the work did by the people before. It is awesome to stand on the shoulders of giants.

I would also like to thank my parents who have always put my education above all of their priorities. And support my further education and encourage me to study aboard.

Last but definitely not least, this past year would not have been the same without the support from the love of Yuxin Zhang, and friendship from Yifeng Han, Junkai Wang and Zhe Cao.

Standing On the Shoulders of Giants.

Contents

List of Tables

List of Figures

Nomenclature

$\gamma_{s:t}$ The sequence of time-independent variables: $\gamma_s, \gamma_{s+1} ... \gamma_t, \gamma_s$

$\hat{x}$ The estimation of a $n \times 1$ vector x

$|d|$ The Euclidean length of vector $d = \sqrt{d_1^2 + d_2^2 + ... + d_n^2}$

d The displacement vector (2×1 in two dimension) between the target and the observer platform $\boldsymbol{d} = \boldsymbol{x_{target}} - \boldsymbol{x_{observer}}$

$E(x)$ The mean of the random variable x

F The system $n \times n$ matrix describing the motion

H The augmented measurement $k \times n$ matrix

$N(\hat{x}, P)$ The Multivariate Gaussian density with mean $\hat{x}$ and covariance P

$p(x)$ The probability density of the random variable x

$p(x|y)$ The conditional probability density of the random variable x given y

Q^m The covariance $k \times k$ matrix of the measurement noise process

Q^s The covariance $n \times n$ matrix of the system noise process

$R_x(s)$ The autocorrelation of the white process x with time difference s, $R_x(s) = E(X_{t+s}X_t)$

u_t^m The measurement input signal ($k \times 1$ vector)

u_t^s The system input signal ($n \times 1$ vector)

$v_t \sim N(0, Q^s)$ The system noise process

$w_t \sim N(0, Q^m)$ The measurement noise process

Chapter 1

Introduction

1.1 General Introduction

The thesis focuses on a class of nonlinear filter used in tracking problems. And the aim of the project is to find some computationally efficient and robust algorithms for the tracking problems with nonlinear noisy measurements like bearing and range measurements.

The main objective of the tracking problems is to estimate the states of the target with the information of the noisy measurement and the initial state. And these tracking problems are known as Bayesian target tracking. The estimation process is usually nonlinear for the characteristics of the measurement sensors. The most common sensors used in this kind of problem can get the the angle information or the range information of the target. And all these equations of the information are nonlinear functions of the state. So the computation and the estimation process become complex, and the efficient algorithms are required in this area.

The thesis is concerned with three typical classes of the tracking problems: Bearing-Only tracking problems, Rang-Only tracking problems and Bearing-Range Mixture tracking problems. The angle measurement and the distance measurement are the most common measurements in real life. Radar and GPS are developed fast recent years and can offer the accurate information of targets quickly. But in some special cases, we also need to consider the degenerate configurations of the measurements, where measurements can only offer very little information of the target. For example, in the deep water environment, the AUV (autonomous underwater vehicle) can only have the accurate bearing information. In the robotics applications, simultaneous localization and the navigation all need the range-only tracking algorithm, so the Range-Only tracking problems arises.[18] And in some case the range and bearing information are all available, so the Bearing-Range Mixture tracking problem also attracts some interests.

In Chapter 1, this thesis will first talk about the target tracking and the filter problems, explain the tracking and filter problem we faced and then focuses on some general filter algorithms used in the tracking problems. In this part, we will state the different categories of the filter, compare the advantage and disadvantage of the different algorithms. This thesis will concern 'analytic moment matching' filter in later (tracking problems based on the probability density calculation instead of the construction of the empirical distribution).

In Chapter 2, we will focus on the general 'Analytic filter' and its construction. We will propose the models of the problems, and apply the 'Analytic filter' into the tracking problems in general. We also will analyze and justify the algorithm in this chapter.

In Chapter 3, the thesis pays special attention to the Bearing-Only tracking problems. And we apply the analytic filter (also known as the Shifted Rayleigh filter in bearing-only tracking) into this kind of tracking problems. And justify the calculation of the parameters used in this filter. In the last, we run the simulations to check the performance of the filter.

In Chapter 4, the thesis focuses on the Range-Only tracking problems. And we apply the analytic filter in range-only tracking. Similarly, we will justify the calculation of the parameters used in this filter and run the simulation to compare the performance of the filter with other algorithms.

In Chapter 5, we will extend the analytic filter into bearing-range mixture tracking. In this chapter, We will model the noise and apply the 'analytic filter' into this problem. Also we run the simulation to compare the performance of the filter with other algorithms.

In Chapter 6, we will summarize all the situation where we apply the analytic filter algorithm and make some conclusions about the filter.

In Chapter 7, we will point out some disadvantages need to be solved and the improvement can be done in the future.

1.2 Target Tracking

Target tracking is a vast filed concerned with extracting information of the target with the sensor data. Common aims of the tracking problems include the estimation of the position, movement and the number of the targets. And recent interests arise in this area are multiple sensors environment, the location of the sensor and intelligent strategy planning. In this thesis, we focus on the nonlinear problems which arise in target tracking ,especially the position estimation problems.

The vector $(x_t, y_t, z_t)^T$ denotes the position of the target in Cartesian coordinate at time t. And the corresponding velocity vector is $(\dot{x}_t, \dot{y}_t, \dot{z}_t)^T$. Also in some special cases, higher order derivatives of the state are needed, but these two vectors can describe most of the tracking problems. These two vectors are the typical description of the states in the tracking problems.

And most motion of the target can be modeled from the basic laws of kinematics and get the function of the $(x_t, y_t, z_t)^T$,$(\dot{x}_t, \dot{y}_t, \dot{z}_t)^T$ to describe the motion. The advantage of this description is its linearity and easy calculations. The famous models described by the linear kinematic models of the state are constant velocity motion, constant accelerating motion and coordinated turn motion.

Sensors devices used in target tracking are naturally nonlinear. Example include radar, infra-red sensors, passive sonar, which typically offer the information: range (r_t), azimuth angle (θ) and elevation angle (ϕ). (3-dimension description in Cartesian position):

$$r_t = \sqrt{x_t^2 + y_t^2 + z_t^2} \tag{1.1}$$

$$\theta_t = \arctan(\frac{y_t}{x_t}) \tag{1.2}$$

$$\phi_t = \arcsin(\frac{z_t}{\sqrt{x_t^2 + y_t^2 + z_t^2}}) \tag{1.3}$$

As we can see the equation (1.1)-(1.3) are all nonlinear, which is the first challenge we need to face. And these sensor measurements are corrupted by the noise in real life, which is the second challenge. These noise are usually considered to be the additive and Gaussian. But in this thesis, we used a new model to describe the noise in this nonlinear systems, we will discuss in the Chapter2.2 latter. After modeling the noise process, we usually use the filter algorithms to eliminate the effect from the noise on the measurements.

Filter Problem is a key block of the tracking tasks, because the noise in the target tracking affects the accuracy of the results and the noise cannot be ignored in general situation. The filter algorithm can provide the results of the target tracking tasks with systematic method of information extraction under noise environment.

1.3 Filter Problem

As already mentioned, the filter problem is to compute the optimal estimation of the state under the noisy measurement related to the states. And the criterion used to decide 'what is the best estimation' depends on the different cost functions. All optimality principle are defined with the cost function, different cost function will result in the different optimal estimators.

The mean squared error, the mean absolute error and the uniform cost are three common cost functions for the Bayesian estimator. They are easily computed and defined well, the corresponding equation of the estimator are given in the Table 1.1. And The main research we did is based on the mean squared error cost function.

	Mean squared error	Uniform error	Mean absolute error
Cost Function	$E[(x-\hat{x})^2]$	$E[f(x-\hat{x})], where f(a)$ $= \begin{cases} 0 & \text{if } \lvert a \rvert < \Delta/2 \\ 1 & \text{if } \lvert a \rvert > \Delta/2 \end{cases}$	$E[\lvert x-\hat{x} \rvert]$
Estimator	$\hat{x} = \int_{-\infty}^{+\infty} xp(x\lvert y)dx$	$\hat{x} = \max p(x\lvert y)$	$\hat{x}$ is s.t $\int_{-\infty}^{\hat{x}} p(x\lvert y)dx = \int_{\hat{x}}^{\infty} p(x\lvert y)dx$

Table 1.1: Common cost functions and estimator $\hat{x}$ with given measurement y

Having the cost function and the estimation method, the filter problems can be solved easily.

The filter problems can be divided into three class:

1. 'Continuous-time state/Continuous-time measurements'
2. 'Continuous-time state/Discrete-time measurements'
3. 'Discrete-time state/Discrete-time measurements'.

The thesis focuses on the 'Discrete-time state/Discrete-time measurements' filter. In the real life, the measurements are all discrete-time, and when the state is continuous, we can do the discretization process first. In this way we can change the continuous systems into discrete-time systems.

There are many good algorithms used in the filter problems. And they can be separated into two main categories: 'The moment matching' class and the 'density computation' class.

a). 'The Moment matching' filters:

This class of filters attempt to match the first and second moments of the conditional densities of the state. Filers in this class are based on the principle: when faced with two distributions of the processes (states), if the first and second moments of the processes are very closed or roughly equaled and then we can just think these two processes are the same process, and make good estimation of the process. And the typical algorithms in the moment matching class are: the extended Kalman Filter[1], 'pseudomeasurement' approaches to bearing-only tracking[20], and Unscented Kalman Filter[16]

The most famous example in this category is the Kalman Filter (KF), which is the best choice for the linear system where the equations of the measurements and states are all linear and the noise processes can be modeled normally distributed. And it is because that Gaussian process can be described by the first and second moment completely.

Traditional ways to estimate the state in the tracking problems are Kalman filter. And when the situation is nonlinear, the estimation can be completed by linearizing the equation and then using the Kalman filter (the Extended Kalman Filter algorithm).

Based on the Kalman Filter, the Extended Kalman Filter (EKF) is proposed to handle with the nonlinear situations. In the Extended Kalman Filter, a first or second order Taylor series expansion of the nonlinear description is used for the linearizing the nonlinear part of the system around the current estimation position, so as to use the Kalman Filter Algorithm[16]. As we know the system can run linearly at local and the distributions can keep the Gaussian distribution locally, so in some cases this algorithm can be used well. And the Extended Kalman Filer also gives rise to a class of 'Moment matching' Filter: algorithm that tries to approximate the mean and the covariance of the posterior distribution and use a Gaussian density to match the moments in latter. And its computation cost will be very small, which can be the same order of magnitude with that of the linear Kalman Filter.

However, the Kalman filter sometimes will fail to get the convergent results (break down) when faced with many challenging tracking problems. For example scenarios with multiple sensors information or degenerate sensors which doesn't provide enough information of the target.

Then this results in the interests recent years in particle filter, which belongs to the 'density' category. And it exploits on-line Monte-Carlo methods to construct a empirical distribution to approximate the conditional density of the state given the measurements.

b). The 'density' filter:
The density class filter is also very useful and accurate. The most famous density filter is Particle filters[2], which is very accurate and practical. Interacting multiple model(IMM) filter[4] is also very common and accurate. The range-parameterized extended Kalman filter[21] is another density filter, which is also very practical. These three 'density' filter are most common or famous algorithms nowadays.

The particle filter is the most typical algorithm in this class. The particle filter are sequential Monte-Carlo methods approximating the probability densities of the state directly. It propagates a set of randomly drawn samples and then estimate the conditional density of the state directly. If the number of particles is big enough, it can reproduce the the exact probability density and get the estimation of the functions of the random variables(e.g. moments,etc) from the examples. And the particle methods can bring us more accurate results than the 'moment matching' filter.

Particle filter Algorithm can be used in many tracking scenarios in which the extended Kalman filter cannot do well. But the main disadvantages of general particle filters are their heavy computations and their sensitivity to the initial parameters.[22] And current researches of the Particle filter focus on the improvement of the computation of the filter.

And in this thesis we focus on the 'moment matching' filters and try some new algorithms to handle with the nonlinear situation and wish to improve the behavior of the filers in the three typical cases: bearing-only, range-only and bearing-range situations.

Chapter 2

Construction of the Analytic filter

2.1 Problem formulation

In this Chapter, we describe an unified approach to the construction of the 'Analytic Filters' for some typical class of problems. This unified approach presented here is elaborated from the approach proposed in paper[11]. And through examining it in some specific cases like: Bearing-Only,Range-Only and Bearing-Range Mixture, we find that theses cases are treated well in this construction.

The basic parts of the standard tracking problem can be considered as the the calculation between two discrete sequences of the n-vector x_t and the k-vector γ_t. where $t \in (1, 2, ...)$. The k-vector γ_t is the measurement information at time t. The n-vector x_t is the state description of the target in the generalized coordinates at time t. (the state x_t typically includes the position and velocity information of target in Cartesian coordinates).

And we wish to get the function in the form: $x_t = f(x_0, \gamma_1, \gamma_2 ... \gamma_t)$ to estimate the x_t (the states of the target) based on the information of the noisy measurements and the initial state (sometimes need the approximation or estimation of initial state).

The tracking problems can be considered essentially as the problems to calculate the conditional density $p(x_t|\gamma_{1:t})$ at the time t given the measurement $\gamma_{1:t}$ before the time t. As we know, the recursive calculation is preferred in the computer computation, so we have more interests in the recursive calculation algorithm.

We wish to express the conditional density $p(x_t|\gamma_{1:t})$ as a function of the the conditional density $p(x_{t-1}|\gamma_{1:t-1})$ and γ_t for $t = 1, 2....$

Actually, If two assumption are satisfied at each t, we can get a recursive solution through the Bayes' rule to the problem:

Assumption 1. γ_t is independent of $\gamma_{1:t}$ given $x_{1:t-1}$

Assumption 2. x_t is independent of $\gamma_{1:t-1}$ given x_{t-1}

From the Bayes' rule, we can get the recursive formula for the problem[13]:

$$p(x_t|\gamma_{1:t}) = 1/c \int p(x_t|x_{t-1})p(x_{t-1}|\gamma_{1:t-1})dx_{t-1} \times p(\gamma_t|x_t) \tag{2.1}$$

where c is a constant chosen to make sure that $\int p(x_t|\gamma_{1:t})dx_t = 1$ which is agreed with the definition.

Theoretically, the problem has already been solved, but sometimes, it is not possible to calculate the equations with two integrals involved in many cases. For this reason, more interests arise in the algorithms which can get computationally, robust and efficient estimation to the above formula.

And we are more interested in the recursive formula to calculate the mean $\hat{x}_t$ and the covariances P_t of the conditional density for each time t. And in further research, the conditional mean can be used to estimate the state (solve the tracking problem), the conditional covariance can be used to estimate the confidence regions of the estimation results.

2.2 Model of the noise: ‘Noise-before-nonlinearity’

There are two main themes in this filter, one is to approximate the densities of the state. It avoids the nonlinear part by the linear approximation to the nonlinear part of the equation and can approximate directly from the moment matching. For example, we can use some efficient and robust algorithms to estimate the equation(2.1). In this way, we can avoid some not feasible calculation including two integral calculation.

The other theme is to change the noise model to simplify the calculation of the probability. In the traditional method, the noise model is hard to handled for its nonlinear construction: $\gamma_t = \psi(d_t) + \omega_t$, where $\psi(d_t)$ is a nonlinear function of the relative state. In this theme, we wish to change the noise model to simplify the calculation. At the same time, we want to make the effect from the changes of the noise model on the relevant distribution as small as possible.

Modifications to the noise model can make sense in some special situations, where the measurements are angles (bearing) or Euclidean lengths (range) of the displacement d between the target and observer platform. It is acceptable because the two distribution are almost indistinguishable in this situation.

For such cases, the customary model is ‘noise-after-nonlinearity’ model, for the measurement γ_t, the traditional noise model is:$\gamma_t = \psi(d_t) + \omega_t$, where the d is the relative distance vector, $\psi()$ is the nonlinear function of the distance, like the equation(1.1)-(1.3). The ω_t is a Gaussian noise variable. And it is because that we assume that the noise is very small compared with the measurements. In linear situation, the ‘adding’ behavior is easy to handle with, but when faced with nonlinear situation it would be very complex.

In this thesis, we try the new model ‘noise-before-nonlinearity’ :$\gamma_t = \psi(d + \omega_t')$ to simplify calculation. This is because in this way, the new model can do the exact calculation of the corresponding distributions and make possible analytic moment calculations. And we can see the difference between two models is very small for the suitable d_t (large enough in range case) and σ (small enough in bearing case). Actually, we use the conditional densities of the measurement given the d_t and the densities of the error to compare two models. And if not suitable we can slightly change the parameter ω_t' to make the conditional densities of the two models almost indistinguishable.

2.3 Models of the state: motion description

2.3.1 Models for the continuous-system

The real models for the target should be continuous and we can make it into the discrete-time system. And the key point in this step is to transfer the noise process and the matrix description mathematically.

A common model for the continuous time model is the stochastic differential equation driven by white noise.

$$\begin{cases} \dot{x}(t) = f((t, x(t))) + b(t)w(t) \\ x(0) = x_0 \end{cases} \tag{2.2}$$

Where the $x(0)$ is the initial state, and $w(t)$ is the Gaussian white noise process which is independent from $x(0)$. ($w(t)$ is the noise such that $E(w(t)) = 0$,$R_w = Q\delta s$)

Actually, we focus on the linear case:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bw(t) \\ x(0) &= x_0 \end{aligned} \tag{2.3}$$

And we have

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-s)}Bw(s)ds$$

2.3.2 Models for the discrete-time system

We wish we could write the equation (2.3) into the discrete time form. And in this way, we can use the filter and do the computer matrix computation efficient.

So we wish to write the discrete-time model in the form of $x_k = Fx_{k-1} + w'_k$. x_k is the n-dimension state vector and x_k means the state x_t when $t = kT$, T is the sample time:

$$\begin{aligned} & x_k = Fx_{k-1} + w'_k \\ & x(0) = x_0 \end{aligned} \tag{2.4}$$

From the definition, we can compute the $F = e^{AT}$ and $w'_k = \int_{(k-1)T}^{kT} e^{A(kT-s)} Gw(s)ds$.

And w_k is independent Gaussian variable with zero mean and covariance:

$$\begin{aligned} & E[w_k] = 0 \\ & Cov[w_k] = \int_{(k-1)T}^{kT} e^{A(kT-s)} GQG^T e^{A^T(kT-s)} ds \\ & = \int_0^T e^{As'} GQG^T e^{A^T s'} ds' \end{aligned} \tag{2.5}$$

So we can write the continuous system into the discrete-time system:

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t & (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m \\ z_t = d_t + w_t & (w_t \sim N(0, Q^m)) \\ \gamma_t = \psi_(z_t) \end{cases} \tag{2.6}$$

The equation (2.6) describes the target motion and its nonlinear noisy measurements.

2.4 Algorithm of the Analytic Filter

In this section, we conclude a general class of the 'Analytic Filter' to approximate the first moment and the second moment of the $p(x_t|\gamma_{1:t})$ recursively. Now, we used the model introduced before:

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t \quad & (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m & \\ z_t = d_t + w_t & (w_t \sim N(0, Q^m)) \\ \gamma_t = \psi_(z_t) & \end{cases} \tag{2.7}$$

The equation (2.4) describes the target motion and its nonlinear noisy measurements.

For this system, we proposed a general moment matching analytic filter, the algorithm calculated the $(x_{t|t}, P_{t|t})$ to approximate the first and second moment of the conditional density: $p(x_t|\gamma_{1:t})$ in a recursive way. It calculate the $(x_{t|t}, P_{t|t})$ with the knowledge of the noisy measurement and $(x_{t-1|t-1}, P_{t-1|t-1})$.

This algorithm uses the Kalman Filter construction and makes a little change and extension. We divide the algorithm into two parts: prediction part and update part. (Notation: We write the $(x_{t|t}, P_{t|t})$ in the form of (x_t, P_t) for simplification and the $\hat{x}$ means the estimation of the state at the time t, the $\hat{x}_{t|t-1}$ means the estimation of the state x based on the information from the past time $1 : t-1$ (the information of the measurements $\gamma_{1:t-1}$), and the matrix $P_{t|t-1}$ means the covariance of the state with given measurements $\gamma_{1:t-1}$

Step 1. Prediction step:

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \end{cases} \tag{2.8}$$

Step 2. Correction step:

$$\begin{cases} S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^T S^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T \\ where: \\ \zeta_t = E[d|\psi(z) = \gamma_t], d \sim N(d_{t|t-1}, S_t) \\ \Gamma_t = Cov[d|\psi(z) = \gamma t], d \sim N(d_{t|t-1}, S_t) \end{cases} \tag{2.9}$$

In the update step, the Q_m is the covariance of the measurements $z_t = d_t + w_t, (w_t \sim N(0, Q^m))$. K_t is 'Kalman parameter used to correction the results. The ζ_t is the conditional mean of the d_t given the γ_t and Γ_t is the covariance of the d_t given the γ_t in this algorithm.

We firstly check the dimension of the matrix computation in 2-dimension. (it is similar to get the matrix dimension in three-dimension.):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

We can see this algorithm is very similar to the Kalman filter, and we change the measurements from the γ_t to the enhanced displacement. And the vector ζ is the conditional mean of the d_t given the new measurement. As we know that in this process, we do some approximation of the range information indirectly, and in this way this method contains too much information than reality has. So we use the equation: $K_t \Gamma_t K_t^T$ to compensate the uncertainty of the algorithm, which can be proved later.

So in this part, we propose a recursive algorithm to do the estimation of the states. And two typical characteristics are: the changes of the noise model and the approximation for the density. But the algorithm is not suitable to all cases, because the calculation of the Γ_t and ζ is very complex in some special cases (the initial parameters and requirements are not very suitable) where other algorithms maybe more suitable and efficient.

2.5 Analysis and Justification

In this part, we wish to do a theoretical analysis of the algorithm.

Firstly, consider the motion description in the equation (2.4)

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t \quad & (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m & \\ z_t = d_t + w_t & (w_t \sim N(0, Q^m)) \\ \gamma_t = \psi_(z_t) & \end{cases}$$

And the things we want to prove is that:
If we assume that the initial conditional density in the recursive process satisfies:

$$p(x_{t-1}|\gamma_{1:t-1}) = N(\hat{x}_{t-1}, P_{t-1})$$

Then we can get the results from the algorithm:

$$E[x_t|\gamma_{1:t}] = \hat{x}$$

$$cov(x_t|\gamma_{1:t}) = P_t$$

The concept stated above shows that the algorithm match the first and second moment of the true mean and covariance of the estimated state. This filter is based on the Kalman filter so it does a 'least mean squared error' estimation, which is same with other Kalman filters. But it is different with some common 'least mean squared error' filter like unscented Kalman filter or extended kalman filter. Extended kalman filter (EKF) uses the method of linearizing the nonlinear equations[4] and the unscented Kalman filter uses the 'statistical linearizion'[17].

And this algorithm uses a different noise model and also a different structure to approximate the conditional density. Then we would prove the statement: this algorithm can make the first and second moment the exactly matched with the true mean and covariance.

Proof: As we know the enhanced measurement in the algorithm is:

$$z_t = d_t + w_t, \quad (w_t \sim N(0, Q^m)).$$

Do the decomposition for the x_t from the equation (2.5) - (2.6):

$$x_t = \zeta_t + (I - K_t H)\hat{x}_{t|t-1} - K_t u^m + K_t z_t \tag{2.10}$$

Here, the ζ_t is independent with z_t , x_t, and ζ_t is the random variable satisfying:

$$\zeta_t \sim N(0, (I - K_t H)P_{t|t-1}) \tag{2.11}$$

The equation(2.7) shows that the x_t can be expressed as the sum of $\hat{x}_{t|t-1}$ and ζ_t. It means that the equation can be represented by the conditional mean of x_t resulting from the algorithm and an estimation error random variable ζ_t. All of these are under the assumption that the $p(x_{t-1}|\gamma_{1:t-1})$ is Gaussian distributed.

To check the moment matching, we calculate the expectation of the x_t given $\gamma_{1:t}$. And use the information that: the $z_t = \psi(\gamma_t)$ and ζ_t is zero mean process which is also independent with γ_t, z_t, so we can get:

$$\begin{aligned} E[x_t|\gamma_{1:t}] &= E[(\zeta_t + (I - K_t H)\hat{x}_{t|t-1} - K_t u^m + K_t z_t)|\gamma_{1:t}] \\ &= E[\zeta_t] + E[(I - K_t H)\hat{x}_{t|t-1}] - E[K_t u^m] + E[K_t z_t|\gamma_{1:t}] \\ &= (I - K_t H)\hat{x}_{t|t-1} - K_t u^m + K_t E[z_t|\gamma_{1:t}] \end{aligned} \tag{2.12}$$

As we know, from definition, the $E[z_t|\gamma_{1:t}]$ is the $E[z_{z\sim N(d_{t|t-1}, S_t)}|\gamma_t] = \zeta_t$, so we get:

$$E[z_t|\gamma_{1:t}] = E[z_{z\sim N(d_{t|t-1}, S_t)}|\gamma_t] = \zeta_t$$

And we have proved that:

$$E[\hat{x}_t|\gamma_{1:t}] = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1})$$

Then compute the covariance from the equation (2.7). In the same way, we use the information: the $z_t = \psi(\gamma_t)$ and ζ_t is zero mean process which is also independent with γ_t, z_t, so we can get:

$$\begin{aligned} Cov[x_t|\gamma_{1:t}] &= Cov[\zeta_t] + K_t Cov[z_t|\gamma_{1:t}]K_t^T \\ &= (I - K_t H)P_{t|t-1} + K_t Cov[z_t|\gamma_{1:t}]K_t^T \end{aligned} \tag{2.13}$$

As we know from the definition, the $Cov[z_t|\gamma_{1:t}]$ is the $Cov[z_{z\sim N(d_{t|t-1},S_t)}|\psi(z) = \gamma_t] = \Gamma_t$, so we get:

$$Cov[z_t|\gamma_{1:t}] = Cov[z_{z\sim N(d_{t|t-1},S_t)}|\psi(z) = \gamma_t] = \Gamma_t$$

And we have proved that

$$Cov(x_t|\gamma_{1:t}) = (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T$$

In conclusion, we have proved two key parts of the algorithm:

$$Cov(x_t|\gamma_{1:t}) = (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T$$

$$E[\hat{x}_t|\gamma_{1:t}] = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1})$$

So the algorithm generates the exact and moment matching conditional density with its mean and covariance in recursive way.

So if $p(x_{t-1}|\gamma_{1:t-1}) = N(\hat{x}_{t-1}, P_{t-1})$ then we can get $p(x_t|\gamma_{1:t}) = N(\hat{x}_t, P_t)$ from the algorithm, and with special interest we also have the formulation: $p(z_t|t_{1:t-1}) = N(d_{t|t-1}, S_t)$.

Chapter 3

Bearing-Only tracking problem

3.1 Background

In bearing-only tracking, measurements are taken from the angle of the relative displacement vector. As we know, the measurement of the angle information is defined as: $\theta = \angle d$, and the 'bearing vector' is defined as $\psi(d_t) = |d|^{-1}d$.
For two-dimensional situation, set the bearing vector $\psi(d_t)$ with azimuth θ_t:

$$\psi(d_t) = |d|^{-1}d = (cos(\theta), sin(\theta))$$

For three-dimensional situation, set the bearing vector $\psi(d_t)$ with azimuth and elevation θ_t, ϕ_t:

$$\psi(d_t) = |d|^{-1}d = (cos(\phi)cos(\theta), cos(\phi)sin(\theta), sin(\phi))$$

The aim of the bearing-only tracking problem is to estimate the position of the target only from the knowledge of the noisy bearing measurement (bearing vector which we set before only contains angle information between the observer platform and the target). Tracking problem with only bearing measurements is a special case, where the formulations are nonlinear and its measurements only offer bearing information which is not enough for the estimation in common sense. And we wish we could use these information to get the recursive Bayesian estimation of the state (the position of the target).

The difficulties in this estimation are derived from the missing information of the range information and its details can be seen in [14][15]. Another challenge we need to face is the nonlinear computation (like equation(1.2)) in the bearing-only tracking. And there is no computational equation of the conditional density of the state given the bearing measurements. So the common practical algorithms usually choose to use the method of the approximations.

In this section, we wish to use a new algorithm known as 'Shifted Rayleigh Filter'[8] based on the construction of the 'Analytical Filter'. This algorithm uses the conditional mean of the relative displacement to replace the nonlinear part in the computation. The approximation is also very efficient and transparent by the calculation of the conditional density of the range information. Compared with other moment-matching algorithms, the SRF (Shifted Rayleigh filter) can do an accurate calculation to correct the conditional density of the states, when the sensors measure new noisy bearing information. It is different from the extended Kalman filter and its evolutions which approximate the prior conditional density by a Gaussian process.

The Shifted Rayleigh algorithm can be described by three steps: first do a normal approximation of the conditional density at the time $t-1$ and then calculate an exact conditional density distribution considering the measurement. In the last step, we use a normal distribution to approximate this conditional density by moment matching method. Then it would be prepared for the next iteration at time t.

The algorithm uses the 'Analytic filter' to save the computational cost, and we can see that the only approximation in this algorithm is the matching process at the each step at time t. So the accuracy can be confirmed and the performance can be improved in some special cases.

Also the shifted motion of the observer can provide more information of the range in some sense. Actually, in the 'benign bearing-only tracking scenarios' (easy scenario where the range information is easy to get), some common filter can all do a good work. For example: the extended Kalman filter and pseudomeasurement filter(we will check the simulations of these two filters in the later section).

Also unscented Kalman filter (derived from the Kalman filter) and shifted Rayleigh filter (we will introduce later) can make sense in this special scenario where the measurements can get enough 'range information'[7][10][12]. But in some challenging cases when noise's covariance is very large or the clutter levels are also high, then common filter cannot do a good job in this situaion, and the 'Shifted Rayleigh filter' can do a better work for its high accuracy and robust performance.

3.2 Formulation of the Bearing-Only tracking problem

The analytic system model ('noise before nonlinearity'):

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t \qquad (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m \\ z_t = d_t + w_t \qquad\qquad\quad (w_t \sim N(0, Q^m)) \\ \gamma_{\theta t} = \psi_1(z_t) \end{cases} \tag{3.1}$$

where the

$$\psi_1(z_t) = |z|^{-1}z = (cos(\theta), sin(\theta))$$

The state contains the velocity and the displacement in different dimensions, for example in two dimensions:

$$x = \begin{bmatrix} x \\ v_x \\ y \\ v_y \end{bmatrix}$$

Matrix F is a $n \times n$ matrix to shows the motion of the state, u^s is an additional signal of the motion, which can model some complex motion like the varying acceleration motion. And v_t is the system noise used to describe the inaccurate part in the motion description.

d_t is the relative distance vector between the target and the observer platform, in the two dimension situation:

$$d_x = x_t - x_{ship}$$

$$d_y = y_t - y_{ship}$$

and the u_m is the motion of the observer platform with the negative sign.

The z_t is the noisy displacement, and here we use the new model of the noise (noise before nonlinearity) introduced before. We add the noise onto the relative displacement. The $\psi(z_t)$ is the noisy measurements of the displacement, and it is a nonlinear function.

For the most of the cases we can set the F and H as:

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

A precise model requires the matrix computation to work well in the formulation we concluded above, so we firstly check the dimension of the matrix computation in 2-dimension. (it is similar to get the matrix dimension in three-dimension):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

Of particular interest are some scenarios when the 'measurement' noise take the form below and it was proved well in the paper[8]:

$$Q_t^m = \sigma^2 E[|d_t|^2 | b_{1:t-1}] I_{k \times k} + Q^{tr}$$

3.3 The Noisy Bearing Model

As the special case of the analytic filter, the noise model of the Shifted Rayleigh filter, is also 'noise before nonlinearity'. The bearing vector b is modeled as the measurement nonlinear function of the noisy displacement $d_t + w_t$. We define this displacement vector $d_t + w_t$ as the 'augmented measurement' vector. And compute the bearing vector from the augmented measurement. We project the augmented vector $d_t + w_t$ onto the unit circle (when the dimension is 2-dimension) or the unit sphere (when the dimension is 3-dimension) and then we can get the bearing vector (equation 3.2):

$$\begin{cases} z_t = d_t + w_t \\ \gamma_t = \Pi z_t \end{cases} \tag{3.2}$$

As we introduced in the Section 3.2, the w is a Normal random variable with co-variance $Q_t^m = \sigma^2 E[|d_t|^2 | b_{1:t-1}] I_{k\times k} + Q^{tr}$, and the w is also independent of the displacement d and state x. In this section, we will discuss the evolution of the bearing model.

In some sense, it is clear that the dominant noise is from the translation noise of the target or observer (the noise from the change between the bearing vector with the angle information and its perturbation). And it is characterized by Q^{tr}, a symmetric matrix which is used to model the noisy perturbation of the calculation. In this filter, we prefer the model of the equation (3.2), and to simplify we set the dimension into the 2-dimension, and $Q^{tr} = 0_{2\times 2}$ for the fair comparison.

The standard model for the bearing noise use the adding noise on the measurements when the noise is very small, and the traditional model is expressed as:

$$\theta = \angle d + n = \arctan(d_y/d_x) + n \tag{3.3}$$

where n is the measurement noise and $n \sim N(0, \sigma^2)$ which is also independent with the state x_t and displacement vector d_t.

The displacement d is used to show the relative position between the observer and the target. Set the position vector of the target is (x_{target}, y_{target}), the position vector of the observer is $(x_{observer}, y_{observer})$, so the displacement is defined in the Cartesian coordinates of the target and the observer in this thesis in the form of:

$$\begin{aligned} d_x &= x_{target} - x_{observer} \\ d_y &= y_{target} - y_{observer} \end{aligned} \tag{3.4}$$

The figure below shows how the relative distance and the bearing information:

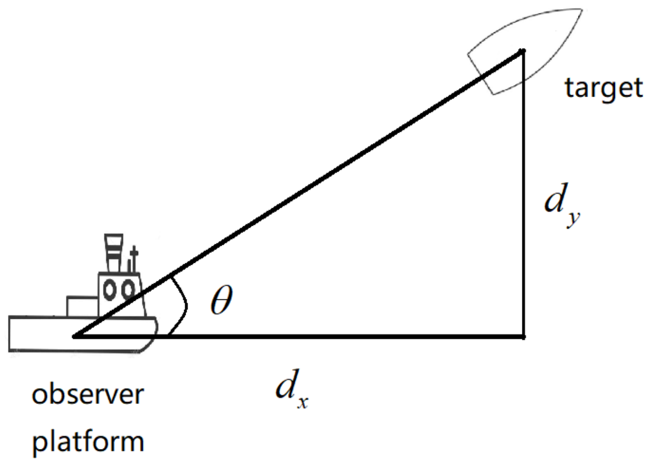

Figure 3.1: The figure of the displacement and bearing measurement in 2-D

To get a recursive and precise formulation, we prefer the matrix calculation and wish to use the equation (3.1) to describe the equation (3.4) then the easiest way is to set some special H and u^m as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

And then set the u^m as the opposite number of the motion function of the observer. For example, the observer moved in constant process or as a function of time $x_t^p = x_0^p + tc$, $y_t^p = y_0^p + tb$. So the displacement can be expressed as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$u^m = \begin{bmatrix} -(x_0^p + tc) \\ -(y_0^p + tb) \end{bmatrix}$$

Then the relative displacement can be expressed as

$$d_t = Hx + u^m$$

After gaining the relative displacement, we can set the noise model in Shifted Rayleigh filter, and to simplify, we wish to wire the equation (3.5) to replace the equation (3.2):

$$\begin{cases} z'_t = d_t + |d_t|e' \\ \gamma_t = \Pi z'_t \end{cases} \tag{3.5}$$

Where e' is a Gaussian random variable which satisfy that $e' \sim N(0, \sigma^2 I_{2\times 2})$. Actually, the equation (3.5) is a easier form of the equation (3.2) and it use the $|d_t|e'$ to construct the augmented noisy measurements. Actually there is no difference between them and we will use the equation(3.5) to compare with the standard form equation (3.3). Firstly, we need to justify that there is no difference between the equation (3.2) and (3.5) in the first and second moment sense:

$$\begin{aligned} &E[|d|e'd^T] = E[E[(|d|e'd^T)|d]] = 0 \times I_{2\times 2} = 0 \\ &E[|d|e'] = 0 = E[w] \\ &Cov[|d|e'] = E[|d|^2 e'e'^T] = E[|d|^2]\sigma^2 I_{2\times 2} = Cov[w] \end{aligned}$$

So we can conclude that $|d|e'$ is independent with d, which is the same with w. And also w and $|d|e'$ both have zero mean and have same covariance: $E[|d|^2]\sigma^2 I_{2\times 2}$. So we use the $|d_t|e'_t$ to replace the w_t.

Then we would check the difference between two models (the 'noise before nonlinearity ' model equation 3.5 and standard 'noise after nonlinearity 'equation 3.3): In the traditional model like equation (3.3):

$$\theta = \angle d + n = \arctan(d_y/d_x) + n \tag{3.6}$$

In the 'noise before nonlinearity' model:

To compare two models, We wish we could get the expression of the θ' from the augmented vector b'. And We can know from the paper[7] the relationship between the θ' and the true bearing information $\arctan(d_y/d_x)$ in the new model can be written as:

$$\theta' = \arctan(d_y/d_x) + n' \tag{3.7}$$

Where the n' in the new noise model is a zero mean random variable which is restricted to $[-\pi, \pi]$ and independent with the displacement vector d. The density of the n' is expressed below:

$$\alpha_\sigma(\theta) = \frac{e^{-\frac{1}{2\sigma^2}}}{2\pi} \times (1 + \sqrt{2\pi}\frac{cos\theta}{\sigma} F_{normal}(\frac{cos\theta}{\sigma})e^{1/2(cos\theta/\sigma)^2}) \tag{3.8}$$

The equation (3.8) includes the F_{normal}, which is the cumulative distribution function. And the F_{normal} can be computed quickly by the matlab, here F_{normal} is used for a normal random variable $x \sim N(0, 1)$. And we can see that the difference between the two models equation (3.6) and equation (3.7) is the new noise n' whose density is (equation 3.8) and n whose density is $N(0, \sigma^2)$.

To compare the models, we choose different σ value to plot the figure of the densities and see that for most small σ (which is less than the radians value), the difference between two densities of the different model seems indistinguishable, the difference actually is less than 0.03.

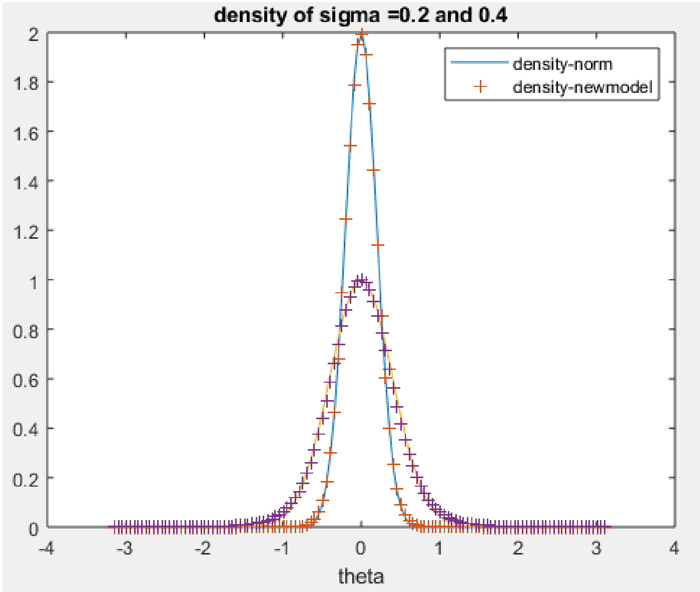

Figure 3.2: The density of two noise models in bearing-only tracking with sigma = 0.2 and 0.4

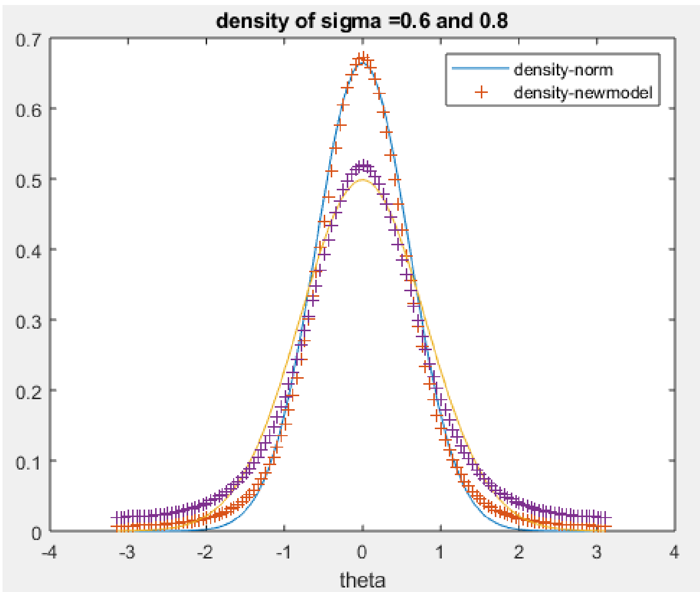

Figure 3.3: The density of two noise models in bearing-only tracking with sigma = 0.6 and 0.8

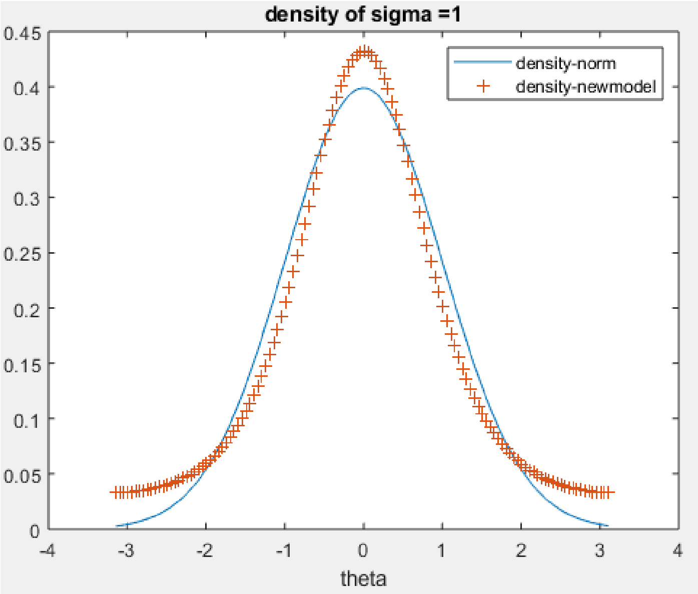

Figure 3.4: The density of two noise models in bearing-only tracking with sigma = 1

From the figure (3.2) - figure (3.4) below, we can conclude that the 'noise before nonlinearity' model is very suitable and indistinguishable when the σ is not very large in the bearing-only case. And even when the σ is relatively much bigger like $\sigma = 1$ (which is rarely seen in the some tough reality), it also works well.

So the model of the noise used in the Shifted Rayleigh algorithm can be seen indistinguishable with the normal standard noise model. The method used to construct the bearing vector by augmented measurement and the noise $|d|e'$ is a good choice.

And it provide the possibility to do the accurate computation of the conditional density with the 'Analytic Filter' algorithm construction.

3.4 The 'Analytic filter': Shifted Rayleigh Filter

After checking that the noisy model is indistinguishable with the traditional noise model, we now introduce the Shifted Rayleigh Filter which is a special form of the 'Analytic filter' in the bearing-only situation.

We wish to get the recursive algorithm to generate the sequence:$x_{t|t}$ and $P_{t|t}$ with the measurement sequence γ_t.

Algorithm outline

Step 1. Initial configuration:

$x_{0|0}$ Estimation for the initial state

$P_{0|0}$ The covariance matrix of the initial state (The confidence of the initial state)

Step 2. Prediction step:

Calculate the $\hat{x}_{t|t-1}$ and $P_{t|t-1}$ and also construct the augmented bearing vector b_t

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \\ b_t = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \end{cases} \tag{3.9}$$

Step 3. Updating step:

$$\begin{cases} S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^TS^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_tH)P_{t|t-1} + K_t\Gamma_tK_t^T \\ where: \\ \zeta_t = (b_t^TS_t^{-1}b_t)^{-1/2}\rho_r(z_t)b_t \\ \Gamma_t = \delta_t^r b_t b_t^T \end{cases} \tag{3.10}$$

Here we introduce the function z_t, $\rho(z)$, δ_t and $F_{normal}(z)$ to compute ζ_t and Γ_t and set the r is the dimension of the filter (r=2 or 3):

$$z_t = (b_t^T S_t^{-1} b_t)^{-1/2} b_t^T S_t^{-1} (H\hat{x}_{t|t-1} + u_t^m) \tag{3.11}$$

$$\rho_r(z) = \frac{\int_0^\infty s^r e^{-1/2(s-z)^2} ds}{\int_0^\infty s^{r-1} e^{-1/2(s-z)^2} ds} \tag{3.12}$$

$$\delta_t(z) = \begin{cases} (b_t^T S_t^{-1} b_t)^{-1}(2 + z_t\rho_2(z_t) - \rho_2^2(z_t)) & r = 2 \\ (b_t^T S_t^{-1} b_t)^{-1}(3 + z_t\rho_3(z_t) - \rho_3^2(z_t)) & r = 3 \end{cases} \tag{3.13}$$

Actually, the ρ is the mean of 'Shifted Rayleigh parameter' and in the two dimension and three dimension situation the $\rho_r(z)$ has different form with a easy calculation process we can get the formulation from the paper[7], and F_{normal} is the cumulative distribution function for the random variable $N(0,1)$. Actually, this function can be computed by the function library from matlab. We can use the $1/2\text{erfc}(-z/\sqrt{2})$, where: erfc($x$)= $2/\sqrt{\pi}\int_x^\infty e^{-t^2} dt$.

$$\rho_r(z) = \begin{cases} \frac{ze^{-z^2/2} + \sqrt{2\pi}(z^2+1)F_{normal}(z)}{e^{-z^2/2} + \sqrt{2\pi}(z)F_{normal}(z)} & r = 2 \\ z + \frac{2}{\rho_2(z)} & r = 3 \end{cases} \tag{3.14}$$

So we can get the ζ_t and Γ_t from the equation (3.11-3.14):

$$\begin{aligned} \zeta_t &= (b_t^T S_t^{-1} b_t)^{-1/2} \rho_r(z_t) b_t \\ \Gamma_t &= \delta_t^r b_t b_t^T \end{aligned} \tag{3.15}$$

So we can complete to do the update process:

$$\begin{aligned} \hat{x}_t &= x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t &= (I - K_t H) P_{t|t-1} + K_t \Gamma_t K_t^T \end{aligned} \tag{3.16}$$

In the update step, the Q_m is the covariance of the measurements. K_t is 'Kalman parameter used to correction the results. The ζ_t is the conditional mean of the d given the γ_t and Γ_t is the conditional covariance of the d given the γ_t.

The proof of the algorithm can be checked from the Chapter 2.4 - Chapter 2.5, and we also will justify the calculation of the ζ_t and Γ_t for the bearing-only case in the next section: 3.5 'Analysis'.

After giving all the parameter we need to compute in the algorithm, we need to check the dimension of the matrix computation again in this bearing-only scenario. Take the two-dimension as an example (it is similar to get the matrix dimension in three-dimension):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

Where the K_t is the Kalman gain, the S_t is the covariance matrix of d_t and ζ_t is the conditional mean of the d_t given the bearing vector b_t, Γ is the conditional variance of d_t given the bearing vector b_t.

3.5 Analysis and Justification

Firstly, we would like to prove that the structure of the SRF we introduced in the equation (3.9-3.10) makes sense:

To prove the mean equation in the updating step:
We need to introduce the augmented vector(displacement). We can assume that the measurement method can directly measure the augmented vector: d_t, and then we can use the standard Kalman filter to get the formulation:

$$\begin{aligned} &\hat{x}_t = x_{t|t-1} + K_t(d_t - d_{t|t-1}) = (I - K_t H)x_{t|t-1} - K_t u_t^m + K_t d_t \\ &P_t = (I - K_t H)P_{t|t-1} \end{aligned} \tag{3.17}$$

From the updating equation in bearing-only case, we can see that when we assume that the bearing vector is Gaussian random variable, then we can just use ζ_t to replace the augmented relative displacement vector d_t, actually the ζ is the conditional mean of d_t given the bearing b_t. (which is the consistent with the equation proof given in the Chapter 2)

And then prove the covariance equation P_t in the updating step:

As we know, the bearing information is not enough for the augmented vector d_t so we need to compensate the covariance use the term $K_t\Gamma_t K_t^T$. From the equation (3.15) we know that the $\Gamma_t = \delta_t b_t b_t^T$. The adding term is the $Cov[K_t y_t|b_t]$ and the δ_t is defined as the conditional covariance: $Cov[|y_t||b_t]$, then we can prove the:

$$Cov[K_t y_t|b_t] = Cov[|y_t|K_t b_t|b_t] = \delta_t K_t b_t b_t^T K_t^T \tag{3.18}$$

The equation(3.18) is consistent with the analysis in the Chapter 2.

And the things left which we need to prove are $\zeta_t = (b_t^T S_t^{-1} b_t)^{-1/2}\rho_r(z_t)b_t$ and $\Gamma = \delta_t b_t b_t^T$:

We have proved the structure of the Shifted Rayleigh Filter is right and then we need to prove that:

$$\begin{aligned}\zeta_t &= (b_t^T S_t^{-1} b_t)^{-1/2} \rho_r(z_t) b_t \\ \Gamma &= \delta_t b_t b_t^T\end{aligned} \tag{3.19}$$

Where the S_t, b_t, ρ_t, z_t, δ_t are given in the equation (3.10-3.14).

For the justification, we need to introduce a Lemma (equation 3.20) from the paper [8]:

The x is a r-dimensional random variable (usually r=2 or 3), $x \sim N(m, P)$.

Define $s = |P^{-1/2}x|$ (the length of the $P^{-1/2}x$,) and $\theta = |P^{-1/2}x|^{-1} P^{-1/2}x$ (the projection of the $P^{-1/2}x$) and we define the $P^{-1/2}x$ is the normalized version of the state. Then we can know the 'the shifted Rayleigh density' $p(s|\theta)$:

$$\begin{aligned}p(s|\theta) &= \frac{s^{r-1} e^{-1/2(s-z)^2}}{\int_0^\infty \delta^{r-1} e^{-1/2(\delta-z)^2} d\delta} \\ z &= m^T P^{-1/2} \theta\end{aligned} \tag{3.20}$$

And what is interesting: $E[s^n] = (n + r - 2)E[s^{n-2}] + zE[s^{n-1}]$, this equation and the equation (3.20) can also be seen in the paper [7].

As we know the relative displacement $d_t \sim N(H\hat{x}_{t|t-1} + u^m, S_t)$, then apply the Lemma cited before: $s_t = |S_t^{-1/2} d_t|$ and $\theta_t = |S_t^{-1/2} d_t|^{-1} S_t^{-1/2} d_t$. Actually we can just write the length and projection into the form of $V_t^{-1/2} d_t = s_t \theta_t$. Notice that the condition on the θ_t is same to the condition on the b_t:

$$\begin{aligned}\theta_t &= (b_t^T V_t^{-1} b_t)^{-1/2} V_t^{-1/2} b_t \\ b_t &= (\theta_t^T V_t^{-1} \theta_t)^{-1/2} V_t^{-1/2} \theta_t\end{aligned} \tag{3.21}$$

We decompose the x_t as

$$x_t = \xi_t + (I - K_t H)x_{t|t-1} + K_t d_t - K_t u_t^m \tag{3.22}$$

where $\xi \sim N(0, (I - K_t H)P_{t|t-1})$

Then we can get the equation (3.23) from the relationship :$V_t^{-1/2} d_t = s_t \theta_t$, we just replace the d_t in the equation (3.22)

$$x_t = \xi_t + (I - K_t H)x_{t|t-1} - K_t u_t^m + s_t K_t V_t^{1/2} \theta_t \tag{3.23}$$

Then we compute the $E[x_t|b_t]$:

$$
\begin{aligned}
&E[x_t|b_t] = E[\xi_t] + (I - K_t H)x_{t|t-1} - K_t u_t^m + E[s_t|b_t]K_t V_t^{1/2}\theta_t \\
&= (I - K_t H)x_{t|t-1} - K_t u_t^m + E[s_t|\theta_t]K_t V_t^{1/2}\theta_t
\end{aligned}
\tag{3.24}
$$

Recall the density of the 'the shifted Rayleigh' we cited in the equation (3.20), then we can compute the $E[s_t|\theta_t] = \rho_r(z_t)$, wherer $\rho_r(z) = \frac{\int_0^\infty s^r e^{-1/2(s-z)^2} ds}{\int_0^\infty s^{r-1} e^{-1/2(s-z)^2} ds}$, which is consistent with the algorithm. And by the equation (3.20) we can compute the $z_t = (b_t^T S_t^{-1} b_t)^{-1/2} b_t^T S_t^{-1}(H\hat{x}_{t|t-1} + u_t^m)$ which is also consistent with the algorithm.

And then we can compute the $E[x_t|b_t]$ from the equations above:

$$
\begin{aligned}
&E[x_t|b_t] = (I - K_t H)\hat{x}_{t|t-1} - K_t u_t^m + K_t(b_t^T S_t^{-1} b_t)^{-1/2}\rho_r(z_t)b_t \\
&E[x_t|b_t] = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\
&\zeta_t = (b_t^T S_t^{-1} b_t)^{-1/2}\rho_r(z_t)b_t
\end{aligned}
\tag{3.25}
$$

In addition, from the equation (3.21 -3.23) we can also compute the $Cov[x_t|b_t]$:

$$
\begin{aligned}
&Cov[x_t|b_t] = Cov[\xi_t] + Cov[s_t|\theta_t]K_t V_t^{1/2}\theta_t(K_t V_t^{1/2}\theta_t)^T \\
&= (I - K_t H)P_{t|t-1} + Cov[s_t|\theta_t]K_t V_t^{1/2}\theta_t\theta_t^T V_t^{1/2} K_t^T
\end{aligned}
\tag{3.26}
$$

And from the Lemma we cited, we know the $Cov[s_t|\theta_t]$ from the 'the shifted Rayleigh density'

$$
Cov[s_t|\theta_t] = \begin{cases} 2 + z_t\rho_2(z_t) - (\rho_2(z_t))^2 & r = 2 \\ 3 + z_t\rho_3(z_t) - (\rho_3(z_t))^2 & r = 2 \end{cases}
\tag{3.27}
$$

So we can compute $Cov[x_t|b_t]$:

$$
\begin{aligned}
&Cov[x_t|b_t] = (I - K_t H)P_{t|t-1} + \delta_t^r K_t b_t b_t^T K_t^T \\
&= (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T \\
&where: \\
&\Gamma_t = \delta_t^r b_t b_t^T \\
&\delta_t^r = \begin{cases} (b_t^T S_t^{-1} b_t)^{-1}(2 + z_t\rho_2(z_t) - \rho_2^2(z_t)) & r = 2 \\ (b_t^T S_t^{-1} b_t)^{-1}(3 + z_t\rho_3(z_t) - \rho_3^2(z_t)) & r = 3 \end{cases}
\end{aligned}
\tag{3.28}
$$

So we have proved the computation of the ζ_t,Γ_t and the structure of the filter. All the equation has been confirmed and proved. We will do the simulation in the next section to check whether this filter make sense and useful.

3.6 Simulation and Conclusion

As the problem formulation and its motion description, we have the state description equation:

$$x_t = Fx_{t-1} + u^s + v_t \quad (v_t \sim N(0, Q^s))$$

We use the x_t which contains the position information and velocity information in two-dimension:

$$x_t = \begin{bmatrix} x \\ V_x \\ y \\ V_y \end{bmatrix}$$

We choose the scenario where the target moves in one direction and the ship is shifted and the whole motion is constant velocity motion.

So the $u^s = 0$ and F is in the form below:

$$x_t = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{t-1} + \begin{bmatrix} T^2/2 \\ T \\ T^2/2 \\ T \end{bmatrix} v_{t-1}$$

We just let the target move along the x-direction and $V_y = 0$. Then we can set different initial value of the V_x and x_0, y_0 to do the simulation. (In this simulation, the $V_x = 1$, $x_0 = 40$, $V_y = 0$, $y_0 = 0$) And v_t is a Gaussian noise $N(0, 0.01)$, the sample time of the discrete-time system is $T = 1s$. So we set:

$$x_t = \begin{bmatrix} x \\ V_x \\ y \\ V_y \end{bmatrix} = \begin{bmatrix} x \\ V_x \\ y \\ 0 \end{bmatrix}$$

Then we organize the motion of the observer platform, we design that the observer ship moved in the x-direction with velocity 4m/s, y-direction with 0m/s velocity.

And also the ship shifted in both x and y direction. So the observer moved in the way:

$$x_{observer} = \begin{bmatrix} x_{observer} \\ y_{observer} \end{bmatrix} = \begin{bmatrix} 4t + w_1 \\ 20 + w_2 \end{bmatrix}$$

w_1 and w_2 are both Gaussian noise with $N(0, 1)$.

For the fair comparison, we add the noise after nonlinearity. So we make the measurements in the form of:

$$\gamma_t = \arctan(\tfrac{y_{target} - y_{observer}}{x_{target} - x_{observer}}) + w_t$$
$$w_t \sim N(0, \sigma^2)$$
$$\sigma = 0.05$$

Approximate the initial value of the state and its covariance to prepare to run the algorithm (we set an error between the approximation and the real position to check whether the algorithm is sensitive to the initial value):

$$x_{0|0} = \begin{bmatrix} 5 \\ 1 \\ 10 \\ 0 \end{bmatrix}, P_{0|0} = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

All the preparations are completed for the filter:

$$u_t^s = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, u_t^m = \begin{bmatrix} -4t \\ -20 \end{bmatrix}$$

$$Q^{tr} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, Q^s = 0.01 \begin{bmatrix} 0.25 & 0.5 & 0 & 0 \\ 0.5 & 0.1 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}$$

$$Q_t^m = Q^{tr} + \sigma^2(|H\hat{x}_{t|t-1} + u_t^m|^2 + tr(HP_{t|t-1}H^T))I_{2\times 2}$$

Then we run the algorithm:

Prediction step:

Calculate the $\hat{x}_{t|t-1}$, $P_{t|t-1}$ and the augmented bearing vector b_t

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \\ b_t = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \end{cases}$$

Updating step:

Compute the recursive equation below and plot the figure

$$\begin{cases} S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^TS^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_tH)P_{t|t-1} + K_t\Gamma_tK_t^T \\ where: \\ \zeta_t = (b_t^TS_t^{-1}b_t)^{-1/2}\rho_r(z_t)b_t \\ \Gamma_t = \delta_t^rb_tb_t^T \end{cases}$$

Simulation result:

We choose the performance of the pseudo-measurement filter[20] to compare with the performance of Shifted Rayleigh filter. We do simulation under the same situation and plot figures of the tracking performance and error of two filters:

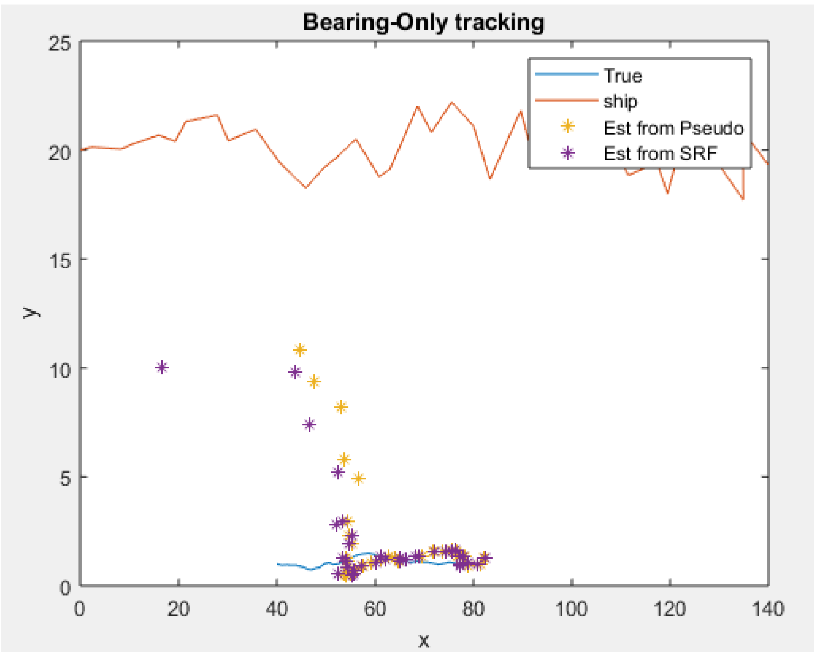

Figure 3.5: The tracking of the SRF and Pseudo-measurement in the bearing-only case

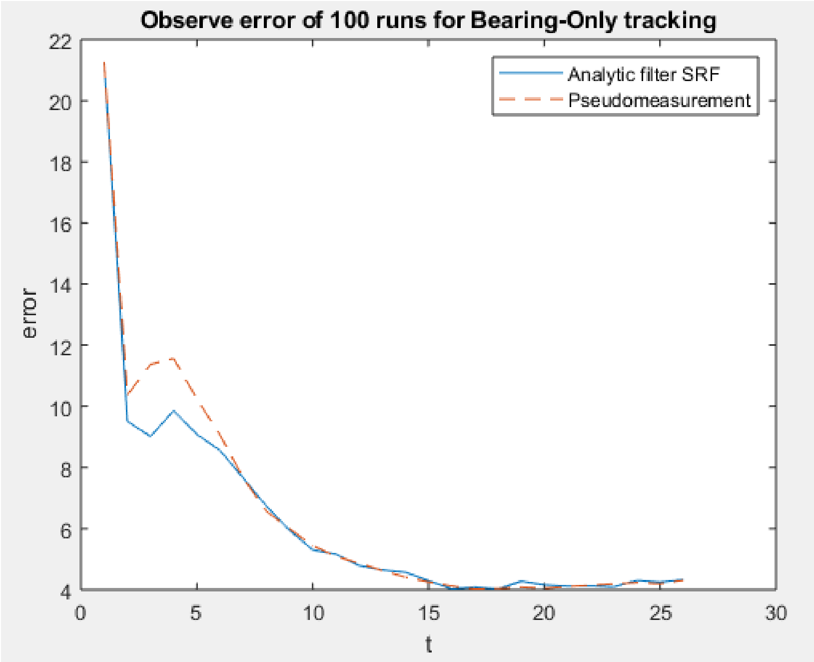

Figure 3.6: The error of the SRF and Pseudo-measurement in the bearing-only case

Then we also choose the EKF to compare with the SRF, and we plot the figures of the tracking performance and the error of the tracking below:

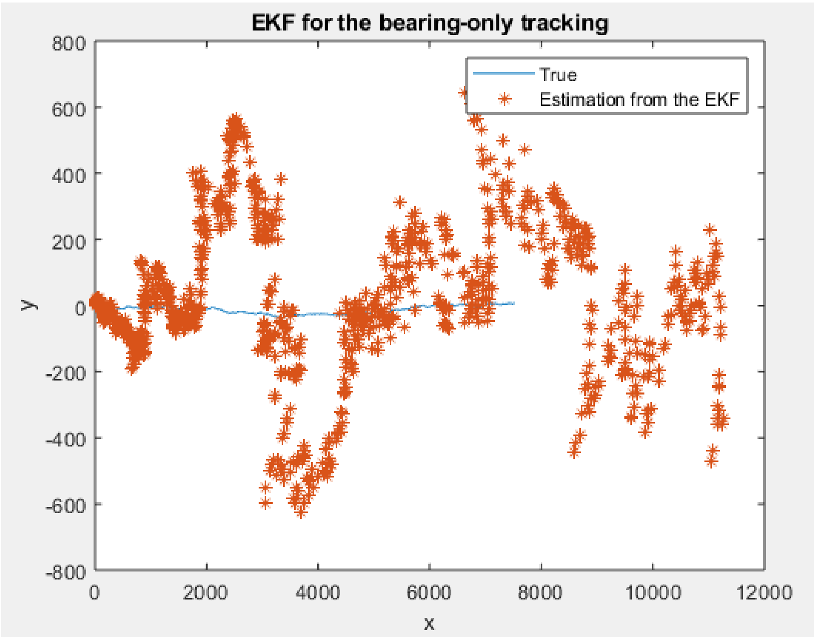

Figure 3.7: The tracking of EKF in the bearing-only case

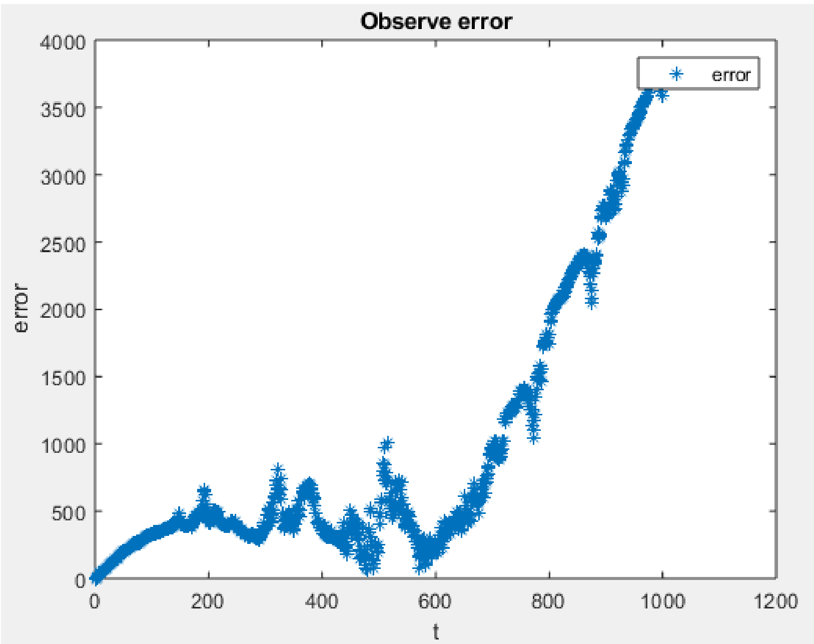

Figure 3.8: The error of EKF in the bearing-only case

Figure 3.5 shows that two filter (SRF and Pseudomeasurement filter)both can track the target quickly (in 5 steps). Figure 3.6 shows the errors of filters all can converge, and the error could be very small while the steps increasing. The SRF and pseudo-measurement filter can work well in this scenario. They can meet all the requirements we want (convergence and fast speed). SRF even can work better.

For the Extended Kalman filter (Figure 3.7-3.8), we can see that sometimes it is not robust at all and it could even lose the target in the bearing-only case

Chapter 4

Range-only tracking Problem

4.1 Background

In range-only tracking, measurements are taken from the euclidean distance of the relative displacement vector. As we know, the measurement of the range information is defined as: $r_t = |d|$.

$$\psi(d_t) = |d| = \sqrt{(d_1^2 + d_2^2 + ... + d_r^2)} \tag{4.1}$$

The range-only tracking problem is the task to estimate the position of the target only from the knowledge of the noisy range measurement (range measurements r_t which we set in equation (4.1) only contains distance information between the observer platform and the target). Tracking problem only with range measurements is very difficult because of its limited measurement information and its nonlinearity.

The first challenge is that the measurements only offer limited information of the target: range information. And the range information is not enough for the estimation of the position in common sense. And we wish we could use these limited information to get the recursive and converge Bayesian estimation of the state (the position of the target). The difficulty from missing information of the bearing information and its details can be seen in [19].

Another difficulty in this estimation is derived from the its nonlinear computation (like equation(4.1)). And there is no direct and practical equation to compute conditional density for states with the range information. So the common practical algorithms usually choose to use the method of the approximations. The extended Kalman filter is a typical algorithm to handle with this nonlinear estimation problem, it approximates the conditional density by a Gaussian process[4]. And the Particle filter is another typical filter which uses a empirical distribution to approximate the conditional density[3].

In this section, we wish to use the 'Analytic filter' introduced in Chapter 2. The aim of this filter is to improve the performance of the filter in the range-only case and control the computation cost in some thresholds. This algorithm uses the conditional mean of the relative displacement to replace the nonlinear part in the computation with Kalman structure. This method has been proved and its details are in the Chapter 2. The approximation is very efficient and transparent by the calculation of the conditional density of the bearing information. And combine the range information we measured with the bearing information we computed, we can get the relative displacement (augmented vector) for the updating of estimation.

The algorithm can be described by three steps: first do a normal approximation of the conditional density at the time $t-1$ and then calculate an exact conditional density considering the measurement to get the augmented vector. In the last step, we use a normal distribution to approximate this conditional density by moment matching method. Then it would be prepared for the next iteration at time t.

The algorithm can save the computational cost for its matrix and trigonometric computation. We can also improve the performance for its direct computation of the conditional density in some scenario. Actually, the only approximation in this algorithm is the matching process at the each step. So the accuracy can be confirmed and the performance can be improved in many cases.

4.2 Formulation of the Range-Only tracking problem

The system for the range-only tracking problem ('noise before nonlinearity'):

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t & (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m & \\ z_t = d_t + w_t & (w_t \sim N(0, Q^m)) \\ \gamma_{rt} = \psi_2(z_t) & \end{cases} \tag{4.2}$$

where the nonlinear function

$$\psi_2(z_t) = |z| = \sqrt{(z_1^2 + z_2^2 + ...z_r^2)}$$

The state contains the velocity and the displacement in different dimensions, for example in two dimensions:

$$x = \begin{bmatrix} x \\ v_x \\ y \\ v_y \end{bmatrix}$$

Matrix F is a $n \times n$ matrix to shows the motion of the state, u^s is an additional signal of the motion, which can describe some complex motion like the varying acceleration motion. And v_t is the system noise used to describe the inaccurate part in the system description.

d_t is the relative distance vector between the target and the observer platform, in the two dimension situation:

$$d_x = x_t - x_{ship}$$

$$d_y = y_t - y_{ship}$$

and the u_m is the motion of the observer platform with the negative sign. Here we use the $d_t = Hx_t + u^m$ to compute the relative displacement.

The z_t is the noisy displacement, and we prefer the new model of the noise (noise before nonlinearity) which is introduced before. We add the noise onto the relative displacement. The $\psi_2(z_t)$ is the noisy measurements of the range information, and it is a nonlinear function of the displacement. For the most of the cases we can set the matrix F and matrix H as the form:

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The matrix computation need to match the dimension of each matrix, so we firstly check the dimensions of the matrix computation. And we take the 2-dimension situation as example. (It is similar to get the matrix dimension in three-dimension):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

Q^s is the covariance matrix of the noise v_t, the Q^m is the covariance matrix of the noise w_t.

4.3 The Noisy Range Model

The noise model of this analytic filter for the range-only case is also 'noise before nonlinearity'. The range measurement result r_t is modeled as the nonlinear function of the noisy displacement $d_t + w_t$. We define this displacement vector $d_t + w_t$ as the 'augmented measurement' vector. We model the range information as the result of the computation by the nonlinear function of the augmented vector $\psi_2(d_t + w_t)$:

$$\begin{cases} z_t = d_t + w_t \\ \gamma_{rt} = \psi_2(z_t) \end{cases} \tag{4.3}$$

The w_t is an independent normal random variable with covariance $Q_t^m = \sigma^2 I_{r\times r}$, r is the dimension of the target. The w_t is independent of the displacement d and state x. In this section, we will discuss the evolution of the 'nose before nonlinearity' model for the range situation and compare it with the traditional noise model.

For comparison, we also write the equations of the traditional noise model which adds noise on the measurements when the noise is very small, and the traditional model is expressed as:

$$\gamma'_{rt} = |d_t| + w'_t \quad , \quad w'_t \sim N(0, \sigma^2) \tag{4.4}$$

where w'_t is the measurement noise and $w'_t \sim N(0, \sigma^2)$ which is also independent with the state x_t and displacement vector d_t.

The displacement d_t is used to show the relative position between the observer and the target. Set the position vector of the target is (x_{target}, y_{target}), the position vector of the observer is $(x_{observer}, y_{observer})$, so the displacement is defined in the Cartesian coordinates of the target and the observer in this thesis in the form of:

$$d_x = x_{target} - x_{observer}$$
$$d_y = y_{target} - y_{observer}$$

The figure below shows how the relative distance and the range information:

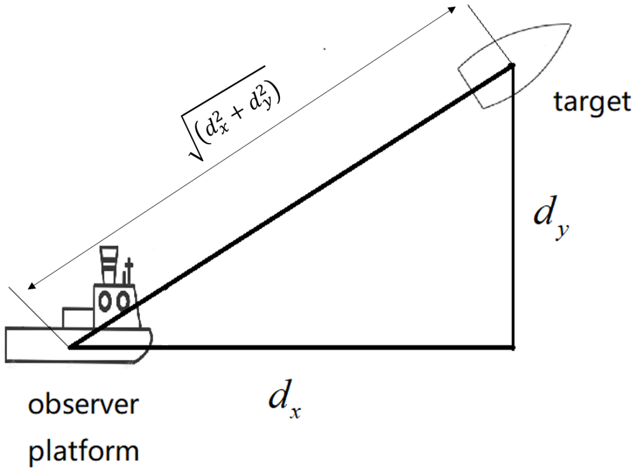

Figure 4.1: The figure of the displacement and range measurement in 2-D

To get a recursive and precise formulation, we prefer the matrix calculation and wish to use the equation (4.2) to describe the equation (4.3) then the easiest way is to set some special H and u^m as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

And then set the u^m as the opposite number of the motion function of the observer. For example, the observer moved in constant process or as a function of time $x_t^p = x_0^p + tc$, $y_t^p = y_0^p + tb$. So the displacement can be expressed as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$u^m = \begin{bmatrix} -(x_0^p + tc) \\ -(y_0^p + tb) \end{bmatrix}$$

Then the relative displacement can be expressed as

$$d_t = Hx + u^m$$

After completing all the preparation for the noisy measurement computation (computing the d_t and designing the matrix H, u^m), we start to compare two noise model from the equation (4.3) and equation(4.4).

The standard noise model :

$$r'_t = \gamma'_{rt} = |d_t| + w'_t, \quad w'_t \sim N(0, \sigma^2)$$

The 'noise before nonlinearity' model:

$$\begin{cases} z_t = d_t + w_t, & w_t \sim N(0, Q^m) \\ r_t = \gamma_{rt} = \psi_2(z_t), & Q^m = \sigma^2 I_{2\times 2} \end{cases}$$

The difference between two equation is the position of the adding noise (traditional one is before the nonlinearity, the new one is after the nonlinearity). And we use the conditional density of the r_t given d_t to compare two models:

For the traditional noise model, the measurement is a Gaussian distribution:

$$p(r'_t|d_t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(r_t - |d|)^2} \tag{4.5}$$

For the 'noise before nonlinearity' model, the measurement can be written by Rice density [9] below:

$$p(r_t|d_t) = \frac{r_t}{\sigma^2} e^{-\frac{1}{2\sigma^2}(r_t^2 + |d|^2)} \frac{1}{\pi} \int_0^{pi} e^{-\frac{r_t|d_t|}{\sigma^2} cos\theta} d\theta \tag{4.6}$$

To compare the models, we choose different $|d_t|$ and σ value to compare the difference in the density of tow models from equation (4.5) and (4.6). We conclude that for most of 'big range value' tracking problems, the two noise models are indistinguishable. Actually when $|d_t|/\sigma > 10$, the density of two model are very close (like figure 4.2). And when the $|d_t|/\sigma$ is bigger, the noise approximation is better. And in the real life, the range information is usually in the level of 'thousand' and the noise in the level of '0.1', so this two models can be indistinguishable in the real life range tracking problems.

We plot the figure of the $|d_t|/\sigma = 20$ in the figure (4.2):

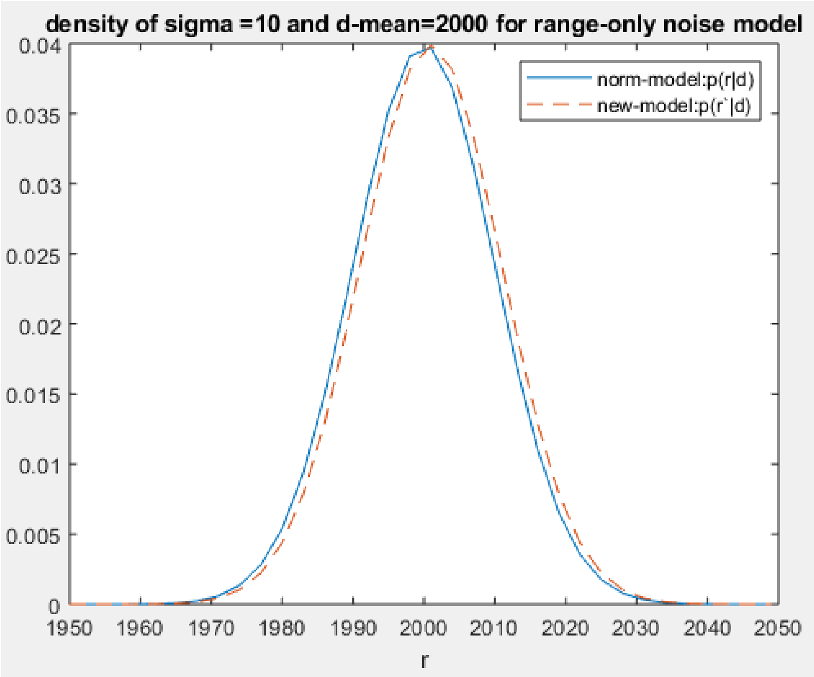

Figure 4.2: The density of two noise models in range-only tracking

From the figure (4.2), we can conclude that the 'noise before nonlinearity' model is very suitable and its approximation to the noise is good. So the model of the noise used in the range-only case can be seen indistinguishable with the normal standard noise model. The method used to construct the augmented vector is a good choice. And it provides the possibility to do the accurate computation of the conditional density with the 'Analytic Filter' algorithm construction.

4.4 The Analytic filter for the Range-Only tracking

Algorithm outline

Step 1. Initial configuration:

$x_{0|0}$ Estimation for the initial state

$P_{0|0}$ The covariance matrix of the initial state (The confidence of the initial state)

Step 2. Prediction step:

Calculate the $\hat{x}_{t|t-1}$ and $P_{t|t-1}$

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \end{cases} \tag{4.7}$$

Step 3. Updating step:

$$\begin{cases} S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^TS^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_tH)P_{t|t-1} + K_t\Gamma_tK_t^T \\ where: \\ \zeta_t = \gamma_{rt}g_t \\ \Gamma_t = \gamma_{rt}^2G_t \end{cases} \tag{4.8}$$

Here we apply the analytic filter into the range-only tracing problem, and list the whole outline of the algorithm above

To complete the computation we need to introduce the function and parameters V_t, g_t, G_t to compute ζ_t and Γ_t:

We define V_t:

$$V_t = S_t^{-1}\begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} \tag{4.9}$$

And then define the parameters:

$$
\begin{aligned}
&a_1 = |V_t d_{t|t-1}| && a_2 = |M_t| \\
&b_1 = \angle(V_t d_{t|t-1}) && b_2 = \angle(M_t) \\
&M_t = \begin{bmatrix} \frac{1}{2}(v_{11} - v_{22}) \\ v_{12} \end{bmatrix}
\end{aligned}
\tag{4.10}
$$

$$
p(\theta_t|r_{1:t}) = c^{-1} e^{(h_t(\theta_t))} \tag{4.11}
$$

where c is the constant to make the the integration of the $p(\theta|r_{1:t})$ from $-\infty$ to ∞ equal to 1. We write the measurement γ_{rt} as r_t for simplification. And the $h_t(\theta)$ is defined as:

$$
h_t(\theta_t) = a_1 r_t cos(\theta_t - b1) - \frac{1}{2} a_2 r_t^2 cos(2\theta_t - b2) \tag{4.12}
$$

Then we can compute the g_t and G_t:

$$
g_t = \int_0^{2\pi} \begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} p(\theta_t|r_{1:t}) d\theta_t \tag{4.13}
$$

$$
G_t = \int_0^{2\pi} \left(\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t \right) \left(\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t \right)^T p(\theta_t|r_{1:t}) d\theta_t \tag{4.14}
$$

And we have completed all the preparation for the updating, we can use the formulation equation (4.15) to update the estimation:

$$
\begin{cases}
\hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\
P_t = (I - K_t H) P_{t|t-1} + K_t \Gamma_t K_t^T \\
where: \\
\zeta_t = \gamma_{rt} g_t \\
\Gamma_t = \gamma_{rt}^2 G_t
\end{cases}
\tag{4.15}
$$

In the update step, the Q_m is the covariance of the measurements: $Q_m = \sigma^2 I_{2\times 2}$. K_t is 'Kalman parameter used to correction the results. The ζ_t is the conditional mean of the d given the γ_{rt} and Γ_t is the conditional covariance of the d given the γ_{rt}.

The proof of the algorithm can be checked later in the Chapter 2.4-2.5 (the structure of the analytic filter has been proved efficient in these sections), and we also will need to justify the calculation of the ζ_t and Γ_t for the range-only case in the next section: 4.5 'Analysis'.

After giving all the parameter we need to compute for the algorithm, we need to check the dimension of the matrix computation again in this bearing-only scenario. Take the two-dimension as an example (it is similar to get the matrix dimension in three-dimension):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

Where the K_t is the Kalman gain, the S_t is the covariance matrix of d_t and ζ_t is the conditional mean of the d_t given the range information, Γ is the conditional variance of d_t given the range information.

4.5 Analysis and Justification

As we know under the assumption: '$p(x_t|r_{1:t})$ is a Gaussian random variable', we can consider that if the augmented vector $z_t = d_t + w_t$ directly can be measured directly, then we can compute the $p(x_t|z_t)$ by the standard Kalman filter directly:

$$p(x_t|z_t) = N(x_{t|t-1} + K_t(z_t - d_t|t-1), (I - K_tH)P_{t|t-1}) \tag{4.16}$$

Actually, we can replace the z_t by $z_t = |z_t|\frac{z_t}{|z_t|} = r_tb_t$, where:

$$b_t = \begin{bmatrix} cos\theta \\ sin\theta \end{bmatrix} = \angle z_t \tag{4.17}$$

$$r_t = |z_t| \tag{4.18}$$

We can get the new formulation:

$$p(x_t|r_{1:t}, b_{1:t}) = N(x_{t|t-1} + K_t(r_tb_t - d_t|t-1), (I - K_tH)P_{t|t-1}) \tag{4.19}$$

We can get the joint density of the r_t and θ_t given $r_{1:t-1}$ by expressing the $N(d_t|t-1, S_t)$ in polar coordinates[11]:

$$\begin{aligned} &p(\theta_t, r_t|r_{1:t-1}) = \tfrac{|V_t|^{1/2}r_t}{2\pi} f_t e^{(h_t(\theta_t))} \\ &f_t = e^{-\frac{1}{4}r_t^2(v_{11}+v_{22})-\frac{1}{2}d_{t|t-1}^T V_t d_{t|t-1}} \\ &h_t(\theta_t) = a_1r_tcos(\theta_t - b1) - \tfrac{1}{2}a_2r_t^2cos(2\theta_t - b2) \\ &p(\theta_t|r_{1:t}) = c^{-1}e^{h_t(\theta_t)} \end{aligned} \tag{4.20}$$

The parameters a_1, a_2, b_1, b_2, V_t can be seen in the equation (4.9)-(4.12).

By combing he equation (4.20) and (4.19), we can compute the

$$p(x_t|r_{1:t}) = \int_0^{2\pi} N(x_{t|t-1} + K_t(r_tb_t(\theta_t) - d_t|t-1), (I - K_tH)P_{t|t-1}) * p(r_t, \theta_t|r_{1:t-1})d\theta_t \tag{4.21}$$

Then we can compute the mean and covariance for the $p(x_t|r_{1:t})$ and do the matching process.

First to prove the $\zeta = r_t g_t$ and calculate the mean of the state estimation:

$$
\begin{aligned}
& x_t = \int_0^{2\pi} x_t p(x_t|r_{1:t})dx_t \\
& = \int_0^{2\pi} (\int_{R^2} x_t p(x_t|r_{1:t}, \theta_t)dx_t)p(\theta_t|r_{1:t})d\theta_t \\
& = x_{t|t-1} - K_t d_{t|t-1} + K_t r_t E[b(\theta_t)|r_{1:t}] \\
& = x_{t|t-1} - K_t d_{t|t-1} + K_t r_t \int_0^{2\pi} \begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} p(\theta_t|r_{1:t})d\theta_t \\
& = x_{t|t-1} - K_t d_{t|t-1} + K_t r_t g_t \\
& = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1})
\end{aligned}
\tag{4.22}
$$

So we ave proved the equation:

$$\zeta = r_t g_t$$

And:

$$g_t = \int_0^{2\pi} \begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} p(\theta_t|r_{1:t})d\theta_t$$

Secondly, we want to prove the $\Gamma = r_t^2 G$ and calculate the covariance of the state estimation:

$$
\begin{aligned}
& P_t = \int_{R^2} (x_t - \hat{x}_t)(x_t - \hat{x}_t)^T p(x_t|r_1 : t)dx_t \\
& = \int_0^{2\pi} (\int_{R^2} (x_t - \hat{x}_t)(x_t - \hat{x}_t)^T p(x_t|r_1 : t, \theta_t)dx_t) \times p(\theta_t|r_{1:t})d\theta_t \\
& = (I - K_t H)P_{t|t-1} + r_t^2 K_t Cov[b_t|r_{1:t}]K_t^T \\
& = (I - K_t H)P_{t|t-1} + r_t^2 K_t \int_0^{2\pi} (\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t)(\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t)^T p(\theta_t|r_{1:t})d\theta_t K_t^T \\
& = (I - K_t H)P_{t|t-1} + r_t^2 K_t G_t K_t^T
\end{aligned}
\tag{4.23}
$$

So we ave proved the equation:

$$\Gamma = r_t^2 G_t$$

And:

$$G_t = \int_0^{2\pi} (\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t)(\begin{bmatrix} cos(\theta_t) \\ sin(\theta_t) \end{bmatrix} - g_t)^T p(\theta_t|r_{1:t})d\theta_t$$

We have finished all the justification of the filter, and confirmed that all the parameters are right for the filter. We would run the simulation to check the algorithm can work well or not in the next section.

4.6 Simulation and Conclusion

As the problem formulation and its motion description, we have the state description equation:

$$x_t = Fx_{t-1} + u^s + v_t \quad (v_t \sim N(0, Q^s))$$

We use the x_t which contains the position information and velocity information in two-dimension:

$$x_t = \begin{bmatrix} x \\ V_x \\ y \\ V_y \end{bmatrix}$$

We choose the scenario where the target moves in one direction (constant velocity situation) and the ship is moving in another direction with constant velocity.

So the $u^s = 0$ and F is in the form below:

$$x_t = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{t-1} + \begin{bmatrix} T^2/2 \\ T \\ T^2/2 \\ T \end{bmatrix} v_{t-1}$$

We can set different initial value of the V_x, V_y and x_0, y_0 to run the simulation. (In this simulation, the $V_x = 15$, $V_y = 12$ $x_0 = 80$, $y_0 = 20$) And v_t is a Gaussian noise $N(0, 0.01)$, the sample time of the discrete-time system is $T = 1s$.
And also the ship shifted in both x and y direction. So the observer moved in the way:

$$x_{observer} = \begin{bmatrix} x_{observer} \\ y_{observer} \end{bmatrix} = \begin{bmatrix} 2t + w_1 \\ 3t + w_2 \end{bmatrix}$$

w_1 and w_2 are both Gaussian noise with $N(0, 1)$. In this way, we make the ship to shift randomly.

For the fair comparison, we add the noise after nonlinearity. So we make the measurements in the form of:

$$\gamma_t = \sqrt{(y_{target} - y_{observer})^2 + (x_{target} - x_{observer})^2} + w_t$$
$$w_t \sim N(0, \sigma^2), \sigma = 0.1$$

Approximate the initial value of the state and its covariance to prepare to run the algorithm (we set an error between the approximation and the real position to check whether the algorithm is sensitive to the initial value):

$$x_{0|0} = \begin{bmatrix} 120 \\ 10 \\ -20 \\ 10 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

All the preparations are completed for the filter:

$$u_t^s = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, u_t^m = \begin{bmatrix} -2t \\ -3t \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, F = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q^s = 0.01 \begin{bmatrix} 0.25 & 0.5 & 0 & 0 \\ 0.5 & 0.1 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}$$

Then we run the algorithm:

Prediction step:

Calculate the $\hat{x}_{t|t-1}$, $P_{t|t-1}$:

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \end{cases}$$

Updating step:

Compute the recursive equation below and plot the figure

$$
\begin{cases}
S_t = HP_{t|t-1}H_T + Q^m \\
K_t = P_{t|t-1}H^T S^{-1} \\
\hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\
P_t = (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T \\
where: \\
\zeta_t = r_t g_t \\
\Gamma_t = r_t^2 G_t
\end{cases}
$$

Simulation result:

We run the algorithm we designed before and EKF for the range-only case. And set different initial values to compare the performance of two filters:

1.Good initial value:

$$
x_{0|0} = \begin{bmatrix} 120 \\ 10 \\ -20 \\ 10 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.24}
$$

2.Bad initial value:

$$
x_{0|0} = \begin{bmatrix} 120 \\ 0 \\ -20 \\ 0 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.25}
$$

1. When we set the good initial values (equation 4.24) for the filter:

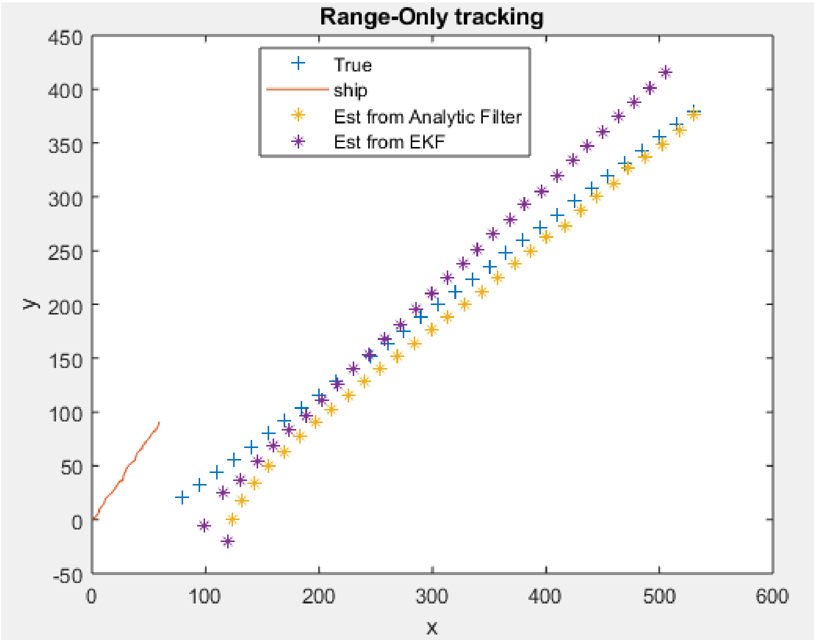

Figure 4.3: The tracking performance of the analytic filter and EKF in range-only case

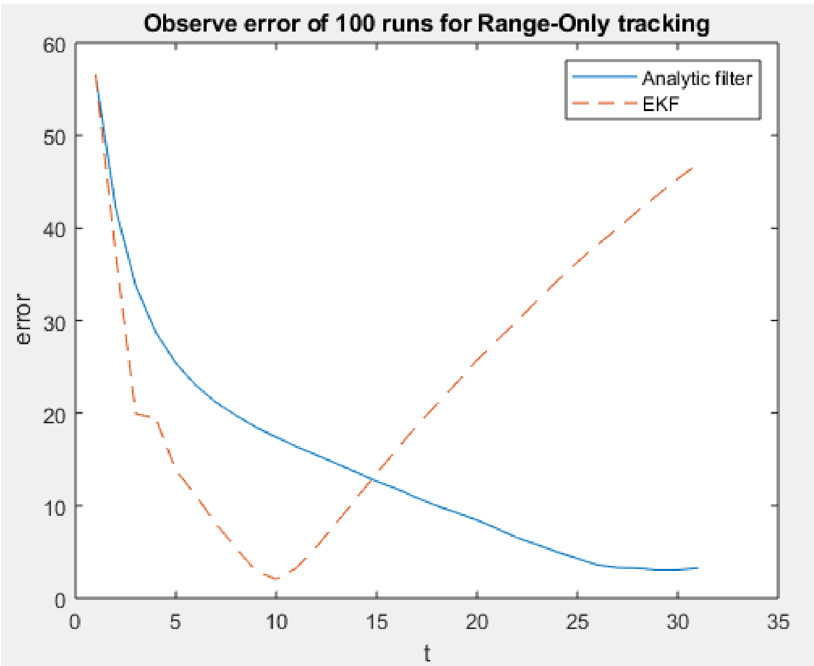

Figure 4.4: Error of 100 runs for analytic filter and EKF in range-only case

2. When we set the bad initial values (equation 4.25) for the filter:

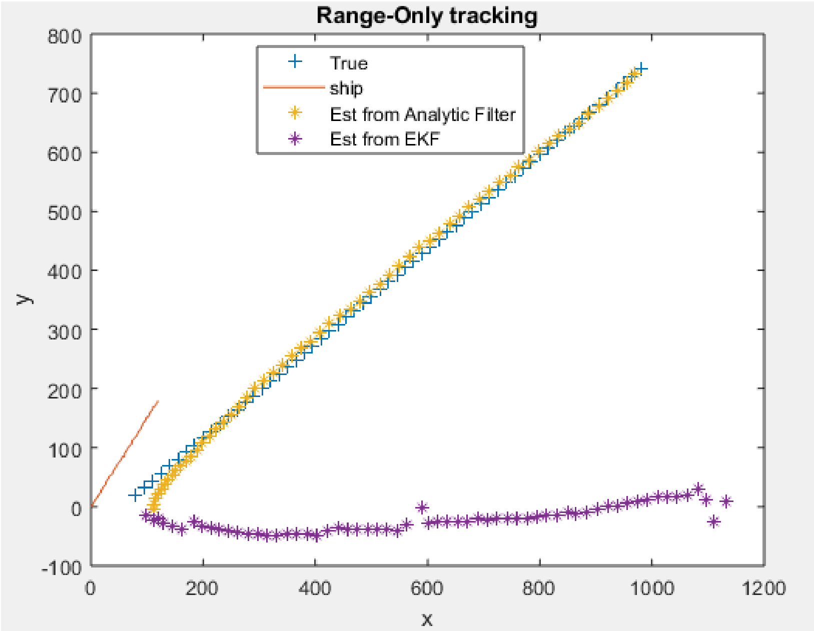

Figure 4.5: The tracking performance of the analytic filter and EKF in range-only case with bad initial value

Figure 4.6: Error of 100 runs for analytic filter and EKF in range-only case with bad initial value

From the figure 4.3-4.4, we can see that both filters can work well in the early steps with the good initial values. They all can converge quickly, but for the EKF, it is not robust, and may lose the convergence while the step increasing. And from the figures of the tracking performance and error, the analytic filter works better than the EKF filter. Analytic filter is faster and more accurate in this scenario.

Figure 4.5-4.6 show that when the initial value is not very good, the analytic filter can converge but the EKF would lose the target. It means that the analytic filter is less sensitive to the initial value than the EKF.

Chapter 5

Bearing-Range mixture tracking Problem

5.1 Background

This section focuses on the bearing-range mixture tracking problems. In some case, we can get both bearing and range information. And we wish that the estimation from the mixture tracking problem could be more accurate than the estimation from only bearing or range cases.

In mixture bearing and range tracking, the challenges are from the nonlinear function of the bearing and range information. And also the different noise level is hard to handle or model.

Bearing measurements are taken from the angle of the relative displacement vector. The measurement of the bearing information is defined as: $\theta = \angle d$, and the 'bearing vector' is defined as $\psi_1(d_t) = |d|^{-1}d$. On the other hand, range measurements are taken from the euclidean distance of the relative displacement vector. The measurement of the range information is defined as: $r_t = |d|$.

The main difficulty for this estimation is derived from the its nonlinear computation. There is no direct and practical equation for the conditional density of the state given the two measurements. So the common practical algorithms usually choose to use the method of the approximations.

The extended Kalman filter is a typical algorithm to handle with this nonlinear estimation problem, it approximates the conditional density by a Gaussian process[4]. And the Particle filter is another typical filter which uses a empirical distribution to approximate the conditional density[3]. Also there is a new research on this mixture tracking problem with robust linear filter[5] and also a approach by the curved wavefronts [6].

In this kind of problem, the EKF do a better work with more efficient performance. But sometimes, it is not robust. The EKF compute the linear equation by the taylor series at each step, and it will have bad performance when the figure is at the turning point. (In real life, it is a common situation where the target choose to change the direction of the motion).

In this section, we wish to apply the 'Analytic filter' introduced in Chapter 2 to this tracking. The aim of this filter is to improve the performance of the filter in the bearing-range mixture case and also not to increase too much computation cost.

This algorithm uses the conditional mean of the relative displacement to replace the nonlinear part in the computation with Kalman structure. This method has been proved and its details are in the Chapter 2. The approximation is very efficient and transparent. And combine the range and bearing information we measured, we can get the relative displacement (augmented vector) for the updating of estimation.

This algorithm has two typical themes, one is to model the noise as 'noise before nonlinearity'. Another is to approximate the state by the knowledge of the 'augmented vector d_t. The algorithm can save the computational cost for its matrix and trigonometric computation. This section is based on the success of the 'Shifted Rayleigh Filter'[8], a typical 'Analytic Filter' proposed before. We apply this analytic filter into the mixture bearing and range cases.

5.2 Problem Formulation and noise model

The figure below shows how the relative distance and the bearing information:

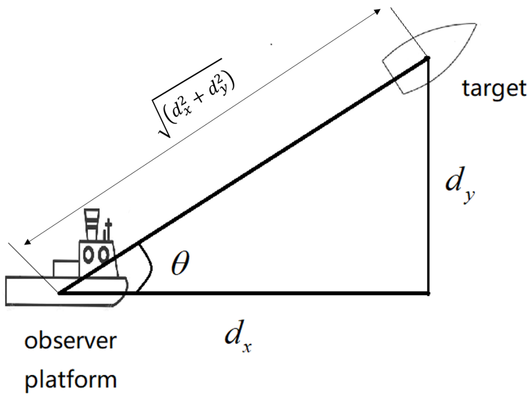

Figure 5.1: The figure of the displacement with bearing-range measurement in 2-D

To get a recursive and precise formulation, we prefer the matrix calculation and wish to use the equation (5.2) then the easiest way is to set some special H and u^m as:

Set the u^m as the opposite number of the motion function of the observer. For example, the observer moved in constant process or as a function of time $x_t^p = x_0^p + tc$, $y_t^p = y_0^p + tb$. So the displacement can be expressed as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$u^m = \begin{bmatrix} -(x_0^p + tc) \\ -(y_0^p + tb) \end{bmatrix}$$

Then the relative displacement can be expressed as

$$d_t = Hx + u^m$$

After gaining the relative displacement, we can set the noise model by equation (5.2).

For the fair comparison, we list two noise models of the system (use 2-dimension as example):

The customary system model is:

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t \quad (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m \\ \gamma_{\theta t} = \angle(d_t) + w_1 \qquad (w_1 \sim N(0, \sigma_b^2)) \\ \gamma_{rt} = |d_t| + w_2 \qquad (w_2 \sim N(0, \sigma_r^2)) \end{cases} \tag{5.1}$$

The analytic system model ('noise before nonlinearity'):

$$\begin{cases} x_t = Fx_{t-1} + u^s + v_t \quad (v_t \sim N(0, Q^s)) \\ d_t = Hx_t + u^m \\ z_t = d_t + w_t \qquad (w_t \sim N(0, Q^m)) \\ \gamma_{\theta t} = \psi_1(z_t) \\ \gamma_{rt} = \psi_2(z_t) \end{cases} \tag{5.2}$$

where the

$$\psi_1(z_t) = |z|^{-1}z = (cos(\theta), sin(\theta))$$

$$\psi_2(z_t) = |z|$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Of particular interest are some scenarios when the 'measurement' noise take the form below: (we can cited this structure from the paper[8])

$$Q_t^m = \sigma_b^2 E[|d_t|^2 | b_{1:t-1}, r_{1:t-1}] I_{k\times k} + Q^{tr} = \sigma'^2 I_{k\times k} \tag{5.3}$$

Actually, we have proved the 'noise before nonlinearity' model works well in range measurement case and bearing measurement case. And what is interesting, the Q_m in two case all can take the form like equation (5.3). And if it is not suitable, we can change the parameter Q^{tr} which characterize the transition of the noise.

Just choose suitable σ'^2: the variance parameter for the noise in the 'augmented vector'. A good choice is to use the equation (5.3), in this way we put the bearing noise model in the first place (we use the noise structure of the bearing case, so the $p(\theta_t|d_t)$ matches well, the proof can be seen in the Chapter 3).

Choosing the structure of the bearing noise is because that the bearing information plays a more important role in this tracking problem. And it is easy to change the Q_{tr} to make the density more close. Also, we set the Q^{tr} in the form of the $Q^{tr}_{1,1} = Q^{tr}_{2,2}$, such that we can matches the range density as close as possible.

A easy way to handle with the covariance is to set the

$$Q^{tr} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Where a is a constant.

$$Q^m_t = Q^{tr} + \sigma_b^2(|H\hat{x}_{t|t-1} + u^m_t|^2 + tr(HP_{t|t-1}H^T))I_{2\times 2}$$

Base on the structure we have finished the bearing match (the structure is suitable for the bearing case), and then we wish we could match the range measurement as close as possible:

As we know the traditional noise model of the range measurement is that:

$$\begin{aligned} d_t &= Hx_t + u^m \\ \theta'_t &= \gamma_{\theta t} = \angle(d_t) + w_1 \quad (w_1 \sim N(0, \sigma_b^2)) \\ r'_t &= \gamma_{rt} = |d_t| + w_2 \quad\quad (w_2 \sim N(0, \sigma_r^2)) \end{aligned} \tag{5.4}$$

The new noise model is that:

$$\begin{aligned} d_t &= Hx_t + u^m \\ z_t &= d_t + w_t \qquad (w_t \sim N(0, Q^m = \sigma'^2 I_{2\times 2})) \\ \theta_t &= \gamma_{\theta t} = \psi_1(z_t) \\ r_t &= \gamma_{rt} = \psi_2(z_t) \end{aligned} \tag{5.5}$$

For the traditional noise model, the measurement is a Gaussian distribution:

$$p(r'_t|d_t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(r_t - |d|)^2} \tag{5.6}$$

For the 'noise before nonlinearity' model, the measurement can be written by Rice density [9] below:

$$p(r_t|d_t) = \frac{r_t}{\sigma'^2} e^{-\frac{1}{2\sigma'^2}(r_t^2 + |d|^2)} \frac{1}{\pi} \int_0^{pi} e^{-\frac{r_t|d_t|}{\sigma'^2} cos\theta} d\theta \tag{5.7}$$

The only work is to check whether the density with the new σ' (5.7) is very close to the density with σ_r (5.6) under assumption that the $|d_t|/\sigma_t > 10$.

$$Q_t^m = Q^{tr} + \sigma_b^2(|H\hat{x}_{t|t-1} + u_t^m|^2 + tr(HP_{t|t-1}H^T))I_{2\times 2} \approx \sigma_r^2 I_{2\times 2} \tag{5.8}$$

So if the assumption 5.8 can be reached then the filter can works very well. And if the assumption cannot be reached, the noise model is to match the bearing measurements firstly, which may bring some errors here. And if more accuracy is required then we need to change the parameters of Q^m.

5.3 The Analytic filter for the bearing-range mixture tracking

After modeling the noise, we then can apply the analytic filter into this tracking:

The algorithm generates $(\hat{x}_t, P_t)$ at each t to approximate the first and second moments of the conditional density $p(x_t|\gamma_{1:t})$ with the new measurement γ_t and the $(\hat{x}_{t-1}, P_{t-1})$ calculated before, we divide the filter into three parts:

Step 1. Initial configuration:

$x_{0|0}$ Estimation for the initial state

$P_{0|0}$ The covariance matrix of the initial state (The confidence of the initial state)

Step 2. Prediction step:

Calculate the $\hat{x}_{t|t-1}$ and $P_{t|t-1}$

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \end{cases} \tag{5.9}$$

Step 3. Updating step:

$$\begin{cases} Q_t^m = Q^{tr} + \sigma^2(|H\hat{x}_{t|t-1} + u_t^m|^2 + tr(HP_{t|t-1}H^T))I_{r\times r} \\ S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^TS^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_tH)P_{t|t-1} + K_t\Gamma_tK_t^T \\ where: \\ \zeta_t = [r_t\cos(\theta_t), r_t\sin(\theta_t)] \\ \Gamma_t = 0 \end{cases} \tag{5.10}$$

Here we apply the analytic filter into the bearing-range tracking problem, and list the whole outline of the algorithm above.

We firstly check the dimension of the matrix computation in 2-dimension. (it is similar to get the matrix dimension in three-dimension.):

$$x : 4 \times 1, F : 4 \times 4, u^s : 4 \times 1, d : 2 \times 1$$

$$H : 2 \times 4, u^m : 2 \times 1, P : 4 \times 4, Q^s : 4 \times 4$$

$$Q^m : 2 \times 2, S : 2 \times 2, K : 4 \times 2$$

As we know under the assumption: '$p(x_t|r_{1:t}, \theta_{1:t})$ is a Gaussian random variable', we can consider that if the augmented vector $z_t = d_t + w_t$ can be measured directly, then we can compute the $p(x_t|z_t)$ by the standard Kalman filter directly (actually the z_t is the determined function of the r_t and θ_t):

$$p(x_t|z_t) = N(x_{t|t-1} + K_t(z_t - d_{t|t-1}), (I - K_t H)P_{t|t-1}) \tag{5.11}$$

Actually, we can replace the z_t by $z_t = |z_t|\frac{z_t}{|z_t|} = r_t b_t$, where:

$$b_t = \begin{bmatrix} cos\theta \\ sin\theta \end{bmatrix} = \angle z_t \tag{5.12}$$

$$r_t = |z_t| \tag{5.13}$$

We can get the new formulation:

$$p(x_t|r_{1:t}, b_{1:t}) = N(x_{t|t-1} + K_t(r_t b_t - d_t|t - 1), (I - K_t H)P_{t|t-1}) \tag{5.14}$$

So in the bearing-range mixture algorithm we just assume that:

$$p(x_t|r_{1:t}, \theta_{1:t}) = N(x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}), (I - K_t H)P_{t|t-1})$$

$$\zeta_t = E[z|\psi(z) = \gamma_t] = [r_t \cos(\theta_t), r_t \sin(\theta_t)]$$

$$\Gamma \approx 0$$

What is interesting is that, it is consistent with the definition we used in the analytic filter:

$$\zeta_t = E[d|\psi(z) = \gamma_t], d \sim N(d_{t|t-1}, S_t)$$

$$\Gamma_t = Cov[d|\psi(z) = \gamma_t], d \sim N(d_{t|t-1}, S_t)$$

Here we use the determined function to compute mean and covariance of the d_t given γ_{tr} and $\gamma_{t\theta}$. So in the ideal situation (the model is really close to the real case), the mean is $\zeta_t = [r_t \cos(\theta_t), r_t \sin(\theta_t)]$, the covariance is $\Gamma = 0$ for its known function.

This algorithm saves lots of computation cost for its linear matrix computation. And also if the noise is suitable, the accuracy can also be confirmed. If it works not well, we can just change the matrix Q^{tr} in the noise model.

5.4 Simulation and Conclusion

As the problem formulation and its motion description, we have the state description equation:

$$x_t = Fx_{t-1} + u^s + v_t \quad (v_t \sim N(0, Q^s))$$

We use the x_t which contains the position information and velocity information in two-dimension:

$$x_t = \begin{bmatrix} x \\ V_x \\ y \\ V_y \end{bmatrix}$$

We choose the scenario where the target moves in one direction (constant velocity situation) and the ship is moving in another direction with constant velocity.

So the $u^s = 0$ and F is in the form below:

$$x_t = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{t-1} + \begin{bmatrix} T^2/2 \\ T \\ T^2/2 \\ T \end{bmatrix} v_{t-1}$$

We can set different initial value of the V_x, V_y and x_0, y_0 to run the simulation. (In this simulation, the $V_x = 15$, $V_y = 12$ $x_0 = 80$, $y_0 = 20$) And v_t is a Gaussian noise $N(0, 0.01)$, the sample time of the discrete-time system is $T = 1s$.
And also the ship shifted in both x and y direction. So the observer moved in the way:

$$x_{observer} = \begin{bmatrix} x_{observer} \\ y_{observer} \end{bmatrix} = \begin{bmatrix} 2t + w_1 \\ 3t + w_2 \end{bmatrix}$$

w_1 and w_2 are both Gaussian noise with $N(0, 1)$. In this way, we make the ship to shift randomly.

For the fair comparison, we add the noise after nonlinearity. So we make the measurements in the form of:

$$\gamma_t = \sqrt{(y_{target} - y_{observer})^2 + (x_{target} - x_{observer})^2} + w_t$$
$$w_t \sim N(0, \sigma^2), \sigma = 0.1$$

Approximate the initial value of the state and its covariance to prepare to run the algorithm (we set an error between the approximation and the real position to check whether the algorithm is sensitive to the initial value):

$$x_{0|0} = \begin{bmatrix} 120 \\ 10 \\ -20 \\ 10 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

All the preparations are completed for the filter:

$$u_t^s = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, u_t^m = \begin{bmatrix} -2t \\ -3t \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, F = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q^s = 0.01 \begin{bmatrix} 0.25 & 0.5 & 0 & 0 \\ 0.5 & 0.1 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}$$

Then we run the algorithm:

Prediction step:

Calculate the $\hat{x}_{t|t-1}$, $P_{t|t-1}$:

$$\begin{cases} x_{t|t-1} = F\hat{x}_{t-1} + u^s \\ d_{t|t-1} = Hx_{t|t-1} + u^m \\ P_{t|t-1} = FP_{t-1}F^T + Q^s \end{cases}$$

Updating step:

Compute the recursive equation below and plot the figure

$$\begin{cases} S_t = HP_{t|t-1}H_T + Q^m \\ K_t = P_{t|t-1}H^T S^{-1} \\ \hat{x}_t = x_{t|t-1} + K_t(\zeta_t - d_{t|t-1}) \\ P_t = (I - K_t H)P_{t|t-1} + K_t \Gamma_t K_t^T \\ where: \\ \zeta_t = [r_t \cos(\theta_t), r_t \sin(\theta_t)] \\ \Gamma_t = 0 \end{cases}$$

Simulation result:

We run the algorithm we designed before and EKF for the bearing-range mixture case. And set different initial values to compare the performance of two filters:

1.Good initial value:

$$x_{0|0} = \begin{bmatrix} 120 \\ 10 \\ -20 \\ 10 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.15}$$

2.Bad initial value:

$$x_{0|0} = \begin{bmatrix} 120 \\ 0 \\ -20 \\ 10 \end{bmatrix}, P_{0|0} = 10 \begin{bmatrix} 20 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.16}$$

1. When we set the good initial values (equation 5.10) for the filter:

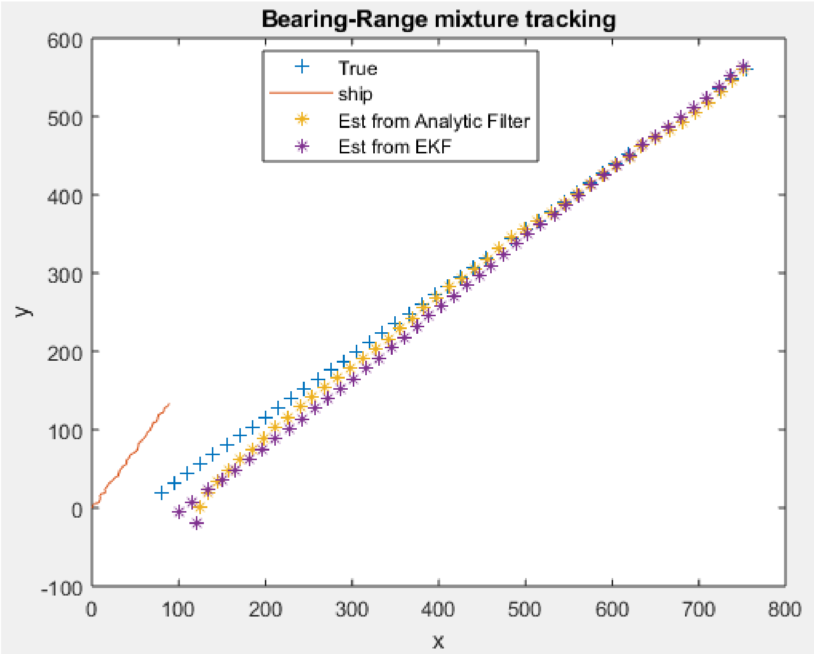

Figure 5.2: The tracking in the bearing-range case

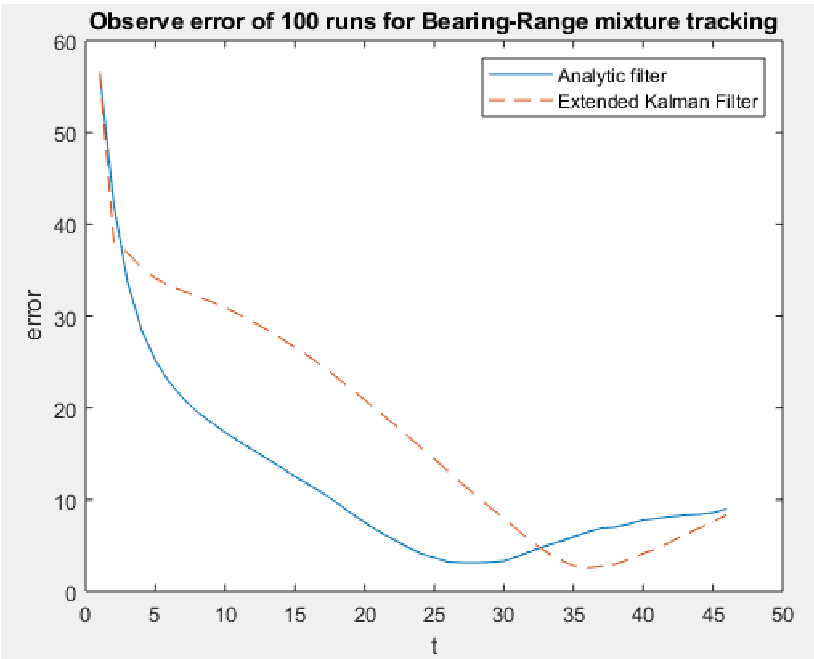

Figure 5.3: The error in the bearing-range case

2.When we set the bad initial values (equation 5.11) for the filter:

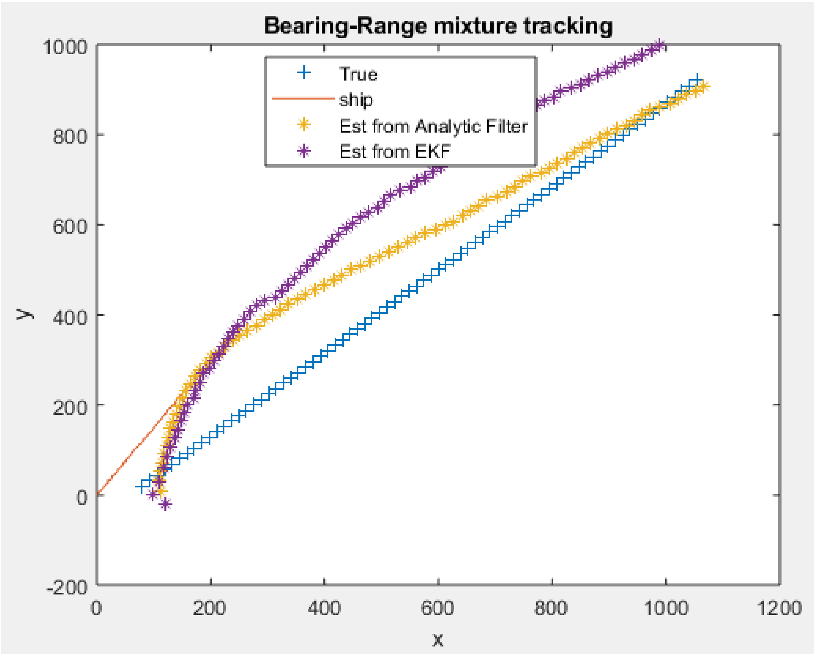

Figure 5.4: Error of 100 runs in the bearing-range case with bad initial value

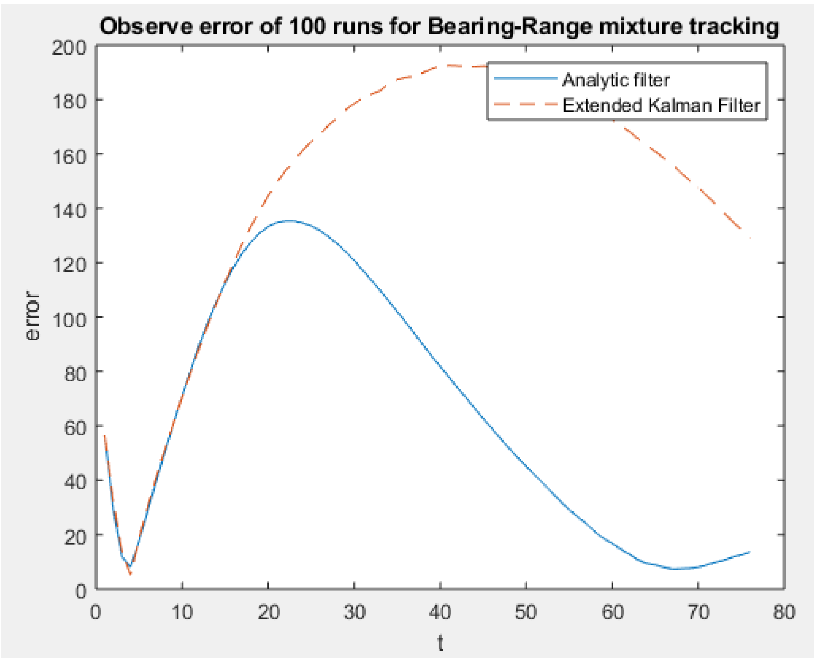

Figure 5.5: Error of 100 runs in the bearing-range case with bad initial value

From the figure 5.2-5.3, we can see that both filters can converge with or good initial values. It is because of its full information of the target. And from the figures of the tracking performance and error, the analytic filter works better than the EKF filter. Analytic filter is faster and more accurate in this scenario.

Figure 5.4-5.5 show that when the initial value is not very good, the analytic filter can converge but the EKF would lose the target. It means that the analytic filter has less requirements about the initial value than the EKF.

Chapter 6

Conclusion and summarization

In this thesis, we have introduced the 'analytic filter' and proved it theoretically. Then we applied it into three typical tracking cases to see that whether it can do a good job. Compared with some common filter, it can converge quickly and save lots of computation cost.

In bearing-only case, the pseudo-measurement filter and analytic filter (Shifted Rayleigh filter) both can work well, and the SRF can even work better in the one direction scenario. But Extended Kalman filter cannot work well in this scenario, it may lose the target.

In range-only case, the analytic filter can work well in constant velocity scenario compared with the Extended Kalman filter. And the analytic filter is not very sensitive to the initial values. For the extended Kalma filter, bad initial values of the state may lead to divergence. But analytic filter can also keep its good performance.

In bearing-range mixture case, the analytic filter and the extended Kalman filter can both work very well in constant velocity scenario under the suitable noise assumption. And the analytic filter is not very sensitive to the initial values. It converge more quickly than the EKF. And the algorithm of analytic filter is very easy to compute. In EKF we need to do the partial differential computation, but the analytic filter just do the matrix computation.

In conclusion, the analytic filter can save much computation cost and it can also have a better performance in many special or common scenarios, which is efficient and robust in some cases.

Here we just summarize the parameters used in analytic filter of different scenarios in the table 6.1 below:

Scenario	ζ_t	Γ_t
Bearing-Only	$\zeta_t = (\gamma_{\theta t}^T S_t^{-1} \gamma_{\theta t})^{-1/2} \rho_r(z_t) \gamma_{\theta t}$	$\Gamma_t = \delta_t^r \gamma_{\theta t} \gamma_{\theta t}^T$
Range-Only	$\zeta_t = \gamma_{rt} g_t$	$\Gamma_t = \gamma_{rt}^2 G_t$
Bearing-Range Mixture	$\zeta_t = [\gamma_{rt} cos(\gamma_{\theta t}), \gamma_{rt} sin(\gamma_{\theta t})]^T$	0

Table 6.1: The parameter table of the analytic filter used in different case

The derivation of the parameters ρ_r, z_t, δ used in bearing-only tracking can be seen in the Chapter 3. And also can be checked in the paper[7]

The derivation of the parameters g_t, G_t used in rang-only tracking can be seen in the Chapter 4. And also can be checked in the paper [9]

Chapter 7

Future Work

In this thesis, we focus on the performance of the 'analytic filter' in three typical tracking cases. Especially, we pay much attention to the tracking performance and the error of the observation. And in the future, we can also do research on the confidence area of the estimation.

Also the less computation cost is one main advantage of the analytic filter, we can do some comparison in this area. We can choose some filters and run the algorithm, compare the computation cost of them.

We compare the filter only with the EKF and the pseudo-measurement filter for the limited time. In the future, we can compare the analytic filter with more filers. Maybe we can find a better one for the common scenarios (constant velocity motion under noisy nonlinear measurements).

The scenarios we designed are all very simple. Actually the SRF can do good performance in many complex scenario[12]. And i think we can find some special scenarios where the SRF or the analytic filter can work best. And in this way to make full use of the computation of the density and moment matching.

About the redesign of the noise part in the algorithm, we just exploit and extend the structure from the conclusion of the paper [7]:

$$Q_t^m = \sigma^2 E[|d_t|^2 | b_{1:t-1}, r_{1:t-1}] I_{k\times k} + Q^{tr}$$

There is an estimation part matrix Q^{tr} to make the whole design flexible. If the noise matrix is not suitable we can change the parameters to make two model more close. But if we can design a theoretical method to compute the noise matrix Q^m of the augmented vector. With different optimal principle, compute different Q^m and check the joint density of the $p(r_t, \theta_t | d_t)$ with the standard noise model. We can complete the strictly justification

From the discussion with Professor Vinter ,we also have anther idea to handle with the bearing-range mixture cases. Do the bearing-only tracking as the main part and use the range-only tracking algorithm to decrease the covariance matrix. But the heavy computation used in the Gaussian mixture algorithm is the hardest part in this idea. Besides, in the range only tracking, the analytic filer contains some integral computation, the value of the P matrix affects the whole computation. When the P become very small (means the situation where estimation is very accurate in the bearing part), then the parameter V_t, $a1$, $a2$, $b1$, $b2$ (the equation can be seen in the Chapter 4) all become bigger. Then the exponential computation may lead to infinity. And in the future, maybe we can find some way to avoid or overcome this situation.

Bibliography

[1] V. J. Aidala. Kalman filter behavior in bearings-only tracking applications. *IEEE Transactions on Aerospace and Electronic Systems*, AES-15(1):29–39, Jan 1979. pages 7

[2] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, Feb 2002. pages 8

[3] M Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on signal processing*, 50(2):174–188, 2002. pages 45, 64

[4] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004. pages 8, 17, 45, 64

[5] Adrian N Bishop, Pubudu N Pathirana, and Andrey V Savkin. Target tracking with range and bearing measurements via robust linear filtering. In *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, pages 131–135. IEEE, 2007. pages 64

[6] E. Boyer, A. Ferreol, and P. Larzabal. Simple robust bearing-range source's localization with curved wavefronts. *IEEE Signal Processing Letters*, 12(6):457–460, June 2005. pages 64

[7] JMC Clark, RB Vinter, and MM Yaqoob. The shifted rayleigh filter for bearings

only tracking. In *Information Fusion, 2005 8th International Conference on*, volume 1, pages 8–pp. IEEE, 2005. pages 22, 28, 32, 35, 79, 81

[8] JMC Clark, RB Vinter, and MM Yaqoob. Shifted rayleigh filter: A new algorithm for bearings-only tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4), 2007. pages 21, 24, 35, 65, 66

[9] John MC Clark, Panagiotis-Aristidis Kountouriotis, and Richard B Vinter. A gaussian mixture filter for range-only tracking. *IEEE Transactions on Automatic Control*, 56(3):602–613, 2011. pages 50, 68, 79

[10] John MC Clark, SA Robbiati, and Richard B Vinter. The shifted rayleigh mixture filter for bearings-only tracking of maneuvering targets. *IEEE transactions on signal processing*, 55(7):3218–3226, 2007. pages 22

[11] M. Clark and R. Vinter. A new class of moment matching filters for nonlinear tracking and estimation problems. pages 108–112, Sept 2006. pages 10, 55

[12] Martin Clark, Simon Maskell, Richard Vinter, and Moeen Yaqoob. A comparison of the particle and shifted rayleigh filters in their application to a multisensor bearings-only problem. In *Aerospace Conference, 2005 IEEE*, pages 2142–2147. IEEE, 2005. pages 22, 80

[13] Dan Crisan and Boris Rozovskii. *The Oxford handbook of nonlinear filtering*. Oxford University Press, 2011. pages 11

[14] Alfonso Farina. Target tracking with bearings–only measurements. *Signal processing*, 78(1):61–78, 1999. pages 21

[15] Alfonso Farina, B Ristic, S Immediata, and Luca Timmoneri. Crlb with pd¡ 1 for bearings only fused tracks. In *Proc. 6th International Conf. on Information Fusion*, volume 2, pages 919–925, 2003. pages 21

[16] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, March 2000. pages 7, 8

[17] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *Proceedings of 1995 American Control Conference - ACC'95*, volume 3, pages 1628–1632 vol.3, June 1995. pages 17

[18] G. Kantor and S. Singh. Preliminary results in range-only localization and mapping. *IEEE International Conference on Robotics and Automation*, 2:1818–1823 vol.2, 2002. pages 2

[19] Thiagalingam Kirubarajan and Yaakov Bar-Shalom. Probabilistic data association techniques for target tracking in clutter. *Proceedings of the IEEE*, 92(3):536–557, 2004. pages 44

[20] A. G. Lingren and K. F. Gong. Position and velocity estimation via bearing observations. *IEEE Transactions on Aerospace and Electronic Systems*, AES-14(4):564–577, July 1978. pages 7, 40

[21] N. Peach. Bearings-only tracking using a set of range-parameterised extended kalman filters. *IEE Proceedings - Control Theory and Applications*, 142(1):73–80, Jan 1995. pages 8

[22] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house, 2003. pages 9