Smoothing Techniques and a Spherical Simplex Unscented Transformation in Solving a SLAM Problem

2012, 9

SAIFUDIN BIN RAZALI

Graduate School of
Natural Science and Technology
(Doctor's Course)
OKAYAMA UNIVERSITY

Abstract

This thesis focuses on the use of unscented transformation method to solve a simultaneous localization and mapping (SLAM) problem. SLAM is the process by which a mobile robot can build a map of an environment and at the same time use this map to compute its location. It can be performed by storing landmarks in a map when they are observed by the robot sensors, using the robot pose estimate to determine the landmark locations, while at the same time, using these landmarks to improve the robot pose estimate. Since the landmarks are repeatedly reobserved, their locations become increasingly certain and the map converges, eventually acquiring the rigidity of a priori map. Many solutions to the SLAM problem are focused on the filtering approaches such as the use of the extended Kalman filter (EKF), the unscented Kalman filter (UKF), the particle filter (PF) as well as their variations.

However, the smoothing approach has received less attention in solving this problem. Therefore, this thesis presented a smoothing approach to solve the SLAM problem, by the implementation of Rauch-Tung-Striebel smoother. In the beginning, a linearization approach has been applied to the Rauch-Tung-Striebel smoother. This smoother named as extended Rauch-Tung-Striebel smoother (ERTSS). The performances of this smoother is better compared to the standard EKF.

In order to minimize errors in the nonlinear estimation, this thesis utilizes the benefit of an unscented transformation over linearization in the ERTSS. In the unscented transformation, the state distribution is represented using a minimal set of carefully chosen sample points, called sigma points. These sigma points are propagated through the true nonlinear function, without any approximation. In addition, the difficulty of Jacobian matrix calculation, which is used in the EKF, is not required in this method. This transformation method is applied to the Rauch-Tung-Striebel smoother to obtain the unscented Rauch-Tung-Striebel smoother (URTSS). The performance of the URTSS is evaluated and compared to the similar filtering method, the UKF. The result shows that the URTSS gives lower errors in solving the SLAM problem, compared to the errors produced by the UKF.

This thesis also investigates other paradigm of solving a SLAM problem known as a FastSLAM approach. In this framework, the approximation used in the standard FastSLAM is replaced by the unscented transformation, in which it is called the unscented FastSLAM (UFastSLAM). The proposed method is evaluated and its performance is compared to the standard FastSLAM. It is shown that, the UFastSLAM gives better result in solving the SLAM problem. In addition, a new sampling technique, which is called a spherical simplex unscented FastSLAM (SSUFastSLAM), is presented. This new sampling technique uses less number of sigma points, compared to the standard one used in the UFastSLAM. For that reason, the computational cost is reduced without giving any effect on its performance, which is proved by the simulation result.

Contents

Page

List of Figures

List of Tables

Table Page

Chapter 1

Introduction

1.1 Introduction

The ability of a mobile robot to determine its location in space is a fundamental competence for autonomous navigation. Knowledge of self location, and the location of other places of interest in the world, is the basic foundation on which all high level navigation operations are built. It enables strategic path planning for tasks such as goal reaching, region coverage, exploration and obstacle avoidance, and makes following of these planned trajectories possible. Without a notion of location, a robot is limited to reactive behavior based solely on local stimuli and is incapable of planning actions beyond its immediate sensing range.

The most basic form of localization is dead reckoning, which is simply estimation of the vehicle pose by integrating estimation of its motion for example the used of inertial sensing and encoder based odometry. The problem with dead reckoning is that each change in pose estimation includes a component of error and these errors accumulate as part of the integration process. Thus, uncertainty in the pose estimation increases monotonically with time although the sensor and the motion model accuracy are improved. At last, uncertainly in robot pose estimation is so large, then it give no useful information to solve the localization problem. For this reason, dead reckoning is an insufficient mechanism to implement for long term localization. However, dead reckoning does retain usefulness as an auxiliary information source in conjunction with map based localization.

Pose estimation with bounded uncertainty is only possible through the availability of absolute rather than incremental pose measurements. The source of absolute information can be found in a map of the environment which is defined by the locations of distinct landmarks. Thus, if the robot has an ability to sense its surroundings, it can obtain absolute pose estimates by registering sensed information with the map. The problem with a priori map based localization is that, the environment should be explored in advance and the landmark locations also have been recognized before the robot can begin to navigate autonomously.

The motivation for SLAM is to overcome the need for a priori maps as a mechanism for bounded pose uncertainty, and to enable map construction that is extensible and adaptive to environmental change. SLAM is performed by storing landmarks in a map as they are observed by the robot sensors, using the robot pose estimate to determine the landmark locations, while at the same time, using these landmarks to improve the robot pose estimate. As the landmarks are repeatedly reobserved, their locations become increasingly certain and the map converges, eventually acquiring the rigidity of an priori map.

The SLAM ideal is to allow immediate navigation capabilities in completely unknown environments, so that a robot might be placed in a new environment and left to explore and map the environment without human intervention. Alternatively a human operator might drive the robot through an environment once for example as a training phase where the operator might demonstrate trajectories and way points. This phase could accumulate sufficient knowledge for the robot to travel competently within the region in the future. For real environments, training by a human operator is currently a more practicable scenario than autonomous exploration especially when operating in the presence of hidden structures. For example, transparent objects and objects outside of the view plane, are invisible to a robot with only a range laser sensor and will not be avoided. Other structures such as stairwells present a potential danger unless the robot equipped with a specific recognition of vertical free space.

The first SLAM algorithm to comprise an explicit and consistent representation of uncertainty, and therefore provide qualification of map convergence, was presented in [1]. This method, referred to here as stochastic SLAM, remains the basic foundation of practically all subsequent SLAM proposals using the landmark based map framework. Early experimental verification of the algorithm with laser [2] and sonar [3] sensors demonstrated its utility in relatively small-scale indoor environments. However, increasing in a computation cost and storage size when involved building a large map, give direct implementation of this algorithm intractable. As a result, more feasible adaptations of stochastic SLAM have been proposed such as removing redundant map features [4], developing efficient estimation procedures [5, 6], and dividing the environment into a network of submaps [7, 8, 6].

The development of a feasible and reliable SLAM system is dependent on the non linearity in the process of constructing a map and localizing from it, computational complexity and storage requirements and data association in recognize on correct correspondence between data obtained from the robot sensor and the data currently stored in the map. The objective of this thesis is to develop a non linear solutions in the process where the robot build the map of it environment and then localize itself on the map.

1.2 Simultaneous Localization and Mapping

The probabilistic simultaneous localization and mapping (SLAM) problem occurred at the 1986 IEEE Robotics and Automation Conference held in San Francisco, California when a probabilistic methods were only just beginning to be introduced into both robotics and artificial intelligence (AI). A number of researchers had been looking at applying estimation theoretic-methods to solve the mapping and localization problems. These led the discussion to recognize that the consistent probabilistic mapping was a fundamental problem in robotics with major conceptual and computational issues that need to be addressed. Work by Smith and Cheesman [1] and Durrant-Whyte [9] established a statistical basis for describing relationships between landmarks and manipulating geometric uncertainty.

A key element of this work was to show that there must be a high degree of correlation between the estimation of different landmarks location in a map and that, indeed, these correlations would grow with successive observations. At the same time Ayache and Faugeras [10] were undertaking early work in visual navigation, Crowley [11] and Chatila and Laumond [12] were working in sonar based navigation of mobile robots using Kalman filter type algorithms. These two strands of research had much in common and resulted soon after in the landmark paper by Smith et al. [13]. This paper showed that as a mobile robot moves through an unknown environment taking relative observations of landmarks, the estimation of these landmarks are all necessarily correlated with each other because of the common error in the estimated vehicle location [3]. It is found that a consistent full solution to the combined localization and mapping problem would require a joint state composed of the vehicle pose and every landmark position that need to be updated by each landmark observation. In turn, this would require the estimator to employ a huge state of vector (on the order of the number of landmarks maintained in the map) with computation scaling as the square of the number of landmarks.

However, this work did not look at the convergence properties of the map or its steady state behavior. Indeed, it was assumed at the time that the estimated map errors would not converge and would instead exhibit a random walk behavior with unbounded error growth. The conceptual breakthrough came with the realization that the combined mapping and localization problem, once formulated as a single estimation problem, was actually convergent. Although most researchers had tried to minimize the correlations between landmarks, it was recognized that the more these correlations grew, the better of the solution.

The structure of the SLAM problem, the convergence result was first presented in a mobile robotics survey paper presented by Durrant-Whyte et al. [14]. The essential theory on convergence and many of the initial results were developed by Csorba [15]. Several groups already working on mapping and localization, notably at the ACFR at Sydney [16, 17], the Massachusetts Institute of Technology [18], Zaragoza [19, 20], and others [7, 21], began working in earnest on SLAM or also called concurrent mapping and localization (CML) in indoor, outdoor, and subsea environments. At this time, the work focused on improving computational efficiency and addressing issues in the data association, or loop closure. The 1999 International Symposium on Robotics Research (ISRR'99) [22] was an important meeting point where the first SLAM session was held and where a degree of convergence between the Kalman filter based SLAM methods and the probabilistic localization and mapping methods introduced by Thrun [23] was achieved. The 2000 IEEE International Conference on Robotics and Automation (ICRA) Workshop on SLAM focused on issues such as algorithmic complexity, data association, and implementation challenges.

1.3 Thesis Contribution

The principal contributions of this thesis are as follows:

- Introduced of extended Rauch-Tung-Striebel smoother to solve a SLAM problem. This estimator used linearization technique to linearize the nonlinear system and based on forward backward smoother equation.
- In order to improved nonlinear estimation, we then implement unscented transformation to Rauch-Tung-Striebel smoother where a set of carefully selected point known as sigma points directly propagate nonlinear system. This smoother named as an unscented Rauch-Tung-Striebel smoother. This smoother then used to solve a SLAM problem.

- In order to apply the unscented transformation more deeply, we applied it to the other paradigm used to solve a SLAM problem known as a FastSLAM. The unscented Kalman filter has been used in conjunction with standard particle filter in the FastSLAM framework.
- We also introduced other sampling technique used in unscented transformation aim to reduced number of sigma points used in the unscented FastSLAM in previous contribution. This techniques known as a Spherical Simplex Unscented FastSLAM (SSUFastSLAM).

1.4 Thesis Overview

This thesis is organized as follows:

Chapter 2 presents the necessary background to this thesis by discussing the common optimal filtering and smoothing. The history and formulation of the discrete time optimal filtering and smoothing as recursive Bayesian inference are presented in the beginning. Then the fundamental knowledge of the classical Kalman filter, extended Kalman filter, unscented Kalman filter and particle filter are discussed in detail. The smoothing methods based on Rauch-Tung-Striebel are explained in detail including the basic Rauch-Tung-Striebel smoother, extended Rauch-Tung-Striebel smoother, and unscented Rauch-Tung-Striebel smoother.

Chapter 3 gives detailed explanations about the SLAM problem. This chapter starts by describing the basic navigational map such as occupancy grid, feature map and topological map which is used in the autonomous mobile robot. Then, the derivation of the SLAM problem in probabilistic term is discussed in detail. Afterwards, this derivation then is used as basic knowledge to solve the SLAM problem. Detailed implementation of the common method in solving the SLAM problem, named the extended Kalman filter (EKF) based SLAM is also presented in detail in this chapter. The structure of the EKF based SLAM including the state augmentation between the robot state and map, the prediction and updating stage, and the augmentation of new feature in the map are described step by step. Then, the derivation of motion model of a nonholonomic vehicle with Ackerman steering model used in this thesis also is demonstrated in detail. In addition, we introduce the extended Rauch-Tung-Striebel smoother (ERTSS) based SLAM and discuss its performance over the standard EKF based SLAM.

Chapter 4 describes another method to solve the SLAM problem known as an unscented transformation used by the unscented Kalman filter (UKF) based SLAM. Instead of using the approximation of nonlinearities in a SLAM problem by the EKF based SLAM, this unscented transformation uses the deterministic sigma points to directly propagate the nonlinear system. The structure of the UKF based SLAM is described in the similar way as in the previous chapter but replacing the linearization with the unscented transformation to estimate the nonlinear term. The comparison between the EKF based SLAM and the UKF based SLAM is also presented. In addition, novel approach known as unscented Rauch-Tung-Striebel smoother (URTSS) is presented to solve the SLAM problem. The filter type algorithms. These two strands of research had much in common and resulted soon after in the landmark paper by Smith et al. [13]. This paper showed that as a mobile robot moves through an unknown environment taking relative observations of landmarks, the estimation of these landmarks are all necessarily correlated with each other because of the common error in the estimated vehicle location [3]. It is found that a consistent full solution to the combined localization and mapping problem would require a joint state composed of the vehicle pose and every landmark position that need to be updated by each landmark observation. In turn, this would require the estimator to employ a huge state of vector (on the order of the number of landmarks maintained in the map) with computation scaling as the square of the number of landmarks.

However, this work did not look at the convergence properties of the map or its steady state behavior. Indeed, it was assumed at the time that the estimated map errors would not converge and would instead exhibit a random walk behavior with unbounded error growth. The conceptual breakthrough came with the realization that the combined mapping and localization problem, once formulated as a single estimation problem, was actually convergent. Although most researchers had tried to minimize the correlations between landmarks, it was recognized that the more these correlations grew, the better of the solution.

The structure of the SLAM problem, the convergence result was first presented in a mobile robotics survey paper presented by Durrant-Whyte et al. [14]. The essential theory on convergence and many of the initial results were developed by Csorba [15]. Several groups already working on mapping and localization, notably at the ACFR at Sydney [16, 17], the Massachusetts Institute of Technology [18], Zaragoza [19, 20], and others [7, 21], began working in earnest on SLAM or also called concurrent mapping and localization (CML) in indoor, outdoor, and subsea environments. At this time, the work focused on improving computational efficiency and addressing issues in the data association, or loop closure. The 1999 International Symposium on Robotics Research (ISRR'99) [22] was an important meeting point where the first SLAM session was held and where a degree of convergence between the Kalman filter based SLAM methods and the probabilistic localization and mapping methods introduced by Thrun [23] was achieved. The 2000 IEEE International Conference on Robotics and Automation (ICRA) Workshop on SLAM focused on issues such as algorithmic complexity, data association, and implementation challenges.

1.3 Thesis Contribution

The principal contributions of this thesis are as follows:

- Introduced of extended Rauch-Tung-Striebel smoother to solve a SLAM problem. This estimator used linearization technique to linearize the nonlinear system and based on forward backward smoother equation.
- In order to improved nonlinear estimation, we then implement unscented transformation to Rauch-Tung-Striebel smoother where a set of carefully selected point known as sigma points directly propagate nonlinear system. This smoother named as an unscented Rauch-Tung-Striebel smoother. This smoother then used to solve a SLAM problem.

- In order to apply the unscented transformation more deeply, we applied it to the othei paradigm used to solve a SLAM problem known as a FastSLAM. The unscented Kalman filter has been used in conjunction with standard particle filter in the FastSLAM framework.
- We also introduced other sampling technique used in unscented transformation aim to reduced number of sigma points used in the unscented FastSLAM in previous contribution. This techniques known as a Spherical Simplex Unscented FastSLAM (SSUFastSLAM).

1.4 Thesis Overview

This thesis is organized as follows:

Chapter 2 presents the necessary background to this thesis by discussing the common optimal filtering and smoothing. The history and formulation of the discrete time optimal filtering and smoothing as recursive Bayesian inference are presented in the beginning. Then the fundamental knowledge of the classical Kalman filter, extended Kalman filter, unscented Kalman filter and particle filter are discussed in detail. The smoothing methods based on Rauch-Tung-Striebel are explained in detail including the basic Rauch-Tung-Striebel smoother, extended Rauch-Tung-Striebel smoother, and unscented Rauch-Tung-Striebel smoother.

Chapter 3 gives detailed explanations about the SLAM problem. This chapter starts by describing the basic navigational map such as occupancy grid, feature map and topological map which is used in the autonomous mobile robot. Then, the derivation of the SLAM problem in probabilistic term is discussed in detail. Afterwards, this derivation then is used as basic knowledge to solve the SLAM problem. Detailed implementation of the common method in solving the SLAM problem, named the extended Kalman filter (EKF) based SLAM is also presented in detail in this chapter. The structure of the EKF based SLAM including the state augmentation between the robot state and map, the prediction and updating stage, and the augmentation of new feature in the map are described step by step. Then, the derivation of motion model of a nonholonomic vehicle with Ackerman steering model used in this thesis also is demonstrated in detail. In addition, we introduce the extended Rauch-Tung-Striebel smoother (ERTSS) based SLAM and discuss its performance over the standard EKF based SLAM.

Chapter 4 describes another method to solve the SLAM problem known as an unscented transformation used by the unscented Kalman filter (UKF) based SLAM. Instead of using the approximation of nonlinearities in a SLAM problem by the EKF based SLAM, this unscented transformation uses the deterministic sigma points to directly propagate the nonlinear system. The structure of the UKF based SLAM is described in the similar way as in the previous chapter but replacing the linearization with the unscented transformation to estimate the nonlinear term. The comparison between the EKF based SLAM and the UKF based SLAM is also presented. In addition, novel approach known as unscented Rauch-Tung-Striebel smoother (URTSS) is presented to solve the SLAM problem. The performance of this proposed method is evaluated by comparing it with the UKF.

Chapter 5 discusses another paradigm in solving a SLAM problem known as the FastSLAM. The FastSLAM decomposes the SLAM problem into robot localization and a collection of landmark estimation problem. The standard FastSLAM framework, which consists of robot estimation, features state estimation and the important weight calculation, is presented in detail. In this chapter, the linearization approaches used in the standard FastSLAM is replaced by the unscented transformation. The performance between the proposed method and the standard FastSLAM will be also discussed. In addition to the standard sampling technique of the unscented transformation, we present the spherical simplex unscented transformation as a new sampling method used in the FastSLAM framework. The evaluation among these approaches is also presented in this chapter.

Chapter 6 presents the conclusions and suggestions for the future directions of this work.

Chapter 2

Optimal Filtering and Smoothing

2.1 Introduction

Optimal filtering refers to the methodology that can be used for estimating the state of time varying system, which is indirectly observed through noisy measurements. The stat of the system refers to the collection of dynamic variables such as position, velocities an accelerations or orientation and rotational motion parameters, which describe the physic state of the system. The noise in the measurements means that the uncertainty in measurements data due to disturbance in the system model, the measurement model as we as environment changes. The time evolution of the state is modeled as a dynamic system which is perturbed by a certain process noise. This noise is used for modeling the uncertainties in the system dynamics and in most cases the system is not truly stochastic, but th stochasticity is only used for representing the model uncertainties.

In this chapter, the history and formulation of the discrete time optimal filtering an smoothing as recursive Bayesian inference are presented in the beginning. Then fundamental knowledge of the classical Kalman filter, extended Kalman filter, unscented Kalman filter and particle filter are discussed in detail. The smoothing methods based on Rauch-Tung-Striebel are explained in detail including the basic Rauch-Tung-Striebel smoother, extende Rauch-Tung-Striebel smoother, and unscented Rauch-Tung-Striebel smoother. Summar of this chapter is presented in the last section in this chapter.

2.1.1 History of Bayesian optimal filtering

The history of optimal filtering starts from the Wiener filter [24], which is a spectral domai solution to the problem of least squares optimal filtering of stationary Gaussian signals. Th Wiener filter is still important in communication applications [25], digital signal processin [26] and image processing [27]. However, the Wiener filter can only be applied to stationar signals and it is often mathematically demanding. Due to these limitations, this filter ca only be applied to a simple low dimensional filtering problems.

The success of optimal linear filtering in engineering applications is begin due t the seminal article by Kalman [28], which describes the recursive solution to the optim discrete time linear filtering problem. The reason to the success is that the Kalman filter ca be understood and applied with very much lighter mathematical machinery than the Wien filter. Also, despite its mathematical simplicity, the Kalman filter or more specifically th Kalman Bucy filter [29] contains the Wiener filter as its limiting special case.

In the early stages of its history, the Kalman filter was soon discovered to belong the class of Bayesian estimators [30, 31, 32]. An historical detail shows that while Kalman and Bucy were formulating the linear theory in the United States, Stratonovich was doing the pioneering work on the Bayesian probabilistic approach in Russia [32, 33].

Because of Kalman filter's useful connection to the theory and history of stochastic optimal control, we use the Bayesian filtering problem approaches from the Kalman filtering point of view. Although the original derivation of the Kalman filter was based on the least squares approach, the same equations can be derived from the pure probabilistic Bayesian analysis and is was covered in the classical book of Jazwinski [32] as well as in the book of Bar-Shalom et al. [34] recently. The discussion on optimal filtering and smoothing as Bayesian Inference is presented in next subsection.

2.1.2 Optimal filtering and smoothing as Bayesian inference

The optimal Bayesian filtering is considered as the statistical inversion problems, where the unknown time series quantity vector $(x_1, x_2, \ldots, x_T)$ which is observed through noisy measurements $(y_1, y_2, \ldots, y_T)$. It can be illustrated as shown in the Fig. 2.1.

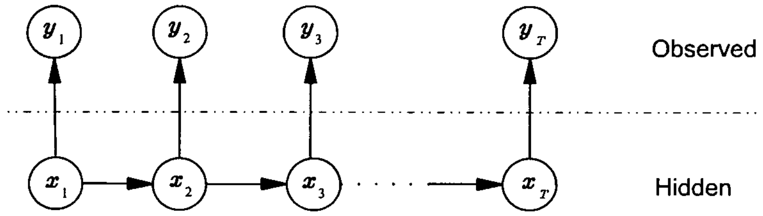

Figure 2.1: In discrete-time filtering a sequence of hidden states x_t is indirectly observed through noisy measurements y_t.

The purpose of the statistical inversion is to estimate the hidden states $\{x_1, \ldots, x_T\}$ given the observed measurements $\{y_1, \ldots, y_T\}$, which means that in Bayesian sense by [35, 36] all we have to do is to compute the joint posterior distribution of all the states given all the measurements. This can be done by straightforward application of the Bayes' rule as follows:

$$p(x_1, \ldots, x_T \mid y_1, \ldots, y_T) = \frac{p(y_1, \ldots, y_T \mid x_1, \ldots, x_T)\, p(x_1, \ldots, x_T)}{p(y_1, \ldots, y_T)} \tag{2.1}$$

where $p(x_1, \ldots, x_T)$ is the prior defined by the dynamic model, $p(y_1, \ldots, y_T \mid x_1, \ldots, x_T)$ is the likelihood model for the measurements, and $p(y_1, \ldots, y_T)$ is the normalization constant which defined as

$$p(y_1, \ldots, y_T) = \int p(y_1, \ldots, y_T \mid x_1, \ldots, x_T)\, p(x_1, \ldots, x_T)\, d(x_1, \ldots, x_T). \tag{2.2}$$

Unfortunately, this full posterior formulation has a serious drawback that each time when a new measurement obtained, the full posterior distribution would have to be recomputed. As a number of time steps increases, the dimensionality of the full posterior distribution also increases. This means that the computational complexity of a single time step also increases. Thus after a sufficient number of time steps the computations will become intractable, independently of available computational resources. Without additional information or approximations, there is no other solution to overcome this problem in the full posterior computation.

However, this problem only arises when we want to compute the full posterior distribution of the states at each time step. The computation can be reliable if we are considering the marginal distribution of the states by restricting the class of dynamic models into probabilistic Markov sequences. In order to achieve that, the model for the states and the measurements are assumed to be in the following condition:

- Initial distribution specifies the prior distribution $p(\boldsymbol{x}_0)$ of the hidden state $\boldsymbol{x}_0$ at initial time step $t = 0$.
- The system dynamics and its uncertainties are modeled by dynamical model as a Markov sequence, defined in terms of the transition distribution $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$.
- The measurement of $\boldsymbol{y}_t$ and its relationship to the current state $\boldsymbol{x}_t$ is modeled using measurement model. This dependence is modeled by specifying the distribution of the measurement given the state $p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$.

On the contrary of the full joint distribution of the states at all time steps which is computationally very inefficient and unnecessary in real time applications, the optimal Bayesian filtering are considered the following marginal distributions:

- Filtering distributions are the marginal distributions of the current state $\boldsymbol{x}_k$ given the previous measurements $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t\}$:

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_t), \qquad t = 1, \ldots, T. \tag{2.3}$$

- Prediction distributions are the marginal distributions of the future states, n steps after the current time step:

$$p(\boldsymbol{x}_{t+n} \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_t), \qquad t = 1, \ldots, T, \qquad n = 1, 2, \ldots. \tag{2.4}$$

- Smoothing distributions are the marginal distributions of the states $\boldsymbol{x}_t$ given a certain interval $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T\}$ of measurements with $t < T$:

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_T), \qquad t = 1, \ldots, T \tag{2.5}$$

There exists a few classes of filtering and smoothing problems which have closed form solutions such as Kalman filter (KF) and Rauch-Tung-Striebel smoother (RTSS). But because the Bayesian optimal filtering and smoothing equations are generally computationally intractable, many kinds of numerical approximation methods such as extended Kalman filter (EKF), extended Rauch-Tung-Striebel smoother (ERTSS), unscented Kalman filter (UKF), unscented Rauch-Tung-Striebel smoother (URTSS), sequential Monte Carlo, unscented particle filter (UPF), Rao-Blackwellized particle filters and smoothers have been developed. To facilitate the discussion, the basic knowledge of filtering and smoothing will be addressed separately in the next section.

2.2 Optimal Filtering

In this section, a state space models can be represented as probabilistic nonlinear form consisting a sequence of conditional probability distributions as:

$$\begin{aligned} \boldsymbol{x}_t &\sim p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) \\ \boldsymbol{y}_t &\sim p(\boldsymbol{y}_t \mid \boldsymbol{x}_t), \end{aligned} \tag{2.6}$$

where $\boldsymbol{x}_t \in \mathcal{R}^n$ is the state, $\boldsymbol{y}_t \in \mathcal{R}^m$ is the measurement at time step t. $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$ is the dynamic model distribution which describes the stochastic dynamics of the system. It can be a probability density, a counting measure or a combination of them depending on the state $\boldsymbol{x}_t$ whether it is continuous, discrete or hybrid. $p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$ is the measurement model , which is the probability distribution of measurements given the state. The model is assumed to be Markovian, which means that it has the following two properties:

1. The states $\{\boldsymbol{x}_t : t = 1, \dots, T\}$ form a Markov sequence or Markov chain. This Markov property means that the current state $\boldsymbol{x}_t$ given the previous state $\boldsymbol{x}_{t-1}$ is independent of any state that has happened before the time step $t-1$. It can be formulate as follows:

$$p(\boldsymbol{x}_t \mid \boldsymbol{x}_{1:t-1}, \boldsymbol{y}_{1:t-1}) = p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}). \tag{2.7}$$

2. The current measurement $\boldsymbol{y}_t$ given the current state $\boldsymbol{x}_t$ is conditionally independent of the previous measurement and state histories as :

$$p(\boldsymbol{y}_t \mid \boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t-1}) = p(\boldsymbol{y}_t \mid \boldsymbol{x}_t). \tag{2.8}$$

With the Markovian assumption and the filtering model as in Eq. (2.6), the joint prior distribution of the states $(\boldsymbol{x}_0, \dots, \boldsymbol{x}_T)$, and the joint likelihood of the measurements $(\boldsymbol{y}_0, \dots, \boldsymbol{y}_T)$ can be describe as follows:

$$p(\boldsymbol{x}_0, \dots, \boldsymbol{x}_T) = p(\boldsymbol{x}_0) \prod_{t=1}^{T} p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}), \tag{2.9}$$

$$p(\boldsymbol{y}_1, \dots, \boldsymbol{y}_T \mid \boldsymbol{x}_0, \dots, \boldsymbol{x}_T) = \prod_{t=1}^{T} p(\boldsymbol{y}_t \mid \boldsymbol{x}_t). \tag{2.10}$$

For a given T we could simply compute the posterior distribution of the states by using Bayes' rule as follows:

$$\begin{aligned} p(\boldsymbol{x}_0, \dots, \boldsymbol{x}_T \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T) &= \frac{p(\boldsymbol{y}_1, \dots, \boldsymbol{y}_T \mid \boldsymbol{x}_0, \dots, \boldsymbol{x}_T)\, p(\boldsymbol{x}_0, \dots, \boldsymbol{x}_T)}{p(\boldsymbol{y}_1, \dots, \boldsymbol{y}_T)} \\ &\propto p(\boldsymbol{y}_1, \dots, \boldsymbol{y}_T \mid \boldsymbol{x}_0, \dots, \boldsymbol{x}_T)\, p(\boldsymbol{x}_0, \dots, \boldsymbol{x}_T). \end{aligned} \tag{2.11}$$

However, this kind of explicit usage of the full Bayes' rule is not feasible in real time applications because the amount of computations per time step increases as new observations arrive. Thus, this way will limit to the application with small datasets, then at some point of time the computations would become intractable. To cope with real time data we need to have an algorithm which does constant amount of computations per time step. As discussed in previous section, filtering distributions, prediction distributions and smoothing distributions can be computed recursively such that only constant amount of computations is done on each time step. For this reason we are concentrate to the above mentioned distributions instead of considering the full posterior computation.

The purpose of optimal filtering is to compute the marginal posterior distribution (also known as prior distribution) of the state $\boldsymbol{x}_t$ at each time step t given the history of the measurements up to the time step t:

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}). \tag{2.12}$$

The fundamental equation of the Bayesian filtering theory are given as follows. The recursive equations for computing the predicted distribution $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})$ and the filtering distribution $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})$ at the time step t are given by the following Bayesian filtering equations:

1. *Initialization.* The recursion starts from the prior distribution $p(\boldsymbol{x}_0)$.
2. *Prediction.* The predictive distribution of the state $\boldsymbol{x}_t$ on time step t given the dynamic model can be computed by the Chapman-Kolmogorov equation

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) = \int p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\, p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) d\boldsymbol{x}_{t-1}. \tag{2.13}$$

3. *Update.* Given the measurement $\boldsymbol{y}_t$ at time step t the posterior distribution of the state $\boldsymbol{x}_t$ can be computed by Bayes' rule

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \frac{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})}{\eta_t}, \tag{2.14}$$

where the normalization constant η_t is given as

$$\eta_t = \int p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) d\boldsymbol{x}_t. \tag{2.15}$$

Note that, if some of the components of the state are discrete, the corresponding integrals are then replaced with summations.

Proof. The joint distribution of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ given $\boldsymbol{y}_{1:t-1}$ can be computed as

$$\begin{aligned} p(\boldsymbol{x}_t, \boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) &= p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}, \boldsymbol{y}_{1:t-1})\, p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) \\ &= p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\, p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}), \end{aligned} \tag{2.16}$$

where the disappearance of the measurement history $\boldsymbol{y}_{1:t-1}$ is due to the Markov property of the sequence $\{\boldsymbol{x}_t, t = 1, 2, \ldots\}$. The marginal distribution of $\boldsymbol{x}_t$ given $\boldsymbol{y}_{1:t-1}$ can b obtained by integrating the distribution in Eq. (2.16) over x_{t-1}, which gives the Chapman-Kolmogorov equation

$$p(x_t \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1})\, p(x_{t-1} \mid y_{1:t-1})\, dx_{t-1}. \tag{2.17}$$

If x_{t-1} is discrete, then the above integral is replaced with summation over x_{t-1}. The distribution of x_t given y_t and $y_{1:t-1}$, that is, given $y_{1:t}$ can be computed by Bayes' rule

$$\begin{aligned} p(x_t \mid y_{1:t}) &= \frac{p(y_t \mid x_t, y_{1:t-1})\, p(x_t \mid y_{1:t-1})}{\eta_t} \\ &= \frac{p(y_t \mid x_t)\, p(x_t \mid y_{1:t-1})}{\eta_t}, \end{aligned} \tag{2.18}$$

where the normalization constant η_t is given by Eq. (2.15). The disappearance of the measurement history $y_{1:t-1}$ in Eq. (2.18) is due to the conditional independence of y_t of the measurement history, given x_t.

□

2.2.1 Kalman filter

The Kalman filter [28] is the closed form solution to the optimal filtering equations of the discrete time filtering model, where the dynamic and measurements models are linear Gaussian:

$$\begin{aligned} x_t &= A_{t-1} x_{t-1} + q_{t-1} \\ y_t &= H_k x_t + r_t, \end{aligned} \tag{2.19}$$

where $x_t \in \mathcal{R}^n$ is the state, $y_t \in \mathcal{R}^m$ is the measurement, $q_{t-1} \sim \mathcal{N}(0, Q_{t-1})$ is the process noise, $r_t \sim \mathcal{N}(0, R_t)$ is the measurement noise and the prior distribution is Gaussian $x_0 \sim \mathcal{N}(\hat{x}_0, P_0)$. The matrix A_{t-1} is the transition matrix of the dynamic model and H_t is the measurement model matrix. The Eq. (2.19) then can be written in the probabilistic terms as

$$\begin{aligned} p(x_t \mid x_{t-1}) &= \mathcal{N}(x_t \mid A_{t-1} x_{t-1}, Q_{t-1}) \\ p(y_t \mid x_t) &= \mathcal{N}(y_t \mid H_t x_t, R_t). \end{aligned} \tag{2.20}$$

The optimal filtering equations for the linear filtering model in Eq. (2.19) can be evaluated in closed form and the resulting distributions are Gaussian as following equation:

$$\begin{aligned} p(x_t \mid y_{1:t-1}) &= \mathcal{N}(x_t \mid \hat{x}_t^-, P_t^-) \\ p(y_t \mid y_{1:t-1}) &= \mathcal{N}(y_t \mid H_t \hat{x}_t^-, S_t) \\ p(x_t \mid y_{1:t}) &= \mathcal{N}(x_t \mid \hat{x}_t, P_t). \end{aligned} \tag{2.21}$$

where the $\hat{x}_t^-$ and P_t^- are the predicted mean and the predicted covariance of the state x in time step t, respectively. S_t is the predicted covariance of the measurement y_t evaluated at time step t. The parameters of the distributions then can be computed with the following Kalman filter prediction and update steps:

1. The prediction step is

$$\begin{aligned} \hat{\boldsymbol{x}}_t^- &= \boldsymbol{A}_{t-1}\hat{\boldsymbol{x}}_{t-1} \\ \boldsymbol{P}_t^- &= \boldsymbol{A}_{t-1}\boldsymbol{P}_{t-1}\boldsymbol{A}_{t-1}^T + \boldsymbol{Q}_{k-1}. \end{aligned} \tag{2.22}$$

2. The measurement update step is

$$\begin{aligned} \boldsymbol{S}_t &= \boldsymbol{H}_t\boldsymbol{P}_t^-\boldsymbol{H}_t^- + \boldsymbol{R}_k \\ \boldsymbol{K}_t &= \boldsymbol{P}_t^-\boldsymbol{H}_t^-\boldsymbol{S}_t^{-1} \\ \hat{\boldsymbol{x}}_t &= \hat{\boldsymbol{x}}_t^- + \boldsymbol{K}_t(\boldsymbol{y}_t - \boldsymbol{H}_t\hat{\boldsymbol{x}}_t^-) \\ \boldsymbol{P}_t &= \boldsymbol{P}_t^- - \boldsymbol{K}_t\boldsymbol{S}_t\boldsymbol{K}_t^-. \end{aligned} \tag{2.23}$$

The initial state has a given Gaussian prior distribution $\boldsymbol{x}_0 \sim \mathcal{N}(\hat{\boldsymbol{x}}_0, \boldsymbol{P}_0)$, which also defines the initial mean and covariance. The Kalman filter equations can be derived as follows:

1. By Lemma A.1 in Appendix A, the joint distribution of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ given $\boldsymbol{y}_{1:t-1}$ is

$$\begin{aligned} p(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) &= p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) \\ &= \mathcal{N}(\boldsymbol{x}_t \mid \boldsymbol{A}_{t-1}\boldsymbol{x}_{t-1}, \boldsymbol{Q}_{t-1})\mathcal{N}(\boldsymbol{x}_{t-1} \mid \hat{\boldsymbol{x}}_{t-1}, \boldsymbol{P}_{t-1}) \\ &= \mathcal{N}\left(\left[\begin{array}{c} \boldsymbol{x}_{t-1} \\ \boldsymbol{x}_t \end{array}\right] \middle| \hat{\boldsymbol{x}}', \boldsymbol{P}'\right), \end{aligned} \tag{2.24}$$

where

$$\begin{aligned} \hat{\boldsymbol{x}}' &= \left(\begin{array}{c} \hat{\boldsymbol{x}}_{t-1} \\ \boldsymbol{A}_{t-1}\hat{\boldsymbol{x}}_{t-1} \end{array}\right), \\ \boldsymbol{P}' &= \left(\begin{array}{cc} \boldsymbol{P}_{t-1} & \boldsymbol{P}_{t-1}\boldsymbol{A}_{t-1}^T \\ \boldsymbol{A}_{t-1}\boldsymbol{P}_{t-1} & \boldsymbol{A}_{t-1}\boldsymbol{P}_{t-1}\boldsymbol{A}_{t-1}^T + \boldsymbol{Q}_{t-1} \end{array}\right). \end{aligned} \tag{2.25}$$

and the marginal distribution of $\boldsymbol{x}_t$ is by Lemma A.2 also in Appendix A page 89

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) = \mathcal{N}(\boldsymbol{x}_t \mid \hat{\boldsymbol{x}}_t^-, \boldsymbol{P}_t^-), \tag{2.26}$$

where

$$\begin{aligned} \hat{\boldsymbol{x}}_t^- &= \boldsymbol{A}_{t-1}\hat{\boldsymbol{x}}_{t-1} \\ \boldsymbol{P}_t^- &= \boldsymbol{A}_{t-1}\boldsymbol{P}_{t-1}\boldsymbol{A}_{t-1}^T + \boldsymbol{Q}_{t-1}. \end{aligned} \tag{2.27}$$

2. Again, using Lemma A.1, the joint distribution of $\boldsymbol{y}_t$ and $\boldsymbol{x}_t$ is

$$\begin{aligned} p(\boldsymbol{x}_t, \boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1}) &= p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) \\ &= \mathcal{N}(\boldsymbol{y}_t \mid \boldsymbol{H}_t\boldsymbol{x}_t, \boldsymbol{R}_t)\mathcal{N}(\boldsymbol{x}_t \mid \hat{\boldsymbol{x}}_t^-, \boldsymbol{P}_t^-) \\ &= \mathcal{N}\left(\left[\begin{array}{c} \boldsymbol{x}_t \\ \boldsymbol{y}_t \end{array}\right] \middle| \hat{\boldsymbol{x}}'', \boldsymbol{P}''\right), \end{aligned} \tag{2.28}$$

where

$$\hat{\boldsymbol{x}}'' = \left(\begin{array}{c} \hat{\boldsymbol{x}}_t^- \\ \boldsymbol{H}_t\hat{\boldsymbol{x}}_t^- \end{array}\right), \quad \boldsymbol{P}'' = \left(\begin{array}{cc} \boldsymbol{P}_t^- & \boldsymbol{P}_t^-\boldsymbol{H}_t^T \\ \boldsymbol{H}_t\boldsymbol{P}_t^- & \boldsymbol{H}_t\boldsymbol{P}_t^-\boldsymbol{H}_t^T + \boldsymbol{R}_t \end{array}\right). \tag{2.29}$$

3. By Lemma A.2, the conditional distribution of $\boldsymbol{x}_t$ is

$$\begin{aligned} p(\boldsymbol{x}_t \mid \boldsymbol{y}_t, \boldsymbol{y}_{1:t-1}) &= p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) \\ &= \mathcal{N}(\boldsymbol{x}_t \mid \hat{\boldsymbol{x}}_t, \boldsymbol{P}_t), \end{aligned} \tag{2.30}$$

where

$$\begin{aligned} \hat{\boldsymbol{x}}_t &= \hat{\boldsymbol{x}}_t^- + \boldsymbol{P}_t^- \boldsymbol{H}_t^T (\boldsymbol{H}_t \boldsymbol{P}_t^- \boldsymbol{H}_t^T + \boldsymbol{R}_t)^{-1} \left[\boldsymbol{y}_t - \boldsymbol{H}_t \hat{\boldsymbol{x}}_t^-\right] \\ \boldsymbol{P}_t &= \boldsymbol{P}_t^- - \boldsymbol{P}_t^- \boldsymbol{H}_t^T (\boldsymbol{H}_t \boldsymbol{P}_t^- \boldsymbol{H}_t^T + \boldsymbol{R}_t)^{-1} \boldsymbol{H}_t \boldsymbol{P}_t^- \end{aligned} \tag{2.31}$$

which can be also written in the form of Eq. (2.23).

2.2.2 Extended Kalman filter

The extended Kalman filter (EKF) [32, 34, 37, 38] is an extension of the Kalman filter to nonlinear optimal filtering problems. We will explain the basic linearization of nonlinear transformation used in the EKF at the beginning followed by the derivation of the EKF. Let consider the following transformation of a Gaussian random variable $\boldsymbol{x}$ into another random variable $\boldsymbol{y}$

$$\begin{aligned} \boldsymbol{x} &\sim \mathcal{N}(\hat{\boldsymbol{x}}, \boldsymbol{P}), \\ \boldsymbol{y} &= \boldsymbol{g}(\boldsymbol{x}) \end{aligned} \tag{2.32}$$

where $\boldsymbol{x} \in \mathcal{R}^n$, $\boldsymbol{y} \in \mathcal{R}^m$, and $\boldsymbol{g} : \mathcal{R}^n \mapsto \mathcal{R}^m$ is a general nonlinear function. $\hat{\boldsymbol{x}}$ and $\boldsymbol{P}$ are the mean and the covariances of state $\boldsymbol{x}$, respectively.

In order to approximate the distribution of $\boldsymbol{y}$, a first order Taylor series based Gaussian approximation needs to be performed, but only the first two terms of it were used in the linear approximation. Let $\boldsymbol{x} = \hat{\boldsymbol{x}} + \delta\boldsymbol{x}$ where $\delta\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{P})$, and the linear approximation of $\boldsymbol{x}$ can be written as

$$\boldsymbol{g}(\boldsymbol{x}) \approx \boldsymbol{g}(\hat{\boldsymbol{x}}) + \nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x} \tag{2.33}$$

where $\nabla \boldsymbol{g}_{\boldsymbol{x}}$ is the Jacobian matrix of $\boldsymbol{g}$,

$$\nabla \boldsymbol{g}_{\boldsymbol{x}} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}=\hat{\boldsymbol{x}})}. \tag{2.34}$$

Computing the expected value with referring to $\boldsymbol{x}$ gives:

$$\begin{aligned} \mathrm{E}\left[\boldsymbol{g}(\boldsymbol{x})\right] &\approx \mathrm{E}\left[\boldsymbol{g}(\hat{\boldsymbol{x}}) + \nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x}\right] \\ &= \boldsymbol{g}(\hat{\boldsymbol{x}}) + \nabla \boldsymbol{g}_{\boldsymbol{x}} \mathrm{E}\left[\delta\boldsymbol{x}\right] \\ &= \boldsymbol{g}(\hat{\boldsymbol{x}}). \end{aligned} \tag{2.35}$$

The covariance can be then approximated as

$$\begin{aligned} &\mathrm{E}\left[(\boldsymbol{g}(\boldsymbol{x}) - \mathrm{E}\left[\boldsymbol{g}(\boldsymbol{x})\right])(\boldsymbol{g}(\boldsymbol{x}) - \mathrm{E}\left[\boldsymbol{g}(\boldsymbol{x})\right])^T\right] \\ &\approx \mathrm{E}\left[(\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\hat{\boldsymbol{x}}))(\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\hat{\boldsymbol{x}}))^T\right] \\ &\approx \mathrm{E}\left[(\boldsymbol{g}(\hat{\boldsymbol{x}}) + \nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x}) - \boldsymbol{g}(\hat{\boldsymbol{x}}))(\boldsymbol{g}(\hat{\boldsymbol{x}}) + \nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x}) - \boldsymbol{g}(\hat{\boldsymbol{x}}))^T\right] \\ &= \mathrm{E}\left[(\nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x})(\nabla \boldsymbol{g}_{\boldsymbol{x}} \delta\boldsymbol{x})^T\right] \\ &= \nabla \boldsymbol{g}_{\boldsymbol{x}} \mathrm{E}\left[\delta\boldsymbol{x} \delta\boldsymbol{x}^T\right] \nabla \boldsymbol{g}_{\boldsymbol{x}}^T \\ &= \nabla \boldsymbol{g}_{\boldsymbol{x}} \boldsymbol{P} \nabla \boldsymbol{g}_{\boldsymbol{x}}^T. \end{aligned} \tag{2.36}$$

The approximation of joint covariance between the variables $\boldsymbol{x}$ and $\boldsymbol{y}$ can be achieved by considering the augmented transformation

$$\tilde{\boldsymbol{g}}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{g}(\boldsymbol{x}) \end{pmatrix}, \tag{2.37}$$

and the resulting mean and covariance as:

$$\begin{aligned} \mathrm{E}\left[\tilde{\boldsymbol{g}}(\boldsymbol{x})\right] &\approx \begin{pmatrix} \hat{\boldsymbol{x}} \\ \boldsymbol{g}(\hat{\boldsymbol{x}}) \end{pmatrix} \\ \mathrm{Cov}\left[\tilde{\boldsymbol{g}}(\boldsymbol{x})\right] &\approx \begin{pmatrix} \boldsymbol{I} \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} \end{pmatrix} \boldsymbol{P} \begin{pmatrix} \boldsymbol{I} \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} \end{pmatrix}^T \\ &= \begin{pmatrix} \boldsymbol{P} & \boldsymbol{P} \nabla \boldsymbol{g}_{\boldsymbol{x}}^T \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} \boldsymbol{P} & \nabla \boldsymbol{g}_{\boldsymbol{x}} \boldsymbol{P} \nabla \boldsymbol{g}_{\boldsymbol{x}}^T \end{pmatrix}. \end{aligned} \tag{2.38}$$

In order to derive of the extended Kalman filter equations, the Eq. (2.32) now can be written in a generalize form as follows:

$$\begin{aligned} \boldsymbol{x} &\sim \mathcal{N}(\hat{\boldsymbol{x}}, \boldsymbol{P}) \\ \boldsymbol{q} &\sim \mathcal{N}(0, \boldsymbol{Q}) \\ \boldsymbol{y} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{q}), \end{aligned} \tag{2.39}$$

where $\boldsymbol{x}$ and $\boldsymbol{q}$ are assumed to be uncorrelated random variables. The mean and covariance can be now computed by substituting the augmented vector $(\boldsymbol{x}, \boldsymbol{q})$ to the vector $\boldsymbol{x}$ in Eq. (2.38). The joint Jacobian matrix can be then written as $\nabla \boldsymbol{g}_{\boldsymbol{x},\boldsymbol{q}} = (\nabla \boldsymbol{g}_{\boldsymbol{x}} \ \nabla \boldsymbol{g}_{\boldsymbol{q}})$. Here $\nabla \boldsymbol{g}_{\boldsymbol{q}}$ is the Jacobian matrix of $\boldsymbol{g}(\cdot)$ with respect to $\boldsymbol{q}$ and both the Jacobian matrices are evaluated at $\boldsymbol{x} = \hat{\boldsymbol{x}}$ and $\boldsymbol{q} = \boldsymbol{0}$. The approximations to the mean and covariance of the augmented transform as in Eq. (2.38) are then given as follows:

$$\begin{aligned} \mathrm{E}\left[\tilde{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{q})\right] &\approx \boldsymbol{g}(\hat{\boldsymbol{x}}, 0) \\ \mathrm{Cov}\left[\tilde{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{q})\right] &\approx \begin{pmatrix} \boldsymbol{I} & 0 \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} & \nabla \boldsymbol{g}_{\boldsymbol{q}} \end{pmatrix} \begin{pmatrix} \boldsymbol{P} & 0 \\ 0 & \boldsymbol{Q} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{I} & 0 \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} & \nabla \boldsymbol{g}_{\boldsymbol{q}} \end{pmatrix}^T \\ &= \begin{pmatrix} \boldsymbol{P} & \boldsymbol{P} \nabla \boldsymbol{g}_{\boldsymbol{x}}^T \\ \nabla \boldsymbol{g}_{\boldsymbol{x}} \boldsymbol{P} & \nabla \boldsymbol{g}_{\boldsymbol{x}} \boldsymbol{P} \nabla \boldsymbol{g}_{\boldsymbol{x}}^T + \nabla \boldsymbol{g}_{\boldsymbol{q}} \boldsymbol{Q} \nabla \boldsymbol{g}_{\boldsymbol{q}}^T \end{pmatrix}. \end{aligned} \tag{2.40}$$

In practical implementation, the state space consists of process models and measurement modes that can be can written in a general form as

$$\begin{aligned} \boldsymbol{x}_t &= \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{q}_{t-1}) \\ \boldsymbol{y}_t &= \boldsymbol{h}(\boldsymbol{x}_t, \boldsymbol{r}_t), \end{aligned} \tag{2.41}$$

where $\boldsymbol{q}_{t-1} \sim \mathcal{N}(0, \boldsymbol{Q}_{t-1})$ and $\boldsymbol{r}_t \sim \mathcal{N}(0, \boldsymbol{R}_t)$ are the Gaussian process and measurement noises, respectively. $\boldsymbol{f}(\cdot)$ is the dynamic model function and $\boldsymbol{h}(\cdot)$ is the measurement model function. The functions $\boldsymbol{f}$ and $\boldsymbol{h}$ can also depend on the step number t, but for notational convenience, this dependence has not been explicitly denoted. The idea of the EKF is to form the Gaussian approximations

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) \approx \mathcal{N}(\boldsymbol{x}_t \mid \hat{\boldsymbol{x}}_t, \boldsymbol{P}_t), \tag{2.42}$$

to the filtering densities. In the EKF this is done by utilizing the linear approximations to the nonlinearities by the following steps:

1. The joint distribution of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ is non-Gaussian, but we can form a Gaussian approximation to it by applying the approximation to the function $\boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{q}_{t-1})$, which results in the Gaussian approximation

$$p(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}) \approx \mathcal{N}\left(\left[\begin{array}{c} \boldsymbol{x}_{t-1} \\ \boldsymbol{x}_t \end{array}\right] \middle| \hat{\boldsymbol{x}}', \boldsymbol{P}'\right), \tag{2.43}$$

where

$$\hat{\boldsymbol{x}}' = \begin{pmatrix} \hat{\boldsymbol{x}}_{t-1} \\ \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1}) \end{pmatrix}$$
$$\boldsymbol{P}' = \begin{pmatrix} \boldsymbol{P}_{t-1} & \boldsymbol{P}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{x}}^T \\ (\boldsymbol{P}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{x}}^T)^T & \nabla\boldsymbol{f}_{\boldsymbol{x}}\boldsymbol{P}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{x}}^T + \nabla\boldsymbol{f}_{\boldsymbol{q}}\boldsymbol{Q}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{q}}^T \end{pmatrix} \tag{2.44}$$

where $\nabla\boldsymbol{f}_{\boldsymbol{x}}$ is the Jacobian matrix of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$, evaluated at $\boldsymbol{x} = \hat{\boldsymbol{x}}$ and $\boldsymbol{q} = \boldsymbol{0}$ with elements

$$\nabla\boldsymbol{f}_{\boldsymbol{x}} = \left.\frac{\partial\boldsymbol{f}_{\boldsymbol{x},\boldsymbol{q}}}{(\partial\boldsymbol{x})}\right|_{(\boldsymbol{x}=\hat{\boldsymbol{x}},\boldsymbol{q}=\boldsymbol{0})} \tag{2.45}$$

and $\nabla\boldsymbol{f}_{\boldsymbol{q}}$ is the corresponding Jacobian matrix with respect to $\boldsymbol{q}$:

$$\nabla\boldsymbol{f}_{\boldsymbol{q}} = \left.\frac{\partial\boldsymbol{f}_{\boldsymbol{x},\boldsymbol{q}}}{(\partial\boldsymbol{q})}\right|_{(\boldsymbol{x}=\hat{\boldsymbol{x}},\boldsymbol{q}=\boldsymbol{0})}. \tag{2.46}$$

From Eq. (2.26), the marginal mean and covariance of $\boldsymbol{x}_t$ are

$$\begin{aligned} \hat{\boldsymbol{x}}_t^- &= \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1}, \boldsymbol{0}) \\ \boldsymbol{P}_t^- &= \nabla\boldsymbol{f}_{\boldsymbol{x}}\boldsymbol{P}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{x}}^T + \nabla\boldsymbol{f}_{\boldsymbol{q}}\boldsymbol{Q}_{t-1}\nabla\boldsymbol{f}_{\boldsymbol{q}}^T. \end{aligned} \tag{2.47}$$

2. The joint distribution of $\boldsymbol{y}_t$ and $\boldsymbol{x}_t$ is also non-Gaussian, but we can again approximate it by applying approximation to the function $\boldsymbol{h}(\boldsymbol{x}_t, \boldsymbol{r}_t)$. We get the approximation

$$p(\boldsymbol{x}_t, \boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1}) \approx \mathcal{N}\left(\left[\begin{array}{c} \boldsymbol{x}_t \\ \boldsymbol{y}_t \end{array}\right] \middle| \hat{\boldsymbol{x}}'', \boldsymbol{P}''\right), \tag{2.48}$$

where

$$\hat{\boldsymbol{x}}'' = \begin{pmatrix} \hat{\boldsymbol{x}}_t^- \\ \boldsymbol{f}(\hat{\boldsymbol{x}}_t^-) \end{pmatrix}$$
$$\boldsymbol{P}'' = \begin{pmatrix} \boldsymbol{P}_t^- & \boldsymbol{P}_t^-\nabla\boldsymbol{h}_{\boldsymbol{x}}^T \\ (\boldsymbol{P}_t^-\nabla\boldsymbol{h}_{\boldsymbol{x}}^T)^T & \nabla\boldsymbol{h}_{\boldsymbol{x}}\boldsymbol{P}_t^-\nabla\boldsymbol{h}_{\boldsymbol{x}}^T + \nabla\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{R}_t\nabla\boldsymbol{h}_{\boldsymbol{r}}^T \end{pmatrix} \tag{2.49}$$