

October 30, 2023

Extended Kalman Filter Simultaneous Localization and Mapping for the FIRST Robotics Competition

Henry LeCompte
FRC Team 2383, The Ninjineers

Contents

1	Introduction	3
1.1	What is the FIRST Robotics Competition?	3
1.2	What is Simultaneous Localization and Mapping (SLAM)	3
1.3	Who should read this paper	3
2	Understanding the Problem	4
2.1	What is the problem?	4
2.2	Why do we need to localize the robot?	4
2.3	Why do we need to map the field?	4
3	Goals of the Project	5
3.1	Goals of the Project	5
3.2	Goals of the Paper	5
4	Background	6
4.1	Kalman Filters	6
5	Kalman Filter Design	7
5.1	State Vector	7
6	Implementation	8
6.1	Motion Model	8
6.2	Measurement Model	8
6.3	Putting it all together	8
6.4	Code Walkthrough	8
6.5	Integration with FRC	8

1 Introduction

1.1 What is the FIRST Robotics Competition?

The FIRST Robotics Competition is an annual high school robotics competition where students compete on an alliance of 3 robots in 3v3 matches culminating in the World Championship in Houston, Texas. The competition is designed to inspire students to pursue careers in STEM fields and to teach students valuable skills such as teamwork, leadership, and problem solving.

1.2 What is Simultaneous Localization and Mapping (SLAM)

Simultaneous Localization and Mapping or SLAM is one of the biggest problems in robotics. The problem comes about from the fact that it is not possible to create a map of an environment without knowing the location of the robot and it is not possible to know the location of the robot without a map of the environment. SLAM attempts to solve these problems by iteratively improving the map and the location estimate of the robot at the same time.

1.3 Who should read this paper

This paper was designed for High School students currently competing in First Robotics Competition (FRC) although nothing in this paper is specific to this competition. These paper assumes only a basic understanding of linear algebra and calculus and will try to motivate intuition for the math used. I myself am a high school student currently taking AP Calc AB and I have tried to write this paper in a way that I would have wanted to read it when I was first learning about SLAM.

2 Understanding the Problem

2.1 What is the problem?

FRC used to be mostly a mechanical competition but as computational power has increased and FIRST has added AprilTags to the field it is more important than ever to know where the robot is on the field. This is where SLAM comes in. SLAM is a way to estimate the location of the robot on the field in a way that does not assume the robot has an accurate map of the field.

2.2 Why do we need to localize the robot?

FRC games have been getting more complicated with smaller scoring zones and more complicated game pieces. The games at the same time have been getting more dynamic, take for example the 2022 game, Rapid React, this game featured large blue and red balls and a large hub to score in the center of the field. The balls were constantly moving around the field and the robots goals where to pick up and shoot the balls as fast as possible. But, how does a robot shoot a ball at a stationary target while it is moving. This requires the robot to know how it is moving in relation to the target which is something that localization can provide. Many teams this year did not have a robust localization system and as a result were only able to shoot accurately while the robot was stationary.

2.3 Why do we need to map the field?

Mapping seems like it should be an easy to solve problem: FIRST gives an official map of the field. The issue is that while FIRST does give a map of the field they also give a plus or minus one inch tolerance for most of the field elements. This is an issue when you want to accurately localize your robot on the field. This is where SLAM comes in. SLAM allows us to create a map of the field that is accurate to within a few millimeters.

3 Goals of the Project

We can split to goals of the project into two sections: the goals of the project and the goals of the paper. The goals of the project are to create a robust SLAM system that can be used in current and future FRC games. The goals of the paper are to teach the reader about SLAM and how to implement it in their own robot.

3.1 Goals of the Project

The goals of the project are to create a robust SLAM system that can be used in current and future FRC games. The system should be able to be used in any FRC game with minimal changes to the code. The system should also be able to be used in any other robotics competition with minimal changes to the code.

3.2 Goals of the Paper

The goals of the paper are to teach the reader about SLAM and how to implement it in their own robot. The paper should be able to be read by any high school student with a basic understanding of linear algebra and calculus. The paper should also be able to be read by any high school student with a basic understanding of programming.

4 Background

4.1 Kalman Filters

4.1.1 What is a Kalman Filter?

A kalman filter is an implementation of the recursive Bayesian filter. The recursive Bayesian filter is a way to estimate the state of a system given a series of measurements. The kalman filter is a specific implementation of the recursive Bayesian filter that assumes that the state of the system is a gaussian distribution and that the measurements are linearly related to the state of the system. The kalman filter is a very powerful tool that can be used in many different applications. The kalman filter is used in everything from self driving cars to the Apollo space program.

These assumptions of gaussian distributions and linear relationship between measurements and the state of the system make the kalman filter a very powerful tool in these cases but these assumptions are not valid in most mobile robotics applications. The linearity assumption is violated by the fact that in order to convert the robots odometry measurements into a movement on the field we need to use trigonometry to rotate the measurements into the field frame. The gaussian assumption in reality is violated by the fact that in some cases there are multiple possible locations for the robot to be in. The violation of the gaussian assumption in practice is not a big issue but the violation of the linearity assumption is a big issue. This is where the extended kalman filter comes in.

4.1.2 What is an Extended Kalman Filter?

An extended kalman filter is an extension of a kalman filter that allows for non linear relationships between measurements and the state of the system. The extended kalman filter does this by linearizing the non linear relationship between measurements and the states of the system through a first order taylor series expansion. This linearization is done at the current state and is recomputed at every time step. This linearization is not perfect and therefore the extended kalman filter is not an optimal solution but it is a good approximation.

5 Kalman Filter Design

5.1 State Vector

The first thing needed to design and implement our kalman filter is the state vector. In the SLAM problem the state vector needs to encompass the full location and rotation of the robot and all the landmarks in the field coordinate frame. There are many different options here for how to represent the state vector but in this paper we will be using a state vector in the form:

$$\mu = \begin{bmatrix} x_0 & y_0 & z_0 & rw_0 & rx_0 & ry_0 & rz_0 & \dots & x_n & y_n & z_n & rw_n & rx_n & ry_n & rz_n \end{bmatrix}^T \quad (1)$$

Where x_0 , y_0 , and z_0 are the robots x, y, and z position in the field coordinate frame. rw_0 , rx_0 , ry_0 , and rz_0 are the robots rotation in the field coordinate frame represented as a quaternion. The landmark positions and rotations then follow in the same format. This state vector is a compact representation of the full map and localization of the field which is needed for the SLAM problem.

6 Implementation

6.1 Motion Model

6.2 Measurement Model

6.3 Putting it all together

6.4 Code Walkthrough

6.5 Integration with FRC