**Basic Kalman Filter Theory**

**Technical Note**

Document XXX

Version: Draft

Authors: Mark Pedley and Michael Stanley

Date: xx Aug 2014



Table of Contents

[1 Introduction 4](#_Toc395019525)

[2 Mathematical Lemmas 5](#_Toc395019526)

[2.1 Lemma 1 5](#_Toc395019527)

[2.2 Lemma 2 5](#_Toc395019528)

[2.3 Lemma 3 6](#_Toc395019529)

[3 Kalman Filter Derivation 8](#_Toc395019530)

[3.1 Process Model 8](#_Toc395019531)

[3.2 Derivation 8](#_Toc395019532)

[3.3 Standard Kalman Equations 11](#_Toc395019533)

**Glossary**

The linear prediction or state matrix at sample.

The measurement matrix relating to at sample.

Expectation operator

The Kalman filter gain at sample

The *a priori* covariance matrix of the linear prediction (*a priori*) error at sample .

The *a posteriori* covariance matrix of the Kalman (*a posteriori*) error at sample .

The covariance matrix of the additive noise on the process

The covariance matrix of the additive noise on the measured process

Variance operator

The additive noise on the measured process at sample

The additive noise on the process of interest at sample

The state vector at time sample of the process of interest

The linear prediction (*a priori*) estimate of the process at sample .

The Kalman filter (*a posteriori*) estimate of the process at sample .

The error in the linear prediction (a priori) estimate of the process .

The error in the *a posteriori* Kalman filter estimate of the process .

The measured process at sample .

The Kronecker delta function. for and zero otherwise.

## Introduction

This document describes the assumptions underlying the basic Kalman filter and derives the standard Kalman equations. It is intended as a primer that should be read before tackling the documentation for the more specialized indirect complementary Kalman filter used for the fusion of accelerometer, magnetometer and gyroscope data.

Section 2 derives some mathematical results used in the derivation. The derivation itself is in section 3.

## Mathematical Lemmas

### Lemma 1

The trace of the sum of two matrices equals the sum of the individual traces.

*Proof*

Eq 2.1.1

### Lemma 2

The derivative with respect to of the trace of the matrix product equals .

*Proof*

Eq 2.2.1

Assuming that the matrix has dimensions and the matrix has dimensions , then has dimensions .

The element of matrix has value:

Eq 2.2.2

Substituting gives:

Eq 2.2.3

By inspection:

Eq 2.2.4

Substituting back gives:

Eq 2.2.5

### Lemma 3

The derivative with respect to of the trace of the matrix product equals .

*Proof*

Eq 2.3.1

If the matrix has dimensions then the matrix must be square with dimensions for the product to exist. The product is always square with dimensions .

The element of the matrix has value:

Eq 2.3.2

The element of matrix has value:

Eq 2.3.3

The trace of matrix ***D*** has value:

Eq 2.3.4

The derivative of with respect to is then:

Eq 2.3.5

Eq 2.3.6

Eq 2.3.7

If is also symmetric then:

Eq 2.3.8

## Kalman Filter Derivation

### Process Model

The Kalman filter models the vector process of interest as linear and recursive:

Eq 3.1.1

If has degrees of freedom then is an linear prediction matrix (possibly time varying but assumed known) and is an noise vector.

The process is assumed to be not directly measurable and must be estimated from a process which can be measured. is modeled as being linearly related to with additive noise .

Eq 3.1.2

is an vector, is an matrix (possibly time varying but assumed known) and is an noise vector.

The noise vectors and are assumed to be zero mean white processes:

Eq 3.1.3

Eq 3.1.4

Eq 3.1.5

Eq 3.1.6

By definition, covariance matrices are symmetric.

Eq 3.1.7

### Derivation

The objective of the Kalman filter is to compute an unbiased *a posterori* estimate   
 of the underlying process from i) extrapolation from the previous iteration's *a posteriori* estimate and ii) from the current measurement :

Eq 3.2.1

The time-varying Kalman gain matrices and define the relative weightingsgiven to the previous iteration's Kalman filter estimate and to the current measurement . If the measurements have low noise then a higher weighting will be given to the term compared to the extrapolated component and vice versa. The Kalman filter is therefore a time varying recursive filter.

***Unbiased estimate constraint (determines )***

For to be an unbiased estimate of , the expectation value of the *a posteriori* Kalman filter error must be zero:

Eq 3.2.2

Subtracting from equation 3.2.1gives:

Eq 3.2.3

Substituting equation 3.1.2 for gives:

Eq 3.2.4

Substituting for from equation 3.1.1 and re-arranging gives:

Eq 3.2.5

Eq 3.2.6

Taking the expectation value of equation 3.2.6 and applying the unbiased estimate constraint gives:

Eq 3.2.7

Since the noise vectors and are zero mean and uncorrelated with the Kalman matrices for the same iteration, it follows that:

Eq 3.2.8

With the additional assumption that the process is independent of the Kalman matrices at iteration :

Eq 3.2.9

Since is not, in general, a zero mean process:

Eq 3.2.10

Eliminating in equation 3.2.1 gives:

Eq 3.2.11

***A priori estimate***

The *a priori* Kalman filter estimate is defined as resulting from the application of the linear prediction matrix to the previous iteration's *a posteriori* estimate :

**Kalman equation 1** Eq 3.2.12

***Definition of a posteriori estimate***

Substituting the a priori estimate into equation 3.2.11 gives:

**Kalman equation 4** Eq 3.2.13

An equivalent form is:

Eq 3.2.14

***as a function of***

The *a priori and a posteriori* error covariance matrices and are defined as:

Eq 3.2.15

Eq 3.2.16

Substituting the definitions of and into equation 3.2.15 gives:

Eq 3.2.17

Eq 3.2.18

Eq 3.2.19

**Kalman equation 2** Eq 3.2.20

***Minimum error covariance constraint (determines )***

The Kalman gain matrix minimizes the *a posteriori* error variance via the trace of the *a posteriori* error covariance matrix :

Eq 3.2.21

Substituting equation 2.1.2 for into equation 3.2.11gives a relation between the *a posteriori* and *a priori* errors:

Eq 3.2.22

Eq 3.2.23

Eq 3.2.24

Substituting this result into the definition of the *a posteriori* covariance matrix gives:

Eq 3.2.25

Eq 3.2.26

Eq 3.2.27

Eq 3.2.28

The Kalman filter gain is that which minimizes the trace of the *a posteriori* error covariance matrix :

Eq 3.2.29

The term has no dependence ongiving:

Eq 3.2.30

Since the trace of a transposed matrix equals the trace of the original matrix and using equation 2.2.5 gives:

Eq 3.2.31

The third term can be simplified using equations 2.3.7 and 2.3.8 exploiting the fact that the covariance matrix is symmetric:

Eq 3.2.32

The final term can be simplified also using equations 2.3.7 and 2.3.8 to give:

Eq 3.2.33

Substituting back into equation 2.2.29 gives the optimal Kalman filter gain matrix :

Eq 3.2.34

Eq 3.2.35

**Kalman equation 3** Eq 3.2.36

***as a function of***

Rearranging equation 3.2.35 gives:

Eq 3.2.37

Substituting equation 3.2.37 into equation 3.2.27 gives:

Eq 3.2.38

**Kalman equation 5** Eq 3.2.39

This completes the derivation of the standard Kalman filter equations.

### Standard Kalman Equations

***Kalman equation 1***

The linear prediction (*a priori*) estimate is made by applying the linear prediction matrix to the previous sample's Kalman (*a posteriori*) filter estimate .

Eq 3.3.1

***Kalman equation 2***

The *a priori* (linear extrapolation) error covariance matrix is then updated using the model matrix and the noise matrix .

Eq 3.3.2

Kalman equations 2 and 5 can be combined to give a recursive update of without explicit calculation of the *a posteriori* error covariance matrix in Kalman equation 5:

Eq 3.3.3

***Kalman equation 3***

The Kalman filter gain matrix is updated:

Eq 3.3.4

***Kalman equation 4***

The Kalman filter (*a posteriori*) estimate is computed from the current *a priori* estimate and the current measurement :

Eq 3.3.5

***Kalman equation 5***

The *a posteriori* Kalman error covariance matrix is updated ready for the next iteration. This equation can be skipped if is updated recursively in terms of itself as in equation 3.3.3.

Eq 3.3.6