Fundamentals of Kalman Filtering and Estimation in Aerospace Engineering

Christopher D'Souza

chris.dsouza@nasa.gov

NASA / Johnson Space Center
Houston, Texas

20 March 2013

Outline

Why Estimate?

- We estimate without even being conscious of it
- Anytime you walk down the hallway, you are estimating, your eyes and ears are the sensors and your brain is the computer
- In its essence, estimation is nothing more than taking noisy sensor data, filtering the noise, and producing the 'best' state of the vehicle

What Do We Estimate?

- As NASA engineers, we estimate a variety of things
  - Position, Velocity, Attitude
  - Mass
  - Temperature
  - Sensor parameters (biases)
- These quantities are usually referred to as the ‘states’ of the system
- We use a variety of sensors to accomplish this task
  - Inertial Measurement Units (IMUs)
  - GPS Receivers (GPSRs)
  - LIDARs
  - Cameras
- These sensors are used to determine the states of the system

A Brief History of Estimation

- Estimation has its origins in the work of Gauss and his innovation called 'Least Squares' Estimation
  - He was interested in computing the orbits of asteroids and comets given a set of observations
- Much of the work through WWI centered around extensions to Least Squares Estimation
- In the interval between WWI and WWII, a number of revolutionary contributions were made to sampling and estimation theory
  - Norbert Weiner and the Weiner Filter
  - Claude Shannon and Sampling Theory
- Much of the work in the first half of the Twentieth Century focused on analog circuitry and the frequency domain

Modern Estimation and Rudolf Kalman

- Everything changed with the confluence of two events:
  - The Cold War and the Space Race
  - The Advent of the Digital Computer and Semiconductors
- A new paradigm was introduced: State Space Analysis
  - Linear Systems and Modern Control Theory
    - Estimation Theory
    - Optimization Theory
- Rudolf Kalman proposes a new approach to linear systems
  - Controllability and Observability

Rudolf Kalman and His Filter

- In 1960 Kalman wrote a paper in an obscure ASME journal entitled "A New Approach to Linear Filtering and Prediction Problems" which might have died on the vine, except:
  - In 1961, Stanley Schmidt of NASA Ames read the paper and invited Kalman to give a seminar at Ames
  - Schmidt recognized the importance of this new theory and applied it to the problem of on-board navigation of a lunar vehicle – after all this was the beginning of Apollo
  - This became known as the 'Kalman Filter'
- Kalman's paper was rather obtuse in its nomenclature and mathematics
  - It took Schmidt's exposition to show that this filter could be easily mechanized and applied to a 'real' problem
- The Kalman Filter became the basis for the on-board navigation filter on the Apollo CSM and LM

Types of Estimation

- There are basically two types of estimation: batch and sequential
- Batch Estimation
  - When sets of measurements taken over a period of time are 'batched' and processed together to estimate the state of a vehicle at a given epoch
  - This is usually the case in a ground navigation processor
- Sequential Estimation
  - When measurements are processed as they are taken and the state of the vehicle is updated as the measurements are processed
  - This is done in an on-board navigation system

Types of Sensors

- Inertial Measurement Units (IMUs)
- GPS Recievers
- Magnetometers
- Optical Sensors
  - Visible Cameras
  - IR Cameras
  - LIDARs (Scanning and Flash)
- RF sensors
  - Radars (S-band and C-band)
  - Range and Range-rate from Comm
- Altimeters
- Doppler Velocimeters

Why do we care about this probability stuff?

"Information: the negative reciprocal value of probability ."

Claude Shannon

Concepts from Probability Theory

- A **random variable** is one whose 'value' is subject to variations due to chance (randomness) – it does not have a fixed 'value'; it can be discrete or continuous
  - A coin toss: can be 'heads' or 'tails' – discrete
  - The lifetime of a light bulb – continuous
- A **probability density function** (pdf), $p(x)$, represents the likelihood that x occurs
  - Always non-negative
  - Satisfies

$$\int_{-\infty}^{\infty} p(\xi)\, d\xi = 1$$

- The **expectation operator**, $E[f(x)]$, is defined as

$$E\left[f(x)\right] = \int_{-\infty}^{\infty} f(\xi)\, p(\xi)\, d\xi$$

Concepts from Probability Theory – Mean and Variance

- The **mean** (or first moment) of a random variable x, denoted by $\bar{x}$, is defined as

$$\bar{x} \triangleq E[x] = \int_{-\infty}^{\infty} \xi \, p(\xi) \, d\xi$$

- The **mean-square** of a random variable x, $E\left[x^2\right]$, is defined as

$$E\left[x^2\right] \triangleq \int_{-\infty}^{\infty} \xi^2 \, p(\xi) \, d\xi$$

- The **variance** (or second moment) of a random variable x, denoted by σ_x^2, is

$$\begin{aligned} \sigma_x^2 \triangleq E\left[[x - E(x)]^2\right] &= \int_{-\infty}^{\infty} (\xi - E(\xi))^2 \, p(\xi) \, d\xi \\ &= E\left[x^2\right] - \bar{x}^2 \end{aligned}$$

Concepts from Probability Theory – Mean and Variance of a Vector

- The **mean** of a random n–vector $\mathbf{x}$, $\bar{\mathbf{x}}$, is defined as

$$\bar{\mathbf{x}} \triangleq E[\mathbf{x}] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \xi \, p(\xi) \, d\xi$$

- The **(co-)variance** of random n–vector $\mathbf{x}$, $\mathbf{P_x}$, is defined as

$$\begin{aligned}\mathbf{P_x} \triangleq E\left[[\mathbf{x}-\bar{\mathbf{x}}][\mathbf{x}-\bar{\mathbf{x}}]^T\right] &= \int_{-\infty}^{\infty} \left[\xi - \bar{\xi}\right]\left[\xi - \bar{\xi}\right]^T p(\xi)\, d\xi \\ &= \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1 x_2} & \cdots & \sigma_{x_1 x_n} \\ \sigma_{x_1 x_2} & \sigma_{x_2}^2 & \cdots & \sigma_{x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{x_1 x_n} & \sigma_{x_2 x_n} & \cdots & \sigma_{x_n}^2 \end{bmatrix}\end{aligned}$$

The covariance is geometrically represented by an *error ellipsoid*.

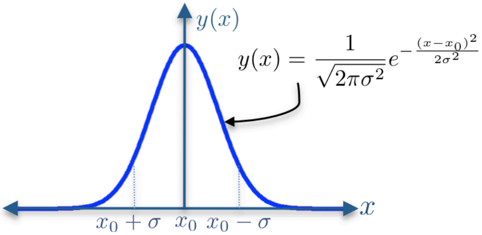
Concepts from Probability Theory –The Gaussian Distribution

- The **Gaussian probability distribution function**, also called the 'Normal distribution'[1] or a 'bell curve', is at the heart of Kalman filtering
- We assume that 'our' random variables have Gaussian pdfs
- We like to work with Gaussians because they are completely characterized by their mean and covariance
  - Linear combinations of Gaussians are Gaussian
- The Gaussian distribution of random n–vector $\mathbf{x}$, with a mean of $\bar{\mathbf{x}}$ and covariance $\mathbf{P_x}$, is defined as

$$p_g(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}\,|\mathbf{P_x}|} e^{-\frac{(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{P_x}^{-1} (\mathbf{x}-\bar{\mathbf{x}})}{2}}$$

[1] Physicist G. Lippman is reported to have said, 'Everyone believes in the normal approximation, the experimenters because they think it is a mathematical theorem, the mathematicians because they think it is an experimental fact.'

Concepts from Probability Theory –The Gaussian Distribution



- We can show that

$$\int_{\mathcal{R}^n} \frac{1}{(2\pi)^{n/2}\,|\mathbf{P_x}|} e^{-\frac{(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{P}_\mathbf{x}^{-1} (\mathbf{x}-\bar{\mathbf{x}})}{2}} d\mathbf{x} = 1$$

- If a random process is generated by a sum of other (non-Gaussian) random processes, then, in the limit, the combined distribution approaches a Gaussian distribution (*The Central Limit Theorem*)

Linear Systems

- A **system** is a mapping from input signals to output signals, written as: $w(t) = L(v(t))$
- A system is **linear** if for all input signals $v(t)$, $v_1(t)$, and $v_2(t)$ and for all scalars α,
  - L is *additive*: $L(v_1(t) + v_2(t)) = L(v_1(t)) + L(v_2(t))$
  - L is *homogeneous*: $L(\alpha v(t)) = \alpha L(v(t))$
- For a system to be linear, if 0 is an input, then 0 is an output:
$$L(0) = L(0 \cdot v(t)) = 0 \cdot L(v(t)) = 0$$
- If the system does not satisfy the above two properties, it is said to be **nonlinear**
- If $L(v(t)) = v(t) + 1$, is this linear?
  - It is not because for $v(t) = 0$, $L(0) = 1 \neq 0$
- Lesson: Some systems may *look* linear but they are not!

Nonlinear Systems and the Linearization Process

- Despite the beauty associated with linear systems, the fact of the matter is that we live in a nonlinear world.
- So, what do we do? We make these nonlinear systems into linear systems by **linearizing**
- This is predicated on a Taylor series approximation which we deploy as follows: Given a nonlinear system of the form: $\dot{\mathbf{X}} = \mathbf{f}(\mathbf{X}, t)$, we linearize about (or expand about) a nominal trajectory, $\mathbf{X}^\star$ (with $\dot{\mathbf{X}}^\star = \mathbf{f}(\mathbf{X}^\star, t)$), as

$$\dot{\mathbf{X}}(t) = f(\mathbf{X}^\star, t) + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{X}}\right)_{\mathbf{X}=\mathbf{X}^\star} (\mathbf{X} - \mathbf{X}^\star) + \cdots$$

Nonlinear Systems and the State Transition Matrix

- If we let $\mathbf{x}(t) = \mathbf{X} - \mathbf{X}^\star$ and let $\mathbf{F}(t) = \left(\frac{\partial \mathbf{f}}{\partial \mathbf{X}}\right)_{\mathbf{X}=\mathbf{X}^\star}$, then we get

$$\dot{\mathbf{x}} = \mathbf{F}(t)\mathbf{x} \quad \text{with} \quad \mathbf{x}(t_0) = \mathbf{x}_0$$

- The solution of this equation is

$$\mathbf{x}(t) = e^{\int_{t_0}^{t} \mathbf{F}(\tau)\, d\tau}\mathbf{x}_0 = \mathbf{\Phi}(t, t_0)\mathbf{x}_0$$

where $\mathbf{\Phi}(t, t_0)$ is the **State Transition Matrix** (STM) which satisfies

$$\dot{\mathbf{\Phi}}(t, t_0) = \mathbf{F}(t)\mathbf{\Phi}(t, t_0) \quad \text{with} \quad \mathbf{\Phi}(t_0, t_0) = \mathbf{I}$$

- The STM can be approximated (for $\mathbf{F}(t) = \mathbf{F} =$ a constant) as

$$\mathbf{\Phi}(t, t_0) = e^{\int_{t_0}^{t} \mathbf{F}(\tau)\, d\tau} = e^{\mathbf{F}(t-t_0)} = \mathbf{I} + \mathbf{F}(t - t_0) + \frac{1}{2}\mathbf{F}^2(t - t_0)^2 + \cdots$$

A Bit More About the State Transition Matrix

The State Transition Matrix (STM) is at the heart of practical Kalman filtering. In its essence it is defined as

$$\mathbf{\Phi}(t, t_0) \triangleq \frac{\partial \mathbf{X}(t)}{\partial \mathbf{X}(t_0)}$$

As the name implies, it is used to 'transition' or **move** perturbations of the state of a nonlinear system from one epoch to another, *i.e.*

$$\mathbf{x}(t) = \mathbf{\Phi}(t, t_0)\mathbf{x}(t_0) \iff \left(\mathbf{X}(t) - \mathbf{X}^\star(t)\right) = \frac{\partial \mathbf{X}(t)}{\partial \mathbf{X}(t_0)}\left(\mathbf{X}(t_0) - \mathbf{X}^\star(t_0)\right)$$

In practical Kalman filtering, we use a first-order approximation[2]

$$\mathbf{\Phi}(t, t_0) \approx \mathbf{I} + \mathbf{F}(t_0)\,(t - t_0) = \mathbf{I} + \left.\frac{\partial \mathbf{f}(\mathbf{X}, t)}{\partial \mathbf{X}}\right|_{\mathbf{X}=\mathbf{X}_0}(t - t_0)$$

[2] In cases of fast dynamics, we can approximate the STM to second-order as:
$$\mathbf{\Phi}(t, t_0) \approx \mathbf{I} + \mathbf{F}(t_0)\,(t - t_0) + \tfrac{1}{2}\mathbf{F}^2(t_0)\,(t - t_0)^2$$

How do we Implement This?

"Never do a calculation unless you already know the answer."

John Archibald Wheeler's First Moral Principle

The Context of Least Squares Estimation

- Least Squares estimation has been a mainstay of engineering and science since Gauss invented it to track Ceres circa 1794
- It has been used extensively for spacecraft state estimation, particularly in **ground-based navigation systems**
- The Apollo program had an extensive ground station network (MSFN/STDN) coupled with sophisticated ground-based batch processors for tracking the CSM and LM
  - A set of measurements (or several sets of measurements) taken over many minutes and over several passes from different ground stations would be 'batched' together to get a spacecraft state at a particular epoch
- Least Squares estimation is predicated on finding a solution which minimizes the square of the errors of the model

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon}$$

The Least Squares Problem

The problem is as follows: given a set of observations, $\mathbf{y}$, subject to measurement errors ($\boldsymbol{\epsilon}$), find the best solution, $\hat{\mathbf{x}}$, which minimizes the errors, i.e.

$$\min J = \frac{1}{2}\boldsymbol{\epsilon}^T\boldsymbol{\epsilon} = \frac{1}{2}\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)^T\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)$$

To do this we take the first derivative of J with respect to $\mathbf{x}$ and set it equal to zero as

$$\frac{\partial J}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}}\left[\frac{1}{2}\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)^T\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)\right]_{\mathbf{x}=\hat{\mathbf{x}}} = -\left(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\right)^T\mathbf{H} = 0$$

Therefore, the optimal solution, $\hat{\mathbf{x}}$, is

$$\hat{\mathbf{x}} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{y}$$

The Weighted Least Squares (WLS) Problem

Suppose now we are given measurements $\mathbf{y}$, whose error has a measurement covariance of $\mathbf{R}$. How can we get the best estimate, $\hat{\mathbf{x}}$ which minmizes the errors weighted by the accuracy of the measurement error ($\mathbf{R}^{-1}$)? The problem can be posed as

$$\min J = \frac{1}{2}\boldsymbol{\epsilon}^T \mathbf{R}^{-1} \boldsymbol{\epsilon} = \frac{1}{2}\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)^T \mathbf{R}^{-1} \left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)$$

Once again, we take the first derivative of J with respect to $\mathbf{x}$ and set it equal to zero as

$$\frac{\partial J}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}}\left[\frac{1}{2}\left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)^T \mathbf{R}^{-1} \left(\mathbf{y} - \mathbf{H}\mathbf{x}\right)\right]_{\mathbf{x}=\hat{\mathbf{x}}} = -\left(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\right)^T \mathbf{R}^{-1}\mathbf{H} = 0$$

Therefore, the optimal solution, $\hat{\mathbf{x}}$, is

$$\hat{\mathbf{x}} = \left(\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{y}$$

The WLS Problem with *A Priori* Information

Suppose we need to find the best estimate of the state, given measurements $\mathbf{y}$, with measurement error covariance $\mathbf{R}$, but we are also given an *a priori* estimate of the state, $\bar{\mathbf{x}}$ with covariance $\bar{\mathbf{P}}$. This problem can be posed as

$$\min J = \frac{1}{2}\left(\mathbf{y}-\mathbf{H}\mathbf{x}\right)^T \mathbf{R}^{-1}\left(\mathbf{y}-\mathbf{H}\mathbf{x}\right) + \frac{1}{2}\left(\bar{\mathbf{x}}-\mathbf{x}\right)^T \bar{\mathbf{P}}^{-1}\left(\bar{\mathbf{x}}-\mathbf{x}\right)$$

As before, we take the first derivative of J with respect to $\mathbf{x}$ and set it equal to zero as

$$\left.\frac{\partial J}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}} = -\left(\mathbf{y}-\mathbf{H}\hat{\mathbf{x}}\right)^T \mathbf{R}^{-1}\mathbf{H} - \left(\bar{\mathbf{x}}-\hat{\mathbf{x}}\right)^T \bar{\mathbf{P}}^{-1} = 0$$

Therefore, the optimal solution, $\hat{\mathbf{x}}$, is

$$\hat{\mathbf{x}} = \left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \bar{\mathbf{P}}^{-1}\right)^{-1}\left[\mathbf{H}^T\mathbf{R}^{-1}\mathbf{y} + \bar{\mathbf{P}}^{-1}\bar{\mathbf{x}}\right]$$

Nonlinear Batch Estimation

In general, the system of interest will be nonlinear of the form

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k, t_k) + \boldsymbol{\epsilon}_k$$

How do we get the best estimate of the state $\mathbf{X}$? Well, first we linearize about a nominal state $\mathbf{X}_k^\star$ (with $\mathbf{x}_k \stackrel{\Delta}{=} \mathbf{X}_k - \mathbf{X}_k^\star$) as

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k^\star + \mathbf{x}_k, t_k) + \boldsymbol{\epsilon}_k = \mathbf{h}(\mathbf{X}_k^\star) + \left.\frac{\partial \mathbf{h}}{\partial \mathbf{X}}\right|_{\mathbf{X}_k = \mathbf{X}_k^\star} \left(\mathbf{X}_k - \mathbf{X}_k^\star\right) + \cdots + \boldsymbol{\epsilon}_k$$

Defining $\tilde{\mathbf{H}}_k \stackrel{\Delta}{=} \left.\frac{\partial \mathbf{h}}{\partial \mathbf{X}}\right|_{\mathbf{X}_k = \mathbf{X}_k^\star}$ we get the following equation

$$\mathbf{y}_k = \tilde{\mathbf{H}}_k \mathbf{x}_k + \boldsymbol{\epsilon}_k$$

Nonlinear Batch Estimation at an Epoch

In batch estimation, we are interested in estimating a state at an epoch, say $\mathbf{X}_0$, with measurements taken after that epoch – say, at t_k. How can we obtain this? Well, we use the state transition matrix as follows

$$\mathbf{X}_k - \mathbf{X}_k^\star = \mathbf{\Phi}(t_k, t_0)\left(\mathbf{X}_k - \mathbf{X}_k^\star\right) \iff \mathbf{x}_k = \mathbf{\Phi}(t_k, t_0)\mathbf{x}_0$$

so that we can map the measurements back to the epoch of interest as

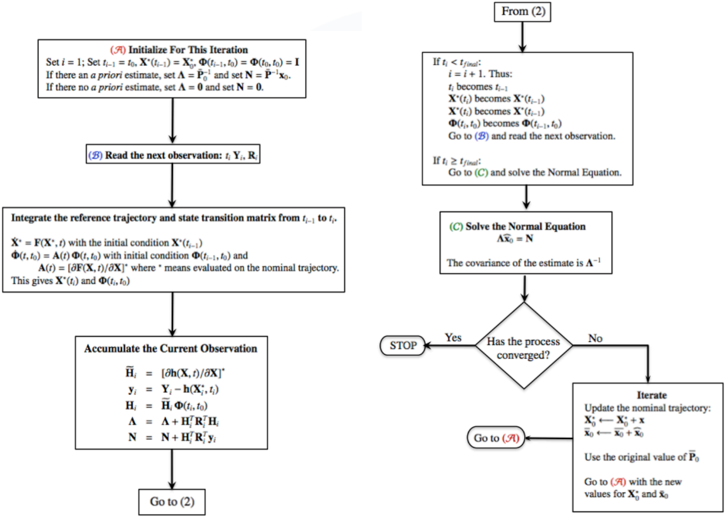
$$\mathbf{y}_k = \tilde{\mathbf{H}}_k \mathbf{\Phi}(t_k, t_0)\mathbf{x}_0 + \boldsymbol{\epsilon}_k = \mathbf{H}_k \mathbf{x}_0 + \boldsymbol{\epsilon}_k$$

The least squares solution (over all the p measurements) is

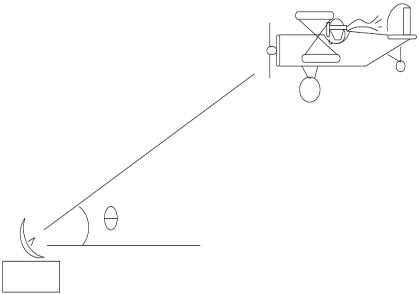
$$\hat{\mathbf{x}}_0 = \left(\sum_{i=1}^{p} \mathbf{H}_i^T \mathbf{R}_i^{-1} \mathbf{H}_i + \bar{\mathbf{P}}_0^{-1}\right)^{-1} \left[\sum_{i=1}^{p} \mathbf{H}_i^T \mathbf{R}_i^{-1} \mathbf{y}_i + \bar{\mathbf{P}}_0^{-1} \bar{\mathbf{x}}_0\right] = \hat{\mathbf{X}}_0 - \mathbf{X}_0^\star$$

This is called the **normal equation**.

The Nonlinear Batch Estimation Algorithm



Batch Filter Example – Aircraft Tracking



Given a ground station tracking an airplane, moving in a straight line at a constant speed, with only bearing measurements, we are interested in knowing the speed of the airplane and its position at the beginning of the tracking pass (x_0, y_0, u_0, v_0). The equations are

$$\begin{aligned} x(t) &= u_0(t - t_0) + x_0 \\ y(t) &= v_0(t - t_0) + y_0 \\ \theta(t) &= \tan^{-1}\left[\frac{y(t)}{x(t)}\right] \end{aligned}$$

Batch Filter Example – Aircraft Tracking (II)

The measurements are:

k	t_k	θ_k (degrees)
0	0	5.4628
1	20	18.9309
2	40	33.4603
3	60	45.1648
4	80	53.7033
5	100	62.3816
6	120	68.1143
7	140	71.9306
8	160	75.7515
9	180	78.5952
10	200	80.8027

The initial guess is

$$\mathbf{X}_0^* = \begin{bmatrix} 985 \\ 105 \\ -1.5 \\ 10 \end{bmatrix}$$

with initial covariance

$$\mathbf{P}_0 = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Batch Filter Example – Aircraft Tracking (III)

After 7 iterations the following results are obtained:

Parameter	Truth	Initial Guess	Converged State
x_0	1000	985	983.5336
y_0	100	105	99.3470
u_0	-3	-1.5	-2.9564
v_0	12	10	11.7763

Lesson: The x–component is not readily observable. But that is not surprising since angles do not provide information along the line-of-sight.

Something to remember

One must watch the convergence of a numerical code as carefully as a father watching his four year old play near a busy road.

J. P. Boyd

The Need for Careful Preparation

"Six months in the lab can save you a day in the library"

Albert Migliori, quoted by J. Maynard
in *Physics Today 49*, 27 (1996)

Stochastic Processes – The Linear First-Order Differential Equation

- Let us look at a first-order differential equation for $x(t)$, given $f(t)$, $g(t)$, $w(t)$ and x_0 as

$$\dot{x}(t) = f(t)x(t) + g(t)w(t) \quad \text{with} \quad x(t_0) = x_0$$

- The solution of this equation is

$$x(t) = e^{\int_{t_0}^{t} f(\tau)d\tau} x_0 + \int_{t_0}^{t} e^{\int_{\xi}^{t} f(\tau)d\tau} g(\xi)w(\xi)d\xi$$

- Suppose now we define $\phi(t, t_0) \triangleq e^{\int_{t_0}^{t} f(\tau)d\tau}$, we can write the above solution as

$$x(t) = \phi(t, t_0)x_0 + \int_{t_0}^{t} \phi(t, \xi)g(\xi)w(\xi)d\xi$$

The Mean of a Linear First-Order Stochastic Process

- Given a first-order *stochastic process*, $\chi(t)$, with constant f and g and white noise, $w(t)$, which is represented as

$$\dot{\chi}(t) = f\chi(t) + g\,w(t) \quad \text{with} \quad \chi(t_0) = \chi_0$$

and the mean and covariance of $w(t)$ expressed as

$$E[w(t)] = 0 \quad \text{and} \quad E[w(t)w(\tau)] = q\,\delta(t-\tau)$$

- The mean of the process, $\bar{\chi}(t)$ is

$$\begin{aligned}
\bar{\chi}(t) = E[\chi(t)] &= e^{\int_{t_0}^{t} f\,d\tau}\bar{\chi}_0 + \int_{t_0}^{t} e^{\int_{\xi}^{t} f\,d\tau} g(\xi)E[w(\xi)]d\xi \\
&= e^{\int_{t_0}^{t} f\,d\tau}\bar{\chi}_0 \\
&= e^{f(t-t_0)}\bar{\chi}_0
\end{aligned}$$

Stochastic Processes – The Mean-Square and Covariance of a Linear First-Order Stochastic Process

- The mean-square of the linear first-order stochastic process, $\chi(t)$ is

$$\begin{aligned} E[\chi^2(t)] &= e^{2f(t-t_0)} E\left[\chi(t_0)\chi(t_0)\right] + \frac{q}{2f}\left[1 - e^{2f(t-t_0)}\right] \\ &= \phi^2(t,t_0) E\left[\chi(t_0)\chi(t_0)\right] + \frac{q}{2f}\left[1 - \phi^2(t,t_0)\right] \end{aligned}$$

- The covariance of $\chi(t)$, $P_{\chi\chi}(t)$, is expressed as

$$\begin{aligned} P_{\chi\chi}(t) &= E\left[(\chi(t) - \bar{\chi}(t))^2\right] = E[\chi^2(t)] - \bar{\chi}^2(t) \\ &= \phi^2(t,t_0) P_{\chi\chi}(t_0) + \frac{q}{2f}\left[1 - \phi^2(t,t_0)\right] \end{aligned}$$

Stochastic Processes – The Vector First-Order Differential Equation

A first-order **vector** differential equation for $\mathbf{x}(t)$, given $\mathbf{x}(t_0)$ and white noise with $E(\mathbf{w}(t)) = \mathbf{0}$, and $E(\mathbf{w}(t)\mathbf{w}(\tau^T)) = \mathbf{Q}\,\delta(t - \tau)$, is

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t)$$

The solution of this equation is

$$\mathbf{x}(t) = \mathbf{\Phi}(t, t_0)\mathbf{x}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}(t, \xi)\mathbf{G}(\xi)\mathbf{w}(\xi)d\xi$$

where $\mathbf{\Phi}(t, t_0)$ satisfies the following equation

$$\dot{\mathbf{\Phi}}(t, t_0) = \mathbf{F}(t)\mathbf{\Phi}(t, t_0), \quad \text{with} \quad \mathbf{\Phi}(t_0, t_0) = \mathbf{I}$$

The Mean and Mean-Square of a Linear, Vector Process

The mean of the stochastic process $\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t)$ is

$$\begin{aligned}\bar{\mathbf{x}}(t) &= E[\mathbf{x}(t)] = \mathbf{\Phi}(t, t_0)E[\mathbf{x}(t_0)] + \int_{t_0}^{t} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)E[\mathbf{w}(\xi)]d\xi \\ &= \mathbf{\Phi}(t, t_0)\bar{\mathbf{x}}(t_0)\end{aligned}$$

The mean-square of the process (with $E[\mathbf{x}(t_0)\mathbf{w}^T(t)] = \mathbf{0}$) is

$$\begin{aligned}E[\mathbf{x}(t)\mathbf{x}^T(t)] &= E\left\{\left[\mathbf{\Phi}(t, t_0)\mathbf{x}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)\mathbf{w}(\xi)d\xi\right]\right. \\ &\qquad \left. \times\left[\mathbf{\Phi}(t, t_0)\mathbf{x}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}(t,\chi)\mathbf{G}(\xi)\mathbf{w}(\chi)d\chi\right]\right\} \\ &= \mathbf{\Phi}(t, t_0)E[\mathbf{x}(t_0)\mathbf{x}^T(t_0)]\mathbf{\Phi}^T(t, t_0) \\ &\qquad + \int_{t_0}^{t} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)\,\mathbf{Q}\,\mathbf{G}^T(\xi)\mathbf{\Phi}^T(t,\xi)\,d\xi\end{aligned}$$

The Covariance of a Linear, Vector Process

The covariance of $\mathbf{x}(t)$, $\mathbf{P_{xx}}(t)$, given $\mathbf{P_{xx}}(t_0)$, is expressed as

$$\begin{aligned}\mathbf{P_{xx}}(t) &= E\left[(\mathbf{x}(t) - \bar{\mathbf{x}}(t))(\mathbf{x}(t) - \bar{\mathbf{x}}(t))^T\right] = E[\mathbf{x}(t)\mathbf{x}^T(t)] - \bar{\mathbf{x}}(t)\bar{\mathbf{x}}^T(t) \\ &= \mathbf{\Phi}(t,t_0)\mathbf{P_{xx}}(t_0)\mathbf{\Phi}^T(t,t_0) \\ &\quad + \int_{t_0}^{t} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)\,\mathbf{Q}\,\mathbf{G}^T(\xi)\mathbf{\Phi}^T(t,\xi)\,d\xi\end{aligned}$$

The differential equation for $\mathbf{P_{xx}}(t)$ can be found to be

$$\dot{\mathbf{P}}_{\mathbf{xx}}(t) = \mathbf{F}(t)\mathbf{P_{xx}}(t) + \mathbf{P_{xx}}(t)\mathbf{F}^T(t) + \mathbf{G}(t)\,\mathbf{Q}\,\mathbf{G}^T(t)$$

In the above development we have made use of *the Sifting Property of the Dirac Delta*, $\delta(t-\tau)$, expressed as

$$\int_{-\infty}^{\infty} f(\xi)\delta(t-\xi)d\xi = f(t)$$

A Discrete Linear, Vector Process

Given the continuous process ($\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t)$), whose solution is

$$\mathbf{x}(t_k) = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{x}(t_{k-1}) + \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)\mathbf{w}(\xi)d\xi$$

the discrete stochastic analog process is

$$\mathbf{x}_k = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{x}_{k-1} + \mathbf{w}_k, \quad \text{with} \;\; \mathbf{w}_k \stackrel{\Delta}{=} \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t,\xi)\mathbf{G}(\xi)\mathbf{w}(\xi)d\xi$$

whose mean is

$$\bar{\mathbf{x}}_k = \mathbf{\Phi}(t_k, t_{k-1})\bar{\mathbf{x}}_{k-1}$$

The Covariance of a Discrete Linear, Vector Process

Likewise, the continuous-time solution for the covariance was

$$\begin{aligned}\mathbf{P_{xx}}(t_k) &= \mathbf{\Phi}(t_k, t_0)\mathbf{P_{xx}}(t_0)\mathbf{\Phi}^T(t_k, t_0) \\ &\quad + \int_{t_0}^{t} \mathbf{\Phi}(t_k, \xi)\mathbf{G}(\xi)\,\mathbf{Q}\,\mathbf{G}^T(\xi)\mathbf{\Phi}^T(t_k, \xi)\, d\xi\end{aligned}$$

whose discrete analog is

$$\mathbf{P}_{\mathbf{xx}_k} = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{P}_{\mathbf{xx}_{k-1}}\mathbf{\Phi}^T(t_k, t_{k-1}) + \mathbf{Q}_k$$

where

$$\mathbf{Q}_k \triangleq \int_{t_0}^{t} \mathbf{\Phi}(t_k, \xi)\mathbf{G}(\xi)\,\mathbf{Q}\,\mathbf{G}^T(\xi)\mathbf{\Phi}^T(t_k, \xi)\, d\xi$$

Practical Considerations

"There is nothing more practical than a good theory"

Albert Einstein

The Context of the Kalman Filter

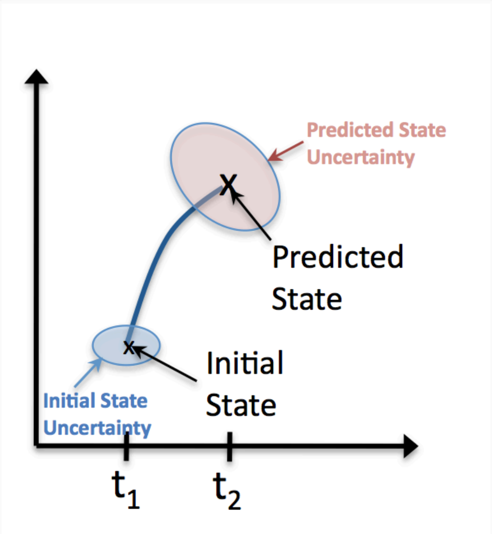
- With the advent of the digital computer and modern control, the following question arose: Can we recursively estimate the state of a vehicle as measurements become available?
- In 1961 Rudolf Kalman came up with just such a methodology to compute an optimal state given linear measurements and a linear system
- The resulting *Kalman filter* is an globally optimal linear, model-based estimator driven by Gaussian, white noise which has two steps
    - Propagation: the state and covariance are propagated from one epoch to the next by integrating model-based dynamics
    - Update: the state and covariance are optimally updated with measurements
- We begin with the same equation as before

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\epsilon}_k \quad \text{with} \quad E(\boldsymbol{\epsilon}_k) = \mathbf{0},\ E(\boldsymbol{\epsilon}_k \boldsymbol{\epsilon}_k^T) = \mathbf{R}_k$$

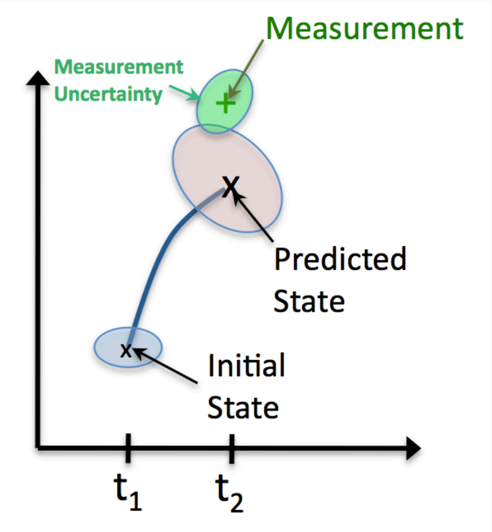
What does a Kalman Filter do?

- Fundamentally, a Kalman filter is nothing more than a predictor (which we call the 'propagation' phase) followed by a corrector (which we call the 'update' phase)
- We use the dynamics (*i.e.* Newton's Laws) to *predict* the state at the time of a measurement
- The measurements are then used to *correct* or update the predicted state.
- It does this in an "optimal" fashion

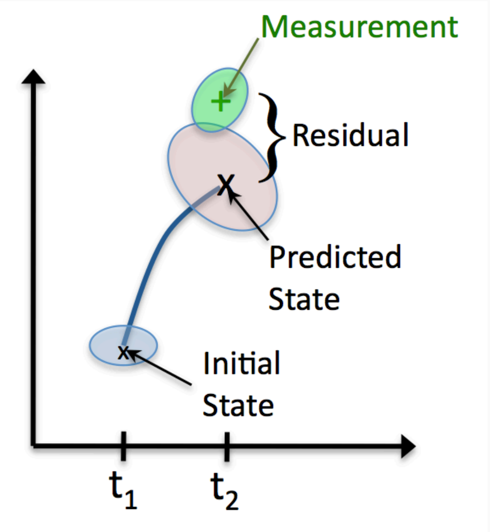
Prediction


Predicted State
Uncertainty
X
Predicted
State
x
Initial
State
Initial State
Uncertainty
t1
t2

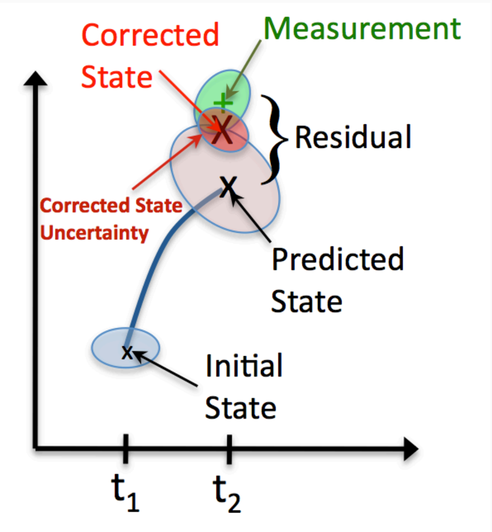
Measurement



Compute Residual



Correction


Corrected State
Measurement
Residual
Corrected State Uncertainty
Predicted State
Initial State
t_1
t_2

The Derivation of the Kalman Filter (I)

Let $\hat{\mathbf{x}}_k^-$ be an unbiased *a priori* estimate (the **prediction**) of $\mathbf{x}_k$ with covariance $\mathbf{P}_k^-$ so that the *a priori* estimate error, $\mathbf{e}_k^-$ is

$$\mathbf{e}_k^- = \mathbf{x}_k - \hat{\mathbf{x}}_k^- \quad \text{with} \quad E(\mathbf{e}_k^-) = \mathbf{0},\ E(\mathbf{e}_k^- \mathbf{e}_k^{-T}) = \mathbf{P}_k^-$$

We hypothesize an unbiased linear update (the **correction**) to $\mathbf{x}_k$, called $\hat{\mathbf{x}}_k^+$, as follows (with $\mathbf{K}_k$ as yet unknown)

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \left(\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- \right)$$

whose *a posteriori* error, $\mathbf{e}_k^+$, is

$$\mathbf{e}_k^+ = \mathbf{x}_k - \hat{\mathbf{x}}_k^+ = \mathbf{e}_k^- - \mathbf{K}_k(\mathbf{H}_k \mathbf{e}_k^- + \epsilon_k) = (\mathbf{I}_k - \mathbf{K}_k \mathbf{H}_k)\mathbf{e}_k^- - \mathbf{K}_k \epsilon$$

If $\mathbf{e}_k^-$ and ϵ_k are uncorrelated, then the *a posteriori* covariance is

$$\mathbf{P}_k^+ = E(\mathbf{e}_k^+ \mathbf{e}_k^{+T}) = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T$$

The Derivation of the Kalman Filter (II)

So far we haven't said anything about $\mathbf{K}_k$. We now choose $\mathbf{K}_k$ to minimize the *a posteriori* error as[1]

$$\begin{aligned} \min J &= \frac{1}{2}E\left[\mathbf{e}_k^{+T}\mathbf{e}_k^{+}\right] = \frac{1}{2}\mathrm{tr}\left\{E\left[\mathbf{e}_k^{+T}\mathbf{e}_k^{+}\right]\right\} = \frac{1}{2}E\left\{\mathrm{tr}\left[\mathbf{e}_k^{+T}\mathbf{e}_k^{+}\right]\right\} \\ &= \frac{1}{2}E\left\{\mathrm{tr}\left[\mathbf{e}_k^{+}\mathbf{e}_k^{+T}\right]\right\} = \frac{1}{2}\mathrm{tr}\left\{E\left[\mathbf{e}_k^{+}\mathbf{e}_k^{+T}\right]\right\} = \frac{1}{2}\mathrm{tr}\left(\mathbf{P}_k^{+}\right) \end{aligned}$$

so we obtain $\mathbf{K}$ by[2]

$$\frac{\partial}{\partial \mathbf{K}_k}\mathrm{tr}\left(\mathbf{P}_k^{+}\right) = \frac{\partial}{\partial \mathbf{K}_k}\mathrm{tr}\left[(\mathbf{I}-\mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^{-}(\mathbf{I}-\mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k^T\right] = \mathbf{0}$$

[1] The *cyclic invariance* property of the trace is: $\mathrm{tr}(\mathbf{ABC}) = \mathrm{tr}(\mathbf{BCA}) = \mathrm{tr}(\mathbf{CAB})$

[2] Recalling that

$$\frac{\partial}{\partial \mathbf{X}}\mathrm{tr}(\mathbf{AXBX}^T) = \mathbf{A}^T\mathbf{XB}^T + \mathbf{AXB};\quad \frac{\partial}{\partial \mathbf{X}}\mathrm{tr}(\mathbf{AXB}) = \mathbf{A}^T\mathbf{B}^T;\quad \frac{\partial}{\partial \mathbf{X}}\mathrm{tr}(\mathbf{AX}^T\mathbf{B}) = \mathbf{BA}$$

The Derivation of the Kalman Filter (III)

This results in the following condition

$$-\mathbf{P}_k^-\mathbf{H}_k^T - \mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{K}_{k_{opt}}\left(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right)^T + \mathbf{K}_{k_{opt}}\left(\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T + \mathbf{R}_k\right) = \mathbf{0}$$

which gives

$$\mathbf{K}_{k_{opt}} = \mathbf{P}_k^-\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}$$

and substituting into the equation[2] for $\mathbf{P}^+$we get

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{P}_k^-\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}\mathbf{H}_k\mathbf{P}_k^- = \left(\mathbf{I} - \mathbf{K}_{k_{opt}}\mathbf{H}_k\right)\mathbf{P}_k^-$$

so the state update is

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_{k_{opt}}\left(\mathbf{y}_k - \mathbf{H}_k\hat{\mathbf{x}}_k^-\right)$$

[2] Recall that $\mathbf{P}^+ = (\mathbf{I} - \mathbf{KH})\mathbf{P}^-(\mathbf{I} - \mathbf{KH})^T + \mathbf{KRK}^T$

The Kalman Filter Revealed

Given the dynamics and the measurements

$$\begin{aligned}
\mathbf{x}_k &= \boldsymbol{\Phi}(t_k, t_{k-1})\mathbf{x}_{k-1} + \boldsymbol{\Gamma}_k \mathbf{w}_k, \quad \text{with } E(\mathbf{w}_k) = \mathbf{0},\ E(\mathbf{w}_k \mathbf{w}_j^T) = \mathbf{Q}_k \delta_{kj} \\
\mathbf{y}_k &= \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\epsilon}_k, \quad \text{with } E(\boldsymbol{\epsilon}_k) = \mathbf{0},\ E(\boldsymbol{\epsilon}_k \boldsymbol{\epsilon}_j^T) = \mathbf{R}_k \delta_{kj}
\end{aligned}$$

The Kalman Filter contains the following phases:

Propagation – the Covariance Increases

$$\begin{aligned}
\hat{\mathbf{x}}_k^- &= \boldsymbol{\Phi}(t_k, t_{k-1})\hat{\mathbf{x}}_{k-1}^+ \\
\mathbf{P}_k^- &= \boldsymbol{\Phi}(t_k, t_{k-1})\mathbf{P}_{k-1}^+ \boldsymbol{\Phi}^T(t_k, t_{k-1}) + \boldsymbol{\Gamma}_k \mathbf{Q}_k \boldsymbol{\Gamma}_k^T
\end{aligned}$$

Update – the Covariance Decreases

$$\begin{aligned}
\mathbf{K}_{k_{opt}} &= \mathbf{P}_k^- \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k\right)^{-1} \\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_{k_{opt}} \left(\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-\right) \\
\mathbf{P}_k^+ &= \left(\mathbf{I} - \mathbf{K}_{k_{opt}} \mathbf{H}_k\right)\mathbf{P}_k^- = \left(\mathbf{I} - \mathbf{K}_{k_{opt}} \mathbf{H}_k\right)\mathbf{P}_k^- \left(\mathbf{I} - \mathbf{K}_{k_{opt}} \mathbf{H}_k\right)^T + \mathbf{K}_{k_{opt}} \mathbf{R}_k \mathbf{K}_{k_{opt}}^T
\end{aligned}$$

A Kalman Filter Example

Given a spring-mass-damper system governed by the following equation

$$\ddot{r}(t) = -0.001 r(t) - 0.005 \dot{r}(t) + w(t)$$

the system can be written (in first-order discrete form, $\mathbf{x}_k = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{x}_{k-1} + \mathbf{\Gamma}_k \mathbf{w}_k$) as

$$\begin{bmatrix} r(t_k) \\ \dot{r}(t_k) \end{bmatrix} = \exp\left\{ \begin{bmatrix} 0 & 1 \\ -0.001 & -0.005 \end{bmatrix} \Delta t \right\} \begin{bmatrix} r(t_{k-1}) \\ \dot{r}(t_{k-1}) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_k$$

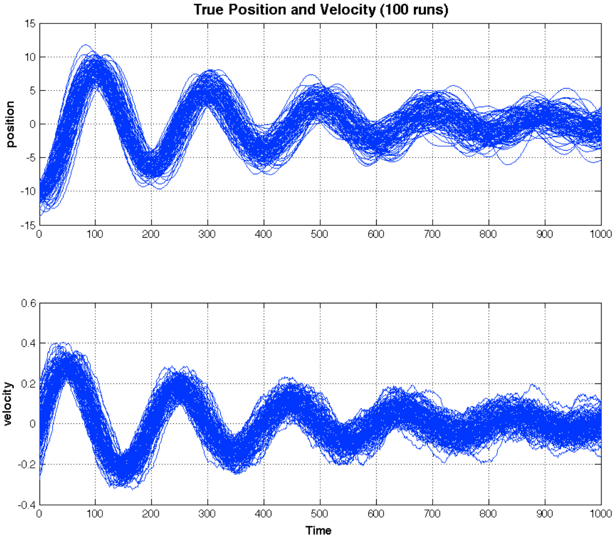
with measurements

$$y_k = r(t_k) + \epsilon_k \;\; \text{with} \;\; E[\epsilon_k] = 0, \;\; E[\epsilon_j \epsilon_k] = 0.001^2 \delta_{jk}$$

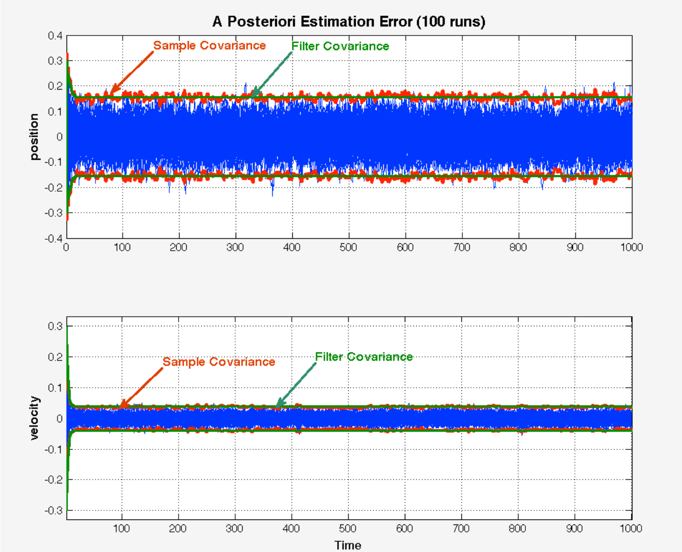
and

$$\mathbf{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 0.1^2 \end{bmatrix} \;\; \text{and} \;\; Q = 0.005^2$$

A Kalman Filter Example (II)



A Kalman Filter Example (III)



Practical Considerations

"A computation is a temptation that should be resisted a long as possible "

John Boyd (paraphrasing T.S. Eliot) , 2000

The Extended Kalman Filter

Since we live in a nonlinear and non-Gaussian world, can we fit the Kalman filter paradigm into the 'real' world? Being engineers, when all else fails, we linearize.

$$\widehat{\mathbf{X}}_k = \mathbf{X}_k^\star + \hat{\mathbf{x}}_k$$

This process results in an algorithm called *the Extended* Kalman filter (EKF). However all guarantees of stability and optimality are lost. *The EKF is a conditional mean estimator with dynamics truncated after first-order by deterministically linearizing about the conditional mean.*

The Development of the Extended Kalman Filter (I)

Begin with the nonlinear state equation

$$\dot{\mathbf{X}}(t) = \mathbf{f}(\mathbf{X}, t) + \mathbf{w}(t) \quad \text{with} \quad E[\mathbf{w}(t)] = \mathbf{0},\ E[\mathbf{w}(t)\mathbf{w}(\tau)] = \mathbf{Q}\,\delta(t - \tau)$$

whose solution, given $\mathbf{X}(t_{k-1})$ is

$$\mathbf{X}(t) = \mathbf{X}(t_{k-1}) + \int_{t_{k-1}}^{t} \mathbf{f}(\mathbf{X}, \xi) d\xi + \int_{t_{k-1}}^{t} \mathbf{w}(\xi) d\xi$$

We expand $\mathbf{f}(\mathbf{X}, t)$ in a Taylor series about $\hat{\mathbf{X}} = E(\mathbf{X})$ as

$$\dot{\mathbf{X}}(t) = \mathbf{f}(\hat{\mathbf{X}}, t) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{X}}\right|_{\mathbf{X}=\hat{\mathbf{X}}} \left(\mathbf{X} - \hat{\mathbf{X}}\right) + \cdots + \mathbf{w}(t)$$

so that $\dot{\hat{\mathbf{X}}}(t)$, neglecting higher than first-order terms,

$$\dot{\hat{\mathbf{X}}}(t) = \mathbf{f}(\hat{\mathbf{X}}, t)$$

The Development of the Extended Kalman Filter (II)

Recalling the definition of $\mathbf{P}\left(\stackrel{\Delta}{=} E\left[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T\right]\right)$, we find that

$$\dot{\mathbf{P}}(t) = \mathbf{F}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^T(t) + \mathbf{Q} \quad \text{where} \quad \mathbf{F} \stackrel{\Delta}{=} \left.\frac{\partial \mathbf{f}}{\partial \mathbf{X}}\right|_{\mathbf{X}=\hat{\mathbf{X}}}$$

which can be integrated as

$$\mathbf{P}(t_k) = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{P}(t_{k-1})\mathbf{\Phi}^T(t_k, t_{k-1}) + \mathbf{Q}_k$$

with $\mathbf{\Phi}(t_{k-1}, t_{k-1}) = \mathbf{I}$ and

$$\dot{\mathbf{\Phi}}(t, t_{k-1}) = \mathbf{F}(t)\mathbf{\Phi}(t, t_{k-1}), \quad \text{and} \quad \mathbf{Q}_k = \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \xi)\,\mathbf{Q}\,\mathbf{\Phi}^T(t_k, \xi)d\xi$$

The Development of the Extended Kalman Filter (III)

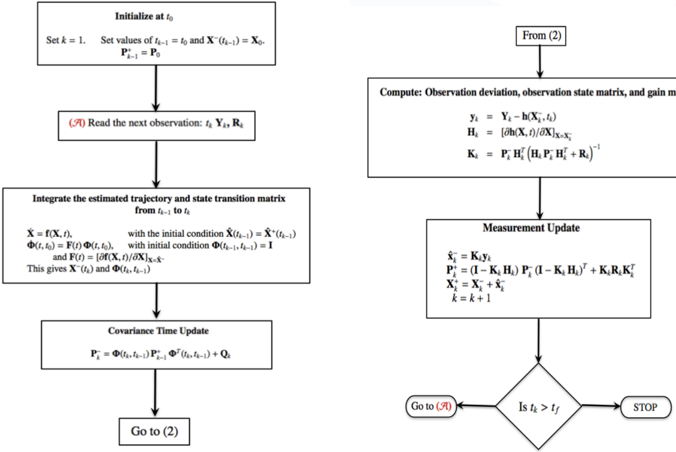
Likewise, the measurement equation can be expanded in a Taylor series about $\hat{\mathbf{X}}_k^-$, the *a priori* state, as

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k) + \epsilon_k = \mathbf{h}(\hat{\mathbf{X}}_k^-) + \left.\frac{\partial \mathbf{h}}{\partial \mathbf{X}_k}\right|_{\mathbf{X}_k = \hat{\mathbf{X}}_k^-} \left(\mathbf{X}_k - \hat{\mathbf{X}}_k^-\right) + \cdots + \epsilon_k$$

In the EKF development, we truncate the Taylor series after first-order. As in the Kalman filter development, we minimize the trace of the *a posteriori* covariance and this results in

$$\begin{aligned}
\mathbf{K}_k(\hat{\mathbf{X}}_k^-) &= \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{X}}_k^-) \left[\mathbf{H}_k(\hat{\mathbf{X}}_k^-) \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{X}}_k^-) + \mathbf{R}_k\right]^{-1} \\
\mathbf{P}_k^+ &= \left[\mathbf{I} - \mathbf{K}_k(\hat{\mathbf{X}}_k^-)\mathbf{H}_k^T(\hat{\mathbf{X}}_k^-)\right] \mathbf{P}_k^- \\
\hat{\mathbf{X}}_k^+ &= \hat{\mathbf{X}}_k^- + \mathbf{K}_k(\hat{\mathbf{X}}_k^-)\left[\mathbf{Y}_k - \mathbf{h}_k(\hat{\mathbf{X}}_k^-)\right] \\
\mathbf{H}_k(\hat{\mathbf{X}}_k^-) &= \left.\frac{\partial \mathbf{h}}{\partial \mathbf{X}_k}\right|_{\mathbf{X}_k = \hat{\mathbf{X}}_k^-}
\end{aligned}$$

The Extended Kalman Filter (EKF) Algorithm



Practical Considerations

“In theory, there is no difference between theory and practice, but in practice there is”

John Junkins, 2012

Practical Matters – Processing Multiple Measurements

- In general, more than one measurement will arrive at the same time
- Usually the measurements are uncorrelated and hence they can be processed one-at-a-time
  - However, even if they are correlated, they can usually be treated as if they were uncorrelated – by increasing the measurement noise variance
- If the measurements are processed one-at-a-time, then

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1} = \frac{\mathbf{P}_k^- \mathbf{H}_k^T}{\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + R_k}$$

- Thus there is no need for a matrix inverse – we can use scalar division
- This greatly reduces the computational throughput, not to mention software complexity

Practical Matters – Processing Non-Gaussian Measurements

- The Kalman Filter is predicated on measurements whose errors are Gaussian
- However, real-world sensors seldom have error characteristics that are Gaussian
  - Real sensors have (significant) biases
  - Real sensors have significant skewness (third moment) and kurtosis (fourth moment)
    - A great deal of information is contained in the tails of the distribution
- Significant sensor testing needs to be performed to fully characterize a sensor and determine its error characteristics
- *Measurement editing* is performed on the innovations process ($\eta_{i_k} = Y_{i_k} - h_i(\hat{\mathbf{X}}_k^-)$ with variance $V_{i_k} = \mathbf{H}_{i_k}\mathbf{P}_k^-\mathbf{H}_{i_k}^T + R_{i_k}$)
  - Don't edit out measurements that are greater than $3V_{i_k}$
  - We process measurements that are up to $6V_{i_k}$

Practical Matters – Dealing with Measurement Latency

- Measurements aren't so polite as to be time-tagged or to arrive at the major cycle of the navigation filter (t_k)
- Therefore, we need to process the measurements at the time they are taken, assuming that the measurements are not too latent
  - Provided they are less than (say) 3 seconds latent
- The state is propagated back to the measurement time using, say, a first-order integrator

$$\mathbf{X}_m = \mathbf{X}_k + \mathbf{f}(\mathbf{X}_k)\Delta t + \frac{\partial \mathbf{f}}{\partial \mathbf{X}}(\mathbf{X}_k)\mathbf{f}(\mathbf{X}_k)\Delta t^2$$

- The measurement partial mapping is done in much the same way as it was done in 'batch estimation'
  - Map the measurement sensitivity matrix at the time of the measurement($\mathbf{H}(\mathbf{X}_m)$) to the filter time (t_k) using the state transition matrix, $\mathbf{\Phi}(t_m, t_k)$.

Practical Matters – Measurement Underweighting

- Sometimes, when accurate measurements are introduced to a state which isn't all that accurate, filter instability results
- There are several ways to handle this
  - Second-order Kalman Filters
  - Sigma Point Kalman Filters
  - Measurement Underweighting
- Since Apollo, measurement underweighting has been used extensively
- What underweighting does is it slows down the rate that the measurements decrease the covariance
  - It approximates the second-order correction to the covariance matrix
- Underweighting is typically implemented as

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T \left((1+\alpha) \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1}$$

- The scalar α is a 'tuning' parameter used to get good filter performance

Practical Matters – Filter Tuning (I)

- Regardless of how you slice it, tuning a navigation filter is an ‘art’
- There are (at least) two sets of ‘knobs’ one can turn to tune a filter
  - Process Noise (also called ‘State Noise’ or ‘Plant Noise’), **Q**, the noise on the state dynamics
  - Measurement Noise, **R**
- Filter tuning is performed in the context of Monte Carlo simulations (1000’s of runs)
- FIlter designers *begin* with the expected noise parameters
  - Process Noise – the size of the neglected dynamics (*e.g.* a truncated gravity field)
  - Measurement Noise – the sensor manufacturer’s noise specifications

Practical Matters – Filter Tuning (II)

- Sensor parameters (such as bias) are modeled as zero-mean Gauss-Markov parameters, x_p, which have two 'tuning' parameters
  - The Steady State Variance ($P_{p_{ss}}$)
  - The Time Constant (τ)

$$
\begin{aligned}
\frac{d}{dt} x_p &= -\frac{1}{\tau_p} x_p + w_p, \quad \text{where } E[w_p(t) w_p(\tau)] = Q_p \delta(t - \tau) \\
Q_p &= 2 \frac{P_{p_{ss}}}{\tau_p}
\end{aligned}
$$

- All of these are 'tuned' in the Monte Carlo environment so that
  - The state error remains mostly within the 3-σ bounds of the filter covariance
  - The filter covariance represents the computed sample covariance

Practical Matters – Filter Tuning (III)

- Sometimes the filter designer inadvertently chooses a process noise such that the covariance of the state gets too small
- When this happens, the filter thinks it is very sure of itself – it is **smug**
- The end result is that the filter starts rejecting measurements
  - Never a good thing
- The solution to this problem is to inject enough process noise to keep the filter 'open'
  - This allows the filter to process measurements appropriately
- There are several spacecraft which have experienced problems because the designers have chosen incorrect (too small) process noise
- Of course, this is nothing more than the classic tension between 'stability' and 'performance'

Practical Matters – Invariance to Measurement Ordering

- Because of its nonlinear foundation, the performance of an EKF can be highly dependent on the order in which measurements are processed
  - For example, if a system processes range and bearing measurements, the performance of the EKF will be different if the range is processed first versus if the bearing were processed first
- To remedy this, on Orion we employ a hybrid linear/EKF formulation
  - The state and covariance updates are accumulated in delta state and covariance variables
  - The state and covariance are updated only after all the measurements are processed

Advanced Topics

- The Kalman-Bucy Filter
- The Schmidt-Kalman Consider Filter
- The Kalman Smoother
- Square Root and Matrix Factorization Techniques
    - Potter Square Root Filter (Apollo)
    - UDU Filter (Orion)
- Nonlinear Filters
    - Second-Order Kalman Filters
    - Sigma Point Kalman Filters
    - Particle Filters
    - Entropy Based / Bayesian Inference Filters
- Linear Covariance Analysis

Conclusions

- Kalman Filtering and Least Squares Estimation are at the heart of the spacecraft navigation
  - Ground-based navigation
  - On-board navigation
- Its purpose is to obtain the 'best' state of the vehicle given a set of measurements and subject to the computational constraints of flight software
- It requires fluency with several disciplines within engineering and mathematics
  - Statistics
  - Numerical Algorithms and Analysis
  - Linear and Nonlinear Analysis
  - Sensor Hardware
- Challenges abound
  - Increase demands on throughput
  - Image-based sensors

To put things in perspective

"I never, never want to be a pioneer . . . Its always best to come in second, when you can look at all the mistakes the pioneers made and then take advantage of them."

Seymour Cray

References

- Maybeck, P.S. *Stochastic Models, Estimation, and Control, Volumes 1-3*, Academic Press, 1982.
- Gelb, A. E. (ed.), *Applied Optimal Estimation*, MIT Press, 1974.
- Brown, R.G., and Hwang, P.Y.C.,*Introduction to Random Signals and Applied Kalman Filtering,* John Wiley, 1992.
- Simon, D., *Optimal State Estimation: Kalman, H-infinity, and Nonlinear Approaches,* John WIley, 2006.
- Tapley, B., Schutz, B., and Born, G., *Statistical Orbit Determination,* Elsevier Inc, 2004.
- Zanetti, R., DeMars, K, and Bishop, R., "Underweighting Nonlinear Measurements," *Journal of Guidance, Control, and Dynamics,* Vol. 33, No. 5, September-October 2010.