Autonomes Fahren
VL Sommer 2015

Probabilities and Uncertainties

Dr. Daniel Clarke
Lecturer

Cranfield Defence and Security
Cranfield University
Shrivenham, Swindon, SN6 8LA, UK
Email – d.s.clarke@cranfield.ac.uk
Intended Learning Outcomes

Lecture Course

• Lecture 1
  – Introduction
  – The sensing problem
  – Automotive sensor operation

• Lecture 2
  – Introduction to probabilities and uncertainty
  – Introduction to data fusion - Stereo Imagery
  – Introduction to filtering – The Kalman Filter

• Lecture 3
  – An introduction to the Particle Filter
  – Factor Graphs
  – Simultaneous Localisation and Mapping

• Lecture 4 (TBC)
  – Communications links in autonomous vehicles
  – Security for autonomous cars
The Way to Autonomous Driving

Key Research Concepts for Sensing

- **Sensors** – The development and application of current and future technologies
- **ICT Architectures** – The integration of hardware and software elements within a system with a specific focus on modular/plug-and-play systems
- **Processing** – The localization of data and information processing (e.g. process closer to the sensing aperture or process in the cloud)
- **Fusion Algorithms** – The processes by which the system state is estimated using available data/information
- **Applications** – The opportunities for multi-sensor systems and challenges in realizing them
- **Infrastructure** – In what ways can infrastructure help and hinder development of the above technologies?
Conclusions

• Sensor systems allow us to **estimate** the physical and semantic state of the world around us
  – Signal processing is used to provide digital representation of physical properties
  – All sensor measurements have a degree of uncertainty associated with them
• Sensor systems play an integral role in autonomous and assisted driving
  – Dynamic environment requires real time sensing
  – Safe operation requires both accurate and precise sensor information
• Many types of different sensors
  – Localization sensors (GPS, IMUs)
  – Range sensors: (Radar, Lidar, Ultra Sound)
  – Vision sensors (later lecture)
Signal Processing

State Uncertainty

- We typically process a signal in order to estimate the physical properties of a target object
  - Even with a perfect signal, digital signal processing introduces uncertainty

- Sources of uncertainty
  - Physical signal is always noisy
  - Electronics of the sensor system
  - Mechanics of the sensor system (i.e. aperture, physical sensing etc.)
  - Missing information (occlusions)
  - Clutter (reflections/multi-path targets of non-interest)
  - Sensor pointing in the wrong direction (lense distortion, mechanical distortion etc.)

Random noise
- Easy to model

Systematic errors
- Difficult to model
Signal Processing – State Estimation

Sensor Fusion

• An observation of the state is generally defined as by:
  \[ y_k = Cx_k + z_k + r_k \]
  
  - \( x_k \) is the true value
  - \( z_k \) is the random noise (uncertainty)
  - \( r_k \) is the systematic noise (bias)

• Integration of sensor measurements (Fusion) allows us to
  - Reduce the uncertainty of state estimates
  - Measure physical states which are otherwise unobservable (e.g. stereo imagery)
Probability Theory
Basic Probability Theory

Random variables

• A classic function:

\[ f : \mathbb{R} \to \mathbb{R}^+, x \mapsto x^2 \]

  – For every input the function \( f \) gives always the same output
  – \( f \) is deterministic

• How do we model randomness?

\[ X = \begin{cases} 
0, & 50\% \text{ of the time} \\
1, & 50\% \text{ of the time} 
\end{cases} \]

  – Each evaluation of \( X \) (may) give a different output
  – The value of \( X \) is subject to randomness
  – This is called a random variable
  – Each outcome \( x \) has a probability \( P(X = x) \)

• Probability theory provides a toolbox to handle randomness
Basic Probability Theory

Classical definition of probability (Hidden)

The ratio of this number to that of all the cases possible is the measure of this probability, which is thus simply a fraction whose numerator is the number of favorable cases and whose denominator is the number of all the cases possible.

\[ P(X = x_i) = \frac{N_i}{\sum_{j=1}^{n} N_j} \]

Random variable \( X \)

Possible outcomes: \( x_1, x_2, \ldots, x_n \)

Probability that \( x_i \) happens

\( x_i \) occurred \( N_i \) times

\( \sum_{j=1}^{n} N_j = \) total number of observations

-- Pierre-Simon Laplace --
Basic Probability Theory

Classical definition of probability - Example

The ratio of this number to that of all the cases possible is the measure of this probability, which is thus simply a fraction whose numerator is the number of favorable cases and whose denominator is the number of all the cases possible.

-- Pierre-Simon Laplace --

Example: Throw of a six sided dice

• $X$ = outcome of a dice roll
• 6 possible outcomes: $x_1 = 1, x_2 = 2, \ldots, x_6 = 6$
• $N_i$ = how often we rolled the number $i$

$$P(X = i) = \frac{N_i}{N_1 + N_2 + N_3 + N_4 + N_5 + N_6}$$

• For a large number of trials $P(X = i)$ will approach $\frac{1}{6}$
Basic Probability Theory

Probability distribution function

- **Discrete random variable** $X$
  - Outcomes are selected from a discrete sample space $\Omega$
  - $P(X = x)$ is called **probability mass function**
  - Probabilities sum up to 1: $\sum_{x \in \Omega} P(X = x) = 1$
  - Example: Roll of dice

- **Continuous random variable** $X$
  - Outcomes are selected from a continuous sample space $\Omega$
  - $X$ has a probability density function $f: \Omega \to \mathbb{R}^+$
  - $P(a \leq X \leq b) = \int_a^b f(x) \, dx$
  - Careful: $P(X = a) = 0!$
  - Probabilities sum up to 1: $\int_{\Omega} P(X = x) \, dx = 1$
  - Example: Normal distribution
  - Advanced mathematical tools required, e.g. measure theory
Basic Probability Theory

The normal distribution

- Normal or Gaussian distribution is a continuous probability distribution
- Probability density function (1 dimension):

\[ f_{\mu, \sigma}(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{x-\mu}{\sigma} \right)^2} \]

- \( \mu \) is called the mean
- \( \sigma \) is called the standard deviation

- Statistically well behaved
- Symmetric about the mean
- Many important applications
Normal Distribution

Noise and Bias

- An observation of the state is defined by:
  \[ y_k = Cx_k + z_k + r_k \]

- State variable
  - useful information

- Stochastic Noise
  - Random variations

- Deterministic Noise
  - A bias on the measurement

- We generally assume the bias to be zero
  - Mathematically more tractable (easier)
Small Number Statistics

A short note

• Consider the following
  – Height of all people in Germany
  – Height of all people in this room
  – How do these differ?

• Probability distributions differ
  – Analytical Function
  – Large sample of measurements
  – Small sample of measurements

• We typically record very few measurements!
  – Differences in actual/estimated noise can lead to over and under confidence
**Multivariate Normal Distribution**

**Error Ellipse**

- Represents an iso-contour of the Multivariate normal distribution
  - Confidence ‘intervals’
  - Usually 95%

- Defines a region which contains a percentage of the data set
  - Given the assumed probability distribution

- Eigenvalue decomposition of the covariance matrix
  - Semi major/minor axes and orientation

[Image of a multivariate normal distribution with an error ellipse]

http://www.visiondummy.com/
Conditional Probabilities
Conditional Probabilities

Joint Probability

• Example: Two random variables A and B on same sample space

- Probability is constant within the area
- Probability is zero outside

\[ P(A = a, B = b) \] is called a joint probability distribution
Conditional Probabilities

Conditional probability

- A conditional probability is the probability of an event, given some other event has already occurred.
- The conditional probability $P(A|B)$ can be defined by
  \[ P(\Omega) P(A \cap B) = P(A|B) P(B) \]

**Diagram:**

- **Absolute probability**
- **Conditioned on B**
Conditional Probabilities

Bayes’ theorem

• From the definition of conditional probability:
  – \( P(A \cap B) = P(A|B) P(B) \)
  – \( P(B \cap A) = P(B|A) P(A) \)

• We have:
  \[
P(A \cap B) = P(B \cap A) \Rightarrow P(A|B) P(B) = P(B|A) P(A)
  \]

• This is called Bayes’ theorem:
  \[
P(A|B) = \frac{P(B|A) P(A)}{P(B)}
  \]
Likelihood of Observation

- Sensors produce observations $y$ of a continuous valued state $x$
  - $p_a(y|x)$ - the observation likelihood or sensor model
  - E.g. consider measuring the angle to a target

- Bayes law
  
  $$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

  - Computes the posterior $p(x|y)$ given the prior $p(x)$ and an observation $p(y|x)$

- $p(y|x)$ takes the role of a sensor model
  - First build the model: fix $x$ and then ask what probability density function on $y$ results
  - Then Observe $y$ and ask what the pdf on $x$ is
Bayes Fusion Example

- Continuous valued state $x$ representing the angle to a target
  - We obtain $y$ which is an observation of the state
  - Gaussian observation model given as
    \[ p(y|x) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(\frac{(y - x)^2}{2\sigma_y^2}\right) \]
    - This is a function of both $y$ and $x$
    - Build the model by fixing the state $x$, use the model by fixing the observation $y$

- The prior $p(x)$ represents our prior knowledge of the system
  - Use a prior distribution such as
    \[ p(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(\frac{(x - x_p)^2}{2\sigma_x^2}\right) \]
Bayes Fusion
Example – Stereo Range Finding

• Measuring the position of an object

• Two sensors from different viewpoints

• Each sensor measures the angle of arrival
Bayes Fusion

Example – Stereo Range Finding

• The posterior for sensor a is given as

\[ p(x|y_a) = \frac{p(y_a|x)p(x)}{p(y_a)} \]

  – We can take \( p(y_a) = 1 \)
  – If it is the first measurement we can take \( p(x) = 1 \)
Bayes Fusion

Example – Stereo Range Finding

• Sensor model:

\[ P(y_a|x) = \frac{1}{\sqrt{2\pi}\sigma^2} e \left( -\frac{(\phi - \theta)^2}{2\sigma^2} \right) \]

  - \( \phi \) is the angle between a point on in the search space \( p_x, p_y \) and the location of the sensor \( s_x, s_y \)
  - \( \theta \) is the measured value of theta
  - \( \sigma \) is the standard deviation of the error

• The posterior for sensor \( a \) is given as

\[ p(x|y_a) = \frac{p(y_a|x)p(x)}{p(y_a)} \]
Bayes Fusion

Example – Stereo Range Finding

• Sensor model:

\[ P(y_b|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e\left(-\frac{(\phi-\theta)^2}{2\sigma^2}\right) \]

- \( \phi \) is the angle between a point on in the search space \( p_x, p_y \) and the location of the sensor \( s_x, s_y \)
- \( \theta \) is the measured value of theta
- \( \sigma \) is the standard deviation of the error

• The posterior for sensor \( b \) is given as

\[ p(x|y_b) = \frac{p(y_b|x)p(x)}{p(y_b)} \]
Bayes Fusion

Example – Stereo Range Finding

• Sensors produce measurements ($y$) estimating the state vector $x$ (e.g. position, orientation)
  – Likelihood of that state function given as $p_a(y_a|x)$ for node $a$
  – Each sensor will produce a posterior pdf $p_a(x|Z_a)$
    • $Z_a$ is all information available to that sensor (e.g. sensor, a priori data, estimates shared by other sensors)

• Bayesian fusion result for a pair of sensors ($a$ and $b$) given as:

$$p(x|Z_a \cup Z_b) \propto \frac{p_a(x|Z_a)p_b(x|Z_b)}{p(x|Z_a \cap Z_b)}$$
Conditional Probabilities

Example – Stereo Range Finding

- Fusion of the two observations is given as

\[ p(x|Z_a \cup Z_b) \propto \frac{p_a(x|Z_a)p_b(x|Z_b)}{p(x|Z_a \cap Z_b)} \]

  - The denominator is the information which is common between them
  - We assume that the measurements are conditionally independent

- The joint estimate becomes

\[ p(x|Z_a \cup Z_b) \propto p_a(x|Z_a)p_b(x|Z_b) \]

  - \( p_a(x|Z_a) \) and \( p_b(x|Z_b) \) are exponential distance functions
Conditional Probabilities

Example – Stereo Range Finding

- \( p_a(x|Z_a) \) and \( p_a(x|Z_b) \) are distance functions of the form
  \[
  p(x|Z) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\phi - \theta}{2\sigma^2}}
  \]

- What is the location with maximum probability?
  \[
  p^* = \arg\max_{p \in \Omega} p_a(x|Z_a) p_b(x|Z_b)
  \]

- Using maximum likelihood estimation:
  \[
  p^* = \arg\max_{p \in \Omega} \ln p_a(x|Z_a)p_b(x|Z_b)
  \]
  - The logarithm is a strictly increasing function!
  - \( \ln P(X) \) is called the likelihood
  - Allows simplification of our problem
We have:

\[ p^* = \arg\max_{p \in \Omega} \ln p_a(x|Z_a)p_b(x|Z_b) \]

\[ = \arg\max_{p \in \Omega} \left[ \ln P(p|s_1) + \ln P(p|s_2) \right] \]

\[ = \arg\max_{p \in \Omega} \left[ \ln C_1 e^{-\frac{1}{2}d(p,s_1)^2} + \ln C_2 e^{-\frac{1}{2}d(p,s_2)^2} \right] \]

\[ = \arg\max_{p \in \Omega} \left[ -\frac{1}{2}d(p,s_1)^2 - \frac{1}{2}d(p,s_2)^2 + \ln C_1 + \ln C_2 \right] \]

\[ = \arg\min_{p \in \Omega} \left[ d(p,s_1)^2 + d(p,s_2)^2 \right] \]
Conditional Probabilities

Example – Stereo Range Finding

• Using maximum likelihood estimation we transformed a probabilistic formulation to an elegant minimization problem!
  – From \( p^* = \arg\max_{p \in \Omega} P(p|s_1) P(p|s_2) \)
  – To \( p^* = \arg\min_{p \in \Omega} [d(p, s_1)^2 + d(p, s_2)^2] \)

• Now it is possible to use various optimization techniques:
  – Gradient Descent
  – Gauss-Newton
  – Levenberg–Marquardt (damped)
  – Evolutionary Algorithms (GA, PSO)
Image Sensors

Hur et al.: “Multi-Lane Detection in Urban Driving Environments Using Conditional Random Fields”, 2013 IEEE Intelligent Vehicles Symposium (IV)
Image Sensor

Basic Principles

- Pixel grid measures incident photonic energy (i.e. CCD or CMOS)
- Focused by a physical aperture or lens (depends on wavelength)
- Sensor optimized for specific radiation wavelength (including filters)
- Digital processing describes grid of normalized radiation intensity (raw image)
Image Sensor

Mathematical Model

- Intrinsic parameters of the pinhole camera model
  - Focal Length $f$
  - Pixel size $s_x, s_y$
  - Image center $o_x, o_y$

- Lens distortion Parameters
  - E.g. barrel distortion
  - Described by the Distortion coefficients $k_1, k_2...$

- Extrinsic Parameters
  - Rotation
  - Translation
Image Sensors

Computer Vision and Image Processing
Image Sensor
Challenges and Opportunities

- Computer vision allows us to infer physical and semantic state from visual representations
  - Pose (location and orientation) of an object
  - Spectral and spatial characteristics
  - Contextual information

- Principal benefits
  - Low cost
  - Passive (low power consumption)
  - Diverse range of capabilities
  - Conceptually easy to interpret (by humans)

- Principal Challenges
  - Highly affected by clutter and obscuration
  - Affected by lighting conditions
  - Conceptually difficult to interpret (by computers)
Image Sensors

Diversity

- Image sensors have found a vast range of applications due to
  - Diversity and Accuracy
  - Human brain is a natural image processor

- We have shaped the world around us for biological image processing
The Filtering/Tracking Problem
Filtering Problem

Hidden Markov Model

- System can be considered as a Hidden Markov Model

\[ x_k \rightarrow x_{k+1} \rightarrow x_{k+2} \rightarrow \ldots \rightarrow x_{k+n} \]

\[ y_k \rightarrow y_{k+1} \rightarrow y_{k+2} \rightarrow \ldots \rightarrow y_{k+n} \]

- System transitions from one state to \( x_k \) another \( x_{k+1} \)
  - Present state is conditionally dependent upon previous states
    - \( P(x_k|x_{k-1}) \)
- We observe a system property \( y_k \) which relates to that state
  - A system property could be an actual observation (i.e. current position)
    - Measurement is conditionally dependent upon the system state
      - \( P(y_k|x_k) \)
Bayesian Update and Prediction

- Given our Markov assumptions
  - current measurement only depends on current state
- The probability distribution of **predicted state** is the sum of the products of
  - Probability distribution associated with the transition from the previous time step
  - Probability distribution associated with the previous state

$$P(x_k \mid y_{1:k-1}) = \int P(x_k \mid x_{k-1}) P(x_{k-1} \mid y_{1:k-1}) dx_{k-1}$$

- **Updated probability** distribution is proportional to the product of the measurement density and the **predicted state density**

$$P(x_k \mid y_{1:k}) = \frac{P(y_k \mid x_k)P(x_k \mid y_{1:k-1})}{P(y_k \mid y_{1:k-1})}$$

$$P(y_k \mid y_{1:k-1}) = \int P(y_k \mid x_k) P(x_k \mid y_{1:k-1}) dx_k$$

- For each step $k$, we are recursively calculating exact posterior density
Bayesian Update and Prediction

• Given our Markov assumptions
  – current measurement only depends on current state
• The probability distribution of predicted state is the sum of the products of
  – Probability distribution associated with the transition from the previous time step
  – Probability distribution associated with the previous state
  \[ P(x_k | y_{1:k-1}) = \int P(x_k | x_{k-1}) P(x_{k-1} | y_{1:k-1}) \, dx_{k-1} \]

• Updated probability distribution is proportional to the product of the measurement density and the predicted state density
  \[ P(x_k | y_{1:k}) = \frac{P(y_k | x_k) P(x_k | y_{1:k-1})}{P(y_k | y_{1:k-1})} \]

\[ P(y_k | y_{1:k-1}) = \int P(y_k | x_k) P(x_k | y_{1:k-1}) \, dx_k \]

• For each step k, we are recursively calculating exact posterior density
Bayesian Update and Prediction

- Given our Markov assumptions, the current measurement only depends on the current state.

The probability distribution of the predicted state is the sum of the products:

\[ P(\mathbf{x}_k | \mathbf{y}_{1:k-1}) = \int P(\mathbf{x}_k | \mathbf{x}_{k-1}) P(\mathbf{x}_{k-1} | \mathbf{y}_{1:k-1}) d\mathbf{x}_{k-1} \]

- The updated probability distribution is proportional to the product of the measurement likelihood and the predicted state:

\[ P(\mathbf{x}_k | \mathbf{y}_{1:k}) = \frac{P(\mathbf{y}_k | \mathbf{x}_k) P(\mathbf{x}_k | \mathbf{y}_{1:k-1})}{P(\mathbf{y}_k | \mathbf{y}_{1:k-1})} \]

\[ P(\mathbf{y}_k | \mathbf{y}_{1:k-1}) = \int P(\mathbf{y}_k | \mathbf{x}_k) P(\mathbf{x}_k | \mathbf{y}_{1:k-1}) d\mathbf{x}_k \]

- For each step k, we are recursively calculating the exact posterior density probability distribution associated with the predicted state.

Transition between states

Previous state

probability distribution associated with the predicted state
The Kalman Filter
Kalman Filter

Summary

- The Kalman Filter is a linear quadratic estimator
  - Optimal Bayesian estimate under conditions of linear-Gaussian uncertainty
  - Noisy measurements recorded over time
  - Produce estimates of state variables which are typically more precise
  - Tracks the estimated state variable and its uncertainty

- Two step process
  - State prediction step
  - State update step
  - Iterative nature provides real time operation

- Extremely fast however limited to
  - Linear systems
  - Uni-modal systems
The Kalman Filter (1)

Unimodal Linear Motion

- Kalman filters are typically used to remove noise from a signal described by the following linear equations:

  \[ x_k = A x_{k-1} + B u_k + w_k \]
  \[ y_k = C x_k + z_k \]

  \( x \) is the system state; \( u \) is the system control vector;
  \( y \) is the measured output; \( w \) is the process noise;
  \( z \) is the measurement noise; \( A, B, C \) are matrices;
  \( k \) is the time index; each of these elements are typically vectors (dimensions)

- The vector \( x \) contains all information about the state of the system
  - Cannot be measured directly!
  - Our measurement \( y \) is an estimate of the state \( x \), corrupted by noise \( z \)
Kalman Filter

Prior knowledge of state $P_{k-1|k-1}$, $\hat{x}_{k-1|k-1}$

Prediction step
Based on e.g. physical model

Next timestep
$k \leftarrow k + 1$

$P_k|k$, $\hat{x}_k|k$

Update step
Compare prediction to measurements

Output estimate of state

Measurements $y_k$

The Kalman Filter
Example – State Equations

- Model a vehicle moving in a straight line
  - The state we wish to estimate is the position (p) and velocity (v)
  - The state vector $x_k$ is given by $x_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix}$
  - We know acceleration, which is our control variable $u$
  - We measure the position $p$, every $T$ seconds (where $\tilde{v}_k, \tilde{p}_k$ are process noises)

\[
\begin{align*}
  v_{k+1} &= v_k + Tu_k + \tilde{v}_k \\
  p_{k+1} &= p_k + \frac{1}{2} T^2 u_k + \tilde{p}_k
\end{align*}
\]

- As our measured output $y_{k+1}$, is equal to the position, our state equations become:

\[
\begin{align*}
x_{k+1} &= \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_k \\
y_k &= [1 \ 0] x_k + z_k
\end{align*}
\]
Kalman Filter

Building the Example

- Matrices A, B and C are used to model our system

\[ x_{k+1} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_k \]

\[ y_{k+1} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + z_k \]

- **Matrix A** – Describes how we transition from one state to the next
- **Matrix B** – Input model which controls the input \( u \)
- **Matrix C** – Observation model which maps the true state space to observed state space
The Kalman Filter

The filter

• We wish to use the available measurements \( y_k \) to estimate the state of the system \( x_k \)
  – We know the relation between the system state and measurements

• Requirements
  – Average value of the state estimate to be equal to the average of the true state
  – Estimated state should vary from the true state as little as possible
The Kalman Filter

Equations

- **Prediction**
  
  Predicted state:
  \[
  \hat{x}_{k|k-1} = (A\hat{x}_{k-1} + Bu_k)
  \]
  
  Predicted covariance:
  \[
  P_{k|k-1} = AP_{k-1}A^T + Q_k
  \]

- **Update**
  
  Innovation Measurement:
  \[
  J_k = y_k - C\hat{x}_{k|k-1}
  \]
  
  Innovation Covariance:
  \[
  S_k = CP_{k|k-1}C^T + R_k
  \]
  
  Kalman Gain:
  \[
  K_k = P_{k|k-1}C^TS_k^{-1}
  \]
  
  Updated state estimate:
  \[
  \hat{x}_k = \hat{x}_{k|k-1} + K_kJ_k
  \]
  
  Updated Covariance
  \[
  P_k = (I - K_kC)P_{k|k-1}
  \]
The Kalman Filter

Equations

- Prediction
  - Predicted state:
    \[ \hat{x}_{k|k-1} = (A\hat{x}_{k-1} + Bu_k) \]
  - Predicted covariance:
    \[ P_{k|k-1} = AP_{k-1}A^T + Q_k \]

- What does it mean?
  - Control Vector: How we expect the system to transition between states
  - \( Q_k \) is a zero mean Gaussian derived from the process noise
  - How we expect the covariance will look given our uncertainty as to how the system transition between states
The Kalman Filter

Equations

- **Update**
  - $R_k$ is the measurement noise and is used to derive the innovation covariance $S_k$
  - The Kalman Gain $K_k$ is inversely proportional to the innovation Covariance ($S_k$)

- **Update**
  Innovation Measurement:
  \[ J_k = y_k - C\hat{x}_{k|k-1} \]
  Innovation Covariance:
  \[ S_k = CP_{k|k-1}C^T + R_k \]
  Kalman Gain:
  \[ K_k = P_{k|k-1}C^TS_k^{-1} \]
  Updated state estimate:
  \[ \hat{x}_k = \hat{x}_{k|k-1} + K_kJ_k \]
  Updated Covariance
  \[ P_k = (I - K_kC)P_{k|k-1} \]
The Kalman Filter

Equations

• Update
  – The state \( \hat{x}_k \) and covariance \( P_k \) are updated using the Kalman gain \( K_k \)
  – High measurement noise means low Kalman gain
  – Low measurement noise means high Kalman gain

• Update

  Innovation Measurement:
  \( J_k = y_k - C\hat{x}_{k|k-1} \)

  Innovation Covariance:
  \( S_k = CP_{k|k-1}C^T + R_k \)

  Kalman Gain:
  \( K_k = P_{k|k-1}C^TS_k^{-1} \)

  Updated state estimate:
  \( \hat{x}_k = \hat{x}_{k|k-1} + K_kJ_k \)

  Updated Covariance:
  \( P_k = (I - K_kC)P_{k|k-1} \)
Kalman Filter
Vehicle Navigation Example

• Consider our initial problem
  – Vehicle travelling along a road (linear system)

• Remember our state Equations:
  \[
  x_{k+1} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_k \\
  y_{k+1} = [1 0] x_k + z_k \\
  x_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix}
  \]

• State
  – Position error is measured at a standard deviation of 10m
  – Input acceleration is 2m/s^2 (with noise of 0.2 m/s^2)
  – Position recorded 10 times per second (T = 0.1)
Kalman Filter

Example – Error Estimation

- $S_z = E(z_k z_k^T)$ - Measurement Noise
  - Position error is measured at a standard deviation of 10m
  - $S_z = E(10 \times 10) = 100$

- $S_w = E(w_k w_k^T)$ - Process Noise
  - $S_w = E\left(\begin{bmatrix} p^2 & pv \\ pv & v^2 \end{bmatrix}\right)$

- Remember $x_{k+1} = \begin{bmatrix} T \\ 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_k$
  - $p = (0.1^2/2)(0.2) = 10^{-3}$
  - $v = (0.1)(0.2) = 0.02$
  - Remember $u_k$ is our known input (a = 0.2 m/s)
Kalman Filter

Example Graphs

- True Position
- Measured Position
- Estimated Position
Kalman Filter

Example Graphs

- Position error based on measurement
- Position error based on estimate

Figure 2 - Position Measurement Error and Position Estimation Error
Kalman Filter

Example Graphs

- Velocity Estimate
- True Velocity

Figure 3 - Velocity (True and Estimated)
Kalman Filter

Example Graphs

- Velocity Error based on Estimate

Figure 4 - Velocity Estimation Error
Kalman Filter

Summary

- Estimates state of a system
  - Position, velocity, timing
  - Any other continuous state variable

- The Kalman Filter maintains
  - Mean state vector
  - Matrix of state uncertainty (Covariance Matrix)

- Two Step Process
  - Sequential prediction
  - Measurement update

- Standard Kalman filter is linear-Gaussian
  - Linear system dynamics, linear sensor model
  - Additive Gaussian noise (independent)
  - Nonlinear extensions: extended KF, unscented KF
Conclusions

What have we learned?

• Basic Probability Theory
  – Probabilities can be defined as relative occurrences
  – Gaussian function

• Conditional Probabilities
  – Bayes Theorem
  – Stereo range finding example

• Filtering
  – Integration of sensor measurements can help to reduce state uncertainty

• Kalman Filter
  – Linear-Gaussian filtering
  – Uni-modal systems
Dr. Daniel Clarke

Cranfield Defence and Security
Cranfield University
Shrivenham, Swindon, SN6 8LA, UK
E: d.s.clarke@cranfield.ac.uk
W: http://www.cranfield.ac.uk