# SENSOR DATA FUSION

BOSCH

# CONTENTS

# CONTENTS
## Slide structure

# INTRODUCTION IN DATA FUSION

# INTRODUCTION IN DATA FUSION
## Why ?

- Each sensor has its uncertainty ( error in measurement )
  - i.e. GPS – give position with an accuracy of 3 meter

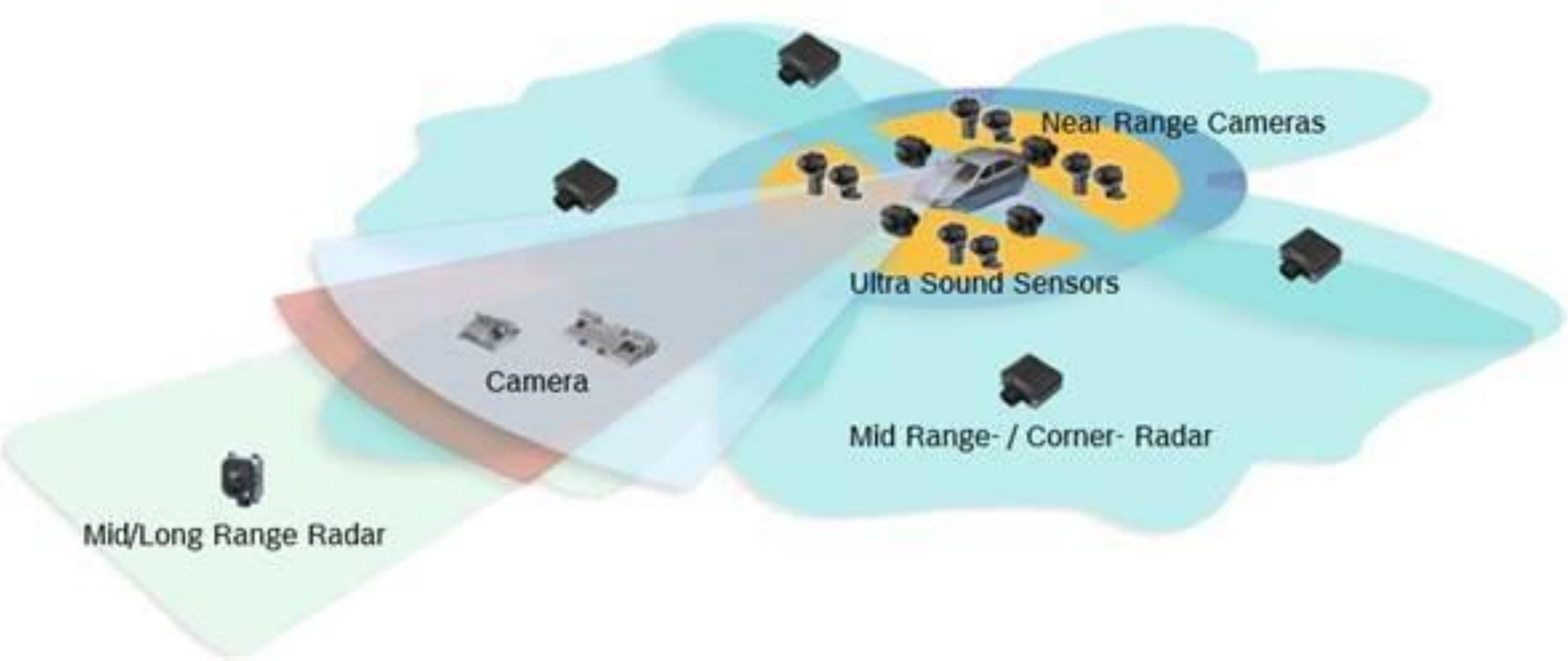
- Each sensor has its drawbacks
  - i.e. GPS – does not work well in tunnels
    IMU (inertial measurement unit )– accumulate error due to the integration process

- Each estimation method does not meet exactly the process
  - i.e. A theoretical model works perfect, but the process has variation due to mechanical issues

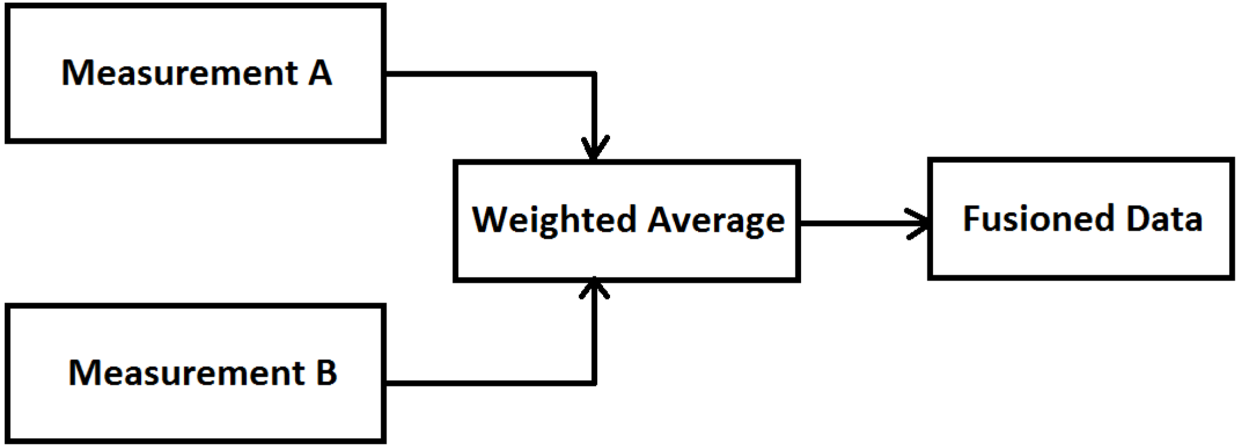


BOSCH

# INTRODUCTION IN DATA FUSION
## Why ?

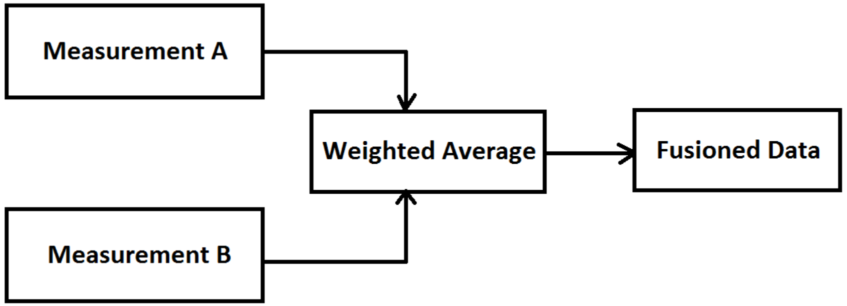


# INTRODUCTION IN DATA FUSION
## How ?




- Which are the weights ?
- Why we want this fusion ?

# INTRODUCTION IN DATA FUSION
## How ?




- Why we want this fusion ?

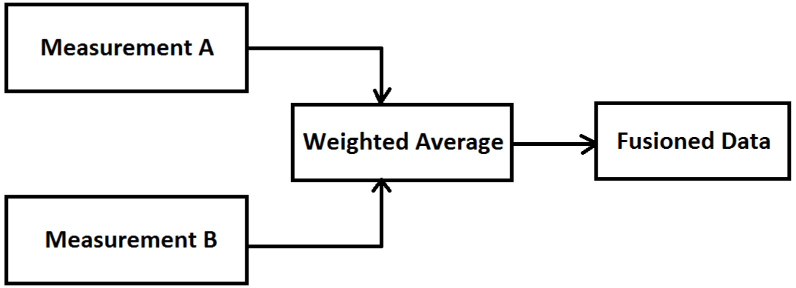
Uncertainty

Some “good” weights

- Which are the weights ?

Greater weight to better data

# INTRODUCTION IN DATA FUSION
## Example





- Estimated relative position
- Accumulated error

A

- Absolute position
- Noisy

B

Which one you trust ?

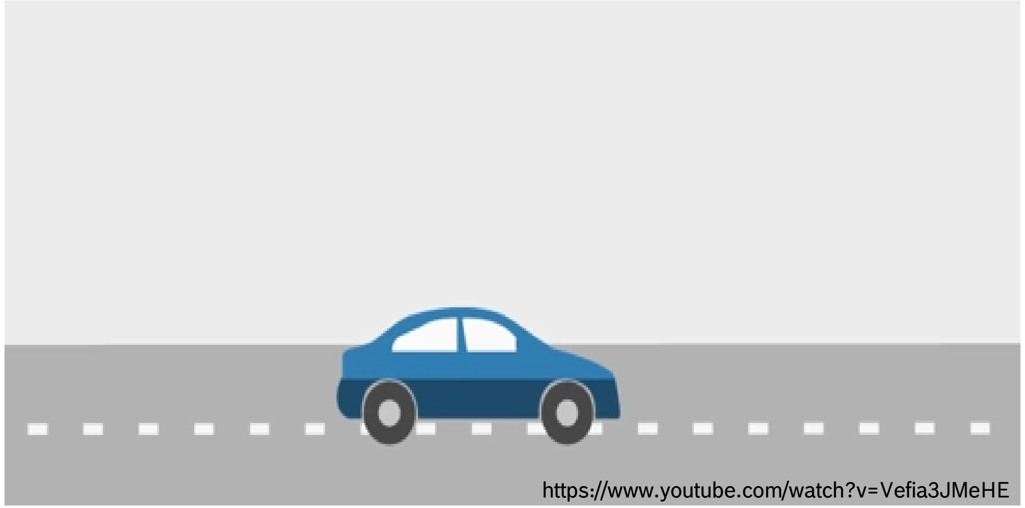


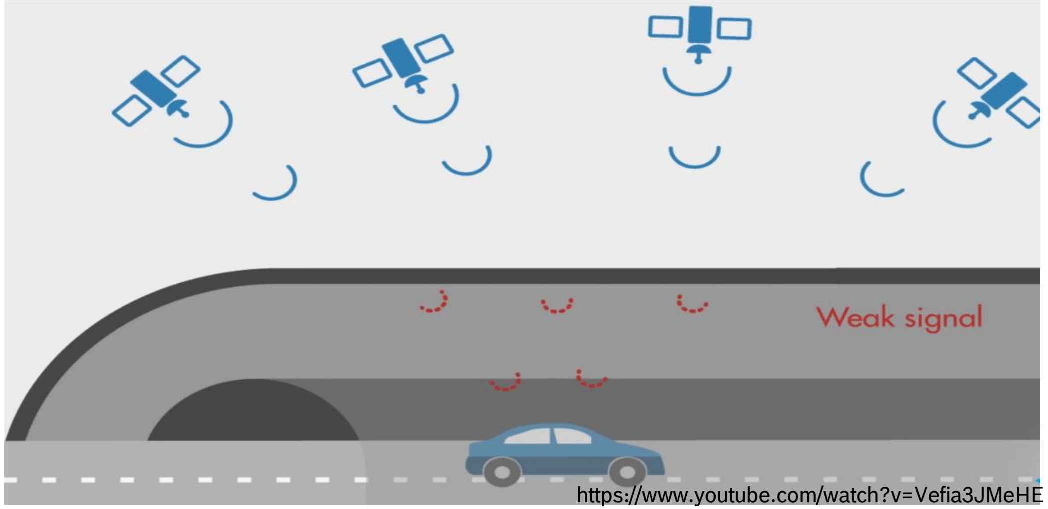
RBRO/ESA1 | 30/08/2017



BOSCH

# INTRODUCTION IN DATA FUSION
## Example




Good GPS – use A only for correction, position is based on B

Bad GPS – use A exclusively to predict from the other sensors

# PRELIMINARY NOTIONS

# Preliminary Notions
## Random Variable

Def.: A function which link a random event with a number.

$$X : E \rightarrow R$$

Example: Roll two dices. One roll represents an event. The random number associated with this event is the sum of the number of dices. This sum is a random variable.

# Preliminary Notions
## Expected Value

Take as random values the exam grades of all students from Control Engineering at math

$$E[X] = \frac{\sum_{i=1}^{N}(x_i)}{N}$$

What we expect to be the grade of a student?

Average of the students grades

# Preliminary Notions
## Variance

Take as random values the exam grades of all students from Control Engineering at math

$$Var(X) = E[(\overline{x} - X)^2] = \frac{\sum_{i=1}^{N}(\overline{x} - x_i)^2}{N}$$

$$[Var] = [X]^2$$

How far from an average student are the best and the worst student

# Preliminary Notions
## Standard Deviation

Take as random values the exam grades of all students from Control Engineering at math

$$\sigma(X) = \sqrt{Var(X)}$$

$$[\sigma] = [X]$$

More convenient

How far from an average student are the best and the worst student





BOSCH

# Preliminary Notions
## Covariance

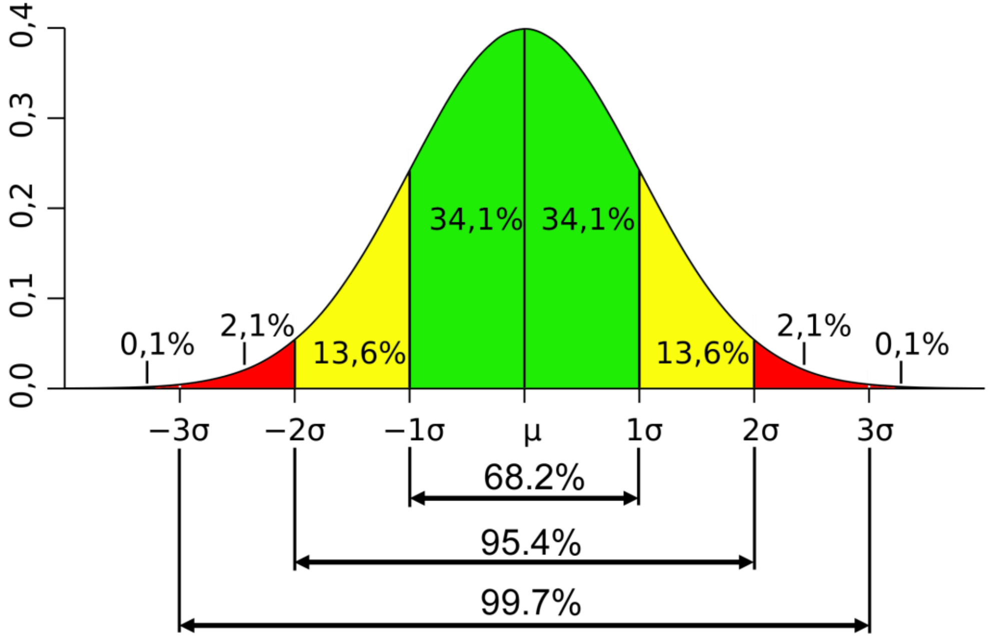
Take as random values the exam grades of all students from Control Engineering at math and English

$$\text{Cov}(X,Y) = E[(\overline{x} - X)(\overline{y} - Y)] = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(\overline{x}-x_i)(\overline{y}-y_i)}{N+M}$$

$$\Updownarrow$$

Express a linear dependence between two dimensions

# Preliminary Notions
## Gaussian Distribution



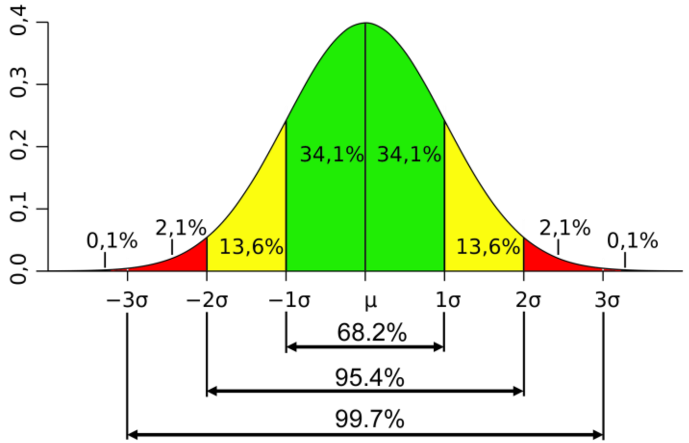

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Probability density function

Take as random value the exam grade of all students from Control Engineering at math

How we interpret this graph ?

# Preliminary Notions
## Gaussian Distribution




$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$E[X_f] = \mu$

Grade of an average student

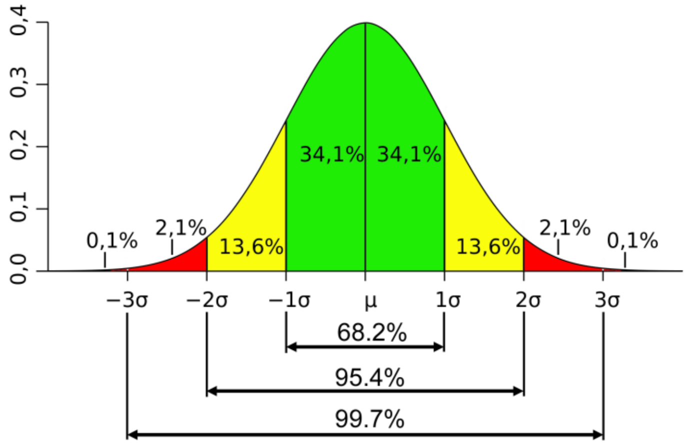
$Var(X_f) = \sigma^2$

How far away is the best and the worst student

Take as random value the exam grade of all students from Control Engineering at math

How we interpret this graph ?

# Preliminary Notions
## White Noise




$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$E[X_f] = \mu = 0$$

We expect the noise to be zero

$$Var(X_f) = \sigma^2$$

Depends on the signal source

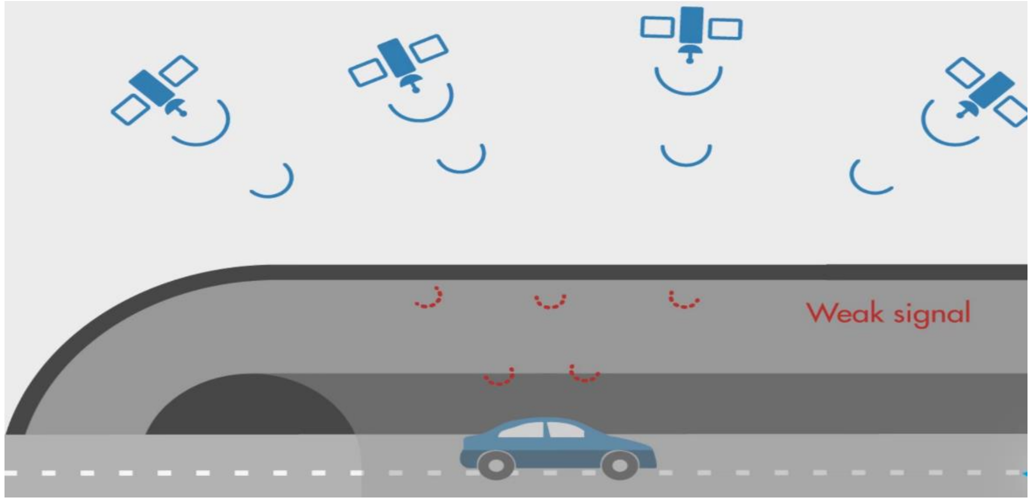
⇔how distorted is the signal

A white noise is a deviation around the 0 which disturb the original signal

Example: Ground variation from an oscilloscope

# PROBLEM TO DISCUSS

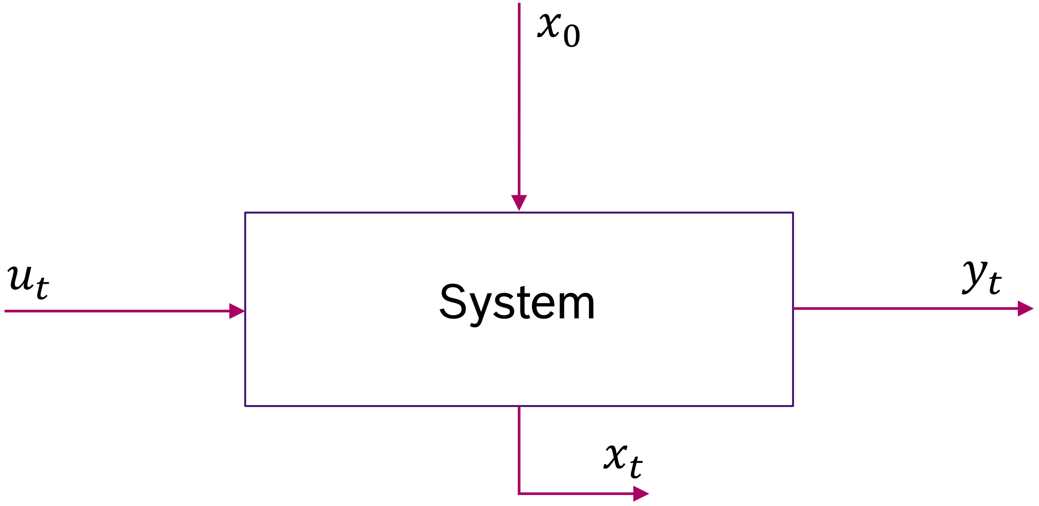
# PROBLEM TO DISCUSS
## Vehicle tracking




- Assume that the trajectory is a line
- You know the velocity of the car as input to the system
- You measure the position of the car via GPS system

# STATE OBSERVER

# STATE OBSERVER
## Introduction




$$x_{t+1} = A_t x_t + B_t u_t$$
$$y_t = H_t x_t$$

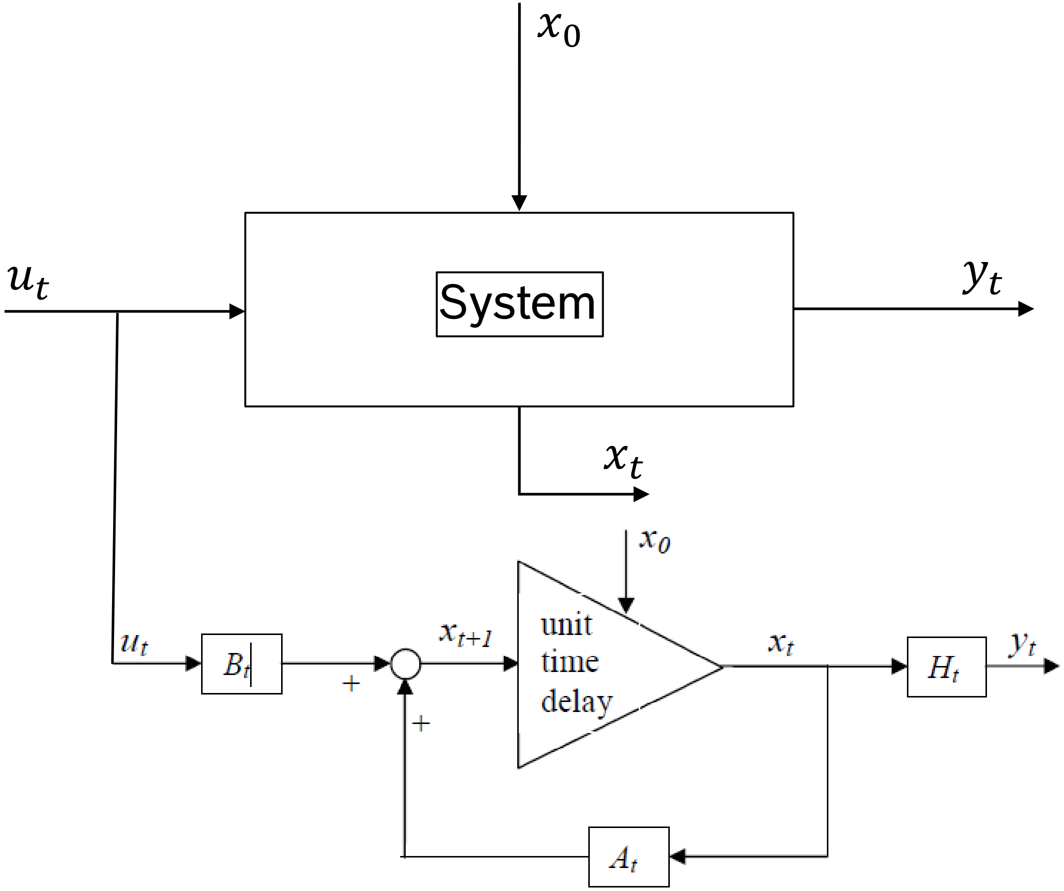
- We measure the output $y_t$
- What about the internal state $x_t$ ? Can estimate the internal state ?

Image taken from www.quora.com



BOSCH

# STATE OBSERVER
## System estimator




$$x_{t+1} = A_t x_t + B_t u_t$$
$$y_t = H_t x_t$$

Is it the same ?

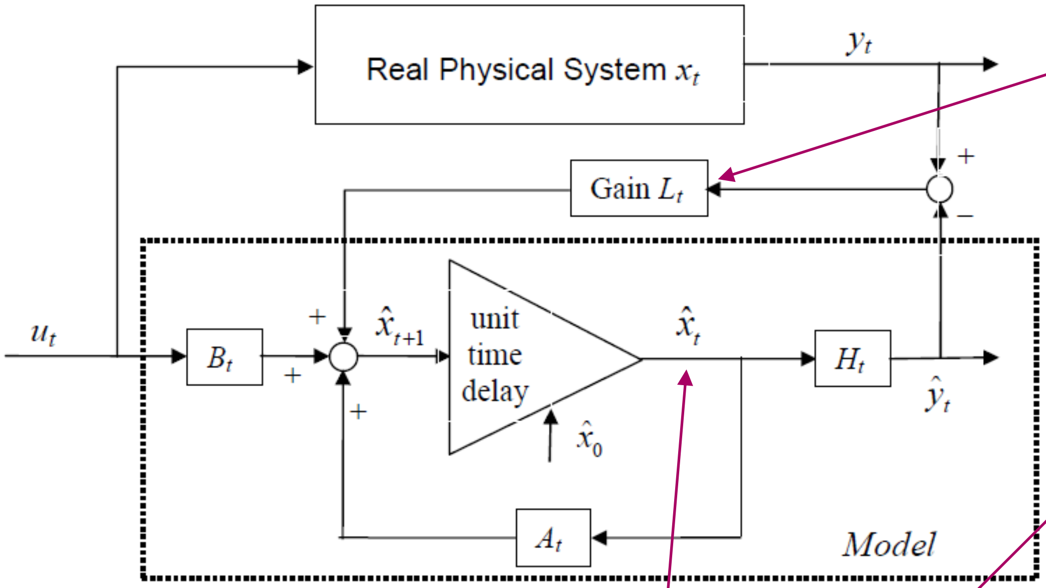
Image taken from www.quora.com




BOSCH

# STATE OBSERVER
## Optimal State Estimator

Feedback ensure the convergence of estimation error to 0




$$\hat{x}_{t+1} = A_t \hat{x}_t + B_t u_t + L_t (y_t - \hat{y}_t)$$
$$\hat{y}_t = H_t \hat{x}_t$$

- ➢What about the noise ?
- ➢How you handle it ?
- ➢How certain is the estimation?

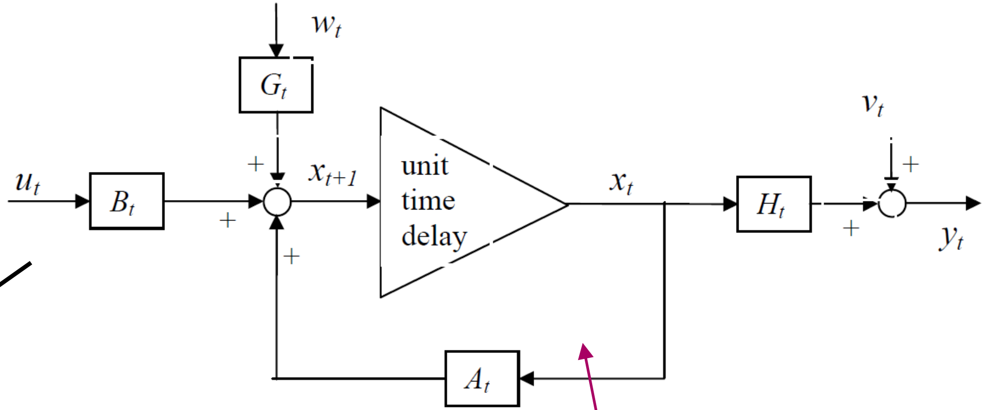
Corrected car’s position

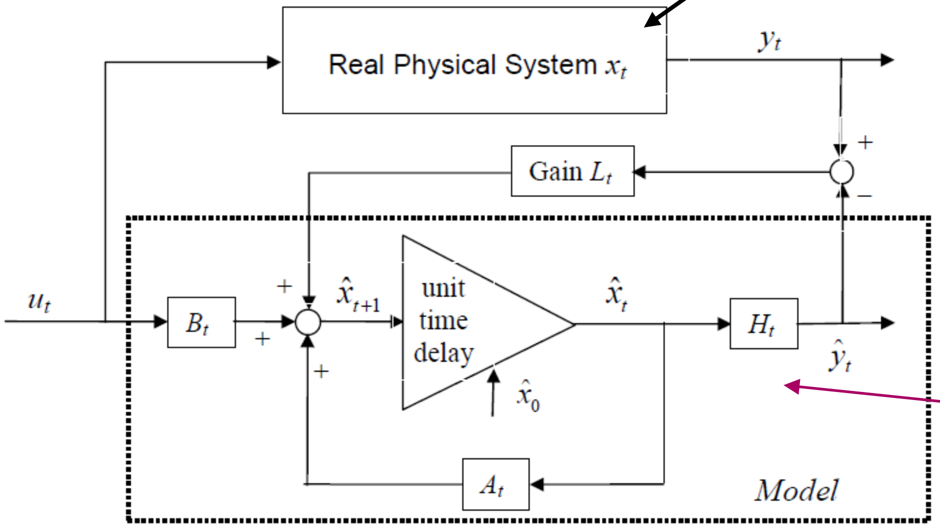
Image taken from www.quora.com




BOSCH

# STATE OBSERVER
## Optimal State Estimator




What you actually have

How you model it

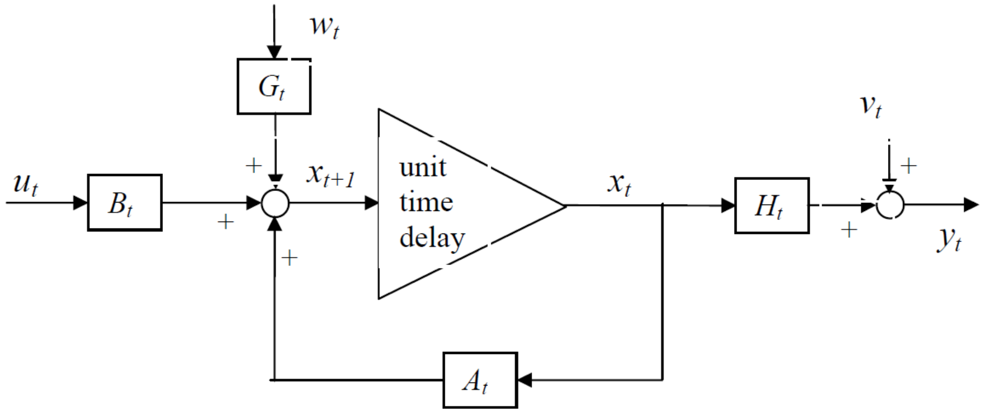
Image taken from www.quora.com




BOSCH

# KALMAN FILTER

# KALMAN FILTER
## Optimal State Estimator For Stochastic Systems

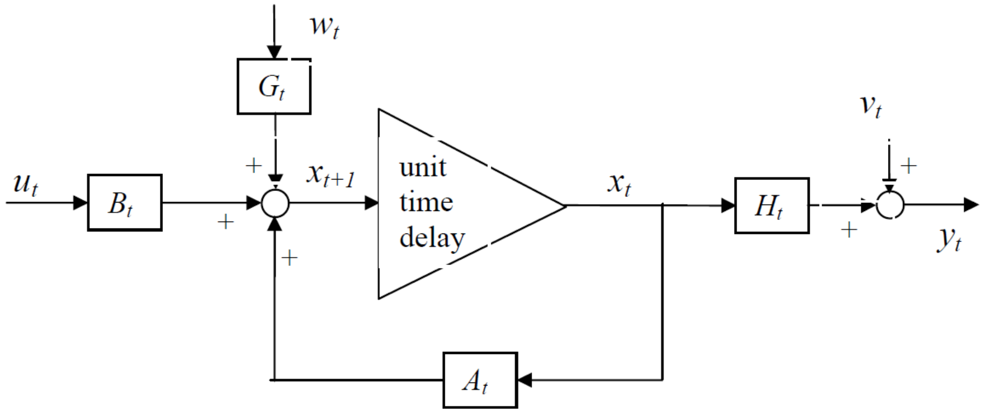



$v_t = measurement\ noise$
$w_t = process\ noise$

- $v_t, w_t \ - is\ assumed\ to\ be\ white\ noises$
- $v_t = N(0, R_t)$, $R_t$ is covariance of measurement noise
- $w_t = N(0, Q_t), Q_t$ is covariance of process noise

Obs.: Process and measurement noise are uncorrelated

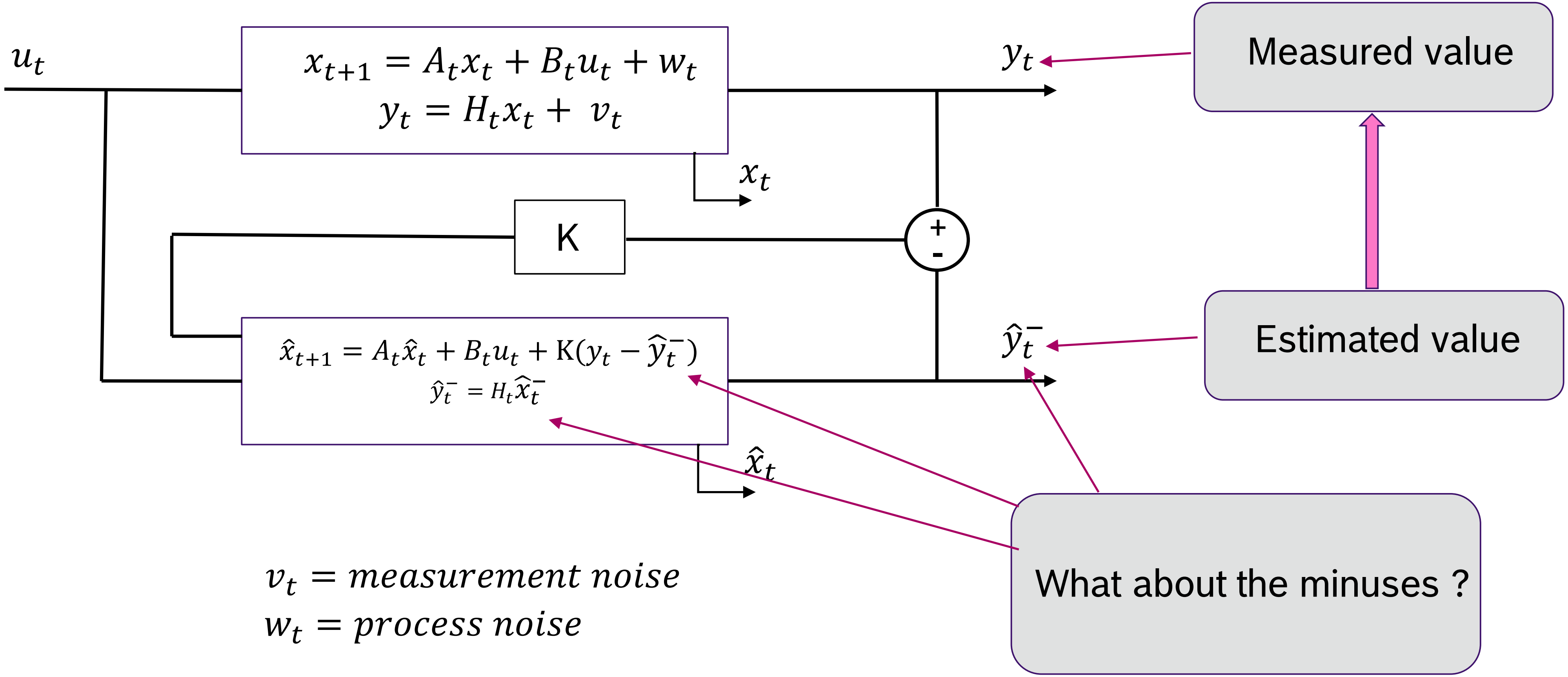
# KALMAN FILTER
## Optimal State Estimator For Stochastic Systems




$$x_{t+1} = A_t x_t + B_t u_t + w_t$$
$$y_t = H_t x_t + v_t$$

$v_t = measurement\ noise$
$w_t = process\ noise$

# KALMAN FILTER
## Optimal State Estimator For Stochastic Systems




$v_t = measurement\ noise$

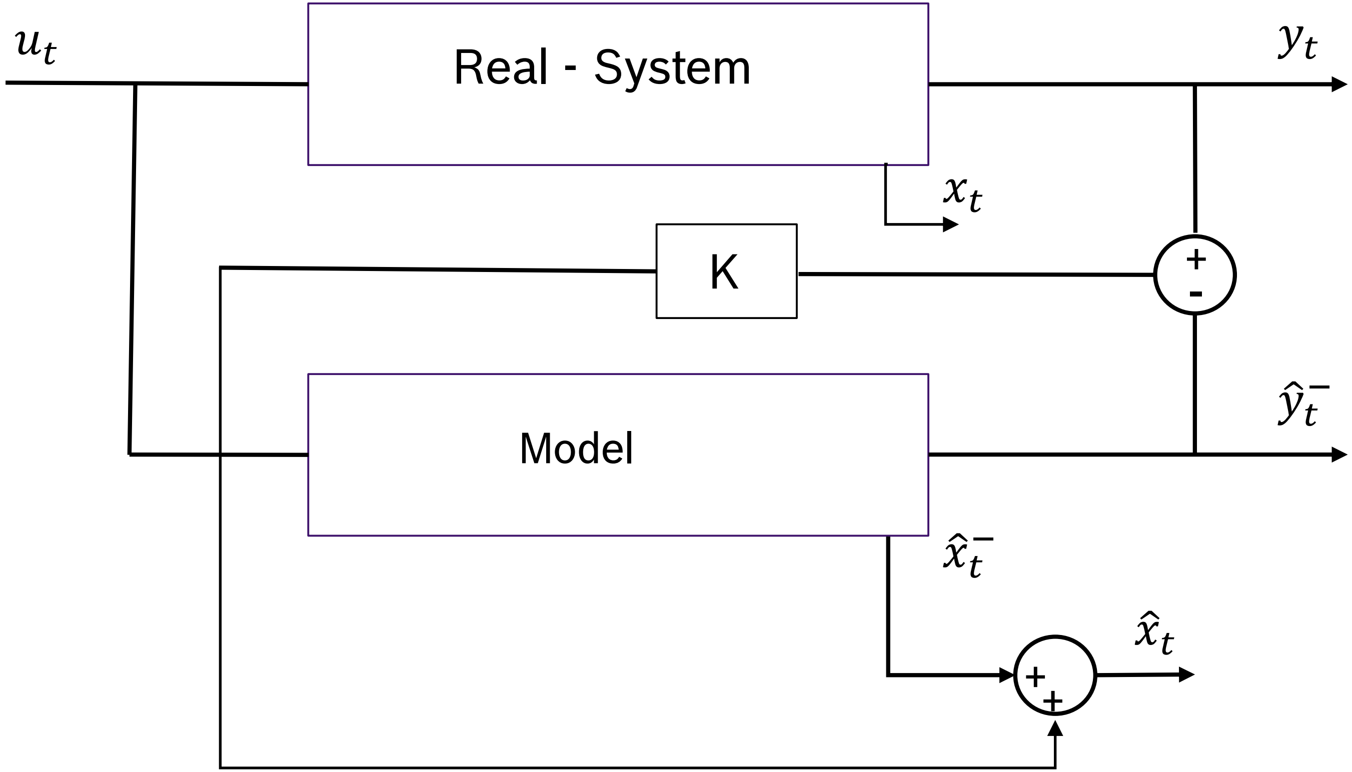
$w_t = process\ noise$


RBRO/ESA1 | 30/08/2017

BOSCH

# KALMAN FILTER

## Optimal State Estimator For Stochastic Systems




$$\hat{x}_t = \underbrace{A\hat{x}_{t-1} + Bu_t}_{\hat{x}_t^-} + K_t(y_t - H\underbrace{(A\hat{x}_{t-1} + Bu_t)}_{\hat{x}_t^-})$$

A priori Estimation

# KALMAN FILTER
## Optimal State Estimator For Stochastic Systems

$$\hat{x}_t = A\hat{x}_{t-1} + Bu_t + K_t(y_t - H(A\hat{x}_{t-1} + Bu_t))$$

$$\hat{x}_t = \hat{x}_t^- + K_t(y_t - H\hat{x}_t^-)$$

A posteriori State Estimation

A priori State Estimation

Correction via Kalman Gain

# KALMAN FILTER
## Kalman Gain

- A posteriori estimation error: $\varepsilon = x_t - \hat{x}_t$

Is a stochastic variable due to process noise $w_t$

- Take the expected value of a posteriori error as an estimation's measure

$$f(K) = E[\varepsilon^T \varepsilon]$$

# KALMAN FILTER
## Kalman Gain

- Take the expected value of a posteriori error as an estimation measure

$$f(K) = E[\varepsilon^T \varepsilon]$$

- Minimizing f with respect to K will give us the best K

$$\frac{df}{dK} = 0 \Rightarrow K_t = \frac{P_t^- H^T}{H P_t^- H^T + R_t},$$ where $P_t^-$ is a priori error covariance

$$P_t^- = \mathrm{E}[(x_t - \hat{x}_t^-)^T (x_t - \hat{x}_t^-)]$$

# KALMAN FILTER
## Updating the error covariance

- Starting from the definition of the a posteriori error covariance

$$P_t = E[\varepsilon^T \varepsilon]$$

- Expanding the above equation by using the model of a posteriori estimation we obtain that

$$P_t = (I - K_t H) P_t^-$$

# KALMAN FILTER
## Estimation of a priori error covariance

- As we estimate a priori the state of the system, we must estimate the a priori error covariance in order to compute the Kalman gain
- Starting from the definition of the a priori error covariance

$$P_t^- = \mathrm{E}[(x_t - \hat{x}_t^-)^T (x_t - \hat{x}_t^-)]$$

- By using the system transition model and considering the expected values of the independent signals to be 0 we obtain the a priori error covariance

$$P_t^- = \mathrm{A}P_{t-1}A^T + Q_t$$

# KALMAN FILTER
## Summary

**Prediction**

$$\hat{x}_t^- = A\hat{x}_{t-1} + Bu_{t-1}$$

$$P_t^- = AP_{t-1}A^T + Q_t$$

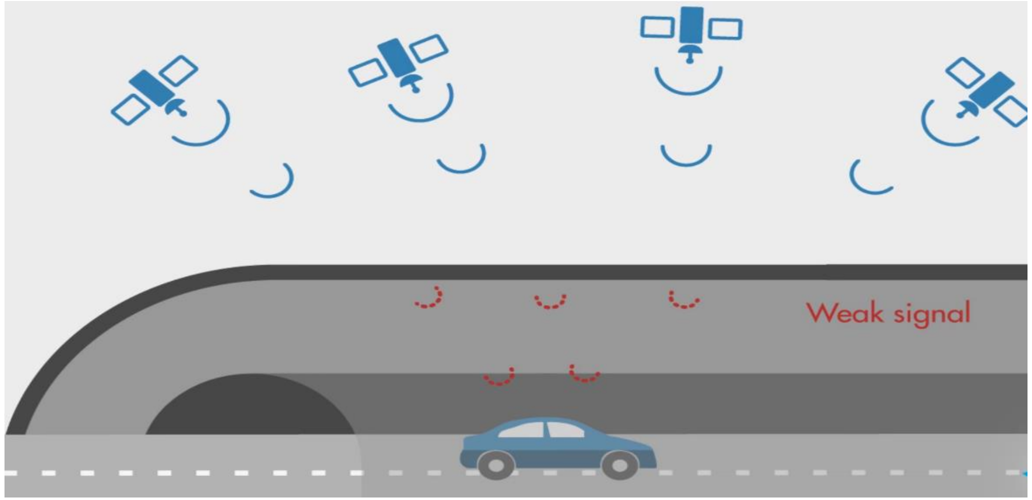
**Update**

$$\hat{x}_t = \hat{x}_t^- + K_t(y_t - H\hat{x}_t^-)$$

$$K_t = \frac{P_t^- H^T}{HP_t^- H^T + R_t}$$

$$P_t = (I - K_t H)P_t^-$$

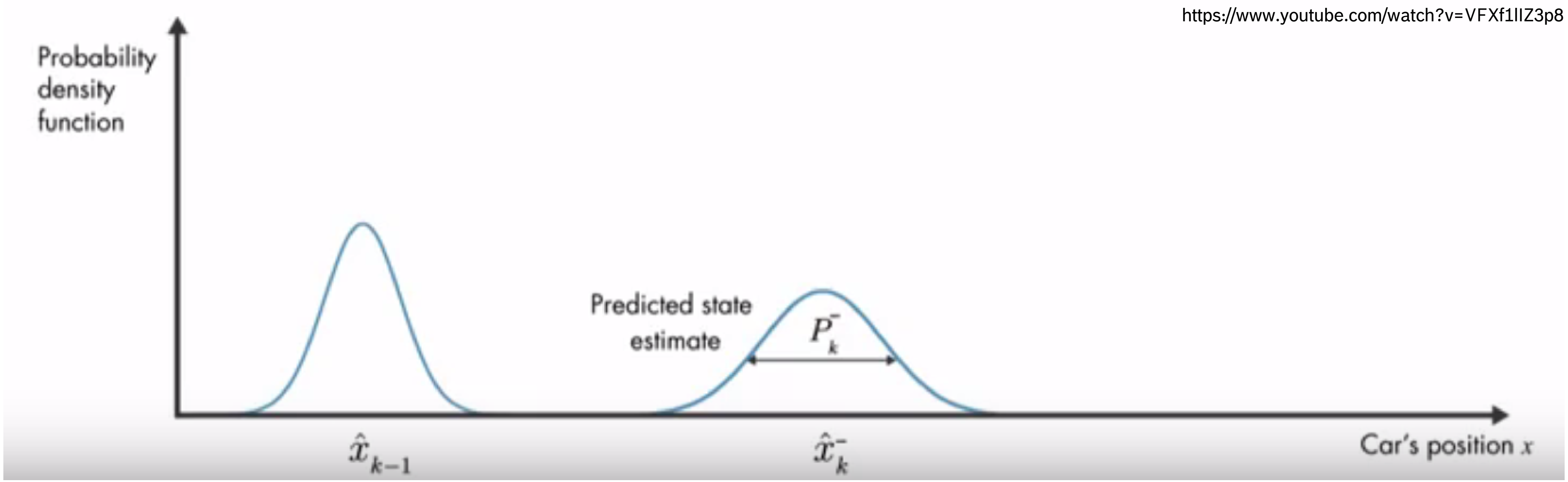
# KALMAN FILTER
## Vehicle tracking




- Assume that the trajectory is a line
- You know the velocity of the car as input to the system
- You measure the position of the car via GPS system

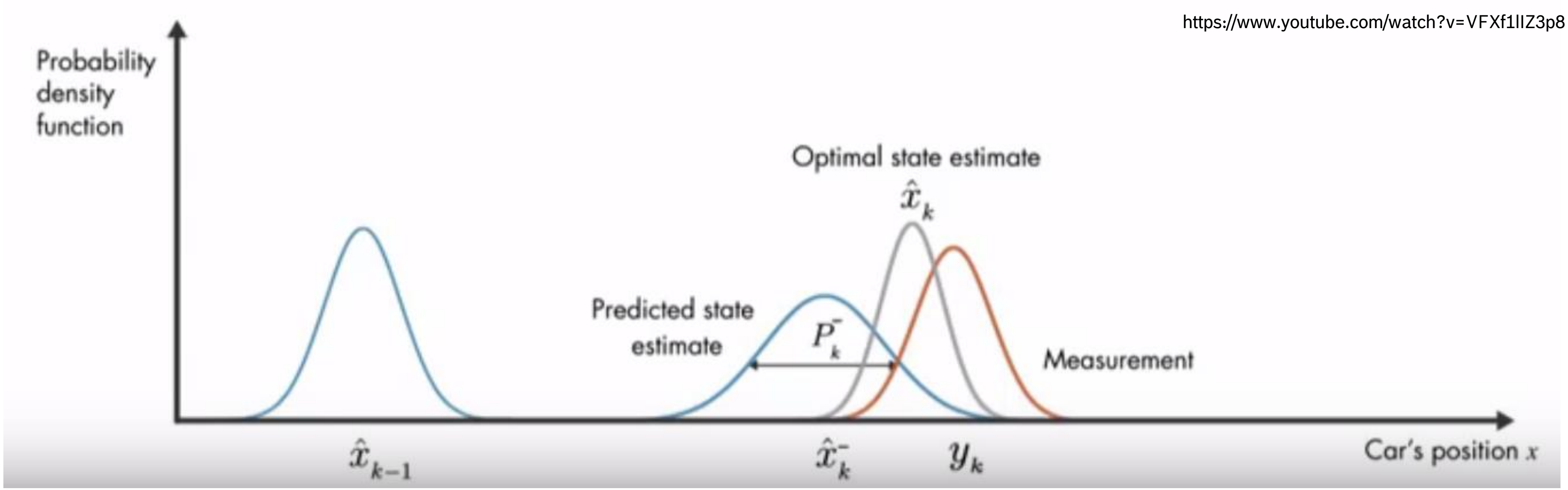
# KALMAN FILTER
## Vehicle tracking - solution

https://www.youtube.com/watch?v=VFXf1lIZ3p8



# KALMAN FILTER
## Vehicle tracking - solution

https://www.youtube.com/watch?v=VFXf1IlZ3p8

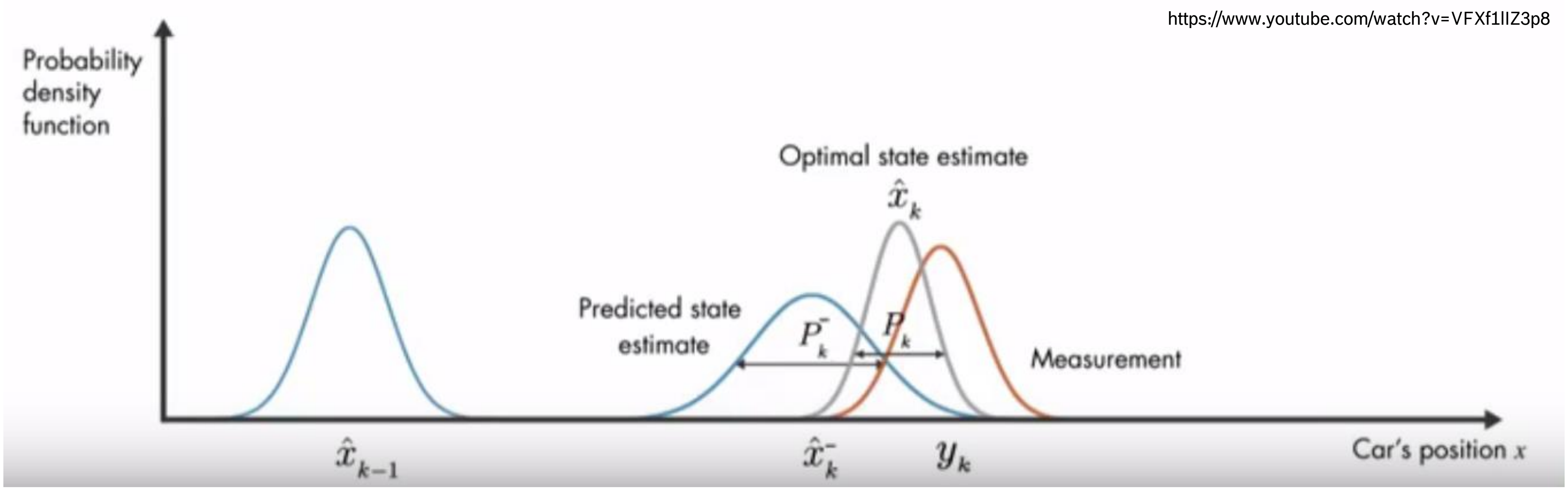




RBRO/ESA1 | 30/08/2017

# KALMAN FILTER
## Vehicle tracking - solution

https://www.youtube.com/watch?v=VFXf1lIZ3p8





RBRO/ESA1 | 30/08/2017

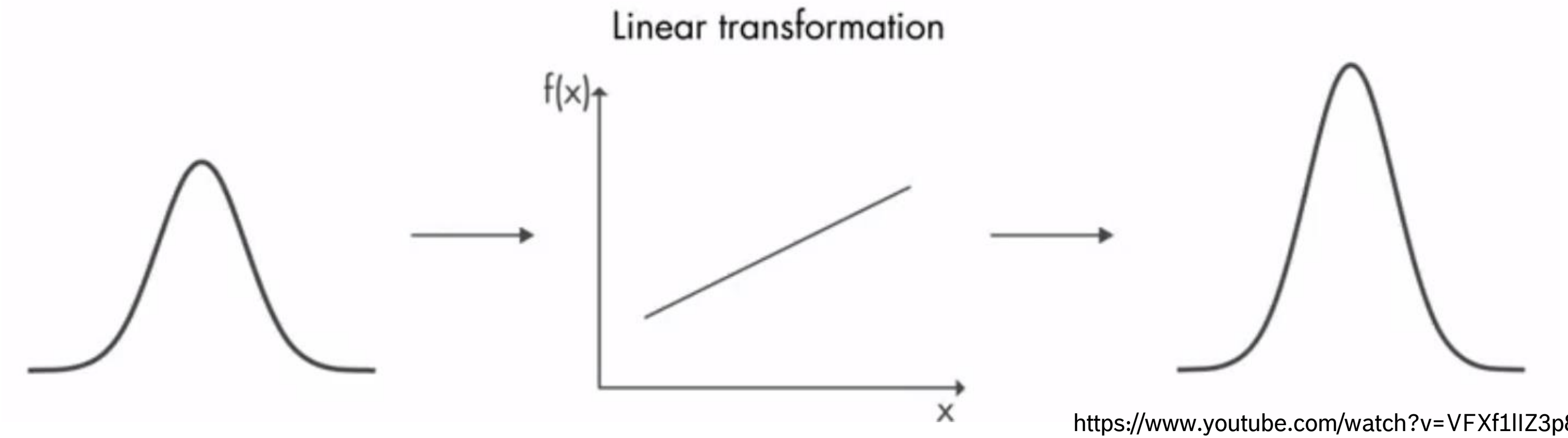
BOSCH

# KALMAN FILTER
## Principal drawback

- Kalman filter assume that the model of the system is linear

$$x_{t+1} = A_t x_t + B_t u_t$$
$$y_t = H_t x_t$$

- If we map an Gaussian distribution with a linear model we obtain another Gaussian distribution




https://www.youtube.com/watch?v=VFXf1lIZ3p8

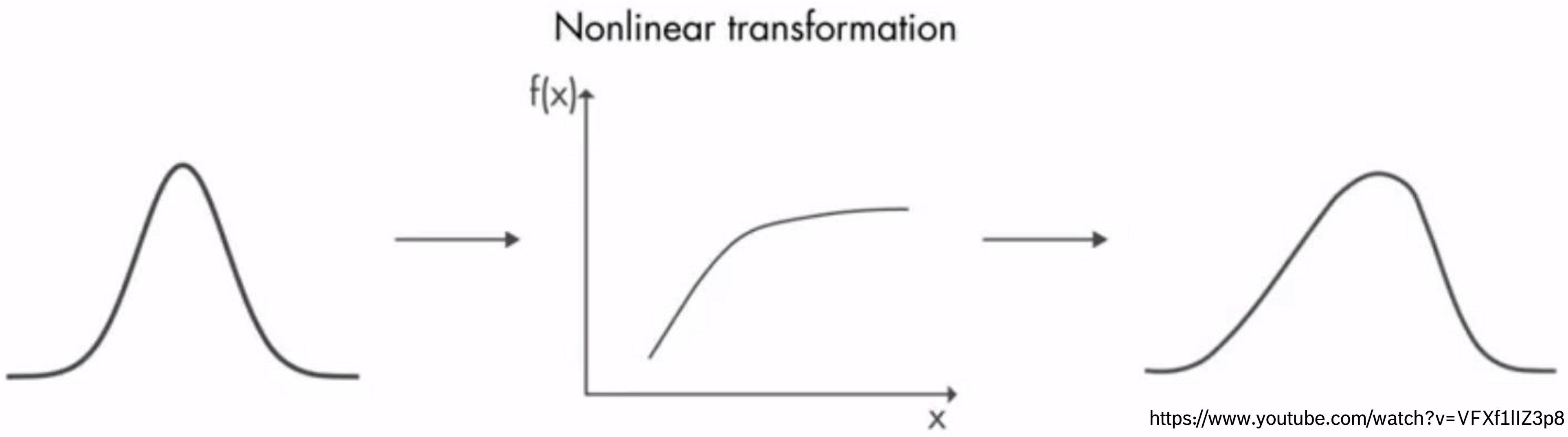


BOSCH

# KALMAN FILTER
## Principal drawback

- If model is nonlinear the Gaussian distribution is distorted

$$x_{t+1} = f(x_t, u_t)$$
$$y_t = g(x_t)$$

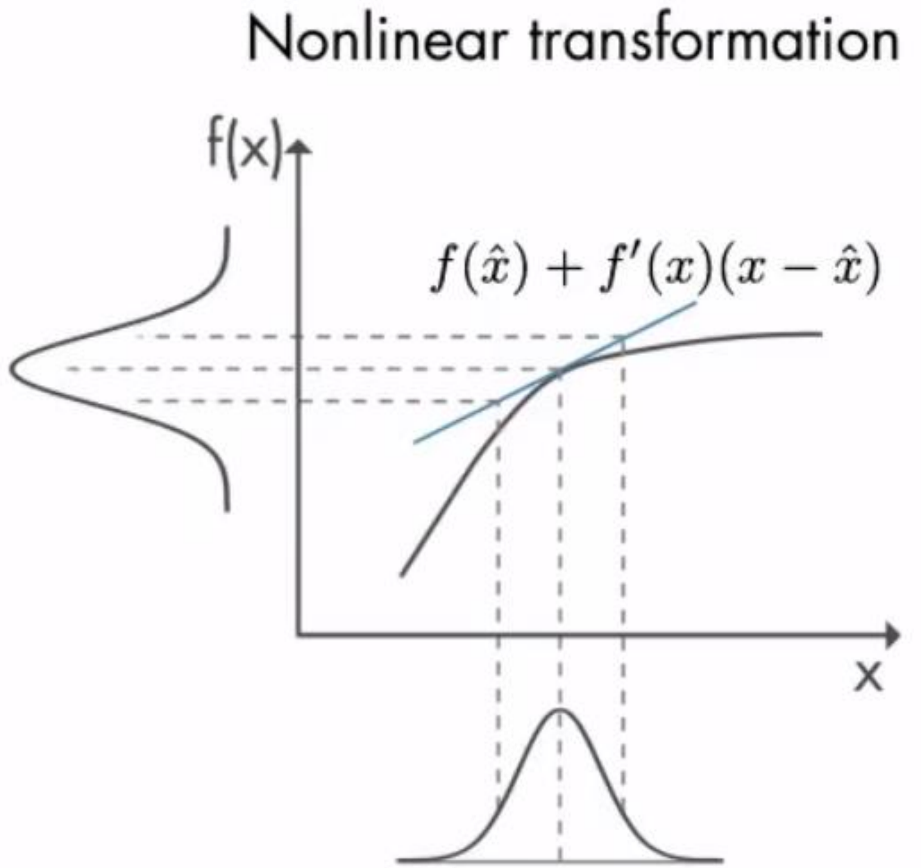



https://www.youtube.com/watch?v=VFXf1IIZ3p8

# EXTENDED KALMAN FILTER

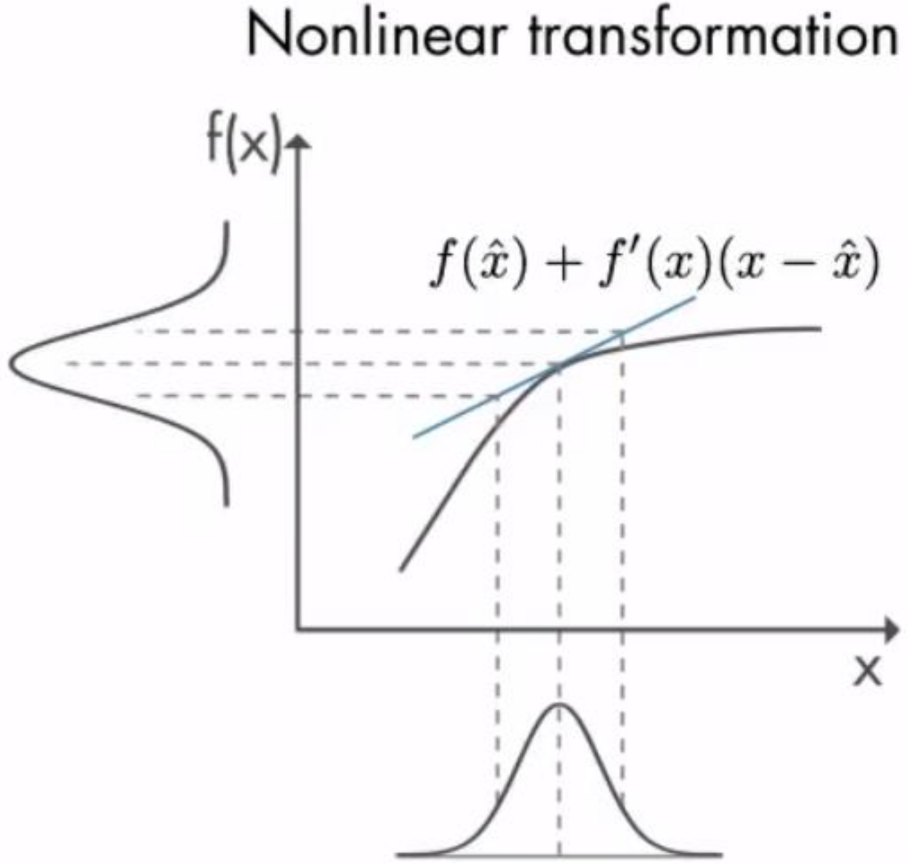
# EXTENDED KALMAN FILTER

## Model linearization




https://www.youtube.com/watch?v=VFXf1lIZ3p8

# EXTENDED KALMAN FILTER
## Model linearization




https://www.youtube.com/watch?v=VFXf1lIZ3p8

$$f(x) = f(x_k) + \frac{\partial f}{\partial x}(x_k - x)$$

For multidimensional function with multivariable input, here we have the Jacobian matrix



BOSCH

# EXTENDED KALMAN FILTER
## Model linearization

$$x_{t+1} = f(x_t, u_t)$$
$$y_t = g(x_t)$$

$$\Rightarrow \begin{cases} F_K = \dfrac{\partial f}{\partial x}\Big|_{\hat{x}_{k-1},\hat{u}_k} \\ \\ G_K = \dfrac{\partial g}{\partial x}\Big|_{\hat{x}_k} \end{cases}$$

Jacobians

# EXTENDED KALMAN FILTER

## Jacobian Matrix

$$\begin{cases} F_K = \dfrac{\partial f}{\partial x}\Big|_{\hat{x}_{k-1},\hat{u}_k} \\[2em] G_K = \dfrac{\partial g}{\partial x}\Big|_{\hat{x}_k} \end{cases}$$

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial f_m}{\partial x_1} & \cdots & \dfrac{\partial f_m}{\partial x_n} \end{bmatrix}$$

# EXTENDED KALMAN FILTER

## Modified Kalman Algorithm

### Prediction

$$\hat{x}_t^- = f(\hat{x}_{t-1}, u_{t-1})$$

$$P_t^- = F P_{t-1} F^T + Q_t$$

### Update

$$\hat{x}_t = \hat{x}_t^- + K_t(y_t - h(\hat{x}_t^-))$$

$$K_t = \frac{P_t^- G^T}{G P_t^- G^T + R_t}$$

$$P_t = (I - K_t G) P_t^-$$

At each step, Jacobians must be recomputed !!




BOSCH

# EXTENDED KALMAN FILTER
## Modified Kalman Algorithm - drawbacks

- Jacobian matrix is difficult to compute analytically
- Numerical computation of Jacobian has a high computational cost
- If system has hard nonlinear parts the first order Taylor approximation fails
- If system has hard nonlinear parts, the kalman filter is not an optimal approach

# References

- https://ocw.mit.edu/courses/mechanical-engineering/2-160-identification-estimation-and-learning-spring-2006/lecture-notes/lecture_5.pdf
- https://ocw.mit.edu/courses/mechanical-engineering/2-160-identification-estimation-and-learning-spring-2006/lecture-notes/lecture_6.pdf
- https://ocw.mit.edu/courses/mechanical-engineering/2-160-identification-estimation-and-learning-spring-2006/lecture-notes/lecture_8.pdf

# THANK YOU FOR YOUR ATTENTION