Event-Data-Recorder for Motorcycles

Low-Cost Accident Reconstruction without GPS

K. Hug[1], R. Filliger[1], Y. Stebler[2], N. Munzinger[1]

[1]**Berner Fachhochschule Technik und Informatik**
Fachbereich Automobiltechnik
Institut für Energie und Mobilität

[2]**École Polytechnique Fédérale de Lausanne**
EPFL-ENAC-IIE
Geodetic Engineering Laboratory TOPO

Financially supported in parts by AXA-Versicherungen and CTI-CH.

Goal: EDR/EA

Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

Goal: EDR/EA

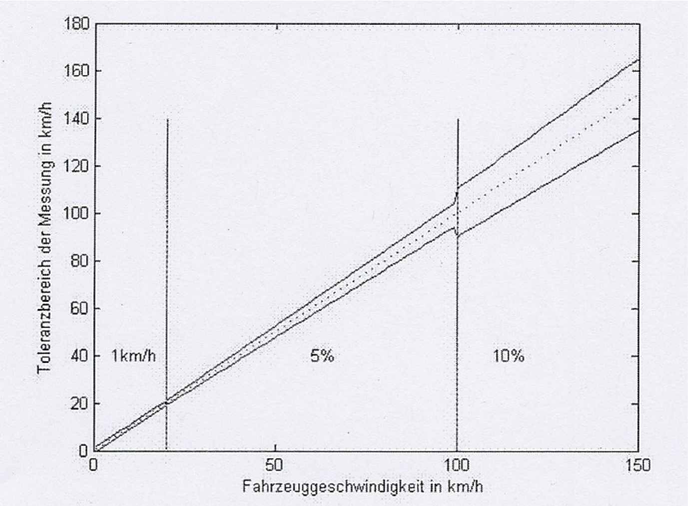
Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

...under the constraints

- **precise**
- do not use GPS
- low cost
- autarkic and easy to fix on the motorcycle
- valid for a large class of motorcycles

precision



180
160
140
120
100
80
60
40
20
0
0
50
100
150
1km/h
5%
10%
Toleranzbereich der Messung in km/h
Fahrzeuggeschwindigkeit in km/h

Goal: EDR/EA

Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

...under the constraints

- **precise**
- **do not use GPS**
- low cost
- autarkic and easy to fix on the motorcycle
- valid for a large class of motorcycles

no GPS

- **you have privacy rights on your global position!**

no GPS

- **you have privacy rights on your global position!**
-
- **uncertain availability of GPS signals**

no GPS

- **you have privacy rights on your global position!**
-
- **uncertain availability of GPS signals**
-
- **low sampling frequency ($\approx 1Hz$)**

Goal: EDR/EA

Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

...under the constraints

- **precise**
- **do not use GPS**
- **low cost**
- autarkic and easy to fix on the motorcycle
- valid for a large class of motorcycles

costs

- **fabrication costs** < 100**CHF**

costs

- **fabrication costs** < 100**CHF**
-
- **fixation costs** < 100**CHF**

Goal: EDR/EA

Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

...under the constraints

- **precise**
- **do not use GPS**
- **low cost**
- **autarkic and easy to fix on the motorcycle**
- valid for a large class of motorcycles

autarkic & easy to fix

- **no access to electronic and mechanic pieces**

autarkic & easy to fix

- **no access to electronic and mechanic pieces**
-
- **“plug and play”**

autarkic & easy to fix

- **no access to electronic and mechanic pieces**
-
- **“plug and play”**
-
- **esthetics**

Goal: EDR/EA

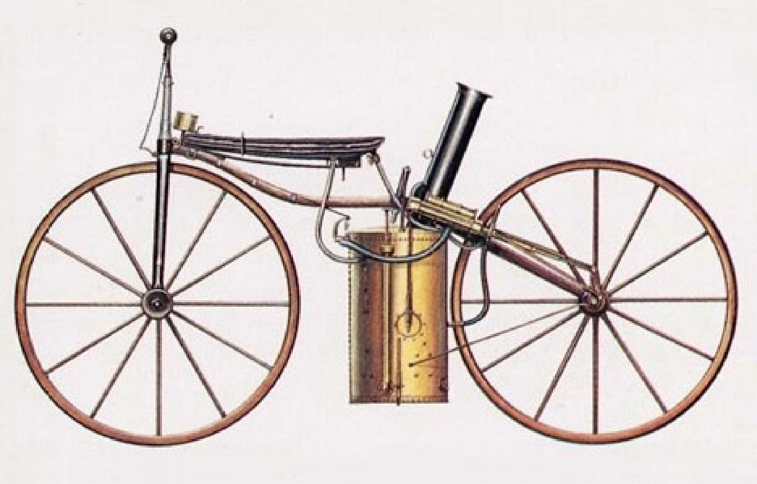
Develop...

Event Data Recorder and associated **off-line Event-Analyzer** for **Motorcycles**.

...under the constraints

- **precise**
- **do not use GPS**
- **low cost**
- **autarkic and easy to fix on the motorcycle**
- **valid for a large class of motorcycles**

valid for different motorcycles



valid for different motorcycles

valid for different motorcycles

Motivation

- **improve claim evaluation** (...deliver objective data complementary to classical accident reconstruction).

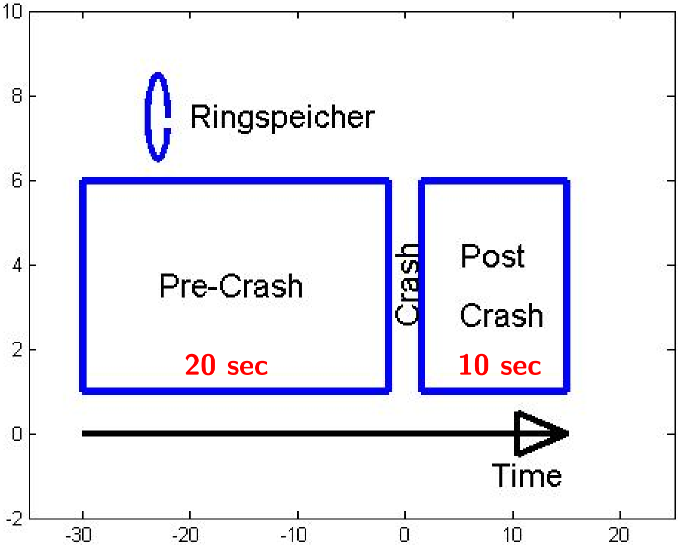
Motivation

- **improve claim evaluation** (...deliver objective data complementary to classical accident reconstruction).
- **speed up claim evaluation**

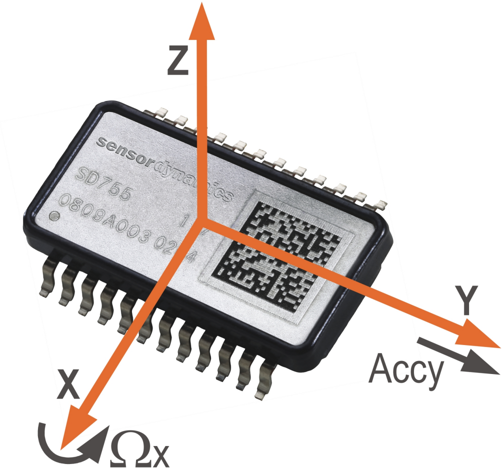
Motivation

- **improve claim evaluation** (...deliver objective data complementary to classical accident reconstruction).
- **speed up claim evaluation**
- **explore selective effect.**

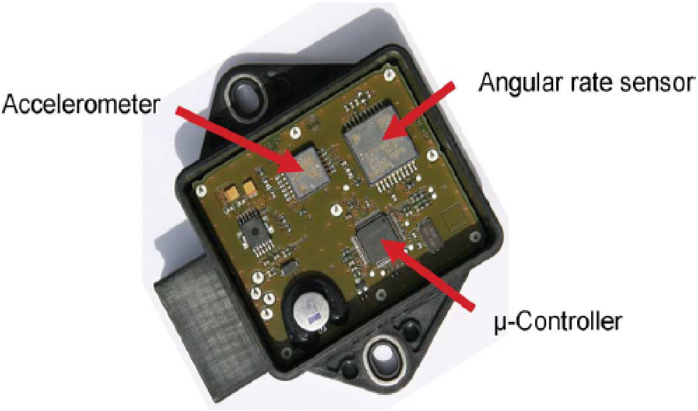
Data from 3 phases



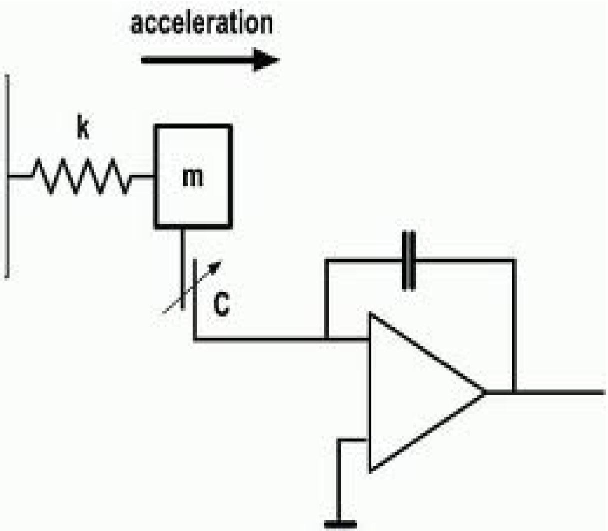
MEMS-Inertial Measurement Unit (IMU)



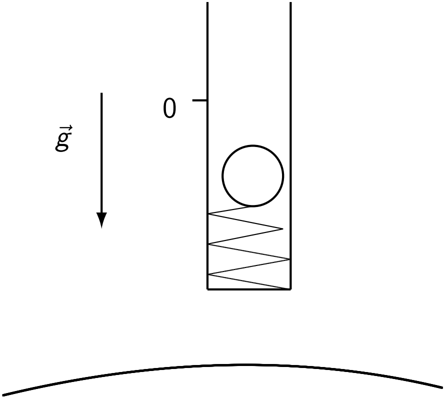
MEMS-IMU



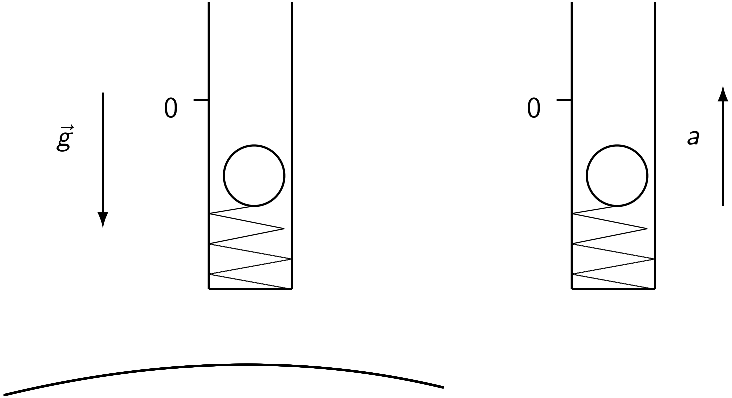
Accelerometer



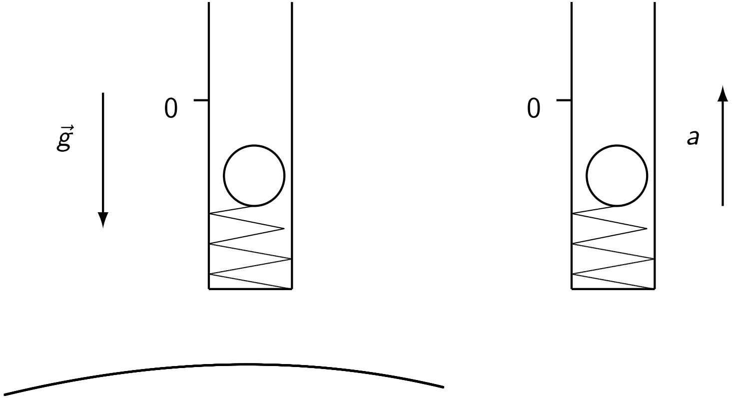
Accelerometer



Accelerometer



Accelerometer



$$\vec{f} = \vec{a} - \vec{g}$$

Speed: integrate inertial measurements (IM)

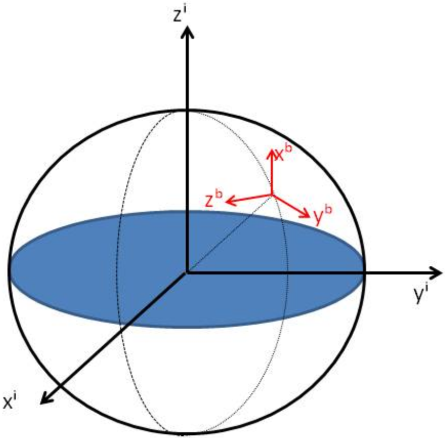
measure specific forces: $\vec{f} = \vec{a} - \vec{g}$
and use basic Newtonian physics:

$$\dot{\vec{v}} = \vec{a}$$

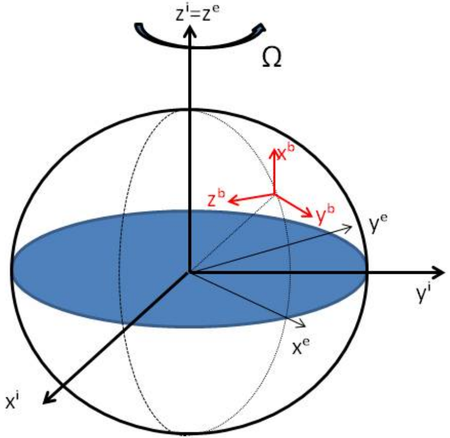
valid in inertial coordinate system!

Speed: integrate inertial measurements (IM)

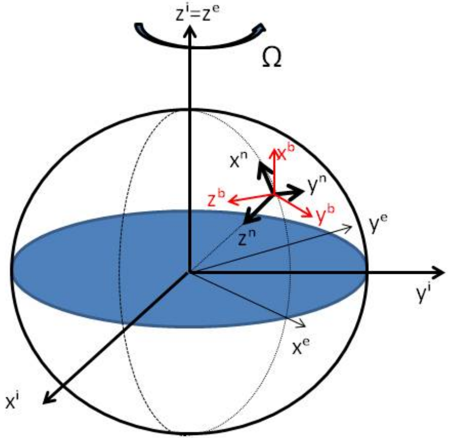
$$\dot{\vec{v}}^i_{ib} = \vec{a}^i_{ib} = \vec{f}^i_{ib} + \vec{g}^i$$



earth fixed frame



navigation frame



Speed: integrate inertial measurements (IM)

measure accelerations $\vec{a}$ (**resp., specific forces:** $\vec{f} = \vec{a} - \vec{g}$)
and use basic Newtonian physics:

$$\dot{\vec{v}}_{ib}^{i} = \vec{a}_{ib}^{i} = \vec{f}_{ib}^{i} + \vec{g}^{i}$$

valid in inertial coordinate system!

Speed: integrate inertial measurements (IM)

measure accelerations $\vec{a}$ (resp., specific forces: $\vec{f} = \vec{a} - \vec{g}$) and use basic Newtonian physics:

$$\dot{\vec{v}}^i_{ib} = \vec{a}^i_{ib} = \vec{f}^i_{ib} + \vec{g}^i$$

valid in inertial coordinate system!
In navigation frame:

$$\dot{\vec{v}}^n_{eb} = \mathbf{C}^n_b \vec{f}^b_{ib} - (2\omega^n_{ie} + \omega^n_{en}) \times \vec{v}^n_{eb} + \vec{g}^n$$

Strapdown-Solution with MEMS-IMU

$$\dot{\vec{v}}_{eb}^{n} = \mathbf{C}_b^n \vec{f}_{ib}^{b} - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times \vec{v}_{eb}^{n} + \vec{g}^{n} \tag{1}$$

Strapdown-Solution with MEMS-IMU

$$\dot{\vec{v}}^n_{eb} = \mathbf{C}^n_b \vec{f}^b_{ib} - (2\omega^n_{ie} + \omega^n_{en}) \times \vec{v}^n_{eb} + \vec{g}^n \quad (1)$$

$$\mathbf{C}^n_b = \begin{pmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix}$$

$c\alpha := cos(\alpha), s\alpha := sin(\alpha)$

Strapdown-Solution with MEMS-IMU

$$\dot{\vec{v}}^n_{eb} = \mathbf{C}^n_b \vec{f}^b_{ib} - (2\omega^n_{ie} + \omega^n_{en}) \times \vec{v}^n_{eb} + \vec{g}^n \quad (1)$$

$$\mathbf{C}^n_b = \begin{pmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix}$$

$c\alpha := cos(\alpha), s\alpha := sin(\alpha)$

$$\begin{aligned} \dot{\phi} &= (\omega^b_{nb,y} s\phi + \omega^b_{nb,z} c\phi) \tan\theta + \omega^b_{nb,x} \\ \dot{\theta} &= (\omega^b_{nb,y} c\phi - \omega^b_{nb,z} s\phi) \\ \dot{\psi} &= (\omega^b_{nb,y} s\phi + \omega^b_{nb,z} c\phi)/c\theta \end{aligned}$$

Strapdown-Solution with MEMS-IMU

$$\dot{\vec{v}}_{eb}^{n} = \mathbf{C}_b^n \vec{f}_{ib}^{b} - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times \vec{v}_{eb}^{n} + \vec{g}^{n} \quad (1)$$

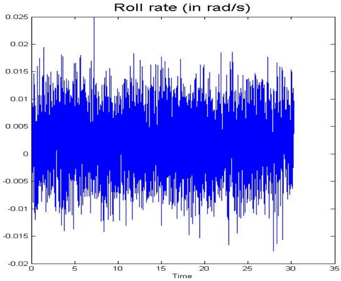
$$\mathbf{C}_b^n = \begin{pmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix}$$

$c\alpha := cos(\alpha), s\alpha := sin(\alpha)$

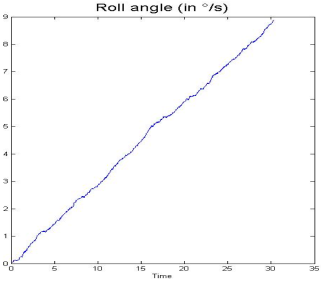
$$\begin{aligned} \dot{\phi} &= (\omega_{nb,y}^{b} s\phi + \omega_{nb,z}^{b} c\phi) \tan\theta + \omega_{nb,x}^{b} \\ \dot{\theta} &= (\omega_{nb,y}^{b} c\phi - \omega_{nb,z}^{b} s\phi) \\ \dot{\psi} &= (\omega_{nb,y}^{b} s\phi + \omega_{nb,z}^{b} c\phi)/c\theta \end{aligned}$$

$$\vec{\omega}_{nb}^{b} = \vec{\omega}_{ib}^{b} - \mathbf{C}_n^b(\vec{\omega}_{ie}^{n} + \vec{\omega}_{en}^{n})$$

IMU drift

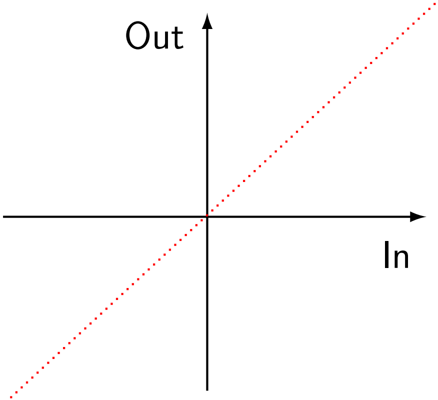


Roll rate (in rad/s)
0.025
0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
-0.02
0 5 10 15 20 25 30 35
Time

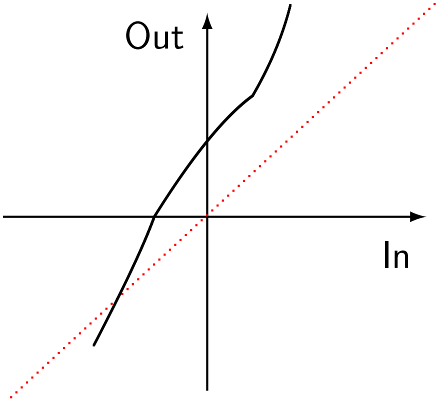


Roll angle (in °/s)
9 8 7 6 5 4 3 2 1 0
0 5 10 15 20 25 30 35
Time

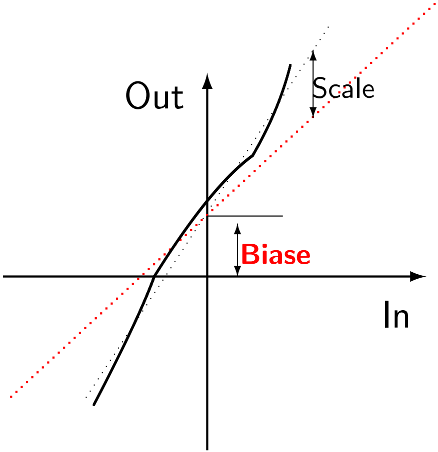
IMU error model



IMU error model



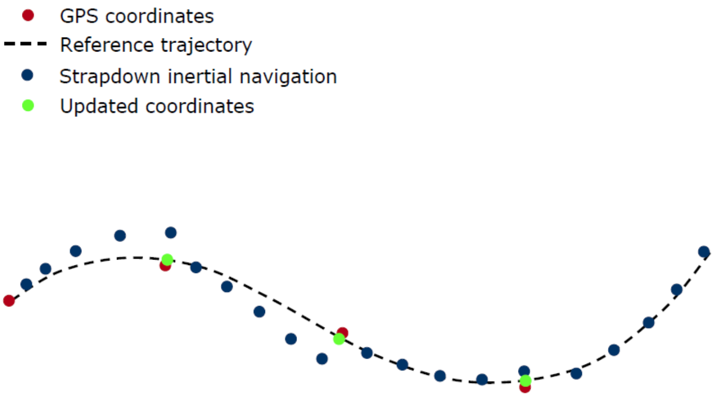
IMU error model



$$\tilde{f}_{ib}^{b} = s_{acc} \cdot \omega_{ib}^{b} + b_a + Noise_a$$

$$\tilde{\omega}_{ib}^{b} = s_{gyro} \cdot \omega_{ib}^{b} + b_\omega + Noise_\omega$$

IMU error-correction



autarkic, low-cost EDR without GPS

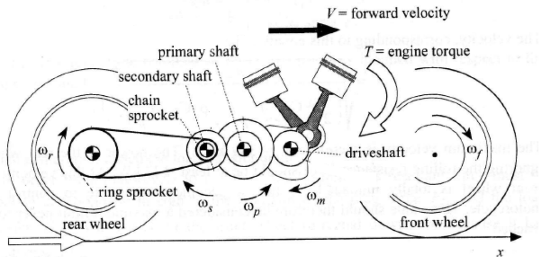
- without GPS $\Rightarrow$ search for efficient **“Navigation aid”**

Navigation aid

- **drift-free** Navigation aid for speed $\|\vec{V}\|$

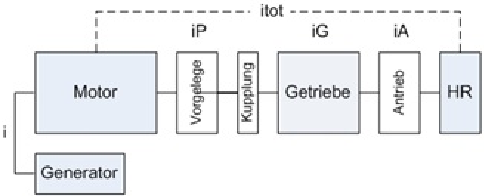
Navigation aid

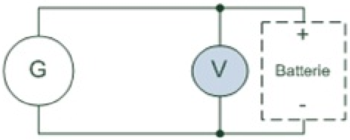
- **drift-free** Navigation aid for speed $\|\vec{V}\|$



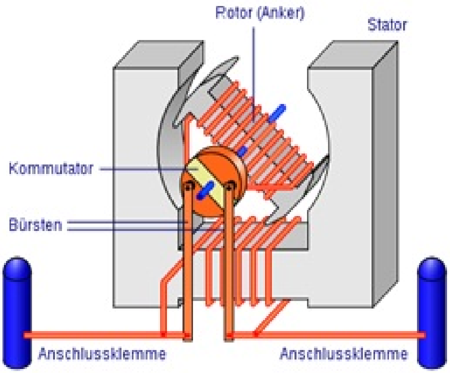
Motor-spin estimation may replace **wheel-spin** measurement if we know the engaged **gear state**.

Navigation aid

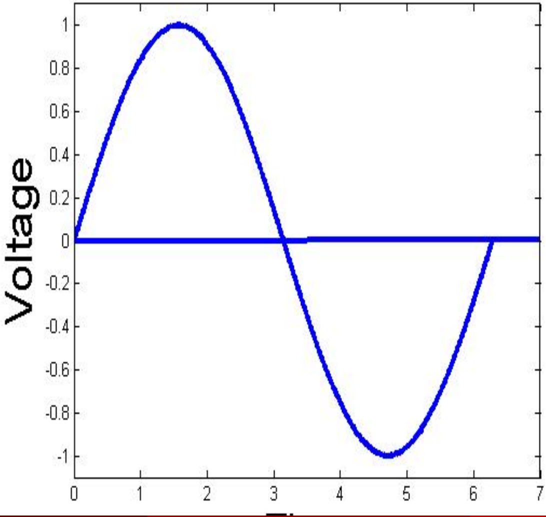




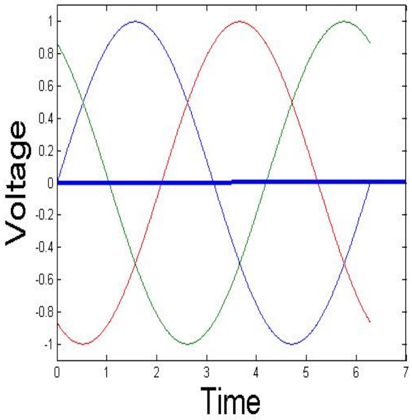
Navigation aid: Generator



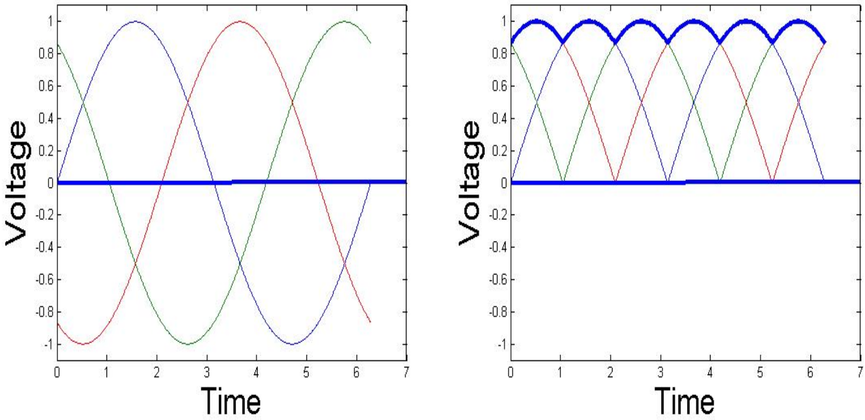
Navigation aid: ripple



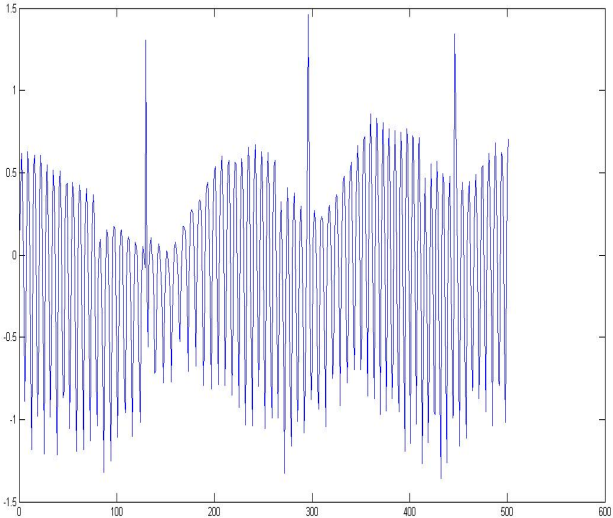
Navigation aid: ripple



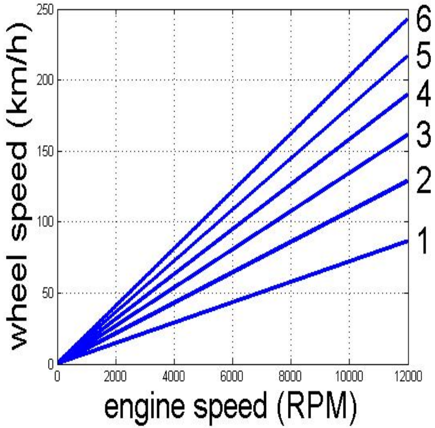
Navigation aid: ripple



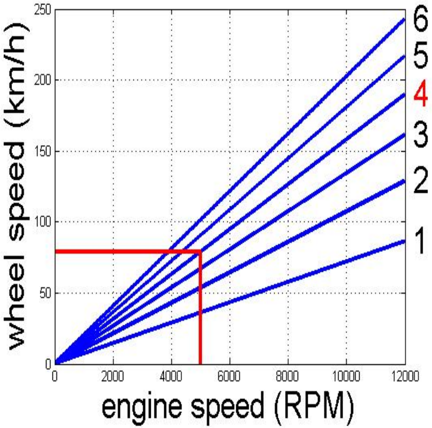
Navigation aid: ripple



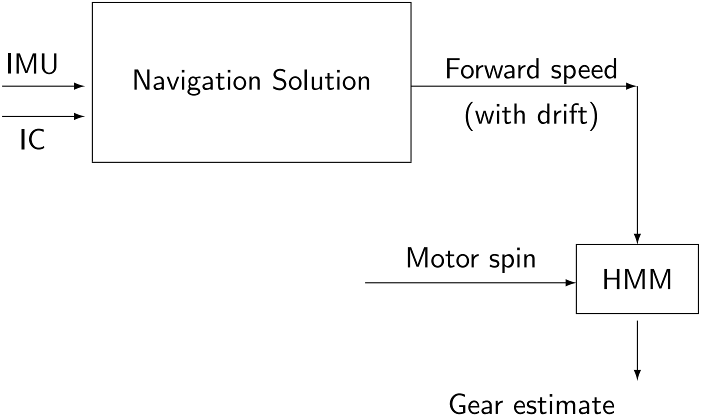
Navigation aid



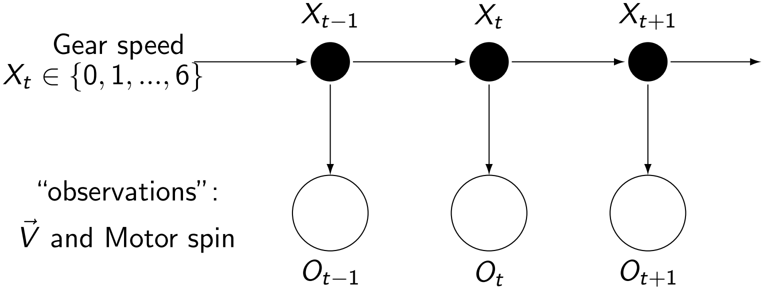
Navigation aid



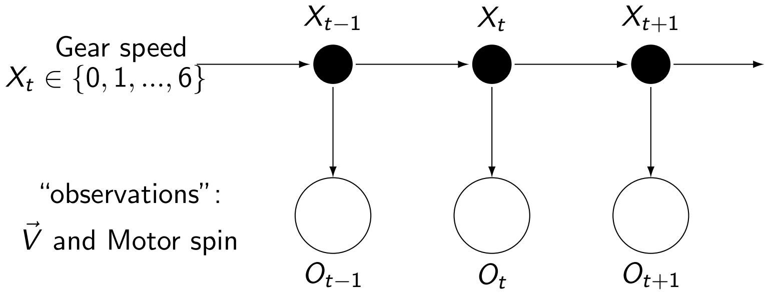
Speed estimation



Hidden Markov Model

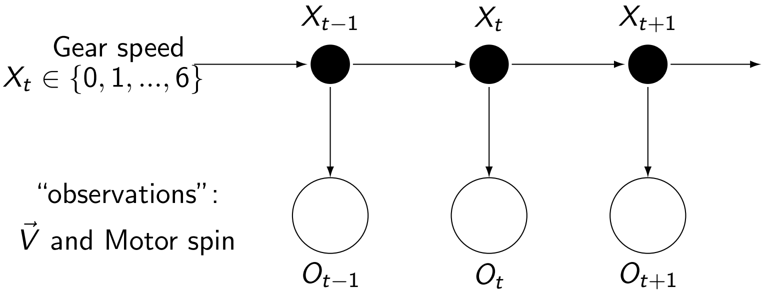


Hidden Markov Model



DBN: Dynamic Bayesian Network

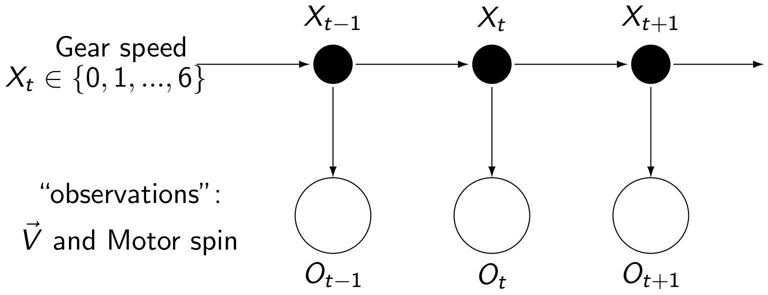
Hidden Markov Model



DBN: Dynamic Bayesian Network

$$P(X_0 = i), \;\; P(X_{t+1} = j \mid X_t = i), \;\; P(O_t \mid X_t)$$

Hidden Markov Model

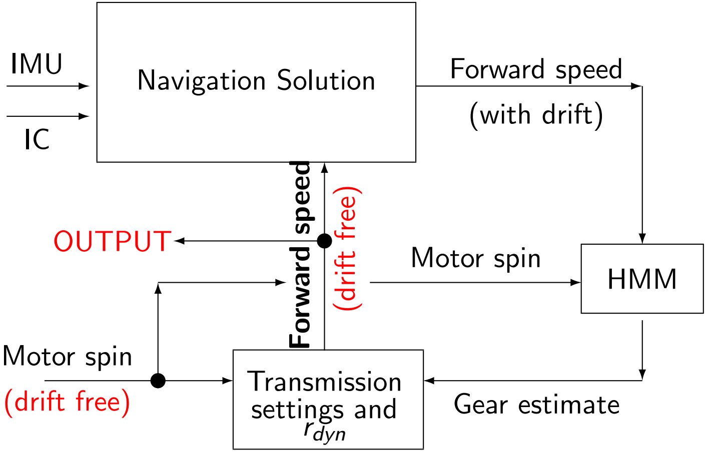


DBN: Dynamic Bayesian Network

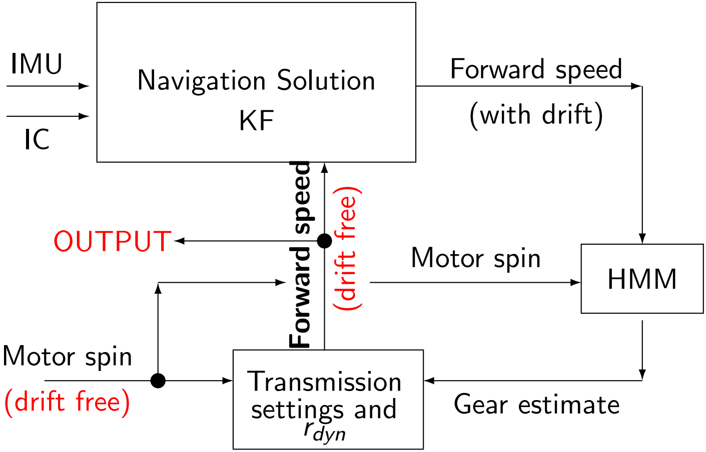
$$P(X_0 = i), \;\; P(X_{t+1} = j \mid X_t = i), \;\; P(O_t \mid X_t)$$

$$\Longrightarrow P(X_t \mid O_t), \quad \hat{x}_t = \mathtt{arg}_i \max \left\{ P(X_t = i \mid O_t) \right\}$$

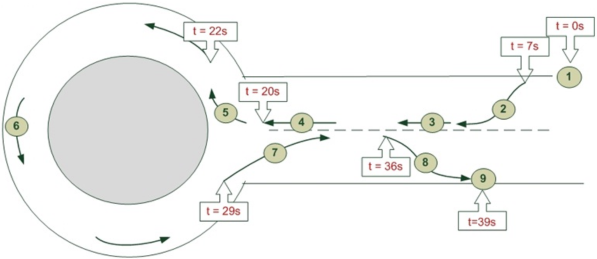
Speed estimation



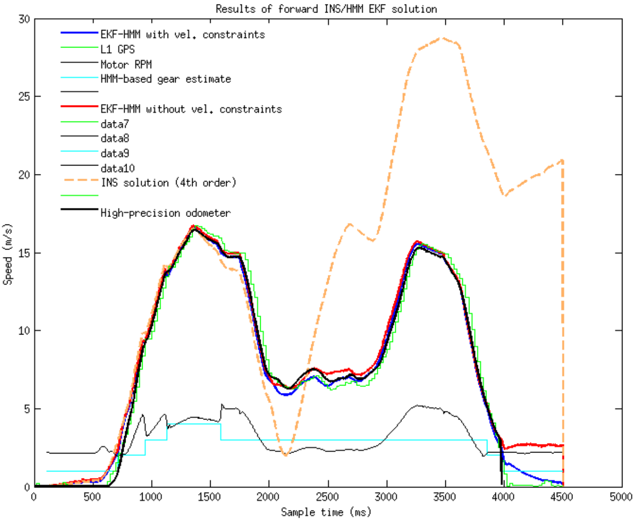
Speed estimation



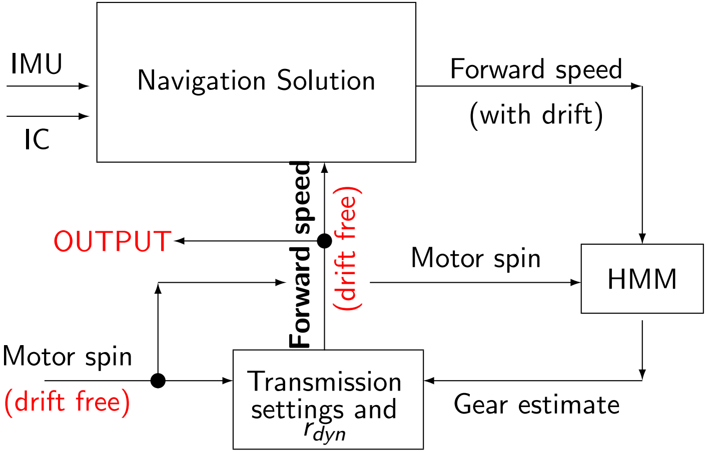
Trajectory



Speed reconstruction



Sensor Fusion



Sensor Fusion

$$\begin{aligned} \dot{v} &= f(v) + Noise \\ o &= h(v) + Noise \end{aligned}$$

Sensor Fusion

$$\begin{aligned}\textbf{Process:}\quad \dot{v} &= f(v) + Noise \\ \textbf{Measurement:}\quad o &= h(v) + Noise\end{aligned}$$

Kalman Filter

$$\begin{aligned}\textbf{Process:}\ \dot{x}(t) &= F(t)x(t) + w(t), \quad w(t) \sim \mathcal{N}(0, Q(t))\\ \textbf{Measurement:}\ o(t) &= H(t)x(t) + v(t), \quad v(t) \sim \mathcal{N}(0, R(t))\end{aligned}$$

state estimate and covariance of estimation error at time t

$$\hat{x}_t := E(x_t \mid \{o_s : 0 \leq s \leq t\}) \tag{2}$$

$$P(t) := E((x_t - \hat{x}_t)(x_t - \hat{x}_t)^T) \tag{3}$$

minimizes L_2 **norm of estimation error**

$$E((x_t - \hat{x}_t)^T(x_t - \hat{x}_t))$$

Kalman Filter

$$\begin{aligned} \textbf{Process: } \dot{x}(t) &= F(t;x) + w(t), \quad w(t) \sim \quad (0, Q(t)) \\ \textbf{Measurement: } o(t) &= H(t;x) + v(t), \quad v(t) \sim \quad (0, R(t)) \end{aligned}$$

state estimate and covariance of estimation error at time t

$$\hat{x}_t := E(x_t \mid \{o_s : 0 \leq s \leq t\}) \quad (4)$$

$$P(t) := E((x_t - \hat{x}_t)(x_t - \hat{x}_t)^T) \quad (5)$$

minimizes L_2 norm of estimation error

$$E((x_t - \hat{x}_t)^T(x_t - \hat{x}_t))$$

Kalman Filter

$$\begin{aligned}
\dot{\hat{x}}_s &= F(s)\hat{x}_s + K(s)\left[o_s - H(s)\hat{x}_s\right], \\
& K = PH^T R^{-1}, \\
\dot{P} &= FP + PF^T + Q - KRK^T,
\end{aligned}$$

Smoothing

$$\begin{aligned} dx_s &= F(x_s)ds + dw_s, \quad E(w_s w_t^t) = Q(s)\delta(t-s) \\ do_s &= H(x_s)ds + dv_s, \quad E(v_s v_t^t) = R(s)\delta(t-s) \end{aligned}$$

$$\hat{\hat{x}}_s = E(x_s \mid \sigma\{o_t : 0 \leq t \leq T\})$$

(off-line data-processing)

Smoothing

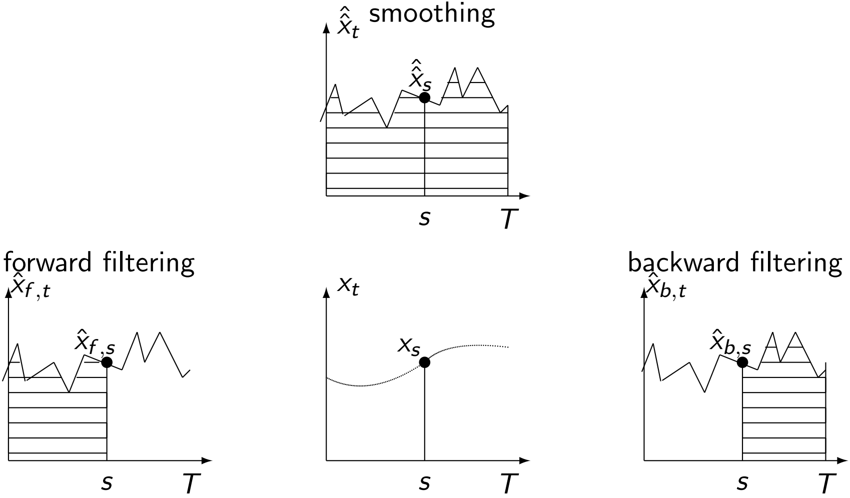
$$
\begin{aligned}
dx_s &= F(x_s)ds + dw_s, \quad E(w_s w_t^t) = Q(s)\delta(t-s) \\
do_s &= H(x_s)ds + dv_s, \quad E(v_s v_t^t) = R(s)\delta(t-s)
\end{aligned}
$$

$$
\hat{\hat{x}}_s = E(x_s \mid \sigma\{o_t : 0 \leq t \leq T\})
$$

(off-line data-processing)

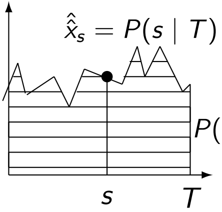
$$
E((x_s - \hat{\hat{x}}_s)^T (x_s - \hat{\hat{x}}_s)) \leq E((x_s - \hat{x}_s)^T (x_s - \hat{x}_s))
$$

Smoothing



$\hat{\hat{x}}_t$ smoothing
$\hat{\hat{x}}_s$
s
T
forward filtering
$\hat{x}_{f,t}$
$\hat{x}_{f,s}$
s
T
x_t
x_s
s
T
backward filtering
$\hat{x}_{b,t}$
$\hat{x}_{b,s}$
s
T

State Estimation



$$\hat{\hat{x}}_s = P(s \mid T)\left(P_f^{-1}(s)\hat{x}_{f,s} + P_b^{-1}(s)\hat{x}_{b,s}\right)$$

$$P(s \mid T)^{-1} = P_f^{-1}(s) + P_b^{-1}(s)$$

forward filtering

$$\dot{\hat{x}}_{f,s} = F(\hat{x}_{f,s}) + K_f(y - H(\hat{x}_{f,s}))$$
$$K_f = P_f H_s R^{-1}$$
$$\dot{P}_f = FP_f + P_f F^t + Q - K_f R K_f^t$$

backward filtering

$$\dot{\hat{x}}_{b,s} = F(\hat{x}_{b,s}) + K_b(y - H(\hat{x}_{b,s}))$$
$$K_b = P_b H_s R^{-1}$$
$$\dot{P}_b = FP_b + P_b F^t + Q - K_b R K_b^t$$