Noise covariance matrix estimation with subspace model identification for Kalman filtering

V. Mussot[1,2], G. Mercère[1] (speaker), T. Dairay[2]
(1) Poitiers University and (2) Michelin

University of Poitiers

GT Identification meeting
12 May 2022

Outline

Université de Poitiers

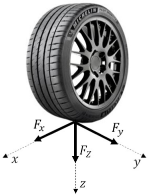
MOTIVATIONS AND FRAMEWORK

PhD thesis with Michelin

- Future autonomous vehicles will require well-developed Advanced Driver Assistance Systems (ADAS) to assist human beings in driving.
- One path chosen by Michelin for ADAS improvement consists in providing the ADAS with information related to the state of the road.
- Such information is included in the grip potential quantity.
- Benefits for passenger security (to name a few) are
  - detection of roads with low-grip area,
  - evaluation of the driving conditions,
  - reduction of the impact of rear end collisions.

Université de Poitiers

Problem formulation



- The grip potential is

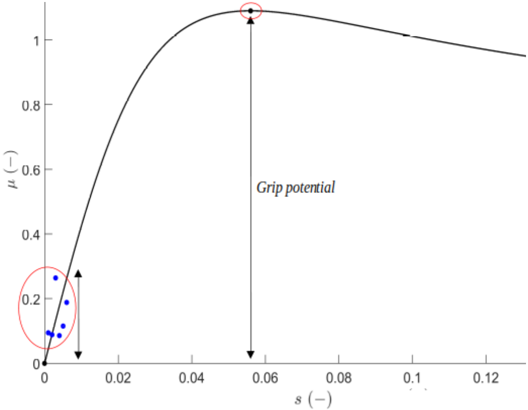
$$\mu_{\max} = \max\left(\frac{\sqrt{F_x^2 + F_y^2}}{F_z}\right),$$

i.e., the maximum effort a tire can generate before sliding on the road.

Problem formulation (cont'd)

Problem

Estimate the grip potential under standard driving conditions from sensors fitted on production vehicles.



Problem formulation (cont'd)

- Getting (noisy) data requires to measure the friction μ and the slip ratio s.
- No dedicated sensors exist on production vehicles.
- These signals must be estimated knowing that, for the longitudinal dynamics,

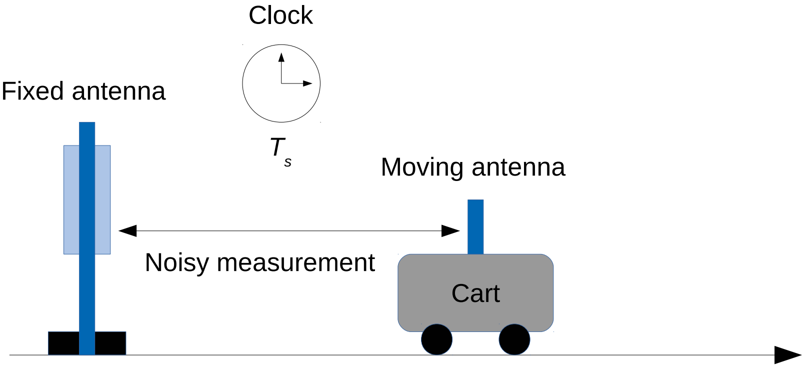
$$\mu = \frac{F_x}{F_z},$$
$$s = \frac{\omega R_{\text{rol}} - v_x}{\max\left(\omega R_{\text{rol}} - v_x\right)}.$$

- A Kalman filter is suggested to reconstruct the components of μ and s accurately.

KALMAN FILTERING: A REMINDER

Toy example

- Let us assume we want[1] to determine from remote noisy measurements the position and speed (state) of a cart moving straightforward.

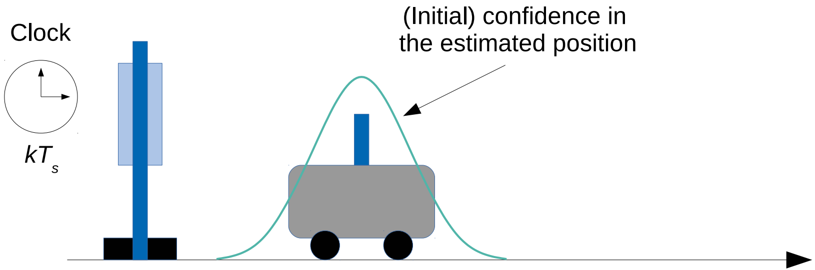


[1] See *Understanding the basis of the Kalman filter via a simple and intuitive derivation*, R. Faragher, IEEE Signal Processing Magazine, 2012.

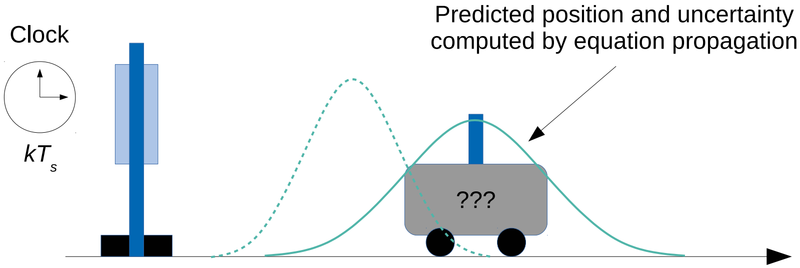
Toy example (cont'd)

- We need
  - a model of the cart dynamics,
  - a model of the measuring process,
  - a description of the noise and uncertainties acting on the system.
- Because we get new measurements every T_s, most of the Kalman filters
  - are based on dynamical systems discretized in the time domain,
  - can be updated recursively (only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state).
- Beside estimating the state vector recursively, the Kalman filters propagate and update its uncertainty as soon as new noisy measurements are available.

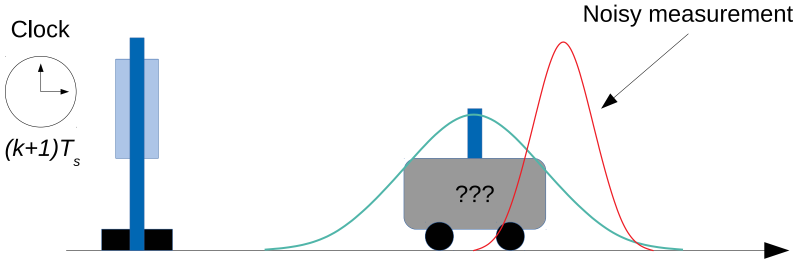
Toy example (cont'd)



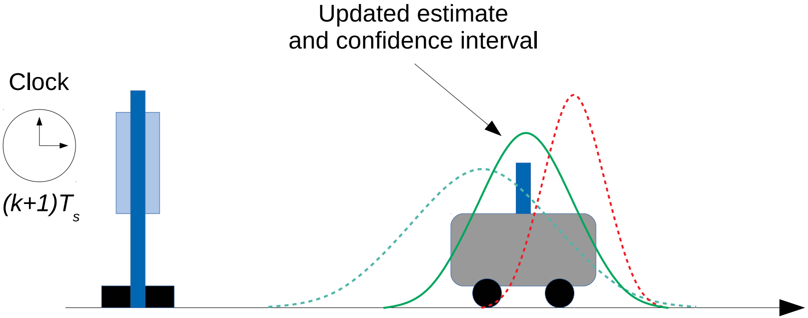
Toy example (cont'd)



Toy example (cont'd)



Toy example (cont'd)



Discrete time linear Kalman filter

- The standard discrete time linear Kalman filter considers models evolving as follows

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{G}_k \boldsymbol{u}_k + \boldsymbol{w}_k,$$
$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k,$$

  where, for $k \in \mathbb{T} = \{0, \cdots, N-1\}$,
  - $\boldsymbol{v}_k \in \mathbb{R}^{n_y \times 1}$ stands for a time sample of the observation or output measurement noise sequence $(\mathbf{v}_k)_{k \in \mathbb{Z}}$,
  - $\boldsymbol{w}_k \in \mathbb{R}^{n_x \times 1}$ stands for a time sample of the process noise sequence $(\mathbf{w}_k)_{k \in \mathbb{Z}}$,
- The sequences $(\boldsymbol{v}_i)_{i \in \mathbb{T}}$ and $(\boldsymbol{w}_i)_{i \in \mathbb{T}}$ are used to describe
  - the noise acting on the real system,
  - the (in)accuracy of the model representation,
  - the confidence we have in the model and the measurements.

- Let us consider a system, the behavior of which is governed by the vector difference equation

$$\mathbf{x}_{k+1} = \boldsymbol{F}_k \mathbf{x}_k + \boldsymbol{G}_k \boldsymbol{u}_k + \mathbf{w}_k,$$

where $\boldsymbol{F}_k$, $\boldsymbol{G}_k$ and $\boldsymbol{u}_k$ are assumed to be perfectly known while the initial state and its symmetric positive definite covariance are defined as follows

$$\mathbb{E}\left\{\mathbf{x}_0\right\} = \boldsymbol{x}_0,$$

$$\mathbb{E}\left\{\left(\mathbf{x}_0 - \boldsymbol{x}_0\right)\left(\mathbf{x}_0 - \boldsymbol{x}_0\right)^\top\right\} = \boldsymbol{X}_0 \succ 0.$$

Discrete time linear Kalman filter (cont'd)

- Let us assume that the process disturbances can be described by a zero mean white noise with a finite and symmetric covariance matrix satisfying

$$\mathbb{E}\left\{\mathbf{w}_k\right\} = \mathbf{0},$$
$$\mathbb{E}\left\{\mathbf{w}_k\mathbf{w}_j^\top\right\} = \boldsymbol{W}_k\delta_{kj},\ \boldsymbol{W}_k \succ 0.$$

- Then, the uncorrected state and error covariance matrix propagate from the previous corrected estimates as follows

$$\hat{\mathbf{x}}_k^- = \boldsymbol{F}_{k-1}\hat{\mathbf{x}}_{k-1}^+ + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1},$$
$$\boldsymbol{X}_k^- = \boldsymbol{F}_{k-1}\boldsymbol{X}_{k-1}^+\boldsymbol{F}_{k-1}^\top + \boldsymbol{W}_{k-1},$$

where the notations $^-$ and $^+$ stand for "before and after new measurements".

- Once a new measurement is available, *i.e.*, once we measure

$$\mathbf{y}_k = \boldsymbol{H}_k \mathbf{x}_k + \mathbf{v}_k,$$

where $\boldsymbol{H}_k$ is known and by assuming that the output disturbances are also described by a zero mean white noise (with finite and symmetric covariance matrix) uncorrelated with the process noise, *i.e.*,

$$\mathbb{E}\left\{\mathbf{v}_k\right\} = \mathbf{0},$$
$$\mathbb{E}\left\{\mathbf{v}_k \mathbf{v}_j^\top\right\} = \boldsymbol{V}_k \delta_{kj},\ \ \boldsymbol{V}_k \succ 0,$$
$$\mathbb{E}\left\{\mathbf{v}_k \mathbf{w}_j^\top\right\} = \mathbf{0},\ \text{for all } k \text{ and } j,$$

we can first update the Kalman gain

$$\boldsymbol{K}_k = \boldsymbol{X}_k^- \boldsymbol{H}_k^\top \left(\boldsymbol{H}_k \boldsymbol{X}_k^- \boldsymbol{H}_k^\top + \boldsymbol{V}_k\right)^{-1}.$$

Discrete time linear Kalman filter (cont'd)

- The state estimate is updated as follows

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

while the error covariance estimate update satisfies

$$\boldsymbol{X}_k^+ = (\boldsymbol{I}_{n_x \times n_x} - \boldsymbol{K}_k\boldsymbol{H}_k)\,\boldsymbol{X}_k^-.$$

Discrete time linear Kalman filter (cont'd)

- The state estimate is updated as follows

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

while the error covariance estimate update satisfies

$$\boldsymbol{X}_k^+ = \left(\boldsymbol{I}_{n_x \times n_x} - \boldsymbol{K}_k\boldsymbol{H}_k\right)\boldsymbol{X}_k^-.$$

- In order to get reliable results, the Kalman filter requires
  - input and output measurements,
  - matrices $\boldsymbol{F}_k$, $\boldsymbol{G}_k$ and $\boldsymbol{H}_k$ generated from (a discretization and linearization of) the physical equations governing the system behavior,
  - values for $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$, $k \in \mathbb{T}$.

Université de Poitiers

Basic idea of our solution

Time invariant conditions

In the sequel, for $k \in \mathbb{T}$,

$$\boldsymbol{F}_k = \boldsymbol{F}, \quad \boldsymbol{G}_k = \boldsymbol{G}, \quad \boldsymbol{H}_k = \boldsymbol{H}, \quad \boldsymbol{V}_k = \boldsymbol{V}, \quad \boldsymbol{W}_k = \boldsymbol{W}.$$

- Because the matrices $\boldsymbol{V}$ and $\boldsymbol{W}$ are used to describe the confidence we have in the model and the measurements, we aim at determining them by comparing
  - the model used in the Kalman filter,
  - a model estimated from the available data sets.
- Herein, the data driven model learning solution is a subspace based model identification approach.

Notations interlude...

- For any vector $\boldsymbol{r}_k \in \mathbb{R}^{n_r \times 1}$ and parameters M, i and $\ell \in \mathbb{N}_*^+$, we define

$$\boldsymbol{r}_{i,M} = \begin{bmatrix} \boldsymbol{r}_i \\ \boldsymbol{r}_{i+1} \\ \vdots \\ \boldsymbol{r}_{i+M-1} \end{bmatrix} \in \mathbb{R}^{Mn_r \times 1},$$

$$\boldsymbol{R}_{i,M} = \begin{bmatrix} \boldsymbol{r}_i & \boldsymbol{r}_{i+1} \cdots & \boldsymbol{r}_{i+M-1} \end{bmatrix} \in \mathbb{R}^{n_r \times M},$$

and the block Hankel matrix as follows

$$\boldsymbol{R}_{i,\ell,M} = \begin{bmatrix} \boldsymbol{r}_i & \boldsymbol{r}_{i+1} & \cdots & \boldsymbol{r}_{i+M-1} \\ \boldsymbol{r}_{i+1} & \boldsymbol{r}_{i+2} & \cdots & \boldsymbol{r}_{i+M} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{r}_{i+\ell-1} & \boldsymbol{r}_{i+\ell} & \cdots & \boldsymbol{r}_{i+M+\ell-2} \end{bmatrix} \in \mathbb{R}^{\ell n_r \times M}.$$

Basic idea of our solution (cont'd)

- The subspace based model learning methods yield estimates of $\boldsymbol{x}_t$, $t \in \{f, \cdots, N-1\} \times T_s$, and $\{\boldsymbol{F}, \boldsymbol{G}, \boldsymbol{H}\}$ up to a similarity transformation.
- We can thus estimate[2]

$$\begin{bmatrix} \hat{\boldsymbol{W}}_{f,M-1} \\ \hat{\boldsymbol{V}}_{f,M-1} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{X}}_{f+1,M} \\ \boldsymbol{Y}_{f,M-1} \end{bmatrix} - \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} \\ \boldsymbol{H} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{X}}_{f,M-1} \\ \boldsymbol{U}_{f,M-1} \end{bmatrix},$$

and

$$\begin{bmatrix} \hat{\boldsymbol{V}} & \hat{\boldsymbol{S}} \\ \hat{\boldsymbol{S}}^\top & \hat{\boldsymbol{W}} \end{bmatrix} = \lim_{M \to \infty} \frac{1}{M} \begin{bmatrix} \hat{\boldsymbol{W}}_{f,M-1} \\ \hat{\boldsymbol{V}}_{f,M-1} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{W}}_{f,M-1}^\top & \hat{\boldsymbol{V}}_{f,M-1}^\top \end{bmatrix}.$$

[2] We have $M = N - f$.

SUBSPACE STATE SPACE MODEL IDENTIFICATION: A REMINDER

Problem formulation

- Subspace model learning (SML) methods mainly focus on state space models of finite dimensional linear time invariant causal and discrete time dynamical systems described by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{w}_k, \tag{1a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k. \tag{1b}$$

where $k \in \mathbb{T}$ whereas the noise sources are assumed to be realizations of zero mean white noise statically independent of the input sequence whereas

$$\mathbb{E}\left[\begin{bmatrix}\mathbf{v}_i \\ \mathbf{w}_i\end{bmatrix}\begin{bmatrix}\mathbf{v}_j^\top & \mathbf{w}_j^\top\end{bmatrix}\right] = \begin{bmatrix}\boldsymbol{V} & \boldsymbol{S} \\ \boldsymbol{S}^\top & \boldsymbol{W}\end{bmatrix}\delta_{ij}. \tag{2}$$

Problem formulation (cont'd)

- By assuming that
  - A1 the input vector sequence is quasi stationary and exciting of sufficient order,
  - A2 the pair $(\boldsymbol{A}, \boldsymbol{C})$ is observable and the pair $(\boldsymbol{A}, \begin{bmatrix}\boldsymbol{B} & \boldsymbol{V}^{1/2}\end{bmatrix})$ is reachable,

  standard SML solutions aim at estimating consistently
  - the order n_x of the system,
  - $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ up to a similarity transformation,
  - $\boldsymbol{K}$ such that an approximated minimum variance estimate of $\boldsymbol{x}_k$, $k \in \mathbb{T}$, can be reconstructed.

Innovation form

- Under Assumption A2, Kalman filtering theory proves that, for every choice of $(\mathbf{v}_i)_{i\in\mathbb{Z}}$ and $(\mathbf{w}_i)_{i\in\mathbb{Z}}$ satisfying Eq. (2), a matrix $\boldsymbol{K} \in \mathbb{R}^{n_y\times 1}$ an a zero mean white noise sequence $(\mathbf{e}_i)_{i\in\mathbb{Z}}$ exist such that the innovation form

$$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}\boldsymbol{e}_k,$$
$$\boldsymbol{y}_k = \boldsymbol{C}\hat{\boldsymbol{x}}_k + \boldsymbol{e}_k,$$

  is equivalent to Eq. (1) (same I/O behavior).
- Furthermore, the matrix $\boldsymbol{K}$ satisfies

$$\lambda_{\max}(\boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}) < 1.$$

- Standard SML solutions estimate $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{K})$ by using this innovation form.

Notations interlude... (cont'd)

- With matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ of appropriate dimensions, for $\ell \geq n_x$, the extended controllability matrix can be defined as follows

$$\boldsymbol{\Omega}_\ell(\boldsymbol{A}, \boldsymbol{B}) = \begin{bmatrix} \boldsymbol{A}^{\ell-1}\boldsymbol{B} & \cdots & \boldsymbol{A}\boldsymbol{B} & \boldsymbol{B} \end{bmatrix}.$$

- We also define the extended observability matrix

$$\boldsymbol{\Gamma}_\ell(\boldsymbol{A}, \boldsymbol{C}) = \begin{bmatrix} \boldsymbol{C}^\top & (\boldsymbol{C}\boldsymbol{A})^\top & \cdots & (\boldsymbol{C}\boldsymbol{A}^{\ell-1})^\top \end{bmatrix}^\top,$$

and the block lower triangular Toeplitz matrix

$$\boldsymbol{\Delta}_\ell(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}) = \begin{bmatrix} \boldsymbol{D} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{C}\boldsymbol{B} & \boldsymbol{D} & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{C}\boldsymbol{A}^{\ell-2}\boldsymbol{B} & \cdots & \boldsymbol{C}\boldsymbol{B} & \boldsymbol{D} \end{bmatrix}.$$

First recursions

- With f and $\ell \in \mathbb{N}^*$,

$$\begin{aligned} \boldsymbol{y}_\ell &= \boldsymbol{C}\hat{\boldsymbol{x}}_\ell + \boldsymbol{e}_\ell, \\ \boldsymbol{y}_{\ell+1} &= \boldsymbol{C}\boldsymbol{A}\hat{\boldsymbol{x}}_\ell + \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_\ell + \boldsymbol{C}\boldsymbol{K}\boldsymbol{e}_\ell + \boldsymbol{e}_{\ell+1}, \\ \boldsymbol{y}_{\ell+2} &= \cdots, \\ &\vdots \end{aligned}$$

i.e.,

$$\boldsymbol{y}_{\ell,f} = \boldsymbol{\Gamma}_f(\boldsymbol{A},\boldsymbol{C})\hat{\boldsymbol{x}}_\ell + \boldsymbol{\Delta}_f^u(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{0})\boldsymbol{u}_{\ell,f} + \boldsymbol{n}_{\ell,f},$$

with

$$\boldsymbol{n}_{\ell,f} = \boldsymbol{\Delta}_f^e(\boldsymbol{A},\boldsymbol{K},\boldsymbol{C},\boldsymbol{I}_{n_y\times n_y})\boldsymbol{e}_{\ell,f}.$$

First recursions (cont'd)

- By using all the available time shifts, we get

$$\boldsymbol{Y}_{\ell,f,M} = \boldsymbol{\Gamma}_f(\boldsymbol{A}, \boldsymbol{C})\hat{\boldsymbol{X}}_{\ell,M} + \boldsymbol{\Delta}_f^u \boldsymbol{U}_{\ell,f,M} + \boldsymbol{N}_{\ell,f,M},$$

where $M = N + 1 - \ell - f$ whereas

$$\begin{aligned} \boldsymbol{\Delta}_f^u &= \boldsymbol{\Delta}_f(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{0}), \\ \boldsymbol{N}_{\ell,f,M} &= \underbrace{\boldsymbol{\Delta}_f(\boldsymbol{A}, \boldsymbol{K}, \boldsymbol{C}, \boldsymbol{I}_{n_y \times n_y})}_{\boldsymbol{\Delta}_f^e} \boldsymbol{E}_{\ell,f,M}. \end{aligned}$$

Predictor state space form

- Because, for $k \in \mathbb{T}$,

$$\boldsymbol{e}_k = \boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k,$$

we get for $k \in \mathbb{T}$

$$\begin{aligned} \hat{\boldsymbol{x}}_{k+1} &= \tilde{\boldsymbol{A}}\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}\boldsymbol{y}_k, \\ \boldsymbol{y}_k &= \boldsymbol{C}\hat{\boldsymbol{x}}_k + \boldsymbol{e}_k, \end{aligned}$$

with

$$\tilde{\boldsymbol{A}} = \boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}.$$

Further recursions

- We can easily see that

$$\begin{aligned}\hat{\boldsymbol{x}}_\ell &= \tilde{\boldsymbol{A}}\hat{\boldsymbol{x}}_{\ell-1} + \boldsymbol{B}\boldsymbol{u}_{\ell-1} + \boldsymbol{K}\boldsymbol{y}_{\ell-1},\\ &= \tilde{\boldsymbol{A}}^2\hat{\boldsymbol{x}}_{\ell-2} + \tilde{\boldsymbol{A}}\boldsymbol{B}\boldsymbol{u}_{\ell-2} + \tilde{\boldsymbol{A}}\boldsymbol{K}\boldsymbol{y}_{\ell-2} + \boldsymbol{B}\boldsymbol{u}_{\ell-1} + \boldsymbol{K}\boldsymbol{y}_{\ell-1},\\ &\vdots\end{aligned}$$

i.e, for $p \in \mathbb{N}^*$

$$\hat{\boldsymbol{x}}_\ell = \tilde{\boldsymbol{A}}^p\hat{\boldsymbol{x}}_{\ell-p} + \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{K})\boldsymbol{y}_{\ell-p,p} + \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{B})\boldsymbol{u}_{\ell-p,p}.$$

- Because $\lambda_{\text{max}}(\boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}) < 1$,

$$\bar{\boldsymbol{x}}_\ell = \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{K})\boldsymbol{y}_{\ell-p,p} + \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{B})\boldsymbol{u}_{\ell-p,p},$$

is the the optimal linear estimate of $\hat{\boldsymbol{x}}_\ell$ (in the mean square error sense) given $\boldsymbol{y}_{\ell-p,p}$ and $\boldsymbol{u}_{\ell-p,p}$.

Data equation

- Thus, with $\ell \geq p$,

$$\hat{\boldsymbol{X}}_{\ell,M} \approx \bar{\boldsymbol{X}}_{\ell,M} = \begin{bmatrix}\boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{B}) & \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{K})\end{bmatrix} \begin{bmatrix}\boldsymbol{U}_{\ell-p,p,\ell+M-1} \\ \boldsymbol{Y}_{\ell-p,p,\ell+M-1}\end{bmatrix}$$

- A straightforward combination of the former equations lead to the data equation

$$\begin{aligned}\boldsymbol{Y}_{\ell,f,M} &= \boldsymbol{\Delta}_f^u \boldsymbol{U}_{f,\ell,M} + \boldsymbol{N}_{f,\ell,M} + \boldsymbol{\Gamma}_f(\boldsymbol{A}, \boldsymbol{C}) \\ &\times \begin{bmatrix}\boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{B}) & \boldsymbol{\Omega}_p(\tilde{\boldsymbol{A}}, \boldsymbol{K})\end{bmatrix} \begin{bmatrix}\boldsymbol{U}_{\ell-p,p,\ell+M-1} \\ \boldsymbol{Y}_{\ell-p,p,\ell+M-1}\end{bmatrix}.\end{aligned}$$

- This is nothing but a standard linear least squares minimization problem with good statistical properties because the past I/O data are uncorrelated with the future noise.

Université de Poitiers

Numerical implementation

- As any linear least squares problem, an LQ factorization is involved. More precisely,

$$\begin{bmatrix} \boldsymbol{U}_f \\ \boldsymbol{U}_p \\ \boldsymbol{Y}_p \\ \boldsymbol{Y}_f \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_{11} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{L}_{21} & \boldsymbol{L}_{22} & \boldsymbol{0} \\ \boldsymbol{L}_{31} & \boldsymbol{L}_{32} & \boldsymbol{L}_{33} \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}_1 \\ \boldsymbol{Q}_2 \\ \boldsymbol{Q}_3 \end{bmatrix}.$$

- It can be proved that

$$\lim_{N \to \infty} \boldsymbol{L}_{32} \boldsymbol{L}_{22}^{-1} \begin{bmatrix} \boldsymbol{U}_p \\ \boldsymbol{Y}_p \end{bmatrix} = \boldsymbol{\Gamma}_f(\boldsymbol{A}, \boldsymbol{C}) \hat{\boldsymbol{X}}_{f,M}.$$

Numerical implementation (cont'd)

- Via the following SVD,

$$\boldsymbol{L}_{32}\boldsymbol{L}_{22}^{-1}\begin{bmatrix}\boldsymbol{U}_p\\ \boldsymbol{Y}_p\end{bmatrix} = \boldsymbol{\mathcal{U}}\boldsymbol{\Sigma}\boldsymbol{\mathcal{V}}^\top,$$

we get an estimate of the system order n_x and

$$\hat{\boldsymbol{\Gamma}}_f(\boldsymbol{A},\boldsymbol{C}) = \boldsymbol{\mathcal{U}}(:,1:\hat{n}_x)\boldsymbol{\Sigma}^{1/2}(1:\hat{n}_x,1:\hat{n}_x),$$
$$\hat{\hat{\boldsymbol{X}}}_{f,M} = \boldsymbol{\Sigma}^{1/2}(1:\hat{n}_x,1:\hat{n}_x)\boldsymbol{\mathcal{V}}^\top(1:\hat{n}_x,:).$$

- Reliable estimates of $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{K})$ can be generated from $\hat{\boldsymbol{\Gamma}}_f(\boldsymbol{A},\boldsymbol{C})$ and $\hat{\hat{\boldsymbol{X}}}_{f,M}$ straightforwardly (up to a similarity transformation $\boldsymbol{T}$).

NOISE COVARIANCE MATRIX ESTIMATION

Similarity transformation

- For our noise covariance matrix estimation problem, we need to have $\hat{\hat{\boldsymbol{X}}}_{f,M}$ in the basis of $(\boldsymbol{F}, \boldsymbol{G}, \boldsymbol{H})$.
- Fortunately, knowing $\hat{\boldsymbol{\Gamma}}_f(\boldsymbol{A}, \boldsymbol{C})$ and $\boldsymbol{\Gamma}_f(\boldsymbol{F}, \boldsymbol{G})$, we have

$$\boldsymbol{\Gamma}_f(\boldsymbol{F}, \boldsymbol{G})\boldsymbol{T} = \hat{\boldsymbol{\Gamma}}_f(\boldsymbol{A}, \boldsymbol{C}).$$

- Thus, we can get $\hat{\hat{\boldsymbol{X}}}_{f,M}$ is the correct basis as follows

$$\hat{\hat{\boldsymbol{X}}}_{f,M_T} = \boldsymbol{\Gamma}_f^{\dagger}(\boldsymbol{F}, \boldsymbol{G})\hat{\boldsymbol{\Gamma}}_f(\boldsymbol{A}, \boldsymbol{C}) \times \boldsymbol{\Sigma}^{1/2}(1:\hat{n}_x, 1:\hat{n}_x)\boldsymbol{\mathcal{V}}^\top(1:\hat{n}_x, :).$$

Covariance matrix estimation

- We can finally estimate

$$\begin{bmatrix} \hat{\boldsymbol{W}}_{f,M-1} \\ \hat{\boldsymbol{V}}_{f,M-1} \end{bmatrix} = \begin{bmatrix} \hat{\hat{\boldsymbol{X}}}_{f+1,M_T} \\ \boldsymbol{Y}_{f,M-1} \end{bmatrix} - \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} \\ \boldsymbol{H} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\hat{\boldsymbol{X}}}_{f,M-1_T} \\ \boldsymbol{U}_{f,M-1} \end{bmatrix},$$

and

$$\begin{bmatrix} \hat{\boldsymbol{V}} & \hat{\boldsymbol{S}} \\ \hat{\boldsymbol{S}}^\top & \hat{\boldsymbol{W}} \end{bmatrix} = \lim_{M\to\infty} \frac{1}{M} \begin{bmatrix} \hat{\boldsymbol{W}}_{f,M-1} \\ \hat{\boldsymbol{V}}_{f,M-1} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{W}}^\top_{f,M-1} & \hat{\boldsymbol{V}}^\top_{f,M-1} \end{bmatrix}.$$

NUMERICAL ILLUSTATIONS

Toy example

- We consider

$$\boldsymbol{F} = \begin{bmatrix} 0.603 & 0.603 & 0 & 0 \\ -0.603 & 0.603,0 & 0 & \\ 0 & 0 & -0.603 & -0.603 \\ 0 & 0 & 0.603 & -0.603 \end{bmatrix},$$

$$\boldsymbol{G} = \begin{bmatrix} 1.1650 & -0.6965 \\ 0.6268 & 1.6961 \\ 0.0751 & 0.0591 \\ 0.3516 & 1.7971 \end{bmatrix},$$

$$\boldsymbol{H} = \begin{bmatrix} 0.2641 & -1.4462 & 1.2460 & 0.5774 \\ 0.8717 & -0.7012 & -0.6390 & -0.3600 \end{bmatrix},$$

$$\boldsymbol{K} = 4 \times \begin{bmatrix} 0.1242 & -0.0895 \\ -0.0828 & -0.0128 \\ 0.0390 & -0.0968 \\ -0.0225 & 0.1459 \end{bmatrix},$$

$$\boldsymbol{R}_e = \begin{bmatrix} 0.0176 & -0.0267 \\ -0.0267 & 0.0497 \end{bmatrix}.$$

- We generate 10^3 realizations of the noise sequence and we select a data length $N = 1000$.

Université de Poitiers

Estimates of the elements of V

		$\hat{v}_{11}$	$\hat{v}_{12}$	$\hat{v}_{22}$
Theo. value		0.0176	-0.0267	0.0497
Sample cov.	avg.	0.0176	-0.0267	0.0497
	std.	0.0008	0.0012	0.0022
ICM	avg.	0.0588	-0.0750	0.0888
	std.	0.0412	0.0149	0.0057
DCM	avg.	0.031	-0.041	0.028
	std.	0.0041	0.0013	0.006
CMM	avg.	0.023	-0.073	0.031
	std.	0.0087	0.0066	0.0092
New meth.	avg.	0.0198	-0.0272	0.0516
	std.	0.0011	0.0013	0.0024

Estimates of the elements of $\boldsymbol{W}$

		$\hat{w}_{11}$	$\hat{w}_{12}$	$\hat{w}_{22}$	$\hat{w}_{23}$	$\hat{w}_{34}$	$\hat{w}_{44}$
Theo. value		0.0202	-0.0045	0.0149	-0.0198	0.0012	-0.0031
Sample cov.	avg.	0.0202	-0.0045	0.0149	-0.0198	0.0012	-0.0031
	std.	0.8886e-03	0.2054e-03	0.6563e-03	0.8750e-03	0.0509e-03	0.1487e-03
ICM	avg.	0.0526	-0.0150	0.0355	-0.0454	0.0041	-0.0103
	std.	0.0146	0.0079	0.0066	0.0067	0.0042	0.0036
DCM	avg.	0.0113	-0.0058	0.0186	-0.0285	0.003	-0.0103
	std.	0.004	0.0033	0.0068	0.0067	0.0039	0.0033
CMM	avg.	0.0170	-0.0041	0.0124	-0.0234	0.0041	-0.0043
	std.	0.0097	0.0082	0.0064	0.0062	0.0052	0.0028
New meth.	avg.	0.0196	-0.0041	0.0145	-0.0190	0.0015	-0.0026
	std.	0.0017	0.0006	0.0011	0.0011	0.0004	0.0005

Estimates of the elements of $\boldsymbol{S}$

		$\hat{s}_{11}$	$\hat{s}_{13}$	$\hat{s}_{21}$	$\hat{s}_{24}$
Theo. value		0.0183	-0.0045	0.0131	-0.0172
Sample cov.	avg.	0.0183	-0.0045	0.0131	-0.0172
	std.	0.0008	0.0002	0.0006	0.0008
New meth.	avg.	0.0181	-0.0039	0.0137	-0.0169
	std.	0.0011	0.0007	0.0010	0.0010

Michelin results

- Data is generated with VI-CRT (realistic simulator).
- A nonlinear state space model is used, *i.e.*,

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t, \boldsymbol{\theta}),$$
$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), t, \boldsymbol{\theta}),$$

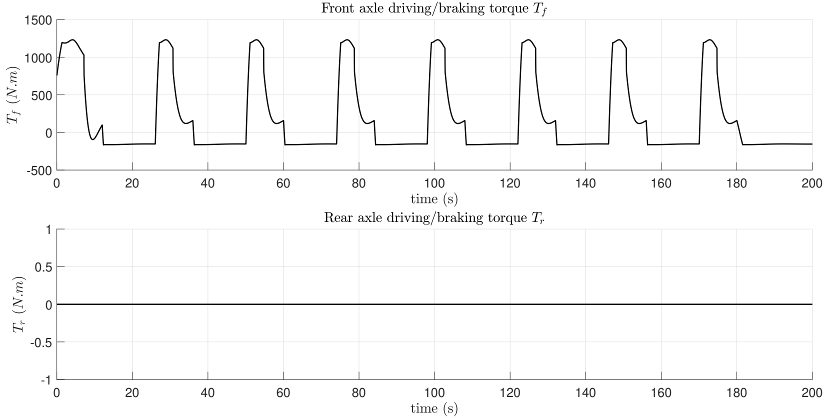
with

$$\boldsymbol{x} = \begin{bmatrix} v_x & \omega_f & \omega_r & F_{x_f} & F_{x_r} & \dot{F}_{x_f} & \dot{F}_{x_r} & \kappa & \dot{\kappa} \end{bmatrix}^\top,$$
$$\boldsymbol{u} = \begin{bmatrix} T_f & T_r \end{bmatrix}^\top,$$
$$\boldsymbol{y} = \begin{bmatrix} v_x & \omega_f & \omega_r & \dot{\kappa} \end{bmatrix}^\top,$$

involving

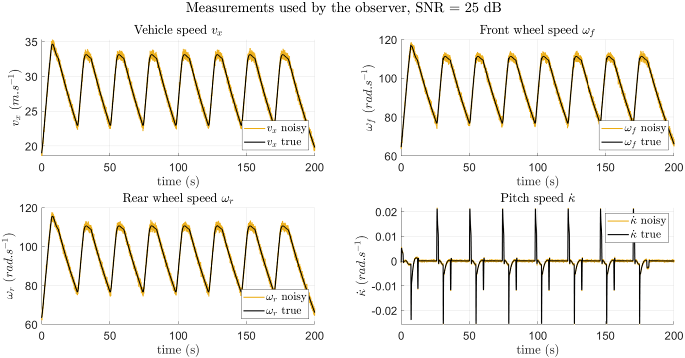
- a single track model for the dynamics,
- an effective tire radius model,
- a suspension model and a load transfer model.

Input data



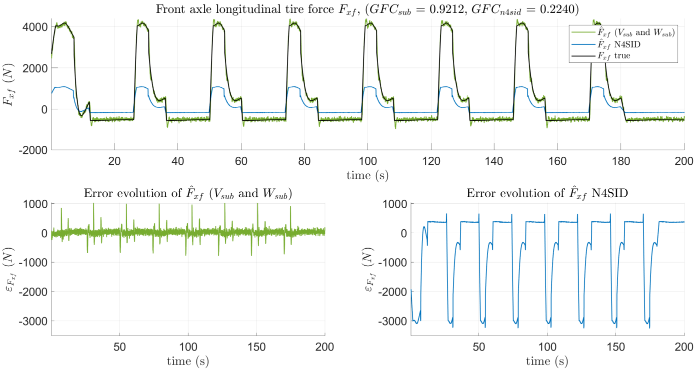
Front axle driving/braking torque T_f
T_f (N.m)
1500
1000
500
0
-500
0
20
40
60
80
100
120
140
160
180
200
time (s)
Rear axle driving/braking torque T_r
T_r (N.m)
1
0.5
0
-0.5
-1
0
20
40
60
80
100
120
140
160
180
200
time (s)

Output data

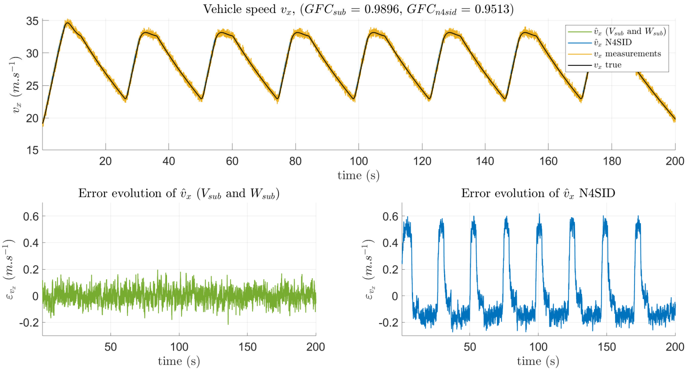


Measurements used by the observer, SNR = 25 dB
Vehicle speed v_x
v_x $(m.s^{-1})$
v_x noisy
v_x true
time (s)
Front wheel speed ω_f
ω_f $(rad.s^{-1})$
ω_f noisy
ω_f true
time (s)
Rear wheel speed ω_r
ω_r $(rad.s^{-1})$
ω_r noisy
ω_r true
time (s)
Pitch speed $\dot{\kappa}$
$\dot{\kappa}$ $(rad.s^{-1})$
$\dot{\kappa}$ noisy
$\dot{\kappa}$ true
time (s)

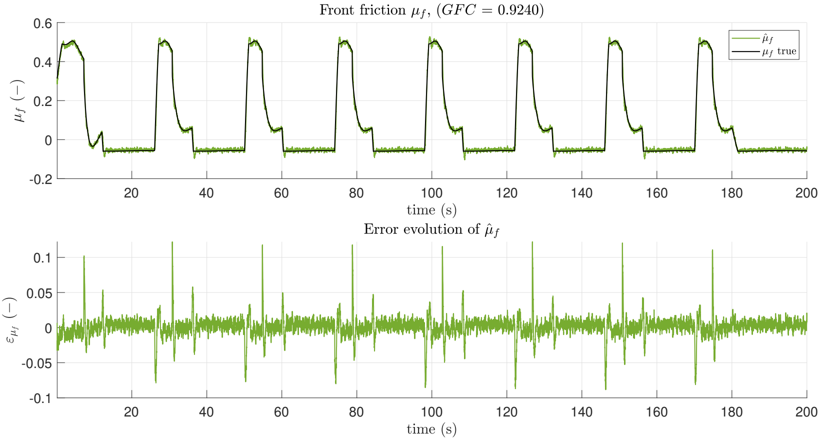
Long. tire force estimation



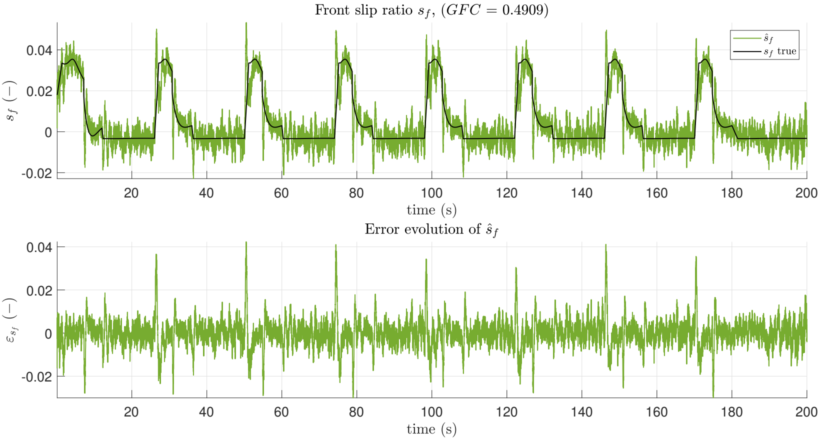
Vehicle speed estimation



Friction reconstruction



Slip ratio reconstruction



Université de Poitiers

CONCLUSIONS AND DISCUSSION

Take home message

- A new solution for noise covariance matrix estimation has been introduced.
- This solution involves SML solutions only.
- No tuning is required by the user.
- The accuracy of this solution was proved in the linear Kalman filter framework.
- This solution can be used with an EKF (no accuracy proof yet).
- See `https://doi.org/10.1002/acs.3213`.
- Do not hesitate to use SML solutions.