Chapter 1: Least Squares Parameter Estimation: A Review

- determine a set of unknown parameters (**x**) from a set of measurements (**y**)
- measurements are contaminated be unavoidable errors ($\boldsymbol{\varepsilon}$)
- the expected value of the measurements is: E(**y**) = **y** - $\boldsymbol{\varepsilon}$
- E(**y**) is related to **x** by a (non-linear) function **f**: E(**y**) = **f**(**x**) ==> **y** = **f**(**x**) + $\boldsymbol{\varepsilon}$
- the error $\boldsymbol{\varepsilon}$ is a continuous random variable (RV) ==> **y** is a continuous RV
- continuous RVs have probability density distributions p($\boldsymbol{\varepsilon}$), p(**y**)
- ==> determine **x** from **y** by taking p(**y**) into account!

The first moment of a RV is the expected value μ of the RV: E(RV)

$$\mathbf{y} = [y_1 \cdots y_i \cdots y_n]^T, p(\mathbf{y}) = p(y_1, \cdots, y_n) \tag{1.1}$$

$$\Rightarrow \mu_i = E(y_i) = \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} y_i \cdot p(y_1, \cdots, y_n) \cdot dy_1 \cdots dy_n$$

Chapter 1: Least Squares Parameter Estimation: A Review - cont'd

A central moment of a RV is the moment calculated for RV- E(RV)

The first central moment of a RV is zero:

$$E(y_i - \mu_i) = \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} (y_i - \mu_i) \cdot p(y_1, \cdots, y_n) \cdot dy_1 \cdots dy_n \tag{1.2}$$

$$= \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} y_i \cdot p(y_1, \cdots, y_n) \cdot dy_1 \cdots dy_n - \mu_i \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} p(y_1, \cdots, y_n) \cdot dy_1 \cdots dy_n = \mu_i - \mu_i$$

The second central moment of a RV is called covariance

$$\sigma_{ij} = E((y_i - \mu_i)(y_j - \mu_j)) = E(y_i y_j) - \mu_i \mu_j$$

$$= \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} (y_i - \mu_i)(y_j - \mu_j) \cdot p(y_1, \cdots, y_n) \cdot dy_1 \cdots dy_n \tag{1.3}$$

If i=j, $\sigma_{ii} = \sigma_i^2$ is called variance

The variances and covariances of the n-dimensional RV **y** can be assembled in the n-dimensional square symmetric covariance matrix

$$\Sigma(\mathbf{y}) = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdot & \sigma_{1n} \\ \sigma_{12} & \sigma_2^2 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \sigma_{1n} & \cdot & \cdot & \sigma_n^2 \end{bmatrix} \tag{1.4}$$

Covariance propagation: if two RV **r** (n-dimensional) and **s** (m-dimensional) are linearly related according to

$$\mathbf{r} = \mathbf{A}\mathbf{s} + \mathbf{c}$$

with **A** being a (constant) matrix of dimension (n x m), and **c** being a m-dimensional constant, than:

$$\Sigma(\mathbf{r}) = \mathbf{A}\Sigma(\mathbf{s})\mathbf{A}^T \tag{1.5}$$

Linearisation: the relation $\mathbf{y} = \mathbf{f}(\mathbf{x}) + \varepsilon$ is non-linear. In order to use the tools of linear algebra, the relation needs to be linearised. Assuming some approximate values for **x** are known, f(x) can be expanded in a series (Taylor-expansion).

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(x_1,\cdots,x_m) \\ f_2(x_1,\cdots,x_m) \\ \cdot \\ f_n(x_1,\cdots,x_m) \end{bmatrix} \tag{1.6}$$

$$= \begin{bmatrix} f_1(x_1,\cdots,x_m)|_0 \\ f_2(x_1,\cdots,x_m)|_0 \\ \cdot \\ f_n(x_1,\cdots,x_m)|_0 \end{bmatrix} + \begin{bmatrix} \partial f_1/\partial x_1|_0 & \partial f_1/\partial x_2|_0 & \cdot & \partial f_1/\partial x_m|_0 \\ \partial f_2/\partial x_1|_0 & \partial f_2/\partial x_2|_0 & \cdot & \partial f_2/\partial x_m|_0 \\ \cdot & \cdot & \cdot & \cdot \\ \partial f_n/\partial x_1|_0 & \partial f_n/\partial x_2|_0 & \cdot & \partial f_n/\partial x_m|_0 \end{bmatrix} \cdot \begin{bmatrix} x_1 - x_1|_0 \\ x_2 - x_2|_0 \\ \cdot \\ x_m - x_m|_0 \end{bmatrix} + t.o.h.o.$$

The first term on the r.h.s. and the partial derivatives in the second term are evaluated using the approximate values for the parameters **x**.

It is assumed that the terms of higher order can be neglected (iterations, if required).
Re-definitions:

$$\mathbf{y} := \mathbf{y} - \begin{bmatrix} f_1(x_1,\cdots,x_m)|_0 \\ f_2(x_1,\cdots,x_m)|_0 \\ \cdot \\ f_n(x_1,\cdots,x_m)|_0 \end{bmatrix}, \mathbf{A} := \begin{bmatrix} \partial f_1/\partial x_1|_0 & \partial f_1/\partial x_2|_0 & \cdot & \partial f_1/\partial x_m|_0 \\ \partial f_2/\partial x_1|_0 & \partial f_2/\partial x_2|_0 & \cdot & \partial f_2/\partial x_m|_0 \\ \cdot & \cdot & \cdot & \cdot \\ \partial f_n/\partial x_1|_0 & \partial f_n/\partial x_2|_0 & \cdot & \partial f_n/\partial x_m|_0 \end{bmatrix}, \mathbf{x} := \begin{bmatrix} x_1 - x_1|_0 \\ x_2 - x_2|_0 \\ \cdot \\ x_m - x_m|_0 \end{bmatrix}$$

==> linearised observation equations $\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\varepsilon}$ (1.7)

dim($\mathbf{y}$) = n, dim($\mathbf{x}$) = m, dim($\boldsymbol{\varepsilon}$) = n, dim($\mathbf{A}$) = n x m, n > m.
Since both $\mathbf{x}$ and $\boldsymbol{\varepsilon}$ are unknown, there exist infinitely many solutions.

Specific solution is obtained by requiring that

1. The estimated solution for $\mathbf{x}$ be a linear function of $\mathbf{y}$.
2. The weighted quadratic norm of the estimated errors be a minimum.
3. The estimated solution for $\mathbf{x}$ be unbiased.

In equations:

$$\hat{\mathbf{x}} = \mathbf{B}\mathbf{y} \tag{1.8}$$

$$\hat{\boldsymbol{\varepsilon}}^T \mathbf{P} \hat{\boldsymbol{\varepsilon}} = \min \tag{1.9}$$

$$E(\hat{\mathbf{x}}) = \mathbf{x} \tag{1.10}$$

Question: How is the matrix $\mathbf{B}$ determined? From eqns. (1.7) and (1.9)

$$\begin{aligned} \hat{\boldsymbol{\varepsilon}}^T \mathbf{P} \hat{\boldsymbol{\varepsilon}} &= (\mathbf{y} - \mathbf{A}\hat{\mathbf{x}})^T \mathbf{P} (\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}) = \mathbf{y}^T\mathbf{P}\mathbf{y} - \hat{\mathbf{x}}^T\mathbf{A}^T\mathbf{P}\mathbf{y} - \mathbf{y}^T\mathbf{P}\mathbf{A}\hat{\mathbf{x}} + \hat{\mathbf{x}}^T\mathbf{A}^T\mathbf{P}\mathbf{A}\hat{\mathbf{x}} \\ &= \mathbf{y}^T\mathbf{P}\mathbf{y} - 2\hat{\mathbf{x}}^T\mathbf{A}^T\mathbf{P}\mathbf{y} + \hat{\mathbf{x}}^T\mathbf{A}^T\mathbf{P}\mathbf{A}\hat{\mathbf{x}} \end{aligned}$$

$$\hat{\boldsymbol{\varepsilon}}^T \mathbf{P} \hat{\boldsymbol{\varepsilon}} = \min \Rightarrow \frac{\partial \hat{\boldsymbol{\varepsilon}}^T \mathbf{P} \hat{\boldsymbol{\varepsilon}}}{\partial \hat{\mathbf{x}}} = 0 \Rightarrow \frac{\partial \hat{\boldsymbol{\varepsilon}}^T \mathbf{P} \hat{\boldsymbol{\varepsilon}}}{\partial \hat{\mathbf{x}}} = -2\mathbf{A}^T\mathbf{P}\mathbf{y} + 2\mathbf{A}^T\mathbf{P}\mathbf{A}\hat{\mathbf{x}} \tag{1.11}$$

Chapter 1: Least Squares Parameter Estimation: A Review - cont'd

From eqns. (1.8) and (1.11) then follows the Least Squares solution:

$$\boxed{\hat{\mathbf{x}} = (\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1}\mathbf{A}^T\mathbf{P}\mathbf{y}} \qquad (1.12)$$

Question: How should the weight matrix **P** be chosen? Gauß proposed to use as weight matrix a (scaled) inverse of covariance matrix of the measurements, c.f. eqn. (1.4)

$$\boxed{\mathbf{P} = \sigma_0^2\boldsymbol{\Sigma}(\mathbf{y})^{-1}} \qquad (1.13)$$

From eqns. (1.5), (1.12) and (1.13) we now can also determine the covariance matrix of the estimated parameters

$$\boldsymbol{\Sigma}(\hat{\mathbf{x}}) = (\mathbf{A}^T\sigma_0^2\boldsymbol{\Sigma}(\mathbf{y})^{-1}\mathbf{A})^{-1}\mathbf{A}^T\sigma_0^2\boldsymbol{\Sigma}(\mathbf{y})^{-1}\boldsymbol{\Sigma}(\mathbf{y})((\mathbf{A}^T\sigma_0^2\boldsymbol{\Sigma}(\mathbf{y})^{-1}\mathbf{A})^{-1}\mathbf{A}^T\sigma_0^2\boldsymbol{\Sigma}(\mathbf{y})^{-1})^T$$

$$\boxed{\boldsymbol{\Sigma}(\hat{\mathbf{x}}) = (\mathbf{A}^T\boldsymbol{\Sigma}(\mathbf{y})^{-1}\mathbf{A})^{-1} = \sigma_0^2(\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1}} \qquad (1.14)$$

For a proof of the unbiasedness insert eqn. (1.12) into eqn. (1.10).

Chapter 1: Least Squares Parameter Estimation: A Review - cont'd

Estimates for the adjusted measurements and their covariance:

$$\boxed{\hat{\mathbf{y}} = \mathbf{A}\hat{\mathbf{x}}; \quad \boldsymbol{\Sigma}(\hat{\mathbf{y}}) = \sigma_0^2\mathbf{A}(\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1}\mathbf{A}^T} \qquad (1.15)$$

Estimates for the measurement errors and their covariance:

$$\boxed{\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}; \quad \boldsymbol{\Sigma}(\hat{\boldsymbol{\varepsilon}}) = \sigma_0^2(\mathbf{P}^{-1} - \mathbf{A}(\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1}\mathbf{A}^T)} \qquad (1.16)$$

Estimate for the variance of unit weight:

$$\boxed{\hat{\sigma}_0^2 = \frac{\hat{\boldsymbol{\varepsilon}}^T\mathbf{P}\hat{\boldsymbol{\varepsilon}}}{n-u}} \qquad (1.17)$$

n: number of measurements
u: number of parameters

<u>Chapter 2:</u> Sequential Least Squares Parameter Estimation

Assumption: the system of (linear) observation equations can be divided into two sets of equations which are uncorrelated.

$$\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix}=\begin{bmatrix}\mathbf{A}_1\\ \mathbf{A}_2\end{bmatrix}\mathbf{x}+\begin{bmatrix}\boldsymbol{\varepsilon}_1\\ \boldsymbol{\varepsilon}_2\end{bmatrix},\quad \mathbf{P}=\begin{bmatrix}\mathbf{P}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{P}_2\end{bmatrix} \tag{2.1}$$

The solution of the first set of equations has already been computed and is available.

$$\boxed{\begin{aligned}\hat{\mathbf{x}}_{(1)}&=(\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1)^{-1}\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{y}_1\\ \hat{\boldsymbol{\varepsilon}}_{1(1)}&=\mathbf{y}_1-\mathbf{A}_1\hat{\mathbf{x}}_{(1)}\\ \hat{\sigma}_{0(1)}^2&=\frac{\hat{\boldsymbol{\varepsilon}}_{1(1)}^{\mathrm{T}}\mathbf{P}_1\hat{\boldsymbol{\varepsilon}}_{1(1)}}{n_{(1)}-u}\\ \boldsymbol{\Sigma}(\hat{\mathbf{x}}_{(1)})&=\hat{\sigma}_{0(1)}^2(\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1)^{-1}\end{aligned}} \tag{2.2}$$

Subscripts 1, 2 for first and second set of equations; subscripts (1), (2) for estimation based on first set of equations and for estimation based on both sets of equations!

<u>Chapter 2:</u> Sequential Least Squares Parameter Estimation - cont'd

Question: Can the solution of eqns. (2.1) be obtained by updating the solution of the first set of equations (2.2) instead of solving the complete system (2.1)?

Formally the solution to (2.1) reads:

$$\begin{aligned}\hat{\mathbf{x}}_{(2)}&=\left[\begin{bmatrix}\mathbf{A}_1^{\mathrm{T}} & \mathbf{A}_2^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbf{P}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{P}_2\end{bmatrix}\begin{bmatrix}\mathbf{A}_1\\ \mathbf{A}_2\end{bmatrix}\right]^{-1}\begin{bmatrix}\mathbf{A}_1^{\mathrm{T}} & \mathbf{A}_2^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbf{P}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{P}_2\end{bmatrix}\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix}\\ &=\left[\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{A}_2\right]^{-1}\left[\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{y}_1+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{y}_2\right]\end{aligned} \tag{2.3}$$

$$\textit{if}\quad \hat{\mathbf{x}}_{(2)}=\hat{\mathbf{x}}_{(1)}+\Delta\hat{\mathbf{x}},\quad \textit{then} \tag{2.4}$$

$$\begin{aligned}&\left[\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{A}_2\right]\left[\hat{\mathbf{x}}_{(1)}+\Delta\hat{\mathbf{x}}\right]=\left[\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{y}_1+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{y}_2\right]\\ &\Rightarrow \cancel{\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1\hat{\mathbf{x}}_{(1)}}+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{A}_2\hat{\mathbf{x}}_{(1)}+\left[\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{A}_1+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{A}_2\right]\Delta\hat{\mathbf{x}}=\left[\cancel{\mathbf{A}_1^{\mathrm{T}}\mathbf{P}_1\mathbf{y}_1}+\mathbf{A}_2^{\mathrm{T}}\mathbf{P}_2\mathbf{y}_2\right]\end{aligned} \tag{2.5}$$

Chapter 2: Sequential Least Squares Parameter Estimation - cont'd

Parameter update: $$\Delta\hat{\mathbf{x}} = \left[\mathbf{A}_1^T\mathbf{P}_1\mathbf{A}_1 + \mathbf{A}_2^T\mathbf{P}_2\mathbf{A}_2\right]^{-1}\mathbf{A}_2^T\mathbf{P}_2\left[\mathbf{y}_2 - \mathbf{A}_2\hat{\mathbf{x}}_{(1)}\right]$$

$$\boxed{\begin{aligned} &\Rightarrow \Delta\hat{\mathbf{x}} = \left[\hat{\sigma}_{0(1)}^2\mathbf{\Sigma}(\hat{\mathbf{x}}_{(1)})^{-1} + \mathbf{A}_2^T\mathbf{P}_2\mathbf{A}_2\right]^{-1}\mathbf{A}_2^T\mathbf{P}_2\left[\mathbf{y}_2 - \mathbf{A}_2\hat{\mathbf{x}}_{(1)}\right] \\ &\Rightarrow \hat{\mathbf{x}}_{(2)} = \hat{\mathbf{x}}_{(1)} + \Delta\hat{\mathbf{x}} \end{aligned}} \qquad (2.6)$$

Updating the variance of unit weight:

$$\hat{\sigma}_{0(2)}^2 = \frac{\hat{\boldsymbol{\varepsilon}}_{1(2)}^T\mathbf{P}_1\hat{\boldsymbol{\varepsilon}}_{1(2)} + \hat{\boldsymbol{\varepsilon}}_{2(2)}^T\mathbf{P}_2\hat{\boldsymbol{\varepsilon}}_{2(2)}}{n_{(2)} - u}$$

$$\hat{\boldsymbol{\varepsilon}}_{1(2)} = \mathbf{y}_1 - \mathbf{A}_1\hat{\mathbf{x}}_{(2)} = \mathbf{y}_1 - \mathbf{A}_1\hat{\mathbf{x}}_{(1)} - \mathbf{A}_1\Delta\hat{\mathbf{x}} = \hat{\boldsymbol{\varepsilon}}_{1(1)} - \mathbf{A}_1\Delta\hat{\mathbf{x}}$$

$$\hat{\boldsymbol{\varepsilon}}_{1(2)}^T\mathbf{P}_1\hat{\boldsymbol{\varepsilon}}_{1(2)} = \hat{\boldsymbol{\varepsilon}}_{1(1)}^T\mathbf{P}_1\hat{\boldsymbol{\varepsilon}}_{1(1)} \cancel{- 2\Delta\hat{\mathbf{x}}^T\mathbf{A}_1^T\mathbf{P}_1\hat{\boldsymbol{\varepsilon}}_{1(1)}} + \Delta\hat{\mathbf{x}}^T\mathbf{A}_1^T\mathbf{P}_1\mathbf{A}_1\Delta\hat{\mathbf{x}}$$

$$\boxed{\hat{\sigma}_{0(2)}^2 = \frac{1}{n_{(2)} - u}\left(\hat{\sigma}_{0(1)}^2\left(n_{(1)} - u + \Delta\hat{\mathbf{x}}^T\mathbf{\Sigma}(\hat{\mathbf{x}}_{(1)})^{-1}\Delta\hat{\mathbf{x}}\right) + \hat{\boldsymbol{\varepsilon}}_{2(2)}^T\mathbf{P}_2\hat{\boldsymbol{\varepsilon}}_{2(2)}\right)} \qquad (2.7)$$

Chapter 2: Sequential Least Squares Parameter Estimation - cont'd

Updating the covariance matrix of the estimated parameters:

$$\boxed{\mathbf{\Sigma}(\hat{\mathbf{x}}_{(2)}) = \hat{\sigma}_{0(2)}^2\left[\hat{\sigma}_{0(1)}^2\mathbf{\Sigma}(\hat{\mathbf{x}}_{(1)})^{-1} + \mathbf{A}_2^T\mathbf{P}_2\mathbf{A}_2\right]^{-1}} \qquad (2.8)$$

Eqns. (2.6) - (2.8) update the parameter estimation based on the previous solution and the new measurements and the corresponding design matrix.

Typical application example: Time series of measurements related to a common set of parameters, uncorrelated between measurement epochs.

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}(t_1) \\ \mathbf{y}(t_2) \\ \cdot \\ \mathbf{y}(t_q) \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \mathbf{A}(t_1) \\ \mathbf{A}(t_2) \\ \cdot \\ \mathbf{A}(t_q) \end{bmatrix}, \mathbf{P} = \begin{bmatrix} \mathbf{P}(t_1) & \mathbf{0} & \cdot & \mathbf{0} \\ \mathbf{0} & \mathbf{P}(t_2) & \cdot & \mathbf{0} \\ \cdot & \cdot & \cdot & \cdot \\ \mathbf{0} & \mathbf{0} & \cdot & \mathbf{P}(t_q) \end{bmatrix} \qquad (2.9)$$

Sequential solution of equation system (2.9)

$$\textit{Abbreviations}: \quad \hat{\mathbf{x}}(t_k) = \hat{\mathbf{x}}_k \quad \textit{etc.}$$

$$\boxed{\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + \left[\hat{\sigma}_{0\,k-1}^2 \mathbf{\Sigma}(\hat{\mathbf{x}}_{k-1})^{-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{P}_k \mathbf{A}_k\right]^{-1} \mathbf{A}_k^{\mathrm{T}} \mathbf{P}_k \left[\mathbf{y}_k - \mathbf{A}_k \hat{\mathbf{x}}_{k-1}\right]} \tag{2.10}$$

$$\boxed{\hat{\sigma}_{0k}^2 = \frac{1}{n_k - u}\left(\hat{\sigma}_{0k-1}^2\left(n_{k-1} - u + \Delta\hat{\mathbf{x}}_k^{\mathrm{T}} \mathbf{\Sigma}(\hat{\mathbf{x}}_{k-1})^{-1} \Delta\hat{\mathbf{x}}_k\right) + (\mathbf{y}_k - \mathbf{A}_k \hat{\mathbf{x}}_k)^{\mathrm{T}} \mathbf{P}_k (\mathbf{y}_k - \mathbf{A}_k \hat{\mathbf{x}}_k)\right)} \tag{2.11}$$

$$\boxed{\mathbf{\Sigma}(\hat{\mathbf{x}}_k) = \hat{\sigma}_{0k}^2 \left[\hat{\sigma}_{0k-1}^2 \mathbf{\Sigma}(\hat{\mathbf{x}}_{k-1})^{-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{P}_k \mathbf{A}_k\right]^{-1}} \tag{2.12}$$

Chapter 3: Ordinary Differential Equations

Ordinary Differential Equations express relations between

- derivatives of a function (**y**', **y**'', **y**''' . . .,$\mathbf{y}^{(m)}$)
- the function itself (**y**)
- and the independent variable (t)

$$\boxed{\mathbf{F}(t,\mathbf{y},\mathbf{y}',\mathbf{y}'',\ldots\mathbf{y}^{(m)})=0} \qquad (3.1)$$

The prime (') denotes the derivative with respect to the independent variable t. The function **y** depends on only one independent variable; therefore equ. (3.1) describes an Ordinary Differential Equation (ODE) in contrast to Partial Differential Equations involving multiple independent variables.

Equ. (3.1) is an ODE of order m; it is solved by integration (m times).

Each of the m integrations requires specification of initial values for some value t_0 of the independent variable t.

$$\boxed{\mathbf{y}^{(i-1)}(t_0)=\mathbf{y}_0^{(i-1)},\ i=1,m} \qquad (3.2)$$

Chapter 3: Ordinary Differential Equations - cont'd

If the relation **F** in equ. (3.1) is linear in the function **y** and its derivatives, it is called a Linear Differential Equation

$$\boxed{\mathbf{y}^{(m)}(t)+\mathbf{A}_1(t)\mathbf{y}^{(m-1)}(t)+\mathbf{A}_2(t)\mathbf{y}^{(m-2)}(t)+\ldots.+\mathbf{A}_{m-1}(t)\mathbf{y}'(t)+\mathbf{A}_m(t)\mathbf{y}(t)=\mathbf{b}(t)} \qquad (3.3)$$

The $\mathbf{A}_i$(t) are square matrices with elements that are functions of t (but do not depend on **y** and its derivatives). Since **y** in equ. (3.3) is a vector valued function, equ. (3.3) represents a System of Linear Differential Equations of mth-order. For a scalar valued function y:

$$\boxed{\mathrm{y}^{(m)}(t)+a_1(t)\mathrm{y}^{(m-1)}(t)+a_2(t)\mathrm{y}^{(m-2)}(t)+\ldots.+a_{m-1}(t)\mathrm{y}'(t)+a_m(t)\mathrm{y}(t)=\mathrm{b}(t)} \qquad (3.4)$$

The a_i(t) are scalar coefficients that are functions of t (but do not depend on y and its derivatives). Equ. (3.4) represents a scalar Linear Differential Equation of mth-order. The corresponding initial values are:

$$\boxed{\mathrm{y}^{(i-1)}(t_0)=\mathrm{y}_0^{(i-1)},\ i=1,m} \qquad (3.5)$$

A scalar Linear Differential Equation of m[th]-order can be transformed into a System of m Linear Differential Equations of 1[st]-order through substitution:

$$\boxed{\begin{aligned} &y_1(t) = y(t) \\ &y_2(t) = y'(t) \\ &\bullet \\ &y_i(t) = y^{(i-1)}(t) \\ &\bullet \\ &y_m(t) = y^{(m-1)}(t) \end{aligned}} \qquad (3.6)$$

$$\boxed{\begin{aligned} &\frac{d}{dt} y_i(t) = y^{(i)}, \quad i = 1, m-1 \\ &\frac{d}{dt} y_m(t) + a_1(t) y_m(t) + a_2(t) y_{m\text{-}1}(t) + \ldots + a_m(t) y_1(t) = b(t) \end{aligned}} \qquad (3.7)$$

Equ. (3.7) can be rearranged to read:

$$\boxed{\frac{d}{dt}\begin{pmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \\ \cdot \\ y_m(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 & 0 & \cdot & 0 \\ 0 & 0 & -1 & \cdot & 0 \\ 0 & 0 & 0 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ a_m(t) & a_{m-1}(t) & a_{m-2}(t) & \cdot & a_1(t) \end{pmatrix}\begin{pmatrix} y_1(t) \\ y_2(t) \\ \cdot \\ \cdot \\ y_m(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \cdot \\ b(t) \end{pmatrix}} \qquad (3.8)$$

Accordingly the initial values (equ. (3.5)) must be transformed:

$$\boxed{\begin{pmatrix} y(t_0) \\ y'(t_0) \\ y''(t_0) \\ \cdot \\ y^{(m\text{-}1)}(t_0) \end{pmatrix} = \begin{pmatrix} y_1(t_0) \\ y_2(t_0) \\ y_3(t_0) \\ \cdot \\ y_m(t_0) \end{pmatrix}} \qquad (3.9)$$

Chapter 3: Ordinary Differential Equations - cont'd

Because any higher order Linear Differential Equation can be transformed into a System of Linear Differential Equations of 1st order, it is sufficient to study the solutions of the latter type. In short we can re-write equs. (3.8), (3.9) as

$$\boxed{\begin{aligned}&\mathbf{y}'(t)+\mathbf{A}(t)\mathbf{y}(t)=\mathbf{b}(t)\\&\mathbf{y}(t_0)=\mathbf{y}_0\end{aligned}} \qquad (3.10)$$

$$\mathbf{y}'(t)=\left(y'_1(t),y'_2(t),\ldots,y'_m(t)\right)^{\mathrm{T}}$$

$$\mathbf{y}_0=\left(y_1(t_0),y_2(t_0),\ldots,y_m(t_0)\right)^{\mathrm{T}}$$

$$\mathbf{b}(t)=\left(0,0,\ldots,b(t)\right)^{\mathrm{T}}$$

Analytical solutions to equ. (3.10) can be obtained in many (but not all) cases as the sum of the solution of the corresponding homogeneous equation

$$\boxed{\mathbf{y}'(t)+\mathbf{A}(t)\mathbf{y}(t)=\mathbf{0}} \qquad (3.11)$$

and a particular solution of the original equation (3.10). Otherwise a numerical solution can be obtained.

Chapter 3: Ordinary Differential Equations - cont'd

Example for a method of numerical solution: Runge-Kutta-Methode. The methode is first explained for a scalar first-order Differential Equation.

$$\boxed{\begin{aligned}&\mathrm{y}'(t)+a(t)\mathrm{y}(t)=\mathrm{b}(t)\Rightarrow \mathrm{y}'(t)=\mathrm{f}(t,\mathrm{y}(t))\\&\mathrm{y}(t_0)=\mathrm{y}_0\end{aligned}} \qquad (3.12)$$

If the solution is available for t_n, than, in principle, the solution for t_{n+1} could be obtained through a Taylor expansion

$$\mathrm{y}(t_{n+1})=\mathrm{y}(t_n)+\mathrm{y}'(t_n)(t_{n+1}-t_n)+\frac{1}{2!}\mathrm{y}''(t_n)(t_{n+1}-t_n)^2+\frac{1}{3!}\mathrm{y}'''(t_n)(t_{n+1}-t_n)^3+\ldots\ldots.$$

with

$$\mathrm{y}'(t_n)=\mathrm{f}$$

$$\mathrm{y}''(t_n)=\frac{\partial \mathrm{f}}{\partial \mathrm{t}}+\frac{\partial \mathrm{f}}{\partial \mathrm{y}}\mathrm{f}$$

etc.

In the Runge-Kutta-Methode of numerical integration, the higher derivatives (y'', y''', etc.) are replaced by first order derivatives. The order of the algorithm indicates, up to which term of the Taylor expansion the algorithm is 'correct'.

Notation:

$$y(t_{n+1}) = y_{n+1}$$
$$(t_{n+1} - t_n) = h$$

First order algorithm:

$$\boxed{y_{n+1} = y_n + h\, f(y_n, t_n)} \tag{3.13}$$

Second order algorithm:

$$\boxed{\begin{aligned} y_{n+1} &= y_n + \frac{h}{2}(k_1 + k_2) \\ k_1 &= f(y_n, t_n) \\ k_2 &= f(y_n + h\, k_1, t_n + h) \end{aligned}} \tag{3.14}$$

Third order algorithm:

$$\boxed{\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 4k_2 + k_3) \\ k_1 &= f(y_n, t_n) \\ k_2 &= f(y_n + \frac{h}{2} k_1, t_n + \frac{h}{2}) \\ k_3 &= f(y_n - h\, k_1 + 2h\, k_2, t_n + h) \end{aligned}} \tag{3.15}$$

Fourth order algorithm:

$$\boxed{\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 &= f(y_n, t_n) \\ k_2 &= f(y_n + \frac{h}{2} k_1, t_n + \frac{h}{2}) \\ k_3 &= f(y_n + \frac{h}{2} k_2, t_n + \frac{h}{2}) \\ k_4 &= f(y_n + h\, k_3, t_n + h) \end{aligned}} \tag{3.16}$$

The Runge-Kutta-Methode can also be applied to the numerical integration of Systems of 1st-order Linear Differential Equations. Example for third order algorithm:

$$\boxed{\begin{aligned}
&\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}(\mathbf{k}_1 + 4\mathbf{k}_2 + \mathbf{k}_3) \\
&\mathbf{k}_1 = \mathbf{f}(\mathbf{y}_n, t_n) \\
&\mathbf{k}_2 = \mathbf{f}(\mathbf{y}_n + \frac{h}{2}\mathbf{k}_1, t_n + \frac{h}{2}) \\
&\mathbf{k}_3 = \mathbf{f}(\mathbf{y}_n - h\,\mathbf{k}_1 + 2h\,\mathbf{k}_2, t_n + h)
\end{aligned}} \tag{3.17}$$

Chapter 4: Linear Dynamic Systems

State space description of a Linear Dynamic System

$$\boxed{\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{w}(t) + \mathbf{L}(t)\,\mathbf{s}(t)} \tag{4.1}$$

x(t): Set of random variables describing the linear system (the state vector)
w(t): Random forcing function
s(t): Deterministic control input
F(t): square matrix
G(t), **L**(t): matrices (not necessarily square!)

Here we will consider only linear dynamic models without control input:

$$\boxed{\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{w}(t)} \tag{4.2}$$

This is formally a non-homogeneous linear differential equation.

Chapter 4: Linear Dynamic Systems

For given initial conditions $\mathbf{x}(t_0)$, the general solution to equ.(4.2) can be written:

$$\boxed{\mathbf{x}(t) = \mathbf{\Phi}(t,t_0)\,\mathbf{x}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}(t,t')\,\mathbf{G}(t')\,\mathbf{w}(t')\,dt'} \tag{4.3}$$

The general solution is the sum of the the solution of the homogeneous equation and a particular solution of the non-homogeneous equation!

$\Phi(t,t_0)$ is called the state transition matrix.The following relations hold for this matrix:

$$\begin{aligned}
&\frac{d}{dt}\mathbf{\Phi}(t,t_0) = \mathbf{F}(t)\,\mathbf{\Phi}(t,t_0) \\
&\mathbf{\Phi}(t_2,t_0) = \mathbf{\Phi}(t_2,t_1)\,\mathbf{\Phi}(t_1,t_0) \\
&\mathbf{\Phi}(t,t) = \mathbf{\Phi}(t,t_0)\,\mathbf{\Phi}(t_0,t) = \mathbf{I} \Rightarrow \mathbf{\Phi}^{-1}(t,t_0) = \mathbf{\Phi}(t_0,t)
\end{aligned} \tag{4.4}$$

Up to now $\Phi(t,t_0)$ is still unknown!

Transition matrix for stationary systems: in stationary systems, the matrix **F** in equs. (4.1) and (4.2) is time-invariant. Stationary systems can often be used to replace approximatively more complex systems over short time periods. A general Taylor expansion gives:

$$\boxed{\mathbf{x}(t) = \mathbf{x}(t_0) + \dot{\mathbf{x}}(t_0)(t - t_0) + \frac{1}{2!}\ddot{\mathbf{x}}(t_0)(t - t_0)^2 + ...} \qquad (4.5)$$

From the homogeneous part of equ. (4.2) we can replace:

$$\begin{aligned} \dot{\mathbf{x}}(t_0) &= \mathbf{F}\mathbf{x}(t_0) \\ \ddot{\mathbf{x}}(t_0) &= \dot{\mathbf{F}}\mathbf{x}(t_0) + \mathbf{F}\dot{\mathbf{x}}(t_0) = \mathbf{F}\mathbf{F}\mathbf{x}(t_0) = \mathbf{F}^2\mathbf{x}(t_0) \\ &. \\ &. \\ \mathbf{x}^{(n)}(t_0) &= \mathbf{F}^n\mathbf{x}(t_0) \end{aligned} \qquad (4.6)$$

Chapter 4: Linear Dynamic Systems

Substituting equ. (4.6) into (4.5):

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \mathbf{F}\mathbf{x}(t_0)(t - t_0) + \frac{\mathbf{F}^2}{2!}\mathbf{x}(t_0)(t - t_0)^2 + ...$$

$$\boxed{\mathbf{x}(t) = \left[\mathbf{I} + \mathbf{F}(t - t_0) + \frac{\mathbf{F}^2}{2!}(t - t_0)^2 .....\right]\mathbf{x}(t_0)} \qquad (4.7)$$

The term in square brackets is by definition the matrix exponential

$$\begin{aligned} e^{\mathbf{A}} &= \mathbf{I} + \mathbf{A} + \frac{\mathbf{A}^2}{2!} + \frac{\mathbf{A}^3}{3!} + ... \\ e^{\mathbf{F}(t - t_0)} &= \mathbf{I} + \mathbf{F}(t - t_0) + \frac{\mathbf{F}^2}{2!}(t - t_0)^2 .... \end{aligned} \qquad (4.8)$$

For stationary systems, the state transition matrix depends only on the time interval (t-t_0) and the matrix **F**

$$\boxed{\mathbf{\Phi}(t, t_0) = e^{\mathbf{F}\,(t - t_0)}} \qquad (4.9)$$

We are now in a position to discretize the continuous system of equ. (4.2):

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

$$\Downarrow$$

$$\boxed{\begin{aligned}&\mathbf{x}(t_n) = \mathbf{\Phi}(t_n, t_{n-1})\mathbf{x}(t_{n-1}) + \mathbf{u}(t_n)\\ &\text{or}\\ &\mathbf{x}_n = \mathbf{\Phi}(t_n, t_{n-1})\mathbf{x}_{n-1} + \mathbf{u}_n\end{aligned}} \qquad (4.10)$$

with

$$\mathbf{u}_n = \int_{t_{n-1}}^{t_n} \mathbf{\Phi}(t, t')\,\mathbf{G}(t')\,\mathbf{w}(t')\,dt' \qquad (4.11)$$

For stationary systems, the state transition matrix is computed from equ. (4.9). The discretization of equ. (4.10) holds also for non-stationary systems; but then the state transition matrix cannot be computed from equ. (4.9)!

Chapter 5: Random Processes

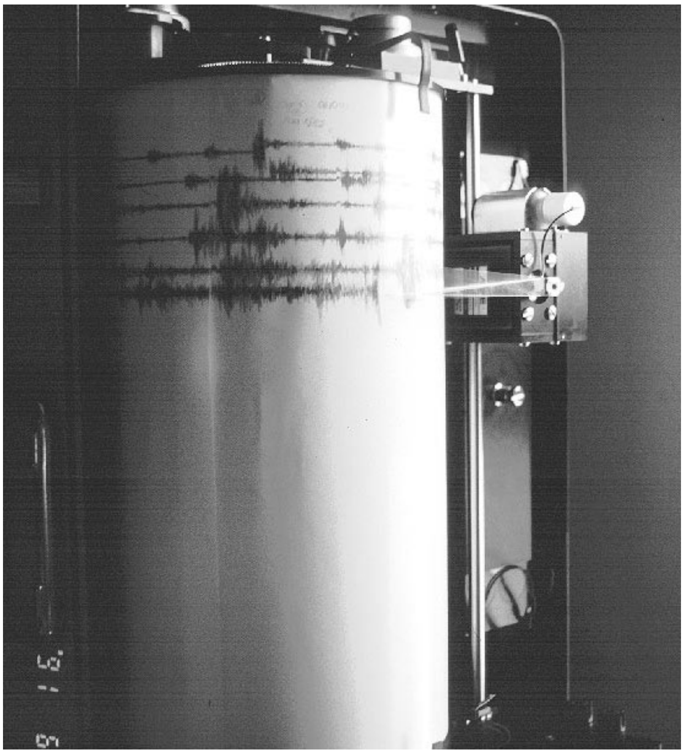
Review: A realisation of a random variable (RV) is the outcome of an experiment (e.g. measurement, c.f. Module 1). Performing the experiment once gives one realisation of the RV. Measurement errors are typically continuous RVs that can take any value according to an associated probability density distribution. Example: the Gaussian bell-shaped probability density distribution.
RVs can be scalar or more-dimensional.

A realisation of a random process (RP, sometimes also called stochastic process) is obtained if the outcome of the experiment is a ‘function’ of an independent variable, usually time. A RP is called continuous, if its argument is continuous time, $x(t)$. A RP is called discrete, if its argument is a discrete variable, $x(t_i)$, $i=1, 2, 3 \ldots$ Performing the experiment once gives one realisation of the RP.
RPs can be scalar or more-dimensional.

Examples for continuous and discrete random variables?

Chapter 5: Random Processes

Examples for continuous and discrete random variables?



Chapter 5: Random Processes - cont'd

Example
(after A. Gelb (ed.), 1984. Applied Optimal Estimation, MIT Press)

Shown are 4 realisations of a scalar RP. There may be infinitely many 'realisations' constituting the ensemble of the RP.

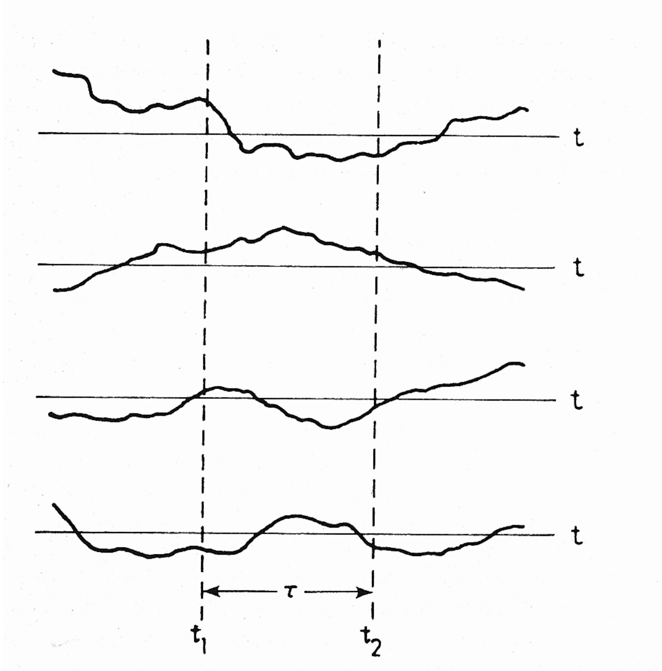


Figure 5.1: Example for scalar random process

Chapter 5: Random Processes - cont'd

If **y**(t) is a RP and p(**y**(t)) is its probability density distribution, than the expected value of the RP at time t is its mean value:

$$\boldsymbol{\mu}(t) = \mathrm{E}(\mathbf{y}(t)) = \int_{-\infty}^{+\infty} \mathbf{y}(\mathrm{t}) \cdot \mathrm{p}(\mathbf{y}(\mathrm{t})) \cdot \mathrm{d}\mathbf{y}(t) \tag{5.1}$$

The auto-covariance and the auto-correlation of the RP are measures for the self-similarity at two epochs in time.

$$\begin{aligned} Cov_{yy}(t_1,t_2) &= \mathrm{E}((\mathbf{y}(t_1)-\boldsymbol{\mu}(t_1))(\mathbf{y}(t_2)-\boldsymbol{\mu}(t_2))^{\mathrm{T}}) \\ Cor_{yy}(t_1,t_2) &= \mathrm{E}(\mathbf{y}(t_1)\mathbf{y}(t_2)^{\mathrm{T}}) \end{aligned} \tag{5.2}$$

The cross-covariance of two RP is a measure for the similarity of the two RP at two epochs in time.

$$Cov_{xy}(t_1,t_2) = \mathrm{E}((\mathbf{x}(t_1)-\boldsymbol{\mu}_x(t_1))(\mathbf{y}(t_2)-\boldsymbol{\mu}_y(t_2))^{\mathrm{T}}) \tag{5.3}$$

Stationarity: A RP is said to be stationary, if its probability density distribution is independent of time, i.e.:

$$\mathrm{p}(\mathbf{y}(\mathrm{t})) = \mathrm{p}(\mathbf{y}(\mathrm{t}+\Delta\mathrm{t})) \tag{5.4}$$

For stationary RPs, the mean value μ is independent of time, and the auto-covariance and auto-correlation functions depend only on the time interval:

$$\begin{aligned} Cov_{yy}(\Delta t) &= \mathrm{E}((\mathbf{y}(t)-\boldsymbol{\mu})(\mathbf{y}(t+\Delta t)-\boldsymbol{\mu})^{\mathrm{T}}) \\ Cor_{yy}(\Delta t) &= \mathrm{E}(\mathbf{y}(t)\mathbf{y}(t+\Delta t)^{\mathrm{T}}) \end{aligned} \tag{5.5}$$

For stationary RPs, the auto-covariance is an even function, which attains its maximum at zero lag; for zero lag, the auto-covariance function value is the variance of the stationary RP:

$$Cov_{yy}(\Delta t) = Cov_{yy}(-\Delta t), \quad \left|Cov_{yy}(\Delta t)\right| \le Cov_{yy}(0) \tag{5.6}$$

For stationary RPs also the cross-covariance function depends only on the time interval:

$$Cov_{xy}(\Delta t) = \mathrm{E}((\mathbf{x}(t)-\boldsymbol{\mu}_x)(\mathbf{y}(t+\Delta t)-\boldsymbol{\mu}_y)^{\mathrm{T}}) \tag{5.7}$$

Equ. (5.6) does not hold for the cross-covariance function.

The Fourier transform of the auto-covariance function is called the power spectral density (psd) of the RP.

$$\Phi_y(f) = \int_{-\infty}^{+\infty} Cov_{yy}(\tau)\cdot e^{-i2\pi f\tau}\,d\tau \tag{5.8}$$

Similarly, the auto-covariance function of the RP is the inverse Fourier transform of the power spectral density of the RP.

$$Cov_{yy}(\tau) = \int_{-\infty}^{+\infty} \Phi_y(f)\cdot e^{i2\pi f\tau}\,df \tag{5.9}$$

==> The variance of the RP equals the area under the psd of the process!

Ergodicity: A stationary RP is called ergodic, if the statistics of the RP (mean, variance, etc.) can be derived from a single realisation of the RP by operations in the time domain. The time average of a single realisation of the RP is:

$$\mathbf{m} = \lim(T \to \infty)\frac{1}{T}\int_{-T/2}^{+t/2} \mathbf{y}(\mathrm{t}) \cdot \mathrm{d}t \qquad (5.10)$$

Ergodicity means, that this time average is equal to the ensemble average (eqn. (5.1)). Similarly, the the auto- and cross-correlation functions for ergodic RPs can be computed from:

$$\begin{aligned} Cov_{yy}(\Delta t) &= \lim(T \to \infty)\frac{1}{T}\int_{-T/2}^{T/2} (\mathbf{y}(t)-\mathbf{m})(\mathbf{y}(t+\Delta t)-\mathbf{m})^T \cdot dt \\ Cov_{xy}(\Delta t) &= \lim(T \to \infty)\frac{1}{T}\int_{-T/2}^{T/2} (\mathbf{x}(t)-\mathbf{m}_x)(\mathbf{y}(t+\Delta t)-\mathbf{m}_y)^T \cdot dt \end{aligned} \qquad (5.11)$$

Chapter 5: Random Processes - cont'd

White noise: A stationary RP is called a white noise process, if its auto-covariance function is zero for non-zero lag (here: zero mean white noise):

$$\begin{aligned} &Cov_{ww}(\Delta t) = \sigma^2 \cdot \delta(\Delta t) \\ &\delta(t) = 0 \;\; for\; t \neq 0, \quad \int_{-\infty}^{\infty} g(t') \cdot \delta(t-t')\, dt' = g(t) \end{aligned} \qquad (5.12)$$

The psd of a white noise process is obtained from eqn. (5.8):

$$\Phi_w(f) = \int_{-\infty}^{+\infty} \sigma^2 \cdot \delta(\tau) \cdot e^{-i2\pi f\tau} d\tau = \sigma^2 \qquad (5.13)$$

The psd of a white noise process is a constant. Its spectral power is evenly distributed; hence the name "white noise". Inserting eqn. (5.13) into (5.9), we obtain a definition for the Dirac function in terms of the integral of a complex exponential function.

$$\int_{-\infty}^{+\infty} \sigma^2 \cdot e^{i2\pi f\tau} df = \sigma^2 \cdot \delta(\tau) \qquad (5.14)$$

Chapter 5: Random Processes - cont'd

Random constants: A random constant is a stationary RP that take on a constant value for all times, but the constant value is a random variable that changes for each realisation. Examples??

The random constant can be described by its differential equation:

$$\dot{R}(t) = 0,\ R(t_0) = R_0 \tag{5.15}$$

The random constant is not an ergodic RP. The auto-covariance function for a random constant is:

$$Cov_{RR}(\Delta t) = \sigma_R^2 \tag{5.16}$$

The RP is fully correlated. The psd of the random constant is obtained from eqn. (5.8):

$$\Phi_R(f) = \int_{-\infty}^{+\infty} \sigma_R^2 \cdot e^{-i2\pi f\tau} d\tau = \sigma_R^2 \cdot \delta(f) \tag{5.17}$$

Random walk: A random walk is a RP described by the following differential equation:

$$\begin{aligned} &\dot{R}(t) = W(t),\ R(t_0) = 0 \\ &\Rightarrow R(t) = \int_{t_0}^{t} W(t)\, dt \end{aligned} \tag{5.18}$$

The random walk process is not a stationary RP. Its auto-covariance function is given by:

$$\begin{aligned} Cov_{RR}(t_1,t_2) &= \sigma^2 \cdot (\mathrm{t}_1 - \mathrm{t}_0), \quad \text{if} \quad \mathrm{t}_2 \geq \mathrm{t}_1 \\ Cov_{RR}(t_1,t_2) &= \sigma^2 \cdot (\mathrm{t}_2 - \mathrm{t}_0), \quad \text{if} \quad \mathrm{t}_1 > \mathrm{t}_2 \end{aligned} \tag{5.19}$$

σ^2 is the amplitude of the psd of the white noise RP *W(t).*

Examples??

Gaussian white noise random process: A white noise RP is called a Gaussian white noise RP, if the probability density distribution of the underlying random variable is the Gaussian distribution (bell-shaped curve).

If the white noise RP underlying the random walk RP is a Gaussian white noise RP, then it is called a *Wiener* process; the *Wiener* process is the integral of the Gaussian white noise RP.

Gauss-Markov-process of first order: The differential equation

$$\dot{X}(t) = -\beta X(t) + W(t),\ \beta \geq 0 \quad (5.20)$$

where *W(t)* is a zero-mean Gaussian white noise RP with psd amplitude equal to $2\sigma^2\beta$ describes a Gauss-Markov-process of first order. Its auto-covariance is given by:

$$Cov_{XX}(t_1, t_2) = Cov_{XX}(\Delta t) = \sigma^2 \cdot e^{-\beta \Delta t} \quad (5.21)$$

σ^2 is the variance of the RP; the RP is stationary.

Chapter 5: Random Processes - cont'd

The psd of a Gauss-Markov-process of first order is given by:

$$\Phi_X(f) = \frac{2\sigma^2\beta}{4\pi^2 f^2 + \beta^2} \quad (5.22)$$

β is the parameter describing the correlation length of the RP.

Relation to Random Constant RP?

Relation to Gaussian white noise RP?

Shape of auto-covariance function? Shape of psd?

Gauss-Markov-process of second order: The differential equation

$$\ddot{X}(t)+2\beta\dot{X}(t)+\beta^2 X(t)=W(t),\ \beta\geq 0 \qquad (5.23)$$

where *W(t)* is a zero-mean Gaussian white noise RP with psd amplitude equal to $4\sigma^2\beta^3$ describes a Gauss-Markov-process of second order. Its auto-covariance and its psd are is given by:

$$Cov_{XX}(t_1,t_2)=Cov_{XX}(\Delta t)=\sigma^2\cdot e^{-\beta\Delta t}(1+\beta\Delta t),\ \Delta t=|t_2-t_1| \qquad (5.24)$$

$$\Phi_X(f)=\frac{4\sigma^2\beta^3}{\left(4\pi^2 f^2+\beta^2\right)^2} \qquad (5.25)$$

Shape of auto-covariance function and psd?

Chapter 5: Random Processes - cont'd

Gauss-Markov-process of third order: The differential equation

$$\dddot{X}(t)+3\beta\ddot{X}(t)+3\beta^2\dot{X}(t)+\beta^3 X(t)=W(t),\ \beta\geq 0 \qquad (5.26)$$

where *W(t)* is a zero-mean Gaussian white noise RP with psd amplitude equal to $16/3\sigma^2\beta^5$ describes a Gauss-Markov-process of third order. Its auto-covariance and its psd are is given by:

$$Cov_{XX}(t_1,t_2)=Cov_{XX}(\Delta t)=\sigma^2\cdot e^{-\beta\Delta t}(1+\beta\Delta t+1/3\beta^2\Delta t^2),\ \Delta t=|t_2-t_1| \qquad (5.27)$$

$$\Phi_X(f)=\frac{16/\sigma^2\beta^5}{\left(4\pi^2 f^2+\beta^2\right)^3} \qquad (5.28)$$

Shape of auto-covariance function and psd?

Gauss-Markov-processes of higher order defined by the appropriate differential equations. Relation between Gauss-Markov-processes of different order?

Discrete random processes: The continuous RPs "random constant", "random walk" and "first order Markov process" can be described by the differential equation

$$\dot{X}(t) = -\beta X(t) + \alpha W(t);\ \alpha, \beta \geq 0 \qquad (5.29)$$

where *W(t)* is a zero-mean Gaussian white noise RP. For the "random constant" both α and β are zero. For the "random walk" β is zero and α is unity. For the "first order Markov process" β is non-zero positive and α is unity. For the discrete counterparts of these continuous RPs the differential equation (5.29) is replaced by the difference equation

$$X_{n+1} = b_n \cdot X_n + a_{n+1} \cdot W_{n+1} \qquad (5.30)$$

where W_n is a zero-mean Gaussian white noise random sequence with variance σ^2.

- For the discrete "random constant", $b_n=1$ and $a_n=0$.
- For the discrete "random walk", $b_n=1$ and $a_n=1$.
- For the discrete "first order Markov process", $b_n \neq 0$ and $a_n=1$.

Variance propagation: (see Module 1, equ. (1.5))

Random constant:

$$X_{n+1} = X_n$$
$$\sigma^2_{X,n+1} = \sigma^2_{X,n} \qquad (5.31)$$

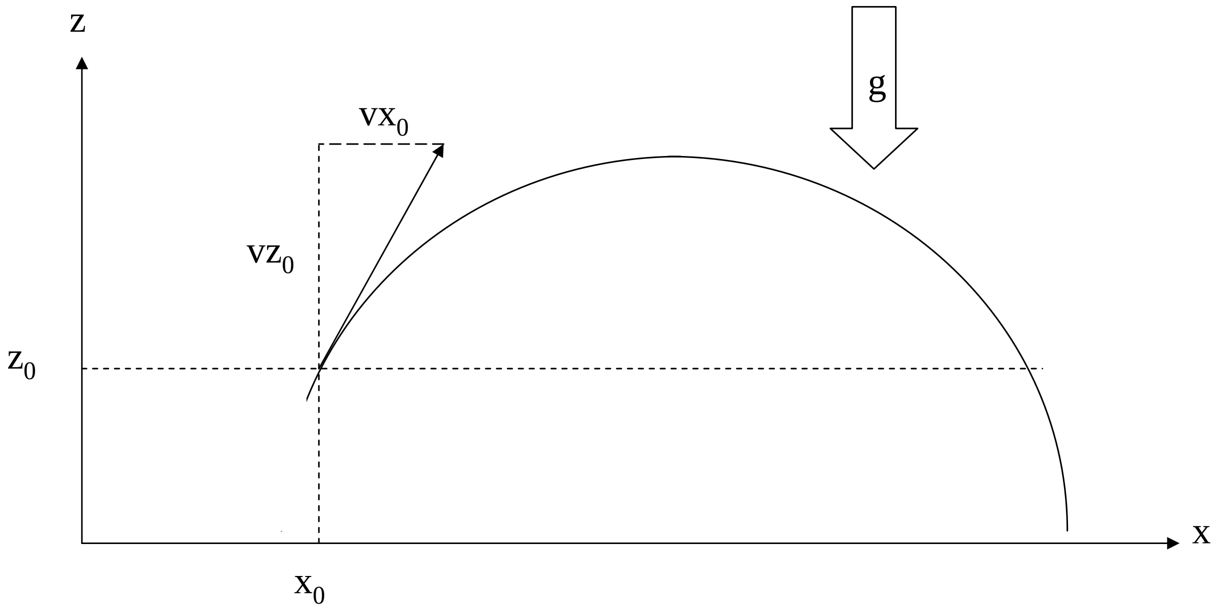
Random walk:

$$X_{n+1} = X_n + W_{n+1}$$
$$\sigma^2_{X,n+1} = \sigma^2_{X,n} + \sigma^2_{W,n+1} \qquad (5.32)$$

First order Markov process (exponential decay in autocorrelation):

$$X_{n+1} = b_n X_n + a_{n+1} W_{n+1}$$
$$\sigma^2_{X,n+1} = b_n^2 \sigma^2_{X,n} + a^2_{n+1} \sigma^2_{W,n+1} \qquad (5.33)$$

Chapter 6: An example



$$d^2z/dt^2 = -g;\ \ z(t_0) = z_0,\ \ vz(t_0) = vz_0$$
$$d^2x/dt^2 = \ 0;\ \ x(t_0) = x_0,\ \ vx(t_0) = vx_0 \qquad (6.1)$$

Chapter 6: An example - cont'd

Übung 1: analytically integrate equ. (6.1)

$$\Rightarrow\ vx(t) = f(vx_0,t);\ x(t) = f(x_0,vx_0,t)$$
$$\Rightarrow\ vz(t) = f(vz_0,g,t);\ z(t) = f(z_0,vz_0,g,t) \qquad (6.2)$$

measurements:
$$lx(t_i) = x(t_i) + \varepsilon x(t_i) \qquad (6.3)$$
$$lz(t_i) = z(t_i) + \varepsilon z(t_i)$$

Combine eqn. (6.2) and (6.3) to form observation equations; solve these equations to determine the (unknown) initial conditions. The results are estimates (least squares) for x_0,vx_0, z_0,vz_0

Übung 2: repeat Übung 1 using sequential least squares estimation

==> same result as in Übung 1!

Chapter 6: An example - cont'd

Übung 3: Assume that the analytical solution for eqn. (6.1) does not exist. Re-formulate eqn. (6.1) as a system of ordinary linear differential equations of first order:

$$dx_1/dt = x_2,\;\; dx_2/dt = 0,\;\; dz_1/dt = z_2,\;\; dz_2/dt = -g$$

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -g \end{bmatrix} \qquad (6.4)$$

With given initial conditions x_0, vx_0, z_0, vz_0, solve equ. (6.4) through numerical integration using a Runge-Kutta algorithm of third order, and a step size of 1s.

==> Result: $x_1(t_i)$, $x_2(t_i)$, $z_1(t_i)$, $z_2(t_i)$

Chapter 6: An example - cont'd

NOW: Assume that it is not known, if there are any outside forces acting on the vehicle other than gravity. In this more general case there is no analytical solution for eqn. (6.1).

Assume that the vehicle carries a platform with two accelerometers measuring the vehicle's acceleration in two orthogonal directions. The platform is stabilised so that the accelerations are always measured in x-direction and in z-direction.

Accelerometer model: The output of the accelerometers is the sum the (integrated) true vehicle acceleration (velocity increments!), a random constant R, and a white noise W:

$$bz(t_n) = \int_{t_{n-1}}^{t_n} \frac{d^2 z}{d\tau^2}(\tau)\, d\tau + R_z + W_z(t_n) - \int_{t_{n-1}}^{t_n} g(\tau)\, d\tau$$

$$bx(t_n) = \int_{t_{n-1}}^{t_n} \frac{d^2 x}{d\tau^2}(\tau)\, d\tau + R_x + W_x(t_n) \qquad (6.4)$$

Chapter 6: An example - cont'd

Integration: Position and velocity are computed from

$$\begin{aligned}
\tilde{x}(t_n) &= \tilde{x}(t_{n-1}) + v\tilde{x}(t_{n-1})\cdot\Delta t + \frac{1}{2}bx(t_n)\cdot\Delta t \\
v\tilde{x}(t_n) &= v\tilde{x}(t_{n-1}) + bx(t_n) \\
\tilde{z}(t_n) &= \tilde{z}(t_{n-1}) + v\tilde{z}(t_{n-1})\cdot\Delta t + \frac{1}{2}\left(bz(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau\right)\cdot\Delta t \\
v\tilde{z}(t_n) &= v\tilde{z}(t_{n-1}) + bz(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau
\end{aligned} \tag{6.5}$$

$$\begin{aligned}
\delta z(t_n) &= \tilde{z}(t_n) - z(t_n), \delta vz(t_n) = v\tilde{z}(t_n) - vz(t_n) \\
\delta x(t_n) &= \tilde{x}(t_n) - x(t_n), \delta vx(t_n) = v\tilde{x}(t_n) - vz(t_n)
\end{aligned} \tag{6.6}$$

are position and velocity errors at epoch t_n.

Chapter 6: An example - cont'd

True position and velocity are (theoretically) computed from:

$$\begin{aligned}
x(t_n) &= x(t_{n-1}) + vx(t_{n-1})\cdot\Delta t + \frac{1}{2}(bx(t_n) - R_x - W_x(t_n))\cdot\Delta t \\
vx(t_n) &= vx(t_{n-1}) + bx(t_n) - R_x - W_x(t_n) \\
z(t_n) &= z(t_{n-1}) + vz(t_{n-1})\cdot\Delta t + \frac{1}{2}\left(bz(t_n) - R_z - W_z(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau\right)\cdot\Delta t \\
vz(t_n) &= vz(t_{n-1}) + bz(t_n) - R_z - W_z(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau
\end{aligned} \tag{6.7}$$

Assumptions for equ. (6.7):

1. gravitational accelerations (*g(t)*) are exactly known
2. kinematic accelerations change linearly within the interval Δt

Taking the difference of equ. (6.5) and (6.7) according to equ. (6.5) describes the error growth:

$$\begin{aligned}
\delta x(t_n) &= \delta x(t_{n-1}) + \delta vx(t_{n-1}) \cdot \Delta t + \frac{1}{2}(R_x + W_x(t_n)) \cdot \Delta t \\
\delta vx(t_n) &= \delta vx(t_{n-1}) + R_x + W_x(t_n) \\
\delta z(t_n) &= \delta z(t_{n-1}) + \delta vz(t_{n-1}) \cdot \Delta t + \frac{1}{2}(R_z + W_z(t_n)) \cdot \Delta t \\
\delta vz(t_n) &= \delta vz(t_{n-1}) + R_z + W_z(t_n)
\end{aligned} \tag{6.8}$$

$$\begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \end{bmatrix}(t_n) = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \end{bmatrix}(t_{n-1}) + \begin{bmatrix} 0.5\Delta t & 0 \\ 1 & 0 \\ 0 & 0.5\Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_x + W_x(t_n) \\ R_z + W_z(t_n) \end{bmatrix} \tag{6.9}$$

Chapter 7: State vector augmentation

The equation (6.9) describes the state of a linear system in the form

$$\mathbf{x}_n = \mathbf{F}\mathbf{x}_{n-1} + \mathbf{G}\mathbf{r}_n \quad (7.1)$$

x describes the state of the system, **F** is the matrix describing the state transition, **G** is a matrix, and **r** consists of realisations of random processes. In general, **r** consists of white noise **w** (uncorrelated in time) and RPs with temporal correlations **u** (random walk, random constant, Gauß-Markov processes, etc.). Correlated RPs can be described by differential equations or their discrete counterparts, equs. (5.31) -(5.34).

$$\begin{aligned} \mathbf{x}_n &= \mathbf{F}\mathbf{x}_{n-1} + \mathbf{G}\mathbf{w}_n + \mathbf{G}\mathbf{u}_n \\ \mathbf{u}_n &= \mathbf{B}\mathbf{u}_{n-1} + \mathbf{A}\mathbf{w}\mathbf{u}_n \end{aligned} \quad (7.2)$$

Inserting the second into the first equation

$$\mathbf{x}_n = \mathbf{F}\mathbf{x}_{n-1} + \mathbf{G}\mathbf{w}_n + \mathbf{G}\mathbf{B}\mathbf{u}_{n-1} + \mathbf{G}\mathbf{A}\mathbf{w}\mathbf{u}_n \quad (7.3)$$

Chapter 7: State vector augmentation - cont'd

Now the second equation (7.2) can be combined with the equ. (7.3) in a new system of equations:

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix}_n = \begin{bmatrix} \mathbf{F} & \mathbf{GB} \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix}_{n-1} + \begin{bmatrix} \mathbf{G} & \mathbf{GA} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{wu} \end{bmatrix}_n \quad (7.4)$$

This equation differs from equ. (7.1) in that respect, that the RPs in the second term on the right hand side are all white noise processes; the correlated RPs have augmented (erweitert) the state vector **x**.

Continuation of example from module 6:

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{G} = \begin{bmatrix} 0.5\Delta t & 0 \\ 1 & 0 \\ 0 & 0.5\Delta t \\ 0 & 1 \end{bmatrix}, \mathbf{w}_n = \begin{bmatrix} W_x(t_n) \\ W_z(t_n) \end{bmatrix}, \mathbf{u}_n = \begin{bmatrix} R_x \\ R_z \end{bmatrix} \quad (7.5)$$

In this example, R_x and R_z are random constants; therefore **B**=**I** and **A**=**0**.

$$\begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \\ R_x \\ R_z \end{bmatrix}(t_n) = \begin{bmatrix} 1 & \Delta t & 0 & 0 & 0.5\Delta t & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0.5\Delta t \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \\ R_x \\ R_z \end{bmatrix}(t_{n-1}) + \begin{bmatrix} 0.5\Delta t & 0 \\ 1 & 0 \\ 0 & 0.5\Delta t \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} W_x(t_n) \\ W_z(t_n) \end{bmatrix} \quad (7.6)$$

State vector augmentation for other types of correlated RPs?

State prediction and state covariance propagation

Equ. (7.6) is of the type (similar to equ. (7.1)):

$$\mathbf{x}_n = \mathbf{F}\mathbf{x}_{n-1} + \mathbf{G}\mathbf{w}_n \quad (7.7)$$

$\mathbf{w}_n$ are uncorrelated White Noise RP. The state of the linear system is predicted

$$\mathbf{x}_{n(p)} = \mathbf{F}\mathbf{x}_{n-1} \quad (7.8)$$

Since $\mathbf{x}_{n-1}$ and $\mathbf{w}_n$ are uncorrelated, the state covariance is propagated by:

$$\mathbf{C}x_{n(p)} = \mathbf{F}\mathbf{C}x_{n-1}\mathbf{F}^T + \mathbf{G}\mathbf{C}w_n\mathbf{G}^T \quad (7.9)$$

Chapter 8: State observation and estimation

Example from Module 6: equation (6.5):

$$\begin{bmatrix} \tilde{x}(t_n) \\ v\tilde{x}(t_n) \\ \tilde{z}(t_n) \\ v\tilde{z}(t_n) \end{bmatrix} = \begin{bmatrix} \tilde{x}(t_{n-1}) \\ v\tilde{x}(t_{n-1}) \\ \tilde{z}(t_{n-1}) \\ v\tilde{z}(t_{n-1}) \end{bmatrix} + \begin{bmatrix} v\tilde{x}(t_{n-1})\cdot\Delta t + 0.5bx(t_n)\cdot\Delta t \\ bx(t_n) \\ v\tilde{z}(t_{n-1})\cdot\Delta t + 0.5(bz(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau)\cdot\Delta t \\ bz(t_n) + \int_{t_{n-1}}^{t_n} g(\tau)d\tau \end{bmatrix} \tag{8.1}$$

State propagation: equation (7.6):

$$\begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \\ R_x \\ R_z \end{bmatrix}(t_n) = \begin{bmatrix} 1 & \Delta t & 0 & 0 & 0.5\Delta t & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0.5\Delta t \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x \\ \delta vx \\ \delta z \\ \delta vz \\ R_x \\ R_z \end{bmatrix}(t_{n-1}) + \begin{bmatrix} 0.5\Delta t & 0 \\ 1 & 0 \\ 0 & 0.5\Delta t \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} W_x(t_n) \\ W_z(t_n) \end{bmatrix} \tag{8.2}$$

Chapter 8: State observation and estimation – Kalman filter

In short notation

$$\tilde{\mathbf{x}}_n = \tilde{\mathbf{x}}_{n-1} + \mathbf{f}(sensor - output, gravity\ \ field) \tag{8.1}$$

$$\delta\mathbf{x}_n = \mathbf{F}\,\delta\mathbf{x}_{n-1} + \mathbf{G}\mathbf{w}_n,\ \ \mathbf{C}w_n,\ \ \mathbf{C}x_{n-1} \tag{8.2}$$

From equation (6.6):

$$\delta\mathbf{x}_n = \tilde{\mathbf{x}}_n - \mathbf{x}_n \tag{8.3}$$

External measurement $\mathbf{y}_n$, non-linearly related to $\mathbf{x}_n$:

$$\mathbf{y}_n = \mathbf{h}(\mathbf{x}_n) + \mathbf{v}_n,\ \ \mathbf{C}v_n \tag{8.4}$$

If the errors (equ. (8.3)) are small, linearize equ. (8.3):

$$\mathbf{y}_n = \mathbf{h}(\tilde{\mathbf{x}}_n) + \left.\frac{\partial\mathbf{h}}{\partial\mathbf{x}_n}\right|_{\tilde{\mathbf{x}}_n}\cdot(\mathbf{x}_n - \tilde{\mathbf{x}}_n) + \mathbf{v}_n,\ \ \mathbf{C}v_n \tag{8.5}$$

Chapter 8: State observation and estimation

Abbreviations:

$$\delta\mathbf{y}_n = \mathbf{y}_n - \mathbf{h}(\tilde{\mathbf{x}}_n) \tag{8.6}$$

$$\mathbf{H}_n = -\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_n}\right|_{\tilde{\mathbf{x}}_n} \tag{8.7}$$

$$\delta\mathbf{y}_n = \mathbf{H}_n \delta\mathbf{x}_n + \mathbf{v}_n,\ \mathbf{C}v_n \tag{8.8}$$

Qu.: How to combine equations (8.2) and (8.8) optimally to estimate the errors $\delta\mathbf{x}_n$ to improve the result from equation (8.1)?

Assumption: $\mathbf{v}_n$, $\mathbf{w}_n$, $\delta\mathbf{x}_{n-1}$ are not correlated.

Notation:
Use $\delta\mathbf{x}_n(-)$ to indicate the result of the prediction from equ. (8.2)
Use $\mathbf{C}x_n(-)$ to indicate the covariance matrix resulting from the prediction using equations (8.2) and (7.9)
Use (+) to indicate result after evaluation the measurement equ. (8.8).

Chapter 8: State observation and estimation

From equ. (8.2):

$$\begin{aligned} \delta\mathbf{x}_n(-) &= \mathbf{F}\,\delta\mathbf{x}_{n-1}(+) \\ \mathbf{C}x_n(-) &= \mathbf{F}\mathbf{C}x_{n-1}(+)\mathbf{F}^T + \mathbf{G}\mathbf{C}w_n\mathbf{G}^T \end{aligned} \tag{8.9}$$

Find a matrix $\mathbf{K}_n$ to combine the measurements and the prediction result linearly:

$$\begin{aligned} \delta\mathbf{x}_n(+) &= \delta\mathbf{x}_n(-) + \mathbf{K}_n(\delta\mathbf{y}_n - \mathbf{H}_n\delta\mathbf{x}_n(-)) \\ \mathbf{C}x_n(+) &= \mathbf{C}x_n(-) + \mathbf{K}_n\mathbf{C}v_n\mathbf{K}_n^T + \mathbf{K}_n\mathbf{H}_n\mathbf{C}x_n(-)\mathbf{H}_n^T\mathbf{K}_n^T \end{aligned} \tag{8.10}$$

or alternatively

$$\begin{aligned} \delta\mathbf{x}_n(+) &= (\mathbf{I} - \mathbf{K}_n\mathbf{H}_n)\delta\mathbf{x}_n(-) + \mathbf{K}_n\delta\mathbf{y}_n \\ \mathbf{C}x_n(+) &= (\mathbf{I} - \mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-)(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n)^T + \mathbf{K}_n\mathbf{C}v_n\mathbf{K}_n^T \end{aligned} \tag{8.11}$$

Kalman Filter: Find the matrix $\mathbf{K}_n$ (Gain matrix) such as to minimize the sum of the diagonal elements of the error covariance matrix $\mathbf{C}x_n(+)$

$$J_n = trace[\mathbf{C}x_n(+)] = \min \quad \Rightarrow \quad \partial J_n / \partial \mathbf{K}_n = 0 \tag{8.12}$$

Chapter 8: State observation and estimation

In general: $$\frac{\partial}{\partial \mathbf{A}}\left[trace\left(\mathbf{ABA}^T\right)\right] = 2\mathbf{AB} \qquad (8.13)$$

Then:

$$\frac{\partial}{\partial \mathbf{K}_n}\left[trace\left((\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-)(\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)^T\right)\right] = 2(\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-)\cdot -\mathbf{H}_n^T$$

$$\frac{\partial}{\partial \mathbf{K}_n}\left[trace\left(\mathbf{K}_n\mathbf{C}v_n\mathbf{K}_n^T\right)\right] = 2\mathbf{K}_n\mathbf{C}v_n$$

$$\frac{\partial}{\partial \mathbf{K}_n}\left[trace(\mathbf{C}x_n(+))\right] = -2(\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-)\mathbf{H}_n^T + 2\mathbf{K}_n\mathbf{C}v_n \qquad (8.14)$$

➔ $$\mathbf{K}_n = \mathbf{C}x_n(-)\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_n(-)\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1} \qquad (8.15)$$

➔ $$\mathbf{C}x_n(+) = (\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-) \qquad (8.16)$$

Chapter 8: State observation and estimation

Kalman Filter equations summary:

Step 1: Prediction

$$\boxed{\begin{aligned}&\delta\mathbf{x}_n(-) = \mathbf{F}\,\delta\mathbf{x}_{n-1}(+)\\ &\mathbf{C}x_n(-) = \mathbf{F}\mathbf{C}x_{n-1}(+)\mathbf{F}^T + \mathbf{G}\mathbf{C}w_n\mathbf{G}^T\end{aligned}}$$

Step 2: Gain matrix computation

$$\boxed{\mathbf{K}_n = \mathbf{C}x_n(-)\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_n(-)\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1}}$$

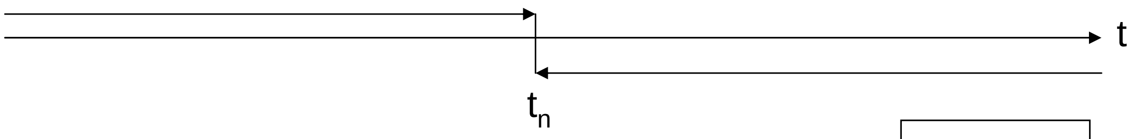
Step 3: Update

$$\boxed{\begin{aligned}\delta\mathbf{x}_n(+) &= \delta\mathbf{x}_n(-) + \mathbf{K}_n(\delta\mathbf{y}_n - \mathbf{H}_n\delta\mathbf{x}_n(-))\\ \mathbf{C}x_n(+) &= (\mathbf{I}-\mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-)\end{aligned}}$$

Chapter 9: Backward filtering and smoothing

Forward Filter:
(real time)

$$\delta\mathbf{x}_n(+) = \delta\mathbf{x}_n(-) + \mathbf{K}_n(\delta\mathbf{y}_n - \mathbf{H}_n\delta\mathbf{x}_n(-))$$
$$\mathbf{C}x_n(+) = (\mathbf{I} - \mathbf{K}_n\mathbf{H}_n)\mathbf{C}x_n(-) \quad (9.1)$$



Backward Filter:
(non-real time)

$$\delta\mathbf{x}_{nb}(+)$$
$$\mathbf{C}x_{nb}(+) \quad (9.2)$$

Qu: How to combine equs. (9.1) and (9.2) for an optimal estimate based on all data?

Chapter 9: Backward filtering and smoothing

Linear combination (weighted average)

$$\delta\mathbf{x}_n = \mathbf{A}\delta\mathbf{x}_n(+) + (\mathbf{I} - \mathbf{A})\delta\mathbf{x}_{nb}(+) \quad (9.3)$$
$$\mathbf{C}x_n = \mathbf{A}\mathbf{C}x_n(+)\mathbf{A}^T + (\mathbf{I} - \mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I} - \mathbf{A})^T$$

A and (**I-A**) are the weight matrices.

Qu: How to choose the weight matrices?

Selection of an optimality criterion! Minimize the trace of the covariance matrix of the result!

$$trace(\mathbf{C}x_n) = trace(\mathbf{A}\mathbf{C}x_n(+)\mathbf{A}^T + (\mathbf{I} - \mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I} - \mathbf{A})^T) \rightarrow Min. \quad (9.4)$$

Take the derivative of equ. (9.4) w.r.t. the matrix A and equate to zero.

Chapter 9: Backward filtering and smoothing

General rule: If the matrix **P** is symmetric, then

$$\frac{\partial}{\partial \mathbf{A}} trace(\mathbf{APA}^T) = 2\mathbf{AP} \tag{9.5}$$

$$\begin{aligned} & trace(\mathbf{AC}x_n(+)\mathbf{A}^T + (\mathbf{I}-\mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I}-\mathbf{A})^T) \\ & = trace(\mathbf{AC}x_n(+)\mathbf{A}^T) + trace((\mathbf{I}-\mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I}-\mathbf{A})^T) \end{aligned} \tag{9.6}$$

$$\begin{aligned} & \frac{\partial}{\partial \mathbf{A}} trace(\mathbf{AC}x_n(+)\mathbf{A}^T + (\mathbf{I}-\mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I}-\mathbf{A})^T) \\ & = \frac{\partial}{\partial \mathbf{A}} trace(\mathbf{AC}x_n(+)\mathbf{A}^T) + \frac{\partial}{\partial \mathbf{A}} trace((\mathbf{I}-\mathbf{A})\mathbf{C}x_{nb}(+)(\mathbf{I}-\mathbf{A})^T) \end{aligned} \tag{9.7}$$

From equ. (9.4):

$$2\mathbf{AC}x_n(+) + 2(\mathbf{I}-\mathbf{A})\mathbf{C}x_{nb}(+)(-\mathbf{I}) = \mathbf{0} \tag{9.8}$$

From equ. (9.8):

$$\begin{aligned} \mathbf{A} &= \mathbf{C}x_{nb}(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1} \\ \mathbf{I}-\mathbf{A} &= \mathbf{C}x_n(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1} \end{aligned} \tag{9.9}$$

Insert equ. (9.9) into equ. (9.3):

$$\begin{aligned} \mathbf{C}x_n &= \mathbf{C}x_{nb}(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1}\mathbf{C}x_n(+)(\mathbf{C}x_{nb}(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1})^T \\ &+ \mathbf{C}x_n(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1}\mathbf{C}x_{nb}(+)(\mathbf{C}x_n(+)(\mathbf{C}x_{nb}(+) + \mathbf{C}x_n(+))^{-1})^T \end{aligned}$$

Use matrix identity: $\boldsymbol{A}(\boldsymbol{A}+\boldsymbol{B})^{-1}\boldsymbol{B} = (\boldsymbol{A}^{-1}+\boldsymbol{B}^{-1})^{-1}$

$$\Rightarrow \boxed{\mathbf{C}x_n = (\mathbf{C}x_{nb}^{-1}(+) + \mathbf{C}x_n^{-1}(+))^{-1}} \tag{9.10}$$

$$\Rightarrow \boxed{\delta\mathbf{x}_n = \mathbf{C}x_n(\mathbf{C}x_n^{-1}(+)\delta\mathbf{x}_n(+) + \mathbf{C}x_{nb}^{-1}(+)\delta\mathbf{x}_{nb}(+))} \tag{9.11}$$

Chapter 10: Comparison between Kalman Filter and Sequential Least Squares Parameter Estimation

Kalman Filter	Sequ. L. Sq. Estimation
state estimation • errors • R.P.-parameters	Estimation of parameters, constant in time
$\delta\mathbf{x}_n(+) = \delta\mathbf{x}_n(-) + \mathbf{K}_n(\delta\mathbf{y}_n - \mathbf{H}_n\delta\mathbf{x}_n(-))$ $\mathbf{K}_n = \mathbf{C}x_n(-)\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_n(-)\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1}$ $\mathbf{C}x_n(+) = \left(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n\right)\mathbf{C}x_n(-)$	$\hat{\mathbf{x}}_n = \hat{\mathbf{x}}_{n-1} + \left[\hat{\sigma}_{0\,n-1}^2\mathbf{\Sigma}(\hat{\mathbf{x}}_{n-1})^{-1} + \mathbf{A}_n^T\mathbf{P}_n\mathbf{A}_n\right]^{-1} \cdot \mathbf{A}_n^T\mathbf{P}_n\left[\mathbf{y}_n - \mathbf{A}_n\hat{\mathbf{x}}_{n-1}\right]$ $\mathbf{\Sigma}(\hat{\mathbf{x}}_n) = \hat{\sigma}_{0n}^2\left[\hat{\sigma}_{0n-1}^2\mathbf{\Sigma}(\hat{\mathbf{x}}_{n-1})^{-1} + \mathbf{A}_n^T\mathbf{P}_n\mathbf{A}_n\right]^{-1}$
$\delta\mathbf{x}_n(-) = \mathbf{F}\,\delta\mathbf{x}_{n-1}(+)$ $\mathbf{C}x_n(-) = \mathbf{F}\mathbf{C}x_{n-1}(+)\mathbf{F}^T + \mathbf{G}\mathbf{C}w_n\mathbf{G}^T$	

Chapter 10: Comparison between Kalman Filter and Sequential Least Squares Parameter Estimation

The Kalman Filter degenerates to Sequ. L. Squ. Estimation, if

- the transition matrix **F** is the Identity matrix
- the process noise covariance matrix $\mathbf{C}_w$ is zero

$$\mathbf{F} = \mathbf{I} \quad \Rightarrow \quad \delta\mathbf{x}_n(\cancel{-}) = \delta\mathbf{x}_{n-1}(\cancel{+})$$

$$\mathbf{C}w_n = \mathbf{0},\ \mathbf{F} = \mathbf{I} \quad \Rightarrow \quad \mathbf{C}x_n(\cancel{-}) = \mathbf{C}x_{n-1}(\cancel{+})$$

((-) and (+) removed, since no prediction step)

$$\mathbf{C}x_n(+) = \left(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n\right)\mathbf{C}x_n(-) \quad \Rightarrow \quad \mathbf{C}x_n = \left(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n\right)\mathbf{C}x_{n-1}$$

$$\mathbf{K}_n = \mathbf{C}x_n(-)\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_n(-)\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1} \quad \Rightarrow \quad \mathbf{K}_n = \mathbf{C}x_{n-1}\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_{n-1}\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1}$$

$$\mathbf{C}x_n = \mathbf{C}x_{n-1} - \mathbf{K}_n\mathbf{H}_n\mathbf{C}x_{n-1} \Rightarrow \mathbf{C}x_n = \mathbf{C}x_{n-1} - \mathbf{C}x_{n-1}\mathbf{H}_n^T\left(\mathbf{H}_n\mathbf{C}x_{n-1}\mathbf{H}_n^T + \mathbf{C}v_n\right)^{-1}\mathbf{H}_n\mathbf{C}x_{n-1}$$

Matrix inversion Lemma, for **B, D** being a pos. def. matrices

$$\left(\mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T\right)^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}\left(\mathbf{D} + \mathbf{C}^T\mathbf{B}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{B}$$

Chapter 10: Comparison between Kalman Filter and Sequential Least Squares Parameter Estimation

Identify: $\mathbf{B} = \mathbf{C}x_{n-1}, \mathbf{C} = \mathbf{H}_n^T, \mathbf{D} = \mathbf{C}v_n$

$$\Rightarrow \mathbf{C}x_n = \left(\mathbf{C}x_{n-1}^{-1} + \mathbf{H}_n^T \mathbf{C}v_n^{-1} \mathbf{H}_n\right)^{-1}$$

Compare to Cov.- propagation for Sequ. L. Squ. Estimation!

Similar derivation for $\delta\mathbf{x}_n$