

# Zusammenfassung Modellbildung und Systemidentifikation

## 1. Grundlagen

### 1.1 Norm

Motivation: Maß für Fehler ist notwendig

Eigenschaften:

- Abbildung von Vektor auf reelle Zahl
- $\|x\| = 0 \Leftrightarrow x = 0$
- $\|\alpha \cdot x\| = |\alpha| \cdot \|x\|$
- $\|x + y\| \leq \|x\| + \|y\|$

Beispiele:

Euklidische Norm (2-Norm):  $\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$

Summennorm (Betragsnorm):  $\|x\|_1 = \sqrt{\sum_{i=1}^n |x_i|}$

Maximumsnorm :  $\|x\|_\infty = \max_i |x_i|$

### Matrixnormen

Matrixnorm heißt verträglich mit Vektornorm wenn:

$$\|A \cdot x\| \leq \|A\| \cdot \|x\|$$

(A: Matrix, V: Vektor)

Durch Vektornorm induzierte Matrixnorm:

$$\|A\| := \max_{x \neq 0} \frac{\|A \cdot x\|}{\|x\|}$$

(A: Matrix, V: Vektor)

Bsp: Spektralnorm (induziert durch 2-Norm):

$$\|A\|_2 = \max_{\|x\|_2=1} \|A \cdot x\| = \sqrt{\{\lambda(A^T \cdot A)\}}$$

wobei  $\lambda$  die Singulärwerte der Matrix  $A$  sind.

Bsp für nicht induzierte Norm: Frobenius-Norm:

$$\|A\|_F = \sqrt{\text{trace}(A^T \cdot A)}$$

## 1.2 Dynamische Systeme

- mind 1 Eingang, mind 1 Ausgang, enthält “Gedächtnis” (=Energiespeicher)
- **linear**, wenn Superpositionsprinzip gilt:

$$G(\alpha_1 u_1 + \alpha_2 u_2) = \alpha_1 \cdot G \circ u_1 + \alpha_2 \cdot G \circ u_2 \text{ mit } \alpha_1, \alpha_2 \in \mathbb{R}$$

- **zeitinvariant**, wenn beliebige zeitl. Verschiebung von Eingangssignal entsprechende zeitl. Verschiebung am Ausgang bewirkt
- nicht-parametrische Modellierung: Sprungantwort, Impulsantwort, Frequenzgang

### 1.2.1 Parametrische Modelle

#### Zustandsraumdarstellung aus physikalischem Modell

physikalische Modellierung  $\rightarrow$  DGL (evtl. Linearisierung um Ruhelage)

$\rightarrow$  Laplace-Transformation

$$G(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots}{s^n + a_{n-1} s^{n-1} + \dots}$$

$\rightarrow$  Zustandsraummodell mit Systemmatrizen  $A_c \ B_c \ C_c$

$$A_c = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & \dots & -a_{n-1} \end{pmatrix} \quad B_c = \begin{pmatrix} 0 \\ 0 \\ \dots \\ 1 \end{pmatrix} \quad C_c = (b_0 \quad \dots \quad b_m \quad 0 \quad \dots \quad 0)$$

Abtastung liefert diskretes Modell:

$$\begin{aligned} x[k+1] &= A_d x[k] + B_d u[k] \\ y[k] &= C_d x[k] \end{aligned}$$

Diskretisierung des zeitkontinuierlichen Modells über Berechnung der Systemmatrizen:

Annahmen:

- Abtastung zu Zeitpunkten  $t = k * \Delta t$

- Stückweise konstantes Eingangssignal

$$A_d = e^{A_e \cdot \Delta t}$$

$$B_d = B_c \cdot \int_{\tau}^{\Delta t} e^{A_e \cdot \tau} d\tau$$

$$C_d = C_c$$

### 1.3 Modellarten

$u(z)$  : Systemeingang,  $v(z)$  : Fehler

#### Klassifikation nach Störung

**AR (autoregressives System)**

$$y = \frac{1}{C(z)} \cdot v$$

**MA (Moving Average)**

$$y = D(z) \cdot v$$

**ARMA (autoregressiv + moving average)**

$$y = \frac{D(z)}{C(z)} \cdot v$$

**ARX (autoregressive with external input)**

$$y = \frac{B(z)}{A(z)} u + \frac{1}{A(z)} v$$

**AR MAX (autoregressive with external input)**

$$y = \frac{B(z)}{A(z)} u + \frac{D(z)}{A(z)} v$$

**Klassifikation nach determ. Eingang (Ausgangsfehler-Modell):**

**FIR**

$$y = B(z)u + v$$

**OE (output error)**

$$y = \frac{B(z)}{A(z)}u + v$$

**BJ (Box-Jenkins)**

$$y(z) = \frac{B(z)}{A(z)} \cdot u + \frac{D(z)}{C(z)} \cdot v$$

## 2. Methode der kleinsten Fehlerquadrate

Gegeben: Messdatenpaare, Modell  $y = f(u, a)$

Ziel: Finde Parameter  $a$ , so dass  $y_i \approx f(u_i, a)$  für Daten möglichst gut erfüllt wird

→ Überbestimmtes Problem → Minimierung von Modellfehler

$$\epsilon_i = y_i - f(u_i, a)$$

Gütekriterium: quadratischer Fehler

$$L(a) = \sum_{i=1}^N \epsilon_i^2 = \epsilon^T \cdot \epsilon$$

### 2.1 MkQ für Statische Systeme

#### Parameterlineare Modelle

**Prinzip:** Kostenfunktion  $\epsilon^T \cdot \epsilon$  definieren und minimieren (Variante: Gewichtete Kostenfunktion)

mit  $\epsilon = \text{Messwert} - \text{Modell}$

Gleichungssystem in Matrix-Form mit  $n$  Messwerten:

$$\begin{pmatrix} y_1 \\ \dots \\ y_n \end{pmatrix} = \begin{pmatrix} \varphi^T(u_1) \\ \dots \\ \varphi^T(u_n) \end{pmatrix} \begin{pmatrix} a_1 \\ \dots \\ a_n \end{pmatrix}$$

$$y = \phi \cdot a$$

$$\text{Fehlerfunktion: } L(a) = \epsilon^T \epsilon = (y - \Phi a)^T (y - \Phi a) = y^T y - 2y^T \Phi a + (\Phi a)^T \Phi a$$

$$\text{Partielle Ableitung nach } a: L'(a) = -2y^T \Phi + 2a^T \Phi^T \Phi \stackrel{!}{=} 0$$

$$y^T \Phi = a^T \Phi^T \Phi$$

$$\Phi^T y = \Phi^T \Phi a$$

Gleichung umstellen ergibt Lösung :

$$a = (\phi^T \cdot \phi)^{-1} \cdot \phi \cdot y = \phi^+ \cdot y$$

Singulärwertzerlegung (SVD) kann für einfache Berechnung von  $\phi^+$  genutzt werden:

$$\phi = U \cdot \Sigma \cdot V^T \rightarrow \phi^+ = V \cdot \Sigma^+ \cdot U^T$$

$$\text{mit } \Sigma^+ = \text{diag}(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \dots, \frac{1}{\sigma_n}, 0, \dots, 0)$$

## ParameterNICHTlineare Modelle

Ansatz wie bei parameterlinearen Modellen.

**Problem:** nichtlineare Gleichungen, Minimum nicht so einfach bestimmbar

**Lösung:** Linearisierung der Fehlergleichung (in jedem Iterationsschritt)

Statt  $\epsilon$  wird  $\epsilon + \Delta\epsilon$  minimiert:

$$\epsilon + \Delta\epsilon = \underbrace{y - f(u, a)}_{\epsilon} - \underbrace{\frac{\partial f}{\partial a}(u, a \cdot \Delta a)}_{\Delta\epsilon}$$

## Gauß-Newton-Verfahren (ggf. mit Dämpfungsfaktor)

- Iterationsvorschrift:  $a_{i+1} = a_i + \Delta a_i = a_i + J_i^T(y - f(u, a_i))$
- keine gesicherte Konvergenz
- $\rightarrow$  mit Dämpfungsfaktor  $\alpha$ :  $a_{i+1} = a_i + \alpha J_i^T(y - f(u, a_i))$

## Gradientenverfahren (line search)

- Iterationsvorschrift:  $a_{i+1} = a_i + \alpha \left( \frac{\partial f}{\partial a}(u, a_i) \right)^T \epsilon$
- $\alpha$  so wählen, dass L minimal wird

## Levenberg-Marquardt-Algorithmus

- robuster als Gauß-Newton-Verfahren  $\rightarrow$  Formel siehe Skript

## 2.2 MkQ für Dynamische Systeme

### Dynamisch zeitdiskrete Systeme

- Dynamische Modelle = ARX (autoregressive) Modelle
- Differenzengleichung der Form:

$$y[k] = b_m u[k-m] + b_{m-1} u[k-m+1] + \dots + b_0 u[k] + \epsilon[k] - (a_n y[k-n] + \dots + a_1 y[k-1])$$

- (Sonderfall: FIR mit  $y[k] = b_m u[k-m] + b_{m-1} u[k-m+1] + \dots + b_0 u[k]$ ; Spezialfall Output-Error Modell)
- Gleichungssystem in Matrix-Form:

$$\underbrace{\begin{pmatrix} y[l] \\ y[l+1] \\ \dots \\ y[N-1] \end{pmatrix}}_{Y_{N-1}} = \underbrace{\begin{pmatrix} -y[l-1] & \dots & -y[l-n] & u[l] & \dots & u[l-m] \\ -y[k] & \dots & -y[l-n+1] & u[l+1] & \dots & u[l-m+1] \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -y[N-2] & \dots & -y[N-1-m] & u[N-1] & \dots & u[N-1-m] \end{pmatrix}}_{\Phi_{N-1}} \underbrace{\begin{pmatrix} a_1 \\ \dots \\ a_n \\ b_1 \\ \dots \\ b_m \end{pmatrix}}_{a_{N-1}} + \underbrace{\begin{pmatrix} e[l] \\ \dots \\ e[N-1] \end{pmatrix}}_{\epsilon_{N-1}}$$

$$Y_{N-1} = \Phi_{N-1} a_{N-1} + \epsilon_{N-1}$$

- Lösung (Minimierung von  $\epsilon_{N-1}$ ):

$$a_{N-1} = \Phi_{N-1}^+ Y_{N-1}$$

## 2.3 Dynamisch zeitkontinuierliche Systeme

Ausgangspunkt DGL:

$$y(kT) = (-y'(kT) \quad \dots \quad -y^{(n)}(kT) \quad u(kT) \quad \dots \quad u^{(m)}(kT)) \begin{pmatrix} a_1^c \\ \dots \\ a_n^c \\ b_0^c \\ \dots \\ b_m^c \end{pmatrix}$$

Problem: Ableitungen beschaffen

Beschaffung von Zeitableitungen:

- Finite Differenzen (Vorwärts/Rückwärtsdifferenzenquotient)
  - $y'(kT) \approx \frac{y(kT) - y((k-1)T)}{T}$

- $y'(kT) \approx \frac{y((k+1)T) - y(kT)}{T}$

Nachteil: Störanfällig, Messrauschen wird verstärkt, schlecht geeignet für höhere Ableitungen

b) Filterung von Ein- und Ausgangssignalen

Idee: Ausnutzen von Eigenschaften des Faltungsoperators

$$d/dt(x(t) * g(t)) = x(t) * d/dt(g(t)) \quad (g(t): \text{Impulsantwort})$$

Zustandsvariablenfilter mit Ansatz:

$$F(s) = \frac{f_0}{f_0 + f_1 s + \dots + s^n}$$

Adaptives Zustandsvariablenfilter

Iterationsvorschrift:

- Schätzung Nennerpolynom  $\hat{a}$
- Anpassung der Filterkoeffizienten

z.B. Butterworth-Filter

**Wann sind physikalische Parameter vollständig identifizierbar?**

- Anzahl physikalischer Parameter  $n_p = n + m + 1$  (Anzahl Modellparameter)
- Jacobi-Matrix  $\partial f / \partial p$  ist regulär



## 2.5 Rechentechnische Umsetzung der MkQ

- Cholesky-Zerlegung von  $\phi$  (in Dreiecksmatrizen) vereinfacht Lösung von  $a = \Phi y$ ; nur ca. 50% Rechenaufwand im Vgl. zu Gauss
- Orthogonalisierungsverfahren, Konditionierung, QR-Zerlegung -> siehe Skript

### 2.5.3 Rekursive MkQ

Herleitungsansatz: Ausgehend von  $a_n$  ergibt sich mit der nächsten Messung  $a_{n+1}$  und damit  $\Phi_{n+1} = \begin{pmatrix} \Phi_N \\ \varphi_{N+1}^T \end{pmatrix} \rightarrow$  2 Seiten Herleitung ergibt Iterationsvorschrift:

$$a_{N+1} = \underbrace{a_n}_{\text{vorheriger Parametervektor}} + \gamma_N \left( \underbrace{y[N+1]}_{\text{neuer Messwert}} - \underbrace{\Phi_{N+1}^T a_N}_{\text{vorhergesagter Ausgang}} \right)$$

Vorteile der rekursiven MkQ:

- Matrix-Inversion gespart
- Rechenaufwand konstant, unabhängig von Menge der Daten
- Rechenaufwand geringer
- online implementierbar
- deutlich weniger Speicherbedarf (vorheriger Parametervektor, aktueller Messwert)

### Wahl der Startwerte

a) Nicht-rekursive MkQ

Nachteile:

- Warten bis  $N$  Werte vorhanden sind ( $N > n + m + 1$ )
  - $P_n^{-1}$  muss erst rechenaufwendig bestimmt werden
- b) Wahl von Standardwerten  $a_0 = 0$  und  $P_0^{-1} = \alpha I$

### 2.5.4 Rekursive MkQ mit exponentiell nachlassendem Gedächtnis

- mit Wichtungsmatrix

$$W_N = \begin{pmatrix} \lambda^{N-1} & & & \\ & \cdots & & \\ & & \lambda & \\ & & & 1 \end{pmatrix}$$

- ältere Messwerte beeinflussen aktuelle Schätzung immer weniger (werden ‘vergessen’)
- Ausgleich von Arbeitspunktwechsel oder Störungen

## 2.6 Identifikation nicht-linearer Systeme

**Wiener-Modell:** dynamisch lineares System + nicht-linear statisches System

### Hammerstein-Modell

- nicht-linear statisches System + dynamisch lineares System

Einfacher Ansatz für nicht-Linearität:  $\tilde{u}[k] = r_0 + r_1 \cdot u[k] + \dots r_p \cdot u[k]^p$

Ergibt lineares Modell mit mehreren Eingängen, darstellbar in der Form  $y[k] = \phi a$ :

```

u^0 -> r_0 -> |B(z^-1)|
u^1 -> r_1 -> |---/-----|----> Y
u^2 -> r_2 -> |A(z^-1)|
u^p -> r_p -> |-----|

```

## 2.7 Modifikationen der MkQ

### 2.7.1 Totale MkQ (orthogonale Regression)

Minimierung des Fehlers der Ausgangsdaten  $\epsilon$  und des Fehlers der Eingangsdaten  $F$ :

$$y + \epsilon = (\Phi + F)a \rightarrow [(\Phi \quad y) + (F \quad \epsilon)] \begin{pmatrix} a \\ -1 \end{pmatrix} = 0$$

=> Minimierung von Fehler  $(F \quad \epsilon)$  im Sinne der Frobeniusnorm

### Einschub: Singulärwertzerlegung

Man kann Matrizen unter gewissen Voraussetzungen folgendermaßen zerlegen:

$$C = U \Sigma V^T$$

- Approximation von C durch “wegwerfen” von unteren Elementen aus  $\Sigma$
- Anwendung: Kompression

### 2.7.2 Methode der Hilfsvariablen

Anwendung: Beseitigung des Mittelwertes/Bias des Ausgangsfehlers

Prinzip: Multiplikation der Modellfehlergleichung mit sog. Hilfsvariablen:  $W^T \epsilon = W^T y - W^T \Phi a$

W ist so wählen, dass Spalten unkorreliert mit  $\epsilon$  sind.

=> Lösung der modifizierten Normalengleichung:  $a_N = (W^T \Phi)^{-1} W^T y$

Ablauf:

1. Schätzen von Parametervektor mit MkQ:  $\hat{a} = \Phi^+ y$
2. Simulation des Models  $\hat{y} = \Phi \hat{a}$
3.  $W = \dots$  (siehe Skript)
4. Schätzung für  $a_N$  mittels Hilfsvariablen (siehe modifizierte Normalengleichung)

Iteratives Wiederholen der Schritte 2-4

### 3. Subspace-based State-Space System Identification (4 SID)

Bisher:

- Identifikation von Differenzengleichungen/Übertragungsfunktionen
- aber: viele Methoden benötigen Zustandsraummodelle
- Problem: keine Infos über Zustand
- Vorteil: Schätzung von Zustandsdimension möglich

#### 3.1 Grundgleichungen, Zustandsraummodelle

Zustandsraummodell:

$$x[k+1] = Ax[k] + Bu[k]$$

- Folgezustand abhängig von aktuellem Zustand + Eingang

$$y[k] = Cx[k] + Du[k]$$

- Ausgang abhängig von Zustand + Eingang

Bekannt: N Messdatenpaare  $u[k]$ ,  $y[k]$

Problem: Weder Zustandsfolge  $x[k]$  noch Zustandsdimension  $n$  bekannt

$$\begin{pmatrix} y[0] \\ y[1] \\ \dots \\ y[k-1] \end{pmatrix} = \underbrace{\begin{pmatrix} C \\ CA \\ CA^2 \\ \dots \\ CA^{k-1} \end{pmatrix}}_{\substack{Q_{B,k} \\ \text{Beobachtbarkeitsmatrix}}} x[0] + \underbrace{\begin{pmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & D & 0 \\ CA^{k-2}B & CA^{k-3}B & CB & D \end{pmatrix}}_{H_k} \begin{pmatrix} u[0] \\ u[1] \\ \dots \\ u[k-1] \end{pmatrix}$$

Zusammenfassung in Blockmatrizen:

$$Y = \begin{pmatrix} y[0] & \dots & y[N-2k] \\ \dots & \dots & \dots \\ y[k-1] & \dots & y[N-k-1] \\ y[k] & \dots & y[N-k] \\ \dots & \dots & \dots \\ y[2k-1] & \dots & y[N-1] \end{pmatrix} = \begin{pmatrix} Y_f \\ Y_p \end{pmatrix}$$

Analog für:  $U = \begin{pmatrix} U_f \\ U_p \end{pmatrix}$

## Subspace-Gleichungen

$$Y_p = Q_{B,k} X_p + H_K U_p$$

$$Y_f = Q_{B,k} X_f + H_K U_f$$

$$X_f = A^k x_p + Q_{S,k} U_p$$

mit  $Q_{S,k} = \begin{pmatrix} A^{k-1}B & A^{k-2}B & AB & B \end{pmatrix}$  (erweiterte Steuerbarkeitsmatrix)

Durch Umformen/Einsetzen ergibt sich:

$$X_f = \dots \text{ (nur abhängig von Vergangenheit)}$$

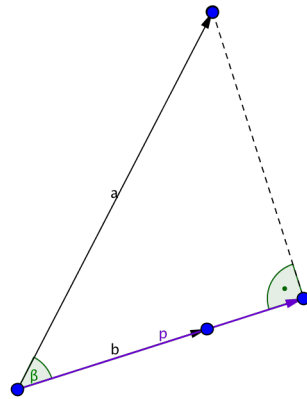
$$Y_f = Q_{B-k} L_{P,k} \begin{pmatrix} U_p \\ y_p \end{pmatrix} + H_k U_f$$

=> Für nächsten Ausgang Wissen der zukünftigen Eingabe erforderlich

## 3.2 Grundlagen: Projektion

### 3.2.1 Orthogonale Projektion

(0.5, 4.75)

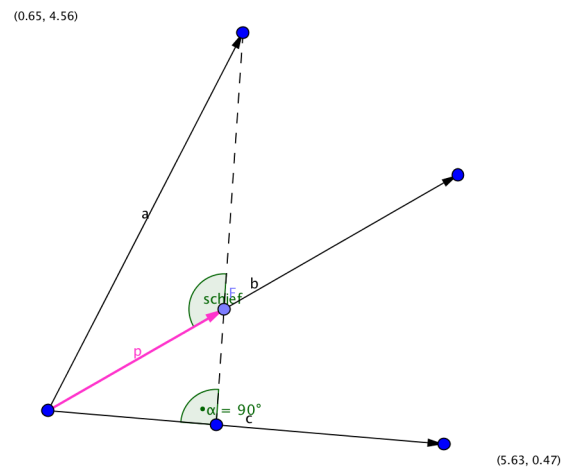


(5.58, 0.58)

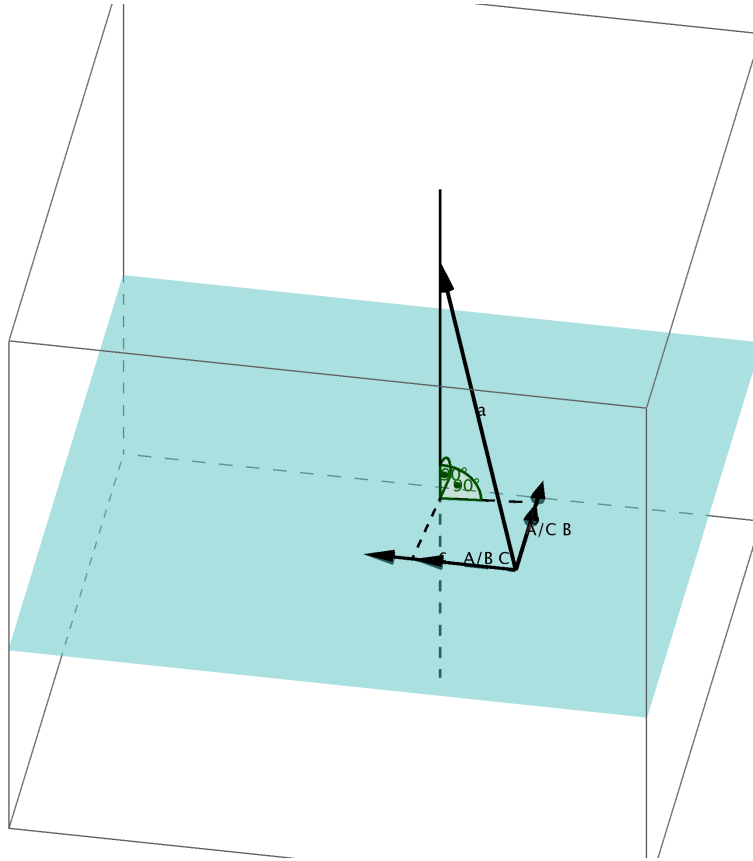
Im zweidimensionalen lässt sich der Projektor  $p$  folgendermaßen bestimmen:

$$p = a \cos \alpha \frac{|a|}{|b|} = a \frac{b^T b}{b b^T}$$

### 3.2.2 Schiefe Projektion



## Allgemeine schiefe Projektion



Vorgehen:

- Senkrechte Projektion in die von B und C aufgespannte Ebene
- Schiefe Projektion der Ebene

Definition: Allgemeine schiefe Projektion entlang des Orthogonalkomplements von C auf B:

$$A/_CB := A \cdot \Pi$$

mit:

$$\Pi = \begin{pmatrix} B^T & C^T \end{pmatrix} \begin{pmatrix} BB^T & BC^T \\ CB^T & CC^T \end{pmatrix}^+ \begin{pmatrix} B \\ 0 \end{pmatrix}$$

## Ablaufschema 4SID

- 1) Messdaten  $u[i]$ ,  $y[i]$  aufnehmen, in Hankelmatrizen  $U$ ,  $Y$  anordnen
- 2) Schiefen Prädiktor  $P$  berechnen
- 3) SVD von  $P$  (Schätzung der Systemordnung (Länge Zustandsvektor); Schätzung für Beobachtbarkeitsmatrix  $Q_{B,k}$ )
- 4) Berechnen von  $A$ ,  $C$
- 5) Berechnen von  $B$ ,  $D$

## Bewertung der 4SID Methode

### Vorteile

- Auch für Mehrgrößensysteme geeignet (mehrere Ein- und Ausgänge)
- Zustandsraumdimension muss vorher nicht bekannt sein und wird ebenfalls geschätzt

### Nachteile

- liefert unter anderem einen verzerrten Schätzer (wie MkQ)



## 4. Kalman-Filter

Ausgangspunkt: System mit Zustandsraummodell

$$\begin{aligned}x[k+1] &= Ax[k] + Bu[k] + v[k] \\ y[k] &= Cx[k] + e[k]\end{aligned}$$

### 4.1 Grundannahmen

- Systemmatrizen A,B,C bekannt
- Eingang  $u[k]$ , Ausgang  $y[k]$  bekannt
- Systemrauschen  $v[k]$  und Messrauschen  $e[k]$ : unkorrelierte, mittelwertfreie Rauschprozesse

#### Ziel

Schätzung  $\hat{x}[k]$  des Zustandsvektors  $x[k]$

#### Ansatz: Filterstruktur

$$\hat{x}[k+1] = \underbrace{A\hat{x}[k] + Bu[k]}_{\substack{\text{Prädiktionsterm} \\ \text{"a priori Schätzung"}}} + \underbrace{K[k]}_{\substack{\text{Kalman-Matrix} \\ \text{"Korrekturmatrix"}}} \underbrace{(y[k] - C\hat{x}[k])}_{\text{Korrekturterm}}$$

- Dieser Schätzer ist besonders gut, da erwartungstreu
- Kalman-Matrix so wählen, dass Kovarianz  $P[k]$  von Schätzfehler  $\tilde{x}[k+1] = \hat{x}[k+1] - x[k+1]$  minimiert wird

$$\begin{aligned}K[k] &= P[k]C^T(Y + CP[k]C^T)^{-1} \\ P[k+1] &= AP[k]A^T + V - K[k]CP[k]\end{aligned}$$

- V: Kovarianz des Systemrauschens
- Y: Kovarianz des Messrauschens (frei wählbar)

**Ergebnis:** erwartungstreuer Schätzer mit kleinster Varianz

## 4.2 Filteralgorithmus

1. Init:  $\hat{x}[0]$  &  $P[0]$  (Anfangszustand aus phys. Vorwissen wählen oder 0 setzen)
2. Prädiktion/Zeitupdate: Schätzung des Zustands auf Basis der Messwerte bis Zeitpunkt  $k$

$$\hat{x}^-[k+1] = A\hat{x}[k] + Bu[k]$$

3. Korrektur/Messupdate: Berechnung der neuen Kalman-Matrix, Korrektur der Zustandsschätzung anhand des neuen Messwertes  $y[k+1] \rightarrow$  a posteriori Schätzung
- Schritte 2. und 3. iterativ für alle Messwerte wiederholen bis Schätzung des internen Zustands konvergiert

## 4.3 Kalman-Filter als Parameterschätzer

Ausgangspunkt: statisches, parameterlineares Modell

$$y[k] = \varphi^T[k]a + e[k]$$

Modell als lineares Zustandsraummodell:

$$a[k+1] = a[k]$$

$\rightarrow$  interner Zustand = konstanter Parameter

$$y[k] = \varphi^T[k]a[k] + e[k]$$

Kalman-Filter entspricht in dieser Form der rekursiven MkQ

## 4.4 Extended Kalman-Filter

$$\begin{aligned} x[k+1] &= \underbrace{f(x[k], u[k])}_{\text{nicht lin. Fkt.}} + v[k] \\ y[k] &= \underbrace{h(x[k])}_{\text{nicht lin. Fkt.}} + e[k] \end{aligned}$$

- Linearisieren der Systemmatrizen  $A[k]$ ,  $B[k]$  und  $C[k]$  um geschätzten Zustand
- a posteriori Schätzung für Filteralgorithmus:  $\hat{x}^-[k+1] = f(\hat{x}[k], u[k])$
- sonst wie im linearen Fall

## 5. Identifikation nichtparametrischer Modelle

### 5.1 Frequenzgang mit period. Anregung

#### Variante 1: Anregung mit harmon. Eingangssignal (Sinus)

- Nach Einschwingen: Amplitude + Phase messen
- Wiederholung für versch. Frequenzen

Problem: reine Sinusschwingungen schwierig zu erzeugen

#### Variante 2: Anregung mit Trapez- oder Rechtecksignal

- Beginn bei hohen Frequenzen
- Bei kleineren Anregungsfrequenzen: Berücksichtigung der höheren harmonischen notwendig
- Aus vorherigen Messungen ist Übertragungsverhalten für hohe Frequenzen bekannt  $\rightarrow$  Signalanteile können subtrahiert werden; somit wird Grundschwingung isoliert

Nachteil: zeitaufwendig, da Warten auf Einschwingen

Ausweg: Signale mit mehreren Frequenzanteilen

Allgemeine Nachteile:

- nur stabile Systeme
- keine passive Messungen
- nur kleine Störsignale

### 5.2 Korrelationsanalyse

#### 5.2.1 Schätzung der Korrelationsfunktion

Def. Kreuzkorrelationsfolge:  $R_{uy}[j] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} u[k-j]y[k]$

Def. Autokorrelationsfolge:  $R_u[j] = R_{uu}[j] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} u[k-j]u[k]$

Eigenschaften:

- $R_{uy}[j] = R_{yu}[-j]$
- $R_u[j] = R_u[-j]$

Problem: Messung nur über endlichen Zeithorizont  $\rightarrow$  Schätzung

Schätzung der Autokorrelationsfolge:  $\hat{R}_u[j] = \frac{1}{N} \sum_{k=0}^{N-|j|-1} u[k-|j|]u[k]$

Bemerkung: Schätzung  $\hat{R}_u[j]$  ist nicht erwartungstreu:  $E\{\hat{R}_u[j]\} = (1 - \frac{|j|}{N})R_u[j]$

Andere Möglichkeit:  $\hat{R}'_u[j] = \frac{1}{N-|j|} \sum_{k=0}^{N-|j|-1} u[k-|j|]u[k]$

Dieser Schätzer ist erwartungstreu, weist aber eine um den Faktor  $N/(N-|j|)$  größere Varianz auf -> Praktisch wird ersterer verwendet

### 5.2.2 Schätzung der Gewichtsfolge / Impulsantwort

Für lineare zeitdiskrete Systeme sind Ein- und Ausgangssignal mittels Faltungssumme verknüpft:

$$y[k] = g * u = \sum_{l=0}^{\infty} g[k-l]u[l] = \sum_{l=0}^{\infty} g[l]u[k-l]$$

Für Kreuzkorrelationsfolge  $R_{uy}[j]$  gilt:

$$R_{uy}[j] = \dots (\text{siehe Skript}) = g * R_u = \sum_{l=0}^{\infty} g[l]R_u[j-l]$$

Annahme:  $R_{uy}[j]$  und  $R_u[j]$  bekannt für  $j$  mit  $-P \leq j \leq M$

=> Gleichungssystem aufstellen (siehe Skript)

Lösung mittels MkQ liefert:

$$\hat{g} = \hat{R}_u^+ \hat{R}_{uy}$$

Falls Eingangssignal weißes Rauschen, dann folgt

$$\hat{g}[i] = \frac{1}{\hat{\sigma}^2} \hat{R}_{uy}[i]$$

### 5.2.3 Parameterschätzung für Differenzengleichungen

- ermöglicht Schätzung der Parameter ohne aktive Anregung des Systems

Ausgangspunkt Differenzengleichung:

$$a_n y[k-n] + a_{n-1} y[k-(n-1)] + \dots + y[k] = b_m u[k-m] + \dots + b_0 u[k]$$

Darstellung mit Hilfe der Korrelationsfunktionen:

$$\hat{R}_{uy}[j] = \begin{pmatrix} -\hat{R}_{uy}[j-1] & \dots & -\hat{R}_{uy}[j-n] & | & \hat{R}_u[j] & \dots & \hat{R}_u[j-m] \end{pmatrix} \begin{pmatrix} a_1 \\ \dots \\ a_n \\ \hline b_0 \\ \dots \\ b_m \end{pmatrix}$$

Für  $j = n, n+1, \dots$  ergibt sich überbestimmtes Gleichungssystem

→ lösbar mittels MqQ

## 5.3 Parameterschätzung aus nicht parametrischen Modellen

### Momentenmethode

Entwicklung von  $G(s) = \int_0^\infty g(t) e^{-st} dt$  in Taylor-Reihe:

$$G(s) = \dots = \underbrace{\int_0^\infty g(t) dt}_{M_0} - s \underbrace{\int_0^\infty g(t) \cdot t dt}_{M_1} + \frac{s^2}{2} \underbrace{\int_0^\infty g(t) \cdot t^2 dt}_{M_2} + \dots$$

$$G(s) = \sum_{k=0}^{\infty} \frac{-1^k}{k!} M_k s^k$$

$M_k = \int_0^\infty g(t) \cdot t^k dt$  sind experimentell oder numerisch zu bestimmen.

=> Gleichsetzen von Taylor-Reihe mit  $G(s) = \frac{B(s)}{A(s)}$  ermöglicht Koeffizientenvergleich

## 5.4 Testsignale

### 5.4.1 Pseudo-Rausch-Binär-Signale

Beim Erzeugen von Testsignalen sind binäre Signale zu bevorzugen.

Diskretes binäres Rauschsignal (DRBs): regelloser Wechsel zwischen Werten  $+a$  und  $-a$  zu diskreten Zeitpunkten  $kT$

Zeitdiskrete Autokorrelationsfolge

$$R_u[k] = \begin{cases} a^2 & k = 0 \\ 0 & \text{sonst} \end{cases}$$

(gleiche AKF wie diskretes weißes Rauschen mit beliebiger Amplitude)

Für endl. Meßzeit ändert sich die AKF, d.h. sie muss im Einzelfall ermittelt werden

-> Übergang zu periodischen Signalen (also deterministisch), die eine ähnliche AKF besitzen (**Pseudo-Rausch-Binär-Signal PRBS**)

- Erzeugung durch (mit XORs) rückgekoppelte Schieberegister
- für Rückkopplung bestimmter Stufen erfolgt Durchlaufen aller  $2^n - 1$  Belegungen des Registers bevor sich eine Belegung wiederholt
- Periodenlänge:  $(2^n - 1) \cdot n$
- Mittelwert:  $\frac{N+1}{2} \cdot a - \frac{N-1}{2} \cdot a = \frac{a}{N}$
- Autokorrelationsfolge:  $R_u[k] = \begin{cases} a^2 & k = 0, \pm N, \pm 2N \\ \frac{a^2}{N} & \text{sonst} \end{cases}$

## 6. Statistische Parameteridentifikation

### 6.1 Maximum-Likelihood-Methode

#### 6.1.1 Grundgedanke

**bisher:** keine Annahme für Verteilungsfunktion der betrachteten Fehlersignale

**jetzt:** stochastische (meist normalverteilte) Fehlersignale mit bekannter Verteilung

Sei  $X$  eine Zufallsvariable mit Wahrscheinlichkeitsdichte  $p(x, a)$  abhängig von Parameter  $a$ . Wahrscheinlichkeit für Auftreten der Stichprobe  $x_1, \dots, x_N$ :

$$L(a) = P(x_1, a) \cdot P(x_2, a) \cdot \dots \cdot P(x_N, a)$$

Diese Funktion wird als **Likelihood-Funktion** bezeichnet.

**Maximum-Likelihood-Schätzung:**  $\hat{a} = \underset{a}{\operatorname{argmax}} L$

Einfacher zu analysieren: **Log-Likelihood-Funktion**

$$\ln(L(a)) = \ln(P(x_1, a)) + \ln(P(x_2, a)) + \dots + \ln(P(x_N, a))$$

Ableitung liefert notwendige Bedingung für Maximum:

$$\frac{\partial(L)}{\partial a} = \sum_{i=1}^N \frac{\partial \ln p(x_i, a)}{\partial a} \stackrel{!}{=} 0$$

#### 6.1.2 Maximum-Likelihood-Schätzer für statische Systeme

- statischer Prozess mit Eingang  $u$ , Ausgang  $y$
- Likelihood-Funktion als bedingte Wahrscheinlichkeit  $p(y|u, a)$

Für parameterlineare Systeme der Form  $y_i = \Phi(u_i) \cdot a + \epsilon_i$  mit normalverteiltem, mittelwertfreiem Fehler  $\epsilon_i$  mit Standardabweichung  $\sigma_\epsilon$  berechnet sich die Wahrscheinlichkeit des Auftretens der Folge  $\epsilon_1, \epsilon_2, \dots, \epsilon_N$  zu:

$$P(\epsilon) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\left(-\frac{\epsilon^T \epsilon}{2\sigma_\epsilon^2}\right)$$

Es wird nun die Wahrscheinlichkeit des Auftretens der Abweichung  $\epsilon = y - \Phi u$  maximiert.

Log-Likelihood-Funktion:

$$\ln L = \ln \left[ \frac{1}{(2\pi\sigma_\epsilon^2)^{N/2}} \cdot \exp\left(-\frac{(y - \Phi a)^T (y - \Phi a)}{2\sigma_\epsilon^2}\right) \right]$$

$$\ln L = -\frac{N}{2} \ln(2\pi\sigma_\epsilon^2) - \frac{1}{2\sigma_\epsilon^2} (y - \Phi a)^T (y - \Phi a)$$

Ableiten liefert:

$$\frac{\partial \ln L}{\partial a} = \frac{1}{\sigma_\epsilon^2} (y - \Phi a)^T \Phi \stackrel{!}{=} 0$$

Gleichung auflösen:

$$y^T \Phi = a^T \Phi^T \Phi$$

$$\Phi^T y = \Phi^T \Phi a$$

$$\hat{a} = (\Phi^T \Phi)^{-1} \Phi^T y$$

MLE liefert also für normalverteilte Fehler den gleichen Schätzer wie MkQ!

### 6.1.3 Maximum-Likelihood-Schätzer für dynamische Systeme

- MkQ: ARX-Modell
- ML: ARX-MAX-Modell? (Allgemeiner)

## 6.2 Bayes-Methode

-> auch Prozessparameter a als stoch. Größe modelliert

Bayes-Methode <-> MLE <-> MkQ

<———— Allgemeiner

————> weniger Rechenaufwand



## Zusammenfassende Fragen

- Ansatz Herleitung der MkQ; Was wird Minimiert?
- Gegeben: Messdaten Eingang + Ausgang
  - Welches Vorgehen?
  - Welcher Ansatz setzt was voraus?
  - linear  $\leftrightarrow$  nicht-linear unterscheiden
  - parameterlinear  $\leftrightarrow$  nicht-parameterlinear unterscheiden
- statisch  $\leftrightarrow$  dynamische Systeme
- Umrechnung kontinuierliche  $\leftrightarrow$  diskrete Systeme
  - Warum notwendig?
  - Wie zurück rechnen?
- 4SID nur knapp
  - Zustandsraum
  - Vorteile (Mehrgrößen und Dimension)
  - Nachteil (Verzerrter Schätzer wie MkQ; Zusammenhang physikalische Systeme, schlechte Abbildung)
- Kalman-Filter (Struktur, nicht unbedingt Details)
- Rekursive MkQ
  - keine Herleitung
  - Was bringt diese Methode?
  - Matrix-Inversion gespart
  - Rechenaufwand gering
  - online implementierbar
  - Wieviele/ welche Daten müssen gespeichert werden? Vergleich mit MkQ
- Nicht-lineare Systeme (Hammerstein-Modell)
- Frequenzgang-Messung (aufwendig, zeitintensiv, nicht-parametrisches Modell nicht besonders nützlich)
- Erzeugung Pseudo-Rausch-Binär-Signal
- Identifikation physikalischer Parameter

## Weniger relevant

- Matrixzerlegung
- Projektion
- Methode der Hilfsvariablen; totale MkQ (jeweils nur Grundkonzept wissen)