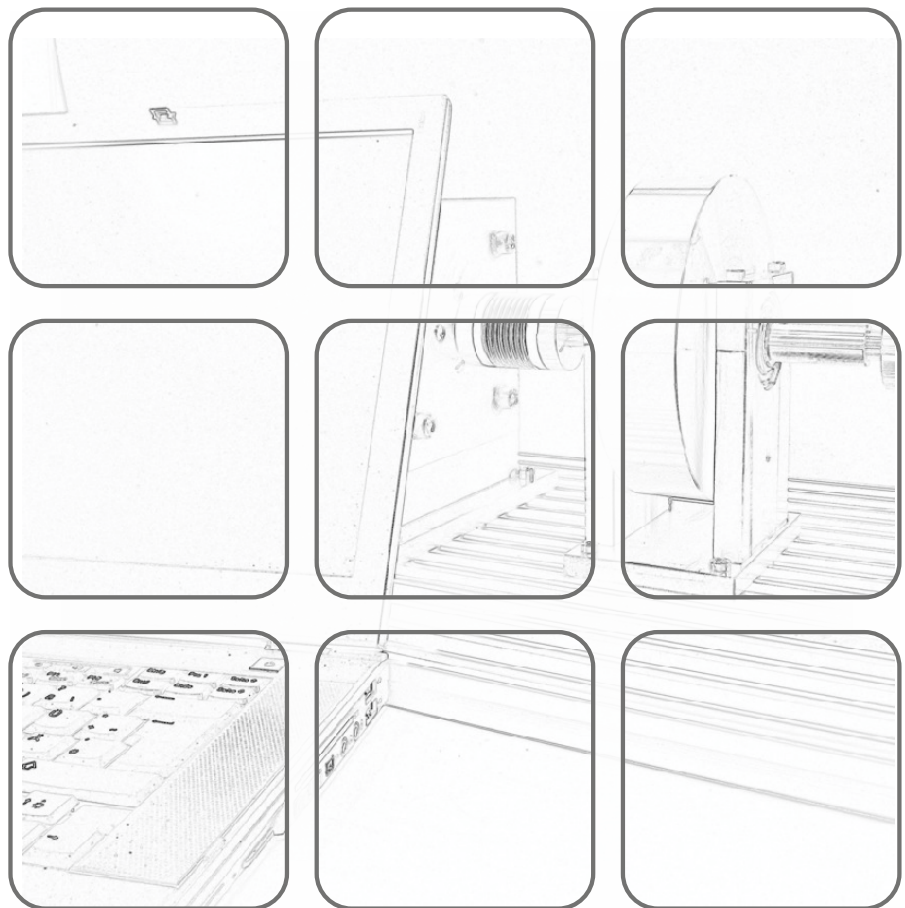


FORTGESCHRITTENE METHODEN DER NICHTLINEAREN REGELUNG

Vorlesung und Übung
Wintersemester 2015

M. Böck, T. Glück, A. Kugi, A. Steinböck



Fortgeschrittene Methoden der nichtlinearen Regelung

Vorlesung und Übung
Wintersemester 2015

M. Böck, T. Glück, A. Kugi, A. Steinböck

TU Wien
Institut für Automatisierungs- und Regelungstechnik
Gruppe für komplexe dynamische Systeme

Gusshausstrasse 27-29
1040 Wien
Telefon: +43 1 58801 – 37615
Internet: <http://www.acin.tuwien.ac.at>

© Institut für Automatisierungs- und Regelungstechnik, TU Wien

Inhaltsverzeichnis

1	Iterativ Lernende Regelung	1
1.1	Fixpunktiterationen	2
1.2	Lifted-System Darstellung	5
1.3	Grundidee der ILR	9
1.4	Stabilitäts- und Konvergenzanalyse	12
1.5	ILR Entwurfsverfahren	15
1.5.1	P-Type und PD-Type ILR	15
	<i>L</i> -Filterentwurf	16
	<i>Q</i> -Filterentwurf	16
	Gauß-Filter	17
	Savitzky-Golay-Filter	18
1.5.2	Inversionsbasierte ILR	24
1.5.3	Normoptimale ILR	25
1.6	ILR mit Feedback	26
1.6.1	Parallele Struktur	26
1.6.2	Serielle Struktur	27
1.7	Literatur	29

1 Iterativ Lernende Regelung

Iterativ Lernende Regelung (ILR) (engl. Iterative Learning Control (ILC)) basiert auf der Idee, dass die Performance eines sich wiederholenden Prozesses verbessert werden kann, indem aus vorangegangenen Versuchen (Iterationen) gelernt wird. Die ILR nutzt Information vorangegangener Iterationen gezielt aus, um den Ausgang eines System einer vorgegebenen Trajektorie folgen zu lassen. Für diese Art von Prozessen würde ein nicht-lernender Regler in jeder Iteration immer den gleichen Regelfehler aufweisen.

Um die ILR anzuwenden, muss ein Prozess folgende Bedingungen erfüllen:

- (i) Jede Iteration endet innerhalb einer festen Iterationsdauer, d. h. $t \in [0, t_f]$.
- (ii) Jede Iteration startet vom (nahezu) gleichen Anfangszustand.
- (iii) Für einen gegebenen Sollausgang $y_d(t)$ existiert eine eindeutige Steuerung $u_d(t)$.

Bei der ILR wird in jeder Iteration $j = 0, 1, \dots$ eine Steuerung $u_j(t)$ auf Basis des Ausgangsfehlers $e_j(t) = y_j(t) - y_d(t)$ berechnet und auf das System geschaltet. Ein zur Berechnung der Steuerung vielfach eingesetztes ILR Gesetz lautet für $L > 0$

$$u_{j+1}(t) = u_j(t) + L(y_j(t) - y_d(t)) . \quad (1.1)$$

Das ILR Gesetz (1.1) stellt eine Fixpunktiteration für die Steuerung $u_j(t)$ dar. Für $j \rightarrow \infty$ resultiert eine Steuerung $u_\infty(t)$, die den Ausgangsfehler $e_\infty(t) = y_\infty(t) - y_d(t) = 0$ zu Null regelt.

Die ILR zeichnet sich insbesondere dadurch aus, dass eine *hohe Regelgüte* erreicht werden kann, sie *robust gegeben Modellunsicherheiten* ist und *sich wiederholende Störungen unterdrücken lassen*. Die ILR unterscheidet sich von anderen lernenden Regler Algorithmen, wie z. B. der Adaptiven Regelung. Bei der Adaptiven Regelung werden zum Erreichen der Regelgüte die Regelparameter modifiziert - bei der ILR wird eine Steuerung modifiziert. Man kann die ILR daher auch als ein Verfahren zur Adaption einer Steuerung auffassen. Die ILR weist einige Vorteile gegenüber einer klassischen Regelung auf. Mithilfe einer Regelung ist es im Allgemeinen schwierig Totzeiten zu kompensieren. Im Rahmen der ILR ist dies einfach möglich, da sie von Iteration zu Iteration arbeitet. Aus diesem Grund ist es auch möglich, akasale Filter für den ILR Entwurf einzusetzen.

In der Literatur [1.1–1.5] werden zwei Arten der Lernenden Regelung unterschieden: *Repetitive Regelung* und *Iterativ Lernende Regelung*. Prinzipiell sind die Verfahren gleich, allerdings mit einem Unterschied.

- Bei der *Iterativ Lernenden Regelung* sind alle Iterationen vollständig unabhängig voneinander. Der Anfangszustand ist in jeder Iteration gleich.
- Bei der *Repetitiven Regelung* sind die Iteration abhängig voneinander. Der Endzustand des Systems am Ende einer Iteration bildet den Anfangszustand des Systems für die nächste Iteration. Der Anfangszustand der Iterationen variiert also von Iteration zu Iteration.

1.1 Fixpunktiterationen

Die Nullstellensuche einer Funktion

$$\gamma(\mathbf{z}) = \mathbf{0} \quad (1.2)$$

mit $\gamma(\mathbf{z}) : \mathbb{R}^N \rightarrow \mathbb{R}^N$ und $\mathbf{z}^T = [z_1 \ \dots \ z_N]$ kann in der Form

$$\psi(\mathbf{z}) = \mathbf{z} \quad (1.3)$$

angeschrieben werden. In dieser Formulierung sucht man nach einem *Fixpunkt*, d. h. einem Vektor \mathbf{z}_∞ , der

$$\psi(\mathbf{z}_\infty) = \mathbf{z}_\infty \quad (1.4)$$

erfüllt. Bei einer *Fixpunktiteration* definiert man eine Folge $\mathbf{z}_0, \mathbf{z}_1, \dots$ durch die Vorschrift

$$\mathbf{z}_{j+1} = \psi(\mathbf{z}_j), \quad j = 0, 1, 2, \dots \quad (1.5)$$

Die Konvergenzeigenschaften der Folge hängen von der Wahl der Funktion ψ ab. Es ist wichtig zu verstehen, dass für ein gegebenes Problem $\gamma(\mathbf{z}) = \mathbf{0}$ mehrere Funktionen ψ definiert werden können. Z. B. können die Funktionen

- $\psi(\mathbf{z}) = \mathbf{z} - \gamma(\mathbf{z})$
- $\psi(\mathbf{z}) = \mathbf{z} + 2\gamma(\mathbf{z})$
- $\psi(\mathbf{z}) = \mathbf{z} - \left(\frac{\partial}{\partial \mathbf{z}} \gamma\right)^{-1}(\mathbf{z})\gamma(\mathbf{z})$

gewählt werden. Die Wahl der Funktion ψ ist entscheidend für die Konvergenzeigenschaften der Fixpunktiteration. Um diese darzustellen, sind einige Definitionen notwendig.

Definition 1.1 (Konvergenz). Die Iteration (1.5) ist

- *lokal konvergent (LK)* gegen \mathbf{z}_∞ , falls es ein $\delta > 0$ gibt, so dass wenn $\|\mathbf{z}_0 - \mathbf{z}_\infty\| < \delta$ gilt, die Iteration (1.5) existiert und gegen \mathbf{z}_∞ konvergiert,
- *global konvergent (GK)*, falls die Iteration (1.5) für alle \mathbf{z}_0 gegen \mathbf{z}_∞ konvergiert.

Definition 1.2 (Stabilität). Der Fixpunkt (1.4) ist

- *stabil* (im Sinne von Lyapunov), falls es für jedes $\varepsilon > 0$ ein $\delta > 0$ gibt, so dass wenn $\|\mathbf{z}_0 - \mathbf{z}_\infty\| < \delta$ gilt, die Folge $\{\mathbf{z}_j\}$ von (1.5) existiert und $\|\mathbf{z}_j - \mathbf{z}_\infty\| < \varepsilon$ für alle $j \geq 1$ gilt,
- *attraktiv* (A), falls es ein $\delta > 0$ gibt, so dass wenn $\|\mathbf{z}_0 - \mathbf{z}_\infty\| < \delta$ gilt, die Folge $\{\mathbf{z}_j\}$ von (1.5) existiert und $\lim_{j \rightarrow \infty} \mathbf{z}_j = \mathbf{z}_\infty$, *global attraktiv* (GA) falls $\delta = \infty$,
- *asymptotisch stabil* (AS), falls er stabil und attraktiv ist und *global asymptotisch stabil* (GAS), falls er stabil und global attraktiv ist.

Attraktivität und Konvergenz sind äquivalente Konzepte. Die folgenden Äquivalenzen gelten [1.6]

$$GAS \implies GA \iff GK \implies A \iff AS . \quad (1.6)$$

Die Iteration (1.5) kann anhand der Konvergenzrate charakterisiert werden:

Definition 1.3 (Konvergenzrate einer Iteration). Die Iteration (1.5) ist

- *q-quadratisch* konvergent gegen \mathbf{z}_∞ , falls es ein $\beta > 0$ gibt, so dass

$$\|\mathbf{z}_{j+1} - \mathbf{z}_\infty\| \leq \beta \|\mathbf{z}_j - \mathbf{z}_\infty\|^2 , \quad (1.7)$$

- *q-superlinear* konvergent gegen \mathbf{z}_∞ , falls es $\{\beta_j\} > 0$ gibt, so dass

$$\|\mathbf{z}_{j+1} - \mathbf{z}_\infty\| \leq \beta_j \|\mathbf{z}_j - \mathbf{z}_\infty\| \quad \text{und} \quad (1.8)$$

- *q-linear konvergent* gegen \mathbf{z}_∞ mit dem *q-Faktor* $0 \leq \beta < 1$, falls

$$\|\mathbf{z}_{j+1} - \mathbf{z}_\infty\| \leq \beta \|\mathbf{z}_j - \mathbf{z}_\infty\| \quad (1.9)$$

für hinreichend große j gilt.

Für eine lineare Iteration

$$\mathbf{z}_{j+1} = \Psi \mathbf{z}_j , \quad j = 0, 1, 2, \dots \quad (1.10)$$

können zusätzlich folgende Aussagen getroffen werden.

Definition 1.4 (Stabilität und asymptotische Stabilität einer linearen Iteration). Eine lineare Iteration (1.10) heißt stabil, falls

$$\sup_{j \geq 1} \|\Psi^j\| < \infty , \quad (1.11)$$

sie heißt asymptotisch stabil, falls

$$\lim_{j \rightarrow \infty} \|\Psi^j\| = 0 . \quad (1.12)$$

Definition 1.5 (Spektralradius). Als Spektrum einer Matrix $\mathbf{\Gamma}$ bezeichnet man die Menge aller Eigenwerte von $\mathbf{\Gamma}$, d. h.

$$\sigma(\mathbf{\Gamma}) = \{\lambda \in \mathbb{C} \mid \det(\lambda \mathbf{E} - \mathbf{\Gamma}) = 0\} \quad (1.13)$$

und

$$\rho(\mathbf{\Gamma}) = \max_{\lambda \in \sigma(\mathbf{\Gamma})} |\lambda| \quad (1.14)$$

ist der Spektralradius von $\mathbf{\Gamma}$.

Satz 1.1. Eine lineare Iteration (1.10) ist genau dann stabil, wenn für den Spektralradius $\rho(\Psi) \leq 1$ gilt und alle Eigenwerte bei 1 nur einfach vorkommen. Sie ist genau dann asymptotisch stabil, wenn für den Spektralradius $\rho(\Psi) < 1$ gilt.

Für den Beweis sei auf die angeführte Literatur verwiesen.

Definition 1.6 (BIBO-Stabilität). Eine lineare Iteration

$$\mathbf{z}_{j+1} = \Psi \mathbf{z}_j + \Lambda \mathbf{v}_j , \quad \mathbf{z}_0 = \mathbf{0} \quad (1.15)$$

heißt BIBO-stabil, wenn zu jeder beschränkten Eingangsfolge $\{\mathbf{v}_j\}$ eine beschränkte Ausgangsfolge $\{\mathbf{z}_j\}$ gehört.

Satz 1.2. Eine lineare Iteration $\mathbf{z}_{j+1} = \Psi \mathbf{z}_j + \Lambda \mathbf{v}_j$, $\mathbf{z}_0 = \mathbf{0}$ ist genau dann BIBO-stabil, wenn für den Spektralradius $\rho(\Psi) < 1$ gilt.

Lemma 1.1. Ist $\{\mathbf{z}_j\}$ eine beschränkte Folge reeller Vektoren mit $\rho > 0, \varepsilon > 0 \in \mathbb{R}$ für die

$$\|\mathbf{z}_{j+1}\| \leq \rho \|\mathbf{z}_j\| + \varepsilon , \quad 0 \leq \rho < 1 \quad (1.16)$$

gilt, dann folgt

$$\limsup_{j \rightarrow \infty} \|\mathbf{z}_j\| \leq \frac{1}{1 - \rho} \varepsilon . \quad (1.17)$$

Beweis. Die Iteration (1.16) lässt auf

$$\begin{aligned} \|\mathbf{z}_1\| &\leq \rho \|\mathbf{z}_0\| + \varepsilon \\ \|\mathbf{z}_2\| &\leq \rho^2 \|\mathbf{z}_0\| + (1 + \rho)\varepsilon \\ &\vdots \\ \|\mathbf{z}_j\| &\leq \rho^j \|\mathbf{z}_0\| + \sum_{i=0}^{j-1} \rho^i \varepsilon = \rho^j \|\mathbf{z}_0\| + \frac{1 - \rho^j}{1 - \rho} \varepsilon \end{aligned} \quad (1.18)$$

schließen. Für $j \rightarrow \infty$ folgt $\rho^j \rightarrow 0$ für $0 \leq \rho < 1$ und damit (1.17). \square

Außerdem können die Monotonieeigenschaften der linearen Iteration (1.10) charakterisiert werden.

Definition 1.7 (Maximaler Singulärwert). Als maximalen Singulärwert einer Matrix Ψ bezeichnet man

$$\bar{\sigma}(\Psi) = \sqrt{\rho_{\max}(\Psi^T \Psi)} . \quad (1.19)$$

Der maximale Singulärwert $\bar{\sigma}(\Psi)$ gibt eine obere Schranke für die Verstärkung einer Abbildung Ψ an, denn es gilt

$$\|\Psi \mathbf{z}_j\|_2 \leq \|\Psi\|_2 \|\mathbf{z}_j\|_2 = \bar{\sigma}(\Psi) \|\mathbf{z}_j\|_2 . \quad (1.20)$$

Satz 1.3 (Monotone Konvergenz einer linearen Iteration). Die lineare Iteration $\mathbf{z}_{j+1} = \Psi \mathbf{z}_j$ ist monoton konvergent gegen $\mathbf{0}$ in der l_2 -Norm, d. h. es gilt

$$\|\mathbf{z}_{j+1}\|_2 \leq \beta \|\mathbf{z}_j\|_2 \quad \text{bzw.} \quad \|\mathbf{z}_{j+1}\|_2 \leq \beta^j \|\mathbf{z}_0\|_2 \quad (1.21)$$

für $0 \leq \beta < 1$, falls

$$\bar{\sigma}(\Psi) < 1 . \quad (1.22)$$

Für den Beweis wird auf die angeführte Literatur verwiesen.

1.2 Lifted-System Darstellung

Betrachtet wird ein lineares, zeitinvariantes SISO-System der Form

$$\dot{\mathbf{x}}_j(t) = \mathbf{A} \mathbf{x}_j(t) + \mathbf{b} u_j(t) , \quad \mathbf{x}_j(0) = \mathbf{x}_0 \quad (1.23a)$$

$$y_j(t) = \mathbf{c}^T \mathbf{x}_j(t) + d u_j(t) + v_j(t) . \quad (1.23b)$$

Es bezeichnet dabei t die Zeit und $j = 0, 1, \dots$ den Iterationsindex. Ferner ist mit $\mathbf{x}_j^T(t) = [x_1(t) \ \dots \ x_n(t)] \in \mathbb{R}^n$ der Zustandsvektor der Dimension n , mit \mathbf{x}_0 die Anfangsbedingung, mit $u(t)$ der Eingang, mit $y(t)$ der Ausgang und mit $v(t)$ das Messrauschen

gegeben. Außerdem ist \mathbf{A} die Systemmatrix, \mathbf{b} der Eingangsvektor, \mathbf{c}^T der Ausgangsvektor und d der Durchgriffparameter.

Verwendet man ein Abtast/Halteglied Nullter Ordnung, um (1.23) zu diskretisieren (ZOH-Diskretisierung (engl. Zero-Order-Hold-Discretization)), so ergibt sich mit dem Zeitindex $k = 0, 1, \dots$ und dem Zustandsvektor $\mathbf{x}_j[k] = \mathbf{x}_j(kT_a) = \mathbf{x}_j(t)$, dem Eingang $u_j[k] = u_j(kT_a) = u_j(t)$ und dem Ausgang $y_j[k] = y_j(kT_a) = y_j(t)$ für $kT_a \leq t \leq (k+1)T_a$

$$\mathbf{x}_j[k+1] = \Phi \mathbf{x}_j[k] + \gamma u_j[k], \quad \mathbf{x}_j[0] = \mathbf{x}_0 \quad (1.24a)$$

$$y_j[k] = \mathbf{c}^T \mathbf{x}_j[k] + du_j[k] + v_j[k] \quad (1.24b)$$

mit

$$\Phi = \exp(\mathbf{A}T_a) \quad \text{und} \quad \gamma = \int_0^{T_a} \exp(\mathbf{A}\tau) d\tau \mathbf{b}. \quad (1.25)$$

Die äquivalente Eingangs-Ausgangs-Darstellung lautet

$$y_j[0] = \mathbf{c}^T \mathbf{x}_0 + du_j[0] + v_j[0], \quad (1.26a)$$

$$y_j[k] = \mathbf{c}^T \Phi^k \mathbf{x}_0 + \mathbf{c}^T \sum_{m=0}^{k-1} \left(\Phi^{k-m-1} \gamma u_j[m] \right) + du_j[k] + v_j[k], \quad k = 1, 2, \dots \quad (1.26b)$$

Definiert man die Markov-Parameter¹

$$g[k] = \begin{cases} d & \text{für } k = 0 \\ \mathbf{c}^T \Phi^{k-1} \gamma & \text{für } k = 1, 2, \dots \end{cases}, \quad (1.27)$$

so findet man eine weitere Darstellung von (1.26)

$$y_j[0] = \mathbf{c}^T \mathbf{x}_0 + du_j[0] + v_j[0], \quad (1.28a)$$

$$y_j[k] = \mathbf{c}^T \Phi^k \mathbf{x}_0 + \sum_{m=0}^k (g[m] u_j[k-m]) + v_j[k], \quad k = 1, 2, \dots \quad (1.28b)$$

Anzumerken ist, dass der zweite Term in (1.28b) die diskrete Faltung $u_j[k] * g[k] = \sum_{m=\mathcal{D}} g[m] u_j[k-m]$ darstellt, wobei \mathcal{D} der diskrete Definitionsbereich ist. Definiert man den Schiebeoperator $\delta : \delta(z_j[k]) = z_j[k+1]$, dann kann für $\mathbf{x}_0 = \mathbf{0}$ die Übertragungsfunktion

$$\begin{aligned} G(\delta) &= \frac{y_j[k]}{u_j[k]} = d + \mathbf{c}^T (\delta \mathbf{E} - \Phi)^{-1} \gamma = d + \delta^{-1} \mathbf{c}^T (\mathbf{E} - \delta^{-1} \Phi)^{-1} \gamma \\ &= d + \delta^{-1} \mathbf{c}^T (\mathbf{E} + \delta^{-1} \Phi^1 + \delta^{-2} \Phi^2 + \dots) \gamma = d + \delta^{-1} \mathbf{c}^T \sum_{k=0}^{\infty} (\delta^{-k} \Phi^k) \gamma \\ &= d + \sum_{k=1}^{\infty} \mathbf{c}^T \Phi^{k-1} \gamma \delta^{-k} = \sum_{k=0}^{\infty} g[k] \delta^{-k} \end{aligned} \quad (1.29)$$

eingeführt werden.

¹ Aus der Vorlesung Automatisierung ist bekannt, dass die Markov-Parameter direkt die Impulsantwortfolge darstellen.

Bemerkung 1.1. In der Herleitung eines ILC Gesetzes spielt der relative Grad des Systems eine wichtige Rolle. Dieser ist für ein kontinuierliches System wie folgt definiert:

Definition 1.8. Das kontinuierliche System (1.23) für $d = 0$ und $v_j(t) = 0$ besitzt den relativen Grad r , wenn

$$(A) \quad \mathbf{c}^T \mathbf{A}^k \mathbf{b} = 0, \quad k = 0, 1, \dots, r-2$$

$$(B) \quad \mathbf{c}^T \mathbf{A}^{r-1} \mathbf{b} \neq 0.$$

Man überzeugt sich leicht, dass der relative Grad eines kontinuierlichen Systems der Anzahl an Zeitableitungen des Ausgangs $y_j(t)$ entspricht, bis erstmalig der Eingang $u_j(t)$ auftritt. Dies wird deutlich, wenn man fortlaufend die Zeitableitung des Ausgangs bildet:

$$y_j(t) = \mathbf{c}^T \mathbf{x}_j(t) \tag{1.30a}$$

$$\dot{y}_j(t) = \mathbf{c}^T \mathbf{A} \mathbf{x}_j(t) + \underbrace{\mathbf{c}^T \mathbf{b}}_{=0} u_j(t) \tag{1.30b}$$

$$\ddot{y}_j(t) = \mathbf{c}^T \mathbf{A}^2 \mathbf{x}_j(t) + \underbrace{\mathbf{c}^T \mathbf{A} \mathbf{b}}_{=0} u_j(t) \tag{1.30c}$$

$$\vdots$$

$$y_j^{(r-1)}(t) = \mathbf{c}^T \mathbf{A}^{r-1} \mathbf{x}_j(t) + \underbrace{\mathbf{c}^T \mathbf{A}^{r-2} \mathbf{b}}_{=0} u_j(t) \tag{1.30d}$$

$$y_j^{(r)}(t) = \mathbf{c}^T \mathbf{A}^r \mathbf{x}_j(t) + \underbrace{\mathbf{c}^T \mathbf{A}^{r-1} \mathbf{b}}_{\neq 0} u_j(t) \tag{1.30e}$$

Ist $r = n$, dann besitzt das System vollen relativen Grad. Für ein diskretes System gilt folgende Definition für den relativen Grad:

Definition 1.9. Das diskrete System (1.24) für $d = 0$ und $v_j[k] = 0$ besitzt den relativen Grad r , wenn

$$(A) \quad \mathbf{c}^T \Phi^k \gamma = 0, \quad k = 0, 1, \dots, r-2$$

$$(B) \quad \mathbf{c}^T \Phi^{r-1} \gamma \neq 0.$$

Der relative Grad eines diskreten Systems entspricht dem Zeitindex k des Ausgangs $y_j[k]$ für den erstmalig der Eingang $u_j[0]$ auftritt. Dies wird deutlich, wenn man in

Analogie zu (1.30) die Ausgangsfolge betrachtet:

$$y_j[0] = \mathbf{c}^T \mathbf{x}_0 \quad (1.31a)$$

$$y_j[1] = \mathbf{c}^T \Phi \mathbf{x}_0 + \underbrace{\mathbf{c}^T \gamma}_{=0} u_j[0] \quad (1.31b)$$

$$y_j[2] = \mathbf{c}^T \Phi^2 \mathbf{x}_0 + \underbrace{\mathbf{c}^T \Phi \gamma}_{=0} u_j[0] + \underbrace{\mathbf{c}^T \gamma}_{=0} u_j[1] \quad (1.31c)$$

\vdots

$$y_j[r-1] = \mathbf{c}^T \Phi^{r-1} \mathbf{x}_0 + \underbrace{\mathbf{c}^T \sum_{m=0}^{r-2} \Phi^{r-m-1} \gamma}_{=0} u_j[m] \quad (1.31d)$$

$$y_j[r] = \mathbf{c}^T \Phi^r \mathbf{x}_0 + \underbrace{\mathbf{c}^T \sum_{m=0}^{r-1} \Phi^{r-m-1} \gamma}_{\neq 0} u_j[m] \quad (1.31e)$$

Es stellt sich die Frage, ob das diskrete System (1.24) dann ebenso einen vollen relativen Grad besitzt. Diese Frage kann beantwortet werden, indem in (1.25) die Exponentialmatrix $\exp(\mathbf{A}\tau)$ durch ihre Reihendarstellung ersetzt, mit \mathbf{c}^T multipliziert und die Integration explizit ausgeführt wird:

$$\mathbf{c}^T \gamma = \mathbf{c}^T \int_0^{T_a} \sum_{m=0}^{\infty} \frac{(\mathbf{A}\tau)^m}{m!} \mathbf{b} d\tau \quad (1.32a)$$

$$= \mathbf{c}^T \int_0^{T_a} \mathbf{b} + \mathbf{A}\mathbf{b}\tau + \dots + \frac{1}{(n-1)!} \mathbf{A}^{n-1} \mathbf{b} \tau^{n-1} + \mathcal{O}(T_a^n) d\tau \quad (1.32b)$$

$$= \mathbf{c}^T \mathbf{b} T_a + \frac{1}{2!} \mathbf{c}^T \mathbf{A} \mathbf{b} T_a^2 + \dots + \frac{1}{n!} \mathbf{c}^T \mathbf{A}^{n-1} \mathbf{b} T_a^n + \mathcal{O}(T_a^{n+1}) \quad (1.32c)$$

Dies zeigt, dass unter Verwendung einer ZOH-Diskretisierung für (1.23) (mit $d = 0$ und $v_j(t) = 0$) das diskrete System (1.24) immer einen relativen Grad von $r = 1$ aufweist, da $\mathbf{c}^T \gamma \neq 0$, vgl. (1.31b).

Der relative Grad eines diskreten Systems entspricht dem Zeitindex r für den der Eingang $u_j[0]$ erstmalig direkt auf den Ausgang $y_j[r]$ wirkt. Der relative Grad kann daher auch als Verzögerung verstanden werden. In einer realen Anwendung kann zusätzlich zur Verzögerung durch den relativen Grad r eine Verzögerung durch die Digital-Analog- bzw. Analog-Digital-Wandlung auftreten. Benötigt die Wandlung w Abtastschritte, dann kann eine Gesamtverzögerung $m = w + r$ eingeführt und die Wandlungszeit in den Modellgleichungen mitberücksichtigt werden. Es wird nun eine Eingangs- und Ausgangsfolge für N

Zeitschritte

$$u_j[k], \quad k = 0, 1, \dots, N-1 \quad (1.33a)$$

$$v_j[k], \quad k = 0, 1, \dots, N-1 \quad (1.33b)$$

$$y_j[k], \quad k = m, m+1, \dots, N+m-1 \quad (1.33c)$$

$$y_0[k] = \mathbf{c}^T \mathbf{\Phi}^k \mathbf{x}_0, \quad k = m, m+1, \dots, N+m-1 \quad (1.33d)$$

als auch eine Sollausgangsfolge

$$y_d[k+1], \quad k = m, m+1, \dots, N+m-1 \quad (1.34a)$$

betrachtet. Der Ausgangsfehler lautet sinngemäß

$$e_j[k] = y_d[k] - y_j[k], \quad k = m, m+1, \dots, N+m-1. \quad (1.35)$$

Definiert man die Vektoren

$$\mathbf{u}_j^T = [u_j[0] \quad u_j[1] \quad \dots \quad u_j[N-1]] \in \mathbb{R}^N \quad (1.36a)$$

$$\mathbf{v}_j^T = [v_j[0] \quad v_j[1] \quad \dots \quad v_j[N-1]] \in \mathbb{R}^N \quad (1.36b)$$

$$\mathbf{y}_j^T = [y_j[m] \quad y_j[m+1] \quad \dots \quad y_j[m+N-1]] \in \mathbb{R}^N \quad (1.36c)$$

$$\mathbf{y}_d^T = [y_d[m] \quad y_d[m+1] \quad \dots \quad y_d[m+N-1]] \in \mathbb{R}^N \quad (1.36d)$$

$$\mathbf{y}_0^T = [y_0[m] \quad y_0[m+1] \quad \dots \quad y_0[m+N-1]] \in \mathbb{R}^N \quad (1.36e)$$

$$\mathbf{e}_j^T = \mathbf{y}_d^T - \mathbf{y}_j^T = [e_j[m] \quad e_j[m+1] \quad \dots \quad e_j[m+N-1]] \in \mathbb{R}^N, \quad (1.36f)$$

so ergibt sich aus (1.28) die sogenannte Lifted-System Darstellung

$$\mathbf{y}_j = \mathbf{y}_0 + \mathbf{G} \mathbf{u}_j + \mathbf{v}_j \quad (1.37)$$

mit der Matrix

$$\mathbf{G} = \begin{bmatrix} g[m] & 0 & \dots & 0 \\ g[m+1] & g[m] & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ g[m+N-1] & g[m+N-2] & \dots & g[m] \end{bmatrix} \in \mathbb{R}^{N \times N}. \quad (1.38)$$

Zu beachten ist, dass zum einen stets $g[m] \neq 0$ gilt und zum anderen die Matrix \mathbf{G} Toeplitz ist. Eine Matrix \mathbf{G} wird Toeplitz-Matrix genannt wenn die Einträge \mathbf{G}_{ij} nur von der Differenz $i - j$ der Indizes abhängen. Es sei angemerkt, dass lineare, zeitvariante Systeme ebenso in einer Lifted-System Darstellung darstellbar sind, siehe z. B. [1.7].

1.3 Grundidee der ILR

Die Grundidee der ILR besteht darin, in jeder Iteration $j = 0, 1, 2, \dots$ eine Steuerung \mathbf{u}_j mithilfe einer Iteration

$$\mathbf{u}_{j+1} = \psi(\mathbf{u}_j, \mathbf{e}_j(\mathbf{u}_j)), \quad j = 0, 1, 2, \dots \quad (1.39)$$

zu berechnen. Dazu muss in jeder Iteration $j = 0, 1, 2, \dots$ der Ausgang \mathbf{y}_j und die Steuerung \mathbf{u}_j zwischengespeichert, der Ausgangsfehler \mathbf{e}_j gebildet und eine Steuerung \mathbf{u}_{j+1} berechnet werden. Abbildung 1.1 veranschaulicht die Grundidee der ILR. Um die Gestalt von (1.39)

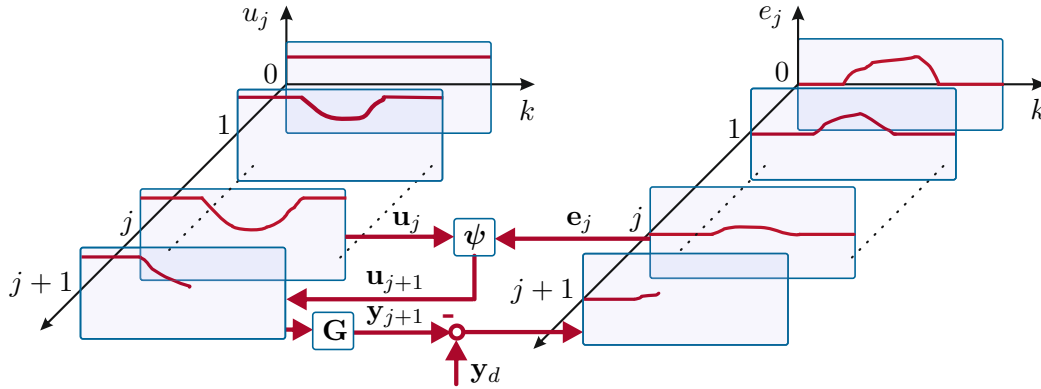


Abbildung 1.1: Grundidee der ILR.

zu motivieren und näher zu spezifizieren, wird die Minimierungsaufgabe

$$\min_{\mathbf{u}_{j+1}} J(\mathbf{u}_{j+1}) = \underbrace{\frac{1}{2} \mathbf{e}_{j+1}^T \mathbf{V} \mathbf{e}_{j+1}}_{J_1(\mathbf{u}_{j+1})} + \underbrace{\frac{1}{2} \mathbf{u}_{j+1}^T \mathbf{S} \mathbf{u}_{j+1} + \frac{1}{2} (\mathbf{u}_{j+1} - \mathbf{u}_j)^T \mathbf{R} (\mathbf{u}_{j+1} - \mathbf{u}_j)}_{J_2(\mathbf{u}_{j+1})} \quad (1.40)$$

$$\text{u.B.v.} \quad \mathbf{e}_{j+1} = \mathbf{y}_d - \mathbf{y}_{j+1} = \mathbf{e}_j + \mathbf{G} \mathbf{u}_j - \mathbf{G} \mathbf{u}_{j+1}$$

mit den positiv definiten Matrizen

$$\mathbf{R} > 0 \quad (1.41a)$$

$$\mathbf{S} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R}) (\mathbf{Q}^{-1} - \mathbf{E}) > 0 \quad (1.41b)$$

$$\mathbf{V} = \mathbf{G}^{-T} \mathbf{R} \mathbf{L} (\mathbf{E} - \mathbf{G} \mathbf{L})^{-1} > 0 \quad (1.41c)$$

betrachtet. Es bezeichnet dabei \mathbf{Q} eine sogenannte Q -Filtermatrix und \mathbf{L} eine Verstärkungsmatrix, die anhand von (1.41b) und (1.41c) auch in der Form

$$\mathbf{Q} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R})^{-1} (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R}) \quad (1.42a)$$

$$\mathbf{L} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{V} \quad (1.42b)$$

angegeben werden können. Setzt man in (1.40) den Ausgangsfehler \mathbf{e}_{j+1} in das Kostenfunktional ein, dann folgt

$$\begin{aligned} J_1(\mathbf{u}_{j+1}) &= \frac{1}{2} (\mathbf{e}_j + \mathbf{G} \mathbf{u}_j - \mathbf{G} \mathbf{u}_{j+1})^T \mathbf{V} (\mathbf{e}_j + \mathbf{G} \mathbf{u}_j - \mathbf{G} \mathbf{u}_{j+1}) \\ &= \frac{1}{2} \left(\mathbf{e}_j^T \mathbf{V} \mathbf{e}_j + \mathbf{u}_j^T \mathbf{G}^T \mathbf{V} \mathbf{e}_j - 2 \mathbf{u}_{j+1}^T \mathbf{G}^T \mathbf{V} \mathbf{e}_j \right. \\ &\quad \left. + \mathbf{e}_j^T \mathbf{V} \mathbf{G} \mathbf{u}_j + \mathbf{u}_j^T \mathbf{G}^T \mathbf{V} \mathbf{G} \mathbf{u}_j - 2 \mathbf{u}_{j+1}^T \mathbf{G}^T \mathbf{V} \mathbf{G} \mathbf{u}_j + \mathbf{u}_{j+1}^T \mathbf{G}^T \mathbf{V} \mathbf{G} \mathbf{u}_{j+1} \right) \end{aligned} \quad (1.43)$$

bzw.

$$\begin{aligned} J_2(\mathbf{u}_{j+1}) &= \frac{1}{2} \mathbf{u}_{j+1}^T \mathbf{S} \mathbf{u}_{j+1} + \frac{1}{2} (\mathbf{u}_{j+1}^T - \mathbf{u}_j^T) \mathbf{R} (\mathbf{u}_{j+1} - \mathbf{u}_j) \\ &= \frac{1}{2} (\mathbf{u}_{j+1}^T \mathbf{S} \mathbf{u}_{j+1} + \mathbf{u}_{j+1}^T \mathbf{R} \mathbf{u}_{j+1} - 2 \mathbf{u}_{j+1}^T \mathbf{R} \mathbf{u}_j + \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j) . \end{aligned} \quad (1.44)$$

Damit \mathbf{u}_{j+1} ein Minimum des statischen Optimierungsproblems (1.40) ist, muss die notwendige Bedingung erster Ordnung

$$\left(\frac{\partial}{\partial \mathbf{u}_{j+1}} J \right) (\mathbf{u}_{j+1}) = \mathbf{0} \quad (1.45)$$

erfüllt sein. Anwendung der Optimalitätsbedingung (1.45) resultiert in

$$(\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R}) \mathbf{u}_{j+1} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R}) \mathbf{u}_j + \mathbf{G}^T \mathbf{V} \mathbf{e}_j \quad (1.46)$$

bzw. mit (1.42) in das ILR Gesetz

$$\mathbf{u}_{j+1} = \mathbf{Q} (\mathbf{u}_j + \mathbf{L} \mathbf{e}_j) . \quad (1.47)$$

Die \mathbf{Q} -Filtermatrix ist dabei gegeben durch

$$\mathbf{Q} = \begin{bmatrix} q[0] & q[-1] & \cdots & q[-(N-1)] \\ q[1] & q[0] & \cdots & q[-(N-2)] \\ \vdots & \vdots & \ddots & \vdots \\ q[N-1] & q[N-2] & \cdots & q[0] \end{bmatrix} \in \mathbb{R}^{N \times N} \quad (1.48)$$

und die Verstärkungsmatrix durch

$$\mathbf{L} = \begin{bmatrix} l[0] & l[-1] & \cdots & l[-(N-1)] \\ l[1] & l[0] & \cdots & l[-(N-2)] \\ \vdots & \vdots & \ddots & \vdots \\ l[N-1] & l[N-2] & \cdots & l[0] \end{bmatrix} \in \mathbb{R}^{N \times N} . \quad (1.49)$$

Das ILR Gesetz (1.47) kann auch in der Form

$$u_{j+1}[k] = Q(\delta)(u_j[k] + L(\delta)e_j[k+m]) \quad (1.50)$$

bzw.

$$u_{j+1}[k] = q[k] * (u_j[k] + l[k] * e_j[k+m]) \quad (1.51)$$

angegeben werden. Damit entsprechen die \mathbf{Q} -Filterung und die Multiplikation mit der Verstärkungsmatrix nicht-kausalen Filterungen mit den Übertragungsfunktionen

$$Q(\delta) = \cdots + q[-2]\delta^2 + q[-1]\delta^1 + q[0] + q[1]\delta^{-1} + q[2]\delta^{-2} + \cdots \quad (1.52a)$$

$$L(\delta) = \cdots + l[-2]\delta^2 + l[-1]\delta^1 + l[0] + l[1]\delta^{-1} + l[2]\delta^{-2} + \cdots \quad (1.52b)$$

und den Impulsantwortfolgen

$$q[k] \quad \text{für} \quad k = -(N-1), -(N-2), \dots, (N-2), (N-1) \quad (1.53a)$$

$$l[k] \quad \text{für} \quad k = -(N-1), -(N-2), \dots, (N-2), (N-1) . \quad (1.53b)$$

Das ILR Gesetz (1.50) kann unterschiedlich interpretiert werden. Zum einen stellt für $Q(\delta) = L(\delta) = 1$ das ILR Gesetz (1.47) einen Integrator im Iterationsbereich dar. Aus der Vorlesung Automatisierung ist bekannt, dass ein Integrator einen konstanten Ausgangsfehler e_j zu Null regeln kann. Zum anderen kann (1.47) als eine Fixpunktiteration für die Nullstellensuche der Funktion $e_j(u_j)$ verstanden werden. Eine weitere Interpretation der ILR ergibt sich, wenn man die ILR als das Online-Lösen der Optimierungsaufgabe (1.40) mit einem Einfeldschießverfahren interpretiert. Das Aufschalten der Steuerung entspricht einem Einfeldschießen (Lösen der Anfangswertaufgabe) und das ILR Gesetz (1.47) einer überlagerten Nullstellensuche. Im Unterschied zu einem Offline-Verfahren muss ein Online-Verfahren robust gegenüber Messrauschen und Variation der Anfangsbedingung sein. Wie später gezeigt wird, unterdrückt die Q -Filterung den Einfluss des Messrauschens und den von sich nicht wiederholenden Störungen.

1.4 Stabilitäts- und Konvergenzanalyse

Es soll nun die Stabilität des ILR Gesetzes (1.47)

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{u}_j + \mathbf{L}\mathbf{e}_j) , \quad (1.54)$$

angewendet auf die Lifted-System Darstellung (1.37)

$$\mathbf{y}_j = \mathbf{y}_0 + \mathbf{G}\mathbf{u}_j + \mathbf{v}_j , \quad (1.55)$$

gezeigt werden. Dazu sind folgende Annahmen zu treffen:

- (i) Jede Iteration endet innerhalb einer festen Iterationsdauer $t_f = NT_a$.
- (ii) Der Prozess startet immer von der gleichen Anfangsbedingung, d. h. $\mathbf{y}_0 = \text{konst.}$
- (iii) Für einen gegebenen Sollausgang \mathbf{y}_d existiert eine eindeutige Steuerung \mathbf{u}_d , d. h. es gilt $\mathbf{y}_d = \mathbf{y}_0 + \mathbf{G}\mathbf{u}_d$.
- (iv) Die Messung \mathbf{y}_j ist ideal, das heißt, dass $\mathbf{v}_j = \mathbf{0}$ gilt.

Mit dem Ausgangsfehler $\mathbf{e}_j = \mathbf{y}_d - \mathbf{y}_j$ nach (1.36f) und der Lifted-System Darstellung (1.37) findet man

$$\mathbf{e}_j = \mathbf{y}_d - \mathbf{G}\mathbf{u}_j - \mathbf{y}_0 \quad (1.56)$$

und damit die lineare Iteration

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{u}_j + \mathbf{Q}\mathbf{L}(\mathbf{y}_d - \mathbf{y}_0) \quad (1.57)$$

mit dem Fixpunkt $\mathbf{u}_{j+1} = \mathbf{u}_j = \mathbf{u}_\infty$

$$\mathbf{u}_\infty = \left(\mathbf{E} - \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G}) \right)^{-1} \mathbf{Q}\mathbf{L}(\mathbf{y}_d - \mathbf{y}_0) . \quad (1.58)$$

Führt man den Eingangsfehler $\bar{\mathbf{u}}_j = \mathbf{u}_j - \mathbf{u}_\infty$ ein, so ergibt sich die Fehlerdynamik

$$\bar{\mathbf{u}}_{j+1} = \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\bar{\mathbf{u}}_j , \quad (1.59)$$

für welche Satz 1.1 folgende Stabilitätsaussage liefert:

Satz 1.4 (Asymptotische Stabilität des ILR Gesetzes). *Die Fehlerdynamik (1.59) des ILR Gesetzes (1.47), angewendet auf die Lifted-System Darstellung (1.37), ist asymptotisch stabil, falls*

$$\rho(\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})) < 1 . \quad (1.60)$$

Für den stationären Ausgangsfehler gilt aufgrund von (1.56) und (1.58)

$$\begin{aligned} \mathbf{e}_\infty &= \mathbf{y}_d - \mathbf{y}_0 - \mathbf{G}\mathbf{u}_\infty \\ &= \left(\mathbf{E} - \mathbf{G}(\mathbf{E} - \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G}))^{-1} \mathbf{Q}\mathbf{L} \right) (\mathbf{y}_d - \mathbf{y}_0) . \end{aligned} \quad (1.61)$$

Man erkennt, dass nur für $\mathbf{Q} = \mathbf{E}$ der Ausgangsfehler verschwindet, d. h.

$$\mathbf{e}_\infty = \mathbf{0} . \quad (1.62)$$

Dies wird auch deutlich, wenn man (1.56) um einen Index verschiebt und das ILR Gesetz (1.47) einsetzt:

$$\mathbf{e}_{j+1} = \mathbf{y}_d - \mathbf{G}\mathbf{Q}(\mathbf{u}_j + \mathbf{L}\mathbf{e}_j) - \mathbf{y}_0 . \quad (1.63)$$

Eliminiert man noch \mathbf{u}_j anhand von (1.56), so folgt

$$\mathbf{e}_{j+1} = \mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}\mathbf{e}_j + (\mathbf{E} - \mathbf{G}\mathbf{Q}\mathbf{G}^{-1})(\mathbf{y}_d - \mathbf{y}_0) . \quad (1.64)$$

Mit dem Satz 1.1 folgt somit für die Ausgangsfehlerdynamik:

Satz 1.5 (Stabilität der Ausgangsfehlerdynamik). *Die Ausgangsfehlerdynamik (1.64) der Lifted-System Darstellung (1.37), (1.47) ist asymptotisch stabil, falls*

$$\rho(\mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}) < 1 . \quad (1.65)$$

In einer Norm lautet (1.64)

$$\|\mathbf{e}_{j+1}\| \leq \underbrace{\|\mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}\|}_{=\sigma} \|\mathbf{e}_j\| + \underbrace{\|\mathbf{E} - \mathbf{G}\mathbf{Q}\mathbf{G}^{-1}\|}_{=\gamma} \underbrace{\|\mathbf{y}_d - \mathbf{y}_0\|}_{=\varepsilon} , \quad 0 \leq \sigma < 1 . \quad (1.66)$$

Lemma 1.1 liefert dann die asymptotische Aussage

$$\limsup_{j \rightarrow \infty} \|\mathbf{e}_j\| \leq \frac{1}{1 - \sigma} \gamma \varepsilon . \quad (1.67)$$

In der Literatur findet man häufig ILR Gesetze, die kein Q -Filter verwenden, d. h. $\mathbf{Q} = \mathbf{E}$. Die vorangegangenen Überlegungen zeigen, dass dies notwendig ist, um perfektes Folgeverhalten ($\mathbf{e}_\infty = \mathbf{0}$) zu gewährleisten. Die Verwendung eines Q -Filters verbessert allerdings die Robustheit gegenüber Variationen der Anfangsbedingungen \mathbf{y}_0 und gegenüber Messrauschen \mathbf{v}_j . Um dies zu plausibilisieren, nimmt man an, dass die Anfangsbedingung nicht in jeder Iteration gleich ist, d. h. $\mathbf{y}_0 = \mathbf{y}_{0,j}$, und dass die Messung nicht ideal ist, d. h. ein Rauschprozess \mathbf{v}_j wirkt. Dann lautet (1.57)

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{u}_j + \mathbf{Q}\mathbf{L}(\mathbf{y}_d - \mathbf{y}_{0,j} - \mathbf{v}_j) . \quad (1.68)$$

Ohne die Verwendung eines Q -Filters ($\mathbf{Q} = \mathbf{E}$) würden das Messrauschen und Variationen der Anfangsbedingung direkt verstärkt und in die Steuerung für die nächste Iteration eingehen. Es ist unmittelbar einsichtig, dass z. B. ein Tiefpassfilter hochfrequentes Rauschen unterdrücken kann. Da das Q -Filter auf der ganzen rechten Seite von (1.68) wirkt, schaltet man damit das Lernen für hochfrequente Signalanteile aus.

Im Folgenden soll die Konvergenz der Ausgangsfehleriteration charakterisiert werden. Setzt man in (1.64) den Ausgangsfehler $\mathbf{e}_{j+1} = \mathbf{y}_d - \mathbf{y}_{j+1}$ ein, so findet man

$$\mathbf{y}_d - \mathbf{y}_{j+1} = (\mathbf{E} - \mathbf{G}\mathbf{Q}\mathbf{L})(\mathbf{y}_d - \mathbf{y}_j) + \mathbf{G}(\mathbf{E} - \mathbf{Q})\mathbf{u}_j \quad (1.69)$$

und mit der Lifted-System Darstellung (1.37) ergibt sich

$$\begin{aligned} \mathbf{y}_d - \mathbf{y}_{j+1} &= (\mathbf{E} - \mathbf{G}\mathbf{Q}\mathbf{L})(\mathbf{y}_d - \mathbf{y}_j) + \mathbf{G}(\mathbf{E} - \mathbf{Q})\mathbf{G}^{-1}(\mathbf{y}_j - \mathbf{y}_0) \\ &= (\mathbf{E} - \mathbf{G}\mathbf{Q}\mathbf{L})\mathbf{y}_d + \mathbf{G}\mathbf{Q}(\mathbf{L} - \mathbf{G}^{-1})\mathbf{y}_j - \mathbf{G}(\mathbf{E} - \mathbf{Q})\mathbf{G}^{-1}\mathbf{y}_0 \end{aligned} \quad (1.70)$$

bzw. folgt die Iteration für den Ausgang zu

$$\mathbf{y}_{j+1} = \mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}\mathbf{y}_j + \mathbf{G}\mathbf{Q}\mathbf{L}\mathbf{y}_d + \mathbf{G}(\mathbf{E} - \mathbf{Q})\mathbf{G}^{-1}\mathbf{y}_0 . \quad (1.71)$$

Außerdem folgt mit dem Fixpunkt der Steuerung (1.58) und $\mathbf{u}_\infty = \mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{y}_0 - \mathbf{e}_\infty)$ die Beziehung

$$\mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{y}_0 - \mathbf{e}_\infty) = \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{y}_0 - \mathbf{e}_\infty) + \mathbf{Q}\mathbf{L}(\mathbf{y}_d - \mathbf{y}_0) \quad (1.72)$$

und mit $\mathbf{y}_d = \mathbf{e}_j + \mathbf{y}_j$ lässt sich

$$(\mathbf{e}_{j+1} - \mathbf{e}_\infty + \mathbf{y}_{j+1} - \mathbf{y}_0) = \mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}(\mathbf{e}_j - \mathbf{e}_\infty + \mathbf{y}_j - \mathbf{y}_0) + \mathbf{G}\mathbf{Q}\mathbf{L}(\mathbf{y}_d - \mathbf{y}_0) \quad (1.73)$$

folgern. Setzt man schließlich die Ausgangsiteration (1.71) ein, so erhält man die Ausgangsfehleriteration

$$\mathbf{e}_{j+1} - \mathbf{e}_\infty = \mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}(\mathbf{e}_j - \mathbf{e}_\infty) \quad (1.74)$$

und in der l_2 -Norm ergibt sich

$$\|\mathbf{e}_{j+1} - \mathbf{e}_\infty\|_2 \leq \|\mathbf{G}\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}\|_2 \|\mathbf{e}_j - \mathbf{e}_\infty\|_2 . \quad (1.75)$$

Definition 1.3 liefert nunmehr die Aussage:

Satz 1.6 (*q-lineare Konvergenz der Ausgangsfehleriteration*). Die Ausgangsfehleriteration (1.74) der Lifted-System Darstellung (1.37), (1.47) ist q-lineare konvergent gegen \mathbf{e}_∞ in der l_2 -Norm, d. h. es gilt

$$\|\mathbf{e}_{j+1} - \mathbf{e}_\infty\|_2 \leq \beta \|\mathbf{e}_j - \mathbf{e}_\infty\|_2 \quad (1.76)$$

für $0 \leq \beta < 1$, falls

$$\bar{\sigma}(\mathbf{GQ}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}) = \beta < 1 . \quad (1.77)$$

Wählt man $\mathbf{L} = \mathbf{G}^{-1}$, dann ist $\beta = 0$ und die Ausgangsfehleriteration (1.74) konvergiert in einem Schritt zu \mathbf{e}_∞ .

Um große Transienten während der Iterationen zu vermeiden, muss monotone Konvergenz der Ausgangsfehleriteration (1.74) sichergestellt werden. Mit der Ausgangsfehleriteration (1.74) folgt anhand von Satz 1.3:

Satz 1.7 (*Monotone Konvergenz der Ausgangsfehleriteration*). Die Ausgangsfehleriteration (1.74) der Lifted-System Darstellung (1.37), (1.47) ist monoton konvergent gegen \mathbf{e}_∞ in der l_2 -Norm, d. h. es gilt

$$\|\mathbf{e}_{j+1} - \mathbf{e}_\infty\|_2 \leq \beta \|\mathbf{e}_j - \mathbf{e}_\infty\|_2 \quad \text{bzw.} \quad \|\mathbf{e}_{j+1} - \mathbf{e}_\infty\|_2 \leq \beta^j \|\mathbf{e}_0 - \mathbf{e}_\infty\|_2 \quad (1.78)$$

für $0 \leq \beta < 1$, falls

$$\bar{\sigma}(\mathbf{GQ}(\mathbf{E} - \mathbf{L}\mathbf{G})\mathbf{G}^{-1}) = \beta < 1 . \quad (1.79)$$

1.5 ILR Entwurfsverfahren

Bisher wurde lediglich die Stabilität und Konvergenz des ILR Gesetzes (1.47) untersucht. Offen ist die Frage, wie das Q-Filter \mathbf{Q} und die Verstärkungsmatrix \mathbf{L} entworfen werden. Aus der Vielzahl von Entwurfsverfahren sollen drei vorgestellt werden:

- (i) P-Type und PD-Type ILR
- (ii) Inversionsbasiertes ILR
- (iii) Normoptimale ILR

1.5.1 P-Type und PD-Type ILR

Bei der P-Type und PD-Type ILR [1.8, 1.9] bedient man sich bekannter Konzepte der PID-Ausgangsregelung. Die Rückführung beinhaltet also je nach Typ einen Proportional- und/ oder einen Differentialanteil des Ausgangsfehlers \mathbf{e}_j . Anzumerken ist, dass meist kein Integralterm verwendet wird, da das ILC Gesetz (1.47) ohnehin schon integrierendes Verhalten aufweist. Das Verfahren ist ein Trial-and-Error Verfahren, das in zwei Schritten

erfolgt. Im **ersten Schritt** wird eine Verstärkungsmatrix \mathbf{L} entworfen und die Einstellparameter so eingestellt, dass die Stabilität nach Satz 1.5 gewährleistet wird und sich eine hinreichend gute Regelgüte einstellt. Im **zweiten Schritt** wird das ILR Gesetz um ein Q -Filter erweitert um Robustheit gegenüber Messrauschen zu sichern. Um das Konzept der P-Type und PD-Type ILR zu erklären, wird zunächst der Entwurf der Verstärkungsmatrix \mathbf{L} aufgezeigt (L -Filterentwurf).

L -Filterentwurf

Die einfachste Möglichkeit die Verstärkungsmatrix \mathbf{L} zu entwerfen, besteht in der Wahl einer diagonalen Verstärkungsmatrix mit einem Freiheitsgrad, d. h. einem P-Type ILR Gesetz in der Form

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{u}_j + k_p \mathbf{E} \mathbf{e}_j) . \quad (1.80)$$

Dabei bezeichnet $k_p > 0$ einen Einstellparameter. Bei der PD-Type ILR wird der Ausgangsfehler und zusätzlich eine numerische Approximation der Zeitableitung des Ausgangsfehlers verwendet. Ein PD-Type ILR Gesetz, welches den Zentralen Differenzenquotienten nutzt, lautet für $k = 1, 2, \dots, N-2$

$$u_{j+1}[k] = q[k] * (u_j[k] + k_p e_j[k+r] + \frac{k_d}{2T_a} (e_j[k+r+1] - e_j[k+r-1])) . \quad (1.81)$$

$k_p > 0$ und $k_d > 0$ sind wiederum Einstellparameter. Für die Randiteration bei $k = 0$ und $k = N-1$ kann z. B. ein Vorwärts- bzw. ein Rückwärts-Differenzenquotient angesetzt werden. Führt man die Ableitungsmatrix

$$\mathbf{D} = \frac{1}{T_a} \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ -1/2 & 0 & 1/2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & -1/2 & 0 & 1/2 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad (1.82)$$

ein, so folgt für den PD-Type die zu (1.81) äquivalente Lifted-System Darstellung

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{u}_j + \underbrace{(k_p \mathbf{E} + k_d \mathbf{D})}_{\mathbf{L}} \mathbf{e}_j) . \quad (1.83)$$

Die Gleichung (1.83) verdeutlicht, dass die Stabilität des PD-Typs wieder anhand von Satz 1.5 charakterisiert werden kann.

Q -Filterentwurf

Der Einsatz eines Q -Filters verhindert das Lernen von sich nicht wiederholenden Störungen und Messrauschen. Da eine akausale Filterung im Rahmen der ILR möglich ist, bietet es sich an, einen gleitenden Mittelwert zu verwenden. Es werden im Folgenden zwei Entwurfsmethoden vorgestellt:

- (i) Gauß-Filter
- (ii) Savitzky-Golay-Filter

Der Entwurf eines Savitzky-Golay-Filters erlaubt es zusätzlich Ableitungsfiler höherer Ordnung zu entwerfen. Diese Ableitungsfiler, die im Grunde Differenzenquotienten darstellen, können zur Konstruktion einer Ableitungsfiltermatrix \mathbf{D} verwendet werden, vgl. (1.82).

Gauß-Filter

Ein einfacher Entwurf eines akausalen Glättungsfilters (Tiefpassfilters) ist mithilfe eines Gauß-Filters [1.10] möglich. Dazu betrachtet man die Gauß-Verteilung

$$q_G(t) = \frac{1}{\sigma_q \sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma_q^2}\right) \quad (1.84)$$

mit der Standardabweichung σ_q , die mit der 3-dB Bandbreite f_c über

$$\sigma_q = \frac{\sqrt{\ln(2)}}{2\pi f_c} \quad (1.85)$$

im Zusammenhang steht. Mit $q_G[k] = q_G(kT_a) = q_G(t)$ für $kT_a \leq t \leq (k+1)T_a$ findet man

$$q_G[k] = \frac{T_a}{\sigma_q \sqrt{2\pi}} \exp\left(-\frac{(kT_a)^2}{2\sigma_q^2}\right), \quad k \in \mathbb{N} \quad (1.86)$$

und eine Fensterung mit einem Rechteckfenster der Länge $M = 2N_q + 1$ und Normierung $\sum_{m=-N_q}^{N_q} q[m] = 1$ ergibt

$$q[k] = \frac{q_G[k]}{\sum_{m=-N_q}^{N_q} q_G[m]}, \quad k \in \mathbb{N}. \quad (1.87)$$

Für $M_q = 2N_q + 1$ Datenpunkte $y[k+m]$, $m = -N_q, \dots, N_q$ erhält man das gefilterte Signal

$$\hat{y}[k] = q[k] * y[k] = \sum_{m=-N_q}^{N_q} q[k] y[k-m]. \quad (1.88)$$

Bemerkung 1.2. Durch die Verwendung eines Gauß-Filters erhält man eine Q -Filtermatrix, die eine Bandstruktur aufweist. Es stellt sich allerdings die Frage, wie die Q -Filtermatrix an den Rändern aufzubauen ist. Dazu ein Beispiel:

Beispiel 1.1. Einfaches Einsetzen der Filterkoeffizienten (1.87) in (1.48) liefert für $N = 5$ und $N_q = 1$ die Q -Filtermatrix

$$\mathbf{Q} = \begin{bmatrix} q[0] & q[-1] & 0 & 0 & 0 \\ q[1] & q[0] & q[-1] & 0 & 0 \\ 0 & q[1] & q[0] & q[-1] & 0 \\ 0 & 0 & q[1] & q[0] & q[-1] \\ 0 & 0 & 0 & q[1] & q[0] \end{bmatrix} \in \mathbb{R}^{5 \times 5}. \quad (1.89)$$

Man erkennt, dass die Filterverstärkung in der zweiten, dritten und vierten Zeile eins ist, d. h. $q[1] + q[0] + q[-1] = 1$. In den Randzeilen ist sie ungleich eins, d. h. $q[0] + q[-1] \neq 1$ und $q[1] + q[0] \neq 1$. Um dies zu vermeiden, kann man die Randzeilen jeweils getrennt normieren. Das Filter wird dann allerdings für die Randpunkte nicht mehr die gewünschte 3-dB Bandbreite f_c einhalten.

Beispiel 1.2 (Gauß-Filter). Abbildung 1.2 zeigt die Impulsantwortfolge $q[k]$ und den diskreten Frequenzgang $G(\exp(j\omega T_a))$ eines Gauß-Filters für $T_a = 1$ ms, $f_c = 20, 100, 200$ Hz und $N_q = 10$. Die Abbildung zeigt deutlich, dass für hohe Frequenzen f_c die Impulsantwortfolge $q[k]$ sehr spitz wird und damit im Grenzfall $f_c = 1/(2T_a)$ keine Filterung mehr stattfindet. Für niedrige Frequenzen ($f_c \lesssim 1/(40T_a)$) wird die Impulsantwortfolge $q[k]$ immer breiter. In diesem Grenzfall gehen alle Datenpunkte gleich in die Mittlung ein. Außerdem verringert sich die Abschneidelänge, was die Ausbildung einer Welligkeit im Sperrband zur Folge hat. Grundsätzlich muss damit beim Entwurf darauf geachtet werden, dass die Impulsantwortfolge nicht zu spitz und nicht zu flach gewählt wird.

Savitzky-Golay-Filter

Das Savitzky-Golay-Filter [1.11, 1.12] ist ebenso wie das Gauß-Filter ein Glättungsfilter. Im Wesentlichen wird dazu eine polynomiale Regression über eine Folge von Datenpunkten durchgeführt. Zusätzlich ist es im Rahmen des Entwurfs des Savitzky-Golay-Filters möglich ein Ableitungsfilter zu erstellen. So ein Ableitungsfilter kann zur numerischen Approximation der Zeitableitung des Ausgangsfehlers verwendet werden. Im Gegensatz zum Entwurf eines Gauß-Filters ist die 3-dB Bandbreite eines Savitzky-Golay-Filters nicht direkt vorgebar und im Unterschied zum Gauß-Filter erhält das Savitzky-Golay-Filter hochfrequente Signalanteile des Nutzsignals auf Kosten einer schlechteren Rauschunterdrückung.

Es werden dazu $M_q = 2N_q + 1$ Datenpunkte $y[k + m]$, $m = -N_q, \dots, N_q$ betrachtet für welche eine polynomiale Regression anhand eines Polynoms d -ten Grades der Form

$$p_m[k] = \sum_{i=0}^d c_i[k] (mT_a)^i, \quad (1.90)$$

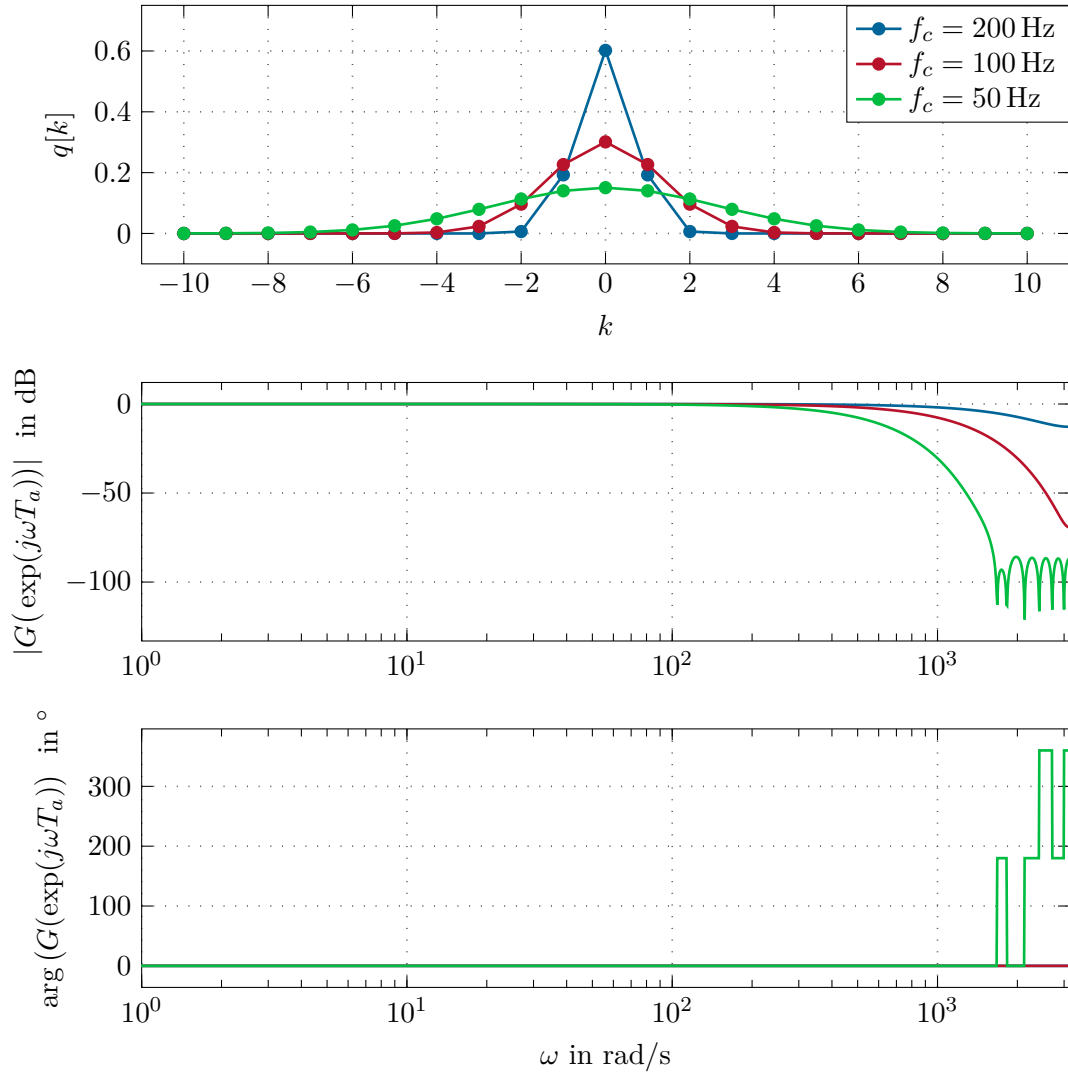


Abbildung 1.2: Impulsantwortfolge $q[k]$ und diskreter Frequenzgang $G(\exp(j\omega T_a))$ eines Gauß-Filters für $T_a = 1$ ms und $N_q = 10$.

mit den Koeffizienten $c_i[k]$, $i = 0, 1, \dots, d$, ausgeführt wird. Die Koeffizienten $c_i[k]$ werden dementsprechend anhand der Minimierungsaufgabe

$$\min_{c_0[k], c_1[k], \dots, c_d[k]} \sum_{m=-N_q}^{N_q} (y[k+m] - p_m[k])^2 \quad (1.91)$$

berechnet. Um die Minimierungsaufgabe zu lösen, fasst man die Datenpunkte $y[k+m]$, $m = -N_q, \dots, N_q$ in einem Vektor

$$\mathbf{y}^T[k] = [y[k - N_q] \quad \dots \quad y[k - 1] \quad y[k] \quad y[k + 1] \quad \dots \quad y[k + N_q]] \quad (1.92)$$

zusammen. Außerdem wird der Parametervektor

$$\mathbf{c}^T[k] = [c_0[k] \quad c_1[k] \quad \dots \quad c_d[k]] \quad (1.93)$$

und ein Vektor

$$\hat{\mathbf{y}}^T[k] = [p_{-N_q}[k] \quad \dots \quad p_{-1}[k] \quad p_0[k] \quad p_1[k] \quad \dots \quad p_{N_q}[k]] \quad (1.94)$$

definiert. Die Minimierungsaufgabe (1.91) lautet damit

$$\min_{\mathbf{c}[k]} (\mathbf{y}[k] - \hat{\mathbf{y}}[k])^T (\mathbf{y}[k] - \hat{\mathbf{y}}[k]) . \quad (1.95)$$

Da (1.90) linear in den Koeffizienten $c_i[k]$, $i = 0, 1, \dots, d$ ist, kann die Beziehung zu

$$\hat{\mathbf{y}}[k] = \mathbf{S}\mathbf{c}[k] \quad (1.96)$$

angegeben werden. Dazu definiert man $d + 1$ Vektoren \mathbf{s}_i , $i = 0, 1, \dots, d$, welche die Einträge

$$\mathbf{s}_i(m + N_q + 1) = m^i, \quad m = -N_q, \dots, N_q \quad (1.97)$$

aufweisen und die Matrix

$$\mathbf{S} = [\mathbf{s}_0 \quad \mathbf{s}_1 \quad \dots \quad \mathbf{s}_d] . \quad (1.98)$$

Bekannterweise lautet die Lösung der Minimierungsaufgabe

$$\mathbf{c}[k] = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \mathbf{y}[k] \quad (1.99)$$

und damit folgt für (1.96)

$$\hat{\mathbf{y}}[k] = \mathbf{S}(\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \mathbf{y}[k] . \quad (1.100)$$

Definiert man die Matrizen

$$\mathbf{H}^T = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T = [\mathbf{h}_0 \quad \mathbf{h}_1 \quad \dots \quad \mathbf{h}_d]^T \quad (1.101a)$$

$$\mathbf{B} = \mathbf{S}\mathbf{H}^T = [\mathbf{b}_{-N_q} \quad \dots \quad \mathbf{b}_{-1} \quad \mathbf{b}_0 \quad \mathbf{b}_1 \quad \dots \quad \mathbf{b}_{N_q}] \quad (1.101b)$$

mit den Spaltenvektoren \mathbf{h}_i , $i = 0, 1, \dots, d$ und \mathbf{b}_m , $m = -N_q, \dots, N_q$, dann gilt für die Koeffizienten

$$\mathbf{c}[k] = \mathbf{H}^T \mathbf{y}[k] \iff c_i[k] = \mathbf{h}_i^T \mathbf{y}[k], \quad i = 0, 1, \dots, d \quad (1.102)$$

und für das gefilterte Signal

$$\hat{\mathbf{y}}[k] = \mathbf{B}\mathbf{y}[k] \iff p_m[k] = \mathbf{b}_m^T \mathbf{y}[k], \quad m = -N_q, \dots, N_q . \quad (1.103)$$

D. h., das gefilterte Signal am Mittelpunkt $m = 0$ ist durch

$$\hat{y}[k] = p_0[k] = \mathbf{b}_0^T \mathbf{y}[k] \quad (1.104)$$

bzw.

$$\hat{y}[k] = \sum_{m=-N_q}^{N_q} \mathbf{b}_0(m + N_q + 1) y[k + m] = \sum_{m=-N_q}^{N_q} \mathbf{b}_0(N_q - m + 1) y[k - m] \quad (1.105)$$

gegeben, wobei $\mathbf{b}_0(m + N_q + 1)$ der $m + N_q + 1$ -te Eintrag von \mathbf{b}_0 ist. Anhand der $d + 1$ Spalten der $M_q \times (d + 1)$ -dimensionalen Matrix \mathbf{H} kann ein Savitzky-Golay-Ableitungsfiler für die $i = 1, 2, \dots, d$ -te Ableitung des Signals $\hat{y}[k]$ aufgebaut werden. Differenziert man (1.90) i -mal und setzt $m = 0$, erhält man

$$\left. \frac{d^i}{dm^i} \hat{y}[k] = \frac{d^i}{dm^i} p_m[k] \right|_{mT_a=0} = i! c_i[k] = i! \mathbf{h}_i^T \mathbf{y}[k] \quad (1.106)$$

bzw.

$$\frac{d^i}{dm^i} \hat{y}[k] = i! \sum_{m=-N_q}^{N_q} \mathbf{h}_i(m + N_q + 1) y[k + m], \quad (1.107)$$

wobei $\mathbf{h}_i(m + N_q + 1)$ der $m + N_q + 1$ -te Eintrag von \mathbf{h}_i ist.

Bemerkung 1.3. Die Behandlung der Randpunkte beim Aufbau einer Q -Filtermatrix kann auf Basis eines Savitzky-Golay-Filters einfacher als auf Basis eines Gauß-Filter erfolgen, da für jede Zeile mit der gleichen Filterlänge M_q gearbeitet werden kann und lediglich für die Randzeilen der Filterpunkt m verschoben werden muss. Gleiches gilt für den Aufbau einer Ableitungsfilermatrix anhand eines Savitzky-Golay-Ableitungsfilters.

Beispiel 1.3 (Savitzky-Golay-Filter). Für $d = 2$ lautet das Polynom (1.90)

$$p_m[k] = c_0[k] + c_1[k]m + c_2[k]m^2, \quad (1.108)$$

und für $N_q = 2$, d. h. $y[k + m]$, $m = -2, \dots, 2$ Datenpunkte, lautet der Vektor (1.92)

$$\mathbf{y}^T[k] = [y[k - 2] \quad y[k - 1] \quad y[k] \quad y[k + 1] \quad y[k + 2]]. \quad (1.109)$$

Der Parametervektor (1.93) schreibt sich folglich als

$$\mathbf{c}^T[k] = [c_0[k] \quad c_1[k] \quad c_2[k]] \quad (1.110)$$

und der Vektor (1.94) ist gegeben durch

$$\hat{\mathbf{y}}^T[k] = [p_{-2}[k] \quad p_{-1}[k] \quad p_0[k] \quad p_1[k] \quad p_2[k]]. \quad (1.111)$$

Die $d + 1$ Vektoren \mathbf{s}_i , $i = 0, 1, \dots, d$ lauten

$$\mathbf{s}_0^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (1.112a)$$

$$\mathbf{s}_1^T = \begin{bmatrix} -2 & -1 & 0 & 1 & 2 \end{bmatrix} \quad (1.112b)$$

$$\mathbf{s}_2^T = \begin{bmatrix} 4 & 1 & 0 & 1 & 4 \end{bmatrix} \quad (1.112c)$$

und können in der Matrix

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_0 & \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix} \quad (1.113)$$

angeordnet werden. Die Matrizen (1.101) ergeben sich zu

$$\mathbf{H}^T = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T = \begin{bmatrix} \mathbf{h}_0 & \mathbf{h}_1 & \mathbf{h}_2 \end{bmatrix}^T = \frac{1}{35} \begin{bmatrix} -3 & -7 & 5 \\ 12 & -3.5 & -2.5 \\ 17 & 0 & -5 \\ 12 & 3.5 & -2.5 \\ -3 & 7 & 5 \end{bmatrix}^T \quad (1.114a)$$

$$\mathbf{B} = \mathbf{S} \mathbf{H}^T = \begin{bmatrix} \mathbf{b}_{-2} & \mathbf{b}_{-1} & \mathbf{b}_0 & \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix} = \frac{1}{35} \begin{bmatrix} 31 & 9 & -3 & -5 & 3 \\ 9 & 13 & 12 & 6 & -5 \\ -3 & 12 & 17 & 12 & -3 \\ -5 & 6 & 12 & 13 & 9 \\ 3 & -5 & -3 & 9 & 31 \end{bmatrix} . \quad (1.114b)$$

D. h., das gefilterte Signal am Mittelpunkt $m = 0$ folgt der Rechenvorschrift

$$\begin{aligned} \hat{y}[k] &= p_0[k] = \mathbf{b}_0^T \mathbf{y}[k] \\ &= \frac{1}{35} (-3y[k-2] + 12y[k-1] + 17y[k] + 12y[k+1] - 3y[k+2]) \end{aligned} \quad (1.115)$$

und das Savitzky-Golay-Ableitungsfiler für die erste und zweite Ableitung am Mittelpunkt $m = 0$ lautet

$$\begin{aligned} \frac{d}{dm} \hat{y}[k] &= \frac{d}{dm} (c_0[k] + c_1[k]mT_a + c_2[k](mT_a)^2) \Big|_{m=0} = c_1[k] = \mathbf{h}_1^T \mathbf{y}[k] \\ &= \frac{1}{35} (-7y[k-2] - 3.5y[k-1] + 3.5y[k+1] + 7y[k+2]) \end{aligned} \quad (1.116)$$

bzw.

$$\begin{aligned} \frac{d^2}{dm^2} \hat{y}[k] &= \frac{d^2}{dm^2} (c_0[k] + c_1[k]mT_a + c_2[k](mT_a)^2) \Big|_{m=0} = 2c_2[k] = \mathbf{h}_2^T \mathbf{y}[k] \\ &= \frac{2}{35} (5y[k-2] - 2.5y[k-1] - 5y[k] - 2.5y[k+1] + 5y[k+2]) . \end{aligned} \quad (1.117)$$

Abbildung 1.3 zeigt die Impulsantwortfolgen $q[k]$ und die diskreten Frequenzgänge $G(\exp(j\omega T_a))$ eines Savitzky-Golay-Filters (S-G-F), eines Savitzky-Golay-Ableitungsfilters erster Ordnung (1. S-G-D-F) und eines Savitzky-Golay-Ableitungsfilters zweiter Ordnung (2. S-G-D-F) für $T_a = 1$ ms, $d = 2$ und $N_q = 2$.

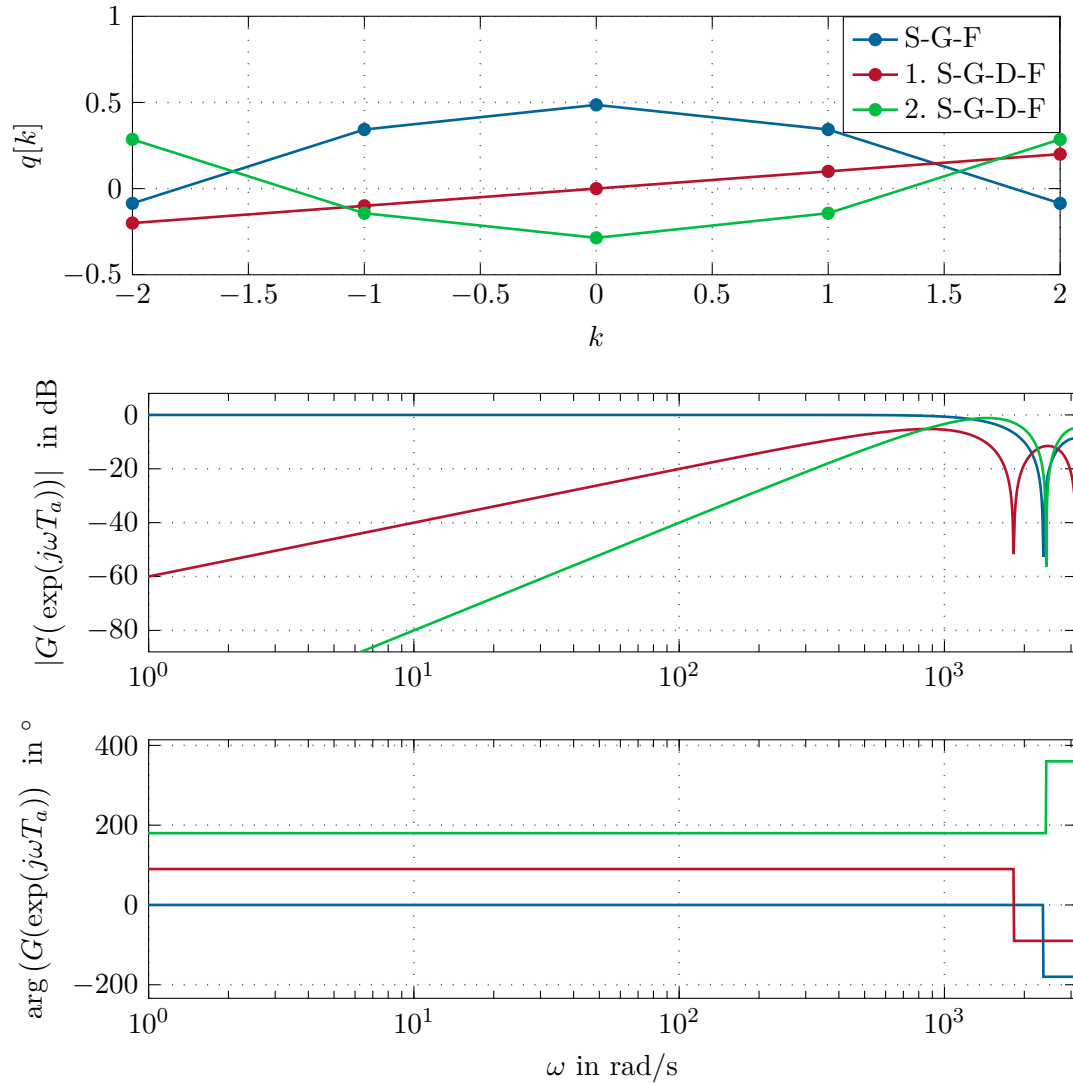


Abbildung 1.3: Impulsantwortfolge $q[k]$ und diskreter Frequenzgang $G(\exp(j\omega T_a))$ eines Savitzky-Golay-Filters (S-G-F), eines Savitzky-Golay-Differentiationsfilters erster Ordnung (1. S-G-D-F) und eines Savitzky-Golay-Differentiationsfilters zweiter Ordnung (2. S-G-D-F) für $T_a = 1$ ms, $d = 2$ und $N_q = 2$.

1.5.2 Inversionsbasierte ILR

Bei der inversionsbasierten ILR [1.13] wird die Verstärkungsmatrix zu $\mathbf{L} = \mathbf{G}^{-1}$ gewählt, da dann

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{u}_j + \mathbf{G}^{-1}\mathbf{e}_j) = \mathbf{Q}(\mathbf{u}_j + \mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{G}\mathbf{u}_j - \mathbf{y}_0)) = \mathbf{Q}(\mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{y}_0)) \quad (1.118)$$

gilt. Mit dieser Wahl ist Dead-Beat-Verhalten erzielbar. D. h., der Ausgangsfehler wird in einer Iteration zu \mathbf{e}_∞ :

$$\mathbf{e}_{j+1} = \mathbf{y}_d - \mathbf{G}\mathbf{Q}(\mathbf{G}^{-1}(\mathbf{y}_d - \mathbf{y}_0)) - \mathbf{y}_0 = \mathbf{e}_\infty. \quad (1.119)$$

Für $\mathbf{Q} = \mathbf{E}$ gilt zudem $\mathbf{e}_\infty = \mathbf{0}$. Selbstverständlich ist dieses Ergebnis von theoretischer Natur. Zum einen ist in einer realen Anwendung \mathbf{G} nie exakt bekannt und die Verwendung eines Q -Filters unabdingbar. Zum anderen kann gezeigt werden, dass die Matrixinversion von \mathbf{G} für nicht-minimalphasige Systeme schlecht konditioniert ist. Einen weiteren Zugang zu dieser Wahl verschafft ein Blick auf die Lifted-System Darstellung (1.37). Für $\mathbf{y}_0 = \mathbf{0}$, $\mathbf{v}_j = \mathbf{0}$, $\mathbf{y}_j = \mathbf{y}_d$ und $\mathbf{u}_j = \mathbf{u}_d$ folgt für die Steuerung

$$\mathbf{u}_d = \mathbf{G}^{-1}\mathbf{y}_d. \quad (1.120)$$

Gleichung (1.120) kann als Parametrierung des Eingangs \mathbf{u}_d durch den Ausgang \mathbf{y}_d verstanden werden. Ein Konzept, welches aus der Theorie der differentiellen Flachheit bekannt ist. Aus dieser Theorie ist bekannt, dass ein kontinuierliches, lineares System bezüglich eines Ausgangs \mathbf{z} differentiell flach ist, wenn der relative Grad r (die Anzahl von Zeitableitungen des Ausgangs bis zum ersten Mal explizit der Eingang auftritt) der Systemordnung n entspricht. Ist $r < n$, dann besitzt (1.23) Nullstellen. Ist (1.23) stabil und befindet sich eine dieser Nullstellen in der rechten offenen Halbebene, dann ist das System nicht-minimalphasig. Durch die Inversion werden dann die Nullstellen zu Polstellen und damit die einzelnen Einträge der Verstärkungsmatrix $\mathbf{L} = \mathbf{G}^{-1}$ sehr groß.

Beispiel 1.4. Die Übertragungsfunktion $G(z) = (z - a)/z$ besitzt eine Polstelle bei $z = 0$ und eine Nullstelle bei $z = a$. Für $|a| > 1$ ist das System nicht-minimalphasig. Die zugehörige Lifted-System Darstellung bzw. deren Inverse lautet [1.14]

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ -a & 1 & 0 & 0 & \cdots \\ 0 & -a & 1 & 0 & \cdots \\ 0 & 0 & -a & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \quad \mathbf{G}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ a & 1 & 0 & 0 & \cdots \\ a^2 & a & 1 & 0 & \cdots \\ a^3 & a^2 & a & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (1.121)$$

Offensichtlich wachsen die Matrixeinträge in \mathbf{G}^{-1} und damit $\mathbf{u}_d = \mathbf{G}^{-1}\mathbf{y}_d$ für $a \gg 1$ sehr stark an. Dies wird auch deutlich, wenn man sich vor Augen führt, dass die erste Spalte von \mathbf{G}^{-1} der Impulsantwortfolge entspricht.

Dieser Umstand kann umgangen werden, indem (1.23) anhand einer regulären Zustandstransformation $\mathbf{z} = \mathbf{T}\mathbf{y}$ auf Regelungsnormalform transformiert wird und für den neuen, flachen Ausgang \mathbf{z} eine Solltrajektorie \mathbf{z}_d geplant wird. In Regelungsnormalform besitzt das System vollen relativen Grad ($n = r$) und damit weist (1.23) keine instabilen Nullstellen auf.

1.5.3 Normoptimale ILR

Bei der Normoptimalen ILR [1.7, 1.15] wird die Q -Filtermatrix bzw. Verstärkungsmatrix nicht direkt entworfen. Vielmehr werden die positiv definiten Matrizen $\mathbf{V} > 0$, $\mathbf{S} > 0$ und $\mathbf{R} > 0$ verwendet, um auf Basis der Minimierungsaufgabe (1.40)

$$\begin{aligned} \min_{\mathbf{u}_{j+1}} J(\mathbf{u}_{j+1}) &= \frac{1}{2} \mathbf{e}_{j+1}^T \mathbf{V} \mathbf{e}_{j+1} + \frac{1}{2} \mathbf{u}_{j+1}^T \mathbf{S} \mathbf{u}_{j+1} + \frac{1}{2} (\mathbf{u}_{j+1} - \mathbf{u}_j)^T \mathbf{R} (\mathbf{u}_{j+1} - \mathbf{u}_j) \\ \text{u.B.v.} \quad \mathbf{e}_{j+1} &= \mathbf{e}_j + \mathbf{G} \mathbf{u}_j - \mathbf{G} \mathbf{u}_{j+1} \end{aligned} \quad (1.122)$$

die Q -Filtermatrix bzw. Verstärkungsmatrix in der Form

$$\mathbf{Q} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R})^{-1} (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R}) \quad (1.123a)$$

$$\mathbf{L} = (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{V} \quad (1.123b)$$

zu entwerfen. Laut Satz 1.4 ist die Fehlerdynamik (1.59) des ILR Gesetzes (1.47), angewendet auf die Lifted-System Darstellung (1.37), asymptotisch stabil, falls

$$\rho(\mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G})) < 1. \quad (1.124)$$

Einsetzen von (1.123) führt auf

$$\begin{aligned} \mathbf{Q}(\mathbf{E} - \mathbf{L}\mathbf{G}) &= (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R})^{-1} (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R}) (\mathbf{E} - (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{V} \mathbf{G}) \\ &= (\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R})^{-1} \mathbf{R}. \end{aligned} \quad (1.125)$$

Da $\|(\mathbf{G}^T \mathbf{V} \mathbf{G} + \mathbf{S} + \mathbf{R})^{-1} \mathbf{R}\| < 1$ gilt, ist die monotone Konvergenz der Normoptimalen ILR immer gewährleistet. Selbstverständlich ist damit auch die Konvergenz gesichert, da für eine beliebige quadratische Matrix $\rho(\cdot) \leq \|\cdot\|$ gilt.

Es bleibt die Frage offen, wie die Einstellmatrizen $\mathbf{V} > 0$, $\mathbf{S} > 0$ und $\mathbf{R} > 0$ gewählt werden sollen. Eine Antwort darauf gibt ein Vergleich der Normoptimalen ILR mit der Linear Quadratischen Regelung (LQR). In der Regel wird die Anzahl der Entwurfsfreiheitsgrade eingeschränkt, indem diagonale Matrizen angesetzt werden, d. h. $\mathbf{V} = v\mathbf{E} > 0$, $\mathbf{S} = s\mathbf{E} > 0$ und $\mathbf{R} = r\mathbf{E} > 0$. Im Einzelnen heißt das:

- Für ein großes v wird der Ausgangsfehler \mathbf{e}_j stärker bestraft und die bleibende Regelabweichung \mathbf{e}_∞ wird kleiner.

- Für ein großes s wird die Stellgröße \mathbf{u}_j stärker bestraft und die bleibende Regelabweichung \mathbf{e}_∞ wird größer. Für $s \rightarrow 0$ gilt $\mathbf{Q} \rightarrow \mathbf{E}$ und damit $\mathbf{e}_\infty \rightarrow \mathbf{0}$.
- Für ein großes r wird die Änderung der Stellgröße $\mathbf{u}_{j+1} - \mathbf{u}_j$ stärker bestraft. Damit wird eine bessere Rauschunterdrückung erreicht. Allerdings auf Kosten der Konvergenzgeschwindigkeit. Für $r \rightarrow 0$ stellt sich Dead-Beat-Verhalten ein.

1.6 ILR mit Feedback

Da die ILR im Grunde eine Feedforward Regelung ist, lässt sie sich im Sinne einer Zwei-Freiheits-Grade-Regelung um einen Feedback Anteil erweitern. In der Literatur unterscheidet man grundsätzlich zwei Strukturen: Die *parallele* und die *serielle Struktur*. Abbildung (1.4) zeigt die parallele (grün) und serielle (orange) Struktur für eine Strecke $G(\delta)$ mit $m = 1$, eine Eingangsstörung $d[k]$ und für die Anfangsbedingung $\mathbf{x}_0 = \mathbf{0}$.

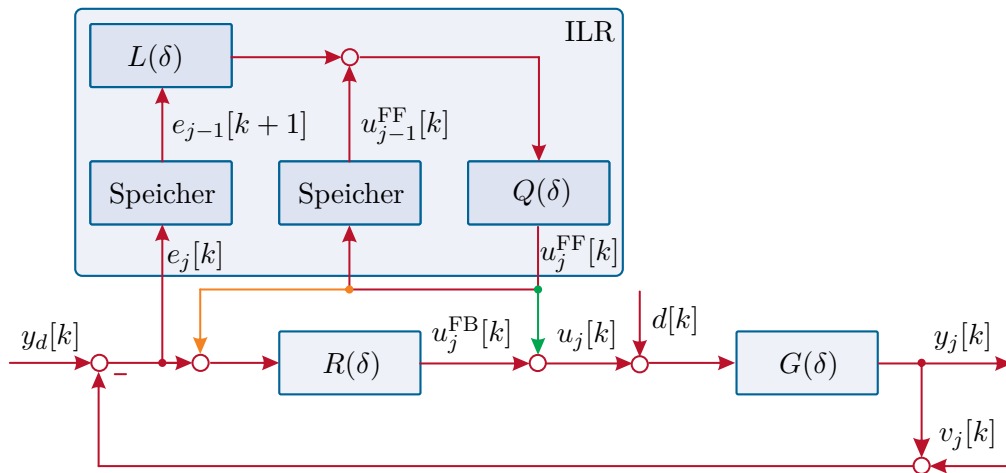


Abbildung 1.4: Parallele (grün) und serielle (orange) Struktur.

1.6.1 Parallele Struktur

Bei der parallelen Struktur wird die Ausgangsregelung

$$u_j^{\text{FB}}[k] = R(\delta)e_j[k] \quad (1.126)$$

und das ILR Gesetz

$$u_j^{\text{FF}}[k] = Q(\delta)(u_{j-1}^{\text{FF}}[k] + L(\delta)e_{j-1}[k+1]) \quad (1.127)$$

verwendet. Der Streckeneingang lautet dementsprechend

$$u_j[k] = u_j^{\text{FF}}[k] + u_j^{\text{FB}}[k] = u_j^{\text{FF}}[k] + R(\delta)e_j[k] . \quad (1.128)$$

Der Ausgang der parallelen Struktur ergibt sich somit zu

$$y_j[k] = \frac{G}{1+GR} \left(u_j^{\text{FF}}[k] + d[k] \right) + \frac{GR}{1+GR} (y_d[k] + v_j[k]) \quad (1.129)$$

und der Regelfehler der parallelen Struktur ist gegeben durch

$$\begin{aligned} e_j[k] &= y_d[k] - y_j[k] = y_d[k] - \frac{G}{1+GR} \left(u_j^{\text{FF}}[k] + d[k] \right) - \frac{GR}{1+GR} (y_d[k] + v_j[k]) \\ &= \frac{1}{1+GR} \left(y_d[k] - G u_j^{\text{FF}}[k] \right) - \frac{G}{1+GR} d[k] - \frac{GR}{1+GR} v_j[k] . \end{aligned} \quad (1.130)$$

Man erkennt, dass $u_j^{\text{FF}}[k]$ über die Störübertragungsfunktion $G/(1+GR)$ auf den Ausgang wirkt und für $u_j^{\text{FF}}[k] = G^{-1}y_d[k]$ der erste Term im Regelfehler (1.130) verschwindet.

Im Folgenden wird vereinfachend $d[k] = 0$ bzw. $v_j[k] = 0$ gefordert. Aus (1.130) ergibt sich einerseits

$$u_{j+1}^{\text{FF}}[k] = \frac{1}{G} y_d[k] - \frac{1}{G} (1+GR) e_{j+1}[k] \quad (1.131)$$

und andererseits

$$Q u_j^{\text{FF}}[k] = \frac{Q}{G} y_d[k] - \frac{Q}{G} (1+GR) e_j[k] . \quad (1.132)$$

Eingesetzt in (1.127) folgt

$$(1+GR) e_{j+1}[k] = Q(1+GR) e_j[k] - \delta G L e_j[k] + (1-Q) y_d[k] \quad (1.133)$$

bzw.

$$e_{j+1}[k] = \left(Q - \frac{\delta L G}{1+GR} \right) e_j[k] + (1-Q) y_d[k] . \quad (1.134)$$

Für $Q = 1$ und $L = 0$ ist $u_{j+1}[k] = u_j[k] = u^{\text{FF}}[k]$ konstant in jeder Iteration und damit der Ausgangsfehler $e_j[k]$ in jeder Iteration gleich, d. h. $e_{j+1}[k] = e_j[k]$.

1.6.2 Serielle Struktur

Für eine serielle Struktur lautet der Streckeneingang

$$u_j[k] = R(\delta) \left(e_j[k] + u_j^{\text{FF}}[k] \right) , \quad (1.135)$$

der Ausgang

$$y_j[k] = \frac{G}{1+GR} d[k] + \frac{GR}{1+GR} \left(y_d[k] + u_j^{\text{FF}}[k] + v_j[k] \right) \quad (1.136)$$

und der Regelfehler der seriellen Struktur ist gegeben durch

$$\begin{aligned} e_j[k] &= y_d[k] - y_j[k] = y_d[k] - \frac{G}{1+GR} d[k] - \frac{GR}{1+GR} \left(y_d[k] + u_j^{\text{FF}}[k] + v_j[k] \right) \\ &= \frac{1}{1+GR} \left(y_d[k] - G R u_j^{\text{FF}}[k] \right) - \frac{G}{1+GR} d[k] - \frac{GR}{1+GR} v_j[k] . \end{aligned} \quad (1.137)$$

Im Gegensatz zur parallelen Struktur wirkt bei der seriellen Struktur $u_j^{\text{FF}}[k]$ über die Führungsübertragungsfunktion $GR/(1 + GR)$ auf den Ausgang. Damit der erste Term im Regelfehler in (1.137) verschwindet, muss das ILR Gesetz auf den offenen Kreis $G_0 = GR$ entworfen werden, da dann $u_j^{\text{FF}}[k] = G_0^{-1}y_d[k]$ gilt. Die beiden Strukturen sind im Grenzfall identisch.

1.7 Literatur

- [1.1] Z. Bien und J.-X. Xu, *Iterative learning control - analysis, design, integration and applications*. London: Springer, 1998.
- [1.2] D. Owens und J. Hätönen, „Iterative learning control - an optimization paradigm“, *Annual Reviews in Control*, Bd. 29, Nr. 1, S. 57–70, 2005.
- [1.3] D. A. Bristow, M. Tharayil und A. Alleyne, „A survey of iterative learning control“, *Control Systems, IEEE*, Bd. 26, Nr. 3, S. 96–114, 2006.
- [1.4] H.-S. Ahn, Y.-Q. Chen und K. Moore, „Iterative learning control: brief survey and categorization“, *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, Bd. 37, Nr. 6, S. 1099–1121, 2007.
- [1.5] H.-S. Ahn, K. L. Moore und Y. Chen, *Iterative learning control - robustness and monotonic convergence for interval system*. London: Springer, 2007.
- [1.6] J. M. Ortega, „Stability of difference equations and convergence of iterative processes“, *SIAM Journal on Numerical Analysis*, Bd. 10, Nr. 2, S. 268–282, 1973.
- [1.7] J. H. Lee, K. S. Lee und W. C. Kim, „Model-based iterative learning control with a quadratic criterion for time-varying linear systems“, *Automatica*, Bd. 36, Nr. 5, S. 641–657, 2000.
- [1.8] S. Arimoto, S. Kawamura und F. Miyazaki, „Bettering operation of robots by learning“, *Journal of Robotic Systems*, Bd. 1, Nr. 2, S. 123–140, 1984.
- [1.9] R. W. Longman, „Iterative learning control and repetitive control for engineering practice“, *International Journal of Control*, Bd. 73, Nr. 10, S. 930–954, 2000.
- [1.10] D. A. Bristow, „Iterative learning control for precision motion of microscale and nanoscale tracking systems“, Diss., University of Illinois at Urban-Champaign, 2007.
- [1.11] A. Savitzky und M. Golay, „Smoothing and differentiation of data by simplified least-squares procedures“, *Anal. Chem.*, Bd. 36, Nr. 8, S. 1627–1639, 1964.
- [1.12] R. W. Schafer, „What is a savitzky-golay-filter“, *IEEE Signal Processing Letters*, S. 111–117, 2011.
- [1.13] T. J. Harte, J. Hätönen und D. H. Owens, „Discrete-time inverse model-based iterative learning control: stability, monotonicity and robustness“, *International Journal of Control*, Bd. 78, Nr. 8, S. 577–586, 2005.
- [1.14] S. Hashemi und J. Hammond, „The interpretation of singular values in the inversion of minimum and non-minimum phase systems“, *Mechanical Systems and Signal Processing*, Bd. 10, Nr. 3, S. 225–240, 1996, ISSN: 0888-3270.
- [1.15] K. Barton und A. Alleyne, „A norm optimal approach to time-varying ilc with application to a multi-axis robotic testbed“, *Control Systems Technology, IEEE Transactions on*, Bd. 19, Nr. 1, S. 166–180, Jan. 2011.