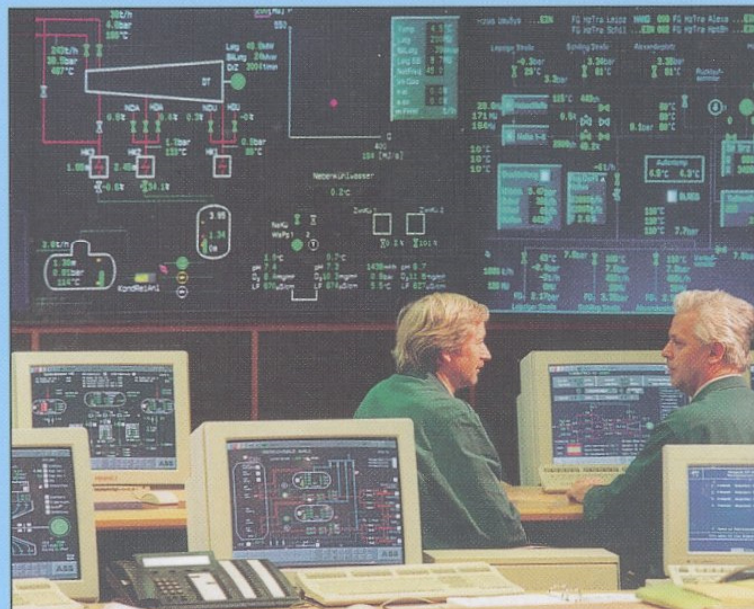


Serge Zacher

# Duale Regelungstechnik



Methoden und Werkzeuge  
der Informationstechnologie  
für die Regelungstechnik  
mit Prozessrechnern

## Zum Autor

Prof. Dr.-Ing. **Serge Zacher** unterrichtet seit 1992 Automatisierungstechnik an der Fachhochschule Wiesbaden. Während seiner gesamten Tätigkeit in der chemischen Industrie und im Hochschulbereich entstanden drei Patente, rund 150 Fachartikel und mehrere Lehrbücher. In der Forschung befasst er sich schwerpunktmäßig mit künstlichen neuronalen Netzen und Softwareagenten für die Steuerung, deren Ergebnisse er auf der CeBit '97 und den Hannover-Industrie-Messen '99 und '02 präsentierte.

Als Herausgeber fasste er die eigenen Beiträge und die von renommierten Autoren aus Industrie, Hochschulen und Universitäten in einem Buch „Automatisierungstechnik kompakt“ zusammen. Gemeinsam mit Prof. Dr.-Ing. M. Reuter bearbeitete er die 10. Auflage des Lehrbuches „Regelungstechnik für Ingenieure“, 2002. Für den VDE VERLAG verfasste er das Buch „SPS-Programmierung mit Funktionsbau-sprache“.

## Zum Buch

Die klassischen Methoden, wie Ziegler-Nichols-Verfahren oder Betragsoptimum, sind im Buch zusammen mit den wissensbasierten Verfahren, wie Fuzzy-Logik und künstliche neuronale Netze, einheitlich behandelt. Die daraus resultierenden dualen Regelkreise bestehen aus Grundgliedern, wie P-, P-T1-, I-Glieder, und aus

zeitvarianten Gliedern, wie Fuzzy-Sets und Backpropagation-Neuronen. Der Entwurf von Regelkreisen mit klassischen und wissensbasierten Elementen erfolgt mit Hilfe von Methoden der linearen Regelungstechnik für einfache Regelkreise und Mehrgrößenregler. Zu diesem Zweck enthält das Buch auch eine Formelsammlung der klassischen Zeit-, Frequenz- und Bildbereichs-Methoden im Anhang.

Die Implementierung erfolgt mit den zum Industrie-Standard gewordenen Software-Paketen MATLAB V.6.1 für Simulation, Concept V 2.2 für IEC-Programmierung und Lookout V.4.5 für Visualisierung.

## Interessentenkreis

Das Buch wendet sich an Ingenieure, die in der Praxis mit Regelungs- und Steuerungssystemen arbeiten. Es ist als Studienbegleitung für Studenten der Elektrotechnik, des Maschinenbaus und der angewandten Informatik gedacht. Beiden Zielgruppen ermöglicht das Buch ein Selbststudium, das durch die Webseite des Verfassers

<http://r5.mnd.fh-wiesbaden.de/prof/zacher>

begleitet und unterstützt wird. Deshalb können auch die Berufsschüler mit dem Buch arbeiten.

ISBN 3-8007-2701-3



9 783800 727018

ISBN 3-8007-2701-3

Das Konzept der Dualität, wie es im Buch behandelt wird, ist im Abschnitt 3.2 formuliert. Um vorher einen Blick in die dualen Systeme zu ermöglichen, ist eine kurze Übersicht der bekannten Dualitäten der Mathematik und der Technik in den Abschnitten 3.1.2 bis 3.1.5 gegeben. Diese Abschnitte kann man beim Lesen ohne Verlust der Vollständigkeit überspringen.

### 3.1.2 Symmetrie und Antisymmetrie als duale Operationen

Der Begriff *dual* bedeutet nach *Webster's New World Dictionary* „bestehend oder gebildet aus zwei Teilen“ (*having or composed of two parts*). In diesem Sinn kann man auch Symmetrieoperationen als duale Operationen betrachten.

Die einfachste Definition der Symmetrie wurde von *I. Schafranowski* gegeben:

„Eine Figur heißt symmetrisch, wenn sie aus gleichen, regelmäßig wiederholbaren Teilen besteht.“

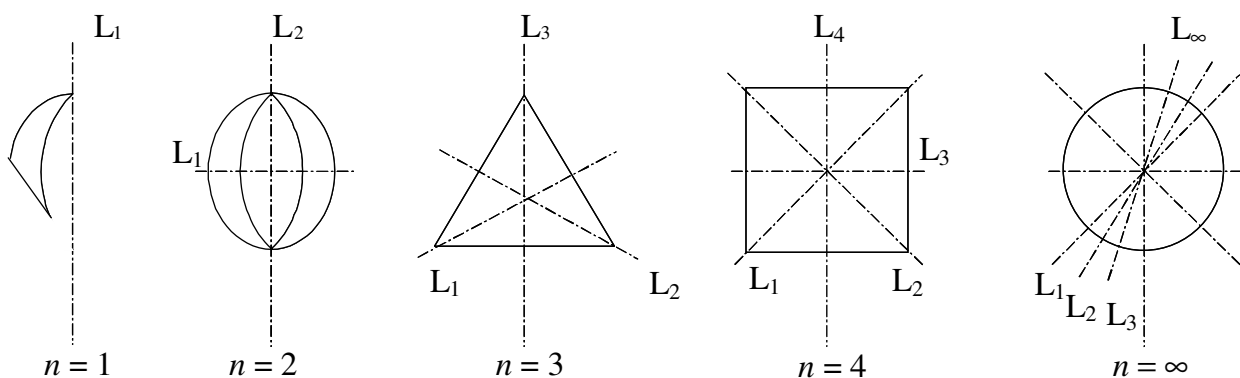
Im Allgemeinen wird Symmetrie wie folgt definiert:

„Ein Objekt ist symmetrisch, wenn man solche Transformationen des Objekts finden kann, die das transformierte Objekt und das ursprüngliche Objekt bezüglich bestimmten Merkmalen identisch lassen.“

Die Symmetrie wird durch Symmetrieelemente, -operationen und -gruppen beschrieben.

Die *Symmetrieelemente* sind die *Symmetrieebene* und die *Symmetrieachse*:

- Symmetrieebene  $m$  ist eine angenommene Ebene, die ein symmetrisches Objekt in spiegelgleiche Teile zerlegt. Die Anzahl der Symmetrieebenen wird als Symmetrieordnung  $p$  bezeichnet. Für eine flache Figur ist  $p = 1$ .
- Symmetrieachse  $L$  ist eine angenommene Linie. Beim Umdrehen eines symmetrischen Objekts rund um die Symmetrieachse um  $360^\circ$  soll das Objekt wenigstens einmal auf sich selbst übereinander liegen (**Bild 3.2**).



**Bild 3.2** Symmetrieelemente einer Ebene ( $p = 1$ )

Wird ein Objekt bei voller Umdrehung  $n$ -mal aufeinander gelegt, so hat die Symmetrieachse die Ordnung  $n$ . Für jede unregelmäßige flache Figur ist  $n = 1$ , für ein Quadrat gilt  $n = 2$ , für einen Kreis ist  $n = \infty$ .

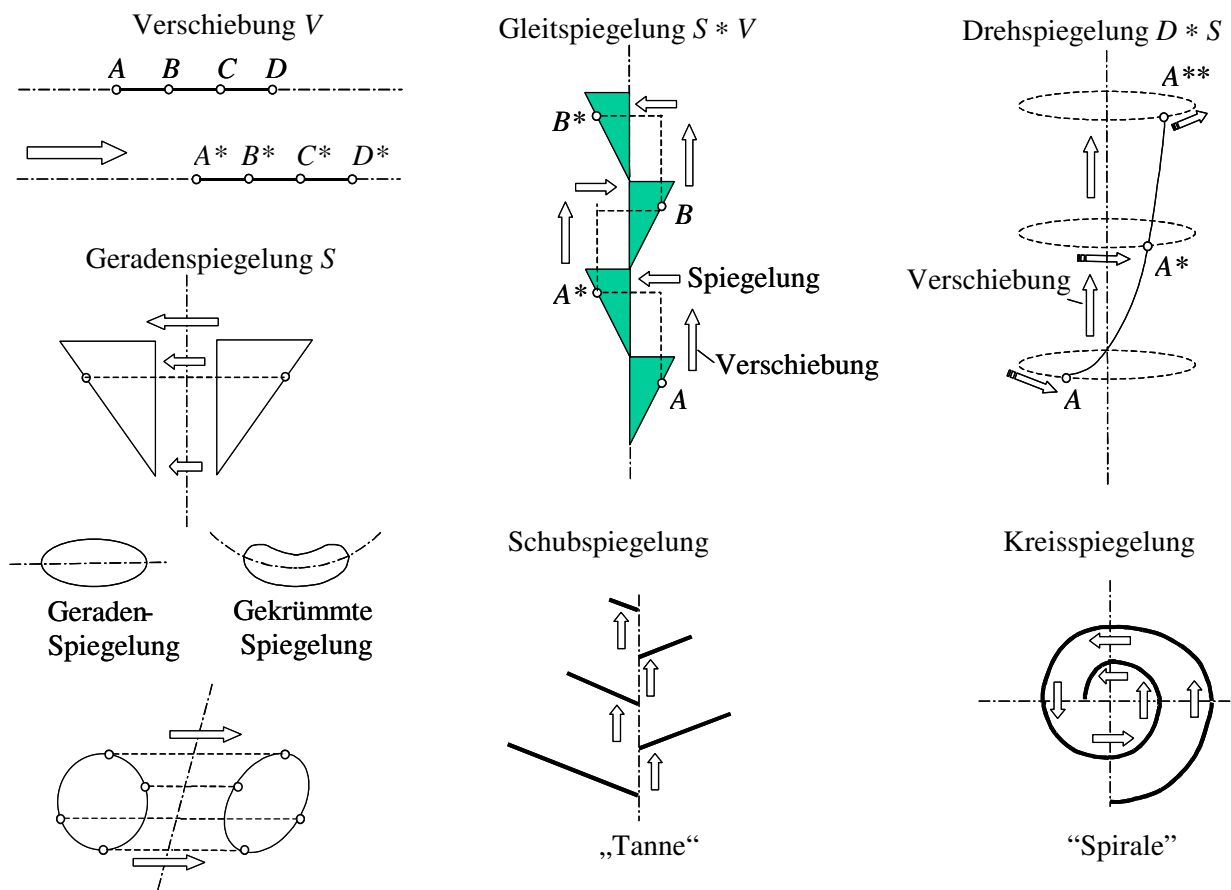
Zu den *Symmetrieeoperationen* gehören folgende Transformationen:

- Geradenspiegelungen  $S$
- Verschiebungen (Translationen)  $V = S1 * S2$
- Drehungen  $D$
- Drehspiegelungen  $S_D = D * S$
- Gleitspiegelungen (Schubspiegelungen)  $S_V = S * V$
- Identitätstransformationen  $S * S$  oder  $D * D$  usw.

Beispiele klassischer Symmetrieeoperationen sind in **Bild 3.3** gezeigt. Im Fall einer gekrümmten Symmetrieebene oder -achse entstehen Symmetrieeoperationen, die zur *Ähnlichkeitssymmetrie* oder *Homologie* führen.

Die mathematische Definition der Symmetrie mit Hilfe von Mengen lautet:

„Eine Teilmenge  $A$  ist symmetrisch, wenn eine andere Teilmenge  $A^{-1}$  existiert, so dass  $A^{-1} = A$  gilt.“

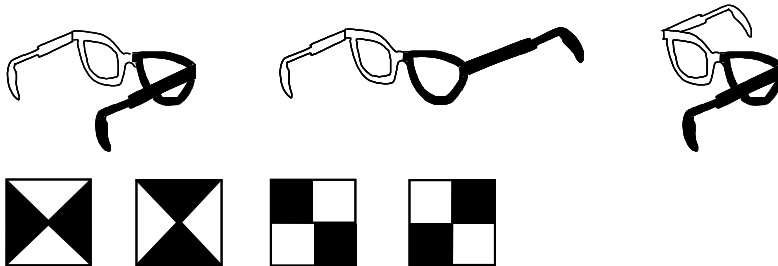


**Bild 3.3** Symmetrieeoperationen

Die allgemeine Definition der Antisymmetrie lautet:

„Die Antisymmetrie besteht aus gleichen Gruppenoperationen, wie die Symmetrie, jedoch mit entgegengesetzten Variablen.“

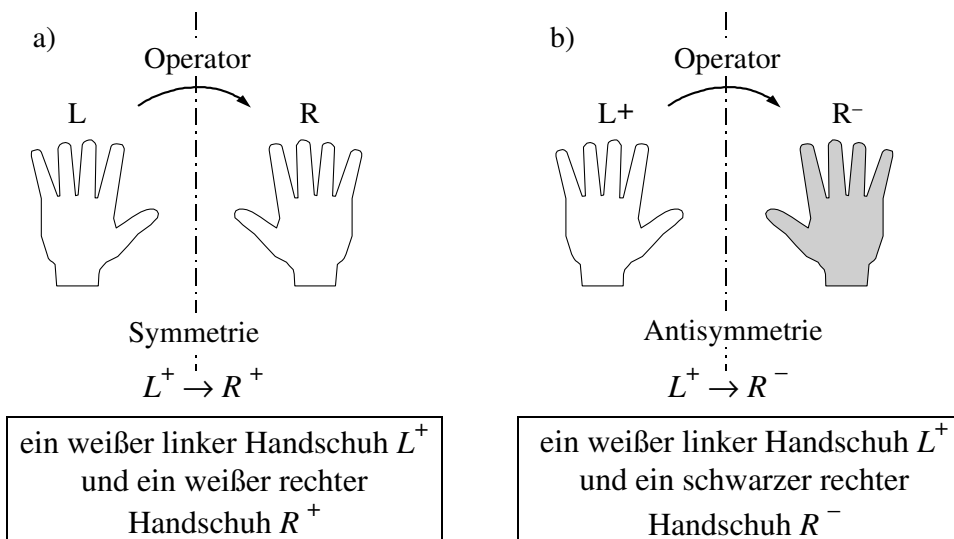
Der Begriff der *Antisymmetrie* wurde 1927 von *Speiser* eingeführt und weiter von *A. Schubnikov* (1945) entwickelt. Die Antisymmetrie als duale Operation der Symmetrie ist von *J. Sivardier* in [20] gründlich behandelt. Einige einfache Beispiele davon sind in **Bild 3.4** demonstriert.



**Bild 3.4** Beispiele der Antisymmetrie nach *J. Sivardier* [20]

Ein anderes Beispiel der Antisymmetrie ist in **Bild 3.5** nach [27] gezeigt. Der Handschuh für die linke Hand ist weiß und durch  $L^+$  bezeichnet. Der Handschuh  $R$  für die rechte Hand entsteht mit Hilfe einer Geradenspiegelung  $S$  und wird im symmetrischen Fall auch weiß. Im Fall der Antisymmetrie ist der Operator der Geradenspiegelung negativ, so wird die Farbe des Handschuhs geändert.

In [27] wurde die Antisymmetrie auf mehrstufige Prozesse mit mehreren Ein- und Ausgangsgrößen verallgemeinert. Unter einem Prozess versteht man dabei ein physikalisches oder informationstechnisches offenes System, das aus einer Reihenschaltung von  $N$  Elementen besteht, deren Ein- und Ausgänge miteinander



**Bild 3.5** Antisymmetrie als Symmetrieoperation mit entgegengesetztem Vorzeichen

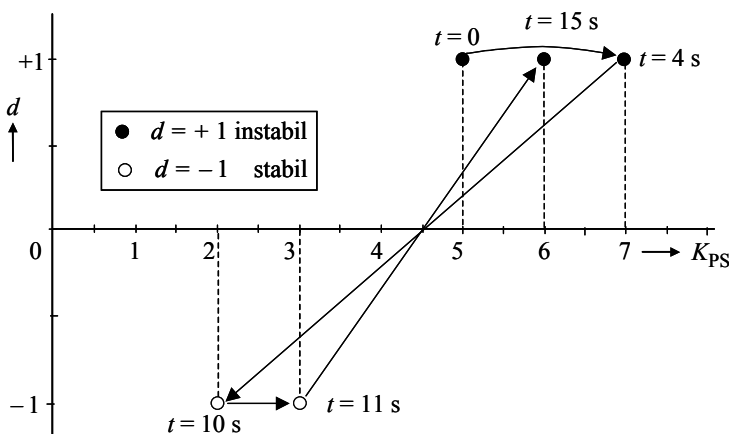
### 3.3.1 Hybride Behandlung

Die hybride Behandlung eines dualen Regelkreises wird am Beispiel einer Regelstrecke mit nichtkonstanten, zeitabhängigen Parametern gezeigt. Es wird angenommen, dass keine mathematische Beschreibung der Strecke vorliegt, jedoch man mit der Strecke experimentieren kann. Die hybride Behandlung wird anhand des MATLAB/Simulink-Modells erklärt. So wird der Regelkreis z. B. mit einem I-Regler gebildet und experimentell untersucht. Am Eingang des Kreises werden die Führungssprünge der Standardhöhe  $w_0 = 1$  gegeben und die Sprungantworten aufgenommen. **Bild 3.25** zeigt, dass der Kreis, bedingt durch die Parameteränderung der Strecke, zu verschiedenen Zeitpunkten unterschiedliche dynamische Eigenschaften aufweist.

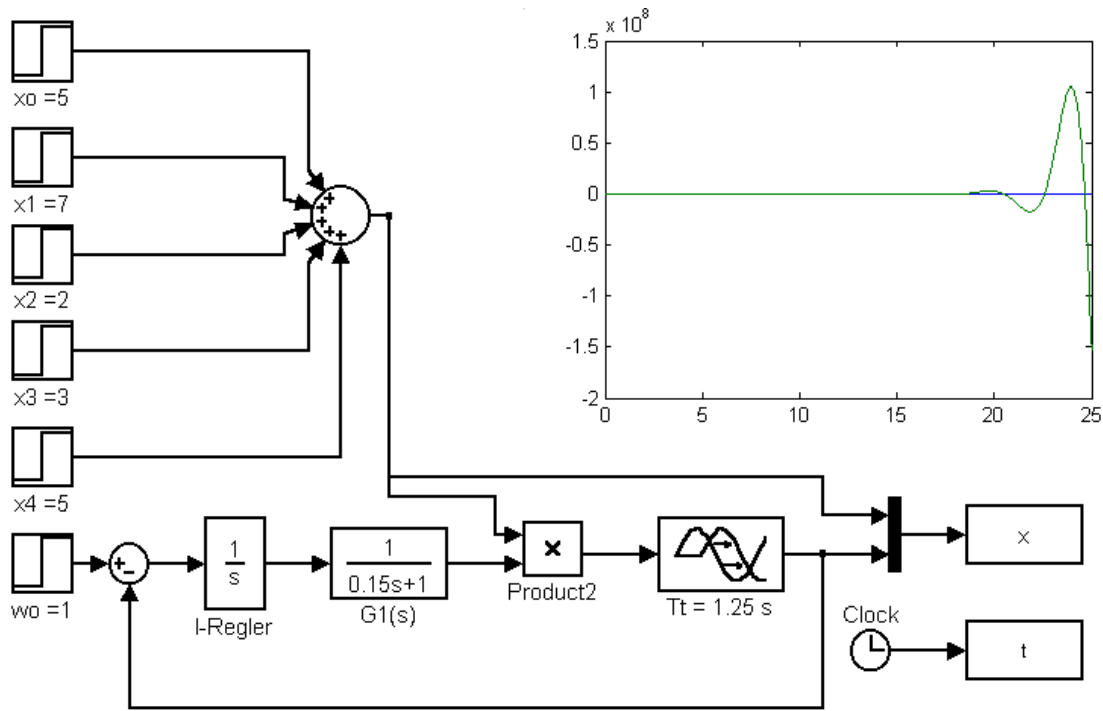
Statt die Strecke zu identifizieren, wie es in der klassischen Regelungstechnik der Fall wäre, wird ein KNN eingesetzt, dessen Aufgabe ist es, die instabilen Zustände zu erkennen und den Kreis durch die Anpassung von eigenen Parametern zu stabilisieren. Dafür wird das im Abschnitt 3.2.3 beschriebene KNN mit Mitkopplung eingesetzt. Um das Beispiel anschaulich zu machen, ist das KNN mit nur einem Neuron aufgebaut. Der Einfachheit halber ist auch eine P- $T_1$ -Strecke mit der Totzeit  $T_t$

$$G_S(s) = \frac{K_{PS}}{1 + sT_1} e^{-sT_t}$$

mit Parametern  $K_{PS} = 1$ ,  $T_1 = 0,15$  s und  $T_t = 1,25$  s gewählt, was dem KNN während des Experiments selbstverständlich nicht bekannt werden soll. Die Parameteränderung ist auch einfach mit Hilfe von fünf Step-Generatoren simuliert (**Bild 3.26**), die dann in einem Subsystem als ein Eingabe-Modul maskiert werden.

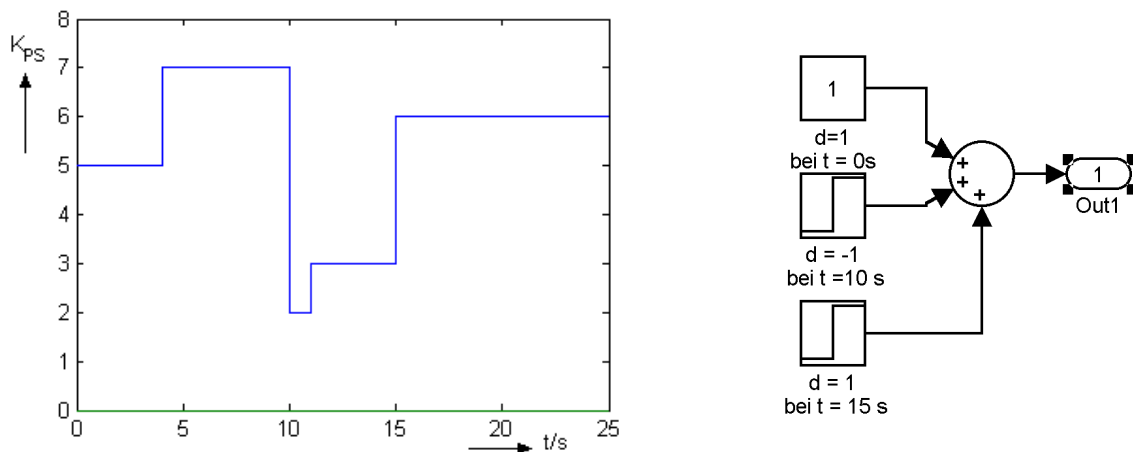


**Bild 3.25** Zustandsgraph des Regelkreises mit stabilem und instabilem Verhalten. Die im Diagramm gezeigten Werte der Streckenparameter sind dem KNN nicht vorgegeben.



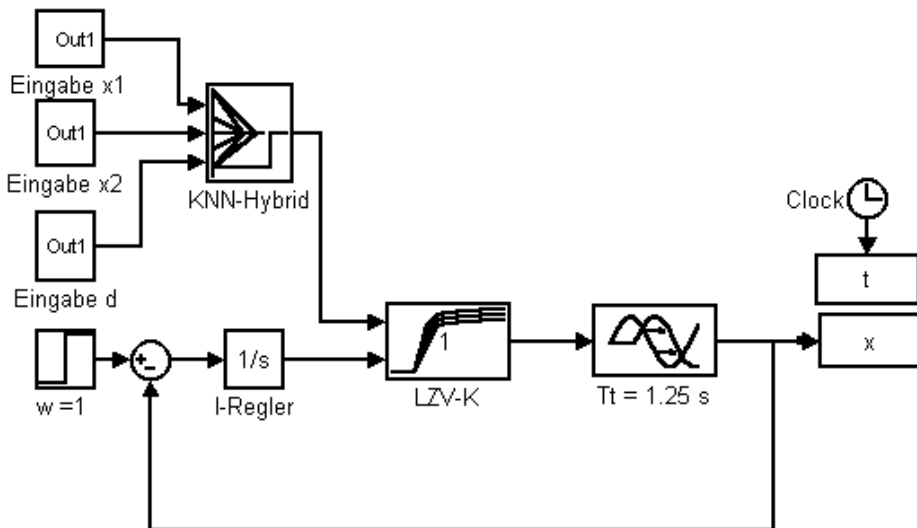
**Bild 3.26** MATLAB/Simulink-Modell des Regelkreises. Die Änderung des Parameters  $K_{PS}$  der Strecke ist mittels fünf Step-Signal-Generatoren simuliert. Die Sprungantwort weist auf ein instabiles Verhalten hin.

Die simulierte Sprungantwort zeigt, dass der Kreis mit dem I-Regler, der einen konstanten Integrierbeiwert  $K_{IR} = 1 \text{ s}^{-1}$  hat, instabil ist. Die entsprechende Änderung der Streckenparameter ist in **Bild 3.27** gezeigt.



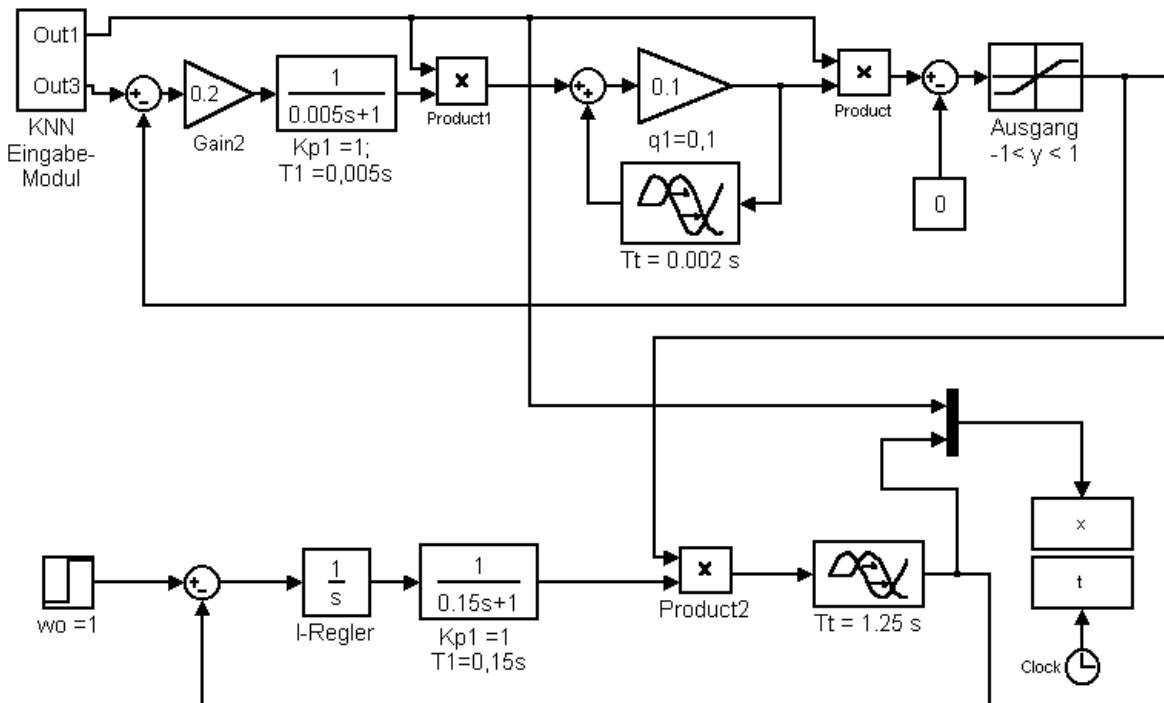
**Bild 3.27** Zeitliche Änderung des Streckenparameters  $K_{PS}$  und die Simulink-Module, die die Eingabe der entsprechenden KNN-Sollwerte simulieren. Die  $d$ -Werte sollen experimentell aus dem Regelkreis ermittelt werden.





**Bild 3.28** Allgemeiner Aufbau des hybriden Systems

Der Regelkreis mit dem KNN ist im Wirkungsplan des **Bildes 3.28** gezeigt. Die detaillierte Darstellung von Simulink-Subsystemen *KNN-Hybrid* und *LZV-K* findet man in **Bild 3.29**. Die Strecke besteht aus dem zeitinvarianten Teil (Totzeitglied mit  $T_t = 1,25$  s) und dem linearen zeitvarianten Teil *LZV* mit konstanter Zeitkonstante  $T_1 = 0,15$  s und variierbarem Proportionalbeiwert  $K_{PS}$ .

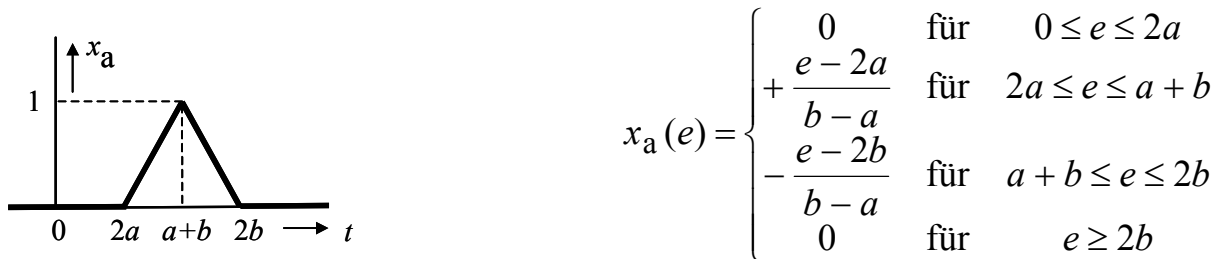


**Bild 3.29** Hybrides KNN und Kommunikation mit dem Regelkreis. Die Lernschrittweite  $\eta$  beträgt 0,2. Der Schwellenwert  $\theta$  ist gleich null.

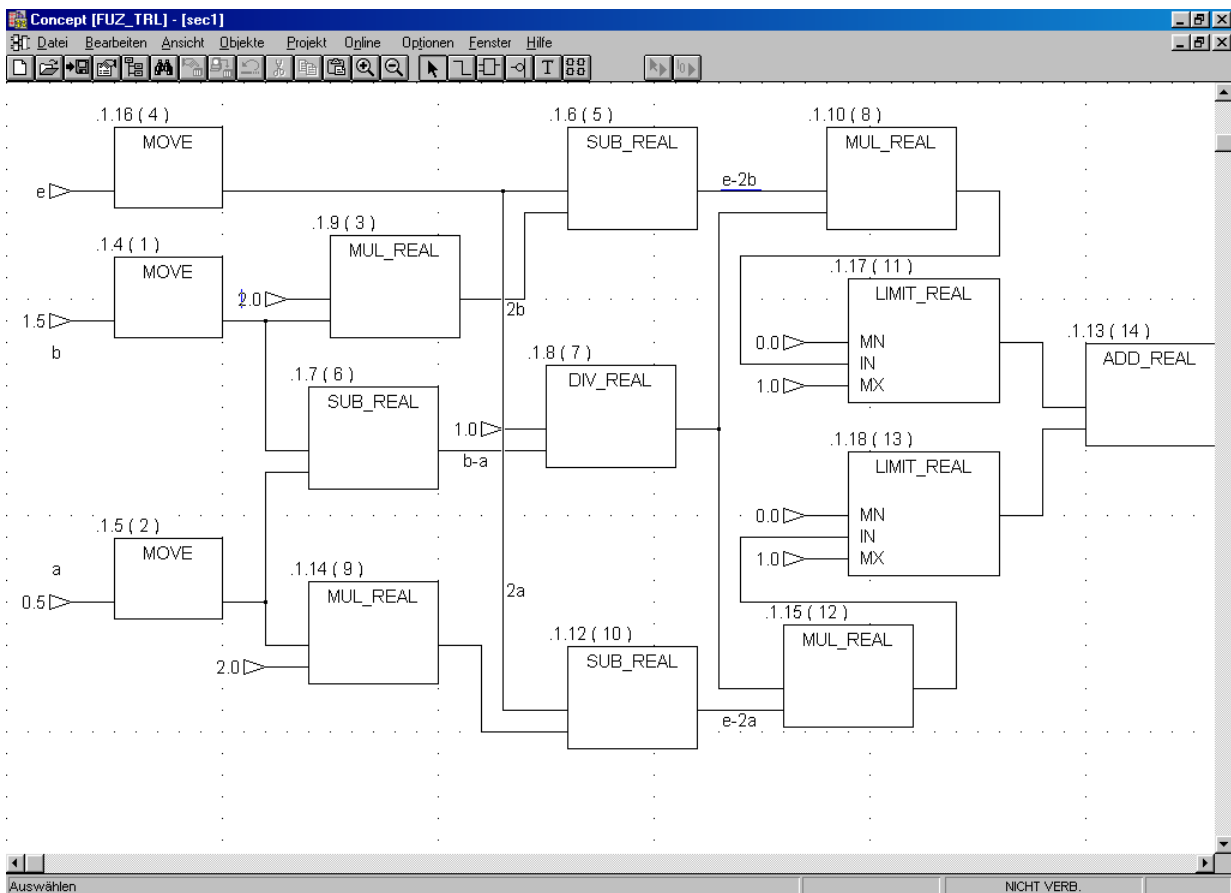


### 3.4.1 Fuzzy-Trial-Elemente mit IEC-Sprachen

Als Beispiel soll die symmetrische Dreiecks-Zugehörigkeitsfunktion mit Stützstellen  $S_1 = 2a$ ,  $S_0 = a + b$  und  $S_2 = 2b$  (**Bild 3.49**) mit der Funktionsbausteinsprache programmiert werden. Das Programm für  $a = 0,5$  und  $b = 1,5$  ist in **Bild 3.50** dargestellt. Mehr über die SPS-Realisierung von Reglern mit Fuzzy-Trial-Elementen findet man in [27].



**Bild 3.49** Dreieckfunktion mit Stützstellen  $S_1 = 2a$ ,  $S_0 = a + b$  und  $S_2 = 2b$



**Bild 3.50** Symmetrisches Fuzzy-Trial-Glied mit IEC-Funktionsbausteinsprache. Die MOVE-Blöcke sind lediglich zur Abzweigung von Eingangssignalen eingesetzt.

### 3.4.2 KNN ohne Backpropagation

Eine Regelstrecke ist gegeben:

$$G_S(s) = \frac{K_{PS1} K_{PS2} K_{IS}}{s(1+sT_1)(1+sT_2)}$$

Die Streckenparameter sind:

$$K_{PS1} = 1,5; \quad T_1 = 1,5 \text{ s}; \quad K_{PS2} = 0,1; \quad T_2 = 0,1 \text{ s}; \quad K_{IS} = 1 \text{ s}^{-1}$$

Soll die Strecke mit einem P-Regler geregelt werden, so wird zuerst eine Ersatzzeitkonstante gebildet:

$$T_E = T_1 + T_2 = 1,5 \text{ s} + 0,1 \text{ s} = 1,6 \text{ s}$$

Dann folgt die Reglereinstellung nach dem Betragsoptimum:

$$G_0(s) = \frac{K_{PR} K_{PS1} K_{PS2} K_{IS}}{s(1+sT_E)} \text{ und}$$

$$K_{PRopt} = \frac{1}{2K_{PS1}K_{PS2}K_{IS}T_E} = \frac{1}{2 \cdot 1,5 \cdot 0,1 \cdot 1 \text{ s}^{-1} \cdot 1,6 \text{ s}} = 2,08$$

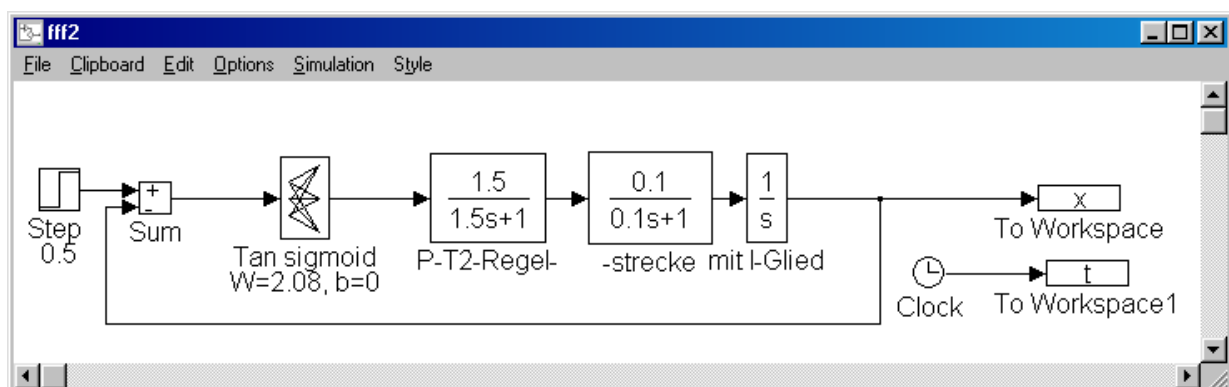
Nun wird ein Neuro-Trial-Glied anstelle des P-Reglers verwendet. Der Wirkungsplan des Regelkreises mit dem MATLAB-Block *tansig* ist in **Bild 3.51** gezeigt. Das Trial-Element besitzt keinen Lernalgorithmus. Seine statische Kennlinie hat hyperbolische tanh-Funktion. Man kann das Gewicht  $W$  von

$$W = K_{PRopt}$$

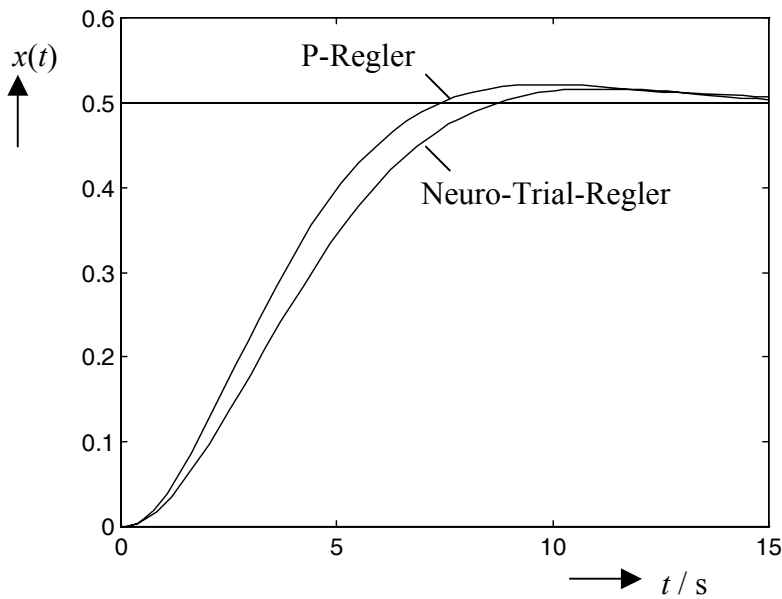
und den Schwellenwert  $b$  entweder direkt im Fenster des Simulink-Blocks eingeben oder über das MATLAB-Command-Fenster mit dem Befehl

$$W = 2.08; \quad b = 0;$$

deklarieren. Der zweite Weg ist zu empfehlen, da man die Deklarationsbefehle bei



**Bild 3.51** Regelkreis mit dem Neuro-Trial-Glied als Regler



**Bild 3.52** Sprungantworten des Regelkreises mit der I-T<sub>2</sub>-Strecke

dem Übergang von Neuro-Trial-Element zu einem vollständigen KNN lediglich mit dem Lernalgorithmus im MATLAB-Command-Fenster zu erweitern braucht.

Die Sprungantworten von beiden Regelkreisen, d. h. mit dem P-Regler und mit dem Neuro-Trial-Glied als Regler, sind bei einem Sprung der Führungsgröße von 0,5 in **Bild 3.52** gezeigt. Da die Kennlinie des Neuro-Trial-Glieds nichtlinear ist, unterscheiden sich die beiden Kurven bei Anstieg und Überschwingung.

Für die bessere Anpassung des Neuro-Reglers an die Strecke soll der Regler mit Neuro-Fuzzy-Elementen, wie in [14] gezeigt, vervollständigt werden.

### 3.1 Abschlussbemerkung

Bei der Behandlung von dualen Regelkreisen, sei es getrennte bzw. kooperative oder simultante bzw. hybride Arbeitsweise, wurden in diesem Kapitel einige Glieder mit neuen Funktionen eingeführt.

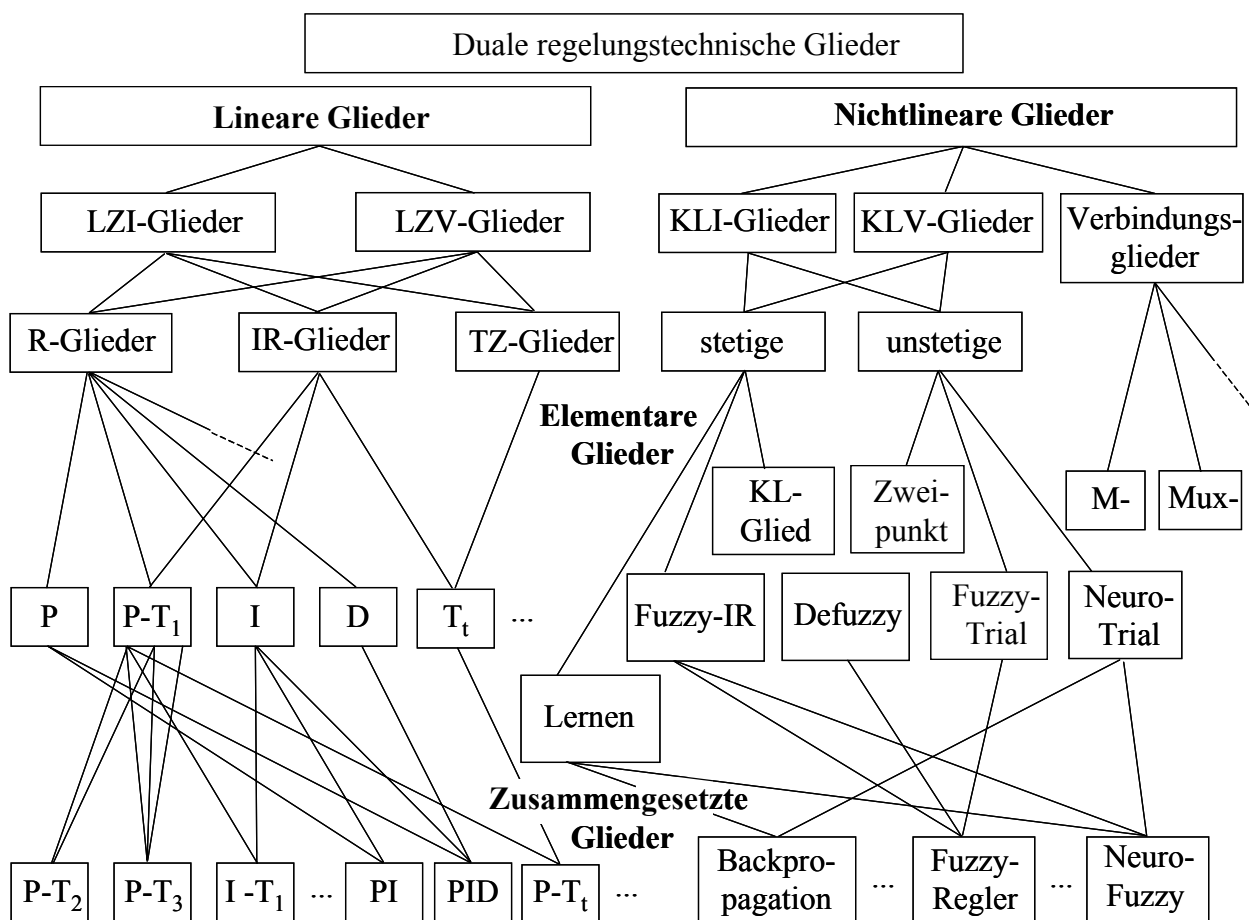
Um die Struktur und das Regelkreisverhalten von beiden Kreisen, des Regelkreises und des Lernkreises, einfach und übersichtlich darzustellen, werden die vorhandenen regelungstechnischen und die neuen wissensbasierten Glieder im nächsten Kapitel einheitlich betrachtet und klassifiziert. Dafür werden neue Symbole für lineare regelungstechnische Grundglieder mit variierbaren Parametern sowie für nichtlineare wissensbasierte Elemente eingeführt. Anhand von Beispielen wird die Bildung von Wirkungsplänen und die Funktionsweise für duale Regelkreise demonstriert.

## 4.1 Duale regelungstechnische Glieder

### 4.3.1 Klassifikation und Zusammenstellung

Die in den vorherigen Abschnitten behandelten klassischen DGL-basierten Grundglieder und die eingeführten wissensbasierten Glieder sind im **Bild 4.7** zusammengefasst. Einige davon sind in der **Tabelle 4.10** als Blöcke des Wirkungsplans dargestellt. Die KL-Glieder (Kennlinienglieder) sind die mittels einer nichtlinearen Funktion bzw. einer nichtlinearen statischen Kennlinie definierten Glieder mit konstanten Parametern (KLI) oder variierbaren Parametern (KLV). Beispielhaft ist gezeigt, dass die Parameter eines LZV-Glieds von außen (z. B. von einem anderen Glied) und die eines KLV-Glieds von innen (z. B. wegen geänderten Arbeitspunkten) angesteuert werden.

Im Allgemeinen kann man kaum einen dualen Kreis finden, der nur aus linearen oder nichtlinearen Gliedern besteht. Dass die zusammengesetzten Glieder für lineare und nichtlineare Bereiche in Tabelle 4.10 getrennt gebildet sind, soll auch als Beispiel betrachtet werden.



**Bild 4.7** Klassifikation von Übertragungsgliedern dualer Regelkreise

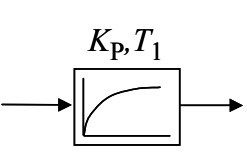
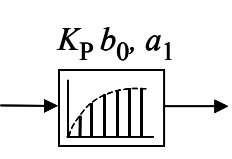
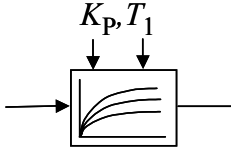
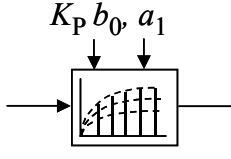
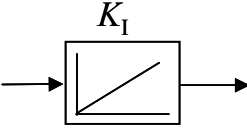
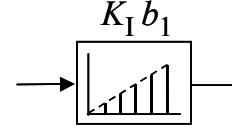
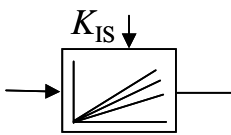
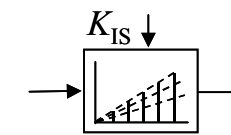
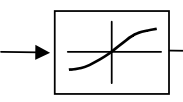
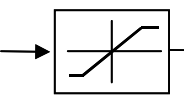
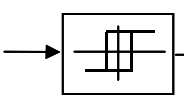
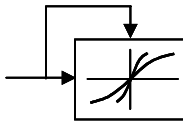
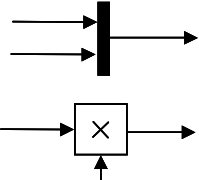
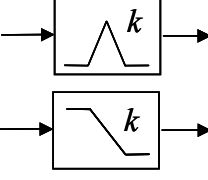
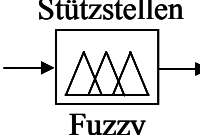
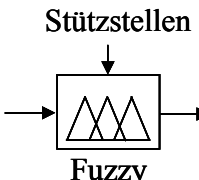
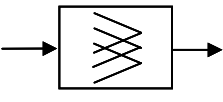
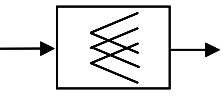

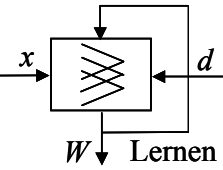
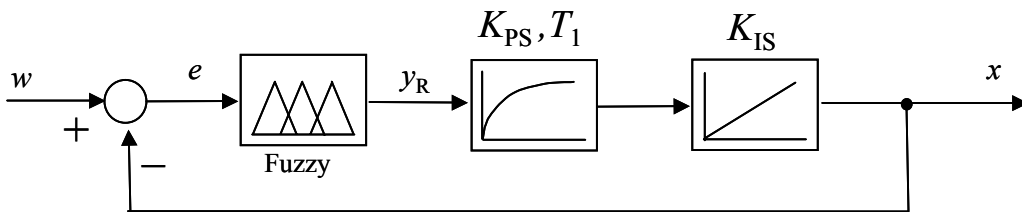
| Lineare Übertragungsglieder                                                                                                  |                                                                                                                                  |                                                                                                                               |                                                                                                                                    |
|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| Zeitinvariante LZI                                                                                                           |                                                                                                                                  | Zeitvariante LZV                                                                                                              |                                                                                                                                    |
| analoge                                                                                                                      | digitale                                                                                                                         | analoge                                                                                                                       | digitale                                                                                                                           |
| <br>$K_P, T_1$<br>P-T <sub>1</sub> -Glieder | <br>$K_P b_0, a_1$<br>P-T <sub>1</sub> -Glieder | <br>$K_P, T_1$<br>P-T <sub>1</sub> -Glieder | <br>$K_P b_0, a_1$<br>P-T <sub>1</sub> -Glieder |
| <br>$K_I$<br>I-Glieder                      | <br>$K_I b_1$<br>I-Glieder                      | <br>$K_{IS}$<br>I-Glieder                   | <br>$K_{IS}$<br>I-Glieder                       |
| Nichtlineare Kennlinienglieder                                                                                               |                                                                                                                                  |                                                                                                                               |                                                                                                                                    |
| KLI<br>(invariante, nicht einstellbare)                                                                                      |                                                                                                                                  |                                                                                                                               | KLV<br>(variante, einstellbare)                                                                                                    |
| stetige                                                                                                                      | unstetige                                                                                                                        |                                                                                                                               | stetige/unstetige                                                                                                                  |
| <br>Nichtlineare Funktion                 | <br>Linearität mit Sättigung                  | <br>Zweipunktglied                        | <br>KLV-Glieder                               |
| <br>Mux- und M-Glieder                    | <br>Fuzzy-Trial                               | <br>Stützstellen<br>Fuzzy                 | <br>Stützstellen<br>Fuzzy                     |
| <br>Neuro Trial (ohne Lernen)             | <br>Neuro Trial (Lernen)                      | <br>KNN-Hybrid                            | <br>Lernen                                    |

Tabelle 4.10 Beispiele von linearen und nichtlinearen Gliedern dualer Regelkreise

### 4.3.2 Beispiele

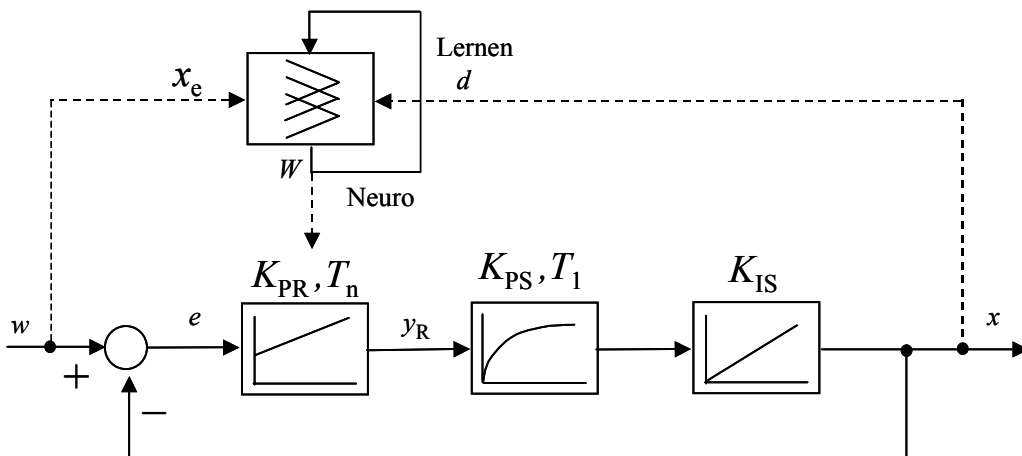
#### Beispiel 1: Fuzzy-Regler mit einer LZI-Strecke (Bild 4.8)



**Bild 4.8** Dualer Regelkreis mit P-Fuzzy-Regler und LZI-Regelstrecke

Der wissensbasierte Regler ist mit konstanten Kennlinien aufgebaut. Die klassische Regelstrecke besteht aus einem P-T<sub>1</sub>- und einem I-Glied, ebenfalls mit konstanten Parametern.

#### Beispiel 2: PI-Regler mit einer LZI-Strecke



**Bild 4.9** Dualer Regelkreis mit PI-Regler und LZI-Regelstrecke

Die klassische LZI-Strecke soll mit dem ebenfalls klassischen PI-Regler geregelt werden. Die Einstellung des Reglers (Autotuning) erfolgt mit Hilfe eines wissensbasierten Gliedes (Backpropagation-KNN), wie in **Bild 4.9** gezeigt ist. Es handelt sich um eine kooperative Behandlung, da erst das KNN trainiert und dann das Ergebnis dem Regler übergeben wird.

#### Beispiel 3: Fuzzy-Regler mit einer KLV-Regelstrecke (Bild 4.10)

Der Arbeitspunkt der Regelstrecke wandelt sich langsam. Der davon abhängige Proportionalbeiwert  $K_{PS}$  soll vom KNN erkannt und dem Fuzzy-Regler in Form von einstellbaren Stützstellen zwecks Optimierung übergeben werden.

## 6.4.1 Diagonaler Fuzzy-Regler

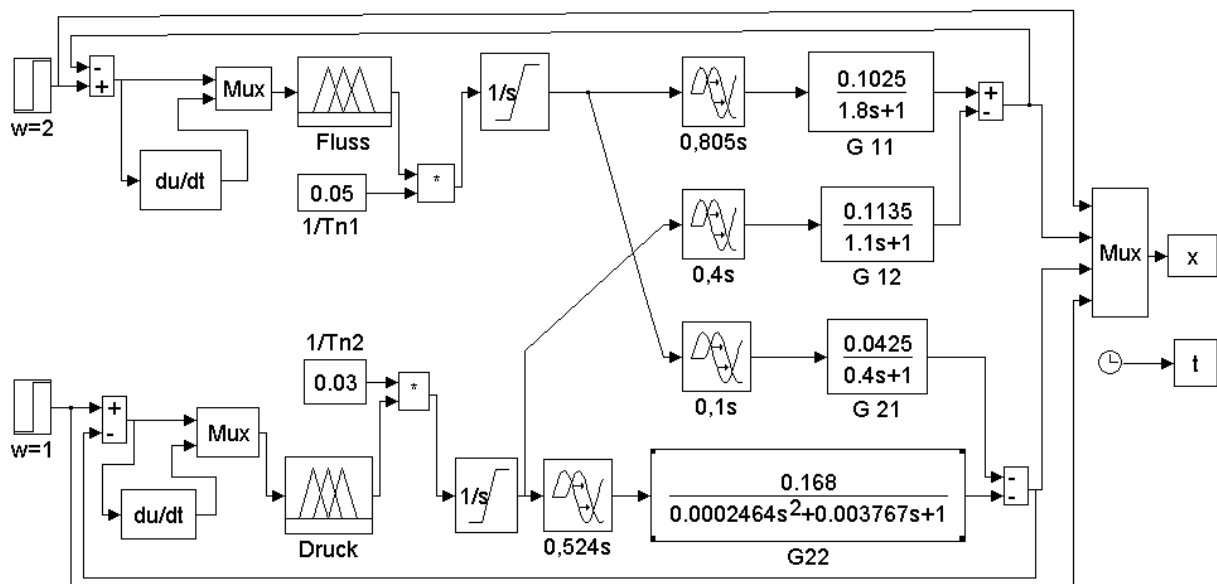
Der Entwurf von Fuzzy-Reglern mit einem PLS für die Regelgrößen *Durchfluss* und *Druck* einer MIMO-Regelstrecke (Molekularfilter) wurde bereits im Abschnitt 2.4 behandelt. Hier wird der Entwurf mit der *Fuzzy-Logic Toolbox* von MATLAB vorgenommen, um dann die Fuzzy-Regler unter Simulink mit dem Streckenmodell zu verbinden. Nach dem Konzept der dualen Mehrgrößenregelung wird dann unter MATLAB ein KNN programmiert, das die Stützstellen der Zugehörigkeitsfunktionen der Fuzzy-Regler an die Arbeitspunkte anpassen wird. Das Gesamtsystem lässt sich unter Simulink einfach verbinden und simulieren.

Eine Einführung in die Fuzzy-Logic-Toolbox findet man in [17]. In **Bild 6.14** sind die entworfenen PI-Fuzzy-Regler, *Fluss* und *Druck*, als Diagonalregler gezeigt. Bevor man eine Simulation unter Simulink startet, sollten die Stellgrößen bzw. die Ausgänge der Fuzzy-Regler deklariert und aus der Fuzzy-Logic-Toolbox im MATLAB-Command-Fenster zugewiesen werden. Dies erfolgt wie üblich durch Doppelklick auf das Block-Symbol *Fluss* oder *Druck*, nachdem ein Fenster zur Parametereinstellung, z. B. *FIS (Mask)*, wie in **Bild 6.15** erscheint. Die Stellgrößen können beliebig bezeichnet werden, z. B. *a* für den Regler *Fluss* und *b* für *Druck*. Sie soll dann im MATLAB-Command-Fenster mit dem folgenden Befehl

*Variable = Befehl ('Dateiname des Fuzzy-Reglers');*

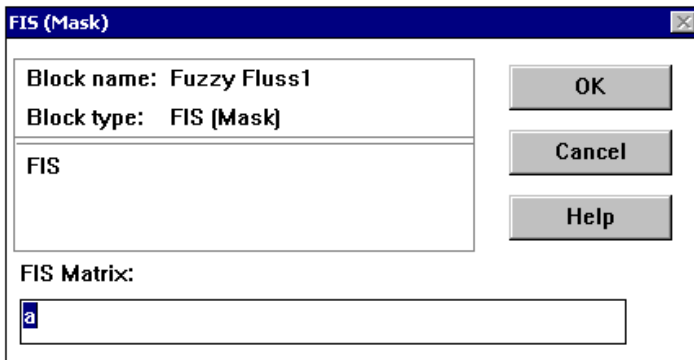
in das Simulink-Programm übertragen werden, z. B. gilt für die Simulink-Blöcke *FuzzyFluss1* und *FuzzyDruck1* mit den MATLAB-Dateien *Fluss.fis* und *Druck.fis*:

```
a = readfis('Fluss');
b = readfis('Druck');
```

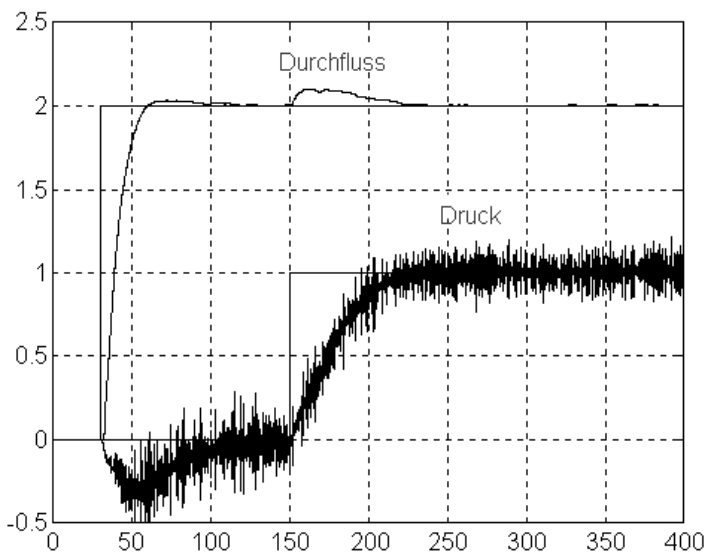


**Bild 6.14** Regelkreis mit Fuzzy-Diagonalregler





**Bild 6.15** Parametereingabe für Regler *FuzzyFluss1*



**Bild 6.16** Sprungantworten des Regelkreises mit Fuzzy-Diagonalregler

**Bild 6.16** zeigt die Eingangssprünge und die simulierten Sprungantworten der Regelgrößen *Durchfluss* und *Druck*. Es ist zu erkennen, dass die Sollwertsprünge einer Führungsgröße auf den anderen Regelkreis als Störgröße wirken. Für den positiven Sprung des Sollwerts *Durchfluss* fällt der Druck zunächst ab und wird dann vom entsprechenden Regler wieder ausgeregelt. Beim Sprung der Führungsgröße *Druck* steigt der Durchfluss an. Das dynamische Verhalten des Regelkreises ist somit nicht zufriedenstellend.

#### 6.4.2 Entkoppelter dualer Mehrgrößenregler

Werden nun die im Abschnitt 6.3.2 berechneten P-Glieder zur Entkopplung verwendet, wie in **Bild 6.17** gezeigt ist, verbessert sich nur das dynamische Verhalten der Regelgröße *Druck* (**Bild 6.18**). Da dabei das Verhalten der Regelgröße *Durchfluss* verschlechtert wird, sollten Fuzzy-Regler optimiert werden.