

7 Feed-Forward-Fuzzy-Regler (FFF-Regler)

Moderne verfahrenstechnische Prozesse werden von Prozessleitsystemen gesteuert, überwacht und dokumentiert. Stetig steigende Ansprüche an die Produktqualität in der Industrie erfordern höchste Steuerungs- und Regelungsqualität. Neben Problemen der Sicherheit und Zuverlässigkeit spielt die optimale Einstellung von Regelkreisen eine wichtige Rolle.

Die Einstellung eines Regelkreises wird grundsätzlich von der Kenntnis der Regelstrecke und der Wahl des hierfür adäquaten Einstellverfahrens bestimmt. In der Praxis versucht man jedoch oft, den für die Identifikation erforderlichen mathematischen Aufwand umzugehen. Immer häufiger beginnt die Dimensionierung des Reglers mit Simulation und geht über Prozessvisualisierung bis hin zur Online-Parametereinstellung durch Ausprobieren. Dass dabei die relativ großen Zeitkonstanten der industriellen Prozessen und die immer kleineren Reaktionszeiten der computerbasierten Prozessleitsysteme eine „gute“ Sprungantwort ermöglichen, ist ein Argument zu Gunsten der Anhänger des „trial-and-error“-Verfahrens. Dafür sprechen auch die zahlreichen Selbsteinstellungsverfahren (Autotuning) sowie die heuristischen Verfahren des Entwurfs von Fuzzy-Reglern.

Die Reglereinstellung im Zeitbereich durch Auswertung von Sprungantworten und die Nachbesserung von Kennwerten des Reglers mittels Prozessleitsystem ist also zum Industriestandard geworden. Will man die konventionellen regelungstechnischen Verfahren weiter erfolgreich für industrielle Regler anwenden, so muss man sich in erster Linie um die neuen Konzepte der Reglereinstellung an Displays bemühen. Eine Kombination von konventionellen Einstellverfahren mit direktem Zugriff an Reglerparameter kann zu einer Kompromisslösung führen.

Im vorliegenden Kapitel wird ein Fuzzy-Regler aus mehreren parallel geschalteten Baugruppen – den so genannten Feed-Forward-Neuronen – beschrieben. Diese Baugruppen, die ausschließlich aus regelungstechnischen Grundelementen bestehen, ermöglichen – wie bei konventionellen Fuzzy-Reglern – die Fuzzifizierung der Regeldifferenz des Regelkreises. Die Inferenzmechanismen – Regelbasis und Defuzzifizierung – sind durch die Bildung mehrerer paralleler Regelkreise ersetzt. Damit lässt sich der Fuzzy-Regler nach konventionellen Verfahren wie Betragsoptimum einstellen. Die so errechneten optimalen Kennwerte des Gesamtreglers werden zwischen den parallel geschalteten Neuronen mit Hilfe von Bedien- und Beobachtungssystem weiterhin so verteilt, dass die gewünschte Kennlinie des Reglers erreicht wird.

Die SPS-Implementierung so eines Feed-Forward-Fuzzy-Reglers (FFF-Reglers) erfolgt über Funktionsbausteine, die unter dem Programm Concept erstellt und mit Modicon Micro 512/612 getestet wurden. Zur Visualisierung dient das Programm Lookout mit Datenübertragung vom Prozess zu SPS über Modbus.

Die Regelung mit dem FFF-Regler wird anhand der Prozesstafel ET722 (siehe Abschnitt 5.1.3) demonstriert und für verschiedene Parameterkombinationen untersucht. Eine optimale Temperaturregelung soll mit Hilfe einer SPS als FFF-Regler erreicht werden. Dabei nimmt die Visualisierung eine bedeutende Stellung sowohl in der Prozessüberwachung als auch in der Konfigurierung und Einstellung der Regler ein. Die Simulationsergebnisse zeigen die wesentliche Verbesserung der Regelgüte im Vergleich zu konventionellen P-, PD-T1- und PID-Reglern.

7.1 Aufbau des FFF-Reglers

7.1.1 Merkmale von Fuzzy-Reglern

Ein konventioneller Fuzzy-Regler stellt einen nichtlinearen Regler dar, der aus gleichen, parallel eingerichteten Baugruppen besteht und hauptsächlich digital realisiert wird.

Die Merkmale von Fuzzy-Reglern folgen der Fuzzy-Logik:

- Fuzzifizierung der Regeldifferenz und ggf. ihrer zeitlichen Ableitung
- Inferenz
- Regelbasis
- Defuzzifizierung der Stellgröße

Die Fuzzifizierung als mathematische Darstellung unscharfer Mengen spielt für die Fuzzy-Regler eine wichtige Rolle. Die Inferenzmechanismen und die Regelbasis bestimmen die Funktion des Reglers. Für die Defuzzifizierung stehen zahlreiche Methoden zur Verfügung, die eine Zusammenfassung mehrerer Regeln ermöglichen.

Den Vorteilen der Fuzzy-Logik, die eine nichtformalisierte Beschreibung von Prozessen ohne Fachwissen ermöglicht, stehen als Nachteile die relativ langen Antwortzeiten und die Diskrepanz zwischen den konventionellen und Fuzzy-Entwurfsmethoden gegenüber, was den Anwendungsbereich von Fuzzy-Reglern stark begrenzt. Die Parametern werden oft durch das einfaches Ausprobieren eingestellt. Die Inferenzmechanismen und die Regelbasis ändern sich für verschiedenen Reglertypen nur gering. Die Defuzzifizierungsmethoden, die eine

einheitliche Stellgröße aus mehreren Baugruppen bilden, werden oft in praktischen Fällen durch einfache Summierung ersetzt.

Ein Fuzzy-Regler, der die Regeldifferenz auf mehrere kleine Wertebereiche zerlegt, jedoch ohne Regelbasis, Inferenz und Defuzzifizierung mit Feed-Forward-Neuronen funktioniert, wird als Feed-Forward-Fuzzy-Regler bezeichnet.

Die zu diesem Zweck entwickelten Feed-Forward-Neuronen bestehen ausschließlich aus regelungstechnischen Grundelementen, die die Einstellung der FFF-Regler mit den konventionellen Methoden der linearen Regelungstechnik ermöglichen.

7.1.2 Feed-Forward-Neuronen

Struktur und Funktionsweise der Feed-Forward-Neuronen wird anhand eines einfachen neuronalen Netzes mit Backpropagation erklärt. Der Algorithmus besteht aus drei Rechenschritten:

Schritt 1: Berechnung der gewichteten Summe der Eingänge x_i unter Beachtung des Schwellenwertes θ (Aktivierungswert α):

$$\alpha = \sum_{i=1}^n W_{ji} \cdot x_i - \theta_i$$

Schritt 2: Berechnung des Neuronenausgangs $y = f(\alpha)$ nach binäre oder sigmoidale Aktivierungsfunktion (Abschnitte 6.2.1 – 6.2.3)

Schritt 3: Eingabe des gewünschten Ausgangs (des Sollwertes) d

Schritt 3: Realisierung des Lernalgorithmus bzw. der Gewichtsänderung:

$$W_{1(\text{neu})} = W_1 + \Delta W_1 \quad \text{mit} \quad \Delta W_1 = \eta \cdot (d - y) \cdot x_1$$

$$W_{2(\text{neu})} = W_2 + \Delta W_2 \quad \text{mit} \quad \Delta W_2 = \eta \cdot (d - y) \cdot x_2$$

Die Berechnung des Ausgangssignals und die nachfolgende Backpropagation-Lernregel ist im **Bild 7.1** gezeigt. Die links im Bild gezeigten P-Glieder sind Gewichte W_1 und W_2 des Netzes. Durch einen Soll-Ist-Wert-Vergleich am Ausgang des Zweipunktgliedes wird das Fehlermaß $e = \eta \cdot (d - y)$ ermittelt und den P-Gliedern der Backpropagation-Lernregel weitergegeben (sind rechts im **Bild 7.1**

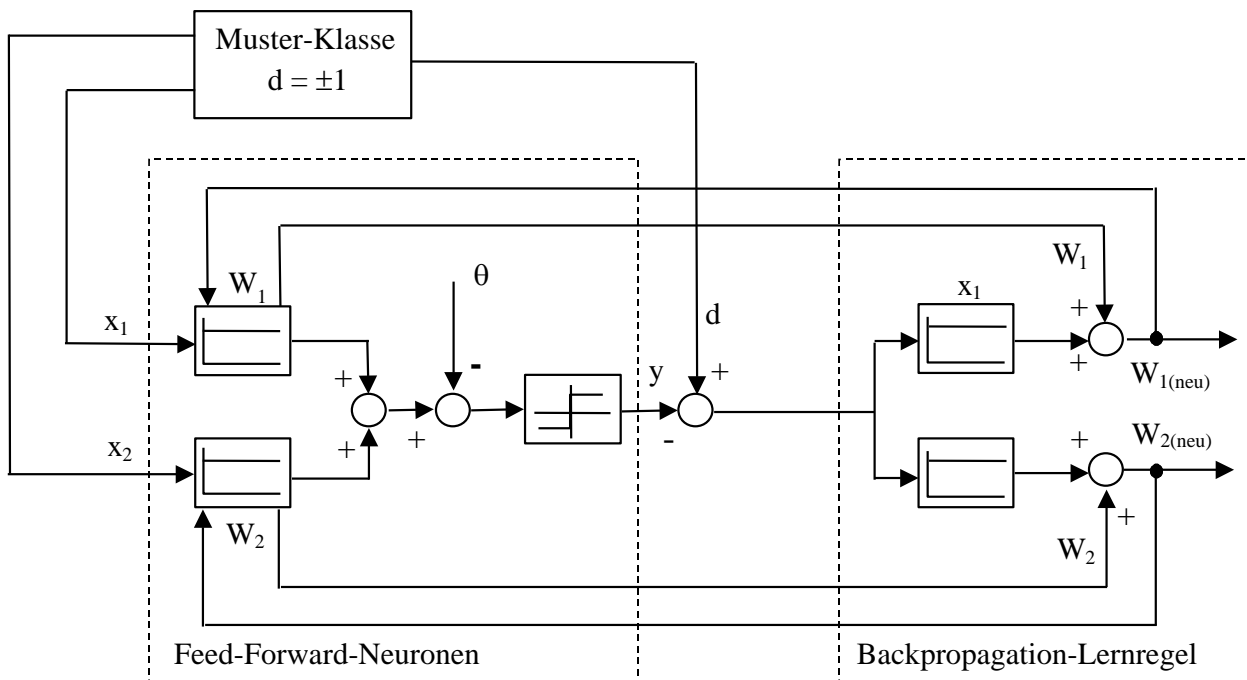


Bild 7.1 Wirkungsplan eines Lernvorgangs mit der Lernschrittweite $\eta = 1$

gezeigt). Die Proportionalbeiwerte dieser P-Glieder sind die aktuellen Werte x_1 und x_2 . Die Ausgänge sind Gewichte $W_{1(\text{neu})}$ und $W_{2(\text{neu})}$, die dem Netz in Form von Proportionalbeiwerten statt der anfänglichen Gewichten W_1 und W_2 übergeben werden. Der Block *Muster-Klasse* gehört zum Lernverfahren und dient der Vorgabe der aktuellen Werte x_1, x_2 und der Bildung von Sollwerten d des Netzes.

Um einen Fuzzy-Regler ohne iterativen Lernvorgang zu strukturieren, wird nur der Wirkungsplan von Feed-Forward-Neuronen betrachtet. Dabei entfallen die Blöcke *Backpropagation-Lernregel* und *Muster-Klasse*. Die Signalübertragung von zwei Feed-Forward-Neuronen ist im **Bild 7.2** in eine für Fuzzy-Bausteine typische Strukturform gebracht.

Im **Bild 7.3** ist nun eine Baugruppe des Fuzzy-Reglers für die linguistische Variable *Null* mit der statischen Kennlinie dargestellt. Mit Hilfe dieser Baugruppe weiterhin als Feed-Forward-Neuron bezeichnet wird die Regeldifferenz des Regelkreises mittels dreieckförmiger Kennlinien fuzzifiziert:

Die Einstellparameter des Neurons sind:

- Steigung der Kennlinie $K_{p\text{NU}}$ (Proportionalbeiwert)
- Schwellenwert θ_{NU} (max. Wert der Kennlinie)
- Stützpunkt $e_{m\text{NU}}$ (Mittelpunkt der Kennlinie)

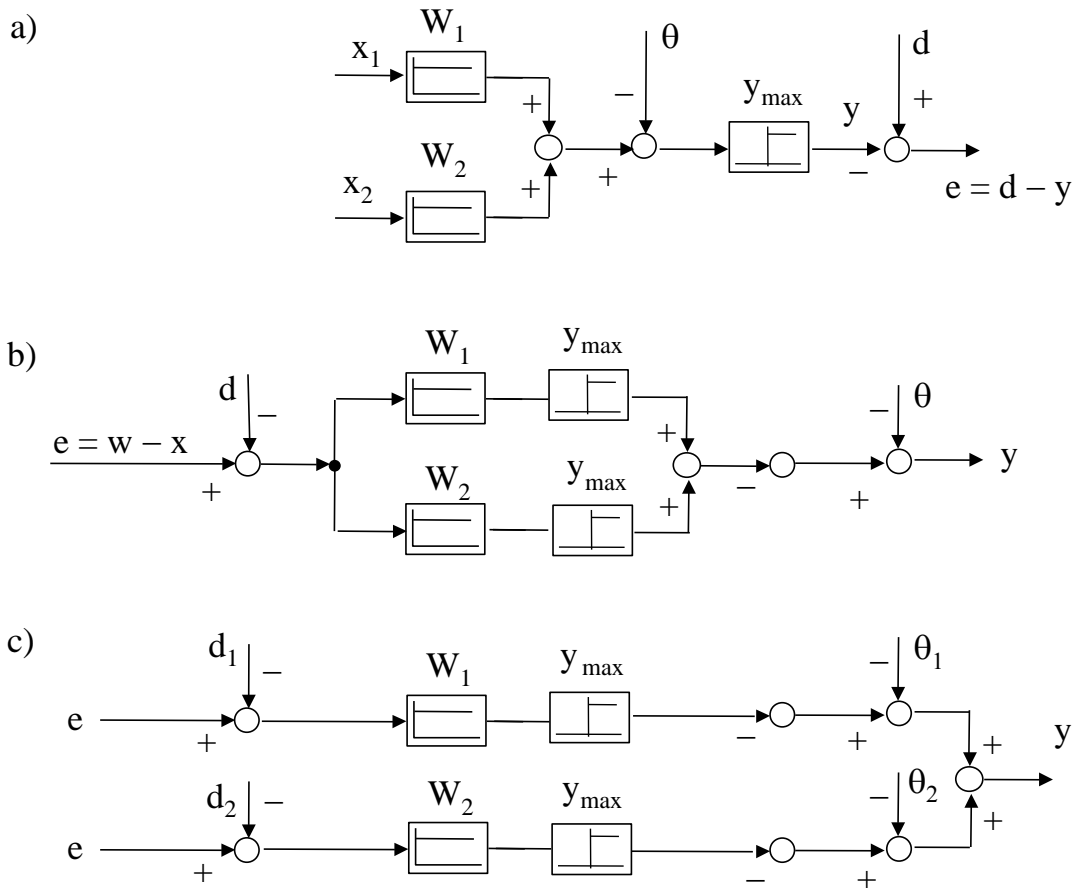


Bild 7.2 Anpassung der Feed-Forward-Neuronen an einen Fuzzy-Regler:

- Originalform (e ist das Fehlermaß mit Sollwert d)
- Umformung des Wirkungsplans (e ist die Regeldifferenz des Regelkreises mit Sollwert w)
- Ergebnis: Fuzzifizierung der Regeldifferenz e mittels zwei Baugruppen. Addition gilt als Defuzzifizierung (y ist die defuzzifizierte Stellgröße)

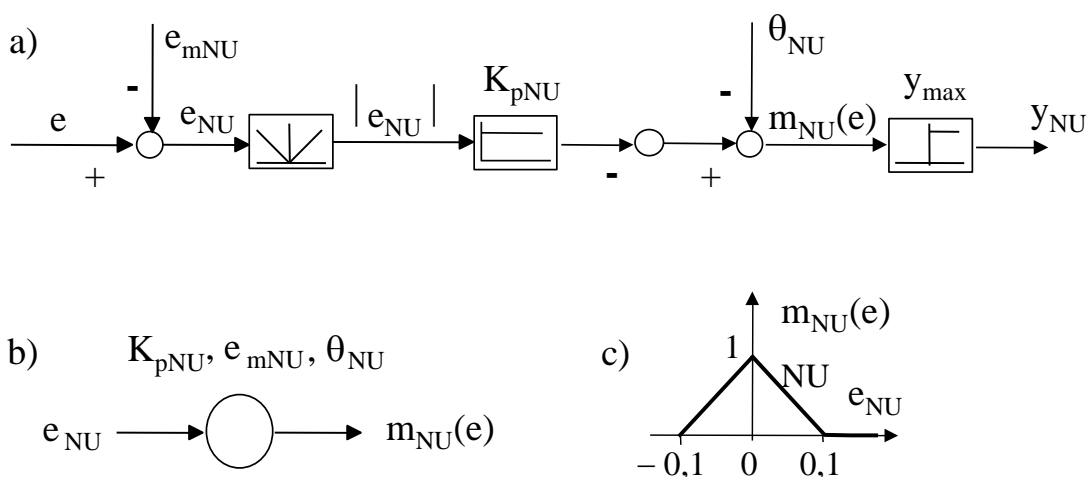


Bild 7.3 Struktur und Variablen des Feed-Forward-Neurons *Null* (NU) eines Fuzzy-Reglers: a) Wirkungsplan; b) Symbolische Darstellung mit Einstellparametern; c) statische Kennlinie.

7.1.3 Einstellregeln für zwei benachbarte Neuronen

Die statische Kennlinie des Neurons NU der Baugruppe *Null* wird durch zwei Gleichungen beschrieben (**Bild 7.3**):

$$m_{\text{NU}} = -K_{\text{pNU}} \cdot e_{\text{NU}} + \theta_{\text{NU}} \quad 0 < e_{\text{NU}} < 0,1$$

$$m_{\text{NU}} = +K_{\text{pNU}} \cdot e_{\text{NU}} + \theta_{\text{NU}} \quad -0,1 < e_{\text{NU}} < 0$$

Das Ausgangssignal dieser Baugruppe wird erstellt, wenn der Aktivierungswert m_{NU} positiv und kleiner als der Schwellenwert θ_{NU} ist.

Die Einstellparameter des Neurons der Baugruppe *Null* sind:

- $K_{\text{sNU}} = 10$ Steigung der Kennlinie
- $\theta_{\text{NU}} = 1$ Schwellenwert (max. Wert)
- $e_{\text{mNU}} = 0$ Mittelpunkt der Kennlinie

Bei mehreren Baugruppen werden für alle Neuronen gleiche Werte angenommen:

- $K_{\text{pi}} = 10$
- $\theta_i = 1$

Die Stützpunkte e_{mi} sind Mittelpunkte der Neuronen-Kennlinien (**Bild 7.4**).

Für die optimale Einstellung eines FFF-Reglers gilt folgende Bedingung: Der Proportionalbeiwert eines Neurons soll linear zwischen Proportionalbeiwerten der zwei benachbarten Neuronen verteilt werden.

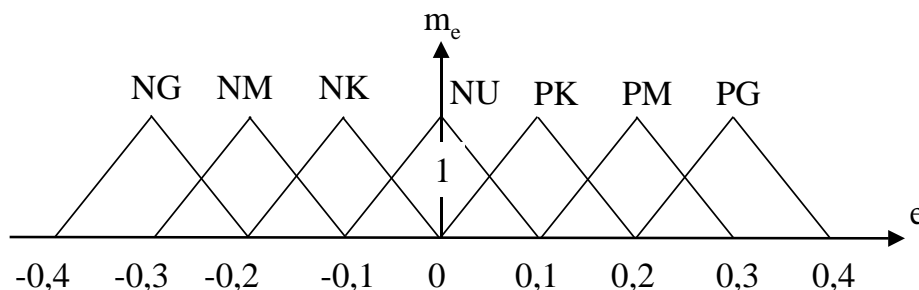


Bild 7.4 Positiv und negativ wirkende Feed-Forward-Neuronen: *Positiv-klein*, *-mittel*, *-groß* (PK), (PM), (PG) und *Negativ-klein*, *-mittel*, *-groß* (NK), (NM), (NG) mit den Stützpunkten:
 $e_{\text{mNG}} = -0,3$ $e_{\text{mNM}} = -0,2$ $e_{\text{mNK}} = -0,1$ $e_{\text{mPK}} = 0,1$ $e_{\text{mPM}} = 0,2$ $e_{\text{mPG}} = 0,3$

Da jedes Neuron drei Einstellparameter hat, sind insgesamt 15 Parameter bei der Dimensionierung eines Reglers aus fünf Baugruppen zu berücksichtigen. Jedoch kann die Reglereinstellung durch entsprechende Konfigurierung wesentlich vereinfacht werden, wenn man annimmt, dass die Proportionalbeiwerte von positiv und negativ wirkenden Neuronen gleich sind und sich nur durch ihr Vorzeichen unterscheiden.

Beispiel: Der Proportionalbeiwert des Neurons NU $K_{pNU} = 6,6$ wird folgendermaßen unter Neuronen NK und NG sowie PK und PG verteilt:

$$K_{pNU} = 0,5 \cdot (K_{pNK} + K_{pNG}) = 6,6 \Rightarrow K_{pNK} = 3,3 \text{ und } K_{pNG} = 9,9$$

$$K_{pNU} = 0,5 \cdot (K_{pPK} + K_{pPG}) = 6,6 \Rightarrow K_{pPK} = 3,3 \text{ und } K_{pPG} = 9,9$$

Die Proportionalbeiwerte einzelner Neuronen kann man aus dem Proportionalbeiwert des mittleren Neurons nach verschiedenen Regeln berechnen. Die Proportionalbeiwerte einzelner Neuronen dürfen allerdings die kritischen Werte, die zu instabilem Regelkreisverhalten führen, nicht überschreiten. Dafür ist eine Stabilitätsuntersuchung nach konventionellen Stabilitätskriterien durchzuführen, indem man das mittlere Neuron durch einen P-Regler ersetzt.

Die Proportionalbeiwerte von Neuronen sind von ihrer Position im Wirkungsplan des Reglers abhängig. Je weiter ein Neuron vom mittleren Neuron im Wirkungsplan entfernt ist, desto größer wird sein Proportionalbeiwert. Das mittlere Neuron stellt man am besten nach konventionellen Entwurfsverfahren ein, um optimales Regelkreisverhalten zu gewährleisten.

Beispiel: Der kritische Proportionalbeiwert eines konventionellen P-Reglers in einem Regelkreis, der als FFF-Regler mit fünf Neuronen zu regeln ist, beträgt $K_{Prkrit} = 30$. Der optimale Proportionalbeiwert des P-Reglers ist $K_{PRopt} = 6$. Dieser Wert dient als Proportionalbeiwert für das mittlere Neuron NU:

$$K_{pNU} = K_{PRopt}$$

Die Proportionalbeiwerte von *positiven* und *negativen* Neuronen sind gleich, d.h. $K_{pNK} = K_{pPK}$ und $K_{pNG} = K_{pPG}$. Sie werden mit K_{pNK_PK} und K_{pNG_PG} bezeichnet. Die Kennwerte werden im Verhältnis 1/9 zwischen Neuronen NK und PK sowie NG und PG verteilt:

$$K_{pNK_PK} = 0,5 \cdot K_{pNU} \Rightarrow K_{pNK} = K_{pPK} = 3$$

$$K_{pNG_PG} = 9 \cdot K_{pNK_PK} \Rightarrow K_{pNG} = K_{pPG} = 27$$

7.1.4 Entwurf eines FFF-Reglers

Ein FFF-Regler besteht aus mehreren parallel geschalteten Feed-Forward-Neuronen. Die Stellgröße $y(t)$ wird als Summe der einzelnen Neuronenausgängen y_i gebildet. Die Einstellung lässt sich an einem Beispiel veranschaulichen.

Beispiel: Eine Regelstrecke mit $K_{PS} = 1,6$, $T_1 = 0,1$ s, $T_2 = 0,5$ s und $K_{IS} = 0,2$ s⁻¹ soll mit einem FFF-Regler geregelt werden (**Bild 7.5**).

Die Kennlinien von fünf Neuronen sind im **Bild 7.6** gezeigt: $K_{pi} = \pm 5$ und $\theta_i = 1$.

Der Entwurf eines FFF-Reglers erfolgt in zwei Stufen. In der ersten Stufe wird der Regelkreis untersucht, in dem der FFF-Regler durch einen konventionellen P-Regler ersetzt. Der berechnete optimale Proportionalbeiwert K_{PR} des P-Reglers dient als Proportionalbeiwert K_{pNU} des mittleren Neurons, meistens *Null* genannt.

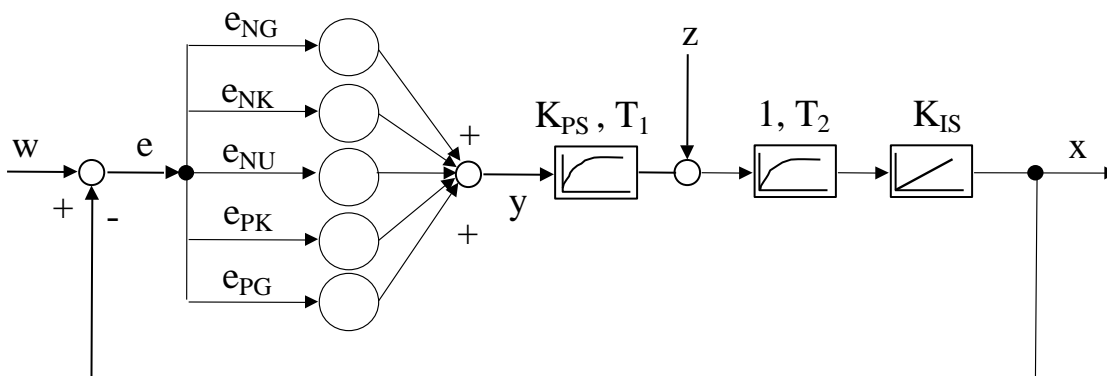


Bild 7.5 Beispiel eines Regelkreises mit dem FFF-Regler mit fünf Neuronen

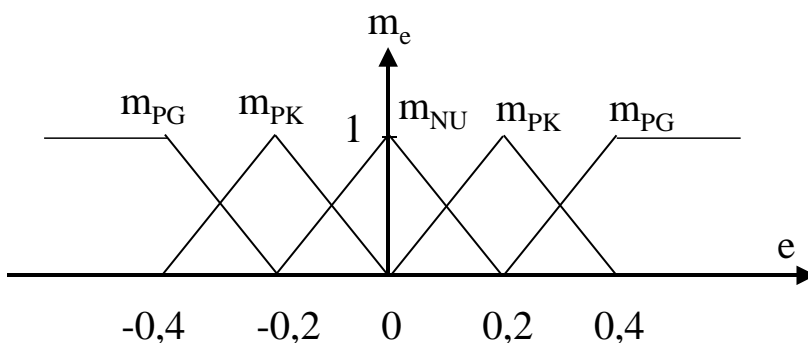


Bild 7.6 Kennlinien des FFF-Reglers mit fünf Neuronen zu obigem Beispiel: *Null* (NU), *Positiv-klein, -groß* (PK), (PG) und *Negativ-klein, -groß* (NK), (NG) mit den Stützpunkten: $e_{mNG} = -0,4$ $e_{mNK} = -0,2$ $e_{mNU} = 0$ $e_{mPK} = 0,2$ $e_{mPG} = 0,4$

In der zweiten Entwurfsstufe werden die Kennlinien der Neuronen gewählt und daraus die Kennwerte bestimmt. Die Proportionalbeiwerte der Neuronen werden aus dem Proportionalbeiwert des mittleren Neurons K_{pNU} berechnet.

Der Entwurf eines FFF-Reglers erfolgt nach folgenden Schritten:

- Die Zahl der Neuronen, die Kennlinien von Neuronen und die Stützpunkte $e_{mNU}, e_{mNK}, \dots, e_{mPG}$ werden aus den gegebenen Wertebereichen gewählt.
- Das Stabilitätsgebiet des Regelkreises mit gegebener Regelstrecke und einem P-Regler wird untersucht, der kritische Proportionalbeiwert K_{PRkrit} bestimmt.
- Der optimale Proportionalbeiwert K_{PRopt} des P-Reglers wird nach konventionellen Methoden berechnet, z.B. nach dem Betragsoptimum.
- Der P-Regler wird im Wirkungsplan des Regelkreises wieder durch den FFF-Regler ersetzt.
- Das mittlere Neuron *Null* wird mit dem Kennwert $K_{pNU} = K_{PRopt}$ eingestellt.
- Der Kennwert K_{sNU} des mittleren Neurons wird zwischen benachbarten Neuronen so verteilt, dass eine gewünschte Sprungantwort erreicht wird. Der kritische Proportionalbeiwert K_{PRkrit} darf dabei nicht überschritten werden.
- Die Schwellenwerte θ_i sind per Visualisierungssystem so einzustellen, dass die gewünschte Kennlinie des Reglers erreicht wird. Damit können die bleibende Regeldifferenz, die Anregelzeit und die maximale Überschwingweite beeinflusst werden. Es wird empfohlen, zuerst alle Schwellenwerte auf $\theta_i = 1$ zu setzen und danach online nachzubessern.

Die Stabilitätsuntersuchung des Regelkreises mit dem konventionellen P-Regler wird nach dem Hurwitz-Kriterium durchgeführt. Aus den Übertragungsfunktionen für den offenen und den geschlossenen Kreis

$$G_0(s) = \frac{K_{PR} \cdot K_{PS} \cdot K_{IS}}{s \cdot (1+sT_1) \cdot (1+sT_2)}$$

$$G_w(s) = \frac{G_0}{1+G_0} \quad \Rightarrow \quad G_w(s) = \frac{K_{PR} \cdot K_{PS} \cdot K_{IS}}{s \cdot (1+sT_1) \cdot (1+sT_2) + K_{PR} \cdot K_{PS} \cdot K_{IS}}$$

erhält man die charakteristische Gleichung

$$T_1 \cdot T_2 \cdot s^3 + (T_1 + T_2) \cdot s^2 + s + K_{PR} \cdot K_{PS} \cdot K_{IS} = 0$$

mit folgenden Koeffizienten:

$$a_3 = T_1 \cdot T_2 \quad a_2 = T_1 + T_2 \quad a_1 = 1 \quad a_0 = K_{PR} \cdot K_{PS} \cdot K_{IS}$$

Der kritische Wert K_{PRkrit} wird nach dem Hurwitz-Kriterium berechnet:

$$a_2 \cdot a_1 - a_3 \cdot a_1 \geq 0$$

$$K_{PRkrit} = \frac{T_1 + T_2}{T_1 \cdot T_2 \cdot K_{PS} \cdot K_{IS}} = \frac{0,1 \text{ s} + 0,5 \text{ s}}{0,1 \text{ s} \cdot 0,5 \text{ s} \cdot 1,6 \cdot 0,2 \text{ s}^{-1}} = 37,5$$

Um den Kennwert K_{PROpt} nach dem Betragsoptimum zu erhalten, eine Ersatzzeitkonstante T_E wird eingeführt:

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

Aufgrund der Übertragungsfunktion

$$G_0(s) = \frac{K_{PR} \cdot K_{PS} \cdot K_{IS}}{s \cdot (1 + sT_E)}$$

ergibt sich der Proportionalbeiwert des Reglers:

$$K_{PROpt} = \frac{1}{2 \cdot K_{PS} \cdot K_{IS} \cdot T_E} = \frac{1}{2 \cdot 1,6 \cdot 0,2 \text{ s}^{-1} \cdot 0,1 \text{ s} \cdot 0,5 \text{ s}} = 2,6$$

Der ermittelte Kennwert wird den *Null*-Neuron des FFF-Reglers zugeordnet:

$$K_{pNU} = K_{PROpt} = 2,6$$

Die Verteilung von Kennwerten der benachbarter Neuronen:

$$K_{pNU} = 0,5 \cdot (K_{pNK} + K_{pNG}) = 2,6$$

$$K_{pNU} = 0,5 \cdot (K_{pPK} + K_{pPG}) = 2,6$$

Zwei Parameter können beliebig gewählt werden, z.B. $K_{pNK} = K_{pPK} = 1,5$.

Dann werden die restlichen zwei Parameter berechnet: $K_{pNG} = K_{pPG} = 3,7$.

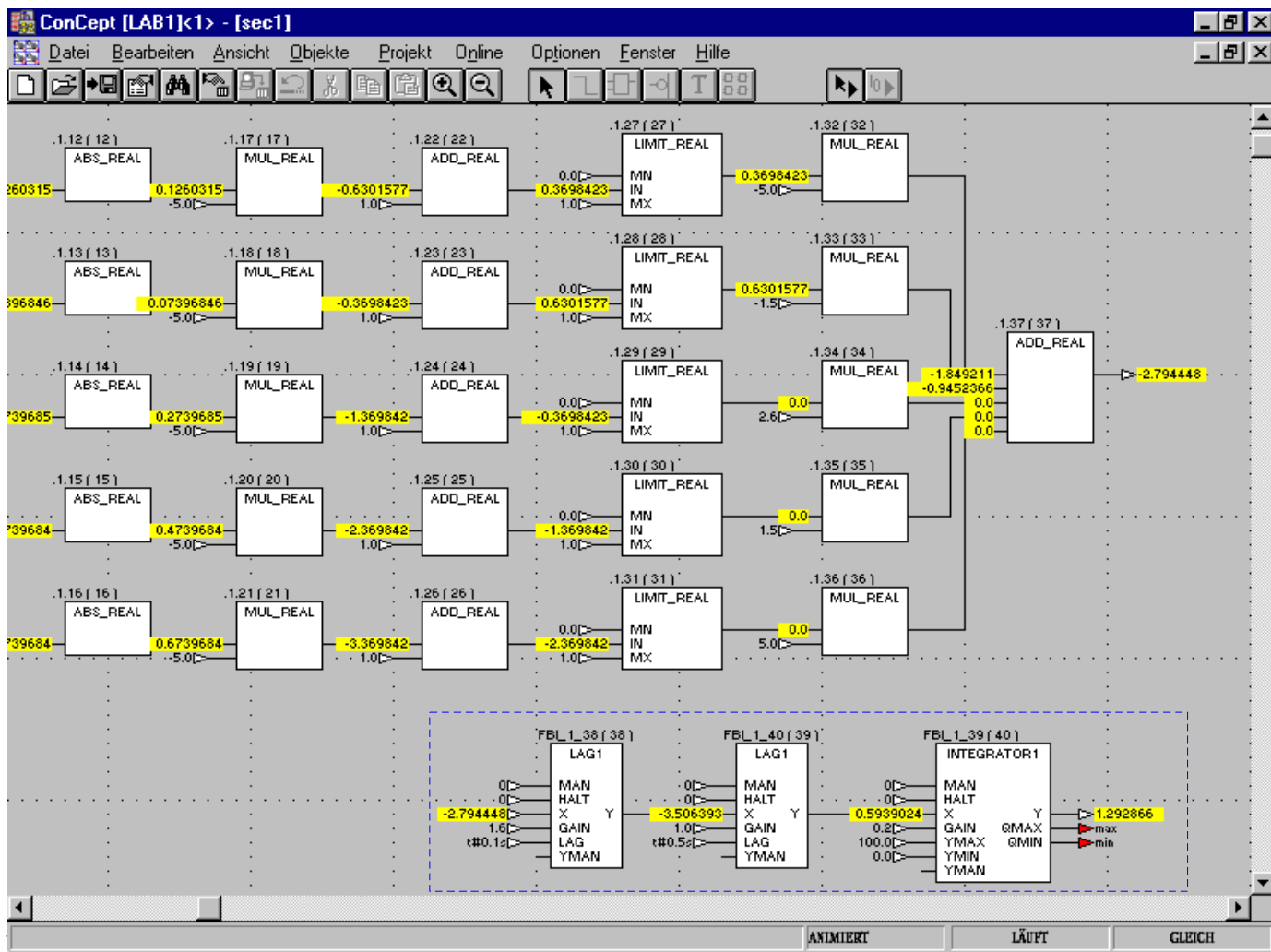
Auf die SPS-Implementierung eines selbseinstellenden FFF-Reglers wird im Abschnitt 7.2 eingegangen. Die Simulation eines Fuzzy-Reglers ohne Autotuning (Abschnitt 7.1.5) lässt seine Eigenschaften feststellen, die im Abschnitt 7.1.6 mit konventionellen PI-, PD- und PID-Reglertypen verglichen werden.

7.1.5 Simulation eines Regelkreises mit FFF-Regler

Ein FFF-Regler lässt sich mit den Funktionsbausteinen ABS (Absolutwert, Gruppe *Numerical*), MUL und ADD (Multiplikation und Addition, Gruppe *Arithmetic*) des Programms Concept problemlos programmieren. Als Zweipunktelement wird der EFB LIMIT (Begrenzung, Gruppe *Selection*) eingesetzt (der Zweipunktregler TWOPOINT der CONT_CLC-Bibliothek ist für diesen Zweck zu kompliziert).

Programmiert man zusammen mit einem FFF-Regler auch eine Regelstrecke, so kann der Regelkreis mittels PC-Simulator auch ohne SPS-Anlage getestet werden. Eine laufende Simulation ist im **Bild 7.7** dargestellt. Der FFF-Regler ist aus fünf Neuronen aufgebaut. Die Stellgröße entsteht am Ausgang des EFB ADD_REAL als gewichtete Summe von aktiven Neuronenausgängen (ein oder zwei Neuronen zu jedem Zeitpunkt):

$$y = K_{pNG} \cdot y_{NG} + K_{pNK} \cdot y_{NK} = -5 \cdot 0,37 + (-1,5) \cdot 0,63 = -2,79 .$$



Das Aufrufen der anliegenden Signale während der Simulation mit Concept V2.1 XL ist nachfolgend dokumentiert. Im animierten Zustand klickt man den entsprechenden Baustein an (Doppelklick), von dem man sich die Signale ansehen möchte. Es erscheint ein Informationsfenster, z.B. im **Bild 7.8** das Fenster des Addierers, der aus einzelnen Neuronenausgängen die Stellgröße des Gesamtreglers bildet.

Mit dem Menübefehl *Advanced* ruft man das Fenster für die Grafiken auf. Danach erscheint das Fenster *Advanced Monitor on ...* mit den Namen des entsprechenden Bausteins (**Bild 7.9**). In diesem Fenster sind die Herkunft, die aktuellen Werte sowie der Datentyp der Eingänge und des Ausgangs zu erkennen.

Um ein Signal anzuzeigen, markiert man den entsprechenden Ein-/Ausgang und wählt dann *Graphics* an. Es erscheint das Trendfenster mit dem entsprechend gekennzeichneten Ein-/Ausgang (**Bild 7.10**).

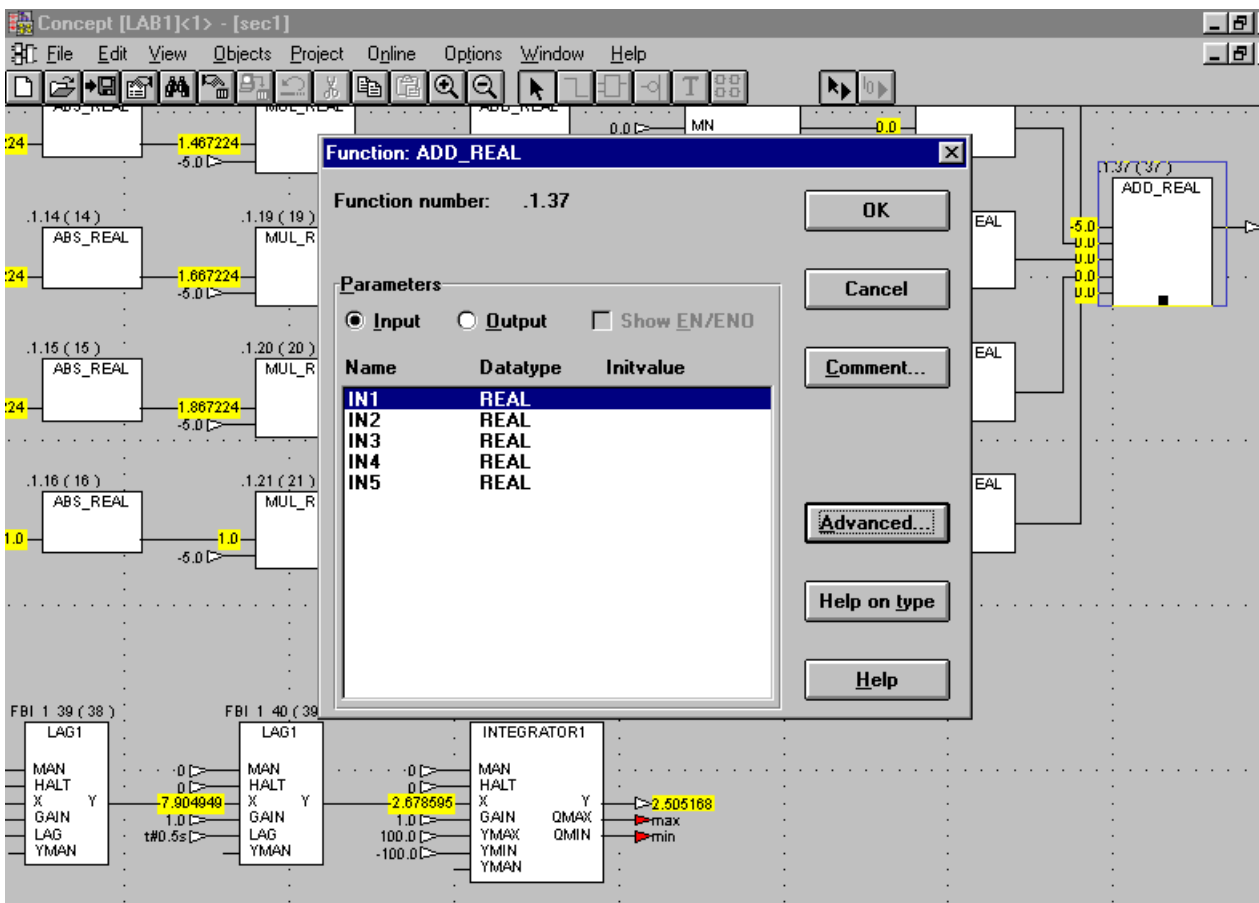
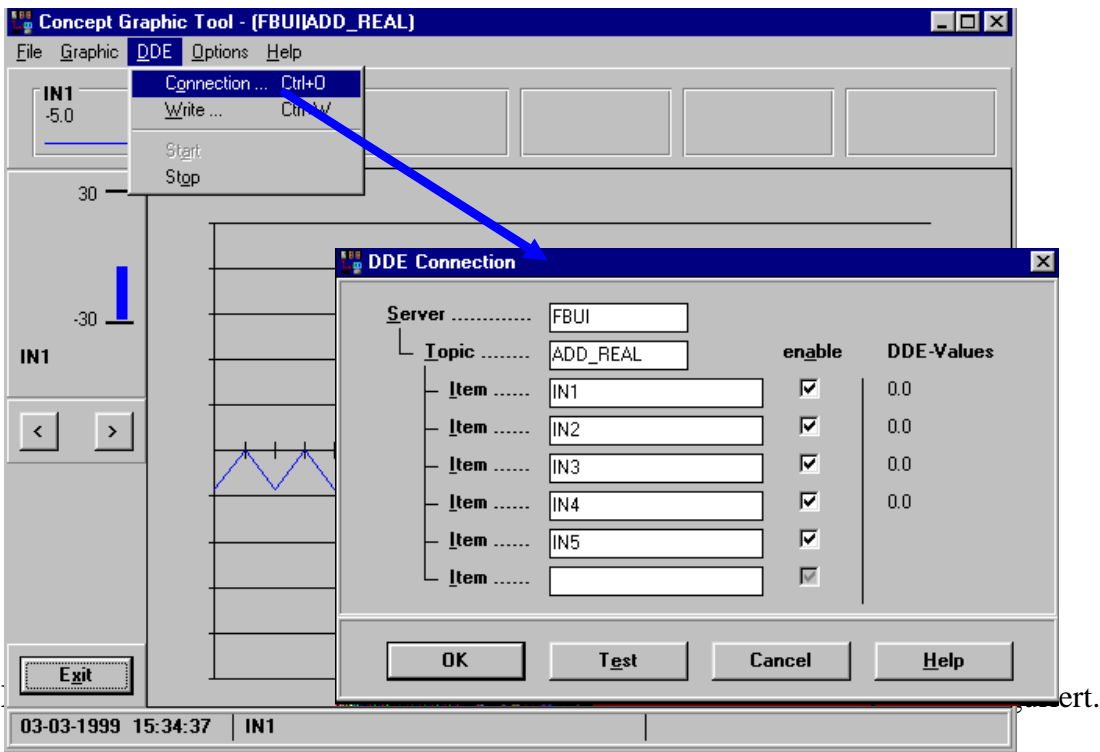


Bild 7.8 Informationsfenster des aufgerufenen Bausteins ADD_REAL. Zu sehen sind die Eingänge des Bausteins, die als fünf Neuronenausgängen addiert werden.



In einem Trendfenster können bis zu sechs Signale angezeigt werden. Man wählt *DDE / Connection* (**Bild 7.11**) und trägt danach die Ein- (IN) oder Ausgänge (OUT) ein, deren Signale abgebildet werden sollen. Nach der Eingabe von Signale ist nur noch die Skalierung zu definieren (**Bild 7.12**).

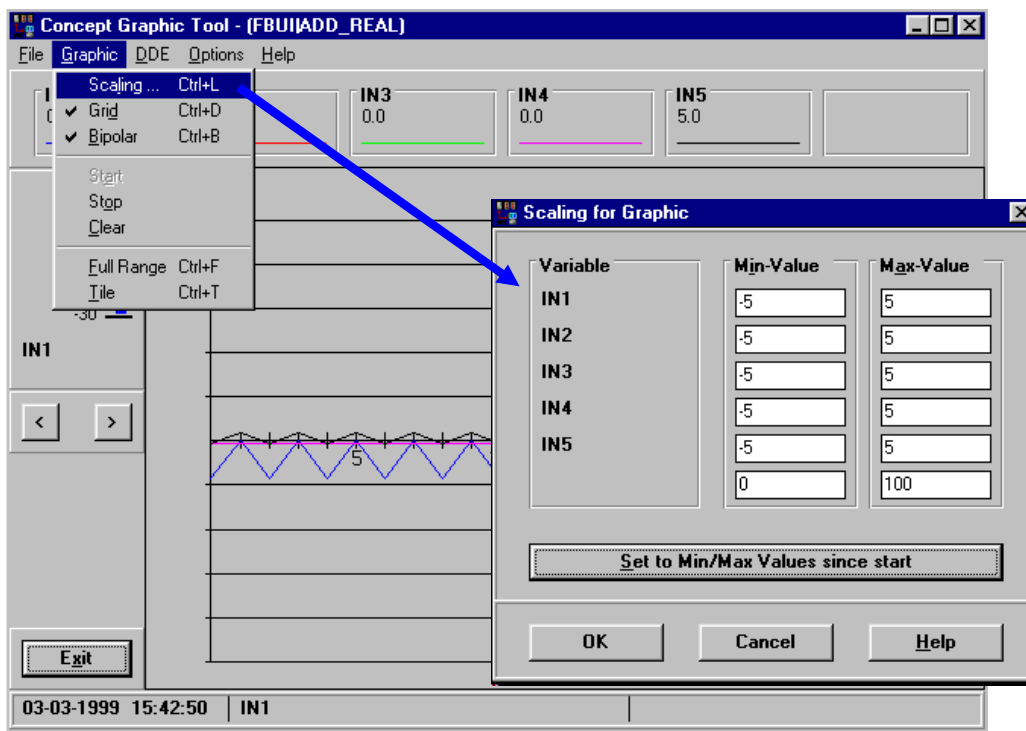


Bild 7.12 Trendfenster: Die Skalierung für alle fünf angezeigten Signale wird konfiguriert.

Die Neuronenausgänge sind im **Bild 7.13** mit folgender Belegung angezeigt:

IN1 = NG-Neuron IN3 = NU-Neuron IN4 = PK-Neuron
 IN2 = NK-Neuron IN5 = PG-Neuron

Noch ein Trendfenster wird für Sprungantworten des Regelkreises erstellt. Eine Simulation mit Dauerschwingung ist im **Bild 7.14** gezeigt.

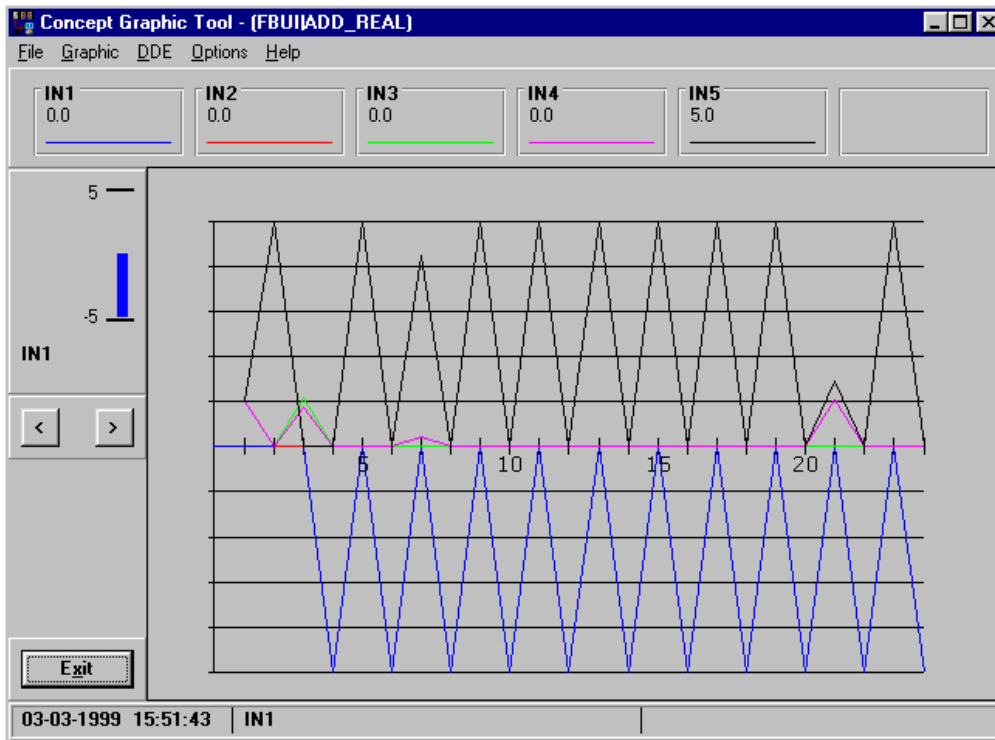
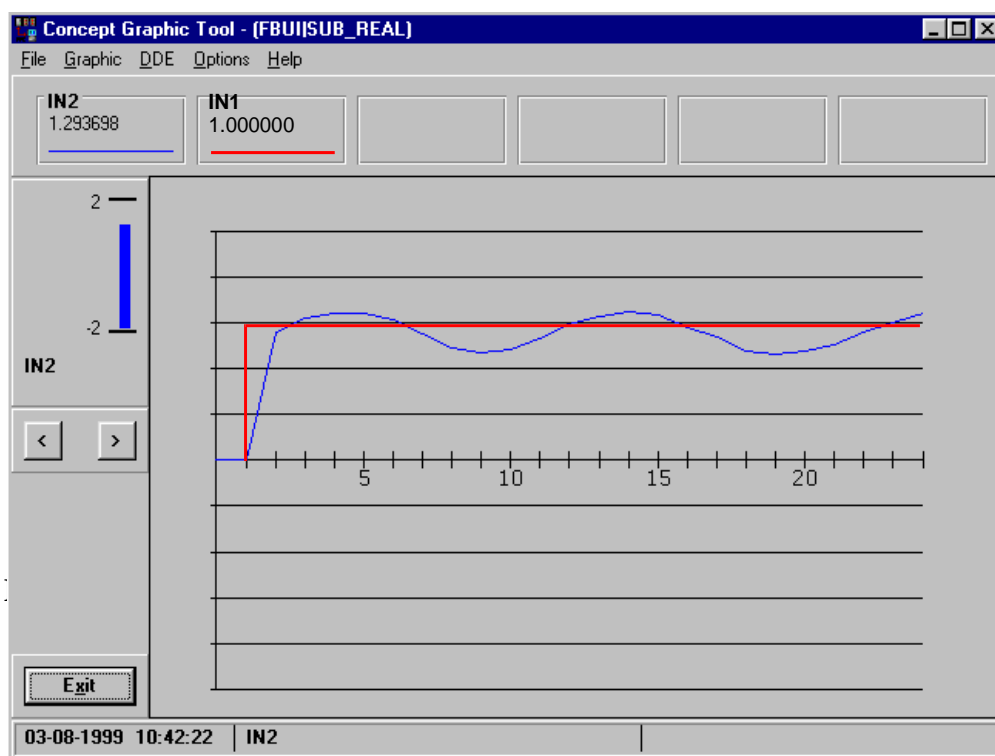


Bild 7.13 Trendfenster: Anzeige von fünf Neuronenausgängen



reises

7.1.6 Wirkung von Neuronen

Wenn sich die aktuelle Regeldifferenz des Regelkreises ändert, werden die Neuronen des FFF-Reglers nacheinander aktiviert. Die Reihenfolge, in der die Neuronen in Betrieb gesetzt werden, kann man beispielsweise im **Bild 7.15** erkennen, wo die Sprungantworten des Regelkreises und die Sprungantworten einzelner Neuronen gezeigt sind. Die Ausgänge einzelner Neuronen sind durch LIMIT-Bausteine begrenzt:

$$y_{\max \text{NK}} = y_{\max \text{PK}} = 0,2 \quad y_{\max \text{NG}} = y_{\max \text{PG}} = 0,1 \quad y_{\max \text{NU}} = 0,5 \quad y_{\min} = 0$$

Die Regelgüte lässt sich entweder über die Begrenzungen y_{\max} oder über die Proportionalbeiwerte K_p einzelner Neuronen. Die Wirkung von Proportionalbeiwerten ist im **Bild 7.16** mit einem optimalen Wert $K_{p\text{NU}} = 2,6$ (Kurve 2) demonstriert. Die Abweichung hiervon (Kurve 1 mit $K_{p\text{NU}} = 0,8$) verschlechtert die Regelgüte und führt zu einer bleibenden Regeldifferenz. Überschwingweite und Ausregelzeit ändern sich nicht. Also kann man mit Hilfe von $K_{s\text{NU}}$ die Sprungantworten nach oben und nach unten verschieben.

Durch die Einstellung der y_{\max} -Werte wird das Regelkreisverhalten auf andere Weise beeinflusst (**Bild 7.17**). Die Kurve 1 zeigt größere Überschwingweite, da nur die Neuronen NG und PG wirken. Werden auch andere Neuronen für die Regelung einbezogen (Kurve 2), kommt eine langsame Sprungantwort zustande. Das NU-Neuron bleibt in beiden Fällen ohne Einwirkung.

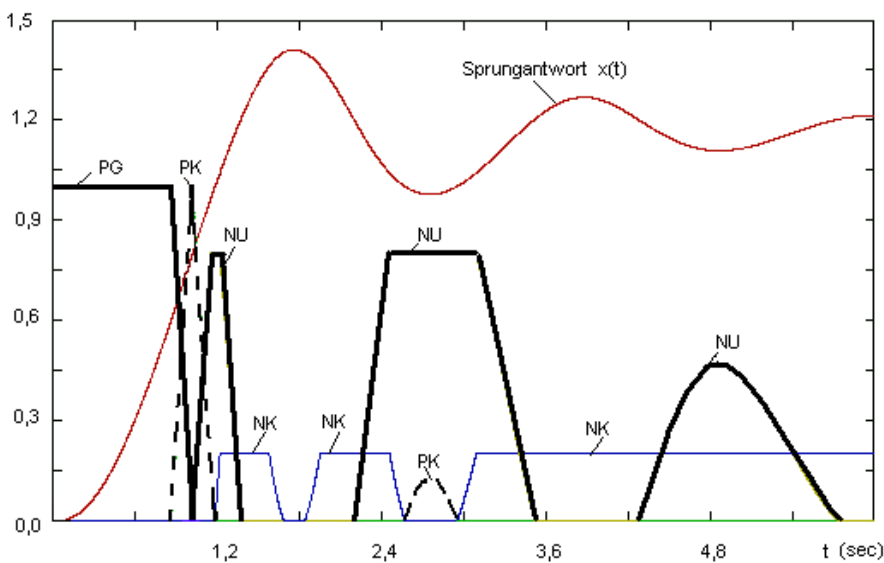


Bild 7.15 Sprungantwort des Regelkreises mit FFF-Regler und die Reaktionen einzelner Neuronen beim Führungsverhalten mit Eingangssprung $w = 1$

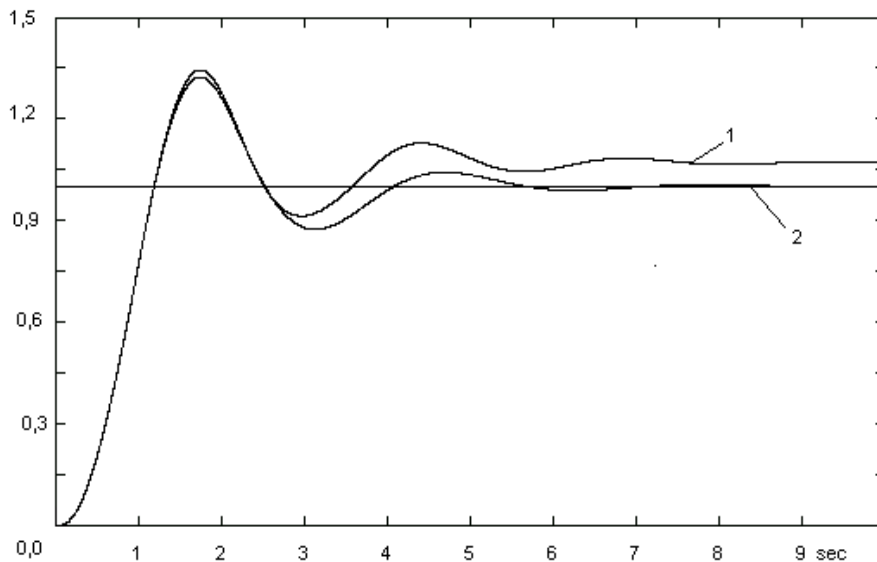


Bild 7.16 Wirkung von Proportionalbeiwerten:
 Kurve 1 mit $K_{pNU} = 0,8$ und Kurve 2 mit $K_{pNU} = 2,6$

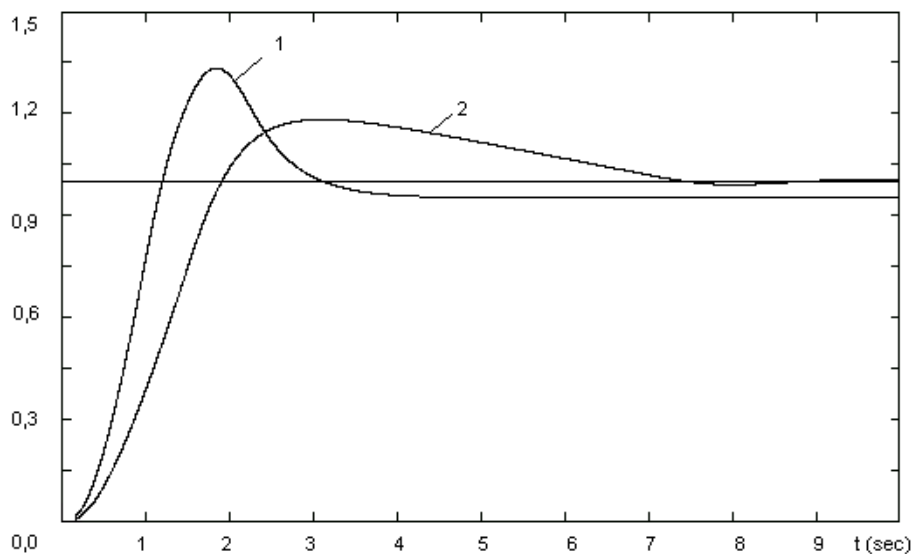


Bild 7.17 Wirkung von Begrenzungen einzelner Neuronen auf Kreisverhalten:
 Kurve 1: $y_{\max NU} = 0$ $y_{\max NK_PK} = 0$ $y_{\max NG_PG} = 1$
 Kurve 2: $y_{\max NU} = 0$ $y_{\max NK_PK} = 0,1$ $y_{\max NG_PG} = 0,5$

Die nächsten zwei Bilder befassen sich mit dem Vergleich von FFF-Reglern mit konventionellen Reglertypen. Im **Bild 7.18** sind die Sprungantworten eines Regelkreises mit dem nach dem symmetrischen Optimum eingestellten PID-Reglers ($K_{PR} = 3,1$, $T_n = 2$ s und $T_v = 0,1$ s) und mit dem FFF-Reglers, der nach dem Betragsoptimum eingestellt ist, zusammengefasst. Zwar unterscheidet sich die

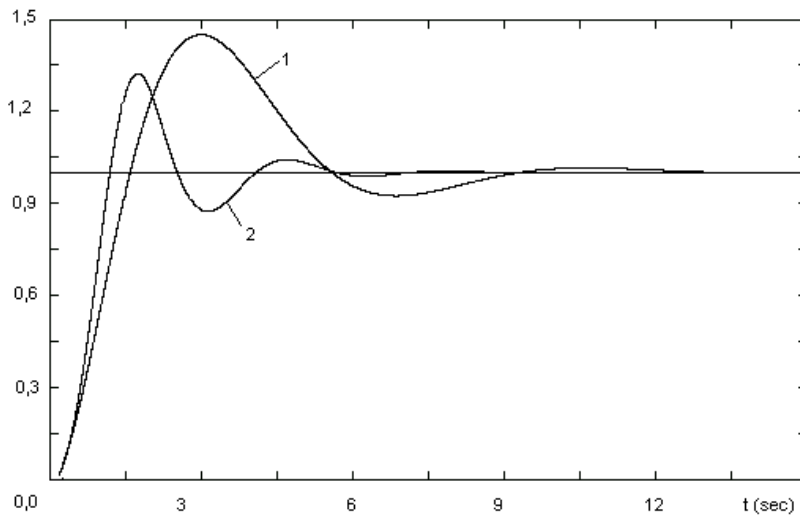


Bild 7.18 Vergleich: PID-Regler (Kurve 1, symmetrisches Optimum) und FFF-Regler (Kurve 2, Betragsoptimum)

Sprungantwort des FFF-Reglers von der beim Betragsoptimum gewohnten Form, seine Regelgüte (Überschwingweite und Ausregelzeit) ist aber wesentlich besser als die des konventionellen PID-Reglers.

Man kann die P-Elemente von Neuronen durch PD-Elemente ersetzen. Damit kommt ein FFF-PD-Regler zustande. Im **Bild 7.19** ist ein solcher FFF-Regler

$$K_{pNG} = K_{pPG} = 9 \quad K_{pNK} = K_{pPK} = 8 \quad K_{pNU} = 15 \quad T_v = 0,5 \text{ s} \quad T_R = 0,01 \text{ s} \quad y_{\max NU} = 0$$

mit konventionellem PD-T1-Regler ($K_{PR} = 15,6$, $T_v = 0,5 \text{ s}$, $T_R = 0,01 \text{ s}$) verglichen und weist dabei bessere Regelgüte auf.

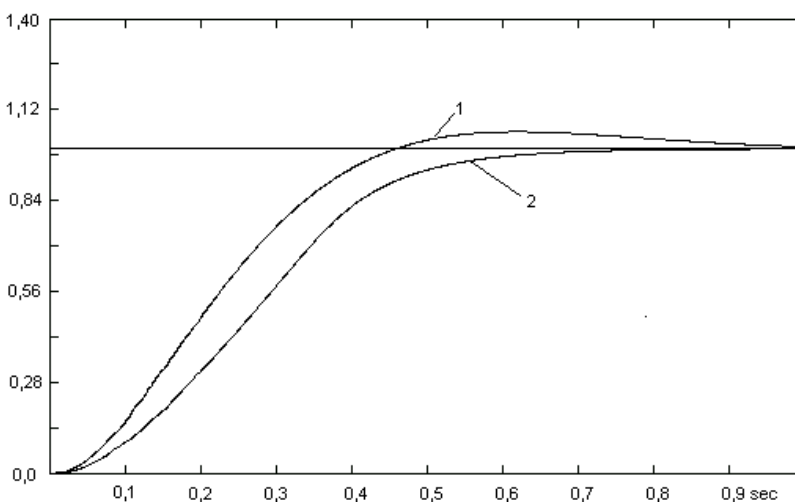


Bild 7.19 Vergleich: PD-T1-Regler (Kurve 1) und FFF-PD-Regler (Kurve 2)