

Gleitende Genauigkeit durch Fuzzy-Logik Teil 4

In den unscharfen Übergängen liegt der Gewinn

Die programmtechnische Realisierung der Fuzzy-Simulations-Software und die ausführliche Darstellung einer Beispielregelung mit Optimierung beschreibt der vorliegende, abschließende Teil dieser Artikelserie.

Programmtechnische Realisierung der Fuzzy-Software

Das Programm läuft auf jedem IBM-kompatiblen PC mit VGA-Graphikkarte. Es ist keine Fuzzy-Hardware notwendig, da die Fuzzifizierung und Defuzzifizierung durch die Software realisiert ist. Um die Simulation wirklichkeitsnah darstellen zu können, sollte ein 386DX-Prozessor im Rechner vorhanden sein.

Die Konzeption des Programms ist so ausgelegt, daß der Anwender weder über Programmierkenntnisse noch über eine große Erfahrung auf dem Gebiet der EDV verfügen muß.

Durch das integrierte Online-Hilfesystem ist es dem Benutzer möglich, zu fast allen Menüpunkten einen entsprechenden Hilfetext aufzurufen. Weiterhin wird beim Eingeben von Zahlen ein kontextbezogener Bereich vorgegeben, innerhalb dessen der Eingabewert liegen muß. Auch werden Menüpunkte deaktiviert, wenn deren Ausführung - bezogen auf die jeweilige Situation - nicht sinnvoll ist.

Nach dem Aufruf des Programms erscheint ein Startmenü, in dem der Anwender sich für eins der drei Simulationsmodelle entscheiden kann. Ist die Wahl getroffen, wird das Hauptmenü des jeweiligen Modells aufgerufen.

Beschreibung der Fuzzy-Software anhand einer Beispiel-Regelung

Zur Verdeutlichung der Funktionsweise des Fuzzy-Simulationsprogramms wird im folgenden eine einfache Regelung unter Verwendung des Kugel-Modells beschrieben.

Die zu entwickelnde Regelung soll den Winkel der Wippe so stellen, daß die Kugel von ihrer Startposition aus in die Mitte der Wippe rollt und dort zum Stehen kommt. Um die Regelung möglichst einfach zu gestalten, teilen wir jede linguistische Va-

riable in nur drei Zugehörigkeitsfunktionen auf:

erste Eingangsvariable:
Geschwindigkeit v {negativ klein, null, positiv klein}

zweite Eingangsvariable:
Position x {negativ klein, null, positiv klein}

Ausgangsvariable:
Winkel der Wippe α {negativ klein, null, positiv klein}.

Der Zusatz „klein“ ist auf die Möglichkeit zurückzuführen, daß bis zu elf Zugehörigkeitsfunktionen mit vordefinierten Namen (...ZR, PK, PG,...) pro linguistischer Variable gewählt werden können. In der Beispiel-Regelung differenzieren wir aber nur zwischen den Bereichen negativer, positiver oder Null-Position bzw. -Geschwindigkeit. Daher steht der Zusatz „klein“ in diesem Fall auch für die maxi-

male Position bzw. Geschwindigkeit.

Es werden nur UND-Regeln benutzt sowie Dreiecke als Zugehörigkeitsfunktionen, die symmetrisch auf den Grundbereich verteilt sind. Aus der Betrachtung von charakteristischen Situationen im Ablauf der Kugelbewegung und -position stellen wir Reaktionen für die Wippenstellung auf, d. h., wir verfolgen einen heuristischen Ansatz zur Formulierung der Regeln:

Liegt die Kugel auf der rechten Wippenhälfte (positive Position; Geschwindigkeit $v = 0$), sollte die Wippe links unten geneigt sein (Ausgangs-Zugehörigkeitsfunktion: Winkel $\alpha = PK$ [positiv klein]), um die Kugel zur Mitte rollen zu lassen. Formal läßt sich diese Regel schreiben als:

WENN $v = ZR$
UND $x = PK$
DANN $\alpha = PK$ (Regel 1)

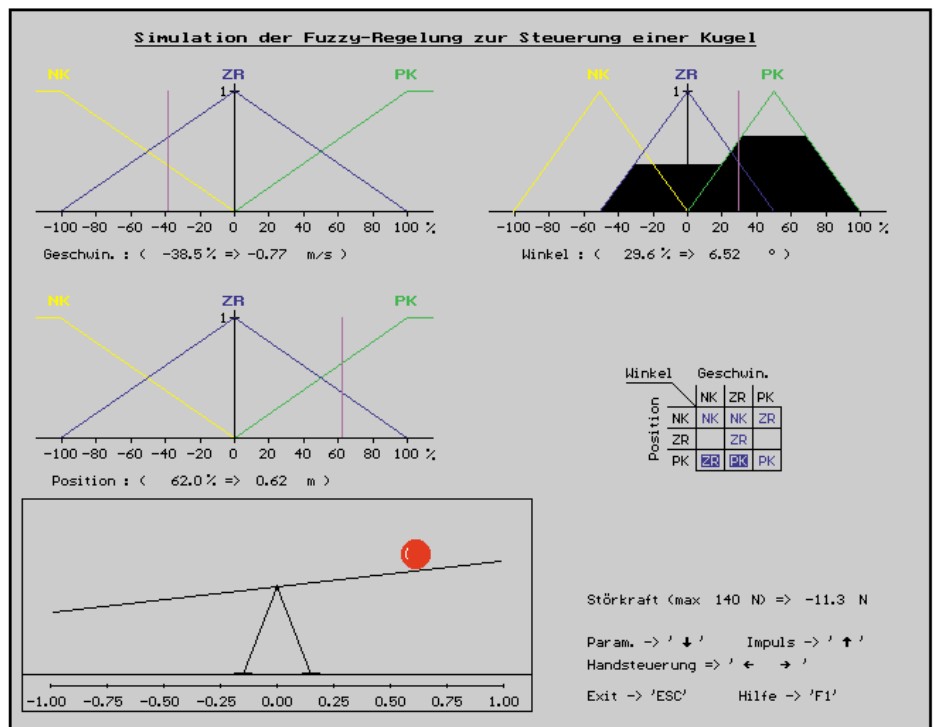


Bild 1: Bildschirmausgabe einer Fuzzy-Regelung (Kugel-Simulation)

Von den 9 möglichen Kombinationen der Eingangsvariablen ziehen wir insgesamt 7 zur Formulierung von Regeln heran:

- WENN $v = ZR$
- UND $x = ZR$
- DANN $\alpha = ZR$ (Regel 2)
- WENN $v = ZR$
- UND $x = NK$
- DANN $\alpha = NK$ (Regel 3)
- WENN $v = PK$
- UND $x = NK$
- DANN $\alpha = ZR$ (Regel 4)
- WENN $v = PK$
- UND $x = PK$
- DANN $\alpha = PK$ (Regel 5)
- WENN $v = NK$
- UND $x = NK$
- DANN $\alpha = NK$ (Regel 6)
- WENN $v = NK$
- UND $x = PK$
- DANN $\alpha = ZR$ (Regel 7)

Übersichtlicher zeigt die zugehörige Regeltabelle den Sachverhalt an, wie sie in der Abbildung 1 zu finden ist.

Diese Graphik erhält der Benutzer, wenn er nach dem Programmstart der Fuzzy-Software das Kugel-Modell auswählt und danach unter dem Menüpunkt „Erstellen einer neuen Regelung“ einen Regelungs-namen angibt. Die Zahl der Zugehörigkeitsfunktionen für die Eingangsvariablen wird im Menüpunkt „Erstellen bzw. Verändern von Fuzzy-Regeln“ auf jeweils 3 festgelegt. Graphisch können jetzt die zu den oben aufgeführten Geschwindigkeits-/Positions-Zugehörigkeitsfunktionen gehörenden Ausgangs-Zugehörigkeitsfunktionen in die Regeltabelle eingetragen werden. Anschließend erstellt der Benutzer durch den Aufruf des Menüpunktes „Automatisches Erstellen von Zugehörigkeitsfunktionen“ ohne weitere Angaben alle Zugehörigkeitsfunktionen.

Danach kann bereits der Menüpunkt „Simulation der Fuzzy-Regelung“ aufgerufen werden, denn die physikalischen Modellparameter und Grundbereiche der linguistischen Variablen besitzen sinnvolle Vorgabewerte, auf die auch hier zurückgegriffen wurde. Die bewegte graphische Bildschirmdarstellung der Simulation vermittelt ihm einen Eindruck von dem Ablauf der Regelung. Durch den Aufruf des Menüpunktes ‚Simulations-Datenaufzeichnung‘ erhält er einen Überblick über den zeitlichen Verlauf der Regelung.

In der Abbildung 2 fällt die große Überschreitung des Zielpunktes und das starke Schwingungsverhalten auf. Um diese Effekte zu minimieren oder gar zu beheben, bietet das Programm Hilfsmittel für eine effizientere Regelungsgestaltung an, wie sie im nächsten Abschnitt beschrieben werden.

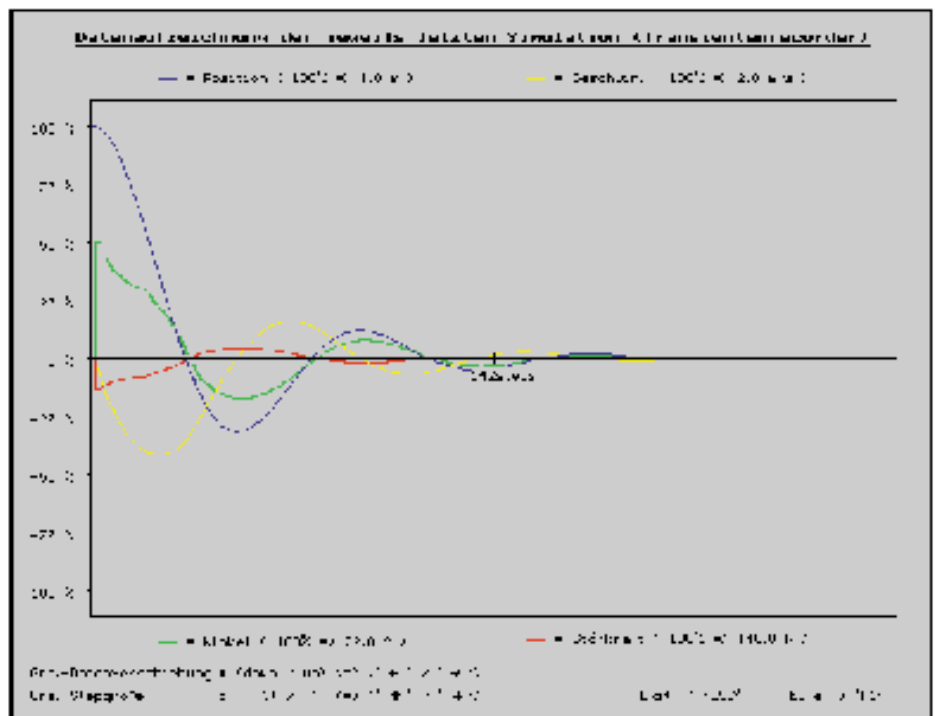


Bild 2: Datenaufzeichnung zur Kugel-Simulation (Regelsatz I)

Optimierung der Beispiel-Fuzzy-Regelung

Zur Verbesserung des Regelverhaltens bietet das Programm - neben den Möglichkeiten der Abspeicherung und des Nachladens von Regelungsdateien zum direkten Vergleich - einfache Methoden, die Zahl der Eingangs- und Ausgangs-Zugehörigkeitsfunktionen zu erhöhen und zu verkleinern, bereits vorhandene Regeln zu modifizieren oder neue Regeln hinzufügen; genauso wie das Ändern der Form einer jeden Zugehörigkeitsfunktion, das Zuweisen von Werten für den Grundbereich der Variablen und der physikalischen Modellparameter.

Greifen wir das Beispiel des letzten Abschnitts auf, gilt es, in Erfahrung zu bringen, warum die Kugel so weit über den Zielpunkt hinausläuft und beim Einlauf so stark schwingt. Wählt der Anwender den Menüpunkt „Fuzzy-Regelungskontrolle“, kann er sich für charakteristische Situationen das Verhalten der Regelung vor Augen führen.

Um herauszufinden, warum die Kugel nicht abgebremst wird, wenn sie sich dem Nullpunkt nähert, bewegt der Anwender - im Menüpunkt „Fuzzy-Regelungskontrolle“ - die Kugel mit Hilfe der Tastatur, z. B. leicht nach rechts vom Nullpunkt und gibt ihr, wiederum mit Hilfe der Tastatur, eine hohe negative Geschwindigkeit. Man erkennt, daß keine Gegensteuerung greift, obwohl die Geschwindigkeit hoch ist und sich die Kugel in der Nähe des Nullpunktes

befindet. Erst wenn die Kugel eine negative Position erreicht hat, wird ein Winkel eingestellt, der dem Kugellauf entgegenwirkt.

Wir schließen aus diesem Sachverhalt, daß die Regelung im Bereich um die Zielposition unvollständig formuliert wurde. Um diesen sensitiven Aspekt zu berücksichtigen, führen wir zwei weitere Regeln ein:

- WENN $v = PK$
- UND $x = ZR$
- DANN $\alpha = PK$ (Regel 8)
- WENN $v = NK$
- UND $x = ZR$
- DANN $\alpha = NK$ (Regel 9)

Damit sind nun in unserem einfachen Beispiel alle Eingangsvariablen-Kombinationen abgedeckt (Abbildung 3).

		Winkel		
		NK	ZR	PK
Position	NK	NK	NK	ZR
	ZR	NK	ZR	PK
	PK	ZR	PK	PK

Bild 3: Regeltabelle zum Regelsatz II

Die Auswirkung dieser Erweiterung sehen wir in Abbildung 4. Bereits vor dem Erreichen des Nullpunktes wird die Kugel abgebremst, da jetzt - im Gegensatz zur ersten Regelung - durch das Vorhandensein der

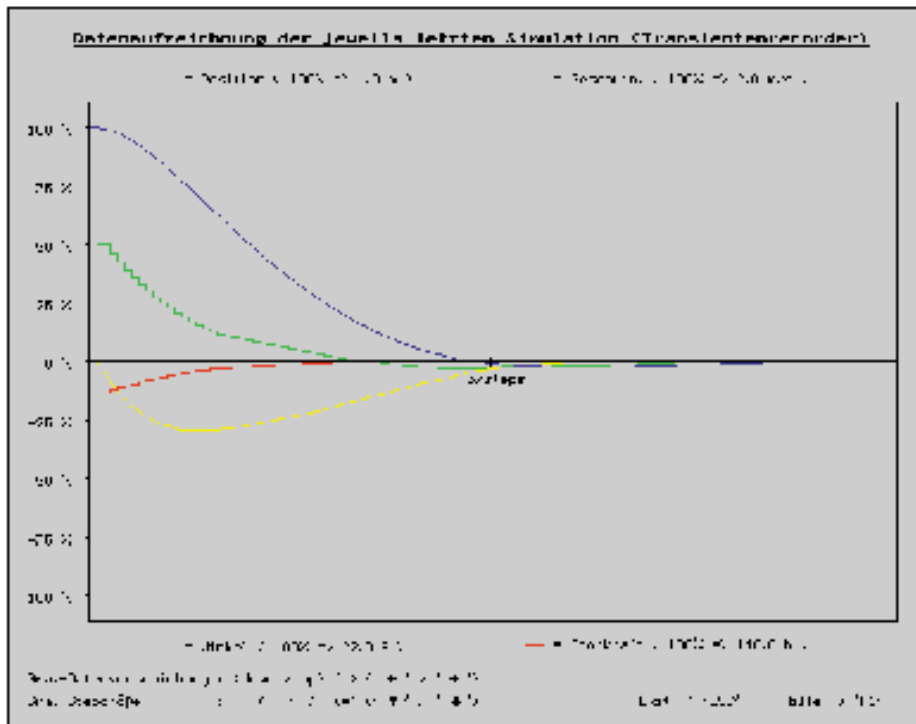


Bild 4: Datenaufzeichnung zur Kugel-Simulation (Regelsatz II)

Regel 9 die zuvor geschilderte Situation zu einem völlig anderen Resultat führt. Während im ersten Fall der Flächenschwerpunkt nur mit den Zugehörigkeitsfunktionen PK und ZR gebildet wurde (woraus sogar noch eine positive Winkelstellung resultierte), dominiert nunmehr die Ausgangs-Zugehörigkeitsfunktion NK. Dies hat zur Folge, daß sich bei der Schwerpunktbildung ein negativer Wert für den Winkel ergibt und somit die Kugel schon vor dem Erreichen der Nullpunkts abgebremst wird.

Zum einen fehlt jetzt das starke Überschwingen und zum anderen erreicht die Kugel nach nur ca. einem Drittel der Zeit bereits ihre Sollposition - insgesamt gesehen, stellt sich ein großer qualitativer Sprung durch die Optimierung des Regelsatzes heraus.

Neben der skizzierten Vorgehensweise gibt es weitere Möglichkeiten, Fuzzy-Regelungen zu optimieren, indem Lage und Form der Zugehörigkeitsfunktionen besser an physikalische Sachverhalte angepaßt werden sowie der Überprüfung, ob der Grundbereich der Variablen korrekt erfaßt wurde.

Tips zur Optimierung von Fuzzy-Regelungen

Die Beschränkung auf wenige Zugehörigkeitsfunktionen, sowohl bei den Eingangsvariablen als auch bei der Ausgangsvariable, führt zu einer wenig differenzierten Betrachtung der Eigenschaften und somit zu einer unvollständiger Beschreibung des Systems. Die Aufspaltung auf

sehr viele Zugehörigkeitsfunktionen läßt das Regelsystem kaum noch überschaubar anwachsen, wodurch die inneren Zusammenhänge überdeckt werden und Optimierungen nur noch schwer durchführbar sind. Bei einer praktischen Umsetzung darf auch nicht die erheblich gesteigerte Rechenzeit zur Auswertung der Eingangsgrößen übersehen werden, die mit der Ausweitung der Anzahl der Zugehörigkeitsfunktionen und der damit einhergehenden signifikanten Erhöhung der Anzahl der Regeln verbunden ist.

Bei der Erstellung der Regelbasis ist insbesondere darauf zu achten, daß möglichst alle im System denkbaren Verknüpfungen in Form von Regeln erfaßt werden. Wird dies außer acht gelassen, so kann es zu unbefriedigenden Regelungsverläufen kommen. Ein Hauptgrund hierfür ist, daß in Situationen, wo der Einsatz von Regeln nicht notwendig erscheint, andere Regeln mit sehr kleinen Erfülltheitsgraden einen starken Einfluß auf den Ausgangswert erlangen. Diese Regeln sind aber aufgrund ihrer geringen Erfüllung nicht für diese Situationen bestimmt und führen daher zum Fehlverhalten. Befindet sich z. B. bei der Mobil-Simulation das „Auto“ in der Nähe des anzufahrenden Punktes, mit einer Geschwindigkeit, die es genau auf den Punkt rollen läßt, ist auf den ersten Blick das Setzen einer Regel nicht notwendig. Doch gerade in diesem Fall ist es wichtig, eine Regel zu definieren, die besagt, daß kein Eingriff erfolgen soll. Unterbleibt dies, kommt es zu dem zuvor beschriebenen Fehlverhalten. Im Fuzzy-Simulationsprogramm kann diese Thematik anhand zwei-

er Beispiel-Datensätze nachvollzogen werden.

Die Breite der Zugehörigkeitsfunktion „ZR“ in der linguistischen Ausgangsvariable, bezogen auf die im Simulationsprogramm verwendeten Modelle, ist ein signifikantes Maß für die Dynamik des Fuzzy-Ausgabewertes. Ist zum Beispiel die Zugehörigkeitsfunktion „ZR“ der linguistischen Ausgangsvariable sehr breit (relativ großer Abstand des Anfangs- und Endwertes vom Nullpunkt), so verlagert sich der Ausgabewert bei der Schwerpunktbildung mehr zum Nullpunkt, d. h. die Ausgabegröße wird verkleinert und gedämpft.

Bei einem Großteil der Fuzzy-Regelungen ist es zweckmäßig, im Grundbereich nur einen bestimmten Ausschnitt der Eingangsvariable zu betrachten. So wird man beispielsweise bei der Mobil-Simulation für die Position des Fahrzeuges einen Grundbereich definieren, der sich auf eine kleine, um den Zielpunkt liegende, Fahrstrecke bezieht. Der Vorteil, der sich daraus ergibt, liegt in der besseren Betrachtung des für die Regelung wichtigen Bereiches. Dies setzt aber voraus, daß Zugehörigkeitsfunktionen und Regeln definiert werden, die festlegen, in welcher Weise das System gesteuert wird, wenn die Eingangsvariable außerhalb des Grundbereiches liegt. Im Fuzzy-Simulationsprogramm sind derartige Zugehörigkeitsfunktionen daran zu erkennen, daß sie in der graphischen Darstellung mit einer Wertigkeit von eins über +/- 100% des Grundbereiches hinausgehen. Betrachtet man z. B. in der Abbildung 1 die Zugehörigkeitsfunktionen „NK“ der linguistischen Variable für die Geschwindigkeit, so ist diese so lange zu eins erfüllt, bis der Wert der Geschwindigkeitsvariable größer als -100% ist.

Ausblick

Das hier vorgestellte Fuzzy-Simulationsprogramm ist eine Software-Realisation für Fuzzy-Einsteiger. In folgenden Artikeln stellen wir eine Hardware-Realisierung eines Fuzzy-Reglers vor. Dieser kann mit der dazugehörigen Windows-Software von einem PC aus konfiguriert werden, arbeitet aber vollkommen eigenständig und ist daher sehr gut für eigene Anwendungen geeignet. **ELV**

Autoren: Tilo Könnecke, Detlef Puchert, Fachhochschule Braunschweig/Wolfenbüttel