9 Realexperimente mit Fuzzy

Bislang haben wir die Funktionstüchtigkeit der Fuzzyregelung nur an simulierten Objekten überprüft. Wird sich die Fuzzyregelung aber auch im Realexperiment bewähren? Mit dieser Frage wollen wir uns in diesem Kapitel eingehend beschäftigen. Dazu wird das universelle Experimentierprogramm *RealExp* eingesetzt. Universell ist es in mehrerlei Hinsicht: Zum Einen kann es mit verschiedenen Interfaces arbeiten; zum Anderen ist es nicht auf ein spezielles Regelungsexperiment festgelegt. Sie können damit zum Beispiel eine Laufkatze, ein invertiertes Pendel oder eine Spielzeuglokomotive, aber auch weitere nach eigenen Vorstellungen konzipierte Systeme regeln.

In diesem Kapitel werden wir zunächst eine kurze Einweisung für den Umgang mit *RealExp* geben. An zwei Beispielen werden wir ausführlich zeigen, wie man mit den verschiedenen Interfaces eine Lokomotive bzw. ein invertiertes Pendel regeln kann. Anschließend geben wir noch Hinweise, wie eigene Konzepte mit *RealExp* umgesetzt werden können.

Wer die Realexperimente nicht durchführen kann oder möchte, kann dieses Kapitel überspringen; statt dessen wird er sich anhand der Videos ipend3.avi, loko1.avi und loko2.avi auf der Fuzzy-CD ein Bild davon machen, ob die Fuzzyregelung sich nun im Realexperiment bewährt.

9.1 Das Programm *RealExp*

RealExp wird bei der Installation des *Fuzzy*-Programms automatisch in Ihr *Fuzzy*-Verzeichnis kopiert. Sie können es aus dem *Fuzzy*-Programm über das Experiment-Menü oder direkt aus dem Start-Menü von Windows 95 starten[1]. Je nach Wahl der Experimentparameter werden die Ressourcen des PCs stark belastet (s.u.). Deshalb sollte Ihr PC mindestens einen Pentium II - Prozessor (oder einen dazu kompatiblen) besitzen, wenn Sie mit *RealExp* arbeiten. *RealExp* unterstützt die in der Tabelle 1 angegebenen Interfaces. Bewusst wurden hier Treiber von höchst unterschiedliche Interfacetypen implementiert.

RealExp kann zwei Messgrößen erfassen und damit einen Motor regeln. Als Sensoren bieten sich z. B. Wendelpotis und LDRs, aber auch solche Bewegungsmesswandler an, die Positionen in Spannungssignale umsetzen. Die Regelung des Motors erfolgt über eine sogenannte Impulsbreitensteuerung. *RealExp* ordnet dazu jedem Stellwert eine bestimmte Zeitspanne zu und versorgt den

Interface	Preis (ungefähr)	A/D-Wandler	Bezug
Sensor-CASSY und Power-CASSY	1500 Euro	12 Bit, -10 V bis +10V	Leybold didactic GmbH Leyboldstraße 1 50354 Hürth
Multiface 7x	ab 200 Euro	8 Bit, 0 bis 2,55 V	Knobloch GmbH Weedgasse 14a, 55234 Erbes-Büdesheim
MAX-Selbstbau	60 Euro	11-12 Bit, 0 bis 5 V	s. Anhang
LTC-Selbstbau	40 Euro	12 Bit, 0 bis 5 V	s. Anhang

Tab. 9.1 Interfaces für *RealExp*

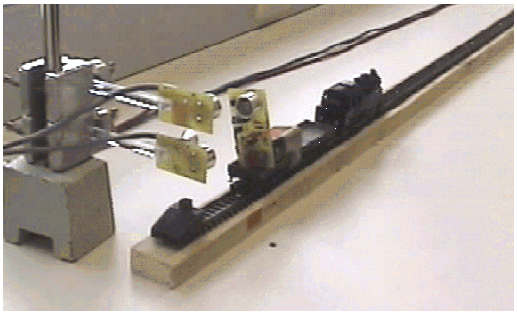
[1] *RealExp* läuft nicht unter Windows 3.x und auch nicht unter Windows NT.

Motor mit entsprechend langen Pulsen konstanter Spannung. Wie hoch dieser Spannungswert sein soll, lässt sich in einem weiten Bereich von ungefähr 5 V bis 20 V direkt am Interface einstellen.[1] Auf diese Weise können höchst unterschiedliche Motoren betrieben werden. Weiter unten werden wir sehen, dass die Impulsbreitensteuerung uns weitere Vorteile bietet.

Wie das Programm *RealExp* bedient wird, wollen wir nun an einem einfachen Experiment kennenlernen.

9.2 Das Lokomotivenexperiment

Ziel dieses Experiments ist es, eine Spielzeuglokomotive möglichst sanft und punktgenau anhalten zu lassen; das zugehörige Simulationsexperiment finden Sie in der Aufgabe 7 des vorletzten Kapitels. Dass ein derartiges Verhalten bei Zügen und U-Bahnen von Vorteil ist, leuchtet unmittelbar ein.[2] Das Lokomotivenexperiment eignet sich besonders gut zur Einführung: Das Fuzzyprojekt ist einfach, Aufbau des Experiments und Kalibrierung des Interfaces sind unkompliziert und erfordern kaum eine ganze Stunde an Zeitaufwand. Sie können außerdem auf Geräte des Lehrmittel- und Spielzeughandels zurückgreifen.



9.1 Eine Fuzzy-geregelte Lokomotive

Hier benutzen wir:

- eine Gleichstromlokomotive mit einem Wagen
- ein ungefähr 100 cm langes Gleisstück inklusive Anschlussgleis
- den Bewegungsmesswandler der Firma Conatex zur Erfassung der Lokomotivenposition
- das Interface Multiface 74 DA der Firma Knobloch (s. o.)
- eine stabilisierte Gleichspannungsquelle mit 18 V, ca. 1 A.

Alternativ können Sie das Multiface durch andere Interfaces ersetzen. Statt des Messwandlers der Firma Conatex können auch andere Bewegungsmesswandler benutzt werden, die die Position durch ein analoges Spannungssignal darstellen; insbesondere wäre hier auch der Einsatz eines leichtgängigen Wendelpotis denkbar.

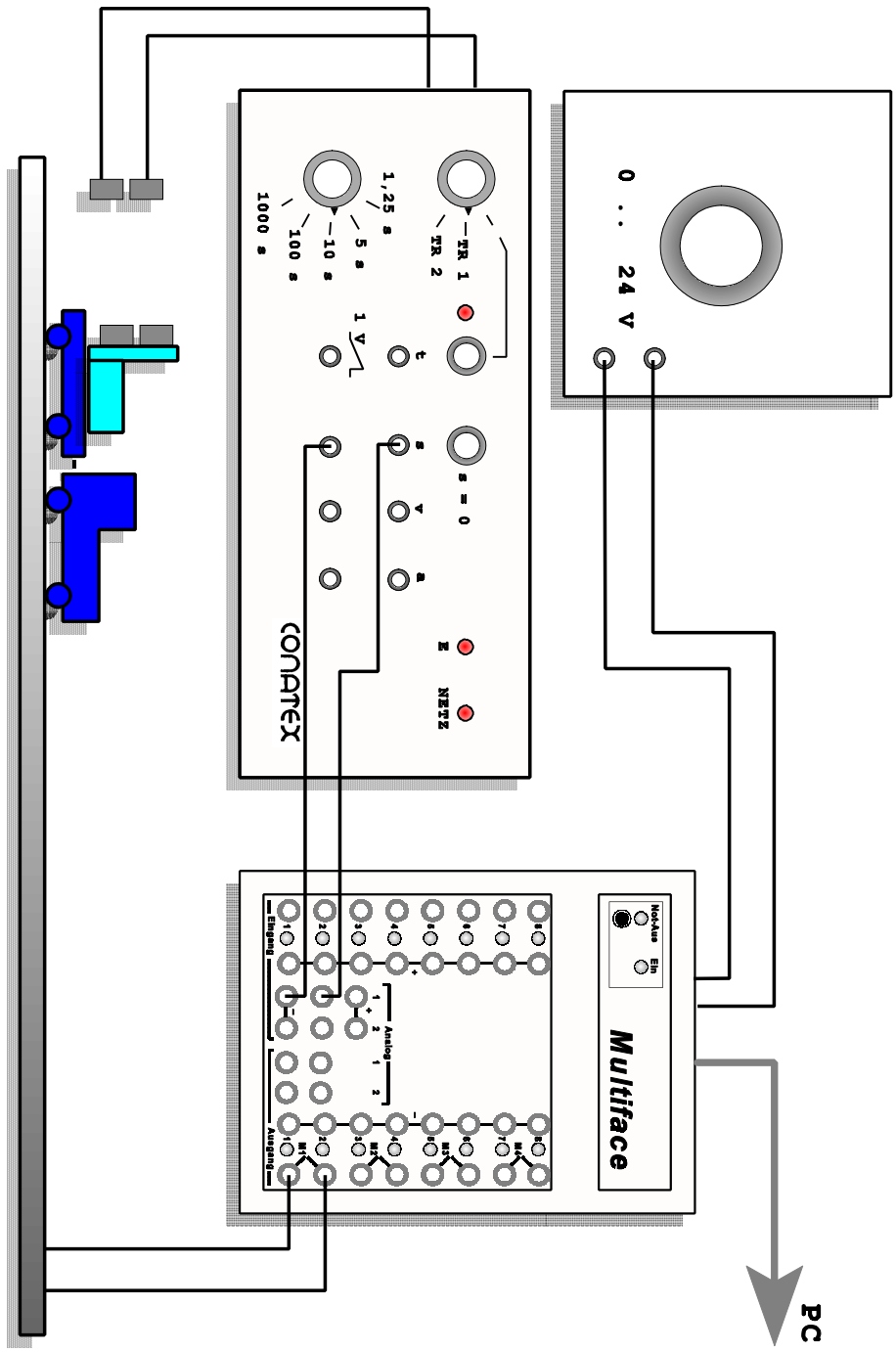
Auf der *Fuzzy*-CD finden Sie ein Video mit der in Abb. 9.1 dargestellten Anordnung (loko1.avi). Zusätzlich ist in dem Video loko2.avi zu sehen, wie das gleiche Experiment mit dem CASSY-Interface und einer Lego-Eisenbahn durchgeführt wird.

Bauen Sie den Versuch nach Abb. 9.2 auf. Achten Sie darauf, dass die Eisenbahn den Conatex-Konverter trägt; das Interface kann zwar auch ohne diesen Konverter Entfernungen messen, allerdings sind Reichweite und auch Genauigkeit geringer. Schalten Sie dann den Rechner, die elektrische Quelle, das Interface und den Messwandler ein und starten Sie das Programm *RealExp*. Die Lokomotive sollte dabei noch nicht auf dem Gleisstück stehen, weil sie bei der Initialisierung der Centronics-Schnittstelle einen kurzen Strompuls erhalten könnte. Die Spannung der elektrischen Quelle sollte bei dem Knobloch-Interface ca. 2 V über der Nennspannung der Lokomotive liegen.

---

[1] Beim Power-CASSY-Interface wird ein Stromstärkewert (maximal 1 A) softwaremäßig vorgegeben.

[2] In der japanischen Stadt Sendai gibt es eine kommerzielle Anwendung dieses Experiments: eine Fuzzy-geregelte U-Bahn-Anlage.



9.2 Versuchsaufbau zur fuzzy-geregelten Lokomotive

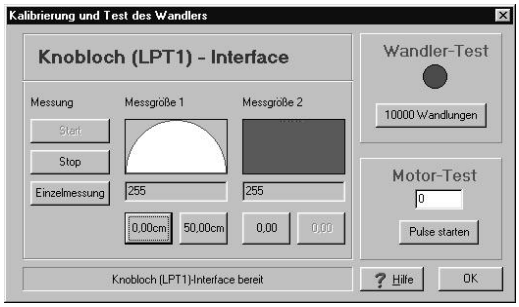
Parameter	Voreinstellung bei loko.exp
benutztes Interface	Knobloch-Interface auf LPT1
Kalibrierung der Sensoren	Kanal für die Nullmarke und Anzahl der Kanäle pro Einheit
Zuordnung der Messeingänge zu den Eingangsgrößen der Fuzzyregelung	Als Messgröße 1 dient die Position des Wagens; sie wird der Eingangsgröße “Position” der Fuzzyregelung zugewiesen. Die zeitliche Veränderung dieser Messgröße (die Ableitung nach der Zeit) wird der Fuzzyvariablen “Geschwindigkeit” zugeordnet. Sie wird berechnet aus dem Differenzenquotienten $$\frac{s_2 - s_1}{t_2 - t_1}.$$ Dabei sind s_1 und s_2 die Positionen der Lokomotive zu den Zeitpunkten t_1 und t_2, und $\Delta t = t_2 - t_1$ ist ein Vielfaches der Zykluszeit T: $\Delta t = n \cdot T$. Hier ist n = 1.
Dauer T der Fuzzyzyklen	50 ms, d. h. jede 50 ms wird aus den Messwerten mittels der Fuzzyregelung ein neuer Stellwert bestimmt und über die Schnittstelle dem Motor zugeführt
benutzte Fuzzyprojekte	loko4.fuz

Tab. 9.2 Einige Parameter des Lokomotiven-Experiments

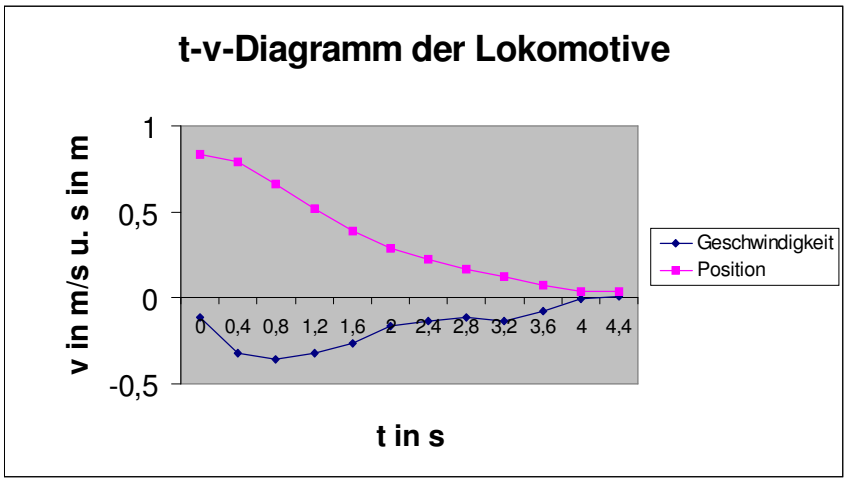
Nun öffnen Sie die Experimente-Datei loko.exp. In dieser Datei sind die für das Experiment wesentlichen Parameter gespeichert. Für das Lokomotivenexperiment sind die wichtigsten in der Tabelle 2 zusammengefasst.

Alle diese Parameter können bei Bedarf den eigenen Vorstellungen und Bedürfnissen angepasst werden. Dies geschieht mit dem Datei- oder mit dem Bearbeiten-Menü. In jedem Fall müssen Sie vor dem Start des Realexperiments aber eine Kalibrierung der Sensoren vornehmen. Dazu klicken Sie im Bearbeiten-Menü die Option Kalibrierung und Test an. Es erscheint das Fenster aus Abb. 9.3. Dort betätigen Sie zunächst den Start-Knopf. *RealExp* ruft nun fortlaufend Messdaten am Interface ab und stellt sie graphisch dar. Hier ist nur eine Messgröße, nämlich die Position der Lokomotive, vorhanden.

Stellen Sie nun die Lokomotive auf die Nullmarke (Zielposition) und betätigen Sie den s=0-Knopf am Conatex-Messwandler; die Zielposition wird dann durch U = 0 V repräsentiert. Diese wird von *RealExp* registriert, wenn man den 0,00cm-Knopf anklickt. Nun stellen Sie die Lokomotive 50 cm vom Ziel entfernt auf und klicken den 50cm-Knopf. Damit ist die Kalibrierung abgeschlossen; Sie können die Test-Messungen beenden.



9.3 Kalibrieren und Testen des Interfaces



9.4 Ergebnisse eines Lokomotiven-Experiments

Wenn Sie sich nicht sicher sind, ob Ihre Rechner-Interface-Kombination schnell genug arbeitet, führen Sie einen Wandler-Test durch. Für zuverlässiges Experimentieren sollten die 10 000 Wandlungen in wenigen Sekunden erfolgen.

Nun sollten Sie noch überprüfen, ob die Motoren korrekt angeschlossen sind. Geben Sie dazu verschiedene Werte zwischen -1,0 und +1,0 für die relativen Pulsbreiten im Motortestfeld ein. Je nach Vorzeichen sollte die Lokomotive vorwärts bzw. rückwärts fahren. Hier sollte sie bei einem positiven Motorwert auf das Ziel zufahren.

Viele Motoren setzen sich aufgrund der Reibung erst bei Pulsbreiten von 0,2 oder 0,3 in Bewegung. Sie sollten diese kritische Pulsbreite experimentell bestimmen und als sogenannten Motoroffset im Parameter-Formular eintragen. Schließen Sie dazu das Fenster zum Kalibrieren und Testen und klicken Sie im Bearbeiten-Menü die Option Parameter an. Tragen Sie Ihren Offsetwert ein und bestätigen Sie die Eingabe mit der OK-Taste. *RealExp* wird nun automatisch die Kraft so korrigieren, dass die Reibung kompensiert wird.

Alle neu vorgenommenen Einstellungen lassen sich mitsamt den restlichen, bereits vorgegebenen Einstellungen dauerhaft in einer neuen Experiment-Datei speichern. Wählen Sie dazu im Datei-Menü die Option Speichern unter... Benutzen Sie als Dateinamen möglichst nicht den Namen der Vorgabe-Datei loko.exp, da diese sonst überschrieben würde.

Nun sind alle Vorbereitungen abgeschlossen, unser erstes Realexperiment zur Fuzzyregelung kann beginnen. Setzen Sie die Lokomotive in einiger Entfernung vom Ziel auf das Gleis und betätigen Sie den Start-Knopf im Hauptfenster. Die Lokomotive startet “mit Volldampf”; etwa 50 cm vor dem Ziel wird sie allmählich langsamer und hält schließlich fast genau am Ziel an. Wir müssen dabei Abweichungen von 1 - 2 cm tollerieren, weil die hier benutzte Kombination aus Bewegungsmesswandler und Interface lediglich eine Auflösung von ungefähr 1 cm bietet. Eine wesentlich präzisere Regelung ließe sich z.B. mit einem 12-Bit-Interface erzielen (vgl. Tabelle 1).

9.3 Optimieren des Lokomotivenexperiments

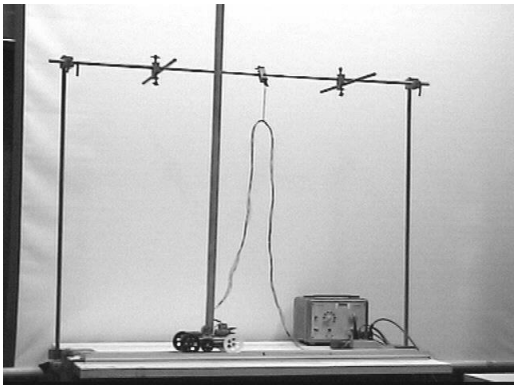
Vielleicht wird Ihre Lokomotive nicht auf Anhieb zufriedenstellen anhalten. In diesem Fall sollten Sie das benutzte Fuzzyprojekt mit dem *Fuzzy*-Programm modifizieren. Gehen Sie dabei in derselben Art und Weise vor wie bei den Simulationsversuchen. Mit der Option Fuzzyprojekt öffnen im Datei-Menü

können Sie Ihr neues Fuzzyprojekt in das Experiment einbinden. Beachten Sie jedoch, dass Ihr Fuzzyprojekt als Eingangsgröße A die Position (in cm) und als Eingangsgröße B die Geschwindigkeit (in cm/s) besitzen muss. Außerdem müssen die Werte der Ausgangsgröße stets im Intervall zwischen -10 und +10 liegen.

Beim Optimieren der Fuzzyregelung kann die Protokoll-Funktion von *RealExp* sehr vorteilhaft sein: Wenn man diese Funktion aktiviert, werden die Mess- und Stellgrößen der ersten 1000 Fuzzyzyklen nach dem Start protokolliert. Diese Daten lassen sich obendrein zur weiteren Auswertung nach EXCEL exportieren. Auf diese Weise wurde auch das Diagramm in Abb. 9.4 hergestellt. Es zeigt deutlich, dass die benutzte Fuzzyregelung noch verbessert werden sollte: Je gleichmäßiger die Geschwindigkeit abnimmt, desto sanfter ist nämlich der Bremsvorgang.

9.4 Das invertierte Pendel im Realexperiment

Im Vergleich zum Lokomotivenexperiment ist das Balancieren eines invertierten Pendels durch einen Wagen eine erheblich anspruchsvollere Aufgabe – und das gleich in mehrfacher Hinsicht: Der Bau des Pendelwagens (inklusive Sensorik) ist aufwendiger; es werden zwei Fuzzyprojekte benötigt, das eine für die Pendelregelung, das andere für die Positionsregelung des Wagens; die Fuzzyzyklen sollten nur 10 bis 20 Millisekunden betragen, der Rechner muss also recht schnell sein. Will man auch die benötigte Elektronik für A/D-Wandler und Motoransteuerung selbst bauen, sollte man ein paar Arbeitstage einplanen. Immerhin erhält man dann aber auch ein fuzzy-geregeltes invertiertes Pendel für weniger als 100 Euro; das ist etwa ein Hundertstel dessen, was man für ein kommerzielles System ausgeben müsste!



9.5 Invertiertes Pendel nach Ihlefeldt im Test

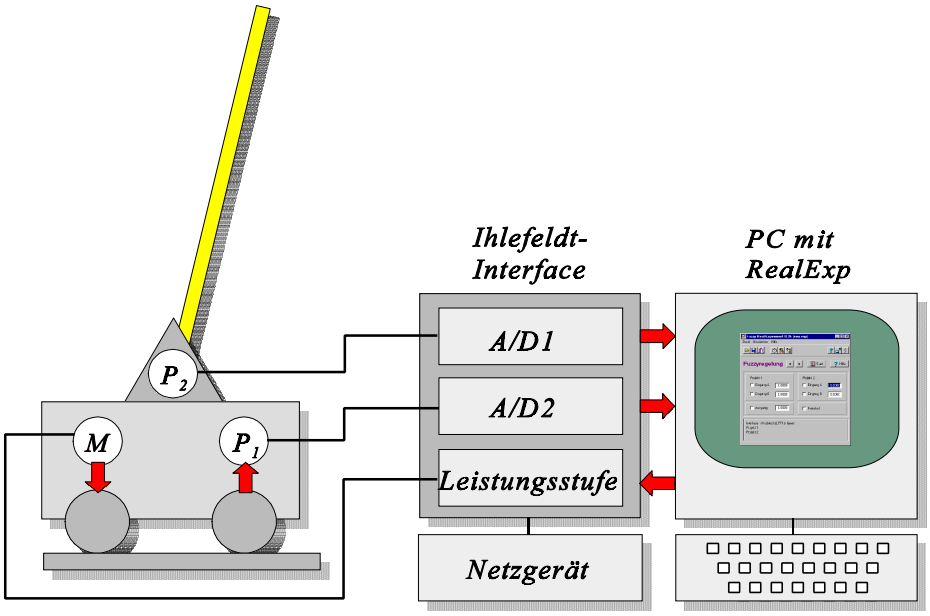
Einen ersten Eindruck von der Funktionsweise des invertierten Pendels können Sie erhalten, wenn Sie sich das Video ipend3.avi anschauen. Der hier benutzte Aufbau ist weiter unten beschrieben; als Fuzzyprojekte wurden dieselben Projekte benutzt wie bei der Simulation mit dem bereits aus Kapitel 7 bekannten Programm *KombiFuz*. Deutlich erkennt man, wie die Fuzzyregelung das Pendel balanciert; wenn der Wagen sich dabei einem Ende der Bahn nähert, wird die Positionsregelung aktiv. So bewegt sich der Wagen fortwährend zwischen diesen Enden hin und her. Kleinere Störungen – etwa durch Antippen der Pendelstange mit dem Finger – werden genauso wie in der Simulation mühelos ausgeglichen.

Unser Wagen (vgl. Abb. 9.6) besteht aus einem Profilblech; an diesem sind befestigt:

- ein Motor mit Getriebe, welcher eine der beiden Achsen antreibt;
- ein Potentiometer P_1, welches über ein Getriebe mit der anderen Achse verbunden ist; es dient zur Positionsbestimmung;
- eine Achse mit dem invertierten Pendel und einem Potentiometer P_2 als Winkelsensor

Um Schlupf zu vermeiden, fährt der Wagen auf Zahnrädern, welche auf einer Zahnstange laufen. Weitere Einzelheiten kann der interessierte Leser dem Anhang entnehmen.

Die beiden Potentiometer können über die Interfaces mit Spannung versorgt werden. Die Spannung zwischen dem Mittelabgriff und Masse wird von den Wandlern ausgewertet und dem Programm *RealExp* als Rohwerte für die Eingangsgrößen “Winkel” und “Position” zur Verfügung gestellt. In



9.6 Invertiertes Pendel nach Ihlefeldt

jedem Fuzzyzyklus werden nun daraus

- der Winkel des Pendels in ° und die Position des Wagens in cm berechnet;
- die Winkelgeschwindigkeit des Pendels und die Geschwindigkeit des Wagens berechnet;
- ein Stellwert aus den Eingangsgrößen Winkel und Winkelgeschwindigkeit sowie Position und Geschwindigkeit des Wagens berechnet;
- eine entsprechend langer Puls an die Leistungsstufe des Interfaces abgegeben.

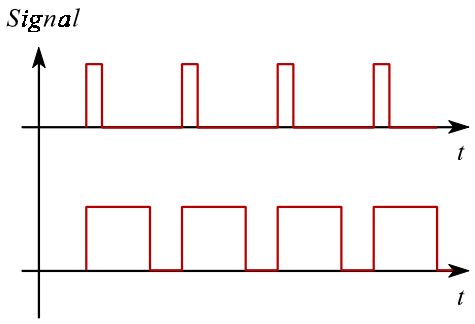
Die Leistungsstufe steuert den Motor über unterschiedlich kurze Strompulse; ihre Länge liegt in der Größenordnung von einigen Millisekunden. Diese Impulsbreitensteuerung bietet gleich mehrere Vorteile: Zum einen ermöglicht sie, Motoren mit unterschiedlichen Nennspannungen zu betreiben; dies hatten wir oben schon erwähnt. Zum anderen lässt sich eine stabile Regelung des Wagens gar nicht erzielen, wenn man den Wagen durch eine zum Stellwert proportionale Spannung steuert. Woran dies liegt, lässt sich mit einigen wenigen physikalischen Betrachtungen klar machen.

Betrachten wir dazu eine typische Situation am invertierten Pendel: Unsere Fuzzyregelung hat es gerade geschafft, durch eine Beschleunigung nach rechts, das Pendel vor dem Umkippen zu bewahren. Sobald das Pendel nun wieder eine nahezu vertikale Stellung erhält, senkt unsere Fuzzyregelung die Ausgangsgröße "Kraft". Bei geringer Reibung bewegt sich der Wagen dann auf Grund seiner Trägheit nahezu gleichförmig weiter. Somit wirkt auf das Pendel kaum ein Drehmoment, es bleibt in seiner günstigen Lage.

Das Experiment zeigt jedoch ein ganz anderes Verhalten, wenn die Ausgangsgröße der Fuzzyregelung als Spannung an den Motor gelegt wird: In dem Augenblick, in dem die Spannung gesenkt wird, bremst der Wagen ab, und das Pendel fällt wieder nach rechts. Sofort versucht die Regelung dies auszugleichen, indem sie eine erneute Beschleunigung nach rechts veranlasst. Auf diese Weise kommt es zu immer hektischeren Bewegungen, die sich in kürzester Zeit aufschaukeln und dazu führen, dass das Pendel bereits nach einigen wenigen Sekunden am Boden liegt.

Dieses merkwürdige Verhalten ist darauf zurückzuführen, dass bei Elektromotoren das Drehmoment und damit auch die Kraft auf den Wagen eben nicht proportional zur Spannung ist. Zum einen ist das Drehmoment bei kleinen Spannungen unverhältnismäßig gering; dies erschwert natürlich ein sanftes Regeln. Zum andern hängt das Drehmoment bei fester Spannung auch von der Belastung des Motors ab.

Es gibt allerdings spezielle Regelsysteme, die in der Lage sind, bei einem Motor ein zur Regelspannung proportionales Drehmoment zu erzeugen. Leider sind diese Systeme sehr teuer. Deswegen wird hier auf andere Weise eine Drehmomentregelung realisiert. Bekanntlich ist die Kraft auf einen stromführenden Leiter im Magnetfeld proportional zur Stromstärke. Eine Drehmomentregelung für den Motor ist also nichts anderes als eine Stromstärkeregelung. Eine solche Regelung lässt sich gerade durch die oben bereits erwähnte Impulsbreitensteuerung zumindest näherungsweise erreichen (Abb 9.7).



9.7 Das obere Signal führt zu einem geringeren Drehmoment als das untere.

Für unseren Aufbau benutzen wir ein Selbstbau-Interface[1], welches eine wichtige, zusätzliche Eigenschaft besitzt: Wenn das Signal vom PC Null ist, wird der Ausgang dieses Interfaces hochohmig. Auf diese Weise wird verhindert, dass in diesen Null-Signal-Phasen der Motor durch Selbstinduktion gebremst wird. Die Wirkung einer solchen Induktionsbremse lässt sich auch auffassen als eine zusätzliche Reibungskraft, die um so größer ist, je kleiner der von der Fuzzyregelung berechnete Stellwert für die Kraft ist. Bei kleinen oder verschwindenden Kräften sollte sich der Wagen aber (nahezu) mit konstanter Geschwindigkeit fortbewegen; die Induktionsbremse würde das Pendel also immer wieder in Bewegungsrichtung des Wagens kippen lassen; mit anderen Worten: der Wagen würde dem Pendel stets hinterherlaufen müssen.

Kommen wir nun zur Durchführung des Experiments. Dazu gehen wir ähnlich wie bei der Regelung der Lokomotive vor: Nach dem Anschließen aller Geräte starten wir das Programm *RealExp* und laden die Experimente-Datei invpend.exp. Anschließend werden Winkel- und Positionsgeber kalibriert; dabei sollte die Markierung des Winkels Null mit großer Sorgfalt erfolgen. Nun wird der Motoroffset bestimmt und als Parameter eingegeben (s. o.). Anschließend sollten alle Angaben unter einem neuen Namen abgespeichert werden.

Setzen Sie nun den Wagen auf die Nullposition der Schiene, umgreifen Sie das Pendel an seinem oberen Ende und halten es lotrecht. Jetzt starten Sie die Fuzzyregelung mit dem Start-Knopf und lockern Sie gleichzeitig Ihren Griff ein wenig. Sollten Sie nun bemerken, dass das obere Pendelende dabei mehr zu einer Seite tendiert, ist die Nullmarke des Winkels nicht ganz korrekt eingestellt. In diesem Fall können Sie die Nullmarke online, d. h. während der laufenden Regelung mit den Knöpfen “<” und “>” verschieben (vgl. Abb. 9.8). Betätigen Sie diese Knöpfe solange, bis das obere Pendelende zwischen Ihren Fingern nicht mehr zu einer Seite tendiert.

Jetzt können Sie das Pendel behutsam loslassen. Vermutlich wird die Fuzzyregelung Ihr Pendel nicht auf Anhieb vernünftig balancieren können; seien Sie deshalb vorsichtig und stoppen Sie das Experiment rechtzeitig. Dabei werden Sie von *RealExp* unterstützt: Wenn der Winkel des Pendels 20° überschreitet, wird die Fuzzyregelung beendet und der Motor gestoppt. Diese Grenze von 20° können Sie nach eigenen Bedürfnissen auch anders festlegen; das geschieht mit Bearbeiten | Zuordnung der Messgrößen.

[1]Weitere Informationen zu diesem Interface finden Sie im Anhang.

Nun gilt es, die Experiment-Parameter, insbesondere die Fuzzyprojekte, zu optimieren. Bei dieser Aufgabe haben sich die verschiedenen Protokollfunktionen von *RealExp* als sehr hilfreich erwiesen. Sie erlauben es zum Beispiel typische Werte für die Eingänge zu ermitteln und die Fuzzyprojekte entsprechend zu modifizieren. Von großer Hilfe waren auch die zahlreichen Erfahrungen, die vor dem Experimentieren mit dem Simulationsprogramm *KombiFuz* gesammelt worden waren; auch während des Experimentierens sorgte der Vergleich von Real- und Simulationsexperiment immer wieder für neue Einsichten. Dies spiegelt sich in der Tatsache wieder, dass beide Experimente, das reale wie auch das simulierte, sich durch ein- und dieselbe Fuzzyregelung balancieren lassen!

9.8 RealExp

9.5 Eigene Fuzzyexperimente

Wenn man ein selbst konzipiertes Fuzzy-Experiment mit *RealExp* durchführen möchte, muss man zuerst eine passende Experimente-Datei erstellen. Bei diesen Dateien handelt es sich um Ini-Dateien, die mit einem gewöhnlichen Editor bearbeitet oder erstellt werden können.

Es bietet sich folgende Vorgehensweise an:

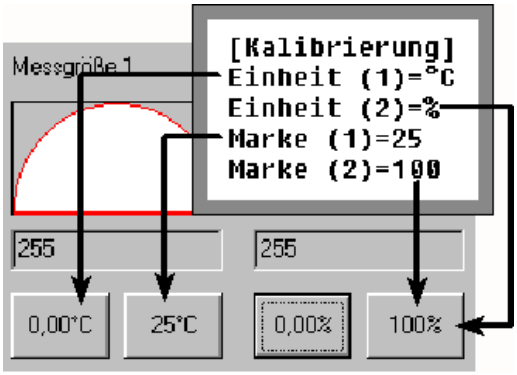
- Legen Sie mit Datei | Neu eine neue Experimente-Datei an und speichern Sie diese Datei unter einem neuen Namen ab.

- Laden Sie diese Datei in einen Editor und geben Sie in der Rubrik “Kalibrierung” die Einheiten der beiden Messgrößen an; der Eintrag bei der zweiten Messgröße kann gegebenenfalls entfallen. Geben Sie dann die Größen für die zweiten Kalibrierungsmarken an (Die ersten Marken werden automatisch immer auf Null gesetzt.). Diese Angaben werden von *RealExp* benötigt, um die Kalibrierknöpfe einzurichten (vgl. Abb. 9.9).

- Anschließend speichern Sie die Datei unter gleichem Namen ab.

- Öffnen Sie nun diese Experimente-Datei mit *RealExp* und legen Sie die restlichen Parameter fest:
  - Interface wählen
  - Fuzzyprojekt(e) wählen
  - Zuordnung der Messgrößen (vgl. Tabelle 9.3) vornehmen und ggfs. das Vorzeichen des Motoroffsets an eine Eingangsvariable koppeln
  - Kalibrierung vornehmen
  - Motoroffset bestimmen und im Parameterfenster eingeben
  - Zykluszeit in ms angeben
  - Faktor n eingeben (vgl. Tabelle 2)

- Zuletzt sollten Sie diese Angaben sichern, indem Sie Ihre Experimente-Datei speichern.



9.9 Definition der Kalibrierungsmarken

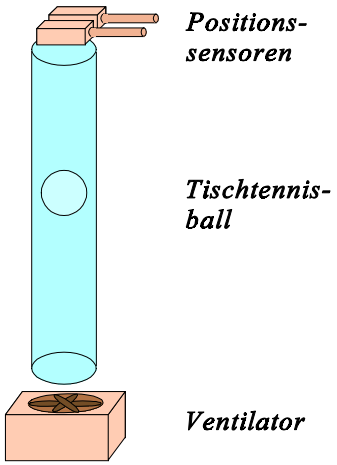
Projekt 1 Eingangsvariable A Eingangsvariable B	Messgröße 1 Ableitung der Messgröße 1
Projekt 2 Eingangsvariable A Eingangsvariable B	Messgröße 2 Ableitung der Messgröße 2

Tab. 9.3 Eine mögliche Zuordnung von Messgrößen und Fuzzyvariablen

9.6 Aufgaben

1. Bauen Sie Ihr invertiertes Pendel um zu einer Laufkatze. Benutzen Sie anfangs als Eingangsgrößen die Position und die Geschwindigkeit des Wagens. Erweitern Sie dann Ihre Regelung durch ein weiteres Fuzzyprojekt, welches als Eingangsgrößen den Winkel und die Winkelgeschwindigkeit besitzt.

2. Ein Tischtennisball schwebt automatisch in dem Luftstrom eines Föhns oder eines anderen ähnlichen Gebläses (vgl. Abb. 9.10). Erheblich schwieriger sind die Verhältnisse, wenn der Ball in einer luftdurchströmten Röhre schweben soll. Bauen Sie die Lokomotivenregelung zu einer Tischtennisballregelung um.
Hinweis: Der Bewegungsmesswandler von Conatex kann bei nicht zu großen Abständen auch ohne aufgesetzten Konverter arbeiten.



9.10 Zu Aufgabe 2