

Studienarbeit

Thema: Alternative Gelenksteuerung für Roboter

Autoren: Christian Benckendorff

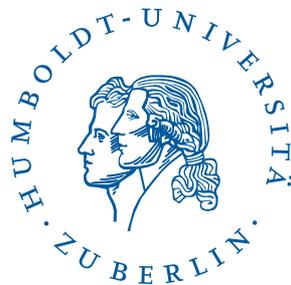
und Thomas Lobig

Betreuer: Manfred Hild

Lehrstuhl für Künstliche Intelligenz

Prof. Hans-Dieter Burkhard

April 2008



Diese Arbeit befasst sich mit der Umsetzung eines neuartigen Antriebs für Robotergelenke. Dabei wird die noch immer vorherrschende Praxis einen Roboter mit direkt gekoppelten Getriebemotoren anzusteuern verworfen. Als neuer Ansatz werden mehrere Servomotoren pro Gelenk benutzt und die Krafteinleitung wird mit Hilfe von Seilzügen und gedämpften Federn durchgeführt. Nach dieser Methode kann man das Getriebespiel verhindern, welches bei einem einzelnen Servo zu Problemen in der Ansteuerung führt. Außerdem werden die Vorteile eines elastischen Antriebs für die nächste Generation von Robotern des Labors für Neurorobotik an der Humboldt Universität zu Berlin nutzbar gemacht. Diese Studienarbeit ist eine Vorarbeit dazu.

This student research project covers the realisation of a new way to drive a robotic joint. The current standard of using directly connected servos on each joint is left aside. More than one servo is used instead and they are coupled with the joint using a kind of tackle and dampened elastic springs system. In this manner backlash can be virtually eliminated, which would normally lead to complex controlling problems. In addition to that the features of elastically propelled joints come at grasp for the next generation of robots in the Neuro Robotic Lab of the Humboldt University of Berlin. This thesis is a first step towards it.

Inhaltsverzeichnis

1 Einführung in das Thema der Arbeit	4
2 Ansatz	6
2.1 Eine mathematische Herleitung über die Eliminierung des Getriebespiels	6
2.2 Überlegungen zur Mechanik und Biomechanik	8
2.2.1 Kupplung	8
2.2.2 Einstellbare Steifigkeit	9
2.2.3 Energieeffizienz	10
2.2.4 Schwingungen im System	10
2.3 Eine Kostenfrage	11
3 Eine Auswahl verwandter Arbeiten	12
3.1 Elastisch angetriebener Roboterarm	12
3.2 Oktavio	13
4 Der Experimentalaufbau	13
5 Experimente	17
5.1 Erste Tests	17
5.1.1 Pseudocontroller ohne Winkelmesser	17
5.1.2 Winkelgeber	17
5.1.3 Federkennlinie der Gummibänder	18
5.2 Experiment 1	20
5.2.1 Controller mit vier Servos	20
5.2.2 Auswertung	21
5.3 Experiment 2	23
5.3.1 Controller mit vier Servos neuronal gesteuert	23
5.3.2 Auswertung	26
5.4 Experiment 3	29
5.4.1 Controller mit zwei Servos neuronal gesteuert	29

5.4.2	Auswertung	29
6	Abschließende Überlegungen und Ausblick	30
6.1	Berechnung der Inversen Funktion	30
6.2	Ausblick	35
7	Anhang	37
7.1	Die Elektronik des Experimentalaufbaus	37
7.2	Die Firmware des Experimentalaufbaus	40
7.3	Die PC-Software des Experimentalaufbaus	43
7.4	Arbeitsaufteilung	46
7.5	Verwendete Software	46

1 Einführung in das Thema der Arbeit

In der Robotik gibt es vielfältige Probleme die sich Wissenschaftlern bei der Arbeit mit und an Robotern stellen. Die Erkennung und interne Modellierung der Umwelt, die Planung von Aktionen müssen bewerkstelligt werden. Nicht zuletzt erweist sich aber auch die Aktuierung von Robotern als eine schwierige Aufgabe.

Für mobile Roboter gibt es im Wesentlichen zwei unterschiedliche Antriebsmethoden, wenn man die Bewegung zu Wasser und zur Luft ausklammert. Zum einen sind da die fahrenden Roboter, welche sich auf Rädern oder Ketten fortbewegen. Die Ansteuerung solcher Roboter ist allgemein eher einfach, darum waren die ersten mobilen Roboter auch solche. Der zweite Ansatz ist Roboter mit Beinen auszustatten. So genannte Laufmaschinen sind allgemein komplizierter in der Ansteuerung. Man benötigt die Möglichkeit die Gelenke in den Beinen präzise anzusteuern um dynamische Laufmuster auszuführen.

Die Natur macht es vor. Emsige Insekten auf sechs Beinen, darunter ausgezeichnete Springer, geschickte Arachniden auf acht Beinen, flinke Pferde auf vier Beinen und nicht zu vergessen der Mensch, als eines der wenigen Lebewesen, welche sich dauerhaft mit zweibeiniger Fortbewegung begnügen. Doch was in der Natur so leicht und selbstverständlich wirkt bereitet Forschern und Entwicklern Kopfzerbrechen bei der technischen Umsetzung für Roboter.

Man kann sagen dass es noch keinen Roboter gibt der zufriedenstellend wie ein Mensch laufen kann. Entweder die Maschinen sind schnell und reaktiv, dafür aber z. B. pneumatisch getrieben und damit nicht autonom, oder sie sind zwar autonom, aber laufen nicht so robust und elegant wie ein Mensch. Das Mittel des Antriebs ist eine wichtige Kernfrage. Es gibt pneumatische Aktuatoren, die für autonome Roboter eigentlich indiskutabel sind, da ein kurzlebiger Druckluftspeicher mitgenommen werden müsste oder ein sehr energiehungriger Kompressor. Es gibt experimentelle Aufbauten[2] mit Drähten aus Formgedächtnislegierung als Muskelfaser, die aber physikalisch sehr empfindlich sind und nicht elastisch wie menschliche Muskeln. Der weit verbreitete Standard ist es mit Servomotoren aller Art an die Sache heran zu gehen.

Servomotoren sind relativ einfach zu steuern, sie haben einen Winkelbereich den sie anfahren können. Dies kann man entweder durch eine direkte Winkelansteuerung unter Nutzung einer internen Steuerung des Servomotors, oder durch Steuerung des Drehmomentes.

Im Vergleich mit natürlichen Muskeln haben Servomotoren zwei gravierende systematische Nachteile. Der eine Nachteil entsteht durch das Getriebe eines Servomotors, welches ein gewisses Getriebespiel mit sich bringt und die Feinsteuerung in einem Winkelbereich von wenigen Grad erschwert (Es bleibt zwar einzuräumen dass z. B. das Harmonic Drive kein Getriebespiel besitzt, aber es ist dafür ungleich viel teurer als ein gewöhnliches Getriebe). Leider benötigt man insbesondere für zweibeinige Laufroboter eine sehr präzise Steuerung, sofern der Roboter nicht mit sehr großen Füßen ausgestattet wird.

Das zweite Problem sei an einem Beispiel erklärt. Man stelle sich einen Menschen vor der durch einen Wald joggt. Durch Unachtsamkeit bleibt der Mensch mit dem Fuß an einer Wurzel hängen. Wäre ein zweibeiniger, rein servogetriebener, Laufroboter in der Situation, so würde der Impuls vom Fuß direkt in die Getriebe gehen. Es ist sehr wahrscheinlich dass der Roboter stürzt. Die Beine sind zu starr um den Impuls abzufangen, der ganze Körper gerät aus dem Gleichgewicht. Beim Mensch ist es so, dass die Muskeln eine gewisse Elastizität besitzen, welche durch Anspannung der Muskeln auch verringert werden kann (vgl. Nachtigall [1]). Das Bein eines Menschen fängt also den Impuls ab, der in Situationen, wie der oben geschilderten, das Laufmuster stören kann. Es können natürlich noch weitere Mechanismen greifen, um ein Stürzen zu verhindern. Einen Ausfallschritt kann auch ein rein servogetriebener Laufroboter probieren. Ein elastischer Antrieb hätte einen systematischen Vorteil. Darüber hinaus gibt es weitere Vorteile, die weiter unten besprochen werden.

2 Ansatz

Der Ansatz in dieser Arbeit verfolgt eine Idee von Manfred Hild, dem Betreuer dieser Studienarbeit. Um trotz der Verwendung von Servomotoren das Getriebeispiel auszu-schalten und die Steifigkeit des Gelenkes einzustellen, wird für ein Rotationsgelenk mit einem Freiheitsgrad mehr als ein Servomotor benutzt und die Kraft über einen gefederten Seilzug eingeleitet. Prinzipiell lässt sich so das Getriebeispiel des einzelnen Motors für den resultierenden Antrieb aufheben. Zusätzlich erschließt sich mit einem elastischen Antrieb ein großartiges Potential für geschmeidigere Bewegungen, für Ener-gieersparnis und -rückgewinnung, sowie eine Verschleißreduktion.

2.1 Eine mathematische Herleitung über die Eliminierung des Getriebeispiels

Eine typische Kennlinie einer Servosteuerung hat eine Form wie in Abbildung 1.

Diese ergibt sich aus der allgemeinen Form

$$\vec{M}(\Delta\varphi) = \begin{cases} (\Delta\varphi + \vartheta) \cdot a - b & | \quad \Delta\varphi \leq -\vartheta \\ 0 & | \quad -\vartheta < \Delta\varphi < \vartheta \\ (\Delta\varphi - \vartheta) \cdot a + b & | \quad \Delta\varphi \geq \vartheta \end{cases}$$

wobei

$$\begin{aligned} \Delta\varphi \cdot a + b &\leq 0 & | & \quad \Delta\varphi \leq -\vartheta \\ \Delta\varphi \cdot a - b &\geq 0 & | & \quad \Delta\varphi \geq \vartheta \end{aligned}$$

$\Delta\varphi$ ist die gemessene Abweichung vom Zielwinkel des Servos. \vec{M} ist das anzusteuernde Drehmoment. a und b sind abhängige Konstanten, welche die Steigung und die Ver-schiebung der Geradenstücken bestimmen. a und b sind frei wählbar, solange die oben angegebene Bedingung erfüllt ist. Die Funktion ist allgemein nicht stetig (Spezialfall $\Delta\varphi \cdot a - b = 0 \quad | \quad \Delta\varphi = \vartheta$) und in jedem Fall an den Stellen $-\vartheta$ und ϑ nicht differen-zierbar. Der Wert b existiert, weil der Motor einen minimalen Anlaufstrom benötigt um in Bewegung zu kommen, daraus ergibt sich auch ein minimales Drehmoment $\pm b$ (Dazwischen existiert nur die Möglichkeit kein Drehmoment zu haben, aber keine feine Abstufung zwischen 0 und b bzw. $-b$. $[-\vartheta, \vartheta]$ ist ein Bereich, in dem der Zielwinkel als

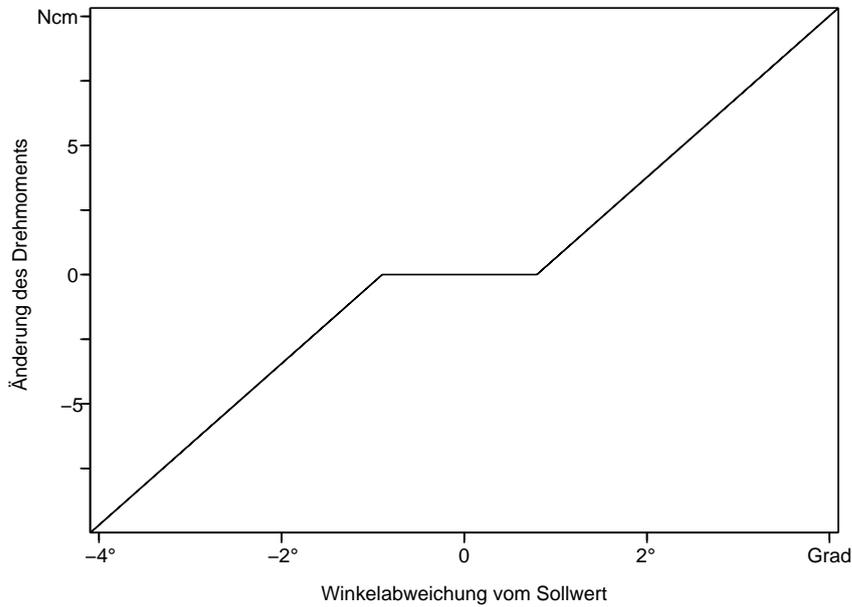


Abbildung 1: Kennlinie eines einzelnen Servos

erreicht gilt. Für ein größeres Getriebespiel muss auch dieser Bereich größer sein. Er dient dazu den Motor des Servos vor Selbstzerstörung zu bewahren. Gäbe es diesen Sicherheitsbereich nicht, würde der Motor ständig um die Nullposition herum die Richtung wechseln, dabei unnötig viel Strom verbrauchen ohne eine merkliche Bewegung zu zeigen und könnte sich dabei aufheizen und sich zerstören. Durch diesen Bereich ist jedoch die Funktion nicht umkehrbar und führt zu einem nichtlinearen Verhalten welches vor allem in der Simulation von Laufmaschinen zu Problemen führt. Ergebnisse aus der Simulation sind dann im Zweifelsfall nicht auf die Realität übertragbar. Angenommen man benutzt zwei Servomotoren die ihre Kraft auf dasselbe Gelenk einleiten. Steuert man diese so an, dass sich die Sicherheitsbereiche nicht überschneiden, so addieren sich die Kennlinien zu einer Kennlinie deren zugehörige Funktion umkehrbar ist. Nun lässt sich eine Bewegungsgleichung aus dem System ableiten, das nichtlineare

Verhalten durch Getriebespiel ist eliminiert.

Verschiebt man die Steuerfunktionen von zwei Servos so, dass der Abstand 2ε zwischen den beiden Sicherheitsbereichen entsteht, dann ergeben sich folgende Funktionen:

$$\tau_1(\Delta\varphi) = \begin{cases} (\Delta\varphi + (2\vartheta + \varepsilon)) \cdot a - b & | \quad \Delta\varphi < -(2\vartheta + \varepsilon) \\ 0 & | \quad -(2\vartheta + \varepsilon) \leq \Delta\varphi < -\varepsilon \\ (\Delta\varphi + \varepsilon) \cdot a + b & | \quad \Delta\varphi \geq -\varepsilon \end{cases}$$

$$\tau_2(\Delta\varphi) = \begin{cases} (\Delta\varphi - \varepsilon) \cdot a - b & | \quad \Delta\varphi < \varepsilon \\ 0 & | \quad \varepsilon \leq \Delta\varphi < 2\vartheta + \varepsilon \\ (\Delta\varphi - (2\vartheta + \varepsilon)) \cdot a + b & | \quad \Delta\varphi \geq 2\vartheta + \varepsilon \end{cases}$$

Aufsummiert ergibt sich:

$$\tau_{1+2}(\Delta\varphi) = \begin{cases} 2(\Delta\varphi + \vartheta) \cdot a - 2b & | \quad \Delta\varphi < -(2\vartheta + \varepsilon) \\ (\Delta\varphi - \varepsilon) \cdot a - b & | \quad -(2\vartheta + \varepsilon) \leq \Delta\varphi < -\varepsilon \\ 2\Delta\varphi \cdot a & | \quad -\varepsilon \leq \Delta\varphi < \varepsilon \\ (\Delta\varphi + \varepsilon) \cdot a + b & | \quad \varepsilon \leq \Delta\varphi < 2\vartheta + \varepsilon \\ 2(\Delta\varphi - \vartheta) \cdot a + 2b & | \quad 2\vartheta + \varepsilon \leq \Delta\varphi \end{cases}$$

2.2 Überlegungen zur Mechanik und Biomechanik

Im Folgenden sind einige Überlegungen aus biomechanischer Sicht zum Ansteuerungsprinzip wiedergegeben. Die Aussagen sind Ergebnis zweijähriger Erfahrung mit humanoiden Robotern und finden sich größtenteils in Nachtigall [1] wieder.

2.2.1 Kupplung

Viel Zeit bei der Arbeit mit Robotern geht durch die Instandhaltung verloren. Hin und wieder muss ein Servomotor ausgetauscht werden, weil dessen Getriebe gebrochen ist. Es kann auch passieren dass ein blockierter Servo durchbrennt. Die Überlastung der Servomotoren hängt auch zusammen mit der direkten Einleitung der Kraft von dem

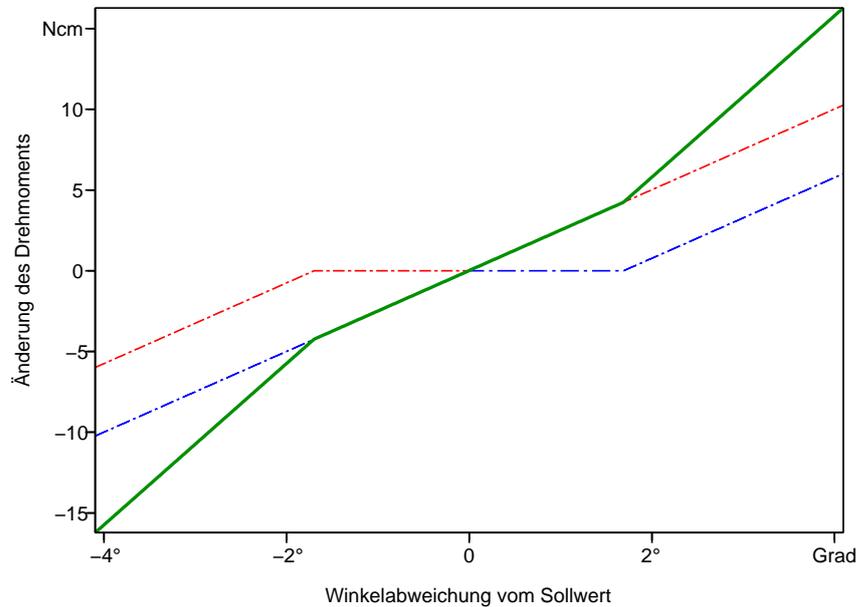


Abbildung 2: Kennlinien von zwei Servos ergänzen sich zu einer umkehrbaren Funktion

Motor über ein Getriebe zu dem angesteuerten Arm. Bekommt der Arm einen Schlag ab, so geht der Impuls direkt in das Getriebe und den Motor. Bei einer elastischen Ansteuerung wird die Kraft über eine Feder indirekt übertragen. Die Feder wirkt sich als Kupplung aus. Der Vorteil liegt klar auf der Hand. Die Getriebe halten länger, starke kurze Kraftimpulse können praktisch keinen Schaden mehr an den Motoren anrichten.

2.2.2 Einstellbare Steifigkeit

Da die Krafteinleitung nach dem vorgestellten Prinzip elastisch ist, fragt sich natürlich wie hart oder weich die Federung sein soll. Idealerweise ist die Antwort darauf eine vollständig dynamisch einstellbare Federung. Basketballer spannen schnell ihre Muskeln an wenn ein Gegenspieler in sie hinein läuft, für einen untrainierten Menschen

ist es als lief er gegen eine Wand. Sonst aber laufen und springen Basketballer dank elastischer Muskeln sehr behände über das Spielfeld.

Im Rahmen dieser Arbeit werden mindestens zwei Motoren pro Gelenk verwendet und damit kann die Steifigkeit des Gelenkes einfach eingestellt werden. Ziehen die Motoren entgegengesetzt, so erhöht sich die Steifigkeit. Dies lässt sich umkehren um die Steifigkeit zu verringern. Diese Parametrisierbarkeit der Steifigkeit ist im Prinzip vorhanden, wurde aber hier nicht konkret verfolgt.

2.2.3 Energieeffizienz

Der Mensch selbst bewegt sich schon von Natur aus sehr effizient vorwärts. Die elastischen Muskeln ermöglichen, einem gedämpften Pendel gleich, ein Zurückschwingen des Beines beim Gehen. Hierbei wird die Schwerkraft genutzt um die Potenzielle Energie des Beines teilweise in Kinetische Energie zurück zu wandeln. Bei Robotern einfacher Bauart sind die Motoren direkt eingekoppelt, so ist z. B. ein Rückschwingen des Beins nicht möglich. Vielmehr muss das Bein unter Energieaufwand getrieben zurück bewegt werden. Allein in dieser Bewegung steckt schon viel Potenzial für Energieeffizienz. Konkrete Vergleiche von Energieaufwand verschiedener Antriebe gibt es in dieser Arbeit nicht, aber sie trägt dazu bei künftig die ungenutzten Einsparpotenziale in eine längere Betriebsdauer von Robotern umzuwandeln.

2.2.4 Schwingungen im System

Natürlich birgt der Ansatz, die Kraft über eine Federung einzuleiten, die Gefahr, dass man ein schwer zu kontrollierendes schwingendes System erhält. Es gibt zwei wesentliche Möglichkeiten dem Problem zu begegnen. Die erste Möglichkeit ist die Schwingungen durch intelligente Motoransteuerung aktiv zu kompensieren. Die zweite Möglichkeit ist gedämpfte Federn zu benutzen. Diese Variante bietet den Vorteil dass Eigenschwingungen schon vom System selbst kompensiert werden. Im Rahmen dieser Arbeit wurden beide Varianten getestet.

2.3 Eine Kostenfrage

Bei neuen Technologien und Methoden ist der Kostenfaktor immer ein wichtiges Entscheidungskriterium über den Lebensweg der neuen Technologie. Übersteigen die Kosten den Nutzen, so tritt die neu entwickelte Technologie nicht in Alltagsanwendung. Es stellt sich also die Frage ob die Steuerung eines Gelenkes mit mehreren Servos sinnvoll ist, wenn es auch mit einem Servo funktioniert, sei es auch nicht ganz perfekt. Wichtig dabei ist zu unterscheiden, dass die Servos, welche anstelle eines einzelnen Servos verwendet werden, kleinerer Bauart sein können, mit weniger Drehmoment. In der Summe sollte das Drehmoment der Servos der des einzelnen entsprechen. Hinzu kommt dass die Elastizität im Antrieb dafür sorgen kann, dass das aufsummierte Drehmoment der mehreren Servos unterhalb des Drehmomentes, des Einzelservos, liegen kann. Definitiv ist die Verwendung des elastischen Antriebs mit höheren Kosten verbunden und sollte nur in Spezialfällen günstiger sein als ein Einzelservo. Die Vorteile des Antriebs dürften aber bei sorgfältiger Konstruktion desselben die höheren Kosten überwiegen. Die Steuerung des Gelenks ist möglich ohne durch Getriebespiel behindert zu werden und es lässt sich in vielen Bewegungen des Gelenks Energie sparen, was in einer längeren Laufzeit des autonomen System resultiert. Ein weiterer Vorteil des elastischen Antriebs ist dass durch die Verspannung mit Seilen ein Roboterarm bei gleicher Traglast leichter gebaut werden kann als ein herkömmlicher Roboterarm. Die Verspannung führt zu einer Biegeentlastung und dadurch zu einer Kostengünstigeren Bauform des Arms (vgl. Abbildung 3).

Im Rahmen dieser Arbeit wäre die Kostenfrage also zunächst geklärt, konkrete Implementationen des Antriebs benötigen jedoch auch einer individuellen Beurteilung der Lage.

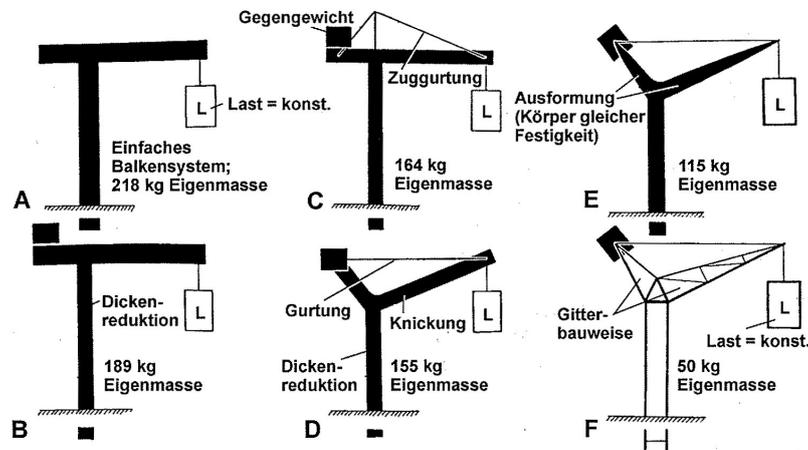


Abbildung 3: Hier wird im Nachtigall [1] demonstriert, wie Zuggurtung zu einer Entlastung des Systems beiträgt, ohne die Belastungsfähigkeit zu verringern.

3 Eine Auswahl verwandter Arbeiten

3.1 Elastisch angetriebener Roboterarm

Eine sehr stark mit der vorliegenden Arbeit verwandte Arbeit ist der „elastisch angetriebene Roboterarm“ [3] von Prof. Dr. Bernhard Möhl von der Uni Saarland. Hierbei wird das herkömmliche Konstruktionsprinzip in der Industrierobotik auf den Kopf gestellt. Statt einer starren Ansteuerung mit hochpräziser Winkelauflösung und der damit verbundenen wuchtigen Bauweise schlägt Professor Möhl vor einen Roboterarm elastisch anzutreiben.

Die Gelenke eines solchen elastisch angetriebenen Roboterarmes werden von zwei Motoren getrieben. Ein Motor ist über gefederte Seilzüge mit dem jeweiligen Gelenk verbunden und bietet die Kraft auf, die das Gelenk benötigt. Der zweite Motor kann entkoppelt werden und ist für die Feinsteuerung des Gelenkes verantwortlich.

Derzeit wird das Konstruktionsprinzip federführend an der Uni Darmstadt weiterentwickelt [4], gefördert durch das Bundesministerium für Bildung und Forschung. Diese Arbeit zeigt, wie viel Potenzial in dem Antriebsprinzip steckt. Der elastisch angetriebene Roboterarm ist zum Patent angemeldet.

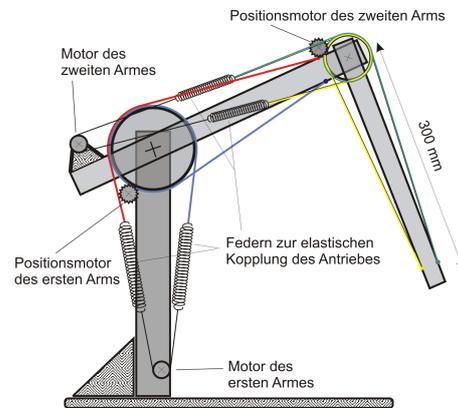


Abbildung 4: Dieses Bild von der Homepage des Darmstädter Projektes zeigt das Konstruktionsprinzip des „elastisch angetriebenen Roboterarmes“.

Der „elastisch angetriebene Roboterarm“ ist ein Vorreiter auf dem Gebiet der elastischen Antriebe für Roboter und in diesem Sinne ein geistiger Vater der hier vorliegenden Arbeit.

3.2 Oktavio

Ein weiteres verwandtes Projekt ist das des mehrbeinigen Laufroboters Oktavio (Abbildung 5). Er wurde von Manfred Hild und Torsten Siedel erdacht, konzipiert und konstruiert. Der Laufroboter ist modular aufgebaut, die baugleichen Beine können einzeln abgenommen und angeschraubt werden. Auch im Oktaviobein wird die Kraft über eine Feder eingeleitet, ohne Seilzug. Es handelt sich hier um eine Drehmomentkupplung, welche die Motoren vor Stößen und anderen äußeren Einflüssen schützen soll. Die Erfahrungen aus dem Oktavioprojekt stützen die Hoffnung dass elastisch getriebene Gelenke auch für humanoide Roboter einsetzbar sind. Eine genauere Beschreibung von Oktavio befindet sich in der Dissertation von Manfred Hild [5]

4 Der Experimentalaufbau

Der schematische Aufbau zeigt sich in Abbildung 6

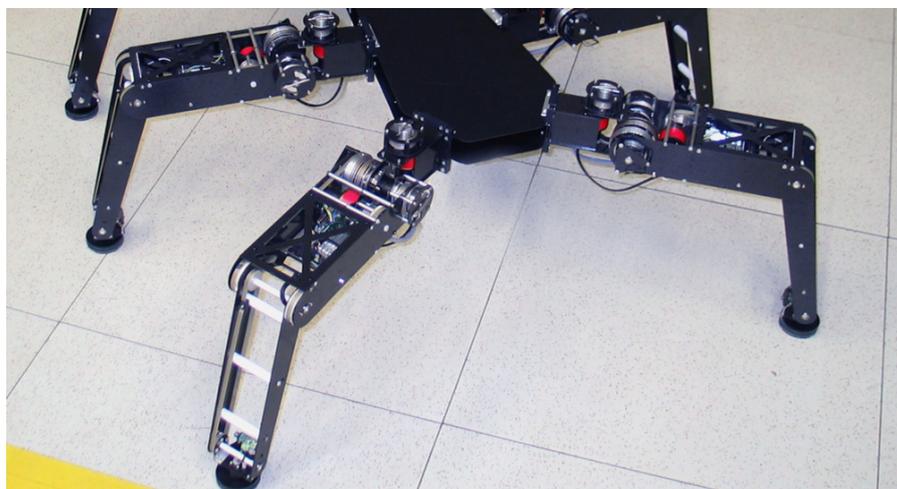


Abbildung 5: Foto vom Laufroboter Oktavio

Die Servomotoren, deren Anzahl skalierbar ist, sind für den Versuchsaufbau fest auf einer horizontal liegenden Holzplatte montiert. Die Achsen der Ruderhörner der Servomotoren sind parallel zur Drehachse des Gelenkarmes. Auf allen Achsen befinden sich Führungsräder für Seilzüge. Dass die Servos mit Führungsrädern anstelle von den sonst auch üblichen Ruderhörnern ausgestattet sind hat den Vorteil dass die Kraftübertragung an dieser Stelle linear ist. Die Zugseile aus festem Garn werden jeweils einmal um das Führungsräder am Servo gewickelt, dann mit einer Feder verbunden und diese Feder wird am anderen Ende mit einem Zugseil verlängert, welches einmal um das Führungsräder an der Achse des Gelenkarmes gewickelt wird und schließlich am Gelenkarm selbst verankert wird. Die Zugseile können auf der Servoseite durch eine Schraubvorrichtung gespannt werden. Die Achse des Gelenkarmes ist mit zwei Kugellagern fixiert. Am Ende des Gelenkarmes befindet sich ein zweiachsiger Beschleunigungssensor, der die radialen und tangentialen Beschleunigungskräfte messen kann. Der Sensor ist über drei flexible Kabel angeschlossen die entlang des Gelenkarmes geführt sind und über dessen Achse geführt werden. So werden die Messungen praktisch nicht durch Torsion der Kabel behindert. Der Winkel des Gelenkarmes wird über einen kapazitiven Winkelgeber gemessen. Dieser besteht aus zwei gegenüberliegenden

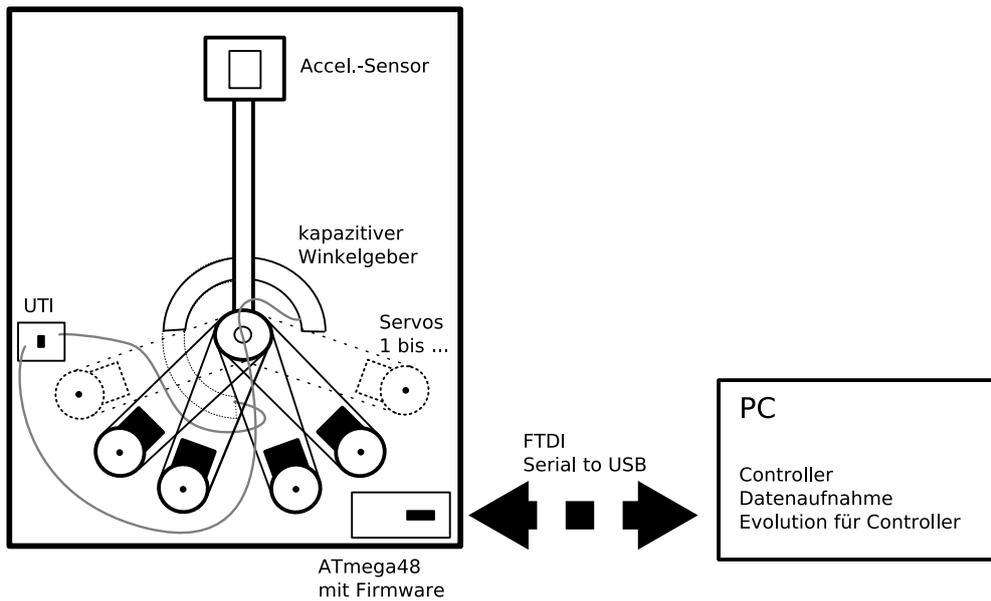


Abbildung 6: Schemaskizze des Experimentalaufbaus

Ringsegmenten die konzentrisch zur Drehachse montiert sind. Das obere Ringsegment ist mit dem Gelenkarm verschraubt und ein Kabel führt von dort über die Drehachse einem Multifunktionssensor (UTI03), der in diesem Falle die Kapazität des Winkelgeber misst.

Weitere Details zur Elektronik, Firmware und Software finden sich im Anhang.

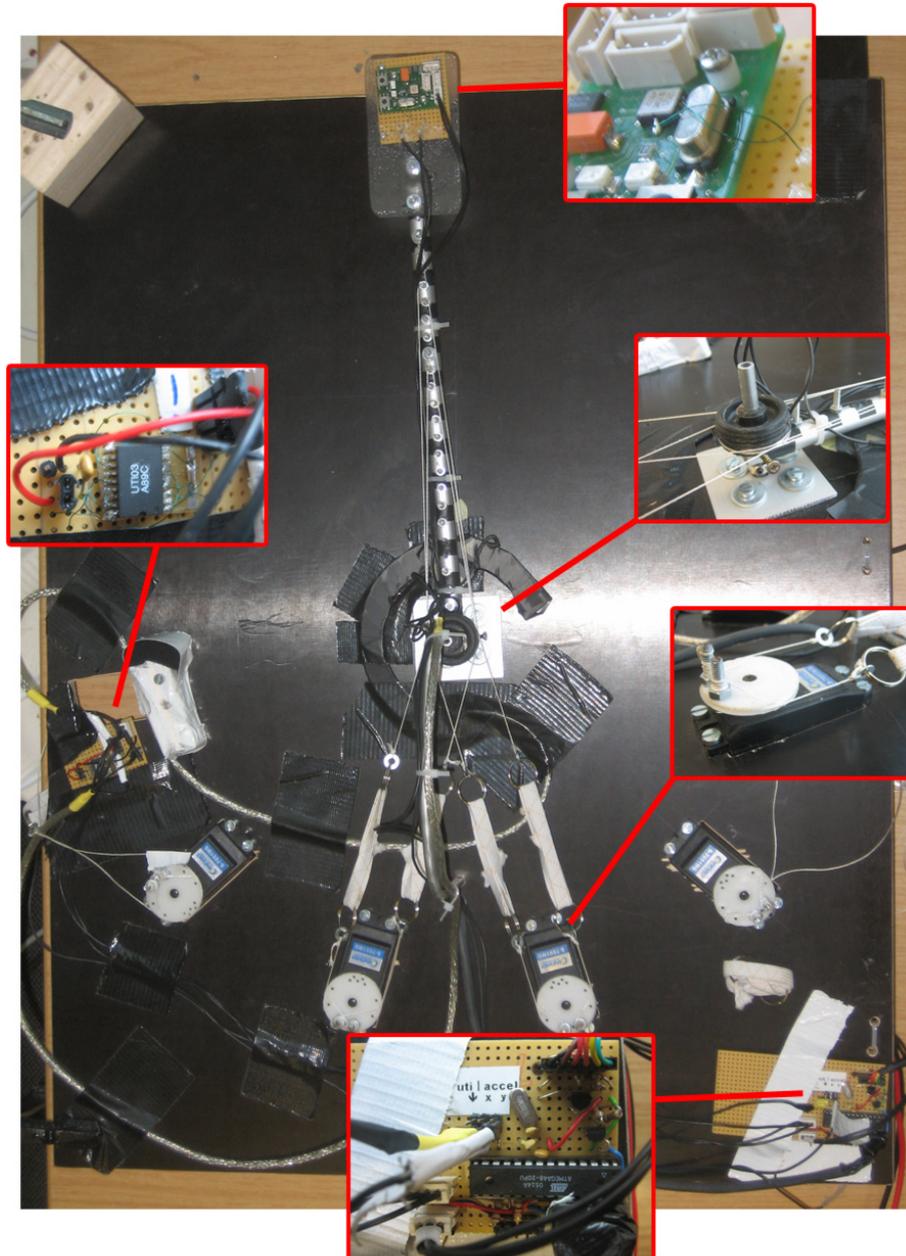


Abbildung 7: Ein Foto des Aufbaus mit einigen Nahaufnahmen

5 Experimente

5.1 Erste Tests

Unsere ersten Tests untersuchten die Funktionstüchtigkeit des Aufbaus und förderten Eckdaten der Sensoren und des Timings zutage.

5.1.1 Pseudocontroller ohne Winkelmesser

Ein erster Funktionstest der Evolution fand statt noch bevor der Winkelsensor konstruiert und implementiert war. Die einfache Aufgabe der Evolution bestand darin die Bewegung eines Motors durch den anderen Motor auszugleichen, so dass die entstehenden Beschleunigungskräfte am Arm minimal sind. Dazu hat ein Motor immer die gleiche Position angefahren und die Werte des Beschleunigungssensors (tangentele Beschleunigungskräfte) wurden als Eingangswert für die Fitnessfunktion der Evolution benutzt. Evolviert wurden die Parameter eines PD-Controllers, wobei der proportionale Anteil der Differenz der angesteuerten Servopositionen entsprach und der differentielle Anteil der gemessenen Beschleunigung.

Auswertung

Schon nach wenigen Generationen hat die Evolution der Parameter ein paar sehr gute Individuen hervorgebracht. Das verwundert nicht, da der Suchraum nicht sehr komplex war. Anscheinend sind jedoch die Individuen sehr angepasst auf den konkreten Versuch. Hält man den Arm fest, so regelt der zweite Motor nahezu identisch. Lässt man den Arm wieder los, so schnell er in die gleiche Position in die er, ohne festzuhalten, gefahren wäre. Das als großen Erfolg zu werten wäre übertrieben, die Funktionsfähigkeit der Evolution und der Servoansteuerung hat sich aber erwiesen.

5.1.2 Winkelgeber

Nachdem der Hardwareteil des Winkelsensors fertiggestellt wurde, galt es dessen Funktionstüchtigkeit zu testen. Dabei wurde schnell klar, dass der Sensor keine lineare Kennlinie hat und eine Kalibrierung implementiert werden musste. Diese ist ausführlich in

Kapitel 7.3 beschrieben. In Abbildung 8 sind die vom Winkelsensor gemessenen Werte linear auf der Abszisse abgetragen. Die Ordinate spiegelt die realen Winkelstellungen des Gelenkarms wider. Der Verlauf des Graphen ist eindeutig nichtlinear, besonders an den Rändern, bei denen die Kondensatorplatten entweder völlig oder gar nicht übereinander liegen, sind starke Anstiegsänderungen zu erkennen.

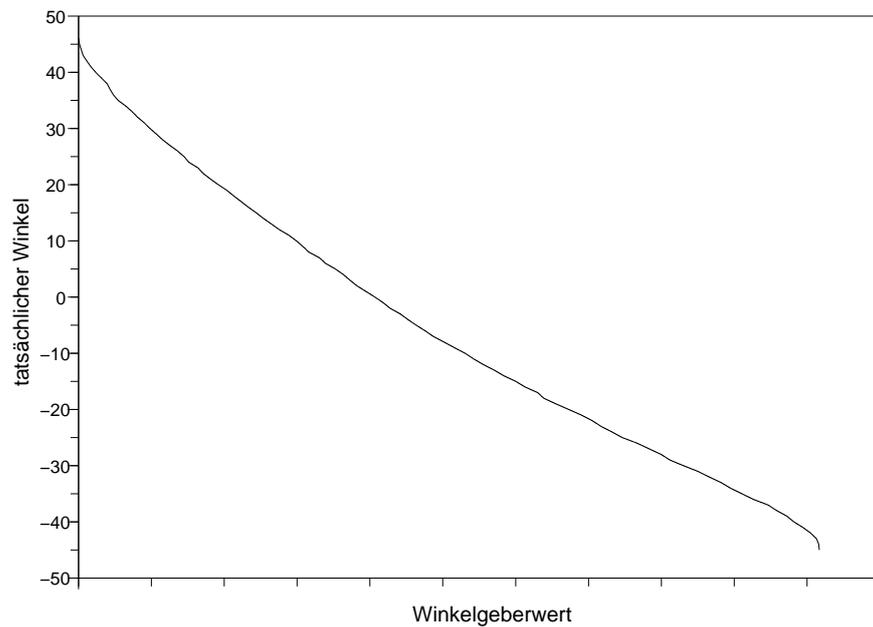


Abbildung 8: Verlauf der gemessenen Werte des Winkelsensors

5.1.3 Federkennlinie der Gummibänder

Die Federkennlinie der von uns verwendeten Gummibänder ist in Abbildung 9 dargestellt. Sie wurde experimentell bestimmt. Dazu wurde ein Gummiband an der einen Seite an einen Zugkraftmesser angebracht und die andere starr befestigt. Anschließend wurde der Zugkraftmesser so weit entfernt, dass die Gummibänder gespannt waren,

aber noch keine messbare Kraft ausgeübt wurde. Die Kraft wurde nun stetig erhöht und immer bei vollen Newton die Länge, um der sich die Gummibänder gedehnt hatten, gemessen.

Auswertung

Anhand der Federkennlinie ist ersichtlich, dass bei geringeren Kräfteinwirkungen bis ca. 10 N sich die Bänder nicht so stark dehnen, wie bei Kräfteinwirkungen zwischen ca. 10 bis 16 N. Ab ca. 20 N war fast die maximale Ausdehnung des Gummibandes erreicht, dies ist in der Abbildung nicht mehr dargestellt. Eine explizite Modellierung, beziehungsweise Kompensation, dieses Verhaltens fand in den Experimenten nicht statt.

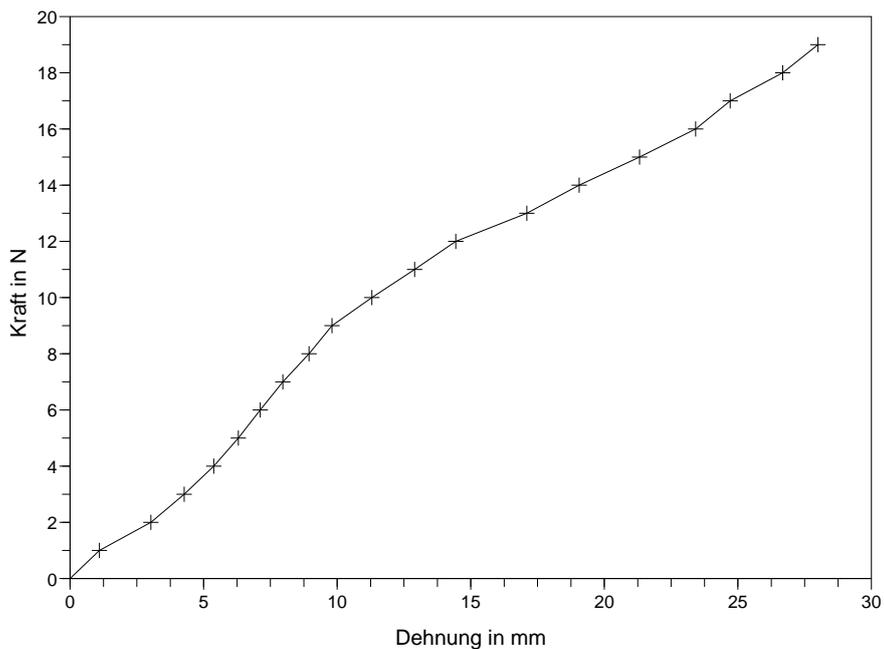


Abbildung 9: Federkennlinie der Gummibänder

5.2 Experiment 1

5.2.1 Controller mit vier Servos

Das erste Experiment mit vier Servomotoren wurde mit Hilfe eines PID-Controllers durchgeführt. Dabei wurde für jeden Servo, unabhängig von den anderen, ein PID-Controller zur Ansteuerung eines Zielwinkels implementiert. Als Sensorwert wird nur der aktuelle Winkel genutzt. Zusammen mit dem Zielwinkel ergibt sich ein Fehler, der zu jedem Zeitschritt (alle 20 ms), berechnet wird und die Grundlage für die PID-Berechnung bildet. Der PID-Controller besteht aus drei Teilen, einem proportionalen, einem integralen und einem differentialen Teil. Jeder dieser Anteile benötigt eine Gewichtung. Es sind also drei Gewichte pro Servo und 12 insgesamt. Diese 12 Gewichte wurden unter Verwendung einer künstlichen Evolution bestimmt.

Die Populationsgröße bei dieser Evolution betrug 20 Individuen, diese wurden für die erste Generation zufällig bestimmt, die Gewichte für den proportionalen Anteil aus dem Intervall $[0, 1]$ und die restlichen aus dem Intervall $[-0.5, +0.5]$. In die Folgegeneration wurden jeweils die 12 besten Individuen übernommen. Die fehlenden Individuen wurden durch Kreuzungen der 12 besten bestimmt, dabei werden zunächst zwei der besten ausgelost, die Eltern. Für jedes Gen wird nun mit einer Wahrscheinlichkeit von 50% ein Elternteil bestimmt, sodass der Nachkomme im Durchschnitt die Hälfte seiner Gene von einem Vorfahr besitzt. Nachdem alle fehlenden Individuen ermittelt wurden, werden alle Individuen mit einer bestimmten Wahrscheinlichkeit mutiert, diese betrug in dem Fall 35%. Bei den zu mutierenden Individuen wurden im Durchschnitt 25% der Gene mutiert, das heißt per Zufall ein neuer Wert im vorgegebenen Intervall ermittelt. Um überhaupt feststellen zu können welches Individuum besser ist als das andere und damit eine Chance auf das Fortleben in der Evolution hat, muss jedes Individuum bewertet werden. Dies geschieht durch einen Testlauf, dessen Ergebnis mit Hilfe einer Fitnessfunktion bewertet wird. Der Testlauf bestand, bei dieser Evolution, aus dem Anfahren von drei nacheinander vorgegebenen Positionen. Die Fitness wurde ausschließlich aus dem Fehler des aktuellen Winkels, in Bezug auf den anzufahrenden, berechnet. Je geringer der Fehler, desto größer die Fitness.

5.2.2 Auswertung

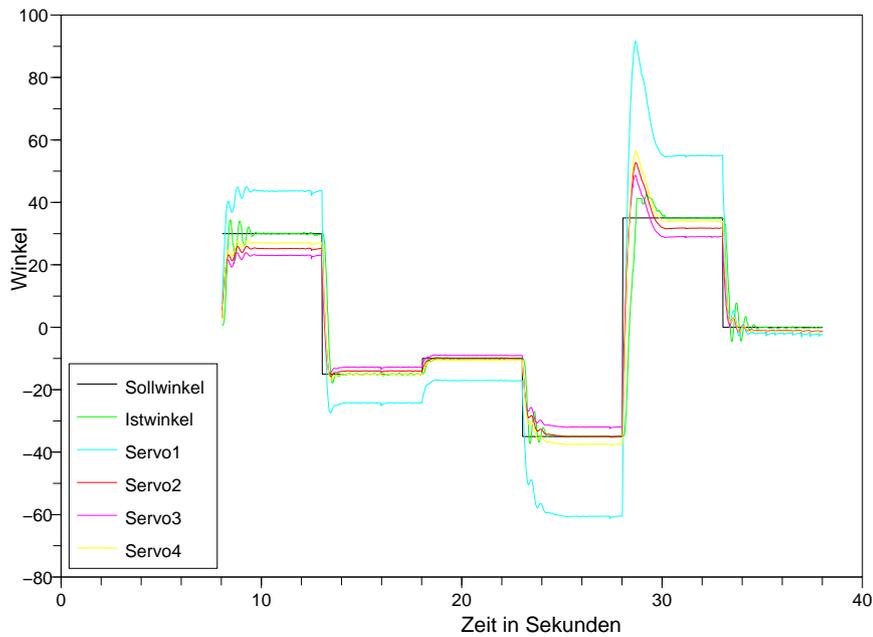


Abbildung 10: Testlauf des PID-Controllers mit 4 Servos

Die künstliche Evolution durchlief insgesamt 32 Generationen, bis ein gutes Individuum gefunden wurde und sich dieses auch im Verlauf nicht mehr verbesserte. In der Abbildung 10 ist das Ergebnis einer Testfahrt eines sehr guten Individuums dargestellt. Bei dem Test wurden nacheinander sechs verschiedene Winkelpositionen vorgegeben, sie sind als schwarze Linie eingezeichnet. Die Positionen wurden jeweils Fünf Sekunden gehalten. Die Trajektorie der Werte des Winkelsensors ist hellgrün dargestellt. Die eingezeichneten Winkelstellungen der Servomotoren beziehen sich auf die vom jeweiligen PID-Controller ermittelten Werte. Diese werden noch auf ganze Zahlen gerundet und entsprechen dann den Positionen, die die Servomotoren real ansteuern können. Anschließend erfolgte die Übertragung zum Mikrocontroller.

Die anzufahrenden Winkel wurden sehr gut erreicht, die aktuelle Position liegt nach kurzem Überschwingen exakt auf der vorgegebenen. Sobald der Gelenkarm ruhig war, hat der Controller auch keine Schwingungen erzeugt. Allerdings konnte der PID-Controller in den meisten Fällen das Nachschwingen nicht zufriedenstellend reduzieren, dies liegt insbesondere an der festen Winkelsteuerung der Servomotoren, mit einer Drehmomentsteuerung wären wahrscheinlich bessere Ergebnisse zu erreichen. Ferner zeigt sich, dass drei Servomotoren (rot, magenta und gelb in Abbildung 10) nahe beieinander liegende Positionen ansteuern. Jedoch sind die Reaktionen auf den Verlauf des aktuell gemessenen Winkels und damit die Trajektorien der Servos leicht Phasenverschoben. Das heißt, dass der Servomotor 3 (magenta) als erster reagiert und danach der Motor 2 und 4 folgt. Der vierte Motor allerdings liegt deutlich entfernt von dem Zielwinkel und übersteuert meistens. Dadurch werden die Gummibänder in den Seilzügen stärker gespannt und der Gelenkarm wird steifer gehalten. Dieses Verhalten entspricht in Ansätzen dem in Teil 2.1 beschriebenen Konzept zur Verringerung des Getriebebeispiels.

Bei geringen Änderungen des anzusteuernenden Winkels vermindert der Controller das Überschwingen des Gelenkarmes deutlich, dies ist in Abbildung 10 bei der Stelle von 18 Sekunden sichtbar. Das geringe Schwingen hat aber auch ein langsames Erreichen der Zielposition zur Folge. Das Gegenteil ist bei großen Änderungen der Fall, z. B. bei der Stelle von 28 Sekunden. Um möglichst schnell den Sollwinkel zu erreichen, übersteuern alle Servomotoren. Servo 1 übersteuert dabei um mehr als den doppelten Winkel der anderen. Beim Testlauf befanden sich die Servomotoren bis zu zwei Sekunden über dem eigentlichen Zielwinkel. Dadurch übersteuert auch der Gelenkarm, der sich erst nach gut zwei Sekunden auf die anzufahrende Position zurück bewegt.

Zusammenfassend erzielt der PID-Controller mit vier Servomotoren gute Ergebnisse. Besonders bei kleinen Änderungen und die Steifigkeit des Gelenkarmes durch versetzte Servowinkel fallen positiv auf, allerdings kann das Nachschwingen oft nicht zufriedenstellend minimiert werden.

5.3 Experiment 2

5.3.1 Controller mit vier Servos neuronal gesteuert

Bei diesem Experiment kamen wieder vier Servomotoren zum Einsatz, allerdings wurden diese nicht mehr durch mehrere PID-Controller gesteuert, sondern durch ein neuronales Netz. Dieses neuronale Netz bestand in diesem Fall aus drei Eingangs-, vier Ausgangs- und zwei Hidden-Neuronen, als Transferfunktion wurde der Tangens Hyperbolicus benutzt. Es können Verbindungen zwischen allen Neuronen bestehen, jedoch haben die drei Eingangs-Neuronen keine hinführenden Verbindungen oder Selbstkopplungen. In der Summe ergibt das 54 mögliche Verbindungen zwischen den Neuronen. Um die Werte des Winkelgebers und des Beschleunigungssensor sinnvoll in das Netz integrieren zu können, wurden sie auf das Intervall $[-1, +1]$ normiert. Zwei der drei Eingangs-Neuronen wurden in jedem Zeitschritt jeweils mit den aktuellen Messwerten des Winkelgebers bzw. des Beschleunigungssensors belegt. Das dritte Eingangs-Neuron wurde mit dem aktuell anzufahrenden Zielwinkel belegt. Die vier Ausgangs-Neuronen steuerten direkt den Zielwinkel der einzelnen Servomotoren an. Die Zielwinkel wurden wieder auf das ganzzahlige Intervall normiert und in jedem Zeitschritt an den Mikrocontroller gesendet.

Zur Bestimmung der 54 Gewichte, die die Verbindungen zwischen den Neuronen widerspiegeln, kam, wie bei den PID-Controllern, eine künstliche Evolution zum Einsatz. Da der Suchraum jetzt viel größer war, musste die Fitnessfunktion und andere Parameter der Evolution viel genauer festgelegt werden. Daher bestand die Fitnessfunktion nicht nur noch aus dem Fehler zwischen Soll- und Istwinkel, sondern setzte sich aus mehreren Komponenten zusammen.

Der Fehler des Winkels stellte, mit einer Gewichtung von ca. 65% von der Gesamtfitness, aber immer noch den größten Teil dar. Ein weiterer Anteil ist der Durchschnitt der einzelnen Servobewegungen, dieser wird anhand der Ausgangs-Neuronen bestimmt. Dazu wurde in jedem Zeitschritt die Differenz zwischen der aktuellen Aktivität und der Aktivität aus dem Zeitschritt davor bestimmt, also die erste Ableitung gebildet. Diese wurden aufsummiert und am Ende des Testlaufs durch die Anzahl der Zeitschritte ge-

teilt. Diese Teilfitness hatte einen Anteil von ca. 16% an der Gesamtfitness und wurde hinzugefügt um unnötige Bewegungen der Servos zu minimieren. Insbesondere beim halten eines Winkels könnte es sonst dazu kommen, dass zwei oder mehr Servomotoren ständig gegeneinander drehen, ohne dass dies der Winkelsensor registrieren würde. Des Weiteren wird noch in jedem Zeitschritt überprüft wie stark sich die Aktivitäten und damit die Stellungen der Servos unterscheiden. Es wird davon wieder der Durchschnitt über alle Zeitschritte während des Testlaufs bestimmt. Damit werden stark gegensteuernde Servomotoren bestraft. Der Einfluss auf die Gesamtfitness beträgt ca. 8%. Der letzte Teilfitness wird nur über die Gewichte der Verbindungen bestimmt, d.h. es wird der Betrag aller Gewichte summiert und durch das Produkt der Anzahl der Verbindungen und des maximalen Betrages eines Gewichtes geteilt. Dies hat zur Folge das es belohnt wird, wenn die Neuronen nicht im stark gesättigten Bereich arbeiten. Außerdem wird das Auslöschen von Verbindungen erleichtert. Die Gewichtung betrug 10% von der Gesamtfitness.

Die Gewichte für die Verbindungen des neuronalen Netzes wurden aus dem Intervall $[-4, +4]$ bestimmt. Aufgrund der großen Anzahl an Genen pro Individuum (54) wurde die Populationsgröße von 20 auf 30 erhöht. Von den 30 Individuen wurden 12 mit in die nächste Generation übernommen, der Rest wurde wieder über Kreuzungen aufgefüllt. Die Mutationsrate, ob ein Individuum mutiert wird, lag bei 50%, von dem zu mutierendem Individuum wurden 15% der Gene, also durchschnittlich 8, verändert. Um zu verhindern, dass sehr gute Individuen durch die Mutation verloren gehen, wurde eine Elite der Größe eins eingeführt. Folglich wurde das beste Individuum in der Folgegeneration nicht mutiert. Da aber trotzdem mutierte Nachkommen entstehen sollten, wurde das Individuum zusätzlich ein weiteres Mal in die Generation eingefügt.

Die Evolution durchlief insgesamt 124 Generationen, der genaue Verlauf der besten, durchschnittlichen und schlechtesten Fitness ist in Abbildung 11 dargestellt. Die maximale Fitness steigt zwar immer langsamer, aber dafür stetig an. Die durchschnittliche Fitness verhält sich ähnlich, unterliegt allerdings größeren Schwankungen. Wie zu erwarten war schwankt die schlechteste Fitness sehr stark am unteren Ende. Die Schwankungen der besten Fitness nach oben kann wie folgt erklärt werden. Mit zu-

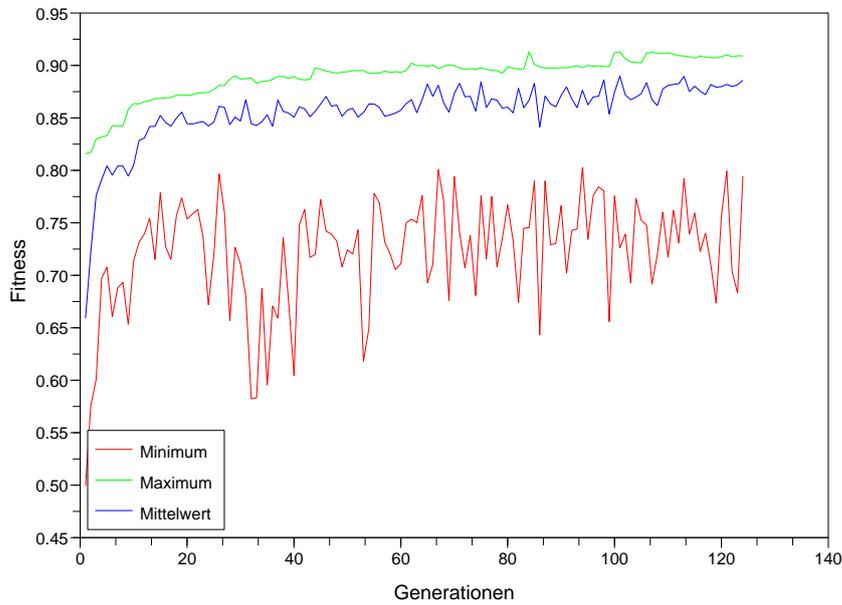


Abbildung 11: Verlauf der Evolution (4 Servos mit neuronalem Controller)

nehmender Dauer des Betriebs der Servomotoren heizen diese sich langsam auf und verhalten sich dadurch etwas anders. Dies führt zu einer verminderten Fitness, da die Winkelabweichungen größer werden. Wird die Evolution also pausiert und die Servomotoren können sich abkühlen, bekommt das gleiche Individuum eine höhere Fitness als vorher. Solange dies aber nicht inmitten einer Generation passiert und die Individuen einer Generation vergleichbar bleiben, ergeben sich daraus für die gesamte Evolution keine Probleme. Die minimale Fitness bewegt sich aufgrund der Normierung meist über 0.5 oder sogar 0.7. Die theoretisch minimale Fitness von 0.0 ist praktisch nicht zu erreichen. Auch Werte schlechter als 0.5 sind nur schwer zu erreichen.

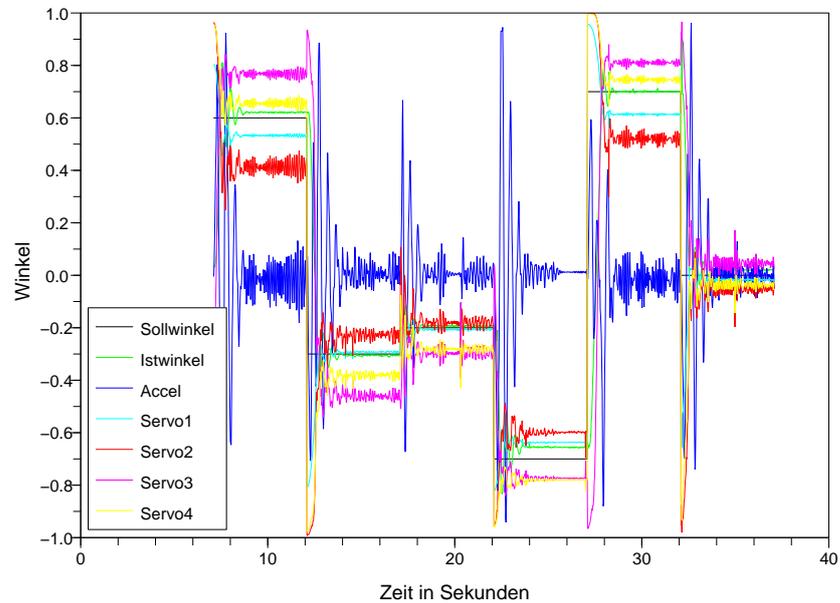


Abbildung 12: Ergebnisse des Testlaufs (4 Servos mit neuronalem Controller)

5.3.2 Auswertung

In Abbildung 12 ist die Testfahrt des besten Individuums dargestellt. Dieses Individuum wurde am Ende der Evolution, durch das Vergleichen von mehreren ausgewählten Individuen der gesamten Evolution, die alle eine sehr gute Fitness erreicht hatten, bestimmt. Die Darstellung entspricht der Abbildung 10 aus dem Kapitel 5.2, nur dass als blaue Linie die Beschleunigungswerte des Sensors vorne am Gelenkarm hinzugefügt wurden.

Betrachtet man das Verhalten der Winkelstellungen der einzelnen Servomotoren, nachdem eine Sollpositionsänderung durchgeführt wurde, so wird ersichtlich, dass es drei Motoren gibt die ähnlich reagieren und einen der sich anders verhält. Servo eins, zwei und vier sollen schnellst möglich in die Sollposition fahren, deshalb regelt der Controller sofort weit über die Sollposition hinaus. Außer bei sehr kleinen Änderungen entspricht

dies meist der maximal möglichen Position in die jeweilige Richtung. Nur der erste Servomotor übersteuert geringer und ist auch derjenige, der sich wieder der Sollposition annähert und diese sogar wieder unterschreitet. Die anderen beiden steuern erst später der Sollposition entgegen, Servo vier verharrt bei einer geringen Übersteuerung und Servo zwei fährt sogar in eine weit untersteuernde Position. Der dritte Servomotor verhält sich vollkommen anders, zu Beginn der Sollwinkeländerung fährt er entgegengesetzt zur eigentlichen Richtung und fährt dann mit ähnlicher Geschwindigkeit wie der Gelenkarm auf eine stark übersteuerte Position. Durch dieses Verhalten der vier Servomotoren ergibt sich, wie beim PID-Controller, eine steifere Haltung des Gelenkarms. Auch hier entstand durch die Evolution ein Verhalten, dass in Ansätzen dem Konzept aus Teil 2.1 entspricht. Im Gegensatz zum PID-Controller sind die einzelnen Servomotoren viel unruhiger und unterliegen auch bei schon erreichter Sollposition meist starken Schwankungen, die sich bei größerer Amplitude bis zum Gelenkarm fortsetzen können. Dies ist besonders an den Ausschlägen des Beschleunigungssensors zu erkennen, nur vereinzelt kommt der Gelenkarm zur Ruhe. Der zweite Servomotor ist der unruhigste im Gegensatz zu Servo eins, der vollkommen ruhig.

Zur genauen Bewertung des evolvierten neuronalen Controllers, wurde ein Vergleich zwischen diesem und keinem externen Controller durchgeführt. Kein externer Controller bedeutet, dass mit allen Servomotoren starr die vorgegebene Position mittels des internen Servo Controllers angefahren wurde. Bei sehr guter Kalibrierung, sollte dann auch der Gelenkarm an dieser Position stehen. Bei dem Vergleich wurde die bekannte Testfahrt, bestehend aus sechs unterschiedlichen Winkelpositionen, benutzt. Die Ergebnisse des Vergleichs wurden in Abbildung 13 zusammengeführt. Es ist deutlich zu erkennen, dass die Position ohne Controller weiter entfernt von der gewünschten Position liegt, als mit neuronalem Controller. Jedoch das eigentliche Ziel das Überschwingen über die Zielposition zu Minimieren nicht erreicht wird, meist ist das Überschwingen länger als beim starren Anfahren und nahezu gleich stark. Das starre Anfahren der Positionen ist aufgrund der hohen Federvorspannung bei vier Servomotoren besser, im Hinblick auf die Verringerung des Überschwingens, als mit weniger Servomotoren, die in der Summe eine geringere Federvorspannung besitzen. Es sei denn, die Vorspannung

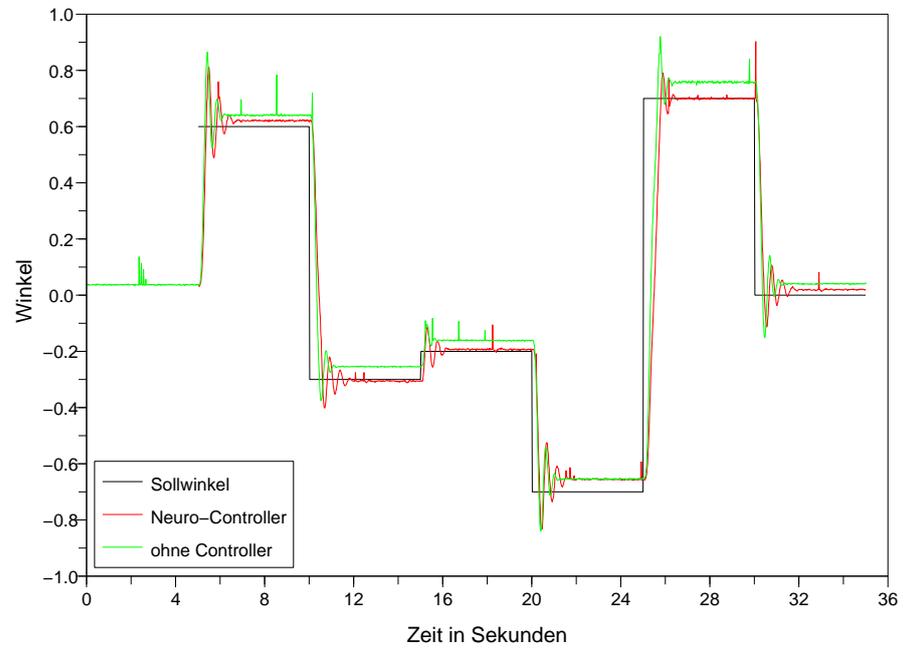


Abbildung 13: Vergleich des Testlaufs zwischen neuronalem und keinem Controller (4 Servos)

der einzelnen Servomotoren wird stark erhöht. Die hohe Federvorspannung hat eine geringere Amplitude und damit eine kürzere Dauer des Überschwingens zur Folge.

5.4 Experiment 3

5.4.1 Controller mit zwei Servos neuronal gesteuert

Bei diesem Experiment kamen nur zwei statt vier Servomotoren zum Einsatz, bei denen die Federvorspannung nicht erhöht wurde. Dadurch ist die gesamte Federvorspannung zwischen Antrieben und Gelenkarm reduziert.

Der neuronale Controller bei diesem Experiment wurde um zwei Ausgangs-Neuronen reduziert, der Rest entspricht dem Controller mit vier Servos. Folglich besteht das Netz aus insgesamt sieben Neuronen. Daraus ergeben sich 28 mögliche Verbindungen.

Die Gewichte der Verbindungen des neuronalen Netzes wurden wieder mit einer künstlichen Evolution bestimmt. Die Populationsgröße lag bei 20 Individuen, von denen jeweils acht in die Folgegeneration übernommen wurden. Die Mutationsrate lag bei 50%, es wurden 20% der Gene mutiert. Des Weiteren gab es eine Elite der Größe eins. Die Gewichte der Fitnessfunktion wurden verändert. Die Gewichtung des Winkelfehlers wurde auf 78.4% erhöht. Die Servobewegungen hatten nur noch einen Einfluss von ca. 13% auf die Gesamtfitness. Die unterschiedliche Stellung der Servos machte ca. 6.5% aus und die Summe der Beträge der neuronalen Verbindungen wurde mit 2% gewichtet. Der Versuch lief an zwei Tagen zu jeweils etwa sechs Stunden.

5.4.2 Auswertung

Dadurch, dass für das neuronale Netz für diesen Versuch deutlich weniger Parameter bestimmt werden mussten, als im Experiment unter 5.3, ging die Evolution auch einiges schneller voran. Der Erkenntnisgewinn gegenüber dem vorhergehenden Versuch war gering. Die Fitness erreichte schon am ersten Tag des Versuches ein Sättigungsniveau und am zweiten Tag wurden die Werte nicht entscheidend besser. Zumindest konnten die Daten des Versuchs genutzt werden um eine mathematische Analyse des Verhaltens der Steuerung durchzuführen (siehe nächstes Kapitel).

6 Abschließende Überlegungen und Ausblick

6.1 Berechnung der Inversen Funktion

Die Messwerte aus den Experimenten suggerieren, dass eine gedämpfte Schwingung vorliegt. Die Formel für eine gedämpfte Schwingung lautet:

$$y(t) = y_0 \cdot \exp^{-\delta t} \cdot \sin(2\pi f t + \varphi_0)$$

y_0 bestimmt die Auslenkung der Amplitude und f legt die Frequenz der Schwingung fest. φ_0 entspricht der Phasenverschiebung. Die Dämpfung wird durch die Abklingkonstante δ bestimmt. Um die Trajektorie mit dieser Funktion anzunähern müssen Frequenz und Abklingkonstante ermittelt werden. Die Frequenz kann direkt aus dem Versuch abgelesen werden, dazu wurde ein Testlauf mit zwei angeschlossenen Servomotoren durchgeführt und anhand der Nulldurchgänge pro Sekunde die Frequenz bestimmt. Ein Zyklus hat eine Länge von $T = 0.72$ s. daraus ergibt sich eine Frequenz von $f \approx 1.39\text{s}^{-1}$. Die Abklingkonstante gibt die Verringerungsrate der Amplitude pro Sekunde an. Zur Berechnung muss zunächst die Abnahme der Amplitude pro Zyklus bestimmt werden. Dazu liest man die Auslenkung zum Zeitpunkt y_t aus der Messung ab, y_t entspricht hierbei einem lokalen Maximum. Des Weiteren muss $y_t + zT$ bestimmt werden, dieser Wert entspricht der Amplitude um genau z Zyklen später, da T die Zykluszeit ist. Diese beiden Werte müssen ins Verhältnis gesetzt werden.

$$\frac{y_t}{y_t + zT} = n$$

Ist das Verhältnis bestimmt kann die Abklingkonstante berechnet werden.

$$\delta = \frac{\ln(n)}{z \cdot T}$$

Aus der Berechnung ergibt sich eine Abklingkonstante von $\delta = 1.1338\text{s}^{-1}$. Die gemessene Amplitude und deren berechnete Approximation ist in Abbildung 14 dargestellt. Die relativ starken Abweichungen am Anfang und Ende der Schwingung sind unter anderem der nichtlinearen Elastizität der von uns verwendeten Gummibändern zu zuschreiben, was im Abschnitt 5.1.3 näher beschrieben wird.

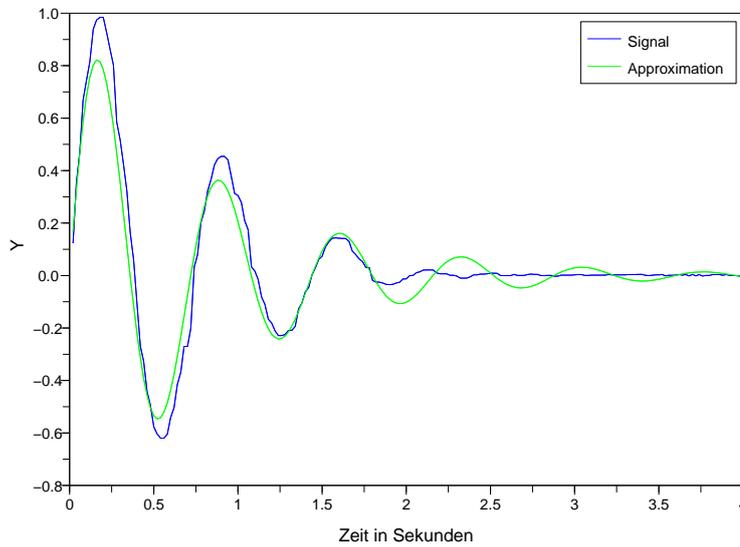


Abbildung 14: Amplitude der gedämpften Schwingung und deren Approximation

Um das Signal gut invertieren zu können, wird es zunächst die z -Transformierte bestimmt. Dazu wird die Schwingung am Einheitskreis betrachtet. φ ergibt sich aus der Frequenz der Schwingung und der Abtastrate.

$$\begin{aligned}\varphi &= \frac{2\pi f_0}{f_s} \\ \varphi &= \frac{2\pi \cdot 1.39}{50} \\ \varphi &= 0.175\end{aligned}$$

r kann aus der Abklingkonstante berechnet werden.

$$\begin{aligned}r &= 1 - \frac{\delta}{f_s} \\ r &= 1 - \frac{1.1338}{50} \\ r &= 0.977\end{aligned}$$

Es gilt:

$$\begin{aligned} z_0 &= r \cdot e^{i\varphi} \\ \bar{z}_0 &= r \cdot e^{-i\varphi} \end{aligned}$$

Die z-Transformierte $H(z)$ kann nun wie folgt bestimmt werden.

$$\begin{aligned} H(z) &= \frac{1}{(z - z_0)(z - \bar{z}_0)} \\ &= \frac{1}{z^2 - (z_0 + \bar{z}_0)z + z_0\bar{z}_0} \\ &= \frac{1}{z^2 - [2r \cos \varphi]z + r^2} \\ &= \frac{z^{-2}}{1 - [2r \cos \varphi]z^{-1} + r^2z^{-2}} \end{aligned}$$

Aus der z-Transformierten können direkt die Filterkonstanten eines rekursiven Filters, dieser wird auch IIR-Filter (infinite duration impulse response filter) genannt, abgelesen werden. Betrachtet man nur das Frequenz- und Dämpfungsverhalten der Schwingung, so reicht es die Koeffizienten der zurückgeführten Daten zu bestimmen. In der Abbildung 15 sind dies d_1 und d_2 . Die anderen Koeffizienten sind vorrangig für die Amplitude des Signals verantwortlich und werden nicht weiter betrachtet. d_1 und d_2 ergeben sich aus dem Faktor vor z^{-1} und z^{-2} im Nenner der z-Transformierten.

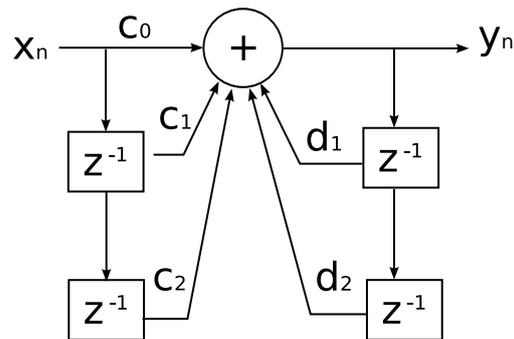


Abbildung 15: Darstellung des IIR-Filters

Würde man einen Dirac-Impuls als Eingangssignal des Filters verwenden, so würde

wieder das Ausgangssignal aus Abbildung 14 entstehen. Betrachtet man die Amplitude des Signals in der Praxis als Auslenkung des Gelenkarms, so macht es keinen Sinn einen Dirac-Impuls als Eingangssignal zu verwenden. Da die Ausgangsstellung der Endstellung entspräche und keine Veränderung der Position vorliegen würde. Daher ist es sinnvoller einen konstanten Wert als Eingangssignal zu benutzen. Dieser kann dann als Offset zur aktuellen Gelenkposition betrachtet werden und würde einer Positionsänderung entsprechen. Der zeitliche Verlauf der daraus entstehenden Schwingung ist in Abbildung 16 dargestellt. Der Offset beträgt in der Abbildung +1, die Auslen-

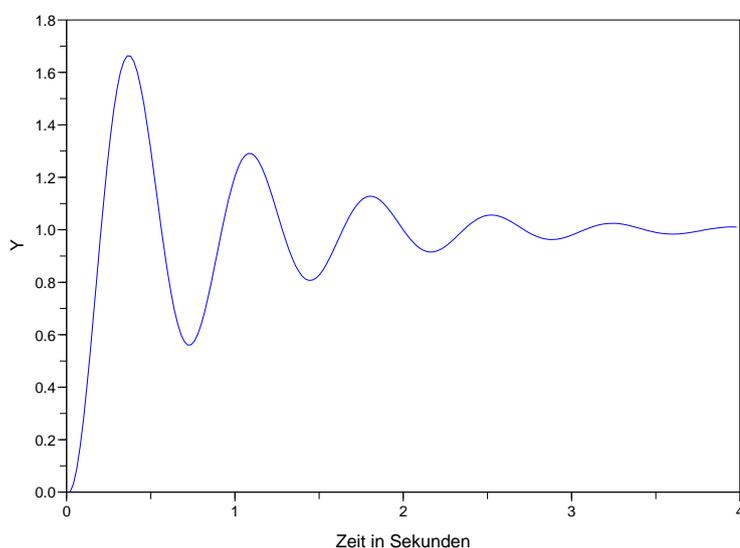


Abbildung 16: Schwingung mit Offset

kung des Armes könnte sich also um einen Grad nach rechts verändert haben. Das zu einer bestimmten Auslenkung passende Eingangssignal kann wie folgt berechnet werden. x entspricht dem Wert des Eingangssignals und y dem des gewünschten Offsets. d_1 und d_2 sind die Filterkoeffizienten.

$$y = x + d_1 y + d_2 y$$

$$\begin{aligned}
 y &= y\left(\frac{x}{y} + d_1 + d_2\right) \\
 1 &= \frac{x}{y} + d_1 + d_2 \\
 \frac{x}{y} &= 1 - d_1 - d_2 \\
 x &= y(1 - d_1 - d_2)
 \end{aligned}$$

Der berechnete Wert muss dann ständig am Eingang des Filters anliegen um die gewünschte Zielposition zu erreichen. Um die Zielposition ohne schwingen zu erreichen muss der Eingangsvektor am Anfang angepasst werden.

Bei diesem idealisierten System ist es möglich die Schwingung komplett zu eliminie-

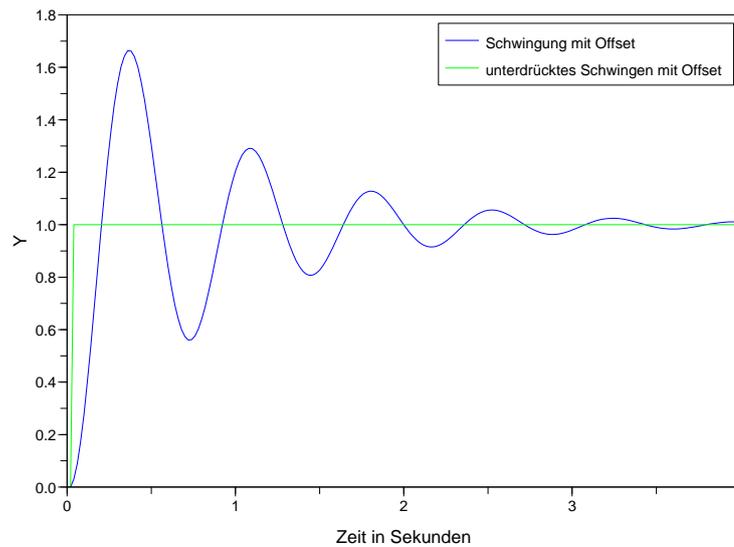


Abbildung 17: Unterdrückte Schwingung im Vergleich

ren. Dazu muss das erste Element des Eingangsvektors auf den gewünschten Zielwert gesetzt werden. Für das zweite Element ergibt sich der Wert aus der Summe des berechneten Eingangssignals und dem Produkt des Zielwinkels und des zweiten Filter-

koeffizienten d_2 :

$$\begin{aligned}x(1) &= y \\x(2) &= y \cdot d_2 + x\end{aligned}$$

Das Ergebnis ist in Abbildung 17 dargestellt. Es ist eindeutig zu erkennen, dass die Schwingung vollständig unterdrückt wurde. Natürlich ist dies nur in der Theorie möglich und ist in der Praxis so nicht zu erreichen, da das aufgestellte System des IIR-Filters nur eine ungefähre Annäherung eines Teilaspektes unseres praktischen Problems darstellt. Jedoch weist das Ergebnis dieser theoretischen Betrachtung schon in die Richtung des in der Praxis benötigten Reglers.

Es wurde gezeigt dass die Steuerfunktion des vorgestellten Antriebssystems umkehrbar ist, was bei einzelnen Servos nicht der Fall ist (vgl. Kapitel 2.1). So lassen sich die Bewegungen mit diesem System vorausberechnen und können zuverlässiger in einer virtuellen Simulation modelliert werden. Das ist natürlich nur möglich weil von dem einzelnen Servo abstrahiert werden kann, denn die Nichtlinearitäten eines einzelnen Servos sind sehr schwer zu modellieren (Zu diesem Thema ist eine Studienarbeit von Matthias Kubisch in Arbeit, sowie ein Paper von Matthias Kubisch, Manfred Hild und Torsten Siedel).

6.2 Ausblick

In dieser Studienarbeit wurden Experimente zu einem Gelenk mit einem einzigen Freiheitsgrad (Rotation) gemacht. In der Natur, zum Beispiel beim Menschen, kommen jedoch auch Gelenke mit zwei (Fußgelenk) oder drei (Hüfte) Freiheitsgraden vor. Die Übertragung des vorgestellten Muskelmodells auf Gelenke mit mehr als einem Freiheitsgrad ist wünschenswert, vor allem wenn man den menschlichen Gehapparat als Vorbild nimmt. Im Labor für Neurorobotik der HU Berlin ist schon ein Experimental-aufbau für ein Kugelgelenk, wie es in der Hüfte des Menschen vorkommt, vorbereitet. Weiterhin müssen die Ergebnisse der Studienarbeit und weiterer Experimente zum Muskelmodell so aufbereitet werden, dass sie direkt in einen Roboter verbaut wer-

den können. Dies ist Bestandteil einer laufenden Diplomarbeit von Torsten Siedel. Er untersucht die Möglichkeit, das Antriebsschema aus unserer Arbeit in ein einzelnes Antriebsmodul zu kapseln. Vor allem der mögliche zusätzliche Instandhaltungsaufwand durch gerissene Seilzüge (etc.) soll vermieden werden. Die Arbeit von Torsten Siedel dürfte mit spannenden Ergebnissen aufwarten. Positiv überraschend zeigte sich, dass die Seilzüge keine Ermüdung zeigten, obwohl der Aufbau ca. 100 Stunden lang in Betrieb war und das größtenteils fünf Stunden am Stück im Dauerbetrieb. Die erwarteten Ermüdungserscheinungen blieben aus und lassen hoffen dass ein solches System eine lange Zeit wartungsfrei laufen kann. Im industriellen Maßstab sind 100 Stunden noch nicht viel, bei einer konkreten Anwendung des elastischen Antriebs müsste es weitere Untersuchungen geben, die zeigen sollen wie stabil der Antrieb läuft.

7 Anhang

7.1 Die Elektronik des Experimentalaufbaus

Beim Bau der Elektronik wurde vor allem auf fertige Schaltpläne zurückgegriffen. Einige Details sorgten dafür, dass dieser Teil an der Arbeit einige Tage Zeit verschlang. Im Folgenden werden kurze Erläuterungen zu den Schaltplänen gegeben.

Das Herz der Ansteuerung ist ein einfacher ATmega48 8bit RISC Mikrocontroller (Abbildung 19). Er ist die Schnittstelle für Sensorik und Aktuatorik. Ein Teil der Steuerlogik ist im Experimentalaufbau auf den PC ausgelagert. So ist es leichter durch künstliche Evolution die geeigneten Steuerparameter zu finden. Im Prinzip kann eine fertige Steuerung mit dem ATmega48 allein funktionieren. In dem Experimentalaufbau dient der Microcontroller eher als Schnittstelle für den PC.

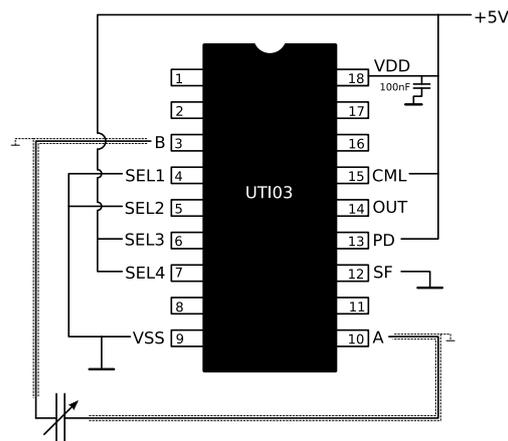


Abbildung 18: Schaltplan für den UTI03

Um den aktuellen Winkel des Gelenkarmes auszulesen wird ein kapazitiver Winkelgeber genutzt. Da dieser berührungslos ist beeinträchtigt er die Messungen weniger als ein Potentiometer, welches zusätzliche Reibungskräfte verursachen würde. Eine weitere Alternative wäre ein optischer Winkelgeber. Durch die zur Verfügung stehenden Mittel empfahl sich die kapazitive Variante eines Winkelsensors. Für die Erfassung

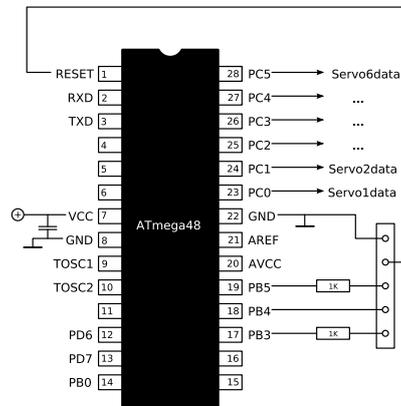


Abbildung 19: Schaltplan mit 8 bit-Prozessor ATmega48

der aktuellen Kapazität des Winkelgebers wird hier ein UTI03 benutzt. Dieser Al-leskönner besitzt einen Betriebsmodus in dem eine Kapazität im Bereich von 0-12 pF kontinuierlich gemessen und ähnlich einer Pulsweitenmodulation in ein digitales Signal umgewandelt wird. Allerdings wird nicht die Pulsweite bei konstanter Periodenlänge moduliert. Stattdessen dient die Zeitdauer zwischen zwei Pegelwechseln als Signal. Leider kann man in diesem Modus nur zwischen zwei, hier mäßig geeigneten, Messgeschwindigkeiten wählen. Entweder die Messdauer war mit durchschnittlich 300 ns zu kurz, weil der zum Pegelwechsel gehörige Interrupt zu oft ausgelöst worden wäre oder mit über 20 ms zu lang. Dennoch wurde die zweite Variante genutzt, da sie in diesem Fall weniger nachteilig ist. Besonders wichtig ist dass die Kabel zwischen jeweils einer Kondensatorplatte mit dem UTI03 abgeschirmt werden. Den Unterschied erkennt man in Abbildung 20. Es wurde eine Sinusförmige Bewegung der Servos angesteuert und dabei ohne Rückkopplung auf die Steuerung der Winkelgeber ausgelesen. Die Unterschiede in der Amplitude sind ein unschöner Nebeneffekt, der eigentliche Fehler sind die Dellen bei fehlender Abschirmung. Für eine serienmäßige Anwendung empfiehlt sich auch den Winkelgeber mit einer elektrischen Abschirmung zu kapseln.

Um die Beschleunigungswerte am Ende des Gelenkarmes zu messen benutzen wir den

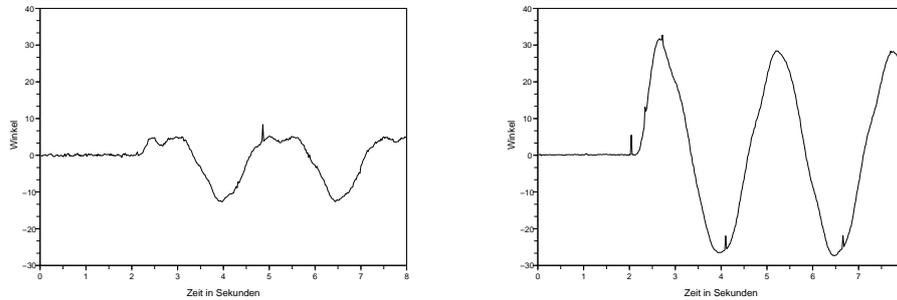


Abbildung 20: Sensorwerte des Winkelsensors bei angesteuerter Sinuskurve links ohne Kabelabschirmung, rechts mit Abschirmung

ADXL213, einen zweiachsigen Beschleunigungssensor der im Bereich von $\pm 1,2$ g misst und die Messwerte in ein pulswidenmoduliertes Signal (im Folgenden kurz PWM) umwandelt. Man kann die Periodenlänge extern über einen Widerstand einstellen. Wir wählten eine Dauer von 4ms.

7.2 Die Firmware des Experimentalaufbaus

Die Firmware für den Mikrocontroller hat im Wesentlichen drei Aufgaben zu erfüllen. Zum einen müssen die Sensordaten des Experimentalaufbaus erfasst werden, zum anderen müssen die Servos über ein PWM angesteuert werden. Die dritte Aufgabe besteht darin die Sensorwerte dem PC zu übermitteln und die Steuerparameter für die Servos vom PC entgegen zu nehmen. Der PC wird eingesetzt um eine Steuerfunktion für den Experimentalaufbau zu entwickeln. Die Parameter der Funktion sollen über das Mittel der künstlichen Evolution ermittelt werden. Dass eine Onboardvolution ebenso funktionieren kann zeigt z. B. die Studienarbeit[6] von Ferry Bachmann zu genau diesem Thema.

Äußerst wichtig ist eine gute zeitliche Koordinierung der Abläufe. Den Rahmen bildet die Vorgabe die Servos mit einem PWM mit einer Periodenlänge von 20 ms, beziehungsweise einer Grundfrequenz von 50 Hz, anzusteuern. Pulsbreite von 1-2 ms entspricht der Winkelposition des Servos von minimalen bis maximalen Winkel (ca. $\pm 100^\circ$ um die Nullposition). Die 20 ms können in fünf Zeitfenster von je 4 ms geteilt werden, dann ist es leichter mit den beschränkten Ressourcen des ATmega48 zu Arbeiten. Somit ergibt sich ein Grundtakt von 50 Hz und fünf Zeitfenster in denen verschiedene Aufgaben erledigt werden.

Der Controller verfügt über zwei 8bit Timer und einen 16bit Timer. Letzterer wird eingesetzt um die Signale der Sensoren auszumessen und damit in Zahlenwerte umzusetzen. Die Erfassung der Sensorwerte ist unabhängig von den vorgegebenen Zeitfenstern. Die 8bit Timer werden benutzt um die Zeitfenster sicher zu stellen und die PWM für die Servos zu generieren. Der Timer, der die Zeitfenster sicherstellt, wird mit einem Zähltakt von 62500 Hz initialisiert. Der Timer erhöht alle $16 \mu s$ den Zählerstand. Bei 250 Zählschritten entspricht das einer Laufzeit von $4 ms$. Wenn dieser Timer startet werden ggf. die steigenden Flanken des PWM für zwei Servomotoren generiert, indem die entsprechenden Pins auf den hohen Pegel gesetzt werden. Der zweite 8bit Timer sichert die fallenden Flanken. Dieser Timer zählt ebenfalls 250 Schritte innerhalb von $4ms$. Bei Erreichen der Zählzeit entsprechend der Pulsbreite, werden die Pins

wieder auf niedrigen Pegel gesetzt. Mit den gewählten Parametern werden die Servos umgerechnet 2° genau angesteuert, das entspricht der realen konstruktionsbedingten Genauigkeit der Servos.

Die Messwerte der Sensoren liegen als wertediskrete Signale an dem ATmega48 an Pin 12 und Pin 14 an. Diesen Pins sind zwei verschiedene Pinchangeinterrupts zugeordnet. Die Maskierung für die Pinchangeinterrupts wird so angepasst, dass genau diese zwei Pins die Interrupts auslösen können. Die Pinchangeinterrupts dienen der Zeitmessung der Sensorsignale. Dies ist für die Signale des UTI besonders leicht, weil die Zeit zwischen zwei Pegelwechseln des Signals vom UTI proportional zur gemessenen Kapazität ist. Der Beschleunigungssensor übermittelt die Sensorwerte als PWM und daher müssen erst Differenzen gebildet werden. Innerhalb eines Zeitfensters von 20 ms treffen vier bis fünf Beschleunigungswerte ein, asynchron zu der Ansteuerung der Motoren. Ein einfacher Medianfilter dritter Ordnung bereinigt das Signal von Ausreißern. So liegt in jedem 20 ms Zeitfenster ein aktueller Beschleunigungsmesswert vor, mit dem der Controller auf PC-Seite rechnen kann. Mit dem Winkelmesswert verhält sich das etwas anders. Teilweise dauert es länger als 20 ms zwischen zwei Pegelwechseln des Signals. Praktisch wirkt sich das nicht merklich aus, wie sich in den Experimenten zeigte.

Das Verhalten der Timerinterruptroutine ist also Abhängig davon welches Zeitfenster gerade bedient wird. Eine einfacher Zustandsautomat sorgt für die Regelung (vgl. Abbildung 21).

In den ersten drei Zeitfenstern werden die Servos angesteuert, im vierten Zeitfenster werden die Sensorwerte an den PC geschickt und die neuen Steuerwerte für die Servos empfangen. Abbildung 22 verdeutlicht noch einmal schematisch das Timing der Firmware.

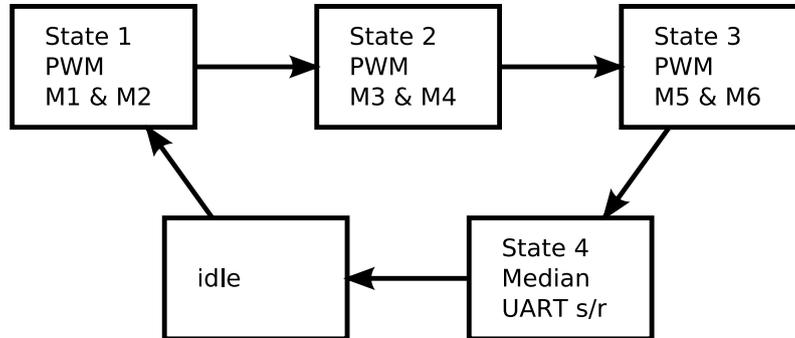


Abbildung 21: Skizze zum Timing als Zustandsautomat

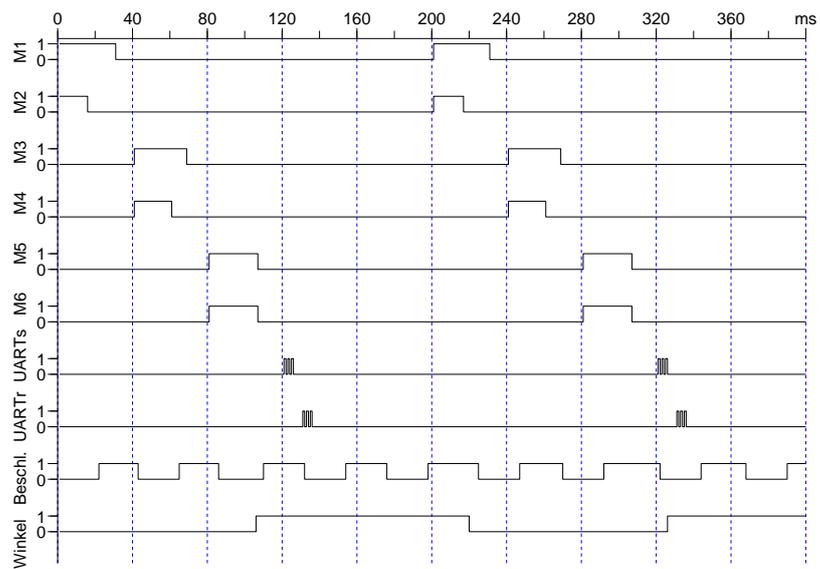


Abbildung 22: Schematische Skizze des Timings in der Firmware

7.3 Die PC-Software des Experimentalaufbaus

Die PC-Software hat viele verschiedene Aufgaben zu erledigen. Die wichtigste ist, dass sie als ausgelagerter Controller für die Servomotoren dienen muss. Die Vorteile gegenüber einem Controller auf dem Atmel sind die weitaus höhere Rechenleistung und der größere Datenspeicher des PCs, die besonders bei der künstlichen Evolution benötigt werden. Diese ist ein weiterer Bestandteil des Programms. Ferner muss der Winkelgeber kalibriert werden können und verschiedene Test- und Loggingfunktionen zur Verfügung gestellt werden. Das Framework für die serielle Kommunikation, zwischen dem Mikrocontroller und dem PC, wurde aus dem Programm SML übernommen und erweitert. Dieses wurde von Christian Thiele im Zuge seiner Studienarbeit [7] entworfen.

Das Framework

Zur Organisation des PCs und des Mikrocontrollers sind bestimmte Regeln notwendig, diese werden durch das Framework definiert. Damit der PC nicht unkontrolliert Steuerungsdaten des Controllers sendet, muss er sich beim Timing dem Mikrocontroller unterordnen. Der Mikrocontroller sendet alle 20ms ein Datenpaket, dass vom PC erwartet und anschließend ausgelesen wird. Das Datenpaket hat immer dasselbe Format. Am Anfang werden 2 Synchronisationsbytes übertragen, gefolgt von 2 Bytes mit Beschleunigungswerten und 2 Bytes mit Werten von dem Winkelsensor. Erst wenn das Datenpaket übertragen wurde, hat der PC knapp 20ms Zeit um eventuelle Berechnungen durchzuführen. Am Ende des Timeslots sollte der PC seine neu berechneten Winkelstellungen für die Servomotoren zurück an den Mikrocontroller übertragen. Dies sollte möglichst nur einmal geschehen, da mehrmaliges senden keinen Vorteil für die Ansteuerung bedeutet, allerdings die Rechenleistung des Mikrocontrollers durch unnötige Interrupts mindert. Um falsche Daten, die zu unkontrollierten Servobewegungen führen könnten, zu vermeiden, werden nach einem Synchronisationsbyte und insgesamt 6 Datenbytes für die Winkelstellungen, noch zusätzlich eine Checksumme übertragen. Falls der PC keine neuen Daten sendet oder diese falsch übertragen wur-

den, werden die alten Winkelstellungen aus dem letzten Timeslot beibehalten.

Die Kalibrierung des Winkelgebers

Der in Eigenproduktion hergestellte Winkelgeber liefert keine linearen Messwerte, das liegt an der ungenauen Fertigung der Kondensatorplatten. Daher muss eine Kalibrierung des Sensors vorgenommen werden. Des Weiteren sollen die maximalen Ausschläge, die der Gelenkarm bis zu den Anschlägen links und rechts anfahren kann, bestimmt werden. Der erste Schritt der Kalibrierung besteht durch das langsame Anfahren des rechten Anschlags, dies geschieht durch langsames weiterbewegen bis der Winkelsensor keine Änderung mehr feststellt. Von der jetzt erlangten Position fährt der Gelenkarm langsam bis zum linken Anschlag, nach derselben Methode. Während des Abfahrens werden die jeweils aktuellen Winkelsensorwerte mit den angefahrenen Winkelwerten der Servomotoren in Verbindung gesetzt, dazu wird der Gelenkarm eine längere Zeit an der selben Position gehalten und die Werte des Winkelsensors gemittelt. Anschließend wird der ermittelte Wert in eine Look-Up-Table geschrieben. Aufgrund der Diskrepanz zwischen der Auflösung der Servomotoren und der des Winkelsensors, der Sensor arbeitet genauer, müssen die Lücken in der Look-Up-Table geschlossen werden. Dazu werden die vorhandenen Werte einfach linear interpoliert.

Der Controller

Der Controller stellt das Herzstück des auf Rechnerseite arbeitendem Programms dar. Es müssen, je nachdem wie viele Servos angeschlossen sind, bis zu sechs Winkelstellungen alle 20 ms berechnet werden. Als Eingangsdaten stehen dem Controller die Beschleunigungs- und Winkelwerte zur Verfügung. Jede der Winkelstellungen werden durch einen eigenen PID-Controller berechnet. Genaueres wird bei den jeweiligen Experimenten erläutert.

Die künstliche Evolution

Die Evolution wird zur Ermittlung der Gewichte der PID-Controller benötigt. Das Programm stellt alle Funktionen einer künstlichen Evolution zur Verfügung. Hierzu

gehören, die Erstellung einer Startpopulation, die Selektion/Kreuzung/Mutation und die Bereitstellung einer Fitnessfunktion, entsprechendem dem Standardverfahren nach Nolfi und Floreano [8]. Dies ist der entscheidende Faktor einer Evolution. Bei den Experimenten wird noch näher auf die Fitnessfunktion und die verwendeten Parameter für die Mutation und Selektion eingegangen. Des Weiteren werden alle während der Evolution durchgeführten Test in einer Logdatei gespeichert. Um sie später fortsetzen zu können. Die Evolution kann schließlich, abhängig von Populationsgröße und Dauer der einzelnen Tests, viele Stunden in Anspruch nehmen kann. Bei der Wahl der Parameter wurde der Rat von Manfred Hild in Anspruch genommen, sowie auf die Eigenen Erfahrungen mit künstlicher Evolution zurückgegriffen.

7.4 Arbeitsaufteilung

Die Studienarbeit ist eine Gemeinschaftsproduktion und die Arbeit daran wurde zu möglichst gleichen Anteilen von den Autoren übernommen. Selbstverständlich ist nicht jeder kleine Teil nur genau einem Autor zu zuweisen. Dieser Abschnitt soll eine ungefähre Aufteilung widerspiegeln.

Im praktischen Teil hat sich Thomas Lobig vor allem um den mechanischen Aufbau des Experiments gekümmert und Lötarbeiten durchgeführt, Christian Benckendorff hat sich vor allem um die Programmierung gekümmert. Den Großteil der Vorrecherchen hat Thomas Lobig übernommen.

Bei der Ausarbeitung war Christian Benckendorff für folgende Kapitel der Hauptautor: 5.1.2 und 5.1.3, 5.2, 5.3, 6.1, 7.3.

Thomas Lobig war Hauptautor der Kapitel: 1 bis 4, 5.1.1, 5.4, 6.2, 7.1, 7.2.

7.5 Verwendete Software

Die vorliegende Arbeit wurde mit \LaTeX erstellt. Hierfür kamen die MiKTeX-Distribution[9] für Windows und der Editor TeXnicCenter[10] zum Einsatz. Für die Erstellung von Vektorgrafiken kam das Programm Inkscape[11], für Graphen SciLab[12], zum Einsatz. Programmiert wurde mit Microsoft Visual Studio[13] und WinAVR[14], die Firmware für den Atmelprozessor wurde mit At-Prog [15] überspielt, außerdem wurden damit die Fusebits gesetzt.

Danksagung

An der Verwirklichung dieser Arbeit wirkten einige Menschen mit, denen wir uns allen zu Dank verpflichtet fühlen. Wir möchten an dieser Stelle nur eine beschränkte Auswahl hervorheben. Manfred Hild, der Betreuer dieser Arbeit, stand uns für fachliche Fragen mit gutem Rat zur Seite. Torsten Siedel hat uns beim Aufbau der Mechanik sehr geholfen. Matthias Kubisch hat bei einer Durchsicht der Arbeit viele gute Hinweise gegeben, welche die Qualität der Arbeit zu verbessern geholfen haben.

Literatur

- [1] Werner Nachtigall, „Biomechanik“, 2. Auflage 2001, Vieweg Verlag
- [2] <http://www.lara-robot.de/> (2.11.2007)
- [3] <http://www.uni-saarland.de/fak8/bi13wn/projekte/roboarm.html> (30.10.2007)
- [4] <http://www.biorob.de/> (30.10.2007)
- [5] Manfred Hild, „Neurodynamische Module zur Bewegungssteuerung autonomer mobiler Roboter“, Dissertation 2007 an der Humboldt Universität zu Berlin
- [6] Ferry Bachmann, „Onboard-Evolution mit der Fahrenden Platine“
- [7] Christian Thiele, „Integrierte Entwicklungsumgebung zur Bewegungssteuerung humanoider Roboter“, Studienarbeit Oktober 2007, Lehrstuhl für Künstliche Intelligenz, Humboldt Universität zu Berlin
- [8] S. Nolfi und D. Floreano, „Evolutionary Robotics“, 2000, MIT Press
- [9] <http://miktex.org/> (30.10.2007)
- [10] <http://www.toolscenter.org/> (30.10.2007)
- [11] <http://www.inkscape.org/> (30.10.2007)
- [12] <http://www.inkscape.org/> (30.10.2007)
- [13] <http://www.microsoft.com/germany/msdn/vstudio/products/default.mspx>
(30.10.2007)
- [14] <http://winavr.sourceforge.net/> (30.10.2007)
- [15] <http://www.speedy-bl.com/avr-prog.htm> (31.10.2007)
- [16] <http://www.humanoidteamhumboldt.de>

Die Datumsangaben hinter Webseiten entsprechen dem Datum an dem zuletzt die Erreichbarkeit der betreffenden Seite überprüft wurde.