



TU Berlin, Sekr. EN11
FG Regelungssysteme
Sokr. EN11
Einsteinufer 17
D-10587 Berlin

FAKULTÄT IV, E-Technik / Informatik
Inst. für Energie- und Automatisierungstechnik
FG Regelungssysteme
AG Nachgiebige Systeme

Diplomarbeit

Mit dem Thema:

**Entwicklung, Optimierung des Modells eines muskelbetriebenen
tripedalen Segmentes eines Roboterrüssels in Matlab/SIMULINK**

vorgelegt von

Cand.-Ing.

Haibei Feng (215601)

Hauptprüfer:

Prof. Dr.-Ing. Jörg Raisch

Betreuer:

Dipl.-Ing. Ivo Boblan

7. März 2011



Eidesstattliche Erklärung

Die selbstständige und eigenhändige Anfertigung der
vorliegenden Diplomarbeit versichere ich an Eides statt.

Berlin, den 2011

.....

Feng Haibei

Danksagung

Mein herzlicher Dank geht an das Forschungsprojekt, das mich mit dieser Forschungsarbeit betraut hat. Dadurch haben sich nicht nur meine Kenntnisse, die ich an der Universität gewonnen habe, konsolidiert, sondern es wurde auch meine Niveau in dem Wissensgebiet gehoben. An dieser Stelle möchte ich mich besonders bei meinem Betreuer, Herrn Ivo Boblan, bedanken, der mich während meiner Diplomarbeit zeitintensiv betreut und viele Denkanstöße gegeben hat.

Feng Haibei

Inhaltsverzeichnis

1 Einleitung.....	9
1.1 Aufgabenbeschreibung	10
2 Vorbetrachtungen	12
2.1 Herkömmliche kinematische Konstruktionen	12
2.2 Tripedale Alternanzkaskade TAK.....	13
2.2.1 Aufbau	13
2.2.2 Zielsetzung.....	18
3 Modellierung	20
3.1 Mathematische Beschreibung der geometrischen Verhältnisse der TAK.....	20
4 Implementierung.....	31
4.1 Algorithmus unter Matlab	31
4.2 SIMULINK und VR-Toolbox	32
4.3 Kopplung MATLAB mit V Realm Builder.....	34
5 Simulation und Auswertung	36
5.1 Geometrische Zusammenhänge	36
5.2 Optimaler Knickwinkel.....	38
5.3 3D-Modell.....	40
6 Zusammenfassung und Ausblick	46
6.1 Zusammenfassung.....	46
6.2 Ausblick.....	47
7 Literatur	48

Abbildungsverzeichnis

ABBILDUNG 1 DARSTELLUNG VON MUSKELSEGMENT IN AUTO CAD	14
ABBILDUNG 2 PNEUMATISCHE MUSKELN DER FIRMA FESTO	15
ABBILDUNG 3 INNER AUFBAU DES PNEUMATISCHEN MUSKELS	15
ABBILDUNG 4 BEMAßUNG VOM PROFIL EINES DREIARMIGEN STERNS	16
ABBILDUNG 5 BEMAßUNG VOM MUSKEL IN SEITENANSICHT	17
ABBILDUNG 6 BEMEßUNG VOM DRUCKSTAB IN SEITENANSICHT.	18
ABBILDUNG 7 LINKE ANSICHT BEIM STRECKENDEN MUSKEL.....	22
ABBILDUNG 8 RECHTE ANSICHT BEIM STRECKENDE MUSKEL.	24
ABBILDUNG 9 KONLISIONSDARSTELLUNG VON MITTLEREN STERNARME BEIM SICH STRECKENDEN MUSKEL	25
ABBILDUNG 10 LINKE ANSICHT BEIM SCHRUMPFEN MUSKEL.....	27
ABBILDUNG 11 KONLISIONSDARSTELLUNG VON MITTLEREN STERNARME BEI DEM SCHRUMPFEN MUSKEL	27
ABBILDUNG 12 RECHTE ANSICHT BEIM SCHRUMPFENDEN MUSKEL	28
ABBILDUNG 13 EREIGNIS DURCH DOPPELKLICK AUF VR SINK BLOCK.....	34
ABBILDUNG 14 DIALOGFENSTER ZUR VRML	35
ABBILDUNG 15 ERGEBNISDARSTELLUNG DES PROGRAMMS STRECKEN.M.....	37
ABBILDUNG 16 3D-STABMODELL MIT MAXIMALE GELENKDREHWINKEL BEI H=300MM.....	38
ABBILDUNG 17 DARSTELLUNG DES MAXIMALEN DREHWINKELS AUS VERSCHIEDENEN K _S -WERTE BEI UNTERSCHIEDLICHEM R ₁ /R ₂ - VERHALTEN	39
ABBILDUNG 18 EULERSCHE WINKEL IN ZXZ-KONVENTION	41
ABBILDUNG 19 EINE AUFNAHME AUS DER KONTROLLEBENE DER SIMULATION	43
ABBILDUNG 20 SIMULATION MIT NEU EINGEGEBENEM PARAMETER.....	44

Tabellenverzeichnis

TABELLE 1 MESSWERT FÜR DREIARMIGEN STERN.	16
TABELLE 2 BEMAßUNG FÜR DEN MUSKELN.	17
TABELLE 3 BEMAßUNG FÜR DAS DRUCKSTAB	18

Abkürzungsverzeichnis

Im folgenden werden die wichtigsten in der Arbeit verwendeten Formelzeichen aufgelistet und erklärt.

Abkürzung:

.mdl	Endung einer SIMULINK-Datei
.WRL	Endung von Software V-Realm Builder
.m	Endung einer MATLAB-Datei
MATLAB	MATrix LABoratory
SIMULINK	Dynamic System Simulation for MATLAB
VR	Virtual Reality
TAK	Tripedale Alternankaskade
VRML	V irtual Reality Modeling Language
CAD	computer-aided design

Symbol	Größe, Bezeichnung	Einheit
L_0	15% verreckende pneumatische Muskelnlänge.	[mm]
L_{15}	ungeänderte pneumatische Muskelnlänge.	[mm]
L_{30}	15% verkürzende pneumatische Muskelnlänge.	[mm]
$L_{L_strecken}$	linke Muskellänge beim linksstehenden sich streckenden Muskel.	[mm]
$L_{R_strecken}$	rechte Muskellänge beim linksstehenden sich streckenden Muskel.	[mm]
$L_{L_schrumpfen}$	linke Muskellänge beim linksstehenden sich verkürzenden Muskel.	[mm]
$L_{R_schrumpfen}$	rechte Muskellänge beim linksstehenden sich verkürzenden Muskel.	[mm]
H:	die gesammte Höhe der Anfangzustand.	[mm]
S:	obere Druckstabslänge.	[mm]
R1:	das Radius von oberem dreiarmigem Stern.	[mm]

R2:	das Radius von unterem dreiarbigem Stern.	[mm]
R1/R2:	das Radiusverhältnis.	[]
α :	Steigung der Anfangzustand der Muskel zur X-Ebene.	[°]
β :	Schnittwinkel zwischen g1 und S.	[°]
ϑ :	Drehwinkel des oberen Druckstabs	[°]
ϑ_{R_max} :	maximaler Drehwinkel des oberen Druckstabs nach rechts	[°]
ϑ_{L_max} :	maximaler Drehwinkel des oberen Druckstabs nach links	[°]
φ :	Steigung der Endzustand der Muskel zur X-Ebene.	[°]
g1:	Hypotenuse von derGeraden des oberen Radius R1 und oberen Dreuckstabs S	[mm]
g2:	Hypotenuse von derGeraden des unteren Radius R2 und unteren Dreuckstabs	[mm]
t2:	Die Senkrechte zwischen dem Kugelgelenk und gestreckende Muskeln.	[mm]
t3:	Abstand zwischen dem mittleren Sternarm und gestreckende Muskeln.	[mm]

1 Einleitung

In diesem Kapitel werden kurz Sinn und Nutzen des flexiblen allgemeinen Roboterarms beschrieben. Am Ende der Einleitung wird der eigentliche Reihenfolge dieser Diplomarbeit angesprochen.

Roboter sind heute unverzichtbare Werkzeuge für die flexible Fertigungsautomatisierung in vielen Bereichen der Industrie sowie für die Durchführung anspruchsvoller oder gefährlicher Missionen in der Kern- und Weltraumtechnik. Zunehmend werden Roboter auch in der Medizin, in verschiedenen Bereichen der Bauwirtschaft, bei der Inspektion und Sanierung von Kanälen und Rohrleitungen sowie in der Landwirtschaft eingesetzt.

In den Bereichen Robotik und Medizin bietet die Entwicklung eines kinematischen Modells für einen anthropomorphen Roboterarm eine wichtige Grundlage mit interessanten Perspektiven. Im Bereich Robotik ist das Modell Ausgangspunkt für die Entwicklung einer neuartigen, anthropomorphen Robotergeneration für Arbeitsumgebungen, die auf den Menschen abgestimmt sind, und für Aufgaben, bei denen die Kooperation von Mensch und Roboter im Vordergrund steht. Im Bereich Medizin bietet das Kinematikmodell zudem eine gute Grundlage für die Entwicklung von Prothesen oder es dient als Demonstrator für die Bewegungen des menschlichen Arms.

Für Anwendungsgebiete, in denen Roboter und Menschen kooperieren oder in denen Roboter vom Menschen lernen, eignet sich insbesondere die anthropomorphe Kinematik.

Weitere Anwendungsbereiche für das Kinematikmodell sind Biomechanik und Medizin. An dem Kinematikmodell können die Bewegungszusammenhänge des menschlichen Arms demonstriert werden. Aufbauend auf dem Modell können auch die Belastungszustände und das Zusammenspiel der Muskeln erforscht werden. Wichtig ist auch hier, dass das Modell den physiologischen Bewegungen des menschlichen Arms angepasst ist und dass die kinematisch

wirksamen Längen und Winkel eingehalten werden. Damit trägt ein solches Modell auch zur Entwicklung von Prothesen und künstlichen Gelenken bei.[1]

Für die Erledigung der gestellten Aufgabe wurden ein vom künstlichen Arm abgeleitetes kinematisches Modell, verschiedene Methoden zur Arbeitsraumanalyse und eine Softwarearchitektur für Roboter entwickelt. Jeder dieser Bereiche trägt zur Entwicklung einer neuartigen, auf den Menschen abgestimmten Robotergeneration bei.

1.1 Aufgabenbeschreibung

Ein Rüssel ist eine verlängerte Nase mit Nasenlöchern. Bei Elefanten ist der Rüssel ein äußerst feinfühliges und langes Organ, das im Lauf der Entwicklung aus Oberlippe und Nase entstand. Etwa 40.000 zu Bündeln verflochtene Muskeln machen den Rüssel sehr beweglich. Der Rüssel enthält kein Nasenbein oder andere Knochen. Er besteht ausschließlich aus Muskelgewebe und ist das auffälligste anatomische Merkmal der Elefanten.

Ziel des BROMMI Projektes ist die Entwicklung, Aufbau und Erprobung eines flexiblen Roboterarms nach dem biologischen Vorbild des Elefantenrüssels zur sicheren Anwendung in der Mensch-Roboter-Interaktion. Auf der Basis von bionischen Prinzip des muskulären Hydrostaten realisiert, d.h. nach dem bionischen Vorbild eines Elefantenrüssels mit seiner flexibel einstellbaren Steifigkeit. Die Konstruktion und der Aufbau des flexiblen Roboterarms bzw. Manipulators erfolgt in Leichtbauweise unter Verwendung von pneumatischen Muskeln und Objekterkennung per Bildverarbeitung zum Greifen und Positionieren von Objekten. An diesem Forschungsfeld habe ich teilgenommen und Erkenntnisse für die künftige Weiterentwicklungen gewonnen.

Diese Arbeit ist wie folgt aufgebaut: Kapitel 2 beschreibt zunächst den Aufbau der TAK und ihre Elemente des Roboterarms in geometrischer Darstellung, das für die Modellierung und Simulation vom Roboterarmsegment im 3D-Modell sowie die Darstellung im Stab-Modell zur Verfügung steht. Anschließend Kapitel 3 wird eine Analyse der Parametrisierung vorgenommen, welche Möglichkeiten der Kollision es gibt. Dann werden die mathematischen Methoden vermittelt, die für die Kollisionsberechnung angewendet sind. Im Anschluss des Kapitels 3 wird die Vorbeugung der Kollision für Programm erläutert. Mit Hilfe dieses Verfahrens werden dann die optimale Parameter für den anthropomorphen Roboterarm berechnet. In darauffolgendem Kapitel 4 werden die Implementierung von Algorithmen unter MATLAB beschrieben. Im ersten Teil wird das Algorithmus und Softwares MATLAB, SIMULINK, VR-Toolbox und V Realm Builder ausführlich geschildert. Im zweiten Teil wird die Kooperation zwischen SIMULINK und V Realm Builder kurz eingeführt. Zunächst wird das Kapitel 5 das Simulationsergebnis gezeigt und eine Analyse der Parametrisierung erklärt.

Die Arbeit wird in Kapitel 6 mit der Zusammenfassung abgeschlossen. Dabei werden sowohl die erzielten Ergebnisse der Arbeit beschrieben als auch ein Ausblick auf die zukünftige Erweiterungen aufbauend auf diese Ergebnisse möglich gegeben.

Der inhaltliche Kern der Diplomarbeit ist das Modellentwurfsverfahren unter der Variation der relevanten Parameter und schließlich die Geometriedarstellung im Raum übertragen.

2 Vorbetrachtungen

In diesem Kapitel wird die Beschreibung sowohl der Standardtechnik, als auch des TAK-Bestandteil eingeführt, die mit dem Forschungsprojekt relevant sind.

Technisch realisiert werden Roboter hauptsächlich im Zusammenspiel der Disziplinen: Mechanik, Elektrotechnik und Informatik. Inzwischen hat sich aus der Verbindung dieser drei Disziplinen die Mechatronik entwickelt. Um autonome Systeme zu entwickeln, die eine gewisse Eigenständigkeit, beispielsweise bei der Wegfindung aufweisen, werden immer mehr wissenschaftliche Disziplinen in die Robotik eingebunden. Hier liegt ein Schwerpunkt bei der Verbindung von Konzepten der Künstlichen Intelligenz oder der Neuroinformatik (Teilgebiete der Informatik) und ihrer biologischen Vorbilder Biokybernetik (Teilgebiet der Biologie). Aus der Verbindung von Biologie und Technik entstand wiederum die Bionik.

2.1 Herkömmliche kinematische Konstruktionen

Die herkömmliche Roboter sind als Industrieroboter gezeichnet und nach der VDI-Richtlinie 2860 universell einsetzbare Bewegungsautomaten mit mehreren Achsen, deren Bewegungen hinsichtlich Bewegungsfolge und Wegen bzw. Winkeln frei programmierbar und ggfs. sensorgeführt sind. Sie sind mit Greifern, Werkzeugen oder anderen Fertigungsmitteln ausrüstbar und können Handhabungs- und/oder Bearbeitungsaufgaben ausführen.

Der Industrieroboter besteht im wesentlichen aus 4 Teilsystemen oder Funktionsgruppen: Die **Kinematik**, hierunter versteht man den Aufbau und die Anordnung der Achsen, Führungen, Gelenke, die die Bewegungsmöglichkeiten des Gerätes bestimmen. Auch ein Greifer oder Werkzeuge am Industrieroboter zählen hierzu. Der **Antrieb**, zum Antrieb der Achsen dienen Motoren und Getriebe, ggf. Auch Ventile und Zylinder. Insbesondere elektrische, aber auch hydraulische und pneumatische Antriebe kommen zur Anwendung. Die **Steuerung**, Sie ist die

Einheit, die das Programm aufnimmt, den Ablauf vorgibt und überwacht sowie nach Bedarf bzw. Auslegung des technischen Gesamtsystems den Kontakt mit anderen Maschinen und Einrichtungen herstellt und aufrecht erhält. Das **Meßsystem**, durch dieses System wird die Stellung der einzelnen Achsen, ihr Weg und ihre Geschwindigkeit ermittelt.[2]

Wichtigste Bestandteile eines Roboters sind die Sensoren, zur Erfassung der Umwelt und der Achspositionen, die Aktoren, zum Agieren innerhalb der erfassten Umgebung, die Robotersteuerung und das mechanische Gestell inklusive der Getriebe des Roboters. Ein Roboter muss nicht unbedingt vollständig autonom handeln können, darum unterscheidet man autonome und ferngesteuerte Roboter.

2.2 Tripedale Alternanzkaskade TAK

Es folgt ein Überblick über die gesamte TAK-Darstellung sowie die Beschreibung aller Einzelbauteile in der Geometrie-Untersuchung

2.2.1 Aufbau

Die folgende Abbildung 1 ist eine Gesamt-Ansicht des Muskelsegments des Roboterarms in CAD-Zeichnung. Dieses Muskelsegment besteht grundsätzlich aus einem Hauptteilelement bezüglich pneumatischen Muskels, dreiarmigen Sternes, Kugelgelenks und Druckstabs .

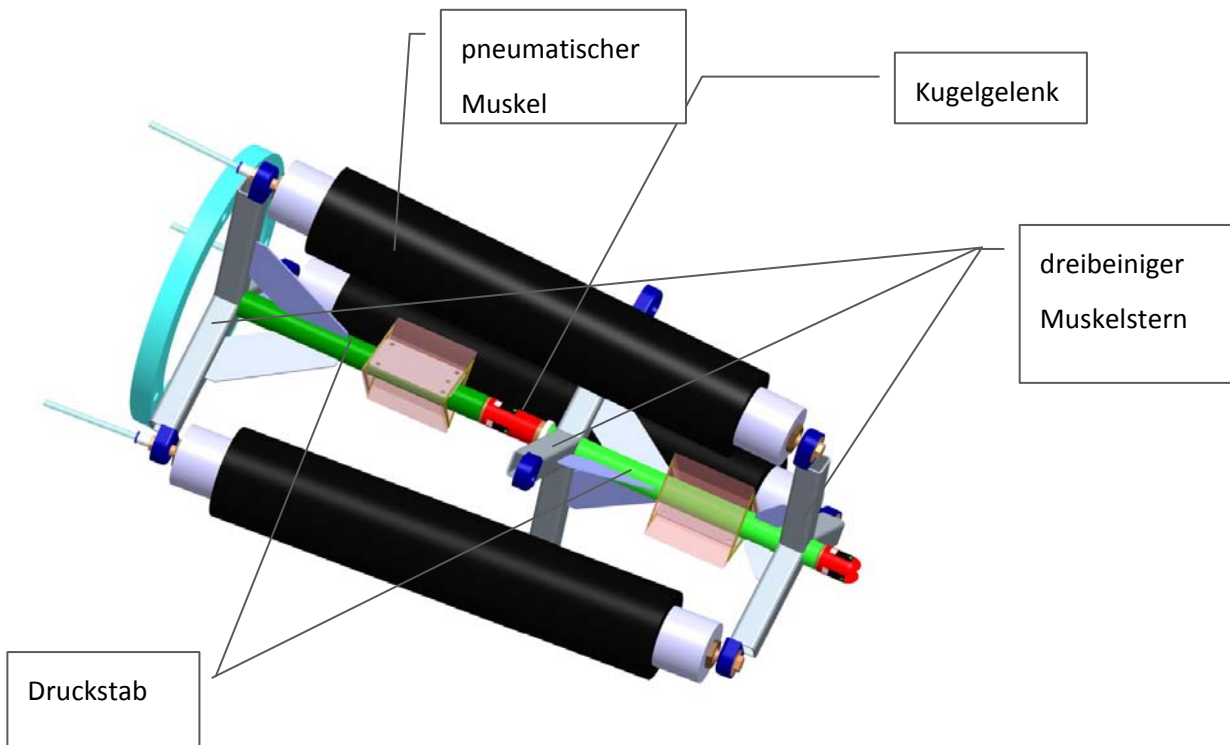


Abbildung 1 Darstellung von Muskelsegment in Auto CAD

Die Abbildung 1 zeigt das Basismuskelsegment. Darauf sind drei schwarzen zylindrischen Muskeln und ein Druckstab zwischen zwei Sterne fixiert. Auf dem Druckstab existieren zwei Kugelgelenke und ein Stern. Das linksstehende Kugelgelenk lässt das Rechtsteil des Druckstabs drehbar durch Variation der Muskellänge. Rechts neben dem Gelenk steht ein Stern, der die gleiche Armlänge wie sein übereinander verschränkende Nachbar auf der linken Seite hat und als Bauelement für das nächste Segment ist. Das rechtsstehende Kugelgelenk steht auch für das nächste Segment zur Verfügung.

Pneumatische Muskeln[3]



Abbildung 2 Pneumatische Muskeln der Firma Festo

Der pneumatische Muskel ist ein Zugaktuator, und ein geschlossenes Luftröhre, der dem biologischen Muskel nachempfunden ist. Er besteht im Wesentlichen aus einem Kontraktionsschlauch (druckdichter Gummi, umgeben von festen Fasern) und den entsprechenden Anbindungsstücken, wie in Abbildung 2 dargestellt. Durch Erhöhung des Innendrucks dehnt sich der Schlauch in radialer Richtung aus. Somit entsteht eine Kontraktionsbewegung in Längsrichtung des Muskels. Demzufolge kann er als eine Art ‚pneumatische Feder‘ angesehen werden, die bei zunehmendem (abnehmendem) Druck zusammengedrückt (ausgedehnt) wird.



Abbildung 3 Inner Aufbau des pneumatischen Muskels

Die Nachgiebigkeit der pneumatischen Muskeln hat eine sehr interessante Eigenschaft. Da dieser Muskel keine starre, mechanische Vorrichtung ist, lässt sich die Interaktion mit Mensch und Umwelt einfacher realisieren. Eine allfällige Kollision mit einem Gegenstand hat dank der nachgiebigen künstlichen Muskeln eine Abfederung wie bei den natürlichen Muskeln zur Folge. Da diese Abfederung material- und morphologiebedingt ist, werden weniger Kontroll- und Steuermechanismen benötigt als mit herkömmlichen, motorbetriebenen Gelenken.

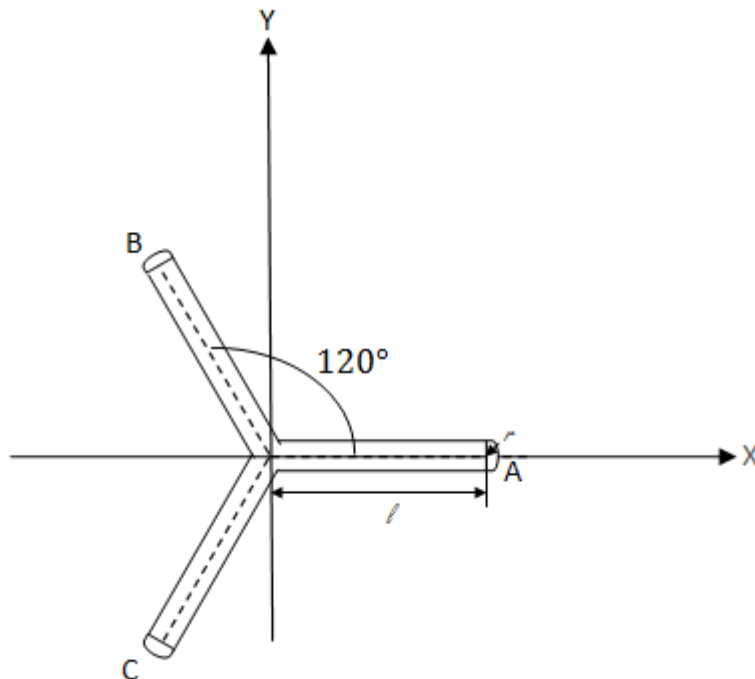


Abbildung 4 Bemaßung vom Profil eines dreiarmligen Sterns

Bemaßung:	/mm
l	20~100
r	11

Tabelle 1 Messwert für dreiarmligen Stern.

In der Abbildung 4 soll ein Profil den dreiarmligen Stern bezeichnen. Das Profil wird so gestaltet, dass jeder Ast zu seinem Nachbarn mit einem Winkel von 120° gedreht ist. Am Ende jedes Asts ist mit einem Halbkreis geformt, auf dem später der Muskel befestigt wird. Sein Radius beträgt 11 mm. Die Armlänge kann von 20 mm bis 100 mm variiert werden, weil die Armlänge in diesem Bereich untersucht wird.

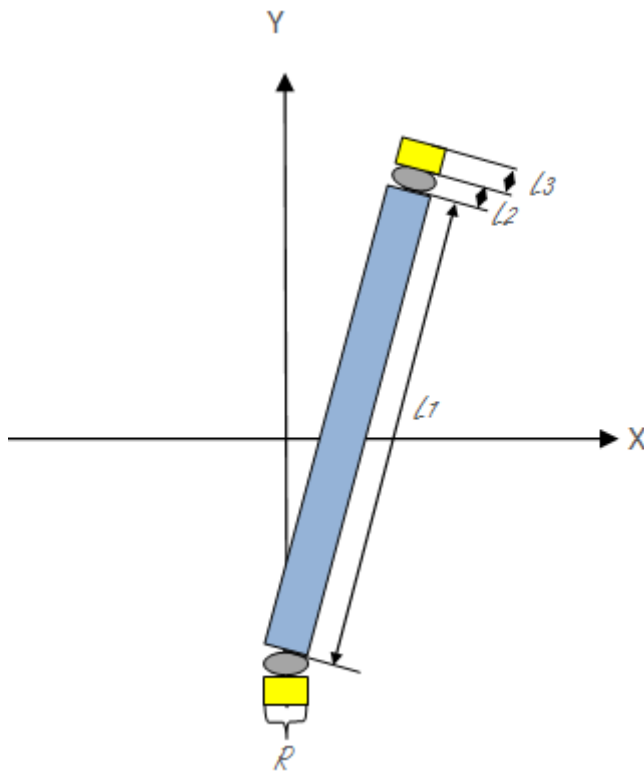


Abbildung 5 Bemaßung vom Muskel in Seitenansicht

Bemaßung:	/mm
l_1	274.492
l_2	1.6
l_3	2
R	16

Tabelle 2 Bemaßung für den Muskeln.

Um den Muskel mit einfacher Geometrie darzustellen, verwenden wir in diesem Fall in der Abbildung 5 drei Zylinder und zwei Ovale, ihre Durchmesser beträgt 16 mm. Zwei kleine Zylinder stehen am Ende der beiden Seiten des Muskels als Verbindungselement, ihre Länge beträgt 2 mm. Ein große Zylinder stellt den Muskel dar, seine Default Länge beträgt 274.492 mm . Zwei Ovale befinden sich dazwischen als Stoßdämpfer, ihre Länge beträgt 1.6 mm.

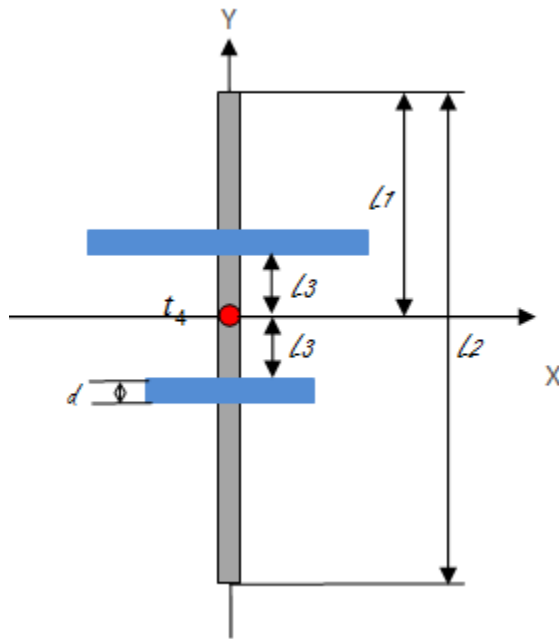


Abbildung 6 Bemaßung vom Druckstab in Seitenansicht.

Bemaßung:	/mm
L_1	138
L_2	276
L_3	18
d	12

Tabelle 3 Bemaßung für das Druckstab

Die Abbildung 6 stellt das Profil des Druckstabs dar. Das besteht aus zwei Zylindern, einem Kugel und zwei Sterne. Die Kugel wird durch ein Kugelgelenk symbolisiert und steht in der Mitte des Druckstabs in Default Zustand. Zwei Sterne werden jeweils neben dem Kugelgelenk in Abstand mit 18mm festgelegt.

2.2.2 Zielsetzung

Ziel der Diplomarbeit ist ein 3D-Modell von einem segmentiger muskelbetriebener Roboterrüssel zu entwickeln. In der Arbeit soll ein geometrisches Modell eines Segmentes inklusive der Komponenten für das jeweils angeschlossene Segment in Matlab/SIMULINK entwickelt werden. Die die Kinematik bestimmenden Parameter sind so zu implementieren,

dass in einem zweiten Schritt über eine Parametervariation optimale Konfigurationen für maximal mögliche Winkel ohne Kollisionen gefunden werden können. In einem dritten Schritt sollen die besten Ergebnisse mittels der VR-Toolbox visualisiert werden.

3 Modellierung

Im Kapitel "Modellierung" werden die mathematische Beschreibung der geometrischen Verhältnisse der TAK untersucht. Und werden weitere Kollisionserkennung und Kollisionsvorbeugung erklärt.

Die Modellierung ist ein wesentlicher Bestandteil der Simulation. Bevor ein neuer Prototyp für ein Automobil-Bremssystem oder ein Multimillionen-Dollar-Flugzeug im Feld getestet wird, ist es alltäglich, dass zu "Test Drive"-Zwecken einzelne Komponenten und das Gesamtsystem in einer simulierten Umgebung auf irgendeiner Form des Modells basieren. Ein Meteorologe prognostiziert die voraussichtliche Entwicklung eines tropischen Sturms mit Wetter-Modellen unter Einbeziehung der jeweiligen klimatischen Variablen und deren Auswirkungen auf den Sturm. Ein Ökonom gibt eine quantitative Prognose der US-Wirtschaft, die auf den wichtigsten ökonomischen Variablen und deren Zusammenhängen beruht und mit Hilfe von Computer-Modellen erstellt ist. Bevor das ein Betriebspersonal von Kernkraftwerken eingesetzt wird, wird eine umfassende Ausbildung in einem modellbasierten Simulator durchgeführt. Dazu gehört, dass der Einzelne mit den Routine- und Notfall-Evakuierungen von Zivilisten aus niedrig gelegenen Gebieten vertraut gemacht wird. Dazu werden Verkehrs-Modelle zu simulieren Fahrzeugbewegungen entlang der wichtigsten Zufahrtsstraßen entwickelt.

3.1 Mathematische Beschreibung der geometrischen Verhältnisse der TAK

Bei der Modellbildung als Grundlage des Simulationsprogrammes ist als erstes eine mathematische Beschreibung der Struktur des Muskelement erforderlich.

Im vorangegangenen Kapitel wurde gezeigt, dass sowohl die gesamte Vorrichtung als auch das einzelne Bauteil geometrisch erfassbar sind. Nachdem bisher die einzelnen Bauelemente

zum Entwurf des Roboterarms betrachtet wurden, wird nun die geometrische Analyse der Kollisionen untersucht. Erste Kollision existiert zwischen einem sich gestreckten Muskel und dem auf dem Druckstab stehenden Kugelgelenk. Zweite Kollision tritt zwischen zwei sich gestreckten Muskeln und dem in der Mitte stehenden Stern auf. Um solche Probleme zu lösen, wurde das 3D-Modell auf eine 2D-Ebene verlegt. Die Überlegung dabei ist, dass alle drei Muskeln den gleichen Effekt mit der gleichen Gestaltung bringen.

In den folgenden Unterkapiteln werden ein paar weitere unbekannte Parameter ermittelt, die durch Schrumpfung und Ausdehnung um die Ausgangslänge des Muskels und die Variation der Parameter verursacht werden. Beispielsweise wird gezeigt, wie der Abstand zwischen Kugelgelenk und Muskel und der maximale Drehwinkel des oberen Druckstabs zur Vertikalen bei jedem Versuche berechnet werden.

Für Kollisionsüberprüfung wurde wie es im Fallunterscheidung für jeweils die beiden Seiten für einen sich ausdehnenden und sich ziehenden Muskel, der auf der XY-Ebene liegt, geteilt.

Sich ausdehnende Muskeln

Zur Untersuchung der Kollisionen wird zuerst der obere Druckstab nach rechts gebogen. Wir diskutieren alle Kollisionsmöglichkeiten beider Seite.

Linke Seite

Hier liegt nur ein Muskel auf der XY-Koordinatenebene. Der Vorteil liegt in der Vereinfachung der Berechnung. Die Abbildung 7 erfolgt in 2D-Darstellung. Ein Blick auf der schwarzen Linien, das ist der Defaultzustand. Alle Muskeln sind mit gleicher Länge und symmetrisch zum Druckstab gebildet. Die roten Linien stellen einen Zustand dar, der obere Druckstab nach rechts gebogen wurde.

Stern jeweils vor und nach der Drehung ϑ des oberen Druckstabs. Die Linien g_1 und S bilden einen Winkel β .

Als ersten Schritt legt man die Länge des Muskels L_{15} nach Pythagoras zugrunde fest:

$$L_{15} = \sqrt{H^2 + (R_2 - R_1)^2} \quad (1)$$

So folgt die Länge $L_{30} = L_{15} * (1 - 0.15)$ bei der Schrumpfung und die Länge

$L_0 = L_{15} * (1 + 0.15)$ beim Strecken.

Der Kosinus des Winkels α ist das Verhältnis zwischen der Gegenkathete (hier: $R_2 - R_1$) und der Hypotenuse (hier: L_0):

$$\alpha = \arccos \frac{R_2 - R_1}{L_0} \quad (2)$$

Der Schneidenwinkel β ist durch das Verhältnis zwischen Gegenkathete (hier: R_1) und Ankathete (hier: S) gegeben:

$$\beta = \arctan \frac{R_1}{S} \quad (3)$$

g_1 liegt auf der Hypotenuse eines rechtwinkligen Dreiecks:

$$g_1 = \sqrt{R_1^2 + S^2} \quad (4)$$

g_2 liegt auch auf der Hypotenuse eines rechtwinkligen Dreiecks: ist:

$$g_2 = \sqrt{R_2^2 + (H - S)^2} \quad (5)$$

t_2 ist der Abstand zwischen Kugelgelenk und sich streckenden Muskel und kann eine Kollision verursachen. Deswegen handeln wir hier t_2 als feste Eingabe und gleich 16mm. So

kann $L_{L_strecken}$ wieder durch den Satz des Pythagoras berechnet werden:

$$L_{L_strecken} = \sqrt{g_1^2 - t_2^2} + \sqrt{g_2^2 - t_2^2} = \sqrt{R_1^2 + S^2 - t_2^2} + \sqrt{R_2^2 + (H - S)^2 - t_2^2} \quad (6)$$

$L_{L_strecken}$ ist die Länge des gestreckten linken Muskels. Eine Voraussetzung davon für diese

Untersuchung ist es, dass $L_{L_strecken}$ im Wertbereich $[L15, L0]$ liegen muss.

Rechte Seite

In der Abbildung 8 ist ein Muskel im Raum zu sehen. Die Linien aus dunkelblau ist die

Muskelebene, die vor der XY-Ebene steht. Die Linien aus heilblau bildet sich eine

Projektionsebene von Muskelebene. Der Winkel zwischen zwei Ebene beträgt 60° .

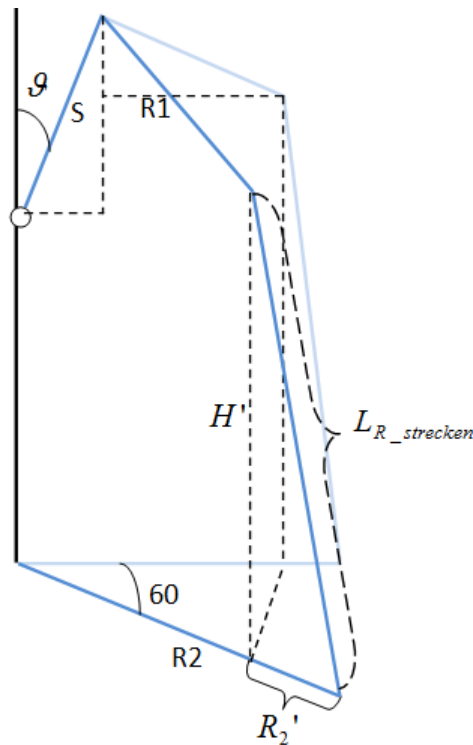


Abbildung 8 Rechte Ansicht beim streckende Muskel.

Wir projektzieren die Muskelebene senkrecht zur XY-Ebene, so ist die Höhe zwischen

Sternarme der Muskelebene gleich wie die Projektionsebene.

$$H' = (H - S) + S * \cos(\vartheta) - \cos(60) * R1 * \sin(\vartheta) \quad (7)$$

Wie die Gleichung (7) dargestellt, die gesamte Höhe ist gleich die Höhe des unteren Druckstabs (H-S) plus die Höhe des oberen Druckstabs ($S \cdot \cos(\vartheta)$) und minus die Höhe des oberen Sternarmes ($\cos(60) \cdot R1 \cdot \sin(\vartheta)$).

$$R_2' = R_2 - \frac{S \cdot \sin(\vartheta) + \cos(60) \cdot R1 \cdot \cos(\vartheta)}{\cos(60)} \quad (8)$$

Die Gleichung (8) ist die Länge R2 minus die Strecke, die R2 durch Projizieren auf der X-Achse zu berechnen ist.

Schließlich ist $L_{R_strecken}$ nach dem Pythagoras gegeben, sieht die Gleichung (9).

$$L_{R_strecken} = \sqrt{(H')^2 + (R_2')^2} \quad (9)$$

$L_{R_strecken}$ soll im Wertbereich [L30,L15] liegen, wenn alle Parameter angepasst sind.

Die nächste Kollisionsmöglichkeit ist die Konflikt der beiden Sternarme, die neben dem Gelenk stehen.

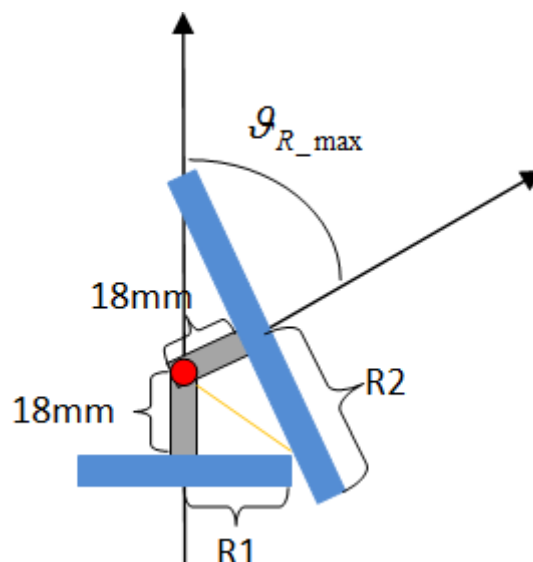


Abbildung 9 Konlisionsdarstellung von mittleren Sternarme beim sich streckenden Muskel.

Wir müssen sicher für das Programm vorstellen, dass ϑ nicht über ϑ_{R_max} schreitet. So ϑ_{R_max}

kann als Definitionsbereich von \mathcal{G} definieren, wenn \mathcal{G}_{R_max} kleiner als \mathcal{G}_{L_max} ist.

$$\mathcal{G}_{R_max} = 180 - \arccos\left(\frac{18}{\sqrt{R_1 + 18^2}}\right) - \arctan\left(\frac{R_1}{18}\right) \quad (10)$$

Die Gleichung (10) ist die Gleichung für den Winkel \mathcal{G}_{R_max} , gleich 180° minus zwei Winkels.

Einer ist durch der Strecke R1 und der Strecke zwischen Kugelgelenk und R1 geschnitten.

Ander ist durch der Strecke R2 und der Strecke zwischen Kugelgelenk und R2 gebildet.

Sich ziehende Muskeln

Auf der rechten Seite sind zwei Muskeln mit gleichem Effekt zu sehen. Wir werden nur einen Muskel betrachten, der vor der XY-Ebene steht. Die tragenden Sternarme stehen jeweils 60° zur XY-Ebene. Zur weiteren Untersuchung der Kollisionen wird dann der obere Druckstab nach links gebogen. Wir diskutieren alle Kollisionsmöglichkeit für beide Seite.

Linke Seite

Die Abbildung 10 zeigt die Geometrie mit einem schrumpfenden Muskel auf der linken Seite.

Hier gibt es eine Bedingung, die wir beachten müssen, nämlich dass $L_{L_schrumpfen}$ im

Wertbereich von [L30,L15] liegen muss.

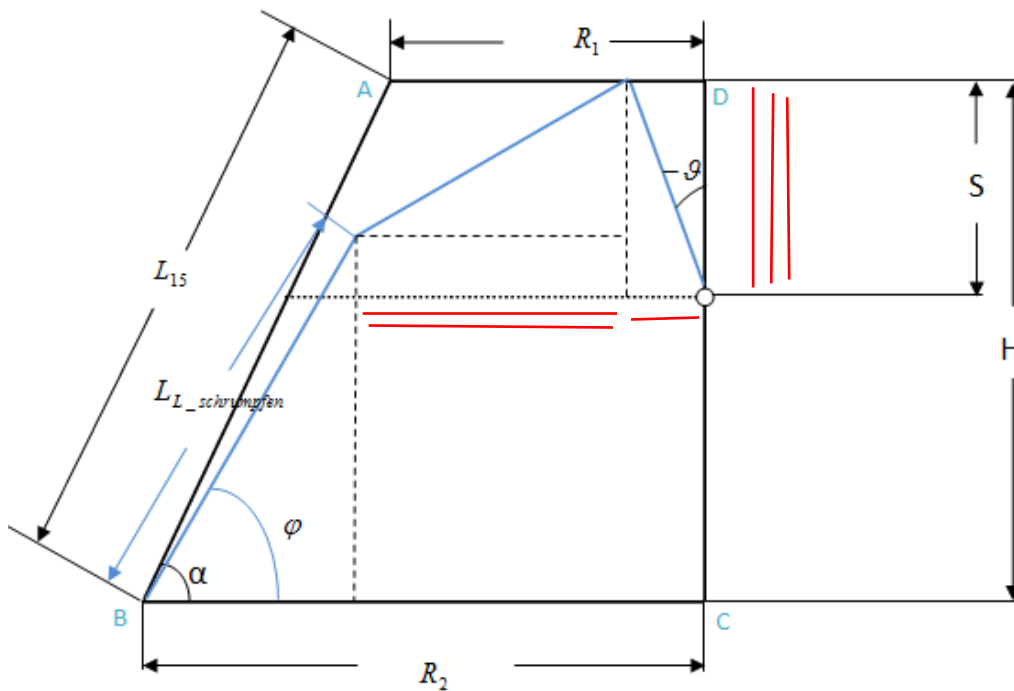


Abbildung 10 Linke Ansicht beim schrumpften Muskel.

$$L_{L_schrumpfen} = \sqrt{(R_2 - \underline{S * \sin(\theta)} - \underline{R1 * \cos(\theta)})^2 + (H - S + \underline{S * \cos(\theta)} - R1 * \sin(\theta))^2} \quad (11)$$

Die Gleichung (11) hat einen ähnlichen Berechnungsweg wie $L_{R_strecken}$.

Gleichfalls ist die Konfliktvorbeugung von zwei Sternarme neben dem Kugelgelenk zu betrachten.

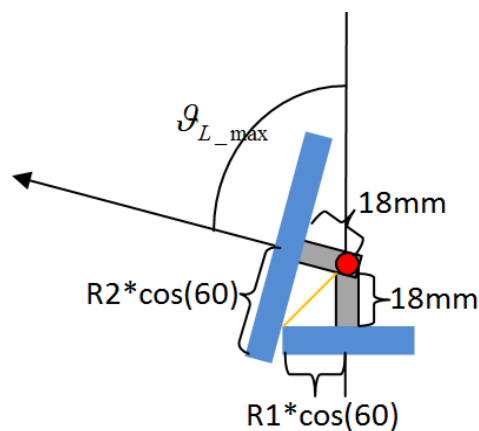


Abbildung 11 Konfliktvorbeugung von mittleren Sternarme bei dem schrumpften Muskel

Der Rechenweg ist ähnlich, aber wir rechnen für diesen Fall mit der Projektionslänge des

Sternarmes.

$$\vartheta_{L_{\max}} = 180 - \arccos\left(\frac{18}{\sqrt{(R_1 * \cos(60))^2 + 18^2}}\right) - \arctan\left(\frac{R_1 * \cos(60)}{18}\right) \quad (12)$$

Rechte Seite

An der rechten Seite des Druckstabs gibt es zwei kritische Stellen. Erstens, ob $L_{R_schrumpfen}$ im Wertebereich liegt. Es kann, zweitens eine Berührung zwischen der sich streckenden Muskeln und dem über dem Kugelgelenk stehenden Sternarm geben. Mit t_3 definieren wir seinen Abstand, er soll nicht kleiner als 16mm sein. 16mm ist die Summe der Radien des Sternarmes und Muskels.

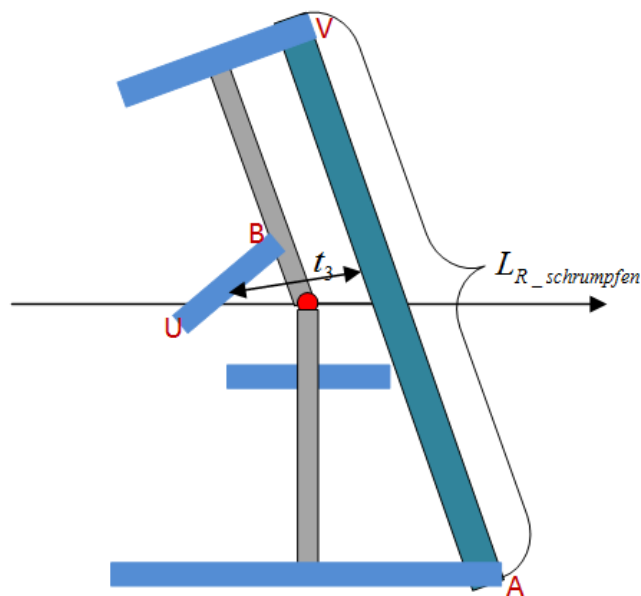


Abbildung 12 Rechte Ansicht beim schrumpfenden Muskel

Die Abbildung 12 ist ein Raumobjekt. Es ist kompliziert, das Objekt in die 2D-Ebene zu zerlegen. So berechnen wir den Abstand t_3 mittels Vektorrechnung zwischen windschiefe Geraden.[4]

A und V sind die Koordinaten des Anfangspunktes und der Endpunktes des Muskels:

$$A = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} R_2 * \cos(60) \\ -R_2 * \sin(60) \\ 0 \end{bmatrix} \quad (13)$$

$$V = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -S * \sin(\vartheta) + R_1 * \sin(\vartheta) * \cos(60) - R_2 * \cos(60) \\ -R_1 * \sin(60) - R_2 * \sin(60) \\ H - S + S * \cos(\vartheta) - R_1 * \sin(\vartheta) * \cos(60) \end{bmatrix} \quad (14)$$

$L_{R_schrumpfen}$ ist die Länge des Vektors \overrightarrow{VA} :

$$L_{R_schrumpfen} = \sqrt{(a_1 - v_1)^2 + (a_2 - v_2)^2 + (a_3 - v_3)^2} \quad (15)$$

B und U sind die Koordinaten des Anfangspunktes und Endpunktes des Sternarmes:

$$B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} -18 * \sin(\vartheta) \\ 0 \\ H - S + 18 * \cos(\vartheta) \end{bmatrix} \quad (16)$$

$$U = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} -R_2 * \cos(\vartheta) * \cos(60) - 18 * \sin(\vartheta) \\ -R_2 * \sin(60) \\ H - S + 18 * \cos(\vartheta) - R_2 * \sin(\vartheta) * \cos(60) - (H - S + 18 * \cos(\vartheta)) \end{bmatrix} \quad (17)$$

Der Abstand zweier windschiefer Geraden $\overrightarrow{VA}, \overrightarrow{UB}$ kann durch die p,q-Formel berechnet werden.

p ist das Kreuzprodukt von V und U:

$$p = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = V \times U = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \times \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (18)$$

q ist Skalarprodukt von (B-A) und p:

$$q = (B - A) \cdot p = \begin{bmatrix} b_1 - a_1 \\ b_2 - a_2 \\ b_3 - a_3 \end{bmatrix} \cdot \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \quad (19)$$

t_3 ist:

$$t_3 = \frac{|q|}{\sqrt{p_1^2 + p_2^2 + p_3^2}} \quad (20)$$

4 Implementierung

4.1 Algorithmus unter Matlab

Das Programm strecken.m ist das Hauptprogramm, das nicht nur das analytische Ergebnis, sondern auch die programmmentschiedene geometrische Darstellung ausgibt. Dieses Programm wird in vielen Teilen geschrieben, variable Definition, Fallentscheidung, Diagramm plotten und Werte speichern.

4.1.1 Exploration mittels Schleifen

Der Teil Fallentscheidung wird durch For- und If-Schleifen gebildet.

```

      for j=1:length(H)
        for r1=1:length(R1)
          for r2=1:length(R2)
            1 { if psi_max_links>psi_max_rechts
                psi_max=psi_max_rechts;
            } else
                psi_max=psi_max_links;
            end;
            for r3=1:length(Ks)
              for psi=0.5:psi_step:psi_max
                2 { if t3>16 & L30_rechts>L30 & L30_rechts<L0 &
                    L0_links>L30 & L0_links<L0 &
                    L0_rechts>L30 & L0_rechts<L0 &
                    L30_links>L30 & L30_links<L0
                } end;
                3 { if t3_counter>t3 & psi_counter<psi ||
                    t3_counter>=t3 & psi_counter+psi_step
                    >=psi_max
                } switch H
                    case 300
                    case 400
                    case 500
                end;
                break;
            end;
          end; end; end; end; end;
        end;
      end;
    
```

The diagram illustrates the nesting of loops in the MATLAB code. Braces on the left side of the code are numbered 1 through 8, indicating the nesting levels. Level 1 is the innermost loop (if statement), level 2 is the psi loop, level 3 is the r3 loop, level 4 is the psi loop, level 5 is the r2 loop, level 6 is the r1 loop, level 7 is the j loop, and level 8 is the outermost loop (j loop).

1.If-Schleife: Hier wird der Wertebereich für den Winkel ϑ definiert. Für eine sichere Drehung entscheidet man sich für einen kleinen Wert von `psi_max_links` und `psi_max_rechts`, damit es keinen Konflikt zwischen zwei Sternarme auf der anderen Seite gibt.

2.If-Schleife: In dieser If-Schleife wird ein Zähler gespeichert, bei dem alle psi- und t3-Bedingungen erfüllt sind, z. B. bis zum Zähler = n.

3.If-Schleife: Mit steigendem psi-Wert wird irgendwann eine Bedingung nicht mehr erfüllt, dann kommt das Programm in dieser If-Schleife, z. B. bis zum Zähler = n+1. Vergleicht man t3 und psi bei einem Zähler gleich n und n+1, wenn $t3(n-1) > t3(n)$ und $\psi(n-1) < \psi(n)$ oder $t3(n-1) > t3(n)$ und psi seinen maximalen Wert erreicht hat, dann werden alle Parameter mit entsprechender Segmenthöhe gespeichert und es tritt die letzte for-Schleife aus.

4.For-Schleife: Schleife für Parameter `psi``.

5.For-Schleife: Schleife für Parameter `Ks``.

6.For-Schleife: Schleife für Parameter `R1``.

7.For-Schleife: Schleife für Parameter `R2``.

8.For-Schleife: Schleife für Parameter `H``.

4.2 SIMULINK und VR-Toolbox

Matlab

Matlab ist eine weltweit angewandte numerische rechnerische Umgebung der vierten Generation der Programmiersprachen, von The MathWorks entwickelt. Sie wurde in dieser Arbeit verwendet, weil man sie verhältnismäßig leicht mit V-Realm Builder 2.0 durch die Funktion VR sink kombinieren kann. Matlab ermöglicht auch das Lösen von

Funktionssystemen, Plotten von Daten, Implementierung von Algorithmen. Der Hauptbestandteil dieser Diplomarbeit soll mit der Programmiersprache Matlab erfüllt werden. In den folgenden Kapiteln werden der grundsätzliche Ablauf dieser Programme für die Muskelsegmentsimulation sowie deren wichtigste Funktionen wie anwenderspezifische Eigenschaften und Besonderheiten beschrieben.

Virtual reality toolbox


Die Virtual Reality Toolbox ermöglicht es, eine virtuelle Welt um Simulink und MATLAB zu verbinden. Das Verständnis der Funktionen der Virtual Reality Toolbox und einige grundlegende VRML Konzepte helfen, dieses Modell noch effektiver einzusetzen. Die Lösung für die Interaktion mit virtuellen Modellen der dynamischen Systeme wird im Laufe der Zeit durch Beschreibung der vielen verfügbaren Funktionen erfüllt. Unterschiede bestehen zwischen VRML MATLAB und Koordinatensystemen, dem Format der VRML-Dateien (VRML-Welten) mit einer Schnittstelle zu Simulink-Block-Diagrammen, ferner zwischen einer Schnittstelle zu MATLAB und zu Objekt-Funktionen. Die Virtual Reality Toolbox enthält Beispiele sowohl für SIMULINK- als auch MATLAB-Schnittstellen.

V-Realm Builder 2.0

V-Realm Builder ist ein leistungsfähiges dreidimensionales Authoring-Paket für die Erstellung von 3D-Objekten und "Welten", damit V-Realm Browser oder ein anderer VRML 2.0 kompatibler Browser betrachtet werden kann. Es ist hinzuzufügen, dass VRML 2.0 als "Netzwerk" für eine virtuelle Datei-Spezifikation entwickelt wurde. V-Realm Builder und VRML waren nicht dazu bestimmt, die heutigen Modellierungs-Tools zu ersetzen. VRML ist mit den Modellierungspaketen in der Lage, unglaublich realistische Objekte zu erstellen. Er bewältigt 5 oder 10 Millionen Polygone. Diese Objekte können für Stand-alone-Rechner oder Anwendungen eingesetzt werden, aber eine Datei von der erwarteten Größe in Echtzeit in das

Netz zu transportieren ist nicht möglich. Aus diesem Grund wurde mit V-Realm Builder 2.0 entwickelt, um die Werkzeuge und die Größe der Dateien zu minimieren und ein Mittel zur Modellierung komplexer Objekte mit Primitiven, ohne Überlastung des Netzes in großen Dateien, zu bekommen.

4.3 Kopplung MATLAB mit V Realm Builder

In diesem Abschnitt wird die Kopplung MATLAB mit V Realm Builder illustriert. Der Funktionsblock `VR sink`  ist Funktionsbestandteil von Virtual Reality Toolbox, das steht im Simulink Library Browser unter Virtual Reality Toolbox.

Zuerst Doppelklick auf VR sink-Block in unserer Modelldatei, dann bekommen wir die folgende Abbildung 13. Darauf sieht man unser Modell im Ruhezustand. Klick auf `Block Parameters` unter Funktion `Simulation` in der Toolbar. Schließlich gibt der Rechner die Abbildung 14 aus.

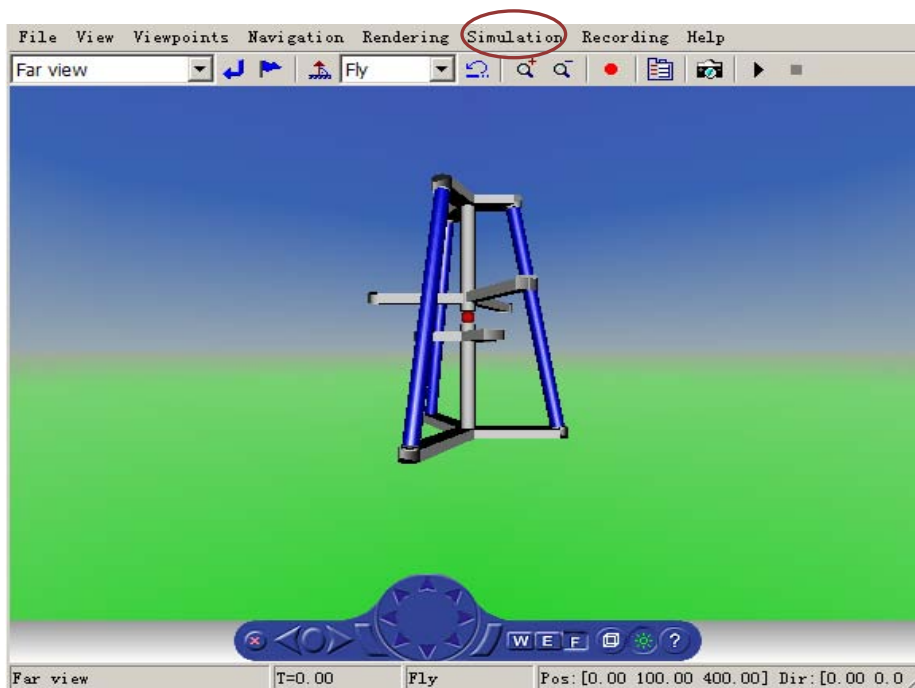


Abbildung 13 Ereignis durch Doppelklick auf VR sink Block.

In der Abbildung 14 ist ein Dialog von VR sink zu sehen. Hiermit können wir die WRL-Datei im Source file-Fenster hochladen, oder die Datei in der Software V Realm Builder öffnen. In der rechten Seite ist ein VRML Tree, auf dem die entsprechende Objekteigenschaft gelistet ist. Die markierte Knoten sollen nachher auf dem VR sink-Block stehen.

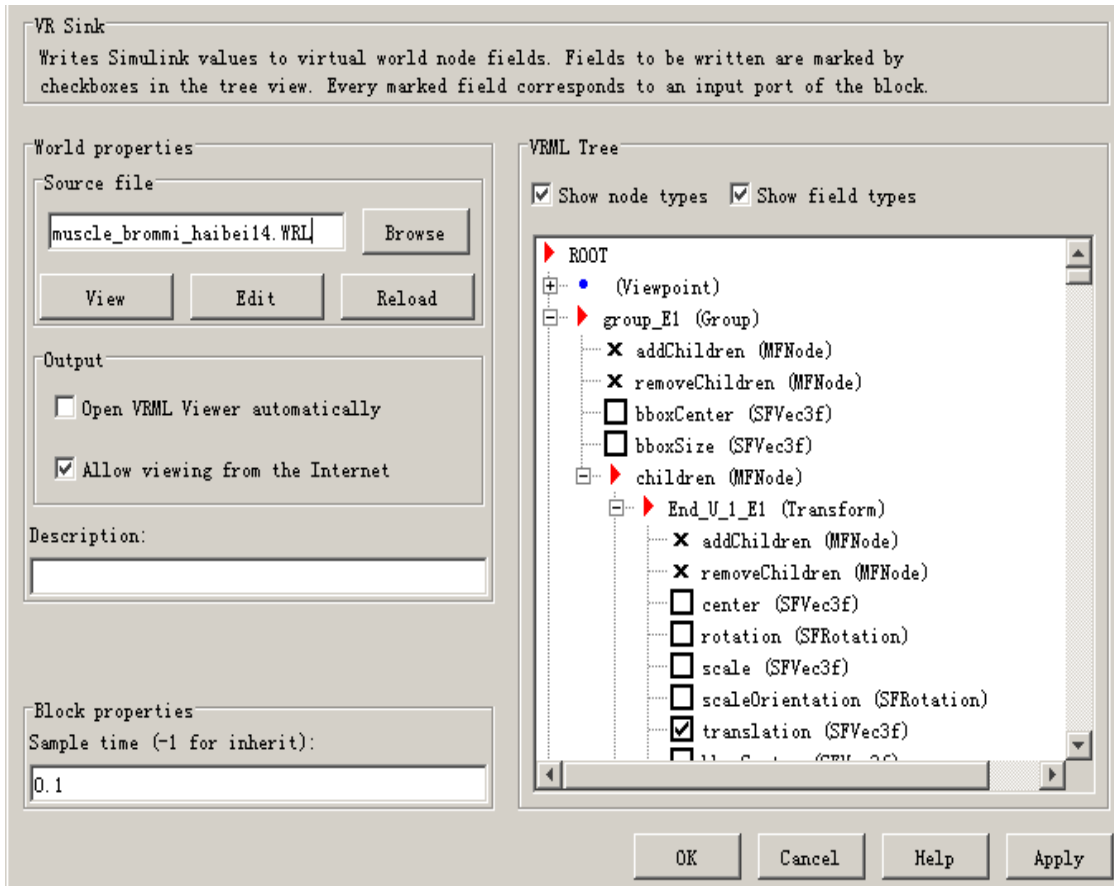


Abbildung 14 Dialogfenster zur VRML

5 Simulation und Auswertung

In diesem Kapitel „Simulation und Auswertung“ geht es um die Beschreibung eines innerhalb des Programms durchgeführten Versuchs. Hierbei wird beschrieben, wie man die beiden Programme benutzt und welche Ereignisse auftreten.

Die Einarbeitung in Matlab erfolgte unter Zuhilfenahme bereits bestehender Programme. Das Ausführen und das teilweise Abändern dieser Programme erlaubte einen Einblick in die Struktur und den Ablauf von Matlab. In der CD wurden drei Dateien gespeichert, sie heißen *strecken.m*; *muscle_brommi_haibei.mdl* und *muscle_brommi_haibei14.wrl*.

Das Programm *strecken.m* untersucht alle Möglichkeiten unter verschiedenen Versuchsbereichen, ob sie sich unter dem System eignen. War das der Fall, wurde die Modellstellung gezeichnet, nachfolgend wurden die Werte in ein Diagramm geplottet und gespeichert. Die Programme *muscle_brommi_haibei.mdl* und *muscle_brommi_haibei14.wrl* wurden für die Simulation zur Verfügung gestellt. Die VRML-Datei *muscle_brommi_haibei14.wrl* ist im Klartext (ASCII bzw. UTF-8) geschrieben, sie kann auch in einem einfachen Texteditor erstellt werden. Sie beschreibt die Geometrie und die Struktur des 3D-Modells. Die von *strecken.m* erstellten Parameter wurden als Eingabe des Simulationsprogramms eingegeben und die Stellung des 3D-Modells wurde gleichzeitig geändert.

5.1 Geometrische Zusammenhänge

Nach der Durchführung des Programms *strecken.m* ergeben sich drei Diagramme, jeweils für H gleich 300mm, 400mm und 500mm. Sie alle zeigen den ausgewählten Winkel ψ und $R1/R2$ in entsprechenden K_s . K_s ist das Verhalten des Höhenstandes des Kugelgelenkes. K_s ist von 0.15 bis 0.85 definiert, weil wir eine Strecke für die Installation des Sterns jeweils

oben und unten neben dem Kugelgelenk leeren müssen. Das K_s -Wert haben wir durchschnittlich in 5-Fach geteilt, und zum Schauen, K_s mit welchem Wert wird maximaler Winkel ψ auftauchen.

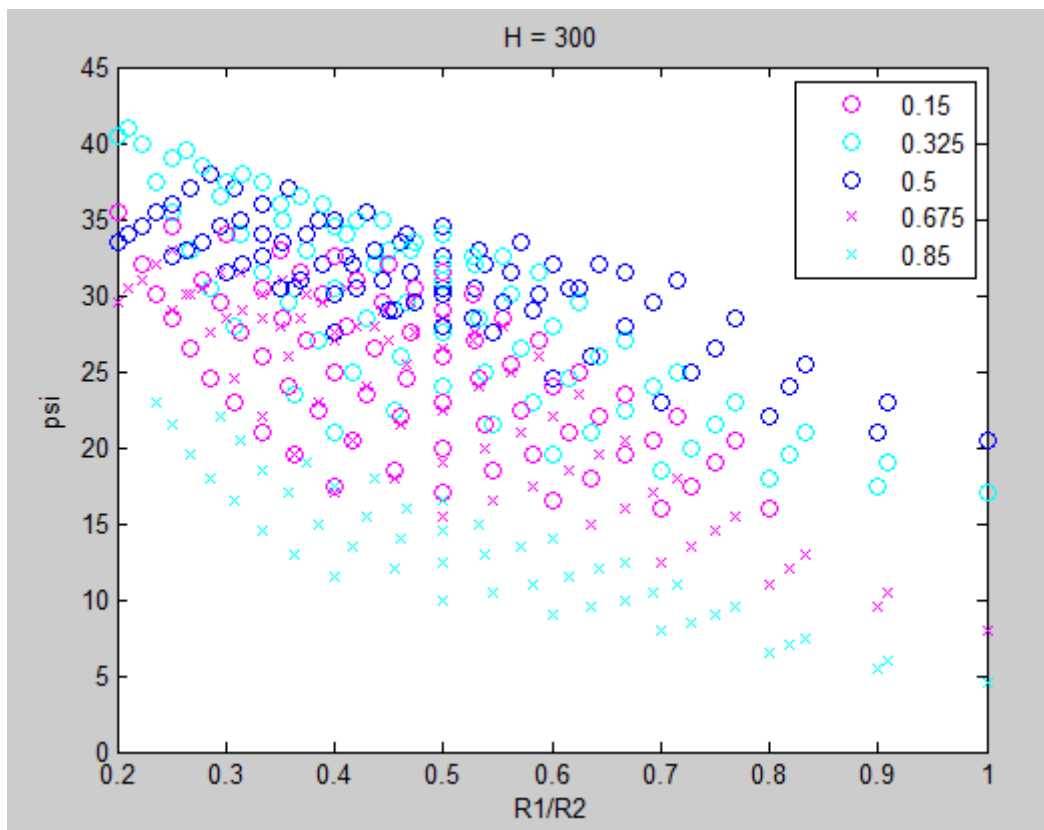


Abbildung 15 Ergebnisdarstellung des Programms strecken.m

Die Abbildung 15 zeigt das Verhältnis zwischen ψ und $R1/R2$ für die Robotersegmenthöhe gleich 300mm. Die farbigen Kreis und Kreuze liefern die Information, welche K_s entsprechen. Es gibt zwei zusätzliche Diagramme für die Robotersegmenthöhe gleich 400mm und 500mm, die können in der CD erhalten werden.

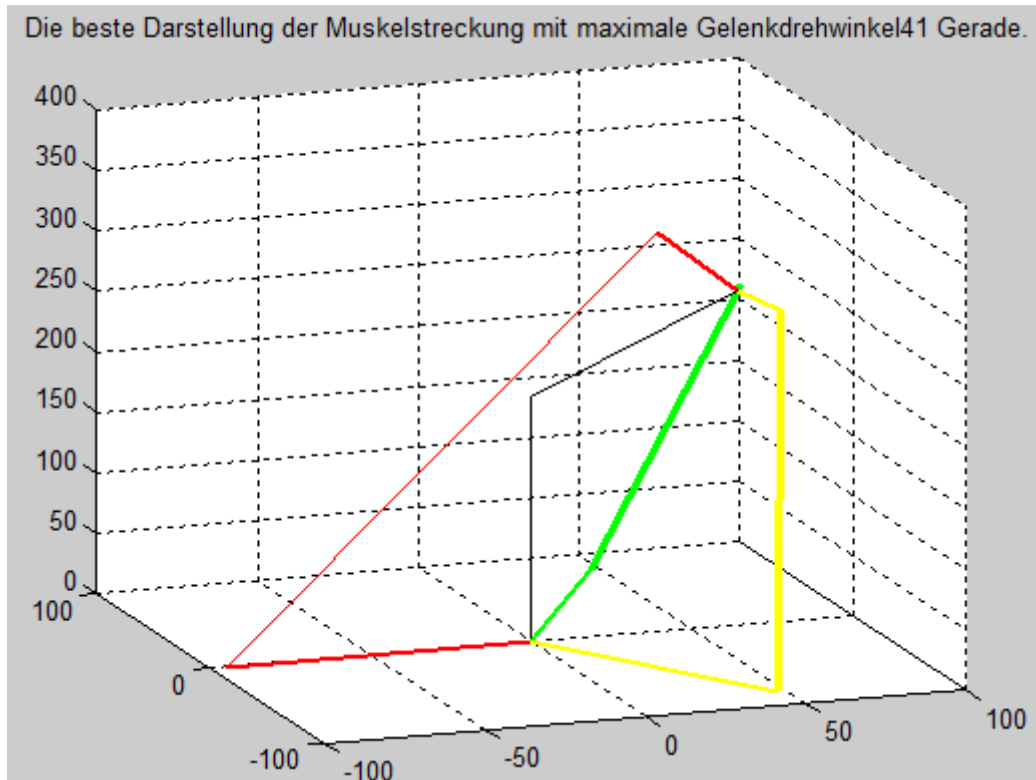


Abbildung 16 3D-Stabmodell mit maximmale Gelenkdrehwinkel bei H=300mm

Aus der Abbildung 16 ist ein 3D-Stabmodell des Muskelsegments ohne Sterne in der Mitte zu sehen. Als Parametergruppe haben wir die ausgesucht, die der maximale Gelenkdrehwinkel in Besitz nimmt. Die roten Linien stellen eine sich ausdehnende Muskelseite dar. Wegen der Verlängerung des Muskels wird dieser mit dünner Linie gezeichnet. Umgekehrt sind sich verkürzende Muskeln mit dicker Linie eingetragen. Da die Achseneinheit in Z-Richtung deutlich kleiner ist, ist das Verhalten des oberen und unteren Druckstabs nicht klar zu ersehen.

5.2 Optimaler Knickwinkel

Optimaler Knickwinkel heißt hier, dass wir immer einen sich nähernden Drehwinkel ϑ für unterschiedliche Segmenthöhen mit ähnlichem Ks- und R1/R2-Verhalten erhalten.

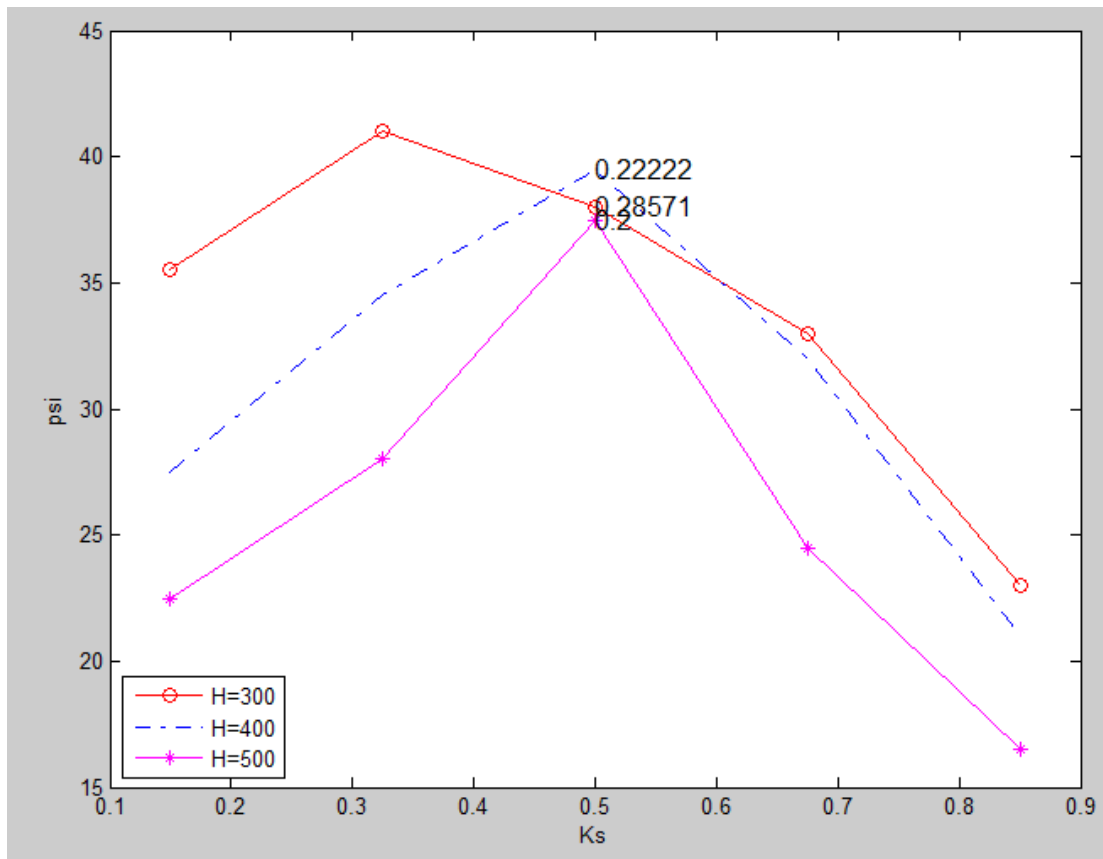


Abbildung 17 Darstellung des maximalen Drehwinkels aus verschiedenen Ks-Werte bei unterschiedlichem R1/R2-Verhalten.

Die Abbildung 17 liefert diese Aussage. Drei Kurven verlaufen spitzförmig. Die rote Kurve hat allein einen maximalen Knickwinkel bei Ks gegen 0.3. Dieses Verhalten bei den anderen Fällen schwer zu erreichen. Aber bei Ks gleich 0.5 treffen alle Kurven mit einen ähnlichen Knickwinkel, die Knickwinkeldifferenz 2° zwischen Muskelsegmentshöhen 400mm und 500mm beträgt. Neben den Kurven sind mit der Zahlen beschriftet, die das Verhalten R1/R2 bedeutet. Sie unterscheiden sich auch klein, nur mit 0.08, d.h. 8 Prozent.

5.3 3D-Modell

Eulerscher Winkel[5]

Der Begriff Euler-Winkel wird in der Mathematik meistens für die Luft- und Raumfahrttechnik verwendet. Der Euler-Winkel wird hier als Hilfsmittel für die Darstellung des 3D-Modells verwendet, das die räumliche Orientierung eines oberen Muskelsterns als eine Zusammensetzung von Drehungen des Bezugssystems eines unteren Muskelsterns simuliert. Im Folgenden werden die festen Systeme mit Kleinbuchstaben (x, y, z) und die gedrehten Systeme mit Großbuchstaben (X, Y, Z) bezeichnet. Als Euler-Winkel sind drei Winkel eingeführt, um die Orientierung eines starren Körpers im 3-dimensionalen euklidischen Raum zu beschreiben. Dies ist gleichbedeutend mit der Aussage, dass jede Orientierung durch die Komposition von drei elementaren Rotationen (Drehungen um eine einzige Achse als Basis) erreicht werden kann, und auch gleichbedeutend mit der Aussage, dass jede Rotationsmatrix, die das Produkt von drei elementaren Rotationsmatrizen ist, zerlegt werden könne.

Zuerst definieren wir eine Linie von Knoten (N) als Schnittlinie der xy - und der XY -Koordinatensystem-Ebenen (mit anderen Worten, der Knotenlinie senkrecht zur z - und zur Z -Achse). Dann definieren wir den Euler-Winkel, der sich auf die Knoten bezieht, zum Beispiel: ψ ist der Winkel zwischen der X -Achse und der Linie der Knoten.

θ ist der Winkel zwischen der z -Achse und der Z -Achse.

ϕ ist der Winkel zwischen der Knotenlinie und der X -Achse.

Nach Eulers Rotations-Theorem kann jede Rotation mit Hilfe von drei Winkeln beschrieben werden. Wenn die Rotationen im Hinblick auf die Rotationsmatrizen beschrieben werden, kann eine allgemeine Drehung als Produkt beschrieben werden:

$$A = B \cdot C \cdot D .$$

Die drei Winkel ergeben die drei Rotationsmatrizen; sie werden Eulerwinkel genannt und jeweils in B, C, D als 3×3 Matrix beschrieben. Es gibt verschiedene Konventionen für die Euler-Winkel, die abhängig von den Achsen über die Rotationen durchgeführt werden. Die Matrix A wird so beschrieben:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Die so genannte "zxz-Konvention" ist, wie oben dargestellt, die häufigste Definition in der XY-Ebene, und zwar

1. die erste Drehung um einen Winkel ϕ um die z-Achse mit D,
2. die zweite Drehung um einen Winkel θ über die ehemalige x-Achse mit C, und
3. die dritte Drehung um einen Winkel ψ von der einstigen z-Achse mit B.

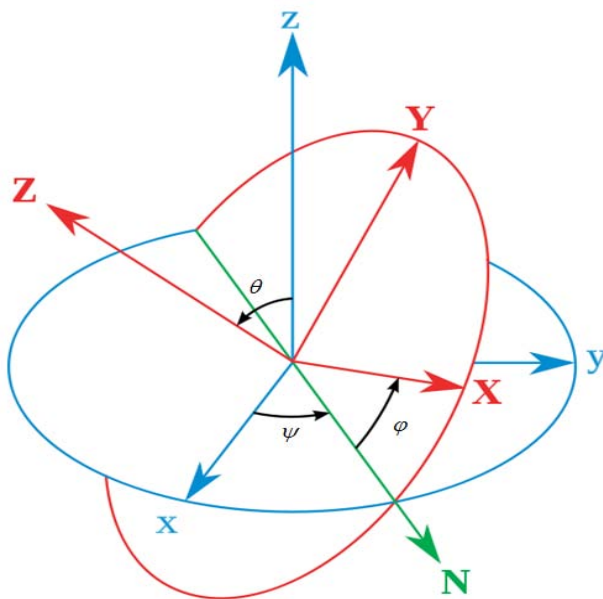


Abbildung 18 Eulersche Winkel in zxz-Konvention

$$D = \begin{bmatrix} \cos \phi & \sin \phi & 0 \\ -\sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}$$

$$B = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Dann folgt nach der Multiplikation der Matrizen:

$$a_{11} = \cos \psi \cos \phi - \cos \theta \sin \phi \sin \psi$$

$$a_{12} = \cos \psi \sin \phi + \cos \theta \cos \phi \sin \psi$$

$$a_{13} = \sin \psi \sin \theta$$

$$a_{21} = -\sin \psi \cos \phi - \cos \theta \sin \phi \sin \psi$$

$$a_{22} = -\sin \psi \sin \phi + \cos \theta \cos \phi \cos \psi$$

$$a_{23} = \cos \psi \sin \phi$$

$$a_{31} = \sin \theta \sin \phi$$

$$a_{32} = -\sin \theta \cos \phi$$

$$a_{33} = \cos \theta$$

Man benutzt die oben genannten Rotationsmatrizen A, weil sich der Eulerwinkel einfach in Rotationsmatrizen umrechnen lässt. Die Rotationsmatrizen lassen sich leichter kombinieren, um eine komplexe Rotation zu erstellen. Hierbei ist jedoch zu beachten, dass die Reihenfolge der Kombination wichtig ist, da die Matrixmultiplikation im allgemeinen nicht kommutativ ist.

Aus der eulernen Matrix bekommen wir den Drehwinkel ω im Raum:

$$\omega = \arccos \left[\frac{a_{11} + a_{22} + a_{33}}{2} \right]$$

E ist der Richtungsvektor, der die Richtung des Drehwinkel ω zeigt.

$$E = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \frac{a_{32} - a_{23}}{2 \cdot \sin(\omega)} \\ \frac{a_{13} - a_{31}}{2 \cdot \sin(\omega)} \\ \frac{a_{21} - a_{12}}{2 \cdot \sin(\omega)} \end{bmatrix}$$

Zum Schluss werden die gestellten Formeln des Drehwinkels ω und Richtungsvektors E ins Programm ‚muscle_brommi_haibei.mdl‘ als Rotationsparameter im SIMULINK umschreiben.

Um das Modell in der Wirklichkeit zu simulieren, ist zuerst das Simulink-Modell („muscle_brommi_haibei.mdl“) zu öffnen. Im Anschluss werden die Sliderfenster durch Doppelklick auf den Block VR-Sink „muscle_brommi_haibei.WRL“ und alle vier Block Slider-Gain geöffnet. Dies wird in der folgenden Abbildung 19 gezeigt.

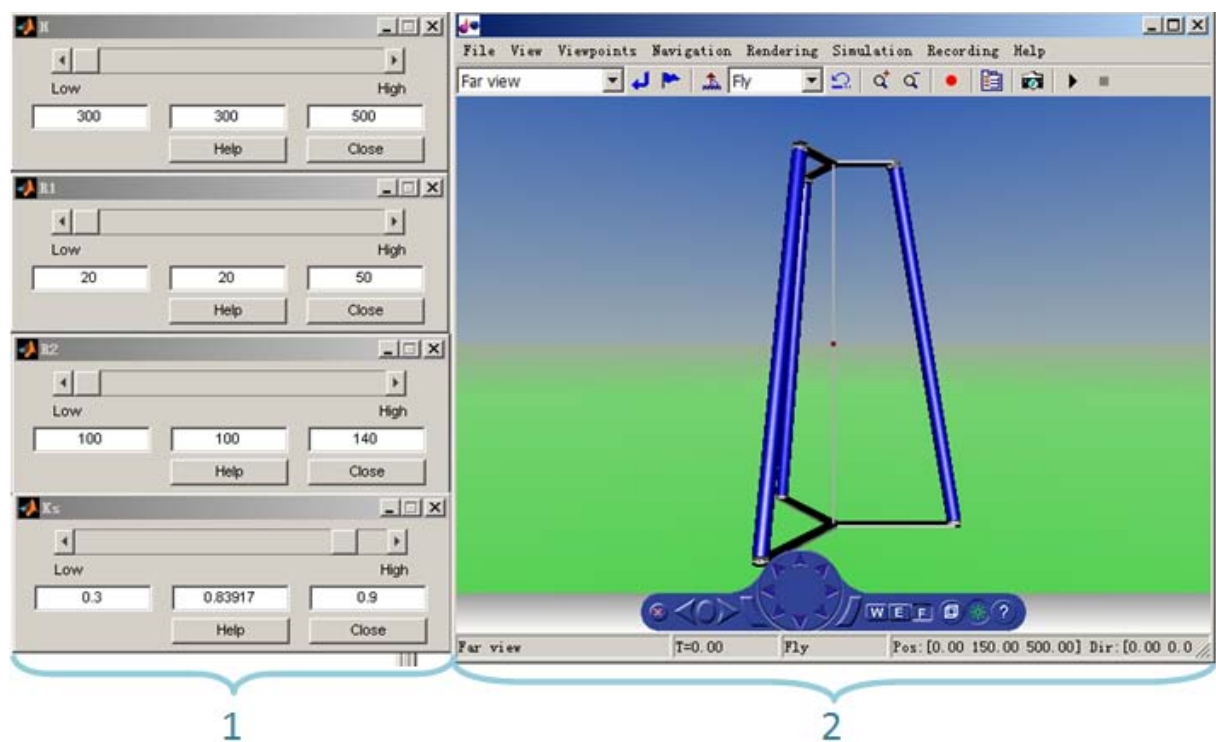



Abbildung 19 Eine Aufnahme aus der Kontrollebene der Simulation

In der Kontrollebene (siehe Nummer 1 auf Abbildung 19) konnten die Höhe des Roboterarmelementes, der Radius des dreiarmligen Sterns und die Lage des Kugelgelenks des Druckstabs dargestellt werden.

Das Roboterarmelement (Nummer 2 auf Abbildung 19) wurde von drei Muskeln an jeder Seite getragen, die jeweils auf einem dreiarmligen Stern befestigt waren. Auf diese Weise konnte in vertikaler Richtung ein starrer Druckstab realisiert werden. Damit sich die Muskeln verformen können, ohne sich zu beschädigen, wurden sie an ihren Enden mit Unterlegscheibe und Kugelgelenk festgelegt. Auf dem Druckstab wurde ein Kugelgelenk mit einer variablen Verhältniszahl angebracht.

Das Programm muss vor der Simulation durch Klicken auf den  Druckknopf kompiliert werden.

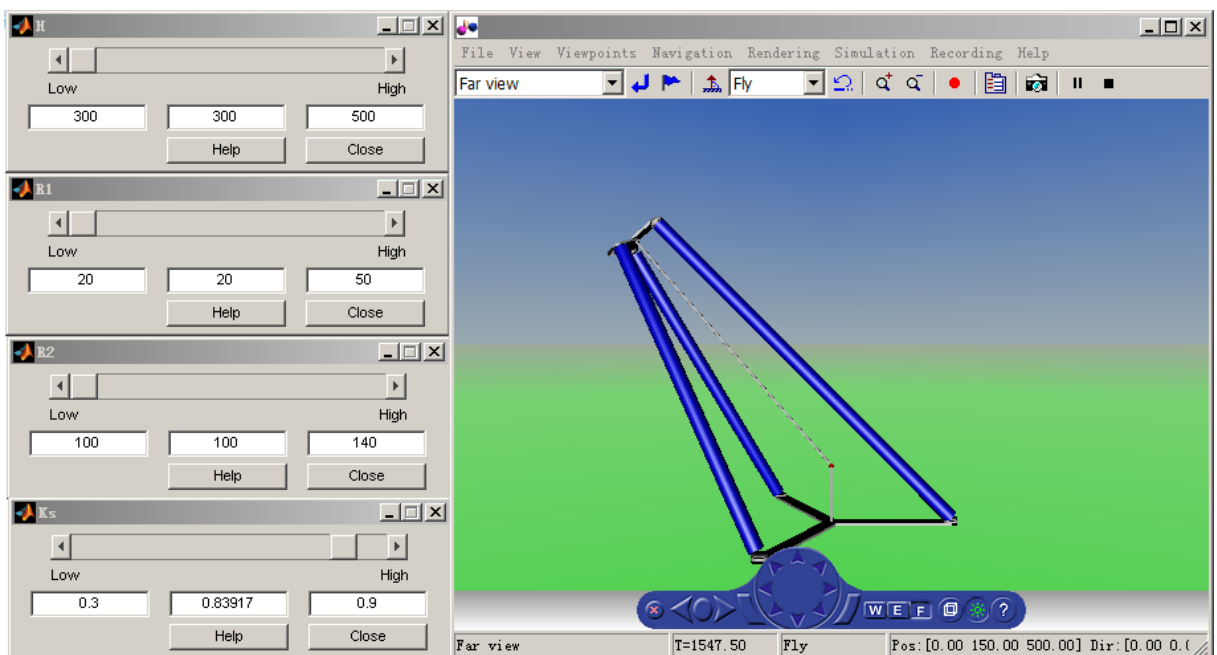


Abbildung 20 Simulation mit neu eingegebenem Parameter

Das Verändern der Ausgangsgröße des „Slider Gain“ erfolgt durch Anklicken der Flächen links/rechts zwischen dem Schieber und den Endmarken des Schiebereiches oder durch

Eingeben der Ausgangsgröße direkt im mittleren weißen Fenster. Alle Eingaben werden auf 2 Nachkommastellen gerundet.

Dieses 3D-Muskelementmodell ist nach links herum gebogen, weil die vorgegebene positive Richtung der X-Koordinatenachse dieses Modells umgekehrt zum Stab-Modell ist.

Die Defaultwerte von H, R1, R2, Ks müssen am Anfang als berechnete und genau passende Werte am mittleren weißen Fenster vor der Kompilation eingegeben werden.

6 Zusammenfassung und Ausblick

Das Teilkapitel ‚Zusammenfassung‘ gibt einen abschließenden Überblick über diese Diplomarbeit. Im Teilkapitel ‚Ausblick‘ wird beschreiben, welche Schritte bei einer Weiterentwicklung des Projekts noch zu machen sind bzw. welche Modifikationen noch durchgeführt werden können und an welchen Stellen noch ein Verbesserungspotenzial vorhanden ist.

6.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde ein biologisches Muskelsegment des Roboterarms unter Kollisionsuntersuchung für das Fachgebiet ‚Elektrotechnik und Informatik‘ entwickelt. Dies beinhaltet sowohl den Vorgang der 3D-Modellierung als naturexistierendes Modell der Elfantennase als auch die Erstellung des Simulationsprogramms. Als Grund für die Entwicklung dieses Modellsystems sind die traditionellen Roboterarme zu nennen, die nur unzureichend die Biomechanik des menschlichen Arms abbilden; außerdem ist das Bewegungsverhalten der bekannten Modelle und Roboterarme durch Singularitäten im Innern des Arbeitsraums stark beeinträchtigt.

Die Aufgabenstellung wurde erfüllt, indem das Programm zur Modellierung sowie Simulation erstellt wurde, einschließlich der zur Analyse des Modellsystems notwendigen Dokumentation. Das entwickelte Roboterarmelement konnte zum Ende der Diplomarbeit anhand einer 3D-Darstellung getestet werden. Somit kann dieses Modell mit passenden Parametern dazu herangezogen werden, einen im Anwendungszustand befindlichen Parameter zu simulieren.

6.2 Ausblick

Der in der Diplomarbeit beschriebene Roboterarm wurde zu einem naturähnlichen Muskel einer Elfantennase entwickelt. Um ihn auch auf der 3D-Objektdarstellung anwenden zu können, wird ein V-Realm Builder 2.0 zur Objektkonstruktion benötigt, da die verwendete Software „Matlab“ keine solche Funktion erfüllen kann. Jedoch ist die im Rahmen dieser Diplomarbeit verwendete Funktion Virtual Reality Toolbox für die Simulationsdurchführung der berechneten Messungen ausreichend. Ein Plus hinsichtlich der Verwendung einer Virtual Reality Toolbox, wie andere Toolbox die über mehr Simulationsblock verfügt, besteht bezüglich der Signalverbindungen und -verarbeitungsmöglichkeiten: Für die einzelnen Aufnehmerarten ist vom V-Realm Builder 2.0 ein Dialog zur Verbindung mit dem 3D-Objekt zu schalten. Um den Ablauf des Roboterarms mit dem bereitgestellten Programm für den Anwender zu entwickeln, ist es ratsam, dafür nach folgenden Schritten vorzugehen:

1. Mehrere Muskelsegmente mit gleichem oder abnehmendem Durchmesser weiter aufbauen und simulieren.
2. Die Systemoptimierung kann mit dem Programm Matlab anhand des Optimierungsergebnisses anhand der Minimal-Energieaufwand-Theorie weiterentwickelt werden. Hierbei ist gemeint, die gewonnenen Daten für den Energieaufwand so aufzubereiten, dass der gesamte Energieaufwand des Muskelements minimiert wird; dabei sind unterschiedliche Phasenverschiebungen zu betrachten. Die Einflüsse der Muskelbelastung und der Geschwindigkeit usw. können zur Systemoptimierung zusammengefasst werden.

7 Literatur

- [1] Christian Schäfer, Sindelfingen, Entwurf eines anthropomorphen Roboterarms: Kinematik, Arbeitsraumanalyse, Softwaremodellierung, VDI Verlag GmbH, Düsseldorf, 2001
- [2] W. Dicke, K. Kuhn, Alles über sichere Industrieroboter, Wirtschaftsverlag NW Bremerhaven, 1993
- [3] Fluidic Muscle DMSP/MAS aus der Firma FESTO
- [4] Ilja N. Bronstein, K. A. Semendjajew, Gerhard Musiol, Heiner Muehlig, Taschenbuch der Mathematik, Harri Deutsch Verlag, Moskau, 2008
- [5] U. Krey, A. Owen, Basic Theoretical Physics , Springer-Verlag Berlin, 2007