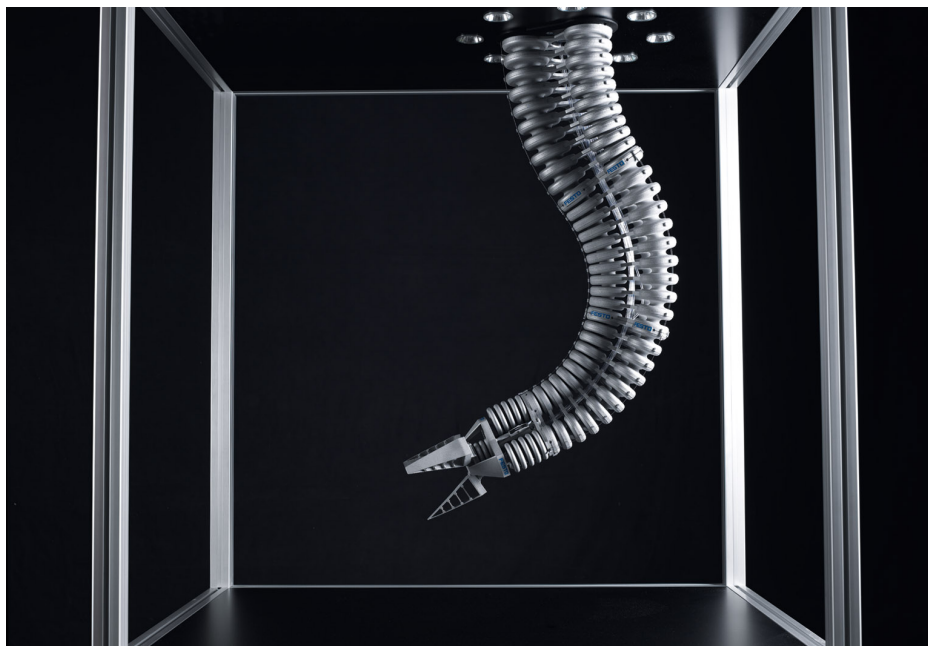


Praktikum Systemdynamik

– Sommersemester 2022 –

Versuch 2: Bionischer Handling-Assistent

Version 1.4



Inhaltsverzeichnis

1	Einleitung	3
2	Versuchsaufbau	4
3	Grundlagen der Robotik	7
3.1	Darstellung von Position und Orientierung	7
3.1.1	Position	7
3.1.2	Orientierung	7
3.2	Koordinatentransformation	8
3.2.1	Homogene Transformation	9
3.3	Kinematik	10
3.3.1	Kinematische Strukturen	10
3.3.2	Vorwärtskinematik	10
3.3.3	Rückwärtskinematik	10
4	Kinematik des BHAs	12
4.1	Definition der Koordinatensysteme	12
4.2	Vorwärtskinematik	13
4.2.1	Manipulatorarm	13
4.2.2	Sektion	13
4.2.3	Zusammenfassung	18
4.3	Rückwärtskinematik	18
4.3.1	Einschränkung der Freiheitsgrade	18
4.3.2	Lokale Optimierung	20
5	Versuchsdurchführung	26
5.1	Starten der Entwicklungsumgebung	26
5.2	Implementierung und Test der Vorwärtskinematik	26
5.3	Implementierung und Test der Rückwärtskinematik	28

1 Einleitung

Im Praktikumsversuch *Bionischer Handling-Assistent* werden Methoden zur kinematischen Steuerung und Regelung von redundanten Handhabungssystemen entworfen. Diese Methoden werden daraufhin in Matlab/Simulink implementiert und experimentell am Versuchsaufbau des Bionischen Handling-Assistenten erprobt. Ziel der jeweiligen Robotersteuerung ist es die Endeffektorgeschwindigkeit in kartesischen Koordinaten mittels eines Joysticks online vorgeben zu können.

Die zentralen Lehrinhalte des Praktikums beinhalten:

- Methoden der kinematischen Modellierung in der Robotik
- Kinematischer Steuerungsentwurf redundanter Handhabungssysteme
- Implementierung der Algorithmen in Matlab/Simulink

Hinweis:

Das Praktikumsskript enthält zwei Arten von Aufgaben welche unterschiedlich gekennzeichnet sind.

Vorbereitungsaufgabe – V2.0 –

Alle Vorbereitungsaufgaben sind selbstständig vor dem Praktikumstermin durchzuarbeiten. Falls nicht anders angegeben, wird eine schriftliche Lösung der Aufgaben verlangt. Diese sind auf gesonderten Blättern, welche mit Name, Studiengang und Matrikelnummer versehen sind, zu lösen. Die angefertigte Ausarbeitung kann gegebenenfalls zur Bewertung des Praktikums genutzt werden.

Praktikumsaufgabe – P2.0 –

Die Praktikumsaufgaben führen Sie durch den experimentellen Teil des Praktikums und werden am Tag der Versuchsdurchführung bearbeitet.

2 Versuchsaufbau

Im Folgenden wird der für das Praktikum zur Verfügung stehende Versuchsaufbau beschrieben. Dieser besteht, wie in Abbildung 1 dargestellt, aus 3 miteinander kommunizierenden Systemen. Dies ist ein

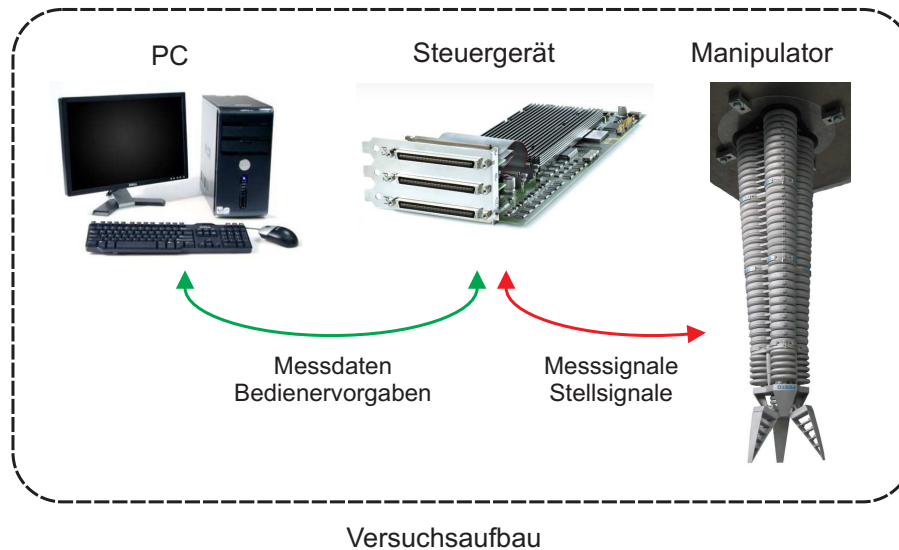


Abbildung 1: Kommunikationsdiagramm

handelsüblicher PC, ein dSpace Rapid Prototyping Steuergerät vom Typ RTI 1007, und ein redundanter 3D Manipulator, für den im Rahmen dieses Praktikums der Bionische Handling-Assistenten (BHA) eingesetzt wird.

Der PC stellt einerseits die notwendigen Programme zur Implementierung und Portierung der Steuerungs- und Regelungsalgorithmen zur Verfügung. Des Weiteren dient er während der experimentellen Versuchsdurchführung als Human-Machine-Interface. Die Bedienoberfläche der Manipulatorsteuerung wird mit dem Softwareprogramm ControlDesk NextGeneration erstellt, welches in der Lage ist dem Steuergerät in soft Real-Time Befehle zu erteilen bzw. Messdaten darzustellen und aufzuzeichnen.

Das Steuergerät führt zu Laufzeit den Programmcode aus, der die zu erstellenden Algorithmen der Robotersteuerung enthält. Es berechnet in Echtzeit die Stellsignale der Aktoren in Abhängigkeit der Messsignale und Bedienervorgaben. Dazu besitzt es neben einer leistungsstarken Recheneinheit eine Vielzahl von digitalen und analogen Ein- und Ausgängen als auch einen CAN-Bus-Controller.

Der BHA ist ein Handhabungssystem mit Greifer als Endeffektor. Wie Abbildung 2 zeigt, besteht der Arm des Manipulators aus drei Sektionen, welche seriell angeordnet sind. Jede Sektion verfügt über drei balgartige Aktoren, die individuell mit Druckluft befüllt werden können. Die Sektion mit dem niedrigsten Sektionsindex i ist mit dem Fundament verbunden. Im Anschluss an die Sektion mit dem höchsten Index ist ein Werkzeug angebracht. Da dieses austauschbar ist, wird im Weiteren davon ausgegangen, dass sich der zu positionierende Tool Center Point (TCP) im Mittelpunkt der Werkzeughalterung befindet.

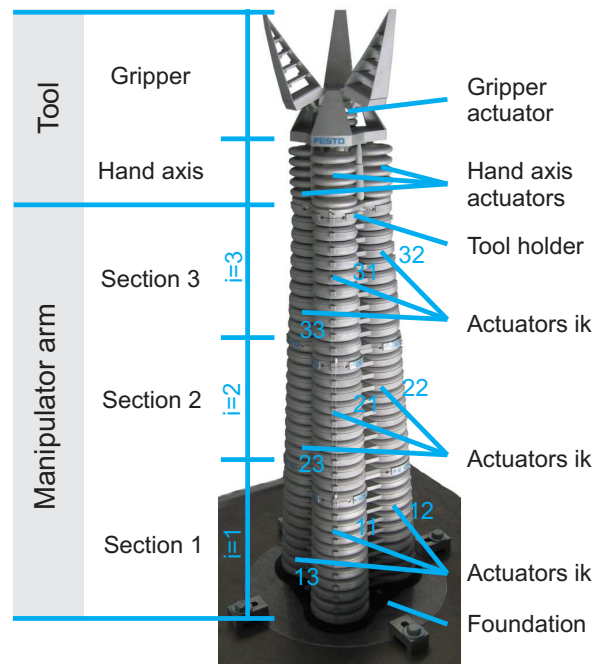


Abbildung 2: Komponenten des BHAs

Betrachtet man eine Sektion genauer, so gestaltet sich deren Aufbau wie in Abbildung 3 dargestellt. Jede Sektion besteht aus einem unteren und oberen Verbindungselement, dem Fuß und dem Kopf,

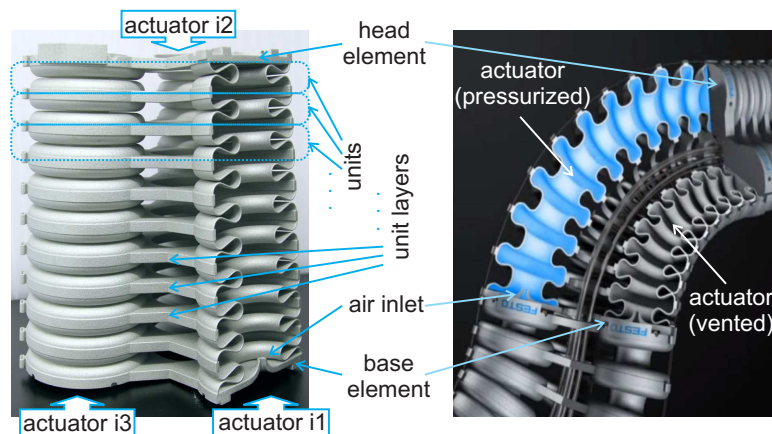


Abbildung 3: Querschnitte durch die Aktoren einer Sektion

als auch aus drei im 120° Winkel zueinander positionierten Balgaktoren. Werden die Balgaktoren mit Druckluft befüllt, so längen sich diese aus. Dabei führt eine unsymmetrische Längung der Aktoren zu einer Biegung der gesamten Sektion. Die individuelle Längung jedes Balgs kann mit Hilfe eines Seilzugpotentiometers pro Balg bestimmt werden. Dazu befinden sich Seilführungen an der Außenseite der Bälge. Eine detaillierte Darstellung der Installation des Seilzugmesssystems ist in Abbildung 4 aufgezeigt.

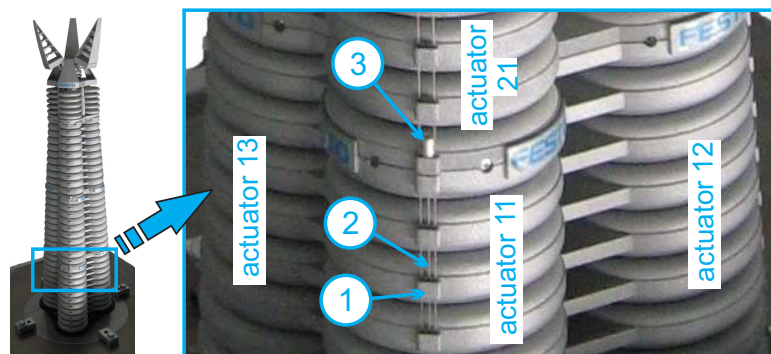


Abbildung 4: Installation des Seilzugmesssystems: ① Seilzugführung, ② Seilzüge der Aktoren 11, 21, und 31, ③ Abschlussbefestigung des Seilzugs von Aktor 11

3 Grundlagen der Robotik

Um die Kinematik des Bionischen Handling-Assistenten beschreiben zu können, gilt es einige grundlegende Methoden der Robotik anzuwenden. Diese sind nachfolgend beschrieben.

3.1 Darstellung von Position und Orientierung

Die Beschreibung der Pose¹ eines beliebigen Objekts im Raum hängt vom gewählten Koordinatensystem ab. Werden mehr als ein Koordinatensystem definiert, so ist das Bezugssystem stets zu kennzeichnen.

3.1.1 Position

Die Position eines Punktes P , bezogen auf das Koordinatensystem \mathcal{R}_j , wird im Folgenden durch den Ortsvektor

$${}^j\mathbf{P} = \begin{bmatrix} P_x & P_y & P_z \end{bmatrix}^T \quad (1)$$

beschrieben. Der Superskript auf der linken Seite kennzeichnet dabei das verwendete Bezugssystem.

3.1.2 Orientierung

Die Orientierung eines Objekts lässt sich mit Hilfe der Drehmatrix ${}^j\mathbf{C}_i$ angeben.

$${}^j\mathbf{C}_i = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \quad (2)$$

Sie stellt eine eindeutige Beschreibung der Rotation des Koordinatensystems \mathcal{R}_i bezüglich dem Koordinatensystem \mathcal{R}_j dar. Für die Berechnung der Drehmatrix existieren eine Vielzahl von unterschiedlichen Methoden. Zwei in der Robotik gängige Methoden sind die Verwendung von Eulerwinkeln als auch die Achse/Winkel Darstellung.

Eulerwinkel Eine Drehung im Raum kann durch drei Eulerwinkel dargestellt werden. Diese beschreiben die Drehungen um je eine Koordinatenachse in einer fest definierten Reihenfolge. Je nach Wahl der Achsen gibt es 12 Möglichkeiten die Eulerwinkel zu definieren. Die Gesamtdrehmatrix ergibt sich durch sukzessive Multiplikation der einzelnen Drehmatrizen. Dabei beschreibt

$$\mathbf{C}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad \text{eine Drehung um die x-Achse,} \quad (3)$$

¹Die Pose definiert die Position und Orientierung eines Objekts.

$$\mathbf{C}_y(\beta) = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \quad \text{eine Drehung um die y-Achse und} \quad (4)$$

$$\mathbf{C}_z(\gamma) = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{eine Drehung um die z-Achse.} \quad (5)$$

Achse/Winkel Ebenfalls kann eine Drehung im Raum durch eine normierte Drehachse \mathbf{n} und einen Drehwinkel ϕ beschrieben werden. Die Drehmatrix berechnet sich dabei wie folgt:

$$\mathbf{C}_{\mathbf{n}}(\phi) = \begin{bmatrix} c + n_x^2(1-c) & n_x n_y(1-c) - n_z s & n_x n_z(1-c) + n_y s \\ n_y n_x(1-c) + n_z s & c + n_y^2(1-c) & n_y n_z(1-c) - n_x s \\ n_z n_x(1-c) - n_y s & n_z n_y(1-c) + n_x s & c + n_z^2(1-c) \end{bmatrix} \quad (6)$$

mit

$$\mathbf{n} = \begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^T, \quad s = \sin(\phi), \quad c = \cos(\phi). \quad (7)$$

Die Bestimmung der Achse als auch des Winkels aus einer gegebenen Drehmatrix kann durch

$$\phi = \arccos\left(\frac{C_{11} + C_{22} + C_{33} - 1}{2}\right) \quad \text{und} \quad \mathbf{n} = \frac{1}{2 \sin(\phi)} \begin{bmatrix} C_{32} - C_{23} \\ C_{13} - C_{31} \\ C_{21} - C_{12} \end{bmatrix} \quad (8)$$

erfolgen.

Vorbereitungsaufgabe – V2.1 –

- Berechnen Sie die Drehmatrix $\mathbf{C}_{ZY'X''}$ in Abhängigkeit der Eulerwinkel α , β und γ für die ZY'X''-Konvention. Die Reihenfolge der Drehungen ist dabei wie folgt definiert:
 - Drehung um z-Achse (Z), Winkel γ
 - Drehung um resultierende y-Achse (Y'), Winkel β
 - Drehung um resultierende x-Achse (X''), Winkel α
- Bestimmen Sie die Drehmatrix $\mathbf{C}_{ZY'X''}$ für die Eulerwinkel $\alpha = \frac{\pi}{3}$, $\beta = 0$ und $\gamma = \frac{\pi}{4}$.
- Geben Sie die Drehachse und den Drehwinkel für die Drehmatrix aus (b) an und überprüfen Sie Ihr Ergebnis mit der Achse/Winkel Darstellung.

3.2 Koordinatentransformation

Die Transformation zwischen zwei Koordinatensystemen besteht im Allgemeinen aus einer Translation und einer Rotation. Ist der Ortsvektor ${}^0\mathbf{P}$ des Punktes P bezüglich des Koordinatensystems \mathcal{R}_0

gegeben, und möchte man diesen im Koordinatensystem \mathcal{R}_1 darstellen, so lautet die Umrechnungsvorschrift

$${}^1\mathbf{P} = {}^1\mathbf{C}_0 {}^0\mathbf{P} + {}^1\mathbf{O}_0. \quad (9)$$

\mathbf{O}_0 bezeichnet den Ortsvektor und \mathbf{C}_0 die Drehmatrix des Koordinatensystems \mathcal{R}_0 .

Die Orientierung eines Objekts ${}^0\mathbf{C}_O$, bezogen auf das Koordinatensystem \mathcal{R}_0 , transformiert sich wie folgt:

$${}^1\mathbf{C}_O = {}^1\mathbf{C}_0 {}^0\mathbf{C}_O \quad (10)$$

3.2.1 Homogene Transformation

Die homogene Transformation ist eine Erweiterung der oben beschriebenen Koordinatentransformation, welche vor allem die Berechnung der in der Robotik häufig vorkommenden kinematischen Ketten vereinfacht. Eine Koordinatentransformation kann damit durch eine einzige Matrix-Vektor-Multiplikation beschrieben werden. Die homogene Repräsentation des Ortsvektors \mathbf{P} lautet

$$\underline{\mathbf{P}} = \begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix}. \quad (11)$$

Die homogene Transformationsmatrix, welche ursprünglich aus dem Feld der Computergrafik entstammt, setzt sich wie folgt zusammen:

$$\mathbf{H} = \begin{bmatrix} \text{Rotation} & \text{Translation} \\ \text{Perspektive} & \text{Skalierung} \end{bmatrix} \quad (12)$$

Damit resultiert eine Koordinatentransformation des Ortsvektors \mathbf{P} von Koordinatensystem \mathcal{R}_0 nach \mathcal{R}_1 zu

$${}^1\underline{\mathbf{P}} = {}^1\mathbf{H}_0 {}^0\underline{\mathbf{P}} = \begin{bmatrix} {}^1\mathbf{C}_0 & {}^1\mathbf{O}_0 \\ 0 & 1 \end{bmatrix} {}^0\underline{\mathbf{P}} = \begin{bmatrix} {}^1\mathbf{C}_0 {}^0\mathbf{P} + {}^1\mathbf{O}_0 \\ 1 \end{bmatrix}. \quad (13)$$

Vorbereitungsaufgabe – V2.2 –

Gegeben ist ein serieller Roboterarm der aus Basis (\mathcal{R}_B), Oberarm (\mathcal{R}_O), Unterarm (\mathcal{R}_U) und Greifhand (\mathcal{R}_G) besteht, die jeweils mit einem körperfesten Koordinatensystem versehen sind. Für die Position des Effektors \mathbf{P}_e erhält man die folgende Transformationsvorschrift:

$${}^B\mathbf{P}_e = {}^B\mathbf{C}_O \left({}^O\mathbf{C}_U \left({}^U\mathbf{C}_G {}^G\mathbf{P}_e + {}^U\mathbf{O}_G \right) + {}^O\mathbf{O}_U \right) + {}^B\mathbf{O}_O \quad (14)$$

Berechnen Sie die homogene Repräsentation der Endeffektorposition ${}^B\underline{\mathbf{P}}_e$ unter Verwendung von 3 homogenen Transformationsmatrizen und verifizieren Sie Ihr Ergebnis durch Ausmultiplizieren dieser.

3.3 Kinematik

In der Robotik beschreibt die Kinematik die Bewegung einer mechanischen Struktur ohne Kräfte und/oder Momente zu berücksichtigen. Zentrales Interesse bei der Beschreibung der Roboterbewegung nimmt dabei die Lösung des direkten als auch des inversen kinematischen Problems ein, welches einerseits durch die Vorwärts-, andererseits durch die Rückwärtskinematik beschrieben wird. Die Komplexität der Kinematiken ist dabei stark von der kinematischen Struktur des Roboters abhängig.

3.3.1 Kinematische Strukturen

Aufgrund der Vielzahl von Aufgaben welche von Robotern durchgeführt werden, existieren sehr unterschiedliche Realisierungsformen der kinematischen Roboterstruktur. Prinzipiell zu unterscheiden sind dabei

1. parallele Mechanismen und
2. serielle Mechanismen.

Erstgenannte weisen geschlossene Ringe in der Führungskette der Gelenke auf. Die kinematische Kette serieller Mechanismen ist hingegen gradlinig oder baumartig verzweigt und weist keine geschlossenen Strukturen auf.

3.3.2 Vorwärtskinematik

In der Robotik existiert im Allgemeinen ein Koordinatensystem, indem die Bewegung des Roboters vorgeben wird (Aufgaben-Koordinatensystem), als auch das Gelenkkoordinatensystem, indem der Roboter die Anweisungen ausführt. Man spricht von einer Vorwärtskinematik oder auch dem direkten kinematischen Problem, wenn aus dem Gelenkzustand des Roboters die Pose der Werkzeugspitze im Aufgaben-Koordinatensystem bestimmt wird. Somit ist die Vorwärtskinematik eine Abbildung Ψ , die den Gelenkzustand \mathbf{q} auf die Pose \mathbf{x} abbildet

$$\mathbf{x} = \Psi(\mathbf{q}). \quad (15)$$

Das Bild der Vorwärtskinematik ist der maximal mögliche Arbeitsraum des Roboters. Zudem wird ein Roboter als *redundant* bezeichnet, wenn die Abbildungsvorschrift nicht injektiv ist. In anderen Worten bedeutet dies, dass mindestens zwei oder mehr Gelenkkonfigurationen zu der selben Pose im Aufgaben-Koordinatensystem führen.

3.3.3 Rückwärtskinematik

Als Rückwärtskinematik oder inverses kinematisches Problem wird die Berechnung des Gelenkzustands aus der Pose des Endeffektors bezeichnet. Da für redundante Kinematiken keine eindeutige Transformationsvorschrift zur Lösung des inversen kinematischen Problems existiert, gibt es verschiedene Ansätze welche es ermöglichen Beschränkungen, Redundanz und Singularitäten bei der

Rückwärtstransformation zu berücksichtigen. Besonders die gezielte Nutzung der zusätzlichen Freiheitsgrade redundanter Roboter steht im Interesse vieler Ansätze. Darunter zählen Sekundärziele wie die Umgehung von Hindernissen, die Steigerung der Geschicklichkeit des Roboters oder die Meidung von singulären Gelenkstellungen. Dabei unterscheidet man drei Klassen von Ansätzen:

1. Einschränkung der Freiheitsgrade
2. Lokale Optimierung
3. Globale Optimierung

Die Einschränkung der Freiheitsgrade ist die einfachste Variante um redundante Lösungen der Rückwärtskinematik zu umgehen. Hierbei werden die Freiheitsgrade der Redundanz durch eine fest vorgegebene Heuristik eingeschränkt. Die Vorteile der Redundanz werden dadurch jedoch vollkommen verschenkt.

Lokale Optimierungsalgorithmen werden bei redundanten Manipulatoren am häufigsten zur Lösung des inversen kinematischen Problems eingesetzt. Diese sind in der Lage die Redundanz entsprechend eines Gütekriteriums, welches lokal minimiert wird, zu nutzen. Durch den geringen Rechenzeitbedarf eignen sie sich zudem für den Echtzeitbetrieb.

Die globale Optimierung der Rückwärtskinematik ist im Allgemeinen sehr rechenzeitaufwändig. Verwendung findet sie deshalb vor allem bei wiederkehrenden Anwendungen für die die Solltrajektorie des Manipulators offline berechnet werden kann.

4 Kinematik des BHAs

Für die kinematische Modellierung des BHAs sind zuerst die verwendeten Koordinatensysteme zu definieren. Anschließend erfolgt die Herleitung der Vorwärts- als auch Rückwärtskinematik des Manipulators. Hinweise zur Implementierung beider Kinematiken finden sich am Ende des Kapitels.

4.1 Definition der Koordinatensysteme

Gemäß dem modularen Aufbau des Manipulators (vgl. Abbildung 2) ist es sinnvoll, die in Abbildung 5 aufgezeigten, körperfesten Koordinatensysteme einzuführen. Jede Sektion i des BHAs besitzt ein

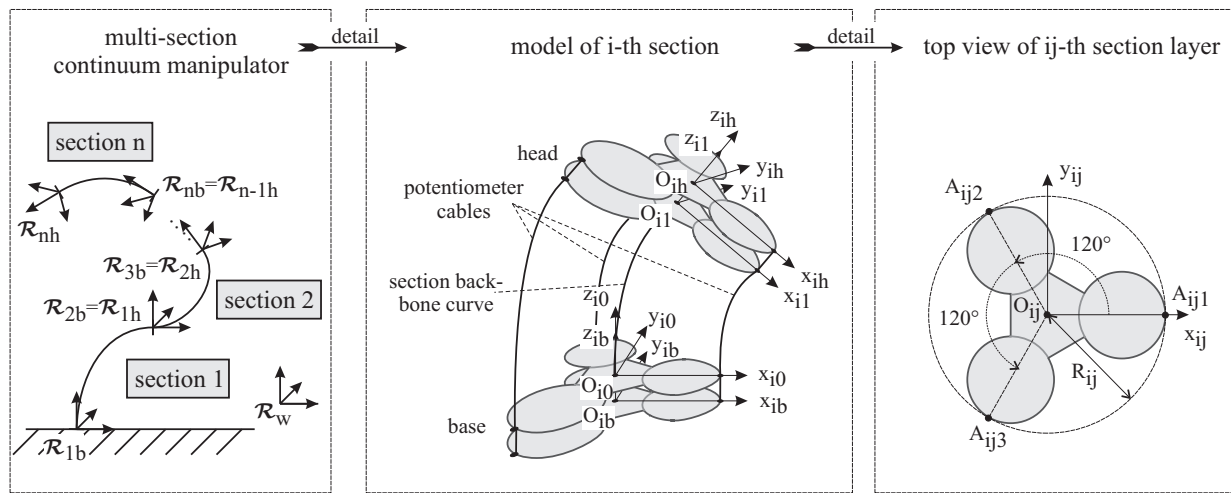


Abbildung 5: Definition der Koordinatensysteme

Fuß- \mathcal{R}_{ib} und Kopfkoordinatensystem \mathcal{R}_{ih} . Durch die serielle Anordnung der Sektionen ist das Kopfkoordinatensystem der $i - 1$ -ten Sektion identisch dem Fußkoordinatensystem der i -ten Sektion. Zudem ist das Fußkoordinatensystem \mathcal{R}_{1b} der 1. Sektion identisch der Aufhängung des Manipulators. Das Kopfkoordinatensystem der letzten Sektion \mathcal{R}_{nh} (beim BHA ist $n = 3$) definiert den TCP.

Jede einzelne Sektion i verfügt darüber hinaus über zwei weitere Koordinatensysteme \mathcal{R}_{i0} und \mathcal{R}_{i1} . Diese definieren den Beginn und das Ende des biegbaren Aktorbereichs einer Sektion. Sie grenzen damit den Aktorbereich vom starren Fuß und Kopf ab.

Der Ursprung der Koordinatensysteme liegt jeweils im Flächenschwerpunkt der Draufsicht auf eine Sektion. Die Ausrichtung der Achsen ist derart gewählt, dass die Seilzugführung des 1. Balgaktors auf der x-Achse zum liegen kommt. Die z-Achse weist lotrecht der Ebene j ($j = \{b, 0, 1, h\}$), welche durch die Seilzugführungen der 3 Balgaktoren aufgespannt wird, in Richtung des Sektionskopfs. Die y-Achse vervollständigt das Koordinatensystem sodass ein karthesisches Rechtssystem entsteht. Somit liegt sie in der von den Seilzugführungen der Balgaktoren aufgespannten Ebene. Die Indizierung k der 3 Aktoren erfolgt im mathematisch positiven Sinn, beginnend bei $k = 1$.

4.2 Vorwärtskinematik

Für die kinematische Modellierung des BHAs wird angenommen, dass sich die Deformation des Aktuatorbereichs einer Sektion durch eine ideale Kreisbahn abschätzen lässt. Diese Annahme des konstanten Biegeradius stellt eine Vereinfachung der realen Kinematik dar. Durch die Verjüngung der Sektionen in Richtung des Kopfs nimmt die tatsächliche Krümmung der Sektionen Richtung Endeffektor kontinuierlich zu. Da die Verjüngung einer Sektion im Vergleich zum Aktuatorabstand jedoch klein ist, sind nur geringe Abweichungen zu erwarten. Im Folgenden wird somit davon ausgegangen, dass sich eine Sektion nicht verjüngt.

Die zu beschreibende Kinematik des TCP ist identisch der Bewegung des Kopfkoordinatensystems der 3. Sektion. Verglichen mit Abbildung 2 ist folglich die Vorwärtskinematik des Manipulatorarms aufzustellen.

4.2.1 Manipulatorarm

Mit der Vorwärtskinematik des Manipulatorarms wird die Pose des Effektors (TCP) ausgehend von dem Gelenkzustand des BHAs beschrieben. Dieser besteht aus den neun Balglängen der drei Sektionen, die in Abhängigkeit der gemessenen Seilzuglängen bestimmt werden können. Als Balglänge wird die Länge der Balgsymmetrielinie bezeichnet, die durch die jeweiligen Flächenschwerpunkte der Balgquerschnitte verläuft. Die Vorwärtskinematik des Manipulatorarms setzt sich aus den Kinematiken der drei Sektionen zusammen, welche seriell miteinander verbunden sind. Die Homogene Transformation des Manipulatorarms ${}^{1b}\mathbf{M}_{3h}$ berechnet sich somit zu

$${}^{1b}\mathbf{M}_{3h} = {}^{1b}\mathbf{S}_{1h} {}^{2b}\mathbf{S}_{2h} {}^{3b}\mathbf{S}_{3h} = {}^{1b}\mathbf{S}_{1h} {}^{1h}\mathbf{S}_{2h} {}^{2h}\mathbf{S}_{3h}. \quad (16)$$

Dabei beschreibt ${}^{ib}\mathbf{S}_{ih}$ die Homogene Transformation der i -ten Sektion, welche nachfolgend beschrieben wird.

4.2.2 Sektion

Die Kinematik der Sektion i setzt sich aus den Untereinheiten konstantem Fuß, variablem Aktuatorbereich und konstantem Kopf zusammen. Damit resultiert die Homogene Transformation einer Sektion zu

$${}^{ib}\mathbf{S}_{ih} = {}^{ib}\mathbf{U}_{i0} {}^{i0}\mathbf{U}_{i1} {}^{i1}\mathbf{U}_{ih}. \quad (17)$$

Die homogenen Transformationen des Fußes ${}^{ib}\mathbf{U}_{i0}$ als auch des Kopfs ${}^{i1}\mathbf{U}_{ih}$ sind reine Translationen in z-Richtung des jeweiligen körperfesten Koordinatensystem \mathcal{R}_{ib} bzw. \mathcal{R}_{i1} . Sie werden wie folgt beschrieben:

$${}^{ib}\mathbf{U}_{i0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \ell_{ib} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^{i1}\mathbf{U}_{ih} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \ell_{ih} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

Die homogene Transformation des Aktuatorbereichs spiegelt die Deformation der Sektion wider. Unter der Annahme, dass sich der Aktuatorbereich mit konstanter Krümmung verbiegt, resultiert dessen homogene Transformation zu

$${}^{i0}\mathbf{U}_{i1}(\phi, \kappa, \ell) = \begin{bmatrix} \cos^2 \phi (\cos \kappa \ell - 1) + 1 & \sin \phi \cos \phi (\cos \kappa \ell - 1) & \cos \phi \sin \kappa \ell & \frac{\cos \phi (1 - \cos \kappa \ell)}{\kappa} \\ \sin \phi \cos \phi (\cos \kappa \ell - 1) & \cos^2 \phi (1 - \cos \kappa \ell) + \cos \kappa \ell & \sin \phi \sin \kappa \ell & \frac{\sin \phi (1 - \cos \kappa \ell)}{\kappa} \\ -\cos \phi \sin \kappa \ell & -\sin \phi \sin \kappa \ell & \cos \kappa \ell & \frac{\sin \kappa \ell}{\kappa} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (19)$$

mit

$$\phi = \phi_i, \quad \kappa = \kappa_i \quad \text{und} \quad \ell = \ell_i. \quad (20)$$

Dabei werden die Parameter ϕ_i , κ_i und ℓ_i als Zustand des Konfigurationsraums der i -ten Sektion bezeichnet. Anders ausgedrückt definiert der Konfigurationszustand das Kreissegment, welches zur Beschreibung der Deformation des Aktuatorbereichs verwendet wird. ϕ_i ist dabei der Winkel zwischen der x-Achse und der auf die xy-Ebene projizierten Biegelinie der Sektion. Die Krümmung κ_i entspricht dem inversen Radius und ℓ_i ist die Kreisbogenlänge der Biegelinie. Eine graphische Darstellung der Konfigurationsparameter ist in Abbildung 6 zu finden.

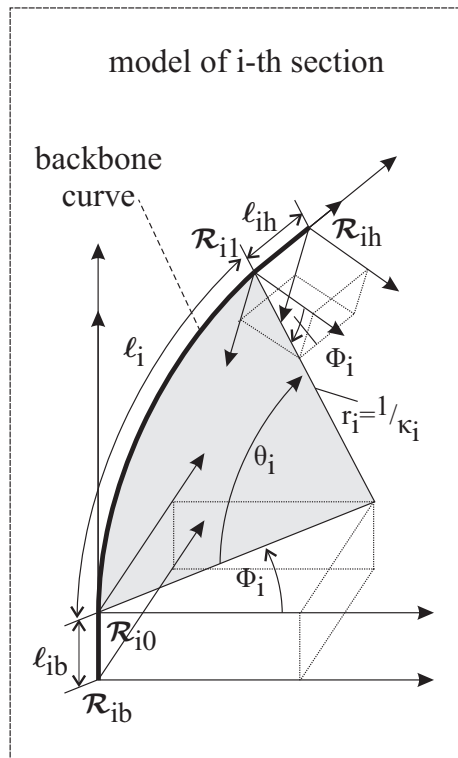


Abbildung 6: Konfigurationsparameter der Biegelinie der i -ten Sektion

Eine ausführliche Herleitung der homogenen Transformation für eine Sektion mit konstanter Krümmung kann in [WJ10] nachgelesen werden.

Hinweis: Die Krümmung κ_i (auch inverse Radius) wird verwendet, da dieser bei einer undeformierten Sektion zu $\kappa_i = 0$ wird. Der Kreisradius r_i wäre in dieser gängigen Position unendlich und somit numerisch nicht darstellbar.

Unter Verwendung der Konfigurationsparameter ist die Kinematik der Sektion, und damit auch des Manipulatorarms, vollständig durch die Gleichungen 16-19 beschrieben. Ziel des direkten kinematischen Problems ist jedoch die Berechnung der TCP-Kinematik in Abhängigkeit der Balglängen und nicht des Konfigurationszustands. Somit gilt es im Folgenden den Zusammenhang zwischen dem Gelenkzustand und dem Konfigurationszustand herzuleiten.

Wird die Biegelinie einer Sektion zum Kreisbogen verformt, so wird der gesamte Aktuatorbereich über einen Zylinder gekrümmt. Die Seilzüge liegen jeweils vollständig in einer Ebene parallel zur Ebene der Biegelinie der Sektion und beschreiben in dieser ebenfalls ein Kreissegment mit dem Kreisbogen $\theta_i = \kappa_i \ell_i$. Ihr individueller Radius ergibt sich als Summe des Radius der Biegelinie r_i abzüglich der Projektion der lokalen Position der Seilzugführung auf die Ebene der Biegelinie Δr_{ik} . Abbildung 7 verdeutlicht die geometrische Beziehung zwischen der Biegelinie einer Sektion und den Seilzügen der Aktuatoren.

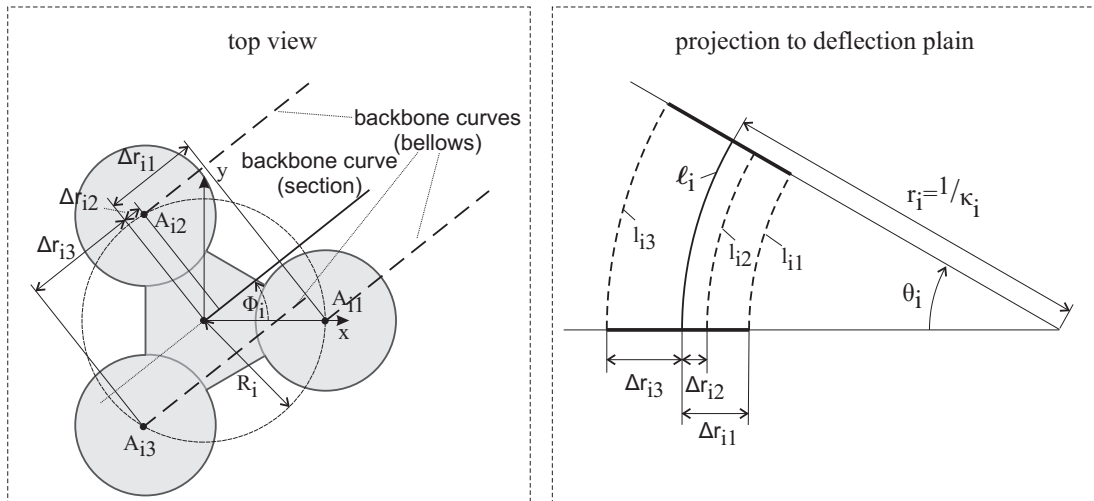


Abbildung 7: Geometrischer Zusammenhang zwischen der Biegelinie der Sektion und den Biegelinien der Aktuatoren.

Mathematisch ausgedrückt resultiert die Projektion der lokalen Position der Balgsymmetrielinie zu

$$\Delta r_{ik} = \begin{bmatrix} \cos(\phi_i) & \sin(\phi_i) & 0 \end{bmatrix} \mathbf{A}_{ik}, \quad \mathbf{A}_{ik} = R_i \begin{bmatrix} \cos\left(\frac{2\pi}{3}(k-1)\right) \\ \sin\left(\frac{2\pi}{3}(k-1)\right) \\ 0 \end{bmatrix}, \quad k = \{1, 2, 3\}. \quad (21)$$

Die Biegelinie L_{ik} der Bälge setzt sich aus den Anteilen des Fußes l_{ib} , des Aktuatorteils l_{ik} und des Kopfs l_{ih} zusammen:

$$L_{ik} = l_{ib} + l_{ik} + l_{ih} \quad \text{mit} \quad l_{ik} = (r_i - \Delta r_{ik})\theta_i, \quad k = \{1, 2, 3\} \quad (22)$$

Wird Gleichung 21 in Gleichung 22 eingesetzt und die Krümmung κ_i als auch die Kreisbogenlänge ℓ_i zur Beschreibung der Konfigurationsparameter verwendet,

$$r_i = \frac{1}{\kappa_i}, \quad \theta_i = \frac{\ell_i}{r_i} = \kappa_i \ell_i, \quad (23)$$

so resultiert ein Gleichungssystem, welches die Biegelinienlänge der Aktuatoren l_{ik} in Abhängigkeit des Konfigurationszustands ϕ_i , κ_i und ℓ_i beschreibt.

$$l_{i1} = \left(\frac{1}{\kappa_i} - R_i \cos(\phi_i) \right) \kappa_i \ell_i \quad (24a)$$

$$l_{i2} = \left(\frac{1}{\kappa_i} + \frac{1}{2} R_i \cos(\phi_i) - \frac{\sqrt{3}}{2} R_i \sin(\phi_i) \right) \kappa_i \ell_i \quad (24b)$$

$$l_{i3} = \left(\frac{1}{\kappa_i} + \frac{1}{2} R_i \cos(\phi_i) + \frac{\sqrt{3}}{2} R_i \sin(\phi_i) \right) \kappa_i \ell_i \quad (24c)$$

Wird das Gleichungssystem 24 nach dem Konfigurationszustand aufgelöst, so resultiert

$$\phi_i = \arctan2 \left(\sqrt{3} (l_{i3} - l_{i2}), l_{i2} + l_{i3} - 2l_{i1} \right) \quad (25a)$$

$$\kappa_i = \frac{2\sqrt{l_{i1}^2 + l_{i2}^2 + l_{i3}^2 - l_{i1}l_{i2} - l_{i2}l_{i3} - l_{i3}l_{i1}}}{(l_{i1} + l_{i2} + l_{i3}) R_i} \quad (25b)$$

$$\ell_i = \frac{1}{3} (l_{i1} + l_{i2} + l_{i3}). \quad (25c)$$

Bemerkung 1: Wikipediaauszug zur Definition des arctan2: *Der arctan2 dient bei der Umrechnung von kartesischen Koordinaten $P(x; y)$ in Polarkoordinaten $P(r; \varphi)$ der Ermittlung des Winkels φ . Da der einfache Arkustangens nicht die Möglichkeit bietet, den Winkel im korrekten Quadranten zu ermitteln, und außerdem die Tangensfunktion für einen Funktionswert von $\pm \frac{\pi}{2}$ nicht umkehrbar ist, gibt es in vielen Programmiersprachen eine Funktion, die mit 2 Argumenten aufgerufen wird. Sie wird üblicherweise mit $\arctan2(y, x)$ bezeichnet. Die Funktion $\arctan2(y, x)$ kann über die folgende Eigenschaft definiert werden: Sind x, y reelle Zahlen und $r = \sqrt{x^2 + y^2}$, so gilt*

$$x = r \cdot \cos(\arctan2(y, x)), \quad y = r \cdot \sin(\arctan2(y, x)).$$

$(r, \arctan2(y, x))$ sind hierbei die Polarkoordinaten des Punktes mit den kartesischen Koordinaten (x, y) . Eine von mehreren in der Praxis vorkommenden Definitionen lautet:

$$\arctan2(y, x) := \begin{cases} \arctan \frac{y}{x} & \text{für } x > 0 \\ \arctan \frac{y}{x} + \pi & \text{für } x < 0, y \geq 0 \\ \arctan \frac{y}{x} - \pi & \text{für } x < 0, y < 0 \\ +\pi/2 & \text{für } x = 0, y > 0 \\ -\pi/2 & \text{für } x = 0, y < 0 \\ 0 & \text{für } x = 0, y = 0 \end{cases} \quad (26)$$

Bemerkung 2: Der Term unter der Wurzel von Gleichung 25b ist für alle Seillängenkombinationen nicht negativ. Er lässt sich als Quadrik $S^T Q S$ schreiben. Q ist positiv semidefinit. Die Eigenwerte sind $\lambda_{1,2} = 1.5$, $\lambda_3 = 0$. Der Eigenvektor zum Eigenwert λ_3 ist $v_3 = [1 \ 1 \ 1]^T$. Somit ist die Wurzel genau dann null wenn alle 3 Balglängen äquivalent sind.

Bemerkung 3: Die von Mathematica bzw. Maple ausgegebene Lösung für ϕ_i , hat in beiden Argumenten des $\arctan2$, im Nenner noch die in Bemerkung 2 behandelte Wurzel stehen. Da diese jedoch positiv ist, ändert sich das Vorzeichen und somit die Lösung durch weglassen derselben nicht. [Siehe Definition des $\arctan2$ nach Gleichung 26].

Bemerkung 4: Die Berechnung der Länge der Balgsymmetrielinie in Abhängigkeit der Seilzuglängen kann auch über geometrische Beziehungen der einzelnen Sektionen hergeleitet werden. Für kleine Krümmungen kann in erster Näherung von einer Identität der Seilzuglängen und Balgsymmetrielinie ausgegangen werden, andernfalls gibt es kleine Unterschiede. Für den Praktikumsversuch kann von gegebenen Balglängen ausgegangen werden.

Vorbereitungsaufgabe – V2.3 –

- Geben Sie die homogene Transformationsmatrix ${}^0\mathbf{H}_1(\phi)$ an, die eine Rotation um die z-Achse mit dem Winkel ϕ beschreibt.
- Gesucht ist die Transformationsmatrix ${}^1\mathbf{H}_2(r, \phi)$, welche die in Abbildung 8 dargestellte Beziehung zwischen den Koordinatensystemen \mathcal{R}_1 und \mathcal{R}_2 beschreibt.
- Berechnen Sie ${}^1\mathbf{H}_2(\kappa, \ell)$, indem Sie in die Transformationsmatrix von (b) die nachfolgenden zwei Beziehungen einsetzen:

$$r = \frac{1}{\kappa}, \quad \theta = \frac{\ell}{r} = \kappa\ell. \quad (27)$$

- Berechnen Sie ${}^0\mathbf{H}_3(\phi, \kappa, \ell) = {}^0\mathbf{H}_1(\phi){}^1\mathbf{H}_2(\kappa, \ell){}^2\mathbf{H}_3(\phi)$ mit ${}^2\mathbf{H}_3(\phi) = {}^0\mathbf{H}_1(-\phi)$ und vergleichen Sie Ihr Ergebnis mit Gleichung 19.

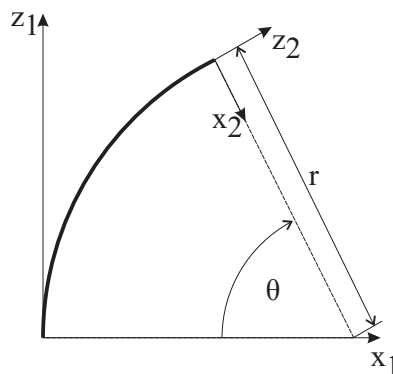


Abbildung 8: Biegung mit konstanter Krümmung in der xz-Ebene

4.2.3 Zusammenfassung

Die Berechnung der Vorwärtskinematik des BHAs erfolgt in zwei Schritten gemäß nachfolgendem Schaubild. Die Abbildung Ψ_{as2cs} berechnet den Zustand der konstanten Krümmungsradien der

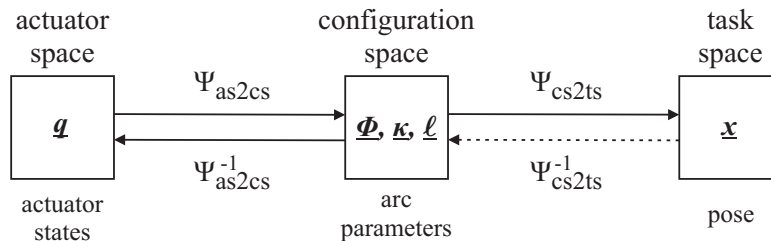


Abbildung 9: Schaubild der Vorwärts- und Rückwärtskinematik des BHAs

Sektionen ausgehend von den gemessenen Balglängen der Balgaktuatoren. Dazu wird das Gleichungssystem 25 verwendet. Im Anschluss daran wird die Pose des TCP durch die Abbildungsvorschrift Ψ_{cs2ts} bestimmt. Hierbei wird die durch den Konfigurationszustand parametrisierte Biegelinie des Manipulators mittels der Gleichungen 16-19 berechnet, an dessen Endpunkt sich der TCP des Manipulators befindet.

Die Rückwärtskinematik des Manipulators wird durch die beiden inversen Abbildungen Ψ_{as2cs}^{-1} und Ψ_{cs2ts}^{-1} beschrieben. Erstgenannte wurde bereits für die Herleitung der Vorwärtskinematik verwendet und ist identisch dem Gleichungssystem 24. Die inverse Abbildungsvorschrift Ψ_{cs2ts}^{-1} ist aufgrund der Redundanz des Manipulators nicht eindeutig. Geeignete Methoden zur Invertierung dieser werden im folgenden Abschnitt behandelt.

4.3 Rückwärtskinematik

Wie bereits erwähnt, existieren für die Bestimmung der Rückwärtskinematik redundanter Manipulatoren im Wesentlichen drei unterschiedliche Methoden. Im Rahmen dieses Praktikums wird sowohl die Methode der Einschränkung der Freiheitsgrade, als auch eine Methode der lokalen Optimierung behandelt.

Um die Komplexität der Rückwärtskinematik zu reduzieren, wird das zu lösende inverse Problem im Weiteren wie folgt definiert: *Zu berechnen ist der Gelenkzustand des BHAs für eine beliebig vorgegebene, erreichbare Soll-Position des TCP. Die Orientierung des TCP ist dabei unerheblich.*

4.3.1 Einschränkung der Freiheitsgrade

Ziel dieser Methode ist es durch eine Heuristik die durch die Redundanz des Manipulators hervorgerufenen Freiheitsgrade bei der Rückwärtstransformation einzuschränken. Die Vorgehensweise zur Berechnung der Rückwärtskinematik erfolgt in drei Schritten.

- (1) Festlegung der Heuristik
- (2) Einsetzen der Heuristik in die Vorwärtskinematik

(3) Auflösen der Vorwärtskinematik nach den Gelenkkoordinaten

Bei der Aufstellung der Heuristik ist darauf zu achten, dass die Anzahl der verbleibenden Freiheitsgrade der Gelenkkoordinaten der Anzahl der Freiheitsgrade der Aufgabenkoordinaten entspricht. Ist eine analytische Lösung des Gleichungssystems von Schritt (3) nicht möglich, können auch iterative Verfahren, wie bspw. das Newton-Raphson-Verfahren, zur Lösung dieses angewandt werden.

Hinweise zur Implementierung

Die Anzahl der Freiheitsgrade der Aufgabenkoordinaten für die oben definierte Aufgabenstellung ist $n_A = 3$. Dies sind die x-, y- und z-Koordinaten der gewünschten TCP Position. Die Anzahl der Freiheitsgrade der Gelenkkoordinaten ist $n_G = 9$. Diese entsprechen den Längen der neun Balgaktoren L_{ik} . Da zudem eine eindeutige Abbildungsvorschrift zwischen dem Gelenkzustand und dem Konfigurationszustand der konstanten Krümmungsradien der BHA-Sektionen existiert (siehe Abbildung 9 oder Gleichung 24), können statt der Gelenkkoordinaten auch die 9 Konfigurationsparameter ϕ_i, κ_i und ℓ_i berechnet werden. Um die Rückwärtskinematik eindeutig lösen zu können, ist die Heuristik so zu wählen, dass 6 Freiheitsgrade der Gelenkkoordinaten eingeschränkt werden.

Im Folgenden wird die inverse Abbildung $\Psi_{cs2ts,EF}^{-1}$ für folgende Einschränkung der Konfigurationsparameter

$$\phi_i = \phi, \quad \kappa_i = \kappa, \quad \ell_i = \ell, \quad i = \{1, 2, 3\} \quad (28)$$

hergeleitet. Diese Heuristik hat zur Folge, dass sich jede Sektion des BHAs identisch verformen wird. Eingesetzt in Gleichung 16 - 19 resultiert die TCP Position \mathbf{P}_{TCP} zu

$${}^{1b}\mathbf{P}_{TCP} = {}^{1b}\mathbf{O}_{3h} = \begin{bmatrix} x_{TCP} \\ y_{TCP} \\ z_{TCP} \end{bmatrix} = \begin{bmatrix} \cos(\phi) \left(\frac{1 - \cos(3\kappa\ell)}{\kappa} + \sum_{i=1}^3 \sin((i-1)\kappa\ell) \ell_{ib} + \sum_{i=1}^3 \sin(i\kappa\ell) \ell_{ih} \right) \\ \sin(\phi) \left(\frac{1 - \cos(3\kappa\ell)}{\kappa} + \sum_{i=1}^3 \sin((i-1)\kappa\ell) \ell_{ib} + \sum_{i=1}^3 \sin(i\kappa\ell) \ell_{ih} \right) \\ \frac{\sin(3\kappa\ell)}{\kappa} + \sum_{i=1}^3 \cos((i-1)\kappa\ell) \ell_{ib} + \sum_{i=1}^3 \cos(i\kappa\ell) \ell_{ih} \end{bmatrix}. \quad (29)$$

Eine analytische Lösung des nichtlinearen Gleichungssystems 29 nach dem eingeschränkten Konfigurationszustand ϕ, κ und ℓ ist nicht möglich. Lediglich der Winkel der Biegelinie ϕ lässt sich analytisch angeben. Er ergibt sich zu

$$\phi = \arctan2(y_{TCP}, x_{TCP}). \quad (30)$$

κ und ℓ müssen numerisch anhand des nachfolgenden Gleichungssystems bestimmt werden:

$$+\sqrt{x_{TCP}^2 + y_{TCP}^2} = \frac{1 - \cos(3\kappa\ell)}{\kappa} + \sum_{i=1}^3 \sin((i-1)\kappa\ell) \ell_{ib} + \sum_{i=1}^3 \sin(i\kappa\ell) \ell_{ih} \quad (31a)$$

$$z_{TCP} = \frac{\sin(3\kappa\ell)}{\kappa} + \sum_{i=1}^3 \cos((i-1)\kappa\ell) \ell_{ib} + \sum_{i=1}^3 \cos(i\kappa\ell) \ell_{ih} \quad (31b)$$

4.3.2 Lokale Optimierung

Die Idee der lokalen bzw. ableitungsbasierten Rückwärtskinematik besteht darin, eine in \dot{x} lineare Abbildung, der Posengeschwindigkeit \dot{x} auf die Gelenkgeschwindigkeit \dot{q} zu finden.

$$\dot{q} = K(q)\dot{x} \quad (32)$$

Der einfachste Ansatz dafür ist die Pseudoinverse der Ableitung der bereits bekannten Vorwärtskinematik zu verwenden. Gleichung (15) ergibt abgeleitet.

$$\dot{x} = \left. \frac{\partial \Psi(q)}{\partial q} \right|_q \dot{q} = J(q)\dot{q} \quad (33)$$

Die Matrix der partiell nach den Gelenkkoordinaten abgeleiteten Abbildungsvorschrift der Vorwärtstransformation wird im weiteren als Jacobimatrix $J(q)$ bezeichnet. Somit ist die lokale Rückwärtskinematik gegeben durch:

$$\dot{q} = J(q)^+ \dot{x} \quad (34)$$

$J(q)^+$ bezeichnet dabei die Pseudoinverse der Jacobimatrix. Die Rückwärtskinematik ist exakt, wenn Gleichung (33) eine Lösung besitzt. Die Norm von \dot{q} ist im Lösungsraum minimal. Hat $J(q)$ vollen Zeilenrang, kann die Norm durch $\|\dot{q}\| \leq \frac{1}{\sigma(J)} \|\dot{x}\|$ abgeschätzt werden. Die Norm wird nur lokal und nicht global minimiert. Sie kann in der Nähe einer Singularität unendlich groß werden.

Die Moore-Penrose Pseudoinverse

Idee: Ein lineares Gleichungssystem kann keine, eine oder unendliche viele Lösungen besitzen. Eine Inverse für das Gleichungssystem

$$Ax = b, \quad b \in \mathbb{R}^m, \quad A \in \mathbb{R}^{n \times m} \quad (35)$$

lässt sich nur dann eindeutig bestimmen, wenn A quadratisch ist und vollen Rang hat. Die Moore-Penrose Pseudoinverse A^+ , ist eine Verallgemeinerung des Konzepts der Inversen auf den Fall, dass A nicht quadratisch ist, es keine eindeutige oder unendlich viele Lösungen gibt.

Definition: Die Moore-Penrose Pseudoinverse kann auf zwei äquivalente Arten definiert werden.

1. Mithilfe der Singulärwertzerlegung $A = U\Sigma V^*$.

$$A^+ := V\Sigma^+U^* = V \left[\begin{array}{ccc|c} \frac{1}{\sigma_1} & & & 0 \\ & \ddots & & \\ & & \frac{1}{\sigma_r} & 0 \\ \hline 0 & & & 0 \end{array} \right] U^* \quad (36)$$

2. Durch 4 eindeutig lösbare Matrixgleichungen.

$$\begin{array}{ll} 1. & AA^+A = A \\ 2. & (AA^+)^T = AA^+ \\ 3. & A^+AA^+ = A^+ \\ 4. & (A^+A)^T = A^+A \end{array} \quad (37)$$

Eigenschaften der Pseudolösung $\tilde{x} = A^+b$:

- (i) Existiert die Inverse, so stimmen die Moore-Penrose Pseudoinverse und die Inverse überein.
- (ii) Besitzt das Gleichungssystem (35) keine Lösung, so ist \tilde{x} die Lösung, die den quadratischen Abstand zwischen Ax und b minimiert.

$$\tilde{x} = \min_{x \in \mathbb{R}^n} \|b - Ax\|_2 \quad (38)$$

- (iii) Besitzt das Gleichungssystem (35) unendlich viele Lösungen, so ist \tilde{x} die Lösung mit der minimalen quadratischen Norm.

$$\tilde{x} = \min_{x \in \mathbb{R}^n} \{\|x\|_2 | Ax = b\} \quad (39)$$

Berechnung von A^+ : Für den Sonderfall, dass A vollen Rang hat, gestaltet sich die Berechnung der Moore-Penrose Pseudoinversen als besonders einfach.

$$m < n \quad \Rightarrow \quad A^+ = A^T (AA^T)^{-1} \quad (40)$$

$$n < m \quad \Rightarrow \quad A^+ = (A^T A)^{-1} A^T \quad (41)$$

Allgemeine Lösung: Die allgemeine Lösung von (35), lässt sich darstellen als:

$$x = A^+b + v, \quad \text{mit} \quad v \in \{x \in \mathbb{R}^n | Ax = 0\} = N(A) \quad (42)$$

Orthogonale Projektion: Seien $B = \text{span}[\{x_1, \dots, x_k\}]$ und $B^\perp = \text{span}[\{x_{k+1}, \dots, x_n\}]$ linear unabhängige Teilräume. X und X^\perp seien die Matrizen, deren Spaltenvektoren durch $[\{x_1, \dots, x_k\}]$ bzw. $[\{x_{k+1}, \dots, x_n\}]$ gegeben sind. So ist die orthogonale Projektion gegeben durch:

$$P_X = [X|X^\perp] \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} [X|X^\perp]^{-1} \quad (43)$$

$$P_{X^\perp} = I - P_X \quad (44)$$

Gilt darüber hinaus, dass die Basisvektoren orthogonal sind $X^T X^\perp = 0$ bzw. $X^\perp{}^T X = 0$, lässt sich die orthogonale Projektion schreiben als:

$$P_X = X(X^T X)^{-1} X^T \quad (45)$$

Die orthogonale Projektion hat die folgenden Eigenschaften: $\|P\| = 1$, $P^2 = P$ und $P^T = P$.

Sei $R(A)$ das Urbild des Bildraumes von A und $N(A)$ der Kern von A. Dann lassen sich die Projektionen auf $R(A)$, $N(A)$, $R(A^\perp)$ und $N(A^\perp)$ mithilfe der Moore-Penrose Inverse ausdrücken.

$$AA^+ = [U_1|U_2] \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} = U_1 U_1^T = P_{R(A)} \quad (46a)$$

$$A^+ A = V_1 V_1^T = P_{R(A^T)} \quad (46b)$$

$$P_{N(A)} = I - P_{R(A^T)} \quad (46c)$$

$$P_{N(A^T)} = I - P_{R(A)} \quad (46d)$$

Gradient Projection Method

Die Idee der Gradient Projection Method ist es, im Lösungsraum des linearen Gleichungssystems eine skalare Gütefunktion $h(\mathbf{q})$ zu optimieren. Wird eine Gelenkgeschwindigkeit die im Kern der Jacobimatrix $N(J)$ liegt zur Pseudolösung addiert, so verändert sich die Posengeschwindigkeit nicht. Zudem ist jede Projektion der negativen Gradientenrichtung $\dot{\mathbf{q}}_0 = -\left(\frac{\partial h(\mathbf{q})}{\partial \mathbf{q}}\bigg|_{\mathbf{q}}\right)^T$ Abstiegsrichtung. Der Wert des Gütefunktional sinkt.

Um zu garantieren, dass sich die Norm der Gelenkgeschwindigkeit durch die Posengeschwindigkeit abschätzen lässt, kann \mathbf{q}_0 beispielsweise mit $\|\frac{\sigma(J)}{\|\dot{\mathbf{x}}\|}\|$ bzw. mit $\|J(\mathbf{q})^+ \dot{\mathbf{x}}\|$ normiert werden. Dann gilt $\|\dot{\mathbf{q}}\| \leq \frac{2}{\sigma(J)} \|\dot{\mathbf{x}}\|$. Mit Gl. 46 erhält man als Vorschrift für die Gradient Projection Method:

$$\dot{\mathbf{q}} = J(\mathbf{q})^+ \dot{\mathbf{x}} + [I - J(\mathbf{q})^+ J(\mathbf{q})] \dot{\mathbf{q}}_0 \quad (47)$$

Vorbereitungsaufgabe – V2.4 –

- (a) Berechnen Sie die Abbildungsvorschrift $\mathbf{x} = \Psi(\phi, \kappa, \ell)$ für eine einzelne Sektion des BHAs. Die Aufgabenkoordinaten \mathbf{x} und die Konfigurationskoordinaten \mathbf{k} setzen sich dabei wie folgt zusammen:

$$\mathbf{x} = {}^{1b}\mathbf{O}_{1h} = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T, \quad \mathbf{k} = \begin{bmatrix} \phi & \kappa & \ell \end{bmatrix}^T \quad (48)$$

Hinweis: Zur Vereinfachung soll in diesem Aufgabenteil die Konfigurationsparameter \mathbf{k} anstatt der sonst üblichen Gelenkzustände \mathbf{q} verwendet werden. Andernfalls würden die analytischen Berechnungen zu komplex werden.

- (b) Berechnen Sie die Abbildungsvorschrift $\mathbf{x} = \Psi(\phi, 0, \ell)$ für $\kappa = 0$. Was bedeutet dies für die Orientierung ϕ für diesen Sonderfall?
- (c) Skizzieren Sie den Biegezustand der Sektion für die Konfigurationskoordinaten \mathbf{k} aus Aufgabenteil (b).
- (d) Berechnen Sie die Jacobimatrix der Abbildungsvorschrift $\Psi(\phi, \kappa, \ell)$ an der Stelle $\kappa = 0$ und $\ell = 1$.
- (e) Berechnen Sie die Posengeschwindigkeit $\dot{\mathbf{x}}$ in Abhängigkeit der Gelenkgeschwindigkeit $\dot{\mathbf{q}}$ für den Biegezustand aus Aufgabenteil (b) gemäß Gleichung (33). Ist eine Bewegung in alle Raumrichtungen möglich?

Hinweise zur Implementierung:

Wahl der Abbildungsvorschrift: Bei der Implementierung der lokalen Rückwärtskinematik können zwei Wege verfolgt werden:

1. Die Kombination aus den Abbildungsvorschriften Ψ_{cs2ts} und Ψ_{cs2ts} kann gemäß Abb. 9 invertiert werden.
2. Die Abbildungsvorschrift Ψ_{cs2ts} wird separat invertiert und anschließend die Berechnung der Gelenkkoordinaten gemäß Gl. 24 durchgeführt.

Da die Systembeschreibung der zweiten Methode jedoch Singularitäten aufweist, welche durch die kombinierte Rückwärtskinematik vermieden werden (siehe Vorbereitungsaufgabe V1.4), wird im Weiteren die erstgenannte Vorgehensweise verfolgt.

Berechnung und Eigenschaften der Jacobimatrix: Die Jacobimatrix hat die Dimension 9×3 , da

$$\mathbf{x} = {}^{1b}\mathbf{O}_{3h} = \begin{bmatrix} x_{TCP} \\ y_{TCP} \\ z_{TCP} \end{bmatrix}$$

und

$$\mathbf{q} = \begin{bmatrix} l_{11} & l_{12} & l_{13} & l_{21} & l_{22} & l_{23} & l_{31} & l_{32} & l_{33} \end{bmatrix}^T.$$

Aufgrund der analytischen Komplexität der Vorwärtskinematik des Manipulators liegt es nahe, sie numerisch mithilfe einer zentralen Differenz zu berechnen. Bei experimentellen Untersuchungen der Singulärwerte der Jacobimatrizen (verteilt über den gesamten Arbeitsraum des Manipulators) lag der minimale Singulärwert bei 0.9760. Die Vermutung liegt somit nahe, dass die Jacobimatrix im gesamten Arbeitsraum nicht singulär wird. Folglich kann der minimale Singulärwert durch 0.9 abgeschätzt werden.

Ableitungsbasierte Rückwärtskinematik: Die Pseudoinverse der Jacobimatrix darf mit der Formel

$$J^+ = J^T (J J^T)^{-1}$$

gemäß Gleichung (40) bestimmt werden, da $J(\mathbf{q})$ stets vollen Rang hat und n größer m ist. Die Geschwindigkeit der Seillängenänderung kann im Arbeitsraum mit

$$\|\dot{\mathbf{q}}\| \leq \frac{10}{9} \|\dot{\mathbf{x}}\|$$

abgeschätzt werden.

Integrationsfehler: Die ableitungsbasierte Rückwärtskinematik berechnet die Gelenkgeschwindigkeiten in Abhängigkeit einer gewünschten Posengeschwindigkeit. Dazu wird die Pseudoinverse der Jacobimatrix verwendet, welche eine punktweise lineare Approximation der nichtlinearen Abbildungsvorschrift ist. Zur Berechnung des Gelenkzustands des Manipulators ist folglich eine Integration der Gelenkgeschwindigkeiten notwendig. Unstabilisiert ist diese Integration grenzstabil, womit geringe Fehler bei der Berechnung der Gelenkgeschwindigkeiten zu einem Drift zwischen Soll- und Istwert der Pose führen.

Einhaltung von Beschränkungen des Gelenkzustands: Die bestehende Rückwärtskinematik ist hinsichtlich einer Beschränkung des Gelenkzustands zu erweitern, welche sich im Fall des BHAs aus den neun

Balglängen (bzw. Seilzuglängen der Balgaktoren) zusammensetzt. Falls eine der Beschränkungen aktiv (erreicht oder überschritten) ist, muss die Rückwärtskinematik eine Gelenkgeschwindigkeit vorgeben, die in entgegengesetzte Richtung weist. Die nachfolgende Forderung muss somit stets erfüllt sein:

$$\begin{aligned} q_{\max,i} - q_i &\leq 0 \Rightarrow \dot{q}_i \leq 0 \\ q_i - q_{\min,i} &\leq 0 \Rightarrow \dot{q}_i \geq 0 \end{aligned} \quad (49)$$

Zur Umsetzung sind unter anderem die folgenden Varianten denkbar:

- Variante (a) Falls Bedingung (49) nicht erfüllt ist, setze $\dot{q} = 0$. Generell ist es jedoch nicht sinnvoll, nur $\dot{q}_i = 0$ zu setzen, da der Roboterarm eine Bewegung in undefinierte Richtung ausführen würde. Das Problem dieser Variante ist somit, dass die Bewegung abgebrochen wird, falls die Pseudolösung in das beschränkte Gebiet zeigt – auch wenn im Lösungsraum eine zulässige Lösung existiert. Dieses Problem kann behoben werden, indem mithilfe der Gradient Projection Method ein gleichmäßiges Erreichen der Beschränkungen garantiert wird.
- Variante (b) Setze $\dot{q}_i = 0$, entferne die i -te Spalte aus der Jacobimatrix und berechne die Pseudolösung erneut. Setze diese Prozedur solange fort, bis eine zulässige Lösung gefunden wird. Das Problem dieser Variante ist, dass nach dem Entfernen von x Spalten keine Abschätzung für den minimalen Singulärwert der reduzierten Jacobimatrix vorliegt und somit keine Garantie für die Nichtsingularität derselben gegeben ist. Die Pseudoinverse muss demnach mithilfe der Singulärwertzerlegung bestimmt werden. Außerdem muss die Seillängengeschwindigkeit nachträglich normiert werden.

Gradient Projection Method: Die Gütefunktion soll eine Annäherung des Gelenkzustands \mathbf{q} an dessen Beschränkungen bestrafen. Es bietet sich somit an, ein normiertes symmetrisches Polynom als innere Beschränkung zu verwenden. Je höher dessen Grad, desto steiler ist der Anstieg bei Annäherung an die Beschränkungen (siehe Abb. 11).

Die Gütefunktion und deren Ableitungen können als

$$h(\mathbf{q}) = \frac{1}{n} \sum_{i=1}^n \left(\frac{q_i - q_{m,i}}{q_{\Delta,i}} \right)^{2k} \quad (50)$$

$$\frac{\partial h(\mathbf{q})}{\partial q_i} = \frac{2k}{n q_{\Delta,i}} \left(\frac{q_i - q_{m,i}}{q_{\Delta,i}} \right)^{2k-1} \quad (51)$$

gewählt werden. Dabei bezeichnet $q_{\Delta,i}$ den maximalen Abstand der Beschränkung des Gelenkzustands i und $q_{m,i}$ dessen Mittelwert

$$q_{\Delta,i} = \frac{q_{\max,i} - q_{\min,i}}{2}, \quad q_{m,i} = \frac{q_{\max,i} + q_{\min,i}}{2}. \quad (52)$$

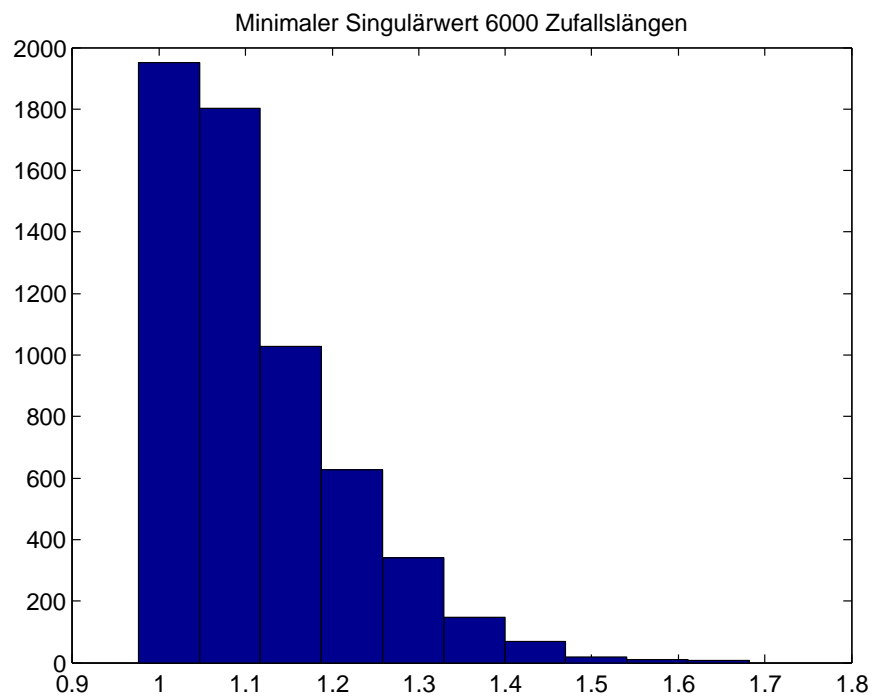


Abbildung 10: Histogramm minimaler Singulärwert

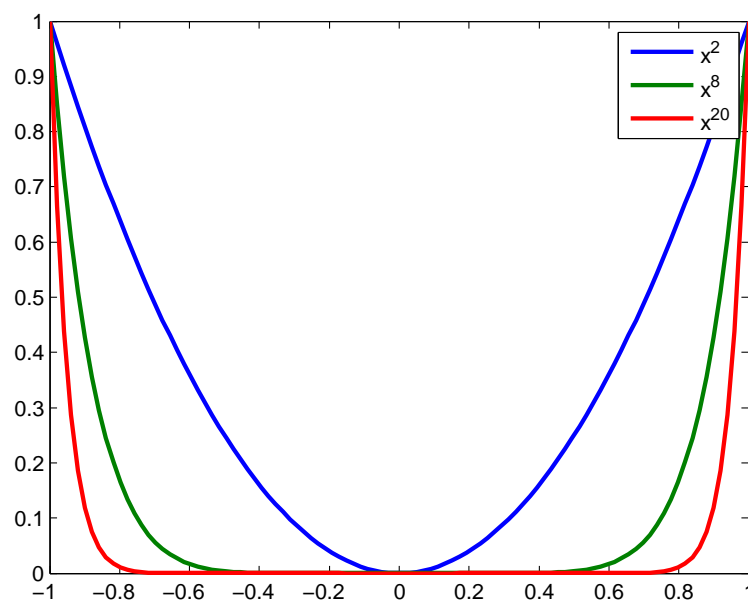


Abbildung 11: Symmetrische Polynome für die Implementierung der Gradient Projection Method.

5 Versuchsdurchführung

Das folgende Kapitel führt Sie durch den experimentellen Teil des Praktikums.

5.1 Starten der Entwicklungsumgebung

Die Umsetzung der Methoden zur Steuerung des Handhabungssystems erfolgt mit der Programmiersprache Matlab/Simulink. Da für jede Aufgabenstellung bereits ein vorgefertigtes Template existiert, ist es wichtig, dass Sie die Aufgaben im richtigen Arbeitsverzeichnis lösen. Dazu

- (i) Starten Sie Matlab
- (ii) Beantworten Sie ggf. die Frage nach dem dSPACE Real-Time Interface mit „RTI 1007“ und
- (iii) Navigieren Sie zum gewünschten Arbeitsverzeichnis.

Das Arbeitsverzeichnis setzt sich wie folgt zusammen:

D:\BHA Praktikum\Versuchsdurchfuehrung\P{Aufgabennummer}

5.2 Implementierung und Test der Vorwärtskinematik

Im Folgenden soll die in Abschnitt 4 beschriebene Vorwärtskinematik des BHAs implementiert werden.

Praktikumsaufgabe – P2.1 –

(a) Implementierung der homogenen Transformation einer Sektion

1. Vergewissern Sie sich, dass Sie sich im korrekten Verzeichnis befinden (D:\BHA Praktikum\Versuchsdurchfuehrung\P1_a).
2. Öffnen Sie das Skript „PAR_Kin_v_Sek_cc.m“ und führen Sie dieses aus. Das Skript lädt die für die Aufgabenstellung notwendigen Parameter ein. Zudem wird der Pfad der Bibliothek „BHA_Lib_Prakt“ in den Matlab Workpath eingetragen, womit diese benutzt werden kann.
3. Öffnen Sie des Weiteren das Simulinkmodell „Kin_v_Sek_cc.slx“ und vervollständigen Sie dieses durch Implementierung der Abbildungsvorschrift $\Psi_{as2ts}(\ell_{ik})$ gemäß Abbildung 9. Verwenden Sie dazu die im Template vorgegebene Struktur.
4. Testen Sie Ihre Vorwärtskinematik der Sektion auf Fehlerfreiheit und Plausibilität. Führen Sie dazu einen Simulationslauf durch und überprüfen Sie, ob das resultierende Simulationsergebnis Ihren Erwartungen entspricht. Passen Sie dazu gegebenenfalls die Seillängenvorgaben Ihren Wünschen an (Parameterstruktur „IData“).
5. Vergleichen Sie Ihr Simulationsergebnis mit der Musterlösung. Gehen Sie dazu im Parameterfile zum Abschnitt „Auswertung“ und aktivieren Sie diese durch setzen der bedingten Anweisung (if) auf *true* bzw. 1. Führen Sie danach das Parameterfile erneut aus. Beachten Sie auch die Abfragen im Matlab Command Window.

6. Wenn Ihre Simulationsergebnisse mit der Musterlösung für alle 3 Sektionen übereinstimmen, Speichern Sie das Subsystem, welches die Abbildungsvorschrift $\Psi_{as2ts}(L_{ik})$ enthält, in einer von Ihnen erstellten Bibliothek ab. Gehen Sie dazu wie folgt vor:

- Legen Sie eine neue Bibliothek über *File*→*New*→*Library* an.
- Kopieren Sie das Subsystem mit der Abbildungsvorschrift $\Psi_{as2ts}(L_{ik})$ in diese Bibliothek.
- Speichern Sie die Bibliothek mit beliebigem Namen unter D:\BHA Praktikum\Versuchsdurchfuehrung\BHA_Lib_Stud ab.
- Testen Sie erneut die Funktionsfähigkeit Ihrer Simulation unter Verwendung des Subsystems der Bibliothek.

(b) Implementierung der homogenen Transformation des Manipulatorarms

1. Wechseln Sie in das für die Aufgabenstellung vorgesehene Verzeichnis.
2. Öffnen Sie das Parameterfile „PAR_Kin_v_BHA.cc.m“ und führen Sie es aus.
3. Öffnen Sie das Simulinkmodell „Kin_v_BHA.cc.slx“ und vervollständigen Sie das Subsystem „BHA_Kin_v“ unter Verwendung der selbst erstellten Bibliothek aus Aufgabenteil (a). Achten Sie auch stets darauf die Namen der Eingangssignale des *Bus Creators* nicht zu verändern. Die Transformation zwischen dem Fußkoordinatensystem von Sektion 1 und dem Koordinatensystem \mathcal{R}_{xx} wird durch den Eingang „H_1b_xx“ beschrieben. Ziel des Subsystems ist die Berechnung von 9 homogenen Transformationen. Diese sind:

- „S_1h_1b“ $\triangleq {}^{1b}\mathbf{S}_{1h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 1 und Kopfkoordinatensystem von Sektion 1
- „M_1h_1b“ $\triangleq {}^{1b}\mathbf{M}_{1h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 1 und Kopfkoordinatensystem von Sektion 1
- „M_1h_xx“ $\triangleq {}^{xx}\mathbf{M}_{1h}$: Homogene Transformation zwischen Koordinatensystem xx und Kopfkoordinatensystem von Sektion 1
- „S_2h_2b“ $\triangleq {}^{2b}\mathbf{S}_{2h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 2 und Kopfkoordinatensystem von Sektion 2
- „M_2h_1b“ $\triangleq {}^{1b}\mathbf{M}_{2h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 1 und Kopfkoordinatensystem von Sektion 2
- „M_2h_xx“ $\triangleq {}^{xx}\mathbf{M}_{2h}$: Homogene Transformation zwischen Koordinatensystem xx und Kopfkoordinatensystem von Sektion 2
- „S_3h_3b“ $\triangleq {}^{3b}\mathbf{S}_{3h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 3 und Kopfkoordinatensystem von Sektion 3
- „M_3h_1b“ $\triangleq {}^{1b}\mathbf{M}_{3h}$: Homogene Transformation zwischen Fußkoordinatensystem von Sektion 1 und Kopfkoordinatensystem von Sektion 3
- „M_3h_xx“ $\triangleq {}^{xx}\mathbf{M}_{1h}$: Homogene Transformation zwischen Koordinatensystem \mathcal{R}_{xx} und Kopfkoordinatensystem von Sektion 3

4. Testen Sie Ihre Vorwärtskinematik der Sektion auf Fehlerfreiheit und Plausibilität.
5. Vergleichen Sie Ihr Simulationsergebnis mit der Musterlösung. Gehen Sie dazu im Parameterfile zum Abschnitt „Auswertung“ und aktivieren Sie diese durch setzen der bedingten Anweisung (if) auf *true* bzw. 1. Führen Sie danach das Parameterfile erneut aus.
6. Wenn Ihr Simulationsergebnis mit der Musterlösung übereinstimmt, speichern Sie die Vorwärtskinematik des BHAs in der von Ihnen erstellten Bibliothek ab. Überprüfen Sie danach die Funktionalität der Simulation unter Verwendung des Blocks aus der Bibliothek.

5.3 Implementierung und Test der Rückwärtskinematik

Im praktischen Teil soll die ableitungsbasierte Rückwärtskinematik entsprechend Abschnitt 4.3.2 implementiert werden. Dabei wird in Aufgabenteil (a) die numerische Berechnung der Jacobimatrix durchgeführt. Aufgabenteil (b) behandelt daraufhin die Invertierung der Jacobimatrix als auch die Implementierung der Gradient Projection Method.

Praktikumsaufgabe – P2.2 –

(a) Numerische Berechnung der Jacobimatrix

1. Wechseln Sie in das für die Aufgabenstellung vorgesehene Verzeichnis.
2. Öffnen Sie das Parameterfile „PAR_ML_Jacobi.m“ und führen Sie es aus.
3. Öffnen Sie das Simulinkmodell „Jq_cc.slx“ und vervollständigen Sie es. Verwenden Sie dabei die in Aufgabe P2.1 angelegte Bibliothek. Beachten Sie zudem, dass das Subsystem „Zentraler Differenzenquotient“ ein *For Each Subsystem* ist. Falls Ihnen dieser Simulink-Block neu ist, lesen Sie sich gegebenenfalls die Matlab-Hilfe dazu durch. Der zentrale Differenzenquotient ist wie folgt definiert:

$$\left. \frac{\Delta \Psi}{\Delta \mathbf{q}} \right|_{\mathbf{q}_0} = \frac{\Psi(\mathbf{q}_0 + \Delta \mathbf{q}) - \Psi(\mathbf{q}_0 - \Delta \mathbf{q})}{2 \|\Delta \mathbf{q}\|}. \quad (53)$$

4. Testen Sie Ihre Berechnung der Jacobimatrix auf Fehlerfreiheit und Plausibilität.
5. Vergleichen Sie Ihr Simulationsergebnis mit der Musterlösung. Gehen Sie dazu im Parameterfile zum Abschnitt „Input: Seillängen“ und aktivieren Sie diese durch setzen der bedingten Anweisung (if) auf *true* bzw. 1. Führen Sie danach das Parameterfile erneut aus. Beachten Sie auch die Fragen im Matlab Command Window.
6. Wenn Ihr Simulationsergebnis mit der Musterlösung übereinstimmt, speichern Sie die Berechnung der Jacobimatrix in der von Ihnen erstellten Bibliothek ab. Überprüfen Sie danach die Funktionalität der Simulation unter Verwendung des Blocks aus der Bibliothek.

(b) Pseudoinverse der Jacobimatrix und Gradient Projection Method

1. Wechseln Sie in das für die Aufgabenstellung vorgesehene Verzeichnis.
2. Öffnen Sie das Parameterfile „PAR_Kin_r_cc.m“ und führen Sie es aus.
3. Öffnen Sie das Simulinkmodell „Kin_r_cc.slx“ und vervollständigen Sie das Subsystem „Kin_r_cc“. Fügen Sie zudem die Vorwärtskinematik des BHAs an der Stelle mit der Beschriftung „BHA_Kin_v“ ein.
4. Testen Sie Ihre Simulation auf Fehlerfreiheit und Plausibilität. Achten Sie vor Allem darauf, ob Ihre Beschränkung des Aktuatorzustands eingehalten wird. Experimentieren Sie auch mit der Polynomordnung k für die Beschränkung des Aktuatorzustands.
5. Speichern Sie die Berechnung der ableitungsbasierten Rückwärtskinematik in der von Ihnen erstellten Bibliothek ab. Überprüfen Sie danach die Funktionalität der Simulation unter Verwendung des Blocks aus der Bibliothek.

Literatur

- [WJ10] WEBSTER, Robert J. ; JONES, Bryan A.: Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review. In: *The International Journal of Robotics Research* 29 (2010), Nr. 13, S. 1661–1683. <http://dx.doi.org/10.1177/0278364910368147>. – DOI 10.1177/0278364910368147