

Aufgabenblätter zur Klausur

Robotik I: Einführung in die Robotik

am 10. April 2017, 14:00 – 15:00 Uhr

- Beschriften Sie bitte gleich zu Beginn jedes Lösungsblatt deutlich lesbar mit Ihrem Namen und Ihrer Matrikelnummer.
- Diese Aufgabenblätter werden nicht abgegeben. Tragen Sie Ihre Lösung deshalb ausschließlich in die für jede Aufgabe vorgesehenen Bereiche der Lösungsblätter ein. Lösungen auf separat abgegebenen Blättern werden nicht gewertet.
- Außer Schreibmaterial sind während der Klausur keine Hilfsmittel zugelassen. Täuschungsversuche durch Verwendung unzulässiger Hilfsmittel führen unmittelbar zum Ausschluss von der Klausur und zur Note „nicht bestanden“.
- Soweit in der Aufgabenstellung nichts anderes angegeben ist, tragen Sie in die Lösungsblätter bitte nur die Endergebnisse ein. Die Rückseiten der Aufgabenblätter können Sie als Konzeptpapier verwenden. Weiteres Konzeptpapier können Sie auf Anfrage während der Klausur erhalten.
- Halten Sie Begründungen oder Erklärungen bitte so kurz wie möglich. (Der auf den Lösungsblättern für eine Aufgabe vorgesehene Platz steht übrigens in keinem Zusammenhang mit dem Umfang einer korrekten Lösung!)
- Die Gesamtpunktzahl beträgt 45 Punkte.

Viel Erfolg und viel Glück!

Aufgabe 1 *Rotationen*

(6 Punkte)

Geben Sie in dieser Aufgabe den **Rechenweg** an und runden Sie sämtlich Ergebnisse auf eine Dezimalstelle.

1. Die Rotationsmatrix R und der Translationsvektor t definieren die Lage des lokalen Koordinatensystems OKS im Weltkoordinatensystem WKS . Geben Sie die homogene Transformationsmatrix ${}^{WKS}T_{OKS}$ an.

1 P.

$$t = (100, 200, 300)^T \quad R = \begin{pmatrix} 0 & -0.6 & 0.8 \\ 1 & 0 & 0 \\ 0 & 0.8 & 0.6 \end{pmatrix}$$

2. Der Punkt $p = (-200, 100, 100)^T$ sei im Weltkoordinatensystem WKS definiert. Transformieren Sie p in das lokale Koordinatensystem OKS aus Aufgabenteil 1.1.

3 P.

3. Geben Sie ein Quaternion q an, das eine Rotation um $\theta = \frac{\pi}{2}$ mit der Rotationsachse $u = (0, 0, 1)^T$ beschreibt.

1 P.

Hinweis: $\sin(\frac{\pi}{4}) = \cos(\frac{\pi}{4}) \approx 0.7$

4. Geben Sie das zu q aus Aufgabenteil 1.3 konjugierte Quaternion q^* an.

1 P.

Aufgabe 2 *Kinematik und Dynamik*

(11 Punkte)

In Abbildung 1 ist ein Roboterarm in seiner Ruhestellung mit den Achsen x_i und z_i der rechtshändigen Gelenkkoordinatensysteme dargestellt. Die Längenangaben sind in Millimetern.

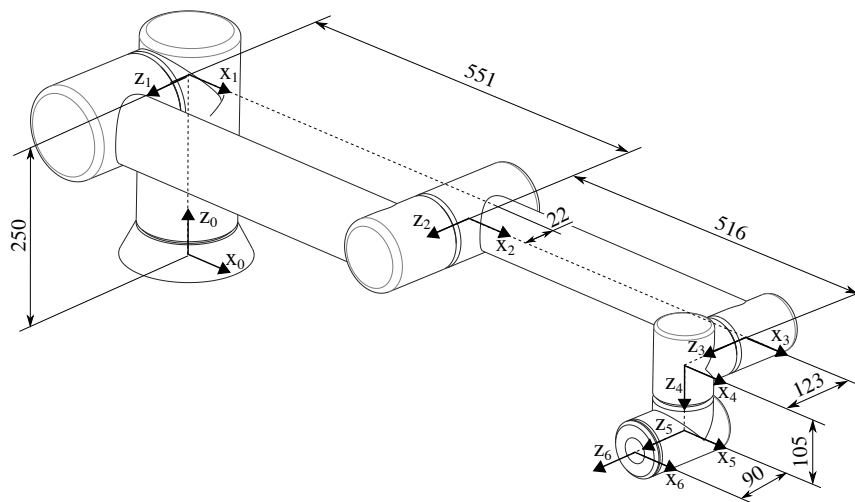


Abbildung 1: Roboterarm in der Ruhestellung

1. Vervollständigen Sie die Tabelle im Lösungsblatt, indem Sie alle fehlenden Denavit-Hartenberg-Parameter eintragen. 4 P.
2. Wie viele Rotationsgelenke und wie viele Translationsgelenke besitzt der Arm? 1 P.
3. Welcher geometrischen Form entspricht der Arbeitsraum des Roboterarms näherungsweise, wenn alle Gelenke die Gelenkwinkelgrenzen $\pm 180^\circ$ besitzen? 1 P.
4. An den Roboterarm soll ein Endeffektor angebracht werden. Bestimmen Sie die Lage des Endeffektor-Koordinatensystems (x_7, y_7, z_7) . Transformieren Sie hierzu das auf dem Lösungsblatt gegebene Koordinatensystem (x_6, y_6, z_6) zeichnerisch in (x_7, y_7, z_7) . Verwenden Sie dabei die folgenden DH-Parameter 3 P.

$$\theta_7 = 90^\circ \quad d_7 = 40 \text{ mm} \quad a_7 = 25 \text{ mm} \quad \alpha = 0^\circ$$

und stellen Sie diese DH-Parameter in Ihrer Zeichnung durch beschriftete Abstände und Winkel dar.

5. Welche zwei Zusammenhänge beschreibt die Jacobi-Matrix im Hinblick auf die Kinematik und Dynamik eines Roboters? 2 P.
Welche Dimension hat die Jacobi-Matrix für einen Roboterarm mit 11 Bewegungsfreiheitsgraden in einem 6D Arbeitsraum?

Aufgabe 3 *Motion Planning*

(7 Punkte)

1. Zellzerlegung mit Line-Sweep
 - (a) Bestimmen Sie die Zellzerlegung des auf dem Lösungsblatt angegebenen Grundrisses mittels des Line-Sweep-Verfahrens. Nummerieren Sie die dabei entstehenden Zellen von links nach rechts. Bei gleichem linkem Rand nummerieren Sie die Zellen von oben nach unten. 2 P.
 - (b) Erstellen Sie den Adjazenzgraphen der ermittelten Zellen. 2 P.
 - (c) Bestimmen Sie den kürzesten Pfad von der Anfangskonfiguration q_{start} zur Endkonfiguration q_{ziel} . Geben Sie dabei die Folge der durchquerten Zellen an. 1 P.
Hinweis: q_{start} und q_{ziel} sind im Grundriss dargestellt.
2. Nennen Sie zwei Unterschiede oder Eigenschaften der Bewegungsplanungsalgorithmen RRT und RRT*. 2 P.

Aufgabe 4 *Bildverarbeitung*

(9 Punkte)

1. Geben Sie die (3×3) -Filtermasken für das Prewitt-X und Prewitt-Y Filter an.

1 P.

2. Gegeben sei das Graustufenbild B :

3 P.

$$B = \begin{pmatrix} 2 & 2 & 2 & 4 & 4 & 4 \\ 2 & 2 & 2 & 4 & 4 & 4 \\ 2 & 2 & 2 & 4 & 4 & 4 \\ 2 & 2 & 2 & 4 & 4 & 4 \end{pmatrix}$$

Wie kann die Filtermatrix M zur Bestimmung des Gradientenbetrags durch Kombination der Prewitt-X und Prewitt-Y Filter approximiert werden?

Wenden Sie anschließend M auf das Graustufenbild B an. Ignorieren Sie im Ergebnisbild die Randpixel, d.h. das Ergebnis ist eine (2×4) -Matrix.

3. Beantworten Sie folgende Fragen zum *Visual Servoing*

3 P.

- (a) Was ist der Unterschied zwischen positions- und bildbasiertem Visual Servoing?
- (b) Was beschreibt die *Interaction Matrix*?

4. Was versteht man unter der Abkürzung SLAM? Wozu wird es benötigt? Wenn SLAM die Abkürzung für ein Problem ist, wie sieht die prinzipielle Vorgehensweise zur Lösung dieses Problems aus?

2 P.

Aufgabe 5 *Programmieren durch Vormachen* (7 Punkte)

1. Nennen und erläutern Sie die vier Hauptfragestellungen (die vier Ws) beim Programmieren durch Vormachen (PdV).

4 P.

2. Was ist der Unterschied zwischen Batch-Lernen und inkrementellem Lernen bei PdV?

1 P.

3. Nennen Sie vier verschiedene Techniken zum Aufnehmen von Demonstrationen, bei denen der Mensch die Aktionen selbst ausführt (Sensoren auf dem Lehrer).

2 P.

Aufgabe 6 *Symbolisches Planen mit STRIPS* (5 Punkte)

Gegeben ist die folgende Planungsdomäne mit initialem Zustand, Zielzustand und verfügbaren Planungsoperatoren.

Initial state: Apple(A), Banana(B), Pineapple(P),
 Table(T), Container(C), Hand(H),
 inHand(B,H), On(A,T), In(P,C), On(C,T)

Goal state: In(A,C)

Actions:

```
// Nimmt Objekt O mit Hand H von Ort L
pickup(O,H,L)
Preconditions:  Empty(H), On(O,L), Hand(H)
Postconditions: !On(O,L), !Empty(H), inHand(O,H)

// Platziert Objekt O mit Hand H auf Ort L
putdown(O,H,L)
Preconditions:  inHand(O,H), Table(L), Hand(H)
Postconditions: On(O,L), Empty(H), !inHand(O,H)

// Platziert Objekt O mit Hand H in Container C
putin(O,H,C)
Preconditions:  inHand(O,H), Container(C), Empty(C), Hand(H)
Postconditions: In(O,C), Empty(H), !inHand(O,H), !Empty(C)

// Nimmt Objekt O mit Hand H aus Container C
takefrom(O,H,C)
Preconditions:  In(O,C), Empty(H), Container(C), Hand(H)
Postconditions: inHand(O,H), Empty(C), !In(O,C), !Empty(H)
```

1. Geben Sie die kürzeste Aktionssequenz an, die den initialen Weltzustand in den Zielzustand überführt. 4 P.
2. Was versteht man unter der Annahme der Weltabgeschlossenheit (*Closed World Assumption*)? 1 P.