KIT-Fakultät für Informatik
Prof. Dr.-Ing. Tamim Asfour

Musterlösungen zur Klausur

Robotik I: Einführung in die Robotik

am 29. Juli 2019, 14:00 – 15:00 Uhr

Name:	Vorname:	Matrikelnummer:
Denavit	Hartenberg	$\frac{\pi}{2}$

Aufgabe 1	von 4 Punkten
Aufgabe 2	von 7 Punkten
Aufgabe 3	von 5 Punkten
Aufgabe 4	von 6 Punkten
Aufgabe 5	von 6 Punkten
Aufgabe 6	von 6 Punkten
Aufgabe 7	von 11 Punkten

Gesamtpunktzahl:	45 von 45 Punkten

	Note:	**1,0**

Aufgabe 1 *Rotationen*

1. Handelt es sich bei R um eine Rotationsmatrix? (Beweis) 3 P.

   Ansatz: Eine Matrix $R \in \mathbb{R}^{3\times3}$ ist eine Rotationsmatrix, wenn $R^TR = I$ (Orthogonalität) und $\det R = 1$ gilt.

   Beweis (Möglichkeit 1): Prüfe Orthogonalität

$$R^TR = \begin{pmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{2} & -\frac{1}{\sqrt{2}} & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{4}+\frac{1}{2}+\frac{1}{4} & \frac{1}{2\sqrt{2}}+\frac{1}{2\sqrt{2}} & \frac{1}{4}-\frac{1}{2}+\frac{1}{4} \\ \frac{1}{2\sqrt{2}}+\frac{1}{2\sqrt{2}} & \frac{1}{2}+\frac{1}{2} & \frac{1}{2\sqrt{2}}+\frac{1}{2\sqrt{2}} \\ \frac{1}{4}-\frac{1}{2}+\frac{1}{4} & \frac{1}{2\sqrt{2}}+\frac{1}{2\sqrt{2}} & \frac{1}{4}+\frac{1}{2}+\frac{1}{4} \end{pmatrix}$$

$$R^TR = \begin{pmatrix} 1 & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 1 \end{pmatrix} \neq I$$

   Alternativ:

$$RR^T = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \neq I$$

   Antwort: Die Matrix R ist nicht orthogonal und deshalb keine Rotationsmatrix.

   Beweis (Möglichkeit 2): Prüfe Determinante

$$\begin{aligned} \det R &= \det \begin{pmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \end{pmatrix} \\ &= \frac{1}{2}\cdot 0 \cdot \frac{1}{2} + \frac{1}{\sqrt{2}} \cdot (-\frac{1}{\sqrt{2}}) \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} - \frac{1}{2} \cdot 0 \cdot \frac{1}{2} - \frac{1}{\sqrt{2}} \cdot (-\frac{1}{\sqrt{2}}) \cdot \frac{1}{2} - \frac{1}{2} \cdot \frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} \\ &= 0 + (-\frac{1}{4}) + \frac{1}{4} - 0 - (-\frac{1}{4}) - \frac{1}{4} = 0 \neq 1 \end{aligned}$$

   Antwort: R ist keine Rotationsmatrix, da ihre Determinante ungleich 1 ist.

2. Translationsvektor und Rotationsmatrix: 1 P.

$$t = \begin{pmatrix} 0.5 \\ 0 \\ -0.5 \end{pmatrix} \quad R = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Aufgabe 2 *Kinematik*

1. Dimension: 6×2 1 P.

2. Jacobi-Matrix: 3 P.

   $\frac{\partial x_1}{\partial \theta_1} = -L \cdot \sin(\theta_1) - L \cdot \sin(\theta_1 + \theta_2)$

   $\frac{\partial x_2}{\partial \theta_1} = L \cdot \cos(\theta_1) + L \cdot \cos(\theta_1 + \theta_2)$

   $\frac{\partial x_3}{\partial \theta_1} = \frac{\partial \alpha}{\partial \theta_1} = \frac{\partial \beta}{\partial \theta_1} = 0$

   $\frac{\partial \gamma}{\partial \theta_1} = 1$

   $\frac{\partial x_1}{\partial \theta_2} = -L \cdot \sin(\theta_1 + \theta_2)$

   $\frac{\partial x_2}{\partial \theta_2} = L \cdot \cos(\theta_1 + \theta_2)$

   $\frac{\partial x_3}{\partial \theta_2} = \frac{\partial \alpha}{\partial \theta_2} = \frac{\partial \beta}{\partial \theta_2} = 0$

   $\frac{\partial \gamma}{\partial \theta_2} = 1$

$$J = \left(\frac{\partial f}{\partial \theta_1}, \frac{\partial f}{\partial \theta_2}\right) = \begin{pmatrix} -L \cdot sin(\theta_1) - L \cdot sin(\theta_1 + \theta_2) & -L \cdot sin(\theta_1 + \theta_2) \\ L \cdot cos(\theta_1) + L \cdot cos(\theta_1 + \theta_2) & L \cdot cos(\theta_1 + \theta_2) \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix}$$

3. Endeffektor-Geschwindigkeit: 3 P.

$$\begin{aligned} v &= J\begin{pmatrix} 90^\circ \\ 0^\circ \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -L \cdot sin(90^\circ) - L \cdot sin(90^\circ + 0^\circ) & -L \cdot sin(90^\circ + 0^\circ) \\ L \cdot cos(90^\circ) + L \cdot cos(90^\circ + 0^\circ) & L \cdot cos(90^\circ + 0^\circ) \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} -2L & -L \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -L \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{aligned}$$

Aufgabe 3 *Dynamik*

1. Vier wichtige Komponenten aus der Bewegungsgleichung: 2 P.

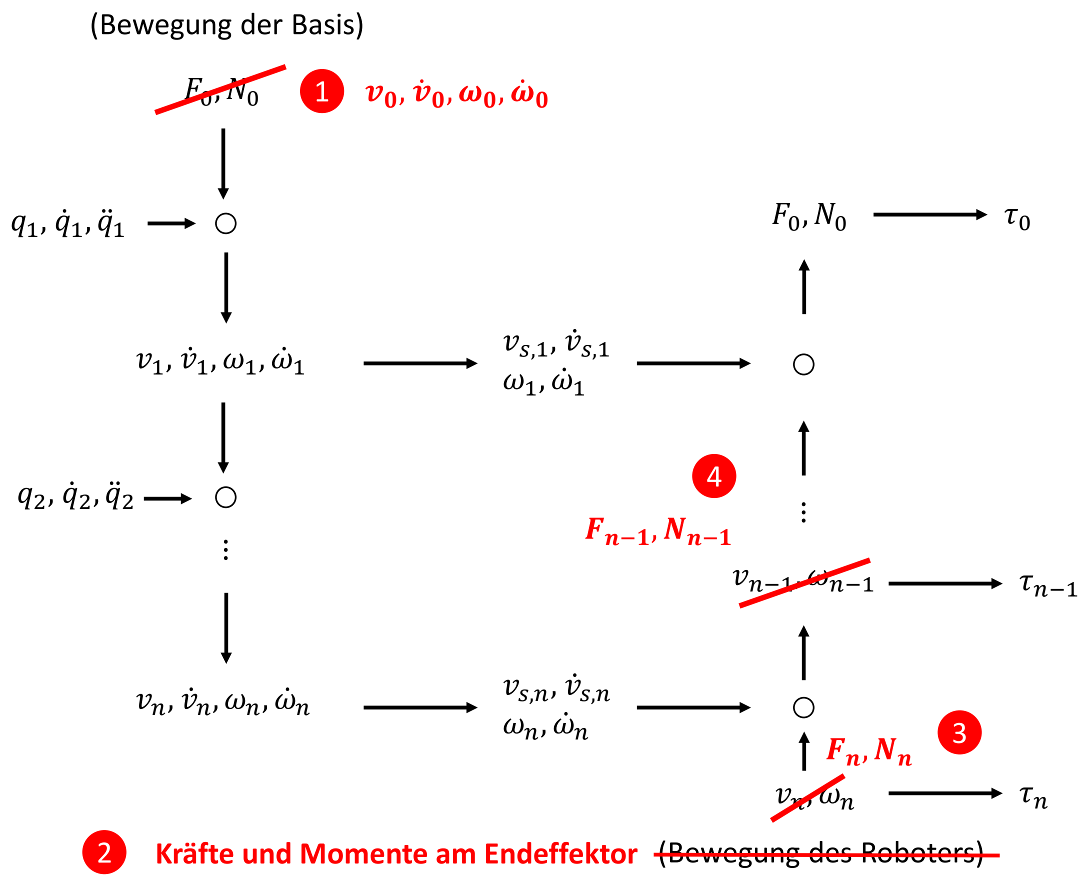
   τ: Generalisierte Kräfte

   $M(q)$: Massenträgheitsmatrix

   $C(q,\dot{q})$: Zentripetal und Corioliskomponent

   $g(q)$: Gravitationskomponente

2. (a) Markieren Sie alle Fehler und korrigieren Sie diese. (Anmerkung: Die Korrektur muss eindeutig einem Fehler zugewiesen werden können.) 2 P.



   (b) Zwei Vorteile der Newton-Euler Methode gegenüber der Lagrange-Methode: 1 P.

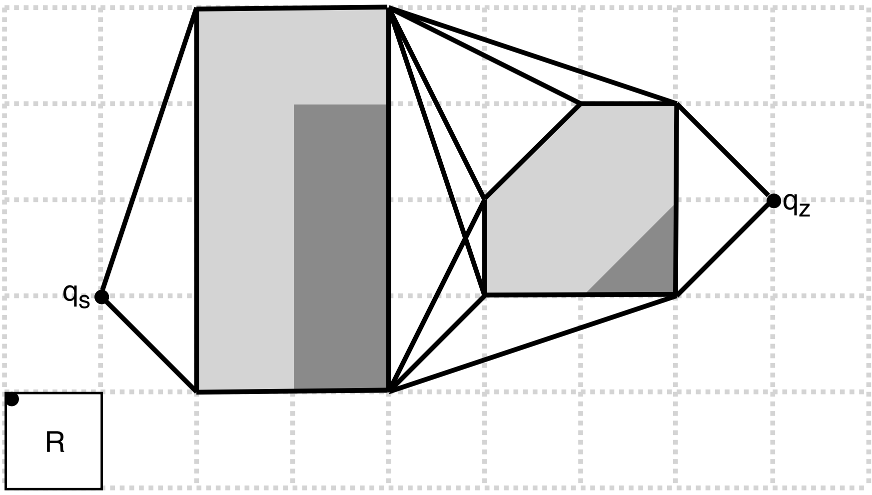
   i. Geringerer Rechenaufwand von RNE mit $O(n)$ gegenüber Lagrange mit $O(n^3)$

   ii. Einfach zu implementieren, aufgrund der Rekursion

   iii. Beliebige Anzahl an Gelenken kann berechnet werden, da sehr effizient

Aufgabe 4 *Bewegungsplanung*

1. Sichtgraph: 3 P.



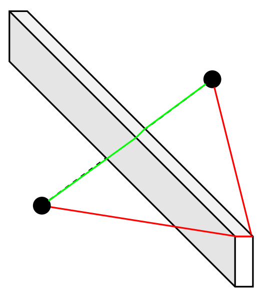
Hellgrau: Erweiterung der Hindernisse unter Beachtung der Form des Roboters.

2. Optimalität: 1 P.

   - Der Roboter und alle Hindernisse sind durch konvexe Polygone darstellbar.
   - Der Arbeitsraum besteht aus zwei translatorischen Freiheitsgraden.

3. Optimalität in $\mathbb{R}^3$: 1 P.

   Nein, in $\mathbb{R}^3$ führt der Sichtgraph über Ecken und der optimale Weg kreuzt ggf. nur Kanten (siehe Visualisierung)



4. Unterschied PRM und DRM: 1 P.

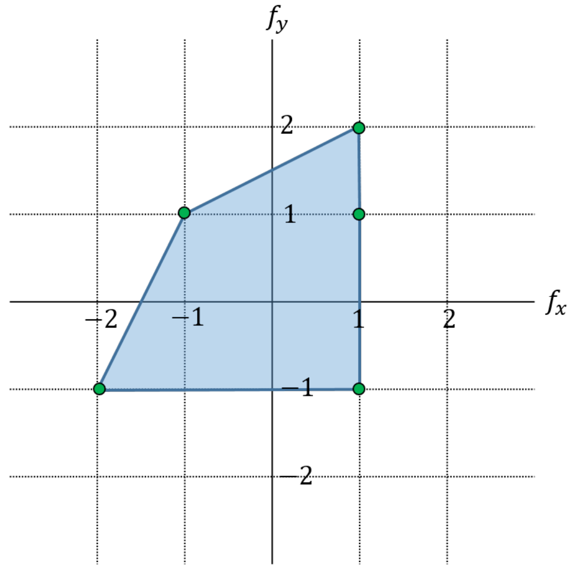
   Beispiele:

   - Die Umgebung darf sich bei DRM ändern
   - Eine DRM hat eine Abbildung vom Arbeits- zum Konfigurationsraum
   - Ein DRM voxilisiert den Arbeitsraum

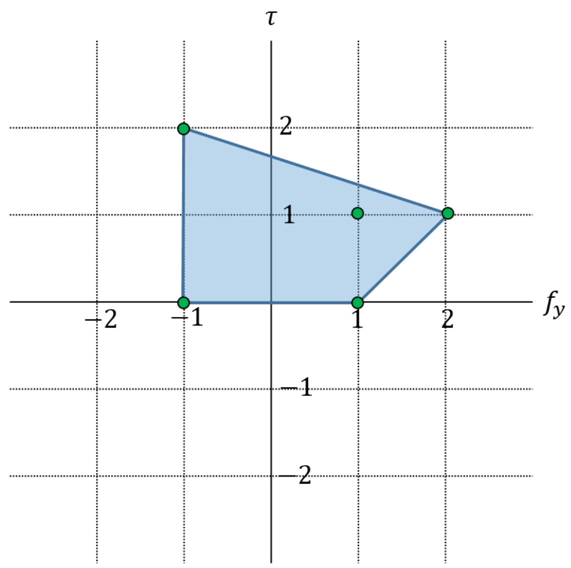
Aufgabe 5 *Greifplanung*

1. Projektion des GWS: **4 P.**

   (a) Projektion auf die (f_x, f_y)-Ebene:



   (b) Projektion auf die (f_y, τ)-Ebene:



2. Kraftgeschlossenheit: **2 P.**

   Der Griff ist nicht kraftgeschlossen

   Zwei alternative Begründungen möglich:

   - ε-Metrik ist 0, da minimaler Abstand zum Ursprung 0 ist (siehe Projektion auf(f_y, τ)-Ebene).
   - Die Wrenches spannen nicht den gesamten $\mathbb{R}^3$ auf. Ein Drehmoment $\tau < 0$ kann nicht als positive Linearkombination erzeugt werden ($pos(w) \neq \mathbb{R}^3$).

Aufgabe 6 *Bildverarbeitung*

1. RGB → HSI: 1.5 P.

$$\theta = \arccos\left(\frac{2R-G-B}{2\sqrt{(R-G)^2+(R-B)(G-B)}}\right) = \arccos\left(\frac{60}{2*30}\right) = 0$$

$$H = \begin{cases} \theta, & \text{falls B < G} \\ 360-\theta, & \text{sonst} \end{cases} = 360-\theta = 360$$

$$S = 1 - \frac{3}{R+G+B}\min(R,G,B) = 1 - \frac{3}{120} * 30 = \frac{1}{4}$$

$$I = \frac{1}{3}(R+G+B) = \frac{120}{3} = 40$$

2. Vorteil HSI gegenüber RGB: 0.5 P.

   Helligkeit ist getrennt von Farbe und Sättigung: Unempfindlich gegenüber Beleuchtungsänderungen

3. 2 P.

   - Beschreibung des Operators Öffnen:
     (a) Teiloperationen: Anwendung von Erosion danach Dilatation
     (b) Effekt: Entfernt dünne Linien oder kleine außenliegende Objekte

   - Beschreibung des Operators Schließen:
     (a) Teiloperationen: Anwendung von Dilatation danach Erosion
     (b) Effekt: Überbrückung kleiner Distanzen und Schließung von inneren Löchern

4. Anwendung von *Öffnen* auf ein Bild 2 P.

   - Teilschritt 1: Erosion $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 255 & 255 & 0 & 0 & 0 & 0 \\ 255 & 255 & 0 & 0 & 0 & 255 \\ 0 & 255 & 0 & 0 & 0 & 0 \end{pmatrix}$

   - Teilschritt 2: Dilatation $\begin{pmatrix} 255 & 255 & 0 & 255 \\ 255 & 255 & 0 & 255 \end{pmatrix}$

Aufgabe 7 *Symbolisches Planen*

1. 2 P.

   Agent: robot, human (not necessary)

   Location: at-fridge, next-to-human, at-bar, at-stove

   Object: apple-juice, glass

2. **Prädikate:** 2 P.

   robotAt(L) grasped(O) handEmpty filled(O) empty(O)

   objectAt(O, L)
   delivered(O)
   closed(O)
   opened(O)

   isLocation(L) (not necessary), isObject(O) (not necessary), isAgent(A) (not necessary)

3. **Aktionen:** 4.5 P.

   moveToPosition(from, to)
   Pre: robotAt(from)
   Add: robotAt(to)
   Del: robotAt(from)

   grasp(object, location)
   Pre: (objectAt(object, location) ∧ robotAt(location) ∧ handEmpty
   Add: grasped(object)
   Del: handEmpty ∧ objectAt(object, location)

   handOver(object)
   Pre: grasped(object) ∧ robotAt(next-to-human)
   Add: delivered(object) ∧ handEmpty
   Del: grasped(object)

4. **Initial State:** 2.5 P.

   robotAt(at-fridge) ∧
   objectAt(glass, at-stove) ∧
   objectAt(apple-juice, at-bar) ∧
   empty(glass) ∧
   handEmpty ∧
   closed(apple-juice)

   Goal predicates:

   delivered(glass) ∧
   filled(glass)