
Modellierung eines Roboters

Nicolas Schäfer
Saarbrücken, 2. November 2025

Inhaltsverzeichnis

1	Modellierung eines Roboters	1
1.1	Two Link Revolute Manipulator	2
1.1.1	Formulierung der Massenmatrix	2
1.1.2	Aufstellen der Coriolis-Matrix	4
1.1.3	Gravitationsterme	5
1.2	Einfaches Greifobjekt	6
2	Theorie optimaler Steuerungsprobleme	8
2.1	Herleitung des Minimumprinzips	9
2.2	Anwendung auf Linear-Quadratische Probleme	10
2.2.1	Finite time problem	10
2.2.2	infinite time problem	11
3	Linearisierung der Dynamik	12
3.1	Linearisierung des two link revolute manipulators	13
3.2	Riccati Regler	15
3.2.1	Implementierung	16
4	Gradientenverfahren	18
4.1	Gradientenverfahren für den two link revolute Manipulator	19
5	Foundations of Reinforcement Learning	21
5.1	Bellman Optimality Equations	22
5.2	Value Iteration	23
5.3	Q-Learning	24
5.4	SARSA	26
6	Reinforcement Learning in Optimal Control	27
6.1	Policy-Based Algorithms	27
6.2	Deep Deterministic Policy Gradient	29
6.3	Two-Link-Revolute Manipulator	29

1 Modellierung eines Roboters

Wir halten uns an [5] und [4]

$$\begin{aligned}
 \underbrace{f - mg}_{\text{Kräftebilanz}} &= \underbrace{m\ddot{x}}_{\text{Newtons 2nd Law}} = \frac{d}{dt} [m\dot{x}] \\
 f - \frac{\partial}{\partial x} \underbrace{[mgx(t)]}_{\text{Potentielle Energie } V} &= \frac{d}{dt} \frac{\partial}{\partial \dot{x}} \underbrace{\left[m\frac{1}{2}\dot{x}(t)^2\right]}_{\text{kinetische Energie } T} \\
 f - \frac{\partial V}{\partial x} = \frac{d}{dt} \frac{\partial T}{\partial \dot{x}} &\Rightarrow f = \frac{d}{dt} \frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x}
 \end{aligned}$$

Wobei $L = T - V = \frac{1}{2}m\dot{x}(t)^2 - mgx(t)$ gilt. Für generalisierte Koordinaten $\mathbf{q} = \theta \in \mathbb{R}^n$

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}_i} \right) - \frac{\partial T}{\partial \mathbf{q}_i} + \frac{\partial V}{\partial \mathbf{q}_i} = \tau_i \quad i = 1, \dots, n$$

Für die kinetische Energie eines starren Körpers gilt:

$$T = \frac{1}{2}m\mathbf{v}^T \mathbf{v} + \frac{1}{2}\boldsymbol{\omega}^T I \boldsymbol{\omega}$$

Wobei \mathbf{v} die Geschwindigkeit des Schwerpunkts, $\boldsymbol{\omega}$ die Winkelgeschwindigkeit und I der Trägheitstensor bezüglich des Schwerpunkts ist. Für den Trägheitstensor gilt:

$$I = \int_V \rho(\mathbf{r})[(\mathbf{r} \cdot \mathbf{r})\mathbf{1} - \mathbf{r} \otimes \mathbf{r}]dV \quad (1)$$

Für ein Mehrkörpersystem mit generalisierten Koordinaten \mathbf{q} lässt sich die Gesamtenergie in kompakter Form schreiben als:

$$T = \frac{1}{2}\dot{\mathbf{q}}^T M(\mathbf{q})\dot{\mathbf{q}}$$

wobei $M(\mathbf{q})$ die konfigurationsabhängige Massenmatrix ist, die alle Massen und Trägheiten der einzelnen Glieder erfasst. Für die Geschwindigkeiten gilt mittels der Kettenregel:

$$\mathbf{v} = J_v(\mathbf{q})\dot{\mathbf{q}} \quad \text{und} \quad \boldsymbol{\omega} = J_\omega(\mathbf{q})\dot{\mathbf{q}} \quad J_v = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \quad \text{und} \quad J_\omega = \frac{\partial \boldsymbol{\omega}}{\partial \dot{\mathbf{q}}}$$

Einsetzen in die kinetische Energie liefert:

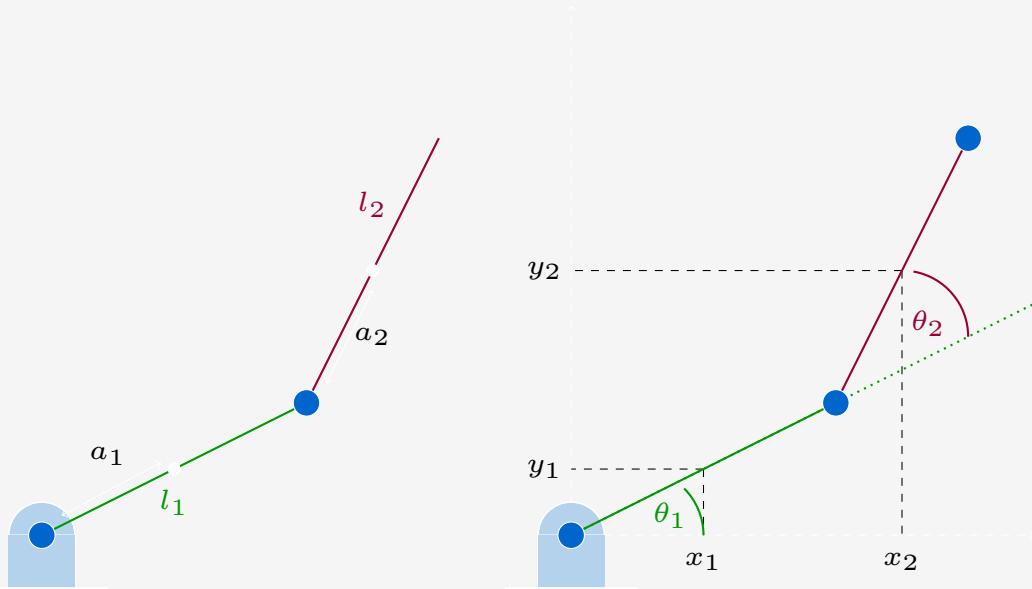
$$\begin{aligned}
 T &= \frac{1}{2}m(J_v\dot{\mathbf{q}})^T (J_v\dot{\mathbf{q}}) + \frac{1}{2}(J_\omega\dot{\mathbf{q}})^T I (J_\omega\dot{\mathbf{q}}) \\
 &= \frac{1}{2}\dot{\mathbf{q}}^T (mJ_v^T J_v)\dot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^T (J_\omega^T I J_\omega)\dot{\mathbf{q}} \\
 &= \frac{1}{2}\dot{\mathbf{q}}^T \underbrace{(mJ_v^T J_v + J_\omega^T I J_\omega)}_{M(\mathbf{q})} \dot{\mathbf{q}}
 \end{aligned}$$

Die Gesamtenergie setzt sich zusammen aus der Summe der Energien aller n Glieder:

$$M(\mathbf{q}) = \sum_{i=1}^n m_i J_{v_i}^T J_{v_i} + J_{\omega_i}^T I_i J_{\omega_i}$$

1.1 Two Link Revolute Manipulator

Wir bezeichnen $\mathbf{q} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ als die generalisierten Koordinaten des Systems. Die Schwerpunkte der beiden Links bezeichnen wir mit (x_1, y_1) und (x_2, y_2) . Die Abstände der Schwerpunkte von den Gelenken werden mit a_1 und a_2 bezeichnet.



Für die Koordinaten der Schwerpunkte gilt:

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = \begin{pmatrix} a_1 \cdot \cos(\theta_1) \\ a_1 \cdot \sin(\theta_1) \\ 0 \end{pmatrix} \quad \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = \begin{pmatrix} l_1 \cdot \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2) \\ l_1 \cdot \sin(\theta_1) + a_2 \sin(\theta_1 + \theta_2) \\ 0 \end{pmatrix}$$

1.1.1 Formulierung der Massenmatrix

Für die Massenmatrix gilt hier:

$$M(\mathbf{q}) = m_1 J_{v_1}^T J_{v_1} + J_{\omega_1}^T I_1 J_{\omega_1} + m_2 J_{v_2}^T J_{v_2} + J_{\omega_2}^T I_2 J_{\omega_2}$$

Hierbei bezeichnet J die Jacobi-Matrix, I_i das Trägheitstensor und m_i die Masse des i -ten Links.

$$J_{v_1} = \frac{\partial \mathbf{r}_1}{\partial \mathbf{q}} = \frac{\partial(x_1, y_1, z_1)}{\partial(\theta_1, \theta_2)} = \begin{pmatrix} \frac{\partial x_1}{\partial \theta_1} & \frac{\partial x_1}{\partial \theta_2} \\ \frac{\partial y_1}{\partial \theta_1} & \frac{\partial y_1}{\partial \theta_2} \\ \frac{\partial z_1}{\partial \theta_1} & \frac{\partial z_1}{\partial \theta_2} \end{pmatrix} = \begin{pmatrix} -a_1 \sin(\theta_1) & 0 \\ a_1 \cos(\theta_1) & 0 \\ 0 & 0 \end{pmatrix}$$

$$J_{v_2} = \frac{\partial \mathbf{r}_2}{\partial \mathbf{q}} = \frac{\partial(x_2, y_2, z_2)}{\partial(\theta_1, \theta_2)} = \begin{pmatrix} \frac{\partial x_2}{\partial \theta_1} & \frac{\partial x_2}{\partial \theta_2} \\ \frac{\partial y_2}{\partial \theta_1} & \frac{\partial y_2}{\partial \theta_2} \\ \frac{\partial z_2}{\partial \theta_1} & \frac{\partial z_2}{\partial \theta_2} \end{pmatrix} = \begin{pmatrix} -l_1 \sin(\theta_1) - a_2 \sin(\theta_1 + \theta_2) & -a_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2) & a_2 \cos(\theta_1 + \theta_2) \\ 0 & 0 \end{pmatrix}$$

Für das Matrizenprodukt $J_{v_1}^T J_{v_1}$ gilt:

$$\begin{aligned} J_{v_1}^T J_{v_1} &= \begin{pmatrix} -a_1 \sin(\theta_1) & a_1 \cos(\theta_1) & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} -a_1 \sin(\theta_1) & 0 \\ a_1 \cos(\theta_1) & 0 \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} a_1^2 \sin^2(\theta_1) + a_1^2 \cos^2(\theta_1) & 0 \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} a_1^2 & 0 \\ 0 & 0 \end{pmatrix} \end{aligned}$$

Für das Matrizenprodukt $J_{v_2}^T J_{v_2}$ gilt:

$$\begin{aligned} J_{v_2}^T J_{v_2} &= \begin{pmatrix} -l_1 \sin(\theta_1) - a_2 \sin(\theta_1 + \theta_2) & l_1 \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2) & 0 \\ -a_2 \sin(\theta_1 + \theta_2) & a_2 \cos(\theta_1 + \theta_2) & 0 \end{pmatrix} \\ &\cdot \begin{pmatrix} -l_1 \sin(\theta_1) - a_2 \sin(\theta_1 + \theta_2) & -a_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2) & a_2 \cos(\theta_1 + \theta_2) \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\theta_2)) & (a_2^2 + l_1 a_2 \cos(\theta_2)) \\ (a_2^2 + l_1 a_2 \cos(\theta_2)) & a_2^2 \end{pmatrix} \end{aligned}$$

Für die Winkelgeschwindigkeiten setzen wir $\omega_1 = (0 \ 0 \ \theta'_1)^T$ und $\omega_2 = (0 \ 0 \ \theta'_1 + \theta'_2)^T$.

$$\begin{aligned} J_{\omega_1} &= \frac{\partial \omega_1}{\partial \dot{\mathbf{q}}} = \frac{\partial (\omega_1)}{\partial (\theta'_1, \theta'_2)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\partial \omega_1}{\partial \theta'_1} & \frac{\partial \omega_1}{\partial \theta'_2} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \\ J_{\omega_2} &= \frac{\partial \omega_2}{\partial \dot{\mathbf{q}}} = \frac{\partial (\omega_2)}{\partial (\theta'_1, \theta'_2)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\partial \omega_2}{\partial \theta'_1} & \frac{\partial \omega_2}{\partial \theta'_2} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \end{aligned}$$

Für den Trägheitstensor I_i gilt:

$$I = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{pmatrix} = \begin{pmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int yx dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int zx dm & -\int zy dm & \int (x^2 + y^2) dm \end{pmatrix}$$

Für symmetrische Körper (z.B. Zylinder, Kugel, Quader) gilt: $I_{xy} = I_{xz} = I_{yz} = 0$ also erhalten wir eine Diagonalmatrix.

$$I_i = \begin{pmatrix} \int_y^2 dm & 0 & 0 \\ 0 & \int x^2 dm & 0 \\ 0 & 0 & \int (x^2 + y^2) dm \end{pmatrix} = \begin{pmatrix} \int_V \rho y^2 dV & 0 & 0 \\ 0 & \int_V \rho x^2 dV & 0 \\ 0 & 0 & \int_V \rho (x^2 + y^2) dV \end{pmatrix}$$

Für die Massenmatrix $M(\mathbf{q})$ gilt somit:

$$\begin{aligned} M(\mathbf{q}) &= m_1 J_{v_1}^T J_{v_1} + J_{\omega_1}^T I_1 J_{\omega_1} + m_2 J_{v_2}^T J_{v_2} + J_{\omega_2}^T I_2 J_{\omega_2} \\ &= m_1 \begin{pmatrix} a_1^2 & 0 \\ 0 & 0 \end{pmatrix} + \underbrace{\left(\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{1,xx} & 0 & 0 \\ 0 & I_{1,yy} & 0 \\ 0 & 0 & I_{1,zz} \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \right)}_{= \begin{pmatrix} 0 & 0 & I_{1,zz} \\ 0 & 0 & 0 \end{pmatrix}} \\ &\quad + m_2 \begin{pmatrix} (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\theta_2)) & (a_2^2 + l_1 a_2 \cos(\theta_2)) \\ (a_2^2 + l_1 a_2 \cos(\theta_2)) & a_2^2 \end{pmatrix} \\ &\quad + \underbrace{\left(\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} I_{2,xx} & 0 & 0 \\ 0 & I_{2,yy} & 0 \\ 0 & 0 & I_{2,zz} \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \right)}_{= \begin{pmatrix} 0 & 0 & I_{2,zz} \\ 0 & 0 & I_{2,zz} \end{pmatrix}} \\ &= \begin{pmatrix} m_1 a_1^2 + I_{1,zz} + m_2 \cdot (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\theta_2)) + I_{2,zz} & m_2 (a_2^2 + l_1 a_2 \cos(\theta_2)) + I_{2,zz} \\ m_2 (a_2^2 + l_1 a_2 \cos(\theta_2)) + I_{2,zz} & m_2 a_2^2 + I_{2,zz} \end{pmatrix} \end{aligned}$$

Die Trägheitselemente $I_{i,xx}$ und $I_{i,yy}$ sind nicht relevant. Für unser Model verwenden wir den Standardansatz eines dünnen Stabes der Länge l_i und Masse m_i . Somit gilt $z = y \approx 0$ und wir erhalten: $I_{i,xx} = I_{i,yy} = 0$

$$I_{i,zz} \approx \int_{-\frac{l_i}{2}}^{\frac{l_i}{2}} \underbrace{\rho}_{\frac{m_i}{l_i}} x^2 dx = \frac{1}{3} \cdot \frac{m_i}{l_i} \cdot \left[\left(\frac{l_i}{2} \right)^3 - \left(-\frac{l_i}{2} \right)^3 \right] = \frac{1}{12} m_i l_i^2 \Rightarrow I_i = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12} m_i l_i^2 \end{pmatrix}$$

1.1.2 Aufstellen der Coriolis-Matrix

Für die partiellen Ableitungen der kinetischen Energie $T(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T M(\mathbf{q}) \dot{\mathbf{q}}$ gilt:

$$\begin{aligned} \frac{\partial T}{\partial \theta_1} &= \frac{1}{2} \frac{\partial M_{11}}{\partial \theta_1} \theta'_1{}^2 + \frac{\partial M_{12}}{\partial \theta_1} \theta'_1 \theta'_2 + \frac{1}{2} \frac{\partial M_{22}}{\partial \theta_1} \theta'_2(t)^2 \\ \frac{\partial T}{\partial \theta_2} &= \frac{1}{2} \frac{\partial M_{11}}{\partial \theta_2} \theta'_1{}^2 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_1 \theta'_2 + \frac{1}{2} \frac{\partial M_{22}}{\partial \theta_2} \theta'_2(t)^2 \\ \frac{\partial T}{\partial \theta'_1} &= M_{11} \theta'_1 + M_{12} \theta'_2 \\ \frac{\partial T}{\partial \theta'_2} &= M_{12} \theta'_1 + M_{22} \theta'_2 \end{aligned}$$

Für die zeitlichen Ableitungen der Größen gilt

$$\begin{aligned} \frac{d}{dt} \frac{\partial T}{\partial \theta'_1} &= \underbrace{\frac{d}{dt} M_{11}}_{\frac{\partial M_{11}}{\partial \theta_1} \theta'_1 + \frac{\partial M_{11}}{\partial \theta_2} \theta'_2} \cdot \theta'_1 + M_{11} \theta''_1 + \underbrace{\frac{d}{dt} M_{12}}_{\frac{\partial M_{12}}{\partial \theta_1} \theta'_1 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_2} \cdot \theta'_2 + M_{12} \theta''_2 \\ &= \frac{\partial M_{11}}{\partial \theta_1} \theta'_1{}^2 + \frac{\partial M_{11}}{\partial \theta_2} \theta'_1 \theta'_2 + M_{11} \theta''_1 + \frac{\partial M_{12}}{\partial \theta_1} \theta'_1 \theta'_2 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_2{}^2 + M_{12} \theta''_2 \\ &= \frac{\partial M_{11}}{\partial \theta_1} \theta'_1{}^2 + \left[\frac{\partial M_{11}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_1} \right] \theta'_1 \theta'_2 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_2{}^2 + M_{11} \theta''_1 + M_{12} \theta''_2 \end{aligned}$$

$$\begin{aligned} \frac{d}{dt} \frac{\partial T}{\partial \theta'_2} &= \underbrace{\frac{d}{dt} M_{12}}_{\frac{\partial M_{12}}{\partial \theta_1} \theta'_1 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_2} \cdot \theta'_1 + M_{12} \theta''_1 + \underbrace{\frac{d}{dt} M_{22}}_{\frac{\partial M_{22}}{\partial \theta_1} \theta'_1 + \frac{\partial M_{22}}{\partial \theta_2} \theta'_2} \cdot \theta'_2 + M_{22} \theta''_2 \\ &= \frac{\partial M_{12}}{\partial \theta_1} \theta'_1{}^2 + \frac{\partial M_{12}}{\partial \theta_2} \theta'_1 \theta'_2 + M_{12} \theta''_1 + \frac{\partial M_{22}}{\partial \theta_1} \theta'_1 \theta'_2 + \frac{\partial M_{22}}{\partial \theta_2} \theta'_2{}^2 + M_{22} \theta''_2 \end{aligned}$$

Es gilt

$$\begin{aligned} \frac{d}{dt} \frac{\partial T}{\partial \theta'_1} - \frac{\partial T}{\partial \theta_1} &= \left[\frac{\partial M_{11}}{\partial \theta_1} - \frac{1}{2} \frac{\partial M_{11}}{\partial \theta_2} \right] \theta'_1{}^2 + \left[\frac{\partial M_{11}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_1} - \frac{\partial M_{12}}{\partial \theta_2} \right] \theta'_1 \theta'_2 + \left[\frac{\partial M_{12}}{\partial \theta_2} - \frac{1}{2} \frac{\partial M_{22}}{\partial \theta_1} \right] \theta'_2{}^2 \\ &\quad + M_{11} \theta''_1 + M_{12} \theta''_2 \end{aligned}$$

$$\begin{aligned} \frac{d}{dt} \frac{\partial T}{\partial \theta'_2} - \frac{\partial T}{\partial \theta_2} &= \left[\frac{\partial M_{12}}{\partial \theta_1} - \frac{1}{2} \frac{\partial M_{11}}{\partial \theta_2} \right] \theta'_1{}^2 + \left[\frac{\partial M_{12}}{\partial \theta_2} + \frac{\partial M_{22}}{\partial \theta_1} - \frac{\partial M_{12}}{\partial \theta_2} \right] \theta'_1 \theta'_2 + \left[\frac{\partial M_{22}}{\partial \theta_2} - \frac{1}{2} \frac{\partial M_{22}}{\partial \theta_2} \right] \theta'_2{}^2 \\ &\quad + M_{12} \theta''_1 + D_{22} \theta''_2 \\ &= \underbrace{\left[\frac{\partial M_{12}}{\partial \theta_1} \theta'_1 - \frac{1}{2} \frac{\partial M_{11}}{\partial \theta_2} \theta'_1 \right]}_{C_{21}} \theta'_1 + \underbrace{\left[\frac{1}{2} \frac{\partial M_{22}}{\partial \theta_2} \theta'_2 + \frac{\partial M_{22}}{\partial \theta_1} \theta'_1 \right]}_{C_{22}} \theta'_2 + M_{12} \theta''_1 + M_{22} \theta''_2 \end{aligned}$$

Für die Einträge der Matrix C gilt

$$\begin{aligned}
 C_{11} &= \frac{1}{2} \underbrace{\frac{\partial M_{11}}{\partial \theta_1}}_{=0} \theta'_1 + \underbrace{\frac{\partial M_{11}}{\partial \theta_2}}_{=-2m_2 l_1 a_2 \sin(\theta_2)} \theta'_2 = -2m_2 l_1 a_2 \sin(\theta_2) \theta'_2 \\
 C_{12} &= \underbrace{\frac{\partial M_{12}}{\partial \theta_2}}_{=-m_2 l_1 a_2 \sin(\theta_2)} \theta'_2 - \frac{1}{2} \underbrace{\frac{\partial M_{22}}{\partial \theta_1}}_{=0} \theta'_1 = -m_2 l_1 a_2 \sin(\theta_2) \theta'_2 \\
 C_{21} &= \underbrace{\frac{\partial M_{12}}{\partial \theta_1}}_{=0} \theta'_1 - \frac{1}{2} \underbrace{\frac{\partial M_{11}}{\partial \theta_2}}_{=-2m_2 l_1 a_2 \sin(\theta_2)} \theta'_2 = m_2 l_1 a_2 \sin(\theta_2) \theta'_1 \\
 C_{22} &= \underbrace{\frac{1}{2} \frac{\partial M_{22}}{\partial \theta_2}}_{=0} \theta'_2 + \underbrace{\frac{\partial M_{22}}{\partial \theta_1}}_{=0} \theta'_1 = 0
 \end{aligned}$$

Es gilt somit

$$C(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} = \begin{pmatrix} -2m_2 l_1 a_2 \sin(\theta_2) \theta'_2 & -m_2 l_1 a_2 \sin(\theta_2) \theta'_2 \\ m_2 l_1 a_2 \sin(\theta_2) \theta'_1 & 0 \end{pmatrix} \begin{pmatrix} \theta'_1 \\ \theta'_2 \end{pmatrix}$$

1.1.3 Gravitationsterme

Für die potenzielle Energie V gilt:

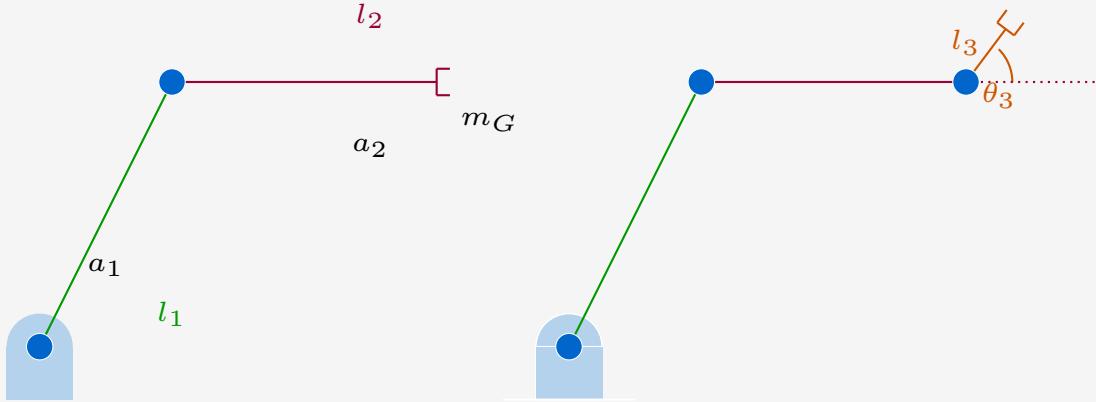
$$V = m_1 \cdot g \cdot y_1 + m_2 \cdot g \cdot y_2 = m_1 g a_1 \sin(\theta_1) + m_2 g (l_1 \sin(\theta_1) + a_2 \sin(\theta_1 + \theta_2))$$

Für die partiellen Ableitungen erhalten wir

$$G(\mathbf{q}) = \frac{\partial V}{\partial \mathbf{q}} = \begin{pmatrix} \frac{\partial V}{\partial \theta_1} \\ \frac{\partial V}{\partial \theta_2} \end{pmatrix} = \begin{pmatrix} m_1 g a_1 \cos(\theta_1) + m_2 g (l_1 \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2)) \\ m_2 g (l_1 \cos(\theta_1) + a_2 \cos(\theta_1 + \theta_2)) \end{pmatrix}$$

1.2 Einfaches Greifobjekt

Wir modellieren ein Variables Greifobjekt, welches am Ende von Link 2 gehalten wird. Wir betrachten nachfolgend das Greifobjekt als Punktmasse mit Masse m_G .



Für die Masse von Link 2 inklusive Greifobjekt gilt:

$$m'_i = m_i + m_G$$

Für den Schwerpunkt von Link 2 inklusive Greifobjekt gilt:

$$\mathbf{a}'_2 = \frac{m_i \cdot \mathbf{a}_2 + m_G \cdot \mathbf{d}}{m_i + m_G} \quad \mathbf{d}(\theta_3) = \begin{pmatrix} l_2 + r \cdot \cos(\theta_3) \\ r \cdot \sin(\theta_3) \\ 0 \end{pmatrix}$$

Für den Trägheitstensor von Link 2 inklusive Greifobjekt gilt:

$$I'_i = I_i + m_i \cdot S(\mathbf{a}_i - \mathbf{a}'_i) + m_G \cdot S(\mathbf{d} - \mathbf{a}'_i) \quad S(\mathbf{v}) = \begin{pmatrix} v_y^2 + v_z^2 & -v_x v_y & -v_x v_z \\ -v_x v_y & v_x^2 + v_z^2 & -v_y v_z \\ -v_x v_z & -v_y v_z & v_x^2 + v_y^2 \end{pmatrix}$$

Wobei $S(\mathbf{v})$ die Steiner-Matrix ist. Wir erhalten die Matrix direkt aus der Definition des Trägheitstensors einer Punktmasse mit $\rho(\mathbf{r}) = m \cdot \delta(\mathbf{r} - \mathbf{d})$. Für die Verschiebungsvektoren gilt:

$$\mathbf{d} = \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{a}_2 = \begin{pmatrix} \frac{l_2}{2} \\ 0 \\ 0 \end{pmatrix} \Rightarrow \mathbf{a}'_2 = \frac{l_2}{m_2 + m_G} \begin{pmatrix} \frac{m_2}{2} + m_G \\ 0 \\ 0 \end{pmatrix}$$

Für die Steiner-Matrizen gilt entsprechend

$$S(\mathbf{a}_2 - \mathbf{a}'_2) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & [\frac{l_2}{2} - a'_2]^2 & 0 \\ 0 & 0 & [\frac{l_2}{2} - a'_2]^2 \end{pmatrix} \quad S(\mathbf{d} - \mathbf{a}'_2) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & [l_2^2 - a'_2]^2 & 0 \\ 0 & 0 & [l_2^2 - a'_2]^2 \end{pmatrix}$$

Somit erhalten wir für den Trägheitstensor von Link 2 inklusive Greifobjekt:

$$I'_2 = \begin{pmatrix} I_{2,xx} & 0 & 0 \\ 0 & I_{2,yy} + m_2 \cdot \left[\frac{l_2}{2} - a'_2 \right]^2 + m_G \cdot \left[\frac{l_2}{2} - a'_2 \right]^2 & 0 \\ 0 & 0 & I_{2,zz} + m_2 \cdot \left[\frac{l_2}{2} - a'_2 \right]^2 + m_G \cdot \left[\frac{l_2}{2} - a'_2 \right]^2 \end{pmatrix}$$

Für die Koordinaten des Schwerpunktes von Link 3 gilt:

$$\mathbf{r}_3 = \begin{pmatrix} l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) \\ l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) \\ 0 \end{pmatrix}$$

Berechnung der Jacobi-Matrizen für Link 3 bezüglich der Geschwindigkeiten:

$$J_{v_3} = \frac{\partial \mathbf{r}_3}{\partial \mathbf{q}}$$

Für die Winkelgeschwindigkeit des Links 3 setzen wir $\boldsymbol{\omega}_3 = (0 \ 0 \ \theta'_1 + \theta'_2 + \theta'_3)$ und erhalten für die Jacobi-Matrix bezüglich der Winkelgeschwindigkeit:

$$J_{\omega_3} = \frac{\partial \boldsymbol{\omega}_3}{\partial \dot{\mathbf{q}}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

2 Theorie optimaler Steuerungsprobleme

Wir halten uns an die Definitionen in [1]

Definition 2.1 (Optimales Steuerungsproblem) Seien $t_0 < t_f$ feste Zeiten und

$$\begin{aligned}\Phi : \mathbb{R}^{n_x} &\rightarrow \mathbb{R}, \\ L : [t_0, t_f] \times \mathbb{R}^{n_x} &\rightarrow \mathbb{R}, \\ f : [t_0, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} &\rightarrow \mathbb{R}^{n_x}\end{aligned}$$

hinreichend glatte Funktionen und $\mathcal{U} \subset \mathbb{R}^{n_u}$ eine abgeschlossene konvexe nichtleere Menge.

$$\min_{\mathbf{u}} \int_{t_0}^{t_f} L(t, \mathbf{x}(t), \mathbf{u}(t)) dt + \Phi(\mathbf{x}(t_f))$$

mit $\mathbf{x} \in W_{1,\infty}^{n_x}([t_0, t_f]), \mathbf{u} \in L_\infty^{n_u}([t_0, t_f])$ und

$$\dot{\mathbf{x}}(t) = f(t, \mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad \mathbf{u}(t) \in \mathcal{U} \text{ f.ü. } t \in [t_0, t_f]$$

Wir betrachten ein Steuerungsproblem und definieren die Value-Function V mittels

$$V(x(t), \textcolor{brown}{t}) := \min_{\textcolor{red}{u}} \left\{ \int_{\textcolor{brown}{t}}^{t_f} L(s, x(s), \textcolor{red}{u}(s)) ds + \Phi(x(t_f)) \right\}$$

Satz 2.2 (Hamilton-Jacobi-Bellman Gleichung) Die Value-Function erfüllt

$$0 = \frac{\partial V(x, t)}{\partial t} + \min_u \left[L(t, x, u) + \frac{\partial V(x, t)}{\partial x} \cdot f(t, x, u) \right], \quad x \in \mathbb{R}^n, t_0 \leq t \leq t_f$$

mit den Randbedingungen

$$V(x, \textcolor{violet}{t}_f) = \Phi(x(\textcolor{violet}{t}_f)) \quad V(x, \textcolor{brown}{t}_0) = \min_u \int_{\textcolor{brown}{t}_0}^{t_f} L(t, x(t), u(t)) dt + \Phi(x(t_f))$$

Beweis:

Die Value-Function $V(x, t)$ beschreibt die minimalen Kosten von Zustand x zum Endzeitpunkt t_f . Für $s \in [t, t_f]$ gilt

- Systemdynamik

$$x'(s) = f(x(s), u(s)), \quad x(t) = x$$

- Kostenfunktional

$$J(u) = \int_t^{t_f} L(s, x(s), u(s)) ds + \Phi(x(t_f))$$

Angenommen, wir wechseln zum Zeitpunkt $t + h$ zu einer optimalen Steuerung, dann gilt für unser Zielfunktional

$$\begin{aligned}J(u) &= \int_t^{t+h} L(s, x(s), u(s)) ds + \int_{t+h}^{t_f} L(s, x(s), u(s)) ds + \Phi(x(t_f)) \\ &= \int_t^{t+h} L(s, x(s), u(s)) ds + V(x(t+h), t+h)\end{aligned}$$

Da $V(x(t), t)$ die minimalen Kosten vom Zustand x zur Zeit t angibt, gilt:

$$\begin{aligned} V(x(t), t) &\leq \int_t^{t+h} L(s, x(s), u(s)) \, ds + V(x(t+h), t+h) \\ \Rightarrow 0 &\leq \int_t^{t+h} L(s, x(s), u(s)) \, ds + V(x(t+h), t+h) - V(x(t), t) \end{aligned}$$

Im Grenzübergang $h \rightarrow 0$ erhalten wir

$$\begin{aligned} 0 &\leq \lim_{h \rightarrow 0} \frac{1}{h} \int_t^{t+h} L(s, x(s), u(s)) \, ds + \lim_{h \rightarrow 0} \frac{V(x(t+h), t+h) - V(x(t), t)}{h} \\ &= L(t, x(t), u(t)) + \frac{d}{dt} V(x(t), t) \\ &= L(t, x(t), u(t)) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \cdot x'(t) \\ &= L(t, x(t), u(t)) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \cdot f(t, x(t), u(t)) \end{aligned}$$

Durch Minimierung bezüglich u erhalten wir die Gleichheit und somit

$$\begin{aligned} 0 &= \frac{\partial V}{\partial t} + \min_u \left[L(t, x(t), u(t)) + \frac{\partial V}{\partial x} \cdot f(t, x(t), u(t)) \right] \\ &= \frac{\partial V}{\partial t} + H \left(t, x(t), \frac{\partial V}{\partial x}(t, x(t)) \right) \end{aligned}$$

□

2.1 Herleitung des Minimumprinzips

Berechnung der totalen Ableitung der Value-Function V entlang der Charakteristik $x(t)$:

$$\frac{dV}{dt} = \frac{\partial V}{\partial x} \frac{dx}{dt} + \frac{\partial V}{\partial t} = p(x, t) \cdot x'(t) - H \left(t, x, \frac{\partial V}{\partial x} \right)$$

Berechnung der totalen Ableitung von $p = \frac{\partial V}{\partial x}$:

$$\frac{dp}{dt} = \frac{d}{dt} \left(\frac{\partial V}{\partial x} \right) = \frac{\partial^2 V}{\partial x^2} \cdot x'(t) + \frac{\partial V}{\partial x \partial t}$$

Differenzieren der HJB bezüglich x liefert

$$0 = \frac{d}{dx} \left[\frac{\partial V}{\partial t} + H \left(t, x, \frac{\partial V}{\partial x} \right) \right] = \frac{\partial V}{\partial t \partial x} + \frac{\partial H}{\partial x} + \frac{\partial H}{\partial p} \cdot \frac{\partial^2 V}{\partial x^2}$$

Einsetzen der Gleichung liefert:

$$\frac{dp}{dt} = \frac{\partial^2 V}{\partial x^2} \cdot x'(t) + \left(-\frac{\partial H}{\partial x} - \frac{\partial H}{\partial p} \cdot \frac{\partial^2 V}{\partial x^2} \right) = \frac{\partial^2 V}{\partial x^2} \cdot \left(x'(t) - \frac{\partial H}{\partial p} \right) - \frac{\partial H}{\partial x}$$

Wir erhalten somit die bekannten, notwendigen Optimalitätsbedingungen:

$$p'(t) = -H_x, \quad x'(t) = f(t, x, u^*), \quad u^* = \arg\min [L(t, x, u) + p(t, x)f(t, x, u)]$$

2.2 Anwendung auf Linear-Quadratische Probleme

2.2.1 Finite time problem

Gegeben sei das Steuerungsproblem

$$\min_u \int_{t_0}^{t_f} x^T Qx + u^T Ru dt + \underbrace{(x(t_f) - x_f)^T S(x(t_f) - x_f)}_{x(t_f)^T Sx(t_f) - 2x_f^T Sx(t_f) + x_f^T Sx_f} = V(t_0, x(t_0)) + x_f^T Sx_f$$

unter der Nebenbedingung $x'(t) = Ax + Bu$ und $x(t_0) = x_0$. Der Term $x_f^T Sx_f$ ist unabhängig von u und kann bei der Minimierung weggelassen werden.

$$V(x, t) = x^T K(t)x + 2s(t)^T x + r(t) \quad V(x, t_f) = x(t_f)^T K(t_f)x(t_f) + 2s(t_f)^T x(t_f) + r(t_f)$$

Wir erhalten für die Endwerte

$$K(t_f) = S, \quad s(t_f) = -Sx_f \quad r(t_f) = 0$$

$$V_t = x^T K'(t)x + 2s'(t)^T x + r'(t) \quad V_x = 2K(t)x + 2s(t)$$

$$\frac{\partial}{\partial u} [x^T Qx + u^T Ru + V_x^T \cdot (Ax + Bu)] = 2Ru + B^T \cdot V_x \Rightarrow u^*(t) = -\frac{1}{2}R^{-1}B^T V_x$$

$$\begin{aligned} (u^*)^T Ru^* &= \left(-\frac{1}{2}R^{-1}B^T V_x \right)^T R \left(-\frac{1}{2}R^{-1}B^T V_x \right) \\ &= \frac{1}{4} \cdot (R^{-1}B^T V_x)^T R (R^{-1}B^T V_x) \\ &= \frac{1}{4} \cdot (2R^{-1}B^T (K(t)x + s(t)))^T R (2R^{-1}B^T (K(t)x + s(t))) \\ &= (K(t)x + s(t))^T BR^{-1}RR^{-1}B^T (K(t)x + s(t)) \\ &= \langle K(t)x + s(t), P(2K(t)x + s(t)) \rangle \\ &= \langle K(t)x, PK(t)x \rangle + \underbrace{\langle K(t)x, Ps(t) \rangle}_{\langle s(t), PK(t)x \rangle} + \langle s(t), PK(t)x \rangle + \langle s(t), Ps(t) \rangle \\ &= x^T K(t)^T PK(t)x + 2 \cdot s(t)^T PK(t)x + s(t)^T Ps(t) \end{aligned}$$

mit $P = BR^{-1}B^T$ und $P^T = P$.

$$\begin{aligned} V_x^T \left(-\frac{1}{2}BR^{-1}B^T V_x \right) &= (2K(t)x + 2s(t))^T \left(-\frac{1}{2}BR^{-1}B^T (2K(t)x + 2s(t)) \right) \\ &= -2 \cdot (K(t)x + s(t))^T P(K(t)x + s(t)) \\ &= -2 \cdot x^T K(t)^T PK(t)x - 4 \cdot s(t)^T PK(t)x - 2 \cdot s(t)^T Ps(t) \end{aligned}$$

$$V_x^T Ax = 2(K(t)x + s(t))^T Ax = 2x^T K(t)^T Ax + 2s(t)^T Ax = x^T (K(t)^T A + A^T K(t))x + 2s(t)^T Ax$$

$$-V_t = \min_u [x^T Qx + u^T Ru + V_x^T \cdot (Ax + Bu)]$$

$$\begin{aligned} -V_t &= x^T Qx + (u^*)^T Ru^* + V_x^T \cdot \left(Ax - \frac{1}{2} BR^{-1} B^T V_x \right) \\ -x^T K'(t)x - 2s'(t)^T x - r'(t) &= x^T (Q - KPK + KA + A^T K) x - 2 \cdot ((PK - A)^T s(t))^T x - s(t)^T Ps(t) \end{aligned}$$

$$\begin{aligned} K'(t) &= -Q + K(t)BR^{-1}B^T K(t) - K(t)A - A^T K(t) \quad K(t_f) = S \\ s'(t) &= (K(t)BR^{-1}B^T - A^T)s(t) \quad s(t_f) = -Sx_f \\ r'(t) &= s(t)^T BR^{-1}B^T s(t) \quad r(t_f) = 0 \end{aligned}$$

2.2.2 infinite time problem

Im Grenzfall $t \rightarrow \infty$ betrachten wir das Steuerungsproblem

$$\min_u \int_{t_0}^{\infty} (x - x_f)^T Q(x - x_f) + u^T Ru dt = V(x(t_0))$$

unter der Nebenbedingung $x'(t) = Ax + Bu$ und $x(t_0) = x_0$. Als Ansatz für die Value Funktion wählen wir erneut

$$V(x) = x^T Kx + 2s^T x + r \Rightarrow V_x = 2Kx + 2s$$

Es gilt erneut

$$\frac{\partial}{\partial u} ((x - x_f)^T Q(x - x_f) + u^T Ru + V_x^T (Ax + Bu)) = 2Ru + B^T V_x = 0 \Rightarrow u^* = -\frac{1}{2} R^{-1} B^T V_x$$

$$(x - x_f)^T Q(x - x_f) = x^T Qx - 2x_f^T Qx + x_f^T Qx_f$$

$$0 = x^T (Q - KPK + KA + A^T K) x - 2((PK - A)^T s + Qx_f)^T x - s^T (P)s + x_f^T Qx_f$$

3 Linearisierung der Dynamik

Wir gehen analog zu [2] vor. Wir definieren den Zustand $\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}$ und schreiben unsere Dynamikgleichung wir folgt um

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + G(\mathbf{q}) = \mathbf{u} \Rightarrow \underbrace{\begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} \dot{\mathbf{q}} \\ M^{-1}(\mathbf{q})[\mathbf{u} - C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - G(\mathbf{q})] \end{bmatrix}}_{f(\mathbf{x}, \mathbf{u})}$$

Wir linearisieren um den Arbeitspunkt $(\mathbf{x}^*, \mathbf{u}^*)$

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}) \\ &\approx \underbrace{f(\mathbf{x}^*, \mathbf{u}^*)}_{=0} + \underbrace{\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*)(\mathbf{x} - \mathbf{x}^*)}_{A(\mathbf{x}^*, \mathbf{u}^*)} + \underbrace{\frac{\partial f}{\partial \mathbf{u}}(\mathbf{x}^*, \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)}_{B(\mathbf{x}^*, \mathbf{u}^*)} \\ &= A(\mathbf{x}^*, \mathbf{u}^*)\bar{\mathbf{x}} + B(\mathbf{x}^*, \mathbf{u}^*)\bar{\mathbf{u}} \end{aligned}$$

Wobei $\bar{\mathbf{x}}(t) = \mathbf{x}(t) - \mathbf{x}^*$ und $\bar{\mathbf{u}}(t) = \mathbf{u}(t) - \mathbf{u}^*$ gilt. Für die Ableitung der Größen gilt

$$\frac{d\bar{\mathbf{x}}}{dt} = \dot{\mathbf{x}} - \underbrace{\frac{d\mathbf{x}^*}{dt}}_{=0} = \dot{\mathbf{x}}(t) \Rightarrow \bar{\mathbf{x}}(t) = A\bar{\mathbf{x}}(t) + B\bar{\mathbf{u}}(t)$$

Somit ist die Rückführung von $\bar{\mathbf{x}}$ auf $\mathbf{0}$ äquivalent zu der Rückführung von \mathbf{x} auf den Arbeitspunkt \mathbf{x}^* . Wir definieren $h(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}) := \mathbf{u} - C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - G(\mathbf{q})$. und wenden die Produktregel auf den Term $M(\mathbf{q})\ddot{\mathbf{q}}$ an, um die Ableitung von f nach \mathbf{x} zu bestimmen.

$$\frac{\partial}{\partial \mathbf{x}} \left[M(\mathbf{q})\ddot{\mathbf{q}} \right] = \frac{\partial}{\partial \mathbf{x}} \left[h(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}) \right] \Rightarrow \frac{\partial M}{\partial \mathbf{x}}\ddot{\mathbf{q}} + M(\mathbf{q})\frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{x}} = \frac{\partial h}{\partial \mathbf{x}}$$

Werten wir h im Arbeitspunkt $(\mathbf{x}^*, \mathbf{u}^*)$ aus, so gilt $\ddot{\mathbf{q}} = \dot{\mathbf{q}} = \mathbf{0}$, somit fällt der erste Term weg und wir erhalten

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*) = \underbrace{\frac{\partial M}{\partial \mathbf{x}}}_{=0}\ddot{\mathbf{q}} + M(\mathbf{x}^*)\frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{x}} \Rightarrow \frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{x}} = M^{-1}(\mathbf{q}^*)\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*)$$

Für die Ableitung der Funktion $h = \mathbf{u} - C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - G(\mathbf{q})$ in $(\mathbf{x}^*, \mathbf{u}^*)$ gilt

$$\begin{aligned} \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*) &= \begin{bmatrix} \frac{\partial h}{\partial \mathbf{q}} & \frac{\partial h}{\partial \dot{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} -\frac{\partial C}{\partial \mathbf{q}}\dot{\mathbf{q}} - \frac{\partial G}{\partial \mathbf{q}} & -C(\mathbf{q}, \dot{\mathbf{q}}) - \frac{\partial C}{\partial \dot{\mathbf{q}}}\dot{\mathbf{q}} \end{bmatrix} (\mathbf{q}^*, \dot{\mathbf{q}}^* = \mathbf{0}) \\ &= \begin{bmatrix} -\frac{\partial G}{\partial \mathbf{q}} & -C(\mathbf{q}^*, \mathbf{0}) \end{bmatrix} \end{aligned}$$

Fassen wir die Ergebnisse zusammen, so gilt für die Ableitung von f nach \mathbf{x} im Arbeitspunkt $(\mathbf{x}^*, \mathbf{u}^*)$

$$A(\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} & \frac{\partial \dot{\mathbf{q}}}{\partial \dot{\mathbf{q}}} \\ \frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{q}} & \frac{\partial \ddot{\mathbf{q}}}{\partial \dot{\mathbf{q}}} \end{bmatrix} (\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} 0 & I \\ M^{-1}(\mathbf{q}^*) \begin{bmatrix} -\frac{\partial G}{\partial \mathbf{q}}(\mathbf{q}^*) \end{bmatrix} & M^{-1}(\mathbf{q}^*)[-C(\mathbf{q}^*, \mathbf{0})] \end{bmatrix}$$

Für die Ableitung von f nach \mathbf{u} gilt

$$B(\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{u}} \\ \frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{u}} \end{bmatrix} (\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} 0 \\ M^{-1}(\mathbf{q}^*) \end{bmatrix}$$

Satz 3.1 Gegeben sei die Dynamikgleichung

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + G(\mathbf{q}) = \mathbf{u} \Rightarrow \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$$

und ein Arbeitspunkt $(\mathbf{x}^*, \mathbf{u}^*)$ mit $f(\mathbf{x}^*, \mathbf{u}^*) = \mathbf{0}$. Dann gilt für die Linearisierung:

$$\begin{aligned} \dot{\mathbf{x}} &= A(\mathbf{x}^*, \mathbf{u}^*)\mathbf{x} + B(\mathbf{x}^*, \mathbf{u}^*)\mathbf{u} \\ &= \begin{bmatrix} 0 & I \\ M^{-1}(\mathbf{q}^*) \begin{bmatrix} -\frac{\partial G}{\partial \mathbf{q}}(\mathbf{q}^*) \end{bmatrix} & M^{-1}(\mathbf{q}^*) [-C(\mathbf{q}^*, \mathbf{0})] \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ M^{-1}(\mathbf{q}^*) \end{bmatrix} \mathbf{u} \end{aligned}$$

3.1 Linearisierung des two link revolute manipulators

Wir definieren $x_1 = q_1$, $x_2 = q_2$, $x_3 = \dot{q}_1$, $x_4 = \dot{q}_2$ und erhalten $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ mit

$$\begin{bmatrix} f_1(\mathbf{x}, \mathbf{u}) \\ f_2(\mathbf{x}, \mathbf{u}) \end{bmatrix} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \quad \begin{bmatrix} f_3(\mathbf{x}, \mathbf{u}) \\ f_4(\mathbf{x}, \mathbf{u}) \end{bmatrix} = M(\mathbf{q})^{-1} [\mathbf{u} - C(\mathbf{x})\dot{\mathbf{q}} - G(\mathbf{q})]$$

Für die Corioliskraft im Punkt $(\mathbf{x}^*, \mathbf{u}^*)$, bzw. für die Matrix $C(\mathbf{q}, \dot{\mathbf{q}} = \mathbf{0})$ gilt:

$$C(\mathbf{q}^*, \mathbf{0}) = \begin{bmatrix} 0 & -m_2 l_1 a_2 \sin(q_2^*) \cdot 0 \\ m_2 l_1 a_2 \sin(q_2^*) \cdot 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Wir berechnen zunächst die Ableitung von h bzgl. \mathbf{x} im Punkt $(\mathbf{x}^*, \mathbf{u}^*)$ und erhalten

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} -\frac{\partial G}{\partial \mathbf{q}} & -\underbrace{C(\mathbf{q}^*, \mathbf{0})}_{=\mathbf{0}} \end{bmatrix} = \begin{bmatrix} -\frac{\partial G_1}{\partial q_1}(\mathbf{q}^*) & -\frac{\partial G_1}{\partial q_2}(\mathbf{q}^*) & 0 & 0 \\ -\frac{\partial G_2}{\partial q_1}(\mathbf{q}^*) & -\frac{\partial G_2}{\partial q_2}(\mathbf{q}^*) & 0 & 0 \end{bmatrix}$$

Für die konkrete Matrix $A = f_{\mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*)$ gilt

$$A(\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \frac{\partial f_1}{\partial x_4} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \frac{\partial f_3}{\partial x_4} \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \\ \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \frac{\partial f_1}{\partial u_3} & \frac{\partial f_1}{\partial u_4} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \frac{\partial f_2}{\partial u_3} & \frac{\partial f_2}{\partial u_4} \\ \frac{\partial f_3}{\partial u_1} & \frac{\partial f_3}{\partial u_2} & \frac{\partial f_3}{\partial u_3} & \frac{\partial f_3}{\partial u_4} \\ \frac{\partial f_4}{\partial u_1} & \frac{\partial f_4}{\partial u_2} & \frac{\partial f_4}{\partial u_3} & \frac{\partial f_4}{\partial u_4} \end{bmatrix} (\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -M^{-1}(\mathbf{q}^*) \frac{\partial G}{\partial \mathbf{q}}(\mathbf{q}^*) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Für die konkrete Matrix $B = f_{\mathbf{u}}(\mathbf{x}^*, \mathbf{u}^*)$ gilt

$$B(\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} \\ \frac{\partial f_3}{\partial u_1} & \frac{\partial f_3}{\partial u_2} \\ \frac{\partial f_4}{\partial u_1} & \frac{\partial f_4}{\partial u_2} \end{bmatrix} (\mathbf{x}^*, \mathbf{u}^*) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ M^{-1}(\mathbf{q}^*) \end{bmatrix}$$

Für die konkret Berechnung der partiellen Ableitungen der Gravitationskraft gilt

$$\begin{aligned} G(\mathbf{q}) &= \begin{bmatrix} m_1 g a_1 \cos(\textcolor{green}{q}_1) + m_2 g (l_1 \cos(\textcolor{green}{q}_1) + a_2 \cos(\textcolor{green}{q}_1 + \textcolor{red}{q}_2)) \\ m_2 g (l_1 \cos(\textcolor{green}{q}_1) + a_2 \cos(\textcolor{green}{q}_1 + \textcolor{red}{q}_2)) \end{bmatrix} \\ \frac{\partial G}{\partial \mathbf{q}} &= \begin{bmatrix} \frac{\partial G_1}{\partial q_1} & \frac{\partial G_1}{\partial q_2} \\ \frac{\partial G_2}{\partial q_1} & \frac{\partial G_2}{\partial q_2} \end{bmatrix} \\ &= \begin{bmatrix} -m_1 g a_1 \sin(\textcolor{green}{q}_1) - m_2 g (l_1 \sin(\textcolor{green}{q}_1) + a_2 \sin(\textcolor{green}{q}_1 + \textcolor{red}{q}_2)) & -m_2 g a_2 \sin(\textcolor{green}{q}_1 + \textcolor{red}{q}_2) \\ -m_2 g (l_1 \sin(\textcolor{green}{q}_1) + a_2 \sin(\textcolor{green}{q}_1 + \textcolor{red}{q}_2)) & -m_2 g a_2 \sin(\textcolor{green}{q}_1 + \textcolor{red}{q}_2) \end{bmatrix} \end{aligned}$$

Die Massenmatrix $M(\mathbf{q})$ ist gegeben durch den Ausdruck:

$$M(\mathbf{q}) = \begin{bmatrix} m_1 a_1^2 + I_{1,zz} + m_2 \cdot (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\mathbf{q}_2)) + I_{2,zz} & m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{q}_2)) + I_{2,zz} \\ m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{q}_2)) + I_{2,zz} & m_2 a_2^2 + I_{2,zz} \end{bmatrix}$$

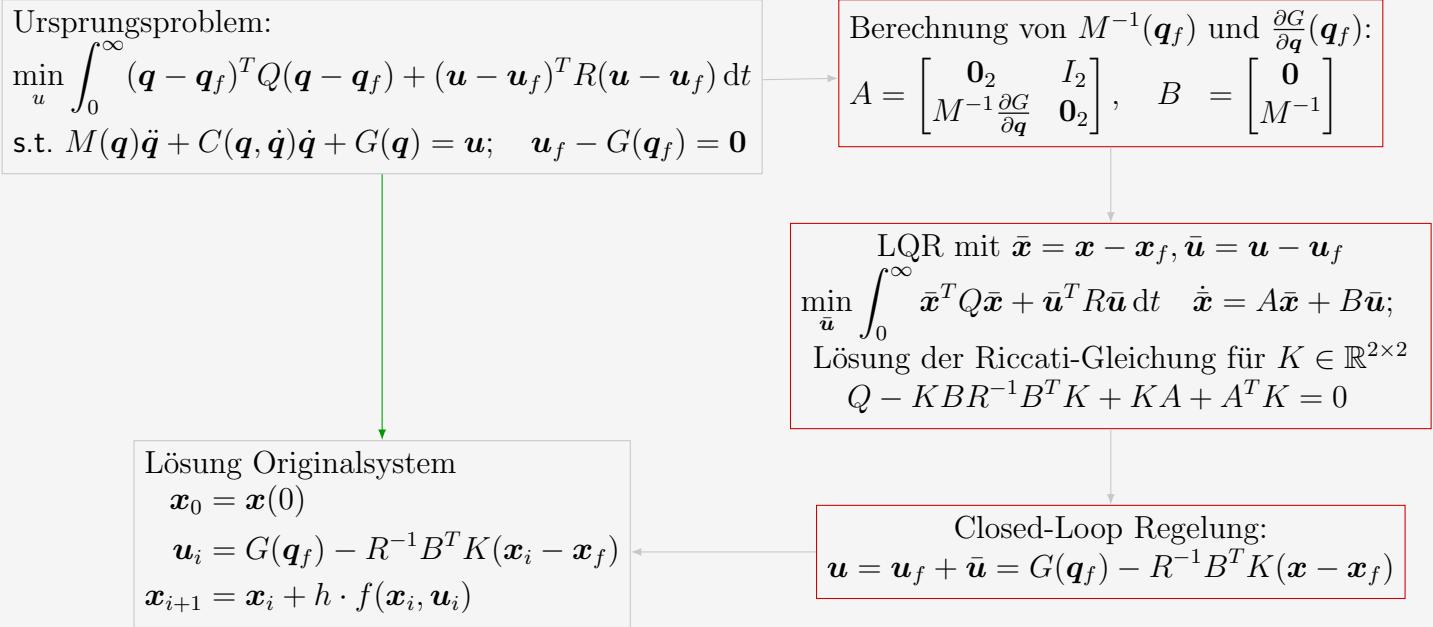
Für die Inverse der Massenmatrix M^{-1} gilt analytisch mittels $F_{\mathbf{q}_2} := m_2 l_1 a_2 \cos(\mathbf{q}_2)$

$$\begin{aligned} \det(M) &= M_{11} M_{22} - M_{12}^2 \\ &= [m_1 a_1^2 + I_{1,zz} + m_2 \cdot (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\mathbf{q}_2)) + I_{2,zz}] \cdot (m_2 a_2^2 + I_{2,zz}) \\ &\quad - (m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{q}_2)) + I_{2,zz})^2 \\ &= \left[m_1 a_1^2 + I_{1,zz} + I_{2,zz} + m_2 l_1^2 + m_2 a_2^2 + 2 \underbrace{m_2 l_1 a_2 \cos(\mathbf{q}_2)}_{F_{\mathbf{q}_2}} \right] (m_2 a_2^2 + I_{2,zz}) \\ &\quad - (m_2 a_2^2 + I_{2,zz})^2 - 2 \underbrace{m_2 l_1 a_2 \cos(\mathbf{q}_2)}_{F_{\mathbf{q}_2}} \cdot (m_2 a_2^2 + I_{2,zz}) - F_{\mathbf{q}_2}^2 \\ &= (m_2 a_2^2 + I_{2,zz}) \cdot \left[m_1 a_1^2 + I_{1,zz} + m_2 l_1^2 + \underbrace{I_{2,zz} + m_2 a_2^2 - m_2 a_2^2 - I_{2,zz}}_{=0} \right] - \underbrace{(m_2 l_1 a_2)^2 (1 - \sin(\mathbf{q}_2)^2)}_{F_{\mathbf{q}_2}^2} \\ &= (m_2 a_2^2 + I_{2,zz}) \cdot (m_1 a_1^2 + I_1 + m_2 l_1^2) - (m_2 l_1 a_2)^2 + (m_2 l_1 a_2)^2 \sin(\mathbf{q}_2)^2 \\ &= (m_2 a_2^2 + I_{2,zz}) \cdot (m_1 a_1^2 + I_1) + \underbrace{(m_2 a_2^2 + I_{2,zz}) \cdot m_2 l_1^2}_{m_2^2 l_1^2 a_2^2 + m_2 l_1^2 I_{2,zz}} - (m_2 l_1 a_2)^2 + (m_2 l_1 a_2)^2 \sin(\mathbf{q}_2)^2 \\ &= (m_1 a_1^2 + I_1 + m_2 l_1^2) \cdot (m_2 a_2^2 + I_2) + m_2 l_1^2 I_{2,zz} + m_2^2 l_1^2 a_2^2 \sin(\mathbf{q}_2)^2 \end{aligned}$$

Für die Inverse der Massenmatrix gilt somit

$$\begin{aligned} M^{-1} &= \frac{1}{\det(M)} \begin{pmatrix} M_{22} & -M_{12} \\ -M_{21} & M_{11} \end{pmatrix} \\ &= \frac{1}{\det(M)} \begin{pmatrix} m_2 a_2^2 + I_2 & -(m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{x}_2)) + I_2) \\ -(m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{x}_2)) + I_2) & m_1 a_1^2 + I_1 + m_2 \cdot (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\mathbf{x}_2)) + I_2 \end{pmatrix} \end{aligned}$$

$$\begin{aligned} M_{11} &= m_1 a_1^2 + I_1 + m_2 \cdot (l_1^2 + a_2^2 + 2l_1 a_2 \cos(\mathbf{x}_2)) + I_2 \\ M_{12} &= m_2 \cdot (a_2^2 + l_1 a_2 \cos(\mathbf{x}_2)) + I_2 \\ M_{21} &= M_{12} \\ M_{22} &= m_2 a_2^2 + I_2 \end{aligned}$$



3.2 Riccati Regler

Es gilt

$$F(K) := Q - KPK + KA + A^T K \Rightarrow F(K) = 0$$

Definition 3.2 Sei $X \in \mathbb{R}^{n \times n}$ dann definieren wir

$$\text{vec}(X) = (X_{11} \ X_{21} \ \dots \ X_{n1} \ X_{12} \ X_{22} \ \dots \ X_{nn})^T \in \mathbb{R}^{n^2}$$

Seien $A \in \mathbb{R}^{n \times m}, B \in \mathbb{R}^{p \times r}$, dann definieren wir

$$A \otimes B = \begin{pmatrix} A_{11}B & \dots & A_{1m}B \\ A_{12}B & \dots & A_{2m}B \\ \vdots & \ddots & \vdots \\ A_{n1}B & \dots & A_{nm}B \end{pmatrix} \in \mathbb{R}^{mp \times nr}$$

Es gilt die Formel

$$\text{vec}(MXN) = (N^T \otimes M)\text{vec}(X)$$

Betrachten wir den Fall $M = 1$ und $N = A$ bzw. $N = 1$ und $M = A^T$ erhalten wir mit $X = K$

$$\text{vec}(KA) = (A^T \otimes 1)\text{vec}(K) \quad \text{vec}(A^T K) = (1 \otimes A^T)\text{vec}(K)$$

Anwendung auf unser Problem liefert

$$\underbrace{\text{vec}(F(K))}_{f(x=\text{vec}(K))} = \underbrace{\text{vec}(Q)}_q - \underbrace{\text{vec}(KPK)}_{g(x=\text{vec}(K))} + \underbrace{\text{vec}(KA)}_{(A^T \otimes 1)\text{vec}(K)} + \underbrace{\text{vec}(A^T K)}_{(1 \otimes A^T)\text{vec}(K)}$$

Durch Einführung von $x = \text{vec}(K) \in \mathbb{R}^{n^2}$ erhalten wir

$$f(x) = q - g(x) + (A^T \otimes 1)x + (1 \otimes A^T)x$$

Da $(KP)^T = KP$ gilt

$$g(\text{vec}(K)) = \text{vec}(KP) = (1 \otimes KP)\text{vec}(K)$$

Wir betrachten die Variation $K(\epsilon) = K + \epsilon \cdot \bar{K}$ und die Funktion $h : \mathbb{R} \rightarrow \mathbb{R}^{n^2}$ mit

$$\begin{aligned} h(\epsilon) &:= g(\text{vec}(K(\epsilon))) = \text{vec}(K(\epsilon)PK(\epsilon)) \\ &= \text{vec}((K + \epsilon \cdot \bar{K})P(K + \epsilon \cdot \bar{K})) \\ &= \text{vec}(KP + \epsilon K P \bar{K} + \epsilon \bar{K} P K + \epsilon^2 \bar{K} P \bar{K}) \end{aligned}$$

Ableitung nach ϵ und Auswertung in $\epsilon = 0$ ergibt

$$h'(0) = \frac{d}{d\epsilon} g(\text{vec}(K + \epsilon \cdot \bar{K})) \Big|_{\epsilon=0} = \text{vec}(KP\bar{K} + \bar{K}PK) = \text{vec}(KP\bar{K}) + \text{vec}(\bar{K}PK)$$

Umschreiben als Matrix-Vektor-Produkt

$$\text{vec}(KP \cdot \bar{K} \cdot 1) = (1 \otimes KP)\text{vec}(\bar{K}) \quad \text{vec}(1 \cdot \bar{K} \cdot PK) = ((PK)^T \otimes 1)\text{vec}(\bar{K})$$

Wir erhalten als Ableitung

$$g'(x) = (1 \otimes KP) + ((PK)^T \otimes 1) \Rightarrow f'(x) = -(1 \otimes KP) - ((PK)^T \otimes 1) + (A^T \otimes 1) + (1 \otimes A^T)$$

3.2.1 Implementierung

```
def vec(X):
    return X.reshape(-1, order='F')

def unvec(v, n):
    return v.reshape((n, n), order='F')
```

Hilfsfunktionen für die Vektorisierung

```

# Newton Kleinman Verfahren
def solv_CARE(A,B,R,Q,tol=1e-8,max_iter=50):
    n = A.shape[0]
    I = np.eye(n)
    q = vec(Q)
    L = np.kron(I, A.T) + np.kron(A.T, I)
    P = B @ np.linalg.solve(R, B.T)
    # Startwert KO hier Einheitsmatrix
    K = I
    x = vec(K)

    for i in range(max_iter):
        X = K @ P #KP
        vec_KPK = np.kron(I,X) @ x
        f = q - vec_KPK + L @ x
        Dg = np.kron(I,X)+np.kron(X,I)
        Df = L - Dg
        dx = np.linalg.solve(Df, -f)
        x_new = x + dx
        # Abbruch
        if np.linalg.norm(dx) / np.linalg.norm(x_new) < tol:
            x=x_new
            break

    x = x_new
    K = unvec(x, n)
return K

```

Erklärung des Codes

Stabilität der Startlösung hier Gershgorin Kreise und Stabilität erklären
Für Y mit $RY = B^T \Rightarrow Y = B^{-1}B^T$
und $B \cdot Y = P$

4 Gradientenverfahren

Satz 4.1 (Gradientenverfahren mit Linesearch) Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}$ differenzierbar,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_j \cdot \nabla f(\mathbf{x}_k) \quad \min \left\{ j : \underbrace{f(\mathbf{x}_i - \alpha_j \nabla f(\mathbf{x}_i))}_{\varphi(\alpha_j)} \leq \underbrace{f(\mathbf{x}_i)}_{\varphi(0)} - \underbrace{\frac{\mathbf{c} \cdot \alpha_j \|\nabla f(\mathbf{x}_i)\|^2}{\mathbf{c} \cdot \varphi'(0) \cdot \alpha_j}}_{\mathbf{c} \cdot \varphi'(0) \cdot \alpha_j} \right\}$$

konvergiert für $k \rightarrow \infty$ gegen die Lösung von $\min_x f(x)$.

Ziel Berechnung des Gradienten $J'(\mathbf{u})$ für das Steuerungsproblem

$$\begin{aligned} J^{aux}(\mathbf{u}) &= \int_{t_0}^{t_f} L(t, \mathbf{x}, \mathbf{u}) + \mathbf{p}(t) \cdot \left[f(t, \mathbf{x}, \mathbf{u}) - \dot{\mathbf{x}}(t) \right] dt + \Phi(\mathbf{x}(t_f)) \\ &= \int_{t_0}^{t_f} L(t, \mathbf{x}, \mathbf{u}) + \mathbf{p}(t) \cdot f(t, \mathbf{x}, \mathbf{u}) dt - \int_{t_0}^{t_f} \mathbf{p}(t) \dot{\mathbf{x}}(t) dt + \Phi(\mathbf{x}(t_f)) \\ &= \int_{t_0}^{t_f} H[t] dt - \left[\mathbf{p}(t) \mathbf{x}(t) \right]_{t_0}^{t_f} + \int_{t_0}^{t_f} \dot{\mathbf{p}}(t) \mathbf{x}(t) dt + \Phi(\mathbf{x}(t_f)) \\ &= \int_{t_0}^{t_f} H[t] + \dot{\mathbf{p}}(t) \mathbf{x}(t) dt - \left[\mathbf{p}(t) \mathbf{x}(t) \right]_{t_0}^{t_f} + \Phi(\mathbf{x}(t_f)) \\ &= \int_{t_0}^{t_f} H[t] + \dot{\mathbf{p}}(t) \mathbf{x}(t) dt - \mathbf{p}(t_f) \mathbf{x}(t_f) + \mathbf{p}(t_0) \mathbf{x}(t_0) + \Phi(\mathbf{x}(t_f)) \end{aligned}$$

Der Term $\mathbf{p}(t_0) \mathbf{x}(t_0)$ ist unabhängig von \mathbf{u} und fällt bei der Optimierung weg. Bilden der Gateau Ableitung in \mathbf{u} entlang \mathbf{h} liefert

$$\begin{aligned} D J^{aux}(\mathbf{u}, \mathbf{h}) &= \int_{t_0}^{t_f} H_x[t] \cdot S(t) + H_u[t] \cdot h(t) + \dot{\mathbf{p}}(t) S(t) dt - \mathbf{p}(t_f) S(t_f) + \Phi_x(\mathbf{x}(t_f)) S(1) \\ &= \int_{t_0}^{t_f} (\underbrace{H_x[t] + \dot{\mathbf{p}}(t)}_{=0}) \cdot S(t) + H_u[t] \cdot h(t) dt + (\underbrace{\Phi_x(\mathbf{x}(t_f)) - \mathbf{p}(t_f)}_{=0}) \cdot S(t_f) \\ &= \int_0^T H_u[t] \cdot h(t) dt \end{aligned}$$

Hierbei bezeichnet $S(t)$ die Sensitivity function von \mathbf{x} in \mathbf{u} in Richtung \mathbf{h} , also

$$S(t) = \frac{\partial \mathbf{x}(t)}{\partial \mathbf{u}} \cdot \mathbf{h}$$

Wir erhalten somit $\nabla J(\mathbf{u}) = H_u[t]$

4.1 Gradientenverfahren für den two link revolute Manipulator

Wir betrachten das Steuerungsproblem

$$\begin{aligned} \min_{\mathbf{u}} \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}) dt + \Phi(\mathbf{x}(t_f)) \\ L(\mathbf{x}, \mathbf{u}) = (\mathbf{u} - \mathbf{u}_f)^T \cdot R \cdot (\mathbf{u} - \mathbf{u}_f) \\ \Phi(\mathbf{x}(t_f)) = (\mathbf{q}(t_f) - \mathbf{q}_f)^T \cdot Q \cdot (\mathbf{q}(t_f) - \mathbf{q}_f) \\ \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{\mathbf{q}} \\ M^{-1}(\mathbf{q}) \cdot (-C(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} - G(\mathbf{q}) + \mathbf{u}) \end{bmatrix} \\ \mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \quad \mathbf{x}(t_0) = \begin{bmatrix} \mathbf{q}(t_0) \\ \dot{\mathbf{q}}(t_0) \end{bmatrix} \end{aligned}$$

Für die adjungierte Gleichung gilt

$$\dot{\mathbf{p}} = -H_{\mathbf{x}} = -\begin{bmatrix} \frac{\partial H}{\partial \mathbf{q}} \\ \frac{\partial H}{\partial \dot{\mathbf{q}}} \end{bmatrix}, \quad \mathbf{p}(t_f) = \Phi_{\mathbf{x}}(\mathbf{x}(t_f)) = \begin{bmatrix} 2Q(\mathbf{q}(t_f) - \mathbf{q}_f) \\ \mathbf{0} \end{bmatrix} \quad H[t] = L(\mathbf{u}) + \mathbf{p}^T \cdot f(\mathbf{x}, \mathbf{u})$$

Für den Gradienten gilt

$$\nabla J(\mathbf{u}) = \frac{\partial H}{\partial \mathbf{u}} = 2R \cdot (\mathbf{u} - \mathbf{u}_f) + \frac{\partial f^T}{\partial \mathbf{u}} \cdot \mathbf{p}$$

Wir nutzen die spezielle Struktur unseres Problems

$$\frac{\partial f}{\partial \mathbf{u}} = \begin{bmatrix} \mathbf{0}_2 \\ M^{-1}(\mathbf{q}) \end{bmatrix} \Rightarrow \frac{\partial f^T}{\partial \mathbf{u}} \cdot \begin{bmatrix} \mathbf{p}_{\mathbf{q}} \\ \mathbf{p}_{\dot{\mathbf{q}}} \end{bmatrix} = M^{-1}(\mathbf{q}) \cdot \mathbf{p}_{\dot{\mathbf{q}}} \Rightarrow \nabla J(\mathbf{u}) = 2R \cdot (\mathbf{u} - \mathbf{u}_f) + M^{-1}(\mathbf{q}) \cdot \mathbf{p}_{\dot{\mathbf{q}}}$$

1) Forward Integration der State Equation: $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$

$$\begin{aligned} \mathbf{x}_0 &= \mathbf{x}(t_0) \\ \mathbf{x}_{i+1} &= \mathbf{x}_i + h \cdot \begin{bmatrix} \dot{\mathbf{q}}_i \\ M^{-1}(\mathbf{q}_i) \cdot (-C(\mathbf{q}_i, \dot{\mathbf{q}}_i) \cdot \dot{\mathbf{q}}_i - G(\mathbf{q}_i) + \mathbf{u}_i) \end{bmatrix} \quad i = 0, \dots, N-1 \end{aligned}$$

Abspeichern Array $\mathbf{X} = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N]$

2) Backward Integration der Adjungierten Gleichung: $\dot{\mathbf{p}} = -H_{\mathbf{x}} = -\frac{\partial f}{\partial \mathbf{x}}$

$$\begin{aligned} \mathbf{p}_N &= \mathbf{p}(t_f) = \begin{bmatrix} 2Q(\mathbf{q}(t_f) - \mathbf{q}_f) \\ \mathbf{0} \end{bmatrix} \\ \mathbf{p}_{i-1} &= \mathbf{p}_i + h \cdot \begin{bmatrix} \mathbf{0}_2 & I_2 \\ \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} & \frac{\partial \dot{\mathbf{q}}}{\partial \dot{\mathbf{q}}} \end{bmatrix} \cdot \mathbf{p}_i \\ \frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{q}} &= \frac{\partial}{\partial \mathbf{q}} (M^{-1}(\mathbf{q}_i) \cdot (-C(\mathbf{q}_i, \dot{\mathbf{q}}_i) \cdot \dot{\mathbf{q}}_i - G(\mathbf{q}_i) + \mathbf{u}_i)) \\ \frac{\partial \ddot{\mathbf{q}}}{\partial \dot{\mathbf{q}}} &= \frac{\partial}{\partial \dot{\mathbf{q}}} (M^{-1}(\mathbf{q}_i) \cdot (-C(\mathbf{q}_i, \dot{\mathbf{q}}_i) \cdot \dot{\mathbf{q}}_i - G(\mathbf{q}_i) + \mathbf{u}_i)) \end{aligned}$$

2.1) Finite Differenzen für $\frac{\partial \ddot{\mathbf{q}}}{\partial x_j}$ $j = 1, 2, 3, 4$

$$\begin{aligned} \mathbf{x}_{plus} &= \mathbf{x} + \epsilon \cdot \mathbf{e}_j & \mathbf{x}_{minus} &= \mathbf{x} - \epsilon \cdot \mathbf{e}_j \\ \ddot{\mathbf{q}}(\mathbf{x}) &= M^{-1}(\mathbf{q}) \cdot (-C(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} - G(\mathbf{q}) + \mathbf{u}) \\ \frac{\partial \ddot{\mathbf{q}}}{\partial x_j} &\approx \frac{\ddot{\mathbf{q}}(\mathbf{x}_{plus}) - \ddot{\mathbf{q}}(\mathbf{x}_{minus})}{2\epsilon} \end{aligned}$$

Abspeichern Array $\mathbf{P} = [\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_N]$

3) Gradientenverfahren $\mathbf{u}_{i+1} = \mathbf{u}_i - \alpha_j \cdot \nabla J(\mathbf{u}_i)$

$$\mathbf{u}_{i+1} = \mathbf{u}_i - \alpha_j \cdot [2R \cdot (\mathbf{u}_i - \mathbf{u}_f) + M^{-1}(\mathbf{q}_i) \cdot \mathbf{P}_{\dot{\mathbf{q}},i}] \quad i = 0, \dots, N-1$$

3.1) Berechnung der Schrittweite α_j über Linesearch

$$\begin{aligned} \alpha_{test} &= \beta^m \cdot \alpha_0 \\ \mathbf{u}_{test} &= \mathbf{u}_k - \alpha_{test} \cdot \nabla J(\mathbf{u}_k) \\ J_{test} &= J(\mathbf{u}_{test}) \\ \text{IF } J_{test} &\leq J(\mathbf{u}_k) - c \cdot \alpha_{test} \cdot \|\nabla J(\mathbf{u}_k)\|^2 : \\ \alpha_k &= \alpha_{test} \\ \Rightarrow &\text{Break} \end{aligned}$$

5 Foundations of Reinforcement Learning

Wir folgen der Notation aus [3].

Definition 5.1 Wir definieren für den Markov decision process (MDP) mit Zustandsraum \mathcal{S} , Aktionsraum \mathcal{A} , Übergangswahrscheinlichkeiten $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$, Belohnungsfunktion $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ und Diskontfaktor $\gamma \in [0, 1)$ die optimale Value-Funktion als

Definition 5.2 (Value-Funktion) Wir definieren die Value-Funktion für eine Policy π als

$$V^\pi(\mathbf{s}) = \mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+1} \mid \mathbf{s}_t = \mathbf{s} \right]$$

$$\begin{aligned} V^\pi(\mathbf{s}) &= \mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+1} \mid \mathbf{s}_t = \mathbf{s} \right] \\ &= \mathbb{E}[r_{t+1} \mid \mathbf{s}_t = \mathbf{s}] + \gamma \cdot \mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+2} \mid \mathbf{s}_t = \mathbf{s} \right] \\ &= \mathbb{E}[r_{t+1} \mid \mathbf{s}_t = \mathbf{s}] + \gamma \cdot \mathbb{E} \left[\underbrace{\mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+2} \mid \mathbf{s}_{t+1} \right]}_{V^\pi(\mathbf{s}_{t+1})} \mid \mathbf{s}_t = \mathbf{s} \right] \\ \Rightarrow V^\pi(\mathbf{s}) &= \mathbb{E}[r_{t+1} + \gamma \cdot V^\pi(\mathbf{s}_{t+1}) \mid \mathbf{s}_t = \mathbf{s}] \end{aligned}$$

Denn es gilt hier für die σ -Algebren, dass $\sigma(\mathbf{s}_t) \subseteq \sigma(\mathbf{s}_{t+1})$ gilt. Explizit ausgeschrieben in Erwartungswerten ergibt sich:

$$\begin{aligned} V^\pi(\mathbf{s}) &= \mathbb{E}[r_{t+1} + \gamma V^\pi(\mathbf{s}_{t+1}) \mid \mathbf{s}_t = \mathbf{s}] \\ &= \mathbb{E}_{a_t \sim \pi(\cdot | \mathbf{s})} \left[\mathbb{E}[r_{t+1} + \gamma V^\pi(\mathbf{s}_{t+1}) \mid \mathbf{s}_t = \mathbf{s}, a_t = a] \right] \\ &= \mathbb{E}_{a_t \sim \pi(\cdot | \mathbf{s})} \left[\mathbb{E}_{\mathbf{s}_{t+1} \sim T_a(\mathbf{s})} [R_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1}) + \gamma V^\pi(\mathbf{s}_{t+1})] \right] \\ &= \mathbb{E}_{a \sim \pi(\cdot | \mathbf{s})} \left[\mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} [R_a(\mathbf{s}, \mathbf{s}') + \gamma V^\pi(\mathbf{s}')] \right] \end{aligned}$$

Definition 5.3 (Q-Function) Wir definieren die Q-Function als

$$Q^\pi(\mathbf{s}, \mathbf{a}) = \mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+1} \mid \mathbf{s}_t = \mathbf{s}, \mathbf{a}_t = \mathbf{a} \right] \Rightarrow V^\pi(\mathbf{s}) = \mathbb{E}_{\mathbf{a} \sim \pi(\cdot | \mathbf{s})} [Q^\pi(\mathbf{s}, \mathbf{a})]$$

Herleitung der Bellmann Gleichung für die Q-Funktion:

$$\begin{aligned}
Q^\pi(\mathbf{s}, a) &= \mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+1} \mid \mathbf{s}_t = \mathbf{s}, a_t = a \right] \\
&= \mathbb{E} \left[\mathbb{E} \left[r_{t+1} + \sum_{i=1}^{\infty} \gamma^i \cdot r_{t+i+1} \mid \mathbf{s}_t = \mathbf{s}, a_t = a, \mathbf{s}_{t+1} = \mathbf{s}' \right] \mid \mathbf{s}_t = \mathbf{s}, a_t = a \right] \\
&= \mathbb{E} \left[r_{t+1} + \gamma \cdot \underbrace{\mathbb{E} \left[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i+2} \mid \mathbf{s}_{t+1} = \mathbf{s}' \right]}_{V^\pi(\mathbf{s}_{t+1})} \mid \mathbf{s}_t = \mathbf{s}, a_t = a \right] \\
&= \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^\pi(\mathbf{s}') \right] \\
&= \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \mathbb{E}_{a' \sim \pi(\cdot | \mathbf{s}')} \left[Q^\pi(\mathbf{s}', a') \right] \right]
\end{aligned}$$

Wobei wir im letzten Schritt die folgende Beziehung genutzt haben:

$$V^\pi(\mathbf{s}) = \mathbb{E}_{a \sim \pi(\cdot | \mathbf{s})} [Q^\pi(\mathbf{s}, a)]$$

5.1 Bellman Optimality Equations

Satz 5.4 (Optimality Equation für V) Die optimale Value-Funktion $V^*(\mathbf{s})$ erfüllt:

$$V^*(\mathbf{s}) = \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}') \right] \quad \pi^*(\mathbf{s}) = \arg \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}') \right]$$

Beweis: Wir setzen $V^*(\mathbf{s}) = \max_\pi V^\pi(\mathbf{s})$. Für jede Policy π gilt nun $V^*(\mathbf{s}') \geq V^\pi(\mathbf{s}')$ und somit:

$$V^*(\mathbf{s}) = \max_\pi \mathbb{E}_{a \sim \pi(\cdot | \mathbf{s})} \left[\mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} [R_a(\mathbf{s}, \mathbf{s}') + \gamma V^\pi(\mathbf{s}')] \right] \leq \max_\pi \mathbb{E}_{a \sim \pi(\cdot | \mathbf{s})} \left[\mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} [R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}')] \right]$$

Das Maximum über alle Policies π wird erreicht durch die Policy, die in jedem Zustand \mathbf{s} die Aktion a auswählt, die das Maximum in der inneren Erwartung erreicht. Somit gilt:

$$V^*(\mathbf{s}) \leq \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}') \right]$$

Wir konstruieren nun die greedy Policy π^* mittels:

$$\pi^*(\mathbf{s}) = \arg \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}') \right]$$

Für diese Policy π^* gilt nun:

$$V^{\pi^*}(\mathbf{s}) = \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V^*(\mathbf{s}') \right] \geq V^*(\mathbf{s}) \Rightarrow V^{\pi^*}(\mathbf{s}) = V^*(\mathbf{s})$$

□

Satz 5.5 (Optimality Equation für Q-Funktion) Die optimale Q-Funktion $Q^*(\mathbf{s}, a)$ erfüllt:

$$Q^*(\mathbf{s}, a) = \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} Q^*(\mathbf{s}', a') \right] \quad \pi^*(\mathbf{s}) = \arg \max_a Q^*(\mathbf{s}, a)$$

Beweis: Für die optimale Q-Funktion gilt somit

$$Q^*(\mathbf{s}, a) = \max_{\pi} Q^\pi(\mathbf{s}, a) \Rightarrow Q^*(\mathbf{s}, a) = \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} Q^*(\mathbf{s}', a') \right]$$

Für die optimale Policy gilt:

$$\pi^*(\mathbf{s}) = a^* = \arg \max_a Q^*(\mathbf{s}, a)$$

□

5.2 Value Iteration

Wir definieren die Abbildung \mathcal{T} als

$$\mathcal{T}[V](\mathbf{s}, a) := \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V(\mathbf{s}', a') \right]$$

Die Bellmann Optimalitätsgleichung 5.4 sagt nun aus, dass V^* der Fixpunkt der Abbildung \mathcal{T} ist, also $\mathcal{T}[V^*] = V^*$ gilt. Die Fixpunktiteration $V_{k+1} = \mathcal{T}[V_k]$ konvergiert somit gegen V^* .

Beweis: Wir zeigen zunächst, dass \mathcal{T} eine Kontraktion ist:

$$\begin{aligned} |\mathcal{T}[V] - \mathcal{T}[U]| &= \left| \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V(\mathbf{s}') \right] - \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma U(\mathbf{s}') \right] \right| \\ &\leq \max_a \left| \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma V(\mathbf{s}') \right] - \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma U(\mathbf{s}') \right] \right| \\ &= \max_a \left| \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[\gamma(V(\mathbf{s}') - U(\mathbf{s}')) \right] \right| \\ &\leq \max_a \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[\underbrace{\gamma \cdot \sup_{\mathbf{s}'} |V(\mathbf{s}') - U(\mathbf{s}')|}_{\|V-U\|_\infty} \right] \\ &= \gamma \|V - U\|_\infty \end{aligned}$$

Nach dem Fixpunktsatz von Banach [6] konvergiert die Iteration $V_{k+1} = \mathcal{T}(V_k)$ gegen V^* für $k \rightarrow \infty$. □

Algorithm 1 Value Iteration

```

Initialize  $V_0(s)$  arbitrarily for all  $s \in \mathcal{S}$ 
 $k \leftarrow 0$ 
repeat
  for each  $s \in \mathcal{S}$  do
    
$$V_{k+1}(s) \leftarrow \max_a \mathbb{E}_{s' \sim T_a(s)} \underbrace{\left[ R_a(s, s') + \gamma V_k(s') \right]}_{Q_k(s, a)} = \max_a \sum_{s'} T_a(s, s') [R_a(s, s') + \gamma V_k(s')]$$

  end for
   $k \leftarrow k + 1$ 
until  $|V_k - V_{k-1}| < \varepsilon$ 
return  $V_k$ 

```

5.3 Q-Learning

Wir betrachten die Optimalitätsgleichung für die Q-Funktion 5.5:

$$Q^*(\mathbf{s}, a) = \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} Q^*(\mathbf{s}', a') \right]$$

und definieren erneut den Operator \mathcal{T} als

$$\mathcal{T}[Q](\mathbf{s}, a) := \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} Q(\mathbf{s}', a') \right]$$

und erhalten erneut die Fixpunktgleichung $\mathcal{T}[Q^*] = Q^*$. Die Fixpunktiteration $Q_{k+1} = \mathcal{T}[Q_k]$ konvergiert somit gegen Q^* , allerdings ist die Berechnung der Erwartungswerte nicht möglich, da wir das Modell $\mathbf{s}' = T_a(\mathbf{s})$ nicht kennen.

Algorithm 2 Q-Learning

Given Environment, learning rate $\alpha \in (0, 1]$, discount factor $\gamma \in [0, 1)$, exploration rate $\epsilon \in (0, 1)$
 Initialize $Q(s, a)$ arbitrarily for all $s \in \mathcal{S}, a \in \mathcal{A}$
 $Q(\text{terminal}, \cdot) = 0$ ▷ Terminale Zustände haben Wert 0
for each episode **do**
 Initialize state s
while state s is not terminal **do**
 Choose action $a = \epsilon\text{-greedy}(Q[s, \cdot], \epsilon)$ ▷ Exploration vs. Exploitation
 $(s', r) = \text{Environment}(s, a)$
 $Q[s, a] \leftarrow (1 - \alpha) \cdot Q[s, a] + \alpha \cdot (r + \gamma \cdot \max_{a'} Q[s', a'])$ ▷ Bootstrapping und TD-Learning
 $s \leftarrow s'$
end while
end for

Beweis: Wir prüfen die Voraussetzungen für die Konvergenz von Q-Learning aus [7] und folgern die Konvergenz gegen Q^* fast sicher.

- Der Beweis, dass \mathcal{T} eine Kontraktion ist, verläuft analog zum Beweis für die Value-Funktion.

$$\begin{aligned}
|\mathcal{T}[Q] - \mathcal{T}[U]| &= \left| \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} Q(\mathbf{s}', a') \right] - \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \max_{a'} U(\mathbf{s}', a') \right] \right| \\
&= \left| \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[\gamma \left(\max_{a'} Q(\mathbf{s}', a') - \max_{a'} U(\mathbf{s}', a') \right) \right] \right| \\
&\leq \gamma \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[\underbrace{\gamma \cdot \max_{a'} |Q(\mathbf{s}', a') - U(\mathbf{s}', a')|}_{\leq \|Q-U\|_\infty} \right] \\
&\leq \gamma \|Q - U\|_\infty
\end{aligned}$$

- Wir verwenden folgende Update-Regel, um die Q-Funktion zu approximieren:

$$\begin{aligned}
Q_{k+1}(\mathbf{s}, a) &= Q_k(\mathbf{s}, a) + \alpha \cdot \left[\underbrace{R_a(\mathbf{s}, \mathbf{s}') + \gamma \cdot \max_{a'} Q_k(\mathbf{s}', a')}_{\text{Sample of } T[Q_k]} - Q_k(\mathbf{s}, a) \right] \\
&= Q_k(\mathbf{s}, a) + \alpha \cdot \left[\mathcal{T}[Q_k](\mathbf{s}, a) - Q_k(\mathbf{s}, a) + \underbrace{R_a(\mathbf{s}, \mathbf{s}') + \gamma \cdot \max_{a'} Q_k(\mathbf{s}', a')}_{\eta_k} - \mathcal{T}[Q_k](\mathbf{s}, a) \right] \\
&= Q_k(\mathbf{s}, a) + \alpha \cdot \left[\mathcal{T}[Q_k](\mathbf{s}, a) - Q_k(\mathbf{s}, a) + \eta_k \right]
\end{aligned}$$

Für den Erwartungswert des Rauschterms η_k gilt:

$$\mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} [\eta_k \mid Q_k] = \mathbb{E}_{\mathbf{s}' \sim T_a(\mathbf{s})} \left[R_a(\mathbf{s}, \mathbf{s}') + \gamma \cdot \max_{a'} Q_k(\mathbf{s}', a') - \underbrace{\mathcal{T}[Q_k](\mathbf{s}, a)}_{\text{deterministic}} \mid Q_k \right] = 0$$

- Wir definieren die ϵ -Greedy-Policy:

$$\pi_\epsilon(a, \mathbf{s}) := \begin{cases} 1 - \epsilon + \frac{\epsilon}{|\mathcal{A}|} & \text{if } a = \arg \max_{a'} Q[\mathbf{s}, a'] \\ \frac{\epsilon}{|\mathcal{A}|} & \text{else} \end{cases} = (1 - \epsilon) \cdot \mathbb{1}_{a=\arg \max_{a'} Q[\mathbf{s}, a']} + \frac{\epsilon}{|\mathcal{A}|}$$

□

5.4 SARSA

Algorithm 3 SARSA

Given Environment, learning rate $\alpha \in (0, 1]$, discount factor $\gamma \in [0, 1)$, exploration rate $\epsilon \in (0, 1)$

$$Q(Terminal) = 0$$

for each episode **do**

 Initialize state s

 Choose action $a = \epsilon\text{-greedy}(Q[s, \cdot], \epsilon)$

while state s is not terminal **do**

$(s', r) = Environment(s, a)$

$Q[s, a] = (1 - \alpha) \cdot Q[s, a] + \alpha \cdot (r + \gamma \cdot Q[s', a'])$

$s = s'$

 Choose action $a' = \epsilon\text{-greedy}(Q[s, \cdot], \epsilon)$

end while

end for

6 Reinforcement Learning in Optimal Control

Wir folgen den Notationen aus [3]. Für eine gegebene Trajektorie τ des MDPs gilt:

$$\tau_t^n := \{\mathbf{s}_t, a_t, r_t, \mathbf{s}_{t+1}, \dots, \mathbf{s}_{t+n}, a_{t+n}, r_{t+n}, \mathbf{s}_{t+n+1}\} = \left\{ \mathbf{s}_{t+i}, a_{t+i}, r_{t+i}, \mathbf{s}_{t+i+1} \right\}_{i=0, \dots, n}$$

In einem unendlichen Zeithorizont notieren wir $\tau_t^\infty = \tau_t$. Für die Verteilung der Trajektorien gilt:

$$p(\tau_0 \mid \pi) = p(\mathbf{s}_0) \cdot \prod_{t=0}^{\infty} \pi(a_t \mid \mathbf{s}_t) \cdot T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1})$$

Für das Zielfunktional definieren wir:

Definition 6.1 (Objective Function) Wir definieren die Objective Function als

$$J(\pi) := V^\pi(\mathbf{s}_0) = \mathbb{E}_{\tau_0 \sim p(\tau_0 \mid \pi)} \left[R(\tau_0) \right] \Rightarrow \pi^*(a \mid \mathbf{s}) = \arg \max_{\pi} V^\pi(\mathbf{s}_0)$$

6.1 Policy-Based Algorithms

Wir betrachten nun nachfolgend eine Familie von parametrisierten Policies $\pi_\theta(a \mid \mathbf{s})$ mit den Parametern $\theta \in \mathbb{R}^d$.

	Unendlichdimensionales Problem	Parametrisiertes Problem
Policy	$\pi : S \rightarrow p(A)$	$\theta \rightarrow \pi_\theta \text{ mit } \pi_\theta(a \mid \mathbf{s})$
$J(\pi)$	$J(\pi) = \mathbb{E}_{\tau \sim p(\tau \mid \pi)} \left[R(\tau) \right]$	$J(\theta) = \mathbb{E}_{\tau \sim p(\tau \mid \theta)} \left[R(\tau) \right]$
$p(\tau_0)$	$p(\tau_0 \mid \pi) = p(\mathbf{s}_0) \cdot \prod_{t=0}^{T-1} \pi(a_t \mid \mathbf{s}_t) \cdot T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1})$	$p(\tau_0 \mid \theta) = p(\mathbf{s}_0) \cdot \prod_{t=0}^{T-1} \pi_\theta(a_t \mid \mathbf{s}_t) \cdot T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1})$
∇J	$\nabla J(\pi) =$	$\nabla J(\theta) = \mathbb{E}_{\tau \sim p(\tau \mid \theta)} \left[R(\tau) \cdot \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t \mid \mathbf{s}_t) \right]$

Für die Gradienten der Objective Function gilt:

Satz 6.2 (Policy Gradient Theorem) Für die Gradienten der Objective Function gilt:

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{\tau \sim p(\tau \mid \theta)} [R(\tau)] = \mathbb{E}_{\tau \sim p(\tau \mid \theta)} [\nabla_\theta R(\tau)]$$

Beweis:

- Direkte Ableitung der Objective Function:

$$\begin{aligned} \nabla_\theta J(\theta) &= \nabla_\theta \int R(\tau) \cdot p(\tau \mid \theta) d\tau \\ &= \int R(\tau) \cdot \nabla_\theta p(\tau \mid \theta) d\tau \\ &= \int R(\tau) \cdot p(\tau \mid \theta) \cdot \frac{\nabla_\theta p(\tau \mid \theta)}{p(\tau \mid \theta)} d\tau \\ &= \int R(\tau) \cdot p(\tau \mid \theta) \cdot \nabla_\theta \log p(\tau \mid \theta) d\tau \\ &= \mathbb{E}_{\tau \sim p(\tau \mid \theta)} [R(\tau) \cdot \nabla_\theta \log p(\tau \mid \theta)] \end{aligned}$$

- Wir berechnen nun $\nabla_\theta \log p(\tau|\theta)$:

$$\begin{aligned}
\nabla_\theta \log p(\tau|\theta) &= \nabla_\theta \log \left[p(\mathbf{s}_0) \cdot \prod_{t=0}^{T-1} \pi_\theta(a_t|\mathbf{s}_t) \cdot T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1}) \right] \\
&= \nabla_\theta \left[\log p(\mathbf{s}_0) + \sum_{t=0}^{T-1} \log \pi_\theta(a_t|\mathbf{s}_t) + \log T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1}) \right] \\
&= \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|\mathbf{s}_t)
\end{aligned}$$

- Wir erhalten somit:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p(\tau|\theta)} [R(\tau) \cdot \nabla_\theta \log p(\tau|\theta)] = \mathbb{E}_{\tau \sim p(\tau|\theta)} \left[R(\tau) \cdot \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|\mathbf{s}_t) \right]$$

□

$$\begin{aligned}
R &\leftarrow \sum_{k=t}^{T-1} \gamma^{k-t} r_k \\
\theta &\leftarrow \theta + \alpha \cdot R \cdot \nabla_\theta \log \pi_\theta(a_t|\mathbf{s}_t)
\end{aligned}$$

Beweis: Es gilt $\pi_\epsilon = \pi + \epsilon \cdot h$

$$\begin{aligned}
\frac{d}{d\epsilon} J(\pi_\epsilon) |_{\epsilon=0} &= \frac{d}{d\epsilon} \mathbb{E}_{\tau \sim p(\tau|\pi_\epsilon)} [R(\tau)] |_{\epsilon=0} \\
&= \frac{d}{d\epsilon} \int R(\tau) \cdot p(\tau|\pi_\epsilon) d\tau |_{\epsilon=0} \\
&= \int R(\tau) \cdot \underbrace{\frac{d}{d\epsilon} p(\tau|\pi_\epsilon)}_{p(\tau|\pi_\epsilon) \cdot \frac{d}{d\epsilon} \log p(\tau|\pi_\epsilon)} d\tau |_{\epsilon=0} \\
&= \int R(\tau) \cdot p(\tau|\pi) \cdot \frac{d}{d\epsilon} \log p(\tau|\pi_\epsilon) |_{\epsilon=0} d\tau
\end{aligned}$$

Wir berechnen nun $\frac{d}{d\epsilon} \log p(\tau|\pi_\epsilon)$:

$$\begin{aligned}
\frac{d}{d\epsilon} \log p(\tau|\pi_\epsilon) &= \frac{d}{d\epsilon} \log \left[p(\mathbf{s}_0) \cdot \prod_{t=0}^{T-1} \pi_\epsilon(a_t|\mathbf{s}_t) \cdot T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1}) \right] \\
&= \frac{d}{d\epsilon} \left[\log p(\mathbf{s}_0) + \sum_{t=0}^{T-1} \log \pi_\epsilon(a_t|\mathbf{s}_t) + \log T_{a_t}(\mathbf{s}_t, \mathbf{s}_{t+1}) \right] \\
&= \sum_{t=0}^{T-1} \frac{d}{d\epsilon} \log (\pi(a_t|\mathbf{s}_t) + \epsilon \cdot h(a_t|\mathbf{s}_t)) \\
&= \sum_{t=0}^{T-1} \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t) + \epsilon \cdot h(a_t|\mathbf{s}_t)} \stackrel{\epsilon=0}{=} \sum_{t=0}^{T-1} \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)}
\end{aligned}$$

Wir erhalten somit:

$$\begin{aligned}
\frac{d}{d\epsilon} J(\pi_\epsilon) |_{\epsilon=0} &= \int R(\tau) \cdot p(\tau|\pi) \cdot \sum_{t=0}^{T-1} \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)} d\tau \\
&= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \cdot \sum_{t=0}^{T-1} \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)} \right] \\
&= \sum_{t=0}^{T-1} \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \cdot \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)} \right]
\end{aligned}$$

Für jeden festen Zeitschritt t gilt somit:

$$\begin{aligned}
\mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \cdot \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)} \right] &= \mathbb{E}_{\mathbf{s}_t, a_t} \left[\mathbb{E}_{\tau \sim p(\tau|\mathbf{s}_t, a_t)} \left[R(\tau) \cdot \frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\mathbf{s}_t)} \right] \right] \\
&= \mathbb{E}_{\mathbf{s}_t, a_t} \left[\frac{h(a_t|\mathbf{s}_t)}{\pi(a_t|\pi, \mathbf{s}_t)} \cdot \underbrace{\mathbb{E}_{\tau \sim p(\tau|\pi)} [R(\tau) | \mathbf{s}_t, a_t]}_{=Q^\pi(\mathbf{s}_t, a_t)} \right]
\end{aligned}$$

□

6.2 Deep Deterministic Policy Gradient

6.3 Two-Link-Revolute Manipulator

Literatur

- [1] Matthias Gerdts. *Optimal Control of ODEs and DAEs*. Berlin: De Gruyter, 2012.
- [2] Amit Kumar, Shrey Kasera und L. B. Prasad. “Optimal Control of 2-Link Underactuated Robot Manipulator”. In: *2017 International Conference on Innovations in Information Embedded and Communication Systems (ICIIECS)*. Gorakhpur, India, 2017.
- [3] Aske Plaat. *Deep Reinforcement Learning*. Springer, 2022. ISBN: 978-981-19-0637-4.
- [4] Bruno Siciliano u. a. *Robotics: Modelling, Planning and Control*. London: Springer, 2010. ISBN: 978-1849964507.
- [5] Mark W. Spong, Seth Hutchinson und M. Vidyasagar. *Robot Dynamics and Control*. 2nd. Hoboken, NJ: Wiley, 2004. ISBN: 978-0471649908.
- [6] Richard S. Sutton und Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.
- [7] John N. Tsitsiklis. “Asynchronous Stochastic Approximation and Q-Learning”. In: *Machine Learning* (1992).