Optimale Steuerung und Regelung

3. Übung, ausgegeben von: Prof. Frank Woittennek
zuletzt geändert am: 5. Juni 2023

Der Lösungsweg sowie der Python-Code sind am Ende der Übungseinheit zur Bewertung abzugeben. Der Lösungsweg soll mit einem geeignetes Textsatzsystem wie LibreOffice Writer oder Latex dokumentiert werden und ist als PDF-Datei abzugeben. Sie können den Lösungsweg auch zusammen mit dem Code in Form eines Jupyter-Notebooks dokumentieren und als HTML-Datei + IPYNB-Datei abgeben. Während der Übung haben Sie die Möglichkeit, die verbleibenden Fragen und Probleme zu klären bzw. zu lösen und Ihre Dokumentation entsprechend anzupassen.

Aufgabe 1

Es wird das Wagen-Pendel-System aus Abb. 1 betrachtet, vergleiche [Ste20, Übung 4]. Die

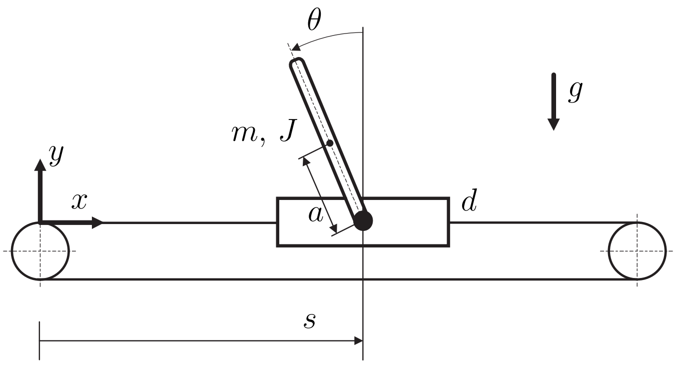

Abb. 1 – Wagen-Pendel-System (Abbildung aus [Ste20])

Zustandsdarstellung für den Zustand $\boldsymbol{x}^T = (\theta, \omega, s, v)$ lautet

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), u(t)) = \begin{pmatrix} \omega(t) \\ \frac{m\,g\,a\,\sin\theta(t) - d\,\omega(t) + m\,a\,u(t)\,\cos\theta(t)}{J + m\,a^2} \\ v(t) \\ u(t) \end{pmatrix}, \tag{1}$$

mit der Wagenposition s, dem Pendelwinkel θ, dem Massenträgheitsmoment $J = 0.0361\,\mathrm{kg\,m^2}$, dem Schwerpunktabstand $a = 0.42\,\mathrm{m}$ und der Masse $m = 0.3553\,\mathrm{kg}$ des Pendelstabes sowie der Erdbeschleunigung $g = 9.81\,\mathrm{m\,m^{-2}}$ und dem Reibkoeffizienten $d = 0.005\,\mathrm{N\,m\,s}$. Die Stellgröße u ist durch die Beschleunigung des Wagens $u = \ddot{s} = \dot{v}$ gegeben.

Für dieses System soll das Optimalsteuerungsproblem

$$\min_{u(\cdot)} \quad J(u(\cdot)) = \varphi(\boldsymbol{x}(t_1)) + \int_{t_0}^{t_1} l(\boldsymbol{x}(t), u(t))\, dt \tag{2a}$$

$$\text{u.B.v.} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, u), \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{\psi}(\boldsymbol{x}(t_1)) = 0 \tag{2b}$$

mit den Endkosten

$$\varphi(\boldsymbol{x}(t_1)) = \frac{1}{2}\boldsymbol{x}^T(t_1)\,\boldsymbol{S}\,\boldsymbol{x}(t_1), \qquad \boldsymbol{S} \geq 0,$$

den integralen Kosten

$$l(\boldsymbol{x}, u) = 1 + \frac{1}{2}\Big(\boldsymbol{x}^T\,\boldsymbol{Q}\,\boldsymbol{x} + R\,u^2\Big), \qquad \boldsymbol{Q}, R > 0,$$

der Endbedingung

$$\boldsymbol{\psi}(\boldsymbol{x}(t_1)) = \boldsymbol{M}\,\boldsymbol{x}(t_1) = 0 \tag{3}$$

und der Anfangsbedingung

$$\boldsymbol{x}_0 = (\pi, 0, 0, 0)^T$$

gelöst werden. Die Gewichtungsmatrizen sind mit

$$\boldsymbol{M} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{Q} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad R = 0.1$$

gegeben.

Die optimale Steuergröße soll durch das Lösen der zu (2) äquivalenten **Zwei-Punkt-Randwertaufgabe** bestimmt werden, also durch Anwendung eines **indirekten Verfahrens**. Studieren Sie zu diesem Zweck das Kapitel 5 aus [Gra19].

Die folgenden Aufgaben sollen für $t_0 = 0$ s und $t_1 = 4$ s gelöst werden.

a) Formulieren Sie das Optimierungsproblem (2) als Zwei-Punkt-Randwertaufgabe. Lesen dafür insbesondere Abschnitt 5.3 und Abschnitt 5.6 aus [Gra19].

b) Zur Lösung der Zwei-Punkt-Randwertaufgabe kann die Funktion `solve_bvp` aus dem SciPy-Modul `scipy.integrate` zum Einsatz kommen. Machen Sie sich mit der Anwendung der Funktion `solve_bvp` vertraut und implementieren Sie ein einfaches Problem aus der Dokumentation von `solve_bvp` selbst. Was für ein Lösungsverfahren [Gra19, vergleiche Abschnitt 5.6] wurde in der Funktion `solve_bvp` implementiert?

c) Bestimmen Sie die Lösung von (2) mit `solve_bvp`. Studieren Sie hier insbesondere die Dokumentation der Argumente/Schlüsselwortargument von `solve_bvp` und variieren Sie diese gegebenenfalls. Achten Sie darauf, aktuelle Programmbibliotheken zu verwenden, z.b. `scipy==1.4.0`, `numpy==1.15.2` und `sympy==1.5` (SymPy ist optional).

d) Bestimmen Sie nun noch eine weitere Lösung so, dass die Eingangsgrößenbeschränkung

$$-12\,\frac{\mathrm{m}}{\mathrm{s}^2} \leq u(t) \leq 12\,\frac{\mathrm{m}}{\mathrm{s}^2} \qquad\qquad \forall \quad t \in [t_0, t_1]$$

nicht verletzt wird. Lesen Sie dafür Abschnitt 5.5 aus [Gra19].

e) Schreiben Sie eine Funktion die die von `solve_bvp` gefundene optimale Steuergröße $u_{\mathrm{opt}}(t)$, die optimale Systemtrajektorie $\boldsymbol{x}_{\mathrm{opt}}(t)$ und das Tupel (t_0, t_1), als Argumente erhält und das System (1) ausgehend von $\boldsymbol{x}_0$ mit $u(t) = u_{\mathrm{opt}}(t)$ von $t = t_0$ bis $t = t_1$ simuliert. Zum Vergleich der Zeitverläufe der simulierten Zustandsgrößen mit denen von `solve_bvp` bestimmten Systemgrößen soll diese Funktion außerdem einen Plot erzeugen. Für die Simulation kann die Funktion `solve_ivp` aus dem SciPy-Modul `scipy.integrate` verwendet werden.

Literatur

[Gra19] K. Graichen. *Skript zur Vorlesung: Numerische Optimierung und modellprädiktive Regelung*. Lehrstuhl für Regelungstechnik, Friedrich-Alexander-Universität Erlangen-Nürnberg, Sommersemester 2019.

[Ste20] A. Steinböck. *Skript zur Vorlesung: Optimierung*. Institut für Automatisierungs- und Regelungstechnik, Technische Universität Wien, Wintersemester 2019/2020. URL: `https://www.acin.tuwien.ac.at/file/teaching/master/optimierung/WS2019/Optimierung_VO.pdf`.