

# Optimale Steuerung und Regelung

1. Übung, ausgegeben von: Thomas Auer  
Frank Woittennek  
zuletzt geändert am: 15. März 2025

Der Lösungsweg sowie der Python-Code sind zwei Tage nach 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

Gegeben ist das mathematische Modell einer harmonisch oszillierenden Punktmasse

$$m\ddot{y}(t) + \omega^2 y(t) = u(t), \quad y(0) = 1, \quad \dot{y}(0) = 0,$$

mit der Masse  $m$  und der Position  $y(t)$ , deren Bewegung über eine externe Kraft  $u(t)$  beeinflusst werden kann. Mit dem PID-Regler

$$u(t) = K_P y(t) + K_D \dot{y}(t) + K_I \int_0^t y(\tau) d\tau$$

soll die Punktmasse optimal bzgl. des Kostenfunktional

$$F(\mathbf{k}, y(\mathbf{k}; t)) = \frac{1}{2} \int_0^\infty \left( q_1 (y(t))^2 + q_2 (\dot{y}(t))^2 + q_3 \left( \int_0^t y(\tau) d\tau \right)^2 + r_1 (u(t))^2 \right) dt$$

auf die Position  $\lim_{t \rightarrow \infty} y(t) = 0$  überführt werden. Verwenden Sie die folgenden Parameter:

$m$	$\omega$	$q_1$	$q_2$	$q_3$	$r_1$
1	2	3	4	5	6

- Führen Sie das oben gegebene dynamische Optimierungsproblem in ein statisches über um diese mit Hilfe eines einfachen Gradientenverfahrens lösen zu können. Wählen Sie dazu einen geeigneten Zustandsvektor.
- Implementieren Sie das Kostenfunktional und den Gradienten davon.
- Implementieren Sie ein Gradientenverfahren [Gra19, Abschnitt 3.3.2] und wenden Sie es auf die vorliegende Problemstellung an.
- Machen Sie sich mit Verfahren zur Schrittweitensteuerung [Gra19, Abschnitt 3.3.1] vertraut und testen Sie eines davon.

- e) Wie kann die Optimierungsaufgabe für den hier vorliegenden Spezialfall, eines linearen Systems mit quadratischem Gütemaß, auch analytisch gelöst werden? Nutzen dieses Verfahren um das Ergebnis Ihres Algorithmus zu validieren. [Woi19, Kapitel 7]
- f) Vergleichen Sie Ihren Algorithmus mit etablierten Implementierungen aus dem Python Paket SciPy [Vir+19].

## Literatur

- [Gra19] K. Graichen. *Skript zur Vorlesung: Numerische Optimierung und modellprädiktive Regelung*. Lehrstuhl für Regelungstechnik, Friedrich-Alexander-Universität Erlangen-Nürnberg, Sommersemester 2019.
- [Vir+19] Pauli Virtanen u. a. “SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python”. In: *arXiv e-prints*, arXiv:1907.10121 (Juli 2019), arXiv:1907.10121. arXiv: 1907.10121 [cs.MS].
- [Woi19] F. Woittennek. *Skript zur Vorlesung: Regelung mechatronischer Systeme*. Institut für Automatisierungs- und Regelungstechni, UMIT Hall in Tirol, Sommersemester 2019.