



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelorarbeit

Liona Multhaup

Automatisierte Auslegung der Steuerung eines Wellenreiters

*Fakultät Technik und Informatik
Department Fahrzeugtechnik und Flugzeugbau*

*Faculty of Engineering and Computer Science
Department of Automotive and
Aeronautical Engineering*

Liona Multhaup
**Automatisierte Auslegung der Steuerung
eines Wellenreiters**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Flugzeugbau
am Department Fahrzeugtechnik und Flugzeugbau
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Deutsches Zentrum für Luft- und Raumfahrt e.V.
Institut für Aerodynamik und Strömungstechnik (Abteilung Raumfahrzeuge)
Lilienthalplatz 7
38108 Braunschweig

Erstprüfer: Prof. Dr.-Ing. Hartmut Zingel
Zweitprüfer: Dipl.-Ing. Marius Franze (DLR)

Industrieller Betreuer: Dipl.-Ing. Marius Franze (DLR)

Abgabedatum: 12.10.2020

Zusammenfassung

Liona Multhaup

Thema der Bachelorthesis

Automatisierte Auslegung der Steuerung eines Wellenreiters

Stichworte

Steuerung, Wellenreiter, SciPy Optimierer, Künstliche Intelligenz, Aerodynamik

Kurzzusammenfassung

In dieser Arbeit werden zur Automatisierung der Auslegung einer Steuerung eines Wellenreiters verschiedene Verfahren untersucht. Es werden dafür vier Optimierungsmöglichkeiten (Nelder-Mead, BFGS, L-BFGS-B, DEAP) und zwei künstliche Intelligenzen (DDPG, TD3) verwendet. Dabei wird sowohl auf die Auslegung selbst als auch die Qualität der daraus folgenden Steuerung eingegangen. Der Erfolg der Ergebnisse wird anhand eines Anwendungsbeispiels eines Wellenreiters untersucht und beurteilt.

Liona Multhaup

Title of the paper

Automatic design of the control of an orbital glider

Keywords

Controller, orbital glider, SciPy optimizer, artificial intelligence, aerodynamics

Abstract

In this thesis different methods are investigated to automate the design of a control of an orbital glider. Four optimization possibilities (Nelder-Mead, BFGS, L-BFGS-B, DEAP) and two artificial intelligences (DDPG, TD3) are used for this. Both the design itself and the quality of the resulting control are discussed. The success of the results is examined and assessed using an application example of an orbital glider.

Inhaltsverzeichnis

Abbildungsverzeichnis	VII
Tabellenverzeichnis	IX
Symbole und Abkürzungen	X
1 Einleitung	1
1.1 Ziel der Arbeit	1
1.2 Struktur der Arbeit	2
2 Theorie	3
2.1 Flugmechanik	3
2.1.1 Achsen und Winkel	3
2.1.2 Auftrieb und Widerstand	4
2.1.3 Gleitzahl und Reichweite	6
2.2 Wellenreiter	7
2.3 Umgebungsparameter für die Berechnungsprogramme	9
2.3.1 SOSE	9
2.3.2 REENT 6D	9
2.3.3 Flugparameter für REENT-SOSE-Kopplung	10
2.4 Flugsteuerung	11
2.4.1 Massenpunktverschiebung	11
2.4.2 Regelung	12
2.5 Optimierung	15
2.5.1 SciPy	16
2.5.2 DEAP	17
2.6 Übersicht KI	18
2.6.1 Machine Learning	18
2.6.2 Reinforcement Learning	19
2.6.3 Deep Learning	20
2.6.4 Deep Reinforcement Learning	20
3 Auslegung	22
3.1 Gegenüberstellung der Optimierungsmethoden	22
3.1.1 Beurteilung des Optimierungsablaufes	22
3.2 Reglerüberprüfung durch Zielfunktion	29
3.3 Gegenüberstellung der KIen	37

3.4	Optimierungsvariable	39
3.4.1	Optimierungsvariablen θ , γ und Höhe	40
3.4.2	Optimierung ohne PID-Regler über die Verschiebung des Schwerpunktes	45
4	Anwendung	46
4.1	Plausibilitätsprüfung der Ergebnisse	46
4.1.1	Simulationsbeiwerte	46
4.1.1.1	Anstellwinkel α	46
4.1.1.2	Temperatur im Staupunkt	49
4.1.2	Massenverschiebung	51
4.2	Implementierung und Anwendung	52
4.2.1	Optimierer	52
4.2.2	KI	54
5	Fazit	55
	Literatur	55

Abbildungsverzeichnis

2.1	Achsen des körperfesten Koordinatensystems, zugehörige Anstellwinkel und angreifende Kräfte [1]	3
2.2	Auftriebsbeiwert über Anstellwinkel α [2]	5
2.3	Schrägstoß Wellen über Mach 2 (a) und Mach 20 (b) [7]	7
2.4	Übersichtshilfe zur Auslegung von PID Regelgrößen[14]	15
2.5	Beispielfunktion zur Verdeutlichung eines Optimierungsvorgangs	16
2.6	KI Übersicht	19
2.7	Grafik zur Verdeutlichung einer RL-Struktur[22]	20
3.1	Targetwerte des Nelder-Mead Optimierers über die Anzahl der Durchläufe	23
3.2	Targetwerte des BFGS Optimierers über die Anzahl der Durchläufe . . .	24
3.3	Ausschnitt aus den Targetwerten des BFGS Optimierers über die Durchläufe	24
3.4	Targetwerte des L-BFGS-B Optimierers über die Anzahl der Durchläufe .	25
3.5	Targetwerte des DEAP Optimierers über jeden 20sten Durchlauf	26
3.6	Vergleich aller errechneten PID-Werte und der Vorgabe	27
3.7	(links:) die mittleren Abweichungswerte über die Anzahl der Durchläufe aller Optimierer, (rechts:) der kleinste Targetwert über den Quotienten aus der mittleren Abweichung zu der Anzahl der Durchläufe	28
3.8	Sprungfunktion als Zielwerte für Optimierer in einem Zeitintervall von $t = 16$	30
3.9	Ausschnitt aus Sprungfunktion von $0 \leq t \leq 4$	31
3.10	Ausschnitt aus Sprungfunktion von $4 \leq t \leq 8$	32
3.11	Ausschnitt aus Sprungfunktion von $8 \leq t \leq 12$	33
3.12	Ausschnitt aus Sprungfunktion von $12 \leq t \leq 16$	34
3.13	Sprungfunktion mit Schrittgröße von 3 (links) und 5 (rechts) als Zielfunktion und die Istwerte aller Optimierer	35
3.14	lineare Zielfunktion und die Istwerte aller vier Optimierer	36
3.15	Sprungfunktion als Zielfunktion für optimierte PID-Werte der KIen DD-PG und TD3 als Variante 1 (links) und Variante 2 (rechts)	39
3.16	Variablenvergleich der Optimierer Nelder-Mead, BFGS, L-BFGS-B über die Anzahl der Durchläufe	40
3.17	Variablenvergleich aller Optimierer über den minimalen Variablenwert aller Optimierungsvariablen	41
3.18	Verlauf von γ aller Optimierer bei konstanter Zielfunktion $\gamma = 0$ im Zeitintervall $t = 16$ bei einer Optimierung nach γ	42
3.19	Vorgabe und Antwortfunktion des Nelder-Meads für den Zielwert $H = 30km$	43

3.20	Simulationenwerte des Nelder-Meads über verschiedene Höhen bei einer Optimierung nach Θ	44
4.1	Verlauf von α für die Optimierung nach $\theta = 0$ beispielhaft für die Optimierer Nelder-Mead, DEAP und die KI $TD3_{var2}$	47
4.2	Verlauf von α für die Optimierung nach $\gamma = 0$ für alle Optimierer	48
4.3	Verlauf von α für die Optimierung nach Höhe = $30km$ mit Nelder-Mead, stellvertretend für alle Optimierer	48
4.4	Verlauf von t_{stau} für die Optimierung nach $\theta = 0$, beispielhaft für die Optimierer Nelder-Mead, DEAP und die KI $TD3_{var2}$	49
4.5	Verlauf von t_{stau} für die Optimierung nach $\gamma = 0$ für alle Optimierer	50
4.6	Verlauf von t_{stau} für die Optimierung nach Höhe = 0 für alle Optimierer	50

Tabellenverzeichnis

0.1	Symbole	X
0.2	Abkürzungen	XI
2.1	Maximal Take Off Weight verschiedener Vergleichsmodelle	8
2.2	Übersicht von verwendeten REENT Modellen	10
3.1	Übersicht, wann der Minimalwert während der Optimierung gefunden wurde	29
3.2	Auszug aus optimierter Wertetabelle des DDPG und TD3 für die PID- Werte sowie die dazugehörigen Targetwerte	38
4.1	Durchschnittliche und maximale Stellzeiten aller Optimierer und KIen bei der Simulation der optimierten PID-Werte nach Θ	51

Symbole und Abkürzungen

Symbole

Symbol	Bedeutung	Einheit
A	Auftriebskraft	N
a	Schallgeschwindigkeit	m/s
CoG	Schwerpunkt entlang einer Körperachse	$\%$
c_A	Auftriebsbeiwert	-
c_W	Widerstandsbeiwert	-
D	Widerstandskraft	N
E	Gleitzahl	-
g	Erdbeschleunigung	m/s^2
H	Flughöhe	km
K	jeweilige Gliederwerte eines PID-Reglers	-
L	Auftriebskraft	N
Ma	Machzahl	-
m_W	Masse des Wellenreiters	kg
m_{MM}	Masse des verschiebbaren Gewichts	kg
p	Druck	
R	Gaskonstante	$kgm^2/(s^2molK)$
S	Fläche des Flugobjektes	m^2
T	Temperatur	K
T	Schubkraft	N
t	Zeit	s
v	Geschwindigkeit	m/s
W	Gewichtskraft	N
α	Anstellwinkel	$^\circ$
γ	Flugbahnwinkel	$^\circ$
Δt	Zeitschrittweite	s
κ	Isentropenexponent	-
ρ	Dichte	kg/m^3
θ	Winkel zw. horizontaler und körperfesten x-Achse	$^\circ$

Tabelle 0.1: Symbole

Abkürzungen

Name	Bedeutung
(L-)BFGS(-B)	(Limited-memory)Broyden-Fletcher-Goldfarb-Shanno(-Bound constraints)
DDPG	Deep Deterministic Policy Gradient (KI-Algorithmus)
DEAP	Distributed Evolutionary Algorithms in Python
DLR	Deutsches Zentrum für Luft- und Raumfahrt
KI	Künstliche Intelligenz
MTOW	Maximales Abfluggewicht
PID-Regler	Proportional-Integral-Differential-Regler
TD3	Twin Delayed DDPG (KI-Algorithmus)
VAR1/2	Variante 1 bzw. Variante 2

Tabelle 0.2: Abkürzungen

1 Einleitung

1.1 Ziel der Arbeit

Nach dem Absturz der Concorde im Jahre 2003 und dem Abschaffen des Space Shuttles im Jahre 2011 wurden fast ein Jahrhundert lang keine neuen Wellenreiter entwickelt und gebaut. Doch heutzutage werden aus der Industrie neue Projekte wie "Lockheed Martin X-59", "Virgin Space Ship (kurz: VSS) Unity" oder "Spike S-512 Supersonic Jet" gestartet. Es sollen neue Flugobjekte für Überschall und sehr große Flughöhen entwickelt werden. Auf Grund der hohen Kosten bei der Entwicklung und Fertigung eines solchen Flugobjektes, sollen schon in der Entwurfsphase Zeit und Kosten gespart werden.

Beim Auslegen eines neuen Flugobjektes sind immer erste grobe Schätzungen erforderlich. Ein Flugregler spielt dabei eine wichtige Rolle, um zu entscheiden, ob das Flugobjekt steuerbar ist. Allerdings ist die Auslegung eines PID-Reglers, welcher zur Regelung einer Maschine meistens verwendet wird, sehr zeitaufwendig und damit teuer. Um nicht unnötig Zeit in mangelnde Entwürfe zu investieren, ist es wichtig eine Methode zur Hand zu haben, die einen Regler zwar grob, aber dafür schnell auslegen kann. Somit wird weniger Zeit in aerodynamische Entwürfe investiert, die später durch unmögliche Steuerungseinstellungen wieder verworfen werden müssen.

In dieser Arbeit soll ein PID-Regler durch Optimierungsprozesse ausgelegt und bewertet werden. Außerdem wird betrachtet, ob eine künstliche Intelligenz diese Aufgabe der Überprüfung der Steuerbarkeit übernehmen kann.

1.2 Struktur der Arbeit

In dieser Arbeit wird zunächst auf die Merkmale und notwendigen aerodynamischen Kenngrößen eingegangen, die zur Entwurfsauslegung wichtig sind. In diesem Teil wird auch erläutert, auf welche Optimierer und künstliche Intelligenzen sich konzentriert wird und wie diese einzuordnen sind.

Die optimierten PID-Regler werden im Darauffolgenden auf ihre Effizienz und Qualität in der Anwendung hin untersucht und bewertet. Dabei wird immer zwischen einem Optimierer und einer künstlichen Intelligenz unterschieden.

Eine Auswertung anhand von praxisnahen Anwendungsbeispielen geschieht im letzten Teil dieser Arbeit und zieht eine Schlussfolgerung nach sich, wie die Auslegung eines PID-Reglers in der Entwurfsphase von Wellenreitern aussehen könnte.

2 Theorie

2.1 Flugmechanik

In diesem Kapitel wird auf die Flugmechanik im stationären Reiseflug eingegangen. Um eine Fluglage ausreichend beschreiben zu können, müssen die verschiedenen Achsen und Winkel bzw. die Bewegung um die jeweiligen Achsen aufgezeigt werden. Hierzu sind diese im Folgenden näher erläutert.

2.1.1 Achsen und Winkel

Im dreidimensionalen Raum gibt es drei translatorische und drei rotatorische Freiheitsgrade. Die translatorischen Freiheitsgrade finden entlang der x -, y - und z -Achse statt (siehe Abb. 2.1). Die Achsen beschreiben ein körperfestes Koordinatensystem, welches seinen Ursprung im Schwerpunkt des Flugobjekts hat. Dabei verläuft die x -Achse von dem Schwerpunkt aus in Richtung der Nasenspitze (vgl. Abb. 2.1). Die y -Achse geht seitwärts Richtung Steuerbord ab und wie in Abb. 2.1 zu sehen, komplettiert die z -Achse das Rechtssystem, indem diese nach unten verläuft.

Die rotatorische Bewegung eines Flugkörpers findet um je eine dieser Achsen statt. In dem Fall wird von Rollen, Nicken und Gieren gesprochen. Dabei versteht man unter dem Rollen eine Drehbewegung um die x -Achse des Objekts. Diese Drehbewegung ist positiv, wenn sie der Rechten-Hand-Regel folgt.

Mit Nicken ist eine Drehbewegung um die y -Achse gemeint und wie später näher erläutert, kann dadurch der Auftrieb angepasst werden.

Beim Gieren dreht sich das Objekt um seine z -Achse. Ein Giermoment wird zur Änderung der Flugrichtung in der x - y -Ebene verwendet.

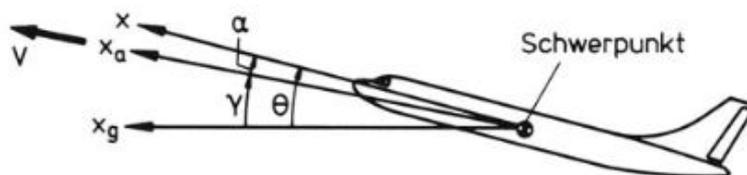


Abbildung 2.1: Achsen des körperfesten Koordinatensystems, zugehörige Anstellwinkel und angreifende Kräfte [1]

In Abb 2.1 sind die nach DIN 9300 genormten Anstellwinkel zu sehen. Sie sind wie folgt definiert.

Der Anstellwinkel α beschreibt den Winkel zwischen der Flugbahn und der körperfesten x-Achse. Wie später näher erläutert, ist α in kleinen Wertebereichen proportional zu dem Auftriebsbeiwert und damit zum Auftrieb. Je größer der Anstellwinkel, um so größer der Auftriebsbeiwert (vgl. Kap. 2.1.2).

γ gibt den Wert des Winkels zwischen der Horizontalen eines geodätischen (erdlotfesten) Koordinatensystems und der Flugbahn an. Dabei ist wichtig zu beachten, dass der Winkel γ von der Horizontalen aus startet, um das Vorzeichen von γ eindeutig zu bestimmen. Dreht γ mit dem Uhrzeigersinn ist der Wert positiv, dreht er gegen den Uhrzeigersinn ist der Wert negativ.

Θ ist die Summe der vorherigen zwei Winkel α und γ . Er zeigt damit den Winkel zwischen der Horizontalen und der körperfesten x-Achse auf.

2.1.2 Auftrieb und Widerstand

Während des Fluges treten verschiedene Kräfte am Wellenreiter auf. Wie in Abb. 2.1 dargestellt, können diese vereinfacht in 4 Kräfte aufgeteilt werden.

Die Gewichtskraft, welche in Abb. 2.1 mit ' W ' (engl.: "Weight") gekennzeichnet ist, zeigt stets vom Schwerpunkt zum Erdmittelpunkt.

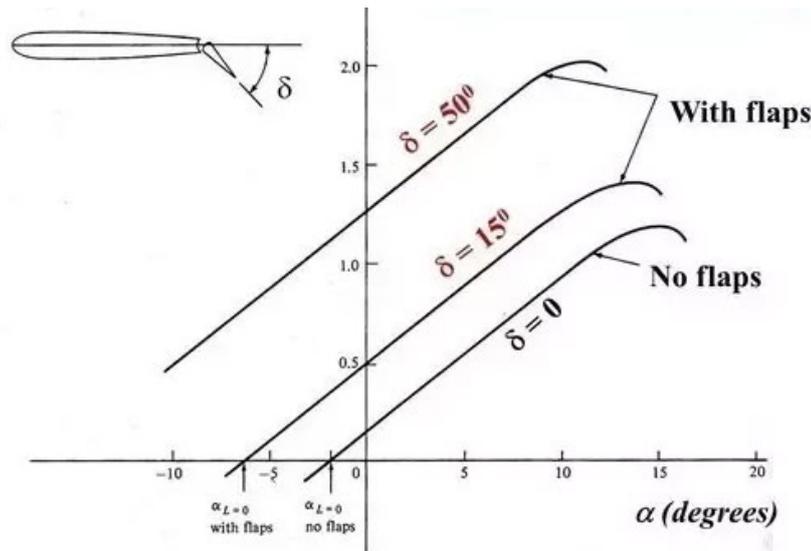
Die Auftriebskraft ' L ' (engl.: "Lift") steht im Lot zum Anströmvektor.

Die Summe dieser beiden Kräfte in geodätischer z-Richtung entscheidet, ob ein Flugobjekt in diesem Flugzustand steigt, sinkt oder konstant in einem Höhenpotential verweilt.

Die Schubkraft ' T ' (engl.: "Thrust") zeigt im Allgemeinen entlang der körperfesten x-Achse in positiver Richtung mit einem kleinen Anstellwinkel, um der Gewichtskraft entgegenzuwirken. Die Schubkraft wird bei Flugzeugen von Triebwerken erzeugt. Im Falle eines Wellenreiters ohne Düsen ist keine Schubkraft gegeben.

Die resultierende Widerstandskraft ' D ' (engl.: "Drag") zeigt immer entgegen der Bewegungsrichtung. Wie später erläutert, setzt sie sich aus mehreren Komponenten zusammen.

In dieser Arbeit wird davon ausgegangen, dass der Wellenreiter keine Triebwerke besitzt, damit also keinen Schub erzeugen kann. Damit ist die Schubkraft $T = 0$. Dies führt außerdem dazu, dass eine quasi-konstante Gewichtskraft vorliegt, da es keinen Massenverlust durch verbrauchten Treibstoff gibt. Allein die Änderung der Erdbeschleunigung $g(H)$ beeinflusst die Gewichtskraft W . Dieser Einfluss ist in niedrigen Höhen allerdings sehr gering und vernachlässigbar.

Abbildung 2.2: Auftriebsbeiwert über Anstellwinkel α [2]

Auftrieb

Beim Bewegen eines Profils in einem Fluid wird es durch seine Krümmung auf der Unter- und Oberseite unterschiedlich schnell umströmt. Dieser Effekt erzeugt einen Druckunterschied und eine dadurch resultierende Kraft. Diese Kraft zeigt bei konventionellen Profilen nach oben und wird als Auftrieb bezeichnet. Dieser lässt sich wie folgt berechnen:

$$A = c_A \frac{\rho}{2} v^2 S \quad (2.1)$$

Hierbei ist ρ die Dichte des Fluids. Im Flugzeugbau lässt sich die Luftdichte anhand von Atmosphärenmodellen über die Höhe bestimmen. Tagesaktuelle Dichte-, Druck- und Temperaturschwankungen werden in der Regel vernachlässigt.

v beschreibt die absolute Geschwindigkeit des Körpers im Raum.

S ist die Bezugsfläche des Profils. Meistens ist im Flugzeugbau damit die Flügelfläche gemeint.

c_A beschreibt einen einheitslosen Beiwert, welcher abhängig ist von der Lage des Körpers in der Strömung. Abb. 2.2 zeigt einen typischen Verlauf für den Auftriebsbeiwert über den Anstellwinkel α eines unsymmetrischen Profils.

Es ist ein linearer Anstieg zu sehen, der bei negativen Anstellwinkeln startet und bis zu $\alpha = 14^\circ$ einen konstanten Anstieg zeigt. Bei größerem α ist ein exponentieller Abfall zu erkennen und danach ein Ende des dargestellten Kurvenverlaufs.

Die Steigung der Linearen, der Schnittpunkt der Geraden mit der y-Achse und die Länge der Gerade sind profilspezifische Werte und außerdem anhängig von weiteren Hochartriebssystemen, die eventuell am Flugobjekt vorhanden sind.

Widerstand

Bei der Bewegung eines Körpers durch ein Fluid entsteht Reibung und Druck an den Oberflächen des Körpers. Die resultierenden Kräfte bezeichnet man als Widerstandskräfte und sie zeigen immer entgegen der Bewegungsrichtung. Die Widerstandskraft lässt sich analog zum Auftrieb wie folgt berechnen:

$$W = c_W * \frac{\rho}{2} * v^2 * S \quad (2.2)$$

Der Widerstandsbeiwert c_W lässt sich in mehrere Anteile aufteilen. Wie bereits erwähnt, kann man Widerstand in Reibung und Druck zerlegen. Analog ist der Beiwert c_W zu unterteilen in c_{WR} für die Reibung zwischen der Körperoberfläche und der Luft und c_{WD} für die unterschiedliche Druckverteilung um den Körper und die daraus resultierende Normalspannung.

Außerdem kommt im Überschall noch der Wellenwiderstand hinzu. Dieser tritt auf, wenn an den angeströmten Körperkanten eine Druckerhöhung auftritt, während die strömungsabgewandten Kanten eine Druckminderung erfahren [3]. Der Wellenwiderstand wird durch den Beiwert c_{WW} berücksichtigt.

Damit ist:

$$c_W = c_{WR} + c_{WD} + c_{WW} \quad (2.3)$$

Im Überschall mit $Ma > 5$ ist der Widerstand allerdings größtenteils druckdominiert.

2.1.3 Gleitzahl und Reichweite

Die Gleitzahl beschreibt, wie weit ein Körper gleiten kann im Bezug auf den Höhenverlust [3]:

$$E = \frac{c_A}{c_W} \quad (2.4)$$

Maximale Gleitzahlen in der Luft- und Raumfahrt schwanken in der Regel von 4,5 (bei Space Shuttles im Endanflug) über 16 (bei Verkehrsflugzeugen im Reiseflug) bis zu 50 (Segelflugzeugen der 18m-Klasse)[4]. Es zeigt, dass, je nach Anwendungsfall, die Gleitzahl, abhängig von den Missionen, unterschiedlich groß gewählt wird.

In dieser Arbeit soll die zurückgelegte Strecke maximiert werden, dadurch ist diese gleich der maximal möglichen Reichweite.

Eine zurückgelegte Strecke ist gleich der Geschwindigkeit in horizontaler Richtung mal die verstrichene Zeit [5].

$$dx = v_h(t)t \quad (2.5)$$

2.2 Wellenreiter

Ein Wellenreiter hat als Charakteristik meistens einen Deltaflügel, also eine negative V-Form als Flügelfläche. Diese Form hat sich im Flug im Überschallbereich besonders bewährt. Grund dafür ist der große Flächenanteil im Mach'schen Kegel und die daraus resultierende Kompressionsauftriebskraft [6].

Durch das Fliegen im Überschall ist es dem Fluid nicht mehr möglich, dem Objekt auszuweichen. Das Objekt schiebt infolgedessen eine Art Druckwand vor sich her. Dies wird auch als Mach'scher Kegel bezeichnet und sieht wie ein Fächer aus, dessen Öffnungswinkel von der Machzahl des Objekts abhängig ist (siehe Abb. 2.3). In Abb. 2.3 ist zu erkennen, dass sich mit größerer Machzahl (vgl. 2.3(a) mit $Mach = 2$ und 2.3(b) mit $Mach = 20$) der Öffnungswinkel verkleinert [7].

Der Winkel des Mach'schen Kegels zeigt außerdem den Winkel des Oblique Shocks (dt.: "Schrägstoß"), also der anliegenden Stoßwelle, an [7]. Der Druck nach einem Schrägstoß ist größer als zuvor. Das heißt, der Druck auf die Körperoberfläche wird größer und damit auch die entgegengesetzte Kraft, die, je nach Anstellwinkel, im Auftrieb resultiert (vgl. Kap. 2.1.2). Je mehr Körperfläche im Mach'schen Kegel liegt, umso größer ist der Auftrieb [6].

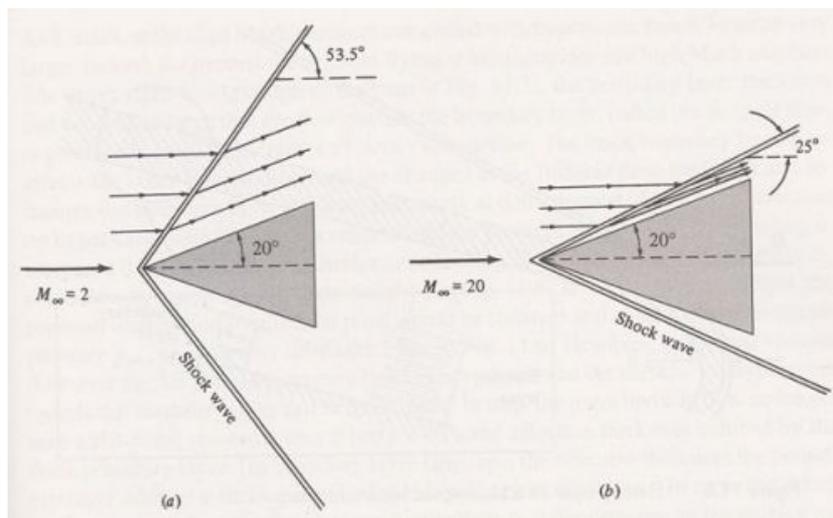


Abbildung 2.3: Schrägstoß Wellen über Mach 2 (a) und Mach 20 (b) [7]

In Abb. 2.3 ist ein Deltaflügel gezeigt, welcher mit Mach 2 (a) und Mach 20 (b) angeströmt wird. Wie in Abb. 2.3 zu erkennen, wird der Winkel der Stoßwelle mit steigender Machzahl kleiner. Damit verkleinert sich die Fläche zwischen dem Stoß und der Körperoberfläche.

Bei großen Reynoldszahlen kann diese außerdem als reibungslos angesehen werden, wes-

halb der Widerstand im Überschall druckdominiert ist. Im Extremfall, wenn Mach gegen Unendlich geht, wird die Fläche zu einer unendlich dünnen Schicht, welche mit einer unendlich großen Dichte einer Scheibe gleich kommt, welche als idealer Auftriebskörper in der Aerodynamik gilt [7].

Aufgrund der guten Einsatzmöglichkeiten im Überschall werden Wellenreiter mit Deltaflügeln als Langstreckenflugzeug oder als Unterstufe eines Raumtransportsystems entwickelt [8].

Für diese Arbeit liegt kein Entwurfsmodell vor, welches aktuell in der Entwicklung ist. Zur numerischen Berechnungen in REENT und SOSE werden trotzdem einige geometrische Werte benötigt, die im Folgenden näher beschrieben werden.

Geometrieannahmen

Für die Abmessungen wurde eine geeignete Referenzlänge und eine Referenzbreite von $1m$ und einer Referenzfläche der Unterseite von $1m^2$ gewählt.

Der Nasenradius wurde mit $0,2m$ gewählt und die Koordianten des aerodynamisches Bezugspunkts bzw. Schwerpunkts im Bezug zur Nasenspitze sind $CoG_{Nx} = 0,6$ und $CoG_{Ny} = CoG_{Nz} = 0$.

Zur Abschätzung einer geeigneten Masse werden folgende Vehikel zum Vergleich herangezogen[9][10][11]:

Modell	MTOW [kg]	Länge [m]	Masse/Längenmeter [kg/m]
NASA X-43A	1.300	3,70	351
Boeing X-51A	1.814	7,62	238
SHEFEX III	2.000	3,36	595

Tabelle 2.1: Maximal Take Off Weight verschiedener Vergleichsmodelle

Die erwartete Masse pro Längenmeter bewegt sich zwischen $238kg$ und $595kg$. Nach interner Absprache wurde für den Wellenreiter ein Gewicht von $280kg$ festgelegt.

2.3 Umgebungsparameter für die Berechnungsprogramme

Die Berechnungen von aerodynamischen Kräften auf einen Flugkörper sind sehr komplex und werden heutzutage von Softwareprogrammen übernommen.

In dieser Arbeit wird eine Verknüpfung zweier Programme zur Berechnung benutzt - REENT und SOSE. Diese sind im Folgenden kurz vorgestellt.

Außerdem wird darauf eingegangen, welche entscheidenden Parameter zur Berechnung benutzt wurden. Hierbei ist darauf hinzuweisen, dass diese nur zur ersten Auswertung der Optimierung gedacht sind. Später in der Arbeit wird darauf eingegangen was diese Ergebnisse für die komplette Mission bedeuten.

2.3.1 SOSE

SOSE ist ein Oberflächenprogramm, welches vom DLR implementiert wurde. Benutzt wird es, um Drücke in Kräfte und Momente auf einer umströmten Oberfläche zu berechnen [12]. Nach der Berechnung gibt SOSE diese Kräfte und Momente an REENT weiter.

2.3.2 REENT 6D

Das Softwareprogramm REENT 6D (Degree of Freedom) wurde ursprünglich in Stuttgart vom IRS (Institut für Raumfahrtsysteme) zur Simulation und Optimierung von Flugbahnen entwickelt und von der Abteilung Raumfahrzeuge des Instituts für Aerodynamik und Strömungslehre des DLR in Braunschweig maßgeblich erweitert [12]. REENT berechnet aus den Momenten und Spannungen, welche in SOSE berechnet wurden, Kräfte, die auf den Körper wirken. Daraus ermittelt REENT dann Geschwindigkeiten und Beschleunigungen in und um alle Achsen [12]. In REENT können verschiedene Atmosphären- und Erdmodelle einbezogen werden.

Die in dieser Arbeit verwendeten Modelle sind in der Tabelle 2.2 aufgelistet.

Als Gravitationsmodell wird das Newton'sche genommen, nach dem sich zwei Massen gegenseitig anziehen. USSTAN62 ist ein klassisches Atmosphärenmodell welches Dichte, Temperatur, Viskosität und Druckverhältnisse über die Höhe gespeichert hat. Dieses wurde 1962 von der NASA erstellt. Auch das thermodynamische Modell basiert auf Daten der NASA. Das verwendete Erdenmodell wird als elliptisch angenommen und Wind nicht in die Berechnung eingebunden. Die Reibung wird in der Simulation nicht vernachlässigt, sondern ist mit in die Berechnung gekoppelt.

h

-MOD	Einstellung
GRAV.	Newtonian
ATMOS.	USSTAN62
EARTH.	elliptic
WIND.	none
THERMO	NASA X-38
AERO.	gekoppelt mit Reibung

Tabelle 2.2: Übersicht von verwendeten REENT Modellen

2.3.3 Flugparameter für REENT-SOSE-Kopplung

Für die numerische Lage- und Kräfteberechnung des Flugobjekts in den Programmen REENT und SOSE werden einige Startparameter benötigt.

In dieser Arbeit werden die Startparameterwerte Anfangsflughöhe, -geschwindigkeit und die Position des Wellenreiters im Raum betrachtet.

Es ist darauf hinzuweisen, dass diese Parameter zwar für die Berechnungen und Optimierungen benutzt wurden, in der Auswertung allerdings darauf eingegangen wird, wie die Ergebnisse für mögliche Startparameter zu werten sind.

Flughöhe

Aus der Information über die Flughöhe beziehen REENT und SOSE die Informationen der Atmosphäre. Dazu zählen die Dichte $\rho(H)$, der Druck $p(H)$ und die Temperatur $T(H)$.

Wie in Kap.2.1.2 erläutert, haben diese Werte Einfluss auf die aerodynamischen Beiwerte, wie zum Beispiel den Auftrieb.

In Anlehnung an die Dissertation von Dr.-Ing.Thino Eggers [8] sind $30km$ als bevorzugte Höhe einer Reiseflugphase zu wählen. Dies erläutert er am Beispiel der Sänger Oberstufe.

Fluggeschwindigkeit

In den Formeln 2.1 und 2.2 wurde gezeigt, dass Auftrieb und Widerstand maßgeblich von der Geschwindigkeit abhängig sind.

Die Flugmachzahl wird über die Höhe in Anlehnung an [8] mit 4,5 gewählt. Nach einer Abschätzung der Temperatur in der Höhe von 30km von $T = -50^\circ\text{C}$, ergibt sich daraus eine Schallgeschwindigkeit von:

$$a = \sqrt{RT\kappa} \approx 300 \frac{\text{m}}{\text{s}} \quad (2.6)$$

Daraus ergibt sich eine Geschwindigkeit von:

$$v_{\text{Start}} = Ma \quad a \approx 1350 \frac{\text{m}}{\text{s}} \quad (2.7)$$

Ausrichtung des Wellenreiters im Raum

Wie zuvor beschrieben, hängt unter anderem c_A und damit der Auftrieb von den Anstellwinkeln ab, weshalb die Position des Wellenreiters im Raum zu Beginn der Simulation definiert werden soll.

Zur Auslegung eines Reglers wurden alle Anstellwinkel zu Beginn der Rechnungen auf 0° gestellt. Das heißt, es findet eine rein translatorische Bewegung entlang des Horizontes statt.

2.4 Flugsteuerung

Die Steuerung eines Flugzeuges geschieht in der Regel durch Quer-, Höhen- und Seitenruder. Diese befinden sich beim Flugzeug an den Trag- und hinteren Ruderflächen oder an Canards. Durch Anstellen dieser Ruderflächen werden Momente auf das Flugzeug aufgebracht, die die absolute Flugrichtung ändern.

Auch der Schub, welcher beispielsweise von den Triebwerken erzeugt wird, kann je nach Anstellwinkel Einfluss auf den Auftrieb nehmen, wie in Kap. 2.1.2 ausführlich erläutert wurde.

In der Auslegung des Wellenreiters in dieser Arbeit wird davon ausgegangen, dass keine Klappen, Ruder oder Triebwerke vorliegen. Die Flugregelung muss also anderweitig geschehen. Eine Möglichkeit ist die Flugregelung durch die Massenpunktverschiebung.

2.4.1 Massenpunktverschiebung

Bei der Massenverschiebung wird im Inneren des Flugobjekts eine Masse platziert, welche entlang ausgewählter Achsen des Flugobjekts beweglich ist und damit den Schwerpunkt

des gesamten Flugobjektes verschiebt. In dieser Arbeit wird vorerst davon ausgegangen, dass die Masse nur entlang der körperfesten x-Achse verschiebbar ist.

Durch eine Verschiebung dieser Masse ändert sich der Angriffspunkt der Gewichtskraft, was zu einem Nickmoment führt. Durch dieses Moment wird der Anstellwinkel α verändert.

Um eine Vorstellung dafür zu bekommen, wie schwer die bewegungsfähige Masse sein müsste, wurde sich am Konzept von SHEFEX III orientiert[13], für welches zur Steuerung des Rollmoments eine verschiebbare Masse ausgelegt wurde.

Nach der Formel [13]

$$\frac{m_{MM}}{m_{SHEFEX}} = \frac{\Delta y_{COG,max}}{\Delta y_{MM,max}} \quad (2.8)$$

kann eine verschiebbare Masse abgeschätzt werden.

Dabei wird die oben abgeschätzte Gesamtmasse m_{SHEFEX} von $280kg$ verwendet. Die weiteren Daten sowie die Masse des abgeschätzten Gewichts werden in der Auswertung näher erläutert. Dort wird in einer Plausibilitätsprüfung untersucht, ob die notwendige verschiebbare Masse umsetzbar erscheint.

Für die weitere Vorgehensweise wurde auf Grund dieser Auslegung von einer Verschiebungsgeschwindigkeit von $10\frac{m}{s}$ ausgegangen. Dies ist eine sehr hohe und unrealistische Geschwindigkeit. In der späteren Auswertung wird darauf eingegangen, ob diese überhaupt notwendig ist.

Da die Position der beweglichen Masse entlang der x-Achse während des Fluges veränderlich ist, wird diese von einem Regler angesteuert. Dies soll in dieser Betrachtung durch einen PID-Regler geschehen, dessen Funktion und Auslegung im folgenden Kapitel näher beschrieben wird.

2.4.2 Regelung

Um zu verstehen, was ein PID-Regler tut und warum er wichtig ist, müssen zunächst ein paar Begriffe aus der Regelungstechnik erklärt werden.

Zunächst ist zu verstehen, was eine Regelung ist. Im Gegensatz zum "Steuern", was einer reinen Vorwärtsbewegung gleich kommt, überwacht eine Regelung fortlaufend Ausgangsgrößen und korrigiert diese in einer Regelstrecke über Stellgrößen [14].

Temperaturregler in einem Kühlschranks oder einer Heizung sind gute Beispiele für Regelungen. Aber auch im Bereich von Geschwindigkeitsregelungen oder in der Belichtungs-

automatik werden Regler eingesetzt [14].

Vereinfacht kann ein Regelkreis einer Regelung zusammengefasst werden in einem sich immer wiederholenden Vorgang von: "Messen", "Vergleichen", "Stellen".

Dabei wird der Regelkreis aufgeteilt in den Regler und die Regelstrecke. Der Regler berechnet sich aus einer Führungsgröße (Sollwert) und einer Regelgröße (Istwert) eine Regelabweichung und gibt als Korrekturmaßnahme eine Stellgröße an die Regelstrecke weiter [14].

Diese Regelstrecke stellt einen Prozess dar, welcher gesteuert werden soll. Der Prozess besteht aus einzelnen Gliedern, die je ein charakteristisches Zeitverhalten aufweisen [14].

PID-Regler

In einem PID-Regler ("Proportional-Integral-Differential") besteht die Regelstrecke aus drei Gliedern. Analog zum Namen lauten diese [14]:

Proportionalglied ("P-Glied")

Integralglied ("I-Glied")

und Differentialglied ("D-Glied").

Der P -Anteil betrachtet den proportionalen Zusammenhang zwischen dem Soll- und Ist-Wert und regelt zeitlich unverzögert mit dem Verstärkungsfaktor K_P [14]. Die zugehörige Übertragungsfunktion lautet:

$$y(t) = K_P e(t) \quad (2.9)$$

Ein einfacher P-Regler stellt die einfachste Form eines Reglers dar, da er nur einen reinen Verstärkungsfaktor verwendet. Er ist anwendbar in Hebeln, Getrieben oder Verstärkern bei denen das Zeitverhältnis vernachlässigbar ist [14].

Als zweites Glied betrachtet der I -Anteil die integrierten Abweichungen, ist also die Summe aller Regelabweichungen. Dieser Anteil ermöglicht eine vollständige Eliminierung der Regelabweichung [14]. In 2.10 ist die Funktion in der Regelstrecke aufgezeigt.

$$y(t) = K_i \int_0^t e(\tau) dt \quad (2.10)$$

Der D -Anteil betrachtet die differentiale Abweichung und reagiert somit nur auf eine Änderungsgeschwindigkeit, nicht die Regelabweichung selbst und multipliziert diese mit dem Faktor K_D [14].

Dargestellt wird dies in der Formel:

$$y(t) = K_d \frac{de(t)}{dt} \quad (2.11)$$

Zusammengefasst ergibt sich für einen PID-Regler dann die Gleichung:

$$y(t) = K_p e(t) + K_i \int_0^t e(\tau) dt + K_d \frac{de(t)}{dt} \quad (2.12)$$

Ein PID-Regler eignet sich also gut für schnelle und dynamische Regelungen aufgrund der Änderung der Regelgeschwindigkeitsänderung im D-Glied, als auch für exakte Regelungen aufgrund des I-Glied, weshalb er ein häufig eingesetzter Regler ist [14].

Dimensionierung PID-Regler

Um einen PID-Regler auszulegen, müssen nach der Formel 2.12 die Werte K_P, K_I und K_D richtig dimensioniert werden. Dazu gibt es verschiedene Methoden.

Zunächst können die Werte nach rein empirischen Methoden erhoben werden. Hierbei ist es üblich, zunächst K_I und K_D auf 0 zu setzen und K_P , angefangen bei sehr kleinen Zahlenwerten, langsam zu steigern. Danach können K_I und K_D der Regelstrecke hinzugefügt werden, bis sich eine ideale Regelung einstellt.

Dieses Verfahren setzt ein wenig Erfahrung voraus, kann dann aber für simple Systeme schnell angewandt werden.

Eine zweite Vorgehensweise ist das Auslegen nach Einstellregeln. Eine sehr bekannte ist die von Ziegler/Nichlos [15], welche sich nochmals unterscheidet in die Schwingungsmethode und in die Einstellung nach der Sprungantwort[14]. Beide stützen sich auf die Formel 2.12. In dieser Arbeit wird sich auf die Schwingungsmethode begrenzt.

Bei diesem Verfahren werden, wie auch bei dem empirischen Verfahren, die Stellgrößen K_I und K_D auf 0 gelassen und der K_P -Anteil langsam erhöht, bis sich eine gleichmäßige Schwingung einstellt [14].

Aus dem Funktionsgrafik, der sich dann ergibt muss dann die Periodenlänge der gleichmäßigen Schwingung als T_{krit} abgelesen werden. Des Weiteren ist das eingestellte K_P als K_{KRkrit} zu notieren.

Mit diesen Zahlen können dann mit Hilfe der Tabelle 2.4 und der dort stehenden Beziehungen $K_I = \frac{K_R}{T_N}$ und $K_D = \frac{K_R}{T_V}$ die restlichen Werte berechnet werden.

Ein Problem bei dieser Art von Auslegung ist, dass das gleichmäßige Schwingen an der Stabilitätsgrenze des Systems geschieht, weshalb dieses Verfahren nicht immer in

Regler	K_R	T_N	T_V
P	$0,5 * K_{Rkrit}$		
PI	$0,45 * K_{Rkrit}$	$0,85 * T_{krit}$	
PD	$0,8 * K_{Rkrit}$		$0,12 * T_{krit}$
PID	$0,6 * K_{Rkrit}$	$0,5 * T_{krit}$	$0,12 * T_{krit}$

mit $K_I = K_R / T_N$ und $K_D = K_R * T_V$

Abbildung 2.4: Übersichtshilfe zur Auslegung von PID Regelgrößen[14]

der Praxis angewendet werden kann. In dieser Arbeit findet diese Analyse allerdings auf Grund von Simulationen statt und kann deswegen problemlos angewandt werden.

2.5 Optimierung

Wie zuvor erwähnt, werden die benötigten Werte von PID-Reglern entweder empirisch oder per Einstellregeln bestimmt. Bestandteil dieser Arbeit ist es, diese Werte mit Hilfe von Simulationen oder einer KI zu optimieren.

Dazu wird in Simulationen eine Funktion durch PID-Beiwerte gesteuert, welche minimiert werden soll. Die verwendeten Beiwerte und das dazugehörige Minimum werden gespeichert und als Referenz für die weitere Optimierung genutzt.

In dieser Arbeit ist die zu minimierende Funktion die Berechnung in der Kopplung von REENT und SOSE. In dieser kann angegeben werden, welcher Wert minimiert werden soll. Zunächst wird sich auf den Winkel $\Theta = 0^\circ$ konzentriert. Durch diesen ist in den Ausgabewerten zu erkennen, mit welchen PID-Beiwerten das Fluggerät horizontal ausgerichtet ist.

Danach findet eine Optimierung zu $\gamma = 0^\circ$ statt. Dies sollte eine möglichst stabile Flughöhe ermöglichen.

Die Addition der Reglerwerte soll dann einen geeigneten Regler produzieren, der sowohl eine stabile Fluglage aufweist, als auch so wenig wie möglich an der gegebenen Höhe verliert.

In einem weiteren Schritt wurde außerdem die Höhe direkt versucht zu optimieren, um zu beurteilen, ob eine zweifache Optimierung nach den Anstellwinkeln Θ und γ notwendig ist.

Optimierungssoftware

Bevor eine Optimierung ausgewertet werden kann, muss zuerst erläutert werden, wie diese Optimierung verläuft. In dieser Arbeit wird auf die folgenden vier Optimierer eingegangen.

”Nelder-Mead”, ”BFGS” und ”L-BFGS-B” sind Methoden implementiert in SciPy, einer Open-Source-Software für pythonbasierte Anwendungsfälle im Bereich Mathematik, Wissenschaften und Ingenieurswesen. Außerdem wird als vierte Optimierungsmethode ”DEAP” gewählt.

2.5.1 SciPy

Zunächst soll verstanden werden, wie SciPy-Algorithmen generell nach Minima suchen. Man stelle sich hierzu einen beliebigen Kurvenverlauf vor. In Abb. 2.5 ist ein solcher beispielhaft dargestellt.

Wie in 2.5 zu erkennen ist, weist dieser Kurvenverlauf zwei lokale Minima (bei $x = 0,3$ und $x = 0,5$) und ein globales Minimum ($x = 0,7$) auf. Da die x-Achse in Abb. 2.5 einheitenlos ist, kann der hier gezeigte Abschnitt unendlich groß sein. Eine Optimierung über einen sehr großen Zahlenbereich dauert dementsprechend lange, da sehr viele Zahlenkombinationen beim Optimieren berechnet werden. Deswegen ist es zu Beginn ratsam, als Anwender diesen Bereich durch Schätzungen einzugrenzen. Es wird außerdem zum Optimieren häufig ein Schätzwert als Startwert angegeben.

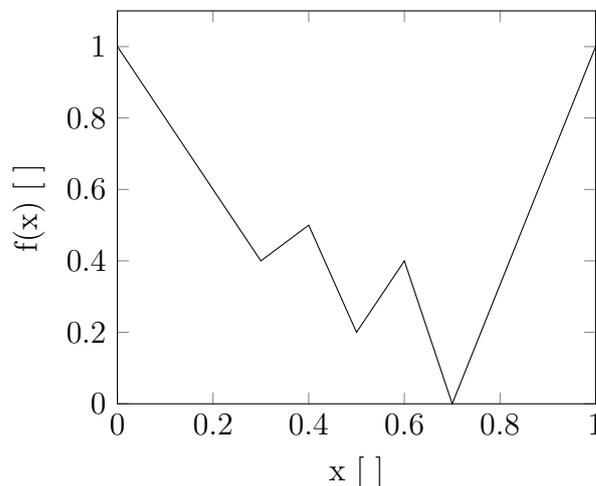


Abbildung 2.5: Beispielfunktion zur Verdeutlichung eines Optimierungsvorgangs

Beispielhaft wird eine Optimierung im Maximalbereich von $x = 0$ bis $x = 1$ gestartet, die den Anfangswert von $x = 0,4$ erhält. Durch Rechenverfahren erlangt ein beispielhafter Optimierer ein Spektrum auf der x-Achse von $\pm 0,15$ (also im Bereich: $0,25 \leq x \leq 0,55$). Die Größe dieses Spektrums ist von zwei Faktoren abhängig: Zum einen können diese Grenzen durch den Anwender gesetzt werden (in dem genannten Beispiel liegen diese allerdings bei 0 und 1). Zum anderen hängt das abgedeckte Spektrum von den Einstellungen im Optimierer ab, wie weit er in eine ungünstige Richtung suchen soll.

Im oben genannten Beispiel findet der Optimierer das Minimum bei $x = 0,5$, da dort der Gradient am größten ist. Für diesen beispielhaften Optimierer ist dies dann, bedingt durch seine Einstellungen, das globale Minimum.

Ein anderer Optimierer testet eventuell noch weiter entlang der steigenden x-Achse. Damit würde dieser über das Maximum bei $x = 0,6$ hinaus gehen und das globale Minimum bei $x = 0,7$ finden. Ein größeres Spektrum ist also erstrebenswert, doch bringt dies auch eine längere Rechenzeit mit sich. Eine größere Schrittweite könnte bei gleicher Rechenzeit das Spektrum auch vergrößern, allerdings würde dadurch die Genauigkeit sinken und Minima eventuell sogar ganz übergangen werden.

Deswegen sollte je nach Anwendungsfall ein Kompromiss aus beidem (Rechenaufwand und Größe des untersuchten Spektrums) gefunden werden. Außerdem ist eine gute Einschätzung der Anfangs- und Grenzwerte empfehlenswert.

Da nun erläutert wurde, wie Minima durch SciPy-Algorithmen in einer Funktion gefunden werden, wird nun erklärt, welche besonderen Eigenschaften und Unterschiede zwischen den Optimierungsmethoden auftreten.

Der Nelder-Mead ist ein sehr bekannter und häufig genutzter Algorithmus in der Optimierung[16]. Er basiert auf einer ableitungsfreien Optimierungsmethode, welche ausschließlich die Werte $f(x)$ benutzt und mit einem $n + 1$ -System ein Minimum durch Verschieben und Verkleinern des X-Achsenabschnittes findet[16]. Ist der Bereich, in dem er verschiebt, zu klein, kann der Effekt eintreffen, der zuvor in dem Beispiel erklärt wurde, dass der Nelder-Mead globale Minima nicht findet. Deshalb ist der Nelder-Mead nicht bei großen Wertebereichen einzusetzen [16].

Der BFGS-Algorithmus ("Broyden-Fletcher-Goldfarb-Shanno") benutzt für seine Optimierung ein lineares Schema, welches sowohl die Optimierungsrichtung als auch die nächste Schrittweite miteinbezieht[16]. Da dieses Schema sehr komplexe Rechnungen beinhaltet, wird an dieser Stelle nur auf die Quelle [16] verwiesen.

Auf Basis des BFGS wurde der L-BFGS-B entwickelt. Dabei steht das "L" für "Limited(-Memory)" und sorgt dafür, dass nur eine limitierte Menge an Computerspeicher verwendet werden darf. Wichtiger ist allerdings der Zusatz "B" für "bound (constrains)", welches ermöglicht, obere und untere Grenzen für Variablen festzulegen[17]. Damit sind physikalisch unmögliche Lösungen limitierbar, falls dies nicht anderweitig in den Funktionen implementiert werden konnte.

2.5.2 DEAP

Bei der Optimierungsmethode des DEAP wird in dieser Arbeit nach der Evolutionsmethode vorgegangen. Das bedeutet, dass DEAP sich zunächst aus einer vorgegebenen Wertemenge eine Population von Individuen zusammenstellt. Jedes Individuum ist eine

mögliche Lösung für das Optimierungsproblem mit einem sogenannten Fitnesswert, der die Wertigkeit dieser Lösung widerspiegelt.

Aus dieser Population sind die Individuen mit ungültiger Fitness auszusortieren. Danach geht die Population in die nächste Generation über. Je Generation wird eine bestimmte Anzahl der besten Individuen in einer externe Datei ausgeschrieben. In der neuen Generation wird dann nach einer statistischen Funktion die Population 1 : 1 mit neuen Individuen zu ersetzen [18]. Danach beginnt die Evaluation von Neuem.

Die Anzahl der Individuen und die Anzahl der Populationen ist dabei vom Anwender festzulegen, ebenso die Anzahl der Generationen und der Individuen.

Die Optimierung von den PID-Beiwerten muss allerdings nicht auf einer rein mathematischen Berechnungssoftware fundieren. Durch die ständige Weiterentwicklung von künstlichen Intelligenzen (kurz: KIen) verspricht dieser Ansatz eine gute Alternative für solch ein Optimierungsproblem zu sein.

Im nächsten Kapitel wird ein Überblick zu diesem Thema gegeben.

2.6 Übersicht KI

Der Bereich der KI ist sehr weit gefächert, weshalb in diesem Kapitel zunächst verdeutlicht wird, was unter einer KI verstanden wird und welcher Bereich für den Anwendungsfall einer Optimierung sinnvoll ist.

Künstliche Intelligenz kann wie folgt definiert werden: "...die Fähigkeit eines Systems, externe Daten korrekt zu interpretieren, aus diesen Daten zu lernen und diese Erkenntnisse zu nutzen, um bestimmte Ziele und Aufgaben durch flexible Anpassung zu erreichen..."[19]

Ein einfaches Beispiel für die Anwendung einer KI wäre die Gesichtserkennung bei heutigen Smartphones. Diese Einsatzmöglichkeit einer KI ist schon sehr fortgeschritten, das Anlernen einer KI basiert auf simplen Prinzipien.

In der Entwicklung einer KI, oft auch als ein "Agent" bezeichnet, kann man sich in verschiedenen Bereichen aufhalten, sogenannten "Challenges" [20]. Wie in Abb. 2.6 zu sehen, fallen darunter Probleme wie die Fähigkeit sich und andere Objekte zu bewegen, automatisierte Planung oder das selbstständigen Lernen, das so genannte "Machine Learning" [20]. Aufgrund des Anwendungsfalls des Erkennen eines Simulationsablaufes und der Optimierung nach verschiedenen Variablen wird sich auf Letzteres in dieser Arbeit konzentriert.

2.6.1 Machine Learning

Das Machine Learning (kurz: ML) hat im Laufe der Zeit unterschiedliche Ansätze entwickelt, wie eine KI angelernt wird. Folgende Methoden zählen zu den bekanntesten und

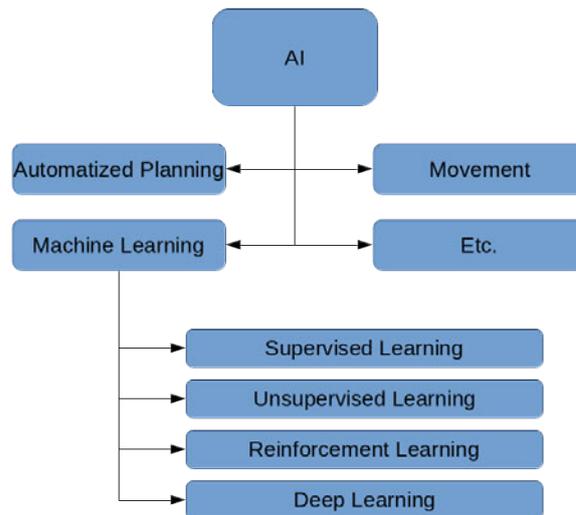


Abbildung 2.6: KI Übersicht

sind in Abb. 2.6 dargestellt [20]:

Es begann mit dem Super- und Unsupervised Learning, welche die klassischen Lernmethoden darstellen, darauf folgte das Reinforcement und Deep Learning, welche heutzutage eher Anwendung finden.

Im Allgemeinen ist zu sagen, dass diese Ansätze sich unterscheiden in den Aktionen der KI in seiner Umgebung und der Art der Rückmeldung/Belohnung, die sie über ihre Aktion bekommen [21]. In dieser Arbeit wird sich auf das "Reinforcement Learning" (kurz: RL) und das "Deep Learning" beschränkt.

2.6.2 Reinforcement Learning

In der Abb 2.7 ist ein typisches RL-System gezeigt. Der sogenannte Agent begeht eine Aktion in einer Umgebung, die dann vom Interpreten bewertet wird und einen neuen Status sowie eine Belohnung, je nachdem wie "gut" die Aktion des Agenten war, an diesen weitergibt.

Der Agent bekommt keinen Satz an Daten über verschiedene Möglichkeiten und deren Auswirkungen, sondern lediglich einen Status. Den Datensatz erstellt er sich durch Trail-and-Error dann selbst. Dadurch liegt der Fokus beim RL größtenteils in der Balance zwischen dem Simulieren und Gewinnen neuer Daten und dem Auswerten der schon vorhandenen Daten [21].

Anwendung findet der RL in vielen Bereichen, besonders allerdings in simulationsbe-

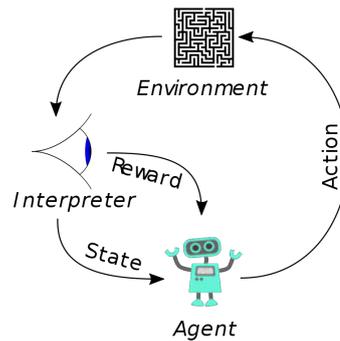


Abbildung 2.7: Grafik zur Verdeutlichung einer RL-Struktur[22]

dingten Spiele-, Operations- und Kontrolltheorien [21].

2.6.3 Deep Learning

Das Deep Learning ist auch ein Algorithmus im Bereich Machine Learning, aber im Gegenteil zum RL benötigt das Deep Learning eine sehr große Datenbank an Informationen. Diese Daten werden in sogenannten "künstlichen neuronalen Netzwerken" gespeichert. Sie bestehen aus mehreren "hidden layers", also sogenannten Schichten, welche untereinander verknüpft sind. Die Besonderheit beim Deep Learning zu vergleichbaren Methoden ist die hohe Anzahl an Schichten, welche sich zwischen dem Input und dem Output befindet [23].

Durch neue Daten lernt das Deep Learning immer mehr und verbessert sich in der Anwendung stetig selbst. Bekannte Beispiele für die Anwendung vom Deep Learning sind: Siri und Cortana [23].

2.6.4 Deep Reinforcement Learning

In dieser Arbeit wird die Kombination des Reinforcement und Deep Learning verwendet, dem sogenannten Deep Reinforcement Learning (kurz Deep RL). Die verbindet das selbstständige Lernen des RL mit einer großen Datenbank des Deep Learning.

Die verwendeten Algorithmen (DDPG und TD3) basieren auf der freien Python Bibliothek "SpinningUp", welche eine freie Implementierung für den Deep RL ist und für OpenAI programmiert ist. Diese stellt Module zum Trainieren einer KI bereit, welche sich im Bereich Deep RL befinden.

Ein wichtiger Punkt in dem Deep RL ist die Unterscheidung zwischen einer modellfreien oder modellbasierenden Rechnung [24]. Mit dem Begriff "Modell" ist eine Funktion

gemeint, welche vorhersagt, wie sich der Status ändert und welche Belohnung der Deep RL dafür bekommt.

In dieser Arbeit wird sich auf zwei Algorithmen beschränkt, die beide modellfrei sind, da sich gezeigt hat, dass modellbasierende Agenten in der gegebenen Umgebung zwar gut performen, aber in einer wahren Umgebung dann sehr schlechte Ergebnisse liefern können [25].

DDPG

Der DDPG ("Deep Deterministic Policy Gradient") ist ein modellfreier Algorithmus, welcher auf einer Q-Funktion und der Optimierung der "Policy" (engl.: Regelwerk) basiert.

Die Q-Funktion ist eine "Optimal Action-Value Function" (optimale Aktions-Wert Funktion). In der Umgebung ist außerdem die "Policy" gegeben, welche beim Anwenden des DDPG optimiert wird [24].

TD3

Der TD3 ("Twin Delayed DDPG") basiert, wie der Name schon verrät, auf dem DDPG, welcher durch die Q-Funktion gerade zu Beginn Eigenschaften hat, die hinderlich beim Lernen sein können. Der TD3 versucht diese mit drei Ansätzen zu beheben. Zum einen lernt der TD3 mit zwei Q-Funktionen und benutzt dann die mit den kleineren Werten, um einem Überschwingen am Anfang entgegenzuwirken. Als Nächstes werden die Richtlinien und Zielwerte verspätet, um eventuell Überschwinger zu dämpfen, und zuletzt wird die Aktion für den Zielwert mit einem Störgeräusch überlagert, damit die Richtlinien sich am Anfang nicht zu sehr in eine Richtung versteifen, die auf das gesamte System gesehen eher negativ zu werten sind [25].

3 Auslegung

Das Kapitel "Auslegung" ist in drei Teile aufgeteilt. Im ersten Teil wird betrachtet, wie die Optimierungsmethoden miteinander verglichen und ihre Effektivität festgestellt werden können. Dabei werden zuerst die Effizienz und danach die Qualität der Regler aufgezeigt und dann gegenübergestellt. Zuletzt wird dargestellt, wie sich die Ergebnisse der vorherigen Betrachtungen je nach Optimierungsvariable voneinander unterscheiden.

3.1 Gegenüberstellung der Optimierungsmethoden

Betrachtet wird zuerst, wie viele Durchläufe die Optimierungsmethode braucht, um ein Minimum zu finden und wie klein die durchschnittliche Abweichung des Soll- und Istwertes ist. Daraus wird erschlossen, wie effizient der Optimierer ist. Außerdem wird ein Durchschnitt aller errechneten Abweichungs-Werte ermittelt, um anhand dieser Information Schlussfolgerungen daraus zu ziehen, in welchem Umfangsbereich der Optimierer nach passenden Minima gesucht hat.

Da die Optimierer nach unterschiedlichen Variablen optimiert haben, sind diese auch getrennt zu betrachten. Zur Erläuterung der Gegenüberstellung wird zunächst nur auf die Optimierung nach θ eingegangen. Daraufhin wird erläutert, wie sich die Optimierungsverfahren bei anderen Variablen verhalten.

Jeder Optimierer hat zu Beginn einen Startwert für die PID-Werte bekommen. Diese sind nach dem Ziegler/Nichlos Verfahren empirisch bestimmt worden. Als Anfangswert für die Optimierung nach $\Theta = 0^\circ$ und $\gamma = 0^\circ$ sind $K_P = 0,3$, $K_I = 0,1$ und $K_D = 0,01$ gewählt worden. Für den Optimierungsfall nach der Höhe mit $H = 30km$, sind für $K_P = 0,5$, $K_I = 0,01$ und $K_D = 0$ als Anfangswerte festgelegt worden.

Die Flugdauer ist bei allen Optimierungen gleich mit $t = 10s$.

3.1.1 Beurteilung des Optimierungsablaufes

Zum besseren Verständnis sollen zwei Begriffe kurz erläutert werden. Ein "Durchlauf" bezeichnet die einmalige Berechnung aller Flugzustände innerhalb des vorgegebenen Zeitintervalls mit der Zeitschrittgröße von $t = 0,01s$. Pro Durchlauf wird somit eine mögliche PID-Werte-Zusammensetzung berechnet. Die Anzahl der Durchläufe ist damit gleich der Anzahl der überprüften Kombinationen aus PID-Werten eines Optimierers. Außerdem ist analog zur Anzahl der Durchläufe auch auf die benötigte Rechenzeit zu schließen,

mit Ausnahme vom DEAP, worauf später eingegangen wird.

Als "Targetwert" wird der Wert bezeichnet, welcher durch den Einfluss von Soll- und Ist-Wert einer Variablen minimiert werden soll. Er setzt sich zusammen aus dem Mittelwert aller betragsmäßigen Standardabweichungen abzüglich dem Mittelwert aller Ist-Werte eines Durchlaufes.

In den Abb.3.1 bis 3.5 sind Grafiken dargestellt, welche die Verläufe des Targetwertes über die Anzahl der Durchläufe darstellt. Es wird betrachtet, wie viele Durchläufe der Optimierer insgesamt benötigt, ob über den Verlauf der Durchläufe eine Tendenz der Verkleinerung des Targetwertes zu erkennen ist und wie groß das Spektrum des abgesuchten Gebietes ist. Letzteres soll vereinfacht durch die Mittelwertbildung der Standardabweichungen geschehen.

Die folgenden Gegenüberstellungen haben alle Anfangseinstellungen wie in Kapitel 2 beschrieben.

Nelder-Mead

Beginnend mit dem Verlauf des Targetwertes unter dem Optimierer Nelder-Mead, ist in Abb.3.1 zu sehen, dass dieser nur 24 Durchläufe startete, um zu einem Ergebnis zu kommen. Aufgrund dieser niedrigen Durchlaufzahl ist zu vermuten, dass der Nelder-Mead kein großes Spektrum an möglichen PID-Werte-Zusammensetzungen abdeckt. Die Abweichungen der Targetwerte untereinander sind alle sehr klein in einem Wertebereich von 0,0017 bis 0,0002. Im Verlauf des Grafiken ist zu sehen, dass sich die Targetwerte tendenziell verkleinern, dabei aber in ihren absoluten Werten immer zwischen hohen und niedrigen Werten schwanken.

Als Endergebnis ist der minimale Targetwert von $2,2125 \cdot 10^{-4}$ vom Nelder-Mead geliefert worden. Die dazugehörigen PID-Werte lauten $K_P = 0,3067$, $K_I = 0,0954$ und $K_D = 0,0107$. Die Abweichungen zu den geschätzten Vorgaben sind sehr gering.

Als Varianz aller Standardabweichungen ergibt sich $1,7665 \cdot 10^{-4}$. Dies unterstützt die Vermutung, dass der Nelder-Mead kein großes Spektrum an möglichen Zusammensetzungen untersucht hat.

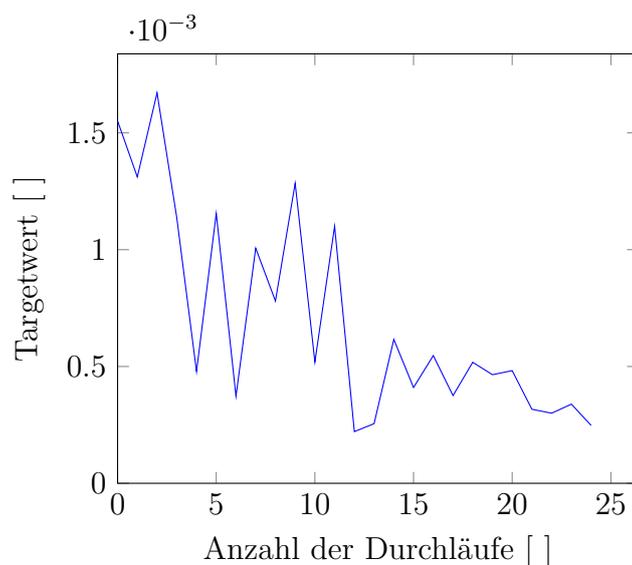


Abbildung 3.1: Targetwerte des Nelder-Mead Optimierers über die Anzahl der Durchläufe

BFGS

In Abb 3.2 ist der Targetverlauf der Optimierung mit dem BFGS dargestellt. Insgesamt macht der BFGS rund 150 Durchläufe, also deutlich mehr als Nelder-Mead, allerdings im Vergleich zu den folgenden Optimierern immer noch relativ wenige. In dieser Darstellung ist deutlich zu erkennen, dass die Werte die meiste Zeit im Bereich $y \ll 10$ bleiben, allerdings zu Beginn der Optimierung auch in höhere Wertebereiche abweichen. Zur besseren Ansicht in kleineren Wertebereichen wird in Abb.3.3 ein Ausschnitt aus der Abb.3.2 gezeigt. Es ist zu erkennen, dass sich die Werte nach dem 20ten Durchlauf in einem viel kleinerem Wertebereiche einpendeln.

Hier ist allerdings keine Tendenz zu erkennen, die eine Verringerung des Targetwertes aufzeigt. Der Targetwert schwankt kontinuierlich in weitaus größeren Wertebereichen als der Nelder-Mead dies tut.

Der Endtargetwert, welcher durch den BFGS errechnet wurde, ist $30,9206 \cdot 10^{-4}$ und die dazugehörigen PID-Werte sind $K_P = 0,3$, $K_I = 0,0999$ und $K_D = 0,01$. Diese sind nahezu identisch mit der Vorgabe der PID-Werte in der ersten Schätzung.

Die Varianz aller Targetwerte beträgt 3,199 und liegt deutlich über der des Nelder-Mead. Es ist davon auszugehen, dass ein größeres Spektrum an Möglichkeiten untersucht wurde. Die kleinere Abweichung der optimierten PID-Werte zur Vorgabe im Vergleich zu den optimierten Werten durch den Nelder-Mead können auf die bessere Annäherung an 0 des minimierten Endwertes zurück geführt werden.

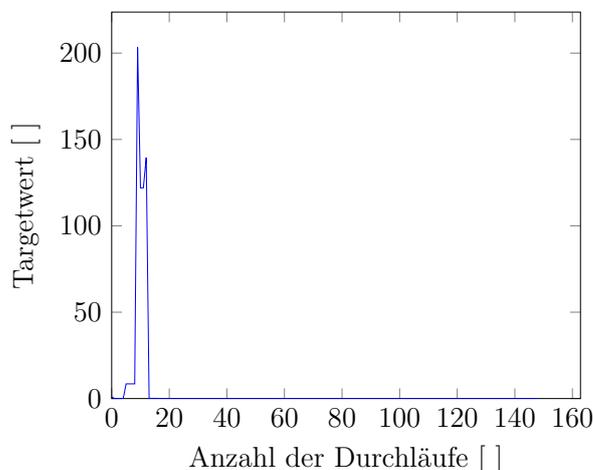


Abbildung 3.2: Targetwerte des BFGS Optimierers über die Anzahl der Durchläufe

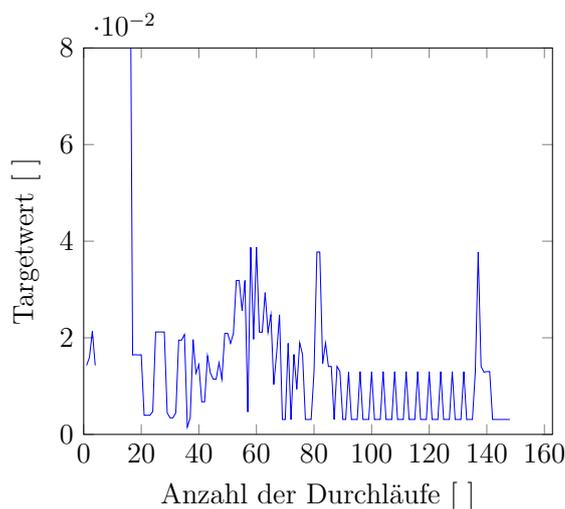


Abbildung 3.3: Ausschnitt aus den Targetwerten des BFGS Optimierers über die Durchläufe

L-BFGS-B

Der Verlauf der Targetwerte der Optimierung durch den L-BFGS-B über die Anzahl der Durchläufe ist in Abb.3.4 zu sehen. Mit deutlich über 6000 Durchläufen ist das abgesuchte Spektrum mehr als 40mal größer als bei den vorherigen Optimierern.

Der Wertebereich schwankt in den ersten 5000 Durchläufen sehr stark zwischen Werten nahe 0 und bis zu 700. Dabei sind die absoluten Werte häufig entweder sehr klein oder im hohen Hunderter Bereich. Das Schwanken zwischen sehr großen Werteunterschieden nimmt im Laufe des Diagrammes zu. Eine Verkleinerung der gemittelten Werte ist nicht zu erkennen.

In den letzten 1000 bis 1500 Durchläufen sind die absoluten Werte zwar auf einen Wertebereich bis ungefähr 200 beschränkt, allerdings sind auch diese im Vergleich zu den anderen Werten von den vorherigen Optimierern sehr große Targetwerte.

Als minimaler Endwert gibt der L-BFGS-B den Wert $0,4140 \cdot 10^{-4}$ aus. Die dazugehörigen PID-Werte lauten $K_P = 0,2745$, $K_I = 3,4954$ und $K_D = 0,1323$. Dieser minimierte Targetwert ist kleiner als bei den vorherigen Optimierern und liefert PID-Werte, die stark von der initialisierten Vorgabe abweichen.

Der Mittelwert der Standardabweichung von 81,8553 bestätigt ein weitaus größeres Abtastspektrum der Möglichkeiten, was in der starken Abweichung des optimierten K_I -Wertes von den Vorgaben auch widerspiegelt wird. Bei der Betrachtung der Optimierer hat der L-BFGS-B den größten Mittelwert der Standardabweichungen und spiegelt damit das größte Abtastspektrum wider.

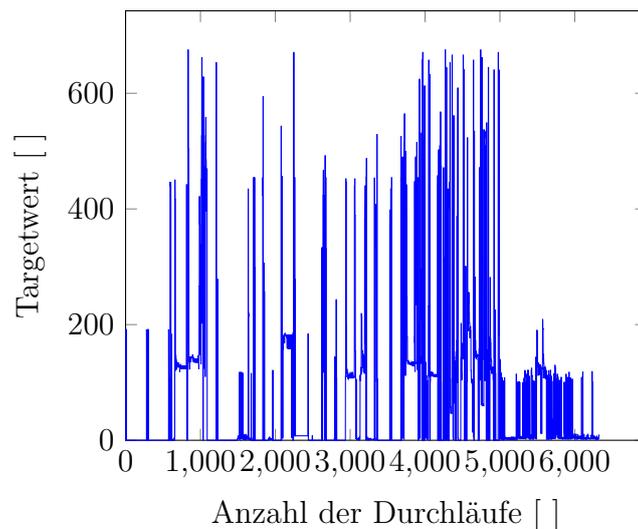


Abbildung 3.4: Targetwerte des L-BFGS-B Optimierers über die Anzahl der Durchläufe

DEAP

Der Optimierer DEAP arbeitet mit Multiprozessoren, das heißt es finden mehrere Rechnungen gleichzeitig statt. Dies führt zu einer geringeren Rechenzeit, allerdings auch mehreren Ausgabedateien, welche nicht ordnungsgemäß in einer Datei sortiert werden können, da die Verteilung der Ergebnisse zwar annähernd, aber nicht exakt auf alle Dateien gleichermaßen aufgeteilt wird. Die Optimierung durch den DEAP wird hier mit 20 Parallelrechnungen gleichzeitig gestartet. Stellvertretend für die Entwicklung der Targetwerte in der Optimierung durch den DEAP, wird die erste Ausgabedatei analysiert. Diese Werte entsprechen vereinfacht jedem 20sten Durchlauf. Der aufgezeichnete Verlauf ist mit dem Verlauf der anderen 19 Dateien vergleichbar.

Die Anzahl der Durchläufe beträgt über 900, insgesamt also knapp unter 20.000 Durchläufen, was ein deutlicher Anstieg an Durchläufen im Vergleich zu den anderen Optimierern ist. Da die Rechnungen aber parallel gerechnet werden, steigt die Rechendauer nicht analog zu den vorherigen Optimierern.

Innerhalb des ersten Drittels ist in der dazugehörigen Abb.3.5 eine kontinuierliche Senkung des durchschnittlichen Targetwertes zu erkennen, wobei die absoluten Werte um mehrere 100 schwanken. In den darauf folgenden zwei Dritteln bewegen sich die Absolutwerte in sehr kleinen Zahlbereichen, mit wenigen Ausnahmen, bei denen der Targetwert kurzzeitig wieder rapide ansteigt. Die Tendenz der Verringerung der Targetwerte ist über die gesamte Optimierung also durchaus zu erkennen.

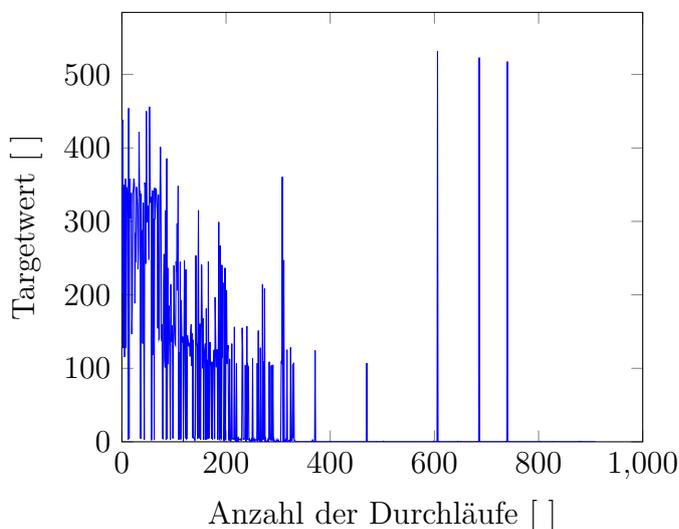


Abbildung 3.5: Targetwerte des DEAP Optimierers über jeden 20sten Durchlauf

Das kleinste gefundene Minimum durch den DEAP lautet $0,14089 \cdot 10^{-4}$ und ist somit der kleinste Endwert aller Optimierer. Die dazu gehörigen PID-Werte lauten $K_P = 0,2849$, $K_I = 6,4796$ und $K_D = 0,1278$. Auffällig ist dabei der hohe K_I -Wert, welcher stark von der erstmaligen Schätzung und allen anderen optimierten K_I -Werte abweicht.

Die Varianz der Targetwerte aller knapp 20.000 Durchläufe ist 46,1060.

Vergleich

Zusammengefasst sind in den folgenden Diagrammen nochmal die PID-Werte aller Optimierer aufgelistet, um einen direkten Vergleich hervorzuheben.

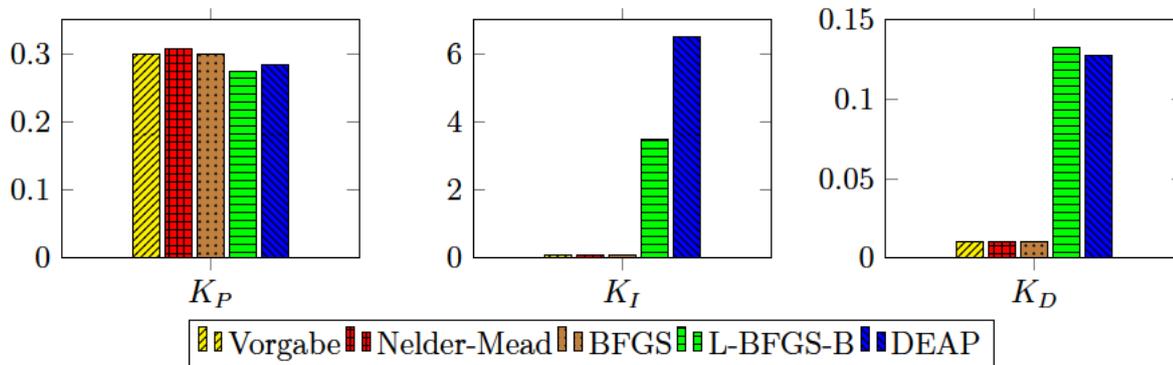


Abbildung 3.6: Vergleich aller errechneten PID-Werte und der Vorgabe

Wie in Abb.3.6(K_P) zu sehen, bleibt der K_P -Wert bei allen Optimierern recht nahe bei der Vorgabe von 0,3.

In Abb.3.6(K_I) ist bei den K_I -Werten zu sehen, dass der Nelder-Mead und der BFGS recht stark bei dem Vorgabewert bleiben, während der L-BFGS-B und der DEAP weitaus höhere Werte annimmt. Mit 3,4954 liegt der L-BFGS-B zehnmal so hoch wie die geschätzte Vorgabe und der DEAP mit 6,4796 20mal so hoch.

Auch beim K_D -Wert in Abb.3.6(K_D) gibt es nennenswerte Unterschiede. Auch hier bleiben der Nelder-Mead und der BFGS nahe der Vorgabe, während der L-BFGS-B und DEAP höhere Werte als Lösung ansetzen, die ungefähr zwölfmal so hoch sind.

In Abb.3.7(links) ist für den Optimierer die mittlere Varianz über die Anzahl der gestarteten Durchläufe aufgetragen.

Wie zuvor schon beschrieben, ist die Varianz der Targetwerte ein Indiz dafür, wie groß das Abtastspektrum eines Optimierers ist.

Diesem wird die benötigte Zeit gegenüber gestellt. Da diese aber stark von dem PC und dessen Rechenleistung abhängt ist wird analog die Anzahl der gestarteten Rechnungen in Betracht gezogen. Da der DEAP mehrere Rechnungen gleichzeitig startet, sind dessen gesamte Durchläufe durch die Anzahl der Parallelrechnungen geteilt worden. In diesem Fall 19539 Rechnungen durch 20 Parallelprozesse.

Ein guter Optimierer hat eine niedrige Anzahl an Durchläufen und weist dabei ein hohes Abtastspektrum auf.

Dies ist leider bei keinem der Optimierer der Fall. Während der Nelder-Mead und BFGS zwar wenige Durchläufe starten, bis sie zu einer Lösung kommen, brauchen der L-BFGS-B und DEAP weitaus mehr Durchläufe und damit Rechenzeit. Der Nelder-

Mead und BFGS weisen dabei ein viel kleineres Abtastspektrum auf als die anderen beiden.

Dabei ist festzuhalten, dass die Anzahl der Durchläufe des DEAPs von dem Benutzer festgelegt wird. Eine Unterscheidung des Optimiers DEAP in der Anzahl der Durchläufe wurde in dieser Arbeit nicht berücksichtigt.

Eine hohe mittlerer Standardabweichung könnte allerdings auch durch viele sehr ungünstige Ergebnisse hervorgerufen werden. Es muss also auch untersucht werden, wie gut die Ergebnisse dieser Optimierer sind. Für diese Überprüfung wurden in Abb.3.7(rechts) die minimalen Endergebnisse der Effizienz der Optimierer gegenübergestellt. Als Effizienz wird an dieser Stelle der Quotient aus der mittleren Standardabweichung zu der Anzahl der Durchläufe angenommen. Gewünscht ist also eine hohe Effizienz mit einem möglichst kleinem optimierten Targetwert.

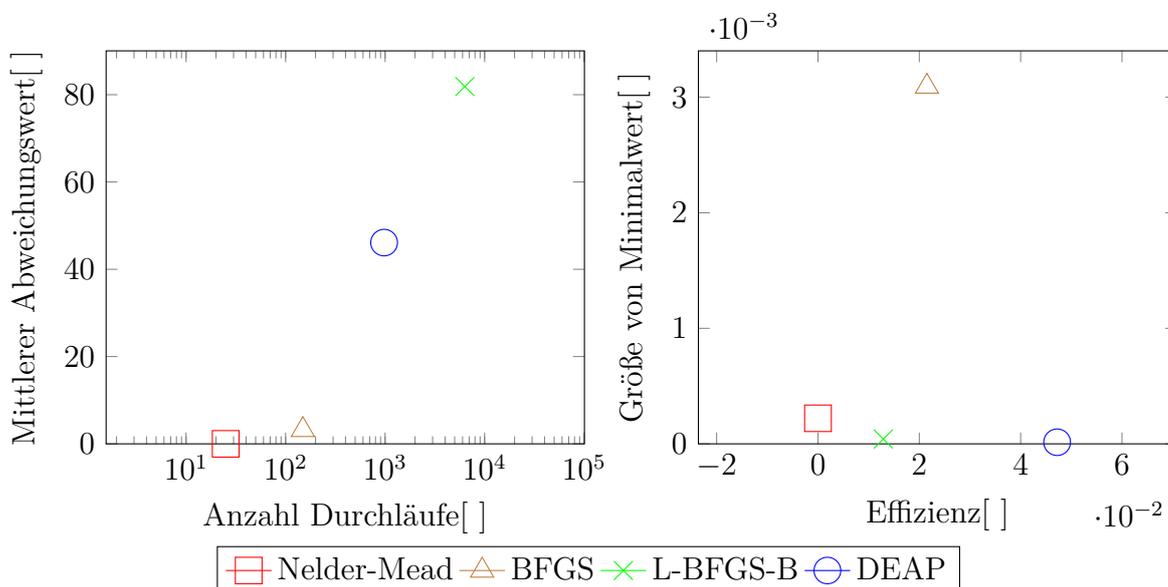


Abbildung 3.7: (links:) die mittleren Abweichungswerte über die Anzahl der Durchläufe aller Optimierer, (rechts:) der kleinste Targetwert über den Quotienten aus der mittleren Abweichung zu der Anzahl der Durchläufe

Da der Nelder-Mead nur einen sehr kleinen Bereich analysiert, ist sein Effektivwert im Vergleich am kleinsten, allerdings ist auch der Minimalwert sehr klein.

Der L-BFGS-B hat wegen seiner großen Anzahl an Durchläufen auch eine kleine Effizienz, findet allerdings auch ein sehr kleines Minimum.

Mit eindeutig dem größten Targetwert erscheint der BFGS zunächst ungeeignet. Mit einer hohen Effizienz und einem absolutem Minimumwert von ca. 0,003 sollte aber auch dieser weiterhin in Betracht gezogen werden.

Der DEAP scheint mit den Kriterien, die bis hierhin aufgestellt wurden, die beste Wahl zu sein. Mit der besten Effizienz und dem kleinsten Targetwert scheint man keinen Kompromiss eingehen zu müssen.

Es ist zu erwähnen, dass die Minimalwerte nicht am Ende der Optimierung gefunden werden. In Tab.3.1 ist aufgetragen, an welcher Stelle der Durchläufe das Minimum je Optimierer gefunden wird und wie viele Durchläufe der Optimierer insgesamt macht. Dies ist hier beispielhaft anhand der Optimierung nach θ aufgelistet. Während der BFGS und DEAP die Minimalwerte im ersten Viertel der Optimierung finden, tut der L-BFGS-B dies erst nach $3/4$ seiner Optimierung. Der Nelder-Mead liegt sehr genau in der Mitte.

Wie zuvor erwähnt, kann die Anzahl des DEAPs variiert werden. Es ist zu erkennen, dass das frühe Finden des Minimums darauf hindeutet, dass eine Reduzierung der Durchläufe sinnvoll ist.

Optimierer	Stelle des Minimalwertes []	Anzahl aller Durchläufe []
Nelder-Mead	13	25
BFGS	37	149
L-BFGS-B	4862	6326
DEAP	1241	19539

Tabelle 3.1: Übersicht, wann der Minimalwert während der Optimierung gefunden wurde

3.2 Reglerüberprüfung durch Zielfunktion

Den vorangegangenen Werten wird die Qualität der optimierten PID-Regler gegenüber gestellt. Es werden drei Zielfunktionen gewählt, welche den zu optimierenden Wert entweder sprunghaft oder linear über einen Zeitintervall ändern. Ein guter Regler würde dieser Zielfunktion so schnell wie möglich folgen ohne groß über deren Wert hinaus zu schwingen und sich bei diesem neuen Wert einpendeln.

Die drei Zielfunktionen sollen verschiedene Aspekte der Antwortfunktion hervorheben. Zunächst soll auf gezeigt werden, wie gut der Regler sich auf einen anderen Wert ungleich 0 einschwingen kann. Die dazugehörige Funktion lautet wie folgt, wobei x die Zeit in Sekunden darstellt und $f(x)$ den Winkelausschlag Θ

$$\begin{aligned}
 f(x) &= 0 \text{ für } x \leq 4 \\
 f(x) &= 1 \text{ für } 4 < x \leq 8 \\
 f(x) &= 2 \text{ für } 8 < x \leq 12 \\
 f(x) &= 1 \text{ für } 12 < x \leq 16
 \end{aligned} \tag{3.1}$$

In Abb. 3.8 ist diese zusammen mit den Antwortfunktionen aufgeführt. Zunächst ist zu erkennen, dass alle Regler nach einem Sprung in der Zielfunktion diesen Werten folgen und um den neuen Zielwert schwingen. Die Größe der Amplituden und ob diese Schwingung sinusförmig verläuft, hängt nicht nur von den Reglern, sondern auch von der verstrichenen Zeit ab.

Letzteres ist darauf zurückzuführen, dass die vorgegebenen Startwerte keinen stabilen Flugzustand beschreiben, sondern getroffene Annahmen und Schätzungen sind. Flugwinkel, -geschwindigkeit, -höhe und andere Parameter verändern sich stetig. Auch ist als Anfangsgröße zwar eine Geschwindigkeit, jedoch keine Beschleunigung des Flugobjektes gegeben. Dies ist der Grund, weshalb alle Antwortfunktionen zunächst unter 0° fallen, bevor sie sich um den ersten Zielwert einschwingen.

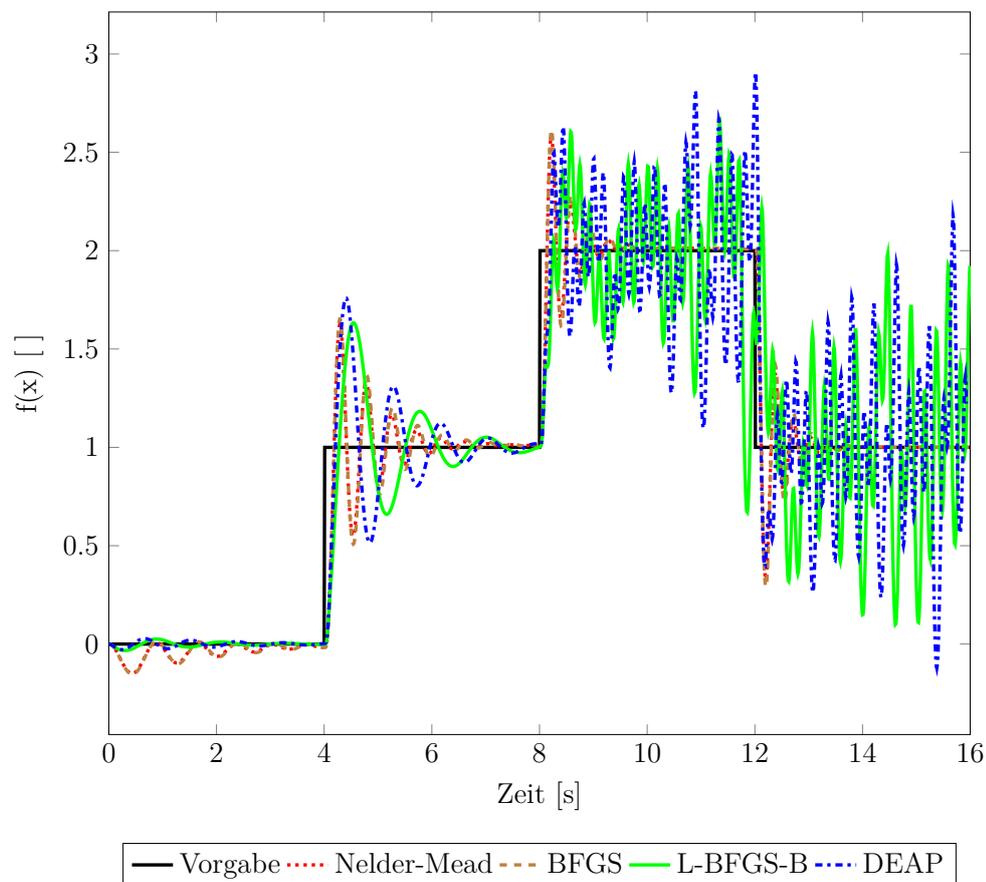
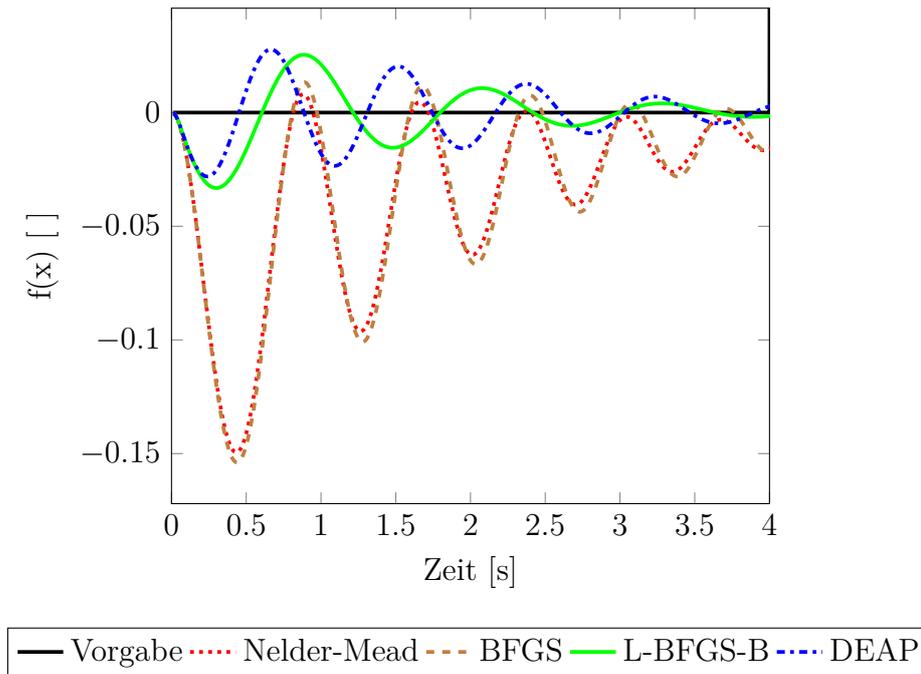


Abbildung 3.8: Sprungfunktion als Zielwerte für Optimierer in einem Zeitintervall von $t = 16$

Der erste Zeitabschnitt mit $0 \leq t \leq 4$ Sekunden ist in Abb. 3.9 hervorgehoben. Alle optimierten PID-Werte rufen eine sinusförmige Schwingung hervor, welche über die gegebene Zeit abklingt.

Abbildung 3.9: Ausschnitt aus Sprungfunktion von $0 \leq t \leq 4$

Es ist zu erkennen, dass die PID-Werte des DEAP und L-BFGS-B, welche beide einen hohen K_I -Wert besitzen, kleinere negative Amplituden haben und ihre durchschnittlichen Werte einer Periode nahe bei dem Zielwert liegen. So kommt der DEAP auch als Erstes aus dem negativen Zahlenraum und der L-BFGS-B erzielt schon bei der zweiten und der DEAP bei der dritten Schwingung einen Amplitudenwert, welches sich sowohl im Positiven als auch im Negativen mit weniger als $0,02^\circ$ vom Zielwert unterscheidet.

Die PID-Werte des Nelder-Meads und des BFGS ähneln den geschätzten Vorgaben und zeigen kaum einen Unterschied auf. Beide schwingen hauptsächlich unterhalb des Zielwertes. Ihre durchschnittlichen Werte innerhalb einer Periode liegen auch unterhalb des Zielwertes und in größerem Abstand zum Zielwert als bei den anderen Optimierern. Dieser Abstand verringert sich innerhalb dieses Zeitintervalls.

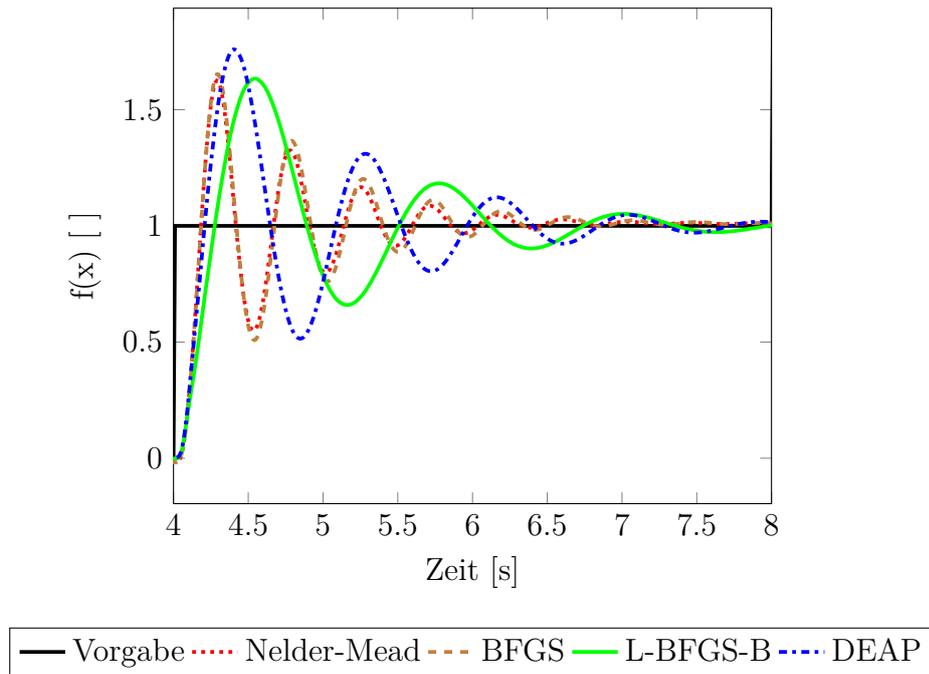


Abbildung 3.10: Ausschnitt aus Sprungfunktion von $4 \leq t \leq 8$

Im zweiten Abschnitt, welcher in der Abb. 3.10 vergrößert dargestellt wird, sind die Sprungantwort von 0° auf 1° und die geregelten Größen aller Optimierer innerhalb des Zeitintervalls von 4 bis 8 Sekunden dargestellt. Bei allen Optimierern ist wieder eine sinusförmige Schwingung um den neuen Zielwert von 1° zu sehen, welche innerhalb des Zeitintervalls deutlich abklingt.

Es ist hervorzuheben, dass der absolute Wert des ersten Übersprungs bei dem DEAP mit rund $0,75^\circ$ größer ist als bei den anderen mit $0,6 - 0,65^\circ$. Die darauffolgenden Amplitudenwerte des DEAPs unterscheiden sich zwar wenig von denen der anderen Optimierer, sie sind dafür zeitlich versetzt, was das Abklingen der Schwingung verlangsamt. Dieser Unterschied ist am Ende des Zeitintervalls kaum noch sichtbar.

Der Nelder-Mead und BFGS liegen immer noch sehr nahe beieinander. Sie klingen schneller als die anderen Schwingungen ab und pendeln sich am Ende des Zeitintervalls knapp oberhalb des Zielwertes ein. Die Unterschreitung einer Abweichung von $0,02^\circ$ liegt bei einer Zeit von $t = 0,8s$ vor.

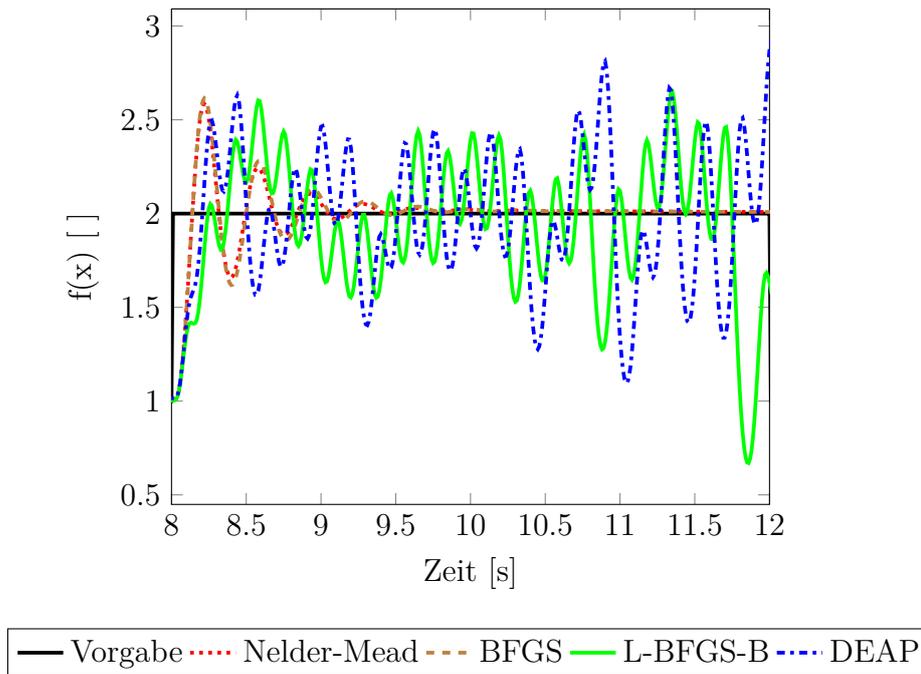


Abbildung 3.11: Ausschnitt aus Sprungfunktion von $8 \leq t \leq 12$

Im darauffolgenden Sprung der Zielfunktion, welche in Abb. 3.11 dargestellt ist, werden die Unterschiede der Optimierer deutlicher. Der Nelder-Mead und der BFGS folgen wieder einer sinusförmigen Schwingung, die sich im Vergleich zum ersten Sprung auf den Wert 1 schneller einschwingt. Die Amplituden sind beim ersten Übersprung wieder bei ca. $0,6^\circ$ und eine positive sowie negative Abweichung von weniger als $0,2^\circ$ liegt ab der zweiten Periode vor.

Bei den optimierten Werten des L-BFGS-B und DEAP ist eine Überlagerung mehrerer Sinusschwingungen zu erkennen. Der durchschnittliche Wert dieser überlagerten Schwingung scheint im Bereich des Zielwertes zu liegen, allerdings sind die absoluten Amplitudenwerte kontinuierlich stark vom Zielwert unterschiedlich. Ein Abklingen der Schwingung ist nicht zu erkennen. Im Gegenteil vergrößern sich die Werte der Abweichung eher, sodass der PID-Regler des L-BFGS-B am Ende des Zeitintervalls sogar den vorherigen Zielwert von 1° unterschreitet.

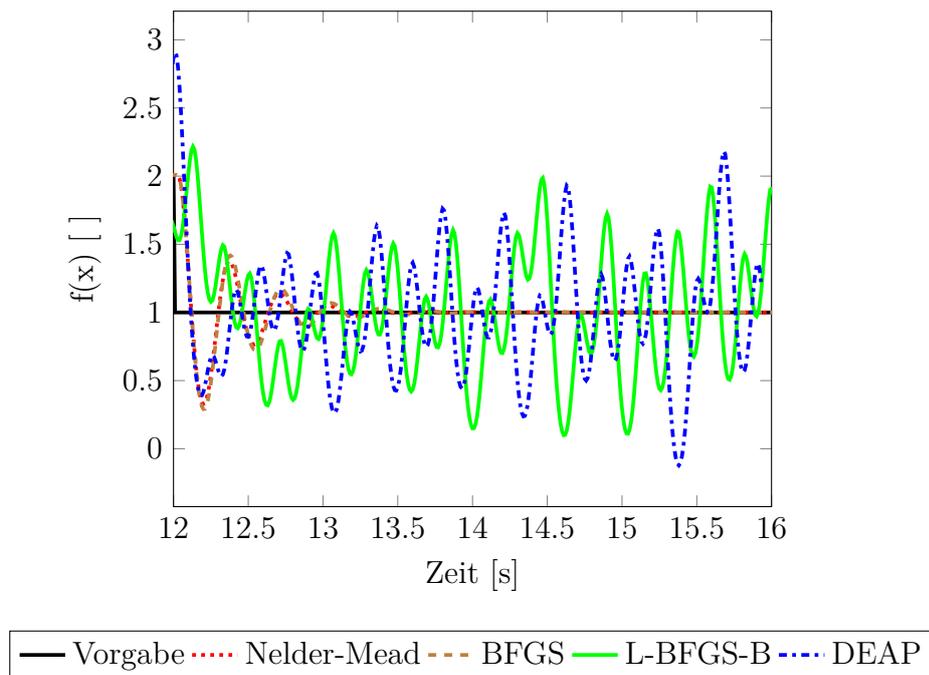


Abbildung 3.12: Ausschnitt aus Sprungfunktion von $12 \leq t \leq 16$

Im vierten und letzten Zeitabschnitt von der 12ten bis zur 16ten Sekunde liegt der Zielwert auf 1 (siehe Abb. 3.12). Die PID-Regler mit den optimierten Werten des Nelder-Mead und BFGS zeigen auch hier eine sinusförmige Schwingung, welche schnell abklingt. Die Abweichung von $0,02^\circ$ ist und bleibt nach der Sekunde 13,5 unterschritten. Bei den anderen beiden PID-Reglern ist analog zum dritten Zeitintervall wieder eine Überlagerung mehrerer Schwingungen sowie die Vergrößerungen der Amplituden zu erkennen.

Im zweiten Schritt wird untersucht, wie groß die positiven Sprünge sein können. Dazu wird beispielhaft ein Sprung auf 3° und ein Sprung 5° als Zielfunktion angesetzt, um zu sehen, ob sie die Regler sich innerhalb von 4 Sekunden auf diesen Wert näherungsweise einschwingen können.

Wie in Abb. 3.13 (links) zu sehen, liegen der Nelder-Mead und BFGS wieder nahe beieinander. Nachdem sie sich 2 Sekunden lang bei dem Wert $\theta = 0^\circ$ eingeschwingen haben, steigt der Zielwert sprunghaft auf $2,5^\circ$ an. Die Funktionen folgen dem Zielwert und schwingen sich knapp oberhalb des Zielwertes ein. Die absolute Abweichung am Ende des Zeitintervalls ist sehr gering. Ein Einschwingen um den Zielwert hat erfolgreich stattgefunden.

Der L-BFGS-B folgt auch der Sprungfunktion. Dabei ist eine Überlagerung mehrerer Sinusschwingungen zu erkennen. Es wird auch ein Abklingen der Schwingung gezeigt. Ein Einschwingen hat demnach mit größeren Amplitudenwerten stattgefunden.

Die Funktion, erzeugt von dem DEAP-Optimierer, zeigt nach dem Sprung des Zielwertes ein aufschwingendes Verhalten. θ schwingt zu immer größeren Werten auf und zeigt damit kein Annähern an den Zielwert, sondern facht die Schwingung an.

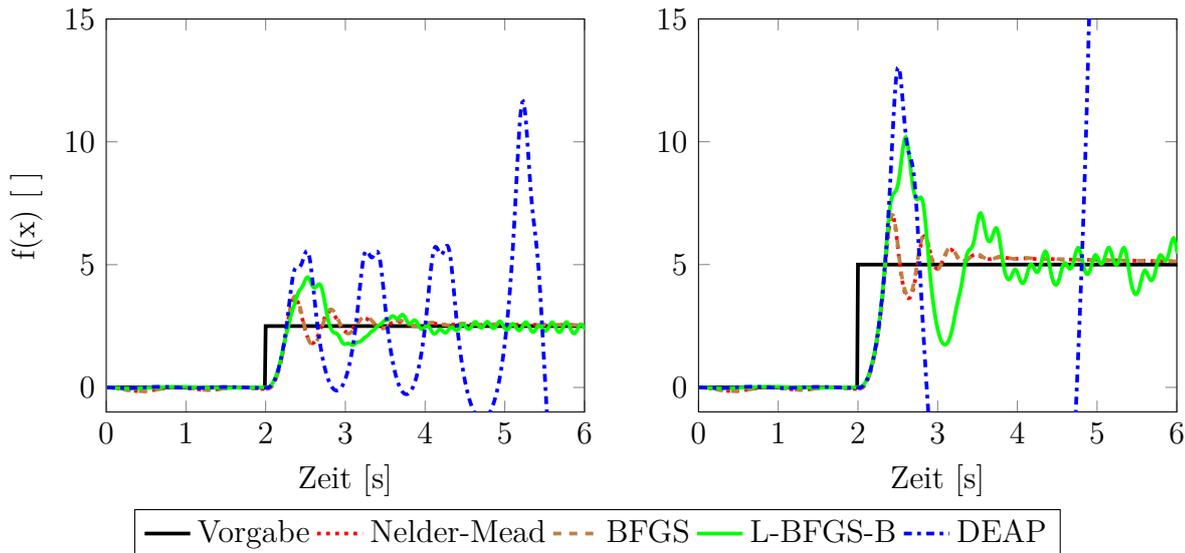


Abbildung 3.13: Sprungfunktion mit Schrittgröße von 3 (links) und 5 (rechts) als Zielfunktion und die Istwerte aller Optimierer

In Abb. 3.13 (rechts) ist ein ähnliches Verhalten zu erkennen. Die Absolutwerte der Abweichungen in den Amplituden steigen bei allen Optimierern an, deren Schwingungsverhalten bleibt allerdings gleich. Sowohl der Nelder-Mead als auch der BFGS zeigen ein Einschwingverhalten und weisen eine geringe Abweichung am Ende des Zeitintervalls auf.

Der L-BFGS-B setzt sich wieder aus einer Überlagerung mehrerer Sinusschwingungen zusammen, wobei ein Abklingverhalten zu erkennen ist.

Der DEAP zeigt wieder eine Überschwingung, die nun sofort nach dem Sprung eintritt und eine starke Aufschwingung zeigt. Da eine Sprunggröße von $2,5^\circ$ bei dem DEAP zu einem Aufschwingen führt, ist dieses Verhalten bei einer Sprunggröße von 5° auch so zu erwarten.

Zuletzt wird ein linearer An- und Abstieg als Zielfunktion gewählt. Diese ist in Abb. 3.14 zu sehen. Wie schon bei den Grafiken zuvor ist zunächst ein Absinken aller PID-Regler-Funktionen zu verzeichnen, welche ihre Ursache in der definierten Anfangsfluglage hat. Bei dem linearen Anstieg haben alle Antwortfunktionen nur wenig Abweichung von den Zielfunktionswerten. Erst kurz vor dem maximalen Wert von 5 weichen der L-BFGS-B und DEAP stärker von der Zielfunktion ab und zeigen ein aufschwingendes Verhalten. Dieses wird durch den Abfall der Zielfunktion unterbrochen. Entlang der Senkung ist ein Abklingen der Schwingungen beider Optimierer zu erkennen. Ab Sekunde 7 liegen beide wieder nahe der Zielwerte, schwingen aber sichtlich weiter.

Auch der Nelder-Mead und BFGS folgen mit ihren Werten denen der Zielfunktion. Eine Schwingung oder Abweichung ist dabei kaum zu erkennen.

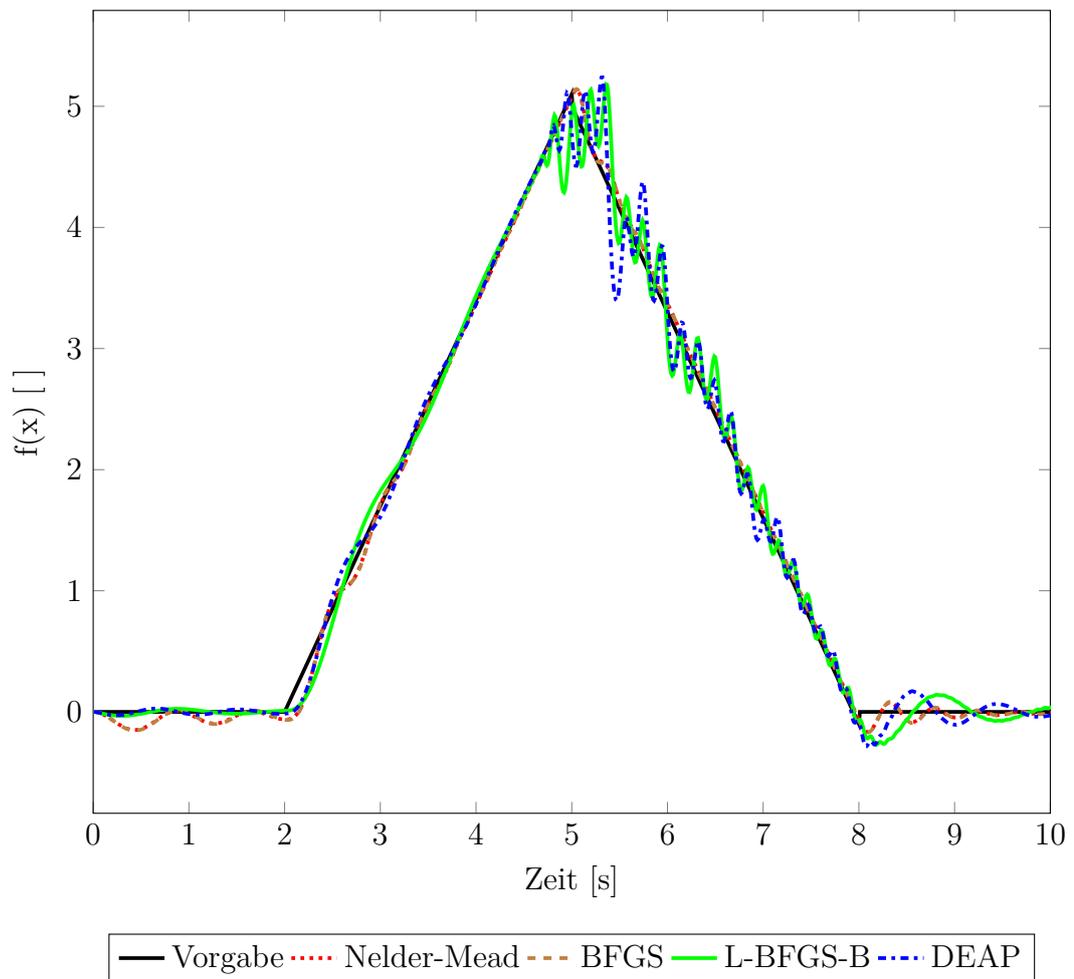


Abbildung 3.14: lineare Zielfunktion und die Istwerte aller vier Optimierer

Im letzten Abschnitt mit dem Zielwert gleich 0° schwingen sich alle Optimierer um diesen Wert ein. Alle zeigen ein Abklingen ihrer Schwingung mit wenig Abweichung vom Zielwert.

3.3 Gegenüberstellung der KIen

Es werden zwei KIen antrainiert, der DDPG und der TD3. Beide führen ein Training zur Steuerung über einen PID-Regler durch sowie eine direkte Steuerung durch die Verschiebung des Schwerpunkts des Flugobjekts entlang der körperfesten x-Achse. Unter dem Begriff "Antrainieren" wird im Bereich KI verstanden, dass diese sich Datensätze aneignet und ein neuronales Netzwerk erstellt. In diesem Fall sind das Simulationsdurchläufe und die Bewertung des Targetwertes. Die KI erlernt dabei nicht den Simulationsablauf, sondern bekommt explizite Ausgabewerte nach oder während einer Simulation.

In diesem Unterkapitel wird dargestellt, welche Ergebnisse die KIen bieten und ob das Trainieren der KIen als erfolgreich angesehen werden kann.

Das Trainieren einer KI kann nie als "abgeschlossen" angenommen werden. Im Allgemeinen ist zu sagen, je mehr Datensätze zur Verfügung stehen, umso ausgeprägter kann ein neuronales Netzwerk gefertigt werden und umso besser sind tendenziell auch die Ergebnisse.

Optimierungsvariable Θ

Bei dem Antrainieren zur Variable Θ hat der TD3 knapp 100.000 Durchläufe und der DDPG über 150.000 Durchläufe durchgeführt. Diese Anzahl liegt um einiges höher als bei den Optimierern und beide liefern minimierte Targetwerte bis $0,04063 \cdot 10^{-4}$ (DDPG) und $0,2196 \cdot 10^{-4}$ (TD3).

Beim Starten der KIen zur optimierten Steuerung eines Durchlaufs sind die Ergebnisse allerdings nicht zufriedenstellend. Der Targetwert beläuft sich auf einen Wert von 1019,38 und die Simulation kommt nach weniger als einer Simulationsflugsekunde an eines der Abbruchkriterien $\alpha > 1000^\circ$.

Die KIen haben demnach minimale Werte gefunden, die vergleichbar sind mit den Optimierern, beim Ausführen der Simulation entscheiden sich die KIen allerdings für eine ungünstige Variante. Es wird angenommen, dass dies der begrenzten Anzahl an Datensätzen geschuldet ist und die guten Minimalwerte als Ausnahme von der KI angesehen und dadurch nicht genutzt werden.

Die Simulation kann, wie bei den Optimierern, per Hand mit den optimierten PID-Werten gestartet werden. Dazu sind die minimalen Targetwerte (siehe Tab. 3.2) vom

Anwender auszulesen und die dazugehörigen PID-Werte in einer Simulation zu starten.

Beide KIen zeigen in ihren besten Minimaltargetwerten zwei starke Tendenzen. Zum einen gibt es viele Ergebnisse, bei denen der K_P -Wert stark erhöht von der Schätzung abweicht und gegen 2 geht. Die anderen Werte, K_I und K_D , sind auch erhöht und liegen im Bereich von 0,5 und 0,1. Diese Tendenz wird im Folgenden als Variante 1 bezeichnet.

Im zweiten Fall, der auch bei beiden KIen auftritt und für kleine Targetwerte sorgt, geht der K_I Wert gegen 2. Der K_P Wert bleibt nahe der Schätzung bei 0,2. Wie im Fall zuvor ist der K_D Wert oberhalb der Schätzung um 0,1. Dies soll Variante 2 genannt werden.

KI	K_P	K_I	K_D	Targetwert _{min}
DDPG _{var1}	2,000	0,500	0,101	$0,406 \cdot 10^{-5}$
DDPG _{var2}	0,240	2,000	0,119	$7,980 \cdot 10^{-5}$
TD3 _{var1}	1,754	0,443	0,111	$2,196 \cdot 10^{-5}$
TD3 _{var2}	0,203	1,601	0,145	$3,465 \cdot 10^{-5}$

Tabelle 3.2: Auszug aus optimierter Wertetabelle des DDPG und TD3 für die PID-Werte sowie die dazugehörigen Targetwerte

Das Antrainieren und Ausführen der KIen ist durch die fehlende positive Ausgabe als nicht erfolgreich einzustufen. Das Konzept der KI zur Optimierung eines PID-Reglers kann trotzdem als Alternative genutzt werden.

Die vorangegangenen vier Varianten (s.Tab. 3.2) wurden mit Hilfe der Zielfunktion aus Formel 3.1 simuliert. In Abb. 3.15 sind diese zusammen mit der Zielfunktion dargestellt.

Die Variante 1 zeigt im ersten Zeitintervall bei beiden KIen ein gutes Einschwingen um den Wert 0° , führt dann aber zu einer Überschwingung nach dem ersten Sprung, welche nach dem zweiten Sprung erhöht wird und nach 12 Sekunden zum Abbruchkriterium $\alpha > 1000^\circ$ führt und die Simulation beendet.

Die zweite Variante ähnelt dem Verlauf der L-BFGS-B und DEAP Optimierer. Auch hier ist ein gutes Einschwingen auf den Wert 0 im ersten Zeitintervall gegeben. Beim zweiten Zeitschritt nach dem ersten Sprung ist auch ein Schwingen um den Zielwert zu sehen, sowie ein Abklingen der Schwingung. Im dritten Zeitintervall liegt eine Überlagerung der Schwingung vor, die im Mittelwert immer noch um den Zielwert schwingt, ein Abklingen der Schwingung ist nicht zu erkennen. Diese Überlagerung ist auch im letzten Zeitintervall zu sehen. Außerdem ist ein leichtes Aufschwingen zu erkennen, da die maximalen Amplitudenwerte im Verlauf größer werden.

Aufgrund dieser beispielhaften Betrachtung und der Ähnlichkeit in der relativen Größe

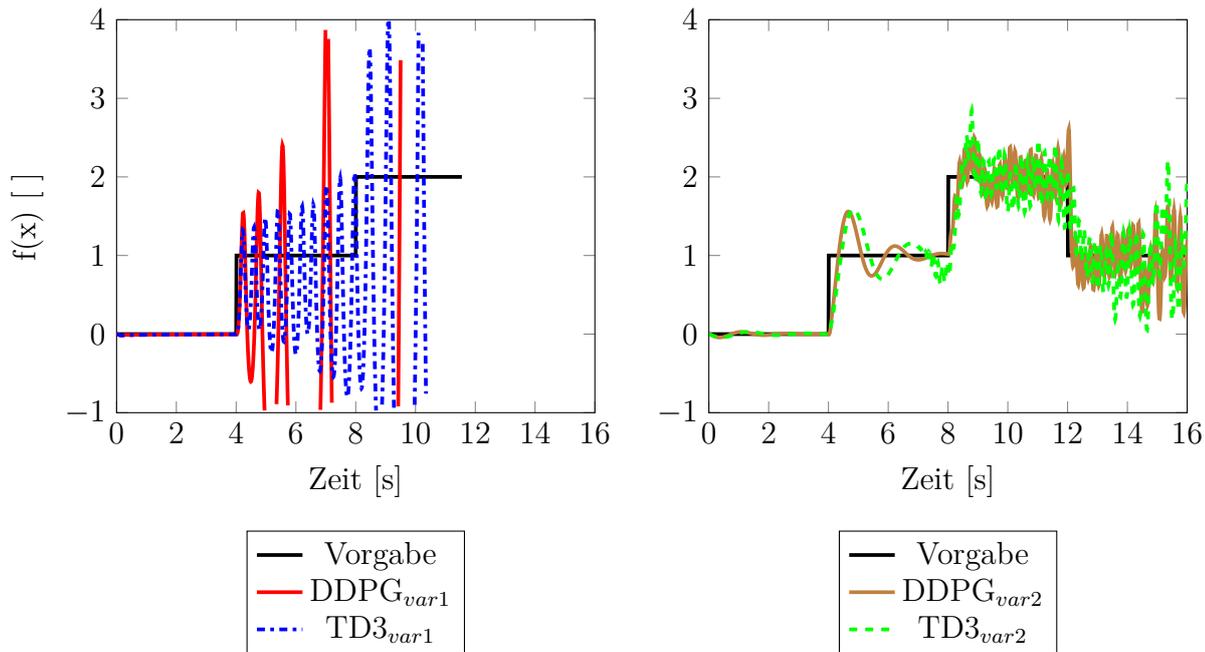


Abbildung 3.15: Sprungfunktion als Zielfunktion für optimierte PID-Werte der Klen DDPG und TD3 als Variante 1 (links) und Variante 2 (rechts)

der K_I -Werte in Bezug zu den anderen, ist davon auszugehen, dass sich beide Klen der zweiten Variante ähnlich zu den Ergebnissen des DEAPs oder dem L-BFGS-B verhalten.

3.4 Optimierungsvariable

In diesem Abschnitt soll untersucht werden, wie die Wahl der Variable auf die Optimierung Einfluss nimmt. Dafür wurde zunächst eine zweite Optimierung nach γ vollzogen, die die gleichen Anfangsdaten wie bei der Optimierung nach Θ hatte.

Danach folgte eine Optimierung nach der Höhe $H = 30\text{km}$. Zusammen mit der Optimierung nach Θ sollen alle drei durch ihre Optimierung zu einer größtmöglichen Reichweite führen. Die Ergebnisse dazu sind im ersten Unterkapitel aufgeführt.

Es wird vermutet, dass das Antrainieren einer KI für drei Variablen mehr Rechenleistung benötigt als nur für eine Variable. Deswegen wurden die Klen zusätzlich noch auf eine Steuerung trainiert, in welcher der Schwerpunkt entlang der körperfesten x-Achse des Flugobjekts direkt geregelt wird ohne Einsetzen eines PID-Reglers. Auf die Ergebnisse wird im zweiten Unterkapitel eingegangen.

3.4.1 Optimierungsvariablen θ , γ und Höhe

Zunächst soll untersucht werden, wie sich die Anzahl der Durchläufe je nach Variable verändert.

In Abb. 3.16 sind dazu die Durchlaufzahlen je Optimierer bei den drei Optimierungsvariablen aufgeführt. Es ist zu sehen, dass alle Optimierer ein recht unterschiedliches Verhalten bei verschiedenen Variablen haben.

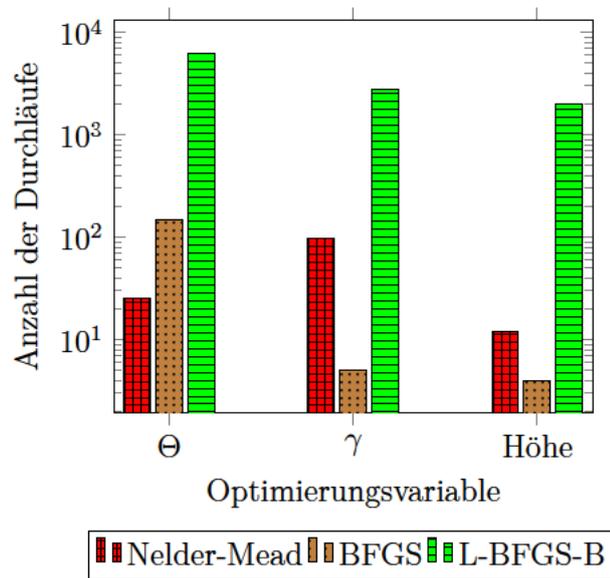


Abbildung 3.16: Variablenvergleich der Optimierer Nelder-Mead, BFGS, L-BFGS-B über die Anzahl der Durchläufe

Während der Nelder-Mead beim Optimieren nach γ weitaus länger braucht, ist die Anzahl von BFGS und L-BFGS-B stark gesunken. Der BFGS geht sogar beinahe auf 0, während der L-BFGS-B nur ein Drittel seiner alten Durchlaufanzahl bei der Optimierung nach θ aufweist.

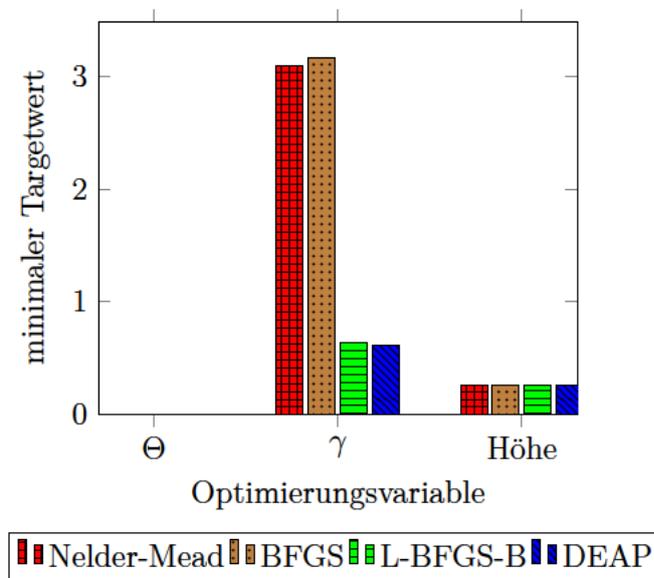


Abbildung 3.17: Variablenvergleich aller Optimierer über den minimalen Variablenwert aller Optimierungsvariablen

Bei der Optimierung zur Höhe sinkt die Durchlaufanzahl vom Nelder Mead wieder und auch der BFGS benötigt nur sehr wenige Durchläufe. Der L-BFGS-B bleibt analog zur Auslegung nach γ bei knapp über 2000 Durchläufen.

Der DEAP ist nicht abgebildet, da er bei jeder Variable nahezu die gleiche Anzahl an Durchläufen macht, weil diese durch den Anwender festgelegt wird.

Die minimierten Targetwerte haben sich stark verändert bei den Variablen γ und der Höhe. So ist in Abb. 3.17 ersichtlich, dass eine Optimierung nach γ bei allen Optimierern die größten Targetwerte liefert. Wie schon zuvor ist zu sehen, dass der L-BFGS-B und der DEAP weitaus kleinere optimierte Targetwerte haben.

Bei der Höhe sinken diese Werte wieder, kommen den optimierten Targetwerten nach der Optimierungsvariable θ aber nicht nahe. Alle Optimierer zeigen denselben optimierten Targetwert, was daran liegt, dass der PID-Regler physikalisch nur minimalen Einfluss auf eine Einschwingung um die Höhe hat.

Variable γ

Eine sprunghafte Zielfunktion für die Optimierung nach γ wurde nicht durchgeführt. Eine solche Untersuchung bringt keine weiteren Erkenntnisse, da sich der Anstellwinkel γ selten sprunghaft ändert.

Bei einer konstanten Zielfunktion $\gamma = 0^\circ$ sind die Unterschiede der optimierten PID-Regler in der Abb. 3.18 aufgeführt.

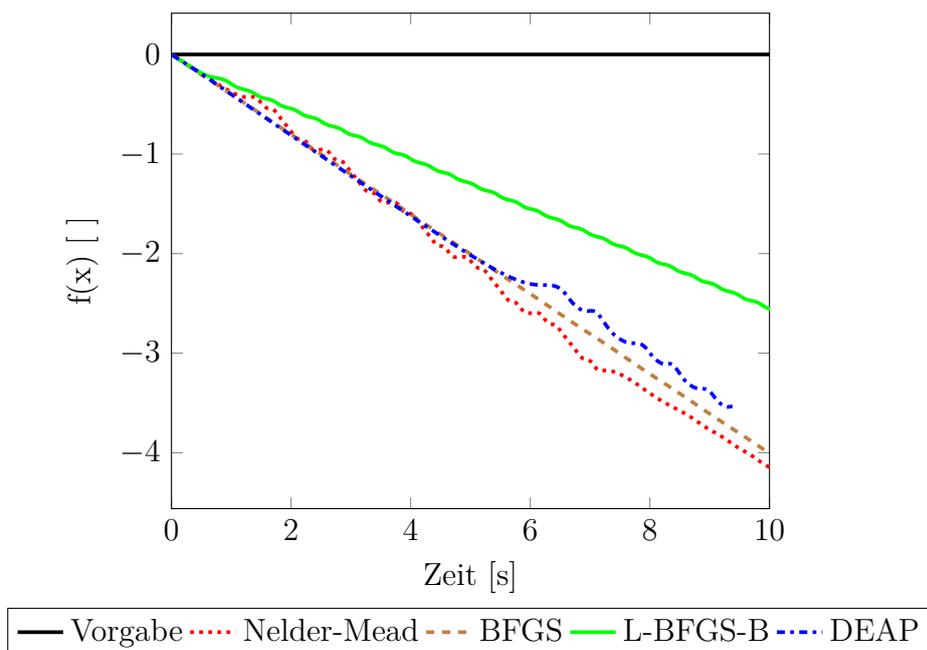


Abbildung 3.18: Verlauf von γ aller Optimierer bei konstanter Zielfunktion $\gamma = 0$ im Zeitintervall $t = 16$ bei einer Optimierung nach γ

Es ist zu sehen, dass kein Regler um den Zielwert schwingt wie bei der Optimierung nach Θ . Es wird vermutet, dass die Anfangssituation ein Steigen nicht zulässt, da nicht genügend Auftrieb erzeugt werden kann, was ein Fallen des Flugobjektes zur Folge hat und γ nur schwach beeinflusst. Einzig der Regler des L-BFGS-B hat etwas mehr Einfluss und lässt γ weniger schnell fallen, als die anderen drei Optimierer.

Variable Höhe

Auch bei dieser Variable wird keine sprunghafte Zielfunktion, sondern eine konstante Zielfunktion gewählt. Es ist zwar sinnvoll aufgrund eines Flugmanövers oder einer Böe von sprunghaften Zielwerten auszugehen, doch in Abb. 3.19 ist zu sehen, dass bei der Optimierung zur Höhe die Antwortfunktion unabhängig von den optimierten Regelgrößen ist.

Es gilt die Annahme, dass eine Schwingung um die Flughöhe nur dann möglich ist, wenn genügend Auftrieb erzeugt werden kann, um ein Steigen zu ermöglichen. Ein stabiler Flugzustand mit einer konstanten Höhe kann also nur dann zustande kommen, wenn mit α genauso viel Auftrieb wie Abtrieb erzeugt werden kann und dabei in einem Rahmen bleibt, welcher eine anliegende laminare Strömung ermöglicht. Geht dies nicht oder hat der Regler zu wenig Einfluss auf die zu optimierende Variable führt dies dazu, dass die Grafiken unabhängig von den PID-Regler-Werten alle gleich aussehen, da der Regler, unabhängig davon, wie er ausgelegt ist, keine Möglichkeit hat, in einen stabilen Zustand hineinzukommen.

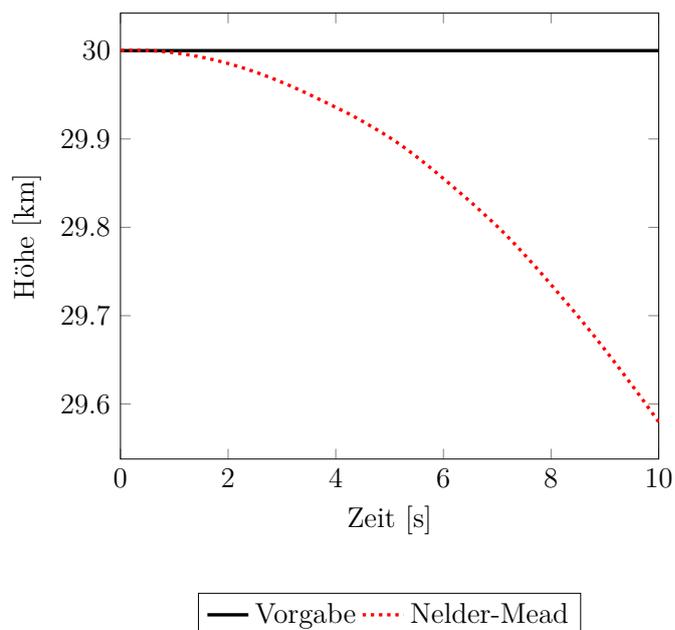


Abbildung 3.19: Vorgabe und Antwortfunktion des Nelder-Meads für den Zielwert $H = 30\text{km}$

Aus Zeit- und Kapazitätsgründen ist ein Antrainieren der Klen nach γ und der Höhe entfallen. Wichtig aufzuzeigen ist allerdings der Unterschied beim Antrainieren zu einer oder drei Variablen, weshalb im folgenden Unterkapitel ein weiteres Antrainieren der Klen zur direkten Steuerung des Schwerpunktes begutachtet wird.

Optimierung aus unterschiedlicher Höhe

Beim Wiedereintritt durchläuft ein Flugobjekt viele verschiedene Luftschichten. Gerade die Dichte, welche direkten Einfluss auf den Auftrieb hat, ändert sich dabei stark. In diesem Unterkapitel wird deswegen erläutert, welchen Einfluss unterschiedliche Anfangshöhen auf die Regelung hat.

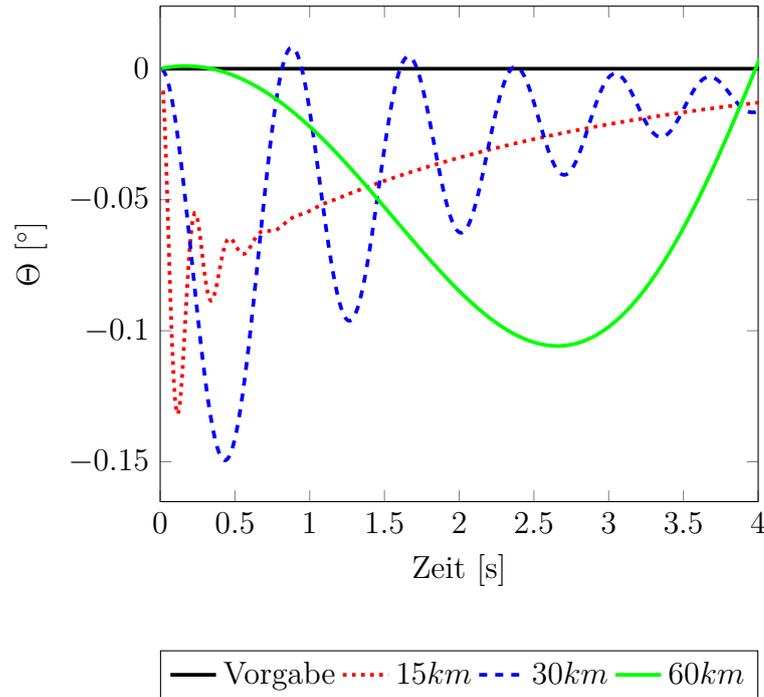


Abbildung 3.20: Simulationswerte des Nelder-Meads über verschiedene Höhen bei einer Optimierung nach Θ

In Abb. 3.20 sind mit Hilfe des PID-Reglers des Nelder-Meads mehrere Simulationen aus den Anfangshöhen 15km , 30km und 60km durchgeführt. Der PID-Regler ist hierbei auf eine Optimierung nach Θ ausgelegt.

Als Referenz ist die Simulation aus 30km aufgeführt, welche zuvor schon diskutiert wurde. Hier sieht man wie sich Θ um den Zielwert schwingt. Die Schwingung hat dabei eine Periodenlänge von weniger als einer Sekunde. Die Amplitudenwerte sinken dabei kontinuierlich über den gewählten Zeitbereich.

Bei dem Verlauf der Simulation aus 15km Höhe ist zu Beginn eine Schwingung mit einer weitaus kleineren Periodenlänge von ca. $0,2$ Sekunden zu sehen. Diese schwächt schnell ab und nähert sich dann asymptotisch an den Zielwert an.

Wird die ursprüngliche Anfangshöhe verdoppelt, vergrößert sich die Periodenlänge so sehr, dass das Zeitfenster von 4 Sekunden nicht ausreicht, um diese vollständig abzubilden. Die Schwingungsamplituden bleiben damit über einen längeren Zeitraum größer. Ob die Schwingung abklingt, kann nach dieser Simulation weder bestätigt noch widerlegt werden.

Eine Regelung zu einem Zielwert ist demnach aus unterschiedlichen Anfangshöhen möglich, da der Regler immer ein Einschwingverhalten um den Zielwert zeigt. Der Ver-

lauf der Schwingung um den Wert verändert sich dabei sehr. Tendenziell ist zu erwarten, dass die Schwingung in höheren Luftschichten langsamer und in niedrigeren Luftschichten schneller verläuft.

3.4.2 Optimierung ohne PID-Regler über die Verschiebung des Schwerpunktes

Aufgrund der Notwendigkeit großer Datensätze zur Entstehung eines neuronalen Netzwerkes wird vermutet, dass eine Regelung über einen Parameter für die KI schneller zu lernen ist. Deswegen werden beide KIen antrainiert auf eine Steuerung der Schwerpunktslage entlang der körperfesten x-Achse.

Wie zuvor schon erläutert, gelingt das Trainieren einer KI, bringt beim Ausüben aber keine optimierten Ergebnisse. Dies ist auch bei der Auslegung der direkten Steuerung des Schwerpunktes entlang der körperfesten x-Achse der Fall. Zu dieser Auslegung ist eine Simulation per Hand nicht ausführbar. Deswegen ist die Auslegung in dieser Richtung als erfolglos zu betrachten.

4 Anwendung

In diesem Kapitel wird beurteilt, wie die praktische Anwendung der Optimierer und KIen stattfinden kann und welche Vor- und Nachteile jeder Lösungsansatz mit sich bringt. Am Ende folgt eine Schlussfolgerung, nach welcher Variable mit welchem Optimierer am besten eine Entwurfsauslegung erfolgen sollte.

4.1 Plausibilitätsprüfung der Ergebnisse

Bei der Anwendung des PID-Wertes darf nicht ausschließlich nur der zu optimierende Wert betrachtet werden. In einer Flugsimulation sind auch andere Flugparameter sowie physikalische Grenzen entscheidend. Auf diese soll in der Plausibilitätsprüfung eingegangen werden.

4.1.1 Simulationsbeiwerte

In der Simulationskopplung von REENT und SOSE werden verschiedene Flugparameter errechnet und ausgegeben. Es wird sich an dieser Stelle auf die zwei Beiwerte α und die Staupunkttemperatur beschränkt. Für alle drei Optimierungsvariablen werden diese Flugparameter betrachtet, um im nächsten Unterkapitel darauf eingehen zu können, nach welchem Parameter es sinnvoll ist zu optimieren.

4.1.1.1 Anstellwinkel α

Der Anstellwinkel α ist eine entscheidende Größe bei der Beurteilung, ob ein stabiler Flugzustand vorliegt. Beim rapiden An- oder Absteigen über bzw. unter kritischen Werten ist davon auszugehen, dass das Flugobjekt nicht mehr ausregelbar ist, bis äußere Umstände sich ändern.

Die Schwerpunktverschiebung hängt direkt mit α zusammen und hat somit großen Einfluss auf ihn.

Optimierungsvariable Θ

In Abb. 4.1 sind die Verläufe von α der Optimierer Nelder-Mead und DEAP sowie der KI TD3_{var2} bei einer Optimierung nach $\theta = 0^\circ$ zu sehen. Die Verläufe der Optimierer BFGS und L-BFGS-B sind analog zu den vorher genannten. Der TD3 steht stellvertretend für

beide KIen, da auch deren Ergebnisse sich stark ähneln.

In Abb. 4.1 ist zu sehen, dass bei der Optimierung nach Θ der Anstellwinkel linear anwächst, was dem Wert γ zu schulden ist, da dieser sich linear absenkt. Da Θ nach 0° optimiert wurde und $\alpha = \theta - \gamma$ ist, ergibt sich dieses Verhalten. Dieses Anwachs ist bei allen Optimieren und KIen sichtbar.

Wenn γ weiter linear sinkt, bedeutet dies, dass der Flugzustand so lange als stabil angenommen werden kann, bis α einen kritischen Wert überschreitet und die Strömung am Flugkörper abreißt. Dieses Verhalten kann in der instabilen Fluganfangssituation durch getroffene Annahmen begründet liegen. Kommt der Flugkörper in tiefere Luftschichten oder wird die Anfangsgeschwindigkeit erhöht, findet das Absinken von γ langsamer statt.

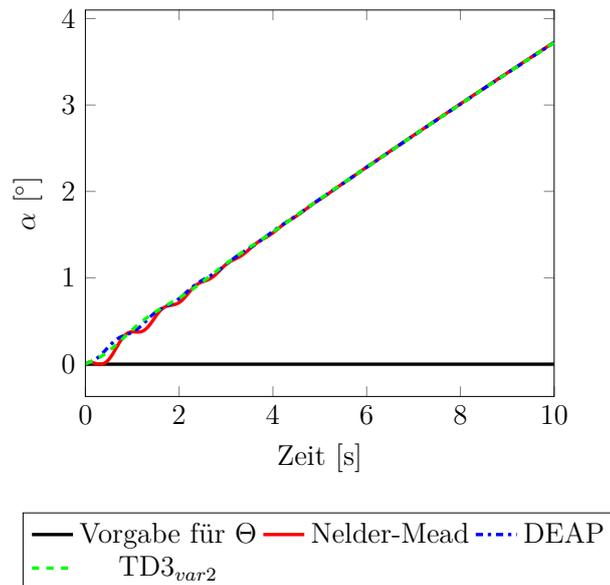


Abbildung 4.1: Verlauf von α für die Optimierung nach $\theta = 0$ beispielhaft für die Optimierer Nelder-Mead, DEAP und die KI $TD3_{var2}$

Optimierungsvariable γ

Bei der Optimierung nach γ sind in Abb. 4.2 die Verläufe von α aller Optimierer aufgezeigt. Es ist deutlich zu erkennen, wie unterschiedlich der Einfluss der PID-Werte auf den Anstellwinkel ist.

Der Nelder-Mead schwingt zunächst mehrfach um 180° mit sehr großen Amplituden, bis er ab der siebten Sekunde sich um 0° anfängt einzuschwingen. Dieses Überschwingen kann auf die instabilen Fluganfangsdaten zurückgeführt werden.

Der BFGS ist kaum zu erkennen, da sein Schwingungsverhalten so nahe an 0° liegt, dass die Amplituden sehr klein sind.

Der L-BFGS-B schwingt mit größeren Amplituden auch um 0° . Die Schwingung zeigt kein Abklingverhalten.

Der DEAP erzeugt beim α ein Abfall seines Wertes und überschreitet um die Sekunde 9 den Grenzwert von $\alpha > |1000|^\circ$. Die negative Steigung ist zunächst gering und nimmt um die fünfte Sekunde exponentiell zu. Es wäre möglich, dass sich das Überschlagen des Flugkörpers in niedrigeren Luftschichten durch die Zunahme der Dichte wieder einpendelt, wie bei dem Nelder-Mead zu sehen war.

Optimierungsvariable Höhe

Bei der Optimierung zur Höhe wurde in Kapitel 3 festgehalten, dass sich die Werte der Optimierer nicht voneinander unterscheiden. Dies ist auch für die anderen Beiwerte der Fall.

Bei der Optimierung zur Höhe fällt der Wert von α nach wenigen Sekunden rapide ab und fängt an sich um einen Wert von 500° einzuschwingen. Sollte sich die Schwingung um diesen Wert weiter verkleinern, käme das einem Flug gleich, bei dem das Flugobjekt um 180° um seine körperfeste y-Achse gedreht wird.

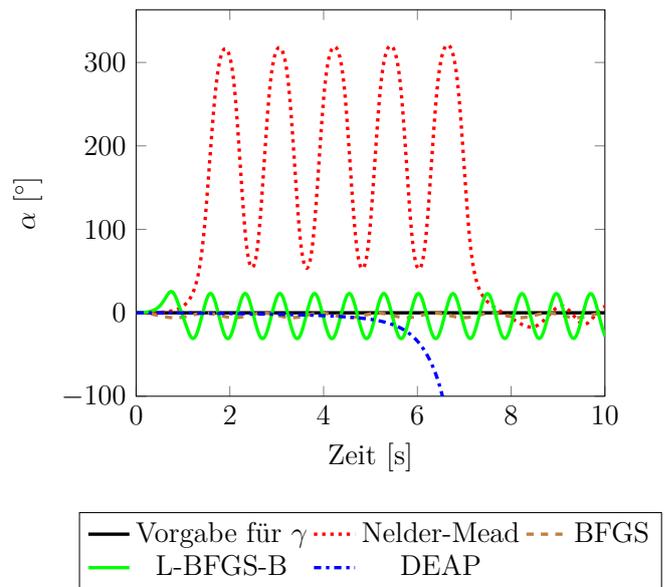


Abbildung 4.2: Verlauf von α für die Optimierung nach $\gamma = 0$ für alle Optimierer

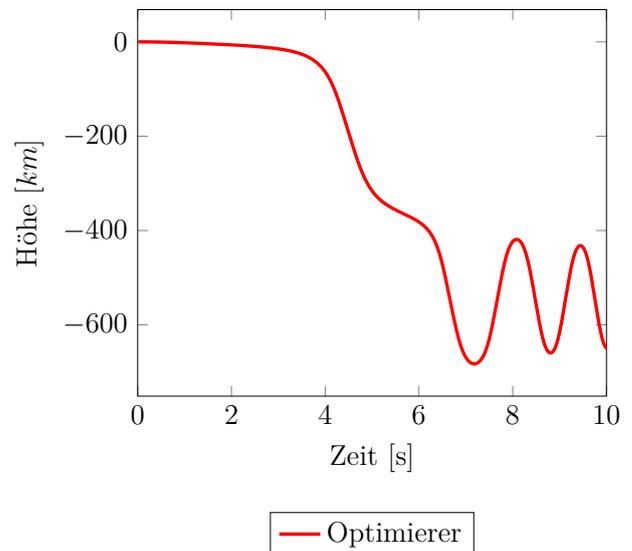


Abbildung 4.3: Verlauf von α für die Optimierung nach Höhe = 30km mit Nelder-Mead, stellvertretend für alle Optimierer

4.1.1.2 Temperatur im Staupunkt

Beim Wiedereintritt von Flugkörpern in die Erdatmosphäre bewegen sich diese Objekte mit einer sehr großen Geschwindigkeit. Dies ist auf die kinetische Energie zurückzuführen, die vorab notwendig war, um auf einer stabilen Umlaufbahn um die Erde zu bleiben. Ein Beispiel dafür ist die internationale Raumstation ISS. Beim Wiedereintritt in die Erdatmosphäre treten große Kräfte durch die Reibung zwischen den Luftmolekülen und der Oberfläche des Flugobjektes auf. Diese Reibung erzeugt eine Wärmeabgabe, die sich auf die Luft und die Oberfläche des Objektes aufteilt.

Auch mit guter Kühlung ist für jedes Material eine maximale Temperatur festgelegt, welche nicht überschritten werden darf, um ein Versagen des Flugobjektes zu verhindern. Die maximal anliegende Temperatur am Flugobjekt liegt im Staupunkt und wird hier mit t_{stau} abgekürzt.

Optimierungsvariable θ

In Abb. 4.4 ist zu sehen, dass sich die Staupunkttemperatur bei allen Optimierern gleich verhält, indem sie parabelförmig ansteigt. Der Zeitintervall beträgt 10s, in der sich die Temperatur im Staupunkt um mehr als 10 Kelvin steigert. Dies ist ein akzeptabler Anstieg für eine kurze Simulation. Während einer kompletten Mission ist dieser Wert erneut zu betrachten, da dieser von der Machzahl abhängig ist, welche sich stetig durch die Veränderung der Geschwindigkeit des Flugobjektes und des Schalls ändert. Entlang einer ganzen Mission sollte dieser Wert nicht über den kritischen Materialwert steigen, um ein Materialversagen der Struktur zu verhindern.

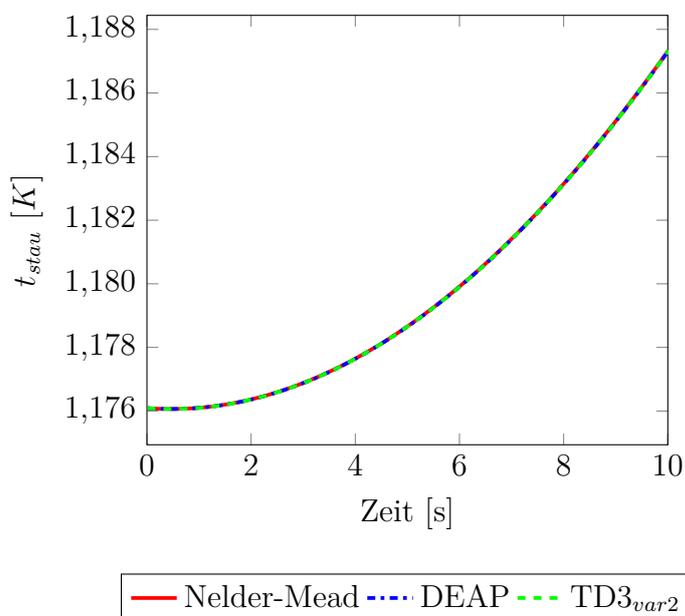


Abbildung 4.4: Verlauf von t_{stau} für die Optimierung nach $\theta = 0$, beispielhaft für die Optimierer Nelder-Mead, DEAP und die KI $TD3_{var2}$

Optimierungsvariable γ

Die Temperatur im Staupunkt ändert sich analog zu der vorhandenen Machzahl der Anströmung. Diese ist nicht als Grafik dargestellt worden, wird aber durch den Verlauf der Staupunkttemperatur in Abb. 4.5 widerspiegelt.

Der Nelder-Mead verzeichnet zunächst einen Temperaturverlust, weil das Flugobjekt durch das Pendeln um sich selbst viel an Geschwindigkeit verliert, die Machzahl also sinkt.

Der BFGS bringt α zu einer Schwingung um den Wert 0° mit nur kleinen Ausschlägen, wie in Abb. 4.2 zu sehen. Dies führt zu einer Zunahme der Temperatur im Staupunkt, weil die stabile Fluglage zu einer Vergrößerung der Machzahl führt.

Der Anstellwinkel α des L-BFGS-B schwingt auch um den Wert 0° (siehe Abb. 4.2), allerdings sind die Amplitudenausschläge größer als bei dem BFGS. Der stabile Flugzustand erzeugt damit mehr Widerstand als bei dem BFGS und verliert leicht an der Flugmachzahl.

Beim DEAP ist α im ersten Teil des Zeitintervalls nahe 0° , was in Abb. 4.5 zu sehen ist. Die Fluglage ist stabil, und analog zum BFGS steigt die Flugmachzahl. Nach dem Abreißen der Strömung verliert der DEAP rapide an Geschwindigkeit. Die Machzahl sinkt und damit auch die Temperatur im Staupunkt.

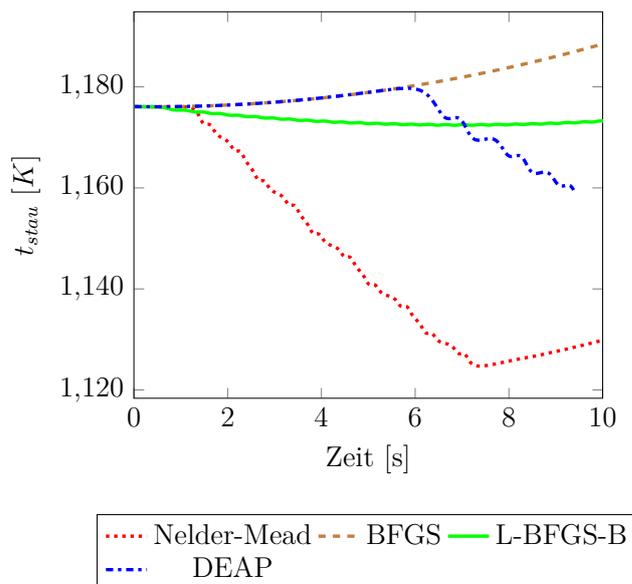


Abbildung 4.5: Verlauf von t_{stau} für die Optimierung nach $\gamma = 0$ für alle Optimierer

Optimierungsvariable Höhe

Wie zuvor mehrfach erwähnt, verhalten sich die Simulationsergebnisse bei der Optimierung zur Höhe stets gleich, sind also unabhängig vom Optimierer. Zunächst bleibt α nahe dem Wert 0° . Danach sinkt der Anstellwinkel rapide ab, was auf ein Abreißen der Strömung hindeutet. Durch das Drehen des Flugobjektes um die körperfeste y -Achse und die damit verbundene Machzahlabnahme, nimmt auch die Temperatur im Staupunkt ab. Dies ist auch in Abb. 4.6 zu sehen. Nach wenigen Sekunden Flugdauer fällt α schnell auf betragsmäßige hohe Werte, was sich auch in Abb. 4.6 wiederfindet, da die Temperatur, die in den ersten Se-

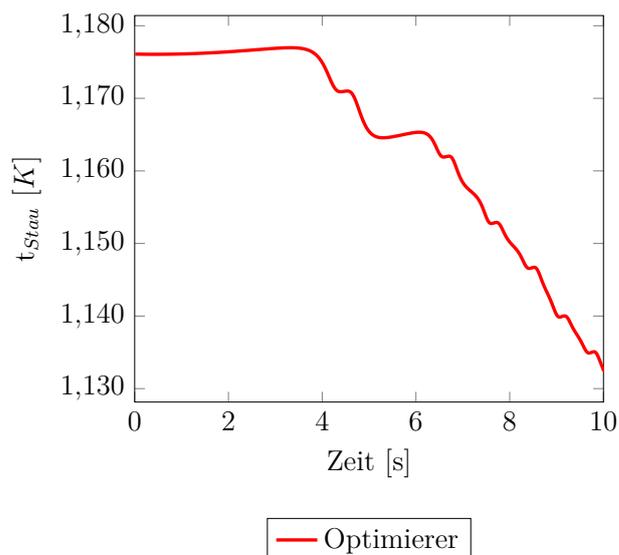


Abbildung 4.6: Verlauf von t_{stau} für die Optimierung nach Höhe = 0 für alle Optimierer

kunden noch recht konstant blieb, dann um mehrere Dutzende Kelvin absinkt.

4.1.2 Massenverschiebung

Ein weiterer wesentlicher Bestandteil der Optimierung ist das Setzen des Schwerpunktes entlang der körperfesten x-Achse, was einen direkten Einfluss auf die Anstellwinkel hat. Das Ändern dieses Schwerpunktes soll, wie in Kapitel 2 beschrieben, durch die Verschiebung einer Masse entlang besagter Achse geschehen. Als mögliche Geschwindigkeit der Massenverschiebung wurde zunächst eine sehr große Annahme getroffen. An dieser Stelle soll nun überprüft werden, ob die maximal genutzte Stellgeschwindigkeit physikalisch umsetzbar erscheint.

Simulation mit dem Zielwert $\Theta = 0^\circ$

In Tabelle 4.1 sind die durchschnittlichen Stellgeschwindigkeiten der einzelnen Optimierer und KIen bei der Simulation mit $\Theta = 0^\circ$ aufgelistet. Der Nelder-Mead und BFGS haben Stellzeiten von durchschnittlich 1 bis 3 Millimeter pro Zeitschritt, was umgerechnet 0,1 bis 0,3 $\frac{m}{s}$ entspricht. Die durchschnittliche Stellgeschwindigkeit des L-BFGS-B liegt bei unter 0,1 $\frac{m}{s}$. Allerdings ist hervorzuheben, dass gerade am Anfang der Simulation die Stellgeschwindigkeit weitaus höher liegt, bei bis zu 1,2 $\frac{m}{s}$. Diese verringert sich nach wenigen Zeitschritten. Das gleiche Bild tut sich auch bei dem DEAP auf. Auch hier liegt die maximale Stellgeschwindigkeit am Anfang sehr hoch, bei bis zu 1,2 $\frac{m}{s}$ und verringert sich nach wenigen Zeitschritten auf einen Wert, der im Durchschnitt unter 0,1 $\frac{m}{s}$ liegt.

Optimierer/KI	\varnothing Stellzeit [m/s]	max. Stellzeit [m/s]
Nelder-Mead	0,1	0,2
BFGS	0,1	0,2
L-BFGS-B	<0,1	1,3
DEAP	<0,1	1,3
DDPG _{var2}	<0,1	1,4
TD3 _{var2}	<0,1	1,1

Tabelle 4.1: Durchschnittliche und maximale Stellzeiten aller Optimierer und KIen bei der Simulation der optimierten PID-Werte nach Θ

Eine Stellzeit von 0,1 $\frac{m}{s}$ des Schwerpunktes wird für technisch umsetzbar gehalten, wohingegen eine Geschwindigkeit von 1,2 $\frac{m}{s}$ als unrealistisch angesehen werden kann. Es wird an dieser Stelle nicht auf eine genaue Umsetzung eingegangen.

Die KIen hatten bei dem Anlernen keinerlei Beschränkung was die Stellgeschwindigkeit anging. Der von der KI vorgegebene Schwerpunktwert für den nächsten Zeitschritt sollte nicht durch das Skript und eine Stellzeitbegrenzung verändert werden, um das Anlernen der KI zu beschleunigen. Trotz keinerlei Beschränkung kommen beide KIen beim Anlernen des PID-Reglers auf ähnliche Werte wie die Optimierer DEAP und LBFGS-B. Zu Beginn werden große Stellzeiten benutzt und nach wenigen Zeitschritten nur noch kleinere.

Zusätzlich zu der Geschwindigkeit soll in diesem Abschnitt das Gewicht der verschiebbaren Masse ermittelt werden. Um die notwendige verschiebbare Masse zu bestimmen, wird auf die Formel 2.8 zurückgegriffen. Die mögliche Schwerpunktverschiebung soll dabei $0,1m$ betragen. Dies ist der Wert, welcher über dem maximalen Wert der größten angewandten Verschiebung des Schwerpunktes in einer Simulation nach Θ auftritt. Bei einer Gesamtmasse von $280kg$ und einer möglichen Massenverschiebung von $0,8m$ ergibt sich daraus eine notwendige Masse von $35kg$.

Dies gilt nur für die Optimierung nach θ , da bei den anderen Auslegungen der Schwerpunktbereich von $0,8m$ von den Optimierern voll ausgenutzt wird und sich die notwendige Masse dem Gesamtgewicht annähert. Dies ist nicht erstrebenswert und erfordert dann weitere Steuermöglichkeiten wie das Einsetzen von Düsen oder Klappen.

Simulation mit sprunghafter Zielfunktion

Sobald die Zielfunktion sich sprunghaft ändert, versucht der PID-Regler so schnell wie möglich diesem Verlauf zu folgen. In einer Simulation bei einem Sprung von $2,5^\circ$ nutzt der Regler die mögliche Verschiebungsgeschwindigkeit von $10\frac{m}{s}$ zum Teil voll aus. Dies ist technisch nicht umsetzbar. Bei einer realistischen Auslegung des Reglers würden die Amplituden und Periodenlängen dadurch vergrößert bzw. verlängert werden, da der Regler nicht so schnell ausregeln kann wie es in der Simulation möglich ist.

4.2 Implementierung und Anwendung

In diesem Teil werden die wichtigsten Ergebnisse der Optimierer und der antrainierten KIen zusammengefasst. Danach soll eine Tendenz aufgezeigt werden, welcher Lösungsansatz im Bereich der Reglerauslegung in der Entwurfsphase am geeignetsten ist.

4.2.1 Optimierer

In der Auslegung eines Reglers über eine Minimierungsfunktion sind vier verschiedene Optimierer angewandt worden.

- Nelder-Mead
- BFGS
- L-BFGS-B
- DEAP

Alle Optimierer ließen sich gut in einem Python Skript implementieren. Besonders hervorgetan hat sich das Parallelrechnen des DEAP, welches eine erhebliche Zeitersparnis bedeuten kann. Die Optimierung mit dem DEAP ist auch die einzige, welche eine ungefähre Durchlaufanzahl festlegen lässt, da die Population und die Generationen vom Anwender selbst bestimmt werden muss. Dies setzt vom Anwender einen Erfahrungswert voraus, der aber innerhalb weniger Simulationsdurchläufe gewonnen werden kann.

Die drei Optimierer Nelder-Mead, BFGS und L-BFGS-B benötigen einen ungefähren Startwert. Dieser muss durch den Anwender selber festgelegt werden (siehe Kapitel 2.4.2 zum Nichlos/Ziegler-Verfahren). Diese Vorarbeit kostet Zeit, die beim Verwenden des DEAP entfällt, welcher mit Zufallswerten arbeitet und nur einen Wertebereich voraussetzt, in dem er nach Minima suchen soll. Dieser Wertebereich sollte sich nach Einarbeitung in das Thema ergeben oder alternativ groß gewählt werden. Dann muss die Anzahl der Populationen und Generationen gesteigert werden, um gute Ergebnisse erzielen zu können.

Der Nelder-Mead und BFGS bleiben in der Regel sehr nahe ihrer vorgegebener Schätzung, während der L-BFGS-B und DEAP mehr von den Vorgaben abweichen, welche mit deutlich kleineren Minima verbunden sein können.

Wie die Anwendung an einer Zielfunktion in Kapitel 3 allerdings zeigte, sind diese Minima sehr instabil und führen bei Schwankungen schnell zu einem Versagen des Reglers. Dies schränkt die Anwendung der Optimierungsergebnisse ein.

Die Wahl der Variablen hat großen Einfluss auf die Optimierung und die folgenden Ergebnissen. So ergeben sich bei der Optimierung nach γ unterschiedliche Lösungen, welche bei der Plausibilitätsprüfung auf einen instabilen Flugzustand hinweisen. Auch die Optimierung nach der Höhe erwies sich als kontraproduktiv, da die Ergebnisse sich nicht von einander unterscheiden und vermutet wird, dass die Regler mit den gegebenen Anfangsbedingungen nie in einen stabilen Flugzustand kommen und keine Möglichkeit gefunden wird, das Flugobjekt zu steuern.

Als Schlussfolgerung kommt nur eine Optimierung nach Θ in Frage, welche sowohl in der Auslegung als auch in der Praxis anwendbar ist.

Bei einer PID-Regler Auslegung zur einfachen Bahnsteuerung, welche unterschiedlichsten Sprüngen ausgesetzt ist, ist entweder der Nelder-Mead oder der BFGS zu wählen. Andererseits kann bei einer Auslegung zum schnellen Einpendeln eines festen Wertes auch der L-BFGS-B oder DEAP bevorzugt werden.

4.2.2 KI

Es wurden zwei KIen in zwei Anwendungen antrainiert. Sowohl der DDPG als auch der TD3 sollten eine PID Regelung auslegen und die Steuerung ohne Regler mit direktem Einfluss auf den Schwerpunkt lernen.

Die Anzahl der Durchläufe zum Trainieren bewegen sich in allen Fällen im unteren sechsstelligen Bereich und es wird vermutet, dass dies nicht ausreicht, um einen stabilen Flugzustand zu erlernen, da die Ergebnisse, welche von den KIen als Simulation ausgegeben werden, keinen stabilen Flugzustand zeigen.

Anhand der Simulationsumgebung für die Optimierer mit Hilfe einer Zielfunktion können die ausgegebenen PID-Werte der KIen ausgewertet werden. Es zeigen sich bei den Ergebnissen dieser PID-Werte zwei Tendenzen auf. Die Erste erweist sich als äußerst instabile und untaugliche Lösung. Die Zweite liefert ähnliche Resultate wie die Optimierer L-BFGS-B und DEAP. Es gibt ein schnelles Einpendeln mit kleinen Amplitudenwerten, wenn der Zielwert konstant ist, sie werden aber schnell instabil, wenn der Zielwert sich sprunghaft verändert.

Das zweite Antrainieren der direkten Schwerpunktsteuerung kann nur mit Hilfe der zur Verfügung gestellte Simulation der KI ausgewertet werden, welche keine sinnhafte Lösung zeigt und damit unbrauchbar ist.

5 Fazit

In dieser Arbeit wird das Problem zur primären Auslegung eines Reglers für den Entwurf eines Wellenreiters betrachtet. Dazu werden vier Optimierer verwendet und zwei KIen angelernt, um nach verschiedenen Variablen automatisiert optimale PID-Werte zu finden.

Jeder Ansatz bringt zur Optimierungsvariable Θ sinnhafte Lösungen. Nach Simulationsdurchläufen mit unterschiedlichen Zielfunktionen sowie dem Betrachten von α und der Staupunkttemperatur, kommt man zu dem Ergebnis, dass der Nelder-Mead sowie der BFGS für eine Optimierung nach Θ am sinnvollsten ist.

Die Optimierung nach anderen Variablen zeigt ungünstige Ergebnisse. α kann betragsmäßig sehr große Werte annehmen, was auf ein Überschlagen des Flugkörpers hinweist. Außerdem erfolgt kein Einpendeln um die zu optimierende Variable, da diese physikalisch nicht erhöhbar ist.

Die Anwendung einer KI ist aufgrund des großen Zeitaufwands zum Erstellen eines Netzwerkes ungeeignet. Auch wird nicht belegt, dass ein größeres Netzwerk zu einer sinnvollen Simulation führt, sondern ausschließlich als Annahme aufgezeigt.

In einer Zeit, in der neue Überschallflugzeuge wie "Lockheed Martin X-59", "VSS Unity" oder "Spike S-512 Supersonic Jet" entwickelt werden, werden effiziente Entwurfstrategien stets benötigt. Mit einer ersten Reglerauslegung in der Entwurfsphase findet die Industrie in den Optimierern Nelder-Mead und BFGS eine gute Möglichkeit Zeit einzusparen.

Aber auch die Verwendung einer KI sollte in zukünftigen Projekten nicht per se verworfen werden. Eine Implementierung von parallel erstellten Netzwerken könnte das Problem des zeitlich langwierigen Anlernens lösen und würde den zeitlichen Aufwand auf die bereitgestellte Rechenkapazität limitieren.

Literatur

- [1] Xaver Hafer und Gottfried Sachs. *Flugmechanik - Moderne Flugzeugentwurfs- und Steuerungskonzepte*. Berlin Heidelberg New York: Springer-Verlag, 2013. ISBN: 978-3-642-86730-9.
- [2] John David Anderson (Jr.) *Introduction to Flight* -. New York: McGraw-Hill Education, 2016. ISBN: 978-9-814-63618-6.
- [3] Cord-Christian Rossow, Klaus Wolf und Peter Horst. *Handbuch der Luftfahrzeugtechnik* -. M: Carl Hanser Verlag GmbH Co KG, 2014. ISBN: 978-3-446-43604-6.
- [4] URL: [https://de.wikipedia.org/wiki/Gleitzahl_\(Flugzeug\)](https://de.wikipedia.org/wiki/Gleitzahl_(Flugzeug)) (besucht am 09.10.2020).
- [5] Dr. D. Nguewo. *Vorlesungsskript: Vorlesung Flugmechanik 1*. HAW Hamburg.
- [6] URL: www.aerospaceweb.org/design/waverider/design.shtml (besucht am 06.08.2020).
- [7] URL: www.aerospaceweb.org/design/waverider/flow.shtml (besucht am 06.08.2020).
- [8] Dipl.-Ing. Thino Eggers. "Aerodynamischer Entwurf von Wellenreiter-Konfigurationen für Hyperschallflugzeuge". Diss. Von der Fakultät für Maschinenbau und Elektrotechnik der Technischen Universität Carolo-Wilhelmina zu Braunschweig, 21.März 1998.
- [9] URL: http://www.aircraftinformation.info/art_x43.htm#vehicles (besucht am 29.07.2020).
- [10] URL: <https://www.af.mil/About-Us/Fact-Sheets/Display/Article/104467/x-51a-waverider/> (besucht am 29.07.2020).
- [11] Tarik Barth Martin Sippel Hendrik Weihs. "Systematic Assessment of Advanced "Sharp-Edged" Lifting Body Re-entry Configurations". In: International Astronautical Congress 2009. 2009.
- [12] Dr.-Ing. Th. Eggers Dr. J. Bartolomé Calvo. "Simulation of Aerospace Vehicles based on a Coupling of Aerodynamics and Flight Mechanics". In: *Report of the Institute of Aerodynamics and Flow Technology* (2012).
- [13] Jürgen Teelar. "Model based attitude control design for SHEFEX III". In: ().
- [14] URL: <https://rn-wissen.de/wiki/index.php/Regelungstechnik> (besucht am 10.08.2020).
- [15] URL: [nachtrgentheorie](#).

- [16] URL: <https://docs.scipy.org/doc/scipy/reference/optimize.html> (besucht am 21.07.2020).
- [17] L. & Nocedal J. Byrd R.H.; Peihuang. “A limited-memory algorithm for bound-constrained optimization”. In: (1996). (Besucht am 12.08.2020).
- [18] URL: <https://deap.readthedocs.io/en/master/api/algo.html> (besucht am 12.08.2020).
- [19] Michael Haenlein Andreas Kaplan. “Siri, Siri, in my hand: Who’s the fairest in the land? On the interpretations, illustrations and implications of artificial intelligence”. In: *Business Horizons* 62, Issue 1 (Jan. 2019), S. 15–25. DOI: <https://doi.org/10.1016/j.bushor.2018.08.004>.
- [20] Robert A. Sottolare und Jessica Schwarz. *Adaptive Instructional Systems - First International Conference, AIS 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26–31, 2019, Proceedings*. Berlin, Heidelberg: Springer, 2019. ISBN: 978-3-030-22341-0.
- [21] URL: <https://www.sigs-datacom.de/trendletter/2018-06/4-ein-kompakter-ueberblick-zu-reinforcement-learning.html> (besucht am 22.07.2020).
- [22] URL: https://en.wikipedia.org/wiki/Reinforcement_learning#/media/File:Reinforcement_learning_diagram.svg (besucht am 09.10.2020).
- [23] URL: <https://www.alexanderthamm.com/de/blog/deep-learning-in-der-praxis/> (besucht am 12.08.2020).
- [24] URL: https://spinningup.openai.com/en/latest/spinningup/rl_intro.html (besucht am 13.08.2020).
- [25] URL: <https://towardsdatascience.com/reinforcement-learning-ddpg-and-td3-for-news-recommendation-d3cddec26011> (besucht am 09.10.2020).



Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „– bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] – ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Dieses Blatt, mit der folgenden Erklärung, ist nach Fertigstellung der Abschlussarbeit durch den Studierenden auszufüllen und jeweils mit Originalunterschrift als letztes Blatt in das Prüfungsexemplar der Abschlussarbeit einzubinden.

Eine unrichtig abgegebene Erklärung kann -auch nachträglich- zur Ungültigkeit des Studienabschlusses führen.

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: Multhaup

Vorname: Liona

dass ich die vorliegende Bachelorarbeit bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

Automatisierte Auslegung der Steuerung eines Wellenreiters

ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

- die folgende Aussage ist bei Gruppenarbeiten auszufüllen und entfällt bei Einzelarbeiten -

Die Kennzeichnung der von mir erstellten und verantworteten Teile der -bitte auswählen- ist erfolgt durch:

Hamburg

Ort

12.10.2020

Datum


Unterschrift im Original