

Entwurf eines robusten Kennfeldreglers für die Längsbewegung eines generischen Jet-Trainers mittels H_∞ -Loop-Shaping

Von der Fakultät Luft- und Raumfahrttechnik und Geodäsie der Universität Stuttgart
zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Abhandlung

Vorgelegt von

Tobias Martin Ulmer

aus Bietigheim-Bissingen

Hauptberichter: Prof. Klaus H. Well, Ph.D.
Mitberichter: Prof. Dr.-Ing. Reinhard Reichel

Tag der mündlichen Prüfung: 5. März 2007

**Institut für Flugmechanik und Flugregelung
Universität Stuttgart
2007**

Kurzfassung

Beim Entwurf eines Reglers über ein mathematisches Modell der Regelstrecke ist das Modell gegenüber der Realität im Normalfall vereinfacht und mit Unsicherheiten behaftet. Schlimmstenfalls führen diese Abweichungen zwischen Realität und verwendetem Entwurfsmodell beim späteren Betrieb des Reglers zur Destabilisierung der Regelstrecke.

Ziel der robusten Regelung ist es, einen Regler zu entwerfen, der beim Betrieb mit der realen Regelstrecke diese Abweichungen zwischen Modell und Realität toleriert.

Eine Möglichkeit zum Erreichen dieses Ziels ist es, die Modellfehler und Unsicherheiten ebenfalls zu modellieren und beim Reglerentwurf zu berücksichtigen. Problematisch ist hierbei, dass in der Flugregelung das reale, zu regelnde Flugzeug im Allgemeinen erst sehr spät vorliegt und in der Phase der ersten Reglerentwürfe daher nur wenige Kenntnisse über die zu erwartenden Unsicherheiten bestehen.

Das H_∞ -Loop-Shaping Verfahren bietet die Möglichkeit, einen robusten Regler zu entwerfen, ohne dass zum Zeitpunkt des Entwurfes explizite Kenntnisse über die zu erwartenden Unsicherheiten der Regelstrecke bekannt sein müssen. Der Regler stabilisiert den Kreis auf robuste Weise, indem er die zur Destabilisierung führende Unsicherheit der coprime-faktoriisierten Form des Streckenmodells, eine Art allgemeiner Unsicherheit, maximiert. Hierzu werden die Übertragungsfunktionen des offenen Regelkreises geformt und anschließend wird der Regelkreis über den H_∞ -Loop-Shaping-Regler geschlossen. Durch die Formung des offenen Regelkreises über Wichtungsfunktionen wird das gewünschte Verhalten im Zeitbereich erreicht.

Die vorliegende Arbeit befasst sich mit der Anwendung des H_∞ -Loop-Shaping-Verfahrens auf einen stabilitätserhöhenden Anstellwinkelfolgeregler für einen generischen Jettrainer basierend auf dem Flugzeug Mako von EADS. Neben Stabilität und Robustheit werden beim Entwurf Flugeigenschaftskriterien (Handling-Qualities) über die Vorgabe eines CAP (Control Anticipation Parameter) sowie einer Mindestdämpfung ζ_{AS} der Anstellwinkelschwingung berücksichtigt.

Das Entwurfsverfahren wird vorgestellt und diskutiert, verschiedene Implementierungsarten des Reglers werden präsentiert und bewertet.

Über einem Raster von Flugzuständen innerhalb der Envelope des Trainers erfolgt mittels beschränkter Optimierung ein automatisierter Reglerentwurf, bei dem unter Berücksichtigung der Handling-Qualities in Form von Nebenbedingungen die Robustheit jedes Einzelreglers maximiert wird. Die Regelung an Nicht-Entwurfspunkten erfolgt mittels einer Kennfeldregelung (Gain-Scheduling). Hierzu werden verschiedene Verfahren zur mehrdimensionalen Interpolation vorgestellt und diskutiert. Strategien zur Durchführung eines sprungfreien Scheduling sowie zum Betrieb des linearen Reglers mit der nichtlinearen Regelstrecke werden hergeleitet. Der entworfene Regler wird über lineare und nichtlineare Simulationen getestet, verschiedene Scheduling-Methoden werden verglichen.

Vorbereitend für den Entwurf der linearen Einzelregler wird ein nichtlineares Modell der Regelstrecke erstellt, anschließend erfolgt eine automatisierte Trimmung und Linearisierung des nichtlinearen Flugzeugmodells an den Flugzuständen des Entwurfsrasters unter besonderer Berücksichtigung der Implizit der Bewegungsgleichungen.

Abstract

When designing a controller by using a mathematical model of the plant, the model compared to reality is usually simplified and afflicted with uncertainties. In the worst case, these deviations between reality and utilized design-model lead to destabilization of the plant at the operation of the controller.

The objective of robust controller design is to create a controller, that tolerates the deviation between design-model and reality at operation.

One possibility to achieve this goal is to model the errors and uncertainties as well and to take them into account during the design procedure. The problem one encounters in this case with respect to flight control is, that the plane that is to be controlled is usually available very late and during the phase of first controller design there is only very little knowledge about the uncertainties to be expected.

The H_∞ -Loop-Shaping procedure offers the possibility, to design a robust controller without the necessity of explicit knowledge about the plant's expected uncertainties at the moment of design. The controller stabilises the closed-loop robustly, by maximizing the destabilizing uncertainty of the plant model's coprime factorization, where this form of uncertainty can be considered as a kind of general uncertainty. During the design procedure, weighting-functions before and after the plant are used to shape the open loop transfer-functions, then the loop is closed by the H_∞ -Loop-Shaping controller.

By shaping the open-loop via weighting-functions, the desired time-domain behaviour is achieved.

The present work deals with the application of the H_∞ -Loop-Shaping procedure on a stability augmenting angle-of-attack tracking-controller for a generic jet-trainer based on the aircraft Mako of EADS. Besides stability and robustness, handling-qualities are considered during the design procedure by demanding a specific *CAP* (Control Anticipation Parameter) and a minimum value ζ_{AS} for the damping of the aircraft's angle-of-attack oscillation.

The design procedure is discussed and different possibilities to implement the controller are presented.

Based on a grid of flight-conditions within the trainer's flight-envelope, an automatic design-procedure is effected during which the robustness of each single controller is maximised with respect to constraints representing handling quality requirements. Control at non-design flight-conditions is achieved by gain-scheduling. For this, different multidimensional interpolation procedures are introduced and discussed. Strategies for smooth scheduling and operation of the linear controller in connection with the nonlinear plant are derived. The designed controller is tested by linear and nonlinear simulations, different scheduling-methods are compared.

Preparing the design of the linear controllers, a nonlinear mathematical model of the aircraft is built. Afterwards linear models are derived at each controller-design flight-condition by trimming and linearizing the aircraft by using automated trim- and linearization routines with special consideration of the implicitness of the equations of motion.

Danksagung

Diese Arbeit entstand in wesentlichen Teilen während meiner Zeit als wissenschaftlicher Angestellter am Institut für Flugmechanik und Flugregelung der Universität Stuttgart in Zusammenarbeit mit EADS in München. Sie wäre ohne die Unterstützung der folgenden Personen nicht zu Stande gekommen:

Mein besonderer Dank gilt meinem Hauptberichter Herrn Professor Klaus H. Well, der es mir ermöglichte, konzentriert am Thema zu arbeiten. Er war immer ansprechbar und erreichbar war und bot mir mit seinen Anregungen stets eine große Hilfe.

Des Weiteren danke ich Herrn Professor Reinhard Reichel für die Übernahme der Zweitberichterstattung und das Interesse an meiner Arbeit. Durch sein kritisches Nachfragen erreichte er, dass ich vieles aus einem anderen Blickwinkel betrachtete und dabei eine Menge lernen konnte.

Vielen Dank schulde ich auch den Herren Dr. Heller, Beck und Paul von EADS für die hervorragende Zusammenarbeit und Unterstützung, für hochinteressante Gespräche, einen anregenden Ideenaustausch und eine stets angenehme Atmosphäre.

Außerdem sei allen ehemaligen Kolleginnen und Kollegen am IFR gedankt, für eine überaus angenehme Arbeitsatmosphäre, einen regen Austausch, fachliche und administrative Unterstützung und ganz einfach für so manche schöne Stunde. An dieser Stelle möchte ich insbesondere auch Herrn Christian Pertermann nennen, dessen Vorarbeit mir einen optimalen Einstieg in die Thematik ermöglichte.

Neben all denen, die mich fachlich unterstützten, möchte ich auch meine Familie nicht vergessen. Ohne sie wäre nichts, wie es ist, und deshalb danke ich ihr von Herzen.

Tamm, April 2007

Tobias Ulmer

Inhaltsverzeichnis

Kapitel 1

Einleitung	1
1.1 Vorgehensweise	1
1.2 Ausgangssituation.....	3
1.3 Stand der Technik	3

Kapitel 2

Die Regelstrecke	5
2.1 Das nichtlineare Modell und die Bewegungs-Differentialgleichungen.....	5
2.1.1 Das 6-DOF-Dynamikmodell.....	6
2.1.2 Das Aerodynamikmodell	12
2.1.3 Das Schubmodell	14
2.1.4 Das Atmosphärenmodell.....	16
2.1.5 Einsatzbereich des Jettrainers	16
2.1.6 Berechnung der Lastvielfachen	18
2.1.7 Klappenfahr Gesetze.....	19
2.1.8 Implementierung in Matlab Simulink.....	20
2.2 Bildung des linearen Entwurfsmodells	22
2.2.1 Trimmung: Motivation.....	22
2.2.2 Trimmung: prinzipielle Durchführung	24
2.2.3 Trimmung: Implementierung in Matlab	28
2.2.4 Linearisierung: mathematische Grundlagen	30
2.2.5 Beschreibung und Bestimmung der Längsbewegung.....	33
2.2.6 Beschreibung und Bildung der Kurzzeitnäherung.....	36
2.2.7 Linearisierung: Numerische Differentiation	38
2.2.8 Linearisierung: Implementierung in Matlab	39
2.2.9 Aktuator- und Sensormodell	42
2.2.10Reihenschaltung zum Entwurfsmodell.....	44
2.3 Analyse des Entwurfsmodells	44
2.3.1 Verlauf des Momentenbeiwerts	44
2.3.2 Stabilitätsanalyse anhand der Pole der Regelstrecke	46

Kapitel 3

Das Reglerentwurfsverfahren	51
3.1 Das H_∞ -Loop-Shaping-Verfahren.....	51
3.1.1 Berechnung des Reglers.....	52
3.1.2 Entwurfsziele	53

3.1.3	Wahl der Wichtungsfunktionen	56
3.1.4	Implementierung des linearen Reglers	57
3.1.5	Vergleich der Implementierungen	63
3.2	Das Gain-Scheduling-Verfahren	64
3.2.1	Wahl der Scheduling-Parameter	65
3.2.2	Interpolationsverfahren	66
3.2.3	Diskretisierung der Envelope zur Festlegung des Reglerentwurfsrasters	74
Kapitel 4		
Der automatisierte Reglerentwurf		77
4.1	Prinzip der Regleroptimierung	77
4.1.1	Berechnung der Zielgröße	78
4.1.2	Formulierung der beschränkenden Nebenbedingungen	83
4.1.3	Berechnung der beschränkenden Nebenbedingungen	84
4.1.4	Durchführung einer Optimierung in Matlab	86
4.2	Automatische Bearbeitung des Entwurfsrasters	87
4.2.1	Bearbeitungsreihenfolge und Startwertgenerierung	87
4.2.2	Ablauf der automatischen Bearbeitung und dabei beteiligte Matlab-Files	92
4.2.3	Tabellenerstellung	101
4.2.4	Auswertung und Nachbearbeitung	102
4.2.5	Kurzzusammenfassung des Entwurfsprozesses (Bedienungs- anleitung)	104
4.3	Zusammenbau zum linearen Gain-Scheduling-Regler	104
4.3.1	Implementierung in Matlab und Simulink	104
4.3.2	Simulation des linearen Regelkreises	106
4.3.3	Darstellung der Simulationsergebnisse	107
4.4	Auswertung der Entwurfsversion <i>V205</i>	108
4.4.1	Sprungantworten	109
4.4.2	Robustheitsmaß	111
4.4.3	Dämpfung und <i>CAP</i> -Wert	112
Kapitel 5		
Vom linearen zum nichtlinearen Regler		117
5.1	Implementierung des nichtlinearen Scheduling-Reglers	117
5.2	Übergang von relativen zu absoluten Größen	119
5.2.1	Referenzwert für Anstellwinkel	120
5.2.2	Referenzwert für Nickrate und Höhenruder	121
5.3	Sprungfreies Scheduling	123
5.4	Das nichtlineare Simulationsmodell	128
5.4.1	Startwertberechnung für nichtlineare Simulation	129
5.4.2	Ausgabe der Ergebnisse	130
5.5	Nichtlineare Simulationen	131
5.5.1	Nichtlineare Simulation an zufälligen Flugzuständen	131

5.5.2 Nichtlineare Simulation für verschiedene Kommandofolgen	137
5.5.3 Vergleich zwischen nichtlinearer und linearer Simulation	138
5.6 Vergleich zwischen ein- und ausgeschaltetem Scheduling.....	139
5.7 Vergleich verschiedener Scheduling-Methoden	140
5.8 Zusammenfassung zum nichtlinearen Regler.....	142
Kapitel 6	
Zusammenfassung	143
6.1 Durchgeführte Arbeiten und erzielte Resultate	143
6.2 Fazit und Ausblick.....	145
Literatur	147
Anhang A	
Simulink-Blockdiagramme	149
A.1 Nichtlineares Simulink-Modell für Trimmung und Linearisierung	149
A.2 Simulink-Blockdiagramm nichtlineares Simulationsmodell.....	150
A.2.1 Gesamtmodell (Ebene I)	150
A.2.2 Gain-Scheduling-Regler (Ebene II)	151
A.2.3 Gain-Scheduling-Regler (Ebene III).....	151
A.2.4 Blockdiagramm Zustandsrücksetzung (State Reset, Ebene IV)	152
Anhang B	
Struktogramme	153
B.1 Struktogramm der Trimmroutine <code>trimenvqdyn</code>	153
B.2 Struktogramm der Linearisierungsfunktion <code>linfun10</code>	155
B.3 Baumstruktur der Wiederversuchslogik	158
Anhang C	
CAP-Bestimmung	159
C.1 Erweiterte MIL-Forderungen für die Auslegung gemäß <i>CAP</i> nach Level 1*	159
Anhang D	
Lineares Entwurfsmodell	161
D.1 Lineares Entwurfsmodell, Regler im Vorwärtszweig.....	161
D.1.1 Zustandsraumdarstellung	161
D.1.2 Blockdiagramm.....	162
D.2 Lineares Entwurfsmodell, Regler in Beobachterdarstellung mit Zustandsvektorrückführung.....	163
D.2.1 Zustandsraumdarstellung	163
D.2.2 Blockdiagramm.....	164
D.3 Lineares Entwurfsmodell, Regler im Rückwärtszweig	165
D.3.1 Zustandsraumdarstellung	165

D.3.2 Blockdiagramm	166
Anhang E	
Durchgeführte Reglerentwürfe (Auswahl)	169
E.1 Tabelle der Einstellungen des automatischen Entwurfs	169

Bildverzeichnis

Abb. 1.1:	gewichtete Strecke und coprime-faktorierte Form mit Unsicherheiten.....	2
Abb. 2.1:	Steuerflächen des Jettrainers Mako	9
Abb. 2.2:	Envelope des Trainers in Ma - h -Darstellung	17
Abb. 2.3:	Envelope des Trainers in Ma - q_{dyn} -Darstellung.	18
Abb. 2.4:	Fahrgesetz für Vorder- und Hinterkantenklappe.....	20
Abb. 2.5:	Annäherung der Ableitung durch eine zentrale Differenz.....	38
Abb. 2.6:	Bode-Plot der Aktuator-Übertragungsfunktion mit Kerbfilter	43
Abb. 2.7:	Blockschaltbild des Entwurfsmodells	44
Abb. 2.8:	Momentenbeiwert über Anstellwinkel für verschiedene Machzahlen, $h = 1000\text{m}$	45
Abb. 2.9:	Momentenbeiwert über Anstellwinkel, verschiedene Höhen, $Ma = 0,3$	45
Abb. 2.10:	Momentenbeiwert über Anstellwinkel für verschiedene Staudrücke, $Ma = 0,3$	46
Abb. 2.11:	Momentenbeiwert über Anstellwinkel für verschiedene Machzahlen, $q_{dyn} = 5000\text{Pa}$	46
Abb. 2.12:	Realteile von Phygoid- und AS-Polen. Vergleich IFR-EADS-Daten.....	48
Abb. 2.13:	Vergleich AS-Pole aus Längsbewegung und Kurzzeitnäherung	48
Abb. 2.14:	AS-Pole über Anstellwinkel für verschiedene Staudrücke, $Ma = 0,3$	49
Abb. 2.15:	Realteil AS-Pole über Staudruck für verschiedene Anstellwinkel und Machzahlen.....	49
Abb. 2.16:	Prozentuale relative Abweichungen zw. Realteilen der AS-Pole und Näherungsparabeln	50
Abb. 2.17:	AS-Pole über Anstellwinkel für verschiedene Machzahlen, $q_{dyn} = 5000\text{Pa}$..	50
Abb. 3.1:	Standard H_∞ -Loop-Shaping Blockdiagramm	52
Abb. 3.2:	Robustheitsgrenzen im Nichols-Diagramm.....	54
Abb. 3.3:	Berechnung der Stabilitätsreserve ρ im Nichols-Diagramm	54
Abb. 3.4:	Reglerimplementierung als Beobachter mit Zustandsvektorrückführung.....	58
Abb. 3.5:	Regler im Rückwärtszweig mit beobachterähnlicher Struktur	60
Abb. 3.6:	Regler im Rückwärtszweig.....	61
Abb. 3.7:	Regler im Vorwärtszweig	62
Abb. 3.8:	Regler im Vorwärtszweig mit beobachterähnlicher Struktur.....	62
Abb. 3.9:	Anstellwinkelverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen	63
Abb. 3.10:	Nickratenverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen	64

Abb. 3.11: Ruderstellungsverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen	64
Abb. 3.12: zweidimensionale Interpolation	66
Abb. 3.13: eindimensionale Interpolation in y -Richtung	67
Abb. 3.14: mehrdimensionale Interpolation durch Dreiecks-Ebenen-Methode	68
Abb. 3.15: Verlauf der Berechnungszeiten für klassische und DE-Methode.....	72
Abb. 3.16: Vergleich der Interpolationsarten: einsetzbarer Bereich	74
Abb. 3.17: interpolierbarer Bereich in Abhängigkeit der Rasterung	75
Abb. 3.18: Dreidimensionales Entwurfsraster	76
Abb. 4.1: Sektoren des Nichols-Diagramms.....	78
Abb. 4.2: zu betrachtende Punkte zur Bestimmung des Berührungspunkts in Abhängigkeit der Rate r zur Reduktion der Zahl der betrachteten Punkte pro Durchlauf ...	81
Abb. 4.3: Berührungspunkt auf Kreisbahn - Knicke im Verlauf des Robustheitsmaßes.....	82
Abb. 4.4: Annäherung des Nichols-Diamanten durch Ellipse	82
Abb. 4.5: Startwertabhängigkeit der Optimierung - verschiedene gültige Lösungen....	88
Abb. 4.6: Bearbeitungsreihenfolge für mäanderförmige Abarbeitung des Entwurfsrasters.....	90
Abb. 4.7: Bearbeitungsreihenfolge bei teilweise mäanderförmiger Abarbeitung.....	91
Abb. 4.8: Bearbeitungsreihenfolge bei bürstenförmiger Abarbeitung.....	91
Abb. 4.9: Struktogramm des Steuerungsskripts regler_opt11_DivAlgAutoV11	93
Abb. 4.10: Auswertung der Ergebnisse automatischer Reglerentwurf $V205$	103
Abb. 4.11: Interpolation mit integrierter und separater Index-Suche	105
Abb. 4.12: Darstellung der untersuchten Sets zufälliger Flugzustände	108
Abb. 4.13: Simulation der Sprungantwort der Sets zufälliger Flugzustände	109
Abb. 4.14: farbliche Kennzeichnung simulierter Flugzustände.....	110
Abb. 4.15: Übersicht über erzielte Robustheitsmaße für Sets zufälliger Flugzustände..	111
Abb. 4.16: Verteilungen der erreichten AS-Dämpfungen für Sets zufälliger Flugzustände.....	113
Abb. 4.17: Verteilungen der erreichten CAP -Werte für Sets zufälliger Flugzustände...	114
Abb. 4.18: Verteilungen der erreichten $BestCAP$ -Werte für Sets zufälliger Flugzustände.....	116
Abb. 5.1: beispielhafter Aufbau einer Matrix mit veränderlichen Elementen.....	118
Abb. 5.2: Übergang zwischen relativen und absoluten Größen	120
Abb. 5.3: Bildung der Referenzwerte für q und h	122
Abb. 5.4: prinzipieller Aufbau des nichtlinearen Simulationsmodells	128
Abb. 5.5: Flugzustandsverlauf für beispielhafte nichtlineare Simulation.....	132
Abb. 5.6: simulierbare und nicht-simulierbare Start-Flugzustände ($t=25s$).....	132
Abb. 5.7: Sprungantworten nichtlinearer Simulationen an zufälligen Flugzuständen.	133
Abb. 5.8: Verteilung der Anstiegszeit für nichtlineare Simulation von Sets zufälliger Flugzustände.....	134

Abb. 5.9:	Verteilung der Überschwingweite für nichtlineare Simulation von Sets zufälliger Flugzustände.....	134
Abb. 5.10:	Überschwingweite und Anstiegszeit in 3D-Envelope für nichtlineare Simulation der Sets 1 bis 6.	135
Abb. 5.11:	simulierte Flugzustände mit Flächen konstanter Geschwindigkeitsableitung	136
Abb. 5.12:	Anstellwinkel-, Machzahl- und Staudruckverlauf für treppenförmiges Kommando	137
Abb. 5.13:	Anstellwinkel-, Machzahl- und Staudruckverlauf für rampenförmiges Kommando	138
Abb. 5.14:	Vergleich zwischen linearer und nichtlinearer Simulation für treppenförmiges Kommando	139
Abb. 5.15:	Vergleich zwischen Simulation mit komplettem Scheduling und α -Scheduling	139
Abb. 5.16:	Vergleich zwischen Simulation mit und ohne Scheduling	140
Abb. 5.17:	Vergleich verschiedener Scheduling-Methoden für treppenförmigen Eingang	141
Abb. 5.18:	Vergleich verschiedener Scheduling-Methoden für treppenförmigen Eingang	141
Abb. A.1:	Modell der nichtlinearen Modellierung für Trimmung und Linearisierung.	149
Abb. A.2:	Simulink-Blockdiagramm der Ebene I des nichtlinearen Simualtionsmodells.....	150
Abb. A.3:	Simulink-Blockdiagramm des nichtlinearen Gain-Scheduling-Reglers (Ebene II).....	151
Abb. A.4:	Simulink-Blockdiagramm des nichtlinearen Gain-Scheduling-Reglers (Ebene III).....	151
Abb. A.5:	Simulink-Blockdiagramm der Zustandsrücksetzung des Gain-Scheduling-Reglers (Ebene IV)	152
Abb. B.1:	Nassi-Shneiderman Struktogramm der Trimmroutine <code>trimenvqdyn</code> , Teil 1	153
Abb. B.2:	Nassi-Shneiderman Struktogramm der Trimmroutine <code>trimenvqdyn</code> , Teil 2	154
Abb. B.3:	Struktogramm der Linearisierungsfunktion <code>linfun10</code> , Teil 1	155
Abb. B.4:	Struktogramm der Linearisierungsfunktion <code>linfun10</code> , Teil 2	156
Abb. B.5:	Struktogramm der Linearisierungsfunktion <code>linfun10</code> , Teil 3	157
Abb. B.6:	Baumstruktur der Wiederversuchslogik des automatischen Reglerentwurfs.....	158
Abb. C.1:	Diagramm zur Reglerauslegung gemäß <i>CAP</i> nach Level 1*	159
Abb. D.1:	Blockdiagramm des linearen Entwurfsmodells, Regler im Vorwärtszweig.	162
Abb. D.2:	Blockdiagramm des linearen Entwurfsmodells, Regler in Beobachterdarstellung	164
Abb. D.3:	Blockdiagramm des linearen Entwurfsmodells, Regler im Rückwärtszweig.....	166
Abb. E.1:	Tabelle der Einstellungen des automatischen Reglerentwurfs	169

Tabellenverzeichnis

Tabelle 2.1:Komponenten des Zustandsvektors	7
Tabelle 2.2:Komponenten des Steuervektors	9
Tabelle 2.3:Eingänge des AeroData-Moduls ADM4.0	13
Tabelle 2.4:Eingänge des Schubmoduls	15
Tabelle 2.5:Einsatzbereich des Jettrainers	17
Tabelle 2.6:Übersicht über Module des nichtlinearen Modells und zugehörige s-functions	21
Tabelle 2.7:Zusammensetzung des Modell-Eingangsvektors und Ausgangsvektors	21
Tabelle 2.8:Trimmfall 1, gegebene und unbekannte Größen, zu lösende Gleichungen	26
Tabelle 2.9:Trimmfall 2, gegebene und unbekannte Größen, zu lösende Gleichungen	26
Tabelle 2.10:Trimmfall 3, gegebene und unbekannte Größen, zu lösende Gleichungen ..	27
Tabelle 2.11:Trimmfall 4, gegebene und unbekannte Größen, zu lösende Gleichungen ..	27
Tabelle 2.12:Schema über bekannte und unbekannte Elemente sowie zu verwendende Gleichungen	28
Tabelle 2.13:Zusammensetzung der Spalten der Ausgabevariablen <i>trimarray</i>	30
Tabelle 2.14:Zusammensetzung der Spalten der Ausgabevariablen <i>linarray</i>	42
Tabelle 2.15:Variablen der Linearisierungsdaten-Workspaces	42
Tabelle 4.1: Bearbeitungsreihenfolge und Verfahren zur Startwertbestimmung.....	92
Tabelle 4.2:Startwertgenerierung in Abhängigkeit des Wertes von <i>StartFlag</i>	98
Tabelle 4.3:Zusammensetzung und Bedeutung der Variablen <i>RetryCaseSingle</i>	99
Tabelle 4.4:Ergebnisse der Entscheidungsfindung	100
Tabelle 4.5:Definition der untersuchten Sets zufälliger Flugzustände	108
Tabelle 4.6:Auswertung Sprungantworten hinsichtlich Anstiegszeiten und Überschwingweiten	110
Tabelle 4.7:minimale Robustheitsmaße und zugehörige Flugzustände	112
Tabelle 4.8:Verteilungen von Dämpfung und <i>CAP</i> für Sets zufälliger Flugzustände	115
Tabelle 5.1:Auswertung Sprungantworten hinsichtlich Anstiegszeiten und Überschwingweiten	136

Bezeichnungen

Großbuchstaben

Symbol	Bedeutung	Einheit
A	Systemmatrix	$[-]^a$
B	Eingangsmatrix	$[-]$
C	Ausgangsmatrix	$[-]$
$C_{m, \alpha}$	Momentenbeiwert	$[-]$
D	Durchgangsmatrix	$[-]$
E	Hilfsmatrix Linearisierung	$[-]$
F	Reglermatrix Rückführung	$[-]$
G	Übertragungsfunktion Strecke	$[-]$
H	Reglermatrix Beobachterfehler	$[-]$
	Hilfsmatrix Linearisierung	$[-]$
I	Einheitsmatrix	$[-]$
	Trägheitstensor	$[\text{kg m}^2]$
J	Hilfsmatrix Linearisierung	$[-]$
K	Übertragungsfunktion Regler	$[-]$
L	Momentenkomponente x -Achse	$[\text{Nm}]$
M	Coprim-Faktor Strecke	$[-]$
	Transformationsmatrix	$[-]$
	Magnitude, Betrag Übertragungsfunktion	$[\text{dB}]$
M	Momentenkomponente y -Achse	$[\text{Nm}]$
Ma	Machzahl	$[-]$
N	Coprim-Faktor Strecke	$[-]$
N	Momentenkomponente z -Achse	$[\text{Nm}]$
O	Hilfsmatrix Linearisierung	$[-]$
P	Optimierungsparameter	$[-]$
\vec{v}	Geschwindigkeitsvektor	$[\text{m/s}]$
V	Vorfilter	$[-]$
W_1	Wichtungsfunktion 1	$[-]$
W_2	Wichtungsfunktion 2	$[-]$
X	Kraftkomponente x -Achse	$[\text{N}]$
	Hilfsmatrix Reglerentwurf	$[-]$

Symbol	Bedeutung	Einheit
Y	Kraftkomponente y -Achse	[N]
Z	Kraftkomponente z -Achse	[N]
	Hilfsmatrix Reglerentwurf	[-]

a. [-]: Keine Einheit bzw. verschiedene Einheiten möglich

Kleinbuchstaben

Symbol	Bedeutung	Einheit
b	Optimierungsnebenbedingungen	[-]
e	Regelabweichung	[rad]
g	Erdbeschleunigung	[m/s ²]
h	Flughöhe	[m]
	Schrittweite numerische Differentiation	[-]
m	Masse	[kg]
	Platzhalter Matizenelement	[-]
n	Lastvielfaches	[g]
p	Rollrate	[deg/s], [rad/s]
q	Nickrate	[deg/s], [rad/s]
	Geraden- bzw. Parabelkoeffizient	[-]
q_{dyn}	Staudruck	[Pa]
r	Gierrate	[deg/s], [rad/s]
	Referenzsignal, Eingang des geschl. Regelkreises	[deg], [rad]
t	Zeit	[s]
\hat{u}	Eingangsvektor	[-]
u	Geschwindigkeitskomponente x_f	[m/s]
v	Geschwindigkeitskomponente y_f	[m/s]
w	Geschwindigkeitskomponente z_f	[m/s]
x	x -Koordinate	[m]
y	y -Koordinate	[m]
z	z -Koordinate	[m]
\hat{x}	Zustandsvektor	[-]
\hat{y}	Ausgangsvektor	[-]

Griechische Buchstaben

Symbol	Bedeutung	Einheit
α	Anstellwinkel	[deg], [rad]
β	Schiebewinkel	[deg], [rad]
δ	Schubhebelstellung	[mm]

Symbol	Bedeutung	Einheit
Φ	Hängewinkel	[deg], [rad]
Θ	Längsneigung, Nickwinkel	[deg], [rad]
Ψ	Azimut	[deg], [rad]
γ	Gleitwinkel	[deg]
	Parameter Reglerentwurf	[-]
η	Höhenruderausschlag	[deg], [rad]
ζ	Seitenruderausschlag	[deg], [rad]
	Dämpfung	[-]
ξ	Hinterkantenklappenausschlag	[deg], [rad]
κ	Vorderkantenklappenausschlag	[deg], [rad]
σ	Triebwerkseinbauwinkel	[deg], [rad]
ω	Kreisfrequenz	[rad/s]
φ	Phase	[rad]
λ	Geradenparameter	[-]
ρ	Robustheitsmaß	[-]
	Spektralradius	[-]

Indices und Abkürzungen

Symbol	Bedeutung	Einheit
0	Ausgangszustand, Trimmzustand	[-]
<i>A</i>	Aerodynamik, aerodynamisch	[-]
<i>Ausl</i>	Auslauf (Triebwerk)	[-]
<i>Act</i>	Aktuator	[-]
ADM	AeroDataModul	[-]
<i>AS</i>	Anstellwinkelschwingung	[-]
<i>CAP</i>	Control Anticipation Parameter	[rad/s ² g]
<i>CG</i>	Schwerpunkt	[-]
<i>d</i>	Differentiel	[-]
<i>EAS</i>	Äquivalente Luftgeschwindigkeit	[-]
<i>Einl</i>	Einlauf (Triebwerk)	[-]
<i>f</i>	Flugzeugfest	[-]
<i>FZ</i>	Flugzustand	[-]
<i>g</i>	Geodätisch	[-]
<i>ges</i>	Gesamt	[-]
<i>i</i>	Zeilenindex	[-]
	Zählindex	[-]
<i>in</i>	Eingang, In	[-]
<i>j</i>	Spaltenindex	[-]
<i>lat</i>	Seitenbewegung, lateral	[-]

Symbol	Bedeutung	Einheit
<i>long</i>	Längsbewegung, longitudinal	[-]
<i>m, o</i>	Mit / ohne	[-]
<i>n, norm</i>	Normiert	[-]
<i>out</i>	Ausgang, Out	[-]
<i>O</i>	Beobachter (Observer)	[-]
<i>P</i>	Proportional	[-]
	Punkt <i>P</i>	
<i>Ph</i>	Phygoide	[-]
<i>Q</i>	Quotient	[-]
<i>rel</i>	relativ	[-]
<i>s</i>	Symmetrisch	[-]
<i>S</i>	Schub, schubbedingt	[-]
<i>Sens</i>	Sensor	[-]
<i>SPA</i>	Kurzzeitznäherung	[-]
<i>ss</i>	Stationärzustand, Steady State	[-]
<i>St</i>	Störklappe	[-]
<i>SV</i>	Steuervektor	[-]
<i>ÜF</i>	Übertragungsfunktion	[-]
<i>uw</i>	Überschwingweite	[%]
<i>ZV</i>	Zustandsvektor	[-]
<i>ZVA</i>	Zustandsvektorableitung	[-]

Kapitel 1

Einleitung

In modernen Flugzeugen erfüllen Flugregler die unterschiedlichsten Aufgaben. Diese reichen von der Erzeugung künstlicher Stabilität über die allgemeine Verbesserung der Flugeigenschaften des Flugzeugs wie beispielsweise die Entkopplung gekoppelter Bewegungen bis zum automatischen Verhindern des Verlassens des zulässigen Flugbereichs (Envelope Protection) und einiger weiterer Anwendungen und Einsatzgebiete.

Das Ziel der vorliegenden Arbeit ist es, für den Jettrainer Mako, ein generisches leichtes Kampfflugzeug, einen stabilitätserhöhenden Anstellwinkelfolgeregler zu entwerfen. Um eine hohe Agilität zu erreichen, ist das betreffende Flugzeug in bestimmten Flugzuständen natürlich instabil. Um es für den Piloten dennoch beherrschbar zu machen, wird ein Flugregler zur Erzeugung künstlicher Stabilität benötigt. Ein wesentliches Merkmal des hier entworfenen Reglers ist die Forderung einer möglichst hohen Robustheit gegenüber Modellunsicherheiten. Im Betrieb des Reglers muss trotz auftretenden Abweichungen zwischen dem Modell, für das und mit dem der Regler entworfen wurde, und dem realen Flugzeug stets ein stabiler Regelkreis gewährleistet sein. Neben einer möglichst hohen Robustheit werden an den Regler weitere Anforderungen in Form von Flugeigenschaftskriterien, auch Handling-Qualities genannt, gestellt.

1.1 Vorgehensweise

Aufgrund der Robusteitsanforderungen wird der Regler über ein robustes Reglerentwurfungsverfahren entworfen. Ein wesentlicher Vorteil dieser Entwurfsverfahren gegenüber klassischen Verfahren besteht in der Möglichkeit, zu erwartende Modellunsicherheiten bereits beim Entwurf zu berücksichtigen. Im klassischen Fall kann erst nach erfolgtem Entwurf geprüft werden, ob der geschlossene Regelkreis vorhandene Unsicherheiten toleriert.

Der Reglerentwurf erfolgt mittels des H_∞ -Open-Loop-Shaping Verfahrens, bei dem der offene Regelkreis G (Open-Loop), wie in Abb. 1.1 schematisch dargestellt, zunächst mittels der Wichtungsfunktionen W_1 und W_2 vor und nach der Regelstrecke gewichtet wird, um im Frequenzbereich einen Wunschverlauf zu erzielen (Shaping), denn bestimmte Anforderungen im Zeitbereich lassen sich in bestimmte Anforderungen im Frequenzbereich übersetzen. Das Entwurfsverfahren liefert dann mittels H_∞ -Optimierung einen Regler K , der den geschlossenen Regelkreis robust stabilisiert bezüglich Unsicherheiten in Form der Diagonal-

matrizen Δ_M und Δ_N in der coprime-faktorierten Strecke, bestehend aus den Faktoren M und N^{-1} , denn der Regler maximiert die zur Destabilisierung führenden Unsicherheiten und wird dadurch toleranter. Bei der Unsicherheit in der coprime Faktorisierung handelt es sich um eine Art allgemeine Unsicherheit, worin ein wesentlicher Vorteil des Verfahrens besteht: Ohne explizites Wissen über zu erwartende Unsicherheiten des Modells und deren Modellierung, kann bereits in einem frühen Entwurfsstadium diese allgemeine Unsicherheit berücksichtigt werden.

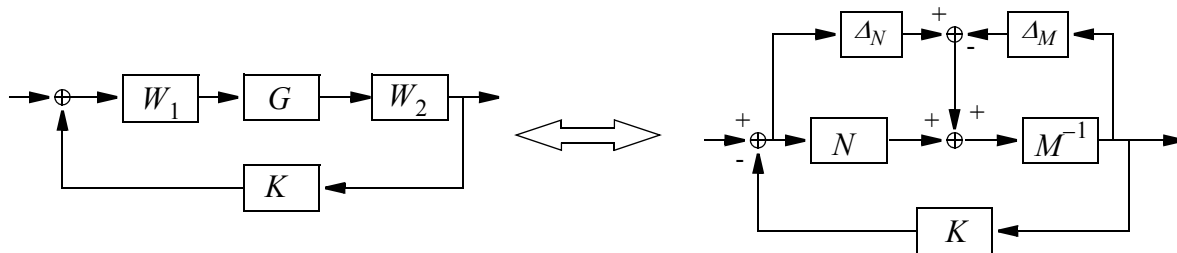


Abb. 1.1: gewichtete Strecke und coprime-faktorierte Form mit Unsicherheiten

Ein weiterer Vorteil des Verfahrens besteht in der relativen Einfachheit der Berechnung des optimalen Reglers. Dessen Parameter werden im Wesentlichen über die Lösung zweier algebraischer Riccati Gleichungen direkt berechnet. Anders als beim allgemeinen H_∞ -Entwurf sind also keine Iterationen zur Bestimmung der optimalen Robustheit notwendig.

Da es sich bei dem verwendeten Verfahren um ein lineares Entwurfsverfahren handelt, wird zum Entwurf eine linearisierte Form der Regelstrecke benötigt. Aufgrund von Nichtlinearitäten in der Regelstrecke ist diese Linearisierung naturgemäß nur in einer Umgebung ihres Auslegungspunktes gültig. Dementsprechend kann von einem Regler, der auf Basis dieser Linearisierung entworfen wurde, ebenfalls nur in der Umgebung des Auslegungspunktes eine hohe Regelqualität erwartet werden.

Um eine gleichbleibend gute Regelung in der gesamten Envelope des Flugzeugs zu erreichen, wird ein Gain-Scheduling-Verfahren angewendet, d.h. die Reglerparameter werden an den momentanen Flugzustand angepasst. Dazu wird die Envelope durch ein Raster von Entwurfspunkten diskretisiert, an denen die Regelstrecke getrimmt und linearisiert wird sowie einzelne, den Anforderungen entsprechende Regler entworfen werden. Diese Regler haben durch das Entwurfsverfahren bedingt alle die gleiche Struktur und unterscheiden sich lediglich durch die Werte einzelner Parameter. Im Betrieb des Reglers werden diese Parameter an Nicht-Entwurfspunkten mittels mehrdimensionaler Interpolation aus dem Raster der Entwurfspunkte berechnet.

Um den Entwurf der Einzelregler zu erleichtern und zu beschleunigen, kommt ein automatisiertes Reglerentwurfsverfahren zum Einsatz. Hierbei werden die zu bestimmenden Parameter der Wichtungsfunktionen mittels nichtlinearer, beschränkter Optimierung berechnet. Die zu optimierende Zielgröße ist die Robustheit des Reglers, die Beschränkungen ergeben sich aus den Anforderungen an die Flugeigenschaftskriterien.

Die Überprüfung des Entwurfs erfolgt sowohl durch lineare als auch nichtlineare Simulationen an zufällig ausgewählten Flugzuständen und deren Auswertung.

1.2 Ausgangssituation

Das zu regelnde Flugzeug liegt in Form von Trägheitsmomenten, Schwerpunkt- und Massendaten sowie einem Schub- und Aerodynamik-Modell vor. Hieraus wird zunächst ein nichtlineares Modell mit sechs Freiheitsgraden aufgebaut. Ausgehend von diesem Modell erfolgt an den Reglerentwurfspunkten die Trimmung und Linearisierung der Regelstrecke. An dieser Stelle sei auf die Besonderheit einer impliziten Aerodynamik hingewiesen. Die Ausgänge des Aerodynamikmodells hängen nicht nur von den Zuständen und Eingängen der Regelstrecke ab, sondern auch von der Zustandsableitung. Für die Zustandsableitungen ergeben sich daher implizite Differentialgleichungen, da die Ableitungen nicht nur auf der linken Seite, der Ausgangsseite der Gleichungen auftreten, sondern aufgrund der Aerodynamik auch auf der rechten Seite, der Eingangsseite. Diese Implizität ist sowohl bei der Trimmung als auch bei der Linearisierung der Gleichungen zu beachten.

1.3 Stand der Technik

Der Jettrainer Mako wurde bereits in früheren Arbeiten untersucht. So erfolgten beispielsweise durch Fischer in [1] eingehende Untersuchungen zu den dynamischen Eigenschaften des Flugzeugs, dessen Trimmung und Linearisierung sowie ein konventioneller Reglerentwurf für die Längsbewegung.

Möglichkeiten zur Implementierung von nichtlinearen Flugzeugmodellen und der Aufstellung der entsprechenden Differentialgleichungssysteme werden beispielsweise von Stevens und Lewis in [2] sowie von Brockhaus in [3] beschrieben. Ebenfalls in [2] finden sich Informationen zur Trimmung und zur numerischen Linearisierung von Regelstrecken. Allerdings beschränken sich die Angaben sowohl bezüglich der Trimmung als auch der Linearisierung auf explizite Differentialgleichungen. Zur Behandlung der im Rahmen dieser Arbeit auftretenden impliziten Differentialgleichungen ist eine Modifikation der Verfahren notwendig.

Das H_∞ -Open-Loop-Shaping-Verfahren (zum Teil auch nur H_∞ -Loop-Shaping oder Open-Loop-Shaping-Verfahren genannt) wird an einigen Stellen der einschlägigen Literatur beschrieben. Beispielsweise beschreiben Skogestad und Postlethwaite in [5] sowohl die Theorie des Verfahrens als auch verschiedene Implementierungsarten geschlossener Regelkreise mit Open-Loop-Shaping Reglern sowie konkrete, beispielhafte Anwendungen. Gleiches gilt für Papageorgiou in [4]. In beiden Fällen werden Regeln zur Bestimmung der freien Entwurfparameter benannt, es erfolgt jedoch kein automatisierter Reglerentwurf. Außerdem beschränken sich beide Autorengruppen auf lineare Regelkreise an einem einzelnen Entwurfspunkt. Ein Übergang zu einer nichtlinearen Betrachtung durch die Addition bzw. Subtraktion von Trimmwerten am Eingang und Ausgang des Reglers geschweige denn eine Anpassung der Reglerparameter an verschiedene Flugzustände erfolgt nicht.

Der Übergang vom linearen Entwurf zu einem nichtlinearen Regelkreis mit vom Flugzustand abhängigen Reglerparametern erfolgt z.B. in [6]. Hier werden des Weiteren verschiedene Strategien zum problemlosen, sprunghaften Umschalten des Reglers im Rahmen der Anpassung

sung an den Flugzustand diskutiert. Allerdings beschränkt sich die Abhängigkeit des Reglers auf einen einzelnen Parameter, im vorliegenden Fall die Fluggeschwindigkeit.

Der Fall einer Regleranpassung in Abhängigkeit von mehreren Parametern wird exemplarisch in [7] beschrieben, jedoch eignen sich die dort aufgeführten Strategien aufgrund ihrer hohen Komplexität nicht zur direkten Anwendung auf die vorliegende Arbeit.

Verfahren zur mehrdimensionalen linearen Interpolation werden in zahlreichen Veröffentlichungen beschrieben, beispielsweise in [8] und [9]. Allerdings beschränken sich diese Verfahren im Allgemeinen auf die Interpolation über einem voll besetzten, kartesischen Koordinatengitter. Je nach Dimension der Interpolation bedeutet dies, dass an einem Punkt nur dann interpoliert werden kann, wenn er im 1D-Fall von zwei Punkten auf einer Geraden umrahmt wird bzw. analog dazu im 2D-Fall von vier Punkten eines Rechtecks und im 3D-Fall von acht Punkten eines Quaders. Die teilweise gekrümmten Begrenzungslinien des dreidimensionalen Einsatzbereichs des Reglers im Falle dieser Arbeit führen in diesem Zusammenhang zu Problemen.

Im Bezug auf den Stand der Technik beinhaltet die vorliegende Arbeit daher folgende Neuerungen:

- Es wird eine Methode zur Trimmung und Linearisierung von impliziten Differentialgleichungen für verschiedene ausgezeichnete Flugzustände (stationärer Horizontalflug und quasi-stationärer Abfangbogen) vorgestellt, entsprechende Matlab-Routinen werden entwickelt (Kapitel 2).
- Das robuste H_∞ -Loop-Shaping Reglerentwurfsverfahren kommt zum Einsatz beim Entwurf eines stabilitätserhöhenden Anstellwinkel-Folgereglers. Dabei erfolgt eine Automatisierung des Entwurfs mittels beschränkter, nichtlinearer Optimierung mit einer automatischen Abarbeitung eines dreidimensionalen Rasters von Reglerentwurfspunkten (Kapitel 3 und Kapitel 4).
- Es erfolgt der Übergang vom linearen Entwurf zum nichtlinearen, angepassten Regler. Die Anpassung des Reglers erfolgt in Abhängigkeit von drei Parametern, eine Strategie zum problemlosen Anpassen des Reglers wird vorgestellt (Kapitel 5).
- Ein dreidimensionales lineares Interpolationsverfahren auf Basis von Prismen mit dreieckiger Grundfläche, dank derer sich der gekrümmte Einsatzbereich des Reglers besser abbilden lässt als mit den Quadern herkömmlicher Verfahren, wird entwickelt und vorgestellt (Kapitel 3).

Kapitel 2

Die Regelstrecke

Bei dem im Rahmen dieser Arbeit verwendeten Reglerentwurfsverfahren handelt es sich um ein lineares Verfahren. Dementsprechend wird eine lineare bzw. linearisierte Darstellung der Regelstrecke benötigt. Das folgende Kapitel beschäftigt sich zunächst mit der Herleitung der nichtlinearen Bewegungsgleichungen des Jettrainers, deren Trimmung und schließlich ihrer Linearisierung. Aus den linearisierten Bewegungsgleichungen wird das Modell der Längsbewegung abgeleitet und bezüglich seiner flugmechanischen Eigenschaften untersucht. Die Längsbewegung wird vereinfacht zur Kurzzeitznäherung, mit der schließlich der Reglerentwurf durchgeführt wird.

2.1 Das nichtlineare Modell und die Bewegungs-Differentialgleichungen

Beim nichtlinearen Modell handelt es sich um eine mathematische Beschreibung des dynamischen Verhaltens des betrachteten Flugzeuges. Hierzu wird der momentane Bewegungszustand des Flugzeuges durch den Zustandsvektor \hat{x} beschrieben. Die Eingänge des Modells werden im Steuervektor \hat{u} zusammengefasst, die Ausgänge des Modells bilden den Ausgangsvektor \hat{y} .

Hierbei beschreiben die nichtlinearen Bewegungs-Differentialgleichungen im Allgemeinen die momentane Ableitung des Zustandsvektors $\dot{\hat{x}}$ in Abhängigkeit des Steuervektors \hat{u} und des Zustandsvektors \hat{x} selbst. Der Ausgangsvektor \hat{y} ist im Allgemeinen ebenfalls ein System von Funktionen in Abhängigkeit von \hat{x} und \hat{u} . Es gilt also:

$$\begin{aligned} \dot{\hat{x}} &= \hat{f}(\hat{x}, \hat{u}) \\ \hat{y} &= \hat{g}(\hat{x}, \hat{u}) \end{aligned} \quad (2.1)$$

wobei es sich bei \hat{f} und \hat{g} um mehrstellige, nichtlineare Funktionen handelt.

Wie sich im Laufe dieses Kapitels zeigen wird, tritt im vorliegenden Fall die Besonderheit auf, dass die Zustandsableitung nicht nur vom Zustand und der Steuerung abhängt, sondern auch von der Zustandsableitung selbst. In diesem Fall entstehen daher implizite Differentialgleichungen der Form

$$\dot{\hat{x}} = \vec{f}(\hat{x}, \hat{u}, \dot{\hat{x}}). \quad (2.2)$$

Im Fall von expliziten Differentialgleichungen ist es durch die Beschreibung des Zustands zu einem bestimmten Zeitpunkt bei bekannten Steuerungen \hat{u} möglich, die Zustandsableitung am betreffenden Zeitpunkt zu berechnen und daraus mittels Integration der Bewegungs-Differentialgleichungen den Zustand bzw. den Zustandsverlauf für nachfolgende Zeitpunkte zu berechnen. Dabei erfolgt die Integration im Falle impliziter Gleichungen entweder unter Berücksichtigung von algebraischen Nebenbedingungen oder Näherungsweisen, indem bei der Berechnung der Zustandsableitung zum Zeitpunkt t für die Zustandsableitung auf der rechten Seite, d.h. der Eingangsseite der Differentialgleichungen, die Zustandsableitung eines vorangegangenen Zeitpunkts verwendet wird.

Die Herleitung der Bewegungs-Differenzialgleichungen erfolgt im Allgemeinen durch eine mehr oder weniger stark vereinfachende Modellierung der Realität.

Im vorliegenden Fall wird der Jettrainer als Starrkörper angenähert, das heißt die elastischen Freiheitsgrade des Flugzeugs werden vernachlässigt. Diese Näherung ist laut [2] für den Entwurf eines Anstellwinkel-Folgereglers zulässig, darüber hinaus handelt es sich aufgrund der relativ kleinen Abmessungen des Trainers um ein verhältnismäßig steifes Flugzeug.

Der Starrkörper wird als Körper mit einer konstanten Masse sowie einer konstanten Massenverteilung angenommen, die Schwerpunktlage ist daher ebenfalls konstant. Somit besitzt der Starrkörper sechs Freiheitsgrade, drei translatorische, die seine Position im Raum bzw. deren Änderung beschreiben, und drei rotatorische, die seine räumliche Orientierung bzw. deren Änderung beschreiben. Er bewege sich über einer als ruhend und flach angenommenen Erde, wobei es sich um Näherungen handelt, die im Bereich der Modellierung und Regelung von Flugzeugen üblich und auch im vorliegenden Fall angesichts der auftretenden Geschwindigkeiten und Flughöhen anwendbar sind.

Die Bewegungsgleichungen des Sechs-Freiheitsgrad-Modells (6DOF-Modell) werden im folgenden Abschnitt beschrieben. Auf eine Herleitung der einzelnen Gleichungen wird verzichtet, hierzu sei auf die einschlägige Literatur wie beispielsweise [2] oder [3] verwiesen.

2.1.1 Das 6-DOF-Dynamikmodell

Die Beschreibung des momentanen Zustands eines Flugzeugs im dreidimensionalen Raum erfolgt durch den Zustandsvektor \hat{x} des Fluggeräts.

Im Falle des 6DOF-Modells setzt sich dieser Vektor zusammen aus Angaben zur momentanen translatorischen Geschwindigkeit des Flugkörpers, seiner momentanen Drehgeschwindigkeit, seiner momentanen Position im Raum sowie seiner momentanen Orientierung. Im vorliegenden Fall setzt sich der Zustandsvektor dementsprechend aus den folgenden Komponenten zusammen:

$$\hat{x} = \left[V_{free} \ \alpha \ \beta \ p \ q \ r \ \Phi \ \Theta \ \Psi \ x_g \ y_g \ z_g \right]^T \quad (2.3)$$

Die einzelnen Komponenten und ihre Bedeutung werden in der folgenden Tabelle beschrieben:

Pos.	Symbol	Beschreibung	Bedeutung
1	V_{free}	Anströmgeschwindigkeit	Definition der translatorischen Geschwindigkeit \vec{v}
2	α	Anstellwinkel	
3	β	Schiebewinkel	
4	p	Rollgeschwindigkeit	Definition der Drehgeschwindigkeit $\vec{\Omega}$
5	q	Nickgeschwindigkeit	
6	r	Giergeschwindigkeit	
7	Φ	Azimut	Eulerwinkel, Definition der Orientierung im Raum
8	Θ	Längsneigung	
9	Ψ	Hängewinkel	
10	x_g	x-Position	Definition der Position im Raum \vec{X}
11	y_g	y-Position	
12	z_g	z-Position	

Tabelle 2.1: Komponenten des Zustandsvektors

Hierbei beziehen sich die einzelnen Komponenten des Zustandsvektors entsprechend [10] auf verschiedene Koordinatensysteme.

Translatorische Geschwindigkeit. Bei der translatorischen Geschwindigkeit \vec{v} handelt es sich um die Geschwindigkeit des Ursprungs des körperfesten Koordinatensystems bezüglich der Luft, die das Flugzeug umgibt. Der Ursprung dieses Koordinatensystems liegt üblicherweise im Schwerpunkt des Flugzeugs, die Längsachse x_f liegt in der Symmetrieebene des Flugzeugs, mit positiver Orientierung zum Bug, die Querachse y_f steht senkrecht zur Symmetrieebene mit positiver Orientierung nach Steuerbord. Die Hochachse z_f vervollständigt das System, sie liegt ebenfalls in der Symmetrieebene und ist positiv zur Unterseite des Flugzeugs orientiert.

Da die Luft im Rahmen dieser Arbeit als ruhend angenommen wird, resultiert die Anströmung des Flugzeugs allein aus der Bewegung des körperfesten Systems gegenüber der umgebenden Luft und der Geschwindigkeitsvektor und der Vektor der Anströmung sind somit genau entgegengesetzt und betragsmäßig gleich.

Daher kann die translatorische Geschwindigkeit im vorliegenden Fall durch die Anströmgeschwindigkeit in Koordinaten des körperfesten Systems dargestellt werden, wobei eine Repräsentierung in Polarkoordinaten verwendet wird. Der Vektor wird also beschrieben durch seinen Betrag, d.h. die Anströmgeschwindigkeit V_{free} und durch die Winkel α und β . Beim Anstellwinkel α handelt es sich um den Winkel zwischen der Längsachse des Flugzeugs, also x_f und der Projektion des Geschwindigkeitsvektors auf die Symmetrieebene des Flugzeugs, also der $x_f z_f$ -Ebene des körperfesten Koordinatensystems.

Der Schiebewinkel β ist der Winkel zwischen dem Geschwindigkeitsvektor \vec{v} und der Symmetrieebene des Luftfahrzeugs.

Ebenso ist eine Repräsentierung in kartesischen Koordinaten (u , v und w) denkbar. Ziel der Arbeit ist jedoch der Entwurf eines Anstellwinkel-Folgereglers, hierbei bietet die Polarkoordinatendarstellung den Vorteil, dass der Anstellwinkel α direkt als Komponente des Zustandsvektors auftritt. Außerdem bietet sie den Vorteil einer vereinfachten Berechnung der aerodynamischen Kräfte und Momente, da für diese sowohl der Anstellwinkel als auch der Schiebewinkel und der Geschwindigkeitsbetrag benötigt werden (siehe auch 2.1.2).

Drehgeschwindigkeit. Die Komponenten des Drehgeschwindigkeitsvektors $\vec{\Omega}$, p , q und r , beziehen sich auf die Rotation des Flugzeugs um die Achsen des körperfesten Koordinatensystems.

Orientierung im Raum. Die Beschreibung der Orientierung des Flugzeugs im Raum erfolgt durch die Beschreibung der Lage des körperfesten Koordinatensystems bezüglich des geodätischen Systems mit körperfestem Ursprung. Die x - und y -Achse des geodätischen Systems, x_g und y_g , liegen dabei in der Horizontalebene, die z -Achse z_g steht senkrecht dazu mit positiver Ausrichtung nach unten bzw. zum Erdmittelpunkt, der Ursprung des Systems fällt mit dem Ursprung des körperfesten Systems, also dem Schwerpunkt des Flugzeugs zusammen. Die Lage- oder Eulerwinkel Φ , Θ und Ψ beschreiben nun die Drehungen, die notwendig sind, um das geodätische Koordinatensystem ins körperfeste Koordinatensystem zu überführen. Eine Alternative hierzu ist die Darstellung durch Quaternionen. Diese bietet den Vorteil, dass, anders als bei den Eulerwinkeln, in den Bewegungsgleichungen keine Singularität bei einer Längsneigung von $\Theta = 90^\circ$ auftritt. Die Entscheidung zu Gunsten der Eulerwinkel erfolgt aufgrund einer besseren Anschaulichkeit sowie einer erheblich geringeren Komplexität. Darüber hinaus zeigt sich im Laufe der Arbeit, dass im gegebenen Einsatzbereich des Jettrainers eine Längsneigung von 90° nicht erfliegbar ist. Die Darstellung über die Eulerwinkel ist also ausreichend.

Position im Raum. Die Komponenten x_g , y_g und z_g beschreiben schließlich die Position des Flugzeugs im Raum in Form einer Beschreibung der Lage des Ursprungs des körperfesten Koordinatensystems (bzw. des geodätischen Systems mit körperfestem Ursprung), also des Schwerpunkts des Flugzeugs bezüglich des erdfesten Koordinatensystems. Dessen Achsen und die Achsen des geodätischen Systems sind parallel, allerdings hat das erdfeste Koordinatensystem einen festen Ursprung. Dieser Ursprung ist laut [10] frei wählbar und wird im vorliegenden Fall auf die Projektion der Startposition zu Beginn einer Betrachtung des Flugzeugs auf die Erdoberfläche gesetzt. Aufgrund der Orientierung des Systems, d.h. der positiven Richtung der z_g -Achse zum Erdmittelpunkt folgt hieraus, dass die Flughöhe h der negative Wert der z_g -Position ist. Es gilt also

$$h = -z_g. \quad (2.4)$$

Zwölf nichtlineare Gleichungen beschreiben nun den Zusammenhang zwischen der Ableitung $\dot{\vec{x}}$ des Zustandsvektors und dem Zustand selbst sowie den am Flugzeug angreifenden äußeren Kräften und Momenten. Hierbei handelt es sich um aerodynamische Kräfte und Momente sowie um Schubkräfte und Momente, die vom Triebwerk des Flugzeugs verursacht werden.

Die aerodynamischen Kräfte und Momente sowie die Schubkräfte und Momente, deren Berechnung in den Abschnitten 2.1.2 und 2.1.3 beschrieben wird, hängen dabei ihrerseits vom Zustandsvektor sowie vom Steuervektor \vec{u} ab. Dieser beinhaltet die Stellungen der diversen Steuerflächen des Flugzeugs sowie die Schubhebelstellung und damit das Leistungsniveau des Triebwerks. Eine Übersicht über die diversen Steuerflächen bietet Abbildung 2.1.

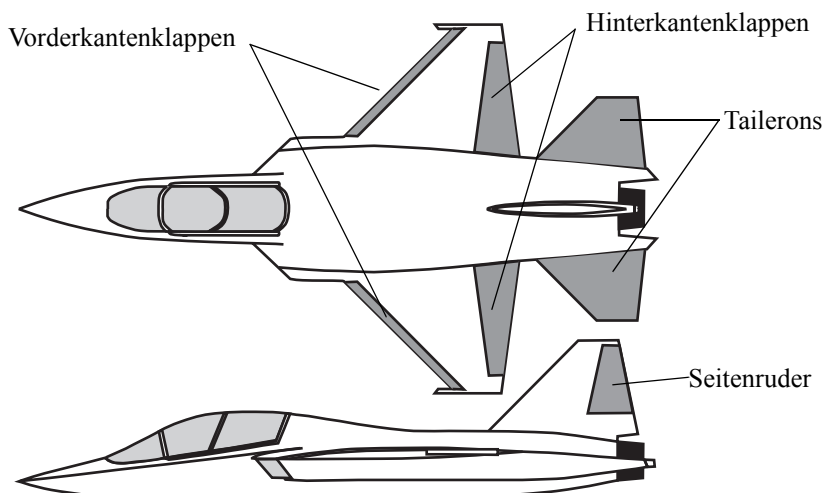


Abb. 2.1: Steuerflächen des Jettrainers Mako

Des Weiteren verfügt der Jettrainer über eine Brems- bzw. Störklappe. Diese steht jedoch bei der im Rahmen dieser Arbeit verwendeten Aerodynamikmodellierung nicht zur Verfügung. Sie wurde deshalb in Abb. 2.1 nicht eingezeichnet. Sie tritt im Steuervektor auf, da die Funktion zur Berechnung der aerodynamischen Kräfte und Momente einen Wert für die Störklappenstellung benötigt, obwohl dieser dann in der vorliegenden Version nicht berücksichtigt wird (siehe auch Kapitel 2.1.2).

Die Zusammensetzung des Steuervektors lautet:

$$\vec{u} = \left[\eta_s \quad \eta_d \quad \zeta \quad \kappa \quad \xi_s \quad \xi_d \quad \eta_S \quad \delta_n \right]^T. \tag{2.5}$$

Die Bedeutung der einzelnen Komponenten des Vektors ist in der folgenden Tabelle zusammengefasst:

Pos.	Symbol	Beschreibung
1	η_s	Taileronausschlag, symmetrisch
2	η_d	Taileronausschlag, differentiell
3	ζ	Seitenruderausschlag
4	κ	Vorderkantenklappenausschlag
5	ξ_s	Hinterkantenklappenausschlag, symmetrisch
6	ξ_d	Hinterkantenklappenausschlag, differentiell

Pos.	Symbol	Beschreibung
7	η_{St}	Störklappenauerschlag
8	δ_n	Schubhebelstellung, normiert auf 100,5 mm

Tabelle 2.2: Komponenten des Steuervektors

Es ist zu beachten, dass die Aerodynamik im vorliegenden Fall außerdem von der Ableitung des Anstellwinkels, also der zweiten Komponente der Ableitung $\dot{\hat{x}}$ des Zustands abhängt. Es ergeben sich daher zwölf zum Teil implizite Bewegungs-Differentialgleichungen der Form

$$\dot{\hat{x}} = \dot{f}(\hat{x}, \hat{u}, \dot{\hat{x}}). \quad (2.6)$$

Die Herleitung der Gleichungen des Systems (2.6) erfolgt entsprechend [3] durch die Anwendung des Drallsatzes und des Impulssatzes auf den Starrkörper. Die Gleichungen lauten im Einzelnen:

Translatorische Geschwindigkeit

$$\begin{aligned} \dot{V}_A &= \frac{u \cdot \dot{u} + v \cdot \dot{v} + w \cdot \dot{w}}{V_{free}} \\ \dot{\alpha} &= \frac{u \cdot \dot{w} - \dot{u} \cdot w}{u^2 + w^2} \\ \dot{\beta} &= \frac{\dot{v} \cdot V_{free} - v \cdot \dot{V}_{free}}{V_{free}^2 \cdot \cos \beta} \end{aligned} \quad (2.7)$$

mit den kartesischen Komponenten der Geschwindigkeit

$$\begin{aligned} u &= V_{free} \cdot \cos \alpha \cos \beta \\ v &= V_{free} \cdot \sin \beta \\ w &= V_{free} \cdot \sin \alpha \cos \beta \end{aligned} \quad (2.8)$$

und deren Ableitungen

$$\begin{aligned} \dot{u} &= X_{ges}/m - g \cdot \sin \Theta + r \cdot v - q \cdot w \\ \dot{v} &= Y_{ges}/m + g \cdot \sin \Phi \cos \Theta + p \cdot w - r \cdot u \\ \dot{w} &= Z_{ges}/m + g \cdot \cos \Phi \cos \Theta + q \cdot u - p \cdot v \end{aligned} \quad (2.9)$$

mit der Flugzeugmasse m und der Erdbeschleunigung g . Letztere hängt vom Abstand des Flugzeugs zum Erdmittelpunkt, also seiner Flughöhe bzw. Position und damit vom Zustandsvektor ab. Ihre Berechnung erfolgt gemeinsam mit weiteren höhenabhängigen Daten im Atmosphärenmodell, welches in Abschnitt 2.1.4 beschrieben wird.

Bei X_{ges} handelt es sich um die Summe aller aerodynamischen (Index A) und schubbedingten Kräfte (Index S) in Richtung der körperfesten x_f -Achse. Analog dazu handelt es sich bei Y_{ges} und Z_{ges} um die Summen der Kräfte in y_f und z_f -Richtung. Wie

bereits erwähnt handelt es sich dabei um Funktionen, die ihrerseits von \dot{x} , \dot{u} und zum Teil von \ddot{x} abhängen. Es gilt daher

$$\begin{aligned} X_{ges} &= X_S(\dot{x}, \dot{u}) + X_A(\dot{x}, \dot{u}, \ddot{x}) \\ Y_{ges} &= Y_S(\dot{x}, \dot{u}) + Y_A(\dot{x}, \dot{u}, \ddot{x}) \\ Z_{ges} &= Z_S(\dot{x}, \dot{u}) + Z_A(\dot{x}, \dot{u}, \ddot{x}) \end{aligned} \quad (2.10)$$

Setzt man die Gleichungen aus (2.10), (2.9) und (2.8) in die Gleichungen (2.7) ein, erhält man schließlich die ersten drei nichtlinearen impliziten Differentialgleichungen für die ersten drei Komponenten der Zustandsableitung in Abhängigkeit des Zustandsvektors, des Steuervektors und der Zustandsableitung.

Für die drei Ableitungen der Drehgeschwindigkeiten gelten die folgenden Beziehungen:

$$\begin{aligned} \dot{p} &= \frac{I_y I_z}{\text{Det}(I)} \cdot L_{ges} + \frac{I_y I_{xz}}{\text{Det}(I)} \cdot N_{ges} \\ \dot{q} &= \frac{I_x I_z - I_{xz}^2}{\text{Det}(I)} \cdot M_{ges} \\ \dot{r} &= \frac{I_y I_{xz}}{\text{Det}(I)} \cdot L_{ges} + \frac{I_x I_y}{\text{Det}(I)} \cdot N_{ges} \end{aligned} \quad (2.11)$$

mit dem Trägheitstensor I des Flugzeugs bezüglich des körperfesten Koordinatensystems

$$I = \begin{bmatrix} I_x & 0 & I_{xz} \\ 0 & I_y & 0 \\ I_{xz} & 0 & I_z \end{bmatrix}. \quad (2.12)$$

Bei L_{ges} , M_{ges} und N_{ges} handelt es sich um die Gesamtmomente um die körperfeste x_f , y_f und z_f -Achse. Sie setzen sich zusammen aus der Summe der Intertialmomente aufgrund der Rotation des trägheitsbehafteten Starrkörpers und den äußeren Momenten aufgrund der Aerodynamik und des Triebwerks. Es gilt:

$$\begin{aligned} L_{ges} &= L_A(\dot{x}, \dot{u}, \ddot{x}) + L_S(\dot{x}, \dot{u}) + q \cdot r \cdot (I_y - I_z) + p \cdot q \cdot I_{xz} \\ M_{ges} &= M_A(\dot{x}, \dot{u}, \ddot{x}) + M_S(\dot{x}, \dot{u}) + I_{xz} \cdot (r^2 - p^2) + p \cdot r \cdot (I_z - I_x) \\ N_{ges} &= N_A(\dot{x}, \dot{u}, \ddot{x}) + N_S(\dot{x}, \dot{u}) + p \cdot q \cdot (I_x - I_y) - r \cdot q \cdot I_{xz} \end{aligned} \quad (2.13)$$

Die Ableitungen der drei Lagewinkel werden direkt aus den Drehgeschwindigkeiten und den Lagewinkeln selbst berechnet. Es gelten folgende Zusammenhänge:

$$\begin{aligned}
\dot{\Phi} &= p + (q \sin \Phi + r \cos \Phi) \cdot \tan \Theta \\
\dot{\Theta} &= q \cos \Phi - r \sin \Phi \\
\dot{\Psi} &= (q \sin \Phi + r \cos \Phi) / \cos \Theta
\end{aligned} \tag{2.14}$$

Bei der Ableitung der Positionskomponenten handelt es sich um die Geschwindigkeit des Luftfahrzeuges über Grund. Diese ist wie oben beschrieben entgegengesetzt gleich der Anströmgeschwindigkeit, die durch die ersten drei Komponenten des Zustandsvektors in körperfesten Koordinaten beschrieben wird. Deshalb erhält man die Komponenten der Positionsableitung durch eine Koordinatentransformation der Anströmgeschwindigkeit vom körperfesten System ins geodätische System. Diese Transformation wird definiert durch die Eulerwinkel, die die Lage des körperfesten Systems bezüglich des geodätischen Systems beschreiben.

Es folgt daher für die Ableitung der Position:

$$\begin{bmatrix} \dot{x}_g \\ \dot{y}_g \\ \dot{z}_g \end{bmatrix} = M_{gf} \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \tag{2.15}$$

mit der Transformationsmatrix M_{gf} :

$$M_{gf} = \begin{bmatrix} \cos \Psi \cos \Theta & \cos \Psi \sin \Theta \sin \Phi - \sin \Psi \cos \Phi & \cos \Psi \sin \Theta \cos \Phi + \sin \Psi \sin \Phi \\ \sin \Psi \cos \Theta & \sin \Psi \sin \Theta \sin \Phi + \cos \Psi \cos \Phi & \sin \Psi \sin \Theta \sin \Phi - \cos \Psi \sin \Phi \\ -\sin \Theta & \cos \Theta \sin \Phi & \cos \Theta \cos \Phi \end{bmatrix}. \tag{2.16}$$

Damit liegt für jede der zwölf Komponenten des Zustandsvektors eine Differentialgleichung vor, die die Ableitung der betreffenden Komponente beschreibt, die zwölf Gleichungen liefern also die zwölf Zustandsableitungen.

Der Ausgangsvektor \dot{y} des nichtlinearen Modells besteht aus dem Zustandsvektor sowie den Lastvielfachen n_x , n_y und n_z . Er lautet ausgeschrieben:

$$\dot{y} = \left[V_{free} \quad \alpha \quad \beta \quad p \quad q \quad r \quad \Phi \quad \Theta \quad \Psi \quad x_g \quad y_g \quad z_g \quad n_x \quad n_y \quad n_z \right]^T. \tag{2.17}$$

Aufgrund der Lastvielfachen, die je nach Berechnung (siehe Abschnitt 2.1.6) entweder die aerodynamischen Kräfte oder die Zustandsableitungen enthalten, handelt es sich wie im Falle der Zustandsableitung auch hier um ein System von Gleichungen, die sowohl von \dot{x} und \dot{u} als auch von \dot{x} abhängen.

Zur Berechnung der Zustandsableitungen sowie des Ausgangs des Modells werden wie oben beschrieben die aerodynamischen Kräfte X_A , Y_A und Z_A , die aerodynamischen Momente L_A , M_A und N_A , die Schubkräfte X_S , Y_S und Z_S sowie die Schubmomente L_S , M_S und N_S benötigt. Ihre Berechnung in Abhängigkeit des Zustands, der Steuerung und ggf. der Ableitung wird in den folgenden Abschnitten beschrieben.

2.1.2 Das Aerodynamikmodell

Der folgende Abschnitt beschäftigt sich mit der Modellierung der Aerodynamik des Jettrainers und der Bestimmung der aerodynamischen Kräfte und Momente. Diese werden, wie im vorangegangenen Abschnitt beschrieben wurde, auf der Eingangsseite der nichtlinearen Bewegungs-Differentialgleichungen benötigt.

Das Aerodynamikmodell des Jettrainers Mako wird für die vorliegende Arbeit in Form des AeroData-Moduls ADM4.0 von EADS zur Verfügung gestellt. Es handelt sich hierbei um einen Matlab-Simulink Block, der auf der s-function `sfun_adm_v4_0.dll` basiert, durch die mittels mehrdimensionaler, linearer Interpolation aus Windkanalmessdaten die aerodynamischen Kräfte und Momente in Abhängigkeit der entsprechenden Komponenten des Zustands, des Steuervektors sowie der Zustandsableitung berechnet werden.

Die Eingänge des ADM4.0 sind mit ihrem jeweiligen Wertebereich in der folgenden Tabelle zusammengefasst:

Pos.	Symbol	Einh.	Min	Max	Beschreibung	Hinweis
1	α	[deg]	-10	+45	Anstellwinkel	2. Komponente ZV ^{a)}
2	β	[deg]	-15	+15	Schiebewinkel	3. Komponente ZV
3	Ma	[-]	0,1	0,5	Machzahl	aus V_{free} u. Schallgeschw. $a(h)$
4	$\dot{\alpha}$	[deg/s]	-360	+360	Abl. Anstellw.	2. Komponente ZVA ^{b)}
5	p	[deg/s]	-360	+360	Rollrate	4. Komponente ZV
6	q	[deg/s]	-360	+360	Nickrate	5. Komponente ZV
7	r	[deg/s]	-360	+360	Gierrate	6. Komponente ZV
8	V_{free}	[m/s]	34	170	Anströmgeschw.	1. Komponente ZV
9	q_{dyn}	[Pa]	500	18000	Staudruck	aus V_{free} u. Luftdichte $\rho(h)$
10	η_s	[deg]	-30	10	Taileron, symmetrisch	1. Komponente SV ^{c)}
11	η_d	[deg]	-20	+20	Taileron, differentiell	2. Komponente SV
12	ζ	[deg]	-30	+30	Seitenruderausschlag	3. Komponente SV
13	κ	[deg]	-45	0	Vorderkantenklappe	4. Komponente SV
14	ξ_s	[deg]	-20	+20	Hinterkantenkl., symm.	5. Komponente SV
15	ξ_d	[deg]	-20	+20	Hinterkantenkl., diff.	6. Komponente SV
16	η_{St}	[deg]	nicht. verw.		Störklappe	7. Komponente SV, nicht verw.
17	x_{CG}	[m]	+9	+10	Schwerp. x -Koord.	benötigt zur Berechnung v. L_A
18	y_{CG}	[m]	-1	+1	Schwerp. y -Koord.	benötigt zur Berechnung v. M_A
19	z_{CG}	[m]	-1	+1	Schwerp. z -Koord.	benötigt zur Berechnung v. N_A
20	$gear$	[-]	nicht. verw.		Fahrwerk	
21	$store$	[-]	nicht. verw.		Beladungszustand	

a) ZV - Zustandsvektor

b) ZVA - Ableitung Zustandsvektor

c) SV - Steuervektor

Tabelle 2.3: Eingänge des AeroData-Moduls ADM4.0

Es ist zu beachten, dass das ADM die Größen Machzahl Ma , Anströmgeschwindigkeit V_{free} und Staudruck q_{dyn} als scheinbar unabhängige Eingänge betrachtet. In der Realität sind diese Eingänge jedoch über das Atmosphärenmodell miteinander gekoppelt, sodass nur zwei von ihnen unabhängig gewählt werden können und sich der dritte aus den beiden anderen ergibt.

Die im Rahmen dieser Arbeit verwendeten Schwerpunktkoordinaten lauten in Koordinaten des flugzeugfesten Systems und der Einheit [m]

$$\dot{\mathbf{x}}_{CG} = [9,444 \ 0 \ -0,074]^T \quad (2.18)$$

Die Ausgabe der Kräfte und Momente erfolgt in einem Koordinatensystem, dessen Achsen parallel zu denen des körperfesten Koordinatensystems sind. Der Ursprung des Systems liegt wie der des körperfesten Systems ebenfalls im Schwerpunkt. Lediglich die Orientierung der x - und z -Achse des Systems ist entgegengesetzt der des körperfesten Systems. Um die Ausgänge der Aerodynamik in die Bewegungs-Differentialgleichungen einsetzen zu können, müssen daher X_A und Z_A mit -1 multipliziert werden. Die Einheit der Kräfte am Ausgang des AeroData-Moduls beträgt [N], Momente werden in [Nm] ausgegeben.

2.1.3 Das Schubmodell

Die Schubkräfte X_S , Y_S und Z_S sowie die Momente L_S , M_S und N_S , die auf das Flugzeug wirken, berechnen sich aus den Kräften F_{Einl} am Einlauf und F_{Ausl} am Auslauf des Triebwerks, den zugehörigen Hebelarmen und dem momentanen Anstellwinkel sowie Schiebewinkel. Die Hebelarme ergeben sich aus den Abständen zwischen den gedachten Angriffspunkten der Kräfte am Triebwerkseinlauf sowie -auslauf und dem Schwerpunkt des Flugzeugs. Für den Angriffspunkt der Kraft am Einlauf gilt

$$\dot{\mathbf{x}}_{Einl} = [x_{Einl} \ y_{Einl} \ z_{Einl}]^T, \quad (2.19)$$

bzw. in körperfesten Koordinaten und der Einheit [m]

$$\dot{\mathbf{x}}_{Einl} = [6,848 \ 0 \ -0,28]^T. \quad (2.20)$$

Die Koordinaten des Angriffspunkts der Kraft am Auslauf lauten

$$\dot{\mathbf{x}}_{Ausl} = [x_{Ausl} \ y_{Ausl} \ z_{Ausl}]^T, \quad (2.21)$$

bzw. in körperfesten Koordinaten und der Einheit [m]

$$\dot{\mathbf{x}}_{Ausl} = [14,933 \ 0 \ -0,115]^T. \quad (2.22)$$

Die Kräfte F_{Einl} und F_{Ausl} charakterisieren das Triebwerk und sind Funktionen der Schubhebelstellung δ , der Flughöhe h und der Machzahl Ma . Sie werden durch das Schubmodul über mehrdimensionale Interpolation aus Tabellendaten berechnet. Die

entsprechenden Tabellen stehen in Form eines Matlab-Simulink Blockes zur Verfügung, der wiederum auf der von EADS gelieferten s-function `sfun_mako_engine` basiert. Die Eingänge mit ihrem Wertebereich sind in der folgenden Tabelle zusammengefasst:

Pos.	Symbol	Einheit	Min	Max	Beschreibung	Hinweis
1	δ	[mm]	15	138,5	Schubhebelstellung	aus 8. Komponente SV ^{a)}
2	h	[m]			Flughöhe	aus 12. Komponente ZV ^{b)}
3	Ma	[-]	0,1	0,5	Machzahl	aus V_{free} und Schallgeschw. $a(h)$

a) SV - Steuervektor

b) ZV - Zustandsvektor

Tabelle 2.4: Eingänge des Schubmoduls

Es ist zu beachten, dass es sich bei der 8. Komponente des Steuervektors um die auf 100,5 mm normierte Schubhebelstellung handelt. Somit entspricht ein Wert von $\delta_n = 1$ gerade einer Schubhebelstellung von $\delta = 100,5$ mm, was wiederum dem Leistungsniveau "maximaler Trockenschub" entspricht. Der Reglerentwurf wird für diese Leistungseinstellung durchgeführt. Ebenso handelt es sich bei der 12. Komponente des Zustandsvektors nicht unmittelbar um die Flughöhe, sondern um die z_g -Position des Flugzeugs, also genau den negativen Wert der Flughöhe (siehe Abschnitt 2.1.1).

Die Berechnung der resultierenden Schubkräfte und -momente erfolgt entsprechend [3]. Für die Kräfte am Einlauf gilt

$$\begin{bmatrix} X_{Einl} \\ Y_{Einl} \\ Z_{Einl} \end{bmatrix} = F_{Einl} \cdot \begin{bmatrix} -\cos \alpha \cos \beta \\ -\sin \beta \\ -\sin \alpha \cos \beta \end{bmatrix}, \quad (2.23)$$

am Auslauf gilt die Beziehung

$$\begin{bmatrix} X_{Ausl} \\ Y_{Ausl} \\ Z_{Ausl} \end{bmatrix} = F_{Einl} \cdot \begin{bmatrix} \cos \sigma \\ 0 \\ -\sin \sigma \end{bmatrix} \quad (2.24)$$

mit dem Triebwerkseinbauwinkel σ , für den im vorliegenden Fall jedoch gilt $\sigma = 0$. Die resultierenden Kräfte lauten:

$$\begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} = \begin{bmatrix} X_{Ausl} \\ Y_{Ausl} \\ Z_{Ausl} \end{bmatrix} + \begin{bmatrix} X_{Einl} \\ Y_{Einl} \\ Z_{Einl} \end{bmatrix}. \quad (2.25)$$

Für die Schubmomente gilt:

$$\begin{bmatrix} L_S \\ M_S \\ N_S \end{bmatrix} = \begin{bmatrix} -Y_{Einl} \cdot \Delta z_{Einl} - Y_{Ausl} \cdot \Delta z_{Ausl} \\ X_{Einl} \cdot \Delta z_{Einl} - Z_{Einl} \cdot \Delta x_{Einl} + X_{Ausl} \cdot \Delta z_{Ausl} + Z_{Ausl} \cdot \Delta x_{Ausl} \\ Y_{Einl} \cdot \Delta x_{Einl} \end{bmatrix} \quad (2.26)$$

mit den Hebelarmen

$$\begin{aligned} \Delta x_{Einl} &= x_{CG} - x_{Einl} \\ \Delta x_{Ausl} &= x_{Ausl} - x_{CG} \\ \Delta z_{Einl} &= z_{CG} - z_{Einl} \\ \Delta z_{Ausl} &= z_{CG} - z_{Ausl} \end{aligned} \quad (2.27)$$

2.1.4 Das Atmosphärenmodell

Die höhenabhängigen Größen Luftdichte ρ , Schallgeschwindigkeit a und Erdbeschleunigung g werden gemäß der Standardatmosphäre DIN ISO 2533 [11] berechnet. Die Berechnung in Matlab-Simulink erfolgt über einen s-function Block, dem die c-mex s-function `atmos04` zu Grunde liegt. Der Bereich zulässiger Eingangswerte erstreckt sich über $-2000 \text{ m} \leq h \leq 32000 \text{ m}$. Dieser Wertebereich ist kleiner als das in [11] angegebene Intervall. Der eingeschränkte Höhenbereich ist jedoch ausreichend, da im Rahmen dieser Arbeit gemäß dem Einsatzbereich des Jettrainers eine maximale Flughöhe von $h_{max} = 55000 \text{ ft}$, also rund 16764 m , angenommen wird (siehe auch Abschnitt 2.1.5).

Der Staudruck q_{dyn} und die Machzahl Ma werden über folgende Formeln ebenfalls in der s-function berechnet:

$$q_{dyn} = \frac{1}{2} \rho(h) \cdot V_{free}^2 \quad (2.28)$$

$$Ma = V_{free} / a(h) \quad (2.29)$$

2.1.5 Einsatzbereich des Jettrainers

Der betrachtete Einsatzbereich des Jettrainers ergibt sich in erster Linie durch den Wertebereich der Eingänge des Aerodynamikmodells. Weitere Beschränkungen ergeben sich durch eine maximal zulässige Flughöhe h_{max} , maximale Lastvielfache $n_{x,max}$, $n_{y,max}$ und $n_{z,max}$ sowie eine maximale äquivalente Luftgeschwindigkeit V_{EAS} . Die Begrenzungen sind mit Einheiten und ihren numerischen Werten in folgender Tabelle zusammengefasst:

Symbol	Beschreibung	Einheit	Min	Max
Ma	Machzahl	[-]	0,1	0,5
q_{dyn}	Staudruck	[Pa]	500	18000
V_{free}	Anströmgeschwindigkeit	[m/s]	34	170

Symbol	Beschreibung	Einheit	Min	Max
h	Flughöhe	[ft]	0	55000
n_x	Lastvielfaches in x	[-]		
n_y	Lastvielfaches in y	[-]		
n_z	Lastvielfaches in z	[-]		
V_{EAS}	äquivalente Luftgeschw.	[kn]		700
α	Anstellwinkel	[deg]	-10	45

Tabelle 2.5: Einsatzbereich des Jettrainers

Hieraus ergibt sich in der $Ma-h$ -Ebene die in Abb. 2.2 dargestellte Envelope. Zu Gunsten einer besseren Übersichtlichkeit wurde an dieser Stelle auf den Anstellwinkel als dritte Achse verzichtet.

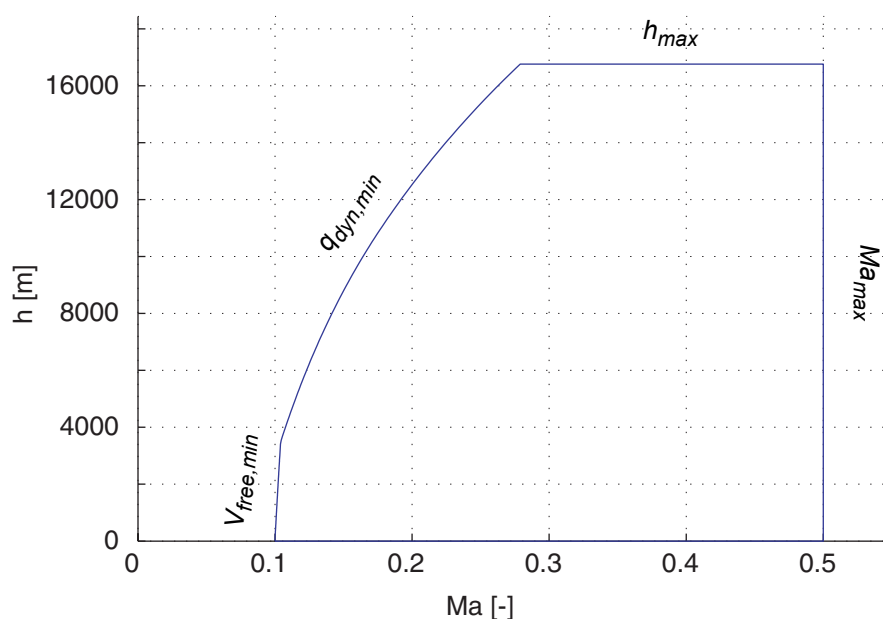


Abb. 2.2: Envelope des Trainers in $Ma-h$ -Darstellung

Die Begrenzungslinien ergeben sich direkt aus den Grenzen des Einsatzbereichs (h_{max} , Ma_{max}) bzw. aus den Kopplungen zwischen den Eingangsgrößen. In Abbildung 2.2 sind die Begrenzungslinien mit den entsprechenden limitierenden Größen versehen.

Eine alternative Darstellung ergibt sich in der $Ma-q_{dyn}$ -Ebene, diese ist in Abb. 2.3 dargestellt. Auch hier wurde auf die dritte Achse verzichtet.

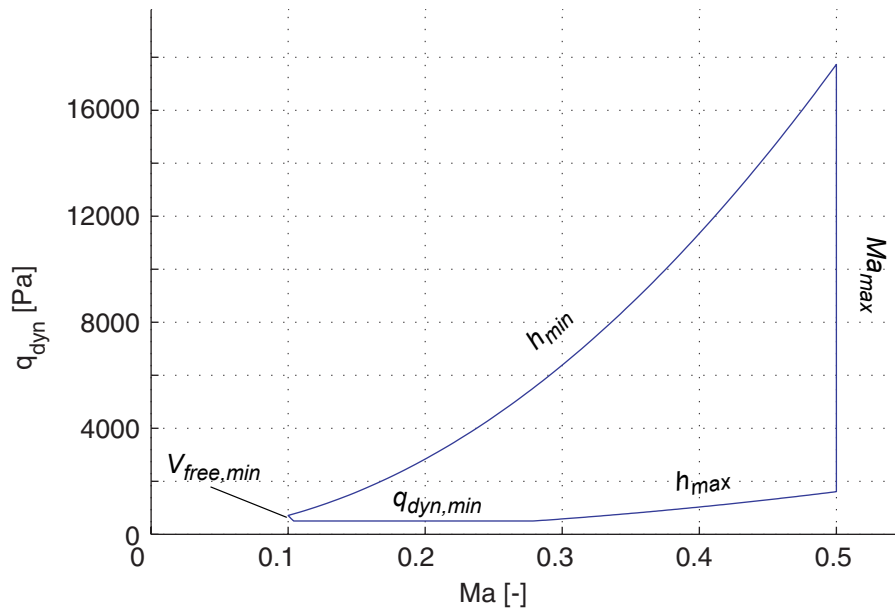


Abb. 2.3: Envelope des Trainers in $Ma-q_{dyn}$ -Darstellung.

2.1.6 Berechnung der Lastvielfachen

Laut Definition handelt es sich bei den Lastvielfachen um den Quotienten aus der Summe der am Flugzeug angreifenden Kräfte in x_f , y_f bzw. z_f -Richtung und der Gewichtskraft des Flugzeugs. Die einfachste Möglichkeit zur Berechnung der Lastvielfachen ist daher die Bestimmung der momentan am Flugzeug angreifenden aerodynamischen und schubbedingten Kräfte über das AeroData-Modul bzw. das Schubmodul. In zukünftigen Versionen des Aerodynamikmodells des Jettrainers sollen jedoch Einflüsse durch elastische Verformungen der Flugzeugstruktur auf die Aerodynamik berücksichtigt werden. Dies geschieht durch die Erweiterung der Eingänge des AeroData-Moduls um die drei Lastvielfachen, da die Verformungen des Flugzeugs von den Lastvielfachen abhängen. Das bedeutet jedoch, dass die Lastvielfachen in diesem Fall nicht mehr über die äußeren Kräfte berechnet werden können, da diese die aerodynamischen Kräfte beinhalten und somit eine algebraische Schleife vorliegen würde. Abhilfe bietet eine alternative Berechnung der Lastvielfachen aus einer Invertierung der Bewegungs-Differentialgleichungen, durch die sich die Lastvielfachen aus den Zustandsgrößen, der Zustandsableitung, der Flugzeugmasse und der Erdbeschleunigung berechnen lassen. Es gelten folgende Zusammenhänge:

$$n_x = \frac{1}{g} \cdot (T_1 \cos \beta \sin \alpha - T_2 \sin \beta + \cos \alpha \cdot (\dot{V}_{free} \cos \beta - T_3 \sin \beta) + g \sin \Theta) \quad (2.30)$$

$$n_y = \frac{\cos \beta}{g} \cdot (T_3 + T_2 \cos \alpha - T_4 \sin \alpha - T_5 \sin \Phi + T_6 / \cos \beta) \quad (2.31)$$

$$n_z = \frac{\cos\beta}{2g} \cdot (-2T_1 \cos\alpha - 2T_5 \cos\Theta + T_6 T_7 + 2T_4 \tan\beta - 2T_3 \sin\alpha \tan\beta - T_6 \tan(\beta)^2) \quad (2.32)$$

mit den Platzhaltern T_1 bis T_7 :

$$\begin{aligned} T_1 &= V_{free} \cdot (q - \dot{\alpha}) \\ T_2 &= r \cdot V_{free} \\ T_3 &= V_{free} \cdot \dot{\beta} \\ T_4 &= p \cdot V_{free} \\ T_5 &= \frac{g \cos\Theta}{\cos\beta} \\ T_6 &= \dot{V}_{free} \cdot \sin\alpha \\ T_7 &= 1 + 1/(\cos\beta)^2 \end{aligned} \quad (2.33)$$

Die Gleichungen (2.30), (2.31) und (2.32) wurden in der c-mex s-function `loads03` implementiert und stehen in Matlab-Simulink zur Verfügung.

2.1.7 Klappenfahr Gesetze

Aufgrund der Konfiguration der Steuerflächen des Jettrainers Mako (siehe Abb. 2.1) ist es möglich, dass sowohl das symmetrisch betätigte Taileron als auch die Vorder- und Hinterkantenklappen ein Nickmoment um die x_f -Achse erzeugen. Bei der Trimmung der Längsbewegung des Flugzeugs ergibt sich hieraus das Problem, dass die Steuerflächenstellungen für ein ausgeglichenes Moment um die y_f -Achse nicht eindeutig bestimmbar sind (siehe auch Trimmung in Abschnitt 2.2.1). In diesem Fall können die Stellungen sämtlicher Ruder und Klappen erst durch zusätzliche Bedingungen berechnet werden, wie beispielsweise durch die Forderung nach minimalem Widerstand im Trimmzustand.

Alternativ dazu können die Stellungen der Steuerelemente, die den kleinsten Einfluss auf das Nickmoment haben, jedoch auch entsprechend festgesetzten Fahr Gesetzen betätigt werden. Im vorliegenden Fall sind die Komponenten Vorderkantenklappe κ und Hinterkantenklappe ξ_{symm} des Steuervektors dementsprechend keine frei wählbaren Steuerungen, sondern werden im Fall der Vorderkante in Abhängigkeit des Anstellwinkels und im Fall der Hinterkante in Abhängigkeit des Anstellwinkels und der Machzahl verfahren. Laut [12] sind die Fahr Gesetze so gewählt, dass im gesamten Machzahl- und Anstellwinkelbereich ein möglichst ähnliches dynamisches Verhalten des Flugzeugs bezüglich der Stabilität der Anstellwinkelschwingung erreicht wird. Als Maß für die Stabilität bzw. Instabilität dient hierbei der Realteil des Pols der Anstellwinkelschwingung.

Abb. 2.4 zeigt eine graphische Darstellung der laut EADS zum AeroData-Modul ADM4.0 gehörenden Fahr Gesetze der Vorderkantenklappe sowie der Hinterkantenklappe für symmetrischen Ausschlag. Im betrachteten Machzahlbereich zwischen $Ma = 0,1$ und $Ma = 0,5$ tritt dabei keine Abhängigkeit von der Machzahl auf. Asymmetrische

Ausschläge der Hinterkantenklappe treten bei einer Anstellwinkelfolgerregelung nicht auf und werden daher im Rahmen dieser Arbeit nicht betrachtet.

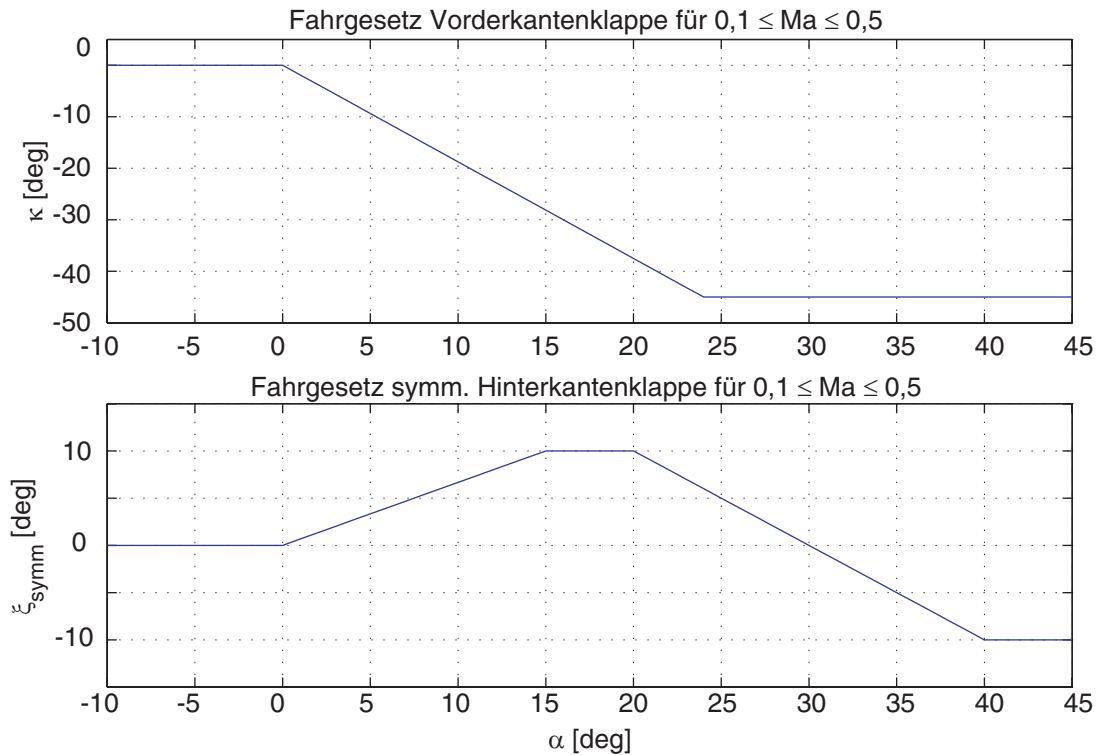


Abb. 2.4: Fahrgesetz für Vorder- und Hinterkantenklappe

2.1.8 Implementierung in Matlab Simulink

Ziel der Implementierung in Matlab Simulink ist es, für die Trimmung, die Linearisierung und die nichtlineare Simulation ein möglichst schnell aufrufbares und auswertbares Modell zu erstellen. Deshalb wurden sämtliche Module des nichtlinearen Gesamtmodells des Trainers durch c-mex s-functions realisiert. Diese stehen nach ihrer Kompilierung als dll-Dateien zur Verfügung, die wiederum in Simulink mit s-function Blöcken aufgerufen werden. Dabei ist die Dauer zur Auswertung einer solchen dll-Datei um ein Vielfaches kleiner als die Zeit, die zur Auswertung einer Matlab s-function oder gar einer Matlab-function benötigt wird, da diese nicht in kompilierter Form vorliegen, sondern erst zur Laufzeit interpretiert werden. Die einzelnen Module des Gesamtmodells und ihre zugehörigen s-functions sind in der folgenden Tabelle zusammengefasst:

Modul	s-function
6DOF Dynamik	dynamik08
Aerodynamik AeroDataModul ADM4.0	sfun_adm_v4_0
Schubmodul, F_{Einl} und F_{Einl}	sfun_mako_engine
Schubmodul, Kräfte und Momente	schub05
Schubmodul, trigonometrische Werte	trigon03
Klappenfahr Gesetze	flaps02

Modul	s-function
Lastvielfache	loads03
Atmosphäre	atmos04

Tabelle 2.6: Übersicht über Module des nichtlinearen Modells und zugehörige s-functions

Es ist möglich, c-mex s-functions so zu programmieren, dass sie zur Simulation eines dynamischen Systems verwendet werden können. Das heißt neben der Berechnung der Zustandsableitungen erfolgt in der s-function auch die Integration der Bewegungsgleichungen. Hierzu greift die s-function auf den in Matlab Simulink zur Simulation angegebenen Integrator zu und steuert diesen. Auf diese Möglichkeit wurde hier jedoch verzichtet. Das heißt, sämtliche verwendete s-functions sind sogenannte direct-feedthrough s-functions, die aus ihren Eingängen direkt, d.h. ohne Integration, ihre entsprechenden Ausgänge berechnen. Dies bietet zum einen den Vorteil eines sehr "durchschaubaren" Modells. Außerdem kann auf diese Art und Weise für die Aufgabenstellungen Trimmung, Linearisierung und nichtlineare Simulation (mit einem nachgeschalteten Integrator) dasselbe Basismodell verwendet werden.

Die Verschaltung der einzelnen Blöcke erfolgt über Bus-Signale. Ein solcher Bus wird zunächst über einen *BusCreator*-Block aus beliebigen Signalen erstellt und bietet den Vorteil, dass anschließend an jeder beliebigen Stelle jedes beliebige Signal über einen *BusSelector*-Block ausgelesen werden kann. Aufgrund der dadurch verringerten Zahl von einzelnen Signalen wird eine hohe Übersichtlichkeit des Modells erreicht.

Das im Rahmen dieser Arbeit für die Trimmung und Linearisierung verwendete Modell `model_2003_02sm.mdl` hat einen Eingangsvektor mit insgesamt 38 Komponenten und liefert einen Ausgangsvektor mit ebenfalls 38 Komponenten. Die folgenden Tabellen bieten eine Übersicht über ihre jeweiligen Zusammensetzungen:

	Komponente	Inhalt
Eingang	1-12	Zustandsvektor \dot{x}
	13-24	Zustandsableitung am Eingang \dot{x}_{in}
	25-32	Steuervektor \dot{u}
	33-38	zusätzliche Parameter, \dot{p}_{zus} (s.u.)
Ausgang	1-9	Zustandsvektor, Komponenten 1-9
	10-21	Zustandsableitung am Ausgang \dot{x}_{out}
	22-24	Geschwindigkeit \vec{V} in kartesischen Koordinaten
	25-27	Lastvielfache, über Kräfte berechnet
	28-30	Lastvielfache, über Eingänge berechnet
	31-36	Gesamtkraft \vec{F}_{ges} und Gesamtmoment \vec{M}_{ges}
	37	Erdbeschleunigung g
	38	ADM-Zustands-Flag (s.u.)

Tabelle 2.7: Zusammensetzung des Modell-Eingangsvektors und Ausgangsvektors

Der Vektor der zusätzlichen Parameter \dot{p}_{zus} beinhaltet die nicht benutzten Eingänge des AeroData-Moduls *gear* und *store*, die Schwerpunktkoordinaten x_{CG} , y_{CG} und z_{CG} sowie

den Parameter *switchmode*. Über dessen Einstellung wird gesteuert, ob die Vorder- und Hinterkantenklappenausschläge über das jeweilige Fahrgesetz berechnet werden oder ob der entsprechende Wert des Steuervektors \vec{u} verwendet werden soll. Diese Unterscheidung ist notwendig, um das Modell sowohl für die Trimmung als auch für die Linearisierung verwenden zu können (siehe hierzu auch die Abschnitte 2.2.1 Trimmung und 2.2.4 Linearisierung).

Bei der 38. Komponente des Ausgangsvektors, dem Zustands-Flag des AeroData-Moduls, handelt es sich um eine Information, ob für die Eingänge des ADM die aerodynamischen Kräfte und Momente berechnet werden konnten oder ob einer (oder mehrere) der Eingangswerte der Aerodynamik außerhalb seines zulässigen Wertebereiches lag. Der Ausgang kann also zu Kontrollzwecken verwendet werden.

Das Simulink-Modell liefert somit in Abhängigkeit des Zustands $\dot{\vec{x}}$, des Steuervektors \vec{u} , der Ableitung des Zustands am Eingang $\dot{\vec{x}}_{in}$ (wegen des Eingangs $\dot{\alpha}$ der Aerodynamik) sowie der Parameter \vec{p}_{zus} die Ableitung des Zustands $\dot{\vec{x}}_{out}$. Damit ist die Trimmung des Jettrainers möglich. Die weiteren Ausgänge dienen zu Kontrollzwecken bzw. werden bei der Linearisierung des Modells verwendet.

Abbildung A.1 in Anhang A.1 zeigt ein Blockdiagramm der ersten Ebene des Simulink-Modells.

2.2 Bildung des linearen Entwurfsmodells

Der folgende Abschnitt beschäftigt sich mit der Bildung eines linearen Entwurfsmodells ausgehend vom nichtlinearen Modell des Jettrainers. Hierzu wird das nichtlineare Modell zunächst getrimmt und anschließend linearisiert. Die linearisierte Form wird auf die Kurzzeitnäherung reduziert, mit der der Reglerentwurf durchgeführt wird.

2.2.1 Trimmung: Motivation

Die Trimmung dient zur Bestimmung stationärer oder quasi-stationärer Flugzustände, die wiederum als Ausgangs- bzw. Arbeitspunkte für die Linearisierung der Differentialgleichungen des Flugzeugs dienen. Ausgehend von einigen gegebenen Komponenten des Zustandsvektors, welche den zu trimmenden Flugzustand beschreiben, werden durch die Trimmung die restlichen Komponenten des Zustandsvektors sowie die veränderbaren Elemente des Steuervektors (diejenigen, die nicht durch Fahrgesetze bestimmt werden) so bestimmt, dass ausgewählte Komponenten der Zustandsableitung zu Null werden. Getrimmte Flugzustände mit verschwindenden Zustandsableitungen werden als Arbeitspunkte für die Linearisierung verwendet, weil linearisierte Gleichungen nur in einer bestimmten Umgebung um ihren Auslegungspunkt gültig sind, d.h. die (nichtlineare) Realität ausreichend gut abbilden. (Die Größe dieser Umgebung richtet sich dabei nach dem Grad der Nichtlinearität der betrachteten Gleichung.) Verschwindende Ableitungen bedeuten nun konstante Zustände, was zur Folge hat, dass sich der Zustand des Flugzeugs im Laufe der Zeit nicht vom Arbeitspunkt der Linearisierung entfernt und die Linearisierung somit gültig und verwendbar bleibt.

Die Trimmung dient zur Bestimmung stationärer oder quasi-stationärer Flugzustände, d.h. die veränderbaren Elemente des Steuervektors und des Zustandsvektors werden so bestimmt, dass die Zustandsableitung ganz oder teilweise verschwindet.

Ein solcher Flugzustand ist der stationäre Horizontalflug. Für einen Trimpunkt, der durch einen Anstellwinkel, eine Geschwindigkeit und eine Flughöhe definiert ist, sind in diesem Fall die Steuerflächen- und die Schubhebelstellung so zu bestimmen, dass die Zustandsableitungen sämtlicher Winkel und Drehraten sowie der Geschwindigkeit und der Position in z -Richtung verschwinden. Lediglich die Ableitungen der x_g - und y_g -Position des Flugzeugs sind ungleich Null, da sich das Flugzeug mit konstanter Geschwindigkeit bewegt. (Die Zustandsableitung verschwindet also nicht komplett. Die x_g, y_g -Position des Flugzeugs hat jedoch keinen Einfluss auf die übrigen Zustände und somit auch keinen Einfluss auf das linearisierte Modell.)

Für ein Hochleistungsflugzeug wie den Jettrainer Mako ist es jedoch nicht ausreichend, ausschließlich für den stationären Horizontalflug Modelle zu bestimmen und Flugregler zu entwerfen, denn das Flugzeug wird neben dem stationären Horizontalflug in vielen anderen Flugzuständen betrieben werden. Aus diesem Grund wird die Flug-Envelope des Trainers durch ein Raster über dem Anstellwinkel, der Machzahl und der Höhe oder alternativ dem Staudruck diskretisiert. An jedem Trimpunkt-Tripel erfolgt dann bei konstanter Schubhebelstellung eine sogenannte Abfangbogen- oder Nicktrimmung. Hierbei werden die Stellungen der nicht an Fahrgesetze gebundenen Steuerflächen so bestimmt, dass sich eine konstante Nickrate $q_0 = \text{const}$ ergibt, wobei hier und im Folgenden der Index 0 den getrimmten Zustand kennzeichnet. Die restlichen Drehraten r_0 und p_0 sowie der Schiebewinkel β_0 sind Null und ebenfalls konstant. Mit einem Rollwinkel $\Phi_0 = 0 = \text{const}$ folgt aus den Bewegungsgleichungen des Flugzeugs, dass gelten muss $\Theta = q_0$. (Unter der Annahme, dass sich das Flugzeug zum Zeitpunkt der Trimmung t_0 am Scheitel des Abfangbogens im Horizontalflug befindet, gilt ferner am Scheitel $\Theta = \alpha$.) Aufgrund der fixierten Schubhebelstellung ergibt sich eine Geschwindigkeitsableitung von $\dot{V}_{free,0} \neq 0$, d.h. das Flugzeug ändert seine Geschwindigkeit. Diese Form der Trimmung ist jedoch im Rahmen des durchzuführenden Reglerentwurfes trotzdem verwendbar, denn der Regler stabilisiert die Anstellwinkelschwingung, an der die Geschwindigkeit jedoch kaum beteiligt ist.

Aufgrund der ständigen Änderungen von Geschwindigkeit und Höhe und der damit verbundenen ständigen Änderung des Staudrucks und der Machzahl wechselt das Flugzeug stets von einem Trimmzustand in den nächsten (wobei im Allgemeinen jedoch nicht gilt $\Theta = \alpha$).

Eine Alternative zur Vorgabe eines konstanten Anstellwinkels ist die Vorgabe eines konstanten Lastvielfachen n_z . Auch in diesem Fall ist der Trimm-Anstellwinkel konstant, er ergibt sich jedoch aus der Größe des Lastvielfachen sowie der Machzahl und dem Staudruck bzw. der Flughöhe des Trimmzustands.

Anmerkung: Der stationäre Horizontalflug ist ein Spezialfall des quasi-stationären Abfangbogens, denn für die Anstellwinkel-Machzahl-Staudruck-Kombination, für die sich im Trimmfall eine konstante Nickrate von $q_0 = 0$ bei einer Geschwindigkeitsänderung von $\dot{V}_{free,0} = 0$ ergibt, erhält man gerade den stationären Horizontalflug. Es ist jedoch zu

beachten, dass durch die Vorgabe einer festen Schubhebelstellung dieser Zustand nur an speziellen Punkten der Envelope auftritt. Bei der Summe aller möglichen Punkte handelt es sich um eine Linie in der Envelope, denn für einen vorgegebenen Anstellwinkel ergeben sich zwar verschiedene Kombinationen von Ma und q_{dyn} , für die entweder gilt $q_0 = 0$ oder $\dot{V}_{free,0} = 0$. Beide Werte werden jedoch nur an genau einer Machzahl-Staudruck-Kombination gleichzeitig zu Null.

Das heißt, für eine feste Schubhebelstellung kann von den drei Größen Anstellwinkel, Machzahl und Staudruck nur eine vorgegeben werden, die beiden anderen ergeben sich aus der Vorgabe des stationären Horizontalfluges, sofern dieser überhaupt möglich ist. (Auch für variable Schubhebelstellung können nicht alle drei Größen frei vorgegeben werden, sondern lediglich zwei, wobei sich die dritte ebenfalls automatisch ergibt, sofern für die Vorgabe ein stationärer Horizontalflug möglich ist.)

2.2.2 Trimmung: prinzipielle Durchführung

Die Trimmung erfolgt durch die Lösung ausgewählter, zu Null gesetzter Gleichungen des Systems der nichtlinearen Bewegungs-Differentialgleichungen. Welche der Gleichungen zu verwenden sind und welche Eingänge dieser Gleichungen zu bestimmen sind, richtet sich nach dem Trimm-Modus, d.h. der Art der durchzuführenden Trimmung.

Hierbei ist zu beachten, dass es sich im vorliegenden Fall um implizite Differentialgleichungen handelt. Die Ableitung des Anstellwinkels ist nicht nur Ausgang der Differentialgleichungen, sondern über die Aerodynamik auch ein Eingang. Für die bisher erwähnten Trimmfälle, dem stationären Horizontalflug und dem quasi-stationären Abfangbogen stellt dies kein Problem dar, denn in beiden Fällen gilt im getrimmten Zustand $\dot{\alpha}_0 = 0$, d.h. die Anstellwinkelableitung im Trimmzustand ist bekannt und somit auch der entsprechende Eingang der Differentialgleichungen. Ein Problem besteht jedoch im Zusammenhang mit den Lastvielfachen. Wie bereits in Abschnitt 2.1.6 erwähnt wurde, sind die Lastvielfachen als Eingänge zukünftiger Aerodynamikmodelle vorgesehen. Gleichung (2.32) zeigt, dass es sich beim Lastvielfachen n_z um eine Funktion der folgenden Art handelt:

$$n_z = f(V_{free}, \alpha, \beta, p, q, \Phi, \Theta, g, \dot{V}_{free}, \dot{\alpha}, \dot{\beta}) \quad (2.34)$$

Hierbei sind die Größen V_{free} , α , β , p , Φ , Θ , g , $\dot{\alpha}$ und $\dot{\beta}$ im getrimmten Zustand bekannt, entweder über den zu trimmenden Flugzustand (V_{free} , α , β , p , Φ , Θ und g) oder über die Bedingungen für getrimmten Flug ($\dot{\alpha} = 0$ und $\dot{\beta} = 0$). Die Werte der Größen \dot{V}_{free} und q , die sich im getrimmten Flug ergeben, sind jedoch nicht bekannt. Das bedeutet, es liegt eine algebraische Schleife vor, denn zur Berechnung der Werte für \dot{V}_{free} und q im getrimmten Zustand werden die betreffenden Werte selbst benötigt.

Zur Lösung dieses Problems wird das Trimmproblem mit einem Vektor der Zustandsableitung auf der Eingangsseite des Modell, $\dot{\mathbf{x}}_{in}$ und einem Vektor der Zustandsableitung auf der Ausgangsseite der Modellgleichungen, $\dot{\mathbf{x}}_{out}$ formuliert. Anstatt nun lediglich die linken Seiten der Modellgleichungen, d.h. die Zustandsableitung $\dot{\mathbf{x}}_{out}$ zu Null zu setzen, wird verlangt, dass gilt

$$\dot{\hat{x}}_{out} = \dot{\hat{x}}_{in} \quad (2.35)$$

das heißt, die Differenz zwischen Ausgang der Modell-Differentialgleichungen und Zustandsableitung auf der Eingangsseite wird zu Null gesetzt:

$$\dot{\hat{x}}_{out} - \dot{\hat{x}}_{in} = 0 \quad (2.36)$$

Zu diesem Satz von zwölf Gleichungen kommen für die Lastvielfachen drei Differenzen zwischen \hat{n}_{in} , berechnet aus Zustand und Zustandsableitung gemäß den Gleichungen (2.30), (2.31) und (2.32) sowie der klassischen Berechnung \hat{n}_{out} aus dem Quotienten aus Gesamtkraft und Gewichtskraft:

$$\begin{cases} \dot{\hat{x}}_{out} - \dot{\hat{x}}_{in} = 0 \\ \hat{n}_{out} - \hat{n}_{in} = 0 \end{cases} \quad (2.37)$$

Anstatt die zum Trimmfall gehörenden Gleichungen aus den Modell-Differentialgleichungen (2.7), (2.11), (2.14) und (2.15) zu lösen, werden die unbekannt Zustandsvariablen, Zustandsableitungen und Steuerungen bestimmt, indem die entsprechenden Gleichungen des Systems (2.37) gelöst werden.

Die Lösung des Gleichungssystems erfolgt über die Matlab Funktion `fsolve_simple`, die auf einem Newton-Verfahren basiert. Hierbei ist zu beachten, dass dieses Verfahren nur zur Lösung von eindeutig bestimmten Systemen verwendet werden kann. Es können also nur genauso viele Gleichungen gelöst werden, wie auch Freiheitsgrade, d.h. Variablen, zur Verfügung stehen.

Es folgt eine Übersicht über verschiedene Trimmfälle, die jeweils gegebenen Größen, die zu bestimmenden Variablen und die dazu verwendeten Gleichungen aus (2.36). Die gegebenen Größen stammen dabei aus dem Flugzustand, für den das Flugzeug getrimmt wird, und aus den Trimmbedingungen für den jeweiligen Trimmfall.

Obwohl es sich ausschließlich um Trimmfälle der Längsbewegung handelt, müssen auch die Stellungen der primär zur Seitenbewegung gehörenden Steuerflächen berechnet werden, da das vorliegende Aerodynamikmodell asymmetrisch ist. Für einen symmetrischen Flugzustand, das heißt eine symmetrische Anströmung und symmetrische Klappenausschläge treten Roll- und Giermomente sowie eine Querkraft auf, welche durch entsprechende Steuerausschläge kompensiert werden müssen.

Trimmfall 1: quasi-stationärer Abfangbogen mit gegebenem Anstellwinkel

Im Falle der Nicktrimmung mit gegebenem α , die als Basis für die Bestimmung von linearen Modellen für den Reglerentwurf dient, erfolgt die Spezifizierung der zu trimmenden Flugzustände wie bereits erwähnt über ein Raster aus Anstellwinkel, Machzahl und Höhen- bzw. Staudruckwerten. Das heißt, die Flughöhe und damit die z_g -Komponente des zu trimmenden Flugzustands folgt im Falle einer Spezifizierung über die Höhe direkt aus dem Höhenraster. Im Falle einer Spezifizierung über den Staudruck wird die zugehörige Höhe über die Kopplungen des Atmosphärenmodells aus der Machzahl und dem Staudruck

berechnet. Die Fluggeschwindigkeit V_{free} des Trimpunktes ergibt sich in beiden Fällen aus der Machzahl und der zur Flughöhe gehörenden Schallgeschwindigkeit. Die gegebenen Größen, die zu bestimmenden Größen sowie die zu verwendenden Gleichungen sind in Tabelle 2.8 zusammengefasst:

gegeben in \hat{x} aus Trimmzustand	V_{free}, α, z_g
gegeben in \hat{x} aus Trimmbedingung	$\beta = 0, p = 0, r = 0, \Phi = 0, \Theta = \alpha, \Psi = 0$
gegeben in \hat{x} aus Trimmbedingung	$\dot{\alpha} = 0, \dot{\beta} = 0, \dot{p} = 0, \dot{q} = 0, \dot{r} = 0,$ $\Phi = 0, \Theta = q, \Psi = 0, \dot{x} = V_{free}, \dot{y} = 0, \dot{z} = 0$
zu bestimmen in \hat{x}	q
zu bestimmen in \hat{x}	$\dot{V}_{free}, (\Theta = q),$
zu bestimmen in \hat{u}	η_s, η_d, ζ
verwendete Gleichungen	$\dot{V}_{free}, \dot{\alpha}, \dot{p}, \dot{q}, \dot{r}$

Tabelle 2.8: Trimmfall 1, gegebene und unbekannte Größen, zu lösende Gleichungen

Trimmfall 2: Stationärer Horizontalflug mit flexiblem Anstellwinkel

Die Spezifizierung von zu trimmenden Flugzuständen erfolgt bei diesem Trimmfall durch die Angabe einer Machzahl sowie einer Höhe bzw. wiederum eines Staudrucks. Im Falle einer Staudruckangabe wird die Höhe wiederum aus Machzahl und zugehörigem Staudruckwert berechnet. Anschließend erfolgt aus der Machzahl und der Flughöhe die Berechnung der zugehörigen Fluggeschwindigkeit. Für diese Bedingungen ergibt sich bei verstellbarem Schub ein Anstellwinkel, für den der stationäre Horizontalflug möglich ist.

Die bekannten Größen, die zu bestimmenden und die zu verwendenden Gleichungen sind in Tabelle 2.9 zusammengefasst.

gegeben in \hat{x} aus Trimmzustand	V_{free}, z_g
gegeben in \hat{x} aus Trimmbedingung	$\beta = 0, p = 0, q = 0, r = 0, \Phi = 0, \Psi = 0$
gegeben in \hat{x} aus Trimmbedingung	$\dot{V}_{free} = 0, \dot{\alpha} = 0, \dot{\beta} = 0, \dot{p} = 0, \dot{q} = 0, \dot{r} = 0,$ $\Phi = 0, \Theta = q, \Psi = 0, \dot{x} = V_{free}, \dot{y} = 0, \dot{z} = 0$
zu bestimmen in \hat{x}	$\alpha, (\Theta = \alpha)$
zu bestimmen in \hat{x}	--
zu bestimmen in \hat{u}	$\eta_s, \eta_d, \zeta, \delta$
verwendete Gleichungen	$\dot{V}_{free}, \dot{\alpha}, \dot{p}, \dot{q}, \dot{r}$

Tabelle 2.9: Trimmfall 2, gegebene und unbekannte Größen, zu lösende Gleichungen

Trimmfall 3: quasi-stationärer Abfangbogen mit gegebenem Lastvielfachen

Wie bereits erwähnt, kann der stationäre Abfangbogen anstatt über einen Anstellwinkel auch über ein Lastvielfaches n_z spezifiziert werden. Auch hier erfolgt die sonstige Spezifizierung des zu trimmenden Flugzustands über eine Machzahl sowie eine Flughöhe oder einen Staudruck. Somit ist die Flughöhe entweder gegeben oder kann aus Machzahl und Staudruck berechnet werden. Die Fluggeschwindigkeit wird wiederum aus der Machzahl und der

Flughöhe berechnet. Die bekannten, die zu bestimmenden Größen sowie die zu verwendenden Gleichungen sind in Tabelle 2.10 zusammengefasst:

gegeben in \hat{x} aus Trimmzustand	$V_{free}, z_g, (n_{z,out})$
gegeben in \hat{x} aus Trimmbedingung	$\beta = 0, p = 0, q = 0, r = 0, \Phi = 0, \Psi = 0$
gegeben in $\dot{\hat{x}}$ aus Trimmbedingung	$\dot{\alpha} = 0, \dot{\beta} = 0, \dot{p} = 0, \dot{q} = 0, \dot{r} = 0,$ $\dot{\Phi} = 0, \dot{\Psi} = 0, \dot{x} = V_{free}, \dot{y} = 0, \dot{z} = 0$
zu bestimmen in \hat{x}	$\alpha, q, (\Theta = \alpha)$
zu bestimmen in $\dot{\hat{x}}$	$\dot{V}_{free}, (\Theta = q)$
zu bestimmen in \hat{u}	η_s, η_d, ζ
verwendete Gleichungen	$\dot{V}_{free}, \dot{\alpha}, \dot{p}, \dot{q}, \dot{r}, n_{z,in} = n_{z,out}$

Tabelle 2.10: Trimmfall 3, gegebene und unbekannte Größen, zu lösende Gleichungen

Trimmfall 4: stat. Horizontalflug, variable Geschwindigkeit und Flughöhe, fester Schub

In diesem Fall wird ein zu trimmender Flugzustand ausschließlich über seinen Anstellwinkel spezifiziert. Da der Schub nicht variiert wird, ergibt sich eine Höhen-Geschwindigkeits-Kombination, in welcher der stationäre Horizontalflug möglich ist (siehe auch 2.1.1).

Die bekannten und unbekanntenen Größen sowie die Gleichungen zu deren Bestimmung sind in Tabelle 2.11 zusammengefasst:

gegeben in \hat{x} aus Trimmzustand	α
gegeben in \hat{x} aus Trimmbedingung	$\beta = 0, p = 0, q = 0, r = 0, \Phi = 0, \Psi = 0$
gegeben in $\dot{\hat{x}}$ aus Trimmbedingung	$\dot{V}_{free} = 0, \dot{\alpha} = 0, \dot{\beta} = 0, \dot{p} = 0, \dot{q} = 0, \dot{r} = 0,$ $\dot{\Phi} = 0, \dot{\Theta} = 0, \dot{\Psi} = 0, \dot{x} = 0, \dot{y} = 0, \dot{z} = 0$
zu bestimmen in \hat{x}	V_{free}, z_g
zu bestimmen in $\dot{\hat{x}}$	--
zu bestimmen in \hat{u}	η_s, η_d, ζ
verwendete Gleichungen	$\dot{V}_{free}, \dot{\alpha}, \dot{p}, \dot{q}, \dot{r}$

Tabelle 2.11: Trimmfall 4, gegebene und unbekannte Größen, zu lösende Gleichungen

Zusammengefasst ergibt sich das Schema in Tabelle 2.12. Hier sind nochmals für die vier Trimmfälle die bekannten sowie die zu bestimmenden Elemente des Zustandsvektors, des Steuervektors sowie der Zustandsableitung mit den zur Lösung verwendeten Gleichungen zusammengestellt.

TF	Zustand \hat{x}						Zustandsableitung $\dot{\hat{x}}$						Steuerung \hat{u}				Gleichungen						
	V_{free}	α	β	p	q	r	\dot{V}_{free}	$\dot{\alpha}$	$\dot{\beta}$	\dot{p}	\dot{q}	\dot{r}	η_s	η_d	ζ	κ	\dot{V}_{free}	$\dot{\alpha}$	\dot{p}	\dot{q}	\dot{r}	n_z	
	Φ	Θ	Ψ	x_g	y_g	z_g	$\dot{\Phi}$	$\dot{\Theta}$	$\dot{\Psi}$	\dot{x}_g	\dot{y}_g	\dot{y}_g	ξ_s	ξ_d	η_{St}	δ_n							
1	+	+	0	0	v_1	0	v_2	0	0	0	0	0	v_3	v_4	v_5	+	g_1	g_2	g_3	g_4	g_5	-	
	0	+	0	0	0	+	0	v_1	0	+	0	0	+	0	0	1							
2	+	v_1	0	0	0	0	0	0	0	0	0	0	v_2	v_3	v_4	+	g_1	g_2	g_3	g_4	g_5	-	
	0	v_1	0	0	0	+	0	0	0	+	0	0	+	0	0	v_5							

TF	Zustand \hat{x}						Zustandsableitung $\dot{\hat{x}}$						Steuerung \hat{u}				Gleichungen						
	V_{free}	α	β	p	q	r	\dot{V}_{free}	$\dot{\alpha}$	$\dot{\beta}$	\dot{p}	\dot{q}	\dot{r}	η_s	η_d	ζ	κ	\dot{V}_{free}	$\dot{\alpha}$	\dot{p}	\dot{q}	\dot{r}	n_z	
	Φ	Θ	Ψ	x_g	y_g	z_g	$\dot{\Phi}$	$\dot{\Theta}$	$\dot{\Psi}$	\dot{x}_g	\dot{y}_g	\dot{z}_g	ξ_s	ξ_d	η_{St}	δ_n							
3	+	v_1	0	0	v_2	0	v_3	0	0	0	0	0	v_4	v_5	v_6	+	g_1	g_2	g_3	g_4	g_5	g_6	
	0	v_1	0	0	0	+	0	v_2	0	+	0	0	+	0	0	1							
4	v_1	+	0	0	0	0	0	0	0	0	0	0	v_3	v_4	v_5	+	g_1	g_2	g_3	g_4	g_5	-	
	0	+	0	0	0	v_2	0	0	0	v_1	0	0	+	0	0	1							
+ aus Trimm-FZ						v_i Variable						g_i Gleichung				0/1 Trimmbedingung							

Tabelle 2.12: Schema über bekannte und unbekannte Elemente sowie zu verwendende Gleichungen

2.2.3 Trimmung: Implementierung in Matlab

Die Implementierung der Trimmung in Matlab erfolgt über die Trimm-Routine `trimenvqdyn`, welche wie nachfolgend erklärt diverse Subroutinen aufruft. Zwar bietet Matlab mit dem Befehl `trim` eine eigene Trimm-Routine an, diese hat jedoch gegenüber einer Trimmung mittels `trimenvqdyn` einige gravierende Nachteile:

- **Geschwindigkeit.** Sowohl die Matlab Trimm-Routine `trim` als auch die Routine `trimenvqdyn` benötigen zur numerischen Lösung der Trimmgleichungen zahlreiche Aufrufe des Simulink-Modells, das die nichtlinearen Bewegungs-Differentialgleichungen enthält (im vorliegenden Fall also das Modell `model_2003_02sm.mdl`). Beim Aufruf eines Simulink-Modells zur Berechnung seiner Ausgänge muss das Modell zunächst kompiliert werden. Anschließend erfolgt die eigentliche Ausgangsberechnung, abschließend muss das Modell terminiert werden. Bei Verwendung des Matlab Trimbefehls erfolgt diese Abfolge bei jedem Aufruf des Befehls `trim`, d.h. bei jedem zu trimmenden Flugzustand. Im Falle der Routine `trimenvqdyn` erfolgt die Kompilierung des zugehörigen Simulinkmodells bzw. dessen Terminierung dagegen nur einmal zu Beginn bzw. am Ende der Routine, bei der Bearbeitung der zu trimmenden Flugzustände kann das Modell also direkt aufgerufen werden, ohne ständiges, zeitraubendes Kompilieren und Terminieren.
- **Implizität der Bewegungsgleichungen.** Mit der Matlab Trimm-Routine ist es nicht möglich, die algebraische Schleife im Falle einer lastabhängigen Aerodynamik aufzubrechen.
- **Verfahren bei nicht trimmbaren Punkten.** Es ist möglich, dass ein Flugzustand mit den gegebenen Vorgaben für den gegebenen Trimmfall nicht getrimmt werden kann. In diesem Fall muss der Punkt als nicht trimmbar gekennzeichnet werden. Dies geschieht bei der Trimm-Routine `trimenvqdyn`. Die Matlab-Trimm-Routine beginnt dagegen in diesem Fall, die Trimmbedingungen zu lockern und versucht, die Bedingungen optimal zu erfüllen. Eine Kennzeichnung als nicht trimmbarer Punkt erfolgt nicht.

Die oben aufgeführten Nachteile der Matlab Trimm-Routine führen dazu, dass die Trimmung des Jettrainers Mako im Rahmen dieser Arbeit über die Routine `trimenvqdyn` erfolgt. Dort wird zunächst die Art der durchzuführenden Trimmung (d.h. der Trimmfall) festgelegt und ob über einem Höhen- oder einem Staudruckraster getrimmt werden soll. Dann werden die zu trimmenden Flugzustände spezifiziert, indem zunächst Listen mit zu

trimmenden Anstellwinkeln, Machzahlen, Staudrücken, Flughöhen und Lastvielfachen erstellt werden. In Abhängigkeit des ausgewählten Trimmfalles und der Rasterung wird aus den entsprechenden Listen über die Funktion `trimplist02` durch Bildung aller möglichen Kombinationen eine Matrix aller zu trimmenden Flugzustände erstellt. Es erfolgt nun für jeden Trimpunkt die Bestimmung des Envelope-Status, d.h. für sämtliche Trimm-Flugzustände erfolgt eine Kennzeichnung, ob sie sich innerhalb oder außerhalb der Envelope befinden. Es können nur Punkte innerhalb der Envelope getrimmt werden, außerhalb liegende Punkte werden bei der eigentlichen Trimmung übersprungen. Zu beachten ist, dass im Trimmfall 1 zu diesem Zeitpunkt das Lastvielfache n_z noch nicht bekannt ist. Dementsprechend erfolgt die Überprüfung, ob ein Trimpunkt innerhalb oder außerhalb der Envelope liegt ohne Berücksichtigung der Begrenzung des Lastvielfachen.

Durch den Aufruf der Unterfunktion `trimfunction` erfolgt nun für jeden Trimpunkt innerhalb der Envelope die eigentliche Trimmung. Wie im vorangegangenen Abschnitt 2.2.2 erläutert, basiert die Trimmung auf der Lösung eines nichtlinearen Gleichungssystems durch die Matlab Funktion `fsolve_simple`, die wiederum auf dem Newton-Verfahren basiert. Welche der Ausgänge des nichtlinearen Modells `model_2003_02sm.mdl` zur Bildung der Linke-Hand-Seiten (LHS) des zu lösenden Gleichungssystems verwendet werden und bei welchen Eingängen des Modells es sich um die zu bestimmenden Variablen des Systems handelt, hängt vom ausgewählten Trimmfall ab und wird über die Funktion `trimpars` spezifiziert. Dieser Funktion liegen die Tabellen 2.8 bis 2.11 zu Grunde, d.h. es wird spezifiziert, welche der Elemente des Zustandsvektors \hat{x} , des Steuervektors \hat{u} und des Vektors der Zustandsableitung am Eingang $\dot{\hat{x}}_{in}$ entweder aufgrund des zu trimmenden Flugzustands oder aufgrund der Trimmbedingungen bekannt sind und welche Elemente zu bestimmen sind. Auch die aus dem System (2.37) zur Lösung des Problems tatsächlich zu verwendenden Gleichungen werden hier festgelegt. Der eigentliche Aufruf des Modells und die Auswertung der Modellausgänge zur Generierung der LHS erfolgt in der Funktion `simple_adm40_fun`.

Das Newton-Verfahren in `fsolve_simple` benötigt zur Lösung des Gleichungssystems Startwerte. Hierbei können wahlweise entweder Standard-Startwerte verwendet werden, oder es werden bei jedem Trimpunkt die Ergebnisse des vorangegangenen Punktes als Startwerte verwendet (sofern diese vorliegen).

Die Kennzeichnung, ob ein Trimpunkt erfolgreich getrimmt werden konnte oder nicht, erfolgt über eine Status-Variable. Falls im Trimmfall 2, das heißt bei variabler Schubhebelstellung, der benötigte Schub zum Erreichen des stationären Horizontalflugs den maximalen Trockenschub übersteigt, erfolgt ebenfalls eine Kennzeichnung (*MaxDryEx*-Status).

Nach der Lösung des Trimmproblems ist nun auch im Trimmfall 1 das zum getrimmten Flugzustand gehörende Lastvielfache n_z bekannt. Deshalb erfolgt nun für jeden Trimpunkt eine Aktualisierung des Envelope-Status, wobei jetzt auch das Lastvielfache berücksichtigt wird. Die Ergebnisse der Trimmung werden für jeden Trimpunkt in dem Cell-Array `trimarray` abgelegt. Das Array hat dementsprechend so viele Zeilen, wie Flugzustände zu

trimmen sind, und setzt sich in seinen Spalten entsprechend Tabelle 2.13 zusammen. Die ungeraden Spalten beinhalten jeweils eine Beschreibung des Inhalts der geraden.

Spalte Nr.	Inhalt	Typ
1/2	Trimpunkt	Vektor (1 x 5)
3/4	Zustandsvektor im getrimmten Flug	Vektor (12 x 1)
5/6	Ableitung Zustandsvektor im getrimmten Flug am Eingang des Modells	Vektor (12 x 1)
7/8	Ableitung Zustandsvektor im getrimmten Flug am Ausgang des Modells	Vektor (12 x 1)
9/10	Steuervektor im getrimmten Flug	Vektor (8 x 1)
11/12	Lastvielfache im getrimmten Flug	Vektor (3 x 1)
13/14	Norm des Gleichungssystems im getrimmten Zustand	Skalar
15/16	Trimmfall	Skalar
17/18	Status	Skalar
19/20	Envelope-Status	Skalar
21/22	MaxDryEx-Status	Skalar

Tabelle 2.13: Zusammensetzung der Spalten der Ausgabevariablen *trimarray*

Nach der Bearbeitung aller Trimpunkte erfolgt die Abspeicherung der Trimmdatei als Matlab-Workspace. Hierzu wird aus dem Trimmfall sowie aus den Daten des Trimm-Rasters ein Dateinamensvorschlag generiert. Dieser ermöglicht es, den Inhalt eines Trimmdatei-Workspaces von seinem Namen abzuleiten.

Ein Nassi-Shneiderman-Diagramm der Routine `trimenvqdyn` befindet sich in Anhang B.1.

2.2.4 Linearisierung: mathematische Grundlagen

Die Linearisierung, d.h. die Bildung einer linearen Zustandsraumdarstellung des Flugzeugmodells, erfolgt durch die Bildung der Taylor-Reihenentwicklung der nichtlinearen Gleichungen, wobei nur die Glieder bis zur ersten Ordnung berücksichtigt werden.

Gegeben ist das System der nichtlinearen Differentialgleichungen nach Gleichung (2.6), das zur Erinnerung an dieser Stelle nochmals angeführt wird:

$$\dot{\hat{x}} = \hat{f}(\hat{x}, \hat{u}, \dot{\hat{x}}) \quad (2.38)$$

Neben den Zustandsableitungen liefert das nichtlineare Modell auch zusätzliche Ausgänge wie beispielsweise die Lastvielfachen n_x , n_y und n_z . Diese Ausgänge \hat{y} hängen ebenfalls vom Zustand \hat{x} , der Steuerung \hat{u} und der Zustandsableitung $\dot{\hat{x}}$ ab, es gilt allgemein:

$$\hat{y} = \hat{g}(\hat{x}, \hat{u}, \dot{\hat{x}}). \quad (2.39)$$

Gesucht wird nun eine lineare Zustandsraumdarstellung des Modells in der Form

$$\Delta \dot{\hat{x}} = A \Delta \hat{x} + B \Delta \hat{u} \quad (2.40)$$

sowie

$$\Delta \dot{y} = C \Delta \dot{x} + D \Delta \dot{u} \quad (2.41)$$

Bei den relativen Größen $\Delta \dot{x}$, $\Delta \dot{u}$ und $\Delta \dot{y}$ handelt es sich um die Abweichungen zum Auslegungs- bzw. Arbeitspunkt x_0 , u_0 und y_0 der Linearisierung, der im Rahmen der Trimmung bestimmt wurde.

Zur Berechnung der Zustandsableitung in der linearen Form nach Gleichung (2.40) wird Gleichung (2.38) zunächst umgeformt:

$$0 = \dot{f}(\dot{x}, \dot{u}, \dot{x}) - \dot{x} = \vec{F}(\dot{x}, \dot{u}, \dot{x}) \quad (2.42)$$

Die Taylor-Reihenentwicklung von (2.42) liefert:

$$0 = \underbrace{\frac{\partial \vec{F}(\dot{x}_0, \dot{u}_0, \dot{x}_0)}{\partial \dot{x}}}_{A_1} \Delta \dot{x} + \underbrace{\frac{\partial \vec{F}(\dot{x}_0, \dot{u}_0, \dot{x}_0)}{\partial \dot{x}}}_{E} \Delta \dot{x} + \underbrace{\frac{\partial \vec{F}(\dot{x}_0, \dot{u}_0, \dot{x}_0)}{\partial \dot{u}}}_{B_1} \Delta \dot{u} \quad (2.43)$$

bzw.

$$0 = A_1 \cdot \Delta \dot{x} + E \cdot \Delta \dot{x} + B_1 \cdot \Delta \dot{u}. \quad (2.44)$$

mit der Jacobimatrix A_1 , bestehend aus den partiellen Ableitungen von \vec{F} nach \dot{x} , der Matrix E mit den partiellen Ableitungen von \vec{F} nach \dot{x} sowie der Matrix C_1 mit den partiellen Ableitungen von \vec{F} nach \dot{u} .

Aufgelöst nach $E \Delta \dot{x}$ folgt

$$-E \Delta \dot{x} = A_1 \Delta \dot{x} + B_1 \Delta \dot{u} \quad (2.45)$$

und unter der Voraussetzung, dass E invertierbar ist, folgt schließlich

$$\Delta \dot{x} = -E^{-1} A_1 \Delta \dot{x} - E^{-1} B_1 \Delta \dot{u} \quad (2.46)$$

bzw.

$$\Delta \dot{x} = A \Delta \dot{x} + B \Delta \dot{u} \quad (2.47)$$

mit

$$A = -E^{-1} A_1 \quad (2.48)$$

und

$$B = -E^{-1} B_1. \quad (2.49)$$

Zur Berechnung der Elemente der jeweiligen Matrizen reicht dabei die Betrachtung der Elemente der Funktion \dot{f} aus, denn es gilt für A_1

$$A_1 = \left. \frac{\partial \vec{F}}{\partial \vec{x}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \vec{x}} \right|_0 - \left. \frac{\partial \dot{\vec{x}}}{\partial \vec{x}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \vec{x}} \right|_0 - 0 \quad (2.50)$$

und für E

$$E = \left. \frac{\partial \vec{F}}{\partial \dot{\vec{x}}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \dot{\vec{x}}} \right|_0 - \left. \frac{\partial \dot{\vec{x}}}{\partial \dot{\vec{x}}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \dot{\vec{x}}} \right|_0 - I \quad (2.51)$$

sowie für B_1

$$B_1 = \left. \frac{\partial \vec{F}}{\partial \vec{u}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \vec{u}} \right|_0 - \left. \frac{\partial \dot{\vec{x}}}{\partial \vec{u}} \right|_0 = \left. \frac{\partial \vec{f}}{\partial \vec{u}} \right|_0 - 0. \quad (2.52)$$

Analog gilt für die Ausgangsgleichung (2.41)

$$0 = \vec{g}(\vec{x}, \vec{u}, \dot{\vec{x}}) - \vec{y} = \vec{G}(\vec{x}, \vec{u}, \dot{\vec{x}}, \vec{y}). \quad (2.53)$$

In diesem Fall gilt für die Taylor-Entwicklung am Arbeitspunkt (Index 0)

$$0 = \underbrace{\left. \frac{\partial \vec{G}}{\partial \vec{x}} \right|_0}_{C_1} \Delta \vec{x} + \underbrace{\left. \frac{\partial \vec{G}}{\partial \dot{\vec{x}}} \right|_0}_{H} \Delta \dot{\vec{x}} + \underbrace{\left. \frac{\partial \vec{G}}{\partial \vec{u}} \right|_0}_{D_1} \Delta \vec{u} + \underbrace{\left. \frac{\partial \vec{G}}{\partial \vec{y}} \right|_0}_{J} \Delta \vec{y} \quad (2.54)$$

bzw.

$$0 = C_1 \Delta \vec{x} + H \Delta \dot{\vec{x}} + D_1 \Delta \vec{u} - J \Delta \vec{y}. \quad (2.55)$$

Analog zum Fall der Matrizen A_1 , E und B_1 gilt hier, dass zur Berechnung der partiellen Ableitungen der Jacobimatrizen nur die Ableitungen der Funktion \vec{g} relevant sind. Es gilt für die Matrix C_1 :

$$C_1 = \left. \frac{\partial \vec{G}}{\partial \vec{x}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \vec{x}} \right|_0 - \left. \frac{\partial \vec{y}}{\partial \vec{x}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \vec{x}} \right|_0 - 0, \quad (2.56)$$

für die Matrix H gilt

$$H = \left. \frac{\partial \vec{G}}{\partial \dot{\vec{x}}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \dot{\vec{x}}} \right|_0 - \left. \frac{\partial \vec{y}}{\partial \dot{\vec{x}}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \dot{\vec{x}}} \right|_0 - 0, \quad (2.57)$$

für die Matrix D_1 gilt

$$D_1 = \left. \frac{\partial \vec{G}}{\partial \vec{u}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \vec{u}} \right|_0 - \left. \frac{\partial \vec{y}}{\partial \vec{u}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \vec{u}} \right|_0 - 0 \quad (2.58)$$

und für die Matrix J

$$J = \left. \frac{\partial \vec{G}}{\partial \dot{y}} \right|_0 = \left. \frac{\partial \vec{g}}{\partial \dot{y}} \right|_0 - \left. \frac{\partial \dot{y}}{\partial \dot{y}} \right|_0 = 0 - I, \quad (2.59)$$

d.h. es handelt sich hierbei um die negative Einheitsmatrix.

Löst man Gleichung (2.55) unter Verwendung von Gleichung (2.59) nach $\Delta \dot{y}$ auf und setzt für $\Delta \dot{x}$ den Ausdruck aus Gleichung (2.46) ein, erhält man die Beziehung

$$\Delta \dot{y} = (C_1 - HE^{-1}A_1)\Delta \dot{x} + (D_1 - HE^{-1}B_1)\Delta \dot{u} \quad (2.60)$$

bzw.

$$\Delta \dot{y} = C\Delta \dot{x} + D\Delta \dot{u} \quad (2.61)$$

mit

$$C = C_1 - HE^{-1}A_1 \quad (2.62)$$

und

$$D = D_1 - HE^{-1}B_1 \quad (2.63)$$

Es sei darauf hingewiesen, dass sowohl die Zustandsgleichung (2.47) als auch die Ausgangsgleichung (2.61) nur unter der Voraussetzung hergeleitet werden können, dass E invertierbar ist. Dies ist jedoch immer der Fall, denn es gilt laut Gleichung (2.51):

$$E = \left. \frac{\partial \dot{f}}{\partial \dot{x}} \right|_0 - I. \quad (2.64)$$

Das bedeutet, die Matrix E besteht in jedem Fall aus einer Linearkombination, die die Einheitsmatrix I enthält. Die Matrix kann also nur dann singulär werden, wenn mindestens eines der Hauptdiagonalelemente der partiellen Ableitung von \dot{f} nach \dot{x} am Arbeitspunkt genau 1 ergibt und alle anderen Elemente der zugehörigen Zeile und Spalte verschwinden. Dies ist nur dann der Fall, wenn für mindestens eine der Zustandsableitungen gilt $\dot{x}_i = \dot{x}_i$. Dieser Fall hat jedoch keinen Sinn und tritt niemals ein, weshalb die Invertierbarkeit der Matrix E stets gewährleistet ist.

2.2.5 Beschreibung und Bestimmung der Längsbewegung

Anstatt sämtliche Zustände, Eingänge und Ausgänge der 6DOF Dynamik zu betrachten, werden im Falle der Längsbewegung nur folgende Größen berücksichtigt:

- die Zustandskomponenten V_{free} , α , q und Θ
- die Ausgänge α , q und n_z sowie
- die Eingänge η_s , ξ_s und δ_n .

Die zugehörigen Matrizen A_{long} , B_{long} , C_{long} und D_{long} können entweder durch Streichung der nicht benötigten Zeilen und Spalten aus den jeweiligen Matrizen des

Gesamtsystems ermittelt werden, oder es werden bereits bei der Linearisierung nur die tatsächlich benötigten partiellen Ableitungen berechnet. Dies ist möglich, da sich beispielsweise die Matrix A entsprechend Gleichung (2.48) aus dem Produkt aus $-E^{-1}$ und A_1 berechnet. Dabei unterscheidet sich die Matrix E nur in ihrer zweiten Spalte von der negativen Einheitsmatrix, denn neben α existieren bei der vorliegenden Aerodynamik und damit bei den nichtlinearen Bewegungs-Differentialgleichungen keine Abhängigkeiten von der Zustandsableitung. Dementsprechend unterscheidet sich die Matrix $-E^{-1}$ nur in ihrer zweiten Spalte von der positiven Einheitsmatrix. Spaltet man die Matrix $-E^{-1}$ auf, in eine Summe aus der Einheitsmatrix I und einer Matrix O , die außer in ihrer zweiten Spalte nur Nullen enthält, folgt aus (2.48)

$$A = -E^{-1}A_1 = (I + O)A_1 = A_1 + OA_1. \quad (2.65)$$

Hierbei bestehen die Elemente der Matrix OA_1 aufgrund der Struktur von O nur aus Produkten aus den Elementen der 2. Spalte von O und der 2. Zeile von A_1 . Somit können über den Summanden OA_1 auch nur Elemente aus der zweiten Zeile von A_1 in anderen Elementen von A auftreten. Gleichung (2.65) zeigt also, dass es zur Berechnung des Elementes $a_{i,j}$ der Matrix A ausreicht, das entsprechende Element $a_{1,i,j}$, das Element $a_{1,2,j}$ der Matrix A_1 sowie die zweite Spalte der Matrix E zu berechnen. (Die Betrachtung des j -ten Elements der 2. Spalte von A_1 reicht aus, weil das Element $a_{1,i,j}$ aus dem Skalarprodukt der i -ten Zeile von O und der j -ten Zeile von A_1 berechnet wird. Alle anderen Spalten haben also keinen Einfluss.) Gleiches gilt für die Matrix B , die Herleitung erfolgt analog wie zum Fall der Matrix A .

Im Falle der Ausgangsgleichung und der Matrix C zeigt Gleichung (2.62), dass zur Berechnung des Elementes $c_{i,j}$ der Matrix C zunächst das Element $c_{1,i,j}$ der Matrix C_1 benötigt wird. Des Weiteren enthält die Matrix den Summanden $-HE^{-1}A_1$. Hierbei gilt für den Anteil $-E^{-1}A_1$ das Gleiche wie für die Matrix A . Die Produktmatrix enthält nur Elemente mit Produkten aus der zweiten Spalte von E^{-1} und der zweiten Zeile von A_1 . Im Fall von Matrix C wird zusätzlich die Matrix H vormultipliziert. Bei dieser Matrix handelt es sich um die Ableitung der Ausgänge des nichtlinearen Modells nach der Zustandsableitung. Auch hier ist die einzige Abhängigkeit von der Zustandsableitung der Einfluss von α auf die Aerodynamik, weshalb sich hier nur die zweite Spalte von H von der Nullmatrix unterscheidet. Wie bei E reicht es also auch im Fall von H aus, lediglich die zweite Spalte der Matrix zu berechnen, und das Produkt $-HE^{-1}A_1$ kann nur Produkte aus den Elementen der zweiten Spalte von H , der zweiten Spalte von E sowie der zweiten Zeile von A_1 enthalten. Somit gilt auch im Fall der Matrix C , dass für die Berechnung des Elementes $c_{i,j}$ die Berechnung des entsprechenden Elementes der Matrix C_1 , die zweite Spalte von E und H sowie das j -te Elemente der zweiten Zeile von A_1 zu bestimmen. Das Gleiche gilt mit der analogen Begründung für die Matrix D .

Dementsprechend genügt zur Berechnung der Matrix A_{long} folgende Matrix $A_{1, long}$:

$$A_{1, long} = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} & \frac{\partial \dot{x}_1}{\partial x_5} & \frac{\partial \dot{x}_1}{\partial x_8} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} & \frac{\partial \dot{x}_2}{\partial x_5} & \frac{\partial \dot{x}_2}{\partial x_8} \\ \frac{\partial \dot{x}_5}{\partial x_1} & \frac{\partial \dot{x}_5}{\partial x_2} & \frac{\partial \dot{x}_5}{\partial x_5} & \frac{\partial \dot{x}_5}{\partial x_8} \\ \frac{\partial \dot{x}_8}{\partial x_1} & \frac{\partial \dot{x}_8}{\partial x_2} & \frac{\partial \dot{x}_8}{\partial x_5} & \frac{\partial \dot{x}_8}{\partial x_8} \end{bmatrix} \quad (2.66)$$

mit den Zustandsableitungen \dot{x}_i aus den entsprechenden Gleichungen des Systems in Gleichung (2.38) und den Zuständen x_i gemäß Gleichung (2.3). Wie verlangt, enthält die Matrix die zu den Zuständen 1, 2, 5 und 8 gehörenden Spalten sowie die zu den entsprechenden Zustandsableitungen gehörenden Zeilen. Die entsprechenden Spalten der zweiten Zeile von A_1 sind dabei wie verlangt enthalten, in diesem Fall sogar automatisch, weil der zweite Zustand, α , ein Zustand der Längsbewegung ist.

Für die Matrix B_{long} wird $B_{1, long}$ benötigt, es gilt hierbei:

$$B_{1, long} = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial u_1} & \frac{\partial \dot{x}_1}{\partial u_5} & \frac{\partial \dot{x}_1}{\partial u_8} \\ \frac{\partial \dot{x}_2}{\partial u_1} & \frac{\partial \dot{x}_2}{\partial u_2} & \frac{\partial \dot{x}_2}{\partial u_8} \\ \frac{\partial \dot{x}_5}{\partial u_1} & \frac{\partial \dot{x}_5}{\partial u_2} & \frac{\partial \dot{x}_5}{\partial u_8} \\ \frac{\partial \dot{x}_8}{\partial u_1} & \frac{\partial \dot{x}_8}{\partial u_2} & \frac{\partial \dot{x}_8}{\partial u_8} \end{bmatrix}, \quad (2.67)$$

mit den Eingängen u_i gemäß Gleichung (2.5).

Die Matrix $C_{1, long}$ zur Berechnung von C_{long} berechnet sich zu:

$$C_{1, long} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \frac{\partial y_1}{\partial x_5} & \frac{\partial y_1}{\partial x_8} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \frac{\partial y_2}{\partial x_5} & \frac{\partial y_2}{\partial x_8} \\ \frac{\partial y_5}{\partial x_1} & \frac{\partial y_5}{\partial x_2} & \frac{\partial y_5}{\partial x_5} & \frac{\partial y_5}{\partial x_8} \\ \frac{\partial y_8}{\partial x_1} & \frac{\partial y_8}{\partial x_2} & \frac{\partial y_8}{\partial x_5} & \frac{\partial y_8}{\partial x_8} \\ \frac{\partial y_{15}}{\partial x_1} & \frac{\partial y_{15}}{\partial x_2} & \frac{\partial y_{15}}{\partial x_5} & \frac{\partial y_{15}}{\partial x_8} \end{bmatrix}, \quad (2.68)$$

mit den Komponenten y_i des Ausgangsvektor \vec{y} des nichtlinearen Modells gemäß Gleichung (2.17).

Die Berechnung der Matrix D_{long} erfolgt ausgehend von der Matrix $D_{1, long}$, die sich aus den partiellen Ableitungen der Modellausgänge nach den Modelleingängen berechnet, es gilt:

$$D_{1, long} = \begin{bmatrix} \frac{\partial y_1}{\partial u_1} & \frac{\partial y_1}{\partial u_5} & \frac{\partial y_1}{\partial u_8} \\ \frac{\partial y_2}{\partial u_1} & \frac{\partial y_2}{\partial u_5} & \frac{\partial y_2}{\partial u_8} \\ \frac{\partial y_5}{\partial u_1} & \frac{\partial y_5}{\partial u_5} & \frac{\partial y_5}{\partial u_8} \\ \frac{\partial y_8}{\partial u_1} & \frac{\partial y_8}{\partial u_5} & \frac{\partial y_8}{\partial u_8} \\ \frac{\partial y_{15}}{\partial u_1} & \frac{\partial y_{15}}{\partial u_5} & \frac{\partial y_{15}}{\partial u_8} \end{bmatrix}. \quad (2.69)$$

2.2.6 Beschreibung und Bildung der Kurzzeitnäherung

In der Längsbewegung eines Flugzeuges lassen sich zwei Eigenbewegungen identifizieren. Die langsame Phygoide, die sich vor allem in einer Änderung der Geschwindigkeit V_{free} und des Lagewinkels Θ bemerkbar macht, und die schnellere Anstellwinkelschwingung, die sich vor allem in den Zuständen Anstellwinkel α und Nickrate q beobachten lässt. Für kurze Betrachtungszeiten dominiert die schnelle Anstellwinkelschwingung die Längsbewegung des Flugzeugs. Deshalb kann die Längsbewegung ausgehend von einem Startzustand für kurze Zeiten durch die Gleichungen für den Anstellwinkel und die Nickrate, die sogenannte

Kurzzeitnäherung oder Short-Period-Approximation (SPA) angenähert werden. Für den Entwurf eines Reglers, der die Anstellwinkelschwingung stabilisieren soll, ist diese Kurzzeitnäherung eine ausreichende Approximation des nichtlinearen Modells. Deshalb erfolgt für die vorliegende Arbeit eine Reduktion der Gleichungen der Längsbewegung des Jettrainers auf die zugehörige Kurzzeitnäherung mit den Zuständen α und q , dem Eingang η_s sowie den Ausgängen α , q und n_z . Hierzu werden die Zeilen und Spalten der nicht betrachteten Zustände in den Matrizen der Längsbewegung weggelassen und man erhält aus der Zustandsraumdarstellung des Gesamtsystems die Zustandsraumdarstellung der SPA

$$\begin{aligned}\dot{\hat{x}} &= A_{SPA}\hat{x} + B_{SPA}u \\ \hat{y} &= C_{SPA}\hat{x} + D_{SPA}u\end{aligned}\quad (2.70)$$

mit der Dynamik-Matrix A_{SPA} mit

$$A_{SPA} = \begin{bmatrix} a_{2,2} & a_{2,5} \\ a_{5,2} & a_{5,5} \end{bmatrix}, \quad (2.71)$$

der Eingangsmatrix B_{SPA} mit

$$B_{SPA} = \begin{bmatrix} b_{2,1} \\ b_{5,1} \end{bmatrix}, \quad (2.72)$$

der Ausgangsmatrix C_{SPA} mit

$$C_{SPA} = \begin{bmatrix} c_{2,2} & c_{2,5} \\ c_{5,2} & c_{5,5} \\ c_{15,2} & c_{15,5} \end{bmatrix} \quad (2.73)$$

sowie der Durchgangsmatrix D_{SPA} mit

$$D_{SPA} = \begin{bmatrix} d_{2,1} \\ d_{5,1} \\ d_{15,1} \end{bmatrix}. \quad (2.74)$$

Da die Kurzzeitnäherung in den Gleichungen der Längsbewegung enthalten ist, kann sie auch direkt aus der Längsbewegung gebildet werden, d.h. zur Bildung der Kurzzeitnäherung ist es nicht notwendig, das komplette System der linearen Bewegungsgleichungen zu bestimmen.

2.2.7 Linearisierung: Numerische Differentiation

Die Bildung der partiellen Ableitungen der Zustandsgleichung (2.47) und der Ausgangsgleichung (2.61) erfolgt durch numerische Differentiation.

Selbstverständlich lassen sich die partiellen Ableitungen der Ausgänge und Bewegungsdifferentialgleichungen des nichtlinearen Modells auch symbolisch berechnen. Es treten dann jedoch die partiellen Ableitungen der aerodynamischen Kräfte und Momente sowie der schubbedingten Kräfte und Momente nach dem Zustandsvektor \hat{x} , dem Steuervektor \hat{u} sowie der Zustandsableitung $\dot{\hat{x}}$ auf, deren numerische Werte am Arbeitspunkt benötigt werden. Diese numerische Werte können nicht anders als durch numerische Differentiation gebildet werden, da sowohl das Aerodynamikmodell als auch das Schubmodell nicht auf Beiwerten, also den gesuchten partiellen Ableitungen basieren, sondern auf Tabellendaten.

Um eine symbolische Berechnung der Ableitungen zu umgehen, erfolgt daher die Bildung der Ableitungen auf rein numerische Art und Weise.

Hierzu werden die partiellen Ableitungen, d.h. die Differentialquotienten, durch finite Differenzenquotienten angenähert. Diese Annäherung erfolgt, wo immer dies möglich ist, durch die Bildung von zentralen Differenzen. Soll beispielsweise am Arbeitspunkt x_0 die partielle Ableitung der Funktion $f(x)$ gebildet werden, wird entsprechend Abbildung 2.5 der Eingang x an der Stelle x_0 um die Schrittweite h nach vorwärts und rückwärts gestört, um mit den Funktionswerten $f(x_0 + h)$ und $f(x_0 - h)$ an den Stellen $x_0 + h$ und $x_0 - h$ die Tangentensteigung der Funktion $f(x)$ an der Stelle x_0 durch die Sekantensteigung anzunähern.

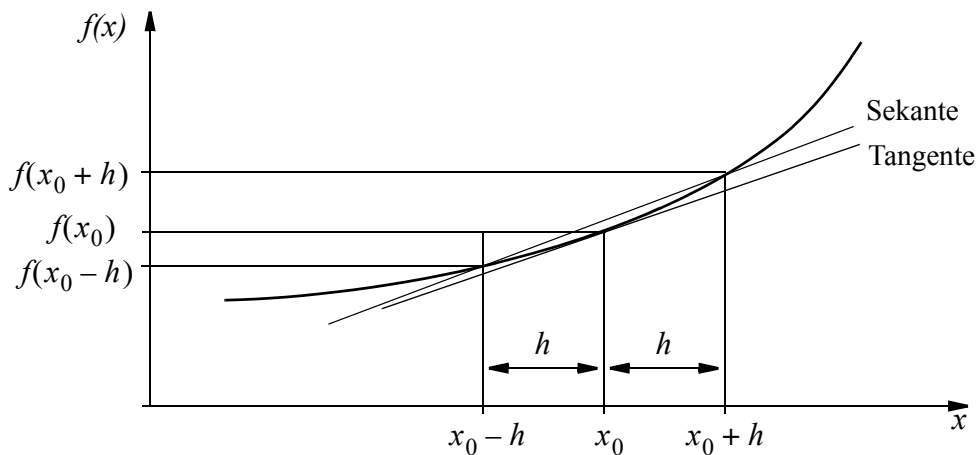


Abb. 2.5: Annäherung der Ableitung durch eine zentrale Differenz

An Arbeitspunkten, die am Rand der Envelope liegen, ist entweder die Störung nach vorwärts oder die Störung nach rückwärts nicht möglich. In diesem Fall erfolgt die Näherung der partiellen Ableitung durch eine Vorwärts- bzw. Rückwärtsdifferenz.

Für die Näherungsformeln der zentralen Differenz sowie der Vorwärts- und Rückwärtsdifferenz 1. Ordnung gelten folgende Formeln:

Zentrale Differenz 1. Ordnung:

$$D_{Z1} = \frac{f(x_0 + h) - f(x_0 - h)}{2h} \quad (2.75)$$

Vorwärtsdifferenz 1. Ordnung:

$$D_{V1} = \frac{f(x_0 + h) - f(x_0)}{h} \quad (2.76)$$

Rückwärtsdifferenz 1. Ordnung:

$$D_{R1} = \frac{f(x_0) - f(x_0 - h)}{h} \quad (2.77)$$

Eine höhere Genauigkeit bieten Differenzenformeln höherer Ordnungen. Anstatt nur mit der Schrittweite h vorwärts und/oder rückwärts zu stören, wird in diesem Fall zusätzlich um Vielfache der Schrittweite gestört, wobei die entsprechenden Funktionswerte in die Differenzenformeln eingehen. Für die zentrale Differenz sowie die Vorwärts- und Rückwärtsdifferenz 2. Ordnung gelten die Formeln:

Vorwärtsdifferenz 2. Ordnung:

$$D_{V2} = \frac{3f(x_0) + 4f(x_0 + h) - f(x_0 + 2h)}{2h} \quad (2.78)$$

Rückwärtsdifferenz 2. Ordnung:

$$D_{R2} = \frac{3f(x_0) - 4f(x_0 - h) + f(x_0 - 2h)}{2h} \quad (2.79)$$

Im vorliegenden Fall erfolgt die Durchführung der finiten Differenzenbildung entsprechend [2] iterativ. Das heißt, die Differenzenbildung startet mit einem Startwert h_{start} , der sukzessive verkleinert wird, bis die relative Abweichung zweier aufeinander folgender Berechnungen eine vorgegebene Toleranz unterschreitet.

Des Weiteren wird an Arbeitspunkten, an denen die Berechnung mittels zentraler Differenz möglich war, nach Erreichen der Toleranz zur Kontrolle eine Vorwärts- und eine Rückwärtsdifferenz gebildet und die jeweiligen Werte werden verglichen.

2.2.8 Linearisierung: Implementierung in Matlab

Die Matlab-Implementierung der Linearisierung erfolgt über das Skript `calculation_lin10`. Analog zum Fall der Trimmung handelt es sich hierbei um ein Konfigurations- und Steuerungsskript, in dem verschiedene Einstellungen getroffen werden und zur eigentlichen Linearisierung verschiedene andere Subroutinen aufgerufen werden.

Es erfolgt zunächst die Spezifizierung des Linearisierungsmodus, d.h. ob das komplette nichtlineare System bearbeitet werden soll, oder ob lediglich die Gleichungen, die zur Längsbewegung gehören, linearisiert werden sollen. Da sich die vorliegende Arbeit mit dem Entwurf eines Reglers für die Längsbewegung beschäftigt, wurde kein Modus für

ausschließliche Betrachtung der Seitenbewegung implementiert. Die entsprechenden Linearisierungen erfolgen jedoch im Rahmen einer Berechnung des Gesamtsystems. Außerdem erfolgt über das Skript `calculation_lin10` die Auswahl eines Workspaces mit Trimmdaten, die als Arbeitspunkte für die Linearisierung verwendet werden sollen. Nach dem Laden der Trimmdaten werden in einer Schleife die Arbeitspunkte in den Zeilen der Variablen `trimarray` abgearbeitet (siehe hierzu auch Abschnitt 2.2.3).

In der Schleife wird für jeden Trimpunkt die eigentliche Linearisierungsfunktion `linfun10` aufgerufen. Dieser Funktion werden die Werte des Zustandsvektors, des Steuervektors sowie der Zustandsableitung am Arbeitspunkt, der Linearisierungsmodus und diverse Toleranzwerte übergeben, deren Bedeutung in der folgenden Funktionsbeschreibung erläutert werden.

In `linfun10` werden die Elemente der Matrizen A_1 , E , B_1 , C_1 , H und D_1 bestimmt. Hierzu werden die zum jeweiligen Element der jeweiligen Matrix gehörenden partiellen Ableitungen der entsprechenden Modellausgänge nach den entsprechenden Modelleingängen durch finite Differenzenbildung angenähert.

Es erfolgt zunächst, sofern möglich, die Bildung je einer Rückwärtsdifferenz erster Ordnung und zweiter Ordnung. Ausgehend von Startschrittweiten, die sich an den Schrittweiten des AeroDataModuls zur Bildung der aerodynamischen Derivativa orientieren, wird die Schrittweite halbiert, bis die relative Abweichung zwischen Rückwärtsdifferenz erster und zweiter Ordnung kleiner als die Toleranz `TolGradUpDo` ist. Zur Bildung der relativen Abweichung wird dabei der Betrag der Abweichung zwischen zwei Werten auf deren Mittelwert bezogen.

Im Anschluss erfolgt analog zur Bildung der Rückwärtsdifferenz die Berechnung einer Vorwärtsdifferenz.

An Trimpunkten, an denen sowohl die Berechnung einer Rückwärts- als auch einer Vorwärtsdifferenz möglich ist, erfolgt ein Vergleich zwischen beiden Ergebnissen. Die relative Abweichung ist im Nominalfall kleiner als der Wert der Toleranz `ConvTol`.

Ebenfalls an Trimpunkten, an denen Vorwärts- und Rückwärtsdifferenzenbildung möglich war, erfolgt abschließend die Bildung einer zentralen Differenz, wobei hier die Schrittweiten so lange halbiert werden, bis die relative Abweichung aufeinander folgender Ergebnisse kleiner als der Wert der Toleranz `CenTol` ist. Es erfolgt eine Kontrolle in Form einer abschließenden Halbierung der Schrittweite, wobei die Abweichung zwischen letzter Berechnung und Kontrollberechnung im Nominalfall weniger als der Wert der Toleranz `FinTol` beträgt.

Wird im Laufe einer Berechnung die Schrittweite so klein, dass die Auflösungsgrenze erreicht wird, das heißt dass die Störung h so klein ist, dass es zu keiner Änderung der Modellausgänge mehr kommt, wird je nach Art der momentan durchgeführten Berechnung, das heißt Vorwärts-, Rückwärts- oder Zentralsdifferenzenbildung und je nachdem, ob es sich um eine normale Berechnung oder eine abschließende Kontrollrechnung handelt, eine Warnung oder eine Fehlermeldung in das Log-File `LinLog` geschrieben. Die Auswahl, welcher der nach den verschiedenen Berechnungsarten bestimmten Werte als Näherung der partiellen Ableitung verwendet wird, richtet sich danach, welche Werte erfolgreich berechnet werden konnten. Im Nominalfall wird so der letzte Wert der zentralen Differenz vor der

abschließenden Kontrollrechnung verwendet. Wenn dieser Wert nicht existiert, weil keine zentrale Differenzenbildung möglich war oder weil im Laufe der Berechnung die Auflösungsgrenze erreicht wurde, wird je nach Vorhandensein der Endwert der Vorwärts- oder der Rückwärtsdifferenz verwendet. Der Endwert der Vorwärts- bzw. Rückwärtsdifferenz ist dabei der Wert der Differenz zweiter Ordnung beim Erreichen der Toleranz $TolGradUpDo$ für die relative Abweichung zwischen Differenz erster und zweiter Ordnung.

Nachdem alle Elemente der beschriebenen Matrizen berechnet wurden, werden entsprechend den Gleichungen (2.48) und (2.49) sowie (2.62) und (2.63) die Matrizen A , B , C und D berechnet.

Zur genaueren Beschreibung der Abläufe im Skript `linfun10` sei auf das Nassi-Shneiderman Diagramm in Anhang B.2 verwiesen.

Die Funktion `linfun10` liefert neben den Matrizen A , B , C und D auch zugehörige Informationsarrays an `calculation_lin10` zurück, die für jedes Matrizenelement Informationen über die erreichten Toleranzen d.h. die Abweichungen am Berechnungsende, die finalen Schrittweiten und eine Statusvariable enthalten. Die Statusvariable gibt Auskunft darüber, welche der diversen Berechnungen erfolgreich abgeschlossen wurden und welche nicht.

In `calculation_lin10` werden die Ausgänge von `linfun10` in die Variable `linarray` geschrieben. Der Aufbau dieses Cell-Arrays ist ähnlich dem der Trimmergebnisvariablen `trimarray`. Sie hat 56 Spalten und für jeden Arbeitspunkt eine Zeile. Die Spalten 1 bis 22 enthalten dabei dieselben Werte wie `trimarray`. Die Besetzung der weiteren Spalten erfolgt gemäß Tabelle 2.14, wobei ähnlich wie im Fall von `trimarray` die Spalten abwechselnd eine Beschreibung des Inhalts der folgenden Zeile enthalten (mit Ausnahme der Spalten 39 und 48, diese enthalten nur Beschreibungen).

Spalte Nr.	Inhalt	Typ
23/24	Matrix A	Matrix (12 x 12)
25/26	Matrix B	Matrix (12 x 8)
27/28	Matrix C	Matrix (27 x 12)
29/30	Matrix D	Matrix (27 x 8)
31/32	Matrix A_{long}	Matrix (4 x 4)
33/34	Matrix B_{long}	Matrix (4 x 3)
35/36	Matrix C_{long}	Matrix (5 x 4)
37/38	Matrix D_{long}	Matrix (5 x 2)
39	Information über Zustände, Eingänge und Ausgänge der Längsbewegung	Character(1x79)
40/41	Matrix A_{lat}	Matrix (4 x 4)
42/43	Matrix B_{lat}	Matrix (4 x 2)
44/45	Matrix C_{lat}	Matrix (5 x 4)
46/47	Matrix $_{lat}$	Matrix (5 x 2)
48	Information über Zustände, Eingänge und Ausgänge der Seitenbewegung	Character(1x71)
49/50	Info Matrix A_1	Cell-Array (1x5)

Spalte Nr.	Inhalt	Typ
51/52	Info Matrix B_1	Cell-Array (1x5)
53/54	Info Matrix C_1	Cell-Array (1x5)
55/56	Info Matrix D_1	Cell-Array (1x5)

Tabelle 2.14: Zusammensetzung der Spalten der Ausgabevariablen `linarray`

Neben der Variablen `linarray` enthalten die Workspaces mit Linearisierungsdaten die in Tabelle 2.15 zusammengefassten Variablen.

Name	Beschreibung
<i>A1Info</i>	Informationen zur Matrix A_1
<i>alpha</i>	Anstellwinkel der linearisierten Trimmdaten
<i>alphalist</i>	Anstellwinkelbereich der zu Grunde liegenden Trimmdaten
<i>hlist</i>	Höhenbereich der linearisierten Trimmdaten ^{a)}
<i>infoarray</i>	Informationen zur numerischen Linearisierung
<i>machlist</i>	Machzahlbereich der linearisierten Trimmdaten
<i>nzlist</i>	Lastvielfache der linearisierten Trimmdaten ^{b)}
<i>qdynlist</i>	Staudruckbereich der linearisierten Trimmdaten ^{c)}
<i>trimws</i>	Name des Workspaces der zu Grunde liegenden Trimmdaten

a) nur relevant bei Spezifizierung der Trimmdaten über Höhe

b) nur relevant bei Spezifizierung der Trimmdaten über Lastvielfache

c) nur relevant bei Spezifizierung der Trimmdaten über Staudruck

Tabelle 2.15: Variablen der Linearisierungsdaten-Workspaces

Nachdem alle Arbeitspunkte bearbeitet wurden, wird auf Basis des Namens des geladenen Trimm-Workspaces ein Name für die Speicherung des Linearisierungs-Workspaces generiert und die Linearisierungsdaten können gespeichert werden.

Da im Falle einer feinen Diskretisierung der Envelope aufgrund sehr vieler Arbeitspunkte sehr große Workspaces entstehen, besteht die Möglichkeit, über das Skript `calculation_lin10auto` eine Linearisierung mit Gliederung der Ergebnis-Workspaces nach dem Anstellwinkel vorzunehmen. Hierbei erfolgt nach der Linearisierung aller Arbeitspunkte mit gleichem Anstellwinkel eine automatische Namensgenerierung und Abspeicherung der Daten. Zudem werden die so generierten Workspaces in einem Zip-Archiv zusammengefasst.

2.2.9 Aktuator- und Sensormodell

Neben der Dynamik eines Festkörpers mit sechs Freiheitsgraden beinhaltet die Modellierung des Jettrainers Mako auch die Dynamiken der Sensoren zur Bestimmung von Messgrößen sowie der Aktuatoren zum Einstellen der Steuerflächen.

Sowohl die Aktuatorik als auch die Sensorik werden durch lineare Übertragungsfunktionen modelliert.

Zur Regelung der Längsdynamik des Flugzeugs steht im Rahmen dieser Arbeit nur das symmetrisch ausgeschlagene Taileron zur Verfügung. Dessen Aktuatordynamik wird entsprechend der technischen Anleitung [13] modelliert. Die entsprechende Übertragungsfunktion lautet

$$:F_{akt} = \frac{1}{3,4 \cdot 10^{-7}s^3 + 0,1417 \cdot 10^{-3}s^2 + 0,02s + 1} \quad (2.80)$$

Neben der Modellierung der Aktuatordynamik beinhaltet diese Übertragungsfunktion einen Notch- bzw. Kerbfilter. Dieser verhindert durch eine Übertragungsnullstelle bei 10 Hz, dass durch den Flugregler und die Aktuatorik Strukturschwingungen angeregt werden. Wie Abbildung 2.6 zeigt, ist die Kerbe im Bode-Plot der Übertragungsfunktion gut zu erkennen.

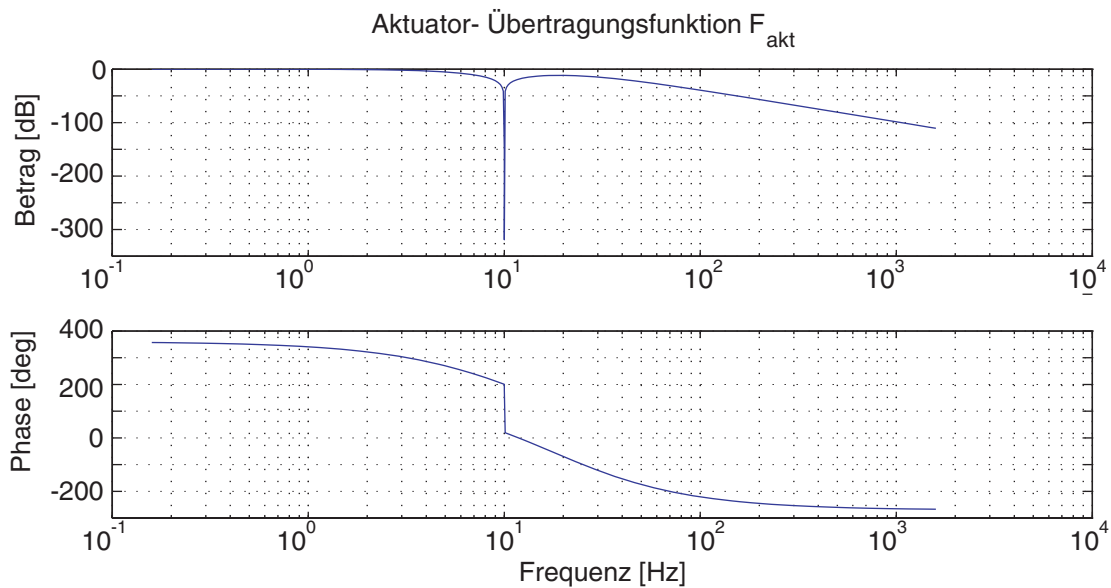


Abb. 2.6: Bode-Plot der Aktuator-Übertragungsfunktion mit Kerbfilter

Auch die Dynamik der Sensoren liegt bereits in Form von linearen Übertragungsfunktionen vor. Für die Messung des Anstellwinkels lautet die Übertragungsfunktion des Sensors gemäß [12]:

$$F_{s, \alpha} = \frac{1}{0,015s + 1} \quad (2.81)$$

Die Übertragungsfunktion des Sensors zur Messung der Nickrate q lautet

$$F_{s, q} = \frac{1}{0,01s + 1} \quad (2.82)$$

Das Entwurfsmodell entsteht nun durch die Reihenschaltung der Übertragungsfunktionen des Aktuators, der Kurzzeitnäherung und der Sensoren.

2.2.10 Reihenschaltung zum Entwurfsmodell

Das lineare Modell G_E , das als Basis für den Reglerentwurf dient, entsteht durch die Reihenschaltung des Aktuatormodells F_a , der Kurzzeitnäherung G_{SPA} und den beiden Sensormodellen $F_{s,\alpha}$ und $F_{s,q}$. Das Modell hat einen Eingang, den symmetrischen Ausschlag des Taileron η_s , sowie zwei Ausgänge, den Ausgang des Anstellwinkelsensors α_s und den Ausgang des Nickratensensors q_s . Das Modell hat neun Zustände, hiervon stammen zwei aus der Kurzzeitnäherung, fünf aus dem Aktuatormodell und jeweils einer aus den beiden Sensoren. Das zugehörige Blockschaltbild ist in Abbildung 2.7 dargestellt.

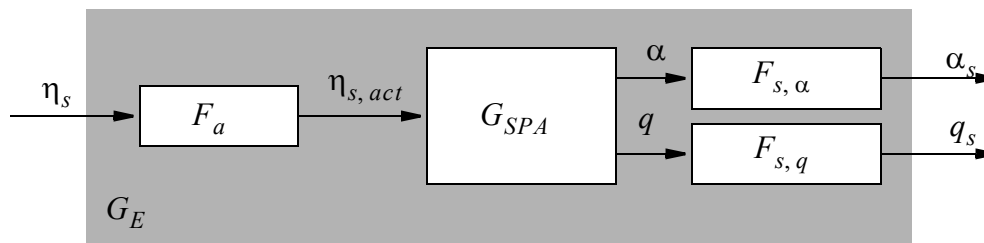


Abb. 2.7: Blockschaltbild des Entwurfsmodells

2.3 Analyse des Entwurfsmodells

Die Analyse des Entwurfsmodells erfolgt hinsichtlich seiner Stabilität in verschiedenen Flugzuständen. Hierzu wird zum einen der Verlauf des Beiwertes $C_{m\alpha}$ des Modells für den Flug auf einem quasi-stationären Abfangbogen (Nicktrimmung) betrachtet. Des Weiteren wird die Lage der Pole der linearisierten Modelle, sowohl des Entwurfsmodells, d.h. der Kurzzeitnäherung, als auch der kompletten Längsbewegung untersucht.

2.3.1 Verlauf des Momentenbeiwerts

Der Momentenbeiwert beschreibt die Ableitung des Momentes M um die Querachse des Flugzeugs nach dem Anstellwinkel α . Er gibt Aufschluss über die statische Stabilität des Flugzeugs, denn je nach dem ob er positiv oder negativ ist, bewirkt eine Störung, die den Anstellwinkel vergrößert, ein auf- oder abnickendes Moment. Für Werte $C_{m\alpha} < 0$ bewirkt die Anstellwinkelvergrößerung ein abnickendes Moment, das wiederum der Störung entgegenwirkt. Für diesen Fall liegt statische Stabilität vor. Für Werte $C_{m\alpha} > 0$ dagegen bewirkt die Störung ein aufnickendes Moment, welches den Anstellwinkel noch weiter vergrößert. In diesem Fall liegt Instabilität vor. Für einen Wert von $C_{m\alpha} = 0$ spricht man von Grenzstabilität.

Abbildung 2.8 zeigt den Verlauf von $C_{m\alpha}$ über dem Anstellwinkel für den Flug auf einem quasi-stationären Abfangbogen in einer Flughöhe von $h = 1000$ m und verschiedenen Machzahlen. Die Abbildung zeigt, dass das Flugzeug für negative Anstellwinkel für alle betrachteten Machzahlen instabil ist. Ab Anstellwinkeln von $\alpha > 26^\circ$ ist das Flugzeug für alle Machzahlen stabil. Zwischen 0° und 26° liegt mit Ausnahme von $\alpha = 15^\circ$ Stabilität

vor, die jedoch unterschiedlich stark ausgeprägt ist. Bei 15° ist das Flugzeug klar instabil. Die Abbildung zeigt außerdem, dass das Verhalten über dem Anstellwinkel stark nichtlinear ist und dass die Machzahl einen verstärkenden Einfluss auf das jeweilige Verhalten bei einem Anstellwinkel hat.

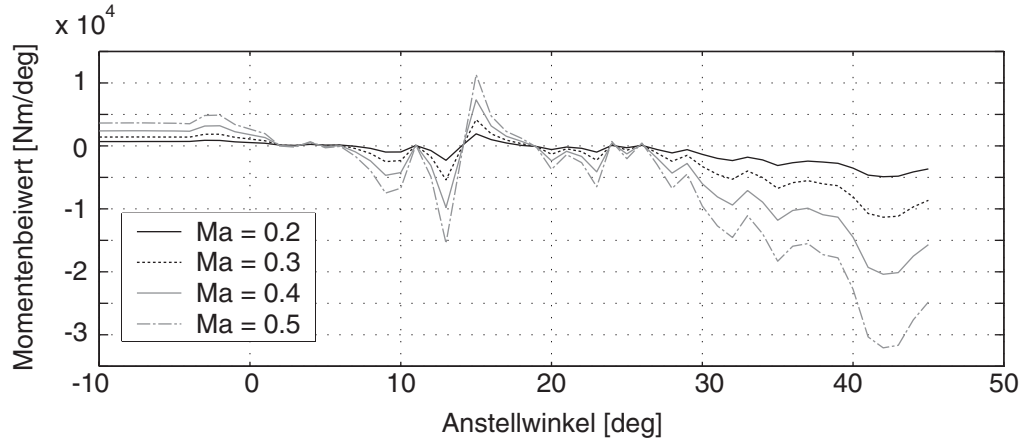


Abb. 2.8: Momentenbeiwert über Anstellwinkel für verschiedene Machzahlen, $h = 1000\text{m}$

Die Verläufe der Kurven für andere Flughöhen unterscheiden sich qualitativ nicht von denen in Abbildung 2.8. Dies zeigt auch Abbildung 2.9, dort ist der Verlauf des Momentenbeiwerts über dem Anstellwinkel für die Machzahl 0,3 und verschiedene Flughöhen aufgetragen.

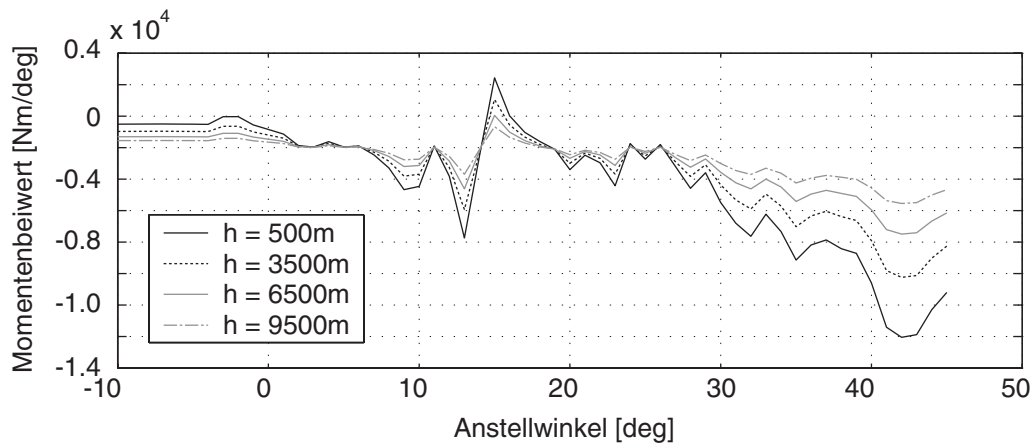


Abb. 2.9: Momentenbeiwert über Anstellwinkel, verschiedene Höhen, $Ma = 0,3$

Hier zeigt sich, dass die jeweiligen Werte des Momentenbeiwerts an einem festen Anstellwinkel für steigende Höhen und fallende Machzahlen betragsmäßig abnehmen. Dies lässt sich durch den starken Einfluss des Staudrucks auf die Aerodynamik des Flugzeugs erklären. Der Staudruck steigt mit sinkender Höhe und steigender Machzahl und mit ihm der Betrag der aerodynamischen Kräfte und Momente. Dies wird deutlich in Abbildung 2.10 und Abbildung 2.11. In Abbildung 2.10 ist wiederum der Momentenbeiwert über dem

Anstellwinkel aufgetragen. Dieses Mal jedoch für eine Machzahl $Ma = 0,3$ und verschiedene Staudrücke.

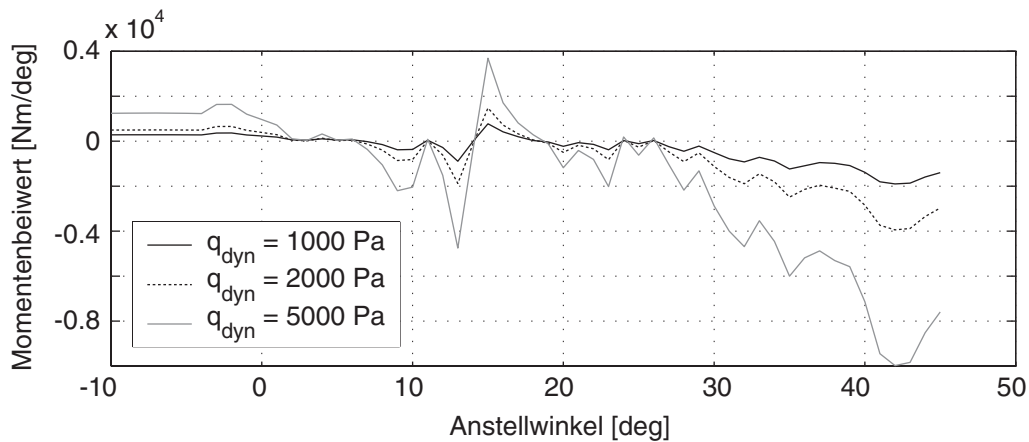


Abb. 2.10: Momentenbeiwert über Anstellwinkel für verschiedene Staudrücke, $Ma = 0,3$

Abbildung 2.11 zeigt den Momentenbeiwert für verschiedene Anstellwinkel bei einem konstanten Staudruck von $q_{dyn} = 5000$ Pa und verschiedenen Machzahlen.

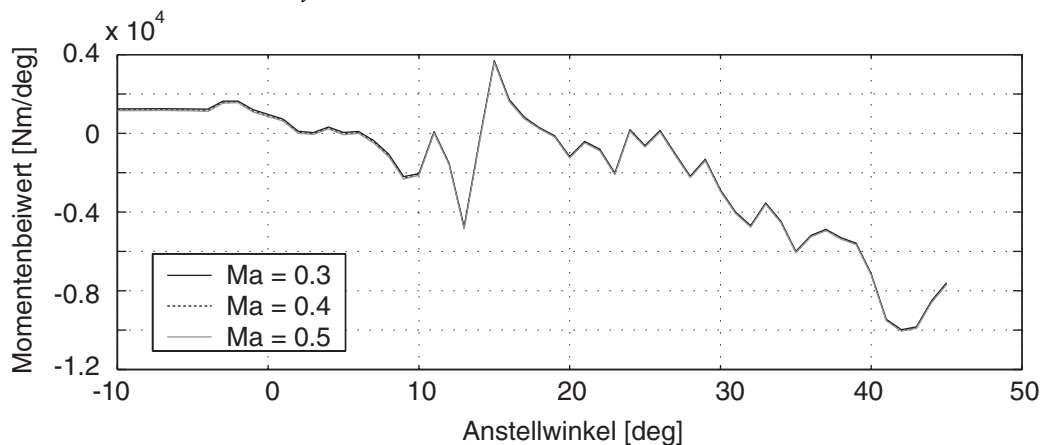


Abb. 2.11: Momentenbeiwert über Anstellwinkel für verschiedene Machzahlen, $q_{dyn} = 5000$ Pa

Diese Abbildung zeigt, dass es für den betrachteten Machzahlbereich praktisch keine reine Abhängigkeit von der Machzahl gibt, denn wenn bei einer Änderung der Machzahl die Höhe so angepasst wird, dass der Staudruck konstant bleibt, ändert sich der Wert des Momentenbeiwerts nicht. Bei konstanter Höhe ändert sich bei veränderter Machzahl dagegen die Fluggeschwindigkeit und mit ihr der Staudruck, was wiederum die in Abbildung 2.8 beobachtbare Änderung des Beiwertes bedingt.

2.3.2 Stabilitätsanalyse anhand der Pole der Regelstrecke

Alternativ zur Betrachtung des Verlaufs des Momentenbeiwerts $C_{m\alpha}$ können zur Analyse des dynamischen Verhaltens des Flugzeugs bezüglich seiner Stabilität auch die Lagen der Pole der Regelstrecke analysiert werden. Hierbei wird als Maß für die Stabilität bzw.

Instabilität eines Pols dessen Realteil betrachtet. In Anlehnung an die Darstellung komplexer Zahlen in der Form $c = \sigma + j\omega$ wird der Wert des Realteils eines Pols mit σ bezeichnet.

Hierbei werden die Pole der Aktuator- und Sensordynamik vernachlässigt, da sie stabil sind und sich für verschiedene Flugzustände nicht ändern. Betrachtet wird zum einen die Dynamik der vollständigen Längsbewegung und zum anderen die Dynamik der Kurzzeitznäherung.

Die Längsbewegung hat aufgrund ihrer vier Zustände vier Pole. Zwei der Pole gehören zur Anstellwinkelschwingung (AS), die beiden anderen zur Phygoide. Die Zuordnung kann dabei auf verschiedene Arten erfolgen. Eine Möglichkeit ist die Modalanalyse der Systemmatrix A_{long} , denn die Eigenbewegung der Phygoide lässt sich primär in den Zuständen V_{free} und Θ beobachten, wohingegen die Anstellwinkelschwingung von den Zuständen α und q dominiert wird. Durch die Bildung der Eigenvektoren von A_{long} und die Untersuchung, welche der Zustände in welchen Eigenvektoren dominieren, lassen sich die Eigenvektoren und damit die zugehörigen Eigenwerte den beiden Eigenbewegungen des Systems zuordnen. Eine andere, schnellere Methode, die ohne eine Modalanalyse auskommt, besteht in der Untersuchung der Lage der Pole. Da die Anstellwinkelschwingung eine signifikant höhere Frequenz als die Phygoide hat, kann die Polzuordnung anhand der Eigenfrequenz der Pole, also ihres Abstands zum Ursprung erfolgen. Für den Fall, dass es sich bei beiden Polpaaren um konjugiert komplexe Polpaare handelt, wird das schnellere, also höherfrequente, weiter vom Ursprung entfernte Paar der Anstellwinkelschwingung zugeordnet. Für den Fall, dass nur ein komplexes Polpaar existiert, geht man davon aus, dass es sich bei diesem schwingungsfähigen Polpaar um dasjenige der Phygoide handelt. Die Pole der Anstellwinkelschwingung sind nun beide reell, das heißt man kann genaugenommen nicht mehr von einer Anstellwinkelschwingung sprechen. Im Normalfall hat der eine der beiden Pole einen negativen und der andere einen positiven Realteil. In diesem Fall ist die Anstellwinkelbewegung also instabil. Im Fall des Jettrainers Mako tritt bei manchen Flugzuständen auch der Spezialfall ein, dass alle vier Pole auf der reellen Achse liegen. Obwohl im Fall beider Eigenbewegungen nun keine Schwingungen mehr vorliegen, geht man weiter davon aus, dass die Anstellwinkelbewegung die schnellere der beiden Eigenbewegungen ist. Deshalb werden die Pole mit dem kleinsten und dem größten Wert der Anstellwinkelbewegung zugeordnet und die beiden verbleibenden Pole der Phygoide.

Abbildung 2.12 zeigt exemplarisch für eine Machzahl von $Ma = 0,31$ und einer Flughöhe von $h = 1000\text{m}$ den Verlauf der Realteile der Pole der Phygoide und der Anstellwinkelschwingung über dem Anstellwinkel. Zum Vergleich enthält die Abbildung auch die entsprechenden Werte der Längsbewegung aus [14], die durch die EADS-Linearisierungsroutine berechnet wurde. Es zeigt sich eine gute Übereinstimmung der Ergebnisse beider Linearisierungsroutinen. Des Weiteren ist erkennbar, dass die Anstellwinkelschwingung bei Anstellwinkeln unterhalb von $\alpha < 7^\circ$ einen Pol mit positivem Realteil hat und somit im betreffenden Bereich instabil ist. Gleiches gilt für Anstellwinkel im Bereich um 11° , 24° und 26° sowie zwischen 14° und 19° . In den übrigen Bereichen fallen beide Realteile der Anstellwinkelschwingungspole zusammen und sind negativ. Es handelt sich dort also um eine stabile Schwingung.

Im Falle der Phygoide handelt es sich fast im gesamten betrachteten Anstellwinkelbereich um eine, wenn auch schwach gedämpfte, stabile Schwingung, da die Realteile beider Pole

zusammenfallen und negativ sind. Lediglich in dem Anstellwinkelschwingungsbereich um 19° wird die Phygoide im Falle der Längsbewegung, die mit der IFR-Linearisierungsroutine berechnet wurde, instabil. Da die Dynamik der Phygoide für den Reglerentwurf nicht verwendet wird, wird die Abweichung zwischen den beiden Linearisierungsroutinen jedoch nicht weiter untersucht.

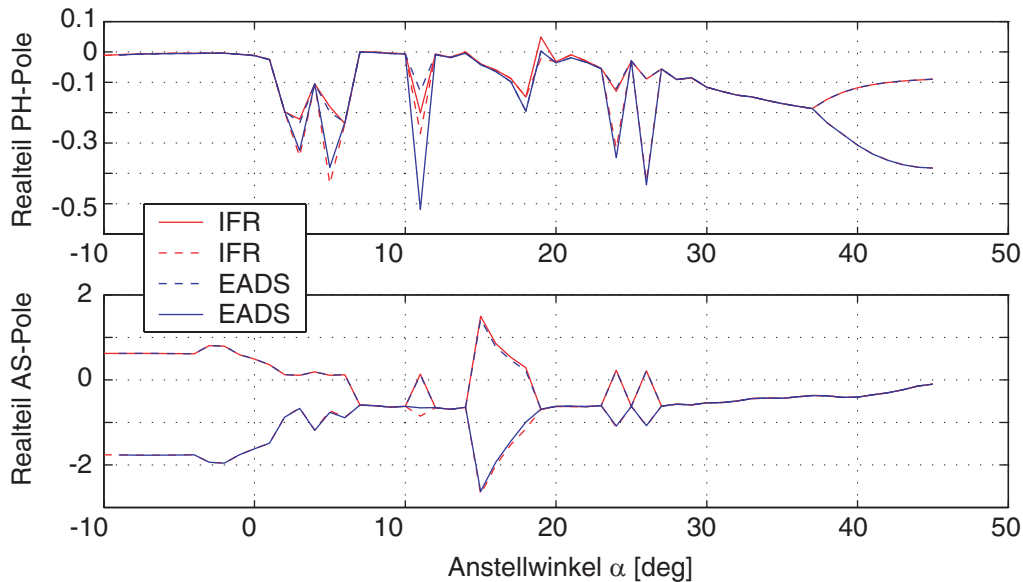


Abb. 2.12: Realteile von Phygoid- und AS-Polen. Vergleich IFR-EADS-Daten

Abbildung 2.13 zeigt für die gleiche Machzahl und Flughöhe wie im Fall von Abbildung 2.12 eine gute Übereinstimmung zwischen den Realteilen der Anstellwinkelschwingungspole aus der kompletten Längsbewegung und den Realteilen der beiden Pole der Kurzzeitnäherung. Hier zeigt sich, dass die Kurzzeitnäherung als Näherung für die Anstellwinkelschwingungsbewegung verwendet werden kann.

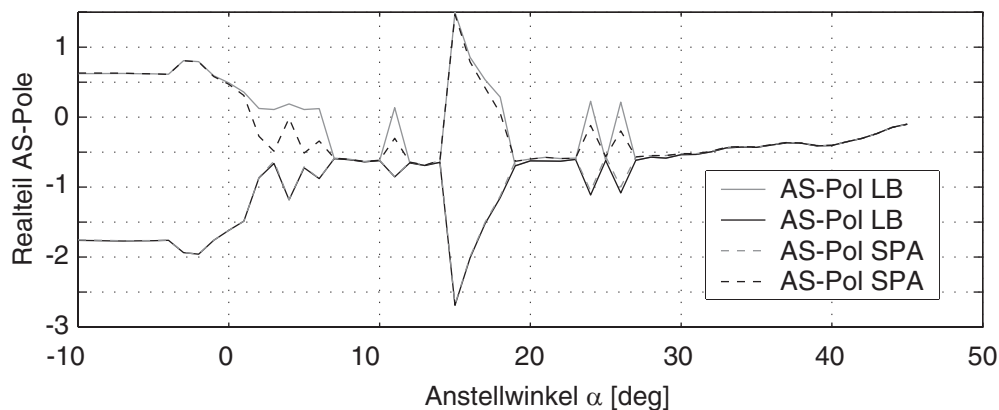


Abb. 2.13: Vergleich AS-Pole aus Längsbewegung und Kurzzeitnäherung

Abschließend erfolgt analog zur Untersuchung des Momentenbeiwerts eine Untersuchung vom Verlauf des Realteils der Pole von der Anstellwinkelschwingung der Kurzzeitnäherung über dem Anstellwinkel für verschiedene Machzahlen bei konstantem Staudruck sowie für verschiedene Staudrücke bei konstanter Machzahl.

Abbildung 2.14 zeigt den Verlauf der Pole der Anstellwinkelschwingung für die Machzahl $Ma = 0,3$ und verschiedene Staudrücke.

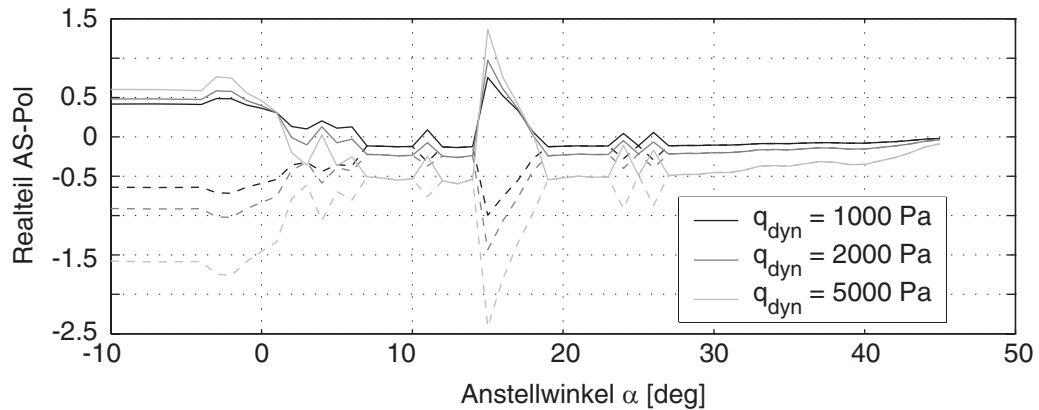


Abb. 2.14: AS-Pole über Anstellwinkel für verschiedene Staudrücke, $Ma = 0,3$

Ähnlich wie beim Momentenbeiwert in Abbildung 2.10 zeigt sich auch hier ein gut sichtbarer Einfluss des Staudrucks. Wie im Fall des Momentenbeiwerts kommt es auch hier im Bereich um $\alpha = 15^\circ$ durch einen ansteigenden positiven Realteil zu einer Verstärkung der Instabilität mit steigendem Staudruck, auch die anderen Anstellwinkelbereiche verhalten sich, wie zu erwarten war, analog zum Momentenbeiwert.

Bei der Abhängigkeit der Pollage vom Staudruck handelt es sich im betrachteten Machzahlbereich um einen näherungsweise quadratischen Zusammenhang. Abbildung 2.15 zeigt den Verlauf der Realteile von beiden Polen der Anstellwinkelschwingung über dem Staudruck für verschiedene Anstellwinkel und Machzahlen sowie die zugehörigen Näherungsparabeln. Es zeigt sich eine hohe Übereinstimmung.

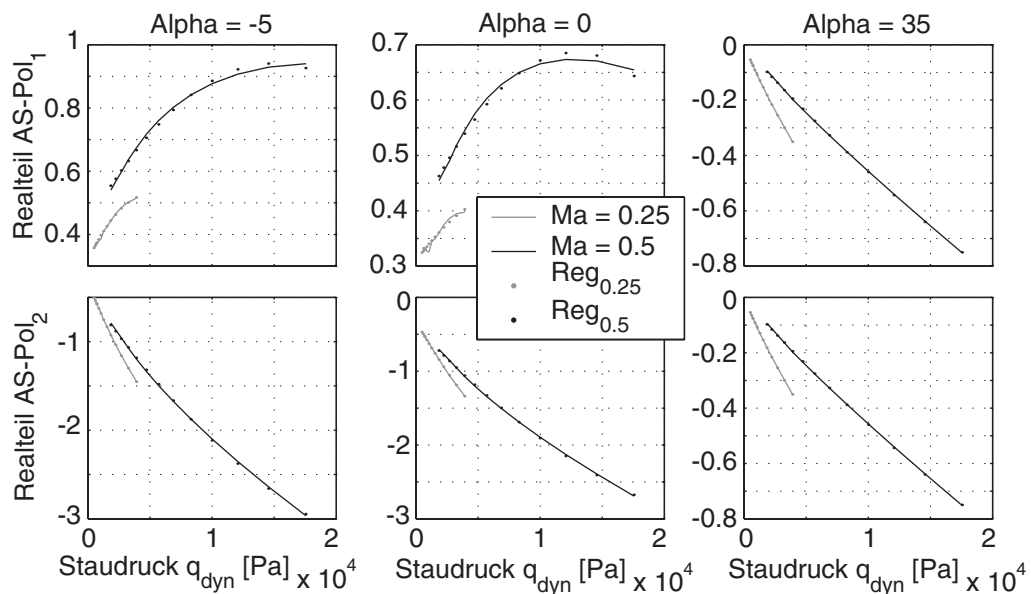


Abb. 2.15: Realteil AS-Pole über Staudruck für verschiedene Anstellwinkel und Machzahlen

Die sichtbar hohe Übereinstimmung in 2.15 wird durch die prozentualen relativen Abweichungen zwischen tatsächlichen Werten und den Werten der Näherungsparabeln in Abbildung 2.16 bestätigt.

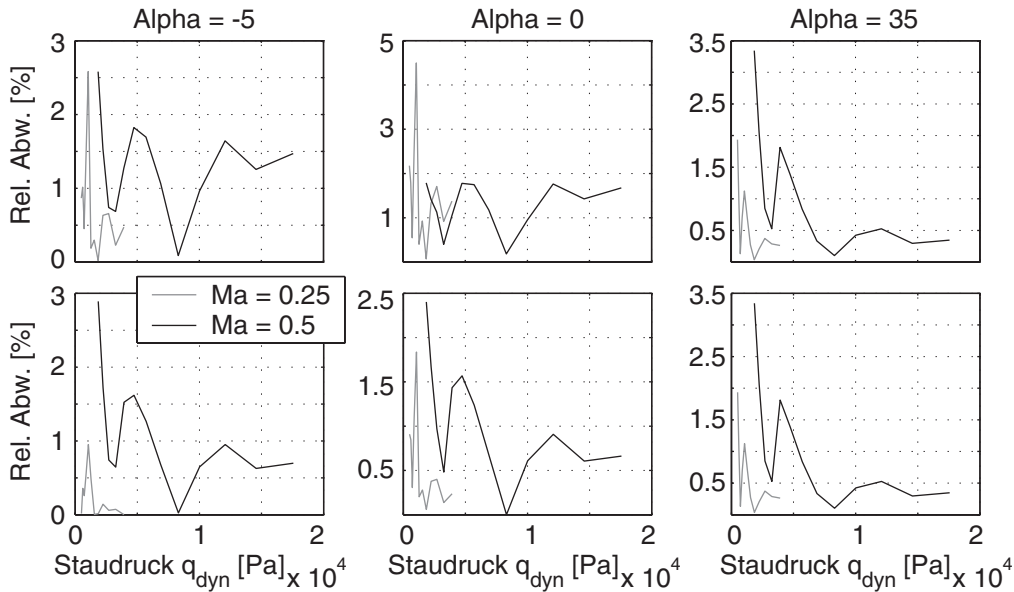


Abb. 2.16: Prozentuale relative Abweichungen zw. Realteilen der AS-Pole und Näherungsparabeln

Die Imaginärteile verhalten sich dort, wo sie ungleich Null sind, analog, auch ihre Verläufe lassen sich gut durch Parabeln annähern.

Abbildung 2.17 zeigt den Verlauf der Realteile der Anstellwinkelschwingung über dem Anstellwinkel für einen konstanten Staudruck von $q_{dyn} = 5000Pa$ und verschiedene Machzahlen. Hier fällt die Änderung der Pollagen mit sich verändernder Machzahl etwas stärker aus als im Fall des Momentenbeiwerts in Abbildung 2.11. Dennoch ist auch hier der Einfluss der Machzahl bei konstantem Staudruck wesentlich geringer als der Einfluss des Staudrucks bei konstanter Machzahl.

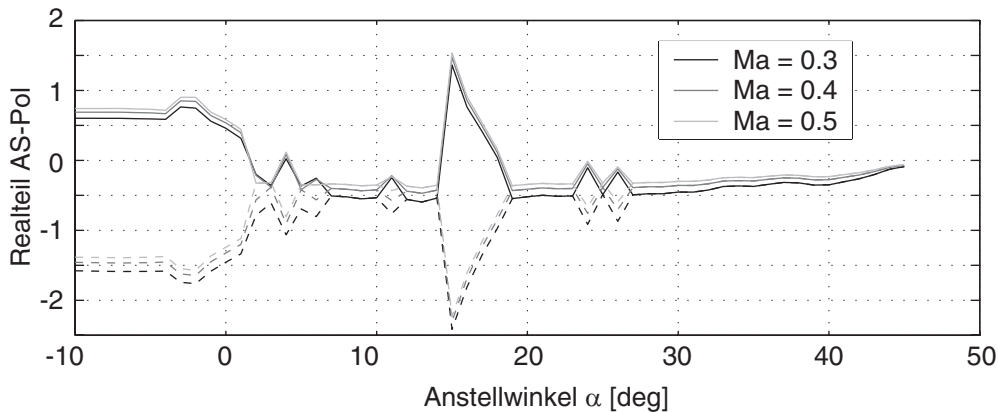


Abb. 2.17: AS-Pole über Anstellwinkel für verschiedene Machzahlen, $q_{dyn} = 5000Pa$

Kapitel 3

Das Reglerentwurfverfahren

Nachdem im vorangegangenen Kapitel die Schritte zur Herleitung eines linearen Entwurfsmodells und dessen Eigenschaften erläutert wurden, folgt in diesem Kapitel die Beschreibung des im Rahmen dieser Arbeit verwendeten Reglerentwurfverfahrens.

3.1 Das H_∞ -Loop-Shaping-Verfahren

Es handelt sich beim H_∞ -Loop-Shaping-Verfahren um ein robustes Reglerentwurfverfahren. Die Motivation für diese Klasse von Entwurfverfahren liegt im Small-Gain-Theorem [21]. Von diesem Theorem kann abgeleitet werden, dass durch die Minimierung der H_∞ -Norm einer stabilen Übertragungsfunktion die Größe (im Sinne ihrer ∞ -Norm) einer unstrukturierten Störung, die die Strecke destabilisiert, maximiert wird.

Im Falle des H_∞ -Loop-Shapings bezieht sich die Robustheit dabei auf Unsicherheiten in der coprime-faktorierten Strecke. Hierbei handelt es sich um eine Art allgemeine Unsicherheit. Hierin liegt einer der Hauptvorteile des Verfahrens gegenüber anderen robusten Reglerentwurfverfahren, denn beim H_∞ -Loop-Shaping sind somit keine expliziten Kenntnisse von den Unsicherheiten in der Regelstrecke geschweige denn deren Modellierung notwendig.

Weitere Entwurfsziele des Reglers neben seiner robusten Stabilität werden berücksichtigt, indem die Singulärwerte der Strecke vor dem Entwurf des eigentlichen stabilisierenden Reglers über Wichtungsfunktionen vor und nach der Regelstrecke so geformt werden, dass der Regelkreis die gegebenen Anforderungen erfüllt. Hierbei kann unter bestimmten Voraussetzungen davon ausgegangen werden, dass der Regler die Singulärwerte der gewichteten Regelstrecke nur wenig verändert [4].

Anforderungen an den Regelkreis im Zeitbereich lassen sich hierbei nicht direkt in Anforderungen an den Verlauf der Singulärwerte der gewichteten Strecke und damit an die Wichtungsfunktionen übersetzen, es existieren jedoch prinzipielle Regeln, nach denen die Wichtungsfunktionen entworfen werden. Die Bestimmung der Wichtungsfunktionen wird in Abschnitt 3.1.3 beschrieben.

Ein weiterer Vorteil des Verfahrens ist die Tatsache, dass der entworfene Regler in Form eines Beobachters mit Zustandsvektorrückführung implementiert werden kann. Hieraus ergibt sich zum einen ein Regler in einer sehr übersichtlichen Darstellung anstatt eines Reg-

lers in Form einer ‘‘Black-Box’’, wie er bei anderen H_∞ -Verfahren oft entsteht. Zum anderen bietet die Beobachterdarstellung die Moglichkeit, variable, vom Flugzustand abhangige Reglerparameter uber ein Gain-Scheduling auf verhaltnismaufig einfache Art und Weise an den Flugzustand anzupassen.

3.1.1 Berechnung des Reglers

Der Entwurf des Reglers geht aus, von der gewichteten Regelstrecke in Abbildung 3.1. Diese kann entsprechend [5] uber den Zwischenschritt der koprim-faktorierten Strecke in eine Form mit externen Ein- bzw. Ausgangen w bzw. z sowie internen Ein- bzw. Ausgangen u_Δ bzw. y_Δ von und zur Matrix Δ , in der die Unsicherheiten der Strecke zusammengefasst sind, umgeformt werden.

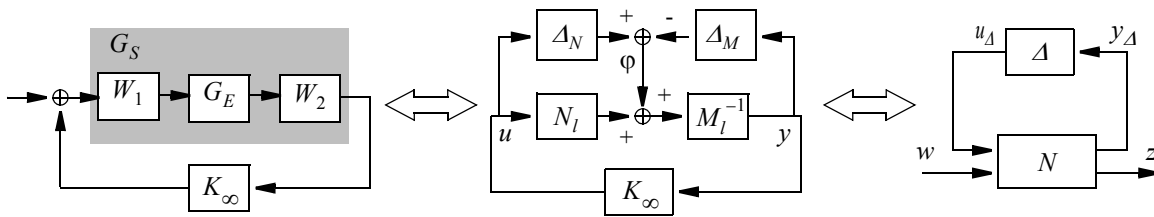


Abb. 3.1: Standard H_∞ -Loop-Shaping Blockdiagramm

In [5] wird gezeigt, dass die Starke $\varepsilon = \|\Delta\|_\infty$ der gerade destabilisierenden Unsicherheit genau dann maximal wird, wenn die ∞ -Norm der Ubertragungsfunktion $N_{1,1}$ (von φ nach u und y) minimal wird. Die Ubertragungsfunktion $N_{1,1}$ lasst sich laut [5] zu dem Ausdruck in Gleichung (3.1) herleiten, und der Regler K_∞ wird entsprechend so entworfen, dass gilt:

$$\|N_{1,1}\|_\infty = \left\| \varphi \rightarrow \begin{bmatrix} u \\ y \end{bmatrix} \right\|_\infty = \left\| \begin{bmatrix} K_\infty \\ I \end{bmatrix} (I - G_S K_\infty)^{-1} M_l^{-1} \right\|_\infty = \frac{1}{\varepsilon} = \gamma \rightarrow \text{Min} \quad (3.1)$$

Hierbei ist M_l^{-1} der erste Faktor der koprim-Faktorisierung der gewichteten Strecke $G_S = M_l^{-1} N_l$ aus Abbildung 3.1, d.h. der Reihenschaltung aus den beiden Wichtungsfunktionen und der Regelstrecke, die wiederum eine Reihenschaltung aus der Ubertragungsfunktion der Aktuatorik, der Kurzzeitnaherung und der Sensordynamiken ist und deren Herleitung im Kapitel 2.2 beschrieben wurde. Es gilt

$$G_S = W_2 G_E W_1 \quad (3.2)$$

mit der Zustandsraumdarstellung

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}. \quad (3.3)$$

Die Losung der Minimierungsaufgabe erfolgt gema uber die Losung der beiden algebraischen Riccati-Gleichungen (Hinweis: es gilt $D = 0$)

$$A^T X + X A - X B B^T X + C^T C = 0 \quad (3.4)$$

und

$$AZ + ZA^T - ZC^T CZ + BC^T = 0. \quad (3.5)$$

Die Lösungen dieser beiden Gleichungen, X und Z werden unter Matlab mit dem Befehl `ric_schr` aus der μ -Synthese-Toolbox berechnet.

Nach Vorgabe eines Wertes γ_{rel} ergibt sich hieraus direkt der Wert für γ gemäß der Vorschrift

$$\gamma = \gamma_{rel} \cdot \sqrt{1 + \rho_s(ZX)}, \quad (3.6)$$

wobei $\rho_s(ZX)$ den Spektralradius von ZX bezeichnet. Der Wert von γ_{rel} wird gemäß [5] bestimmt und beträgt im vorliegenden Fall $\gamma_{rel} = 1,05$.

Ein weiterer wichtiger Vorteil des H_∞ -Shaping-Verfahrens zeigt sich in dieser Entwurfsprozedur, denn sie ist anders als andere gängige H_∞ -Verfahren nicht iterativ, sondern das optimale γ wird direkt berechnet.

Aus γ , X und Z wird die Matrix W berechnet, es gilt

$$W = I - \gamma^{-2}(I + XZ), \quad (3.7)$$

womit nun die eigentlichen Reglermatrizen F und H gemäß den folgenden Formeln berechnet werden

$$F = -B^T(W^{-1}X) \quad (3.8)$$

und

$$H = -ZC^T \quad (3.9)$$

Bei den Matrizen B und C handelt es sich um die B - und C -Matrix der Zustandsraumdarstellung der gewichteten Strecke. Es gibt nun verschiedene Möglichkeiten, die berechneten Matrizen zu verschalten. Diese verschiedenen Reglerimplementierungen werden in Abschnitt 3.1.4. eingehend beschrieben.

3.1.2 Entwurfsziele

Erstes Ziel des Reglerentwurfes ist eine möglichst hohe robuste Stabilität des geschlossenen Regelkreises. Zur Beurteilung der Robustheit eines Entwurfes dient zum einen der Parameter γ , der die eigentliche Zielgröße der H_∞ -Optimierung darstellt und beim Entwurf bestimmt wird. Darüber hinaus erfolgt im Rahmen dieser Arbeit eine Beurteilung der Robustheit anhand der Phasen- und Amplitudenreserve des offenen Regelkreises im Nichols-Diagramm. Hierzu wird der geschlossene Regelkreis zwischen Aktuatorik und Flugzeugmodell aufgeschnitten und es wird der Amplitudengang in dB über dem Phasengang in Hz aufgetragen. Dabei wird gefordert, dass die Ortskurve einen Mindestabstand vom kritischen Punkt bei der Phase -180° und der Amplitude 0dB einhält. Dieser Mindestabstand wird beim Nichols-Robustheitskriterium definiert durch ein halb-diamantenförmiges Gebiet, den sogenannten Nichols-Diamanten, welchen die Ortskurve nicht schneiden darf. In der Literatur existieren

verschiedene, leicht voneinander abweichende Definitionen dieses Nichols-Diamanten, in der vorliegenden Arbeit wird er in der Version von [12] verwendet und es ergibt sich die in Abbildung 3.2 dargestellte Form.

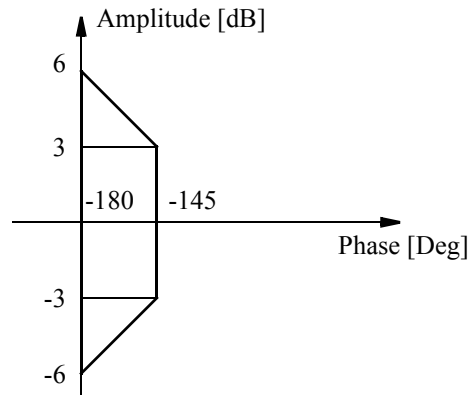


Abb. 3.2: Robustheitsgrenzen im Nichols-Diagramm

Um nun die Robustheit eines Entwurfes zu quantifizieren, wird, wie in [22] vorgeschlagen, die Berechnung einer Stabilitätsreserve ρ vorgenommen (und ist nicht mit dem oben erwähnten Spektralradius ρ_s zu verwechseln). Der Wert der Stabilitätsreserve beträgt $\rho = 1$ für den Fall, dass die Ortskurve den Nichols-Diamanten in Abbildung 3.2 gerade berührt. Für Ortskurven, die den Nicholsdiamanten nicht berühren bzw. schneiden, wird der Diamant durch eine zentrische Streckung in Bezug auf den kritischen Punkt so skaliert, dass die Ortskurve den gestreckten bzw. gestauchten Diamanten gerade berührt. Die Stabilitätsreserve ergibt sich dann aus dem Faktor der zentrischen Streckung, d.h. für Fälle, in denen die Ortskurve den Diamanten schneidet und das Nichols-Robustheitskriterium nicht erfüllt ist, ergibt sich ein Wert $\rho < 1$, für Fälle, in denen die Ortskurve den Diamanten nicht berührt und das Robustheitskriterium übererfüllt ist, ergibt sich ein Wert von $\rho > 1$. Abbildung 3.3 zeigt einen beispielhaften Verlauf einer Ortskurve im Nichols-Diagramm.

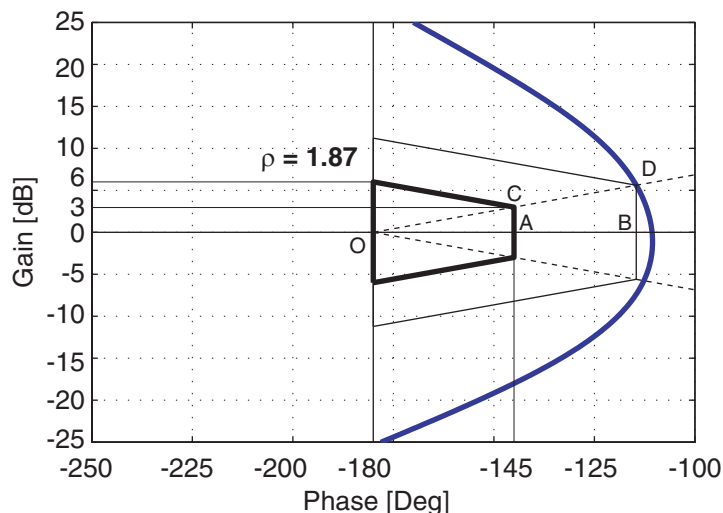


Abb. 3.3: Berechnung der Stabilitätsreserve ρ im Nichols-Diagramm

Die Kurve schneidet den Diamanten nicht, der Wert für die Stabilitätsreserve ergibt sich aus dem Verhältnis der Strecken OB zu OA:

$$\rho = \frac{\overline{OB}}{\overline{OA}} = 1,87. \quad (3.10)$$

Neben einer möglichst hohen Robustheit werden an den Regler Anforderungen in Form von Handling-Qualities gestellt. So wird verlangt, dass die Dämpfung der Anstellwinkelschwingung ζ_{AS} des geregelten Flugzeugs einen Minimalwert von 0,7 nicht unterschreitet. Es gilt also

$$\zeta_{AS} \geq \zeta_{AS, \min} = 0,7. \quad (3.11)$$

Des Weiteren wird ein Control Anticipation Parameter (*CAP*) gemäß den MIL-Anstellwinkelforderungen nach Level 1* verlangt [13]. Hierbei handelt es sich um ein Flugeigenchaftskriterium, über das quantifiziert wird, inwiefern es dem Piloten möglich ist, aus der Anfangsreaktion des Flugzeugs auf eine sprungförmige Änderung der Höhenruderstellung die folgende Bahnänderung abzuschätzen.

Dementsprechend berechnet sich der Wert des *CAP* definitionsgemäß aus der Reaktion des Flugzeugs auf einen Sprungeingang am Höhenruder über die Formel

$$CAP = \frac{\dot{q}(0)}{n_z(\infty)}. \quad (3.12)$$

Der *CAP* ist also der Quotient aus der Nickrate zum Zeitpunkt $t = 0$ und dem sich für $t \rightarrow \infty$ einstellenden stationären Lastvielfachen in z -Richtung. Aus den erweiterten MIL-Forderungen (Category A) für die Anstellwinkelreglerauslegung gemäß *CAP* nach Level 1* entsprechend Abbildung C.1 in Anhang C.1 ergibt sich durch Vereinfachung für den vorliegenden Reglerentwurf die Forderung von $CAP = 1$.

(Anmerkung: Die MIL-Forderungen bestehen aus einer Vorgabe des Quotienten Q_{CAP} aus dem Quadrat der Eigenfrequenz der Anstellwinkelschwingung ω_{AS} und der Übertragung $n_{z, \alpha}$ des Anstellwinkels α auf das Lastvielfache n_z der Kurzzeitnäherung. Gleichzeitig handelt es sich bei diesem Quotienten um eine Näherung für den *CAP*-Wert (siehe auch Abschnitt 4.1.3) und es gilt:

$$Q_{CAP} = \frac{\omega_{AS}^2}{n_{z, \alpha}} \approx CAP. \quad (3.13)$$

Laut den MIL-Forderungen für *CAP* nach Level 1* muss nun gelten

$$Q_{CAP} \approx CAP = \begin{cases} 1 & \text{für } n_{z, \alpha} \geq 3,5714 \\ 3,6 / \omega_{AS}^2 & \text{für } n_{z, \alpha} < 3,5714 \end{cases}. \quad (3.14)$$

Im Fall der vorliegenden Arbeit wird die *CAP*-Bedingung jedoch insofern vereinfacht, dass die Forderung $CAP = 1$ für alle $n_{z, \alpha}$ angenommen wird. Im Diagramm C.1 bedeutet dies die Vernachlässigung des Knickes in der Level 1* Linie.)

Sowohl die Bestimmung der Dämpfung der Anstellwinkelschwingung als auch die Berechnung des *CAP*-Wertes sind nicht trivial. Zum einen ist es schwierig, am geschlossenen

Regelkreis die Pole der Anstellwinkelschwingung zu identifizieren. Zum anderen bezieht sich der *CAP*-Wert definitionsgemäß auf unregelte Flugzeuge. Die Ansätze zur Lösung dieser Problematiken werden eingehend in Abschnitt 4.1.3 beschrieben.

Laut technischer Anleitung [13] wird außerdem für kommandierte Sprünge im Anstellwinkel ein möglichst geringes Überschwingen gefordert.

3.1.3 Wahl der Wichtungsfunktionen

Durch die Wichtungsfunktionen erfolgt eine Formung der Singulärwerte der Regelstrecke. Diese Formung erfolgt in erster Linie hinsichtlich eines guten Folgeverhaltens des Reglers sowie einer akzeptablen Unempfindlichkeit gegenüber Messrauschen. Diese Anforderungen lassen sich erreichen mit einer Regelstrecke mit großen Verstärkungen im niederen Frequenzbereich (Folgeverhalten) sowie kleinen Verstärkungen im hohen Frequenzbereich (Rauschempfindlichkeit). Dementsprechend wird als Wichtungsfunktion W_1 eine Übertragungsfunktion mit integrierendem Verhalten verwendet, sie stellt große Verstärkungen bei kleinen Frequenzen sicher. Bei der Wichtungsfunktion W_2 handelt es sich dagegen um eine Übertragungsfunktion mit Tiefpasscharakter, um kleine Verstärkungen für große Frequenzen zu gewährleisten. Hinsichtlich der Ordnung der Übertragungsfunktionen ist zu beachten, dass sie in die Ordnung des Reglers eingeht. Die Ordnung des Reglers ist gleich der Ordnung der Regelstrecke, das heißt sie setzt sich zusammen aus der Ordnung der Aktuatorik und Sensorik, der Kurzzeitnäherung sowie der Wichtungsfunktionen. Um einen übermäßig komplexen Regler zu vermeiden, wird die Ordnung der Wichtungsfunktionen daher möglichst gering gehalten.

Für die Wichtungsfunktion W_1 wird daher ein PI-Filter verwendet. Dieser wird über einen festen Proportionalanteil von $K_{P1} = 1$ und eine variable Zeitkonstante ω_1 definiert, die wiederum Einfluss auf die Bandbreite und damit die Geschwindigkeit der gewichteten Strecke und damit des Regelkreises hat.

Für die Wichtungsfunktion W_2 verwendet man im Fall der Wichtung der Anstellwinkelmessung ein Tiefpassfilter mit einer Eckfrequenz von $\omega_2 = 50\text{Hz}$ und einem variablen Proportionalanteil $K_{P2,1}$. Im Fall der Nickratenmessung wird dagegen ein Bandpassfilter verwendet. Ein Tiefpassfilter wie im Falle der Messung des Anstellwinkels reicht hier nicht aus, da die Wichtungsfunktion die stationäre Übertragung der Nickratenmessung unterdrücken muss. Der Grund liegt in der Verwendung der Kurzzeitnäherung für den Folgereglerentwurf. Für die Kurzzeitnäherung ergibt sich für einen Sprung an ihrem Eingang nicht nur ein neuer Stationärwert des Anstellwinkels, sondern auch ein von Null verschiedener Stationärwert der Nickrate. Wird nun dieser im Falle einer Anstellwinkelregelung über den Regler auf die Summationsstelle zwischen dem Reglerausgang und dem kommandierten Anstellwinkel α_c zurückgeführt, dann ergibt sich nach der Summationsstelle aufgrund des Nickratenanteils im Reglerausgang auch für $\alpha = \alpha_c$ ein Wert ungleich Null, was eine Folgeregelung für α unmöglich macht. Deshalb muss die Wichtungsfunktion W_2 im q -Kanal für eine Unterdrückung des stationären q -Ausgangs der Kurzzeitnäherung sorgen.

Der q -Kanal der Wichtungsfunktion W_2 wird daher durch einen Bandpass zweiter Ordnung realisiert. Dieser wird definiert über eine ebenfalls feste Knickfrequenz bei $\omega_2 = 50\text{Hz}$,

einem variablen Proportionalanteil $K_{P2,2}$ sowie seiner zweiten Knickfrequenz, die der Knickfrequenz ω_1 des PI-Filters der Wichtungsfunktion W_1 entspricht.

Für die Übertragungsfunktion W_1 gilt:

$$W_1 = \frac{K_{P1}s + K_{P1}\omega_1}{s}, K_{P1} = 1 \quad (3.15)$$

Für die Wichtungsfunktion der α -Messung gilt

$$W_{2,1} = \frac{K_{P2,1}\omega_2}{s + \omega_2}, \quad (3.16)$$

sowie für die Wichtung der q -Messung

$$W_{2,2} = \frac{K_{P2,2}\omega_2 s}{s^2 + (\omega_1 + \omega_2)s + \omega_1\omega_2} \quad (3.17)$$

Somit ergeben sich für Reglerentwurf drei freie Parameter, ω_1 , $K_{P2,1}$ und $K_{P2,2}$, deren Werte so zu bestimmen sind, dass die Entwurfsziele in Abschnitt 3.1.2 erreicht werden. Eine Methode zur systematischen Bestimmung der Entwurfsparameter durch nichtlineare Optimierung wird in Kapitel 4 vorgestellt und beschrieben.

3.1.4 Implementierung des linearen Reglers

Wie bereits erwähnt, existieren verschiedene Arten, die gemäß Abschnitt 3.1.1 berechneten Matrizen F und H zu einem Regler und mit der Strecke zu einem Regelkreis zu implementieren. Diese Arten unterscheiden sich zum einen durch die Struktur des Reglers und zum anderen durch die Anordnung des Reglers im Regelkreis. Es gibt folgende Implementierungsarten:

- Regler im Rückwärtszweig als dynamisches System in Zustandsraumdarstellung
- Regler im Rückwärtszweig als dynamisches System mit beobachterähnlicher Struktur
- Regler in Form eines Beobachters mit Zustandsvektorrückführung
- Regler im Vorwärtszweig als dynamisches System in Zustandsraumdarstellung
- Regler im Vorwärtszweig als dynamisches System mit beobachterähnlicher Struktur

Die ersten beiden Implementierungsarten unterscheiden sich dabei nur in der Darstellung des Reglers, die dynamischen Eigenschaften der Regelkreise sind identisch. Die Darstellungen lassen sich ineinander überführen. Das Gleiche gilt für die letzten beiden Implementierungsarten. Dementsprechend können bezüglich ihrer dynamischen Eigenschaften drei Implementierungsarten unterschieden werden, die Implementierung im Vorwärtszweig, die Implementierung im Rückwärtszweig und die Beobachterdarstellung mit Zustandsvektorrückführung.

Zunächst wird die **Beobachterdarstellung mit Zustandsvektorrückführung** beschrieben. Abbildung 3.4 zeigt ihren Aufbau:

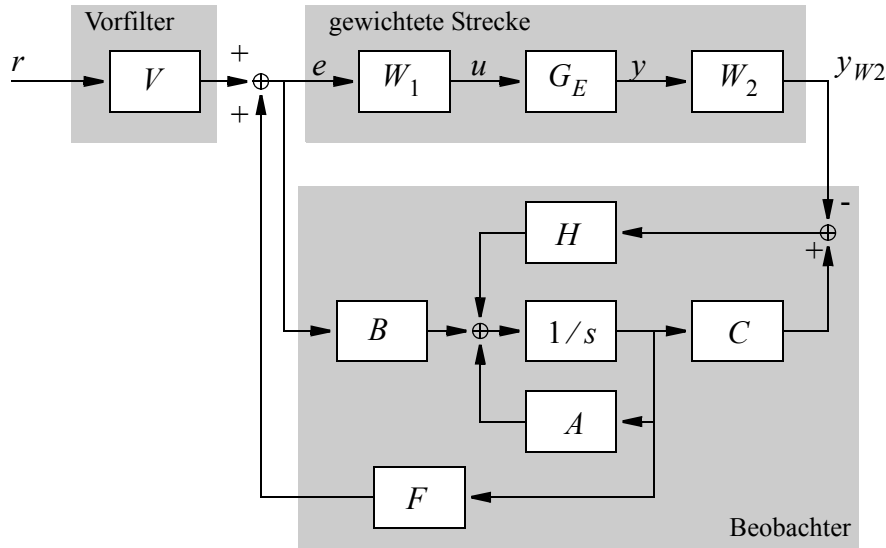


Abb. 3.4: Reglerimplementierung als Beobachter mit Zustandsvektorrückführung

Bei den Matrizen A , B und C handelt es sich um die entsprechenden Matrizen der Zustandsraumdarstellung der gewichteten Strecke. Die Berechnung der Matrizen F und H erfolgt nach Gleichung (3.8) bzw. (3.9) in Abschnitt 3.1.1. Das Vorfilter V wird benötigt, um stationäre Regelgenauigkeit zu erreichen. Für die Anstellwinkelfolgeregelung muss im Stationärfall (Index ss) für den Anstellwinkel, also den ersten Ausgang der Strecke G_E , gelten

$$y_{1,ss} = \alpha_{ss} = r. \quad (3.18)$$

Genau genommen wird natürlich nicht vom Ausgang der Strecke G_E stationäre Genauigkeit gefordert, denn beim Ausgang von G_E handelt es sich um den Sensorausgang, d.h. um die *gemessenen* Werte von Anstellwinkel und Nickrate anstatt der *tatsächlichen*. Die stationäre Übertragung des Sensors für konstante Anstellwinkel und Nickraten ist jedoch $G_{SENS,ss} = 1$, weshalb der Einfachheit halber hier der Ausgang y der Strecke G_E betrachtet werden kann und eine Aufspaltung von G_E in Aktuator, Sensor und eigentliche Regelstrecke nicht notwendig ist.

Aufgrund des Integrators in der Wichtungsfunktion W_2 muss außerdem die Bedingung

$$e_{ss} = 0 \quad (3.19)$$

erfüllt sein. Bei der Reglerabweichung e handelt es sich um die Differenz aus dem Beobachteraussgang und dem Produkt aus Referenzsignal r und der Verstärkung des Vorfilters V . Im stationären Zustand muss also gelten

$$e_{ss} = -Vr + K_{ss,y}y_{W2,ss} + K_{ss,e}e_{ss} = 0. \quad (3.20)$$

Hierbei sind $y_{W_2,ss}$ der Ausgang der Wichtungsfunktion W_2 im stationären Zustand und $K_{ss,y}$ bzw. $K_{ss,e}$ die stationären Übertragungen des Reglers von dessen Eingang y_{W_2} bzw. e auf seinen Ausgang, jeweils für konstante Eingänge. Es gilt also

$$y_{W_2,ss} = W_{2,ss} y_{ss}. \quad (3.21)$$

Zur Berechnung der stationären Übertragungen des Beobachters, $K_{ss,y}$ und $K_{ss,e}$, wird zunächst dessen Zustandsableitung betrachtet. Für den Beobachterzustand \tilde{x} folgt aus Abbildung 3.4:

$$\dot{\tilde{x}} = (A + HC)\tilde{x} + Be - Hy_{W_2}. \quad (3.22)$$

Mit der Bedingung $\dot{\tilde{x}}_{ss} = 0$, der Gleichung $\tilde{y} = F\tilde{x}$ für den Ausgang des Beobachters sowie der Gleichung (3.19) folgt hieraus für die stationäre Übertragung des Beobachters

$$\tilde{y}_{ss} = -F(A + HC)^{-1}Hy_{W_2,ss} = K_{ss,y}y_{W_2,ss}, \quad (3.23)$$

der Anteil vom Eingang e entfällt, da gelten muss $e_{ss} = 0$.

Setzt man die Gleichungen (3.23) und (3.21) in Gleichung (3.20) ein, folgt

$$0 = -Vr - F(A + HC)^{-1}HW_{2,ss}y_{ss} \quad (3.24)$$

mit

$$y_{ss} = \begin{bmatrix} \alpha_{ss} \\ q_{ss} \end{bmatrix} = \alpha_{ss} \begin{bmatrix} 1 \\ q_{ss}/\alpha_{ss} \end{bmatrix}. \quad (3.25)$$

Setzt man Gleichung (3.25) unter Verwendung der Bedingung (3.18) in Gleichung (3.24) ein, folgt

$$0 = -Vr - F(A + HC)^{-1}HW_{2,ss} \begin{bmatrix} 1 \\ r/q_{ss} \end{bmatrix} r, \quad (3.26)$$

woraus sich schließlich V bestimmen lässt zu

$$V = -F(A + HC)^{-1}HW_{2,ss} \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (3.27)$$

Es fällt auf, dass der Term r/q_{ss} zu Null gesetzt wurde. Dies ist zulässig, da es sich beim q -Kanal der Wichtungsfunktion W_2 um einen Bandpass handelt, dessen stationäre Übertragung für konstante Eingänge $q_{ss} = \text{const}$ Null ist. Die stationäre Übertragung von W_2 hat also die Struktur

$$W_{2,ss} = \begin{bmatrix} w_{1,1} & 0 \\ 0 & 0 \end{bmatrix}. \quad (3.28)$$

Deshalb kann die zweite Komponente des Vektors y_{ss} durch eine beliebige (konstante) Komponente, in diesem Fall Null, ersetzt werden.

Aus dem Blockschaltbild in Abbildung 3.4 lässt sich für den Beobachter mit Zustandsvektorrückführung folgende Zustandsraumdarstellung für die Übertragungsfunktion des geschlossenen Regelkreises von r nach y_{W2} ableiten:

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \begin{bmatrix} A & BF \\ -HC & A + HC + BF \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} -BV \\ -BV \end{bmatrix} r \quad (3.29)$$

und

$$y_{W2} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} \quad (3.30)$$

Wie bereits erwähnt, im Betrieb des Reglers interessiert (genau genommen) nicht der Ausgang y_{W2} , also der Ausgang der Wichtungsfunktion W_2 , sondern der tatsächliche Ausgang der Regelstrecke, hier also der Kurzzeitznäherung. Um die Übertragungsfunktion von r auf y_{SPA} zu bilden, müsste jedoch die gewichtete Strecke G_S in ihre Bestandteile Wichtungsfunktion W_1 , Aktuatorübertragungsfunktion G_{ACT} , Kurzzeitznäherung G_{SPA} , Sensorübertragung G_{SENS} und Wichtungsfunktion W_2 zerlegt werden, um den Ausgang y_{SPA} über den Zustand x_{SPA} zu erreichen. Die Übertragungsfunktion hätte dann eine erheblich größere Komplexität als die Zustandsraumdarstellung in Gleichung (3.29) und (3.30). Für den Vergleich der verschiedenen Reglerimplementierungen ist diese Darstellung jedoch ausreichend. Eine ausführlichere Darstellung mit den Zuständen der Wichtungsfunktionen und der Entwurfstrecke G_E befindet sich in Anhang D.

Die **Implementierung im Rückwärtszweig** ist in Abbildung 3.5 dargestellt, hier mit dem Regler in beobachterähnlicher Struktur.

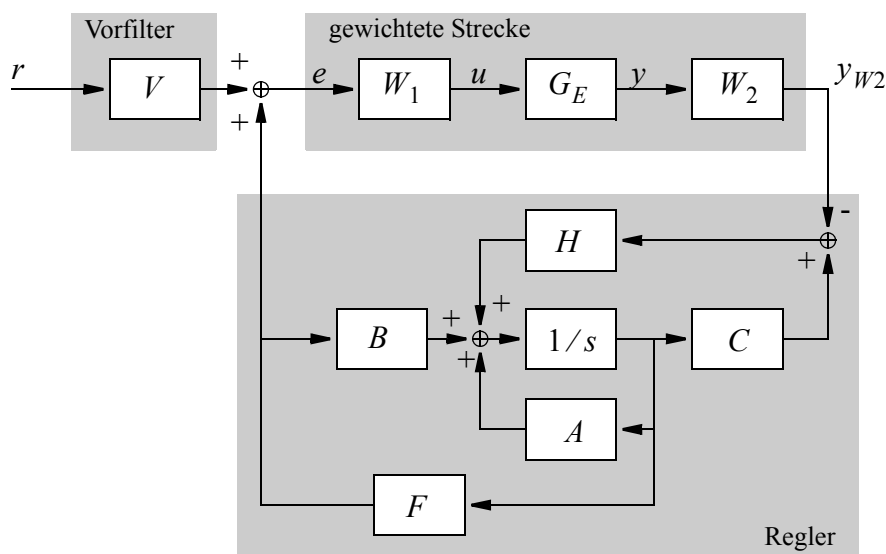


Abb. 3.5: Regler im Rückwärtszweig mit beobachterähnlicher Struktur

Auch hier stehen die Matrizen A , B und C für die entsprechenden Matrizen der Zustandsraumdarstellung der gewichteten Strecke. Die Reglermatrizen H und F berechnen sich wie im Fall des Beobachters nach den Gleichungen (3.8) und (3.9). Wie im Fall des Beobachters wird auch hier ein Vorfilter V für stationäre Regelgenauigkeit benötigt. Die Herleitung erfolgt analog zum Fall des Beobachters mit Zustandsvektorrückführung und es ergibt sich

$$V = -F(A + BF + HC)^{-1}HW_{2,ss} \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (3.31)$$

Der Unterschied zum Beobachter mit Zustandsvektorrückführung besteht darin, dass der Regler nun nur noch einen Eingang, den Ausgang y_{W2} der Wichtungsfunktion W_2 , hat. Das Blockschaltbild in Abbildung 3.5 lässt sich also vereinfachen zu der Darstellung in Abbildung 3.6. Zu beachten ist auch das geänderte Vorzeichen an der Summationsstelle am Ausgang der Matrix C sowie an der Summationsstelle am Eingang des Integrators des Reglers.

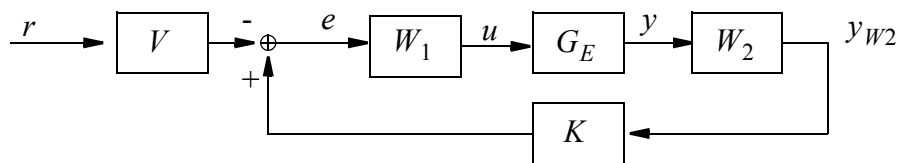


Abb. 3.6: Regler im Rückwärtszweig

Wie bereits erwähnt, kann der Regler K dabei auch in Form einer Zustandsraumdarstellung mit den Matrizen A_K , B_K , C_K und D_K dargestellt werden. Beide Varianten der Implementierung im Rückwärtszweig sind jedoch hinsichtlich ihres dynamischen Verhaltens identisch, weshalb auf diese Variante nicht näher eingegangen wird.

Die Zustandsraumdarstellung des geschlossenen Regelkreises von r nach y_{W2} ergibt sich in diesem Fall nach Abbildung 3.5 zu

$$\begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{x} \end{bmatrix} = \begin{bmatrix} A & BF \\ -HC & A + HC + BF \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} -BV \\ 0 \end{bmatrix} r \quad (3.32)$$

und

$$y_{W2} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix}. \quad (3.33)$$

Abbildung 3.7 zeigt die **Implementierung im Vorwärtszweig**. Die Summationsstelle zwischen Reglerausgang und Referenzsignal befindet sich nun nicht mehr zwischen dem Regler und der Wichtungsfunktion W_1 , sondern zwischen dem Streckenausgang und der Wichtungsfunktion W_2 . In diesem Fall wird kein Vorfilter benötigt. Die Matrizen A , B und C

sind dieselben wie im Fall des Reglers im Rückwärtszweig, auch die Reglermatrizen F und H werden wiederum gemäß den Gleichungen (3.8) und (3.9) berechnet.

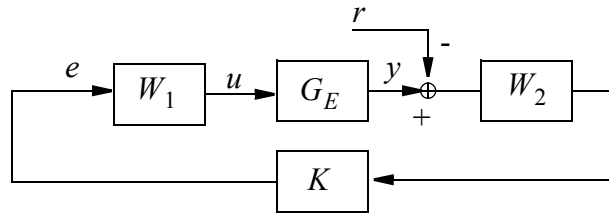


Abb. 3.7: Regler im Vorwärtszweig

Auch hier lässt sich der Regler wie in Abbildung 3.8 dargestellt, in Form einer beobachterähnlichen Struktur realisieren oder in Form einer Zustandsraumdarstellung mit den Matrizen A , B , C , F und H .

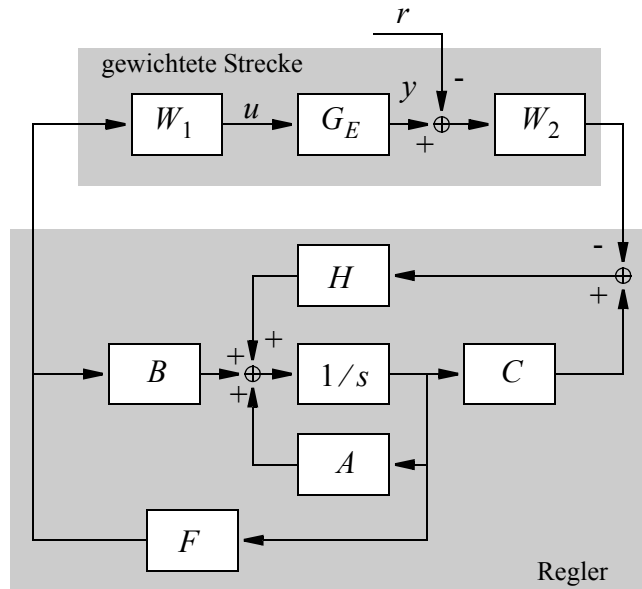


Abb. 3.8: Regler im Vorwärtszweig mit beobachterähnlicher Struktur

Die Zustandsraumdarstellung des geschlossenen Regelkreises mit dem Eingang r und dem Ausgang y_{W2} ergibt sich für diese Implementierung gemäß Abbildung 3.8 zu

$$\begin{bmatrix} \dot{x} \\ \tilde{x} \end{bmatrix} = \begin{bmatrix} A & BF \\ -HC & A + BF + HC \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} \check{f}(B_{W2}) \\ 0 \end{bmatrix} r \tag{3.34}$$

und

$$y_{W2} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix}. \tag{3.35}$$

Für die ausführliche Darstellung des geschlossenen Regelkreises mit der kompletten Darstellung des Vektors $\check{f}(B_{W2})$ sei auf Anhang D verwiesen.

3.1.5 Vergleich der Implementierungen

Die Betrachtung der Zustandsraumdarstellungen der geschlossenen Regelkreise in den Gleichungen (3.29), (3.32) und (3.34) zeigt, dass alle Implementierungsarten dieselbe Dynamikmatrix A und damit dieselben Pole haben. Auch die Robustheitseigenschaften unterscheiden sich nicht, denn die Robustheit in Form des Abstands vom Nicholsdiamanten wird anhand der Ortskurve des offenen Regelkreises ermittelt. Da sich die Regelkreise nur durch die Lage der Summationsstelle zwischen Mess- und Referenzsignal unterscheiden, diese Lage im Fall des offenen Regelkreises jedoch keinen Einfluss hat, entsprechen sich die aufgeschnittenen Regelkreise.

Unterschiede ergeben sich jedoch bezüglich des Verhaltens der Regelkreise im Zeitbereich. Dies leuchtet unmittelbar ein, denn befindet sich der Regler beispielsweise im Vorwärtszweig, so wird er von jeder Abweichung zwischen Referenzsignal und Messung direkt ange-regt. Es ergibt sich dadurch eine schnellere Reaktion als im Fall der Anordnung des Reglers im Rückwärtszweig. In diesem Fall tritt der Regelfehler erst über die Strecke verzögert in den Regler ein. Abbildung 3.9 belegt diese Überlegungen. Im Schaubild ist die Zeitantwort auf einen Sprungeingang für die drei Implementierungsarten für einen exemplarischen Reglerentwurf dargestellt. Der Entwurf erfolgt für die Kurzzeitznäherung für den Flug auf einem quasi-stationären Abfangbogen bei einem Anstellwinkel von $\alpha = 15^\circ$, einer Machzahl von $Ma = 0,3$ und einem Staudruck von $q_{dyn} = 5397\text{Pa}$. Die freien Parameter der Wichtungsfunktionen wurden zu $\omega_1 = 5\text{Hz}$, $K_{P2,1} = -10\text{dB}$ und $K_{P2,2} = 5\text{dB}$ gewählt. Die zeitlichen Verläufe zeigen, dass der Regler im Vorwärtszweig zwar die schnellste Reaktion erzielt, aufgrund der direkten Anregung der Reglerdynamik kommt es jedoch auch zu starkem Überschwingen. Der Regelkreis mit dem Regler im Rückwärtszweig ist wie erwartet langsamer, schwingt jedoch dafür auch nicht über. Sehr gute Ergebnisse erzielt der Beobachter mit Zustandsvektorrückführung. Er ist aufgrund seines zweiten Eingangs, über den der Regelfehler direkt auf den Regler wirkt, schneller als der Regler im Rückwärtszweig und neigt dabei trotzdem kaum zum Überschwingen. Aufgrund dieser positiven Eigenschaften erfolgt die Realisierung des im Rahmen dieser Arbeit entworfenen Reglers in Form der Implementierungsart des Beobachters mit Zustandsvektorrückführung.

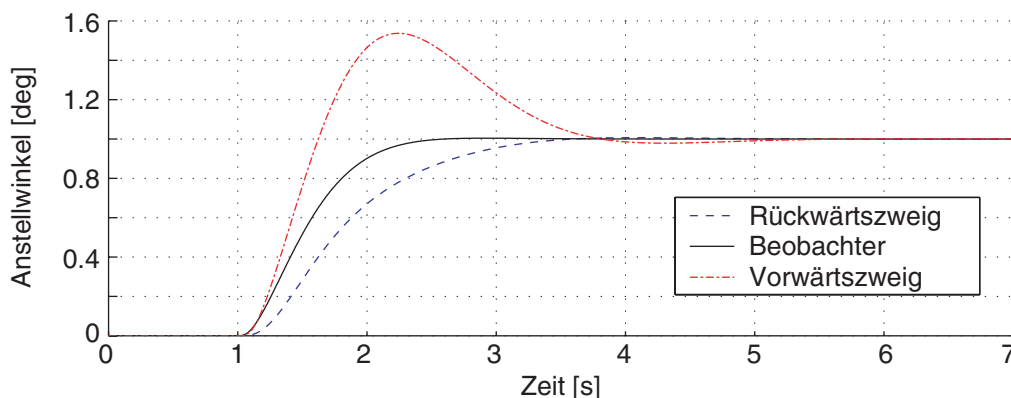


Abb. 3.9: Anstellwinkelverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen

Auch der Verlauf der Nickrate in Abbildung 3.10 zeigt, dass im Falle des Reglers im Vorwärtszweig eine stärkere Anregung der Dynamik erfolgt. Der Nickratenverlauf im Falle des

Beobachters verläuft analog zum Anstellwinkel zwischen den Kurven der beiden anderen Implementierungen.

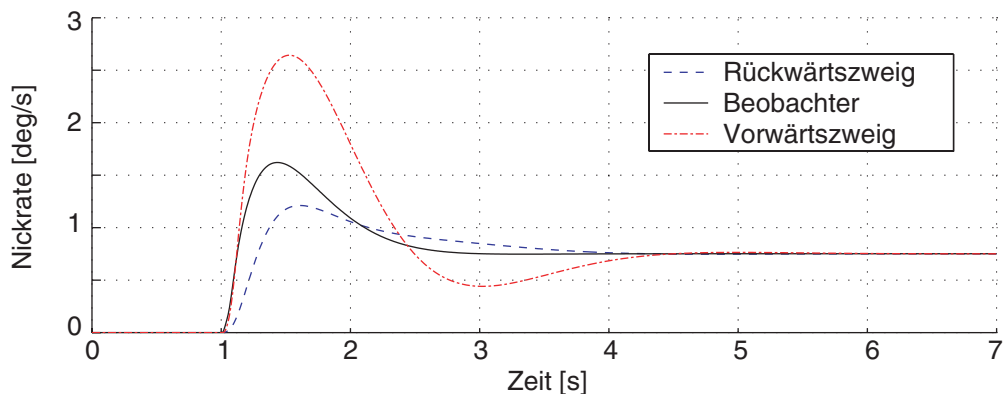


Abb. 3.10: Nickratenverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen

In Anbetracht der Verläufe von Anstellwinkel und Nickrate in den Abbildungen 3.9 und 3.10 ist zu erwarten, dass der Regler im Vorwärtszweig eine stärkere Regelaktivität und damit größere Ruderausschläge aufweist als bei einer Implementierung als Beobachter oder im Rückwärtszweig. Der Verlauf des symmetrischen Ausschlags des Tailerons in Abbildung 3.11 bestätigt diese Annahme. Im Fall des Reglers im Vorwärtszweig ist der Ruderausschlag sowohl betragsmäßig am größten als auch bezüglich seiner Rate. Der Regler im Rückwärtszweig weist passend zu seiner relativ langsamen Anstellwinkelsprungantwort die geringste Aktuatortätigkeit auf. Auch hier liegt der Verlauf für den Beobachter zwischen den beiden anderen Kurven.

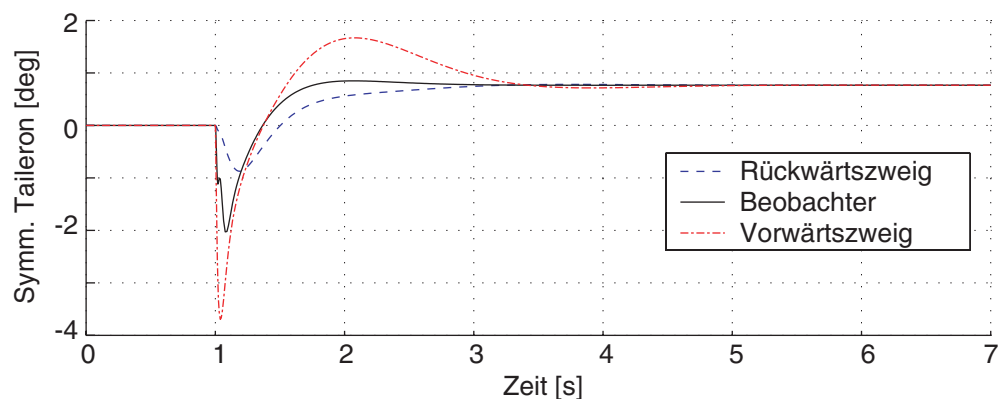


Abb. 3.11: Ruderstellungsverlauf für Sprungeingang bei verschiedenen Reglerimplementierungen

3.2 Das Gain-Scheduling-Verfahren

Obwohl es sich beim H_∞ -Loop-Shaping-Verfahren um ein robustes Reglerentwurfsverfahren handelt, das Abweichungen zwischen Realität und Entwurfsmodell tolerieren kann, handelt es sich dennoch um ein lineares Verfahren, das auf einer linearisierten Darstellung der Entwurfstrecke basiert. Aufgrund der starken Nichtlinearitäten des Modells des Jettrainers,

die sich im Rahmen der Analyse des Entwurfsmodells in Abschnitt 2.3 zeigten, sind für Flugzustände, die vom Arbeitspunkt der Linearisierung abweichen, so große Abweichungen des linearen Entwurfsmodells von der Realität zu erwarten, dass sie nicht mehr durch die Robustheit des Reglers abzufangen sind. Die Stabilität des Flugzeugs kann dann an Flugzuständen, die vom Entwurfsflugzustand abweichen, nicht mehr garantiert werden, geschweige denn die Einhaltung der Handling-Quality-Entwurfsziele.

Abhilfe schafft das Gain-Scheduling-Verfahren. Hierbei macht man sich zunutze, dass die über das H_∞ -Loop-Shaping-Verfahren entworfenen Regler auch für unterschiedliche Entwurfspunkte und dementsprechend unterschiedliche Parameter der Wichtungsfunktionen immer die gleiche Struktur haben. Die Regler werden immer aus den Matrizen A , B , C , F und H aufgebaut, lediglich die Elemente der einzelnen Matrizen unterscheiden sich voneinander. Es werden deshalb für ein Raster von Entwurfspunkten Regler entworfen, um anschließend die Matrizenelemente an Nicht-Entwurfspunkten mittels Interpolation aus dem Raster der Entwurfspunkte und ihrer Reglerdaten zu bestimmen. Die Rasterung der Envelope und die Anpassung der Matrizenelemente erfolgt über die sogenannten Scheduling-Parameter \vec{p}_s , jedes variable Matrizenelement wird dargestellt durch eine Funktion in Abhängigkeit dieser Parameter, es gilt also

$$m_{i,j} = f_{i,j}(\vec{p}_s) \quad (3.36)$$

Es kann sich dabei um einen einzigen Parameter handeln, wie etwa die Fluggeschwindigkeit im Beispiel [6], oder auch um mehrere Parameter, wobei die Anpassung an den Flugzustand entsprechend durch mehrdimensionale Interpolation erfolgt.

3.2.1 Wahl der Scheduling-Parameter

Das Kapitel 2.3 zeigte die starken Nichtlinearitäten des Entwurfsmodells über dem Anstellwinkel. Je nach Anstellwinkel ändert das Flugzeug seine Eigenschaften grundlegend, ist beispielsweise stabil oder instabil. Es erfolgt im vorliegenden Fall daher ein Scheduling über den Anstellwinkel.

Obwohl sich in Abschnitt 2.3 zeigte, dass die Machzahl im gegebenen Bereich nur einen sehr kleinen Einfluss auf das Flugzeugverhalten hat, erfolgt dennoch auch ein Scheduling über die Machzahl. Dies ist sinnvoll im Hinblick auf die Tatsache, dass der Jettrainer nicht nur bis zu einer maximalen Machzahl von $Ma = 0,5$ betrieben werden kann und dass bei höheren Machzahlen, die mit entsprechenden Aerodynamikmodellen untersucht werden können, nichtlineare Machzahleffekte zu erwarten sind.

Ebenfalls in Abschnitt 2.3 zeigte sich eine Abhängigkeit des Flugzeugverhaltens von der Höhe bzw. vom Staudruck. Als dritter Scheduling-Parameter kommen daher entweder die Höhe oder alternativ dazu der Staudruck in Betracht. Ein Scheduling über beide dieser Parameter ist nicht notwendig, da Machzahl, Höhe und Staudruck über die Modellierung der Atmosphäre gekoppelt sind, durch die Vorgabe von zweien dieser Parameter ergibt sich der dritte also automatisch. Im vorliegenden Fall wird als dritter Scheduling-Parameter der Staudruck verwendet, denn wie in Abschnitt 2.3.2 gezeigt wurde, hat er den Vorteil, dass die Dynamik des Entwurfsmodells für konstanten Anstellwinkel und konstante Machzahl eine

quadratische Abhängigkeit vom Staudruck aufweist. Unter der Annahme, dass diese Abhängigkeit auch bei den Matrizenelementen des Reglers auftritt, oder zumindest bei einigen von ihnen, besteht durch die Verwendung des Staudrucks als drittem Scheduling-Parameter die Möglichkeit, eine dreidimensionale Interpolation zu vermeiden: Für drei Scheduling-Parameter $p_{s1} = \alpha$, $p_{s2} = Ma$ und $p_{s3} = q_{dyn}$ gilt gemäß Gleichung (3.36) für das Matrizenelement $m_{i,j}$

$$m_{i,j} = f(\alpha, Ma, q_{dyn}). \quad (3.37)$$

Für den Fall, dass die Abhängigkeit dieser Funktion vom Staudruck quadratischer Art ist, kann Gleichung (3.37) umgeschrieben werden zu

$$m_{i,j} = q_2(\alpha, Ma) \cdot q_{dyn}^2 + q_2(\alpha, Ma) \cdot q_{dyn} + q_3(\alpha, Ma). \quad (3.38)$$

Das heißt, das Matrizenelement wird durch eine quadratische Gleichung beschrieben, deren Parameter q_1 , q_2 und q_3 jeweils nur von zwei Scheduling-Parametern abhängen. Anstatt das Element durch eine dreidimensionale Interpolation zu berechnen, werden die Parameter der Gleichung durch die einfachere zweidimensionale Interpolation bestimmt und die Berücksichtigung des dritten Parameters erfolgt multiplikativ.

Durch die dreidimensionale Rasterung der Envelope und den Reglerentwurf über diesem Raster entsteht für jedes variable Matrizenelement eine dreidimensionale Tabelle. Aus diesen Tabellen können ggf. durch Bildung von Ausgleichsparabeln in Richtung der q_{dyn} -Dimension jeweils drei zweidimensionale Tabellen bestimmt werden, die die Parabelparameter q_1 , q_2 und q_3 enthalten.

3.2.2 Interpolationsverfahren

Zur Bestimmung der Reglermatrizenelemente an Nicht-Entwurfspunkten aus den mehrdimensionalen Tabellen sind verschiedene Verfahren denkbar. Zwei dieser Verfahren werden im Folgenden zunächst für die zweidimensionale Interpolation vorgestellt.

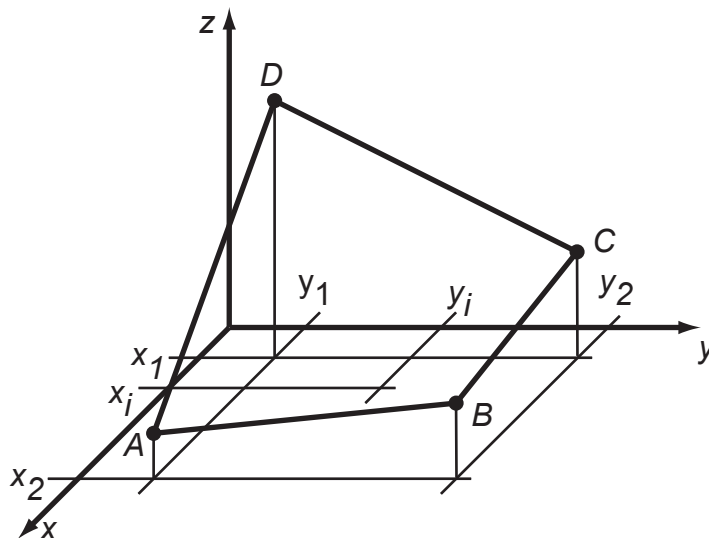


Abb. 3.12: zweidimensionale Interpolation

Gegeben sei ein zweidimensionales, kartesisches Stützstellen-Gitter mit zugehörigen Funktionswerten, das die Stützstellen und Stützwerte für eine 2D-Interpolation liefert. Ausgehend von diesem Gitter soll an der Interpolationsstelle (x_i, y_i) der Funktionswert $z_i = f(x_i, y_i)$ des Interpolationspunktes P_i bestimmt werden, Abbildung 3.12 dient zur Veranschaulichung der Ausgangssituation.

Möglichkeit 1: klassische Methode

Hierbei erfolgt die mehrdimensionale lineare Interpolation durch mehrfache eindimensionale lineare Interpolation. Entsprechend Abbildung 3.13 werden durch eindimensionale Interpolation in y -Richtung bei $x = x_1$ bzw. $x = x_2$ zunächst die Zwischenwerte z_1 bzw. z_2 der Punkte P_1 bzw. P_2 berechnet.

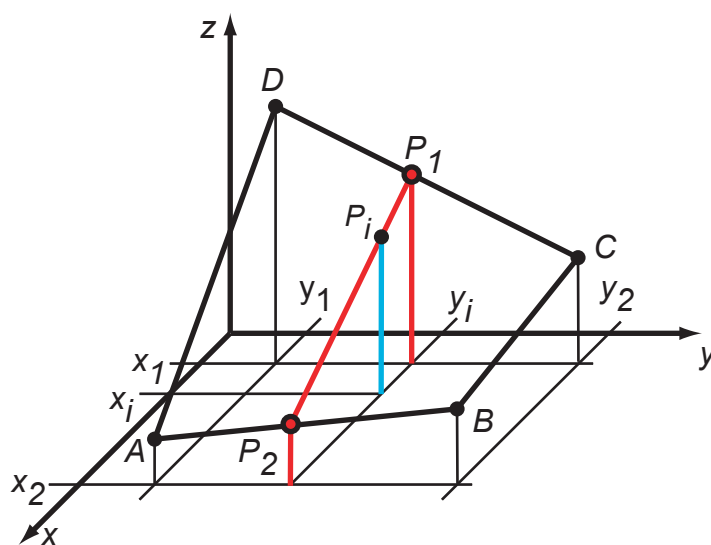


Abb. 3.13: eindimensionale Interpolation in y -Richtung

Anschließend wird durch eine eindimensionale Interpolation in x -Richtung bei $y = y_i$ aus den Werten z_1 und z_2 der interpolierte Funktionswert z_i des Punktes P_i berechnet.

(Analog dazu kann auch zuerst zwei Mal in x -Richtung und anschließend in y -Richtung interpoliert werden.)

Für die Interpolation in y -Richtung bei x_1 gilt

$$z_1 = z_D + (z_C - z_D) \cdot \psi \quad (3.39)$$

mit

$$\psi = \frac{y_i - y_1}{y_2 - y_1} \quad (3.40)$$

und für die Interpolation in y -Richtung bei x_2 gilt

$$z_2 = z_A + (z_B - z_A) \cdot \psi. \quad (3.41)$$

Der z -Wert des Interpolationspunktes P_i berechnet sich nun durch lineare Interpolation in x -Richtung bei $y = y_i$.
Es gilt

$$z_i = z_1 + (z_2 - z_1) \cdot \chi \quad (3.42)$$

mit

$$\chi = \frac{x_i - x_1}{x_2 - x_1}. \quad (3.43)$$

Setzt man in den Ausdruck für z_i die Ausdrücke für z_1 und z_2 ein, folgt nach Vereinfachung der Gleichung folgende Beziehung:

$$z_{i,1} = z_D + \psi \cdot (z_C - z_D) + \chi \cdot (z_A - z_D) + \psi\chi \cdot (-z_A + z_B - z_C + z_D). \quad (3.44)$$

Im zweidimensionalen Fall wird der Funktionswert eines Interpolationspunktes also über Gleichung 3.44 aus den Funktionswerten von vier Stützpunkten A , B , C und D sowie den Abständen des Interpolationspunktes zu den Stützstellen dieser Punkte berechnet. Eine entsprechende Routine ist in Matlab implementiert und steht sowohl als Funktion, z.B. `interp`, als auch als Simulink-Block zur Verfügung.

Möglichkeit 2: Dreiecks-Ebenen-Methode (DEM)

Anstatt über dreimalige eindimensionale Interpolation wie im klassischen Fall kann der Funktionswert des Interpolationspunktes P_i auch entsprechend Abbildung 3.14 über eine Ebeneninterpolation berechnet werden.

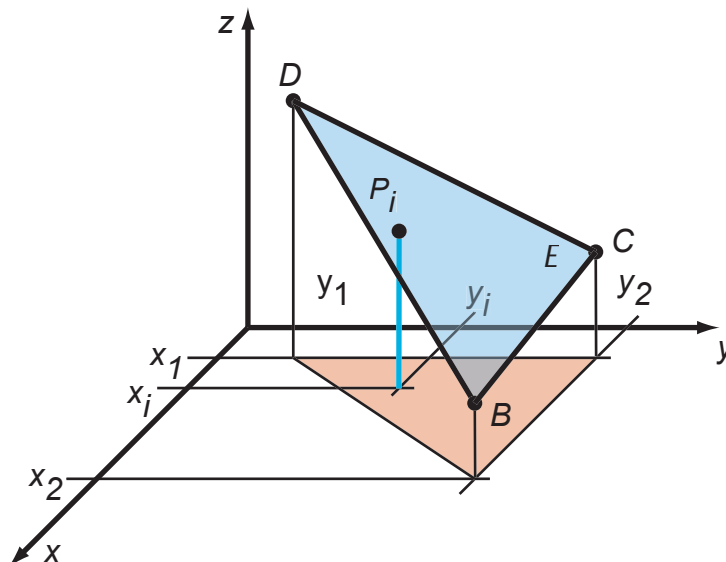


Abb. 3.14: mehrdimensionale Interpolation durch Dreiecks-Ebenen-Methode

Durch drei benachbarte Stützpunkte von P_i wird die Ebene E definiert. Benachbart heißt hierbei, dass die Stützpunkte in der Projektion in die xy -Ebene ein Dreieck bilden, innerhalb dessen die Projektion des Interpolationspunktes liegt.

Unter der Annahme, dass der Interpolationspunkt in der Ebene E liegt, wird über die Ebenengleichung aus den Werten für x_i und y_i der Funktionswert z_i berechnet.

Es gilt

$$\overrightarrow{CB} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_B - z_C \end{bmatrix} = \begin{bmatrix} x_2 - x_1 \\ 0 \\ z_B - z_C \end{bmatrix} \quad (3.45)$$

sowie

$$\overrightarrow{CD} = \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \\ z_D - z_C \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 - y_2 \\ z_D - z_C \end{bmatrix}. \quad (3.46)$$

Hieraus berechnet sich der Normalenvektor der Ebene E gemäß

$$\vec{n} = \overrightarrow{CB} \times \overrightarrow{CD} = \begin{bmatrix} (z_C - z_B) - (y_1 - y_2) \\ (z_C - z_D) - (x_2 - x_1) \\ (x_2 - x_1) - (y_1 - y_2) \end{bmatrix}. \quad (3.47)$$

Damit folgt für die Ebene E

$$E: \vec{n}^T \times \vec{x} = \vec{n}^T \times \vec{c} \quad (3.48)$$

bzw. ausgeschrieben:

$$\begin{bmatrix} (z_C - z_B)(y_1 - y_2) \\ (z_C - z_D)(x_2 - x_1) \\ (x_2 - x_1)(y_1 - y_2) \end{bmatrix}^T \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (z_C - z_B)(y_1 - y_2) \\ (z_C - z_D)(x_2 - x_1) \\ (x_2 - x_1)(y_1 - y_2) \end{bmatrix}^T \cdot \begin{bmatrix} x_1 \\ y_2 \\ z_C \end{bmatrix}. \quad (3.49)$$

Aufgelöst nach z folgt hieraus:

$$z = \frac{z_C - z_B}{x_2 - x_1} \cdot x_1 + \frac{z_C - z_D}{y_1 - y_2} \cdot y_2 + z_C - \frac{z_C - z_B}{x_2 - x_1} \cdot x - \frac{z_C - z_D}{y_1 - y_2} \cdot y. \quad (3.50)$$

An der Interpolationsstelle (x_i, y_i) folgt hieraus nach einigen Umformungen:

$$z_{i,2} = z_D + \chi \cdot (z_B - z_C) + \psi \cdot (z_C - z_D) \quad (3.51)$$

mit χ und ψ aus Gleichung (3.43) bzw. (3.40).

Auch in diesem Fall gehen in die Berechnung des Funktionswerts des Interpolationspunktes also die Abstände des Interpolationspunktes zu den Stützpunkten ein. In diesem Fall werden

jedoch nur drei Stützpunkte verwendet. Entsprechende Matlab-Routinen wurden im Rahmen dieser Arbeit entwickelt und stehen sowohl als Funktionen als auch in Form von Simulink-Blöcken zur Verfügung. Da im Rahmen dieser Arbeit sowohl aus zweidimensionalen als auch aus dreidimensionalen Tabellen interpoliert werden muss, wurden die Funktionen so gestaltet, dass eine dreidimensionale Interpolation möglich ist. Hierzu wird zunächst an zwei Stützwerten der dritten Dimension jeweils in Richtung der beiden anderen Dimensionen mittels der Dreiecks-Ebenen-Methode interpoliert. Aus den beiden Ergebnissen folgt durch eine abschließende 1D-Interpolation in Richtung der dritten Dimension das Ergebnis.

Die Funktionen `TriSel2dp1Tri` und `TriSel2dp1Sel` dienen dabei zur Auswahl der für eine Interpolation zu verwendenden Rasterpunkte. Die Funktion `TriSel2dp1Tri` dient zur Erstellung einer Delaunay-Triangulation aus allen gegebenen Stützstellen, d.h. aus sämtlichen Stützstellen wird ein Netz aus Dreiecken generiert, siehe hierzu auch [15]. Über die Funktion `TriSel2dp1Sel` erfolgt dann die Auswahl des Dreiecks der Stützstellen, das die Interpolationsstelle umschließt. Die eigentliche Ebeneninterpolation erfolgt über die Funktion `TriSel2dp1Int`. Unter Simulink erfolgt die Dreiecksbildung und Stützstellen-selektion über einen Block, der auf der Matlab-s-function `TriSel2dp1` beruht, die Interpolation erfolgt über einen Block, der die s-function `inter_plane_3Dred` aufruft. Die Matlab-Implementierung der Dreiecks-Ebenen-Methode wird im Folgenden kurz als `TriSel`-Methode bezeichnet.

Unterschied zwischen den Methoden

Die klassische Methode erscheint zunächst als die genauere, weil in ihrem Fall eine Stützstelle mehr berücksichtigt wird als im Fall der DE-Methode.

Aus den Gleichungen (3.44) und (3.51) berechnet sich die Differenz zwischen den Interpolationswerten der beiden Methoden zu

$$\Delta = z_{i,1} - z_{i,2} = (z_A - z_B + z_C - z_D) \cdot \chi \cdot (1 - \psi) = \Delta_z \cdot \chi \cdot \tilde{\psi} \quad (3.52)$$

mit

$$\tilde{\psi} = 1 - \psi = \frac{y_i - y_2}{y_2 - y_1}. \quad (3.53)$$

Unter der Voraussetzung eines äquidistanten kartesischen Stützstellengitters mit Gitterweiten Δx und Δy und den Stützpunkten A , B , C und D mit

$$A = \begin{bmatrix} x_A \\ y_A \\ z_A \end{bmatrix} = \begin{bmatrix} x_2 \\ y_1 \\ f(x_2, y_1) \end{bmatrix}, B = \begin{bmatrix} x_2 \\ y_2 \\ f(x_2, y_2) \end{bmatrix}, C = \begin{bmatrix} x_1 \\ y_2 \\ f(x_1, y_2) \end{bmatrix}, D = \begin{bmatrix} x_1 \\ y_1 \\ f(x_1, y_1) \end{bmatrix} \quad (3.54)$$

gilt für beliebige, nur aus Monomen bestehende Funktionen $f(x,y)$

$$\Delta = 0, \quad (3.55)$$

denn für den Spezialfall, dass f nur aus Monomen besteht, können die von x und von y abhängigen Anteile in f voneinander getrennt werden:

$$f(x, y) = f_1(x) + f_2(y). \quad (3.56)$$

Für die Punkte A , B , C und D folgt damit

$$\begin{aligned} z_A &= f(x_2, y_1) = f_1(x_2) + f_2(y_1) \\ z_B &= f(x_2, y_2) = f_1(x_2) + f_2(y_2) \\ z_C &= f(x_1, y_2) = f_1(x_1) + f_2(y_2) \\ z_D &= f(x_1, y_1) = f_1(x_1) + f_2(y_1) \end{aligned} \quad (3.57)$$

Setzt man (3.57) in (3.52) ein, folgt

$$\Delta_z = f_1(x_2) + f_2(y_1) - f_1(x_2) - f_2(y_2) + f_1(x_1) + f_2(y_2) - f_1(x_1) - f_2(y_1) = 0. \quad (3.58)$$

Für den beschriebenen Spezialfall liefern beide Interpolationsverfahren also identische Resultate.

Für den allgemeinen Fall einer beliebigen Funktion $f(x, y)$ lässt sich der Unterschied Δ durch die Taylor-Reihenentwicklung abschätzen.

Es gilt

$$\begin{aligned} f(x_1 + \Delta x, y_1 + \Delta y) &= f(x_1, y_1) + \left. \frac{\partial f}{\partial x} \right|_1 \Delta x + \left. \frac{\partial f}{\partial y} \right|_1 \Delta y + \\ &\frac{1}{2} \cdot \left(\left. \frac{\partial^2 f}{\partial x \partial x} \right|_1 \Delta x^2 + 2 \cdot \left. \frac{\partial^2 f}{\partial x \partial y} \right|_1 \Delta x \Delta y + \left. \frac{\partial^2 f}{\partial y \partial y} \right|_1 \Delta y^2 \right) + R_2 \end{aligned} \quad (3.59)$$

bzw. in verkürzter Schreibweise

$$f(x_1 + \Delta x, y_1 + \Delta y) = f_1 + f_x \Delta x + f_y \Delta y + \frac{1}{2} \cdot (f_{xx} \Delta x^2 + 2f_{xy} \Delta x \Delta y + f_{yy} \Delta y^2) + R_2. \quad (3.60)$$

Für die Punkte A , B , C und D gilt somit

$$\begin{aligned} z_A &= f_1 + f_x \Delta x + 1/2 \cdot f_{xx} \Delta x^2 + R_{2,A} \\ z_B &= f_1 + f_x \Delta x + f_y \Delta y + 1/2 \cdot f_{xx} \Delta x^2 + f_{xy} \Delta x \Delta y + 1/2 \cdot f_{yy} \Delta y^2 + R_{2,B} \\ z_C &= f_1 + f_y \Delta y + 1/2 \cdot f_{yy} \Delta y^2 + R_{2,C} \\ z_D &= f_1 \end{aligned} \quad (3.61)$$

Für die Differenz zwischen den beiden Methoden folgt somit gemäß (3.52)

$$\Delta = (f_{xy} \Delta x \Delta y + R_{2,A} - R_{2,B} + R_{2,C}) \cdot \chi \cdot \tilde{\psi}. \quad (3.62)$$

Hierbei gilt für die Restglieder 2. Ordnung (in Operatorschreibweise)

$$R_2 = \frac{1}{6} \cdot \left(\frac{\partial f}{\partial x} \cdot \Delta x + \frac{\partial f}{\partial y} \cdot \Delta y \right)^3 \cdot f(x_1 + \theta \Delta x, y_1 + \theta \Delta y), \text{ für } 0 < \theta < 1. \quad (3.63)$$

Das heißt, es handelt sich bei den Restgliedern um Terme 3. Ordnung. Die gemischte partielle Ableitung $f_{xy} \Delta x \Delta y$ ist ein Term 2. Ordnung, χ und ψ nehmen jeweils Werte an zwischen 0 und 1, sind also erster Ordnung. Somit ergibt sich, dass die Abweichung Δ insgesamt in der Größenordnung des Quadrates der Gitterweite liegt.

Zur Veranschaulichung wird folgende Funktion betrachtet:

$$f(x, y) = x^2 - y^2 + xy^2 + 150. \quad (3.64)$$

Diese Funktion wird in einem Bereich $-5 \leq x \leq 5$ und $-5 \leq y \leq 5$ an 500 zufällig ausgewählten Interpolationspunkten ausgewertet. Das zu Grunde liegende Gitter hat eine Weite von $\Delta x = \Delta y = 0,2$.

Es ergibt sich eine mittlere Abweichung zwischen klassischer und Dreiecks-Ebenen-Methode von etwa 0,02. Das Quadrat der Gitterweite beträgt 0,04 und wie erwartet haben beide Werte die gleiche Größenordnung.

Für hinreichend feine Rasterweiten ist der zu erwartende Genauigkeitsverlust aufgrund des Informationsverlusts im Falle der DE-Methode also tolerierbar klein.

Geschwindigkeitsvergleich

Die beiden Verfahren zur Bestimmung der Interpolationsstützpunkte führen zu unterschiedlichen Berechnungsgeschwindigkeiten. Zu Testzwecken wird die Funktion

$$f(x, y) = x^2 + y^2 + 3x \quad (3.65)$$

für verschiedene Anzahlen zufällig verteilter Punkte für verschiedene Gitterweiten mit beiden Interpolationsmethoden ausgewertet, um anschließend die Berechnungszeiten zu vergleichen. Die Funktion wird ausgewertet für $0,1 \leq x \leq 5$ und $0,1 \leq y \leq 5$, das Gitter wird spezifiziert durch die Zahl der Stützstellen innerhalb der Grenzen des x - und y -Wertebereichs.

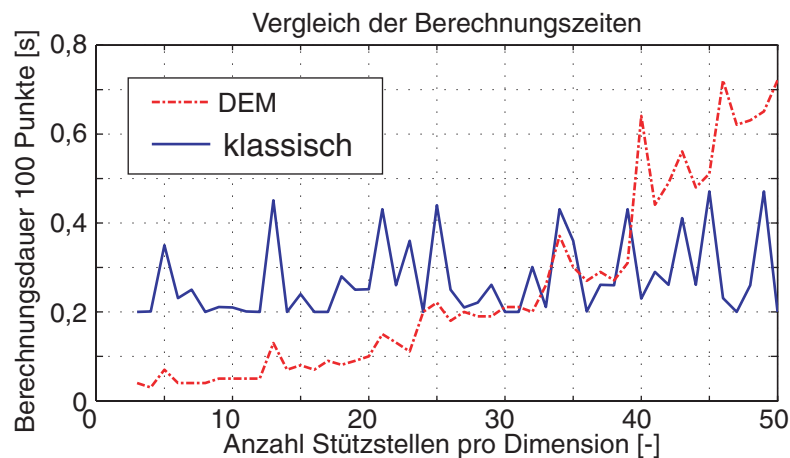


Abb. 3.15: Verlauf der Berechnungszeiten für klassische und DE-Methode

Für Gitter mit einer Stützstellenzahl pro Koordinatenrichtung zwischen 3 und 50 und einer Auswertung an jeweils 100 Punkten ergibt sich der in Abbildung 3.15 dargestellte Verlauf der Berechnungszeiten. Es zeigt sich hierbei: Bis ca. 30 Stützstellen pro Dimension ist die Ebenen-Methode TriSel schneller als die klassische Methode.

Auswahl einer Methode

Mit der dreiecksbasierten Ebeneninterpolation lassen sich Stützstellenbereiche mit gekrümmten Begrenzungen besser abdecken als mit der klassischen Interpolationsmethode.

Im Fall der klassischen Methode sind an jedem Interpolationspunkt vier benachbarte Stützpunkte nötig, deshalb setzt sich der Bereich, in dem die Interpolation möglich ist, aus rechteckigen Kacheln zusammen.

Im Gegensatz dazu werden im Falle der Ebeneninterpolation an jedem Interpolationspunkt nur die drei Punkte benötigt, die ein Dreieck bilden, in dem der Interpolationspunkt liegt.

Bei einem gegebenen Stützstellenraster und einem Interpolationsbereich mit gekrümmter Begrenzung ist dabei mit dreieckigen Kacheln eine bessere Überdeckung des Interpolationsbereichs zu erreichen als mit rechteckigen.

Im Fall der Envelope des Jettrainers Mako treten solche Interpolationsbereiche mit gekrümmten Begrenzungen auf, denn die Begrenzung erfolgt hier durch mehrere, miteinander gekoppelte Variablen. So entsteht beispielsweise durch die Vorgabe von zulässigen Machzahlen und Staudrücken zunächst ein rechteckiger Bereich in der $Ma-q_{dyn}$ -Ebene. Durch die zusätzliche Vorgabe einer Höhenbegrenzung werden jedoch bestimmte Machzahl-Staudruck-Kombinationen unzulässig. Sowohl die Schallgeschwindigkeit als auch die Luftdichte sind von der Flughöhe abhängig. Dementsprechend existiert eine Kopplung zwischen der Machzahl, dem Staudruck sowie der Flughöhe und zu jeder Kombination von Staudruck und Machzahl gehört eine eindeutig zuordenbare Flughöhe, die ggf. außerhalb der angegebenen Begrenzung liegt. Sämtliche Machzahl-Staudruck-Kombinationen, die zu einer bestimmten Höhe gehören, erscheinen im $Ma-h$ -Diagramm als gekrümmte Linie, deren Koordinaten über das Atmosphärenmodell berechnet werden. Da für alle Anstellwinkel die gleichen Machzahl-, Staudruck-, Höhen- und Geschwindigkeitsbegrenzungen vorliegen, entsteht als dreidimensionale Envelope ein Prisma mit einer Grundfläche in Form der Envelope in ihrer $Ma-h$ - bzw. $Ma-q_{dyn}$ -Darstellung entsprechend den Abbildungen 2.2 bzw. 2.3 aus Abschnitt 2.1.5. Für die mehrdimensionale Interpolation innerhalb dieses Gebiets müssen nun im Fall der klassischen Methode aus dem Stützstellenraster Prismen mit rechteckiger Grundfläche, also Quader gebildet werden. Für die Bestimmung des Funktionswertes an einem Interpolationspunkt muss dann der Quader verwendet werden, der den Interpolationspunkt umrahmt. Im Fall der TriSel-Methode müssen dagegen Prismen mit dreieckiger Grundfläche gebildet werden, wobei zur Interpolation wiederum das Prisma verwendet wird, das den Interpolationspunkt umrahmt. Dabei ist eine Fläche wie die Envelope des Jettrainers für einen bestimmten Anstellwinkel in $Ma-q_{dyn}$ -Darstellung (d.h. ein Schnitt durch die dreidimensionale Envelope bei einem bestimmten Anstellwinkel) wesentlich besser durch Dreiecke abzudecken als durch Rechtecke. Abbildung 3.16 zeigt diese Abdeckung für eine Rasterung der Machzahl und des Staudrucks gemäß

$$Ma = [0,1 \ 0,2 \ 0,3 \ 0,4 \ 0,5] \quad (3.66)$$

und

$$q_{dyn} = [500 \ 703 \ 780 \ 1217 \ 1890 \ 2964 \ 4625 \ 7216 \ 11261 \ 17571], \quad (3.67)$$

die Einheit des Staudrucks beträgt dabei [Pa].

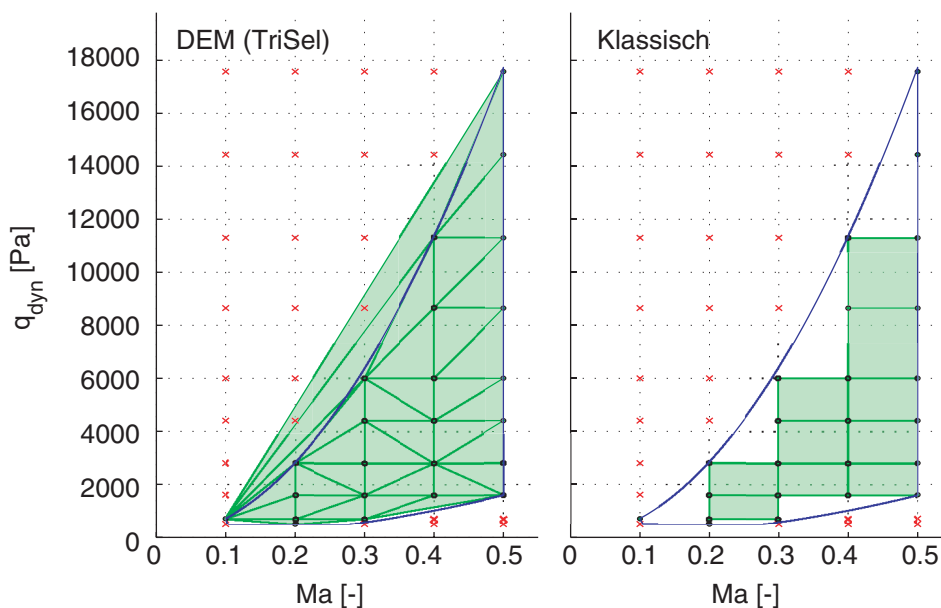


Abb. 3.16: Vergleich der Interpolationsarten: einsetzbarer Bereich

Die blaue Linie kennzeichnet die Envelope. Die Kreise markieren die Punkte des Rasters, die innerhalb der Envelope liegen. Für sie kann ein Modell linearisiert werden und ein Reglerentwurf erfolgen. Sie dienen daher als Stützpunkte für die Interpolation. Die Kreuze sind Punkte, die außerhalb der Envelope liegen und für die daher auch kein Reglerentwurf erfolgen kann. Sie können nicht als Stützpunkte für die Interpolation verwendet werden. Es zeigt sich, dass im Fall der TriSel-Methode sogar bei Punkten außerhalb der Envelope eine Interpolation möglich ist, im Fall der klassischen Methode deckt der Bereich, in dem eine Interpolation möglich ist bei gleicher Rasterung die Envelope dagegen nur lückenhaft ab.

Im Rahmen der vorliegenden Arbeit wird daher für die mehrdimensionale Interpolation die DE-Methode verwendet. Für die gegebenen Anzahlen an Stützpunkten (siehe Abschnitt 3.2.3) ist die Methode ausreichend schnell und der Informationsverlust aufgrund der Berücksichtigung von einer Stützstelle weniger als im Fall der klassischen Interpolationsmethode ist tolerierbar.

3.2.3 Diskretisierung der Envelope zur Festlegung des Reglerentwurfsrasters

Bei der Diskretisierung der Envelope zur Festlegung des Reglerentwurfsrasters ist einerseits zu beachten, dass die Envelope fein genug aufgelöst wird, sodass im gesamten Einsatzbereich des Jettrainers die Stabilität des Regelkreises garantiert ist. Andererseits bedeutet eine

fein aufgelöste Envelope auch eine hohe Anzahl an Reglerentwurfspunkten und damit einen hohen Berechnungsaufwand beim Entwurf aller Einzelregler. Ziel ist es also, die Envelope gerade fein genug bzw. möglichst grob aufzulösen. Weiter ist zu beachten, dass die Einteilung des Staudruck- und Machzahlbereichs zur Generierung des Entwurfspunkterasters so erfolgt, dass möglichst viele Entwurfspunkte innerhalb der Envelope liegen. Wie Abbildung 3.17 zeigt, bedeutet jeder Entwurfspunkt, der außerhalb der Envelope liegt, für den interpolierbaren Bereich der Envelope den Verlust eines Rechtecks (im Fall der klassischen Interpolation) bzw. eines Dreiecks (im Fall der DE-Methode). Im Vergleich zu Fall A ist das Raster im Fall B nur leicht verschoben, durch einen Punkt weniger in der Envelope geht jedoch ein Dreieck des interpolierbaren Bereichs verloren.

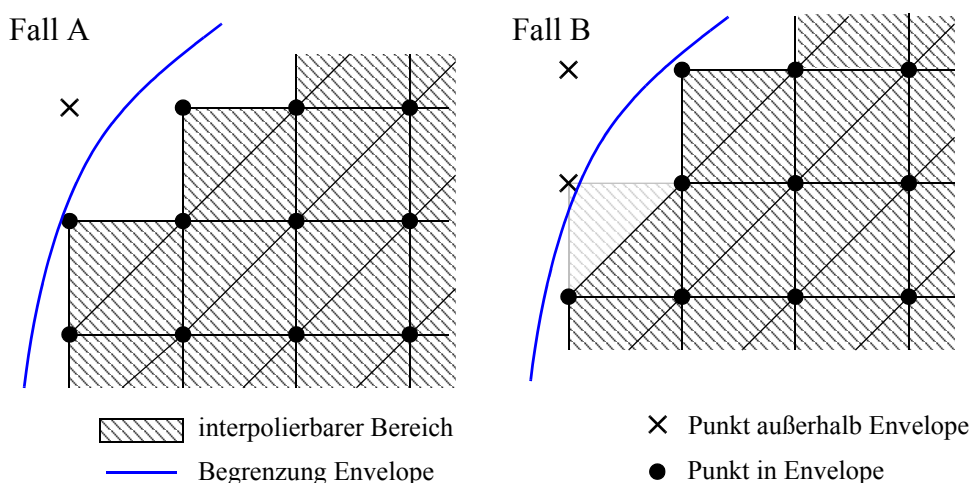


Abb. 3.17: interpolierbarer Bereich in Abhängigkeit der Rasterung

Bei Tests mit verschiedenen Diskretisierungen hat sich gezeigt, dass folgende Einteilung des Machzahl- und Staudruckbereichs zu guten Ergebnissen führt (siehe hierzu auch die Auswertung in Abschnitt 5.5.2):

Die Machzahl wird zwischen $Ma_{min} = 0,1$ und $Ma_{max} = 0,5$ in gleich große Intervalle der Breite 0,1 eingeteilt. Für das Machzahlraster folgt:

$$Ma_R = [0,1 \ 0,2 \ 0,3 \ 0,4 \ 0,5] \tag{3.68}$$

Der Staudruck wird zwischen $q_{dyn,min} = 500\text{Pa}$ und einem maximalen Staudruck von $q_{dyn,max} = 17571\text{Pa}$ eingeteilt. Dieser maximale Staudruck weicht von der oberen Staudruckgrenze des Aerodynamikmodells ab (18000 Pa), denn selbst für die Machzahl $Ma = 0,5$ ist ein Staudruck von $q_{dyn} = 18000\text{Pa}$ nur für negative Höhen erfliegbar, d.h. die betreffende Machzahl-Staudruck-Kombination liegt außerhalb der Envelope. Mit einem Staudruck von 17571Pa ergibt sich bei einer Machzahl von 0,5 jedoch einerseits eine positive und damit zulässige Flughöhe und andererseits eine zugehörige Fluggeschwindigkeit, die ebenfalls gerade innerhalb des zulässigen Geschwindigkeitsbereichs liegt. Der betreffende Punkt liegt also gerade noch innerhalb der Envelope.

Die Einteilung zwischen den Staudruckgrenzen erfolgt in erster Näherung logarithmisch in acht Intervalle. Anschließend werden die Staudruckwerte so verschoben bzw. ergänzt, dass folgende Unterteilung in neun Intervalle entsteht:

$$q_{dyn,R} = [500 \ 703 \ 780 \ 1217 \ 1890 \ 2964 \ 4625 \ 7216 \ 11261 \ 17571]. \quad (3.69)$$

Durch die Verschiebung wird erreicht, dass eine größere Zahl von Entwurfspunkten innerhalb der Envelope liegt und sich somit eine bessere Abdeckung ergibt als bei einer streng logarithmischen Teilung.

Es handelt sich bei dieser Machzahl-Staudruck-Kombination um die gleiche, die im vorangehenden Kapitel in Abbildung 3.16 dargestellt wurde.

Eine Möglichkeit zur Auswahl von Rasterungen hinsichtlich ihrer Envelope-Abdeckung bietet das Matlab-Skript `GridChecker`, über das die Abdeckung einer frei wählbaren Rasterung dargestellt wird.

Die Einteilung des Anstellwinkelbereichs erfolgt zwischen $\alpha_{min} = -9,5^\circ$ und $\alpha_{max} = 44,5^\circ$ in Schritten von einem Grad.

Es ergibt sich hieraus die in Abbildung 3.18 dargestellte dreidimensionale Rasterung der Envelope. Pro Anstellwinkel ergeben sich $5 \times 10 = 50$ Rasterpunkte, von denen 24 innerhalb der Envelope liegen und somit für den Reglerentwurf verwendet werden können. Bei insgesamt 55 Anstellwinkeln ergibt sich so eine Summe von 1320 Entwurfspunkten. Zu Gunsten der Übersichtlichkeit, wurden außerhalb der Envelope liegende Rasterpunkte in Abbildung 3.18 nicht eingezeichnet.

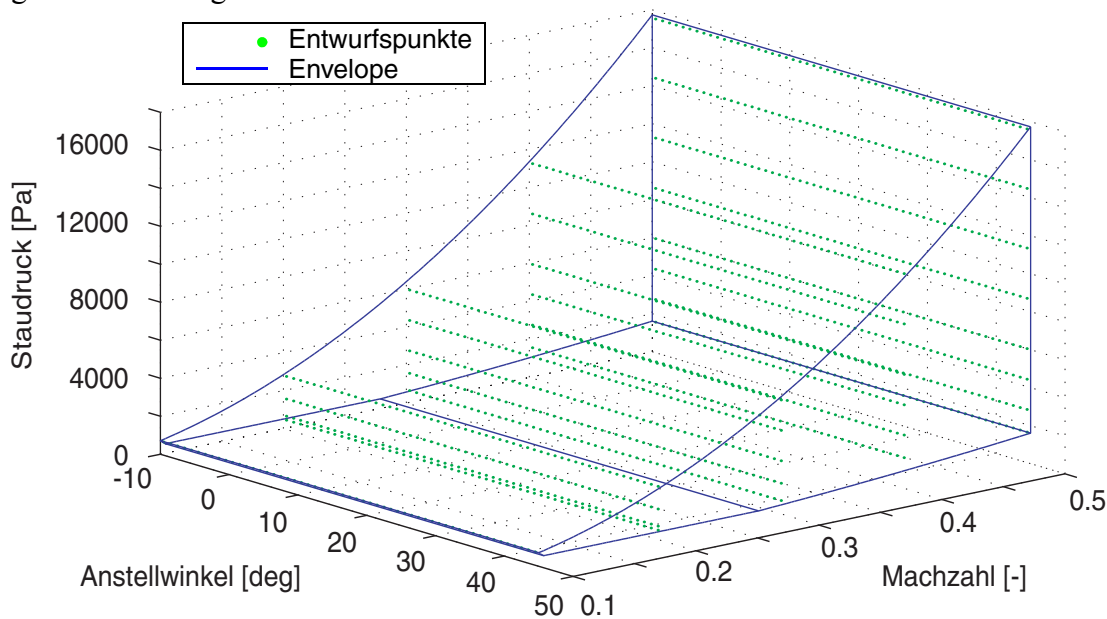


Abb. 3.18: Dreidimensionales Entwurfsraster

Kapitel 4

Der automatisierte Reglerentwurf

Durch die im vorangegangenen Kapitel beschriebene Rasterung der Envelope des Jettrainers zur Festlegung von Reglerentwurfspunkten als Basis für einen Gain-Scheduling-Regler entsteht eine große Anzahl zu entwerfender Regler. Obwohl bei jedem Einzelregler nur drei Parameter festzulegen sind (siehe Abschnitt 3.1.3) und obwohl bei der Abhängigkeit der Entwurfsziele aus Abschnitt 3.1.2 von diesen Parametern Tendenzen erkennbar sind, ist eine Automatisierung des Reglerentwurfsprozesses unumgänglich.

Ziel dieser Arbeit ist der Entwurf eines möglichst robusten Flugreglers für den Jettrainer Mako. Dementsprechend basiert das verwendete Reglerentwurfsverfahren auf der Optimierung der Robustheit des Regelkreises gegenüber Störungen in der coprime-faktorierten Regelstrecke. Ausgehend von drei Parametern, die die Wichtungsfunktionen W_1 und W_2 so definieren, dass der gewichtete Regelkreis die gestellten Handling-Quality-Forderungen erfüllt, liefert das H_∞ -Loop-Shaping Verfahren einen Regler, der den Regelkreis optimal robust stabilisiert. Die erreichte Robustheit, die über die Robustheitsreserve ρ nach Abschnitt 3.1.2 quantifizierbar ist, hängt dabei von den konkreten Zahlenwerten der Parameter der Wichtungsfunktionen ab. Da es unendlich viele Kombinationen von Wichtungsfunktions-Parametern gibt, für die die Handling-Quality-Forderungen des Reglers erfüllt werden, gilt es nun die Kombination zu finden, für die der Regelkreis die größtmögliche Robustheitsreserve aufweist. Der naheliegende Schluss ist eine Bestimmung der Wichtungsfunktions-Parameter mittels eines nichtlinearen, beschränkten Optimierungsverfahrens. Der automatisierte Aufruf eines solchen Verfahrens für jeden Reglerentwurfspunkt ermöglicht somit die selbständige Abarbeitung aller Entwurfspunkte

4.1 Prinzip der Regleroptimierung

Wie in der Einführung dieses Kapitels beschrieben wurde, erfolgt die Bestimmung der drei Wichtungsfunktions-Parameter mittels nichtlinearer, beschränkter Optimierung. Im vorliegenden Fall stehen hierzu zwei Optimierungsroutinen zur Verfügung, zum einen die Matlab-Routine `fmincon` aus der Optimisation Toolbox, zum anderen die IFR-Routine `iffrslsqp`. Beide Routinen basieren auf dem Optimierungsverfahren der Sequentiell Quadratischen Programmierung (SQP). In beiden Fällen wird eine skalare Zielfunktion $f(\vec{p})$ definiert, deren Funktionswert von den zu bestimmenden Optimierungsparametern \vec{p}

abhängt. Durch das Optimierungsverfahren wird nun ausgehend von Startwerten \vec{p}_{start} der Parametervektor \vec{p} so bestimmt, dass der Funktionswert von f unter Einhaltung gegebener Nebenbedingungen minimiert wird. Die Nebenbedingungen sind dazu ebenfalls in Form von Funktionen in Abhängigkeit der Optimierungsparameter anzugeben. Man unterscheidet hierbei zwischen Gleichungsnebenbedingungen der Form $g(\vec{p}) = 0$ und Ungleichungsnebenbedingungen der Form $g(\vec{p}) > 0$. Für weiterführende Informationen zum Thema der nichtlinearen Optimierung sei auf die Literatur verwiesen, wie beispielsweise [16] oder auch die Dokumentation der Matlab-Routine `fmincon` in [17].

4.1.1 Berechnung der Zielgröße

Als Zielgröße für die Optimierung dient das unter den Entwurfszielen in Abschnitt 3.1.2 eingeführte Robusteitsmaß ρ . Es handelt sich dabei um den Faktor, um den der Nichols-Diamant gestreckt werden kann bzw. gestaucht werden muss, bis die Ortskurve des offenen Regelkreises den Diamanten gerade berührt.

Zur Bestimmung des Berührungspunktes und des zugehörigen Skalierungsfaktors werden zunächst für diskrete Frequenzen Punkte der Ortskurve, also Phase und Betrag der Übertragungsfunktion des offenen Regelkreises gebildet. Für jeden dieser Punkte wird dann der Faktor ermittelt, mit dem der Nichols-Diamant skaliert werden muss, damit er den Punkt gerade berührt. Als Berührungspunkt der Ortskurve folgt aus allen untersuchten Punkten derjenige mit dem kleinsten zugehörigen Skalierungsfaktor. Dieser ist wiederum gleich dem Robusteitsmaß ρ . (Anmerkung: Theoretisch müssen zu einer exakten Bestimmung des Robusteitsmaßes kontinuierlich alle Punkte der Ortskurve also alle Frequenzen zwischen 0 und ∞ untersucht werden. Dies ist jedoch technisch nicht machbar, weshalb eine Diskretisierung des Frequenzbereiches erfolgt.)

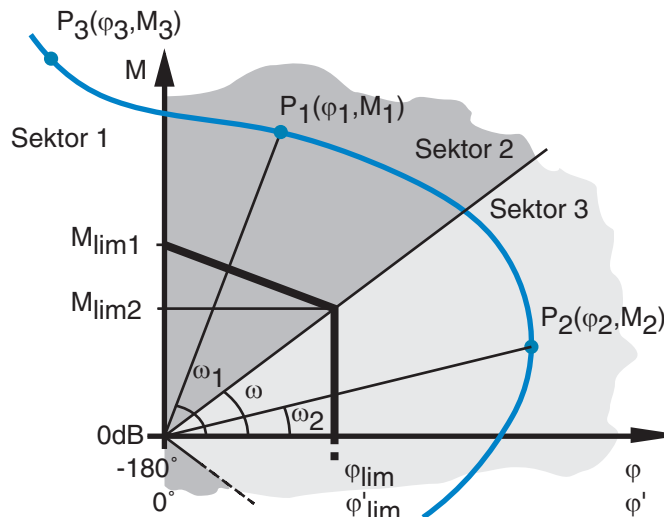


Abb. 4.1: Sektoren des Nichols-Diagramms

Der Skalierungsfaktor eines einzelnen Punktes der Ortskurve hängt davon ab, in welchem Bereich der Umrandung des skalierten Nichols-Diamanten er diesen berührt. Wie Abbildung 4.1 zeigt, wird die Begrenzung des Nichols-Diamanten durch seinen linken Rand bei der

Phase des kritischen Punktes von $\varphi = -180^\circ$, seinen rechten Rand bei der Phase $\varphi = \varphi_{lim}$ sowie durch die schrägen Linien über die Beträge M_{lim1} und M_{lim2} definiert. (Anmerkung: in Abbildung 4.1 ist nur die obere Hälfte des Nichols-Diamanten dargestellt.)

Dementsprechend wird die Betrags-Phasen-Ebene des Nichols-Diagramms durch die beiden vertikalen sowie die beiden schrägen Begrenzungen des Nichols-Diamanten in drei Sektoren eingeteilt. Für jeden Sektor gelten unterschiedliche Vorschriften zur Berechnung des Skalierungsfaktors. Deshalb muss zur Bestimmung des Skalierungsfaktors eines bestimmten Punktes P , gegeben durch seine Phase φ_P und seinen Betrag M_P , zunächst bestimmt werden, in welchem Sektor er sich befindet. Hierzu wird die Koordinatentransformation

$$\varphi' = \varphi + 180^\circ \quad (4.1)$$

durchgeführt. Somit lassen sich alle Phasenwinkel problemlos auf den kritischen Punkt beziehen, der in transformierten Koordinaten die Phase $\varphi' = 0^\circ$ hat. Ein Punkt $P(\varphi'_P, M_P)$ der Ortskurve liegt dann in

- Sektor 1, falls gilt:

$$\varphi'_P \leq 0^\circ \quad (4.2)$$

- Sektor 2, falls gilt:

$$\varphi'_P > 0 \quad (4.3)$$

und

$$\tan(\omega_P) = \frac{M_P}{\varphi'_P} > \frac{M_{lim2}}{\varphi'_{lim}} = \tan(\omega) \quad (4.4)$$

- Sektor 3, falls gilt:

$$\tan(\omega_P) = \frac{M_P}{\varphi'_P} \leq \frac{M_{lim2}}{\varphi'_{lim}} = \tan(\omega). \quad (4.5)$$

Für den Fall, dass der untersuchte Punkt in Sektor 3 liegt und den skalierten Nichols-Diamanten also an seiner rechten vertikalen Beschränkung berührt, berechnet sich der Skalierungsfaktor bzw. das Robustheitsmaß gemäß der Formel

$$\rho = \frac{\varphi'_P}{\varphi'_{lim}}. \quad (4.6)$$

Berührt die Ortskurve den Diamanten dagegen an einer seiner schrägen Begrenzungslinien, d.h. liegt der Berührungspunkt in Sektor 2, berechnet sich ρ nach der Formel

$$\rho = M_{lim1}^{-1} \cdot \sqrt{\varphi'_P{}^2 + M_P^2} \cdot \frac{\sin\beta}{\sin\gamma} \quad (4.7)$$

mit

$$\gamma = \frac{\pi}{2} - \operatorname{atan}\left(\frac{M_{lim1} - M_{lim2}}{\Phi'_{lim}}\right) = \frac{\pi}{2} - \eta \quad (4.8)$$

und

$$\beta = \omega + \eta \quad (4.9)$$

wobei gilt

$$\omega = \operatorname{atan}\frac{\Phi'_P}{M_P}. \quad (4.10)$$

Liegt der momentan betrachtete Punkt der Ortskurve dagegen in Sektor 1, ist der Skalierungsfaktor nicht definiert und wird pauschal auf den Wert 1000 gesetzt. Dies ist zulässig, denn falls die Ortskurve den kritischen Punkt überhaupt umrundet, dann folgen weitere Ortskurvenpunkte in den Sektoren 2 und 3, deren Skalierungsfaktor kleiner ist als 1000.

Durch die Diskretisierung des Frequenzbereichs handelt es sich bei dem ermittelten Berührungspunkt und dem zugehörigen Stabilitätsmaß um eine Näherungslösung. Die Genauigkeit dieser Lösung wird gesteigert, indem mehrere geschachtelte Durchläufe der Berührungspunktbestimmung erfolgen. Zunächst erfolgt eine Diskretisierung des Frequenzbereichs zwischen $\omega_1 = 10^{-4}\text{rad/s}$ und $\omega_2 = 10^4\text{rad/s}$ in Z_1 Schritten bei logarithmischer Teilung, es gilt damit

$$\omega_i = 10^{p(i)}, \quad i = 1 \dots Z_1, \quad (4.11)$$

wobei für den Exponenten p folgende lineare Beziehung gilt:

$$p(i) = p_1 + \frac{p_2 - p_1}{Z_1 - 1} \cdot (i - 1) \quad (4.12)$$

mit

$$p_1 = \log(\omega_1) \quad (4.13)$$

und

$$p_2 = \log(\omega_2). \quad (4.14)$$

Anschließend wird die Frequenz ω_{imin} des am nächsten liegenden Punktes bestimmt. Zwischen den beiden benachbarten Frequenzen ω_{imin-1} und ω_{imin+1} erfolgt dann eine erneute Diskretisierung des Frequenzbereichs mit Z_2 Schritten. Diese Verfeinerung des Frequenzbereichs erfolgt so lange, bis sich zwei nacheinander berechnete Werte des Abstands des Punktes mit dem kleinsten Abstand nicht mehr voneinander unterscheiden. Die Anzahl Z der Schritte der Diskretisierung des Frequenzbereichs wird im vorliegenden Fall bei jedem Berechnungsdurchgang reduziert. Für die Zahl Z in Abhängigkeit des Durchgangs d gilt die Beziehung

$$Z_d = Z_1 \cdot \exp(-r \cdot (d - 1)), \quad (4.15)$$

das heißt die Anzahl der Diskretisierungsschritte wird mit einer Rate $r > 0$ exponentiell reduziert. Hierdurch wird erreicht, dass die Anzahl der insgesamt betrachteten Punkte zur Bestimmung des Punktes mit dem geringsten Abstand reduziert wird. Es ist hierbei allerdings zu beachten, dass pro Berechnung mindestens vier Punkte betrachtet werden, andernfalls kann es bei der Bildung des Diskretisierungsintervalls aus den beiden Nachbarfrequenzen zu keiner Verkleinerung des Intervalls kommen, es muss also in jedem Fall gelten $Z_d \geq 4$.

Zur Veranschaulichung ist in Abbildung 4.2 für einen exemplarischen Regelkreis die Zahl der insgesamt zu betrachtenden Punkte über der Rate r dargestellt, ausgehend von einer Anzahl $Z_1 = 750$.

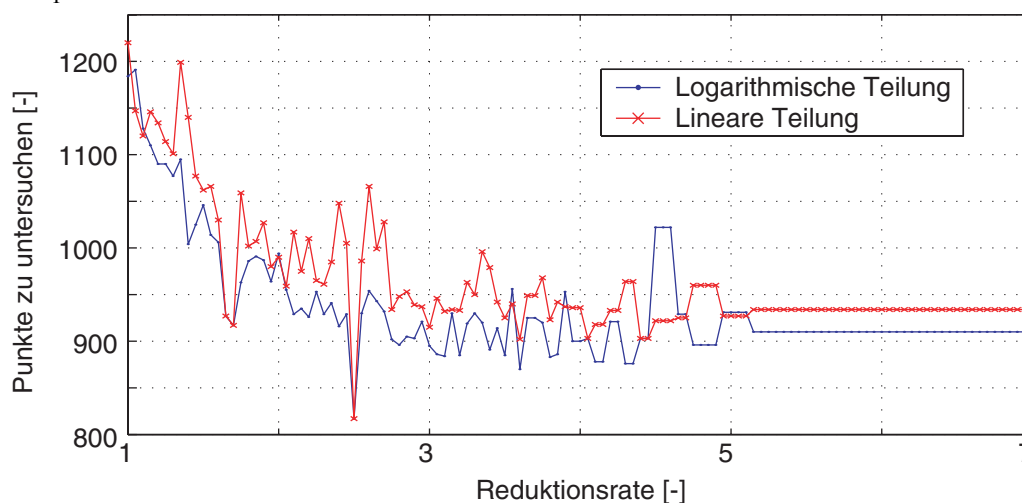


Abb. 4.2: zu betrachtende Punkte zur Bestimmung des Berührungspunktes in Abhängigkeit der Rate r zur Reduktion der Zahl der betrachteten Punkte pro Durchlauf

Es zeigt sich, dass ab einer Rate von etwa $r = 5,5$ für steigende r keine Veränderung mehr auftritt. Im Fall einer logarithmischen Teilung müssen dabei insgesamt weniger Punkte ausgewertet werden als im Fall einer linearen Teilung. Ob eine logarithmische oder eine lineare Teilung des Frequenzbereichs schneller ist, hängt davon ab, bei welcher Frequenz der minimale Abstand zum Nichols-Diamanten vorliegt. Je kleiner diese Frequenz ist, desto näher liegen bei einer logarithmischen Teilung die einzelnen Frequenzstützwerte beieinander und desto schneller wird das betrachtete Frequenzintervall klein genug, dass sich die Abstandswerte zweier aufeinander folgender Berechnungen nicht mehr unterscheiden. Beim automatischen Reglerentwurf liegt die Frequenz des Punktes des geringsten Abstands im Allgemeinen in einem Frequenzbereich, in dem eine logarithmische Teilung schneller zur Bestimmung des Abstands des Punktes führt. Die beschriebene Berechnungsart bietet eine definitionsgemäße Bestimmung des Robustheitsmaßes. Allerdings hat sie einen entscheidenden Nachteil: Betrachtet man in Abbildung 4.3 das Robustheitsmaß für einen Verlauf des Berührungspunktes auf einem Halbkreis um den Nichols-Diamanten mit Ursprung im kritischen Punkt, so stellt man fest, dass es an den Sektorengrenzen zu Knicken im Verlauf des Robustheitsmaßes über dem Kreiswinkel kommt. Das bedeutet, wenn sich die Ortskurve für sich verändernde Optimierungsparameter verschiebt und der Punkt des minimalen Abstands deshalb von einem Sektor in den anderen wandert, dann ändert sich die Ableitung der Änderung (d.h. die Vergrößerung oder Verkleinerung) des minimalen Abstands beim Übergang vom

einen in den anderen Sektor sprunghaft. Dies kann zu Problemen bei der Optimierung führen, weshalb alternativ zur definitionsgemäßen Berechnung von ρ eine weitere Berechnungsmethode eingeführt wird. Anstelle eines Diamants zur Definition des zu meidenden Gebietes um den kritischen Punkt wird eine Ellipse eingeführt, die den Nichols-Diamanten umschließt.

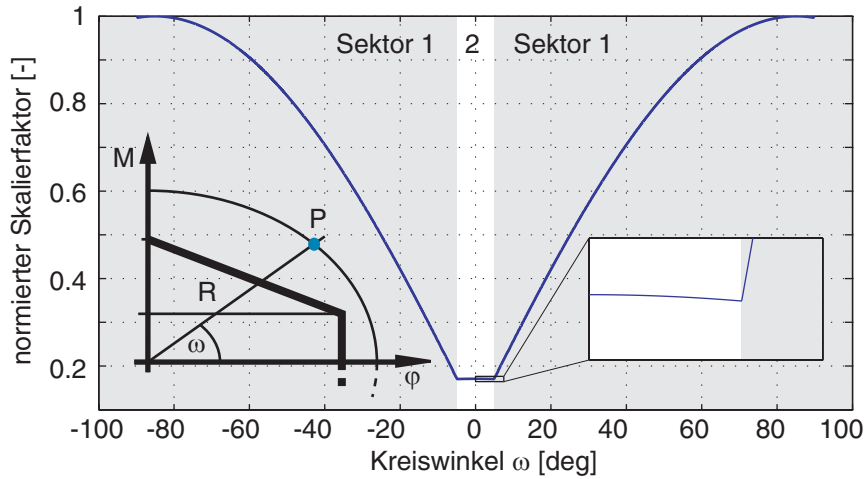


Abb. 4.3: Berührungspunkt auf Kreisbahn - Knicke im Verlauf des Robustheitsmaßes

Dadurch wird der zu meidende Bereich vergrößert, d.h. die Maßnahme ist konservativ, aber wie Abbildung 4.4 zeigt, gibt es keine unterschiedlichen Sektoren. Für einen Berührungspunkt P , der den kritischen Punkt auf einem Halbkreis umfährt, treten nun keine Knicke im Verlauf des Skalierungsfaktors über dem Kreiswinkel ω mehr auf.

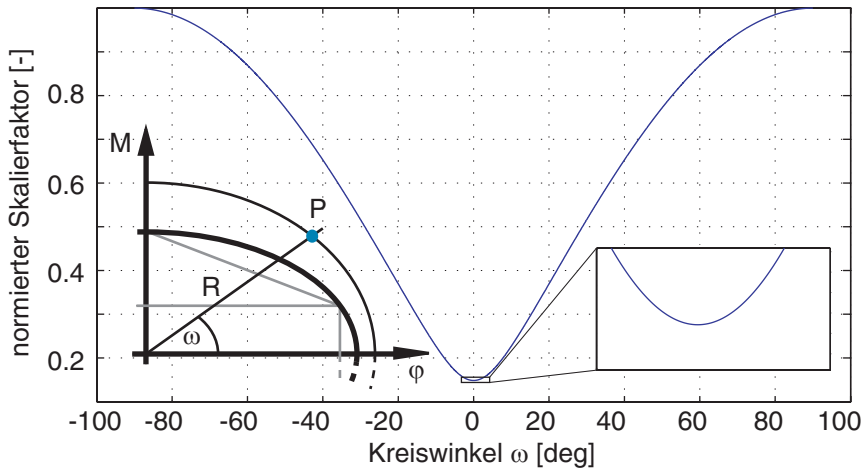


Abb. 4.4: Annäherung des Nichols-Diamanten durch Ellipse

Die Halbachsen der Ellipse berechnen sich aus den Begrenzungen des Nichols-Diamanten, es gilt für die kleinen Halbachse b :

$$b^2 = M_{lim1}^2 \tag{4.16}$$

und für die große Halbachse a :

$$a^2 = \varphi_{lim}^2 \cdot b^2 / (b^2 - M_{lim2}^2). \tag{4.17}$$

Für den Skalierungsfaktor bzw. das Robustheitsmaß gilt für alle Punkte der Ortskurve mit einem Phasenwinkel $\varphi' \geq 0^\circ$ oder $\varphi \geq -180^\circ$:

$$\rho = \sqrt{\frac{\varphi'^2}{a^2} + \frac{M^2}{b^2}}. \quad (4.18)$$

Auch hier wird der Skalierungsfaktor für Punkte mit Phasen $\varphi < -180^\circ$, also Punkte des früheren Sektors 1, pauschal auf den Wert 1000 gesetzt.

4.1.2 Formulierung der beschränkenden Nebenbedingungen

Beim automatischen Reglerentwurf mittels nichtlinearer Optimierung werden die Forderungen an die Handling-Qualities des Reglers als Ungleichungs-Nebenbedingungen formuliert. Die Vorgabe einer Mindestdämpfung der Anstellwinkelschwingung wird formuliert als Bedingung

$$b_1: \zeta_{AS} \geq 0,7 \quad (4.19)$$

bzw.

$$b_1 = \zeta_{AS} - 0,7 \geq 0. \quad (4.20)$$

Die Vorgabe eines CAP-Wertes von 1 wird umgesetzt in Form zweier Bedingungen

$$b_2: CAP \geq 1 - \varepsilon \quad (4.21)$$

bzw.

$$b_2 = CAP - 1 + \varepsilon \geq 0 \quad (4.22)$$

sowie

$$b_3: CAP \leq 1 + \varepsilon \quad (4.23)$$

bzw.

$$b_3 = -CAP + 1 + \varepsilon \geq 0, \quad (4.24)$$

d.h. für den Wert des CAP werden Werte aus dem Korridor $1 \pm \varepsilon$ zugelassen, wobei im vorliegenden Fall $\varepsilon = 0,05$ gewählt wurde.

Aus der Vorgabe einer maximalen Überschwingweite ergibt sich die Nebenbedingung

$$b_4: \alpha_{over} \leq \alpha_{over, max} \quad (4.25)$$

bzw.

$$b_4 = \alpha_{over, max} - \alpha_{over} \geq 0. \quad (4.26)$$

Der Wert der maximalen Überschwingweite $\alpha_{over, max}$ wird dabei auf 10% gesetzt.

Zusätzlich wird die eigentliche Zielgröße ρ nach unten begrenzt, um einen Mindestwert für das Stabilitätsmaß zu garantieren, die entsprechende Nebenbedingung lautet

$$b_5: \rho \geq \rho_{min} \quad (4.27)$$

bzw.

$$b_5 = \rho - \rho_{min} \geq 0 \quad (4.28)$$

mit einem Mindest-Robustheitsmaß von $\rho_{min} = 1, 15$.

4.1.3 Berechnung der beschränkenden Nebenbedingungen

Zur Auswertung der Nebenbedingungen durch das Optimierungsverfahren müssen die Zahlenwerte der beschränkten Größen berechnet werden, um diese mit ihren entsprechenden Begrenzungen zu vergleichen.

Anstellwinkelschwingung. Im Falle der Dämpfung der Anstellwinkelschwingung muss also zunächst das Polpaar der Anstellwinkelschwingung des geregelten Flugzeugs, also des geschlossenen Regelkreises identifiziert werden. Dies ist nicht trivial, denn im Gegensatz zum offenen Regelkreis, bei dem sich das entsprechende Polpaar zweifelsfrei aus der Übertragungsfunktion der Kurzzeitznäherung bestimmen lässt, können die Pole des geschlossenen Regelkreises nicht unmittelbar zugeordnet werden, denn die Pole der Übertragungsfunktionen der einzelnen Elemente des Regelkreises verschieben sich durch das Schließen des Kreises. Letztendlich geht es jedoch bei der Vorgabe der Anstellwinkelschwingungsdämpfung um eine positive Beeinflussung der Anstellwinkel-Sprungantwort. Aus diesem Grund wird in dieser Arbeit das Polpaar der Anstellwinkelschwingung mit demjenigen gleichgesetzt, das die Antwort des Anstellwinkels auf einen Sprung am Eingang des geschlossenen Regelkreises dominiert. Dazu werden im geschlossenen Regelkreis im Übertragungspfad des kommandierten Anstellwinkels auf den tatsächlichen Anstellwinkel Pole gegen eventuell vorhandene entsprechende Nullstellen gekürzt, um die betrachtete Übertragungsfunktion zu vereinfachen. (Es handelt sich beim kompletten geschlossenen Regelkreis um ein SIMO-System, da jedoch lediglich der Übertragungspfad des kommandierten Anstellwinkels α_c auf den Anstellwinkel α betrachtet wird, handelt es sich hierbei nicht um Pol-/Nullstellenkürzungen im eigentlichen Sinn, sondern nur hinsichtlich des betrachteten Signalpfades.)

Anschließend erfolgt die Identifizierung des dominanten Polpaares über die Methode nach Föllinger [18], S. 218. Als Dämpfung der Anstellwinkelschwingung wird dann die Dämpfung des dominanten Poles verwendet. In Analogie zum unregulierten Flugzeug muss es sich dabei nicht notwendigerweise um ein schwingfähiges Polpaar handeln, sondern es kann auch ein reeller Pol der Anstellwinkelschwingung zugeordnet werden.

Alternativ zur Dominanz-Methode nach Föllinger besteht die Möglichkeit, die Dämpfung der Anstellwinkelschwingung aus diversen Näherungsformeln für die Überschwingweite der Sprungantwort des Anstellwinkels zu bestimmen. Auch der Weg über eine Modalanalyse ist denkbar, d.h. aus allen Eigenvektoren des geschlossenen Regelkreises werden die Vektoren der Anstellwinkelschwingung identifiziert und über sie die zugehörigen Pole und deren Dämpfung. Die beschriebenen Methoden wurden alle im Algorithmus zum automatischen

Reglerentwurf implementiert, wobei die Methode nach Föllinger die besten Ergebnisse erzielt.

Control Anticipation Parameter CAP. Auch die Bestimmung eines Wertes für den *CAP* ist nicht trivial. Definitionsgemäß handelt es sich beim *CAP* um den Quotienten aus der Anfangsreaktion eines Flugzeugs auf einen Sprungeingang am Höhenruder in Form der Anfangsnickbeschleunigung $\dot{q}(t)$ zum Zeitpunkt des Sprunges $t = 0$ und dem sich stationär einstellenden Lastvielfachen $n_z(t \rightarrow \infty)$. Die Kenngröße beschreibt also den Zusammenhang zwischen der Anfangsreaktion des Flugzeugs und der sich darauf einstellenden Bahnänderung und gibt somit Aufschluss über das Verhältnis zwischen Agilität und Steuergenauigkeit des Flugzeugs. Je höher der *CAP*-Wert desto agiler verhält sich das Flugzeug, aber desto ungenauer kann das Flugzeug auch vom Piloten ausgerichtet werden. Je niedriger der Wert desto langsamer aber auch genauer verhält sich das Flugzeug.

Allerdings treten bei der Verwendung der eigentlichen Definition des *CAP*-Wertes nach Gleichung 3.12 unter den Entwurfszielen in Abschnitt 3.1.2 Probleme auf. Zum einen wird beim vorliegenden Anstellwinkelfolgeregler mit Integrator im Vorwärtszweig (aufgrund der Wichtungsfunktion W_1) ein Sprungeingang am Höhenruder als Störung interpretiert und ausgegletet. Dieses Problem kann umgangen werden, indem der *CAP*-Wert statt über einen Höhenrudersprung über einen Sprung des kommandierten Anstellwinkels, also quasi einen Sprung am Steuerknüppel gebildet wird. Ein weiteres Problem ist jedoch die Aktuatordynamik. Diese sorgt zum Zeitpunkt $t = 0$ des Sprungeingangs für eine Nickbeschleunigung von $\dot{q}(0) = 0$, d.h. es ergibt sich stets $CAP = 0$. Di Franco schlägt in [19] als Lösung für dieses Problem vor, anstatt der Nickbeschleunigung zum Zeitpunkt $t = 0$ die maximal auftretende Beschleunigung \dot{q}_{max} zu verwenden. Allerdings löst auch dies nicht alle Probleme, denn im Fall des vorliegenden Reglerentwurfs, das heißt für Entwurfsmodelle, die für den Flug auf einem quasi-stationären Abfangbogen getrimmt und linearisiert wurden, ist das Lastvielfache $n_z(t \rightarrow \infty)$, das sich nach einem kommandierten Anstellwinkelsprung ergibt, nicht vom Regler, sondern vom momentanen Flugzustand abhängig. Solange sich das Flugzeug auf einem Abfangbogen befindet, bleiben sowohl der Anstellwinkel als auch das Lastvielfache konstant. Durch einen Sprung im kommandierten Anstellwinkel verändert das Flugzeug seinen Anstellwinkel, um ihn anschließend wieder konstant zu halten. Das Flugzeug befindet sich dann auf einem neuen Abfangbogen mit konstantem Anstellwinkel und Lastvielfachen. Bei Betrachtung der Kurzzeitdynamik über die Kurzzeitnäherung ändern sich dabei die Flughöhe und Fluggeschwindigkeit nicht. Das Lastvielfache nach dem Sprung entspricht daher dem des getrimmten Flugzustands mit neuem Anstellwinkel, alle anderen Größen des Flugzustandes bleiben gleich. Dementsprechend ist bei der Berechnung des *CAP* nach [19] der einzige Freiheitsgrad, der sich durch den Regler verändern lässt, die maximale Nickbeschleunigung. Es zeigt sich jedoch, dass dieser Parameter relativ kleinen Einfluss auf das Aussehen der Anstellwinkel-Sprungantwort für Zeitpunkte $t > 0$ hat, sondern nur auf ihren Beginn. So ist es beispielsweise möglich, dass die Anstellwinkelantwort am Anfang schnell ansteigt, dann jedoch flacher wird und sich nur sehr langsam ihrem stationären Endwert annähert. Regelkreise mit identischen Werten für *CAP* können daher stark unterschiedliche Sprungantworten aufweisen.

Dabei wurde der *CAP* ursprünglich für unregelte Flugzeuge entwickelt, für deren Kurzzeitdynamik das Verhalten eines PT2-Gliedes angenommen wurde [1]. Das bedeutet einer-

seits, dass die Berechnung des CAP näherungsweise aus der Kreisfrequenz der Anstellwinkelschwingung und der Übertragung n_α vom Anstellwinkel auf das Lastvielfache (also dem Element der 1. Spalte der 3. Zeile der C -Matrix der Kurzzeitznäherung) gemäß

$$CAP \approx \frac{\omega_{AS}^2}{n_\alpha} \quad (4.29)$$

erfolgen kann (siehe zur Herleitung beispielsweise [1]). Andererseits bedeutet es aber auch, dass eine Betrachtung des CAP -Wertes nur für ein Flugzeug mit PT2-Verhalten Sinn hat und über den CAP die PT2-ähnliche Sprungantwort beeinflusst wird. Aus diesem Grund wird für die Berechnung des CAP -Wertes im Rahmen dieser Arbeit die Näherungsformel nach Gleichung (4.29) verwendet. Der Wert n_α ist dabei eine charakteristische Größe des Entwurfsmodells, weshalb über den CAP -Wert letztendlich die Frequenz der Anstellwinkelschwingung festgelegt wird. In Zusammenhang mit der Forderung nach einer Anstellwinkel-dämpfung $\zeta_{AS} \geq 0,7$ kommt es daher zu einer Art Polplatzierung der Anstellwinkelschwingung, die dem Flugzeug aufgrund ihrer Dominanz wiederum PT2-Verhalten aufprägt. Der Wert für ω_{AS} folgt aus der Polbestimmung im Rahmen der Ermittlung der Dämpfung der Anstellwinkelschwingung. Hierbei erfolgt allerdings die zusätzliche Einschränkung, dass nur Pole mit einer Kreisfrequenz von $\omega_0 \leq 6,5 \text{ rad/s}$ in Betracht gezogen werden.

Überschwingweite. Die Bestimmung der Überschwingweite ist im Vergleich zur Bestimmung von ζ_{AS} und CAP verhältnismäßig einfach. Für einen entworfenen Regler wird die Systemantwort für einen Sprung im kommandierten Anstellwinkel berechnet. Der Maximalwert des Anstellwinkelausgangs wird dann auf den Stationärwert bezogen.

Die Implementierung der Berechnung aller Zielgrößen erfolgt durch die Matlab-Funktion `EntwurfFun10`. Diese Funktion benötigt als Eingänge die Matrizen der Zustandsraumdarstellungen aller Elemente des Regelkreises, also Kurzzeitznäherung, Aktuator und Sensor sowie Werte für die drei Entwurfsparameter, die die Wichtungsfunktionen spezifizieren.

Darüber hinaus benötigt die Funktion diverse Informationen wie beispielsweise die Art der Reglerimplementierung und auf welche Art die Zielgrößen berechnet werden sollen für den Fall, dass mehrere Möglichkeiten bestehen.

In der Funktion wird dann zunächst die gewichtete Strecke gebildet, für die über die Lösung der beiden algebraischen Riccati-Gleichungen (3.4) und (3.5) die Reglermatrizen berechnet werden. Anschließend erfolgt die Bildung des geschlossenen Regelkreises und die Berechnung der Größen ζ_{AS} , CAP sowie die Überschwingweite des Anstellwinkels. Die Bildung des offenen Regelkreises mit der Ermittlung des Robustheitsmaßes ρ bildet den Abschluss. Der Aufruf dieser Funktion ermöglicht es damit den Optimierungsverfahren, die Optimierung der Zielgröße unter Erfüllung der Nebenbedingungen zu erreichen.

4.1.4 Durchführung einer Optimierung in Matlab

Die Durchführung einer einzelnen Optimierung in Matlab erfolgt durch das Skript `Entwurf10`. Dort ist zunächst der Flugzustand, für den der Regler entworfen werden soll, zu spezifizieren. Des Weiteren ist festzulegen, welches Optimierungsverfahren verwendet

werden soll. Nach der Festlegung, wie die einzelnen Zielgrößen berechnet werden sollen (falls mehrere Möglichkeiten vorliegen, wie beispielsweise im Fall von ζ_{AS}), und der Angabe von Start- und Begrenzungswerten für die Optimierungsparameter erfolgt nach dem Laden von Linearisierungsdaten, die ein linearisiertes Modell des Flugzeugs am betreffenden Flugzustand enthalten, der Aufruf der eigentlichen Optimierungsroutine. Diese ruft im Fall von `fmincon` zur Berechnung der Zielgröße die Funktion `regler_cost10fmincon` auf sowie zur Berechnung der Beschränkungen die Funktion `regler_const10fmincon`. Beide rufen ihrerseits die bereits erwähnte Funktion `EntwurfFun10` auf. Die Optimierungsroutine `ifrslsqp` verfährt mit den Funktionen `regler_cost10IFRSLSQP` und `regler_const10IFRSLSQP` analog. Nach erfolgter Optimierung werden diverse Plots, wie Sprungantwort, Bode-Plots der Wichtungsfunktionen und ein Nichols-Diagramm erstellt sowie im Matlab Kommandofenster eine Information ausgegeben.

4.2 Automatische Bearbeitung des Entwurfsrasters

Die Bearbeitung aller Entwurfspunkte einer gemäß Abschnitt 3.2.3 diskretisierten Envelope macht die Verwendung eines automatisierten Entwurfsverfahrens notwendig. Im Prinzip handelt es sich hierbei um den automatisierten Aufruf der Optimierungsroutine in Abschnitt 4.1.4 in einer Schleife über allen Entwurfspunkten. Die einzelnen Abschnitte der Implementierung eines solchen automatisierten Verfahrens sind Thema dieses Kapitels.

4.2.1 Bearbeitungsreihenfolge und Startwertgenerierung

Wie in Abschnitt 4.1 erwähnt wurde, benötigen numerische Optimierungsverfahren einen Startwert \vec{p}_{start} zur Bestimmung eines optimalen Satzes von Parametern \vec{p}_{opt} . Dementsprechend müssen zur automatisierten Abarbeitung eines Entwurfsrasters auch für jeden Entwurfspunkt Parameter-Startwerte generiert werden. Hierzu sind verschiedene Strategien denkbar. Beispielsweise können für jeden Entwurfspunkt dieselben Startwerte verwendet werden. Hierbei kommt es jedoch zu Problemen, denn Startwerte, die für einen Flugzustand direkt eine zwar suboptimale aber dennoch im Sinne der Nebenbedingungen gültige Lösung liefern, können an anderen Entwurfspunkten zu ungültigen Lösungen führen. Im besten Fall schafft es das Optimierungsverfahren dennoch, zunächst gültige Lösungen zu finden und von diesen die bestmögliche zu bestimmen. Unter ungünstigen Umständen führen solche im Sinne der Nebenbedingungen ungültige Startwerte nicht ans Ziel. Darüber hinaus ist die Regleroptimierung aufgrund der Beschaffenheit der Zielfunktion und der Nebenbedingungen startwertabhängig: Die Zielfunktion hat neben einem globalen Minimum weitere, relative Minima, die Nebenbedingungen sind in mehreren Bereichen innerhalb des Optimierungs-Parameterraumes gültig. Dabei sind beide zur Anwendung kommende Optimierungsverfahren gradientenbasiert, weshalb die Optimierung von einem Punkt im Parameterraum zum folgenden in erster Näherung in Richtung einer Verbesserung der Zielfunktion bzw. der Nebenbedingungen voranschreitet, bis keine Verbesserung der Zielfunktion mehr auftritt, d.h. bis ein relatives Minimum der Zielfunktion erreicht wird und der Gradient der Zielfunktion in Abhängigkeit der Optimierungsparameter Null wird. Hieraus folgt unmittelbar, dass

es vom Startpunkt der Optimierung abhängt, welches der existierenden relativen Optima gefunden wird. Abbildung 4.5 illustriert die Situation. Für den Flugzustand bei $\alpha = 13, 5^\circ$, $Mach = 0,3$ und $q_{dyn} = 1899,6 \text{ Pa}$ werden ausgehend von den Startpunkten \vec{P}_1 und \vec{P}_2 zwei Optimierungen durchgeführt, die als Ergebnis die Punkte $\vec{P}_{opt,1}$ und $\vec{P}_{opt,2}$ liefern. Die Abbildung zeigt den Verlauf der Zielgröße ρ sowie der Begrenzungen CAP und ζ_{AS} für Wichtungsfunktionsparameter, die auf der Verbindungsgerade g der Punkte $\vec{P}_{opt,1}$ und $\vec{P}_{opt,2}$ liegen. Für die Gerade gilt

$$\vec{x}_g = \vec{P}_{opt,1} + \lambda \cdot \overrightarrow{P_{opt,1}P_{opt,2}}, \quad (4.30)$$

mit

$$-0,2 \leq \lambda \leq 1,2. \quad (4.31)$$

Es zeigt sich, dass es bereits entlang der Verbindungsgeraden mehrere Bereiche gibt, in denen die Nebenbedingungen erfüllt sind. So ist für alle Werte $-0,2 \leq \lambda \leq 1,0$ die Bedingung $0,95 \leq CAP \leq 1,05$ erfüllt, für die Vorgabe einer minimalen Dämpfung ergeben sich zwei gültige Bereiche, zum einen $-0,2 \leq \lambda \leq 0,53$ und $0,854 \leq \lambda \leq 1,2$. Beide Nebenbedingungen sind somit in den Bereichen $-0,2 \leq \lambda \leq 0,53$ und $0,854 \leq \lambda \leq 1,0$ erfüllt. Je nachdem, wo die Optimierung startet, wird das Optimum der Zielfunktion des einen oder des anderen Bereiches gefunden. (Da es sich bei der Gerade g um die Verbindung der beiden gefundenen Optima handelt, entspricht der Wert $\lambda = 0$ dem Punkt $\vec{P}_{opt,1}$, sowie der Wert $\lambda = 1$ dem Punkt $\vec{P}_{opt,2}$).

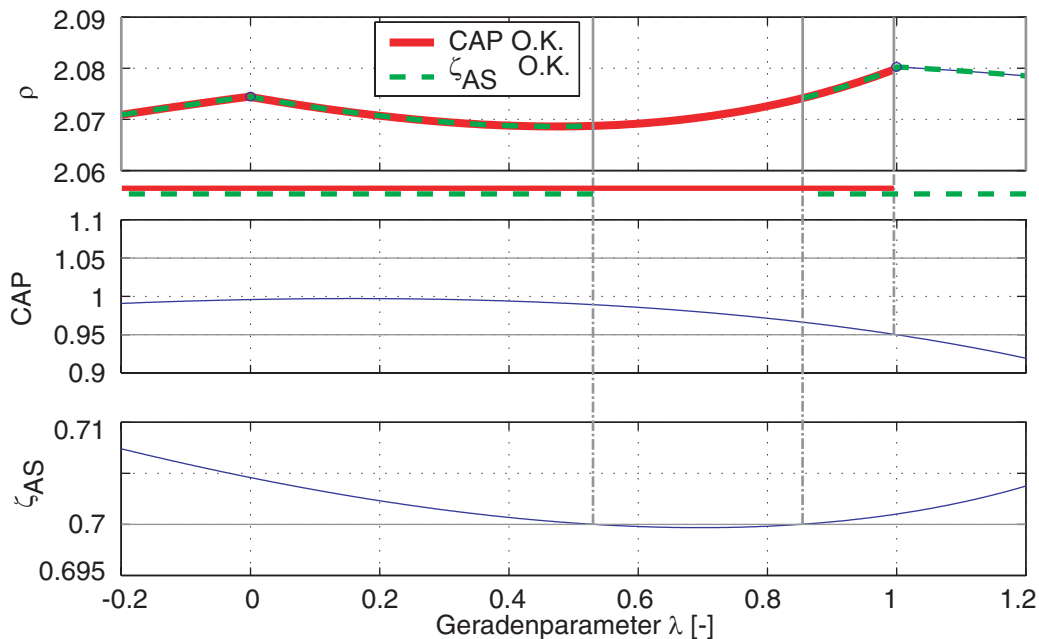


Abb. 4.5: Startwertabhängigkeit der Optimierung - verschiedene gültige Lösungen

Diese Startwertabhängigkeit ist jedoch nicht unbedingt ein Nachteil, denn sie bedeutet, dass mit der Wahl des Startpunktes in gewisser Weise das Ergebnis der Optimierung beeinflusst werden kann. Dies ist ein Vorteil, denn so kann beispielsweise erreicht werden, dass sich die Optimierungsergebnisse benachbarter, also sich ähnelnder Entwurfspunkte ebenfalls ähneln.

Laut [6] ist dies eine Voraussetzung für Gain-Scheduling-fähige Regler, denn laut Hyde muss der Verlauf der Reglerparameter über den Scheduling-Parametern stetig differenzierbar sein. In Richtung des Anstellwinkels kann aufgrund der starken Veränderungen des dynamischen Verhaltens der Regelstrecke für verschiedene Anstellwinkel kein stetig differenzierbarer Verlauf der durch Optimierung ermittelten Reglerparameter erwartet werden. Aus diesem Grund wird in Abschnitt 5.3 ein Verfahren vorgestellt, das es ermöglicht, auch bei nicht stetig differenzierbaren Verläufen der Reglerparameter ein Scheduling vorzunehmen. In Richtung der Machzahl und vor allem in Richtung des Staudrucks sind jedoch stetig differenzierbare Verläufe möglich und erwünscht, denn nur dann ist die Annäherung der Verläufe über dem Staudruck durch Parabeln gemäß Abschnitt 3.2.2 möglich.

Anstatt also für alle Entwurfspunkte dieselben Startwerte zu verwenden, werden die Startwerte so generiert, dass sich wenn möglich ein gewünschter Ergebnisverlauf ergibt. Für einen gegebenen Anstellwinkel und eine gegebene Machzahl werden dabei zunächst alle vorkommenden Staudrücke bearbeitet. Die Ergebnisse einer Berechnung dienen als Basis für die Bestimmung der Startwerte für die folgende Berechnung. Im einfachsten Fall werden dazu die Ergebnisse einer Optimierung direkt als Startwerte für die Folgeberechnung verwendet. Alternativ dazu können auch mehrere vorangegangene Ergebnisse zur Berechnung einer Regressionsgeraden oder -parabel verwendet werden, wobei als Startwert der entsprechende extrapolierte Wert der Regressionskurve verwendet wird. Theoretisch lassen sich hierdurch bereits parabelförmige Verläufe der Optimierungsparameter über dem Staudruck erzeugen. In der Praxis kommt es allerdings leicht zu Ausreißern, denn aufeinander folgende Resultate, die in eine Richtung der Krümmung der Regressionsparabel von der Kurve abweichen, führen zu einer Verlagerung der Regression in die betreffende Richtung, was zu einer starken Verkrümmung der Parabel führt und Startwerte daher in Bereiche gesetzt werden, die von der eigentlichen Tendenz in Richtung des Staudrucks stark abweichen. Im Zuge verschiedener Optimierungsdurchläufe hat sich gezeigt, dass die besten Ergebnisse erzielt werden, wenn vor der Bildung von regressionsbasierten Startwerten zunächst eine bestimmte Anzahl von Startwerten direkt aus den vorangegangenen Ergebnissen übernommen wird.

Mit der direkten Verwendung von Ergebnissen als Startwerte wurden daher im Zuge verschiedener Optimierungsdurchläufe die besten Ergebnisse erzielt.

Erst wenn alle Staudrücke eines bestimmten Anstellwinkels und einer bestimmten Machzahl abgearbeitet wurden, erfolgt die Bearbeitung aller Staudrücke bei der nächsten Machzahl. Wurden alle Staudruck-Machzahl-Kombinationen eines Anstellwinkels bearbeitet, erfolgt die Bearbeitung des nächsten Anstellwinkels. Die Startwertgenerierung bei Anstellwinkel- und Machzahlwechseln steht in Zusammenhang mit der Gesamt-Bearbeitungsreihenfolge der Entwurfspunkte und es gibt verschiedene Strategien.

Mäanderförmige Abarbeitung mit einem Start-Startwert-Vektor. Die Entwurfspunkte werden mäanderförmig abgearbeitet, sodass sich nacheinander abgearbeitete Entwurfspunkte stets möglichst wenig, d.h. in möglichst wenigen Komponenten des zugehörigen Flugzustands voneinander unterscheiden. Unter der Annahme, dass sich bei ähnlichen Flugzuständen auch ähnliche Optimierungsergebnisse ergeben, werden Berechnungsergebnisse nicht nur im Fall von Staudruckwechseln als Startwert für den folgenden Flugzustand verwendet, sondern auch, wenn sich die Machzahl oder der Anstellwinkel zwischen zwei aufeinander folgenden Bearbeitungspunkten geändert hat. Dementsprechend erfolgt die Abarbeitung aller Entwurfs-

punkte gemäß Abbildung 4.6. Hier ist ein exemplarisches Entwurfsraster mit einer mäanderförmigen Bearbeitungsreihenfolge dargestellt. Im Diagramm ist ersichtlich, dass beim Wechseln des Anstellwinkels zwischen zwei Entwurfspunkten sowohl die Machzahl als auch der Staudruck konstant bleiben. Darüber hinaus kehrt sich die Bearbeitungsrichtung bei Abarbeitung der Staudrücke mit jedem Machzahlwechsel um. Es entsteht eine mäanderförmige Bearbeitungsreihenfolge. Dabei genügt in diesem Fall die Vorgabe eines einzelnen Start-Startwerte-Vektors. Dieser wird im Skript `SettingsV11` durch den Parameter `IPStart` spezifiziert.

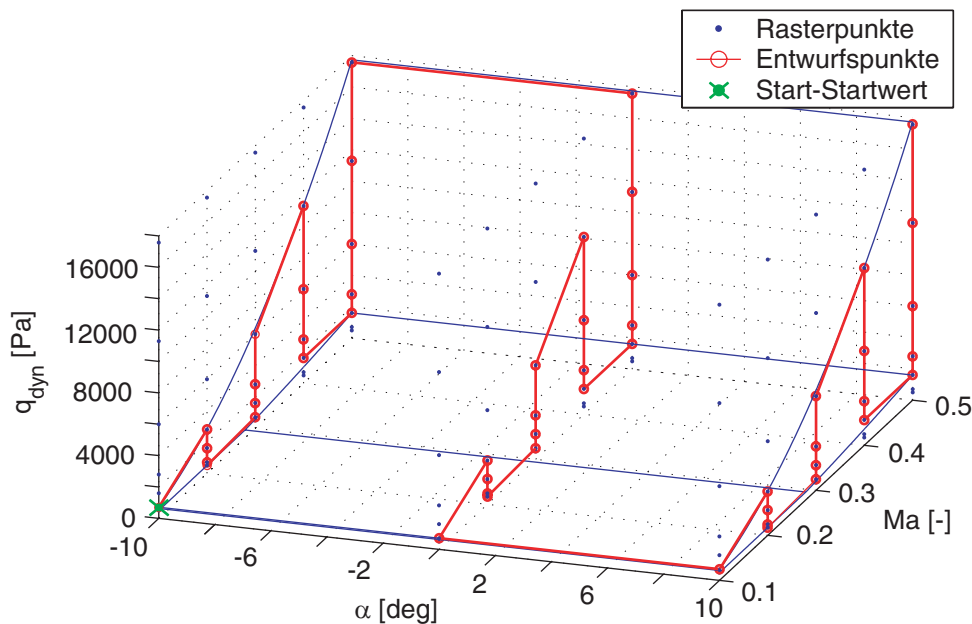


Abb. 4.6: Bearbeitungsreihenfolge für mäanderförmige Abarbeitung des Entwurfsrasters

Teilweise mäanderförmige Abarbeitung. Alternativ zu einer komplett mäanderförmigen Abarbeitung des Entwurfsrasters erfolgt eine Gliederung nach dem Anstellwinkel. In diesem Fall werden nur die Punkte aller Staudrücke und Machzahlen eines Anstellwinkels mäanderförmig abgearbeitet. Die Bearbeitung startet dabei, wie Abbildung 4.7 zeigt, für jeden Anstellwinkel bei der kleinsten Machzahl und dem zugehörigen kleinstmöglichen Staudruck. Bei jedem Anstellwinkelwechsel wird nicht das letzte Optimierungsergebnis verwendet, sondern es werden erneut Start-Startwerte eingelesen. Hierbei können wahlweise stets die gleichen Startwerte verwendet werden, das heißt Werte, die über die Variable `IPStart` in Skript `SettingsV11` spezifiziert werden, oder es werden zuvor generierte und separat abgespeicherte Startwerte eingelesen. Die Startwertberechnung erfolgt in einem separaten Optimierungsdurchgang für alle Anstellwinkel bei der kleinstmöglichen Machzahl und dem kleinstmöglichen Staudruck. Das Skript `ResMatGen11DivAlg` bietet die Möglichkeit, mittels verschiedener Optimierungsalgorithmen solche Startwerte zu generieren. Mit dem Skript `ResMatEdit01` kann ein derart erzeugter Startwerte-Datensatz zusätzlich editiert werden. Der Vorteil dieser Methode ist, dass ausgehend von der ersten Machzahl-Staudruck-Kombination jedes Anstellwinkels in Richtung des Anstellwinkels allzu große Sprünge der Startwerte und damit der Optimierungsergebnisse verhindert werden. Nach dem Einlesen der

Startwerte nach jedem Anstellwinkelwechsel erfolgt wiederum eine mäanderförmige Abarbeitung des Entwurfsrasters.

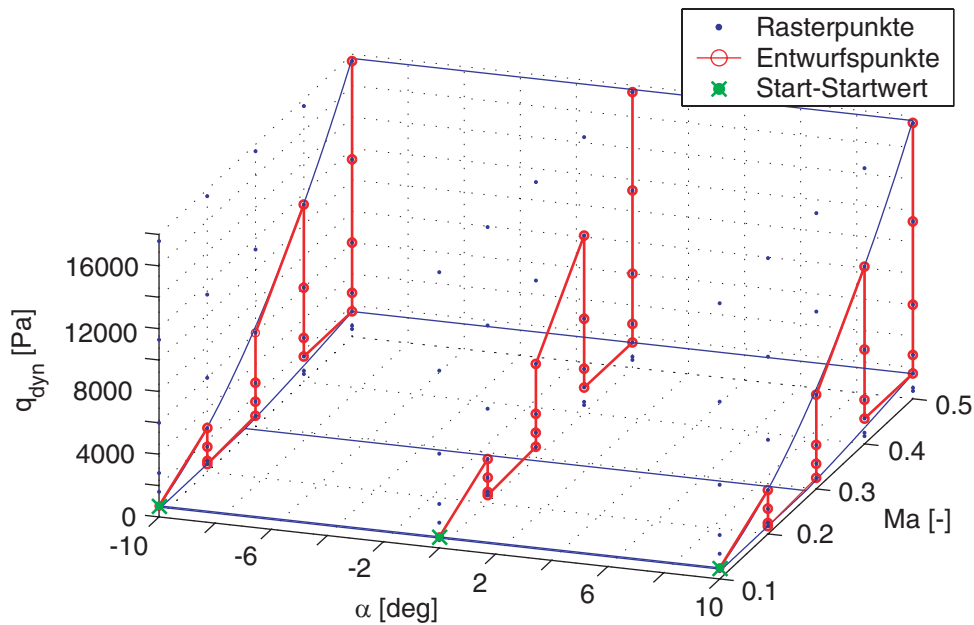


Abb. 4.7: Bearbeitungsreihenfolge bei teilweise mäanderförmiger Abarbeitung

Bürstenförmige Abarbeitung. Eine weitere Methode ist die Generierung von Startwerten für jeden Anstellwinkel und jede Machzahl beim größtmöglichen Staudruck bzw. bei der kleinstmöglichen Flughöhe bei einer anschließenden Bearbeitung in Richtung des Staudrucks. Wie Abbildung 4.8 zeigt, ist diese Bearbeitungsreihenfolge vergleichbar mit einer Bürste: die gekrümmte Fläche, in der Startwerte zur Verfügung stehen, entspricht dem Kopf, es erfolgt dann die Bearbeitung der verschiedenen Staudruckwerte entlang der Borsten.

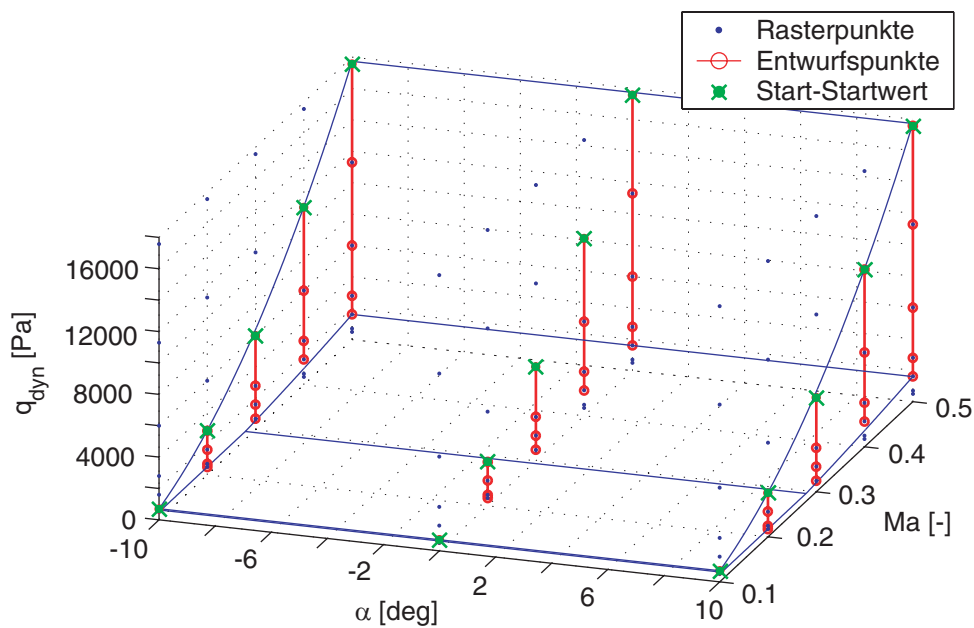


Abb. 4.8: Bearbeitungsreihenfolge bei bürstenförmiger Abarbeitung

Der Vorteil dieses Verfahrens besteht in einer noch besseren Beeinflussbarkeit der Optimierungsergebnisse hinsichtlich eines ruhigen Verlaufs in Richtung des Anstellwinkels. Der Nachteil ist jedoch ein verhältnismäßig großer Aufwand zur Berechnung der Startwerte im Bürstenkopf. Anders als im Fall der teilweise mäanderförmigen Abarbeitung müssen hier nicht nur alle Anstellwinkel bei der niedrigsten Machzahl und dem maximalen zugehörigen Anstellwinkel bearbeitet werden, sondern alle Anstellwinkel für alle Machzahlen. Diese Startwertgenerierung kann ebenfalls über das Skript `ResMatGen11DivAlg` erfolgen, bei einer eventuellen Nachbearbeitung über das Skript `ResMatEdit01`. In Richtung des Staudrucks werden wie im Fall der anderen Verfahren Optimierungsergebnisse zur Startwertgenerierung für nachfolgende Entwurfspunkte verwendet.

Bei der Durchführung eines automatischen Reglerentwurfs erfolgt die Einstellung, welche der beschriebenen Strategien verwendet wird, im Skript `SettingsV11` über die Parameter *meander* und *IPread* entsprechend der Tabelle 4.1. Bei den im Rahmen dieser Arbeit durchgeführten Reglerentwürfen liefert das Verfahren Nr. 4 bei einem vertretbaren Aufwand zur Vorabermittlung von Startwerten die besten Ergebnisse.

Nr.	Verfahren	<i>IPread</i>	<i>meander</i>
1	teilweise mäanderförmig, gleiche Start-Startwerte für jeden Anstellwinkel	0	1
2	bürstenförmige Bearbeitung	1	0
3	komplett mäanderförmig	2	1
4	teilweise mäanderförmig, Start-Startwerte für jeden Anstellwinkel einlesen	3	1

Tabelle 4.1: Bearbeitungsreihenfolge und Verfahren zur Startwertbestimmung

4.2.2 Ablauf der automatischen Bearbeitung und dabei beteiligte Matlab-Files

Die Steuerung der automatischen Abarbeitung der Entwurfspunkte erfolgt durch das Matlab Skript `regler_opt11_DivAlgAuto`. Abbildung 4.9 zeigt ein vereinfachtes Struktogramm dieses Skripts. Dieses Prinzip-Struktogramm enthält dabei kursiv gedruckt eine kurze Beschreibung der jeweils ausgeführten Aktion sowie in Klammern den Namen der zugehörigen Funktion. Wie Abbildung 4.9 zeigt, erfolgt die Abarbeitung aller Entwurfspunkte über mehrere ineinander geschachtelte Schleifen. In einer äußeren `for`-Schleife werden nacheinander verschiedene Anstellwinkel abgearbeitet, in einer inneren `for`-Schleife erfolgt für den jeweiligen Anstellwinkel die Bearbeitung der verschiedenen Staudrücke und Machzahlen. Die Bearbeitungsreihenfolge entspricht einer der im vorangegangenen Abschnitt beschriebenen Varianten. An jedem Entwurfspunkt erfolgt die Berechnung der Reglerparameter innerhalb einer `while`-Schleife, die so lange ausgeführt wird, bis die Parameter erfolgreich optimiert wurden oder bis eine Maximalzahl von Schleifendurchgängen erreicht wird.

Über das Steuerungsskript `regler_opt11_DivAlgAuto` werden die verschiedenen beteiligten Skripte und Funktionen zu gegebener Zeit aufgerufen.

An dieser Stelle erfolgt über Abbildung 4.9 ein Überblick über diese Funktionen und Skripte sowie den Ablauf der automatischen Bearbeitung. Zur Ergänzung sei auf die Kommentare im Programmcode der entsprechenden Skripte und Funktionen verwiesen.

<i>Auswahl Linearisierungsdaten (ChooseLinDataV11)</i>
<i>Festlegung Einstellungen (SettingsV11)</i>
<i>Festlegung Versionsnummer (SaveVersionNumberV11)</i>
<i>Spezifizierung Startwertdaten (ChooseInitialDataV11)</i>
<i>Eintragung Logfile (CreateLogV11)</i>
For Anstellwinkel-Schleife
<i>Laden der Linearisierungsdaten</i>
<i>Anordnen der Flugzustände (OrderV11)</i>
<i>Festlegung des Start-Startwertes</i>
<i>ÜF Aktuator und Sensor</i>
For Machzahl-Staudruck-Schleife
<i>Detektierung Machzahlwechsel</i>
<i>Festlegung Parameter StartFlag</i>
<i>Bildung (SPA)</i>
<i>Startwertgenerierung, Festlegung Parametergrenzen und Algorithmus- Einstellungen (StartWertGenerierungV11 und LimitsV11 bzw. RetryStartWertGenerierungV11 und RetryLimitsTolV11)</i>
<i>Durchführung eigentliche Optimierung</i>
<i>Analyse Ergebnisse (AnalyseOptimV11)</i>
<i>Entscheidungsfindung (DecissionV11)</i>
While Optimierungsschleife
<i>Aktualisierung Info-Textfile (CreateTextInfoV11)</i>
<i>Ausgabe Ergebnisse (DisplayResultsV11)</i>
<i>Abspeichern Ergebnisse (SaveOptimDataV11)</i>
<i>Fortsetzungsmodus (ZipMergeV11)</i>
<i>Eintragung ins LogFile</i>

Abb. 4.9: Struktogramm des Steuerungsskripts regler_opt11_DivAlgAutoV11

Auswahl der zu bearbeitenden Linearisierungsdaten. Nach dem Start des Skriptes erfolgt zunächst über die Funktion ChooseLinDataV11 eine Auswahl der Linearisierungsdaten, für die Regler zu entwerfen sind. Wie bereits in Abschnitt 2.2.8 beschrieben wurde, handelt es sich bei den zur Auswahl stehenden Workspaces um Daten, die über das Skript calculation_lin10_Auto generiert wurden, das heißt es handelt sich um Linearisierungsdaten, die nach Anstellwinkeln gegliedert und in Zip-Archiven komprimiert und zusammengefasst wurden. Ein entsprechendes Zip-Archiv enthält daher mehrere Workspaces, die jeweils für einen Anstellwinkel die Linearisierungsdaten verschiedener Machzahlen und Staudruckwerte enthalten. Das Machzahl-Staudruck-Raster aller zu einem Zip-Archiv zusammengefassten Daten ist dabei für alle enthaltenen Anstellwinkel gleich. Wie im vorangegangenen Abschnitt erklärt wurde, erfolgt die Bearbeitung des Entwurfsrasters zunächst für alle Staudrucke und Machzahlen eines Anstellwinkels. Bei jedem zu bearbeitenden Anstellwinkel sind daher erneut Linearisierungsdaten zu laden. Die Identifizierung der zum

momentan bearbeiteten Anstellwinkel gehörenden und zu ladenden Workspace-Datei erfolgt über ihren Dateinamen. In `ChooseLinDataV11` wird durch die Auswahl eines Zip-Archivs ein gemeinsamer Namensstamm für alle Linearisierungsdaten definiert. In diesen Namensstamm wird vor dem Laden der Daten eines Anstellwinkels der entsprechende Winkel eingesetzt. Damit steht der Name des Workspaces, der die benötigten Daten enthält, fest und die Daten werden geladen. Die Linearisierungsdaten müssen dabei nicht unbedingt in Form eines Zip-Archives vorliegen. Die Bearbeitungsroutine kann Workspaces mit Linearisierungsdaten auch direkt laden. Dies ist interessant, falls Reglerentwürfe für Linearisierungsdaten durchgeführt werden sollen, die nicht mit dem Skript `calculation_lin10_Auto` generiert wurden. Hierbei ist allerdings zu beachten, dass die Namen der Workspaces folgende Eigenschaften haben:

- Die Zeichenkette “AL” muss enthalten sein, gefolgt von dem Anstellwinkel, für den der Workspace Daten enthält. Ein Komma wird dabei durch das Zeichen “d” ersetzt. Beispiel: Der Workspace enthalte Daten für den Anstellwinkel $44,5^\circ$, dann muss der zugehörige Name die Zeichenkette “AL44d5” enthalten.
- Die Zeichenkette “QDYN” bzw. “H” muss enthalten sein, weil über sie die Entscheidung erfolgt, ob ein Staudruck- oder Höhenraster vorliegt.
- Zwischen der Beschreibung des Anstellwinkels und des Staudrucks bzw. der Höhe ist zur Trennbarkeit das Zeichen “_” notwendig. Im vorliegenden Beispiel muss der betrachtete Dateiname im Falle einer Staudruckrasterung also die Zeichenkette “AL44d5_QDYN” enthalten.

Die zur automatischen Bearbeitung der Entwurfspunkte verwendbaren Workspaces mit Linearisierungsdaten müssen die in Tabelle 2.15 zusammengestellten Variablen sowie die Variable `linarray` enthalten. Die Zusammensetzung der Variable `linarray` muss dabei derjenigen in Tabelle 2.14 entsprechen.

Festlegung der Einstellungen. Bevor in der äußersten Schleife, der Anstellwinkelschleife, erstmals Linearisierungsdaten geladen werden, erfolgt zunächst der Aufruf des Skriptes `SettingsV11`. In diesem Skript werden sämtliche Einstellungen vorgenommen, die den automatisierten Entwurf charakterisieren. In `SettingsV11` wird festgelegt, welche der zur Verfügung stehenden Anstellwinkel bearbeitet werden und welche ggf. übersprungen werden sollen. Die Art der Berechnung der Zielgröße ρ und der Begrenzungen ζ_{AS} und CAP sowie deren Begrenzungswerte werden festgelegt. Ein Start-Startwert für die Optimierung wird angegeben, dieser wird jedoch nur dann verwendet, falls nicht extern generierte Startwerte eingelesen werden sollen. Die Spezifizierung der Bearbeitungsreihenfolge entsprechend Abschnitt 4.2.1 sowie die Entscheidung, ob Startwerte eingelesen werden oder nicht, erfolgt ebenfalls in `SettingsV11`. Ebenso erfolgt die Auswahl des zu verwendenden Optimierungsalgorithmus und dessen Konfiguration in Form der Angabe einer Abbruchtoleranz, einer Toleranz für die Erfüllung der Nebenbedingungen, ggf. einer Schrittweite für die numerische Gradientenbestimmung sowie der Angabe von Parametergrenzen zur Bestimmung eines zulässigen Bereichs für die Optimierungsparameter. Hierbei ist zu beachten, dass die Bearbeitung des Entwurfsrasters in verschiedenen Bearbeitungsebenen erfolgt, die sich in der Art der Startwertgenerierung und der Konfiguration des Optimierungsalgorithmus unterscheiden. Die Identifizierung der Bearbeitungsebenen erfolgt über den Parameter `StartFlag`, nach der erfolgreichen Bearbeitung eines Entwurfspunktes wechselt

die Bearbeitungsroutine zur Berechnung des Entwurfspunktes mit dem nächsten Staudruckwert durch eine Erhöhung des Parameters *StartFlag* auf eine höhere Ebene. Der Hintergrund ist der, dass nach einer erfolgreichen Berechnung das Ergebnis als Startwert für die folgende Berechnung in Richtung des Staudruckes verwendet werden kann. Nach zwei erfolgreichen Berechnungen kann theoretisch ein Mittelwert als Startwert gebildet werden oder es kann eine Geradenregression zur Startwertbildung erfolgen, nach dreien eine Parabelregression. Geht man zudem davon aus, dass ein Startwert umso besser wird, je mehr Ergebnisse zu seiner Bildung verwendet wurden, dann hat es Sinn, dem Optimierungsalgorithmus bei der Bearbeitung höherer Bearbeitungsebenen den zulässigen Bereich der Optimierungsparameter einzuschränken. Hierdurch wird einerseits eine Beschleunigung des Verfahrens erzielt, weil das Optimum innerhalb eines kleineren Bereichs gesucht wird, andererseits lässt sich so die Lage des Optimums besser beeinflussen, wodurch im besten Fall ein Wunschverlauf der Optimierungsparameter in Richtung des Staudrucks erreicht werden kann. Neben einer Einschränkung des Parameterbereichs erfolgt mit der Erhöhung der Bearbeitungsebene auch eine Vergrößerung der Abbruchtoleranz, d.h. die Optimierung eines Bearbeitungspunktes ist schneller beendet, was wiederum ebenfalls zum einen das Verfahren beschleunigt und andererseits die Beeinflussbarkeit des Ergebnisses verbessert. Die Konfigurationseinstellungen des Optimierungsalgorithmus' hängen also von der Bearbeitungsebene ab, weshalb in *SettingsV11* Einstellungen für jede Ebene getroffen werden.

Wie bereits erwähnt, kann eine Parabelregression bei der Startwertgenerierung auch zu Problemen führen, weshalb in *SettingsV11* zum einen eine Anlauflänge für die Parabelregression definiert werden kann, das heißt es wird angegeben, wieviele erfolgreiche Berechnungen nacheinander erfolgen müssen, bis erstmals ein Startwert durch Parabelregression gebildet wird. Zum anderen kann eine maximale Berechnungsebene manuell festgelegt werden, deren Wert der Parameter *StartFlag* auch nach mehreren erfolgreichen Berechnungen in Folge nicht überschreiten darf.

Die Parameterbegrenzungen werden im Laufe des Programms von den Startwerten ausgehend festgelegt (s.u.), in *SettingsV11* werden jedoch auch globale Grenzen festgelegt, d.h. eine globale Ober- und Untergrenze für die Optimierungsparameter, die in keinem Fall über- bzw. unterschritten werden darf. Ebenso wird die Größe eines Mindest-Korridors um den jeweiligen Startwert für die Optimierungsparameter festgelegt.

Neben der Spezifikation der Reglerimplementierung für die Bearbeitung über den Parameter *ControllerType* wird in *SettingsV11* auch die Anzahl der Wiederholungsversuche nach der erfolglosen Berechnung eines Entwurfspunktes spezifiziert, bis die Bearbeitung des Punktes abgebrochen wird und mit dem folgenden Entwurfspunkt fortgefahren wird.

Festlegung der Versionsnummer. Zur Unterscheidung verschiedener Durchläufe des automatischen Reglerentwurfs erfolgt über das Skript *SaveVersionNumberV11* die Festlegung einer Versionsnummer. Aus ihr wird zusammen mit dem Namensstamm der zur Bearbeitung ausgewählten Linearisierungsdaten der Namensstamm der Workspaces gebildet, unter denen die Ergebnisse der automatischen Bearbeitung abgespeichert werden.

Spezifizierung der Startwertdaten. Je nach Einstellung für die Startwertgenerierung wird an dieser Stelle der Workspace ausgewählt, der die ggf. zu ladenden Startwerte enthält. Die

Generierung einer Anzeige der zur Verfügung stehenden Daten und die Auswahl eines Datensatzes erfolgt über die Funktion `ChooseInitialDataV11`.

Eintragung ins Logfile. Zur Dokumentation verschiedener Durchläufe des automatischen Reglerentwurfs werden die wichtigsten Einstellungsparameter aus `SettingsV11`, die ausgewählten Linearisierungsdaten und die ggf. ausgewählten Startwertdaten in ein Text-File geloggt. Optional können die Angaben zusätzlich in eine Excel-Datei geloggt werden. Der Name dieser Datei setzt sich zusammen aus dem Namen des Computers, auf dem die Bearbeitung erfolgt und dem Wert der Variablen `NamLogFile`, der in `SettingsV11` spezifiziert wird. Mögliche Namen sind z.B. `IFR036_OptimLog02.xls` bzw. `IFR036_OptimLog02.txt`.

Laden der Linearisierungsdaten. In der äußeren Schleife, der Anstellwinkelschleife, werden zunächst die Linearisierungsdaten geladen, die zum momentan bearbeiteten Anstellwinkel gehören. Hierzu wird aus dem Namensstamm, der zuvor in `ChooseLinDataV11` bestimmt wurde, und dem momentanen Anstellwinkel der Name der zu ladenden Datei generiert. Anschließend wird der entsprechende Workspace geladen.

Anordnen der Flugzustände. Über das Skript `OrderV11` erfolgt die Anordnung der Entwurfspunkte entsprechend der gewählten Bearbeitungsreihenfolge und Startwertgenerierungsmethode. Siehe hierzu auch Abschnitt 4.2.1.

Festlegung des Start-Startwertes. Für die erste zu bearbeitende Machzahl-Staudruck-Kombination des aktuellen Anstellwinkels müssen Startwerte für die Optimierungsparameter festgelegt werden. Im Falle der Kombination von Startwertgenerierung und Bearbeitungsreihenfolge entsprechend Methode 1 aus Tabelle 4.1 werden hierbei für jeden Anstellwinkel die gleichen Werte, genauer gesagt der Startwertvektor `IPStart` verwendet. Im Fall der komplett mäanderförmigen Bearbeitung entsprechend Methode 3 werden für den ersten zu bearbeitenden Anstellwinkel die Werte aus `IPStart` verwendet, bei den folgenden Anstellwinkeln wird das Ergebnis der vorangegangenen Berechnung als Startwert verwendet. Im Falle von Methode 4 werden Startwerte eingelesen. Dazu wird der über das Skript `ChooseInitialDataV11` spezifizierte Startwerte-Workspace geladen und die zum aktuellen Anstellwinkel gehörenden Werte werden extrahiert. Im Falle von Methode 2 werden ähnlich wie bei Methode 4 Startwerte eingelesen. Hier erfolgt das Einlesen jedoch nicht nur bei der ersten Machzahl-Staudruck-Kombination, sondern auch bei jedem Machzahlwechsel. In diesem Fall erfolgt die Start-Startwertbestimmung daher erst, wenn ein Machzahlwechsel detektiert wurde (s.u.).

Bildung der Übertragungsfunktionen von Aktuator und Sensor. Die Funktion `EntwurfFun10` benötigt zur Berechnung der Zielgröße ρ und der Nebenbedingungen für verschiedene Werte der zu optimierenden Parameter der Wichtungsfunktionen unter anderem die Übertragungsfunktionen der Sensorik und Aktuatorik. Letztere ändern sich für verschiedene Parameterwerte nicht. Deshalb ist es sinnvoll, die Berechnung dieser Übertragungsfunktionen nur einmal durchzuführen, um sie an die Funktion zu übergeben, anstatt sie dort bei jedem Aufruf erneut zu berechnen. Die Berechnung erfolgt entsprechend Abschnitt 2.2.9 über die Funktionen `mkact` und `mksens`.

Detektierung von Machzahlwechseln. In der inneren Schleife, in der für einen bestimmten Anstellwinkel alle zur Verfügung stehenden Machzahl-Staudruck-Kombinationen bearbeitet

werden, müssen Änderungen in der Machzahl zweier aufeinander folgender Bearbeitungspunkte detektiert werden. Dies ist notwendig, da beispielsweise im Fall der Startwertgenerierung und Bearbeitungsreihenfolge nach Methode 2 aus Tabelle 4.1 bei jedem Machzahlwechsel erneut Startwerte eingelesen werden müssen. In diesem Fall erfolgte im Rahmen der Auswahl der Startwertdaten über `ChooseInitialDataV11` die Spezifizierung einer Startwertdaten-Workspacefamilie über einen zugehörigen Namensstamm. In diesem Namensstamm wird bei jedem Machzahlwechsel die aktuelle Machzahl eingesetzt, anschließend werden die Startwerte geladen.

Auch im Fall der übrigen Methoden zur Startwertgenerierung und Festlegung der Bearbeitungsreihenfolge ist die Erkennung von Machzahlwechseln notwendig, da der Zähler *SuccCount*, über den die aufeinanderfolgend erfolgreich bearbeiteten Reglerentwürfe gezählt werden, zurückgesetzt werden muss. Dies ist notwendig, da bei der Bearbeitung von Entwurfspunkten der gleichen Machzahl und verschiedener Staudrücke nach mehreren erfolgreichen Berechnungen in Folge die Startwertgenerierung (je nach Einstellung) beispielsweise durch Mittelwertbildung oder Geradenregression in Richtung des Staudrucks erfolgt. Bei Machzahlwechseln dürfen jedoch die vorangegangenen Ergebnisse nicht mehr zur Bildung von Regressionen verwendet werden. Je nach Einstellung werden also im Falle eines Machzahlwechsels entweder neue Startwerte geladen oder es wird das letzte gültige, vorangegangene Optimierungsergebnis als Startwert verwendet.

Festlegung des Parameters StartFlag. Wie bereits im Abschnitt über die Festlegung der Einstellungen durch das Skript `SettingsV11` beschrieben wurde, erfolgt die Unterscheidung der verschiedenen Bearbeitungsebenen über den Parameter *StartFlag*. Der Wert dieses Parameters wird in Abhängigkeit des Wertes von *SuccCount* festgelegt. Im Falle von Machzahl- und Anstellwinkelwechseln wird der Parameter immer dann, wenn externe Startwerte zur Verfügung stehen und verwendet werden, auf den Wert 0,5 gesetzt. In diesem Fall wird davon ausgegangen, dass es sich bei den Startwerten bereits um Optimierungsergebnisse handelt und das Ergebnis der anstehenden Optimierung daher gleich den Startwerten sein soll bzw. nur wenig abweicht. Deshalb werden im Fall von *StartFlag* = 0, 5 die Parametergrenzen so festgelegt, dass für die Parameterwerte nur ein kleiner Bereich um die Startwerte zugelassen wird. Des Weiteren wird die Abbruch-Toleranz der Optimierung erhöht, d.h. die Optimierung ist schneller beendet, weil Optimierungszwischenergebnisse schneller als optimal eingestuft werden. Über den Parameter *MaxStartFlag* kann eine Abgrenzung des Parameter *StartFlag* nach oben erfolgen.

Bildung der Kurzzeitnäherung (SPA). Aus den ausgewählten und geladenen Linearisierungsdaten werden über die Funktion `linex01` die zum momentan betrachteten Flugzustand gehörenden Matrizen der Längsbewegung extrahiert. Anschließend erfolgt die Vereinfachung zur Kurzzeitnäherung sowie die Bildung einer Reihenschaltung aus Aktuatorik, Kurzzeitnäherung und Sensorik.

Startwertgenerierung, Festlegung der Parametergrenzen und Algorithmus-Einstellungen. In der Optimierungsschleife, die so lange ausgeführt wird, bis der aktuelle Entwurfspunkt entweder erfolgreich bearbeitet wurde oder eine Maximalzahl an Optimierungsversuchen erreicht wurde, müssen zunächst Startwerte für die Optimierungsparameter festgelegt werden. Dies geschieht im Normalfall, d.h. beim ersten Versuch der Bearbeitung eines Entwurfspunktes in Abhängigkeit der momentanen Optimierungsebene bzw. in Funktion des

Wertes von *StartFlag* über das Skript *StartWertGenerierungV11*. Wie bereits in Abschnitt 4.2.1 beschrieben wurde, erfolgt die Startwertgenerierung gemäß Tabelle 4.2.

<i>StartFlag</i>	Startwertgenerierung
0	verwende Komponenten von <i>InitialParameter</i> als Startwerte
0,5	eingeleseene Startwerte
1	Startwerte aus letztem Optimierungsergebnis
2	Startwerte aus Mittelwerten der letzten Optimierungsergebnisse
3	Startwerte aus Geraden- oder Parabelregression oder aus Mittelwert aus beiden

Tabelle 4.2: Startwertgenerierung in Abhängigkeit des Wertes von *StartFlag*

Die Festlegung der Algorithmus-Einstellungen Abbruchtoleranz, maximale Iterationszahl und ggf. Schrittweite zur Gradientenbildung erfolgen im Normalfall ebenfalls in Abhängigkeit des Wertes von *StartFlag* über das Skript *StartWertGenerierungV11*. Die einzelnen Zahlenwerte der diversen Parameter für die verschiedenen Optimierungsebenen wurden über das Skript *SettingsV11* festgelegt, nun werden die Parameter auf die entsprechenden Werte gesetzt.

Die Festlegung der Parametergrenzen erfolgt im Fall des ersten Berechnungsversuchs über das Skript *LimitsV11*. Dort werden die Beträge der Parameter-Startwerte mit einem von *StartFlag* abhängigen Faktor multipliziert, anschließend werden die Produkte zur Bildung der Untergrenzen von den Startwerten abgezogen sowie zur Bildung der Obergrenzen zu den Startwerten addiert, wodurch startwertabhängige, relative Parametergrenzen entstehen. Die Spezifizierung der Multiplikationsfaktoren für die verschiedenen Optimierungsebenen erfolgt in *SettingsV11*, in *LimitsV11* wird der Faktor in Abhängigkeit vom momentanen Wert von *StartFlag* auf den entsprechenden Wert gesetzt. Ein Mindestabstand von Ober- bzw. Untergrenze vom jeweiligen Startwert garantiert dabei auch für betragsmäßig kleine Startwerte eine Mindestbreite des zulässigen Parameterbereichs.

Im Fall von erneuten Berechnungsversuchen erfolgt unter Umständen eine Modifizierung der generierten Startwerte durch das Skript *RetryStartWertGenerierungV11*, die Parametergrenzen und Algorithmusparameter werden durch das Skript *RetryLimitsTolV11* modifiziert. Ob und wenn ja welche Modifikationen durchgeführt werden, hängt dabei von der Versagensart des Optimierungsversuchs sowie von der Vorgeschichte ab. Die verschiedenen möglichen Aktionen sowie die Prozedur zur Auswahl der zu ergreifenden Maßnahmen werden im Abschnitt *Entscheidungsfindung* erläutert.

Durchführung der eigentlichen Optimierung. Nachdem alle Vorbereitungen für die Optimierung getroffen wurden, erfolgt nun der eigentliche Aufruf des Optimierungsalgorithmus'. Im vorliegenden Fall stehen die Optimierungsroutinen *ifrslsqpXLPuffer* und *fminconUL* zur Verfügung. Beide berechnen mit Hilfe der Funktion *EntwurfFun10* in ihrer zugehörigen Kostenfunktion (*regler_cost10fmincon* bzw. *regler_cost10IFRSLSQP*) die zu optimierende Zielgröße sowie in ihrer Beschränkungsfunktion (*regler_con10fmincon* bzw. *regler_const10IFRSLSQP*) die Einhaltung bzw. Verletzung der Nebenbedingungen. Die beiden Optimierungsfunktionen basieren auf der Matlab-Optimierungsroutine *fmincon* bzw. der IFR-Implementierung des SQP-Verfahrens *ifrslsqp*. Beide Routinen wurden jedoch dahingehend modifiziert, dass

nicht-optimale Zwischenlösungen der Optimierung, die im Sinne der Nebenbedingungen gültig sind, in einen Puffer geschrieben werden. Sobald eine Zwischenlösung mit einem besseren Wert des Zielfunktional gefunden wird, wird der Puffer aktualisiert. Auf diese Art und Weise wird sichergestellt, dass nicht-optimale, aber gültige Lösungen im weiteren Verlauf der Optimierung nicht “verloren” gehen, was aufgrund des zerklüfteten Verlaufs der beschränkten Parameter CAP und ζ_{AS} in Abhängigkeit der Optimierungsparameter ansonsten passieren kann.

Analyse der Ergebnisse. Nach der eigentlichen Optimierung muss das vom Optimierungsalgorithmus gelieferte Ergebnis analysiert werden. Diese Analyse erfolgt bezüglich der folgenden Fragen:

- Wurde ein Optimum gefunden?
- Ist der Ausgang des Optimierungsalgorithmus’ ein gültiges Ergebnis im Sinne der Nebenbedingungen und gilt für das erzielte Stabilitätsmaß tatsächlich $\rho > 1$?
- Trat im Lauf der Optimierung ein Fehler auf?
- Wurden im Lauf der Optimierung Zwischenergebnisse in den Puffer geschrieben?

Im Skript `AnalyseOptimV11` werden die Fragen abgearbeitet, wobei die Antworten in der Variablen `RetryCaseSingle` erfasst und gespeichert werden. Es handelt sich bei dieser Variable um einen String aus 3 Zeichen, deren Werte und Bedeutungen in Tabelle 4.3 zusammengefasst sind:

Zeichen	Wert	Bedeutung
1	1	Optimierung erfolgreich
1	2	kein Optimum gefunden
1	3	Fehler während der Optimierung
2	1	Lösung gültig
2	3	Lösung nicht gültig
2	3	Fehler während der Optimierung
3	1	Wert(e) im Puffer
3	3	Puffer leer

Tabelle 4.3: Zusammensetzung und Bedeutung der Variablen `RetryCaseSingle`

Um bei einem erneuten Versuch nach einem gescheiterten Optimierungsdurchgang den Wert von `RetryCaseSingle` zu erhalten und somit nach eventuell mehreren gescheiterten Versuchen auf die Vorgeschichte des letzten Durchgangs zurückgreifen zu können, wird aus Werten von `RetryCaseSingle` durch Aneinanderhängen die Variable `RetryCaseCum` gebildet. Die einzelnen Dreier-Zeichenketten werden dabei durch Punkte voneinander getrennt. Maximal sind drei Wiederversuche, also insgesamt vier Berechnungsversuche möglich. Falls es sich beim untersuchten Bearbeitungsdurchgang bereits um den dritten Versuch handelt, wird für den Fall, dass gilt $StartFlag = 0$, zur Kennzeichnung zusätzlich die Zeichenkette `’.SF0’` an die Zeichenkette `RetryCaseCum` angehängt. Diese Kennzeichnung ist notwendig, da es unter bestimmten Voraussetzungen sinnvoll ist, nach dem dritten erfolglo-

sen Versuch eine Berechnung mit der Einstellung *StartFlag* = 0 durchzuführen, allerdings nur dann, wenn nicht bereits im dritten Versuch mit *StartFlag* = 0 optimiert wurde.

Entscheidungsfindung. In Abhängigkeit des Parameters *RetryCaseCum* erfolgt die Entscheidung, ob die Bearbeitung des betrachteten Entwurfspunktes als erfolgreich bewertet werden und die Optimierungsschleife damit verlassen werden kann, oder ob ein erneuter Optimierungsversuch gestartet wird bzw. ob die Schleife ohne abgeschlossene Bearbeitung des Entwurfspunktes verlassen werden soll. Im Fall eines erneuten Versuches sind in Abhängigkeit der Art des Versagens der vorangegangenen Optimierungen verschiedene Maßnahmen sinnvoll. Die Steuerung dieser Maßnahmen erfolgt über die Variable *ActionToTake*. Diese bewirkt je nach ihrem Wert im Skript StartwertGenerierungV11 bzw. RetryStartwertGenerierungV11 eine Modifizierung der standardmäßigen Berechnung der Startwerte sowie in RetryLimitsTo1V11 eine Modifizierung der Algorithmusparameter und der Begrenzungen der Optimierungsparameter.

Tabelle 4.4 enthält die verschiedenen Ergebnisse, die bei der Untersuchung von *RetryCaseCum* auftreten können.

Fall	Beschreibung	ActionToTake
1	Optimierung erfolgreich, verlasse Schleife	--
2	Nicht optimal, aber Lösung gefunden, verlasse Schleife da wiederholt aufgetreten.	--
3	Keine Lösung gefunden, nichts im Puffer kein erneuter Versuch, verlasse Schleife da wiederholt aufgetreten.	--
4	Nicht optimal, aber Lösung gefunden bzw. Lösung im Puffer.	ActionA. Erneuter Versuch mit Startwerten aus Puffer, engere Parametergrenzen, größere Toleranz.
5	Fehler während der Optimierung	ActionB. Erneuter Versuch mit gleichen Startwerten und weiteren Parametergrenzen.
5.1	Keine Lösung gefunden, nichts im Puffer, <i>StartFlag</i> > 0	ActionC. Erneuter Versuch mit Startwerten aus Methode <i>StartFlag</i> - 1, weitere Parametergrenzen.
5.2	Keine Lösung gefunden, nichts im Puffer, <i>StartFlag</i> = 0	ActionD. Erneuter Versuch mit <i>StartFlag</i> = 0 (Start-Startwert, Standardgrenzen, Standardtoleranz)

Tabelle 4.4: Ergebnisse der Entscheidungsfindung

Hierbei ist zu beachten, dass der jeweilige Fall wie bereits erwähnt von der Vorgeschichte des aktuellen Versuchs abhängt. So kommt es beispielsweise beim Erreichen der maximalen Anzahl von Berechnungsversuchen im Fall eines Misserfolgs der Berechnung zum Verlassen der Schleife, während es andernfalls zu einem Wiederversuch kommt. Genauso führt die Berechnung einer gültigen, jedoch nicht optimalen Lösung zu einem Wiederversuch, falls dieses Ergebnis zum ersten Mal auftritt, beim wiederholten Mal kommt es jedoch zum Verlassen der Schleife, weil ein erneuter Versuch dann als nicht sinnvoll betrachtet wird. Die

Zuordnung der verschiedenen Werte von *RetryCaseCum* zu den verschiedenen Fällen erfolgt gemäß einer Wiederversuchslogik entsprechend der Baumstruktur in Anhang B.3. Die verschiedenen Fälle an sich sind in Tabelle 4.4 zusammengefasst. Neben dem Wert der Variablen *ActionToTake* für die Fälle, in denen es zu einem Wiederversuch kommt, enthält die Tabelle eine kurze Beschreibung des Falls sowie ggf. der zu ergreifenden Maßnahmen. Über das Skript *DecisionMessageV11* wird ggf. eine Meldung über die ergriffenen Maßnahmen generiert.

Aktualisierung des Informations-Textfiles. Nach dem Verlassen der Optimierungsschleife wird das Ergebnis der Bearbeitung des Entwurfspunktes durch das Skript *CreateTextInfoV11* in ein Textfile geschrieben. Dieses Textfile kann unabhängig von Matlab geöffnet werden, wodurch auch bei voller Auslastung von Matlab jederzeit Informationen über den Stand einer laufenden Berechnung abgerufen werden können.

Ausgabe der Berechnungsergebnisse beim Anstellwinkelwechsel. Nachdem alle Entwurfspunkte eines Anstellwinkels abgearbeitet worden sind, erfolgt über das Skript *DisplayResultsV11* eine Ausgabe der Ergebnisse.

Abspeichern der Berechnungsergebnisse. Ebenfalls nach Bearbeitung aller Entwurfspunkte eines Anstellwinkels, das heißt am Ende jedes Durchgangs der Anstellwinkelschleife, werden die Berechnungsergebnisse durch das Skript *SaveOptimDataV11* gespeichert. Hierbei werden die Ergebnisse zunächst als Workspace-Datei gespeichert. Der Name der Datei wird dabei automatisch generiert und basiert auf dem Namensstamm der zur Bearbeitung ausgewählten Linearisierungsdaten. Anschließend wird die Datei komprimiert und zu einem Zip-Archiv hinzugefügt, dessen Name ebenfalls auf dem Namen der Linearisierungsdaten basiert.

Fortsetzungsmodus. Unterbrochene Bearbeitungskampagnen können am ersten Entwurfspunkt nach dem letzten vollständig abgearbeiteten Anstellwinkel fortgesetzt werden. Hierbei bietet die Routine die Möglichkeit, im Falle der Start-Startwertgenerierung nach Methode 3 (siehe Tabelle 4.1) das letzte Ergebnis des vorangegangenen Anstellwinkels als Startwert einzulesen. Des Weiteren werden im Fall der Fortsetzung einer unterbrochenen Berechnung die beiden angelegten Zip-Archive über das Skript *ZipMergeV11* zusammengefasst.

Eintragung ins LogFile. Nach erfolgreicher Beendigung des Entwurfsdurchlaufs erfolgt ein entsprechender Eintrag ins Excel-LogFile, sofern beim Beginn der Routine einer Eintragung zugestimmt wurde.

4.2.3 Tabellenerstellung

Durch Ausführung des Skriptes *tabcreate01* werden aus den Ergebnisdaten eines automatisierten Reglerentwurfes Reglertabellen erstellt. Hierbei handelt es sich zum einen um dreidimensionale Tabellen, die die variablen Reglerelemente für die Punkte des Entwurfsrastrers enthalten. Zum anderen werden zweidimensionale Tabellen erstellt, die für jedes variable Element des Reglers die Parameter der zugehörigen Regressionsparabel und -geraden in Abhängigkeit vom Anstellwinkel und der Machzahl enthalten. Die Bestimmung, welche Elemente des Reglers variabel, welche identisch Null und welche konstant sind, kann wahlweise vorgegeben oder automatisch aus den Ergebnisdaten bestimmt werden. Hierzu werden die

Verläufe aller Reglermatrizenelemente für verschiedene Entwurfspunkte hinsichtlich ihrer Mittelwerte und Streuungen untersucht. Falls die Streuung eine festlegbare Toleranz überschreitet, wird das betreffende Element als variabel deklariert, ansonsten wird ihm der als konstant betrachtete Mittelwert zugewiesen.

4.2.4 Auswertung und Nachbearbeitung

Eine schnelle Möglichkeit zur Untersuchung der Ergebnisse eines automatisierten Reglerentwurfes bietet das Skript `RegressionsKontrolle3d`. Die Routine untersucht für jeden Punkt des Entwurfsrasters, ob ein Regler entworfen werden konnte, und bestimmt ggf. das zugehörige Robustheitsmaß ρ sowie die Werte für CAP und ζ_{AS} . Für jedes variable Reglermatrizenelement wird zusätzlich der Flugzustand bestimmt, für den sich die maximale Abweichung ergibt zwischen dem exakten Wert und einem durch Parabel- bzw. Geradenregression bestimmten Wert. Neben einer Ausgabe der Entwurfspunkte mit minimalem Robustheitsmaß, der maximalen Abweichung der Werte CAP und ζ_{AS} von den Vorgabewerten sowie den maximalen Abweichungen zwischen exaktem Wert und Regressionswert erfolgt durch den automatischen Aufruf des Skriptes `RegressionsKontrolleVisual` eine graphische Darstellung der Entwurfsergebnisse.

Punkte, für die trotz mehrmaliger Wiederversuche kein Regler entworfen werden konnte, werden ebenfalls ausgegeben. Für diese Entwurfspunkte ist eine manuelle Nachbearbeitung notwendig. Die Skripte `ValRepCalc11` und `ValRep11` bieten hierzu eine bequeme Möglichkeit. Nach der Spezifizierung des zu korrigierenden Entwurfspunktes sowie der Auswahl des zu korrigierenden Entwurfsdaten-Workspaces in `ValRepCalc11` generiert das Skript eine graphische Darstellung des Verlaufs der Werte der Entwurfparameter p_1 , p_2 und p_3 der Wichtungsfunktionen W_1 und W_2 in der Umgebung des Entwurfspunktes. Das heißt, der Verlauf der Entwurfparameter wird für den Anstellwinkel und die Machzahl des Entwurfspunktes für die verschiedenen zugehörigen Staudruckwerte in drei Plot-Fenstern dargestellt. Durch Betätigung der linken Maustaste wird nun der zu editierende Entwurfspunkt markiert, durch Betätigung der rechten Maustaste wird der Entwurfparameterwert des markierten Punktes an die Stelle des Mauszeigers gesetzt. Zum Wechseln zwischen den drei Plot-Fenstern, d.h. zum Wechseln zwischen den drei Entwurfparametern können entweder die Pfeiltasten (\uparrow bzw. \downarrow) oder die mittlere Maustaste verwendet werden. Durch die Pfeiltasten (\leftarrow bzw. \rightarrow) wird der editierbare Entwurfspunkt gewechselt. Nach jedem Setzen eines Parameterwertes durch die rechte Maustaste erfolgt ein Reglerentwurf mit den entsprechenden Parameterwerten und die berechneten Werte für ρ , CAP , ζ_{AS} und die Anstiegszeit t_{Rise} werden ausgegeben. Zusätzlich wird in den Plot-Fenstern die zu den bisherigen Parameterwerten gehörende Regressionsparabel bzw. -gerade dargestellt. Somit können auf relativ bequeme Art und Weise Entwurfparameter für die fehlenden Entwurfspunkte bestimmt werden, die einerseits die Anforderungen an den Reglerentwurf bezüglich Robustheit und Handling-Qualities erfüllen und die sich zudem in den Trend der Entwurfparameter in Richtung des Staudrucks einfügen. Das Auswählen von Entwurfspunkten und Setzen der entsprechenden Entwurfparameter wird durch Betätigung der Enter-Taste beendet. Anschließend erfolgt nach der Bestätigung einer Sicherheitsabfrage die Aktualisierung der Reglerentwurfsdaten-Workspaces sowie der Reglerdaten-Tabellen über das Skript `ValRep11`.

Stellt sich beim Test von automatisch entworfenen Reglern heraus, dass das Entwurfsraster in Richtung des Anstellwinkels zu grob gewählt wurde, können die zum Entwurf gehörenden Reglerdaten-Tabellen über das Skript `TabMerge` nachträglich erweitert werden. Hierzu erfolgt über die Routine `regler_opt11DivAlgAuto` der Entwurf von Reglerdaten für die einzufügenden Anstellwinkel. Über `TabCreate01` werden die entsprechenden Reglerdatentabellen erzeugt, die abschließend über `TabMerge` in die zu erweiternden Reglerdaten-Tabellen eingefügt werden.

Abbildung 4.10 zeigt die Auswertung der Ergebnisse der Variante *V205* des automatischen Reglerentwurfs für das in Abschnitt 3.2.3 definierte Entwurfspunkteraster nach erfolgter Nachbearbeitung. Die Optimierungseinstellungen für Variante *V205* sind in der Tabelle in Abbildung E.1 in Anhang E zusammengefasst.

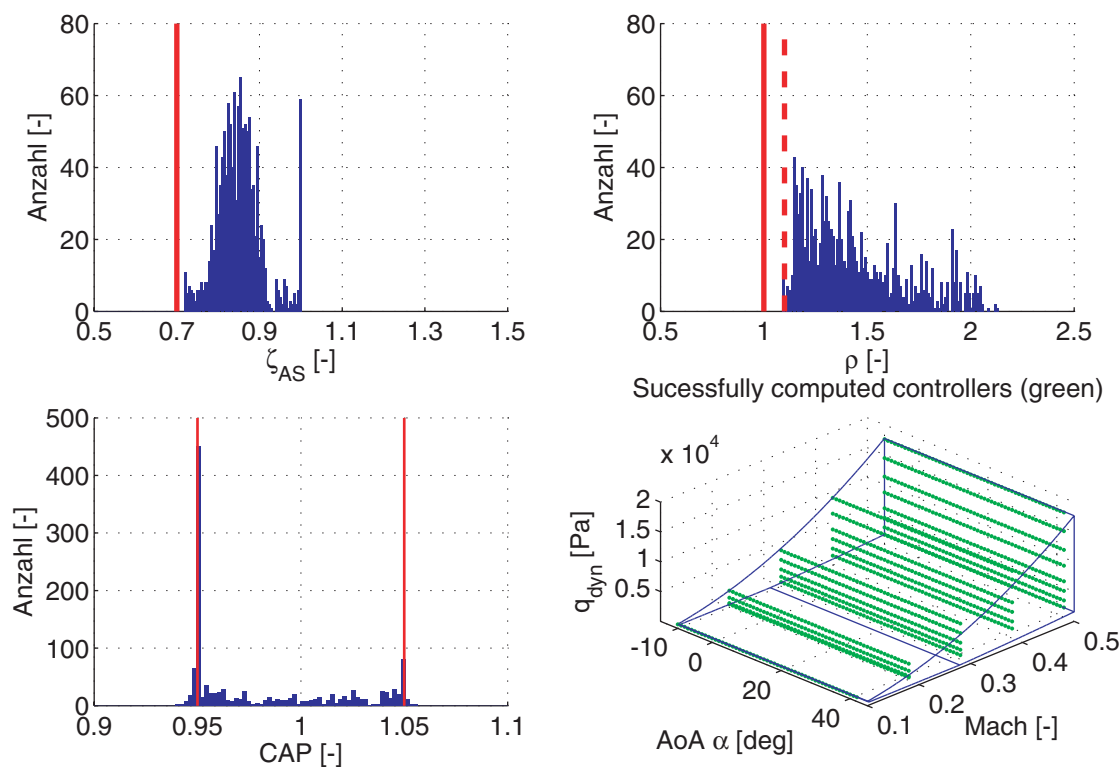


Abb. 4.10: Auswertung der Ergebnisse automatischer Reglerentwurf *V205*

Dargestellt sind drei Histogramme mit der Verteilung der Werte für die Handling-Kriterien ζ_{AS} und CAP sowie das Robustheitsmaß ρ . Die einzelnen Histogramme zeigen, dass die Regelkreise an allen Entwurfspunkten wie verlangt eine Dämpfung von $\zeta_{AS} > 0,7$ aufweisen.

Ebenso wurde an allen Entwurfspunkten ein Robustheitsmaß von $\rho > 1$ erreicht, der Großteil der entworfenen Regler weist ein Robustheitsmaß von $\rho > 1,1$ auf (gestrichelte Linie im entsprechenden Schaubild), nur 1,1% der Entwurfspunkte weisen einen Wert von $\rho < 1,1$ auf, für alle Regler gilt jedoch $\rho > 1,08$. Die bei den Entwurfseinstellungen festgelegte Untergrenze des Robustheitsmaßes von $\rho_{min} = 1,15$ gemäß Abschnitt 4.1.2 wird hierbei aufgrund der manuell nachbearbeiteten Entwurfspunkte verletzt. Diese Verletzung wurde zu

Gunsten einer verbesserten Form der Sprungantwort in Form einer Verbesserung des Überschwingverhaltens sowie der Anstiegszeit toleriert.

Im Falle des Control Anticipation Parameters liegen nahezu alle erzielten Werte innerhalb des verlangten Bereichs von $0,95 \leq CAP \leq 1,05$. Lediglich ca. 1% aller *CAP*-Werte liegt außerhalb des Intervalls $0,945 \leq CAP \leq 1,055$ und alle Werte liegen innerhalb des Intervalls $0,94 \leq CAP \leq 1,06$. Dementsprechend werden die Regler aller Entwurfspunkte als erfolgreich entworfen klassifiziert und in der dreidimensionalen Envelope im vierten Schaubild von Abbildung 4.10 als grüne Punkte dargestellt.

4.2.5 Kurzzusammenfassung des Entwurfsprozesses (Bedienungsanleitung)

Es folgt abschließend eine Kurzzusammenfassung der Schritte des automatischen Reglerentwurfes.

1. Spezifizierung eines Entwurfsrasters und Erzeugung von Trimmdateien über `trimenv-dyn`.
2. Erzeugung zugehöriger Linearisierungsdateien über `calculation_lin10_Auto`.
3. Setzen aller Einstellungen des automatischen Reglerentwurfes in `SettingsV11`.
4. Starten des automatischen Reglerentwurfes durch Ausführung des Skriptes `regler_opt11DivAlgAuto`.
5. Überprüfung der automatisch erzeugten Reglerdateien über das Skript `RegressionsKontrolle3d`.
6. Erzeugung von Reglerdatentabellen über `tabcreate01`.
7. Ggf. Korrektur von Reglerentwurfsdateien und Reglerdatentabellen über `ValRepCalc11`.
8. Ggf. Einfügen bzw. Ersetzen von Daten in Reglerdatentabellen über das Skript `Tab-Merge`.

4.3 Zusammenbau zum linearen Gain-Scheduling-Regler

Im vorangegangenen Abschnitt wurde gezeigt, wie für ein gegebenes Raster von Entwurfspunkten lineare Einzelregler entworfen werden. Das vorliegende Kapitel beschreibt den Zusammenbau dieser Einzelregler zu einem linearen Regler mit Gain-Scheduling.

4.3.1 Implementierung in Matlab und Simulink

Entsprechend Abschnitt 3.2 dienen die Parameter der linearen Einzelregler als Stützstellen mehrdimensionaler Tabellen, aus denen in Abhängigkeit des Anstellwinkels, der Machzahl sowie wahlweise des Staudrucks oder der Flughöhe die zum momentanen Flugzustand gehörenden Reglerparameter interpoliert werden. Zur Implementierung dieses Interpolationsverfahrens in Simulink werden die Matrizen des Reglers aus Einzelblöcken aufgebaut. Für Matrixelemente, deren Wert Null oder konstant ist, werden hierzu Blöcke vom Typ *Source*

verwendet, variable Matrizenelemente werden wahlweise durch dreidimensionale Tabellen-Interpolationsblöcke vom Typ *Interpolation (n-D) using PreLook-Up* für eine klassische dreidimensionale Interpolation oder vom Typ *Plane-Interpol_3D* für eine Berechnung basierend auf der Dreiecks-Ebenen-Methode nach Abschnitt 3.2.2 realisiert. In beiden Fällen werden die Indices der zur Interpolation verwendeten Stützstellen getrennt von der eigentlichen Interpolation berechnet, denn die Indices sind für alle zu berechnenden Matrizenelemente dieselben. Um den Berechnungsaufwand zu reduzieren ist es daher sinnvoll, die Indices für alle Matrizenelemente gemeinsam zu berechnen, anstatt sie für jedes Element einzeln zu bestimmen. Die Bestimmung der Indices sowie der zur Interpolation benötigten Abstände des Interpolationspunktes zu den Stützpunkten erfolgt über die Blöcke *PreLook-Up Index Search* bzw. *TriSel2D+1*. Anschließend werden Indices (und Abstände) an alle Interpolationsblöcke übergeben, anstatt wie sonst üblich lediglich die Interpolationspunkt-Koordinaten an die Interpolationsblöcke zu übergeben. Abbildung 4.11 illustriert diese Vorgehensweise.

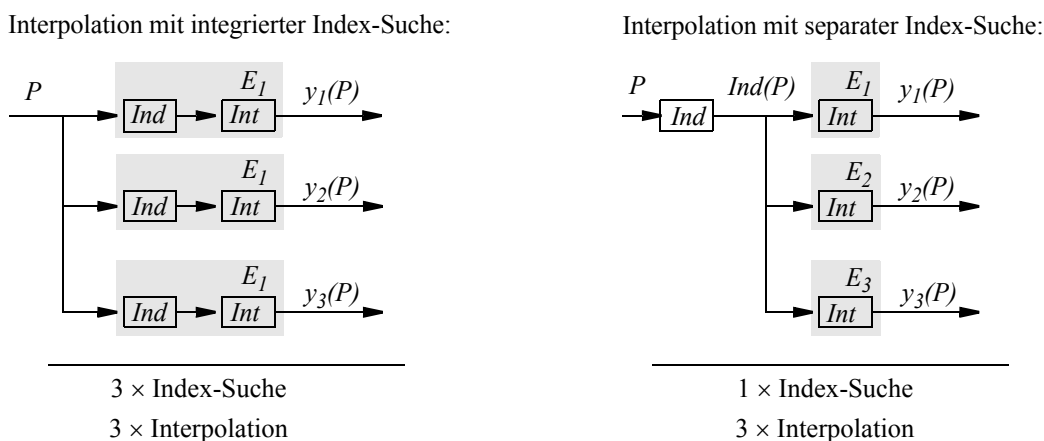


Abb. 4.11: Interpolation mit integrierter und separater Index-Suche

Alternativ zu dreidimensionalen Interpolationsblöcken können zum Aufbau der Reglermatrizen auch Blöcke vom Typ *ParabelRegression* bzw. *GeradenRegression* verwendet werden. In diesem Fall erfolgt die Bestimmung der Matrizenelemente in Abhängigkeit des Flugzustands über die Parabel- bzw. Geradenregressionsmethode entsprechend Abschnitt 3.2.2. (Anmerkung: Außer bei den Standard-Simulink-Blöcken *PreLook-Up Index Search* und *Interpolation (n-D) using PreLook-Up* handelt es sich bei den erwähnten Blöcken um Simulink-Blöcke, die im Rahmen dieser Arbeit erstellt wurden und auf c-mex s-functions bzw. gewöhnlichen Matlab-Functions beruhen).

Die Erstellung der Simulink-Modelle der Reglermatrizen A , C , F und H erfolgt über das Skript `Matrix_createXLR13`. In diesem Skript ist zunächst zu spezifizieren, welches Verfahren zur Bestimmung der variablen Matrizenelemente verwendet werden soll (Interpolation oder Regression). Außerdem ist ein Workspace mit Regler-Tabellendaten anzugeben, für den die Reglermatrizen gebildet werden sollen. Dies ist notwendig, da die Tabellendaten-Workspaces die Variablen `MatAStruct`, `MatCStruct`, `MatFStruct` und `MatHStruct` enthalten. Hierbei handelt es sich um Structure-Arrays, die die gleichen Dimensionen wie die zugehörigen Reglermatrizen aufweisen und die in ihren Elementen die Informationen beinhalten, ob das entsprechende Element der entsprechenden Reglermatrix variabel oder kon-

stant ist. Im Falle der variablen Matrizenelemente enthalten die Structure-Arrays den Buchstaben 'V'. Im Fall von konstanten Elementen enthalten sie den entsprechenden Wert.

Aus den über `Matrix_createXLR13` generierten Simulink-Modellen für die Reglermatrizen kann nun entsprechend Abbildung 3.4 in Abschnitt 3.1.4 der lineare Regler in Form eines Beobachters mit Zustandsvektor-Rückführung aufgebaut werden.

Alternativ zu einer Realisierung des Reglers in Simulink besteht die Möglichkeit, die Matrizen einer Zustandsraumdarstellung des geschlossenen Regelkreises für einen bestimmten Flugzustand in Matlab aufzustellen, um anschließend über die Befehle `step` oder `lsim` (ohne die Verwendung eines Simulink-Modells) die Sprungantwort des linearen Systems am betreffenden Flugzustand zu simulieren. Hierzu werden ebenfalls ausgehend von den Struktur-Informationen-Arrays *MatAStruct*, *MatCStruct*, *MatFStruct* und *MatHStruct* die Reglermatrizen erstellt. Variable, vom betrachteten Flugzustand abhängige Matrizenelemente werden wiederum entsprechend der angegebenen Methode (klassische 3D-Interpolation bzw. Dreiecks-Ebenen-Methode oder Parabel- bzw. Geraden-Regression) berechnet. Die Werte der Parameter P_1 , P_2 und P_3 zur Bestimmung der Übertragungsfunktionen der Wichtungsfunktionen W_1 und W_2 werden analog berechnet.

4.3.2 Simulation des linearen Regelkreises

Die Simulation des linearen Regelkreises erfolgt über das Skript `ControllerCheckLinCalc`. In diesem Skript werden zunächst entweder konkrete Flugzustände spezifiziert, für die eine Simulation durchgeführt werden soll, oder es werden Ober- und Untergrenzen für den Anstellwinkel, die Machzahl sowie den Staudruck angegeben. Aus diesen Bereichen wird eine vorgegebene Anzahl von Flugzuständen mit zufälligen Werten für Anstellwinkel, Machzahl und Staudruck bestimmt. Anschließend muss der Workspace ausgewählt werden, der die Reglerdaten-Tabelle enthält, für die die Simulation durchzuführen ist. Des Weiteren muss die Interpolationsart, mit der im Lauf der Simulation die Reglerparameter zu bestimmen sind, angegeben werden.

Für jeden zu simulierenden Flugzustand wird anschließend die Kurzzeitnäherung ermittelt, indem das nichtlineare Flugzeugmodell zunächst für den Flug auf einem quasi-stationären Abfangbogen getrimmt und anschließend linearisiert wird. Anschließend erfolgt über die Funktion `ControllerCheckLinFun` die Berechnung der Übertragungsfunktion des offenen sowie des geschlossenen Regelkreises. Wie im vorangegangenen Abschnitt 4.3.1 beschrieben, werden in der Funktion hierzu in Abhängigkeit der ausgewählten Art zur Berechnung der Werte variabler Matrizenelemente (Interpolation oder Regression) die Reglermatrizen für den momentan betrachteten Flugzustand berechnet. Es ist zu beachten, dass neben den Reglermatrizen A , C , F und H auch der Vorfilter V sowie die Wichtungsfunktionen W_1 und W_2 berechnet werden müssen. Hierzu werden zunächst über die entsprechende Berechnungsmethode (Interpolation oder Regression) die Werte der Parameter P_1 , P_2 und P_3 der Wichtungsfunktionen in Abhängigkeit des Flugzustands berechnet, danach werden die Übertragungsfunktionen der Wichtungsfunktionen gebildet. Die Berechnung des Vorfilters V erfolgt je nach Implementierung des Reglers entsprechend Gleichung 3.27 bzw. 3.31 in Abschnitt 3.1.4.

Der Regelkreis wird ausgewertet hinsichtlich seiner Stabilität, seines CAP -Wertes, der Dämpfung der Anstellwinkelschwingung ζ_{AS} sowie seines Stabilitätsmaßes ρ . Des Weiteren erfolgt die Simulation eines Einheitsprunges am Eingang des kommandierten Anstellwinkels sowie eine entsprechende Auswertung hinsichtlich der relativen Überschwingweite $\delta_{\alpha,rel}$ sowie der Anstiegszeit $t_{\alpha,90}$, bis der Anstellwinkel 90% des kommandierten Wertes erreicht hat. Zur Berechnung von CAP und ζ_{AS} wird dieselbe Methode verwendet wie bei der automatischen Berechnung der Reglerdaten, auf denen die betrachtete Reglerdatentabelle basiert. Alternativ hierzu können verschiedene Berechnungsmethoden zur CAP -Bestimmung verwendet werden, darunter auch die *BestCAP*-Methode, bei der der CAP -Wert nacheinander über mehrere Methoden bestimmt wird. Anschließend wird das am nächsten am Zielwert $CAP = 1$ liegende Ergebnis als CAP -Wert verwendet.

Zur Untersuchung des Einflusses von verschiedenen Methoden zur Berechnung der Reglermatrizenelemente besteht die Möglichkeit, das Skript `ControllerCheckLinCalc` in einem Batch-Modus auszuführen. In diesem Fall muss die Variable `Loops` auf einen Wert $Loops \geq 1$ gesetzt werden. Die Simulation der angegebenen Anzahl von Flugzuständen erfolgt dann insgesamt $Loops$ Mal. Nach jeder Simulationskampagne werden die Simulationsergebnisse abgespeichert. Hierzu ist mit der Variablen `SaveNameCommonPart` ein gemeinsamer Teil der Workspace-Namen anzugeben, unter denen die Daten nach Erweiterung durch die Nummer des aktuellen Durchgangs abgespeichert werden. Damit nicht bei jedem Durchgang eine Abfrage nach der zu simulierenden Reglerdaten-Tabelle erfolgt, muss außerdem über die Variable `BMWSNum` die Nummer der zu untersuchenden Reglerdaten-Tabelle angegeben werden. Der entsprechende Workspace wird dann über die Funktion `SelectLoadDef` ohne weiteres Nachfragen geladen.

4.3.3 Darstellung der Simulationsergebnisse

Nach Berechnung der Sprungantworten für alle spezifizierten Flugzustände erfolgt automatisch ein Aufruf des Skriptes `ControllerCheckLinVis`. Dieses Skript generiert eine Ausgabe im Kommando-Fenster von Matlab mit Informationen über die erreichten Minimalwerte des Robustheitsmaßes ρ , der maximalen Abweichungen der CAP -Werte vom Sollwert $CAP = 1$ und den kleinsten aufgetretenen Dämpfungen ζ_{AS} der Anstellwinkelschwingung. Darüber hinaus erfolgt ggf. eine Ausgabe von Flugzuständen mit instabilem Regelkreis sowie die Angabe der durchschnittlichen Anstiegszeit und des Flugzustands mit der schnellsten und der langsamsten Sprungantwort.

Neben einer textbasierten Ausgabe der Simulationsergebnisse werden in `ControllerCheckLinVis` diverse Graphiken generiert. Für sämtliche Sprungantworten werden der zeitliche Verlauf des Anstellwinkels α , der Nickrate q sowie des Lastvielfachen n_z dargestellt. Verteilungen der ermittelten Werte für CAP , ζ_{AS} , der Anstiegszeit t_{90} und der relativen Überschwingweite $\delta_{\alpha,rel}$ werden generiert. Die Pole der simulierten Regelkreise werden in der komplexen Zahlenebene dargestellt, wobei die als dominant ermittelten Pole gesondert gekennzeichnet werden.

4.4 Auswertung der Entwurfsversion V205

Im folgenden Abschnitt erfolgt eine Auswertung des manuell nachbearbeiteten automatischen Reglerentwurfs V205. Die Auswertung erfolgt mittels der in den Abschnitten 4.3.2 und 4.3.3 beschriebenen Routinen für verschiedene Sets von jeweils 101 zufällig verteilten Flugzuständen. Ihre Definition erfolgt über die Angabe von Begrenzungen für den Anstellwinkel, die Machzahl sowie den Staudruck. Die entsprechenden Ober- und Untergrenzen sind in Tabelle 4.5 zusammengestellt.

Set Nr.	Anstellwinkel α [deg]		Machzahl Ma [-]		Staudruck q_{dyn} [Pa]	
	min	max	min	max	min	max
1	-10	40	0,1	0,5	1000	
2	-10	40	0,1	0,5	5000	
3	-10	40	0,1	0,5	10000	
4	-10	40	0,21		500	17571
5	-10	40	0,39		500	17571
6	-10	40	0,30		500	17571

Tabelle 4.5: Definition der untersuchten Sets zufälliger Flugzustände

In Abbildung 4.12 sind die zu den Sets gehörenden Flugzustände in verschiedenen Farben in die Envelope des Jettrainers eingezeichnet. Links im Bild befindet sich die Envelope in einer zweidimensionalen Darstellung des Staudrucks (oben) bzw. der Flughöhe (unten) über der Machzahl, im rechten Teil der Abbildung sind die beiden Envelopedarstellungen zusätzlich mit dem Anstellwinkel als dritter Achse versehen.

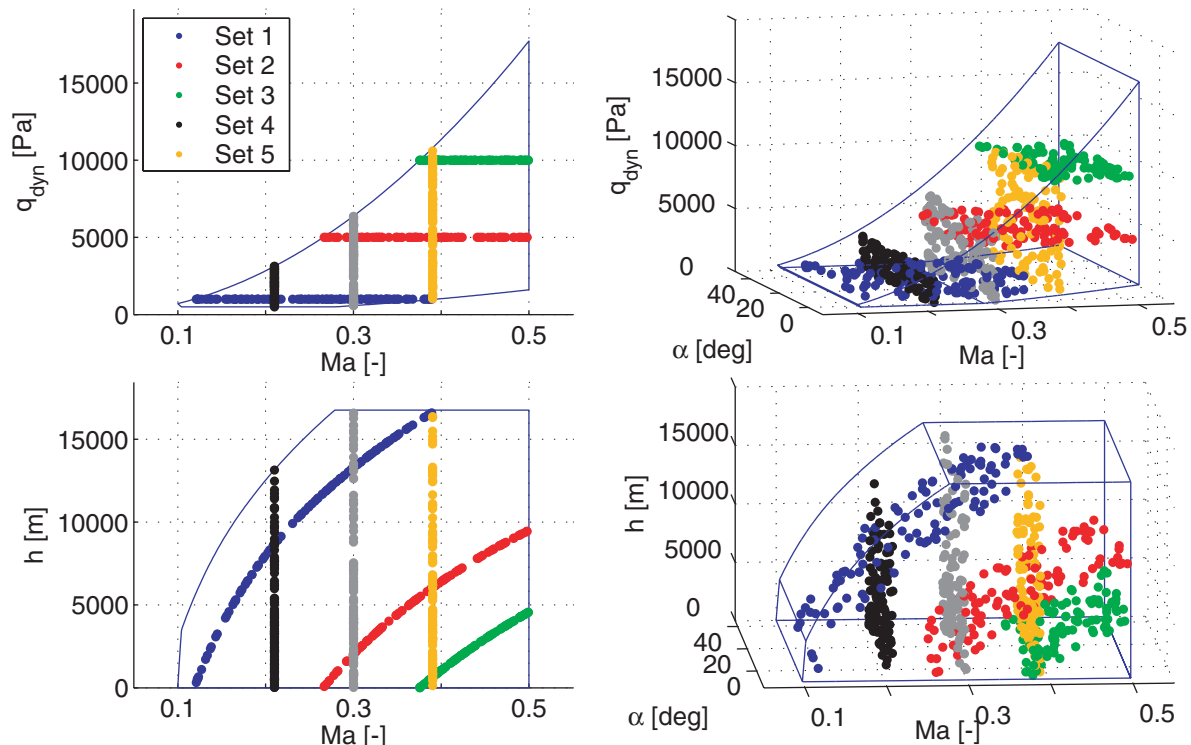


Abb. 4.12: Darstellung der untersuchten Sets zufälliger Flugzustände

Die Auswertung der Simulationskampagnen erfolgt hinsichtlich der sich ergebenden Sprungantworten, der Stabilitätsreserve ρ sowie der Handling-Kriterien CAP und Anstellwinkelschwingungsdämpfung ζ_{AS} . Es erfolgt hierbei eine Beschränkung auf die Betrachtung von Reglern, deren Matrizenelemente über die Dreiecks-Ebenen-Methode berechnet werden, denn für die Reglerparameterbestimmung über die Geraden- bzw. Parabelregressionsmethode ergeben sich für die gegebenen Sets zufälliger Flugzustände einzelne instabile Regelkreise. Die Parabel- und Geradenregressionsmethode wird deshalb an dieser Stelle nicht weiter betrachtet bzw. ausgewertet.

4.4.1 Sprungantworten

Die sich bei der Simulation der Sets 1 bis 6 ergebenden Sprungantworten sind in Abbildung 4.13 dargestellt. Zur leichteren Orientierung wurde die Farbgebung von Abbildung 4.12 übernommen. Es zeigt sich, dass alle Regelkreise asymptotisch stabil sind und dass die prinzipielle Form aller Sprungantworten ähnlich ist. Die Sprungantworten einzelner Sets, das heißt für konstante Staudrücke (Set 1 bis 3) bzw. konstante Machzahlen (Set 4 bis 6) haben größtenteils vergleichbare Anstiegszeiten (Rise-Time) und Überschwingweiten (Overshoot). Eine quantitative Auswertung der Anstiegszeiten und Überschwingweiten führt zu Tabelle 4.6. Durchschnittlich liegen rund 75% der Anstiegszeiten aller simulierten Punkte innerhalb eines Bereiches von $t_{90} = 1\text{ s}$ und $t_{90} = 3\text{ s}$. Etwa 84% aller simulierten Sprungantworten weisen eine Überschwingweite von $uw \leq 10\%$ auf. Bei der Betrachtung der Sprungantworten zeigt sich jedoch auch, dass bei sämtlichen Sets Flugzustände mit teilweise deutlich sichtbaren Ausreißern hinsichtlich Anstiegszeit und Überschwingweite auftreten.

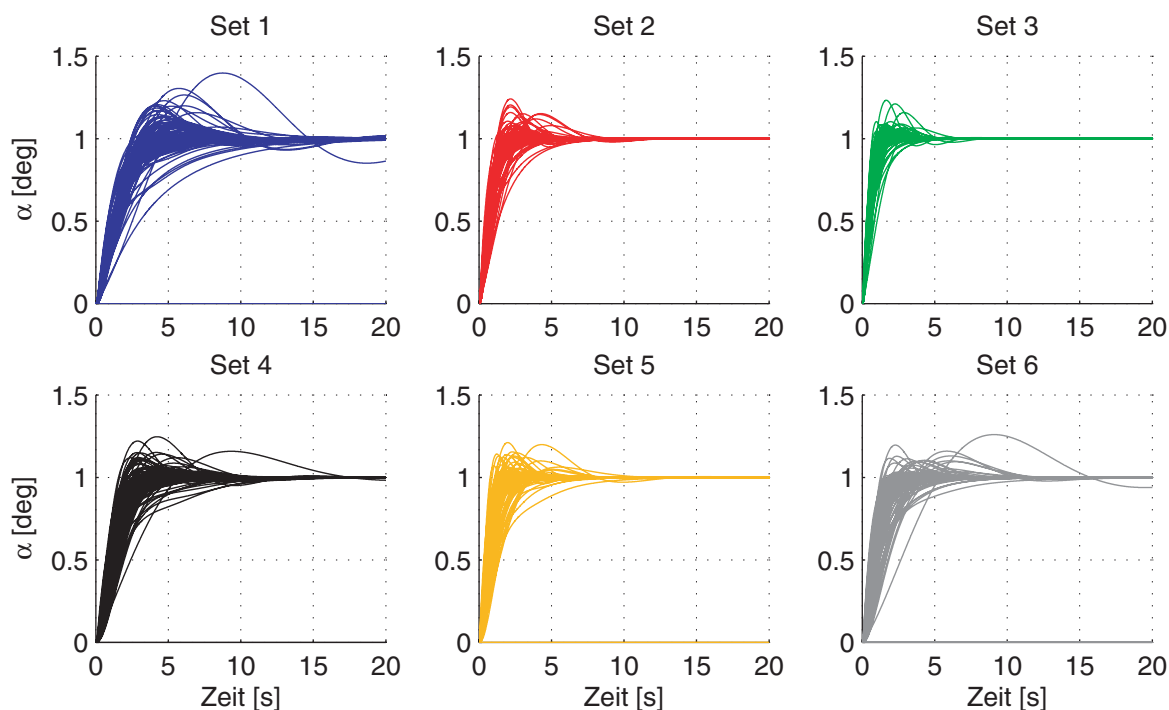


Abb. 4.13: Simulation der Sprungantwort der Sets zufälliger Flugzustände

Prinzipiell ist zur Beseitigung dieser Ausreißer eine Nachbereitung der Reglerentwurfsergebnisse über das Skript `ValRepCalc11` entsprechend Abschnitt 4.2.4 denkbar. Im vorliegenden Fall erfolgte eine derartige Nachbereitung jedoch bereits und als weiterführende Maßnahme kommt daher nur eine Verfeinerung des Reglerentwurfsrasters in Frage. Hinsichtlich der daraus resultierenden, unpraktikabel großen Reglerdatentabellen wird auf diese Maßnahme jedoch verzichtet.

Merkmal		Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Mittelwert
Risettime t_{90} [s]	$t_{90} < 1$	0,0 ^{a)}	2,0	32,7	0,0	10,9	0	7,6
	$1 \leq t_{90} \leq 3$	46,5	96,0	67,3	74,3	86,1	82,2	75,4
	$t_{90} > 3$	53,5	2,0	0,0	25,7	3,0	17,8	17,0
Overshoot uw [%]	$uw \leq 10$	69,4	86,4	94,2	82,9	85,5	87,2	84,1
	$uw > 10$	30,6	13,6	5,8	17,1	14,5	12,8	15,9
	$uw > 15$	18,1	7,4	3,5	6,1	4,0	3,8	8,0
	$uw > 20$	6,9	2,5	2,3	2,4	1,3	1,3	3,0

a) alle Angaben in Prozent

Tabelle 4.6: Auswertung Sprungantworten hinsichtlich Anstiegszeiten und Überschwingweiten

Abbildung 4.14 zeigt eine grobe Zuordnung der erzielten Werte für Anstiegszeit und Überschwingweite zu den betrachteten Flugzuständen.

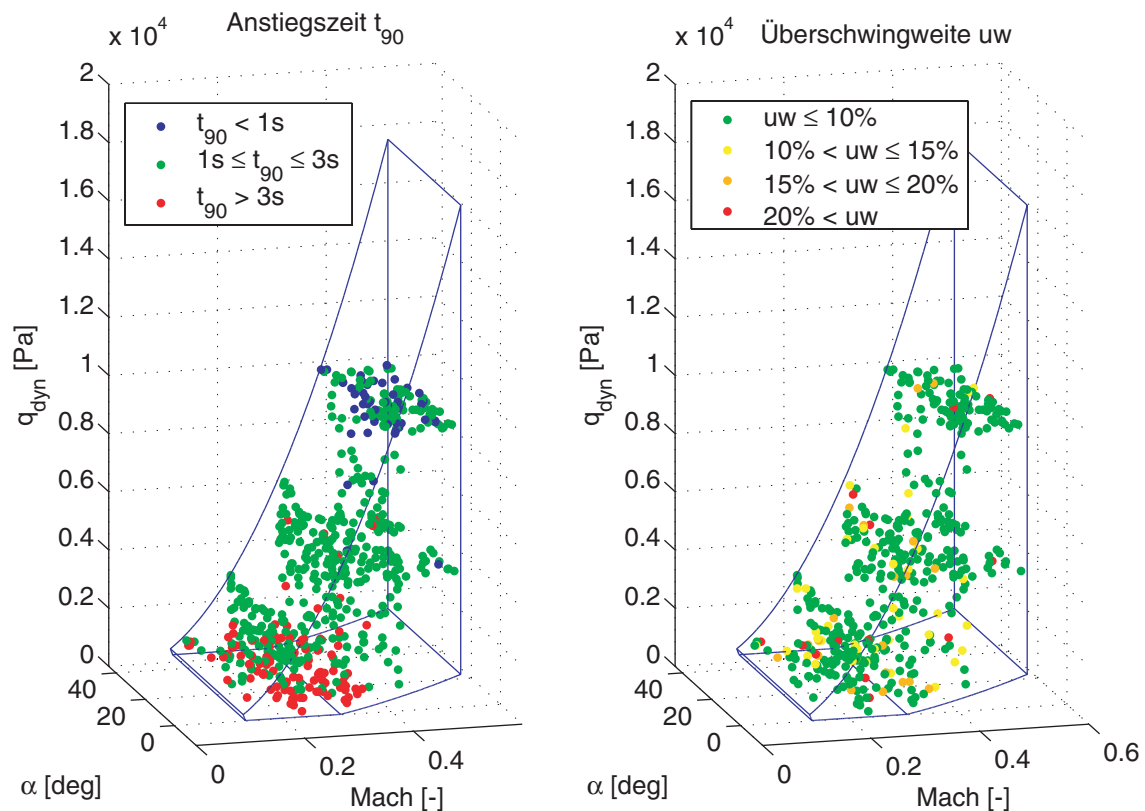


Abb. 4.14: farbliche Kennzeichnung simulierter Flugzustände

Es ist wiederum eine dreidimensionale Envelope dargestellt, in der die simulierten Flugzustände eingetragen sind. Es erfolgt eine farbliche Kennzeichnung des Flugzustands in Abhängigkeit der zum Flugzustand gehörenden Anstiegszeit bzw. Überschwingweite.

Wie sich bereits in Tabelle 4.6 zeigte wird hier ebenfalls deutlich, dass es sich bei Flugzuständen mit größerer Anstiegszeit hauptsächlich um Flugzustände mit niedrigem Staudruck handelt (Set 1). Dies war insofern zu erwarten, da sich bei Flugzuständen mit niedrigem Staudruck verhältnismäßig kleine Werte für n_α ergeben. Dementsprechend müssen sich zur Einhaltung der Vorgabe $CAP = 1$ auch kleine Werte für die Eigenfrequenz ω_{AS} der Anstellwinkelschwingung ergeben. Dementsprechend sind die Anstellwinkelantworten in den betreffenden Flugzuständen langsamer als im hohen Staudruckbereich. Dort kommt es wiederum aufgrund von höheren Werten von n_α analog zu schnelleren Sprungantworten (blau gekennzeichnet). Bezüglich der Überschwingweite sind keine derart deutlichen Tendenzen zu erkennen. Allerdings zeigt sich in Tabelle 4.6 und Abbildung 4.14, dass verstärktes Überschwingen vor allem im Set 1, also bei niedrigem Staudruck, auftritt.

4.4.2 Robustheitsmaß

Eine Übersicht über die für die einzelnen Sets erzielten Robustheitsmaße ist in Abbildung 4.15 zusammengestellt. Es zeigt sich hierbei, dass für die Sets 2, 3 und 6 einzelne Flugzustände mit einem Robustheitsmaß $\rho < 1$ auftreten. Es handelt sich dabei jedoch lediglich um acht Flugzustände, was bei einer Gesamtzahl von 600 simulierten Punkten einem Anteil von 1,3% entspricht.

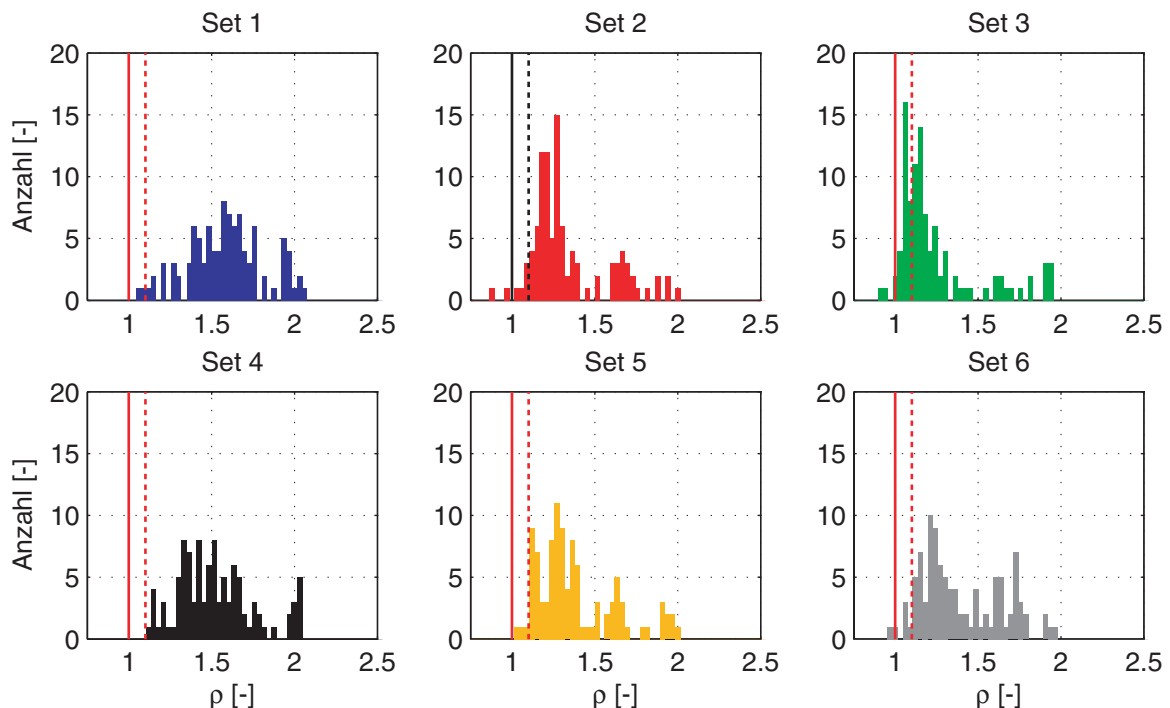


Abb. 4.15: Übersicht über erzielte Robustheitsmaße für Sets zufälliger Flugzustände

Darüber hinaus ist zu bedenken, dass es sich beim Robustheitsmaß ρ um die Über- bzw. Untererfüllung des Nichols-Kriteriums handelt. Ein Wert von $\rho < 1$ bedeutet also, dass das

Nichols-Kriterium in der Form, in der es im Rahmen dieser Arbeit vorliegt, am betreffenden Flugzustand nicht erfüllt wurde. Dies bedeutet nicht, dass am betreffenden Flugzustand automatisch ein instabiler Regelkreis zu erwarten ist.

Trotz der Verletzung des Nichols-Kriteriums meidet der Amplituden- und Phasengang des offenen Regelkreises den kritischen Punkt. In Tabelle 4.7 sind die minimal erreichten Robustheitsmaße für jedes der 6 Flugzustands-Sets zusammengestellt. Das minimale Robustheitsmaß beträgt $\rho_{min} = 0,87$. Dieser Wert wird im Hinblick auf den geringen Anteil von Flugzuständen mit verletztem Nichols-Kriterium zunächst toleriert. Alternativ besteht auch hier wie im Fall der Überschwingweiten und Anstiegszeiten die Möglichkeit, das Entwurfsraster zu verfeinern.

Set Nr	Min. Robustheitsmaß ρ_{min} [-]	Erreicht bei Flugzustand		
		α [deg]	Ma [-]	q_{dyn} [Pa]
1	1,06	-0,4	0,31	1000
2	0,87	-3,8	0,27	5000
	0,98	14,1	0,36	5000
3	0,92	14,1	0,50	10000
	0,94	-4,1	0,38	10000
	0,99	6,2	0,49	10000
4	1,12	14,6	0,21	2785
5	1,04	14,6	0,39	7444
6	0,96	-1,8	0,3	4102
	0,99	14,5	0,3	3674

Tabelle 4.7: minimale Robustheitsmaße und zugehörige Flugzustände

Bei der Betrachtung von Tabelle 4.7 fällt auf, dass die meisten Minimalwerte zwischen einem Anstellwinkel von 14° und 15° auftreten. Dies ist einer der Anstellwinkelbereiche, in dem die Regelstrecke starke Nichtlinearitäten aufweist (siehe Abschnitt 2.3). Insofern ist es nicht überraschend, dass das Entwurfsraster in diesem Bereich in Richtung des Anstellwinkels nicht fein genug ist, um auch in diesem Bereich über ein Gain-Scheduling für alle Flugzustände ein Robustheitsmaß von $\rho \geq 1$ zu garantieren. Auch hier ergeben sich jedoch durch eine Verfeinerung unpraktikabel große Reglerdatentabellen. Eine explizite Robustheitsuntersuchung, beispielsweise in Form einer μ -Analyse muss daher abschließend klären, ob die vorliegende, erreichte Robustheit unter der Voraussetzung bestimmter, gegebener Unsicherheiten ausreichend ist, um einen stabilen Regelkreis zu garantieren oder nicht.

4.4.3 Dämpfung und CAP-Wert

Die Abbildungen 4.16 und 4.17 zeigen die Verteilungen der ermittelten Anstellwinkelschwingungsdämpfungen und CAP-Werte. Wie im Fall des Robustheitsmaßes ρ zeigt sich auch hier, dass die Entwurfsforderungen $\zeta_{AS} \geq 0,7$ und $0,95 \leq CAP \leq 1,05$ nicht in allen getesteten Flugzuständen eingehalten werden, obwohl dies für alle Stützpunkte des Gain-Scheduling-Reglers der Fall war. Tabelle 4.8 gibt Aufschluss über die quantitative Verteilung

lung der erreichten AS-Dämpfungen sowie CAP -Werte. Durchschnittlich liegen rund 78% aller betrachteten Dämpfungen im vorgeschriebenen Bereich. Für knapp 90% aller Dämpfungen gilt $\zeta_{AS} \geq 0,6$, die kleinste aufgetretene Dämpfung beträgt $\zeta_{AS, min} = 0,30$ und tritt im Set 1 bei einem Anstellwinkel von $\alpha = 0,35^\circ$, einer Machzahl von $Ma = 0,31$ und einem Staudruck von $q_{dyn} = 1000\text{Pa}$ auf.

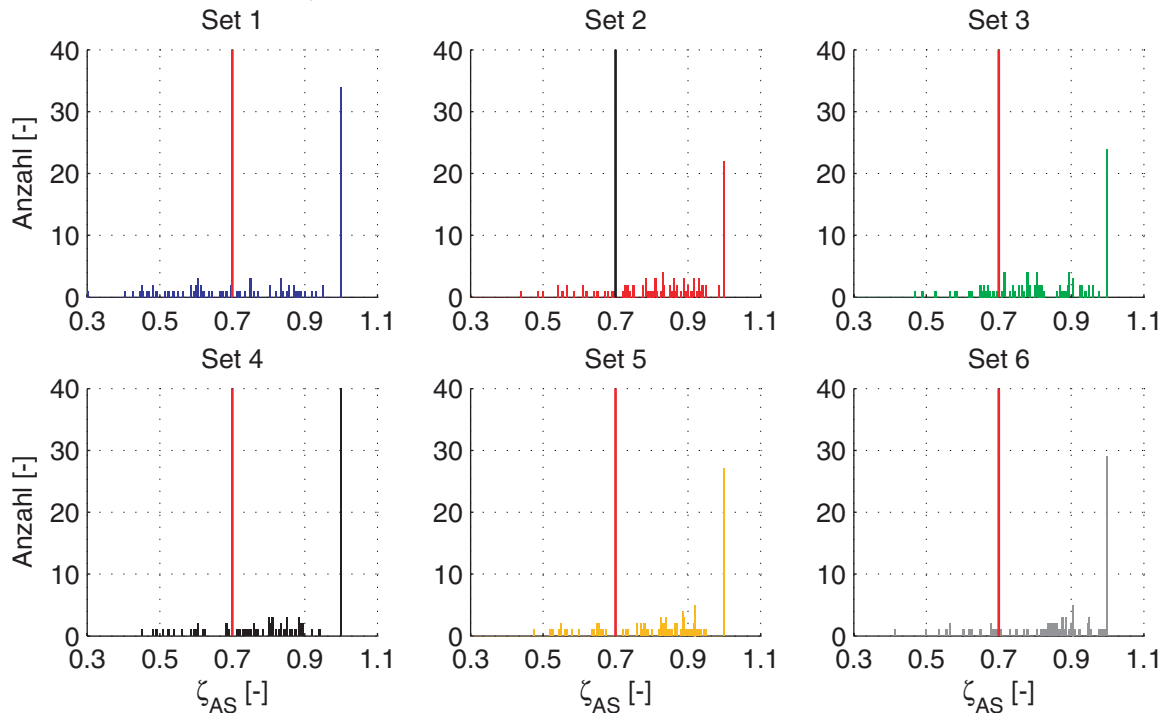


Abb. 4.16: Verteilungen der erreichten AS-Dämpfungen für Sets zufälliger Flugzustände

Im Fall der CAP -Werte treten größere Abweichungen vom Sollbereich auf. So liegen gemäß Tabelle 4.8 durchschnittlich nur etwa 10% aller Werte innerhalb des Intervalles $[0,95 \ 1,05]$. Die graphische Darstellung aller ermittelten Werte in Abbildung 4.17 zeigt zwar eine Häufung in der Umgebung von $CAP = 1$, es wird aber auch deutlich, dass ein Großteil der Flugzustände aller Sets CAP -Werte weit außerhalb des Sollbereiches aufweist.

Dies ist insofern verwunderlich, da alle Entwurfs- bzw. Interpolationsstützpunkte die Dämpfungs- und CAP -Anforderungen erfüllen, wie in Abschnitt 4.2.4 und Abbildung 4.10 gezeigt wurde. Dennoch sind die Streuungen erklärbar: Die Bildung der Reglermatrizen an Nicht-Entwurfspunkten mittels Interpolation führt im Allgemeinen zu einer Verschiebung der Pole des geschlossenen Regelkreises bezüglich der Polstellen der zu den Stützstellen gehörenden Regelkreise. Aufgrund dieser Polverschiebung kommt es zu Veränderungen der Eigenfrequenz des dominanten Poles, wobei sogar der als dominant identifizierte Pol an sich wechseln kann, was wiederum ebenfalls eine Veränderung der entsprechenden Eigenfrequenz bewirkt. Dies führt wiederum zu Abweichungen des berechneten CAP -Wertes vom Sollwert. Besonders kritisch sind dabei Flugzustände mit einem kleinen Wert von $n_{z, \alpha}$. In diesem Fall ergibt sich aus der Näherungsgleichung (4.29) für CAP die Notwendigkeit einer entsprechend kleinen Eigenfrequenz des dominanten AS-Poles, um einen Wert von $CAP = 1$ zu erreichen. Kleine Schwankungen des Absolutbetrages der Eigenfrequenz bewirken nun jedoch große relative Änderungen und damit auch starke Abweichungen des

CAP vom Sollwert 1, obwohl sich der Verlauf der α -Sprungantwort aufgrund der kleinen Polverschiebung nur unmerklich vom Sollverlauf an den umliegenden Stützpunkten unterscheidet. Bemerkenswert ist hierbei der Fall, wenn es sich beim dominanten Pol um einen Pol auf der reellen Achse handelt, der auf der reellen Achse wandert. In diesem Fall bewirkt die Verschiebung des Poles zwar eine Veränderung des CAP - Wertes und mitunter eine starke Abweichung von 1, jedoch keine Veränderung der Dämpfung der Anstellwinkelschwingung. Angesichts vieler Flugzustände mit zugeordneter Dämpfung $\zeta_{AS} = 1$, siehe Abbildung 4.16, erklärt dies wiederum, weshalb es weit weniger Punkte gibt, die das Dämpfungskriterium verletzen. Polverschiebungen bewirken in diesem Fall entweder keine Änderung der Dämpfung oder, im Fall einer Reduzierung, zumindest eine Dämpfung in der Umgebung von 1. Das Dämpfungskriterium ist somit leichter einzuhalten.

Im Hinblick auf die Problematik mit dominanten Polen mit kleiner Eigenfrequenz ist sicherlich auch die in Abschnitt 3.1.2 getroffene Vereinfachung des CAP -Kriteriums zur Reglerauslegung nach Level 1* kritisch zu hinterfragen, denn der im Rahmen dieser Arbeit vernachlässigte Knick im Verlauf der Level 1* Kennlinie verhindert eben diese kleinen Werte von ω_{AS} .

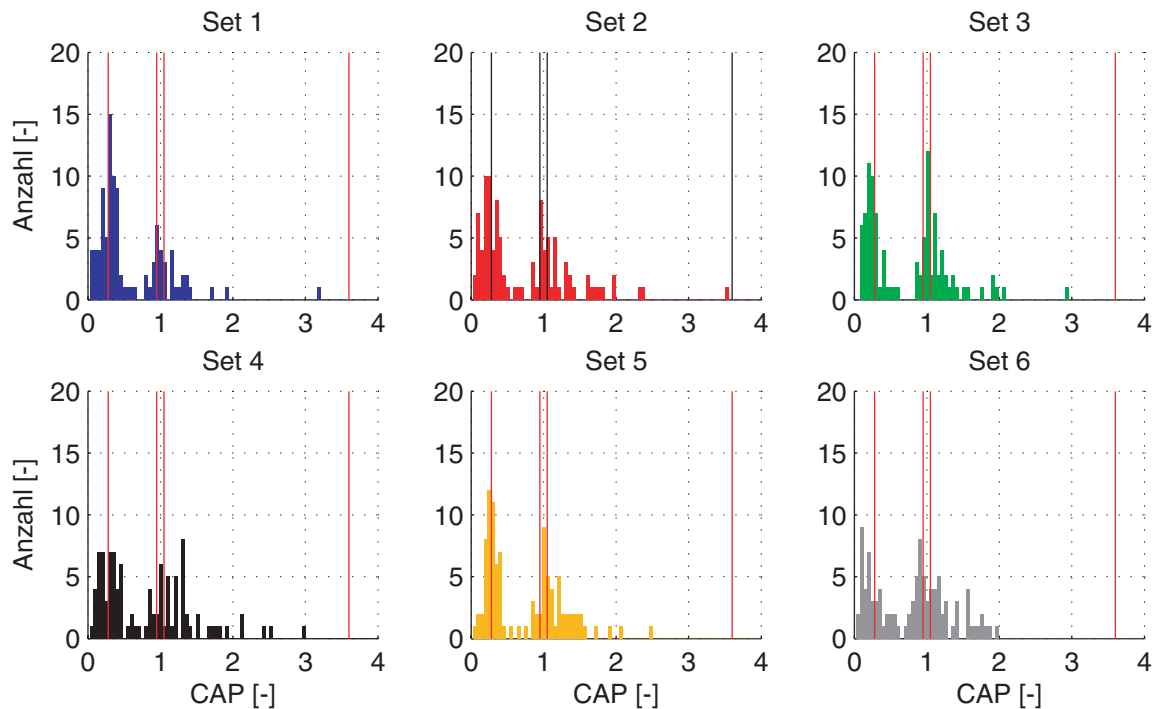


Abb. 4.17: Verteilungen der erreichten CAP -Werte für Sets zufälliger Flugzustände

Zur Umgehung der Problematik von wechselnden, als dominant identifizierten Polen erfolgt neben einer Berechnung des CAP -Wertes über die Identifizierung des dominanten Polpaares bzw. Poles nach Föllinger eine Berechnung über mehrere Alternativmethoden. Diese Methoden basieren zum Teil ebenfalls auf der Zuordnung eines Poles zur Anstellwinkelschwingung und anschließender Berechnung des CAP aus der Näherungsformel (4.29). Die Zuordnung erfolgt hierbei entweder durch direkte Betrachtung der Pole des geschlossenen Regelkreises hinsichtlich Dämpfung und/oder Eigenfrequenz oder aber durch Modalanalyse. Alternativ dazu erfolgt eine direkte Auswertung des zeitlichen Verlaufs der Sprungantwort

mit einer Näherungsweise Berechnung von Dämpfung und Eigenfrequenz nach [18] und anschließender *CAP*-Berechnung über Gleichung (4.29). Aus den so berechneten, unterschiedlichen *CAP*-Werten folgt der Wert *BestCAP*, indem der am nächsten am Sollwert 1 liegende Wert ausgewählt wird. Es ergibt sich hieraus die Verteilung in Abbildung 4.18. Auch hier kann nicht für alle Testpunkte ein *CAP*-Wert von 1 erreicht werden, die Häufung beim Sollwert ist jedoch deutlicher als im Fall des ursprünglich berechneten Wertes in Abbildung 4.17.

Merkmal		Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Mittelwert	
AS-Dämpfung	ζ_{AS} [-]	$\geq 0,7$	62,4 ^{a)}	82,2	81,2	82,2	81,2	81,2	78,4
		$\geq 0,6$	78,2	91,1	94,1	91,1	91,1	93,1	89,8
		$\geq 0,5$	88,1	98,0	98,0	97,0	99,0	98,0	98,0
		$< 0,5$	11,9	2,0	2,0	3,0	1,0	2,0	3,6
		(Minimum)	(0,30)	(0,44)	(0,47)	(0,45)	(0,47)	(0,41)	(0,42)
<i>CAP</i>	[rad/s ² /g]	[0,95 1,95]	6,9	11,9	15,8	6,9	13,9	7,9	10,6
		[0,50 1,50]	32,7	36,6	46,5	43,6	44,6	51,5	42,6
		Level 1 ^{b)}	72,3	67,3	66,3	78,2	75,2	75,2	72,4
<i>BestCAP</i>	[rad/s ² /g]	[0,95 1,95]	17,8	22,8	19,8	15,8	26,7	17,8	20,1
		[0,50 1,50]	75,2	89,1	87,1	83,2	83,2	88,1	84,3
		Level 1	82,2	95,0	92,0	90,1	93,1	96,0	91,4

a) Werte in Prozent

b) Level 1 modifiziert (ohne Knick)

Tabelle 4.8: Verteilungen von Dämpfung und *CAP* für Sets zufälliger Flugzustände

Es zeigt sich also, dass für das gegebene Flugzeugmodell mit der gegebenen Aerodynamik mit dem verwendeten Stützstellenraster die Vorgabe an den CAP -Wert nicht für alle Testpunkte erfüllt werden kann.

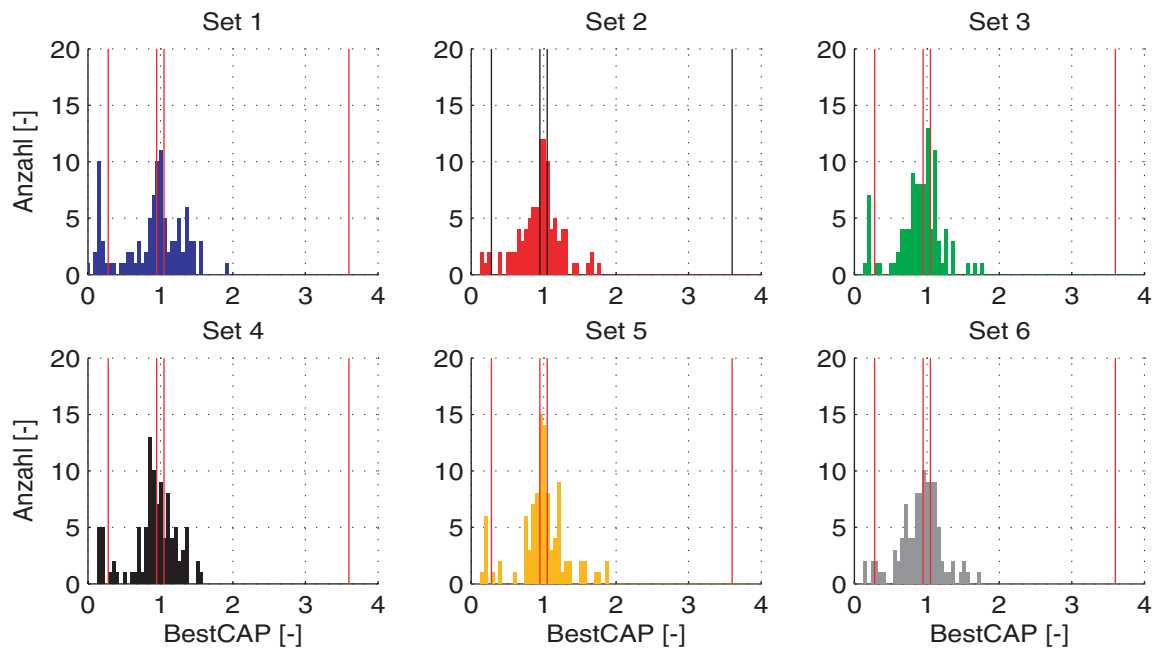


Abb. 4.18: Verteilungen der erreichten $BestCAP$ -Werte für Sets zufälliger Flugzustände

Im Hinblick auf den eigentlichen Zweck einer Vorgabe eines CAP -Wertes - der Vorgabe der AS-Eigenfrequenz zur Beeinflussung der Sprungantwort - und unter der Prämisse, dass sich die Sprungantworten an den Testpunkten tatsächlich überwiegend ähnlich zu denen der Entwurfspunkte verhalten, werden die Abweichungen jedoch toleriert.

Wie Abbildung 4.18 und Tabelle 4.8 zeigen, liegt zudem ein Großteil der bestimmten CAP - bzw. $BestCAP$ -Werte innerhalb der modifizierten Grenzen der MIL-Forderungen nach Level 1.

Die entsprechenden Grenzen bei $CAP = 0,28$ und $CAP = 3,6$ wurden in Abbildung 4.17 und 4.18 eingezeichnet und ergeben sich aus der Vernachlässigung des Knickes in der unteren Kennlinie für CAP nach Level 1 in Diagramm C.1 in Anhang C.1. Diese Vereinfachung der Kennlinie für Level 1 ist dabei konsistent zur bereits in Abschnitt 3.1.2 erwähnten Vereinfachung der Kennlinie für Level 1*.

Durch eine Verfeinerung des Entwurfsrasters ist zwar eine Verbesserung der Situation zu erwarten, allerdings wird, wie bereits erwähnt, von dieser Maßnahme abgesehen, da sich sonst ein Gain-Scheduling Regler mit unpraktikabel großer Stützstellenanzahl ergibt.

Kapitel 5

Vom linearen zum nichtlinearen Regler

In den vorangegangenen Kapiteln wurde gezeigt, wie für verschiedene Flugzustände auf quasi-stationären Abfangbögen und die zugehörigen linearen Modelle Flugregler entworfen werden und wie an beliebigen Flugzuständen interpolierte Regler mit *linearen* Modellen betrieben werden. Das vorliegende Kapitel beschäftigt sich mit der Implementierung des Reglers, wie er in Zusammenhang mit dem *nichtlinearen* Modell des Jettrainers zum Einsatz kommt. Diese Implementierung wird im Folgenden als nichtlinearer Scheduling-Regler bezeichnet.

5.1 Implementierung des nichtlinearen Scheduling-Reglers

Der nichtlineare Regler wird als Simulink Modell aufgebaut. Hierzu werden sämtliche Matrizen elementweise aus einzelnen Blöcken aufgebaut, konstante Elemente werden durch *Source*-Blöcke repräsentiert, variable Elemente je nach Art der Bestimmung ihres Wertes durch mehrdimensionale Interpolationsblöcke oder durch Geraden- bzw. Parabelregressionsblöcke. Die Signale dieser Einzelblöcke werden zunächst spaltenweise durch vertikale Matrizen-Konkatinierung zu Spaltenvektoren zusammengefasst, die Spalten werden anschließend durch horizontale Konkatinierung zur eigentlichen Matrix verschaltet. Abbildung 5.1 zeigt die prinzipielle Vorgehensweise für eine 2x2-Matrix M , deren Element $M_{1,1}$ variabel ist und vom Flugzustand abhängt. Im Beispiel erfolgt die Berechnung in Abhängigkeit des Anstellwinkels, der Machzahl und des Staudrucks über einen Standard Simulink Block für mehrdimensionale Interpolation. Für den Staudruck und die Machzahl werden für das Scheduling die aktuellen, zum momentanen Flugzustand gehörenden Werte verwendet. Um versteckte Rückführschleifen zu vermeiden wird für das Anstellwinkel-Scheduling dagegen nicht der aktuelle Wert von α verwendet, sondern der kommandierte Anstellwinkel α_c , der über ein PT2-Glied mit variablen Koeffizienten ω_{PT2} und ζ_{PT2} gefiltert wird. Gute Ergebnisse werden für $\omega_{PT2} = 3$ und $\zeta_{PT2} = 1$ erzielt. Die Elemente $M_{1,2}$ und $M_{2,1}$ seien konstant Null, das verbleibende Element sei ebenfalls konstant jedoch von Null verschieden und es gelte $M_{2,2} = K$. Im vorliegenden Fall bestehen die Matrizen A , C , F und H des Reglers aus insgesamt 234 Elementen. Um den Aufbau der zugehörigen Simulink-Subsysteme zu vereinfachen, erfolgt die Matrizenerstellung nicht von Hand, sondern über das Skript `Matrix_createXLR13`. Ausgehend von einem ausgewählten Workspace mit Reglerdatentabellen erfolgt der Aufbau der Matrizen aus konstanten und

variablen Blöcken gemäß den Informationen über die Matrizenstruktur aus den Variablen *MatAStruct*, *MatCStruct*, *MatFStruct* und *MatHStruct*. Diese Variablen wurden bei der Erstellung der Reglerdatentabellen über das Skript *TabCreate01* generiert und gemeinsam mit den Tabellen abgespeichert. Für die Implementierung der variablen Blöcke stehen verschiedene Varianten zur Verfügung. Je nachdem, ob die Berechnung eines variablen Elementes über Geraden- oder Parabelregression bzw. mehrdimensionale Interpolation erfolgen soll, wird das Element durch einen entsprechenden Block repräsentiert. Im Falle der mehrdimensionalen Interpolation stehen zudem die Varianten der klassischen Matlab Standard-3D-Interpolation oder alternativ der Dreiecks-Ebenen-Methode zur Verfügung.

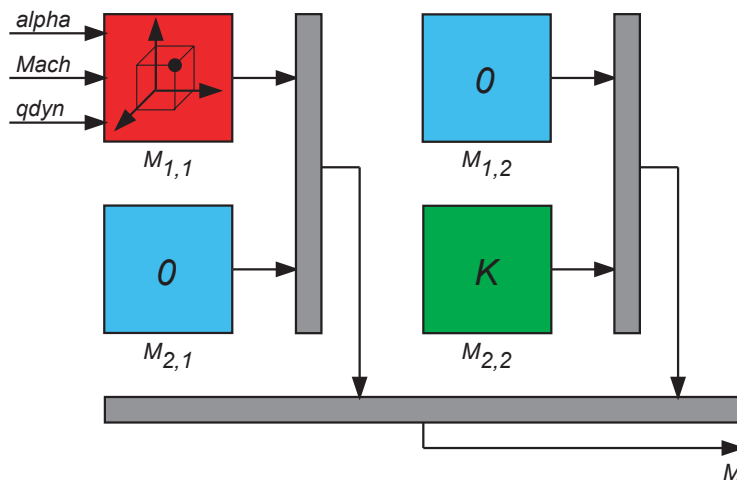


Abb. 5.1: beispielhafter Aufbau einer Matrix mit veränderlichen Elementen

Beide Varianten bieten zudem die Möglichkeit einer Bestimmung der zur Interpolation zu verwendenden Stützwerte dezentral in jedem variablen Element oder alternativ dazu einer zentralen Bestimmung. Die zentrale Bestimmung ist ausreichend, da die Indices der zur Interpolation zu verwendenden Tabellenwerte ausschließlich vom Flugzustand abhängen und für alle variablen Elemente identisch sind. Somit erfolgt eine Beschleunigung der Berechnung der Werte der variablen Elemente, da im letzteren Fall im Matrizenelement lediglich die Berechnung des Elementwertes gemäß dem ausgewählten Verfahren erfolgt, die Index-Bestimmung jedoch nur einmalig pro Flugzustand ausgeführt werden muss (siehe hierzu auch Abschnitt 3.2.2).

Neben den Matrizen des Reglers in beobachterähnlicher Darstellung sind auch die Übertragungsfunktionen der Wichtungsfunktionen als vom Flugzustand abhängige Subsysteme zu implementieren. Hierzu werden die Übertragungsfunktionen der Wichtungsfunktionen zunächst in Zustandsraumdarstellungen umgewandelt, deren Matrizen A , B , C und D in Abhängigkeit der Optimierungsparameter p_1 , p_2 und p_3 dargestellt werden. Für die Wichtungsfunktion W_1 gelten die folgenden Beziehungen:

$$A_{W_1} = 0 \quad (5.1)$$

$$B_{W_1} = 1 \quad (5.2)$$

$$C_{W1} = -p_3 \quad (5.3)$$

$$D_{W1} = 1. \quad (5.4)$$

Für die Wichtungsfunktion W_2 gilt:

$$A_{W2} = \begin{bmatrix} -100\pi & 0 & 0 \\ 0 & p_3 - 100\pi & p_3 \cdot 100\pi \\ 0 & 1 & 0 \end{bmatrix} \quad (5.5)$$

$$B_{W2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (5.6)$$

$$C_{W2} = \begin{bmatrix} p_1 \cdot 100\pi & 0 & 0 \\ 0 & p_2 \cdot 100\pi & 0 \end{bmatrix} \quad (5.7)$$

$$D_{W2} = 0. \quad (5.8)$$

Der Aufbau der Matrizen erfolgt analog zu den Reglermatrizen durch einzelne Blöcke sowie ggf. durch Summations- und Multiplikationsblöcke. Die Verschaltung einzelner Elemente zu Spaltenvektoren sowie deren Verschaltung zu Matrizen erfolgt wiederum durch horizontale und vertikale Matrizen-Konkatinierungsblöcke. Die Parameter p_1 , p_2 und p_3 werden dabei analog zu den variablen Matrizenelementen über Parabel- bzw. Geradenregression oder durch mehrdimensionale Interpolation abhängig vom Flugzustand berechnet.

Beim Vorfilter V handelt es sich um eine skalare Größe, deren Wert in Abhängigkeit des Flugzustands aus einer zugehörigen Tabelle analog zu den variablen Matrizenelementen bestimmt wird.

5.2 Übergang von relativen zu absoluten Größen

Der nichtlineare Scheduling-Regler basiert auf linearen Einzelreglern. Diese wurden mit Hilfe von linearen Modellen entworfen, welche durch Linearisierung des nichtlinearen Flugzeugmodells an verschiedenen Trimpunkten abgeleitet wurden. Dementsprechend arbeiten die linearen Einzelregler mit relativen Größen bezüglich des Trimmzustands (Index 0), für den das lineare Modell abgeleitet wurde. Das heißt, der lineare Regler enthält als Eingänge zum einen die Messungen $\Delta\alpha = \alpha_{mess} - \alpha_0$ und $\Delta q = q_{mess} - q_0$ sowie das Kommando $\Delta\alpha_c = \alpha_c - \alpha_{c,0}$ und generiert daraus den Streckeneingang $\Delta\eta = \eta - \eta_0$. Das nichtlineare Modell des Jettrainers benötigt bzw. liefert an seinem Eingang bzw. Ausgang jedoch absolute Größen. Wie Abbildung 5.2 zeigt, besteht eine Möglichkeit des Übergangs zwischen relativen und absoluten Größen in der Addition bzw. Subtraktion entsprechender

Referenzwerte. Hinweis: in der Abbildung wurden der Vorfilter V sowie die Wichtungsfunktionen W_1 und W_2 bereits in den Regler integriert.

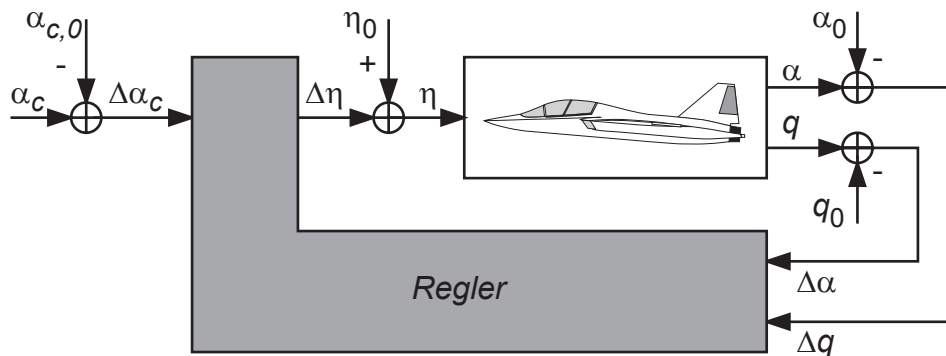


Abb. 5.2: Übergang zwischen relativen und absoluten Größen

Bei dieser Vorgehensweise stellt sich nun jedoch die Frage nach der zum momentanen Flugzustand gehörenden Größe der Referenzwerte. Im Fall der Nickrate q_0 und dem Höhenruderwinkel η_0 ist die Bestimmung problemlos möglich, indem Werte aus den entsprechenden Trimmstabellen für den Flug auf einem quasi-stationären Abfangbogen verwendet werden. Zu jedem Zeitpunkt wird angenommen, dass sich das Flugzeug auf einem Abfangbogen befindet, der über den momentanen Anstellwinkel, die momentane Machzahl und den momentanen Staudruck definiert ist. Aus den Ergebnissen der Trimmung, den Trimmstabellen werden dann die zugehörige Höhenruderstellung sowie die sich einstellende Nickrate ermittelt und als Referenzwerte verwendet.

5.2.1 Referenzwert für Anstellwinkel

Im Fall des Anstellwinkels ist die Wahl eines Referenzwertes schwieriger. Der momentane Flugzustand scheidet als Referenzierung aus, denn ansonsten wird vom Ausgang des Modells, also dem momentanen Anstellwinkel stets der momentane Anstellwinkel selbst abgezogen. Der entsprechende Eingang des Reglers ist dann immer identisch Null. Die Verwendung des momentan kommandierten Anstellwinkels α_c als Referenzwert α_0 ist zwar denkbar, scheidet aber im vorliegenden Fall ebenfalls aus. In diesem Fall führt eine Änderung des kommandierten Anstellwinkels zu einer direkten Anregung der Reglerdynamik, was einer Implementierung des Reglers im Vorwärtszweig entspricht. Entsprechend den Ergebnissen aus Kapitel 3.1.5 sollte dies jedoch vermieden werden.

Es kann jedoch gezeigt werden, dass zur Referenzierung der Anstellwinkelmessung ein beliebiger konstanter Winkel $\alpha_0 = const$ verwendet werden kann, sofern der gleiche Referenzwinkel auch zur Bildung von $r = \Delta\alpha_c = \alpha_c - \alpha_0$ am Regler- bzw. Vorfiltereingang verwendet wird.

Analog zur Bestimmung des Vorfilters in Abschnitt 3.1.4 muss auch hier im stationären Zustand gelten

$$V \cdot (\alpha_c - \alpha_0) + y_{R,ss} = 0. \quad (5.9)$$

Für $y_{R,ss}$, den Ausgang des Reglers im stationären Zustand, gilt dabei unter der Voraussetzung eines konstanten stationären Anstellwinkels $\Delta\alpha_{ss}$ und einer konstanten stationären Nickrate Δq_{ss} :

$$y_{R,ss} = K_{ss} \cdot W_{2,ss} \cdot \begin{bmatrix} \Delta\alpha_{ss} \\ \Delta q_{ss} \end{bmatrix}. \quad (5.10)$$

Der Anstellwinkel $\Delta\alpha_{ss}$ ergibt sich mit dem Referenzwert α_0 aus dem gemessenen Anstellwinkel α_{mess} zu

$$\Delta\alpha_{ss} = \alpha_{mess} - \alpha_0. \quad (5.11)$$

Dabei muss im eingeregelten Zustand gelten

$$\alpha_{mess} = \alpha_c. \quad (5.12)$$

Mit dem Ausdruck für K_{ss} aus Kapitel 3.1.4 folgt hieraus schließlich

$$0 = V \cdot (\alpha_c - \alpha_0) + F(A + HC)^{-1}H \cdot W_{2,ss} \cdot \begin{bmatrix} 1 \\ \Delta q_{ss}/(\alpha_c - \alpha_0) \end{bmatrix} \cdot (\alpha_c - \alpha_0), \quad (5.13)$$

wobei der Term $\Delta q_{ss}/(\alpha_c - \alpha_0)$ zu Gunsten einer besseren Übersichtlichkeit zu Null gesetzt wird. Dies ist zulässig, da die stationäre Übertragung der Wichtungsfunktion W_2 im q -Kanal Null ist. Aus Gleichung (5.13) folgt damit wie bereits im Kapitel 3.1.4 für den Vorfilter:

$$V = -F(A + HC)^{-1}H \cdot W_{2,ss} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (5.14)$$

Der Einfluss des Referenzwinkels α_0 hebt sich also auf. Er bildet im Eingang $\Delta\alpha$ des Reglers einen konstanten Anteil, der zwar über den Regler zur Summationsstelle zurückgeführt wird, der zugehörige Anteil des Reglerausgangs wird jedoch über einen entsprechenden Anteil im Ausgang des Vorfilters kompensiert. Der numerische Wert des Referenzwinkels ist daher unerheblich und kann frei gewählt werden. Lediglich am Anfang einer Simulation des nichtlinearen Flugzeugmodells mit Regler spielt der tatsächliche Wert des Referenzwertes α_0 eine Rolle, da er quasi sprunghaft am Reglereingang auftritt. Nach dem Abklingen der transienten Antwort des Reglers auf diesen Sprung tritt aber die Kompensation des stationären Anteils der Übertragung von α_0 durch den Vorfilter ein. Durch die Wahl $\alpha_0 = 0$ wird dieser Effekt jedoch umgangen.

5.2.2 Referenzwert für Nickrate und Höhenruder

Im Fall der Nickrate muss, wie bereits erwähnt, der Reglereingang Δq im eingeregelten Zustand konstant sein, es muss gelten $\Delta\dot{q} = 0$. Nur für diesen Fall ist die stationäre Übertragung der Wichtungsfunktion W_2 Null, das heißt der Einfluss der Nickrate im

getrimmten Flugzustand klingt mit der Zeit ab. Im Fall des linearen Reglers, der die Kurzzeitznäherung des Jettrainers regelt, stellt sich für einen Sprungeingang in α_c automatisch eine konstante Nickrate ein und es gilt $\dot{q} = 0$. Im Fall der nichtlinearen Flugzeugdynamik ist dies jedoch im Allgemeinen nicht der Fall. Für die Annahme, dass sich das Flugzeug stets auf einem quasi-stationären Abfangbogen befindet kann aus der Trimmtablelle zwar eine zugehörige konstante Nickrate bestimmt werden, aufgrund der sich ändernden Fluggeschwindigkeit wechselt das Flugzeug jedoch ständig von einem Abfangbogen zu einem anderen, weshalb sich auch die zugehörige Nickrate ständig ändert. In diesem Fall gilt also $\dot{q}_{mess}(t \rightarrow \infty) \neq 0$, weshalb der Einfluss der Nickrate ohne entsprechende Referenzierung auch für $t \rightarrow \infty$ nicht abklingt. Deshalb ist im Fall der Nickrate die Differenzbildung entsprechend $\Delta q = q_{mess} - q_0$ mit einem Referenzwert $q_0 \neq const$ zwingend notwendig, um stationäre Regelgenauigkeit zu erreichen. Als Referenzwert wird beispielsweise die zum momentanen Flugzustand gehörende Trimm-Nickrate verwendet, die über die momentane Machzahl, den momentanen Staudruck sowie den aktuell kommandierten Anstellwinkel aus der Trimmtablelle der Nickrate bestimmt wird. Im eingeregelteten Zustand gilt dann $q_0 = q_{mess}$ und somit $\Delta q = 0 = const$. Bei sprunghaftigen Änderungen des kommandierten Anstellwinkels führt diese Vorgehensweise jedoch zu ebenfalls sprunghaften Änderungen des Referenzwertes q_0 , was wiederum im geschlossenen Regelkreis zu Störungen im Verlauf des Anstellwinkels führt. Hierbei ist zu bedenken, dass die Bedingung für stationäre Regelgenauigkeit lediglich lautet $\Delta q = const$ bzw. $\Delta \dot{q} = 0$. Um diese Bedingung zu erfüllen, muss Δq jedoch nicht Null werden. Es reicht also aus, ausgehend von einem Startwert für q_0 die Änderung der Nickrate im eingeregelteten Zustand zu kompensieren.

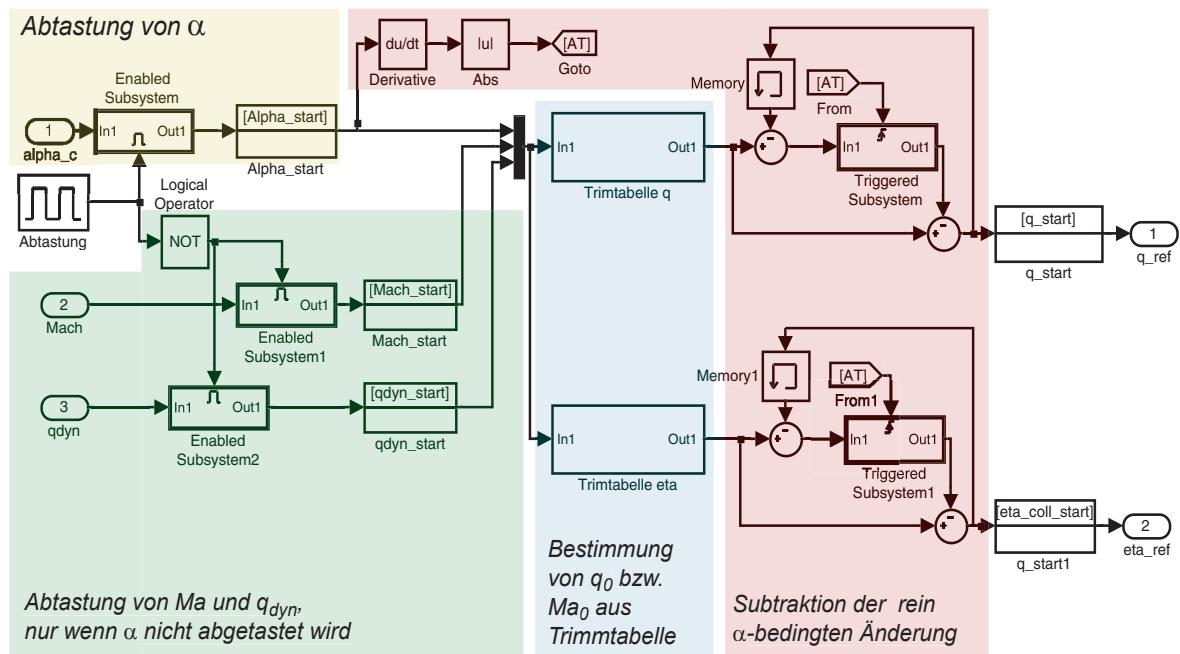


Abb. 5.3: Bildung der Referenzwerte für q und η

Dabei ist der Anstellwinkel im eingeregelteten Zustand konstant, die Änderung der Trimm-Nickrate ergibt sich also rein aus der Änderung der Machzahl und des Staudrucks.

Dementsprechend wird der Referenzwert q_0 zwar aus der Trimmtablelle der Nickrate bestimmt, jedoch für die momentane Machzahl, den momentanen Staudruck und einen *konstanten* Anstellwinkel. Berücksichtigt wird also die Steigung der Trimmtablelle in Richtung der Machzahl und des Staudrucks. Diese Steigung hängt jedoch wiederum vom aktuellen Anstellwinkel ab. Damit die Änderung des Referenzwertes mit sich verändernden Machzahlen und Staudrücken zum momentanen Anstellwinkel passt, erfolgt die Ermittlung von q_0 daher getaktet und Änderungen des Staudrucks und der Machzahl einerseits sowie Änderungen des Anstellwinkels andererseits werden abwechselnd und getrennt voneinander berücksichtigt. Bei einer Änderung des Anstellwinkels wird somit an die korrekte Stelle in der Trimmtablelle gesprungen, wobei die rein anstellwinkelbedingte Änderung des Referenzwertes q_0 kompensiert wird. Änderungen der Machzahl und des Staudrucks führen somit im korrekten Anstellwinkelbereich der Trimmtablelle zu Veränderungen des Referenzwertes, die wiederum die Veränderungen des Reglereingangs Δq kompensieren. Die Bedingung $\Delta \dot{q} = 0$ ist also erfüllt. Abbildung 5.3 zeigt eine entsprechende Realisierung in Matlab-Simulink.

Neben der Aktualisierung des Nickraten-Referenzwertes erfolgt hier analog auch die Berechnung eines Referenzwertes für das Höhenruder η . In diesem Fall ist die Bildung eines Absolutwertes aus dem Relativwert nach der Summationsstelle und einem Referenzwert η_0 gemäß $\eta = \Delta\eta + \eta_0$ nicht zwingend notwendig. Der Integrator im Regler, genauer gesagt in der Wichtungsfunktion W_2 , sorgt auch ohne Addition eines Referenzwertes für einen korrekten Absolutwert der Höhenruderstellung. Die Berücksichtigung der sich verändernden Trimmstellung des Ruders aufgrund sich verändernder Machzahlen und Staudrücke über die entsprechende Trimmtablelle nimmt dem Regler jedoch diese zusätzliche Trimm-Aufgabe ab und verbessert somit die Regelgüte durch eine etwas schneller erreichte stationäre Genauigkeit.

5.3 Sprungfreies Scheduling

Für Regler mit Gain-Scheduling wird im Allgemeinen ein glatter Verlauf der variablen Parameter über den Scheduling-Variablen gefordert und damit verbunden eine langsame Änderung der variablen Reglerparameter [6], [20]. Anderfalls ist es möglich, dass die Regelstrecke zwar für festgehaltene Schedulingparameter an allen Entwurfsunkten durch den zugehörigen Regler stabilisiert wird, bei sich ändernden Reglerparametern im nichtlinearen Betrieb jedoch Instabilitäten auftreten. Des Weiteren wird gefordert, dass über die variablen Reglerparameter die Nichtlinearitäten der Regelstrecke erfasst werden. Beide Forderungen führen im vorliegenden Fall jedoch zu einem Widerspruch. Die Nichtlinearitäten der Regelstrecke werden durch den Entwurf der Einzelregler durch beschränkte Optimierung tatsächlich erfasst. Eben deshalb ergibt sich jedoch vor allem für die Abhängigkeit der Schedulingparameter vom Anstellwinkel ein stark nichtlinearer, nicht stetig differenzierbarer Verlauf. Bei einer Änderung des Anstellwinkels aufgrund einer Kommandoänderung ändern sich somit die Schedulingparameter schnell. Dies gilt auch, wenn der Anstellwinkel vor dem eigentlichen Scheduling der Reglerparameter über ein PT2-Glied verzögert wird. Hierdurch wird die zeitliche Änderung der Schedulingvariablen zwar verlangsamt, es wird jedoch trotzdem der gesamte Bereich vom Start- bis zum

Zielanstellwinkel durchlaufen, wodurch sich schnelle Änderungen der Reglerparameter ergeben. Diese führen wiederum zu Störungen bei der Bildung der Zustandsableitung des Reglers und damit bei der Integration der Reglerzustände. Diese Störungen führen schließlich zu Störungen in Form von Ausreißern im Verlauf des Anstellwinkels.

Betrachtet man den Ablauf der numerischen Integration der Differentialgleichungen des Reglers, so lässt sich der Effekt nachvollziehen. Im anschaulichen Fall der Lösung der Differentialgleichungen über das Euler-Verfahren wird ausgehend vom Zustand $x(t_i)$ und Eingang $u(t_i)$ des Reglers zum Zeitpunkt t_i die zugehörige Zustandsableitung $\dot{x}(t_i)$ berechnet. Aus dem Zustand $x(t_i)$ und der Zustandsableitung $\dot{x}(t_i)$ folgt über die Schrittweite des Integrationsverfahrens Δt der Zustand $x(t_{i+1})$ des folgenden Zeitschrittes. Es gilt

$$x(t_{i+1}) = x(t_i) + \dot{x}(t_i)\Delta t. \quad (5.15)$$

Der im Rahmen dieser Arbeit betrachtete Beobachterregler liegt in Form einer Zustandsraumdarstellung vor, deshalb berechnet sich die Zustandsableitung $\dot{x}(t_i)$ gemäß der Formel

$$\dot{x}(t_i) = A(t_i)x(t_i) + B(t_i)u(t_i), \quad (5.16)$$

die Zeitabhängigkeit der Matrizen A und B folgt hierbei aus der Zeitabhängigkeit der Schedulingparameter.

Nach der Integration entsprechend Gleichung (5.15) wird wiederum die Zustandsableitung zum Zeitpunkt t_{i+1} entsprechend Gleichung (5.16) gebildet u.s.w. Kommt es nun zwischen den Zeitpunkten t_i und t_{i+1} zu einer schedulingbedingten Änderung der Matrix A , folgt hieraus eine Veränderung der Zustandsableitung $\dot{x}(t_{i+1})$ im Vergleich zum Wert, der sich für eine konstante Matrix A ergeben hätte. Diese Veränderung ist für hinreichend große Änderungen der Reglerparameter, hier der Matrix A , als Störung aufzufassen, die unter Umständen zur Destabilisierung des Regelkreises führt. Das Gleiche gilt für Änderungen der Matrix B und im Fall der Ausgangsgleichung des Reglers für die Matrizen C und ggf. D .

Das Problem wird gelöst, indem garantiert wird, dass sich trotz veränderter Reglermatrizen die Ableitung des Zustands des Beobachterreglers und der Wichtungsfunktion W_2 sowie der Ausgang der Wichtungsfunktion W_1 nicht verändert. Hierzu wird der Regler nicht kontinuierlich an den momentanen Flugzustand angepasst, sondern nur zu diskreten Zeitpunkten. Bei jedem Scheduling-Zeitpunkt werden nun die Zustandsvektoren so zurückgesetzt, dass sich im Verlauf der Zustandsableitung bzw. des Ausgangs trotz sich verändernder Reglerparameter keine Änderung und somit auch kein Knick ergibt. Für die Integration der Reglerzustände mittels des Euler-Verfahrens beruhigt sich auf diese Weise der Verlauf der Reglerzustände und somit des Reglerausgangs und damit auch der Verlauf des Anstellwinkels.

Für den Fall, dass der Ausgang einer Übertragungsfunktion mit erfolgtem Scheduling y_m gleich sein soll wie der Ausgang y_o ohne Scheduling, muss also gelten

$$y_m = y_o. \quad (5.17)$$

Für eine Übertragungsfunktion, bei der sich die Zustandsableitung durch das Scheduling nicht verändern soll, gilt die Bedingung

$$\dot{x}_m = \dot{x}_o. \quad (5.18)$$

Allgemein gilt dabei im Fall der Ausgangsgleichung (5.17)

$$y_m = C_m x_m + D_m u_m = C_o x_o + D_o u_o = y_o. \quad (5.19)$$

Zum Zeitpunkt des Scheduling t_S muss der Zustandsvektor auf den Wert x_m gesetzt werden, für den sich aus Gleichung (5.19) ergibt:

$$x_m = C_m^{-1}(C_o x_o + D_o u_o - D_m u_m). \quad (5.20)$$

Im Fall der Zustandsableitung (5.18) gilt

$$\dot{x}_m = A_m x_m + B_m u_m = A_o x_o + B_o u_o = \dot{x}_o, \quad (5.21)$$

woraus für den Rücksetzzustand folgt:

$$x_m = A_m^{-1}(A_o x_o + B_o u_o - B_m u_m). \quad (5.22)$$

Die Werte der Reglermatrizen nach dem Scheduling (Index m) ergeben sich aus der Anpassung an den Flugzustand $FZ(t_S)$ zum Zeitpunkt t_S des Scheduling, sie werden im Folgenden ohne Index angegeben. Für die Werte der Reglermatrizen ohne Scheduling (Index o) wird die Anpassung an den Flugzustand zum Zeitpunkt unmittelbar vor dem Scheduling-Zeitpunkt verwendet, genauer gesagt einen Integrationszeitschritt Δt vor t_S . Sie werden im Folgenden durch den Index 'o' gekennzeichnet. Im Fall einer allgemeinen Matrix M gilt also

$$M_m = M(FZ(t_S)) = M \quad (5.23)$$

und

$$M_o = M(FZ(t_S - \Delta t)) = M_o. \quad (5.24)$$

Der Eingang u_o bzw. u_m wird im Allgemeinen als während des Scheduling konstant angenommen.

Im Fall des Zustands x_o und des Eingangs u_o können ebenfalls die Werte zum Zeitpunkt $t = t_S - \Delta t$ verwendet werden. Dann ist allerdings zu beachten, dass die Taktung des Scheduling nicht zu fein gewählt wird, da die Reglerzustände bei jedem Scheduling auf ihren vorangegangenen Wert gesetzt werden. Bei einem kleinen zeitlichen Abstand aufeinander folgender Regleraktualisierungen im Gegensatz zur Schrittweite der Integration führt dies zu einer Art Passivierung des Reglers. Anstatt zur Berechnung des Rücksetzzustands also die Zustandsgrößen vom vorangegangenen Integrationschritt zu verwenden ist es besser, den State-Port der vorkommenden Integratoren zu verwenden.

Laut Simulink Programm-Dokumentation [23] werden die Zustandswerte am State-Port Ausgang vor der eigentlichen Zustandsberechnung und Ausgabe berechnet und gesetzt. Somit liegt am State-Port eines zurückgesetzten Integrators der Zustand an, der sich aus der

Integration ergeben hätte, falls der Integrator nicht zurückgesetzt worden wäre. Der State-Port eines Simulink-Integrators kann jedoch ausschließlich zur Berechnung von Startwerten desselben Integrators verwendet werden. Aus diesem Grund ist es zunächst notwendig, alle Integratoren des Reglers zusammenzufassen. Hierzu wird aus dem Regler gemäß Schaubild 3.4 aus Kapitel 3.1.4 eine Ersatz-Übertragungsfunktion in Zustandsraumdarstellung gebildet. Aus der Verschaltung der Reglermatrizen A , B , C , H und F sowie der Wichtungsfunktionen W_1 mit A_{W1} , B_{W1} , C_{W1} und D_{W1} sowie W_2 mit A_{W2} , B_{W2} , C_{W2} und D_{W2} und dem Vorfilter V ergibt sich folgende Zustandsraumdarstellung:

$$A_{ges} = \begin{bmatrix} A + HC + BF & 0 & -HC_{W2} \\ B_{W1}F & A_{W1} & 0 \\ 0 & 0 & A_{W2} \end{bmatrix} \quad (5.25)$$

$$B_{ges} = \begin{bmatrix} BV & 0 \\ B_{W1}V & 0 \\ 0 & B_{W2} \end{bmatrix} \quad (5.26)$$

$$C_{ges} = [D_{W1}F \ C_{W1} \ 0] \quad (5.27)$$

$$D_{ges} = [D_{W1}V \ 0] \quad (5.28)$$

mit dem Gesamtzustandsvektor \hat{x}_{ges} , der sich folgendermaßen aus dem Reglerzustand \tilde{x} sowie den Zuständen x_{W1} und x_{W2} der Wichtungsfunktionen zusammensetzt:

$$\hat{x}_{ges} = [\tilde{x} \ x_{W1} \ x_{W2}]^T. \quad (5.29)$$

Der Eingangsvektor des Gesamtsystems setzt sich zusammen aus dem Kommando r sowie den Messungen von Anstellwinkel und Nickrate:

$$\hat{u}_{ges} = [r \ \Delta\alpha \ \Delta q]^T. \quad (5.30)$$

Der Ausgang des Systems ist die Höhenruderstellung:

$$y_{ges} = \Delta\eta. \quad (5.31)$$

Aus dem Gesamtsystem berechnen sich die zum Beobachter gehörenden Komponenten des Rücksetzzustands \tilde{x}_R über Gleichung (5.22) mit Hilfe der entsprechenden Komponenten des State-Port Ausgangs x_{SP} zu folgendem Ausdruck:

$$\tilde{x}_{Reset} = (A + HC + BF)^{-1}[(A + HC + BF)\tilde{x}_{SP} + B(V_- - V)\alpha_c + Hy_{W2,m} - Hy_{W2,o}]. \quad (5.32)$$

Hierbei gilt für den Ausgang der Wichtungsfunktion W_2 nach bzw. mit Scheduling wegen $D_{W2} = 0$:

$$y_{W2,m} = x_{W2,Reset} C_{W2} \quad (5.33)$$

sowie für den Ausgang vor bzw. ohne Scheduling

$$y_{W2,o} = x_{W2,SP} C_{W2}. \quad (5.34)$$

Die Komponenten des Zustandsvektors, die zur Wichtungsfunktion W_1 gehören, berechnen sich zu dem Ausdruck

$$x_{W1,Reset} = C_{W1}^{-1} [C_{W1} x_{W2,SP} + D_{W1} (u_{W1,o} - u_{W1,m}) + \eta_{0,m} - \eta_{0,o}] \quad (5.35)$$

mit dem Eingang der Wichtungsfunktion W_1 nach bzw. mit Scheduling

$$u_{W1,m} = V \alpha_c + F \tilde{x}_{Reset} \quad (5.36)$$

und vor bzw. ohne Scheduling

$$u_{W1,o} = V \alpha_c + F \tilde{x}_{SP}. \quad (5.37)$$

Der Eingang des kommandierten Anstellwinkels α_c wird dabei als während des Scheduling konstant angenommen.

Die Komponenten des Rücksetz-Zustandsvektors der Wichtungsfunktion W_2 ergeben sich aus Gleichung (5.22) zu dem Ausdruck

$$x_{W2,Reset} = A_{W2}^{-1} [A_{W2} x_{W2,SP} + B_{W2} (u_{W2,o} - u_{W2,m})]. \quad (5.38)$$

Beim Eingang u_{W2} der Wichtungsfunktion W_2 handelt es sich dabei um den Ausgang der nichtlinearen Strecke, d.h. die Relativwerte der Messungen $\Delta\alpha$ und Δq . Diese ändern sich aufgrund eines Regler-Scheduling nicht, deshalb gilt

$$u_{W2,o} - u_{W2,m} = 0 \text{ für } t > 0. \quad (5.39)$$

Lediglich zum Zeitpunkt $t = 0$ einer Simulation muss gelten

$$u_{W2,o} - u_{W2,m} = u_{W2,0} = \Delta y_{mess,0}. \quad (5.40)$$

Somit gilt zu Beginn einer Simulation

$$x_{W2,0} = x_{W2,Reset,0} = A^{-1} [A x_{W2,SP,0} - B \Delta y_{mess,0}] = -A^{-1} B \Delta y_{mess,0} \quad (5.41)$$

wegen

$$A x_{W2,SP,0} = 0. \quad (5.42)$$

Daraus folgt wiederum

$$y_{W2,0} = C x_{W2,0} = -C A^{-1} B \Delta y_{mess,0}, \quad (5.43)$$

wobei es sich bei dem Term $-CA^{-1}B$ um die stationäre Antwort der Wichtungsfunktion W_2 handelt, d.h. auf diese Weise wird erreicht, dass die Wichtungsfunktion zu Beginn der Simulation die Messung des Ausgangs der nichtlinearen Strecke weiterleitet.

Der Rücksetz-Zustandsvektor des Gesamtsystems setzt sich zusammen aus den Komponenten in den Gleichungen (5.32), (5.35) und (5.38) zu

$$x_{Reset} = \left[\tilde{x}_{Reset} \ x_{W1,Reset} \ x_{W2,Reset} \right]^T. \quad (5.44)$$

Die Simulink Implementierung der Berechnung des Rücksetz-Zustands ist in Abbildung A.5 in Anhang A.2.4 dargestellt.

5.4 Das nichtlineare Simulationsmodell

In Kapitel 2 wurden die nichtlinearen Differentialgleichungen des Jettrainers hergeleitet. Diese liefern ausgehend vom momentanen Zustands- und Eingangsvektor den Vektor der Zustandsableitung. Dieser Vektor wird integriert und zum Eingang der Flugzeugdynamik zurückgeführt. Die vorliegende Modellierung hat die Besonderheit, dass aufgrund der aerodynamisch bedingten Implizitat der Differentialgleichungen auch die Zustandsableitung als Eingang benotigt wird. Eine direkte Ruckfuhrung der Zustandsableitung zum Eingang der Differentialgleichungen fuhrt jedoch zu einer algebraischen Schleife. Im vorliegenden Fall wird dieses Problem gelost, indem die Zustandsableitung vor der Ruckfuhrung durch einen *TransportDelay*-Block bzw. einen *Memory*-Block um einen Integrationszeitschritt verzogert wird. Dementsprechend wird ausgehend von einem Startwert 0 die Zustandsableitung zum Zeitpunkt t_i mit Hilfe der Ableitung zum Zeitpunkt t_{i-1} berechnet. Fur hinreichend kleine Integrations-schrittweiten wie beispielsweise $\Delta t = 0,005$ s im vorliegenden Fall ist der dadurch entstehende Fehler vernachlassigbar. Die nichtlineare Flugzeugdynamik wird entsprechend dem Schaubild in Abbildung 5.4 mit dem nichtlinearen Gain-Scheduling-Regler zum nichtlinearen Simulationsmodell verschaltet.

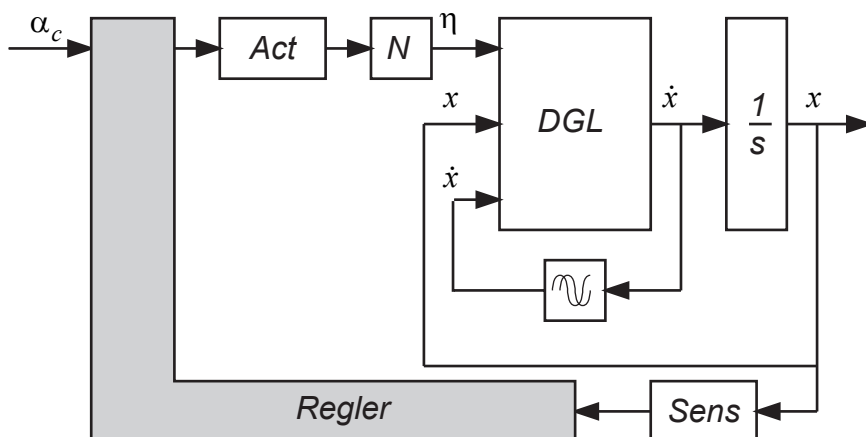


Abb. 5.4: prinzipieller Aufbau des nichtlinearen Simulationsmodells

Diverse Abbildungen des zugehörigen Simulink-Modells befinden sich in Anhang A.2.1. Zu beachten ist, dass der Notch-Filter N anders als bei der Bildung des linearen Entwurfsmodells (siehe Abschnitt 2.2.9) nicht in den Aktuator Act integriert ist, sondern als gesondertes Modell in Zustandsraumdarstellung vorliegt. Die Modellierung der Sensorik $Sens$ entspricht dem linearen Modell in Abschnitt 2.2.9.

Neben dem Regler für die Längsbewegung wird im Fall des nichtlinearen Gesamtmodells auch ein Regler für die Seitenbewegung benötigt, denn die Aerodynamik des Jettrainers ist asymmetrisch. Für einen symmetrischen Flugzustand und symmetrische Steuereingänge liefert die Aerodynamik im Allgemeinen wider Erwarten ein Moment um die Hoch- und die Längsachse des Flugzeugs. Um dennoch einen Flug in der Vertikalebene zu ermöglichen müssen daher die asymmetrischen Steuerflächen so ausgeschlagen werden, dass die asymmetrischen Momente kompensiert werden. Dies geschieht im vorliegenden Fall durch einfache PID-Regler, auf deren Auslegung nicht näher eingegangen wird. Das Steuerkommando α_c wird beispielsweise über einen Simulink *SignalBuilder*-Block erzeugt, um treppen- oder rampenförmige Änderungen des kommandierten Anstellwinkels zu simulieren. Alternativ dazu besteht im Simulink Gesamtmodell die Möglichkeit, einen Joystick anzuschließen um interaktive Simulationen durchzuführen. Darüber hinaus kann ein zuvor aufgezeichneter Kommandoverlauf eingespeist werden.

5.4.1 Startwertberechnung für nichtlineare Simulation

Zur Ausführung nichtlinearer Simulationen sind die Zustände des Simulationsmodells auf korrekte Startwerte zu setzen. Bei den Zuständen des nichtlinearen Gesamtmodells handelt es sich um folgende Vektoren:

- Zustandsvektor der nichtlinearen Flugzeugdynamik entsprechend Kapitel 2

$$\dot{\mathbf{x}} = \left[V_{free} \ \alpha \ \beta \ p \ q \ r \ \Phi \ \Theta \ \Psi \ x_g \ y_g \ z_g \right]^T \quad (5.45)$$

- Zustandsvektor des Aktuatormodells

$$\dot{\mathbf{x}}_{Act} = \left[x_{Act,1} \ x_{Act,2} \ x_{Act,3} \right]^T \quad (5.46)$$

- Zustandsvektor des Notch-Filters

$$\dot{\mathbf{x}}_N = \left[x_{N,1} \ x_{N,2} \right]^T \quad (5.47)$$

- Zustandsvektor der Sensorik

$$\dot{\mathbf{x}}_{Sens} = \left[x_{Sens,1} \ x_{Sens,2} \right]^T \quad (5.48)$$

- Zustandsvektor des Reglers

$$\dot{\mathbf{x}}_{Regler} = \left[\tilde{x} \ x_{W1} \ x_{W2} \right]^T \quad (5.49)$$

Geht man von einem Start-Flugzustand aus, der durch α_{Start} , Ma_{Start} und $q_{dyn,Start}$ definiert ist, folgen hieraus die Startwerte des Zustandsvektors der Flugzeugdynamik. Aus der Kombination aus Ma_{Start} und $q_{dyn,Start}$ ergibt sich eine zugehörige Starthöhe h_{Start} , aus welcher der Startwert der z -Komponente folgt. Mit gegebener Höhe folgt aus der Machzahl die Startgeschwindigkeit $V_{free,Start}$. Der Anstellwinkel ist über den Start-Flugzustand direkt gegeben. Der Schiebewinkel β_{Start} sowie Roll- und Gierwinkel Φ_{Start} und Ψ_{Start} sind Null, ebenso die entsprechenden Raten p_{Start} und r_{Start} . Die Nickrate q_{Start} folgt aus der Trimmtable, es handelt sich um die Nickrate für den Flug auf dem zum Start-Flugzustand gehörenden quasi-stationären Abfangbogen. Der Nickwinkel Θ_{Start} folgt wiederum aus dem Anstellwinkel, beide Winkel sind beim Start der Simulation identisch.

Im Fall des Aktuators muss gelten, dass beim Start der Simulation der Ausgang des Aktuators gleich dem Trimm-Klappenwinkel η_{Start} ist, der zum Start-Flugzustand gehört. Dementsprechend folgt

$$\dot{\mathbf{x}}_{Act,Start} = \begin{bmatrix} 0 & 0 & \eta_{Start}/C_{Act,1,3} \end{bmatrix}^T \quad (5.50)$$

mit dem dritten Element der ersten Zeile der C -Matrix des Aktuatorsmodells $C_{Act,1,3}$.

Das Gleiche gilt für den Notch-Filter, hier gilt

$$\dot{\mathbf{x}}_{N,Start} = \begin{bmatrix} 0 & -\eta_{Start}/A_{N,1,2} \end{bmatrix}^T \quad (5.51)$$

mit dem zweiten Element der ersten Zeile der A -Matrix des Notch-Filter-Modells.

Der Sensor muss beim Start den Start-Anstellwinkel sowie die Start-Nickrate liefern, für seinen Start-Zustand gilt dementsprechend

$$\dot{\mathbf{x}}_{Sens,Start} = \begin{bmatrix} \alpha_{Start}/C_{Sens,1,1} & q_{Start}/C_{Sens,2,2} \end{bmatrix}^T \quad (5.52)$$

mit dem (1,1)- sowie dem (2,2)-Element der C -Matrix des Sensormodells.

Der Startzustand des eigentlichen Reglers wird entsprechend Abschnitt 5.3 berechnet.

5.4.2 Ausgabe der Ergebnisse

Der Verlauf der Zustände bzw. Ausgänge des Gesamtmodells wird während einer laufenden Simulation im Sub-System `CockpitLight` in Form von *Scope*- und *Display*-Blöcken dargestellt. Darüber hinaus werden die Verläufe des gemessenen Anstellwinkels, des kommandierten Anstellwinkels, der Machzahl, des Staudrucks sowie der Klappenstellung des Höhenruders über *ToWorkspace*-Blöcke zur weiteren Auswertung in den Matlab-Workspace geschrieben.

5.5 Nichtlineare Simulationen

Zur Beurteilung des nichtlinearen Reglers erfolgt abschließend die Auswertung diverser nichtlinearer Simulationen. In Analogie zu den Simulationskampagnen in Abschnitt 4.4 erfolgt auch hier eine Simulation an den Flugzuständen der Sets 1 bis 6. Darüber hinaus erfolgen Simulationen, bei denen ausgehend von einem Start-Flugzustand verschiedene Folgen von Anstellwinkelkommandos über einen Zeitraum von rund 200s simuliert werden. Es erfolgt eine Auswertung der Simulationsergebnisse hinsichtlich des Anstiegs- und Überschwingverhaltens der simulierten Reglerkreise, ein Vergleich zwischen der linearen und nichtlinearen Simulation sowie ein Vergleich verschiedener Scheduling-Methoden.

5.5.1 Nichtlineare Simulation an zufälligen Flugzuständen

Es erfolgt zunächst eine Betrachtung des nichtlinearen Regelkreises für Sprünge des kommandierten Anstellwinkels von einem Grad an Start-Flugzuständen entsprechend der Sets zufälliger Flugzustände aus Abschnitt 4.4.1. Während bei der Simulation des linearen Regelkreises in Kapitel 4 der Flugzustand als konstant angenommen wurde und sich dementsprechend das lineare Simulationsmodell während der Simulation nicht veränderte, sind die Flugzustandswerte im nichtlinearen Fall nicht konstant. Dies hat zur Folge, dass je nach Start-Flugzustand einer Simulation einer oder mehrere Zustandswerte die Envelope des Trainers verlassen können. Wird beispielsweise bei einem Flugzustand mit hoher Machzahl und einem Anstellwinkel, für den sich eine hohe positive Geschwindigkeitsableitung ergibt, gestartet, steigt die Machzahl nach Simulationsbeginn weiter an und kann die zulässige Obergrenze von $Ma_{max} = 0,5$ überschreiten. In diesem Fall kann die Simulation nicht beendet werden, das Verhalten am entsprechenden Flugzustand kann also nicht beurteilt werden. Erschwerend kommt hinzu, dass zur Beurteilung des Sprungverhaltens über einen längeren Zeitraum simuliert werden muss als im linearen Fall. Während im linearen Fall für eine Dauer von 20s simuliert wurde und der Anstellwinkelsprung bereits bei $t_S = 0s$ erfolgte, ist im nichtlinearen Fall eine Simulationsdauer von 25s notwendig, wobei der Sprung erst bei $t_S = 12,5s$ erfolgt. Der relativ lange Vorlauf vor dem Sprung ist notwendig, weil sich bei vielen Startflugzuständen eine hohe Ableitung \dot{V}_{free} der Flugeschwindigkeit ergibt. Die große Geschwindigkeitsänderung bewirkt eine schnelle Änderung des Trimmzustands, weshalb es nicht gelingt, den Start-Anstellwinkel zu halten. Die Simulation beginnt zwar mit dem korrekten Anstellwinkel, das schnelle Verlassen des Startzustands führt jedoch dazu, dass sich der Anstellwinkel ebenfalls ändert und der Start-Anstellwinkel quasi eingeregelt werden muss.

Hinweis: Die Verweildauer zwischen Simulationsbeginn und Sprungzeitpunkt bewirkt eine teilweise beträchtliche Veränderung des Flugzustands zwischen Simulationsbeginn und Anstellwinkelsprung. Im linearen Fall erfolgt der Sprung dagegen beim gleichen Flugzustand wie beim Beginn der Simulation. Abbildung 5.5 zeigt exemplarisch die Veränderung des Flugzustands in einer Envelope in $\alpha - Ma - h$ -Darstellung sowie in einer Envelope in $\alpha - Ma - q_{dyn}$ -Darstellung. Die Simulation startet bei einem Flugzustand von $\alpha_{Start} = 20,01^\circ$, $Ma_{Start} = 0,259$ und $q_{dyn,Start} = 1000Pa$. Der Startpunkt ist grün gekennzeichnet, der Flugzustand zum Zeitpunkt des Sprunges rot (es handelt sich im

Beispiel um einen 5°-Sprung von ca. 20° auf 25°) und der Flugzustand am Ende der Simulation ist schwarz eingezeichnet. Bei den grauen Linien handelt es sich um die Projektionen des Flugzustandsverlaufs in die diversen Ebenen der Koordinatensysteme.

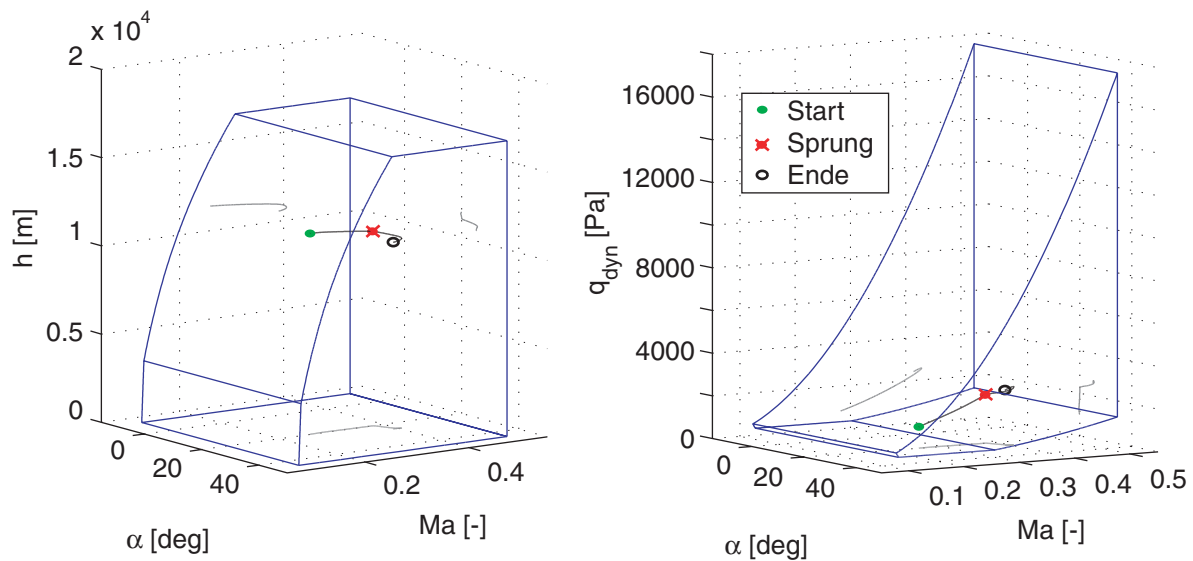


Abb. 5.5: Flugzustandsverlauf für beispielhafte nichtlineare Simulation

Wie aus dem linken Schaubild hervorgeht, ist die Höhenänderung während des Simulationszeitraums bis zum schwarz gekennzeichneten Endpunkt nur relativ gering, das rechte Schaubild zeigt jedoch eine beträchtliche Staudruck- und Machzahlveränderung.

Wie Abbildung 5.5 zeigt, kann bei einer nichtlinearen Simulation ein großer Bereich der Envelope durchlaufen werden und je länger simuliert wird, desto höher ist prinzipiell die Wahrscheinlichkeit, dass das Flugzeug im Laufe der Simulation die Envelope verlässt. Dementsprechend ist es nicht möglich, eine nichtlineare Simulation mit einer Dauer von 25s für alle Startflugzustände abzuschließen. Abbildung 5.6 zeigt analog zu Abschnitt 4.4.1 alle Simulationsstartpunkte in einer dreidimensionalen Envelope in $\alpha - Ma - q_{dyn}$ -Darstellung.

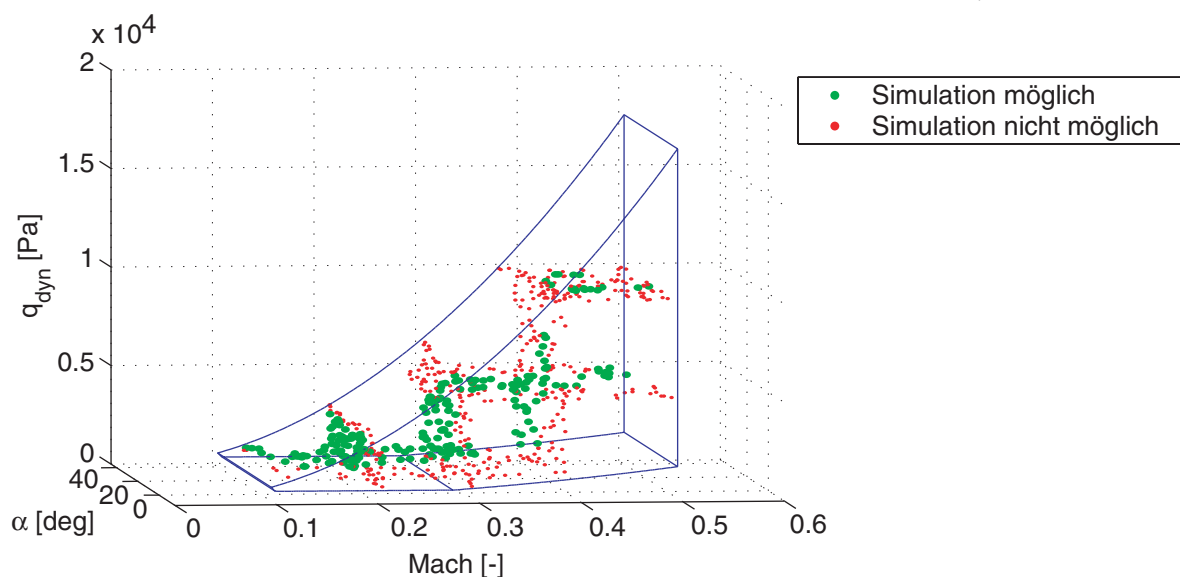


Abb. 5.6: simulierbare und nicht-simulierbare Start-Flugzustände ($t=25s$)

Punkte, für die eine vollständige Simulation möglich ist, sind grün dargestellt, Punkte, für die der Flugzustand während der Simulation den zulässigen Bereich verlässt, sind rot eingezeichnet. Es ist allerdings zu erwarten, dass auch für Start-Flugzustände, für die eine Simulation über den gesamten Zeitraum von 25s möglich ist, Bereiche der Envelope erreicht werden, in denen sich ein problematisches Regelverhalten ergibt. Hierbei handelt es sich einerseits um Flugzustände mit sehr niedrigem Staudruck und andererseits um Punkte am unteren rechten Rand der Envelope für Machzahlen $Ma > 0,3$ und Staudrücken $q_{dyn} < 2000\text{Pa}$, an denen die Interpolation der Reglermatrizenelemente auf Stützstellen basiert, die weit voneinander entfernt liegen. Hierbei ist zu beachten, dass für Punkte, die in Abbildung 5.6 rot gekennzeichnet sind, lediglich kein Start einer Simulation über 25s möglich ist, das Flugzeug kann sich trotzdem im Laufe einer anderen Simulation in einem bzw. in der Nähe eines solchen Flugzustands befinden.

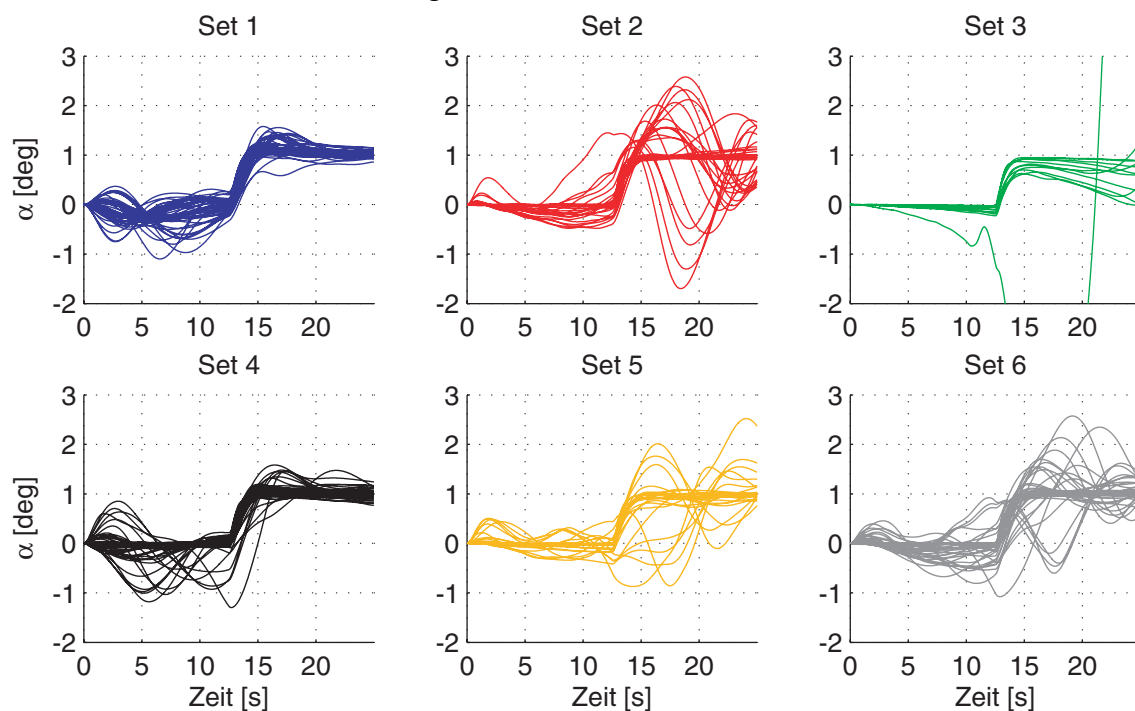


Abb. 5.7: Sprungantworten nichtlinearer Simulationen an zufälligen Flugzuständen

Für die Start-Flugzustände, für die die nichtlineare Simulation möglich ist, ergeben sich für die 6 Sets zufälliger Flugzustände die in Abbildung 5.7 dargestellten Sprungantworten. Zur besseren Darstellbarkeit wurde der Start-Anstellwinkel der jeweiligen Simulation vom Simulationsergebnis subtrahiert. Es ergibt sich so für alle Simulationen ein Anstellwinkelsprung von 0° auf 1° . Der Sprung im Anstellwinkel zum Zeitpunkt $t = 12,5\text{s}$ ist zu erkennen, es zeigt sich eine Häufung der Simulationsergebnisse bei der gewünschten Sprungantwort. Allerdings weisen sämtliche Sets eine relativ große Zahl von Flugzuständen auf, für die starkes Über- bzw. Unterschwingen auftritt, das durch den Regler nicht kompensiert werden kann. Dies wird auch bei Betrachtung der Abbildungen 5.8 und 5.9 deutlich. Hier sind die Verteilungen der Anstiegszeit t_{90} sowie der Überschwingweite uw dargestellt. Vor allem für die Sets 3 und 5 ergibt sich ein großer Anteil von Flugzuständen mit Anstiegszeiten $t_{90} > 5\text{s}$, beim Set 3 wird der Ziel-Anstellwinkel für einen Großteil der

simulierten Flugzustände überhaupt nicht erreicht. Entsprechend ergibt sich für den Großteil der Flugzustände in Set 3 eine negative Überschwingweite.

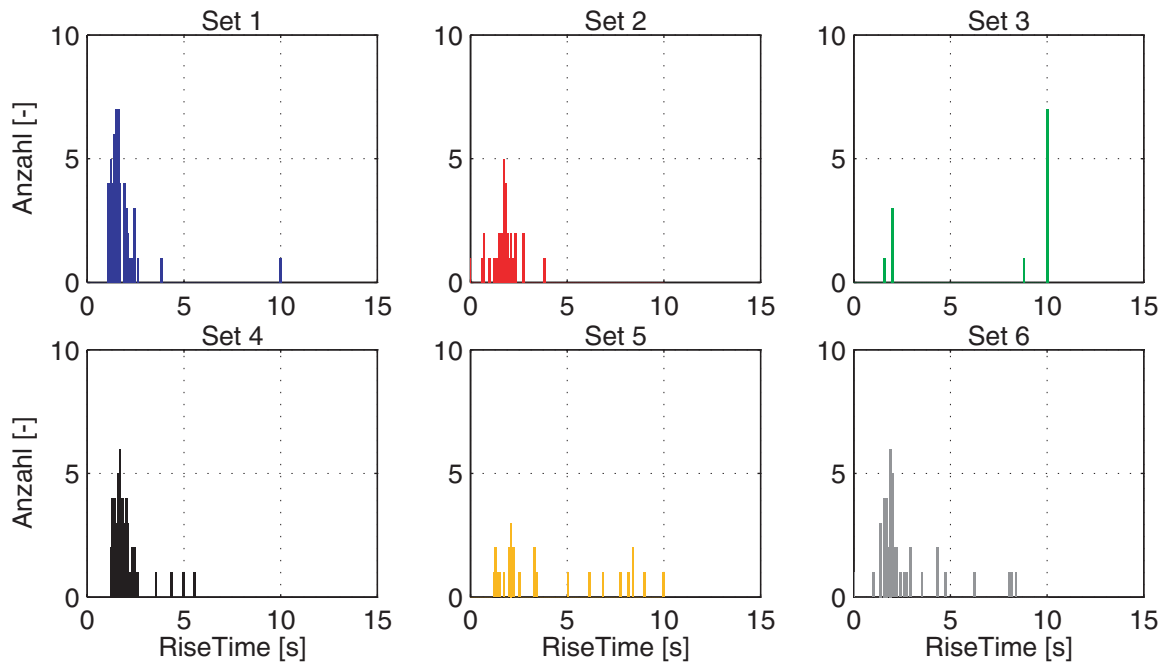


Abb. 5.8: Verteilung der Anstiegszeit für nichtlineare Simulation von Sets zufälliger Flugzustände

Die übrigen Sets zeigen eine Häufung bei der Überschwingweite $uw = 0\%$, in allen Fällen treten jedoch auch mehrere Flugzustände mit sehr großen Überschwingweiten auf.

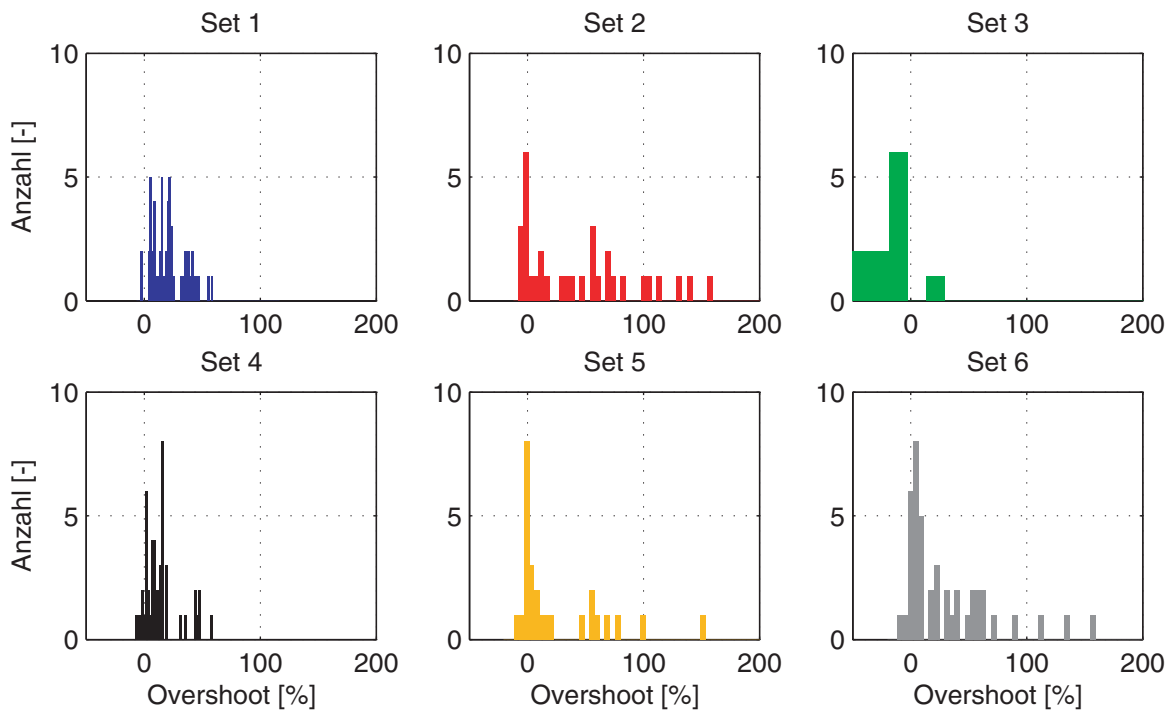


Abb. 5.9: Verteilung der Überschwingweite für nichtlineare Simulation von Sets zufälliger Flugzustände

Abbildung 5.10 zeigt die Start-Flugzustände der nichtlinearen Simulationen farblich gekennzeichnet nach den erreichten Anstiegszeiten und Überschwingweiten. Hierbei zeigt sich, dass sowohl die Flugzustände mit verstärktem Überschwingen als auch die Flugzustände mit erhöhter Anstiegszeit überwiegend im oberen Anstellwinkelbereich liegen. Eine Begründung hierfür sind die für hohe Anstellwinkel verhältnismäßig großen Verzögerungen des Flugzeugs im Trimmzustand auf dem quasi-stationären Abfangbogen. In diesen Fällen verliert das Flugzeug sehr schnell an Geschwindigkeit und befindet sich nach kurzer Zeit in Bereichen der Envelope mit niedrigem Staudruck, in denen sich das Regelverhalten als problematisch erweist.

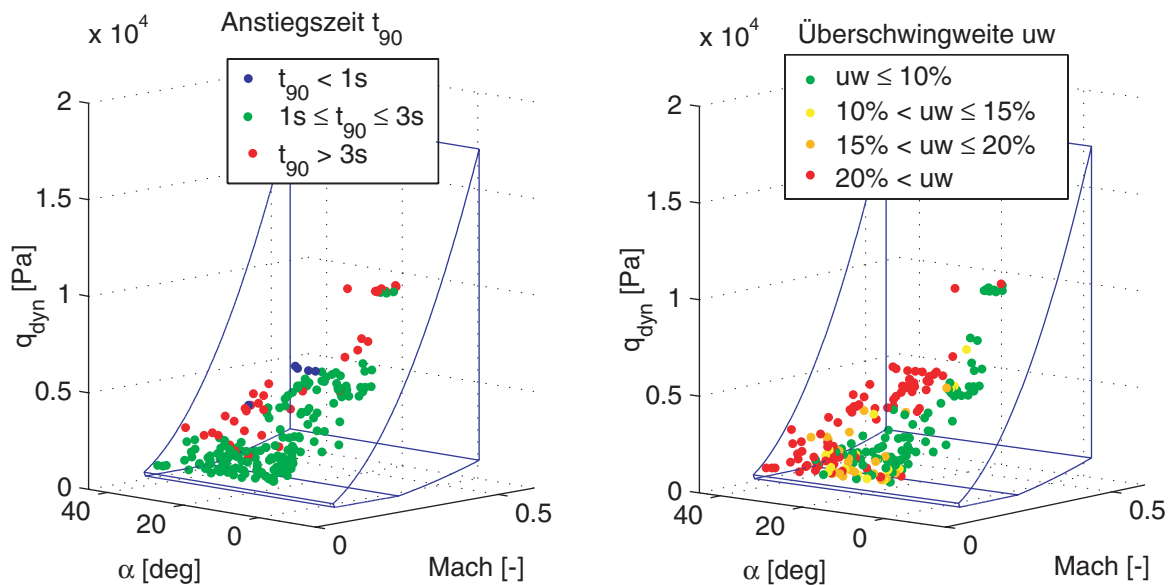


Abb. 5.10: Überschwingweite und Anstiegszeit in 3D-Envelope für nichtlineare Simulation der Sets 1 bis 6.

Der beschriebene Test des Reglers in Form der Simulation von Sprungantworten an zufälligen Flugzuständen führt zu eher ernüchternden Ergebnissen. Mit den verwendeten Entwurfsdaten und dem verwendeten Stützstellenraster ergeben sich keine zufrieden stellenden Simulationsergebnisse, wie auch Tabelle 5.1 verdeutlicht. Zwar haben im Mittel ca. drei Viertel aller Flugzustände eine Anstiegszeit zwischen $t_{90} = 1\text{ s}$ und $t_{90} = 3\text{ s}$, ein Viertel weist jedoch Anstiegszeiten von teilweise deutlich mehr als 3s auf. Hinsichtlich des Überschwingverhaltens ist das Ergebnis noch drastischer, denn nur rund die Hälfte aller getesteten Flugzustände weist eine Überschwingweite von $uw < 10\%$ auf, in über 35% aller Fälle beträgt die Überschwingweite jedoch mehr als 20%.

Merkmal		Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Mittelwert
Risettime	$t_{90} < 1$	0 ^{a)}	12,5	0	0	0	2,3	2,5
	$1 \leq t_{90} \leq 3$	96,3	84,4	33,3	91,8	53,8	79,1	73,1
	$t_{90} > 3$	3,7	3,1	66,7	8,2	46,2	18,3	24,4

Merkmal		Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Mittelwert
Overshoot uw [%]	$uw \leq 10$	29,6	34,4	83,3	46,9	57,7	44,2	49,4
	$uw > 10$	70,4	65,6	16,7	53,1	42,3	55,8	50,6
	$uw > 15$	61,1	59,4	16,7	38,8	38,5	51,2	44,3
	$uw > 20$	48,1	56,3	16,7	16,3	30,8	46,5	35,8

a) alle Angaben in Prozent

Tabelle 5.1: Auswertung Sprungantworten hinsichtlich Anstiegszeiten und Überschwingweiten

Diese schlechten Resultate relativieren sich jedoch aufgrund der Tatsache, dass es sich beim durchgeführten Test um einen Test mit extrem hohen Anforderungen handelt, da sich das Flugzeug aufgrund der Simulationsdauer und der Wahl der Start-Flugzustände teilweise in Flugbereichen mit sehr schwierigen Bedingungen aufhält (niedriger Staudruck, hohe Verzögerungen bzw. Beschleunigungen). Der Test mit Sprüngen des kommandierten Anstellwinkels an zufälligen Flugzuständen im gesamten Einsatzbereich des Flugzeuges zeigt somit die Schwächen des entworfenen Reglers, allerdings für Flugzustände, deren Auftreten in der Realität fraglich ist. Der Vergleich der in Abbildung 5.10 als akzeptabel eingestuften Flugzustände mit den Isoflächen der Geschwindigkeitsableitung auf dem quasi-stationären Abfangbogen in Abbildung 5.11 zeigt, dass für Startpunkte aus einem moderaten Geschwindigkeitsableitungsbereich passable Ergebnisse möglich sind.

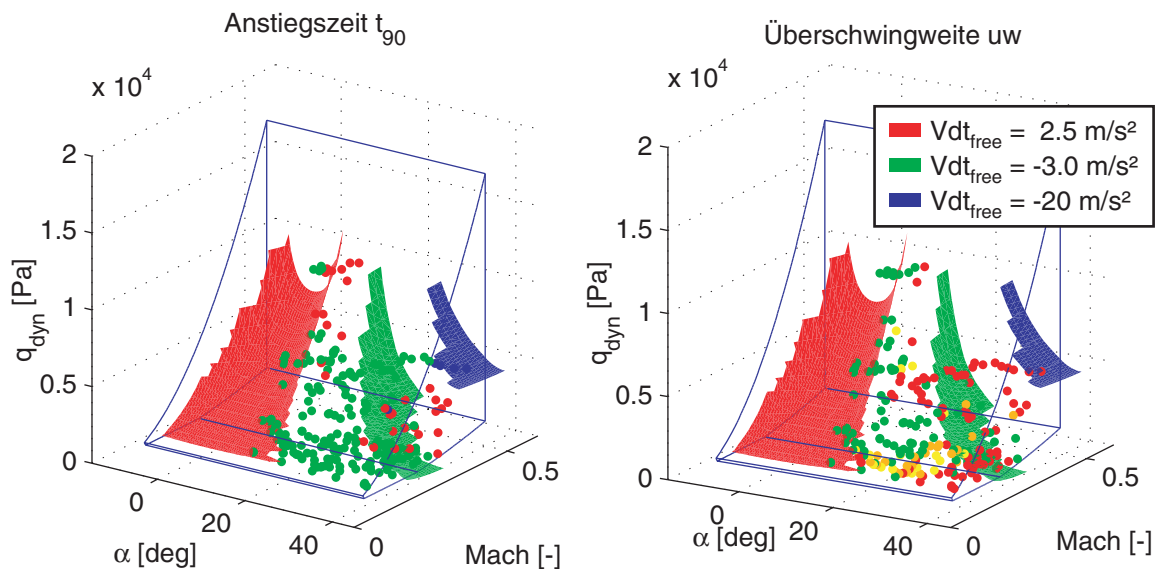


Abb. 5.11: simulierte Flugzustände mit Flächen konstanter Geschwindigkeitsableitung

Eingezeichnet sind die angenäherten Flächen aller Flugzustände mit Geschwindigkeitsableitungen im Trimmfall von $\dot{V}_{free} = 2,5 \text{ m/s}^2$ sowie -3 m/s^2 und -20 m/s^2 . Innerhalb des Bereichs von $2,5 \text{ m/s}^2 \geq \dot{V}_{free} \geq -3 \text{ m/s}^2$ ergeben sich passable Ergebnisse. Für Flugzustände mit sehr hohen Beschleunigungen bzw. Verzögerungen versagt dagegen die Regelstrategie. Dies war zu erwarten, denn es handelt sich um eine lineare Regelung, und obwohl sich das Flugzeug ständig auf einem quasi-stationären Abfangbogen und somit in einem getrimmten Flugzustand mit zugehöriger linearer Strecke befindet, wechselt das Flugzeug zu schnell von einem Trimmzustand zum nächsten.

5.5.2 Nichtlineare Simulation für verschiedene Kommandofolgen

Im vorangegangenen Abschnitt zeigte sich, dass eine Beurteilung der Qualität des Reglers über die Auswertung von Sprungantworten an zufälligen Flugzuständen dazu führt, dass der Regler in Flugzuständen betrieben wird, für die sich ein inakzeptabel schlechtes Regelverhalten ergibt. Im vorliegenden Abschnitt werden deshalb nichtlineare Simulationen für ein treppenförmiges sowie ein rampenförmiges Steuerkommando ausgehend von einem gemeinsamen Startzustand bei $\alpha_{Start} = 15^\circ$, $Ma_{Start} = 0,3$ und $q_{dyn,Start} = 3400\text{Pa}$ durchgeführt. Der Startzustand und die Kommandofolgen sind so gewählt, dass das Flugzeug während der Simulationsdauer von ca. 200s innerhalb der Envelope bleibt und ein möglichst großer Anstellwinkelbereich durchfliegen wird.

Abbildung 5.12 zeigt das Simulationsergebnis für einen treppenförmigen Verlauf des kommandierten Anstellwinkels zwischen 11° und 28° . Der durchflogene Machzahlbereich reicht in diesem Fall von $Ma_{min} \approx 0,21$ bis $Ma_{max} \approx 0,38$, der Staudruckbereich erstreckt sich von $q_{dyn,min} \approx 1500\text{Pa}$ bis $q_{dyn,max} \approx 5050\text{Pa}$. Es ergibt sich hier eine hohe Übereinstimmung zwischen kommandiertem und tatsächlichem Anstellwinkel. Lediglich im Bereich um $\alpha \approx 25^\circ$ ergibt sich ein leicht verstärktes Überschwingen, im Bereich um $\alpha \approx 13^\circ$ ist die Sprungantwort etwas langsamer als bei den übrigen untersuchten Anstellwinkeln.

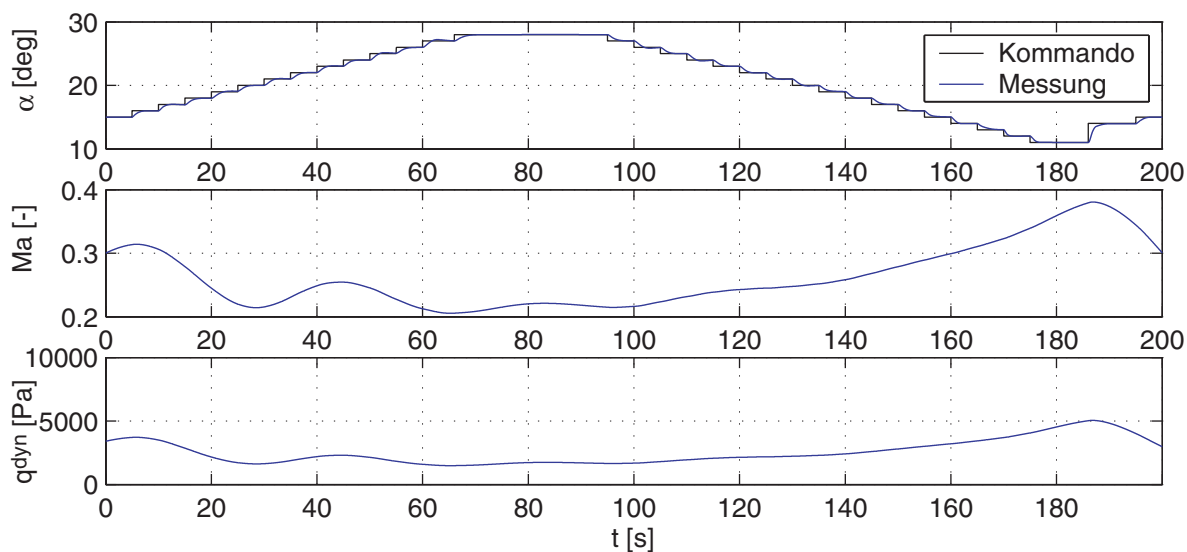


Abb. 5.12: Anstellwinkel-, Machzahl- und Staudruckverlauf für treppenförmiges Kommando

Abbildung 5.13 zeigt den Verlauf des Anstellwinkels sowie der Machzahl und des Staudrucks für ein manuell erzeugtes, rampenförmiges Anstellwinkelkommando.

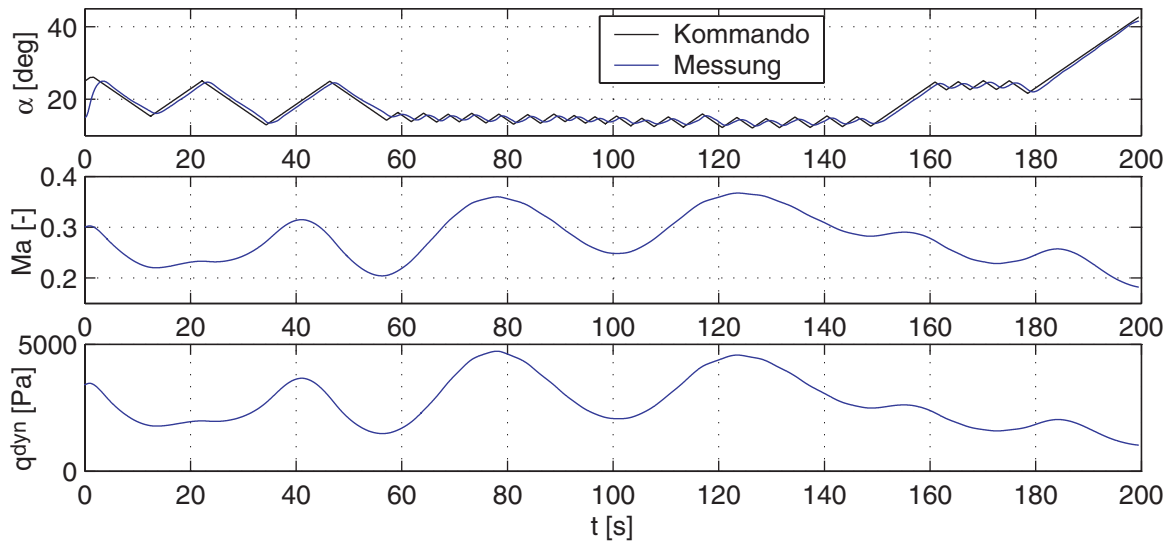


Abb. 5.13: Anstellwinkel-, Machzahl- und Staudruckverlauf für rampenförmiges Kommando

Die Erzeugung des Kommandos erfolgte über einen Joystick, dessen Eingabe über einen *RateLimiter*-Block auf eine Rate von $\pm 1^\circ/s$ limitiert wurde. In diesem Fall werden Anstellwinkel zwischen ca. 12° und 42° erfolgen, der durchflogene Machzahlbereich reicht von $Ma_{min} \approx 0,18$ bis $Ma_{max} \approx 0,37$, der Staudruckbereich erstreckt sich von $q_{dyn,min} \approx 1030\text{Pa}$ bis $q_{dyn,max} \approx 4730\text{Pa}$. Auch in diesem Fall ergibt sich ein gutes Folgeverhalten.

Die nichtlineare Simulation von Flugzuständen mit Anstellwinkeln von weniger als 10° ist prinzipiell möglich, erweist sich jedoch aufgrund der auftretenden großen Beschleunigungen als schwierig, da das Flugzeug die Envelope in diesem Fall sehr schnell verlässt.

5.5.3 Vergleich zwischen nichtlinearer und linearer Simulation

Zur Beurteilung der Verfahren zur Bildung der Referenzzustände beim Übergang vom linearen zum nichtlinearen Regler (siehe Kapitel 5.2) sowie zur Bildung des Rücksetz-Zustands des Gain-Scheduling-Reglers (siehe Kapitel 5.3) erfolgt ein Vergleich zwischen dem Ergebnis der nichtlinearen Simulation mit treppenförmigem Eingang und einer linearen Simulation der entsprechenden Einzelsprünge. Ausgehend vom Ergebnis einer nichtlinearen Simulation mit einem beliebigen sprungförmigen Eingangskommando wird die Regelstrecke über das Skript `LinNonLinCompare01.m` an jedem Flugzustand, an dem ein Kommando-Sprung erfolgte, getrimmt und linearisiert, um anschließend die lineare Simulation des entsprechenden Sprunges durchzuführen. Für den treppenförmigen Eingang aus Abschnitt 5.5.2 ergibt sich bei gemeinsamer Darstellung des nichtlinearen und des linearen Simulationsergebnisses der Verlauf in Abbildung 5.14. Es zeigt sich bis auf wenige Ausnahmen eine hohe Übereinstimmung zwischen der linearen und der nichtlinearen

Simulation. Hieraus folgt, dass die Maßnahmen beim Übergang vom linearen zum nichtlinearen Regler offensichtlich zulässig waren und ihren Zweck erfüllen.

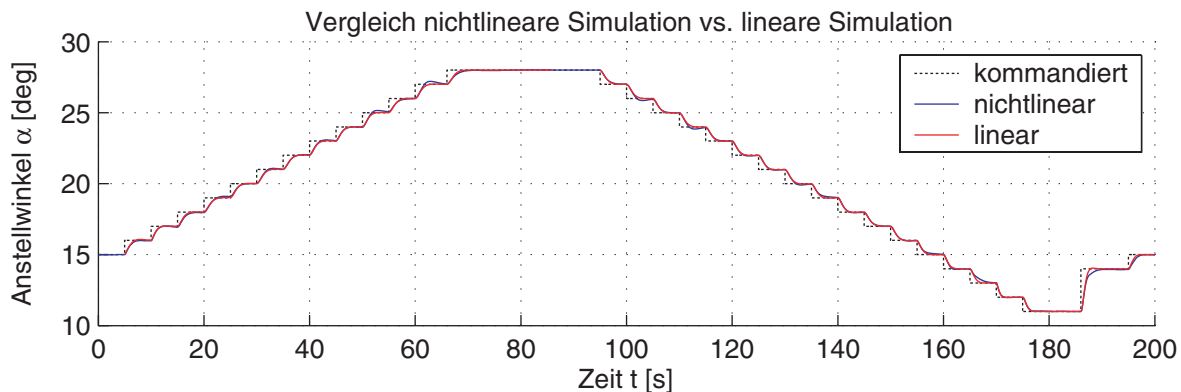


Abb. 5.14: Vergleich zwischen linearer und nichtlinearer Simulation für treppenförmiges Kommando

5.6 Vergleich zwischen ein- und ausgeschaltetem Scheduling

Angesichts des Aufwandes beim Entwurf und Betrieb eines Gain-Scheduling-Reglers stellt sich berechtigterweise die Frage nach der Verbesserung der Regelqualität durch das Scheduling. Hinsichtlich der robusten Stabilität zeigte sich bereits in Abschnitt 4.4.2, dass durch das Scheduling innerhalb der gesamten Envelope gute Robustheitseigenschaften erzielt werden. Hinsichtlich einer Beurteilung des Folgeverhaltens werden die Simulationsergebnisse für den treppenförmigen Eingang aus Abschnitt 5.5.2 bei ein- und ausgeschaltetem Scheduling miteinander verglichen. Abbildung 5.15 zeigt die Verläufe für komplett aktiviertes Scheduling sowie reines α -Scheduling, d.h. die Machzahl- und Staudruckwerte zur Berechnung der Reglermatrizen werden nicht an den momentanen Flugzustand angepasst, sondern sind konstant und betragen $Ma_{Sch} = 0,3$ und $q_{dyn, Sch} = 3400\text{Pa}$.

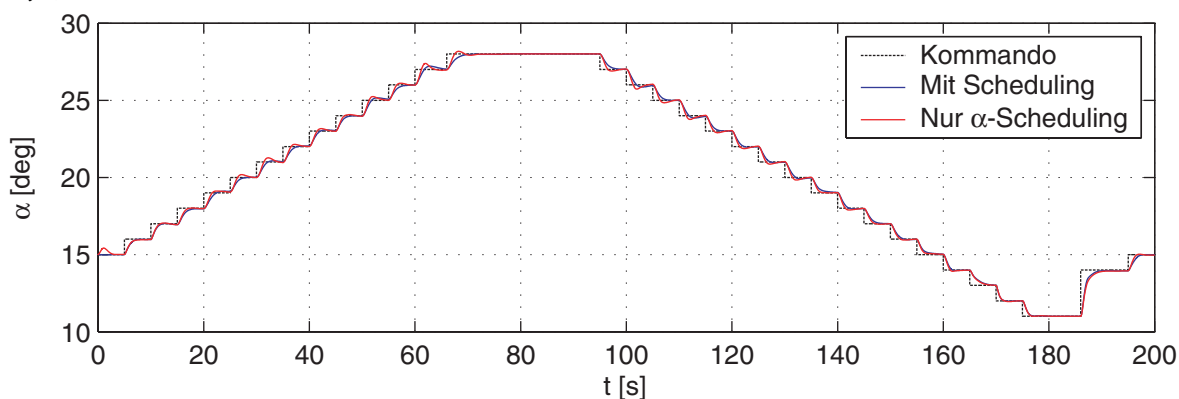


Abb. 5.15: Vergleich zwischen Simulation mit komplettem Scheduling und α -Scheduling

Die Unterschiede erweisen sich als eher gering, bei vielen Sprüngen ergibt sich bei deaktiviertem Ma - und q_{dyn} -Scheduling jedoch ein leicht verstärktes Überschwingen gegenüber dem Anstellwinkelverlauf bei komplett aktiviertem Scheduling.

Ein wesentlich stärkerer Unterschied ergibt sich, wenn außerdem das Anstellwinkel-Scheduling deaktiviert wird. Abbildung 5.16 zeigt den simulierten Anstellwinkelverlauf für konstante α -, Ma - und q_{dyn} -Schedulingwerte und den gleichen treppenförmigen Eingang wie in Abbildung 5.15.

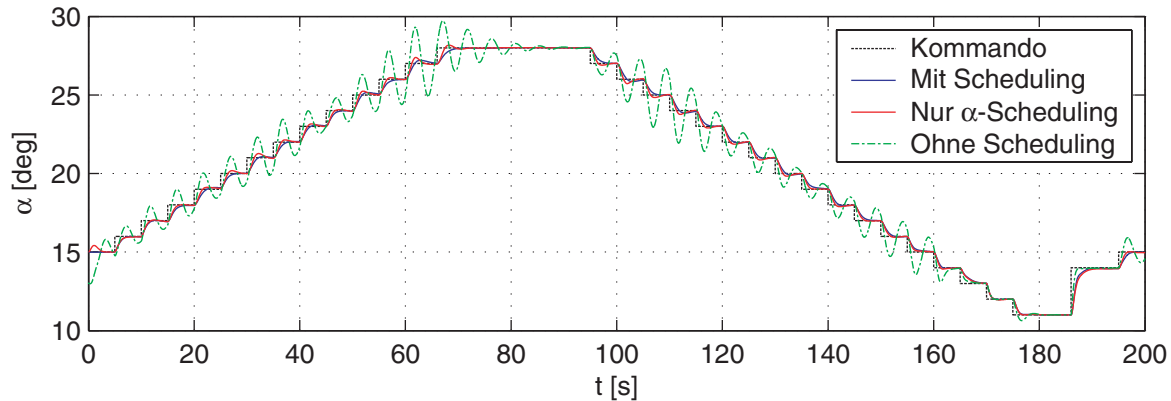


Abb. 5.16: Vergleich zwischen Simulation mit und ohne Scheduling

Im Fall der Machzahl und des Staudrucks werden wiederum die Werte $Ma_{Sch} = 0,3$ und $q_{dyn,Sch} = 3400\text{Pa}$ verwendet. Im Fall des Anstellwinkels wird der Schedulingwert auf $\alpha_{Sch} = 13^\circ$ gehalten. Mit Ausnahme der Sprünge im Anstellwinkelbereich um 13° ergibt sich hierdurch ein sehr starkes Überschwingen. Der Regler, der sich für die gegebenen Scheduling-Parameter ergibt, ist in weiten Teilen des Anstellwinkels zu schnell. Im Gegenzug ergibt sich für einen konstanten Wert $\alpha_{Sch} = 17^\circ$ zwar für die meisten Anstellwinkel ein gutes Folgeverhalten, im Bereich um $\alpha = 13^\circ$ ist die Sprungantwort jedoch zu langsam. Insgesamt zeigen die Abbildungen 5.15 und 5.16 also, dass der Gain-Scheduling-Regler seinen Zweck auch hinsichtlich des Folgeverhaltens erfüllt.

5.7 Vergleich verschiedener Scheduling-Methoden

In Kapitel 3.2 wurden neben dem Verfahren zur Bestimmung der Reglermatrizen an Nicht-Entwurfspunkten durch dreidimensionale Interpolation die Bestimmung durch Geraden- und Parabelregression vorgestellt. Wenngleich in Kapitel 4 aufgrund auftretender Instabilitäten keine Auswertung von linearen Regressions-Reglern durchgeführt wurde, erfolgt an dieser Stelle dennoch ein Vergleich der verschiedenen Implementierungsarten durch Betrachtung der Ergebnisse von nichtlinearen Simulationen. Die nichtlinearen Regelkreise werden wiederum durch den treppen- bzw. rampenförmigen Eingang aus Abschnitt 5.5.2 angesteuert. Für den treppenförmigen Kommandoverlauf ergibt sich der Anstellwinkelverlauf in Abbildung 5.17. Der obere Teil der Abbildung zeigt die Verläufe des kommandierten Anstellwinkels sowie des tatsächlichen Anstellwinkels für eine Interpolation nach der Dreiecks-Ebenen-Methode sowie für eine Geraden- und Parabelregression. In der unteren Hälfte sind die jeweiligen Beträge der Abweichungen zwischen Kommando und tatsächlichem Anstellwinkel dargestellt. Es zeigt sich, dass die Simulationsergebnisse der drei verschiedenen Berechnungsmethoden sehr nahe beieinander liegen. Für den treppenförmigen Kommandoeingang und für den durchflogenen Bereich der Envelope

ergeben die verschiedenen Berechnungsmethoden eine vergleichbar gute Regelqualität. Dies gilt auch für den rampenförmigen Kommandoegang.

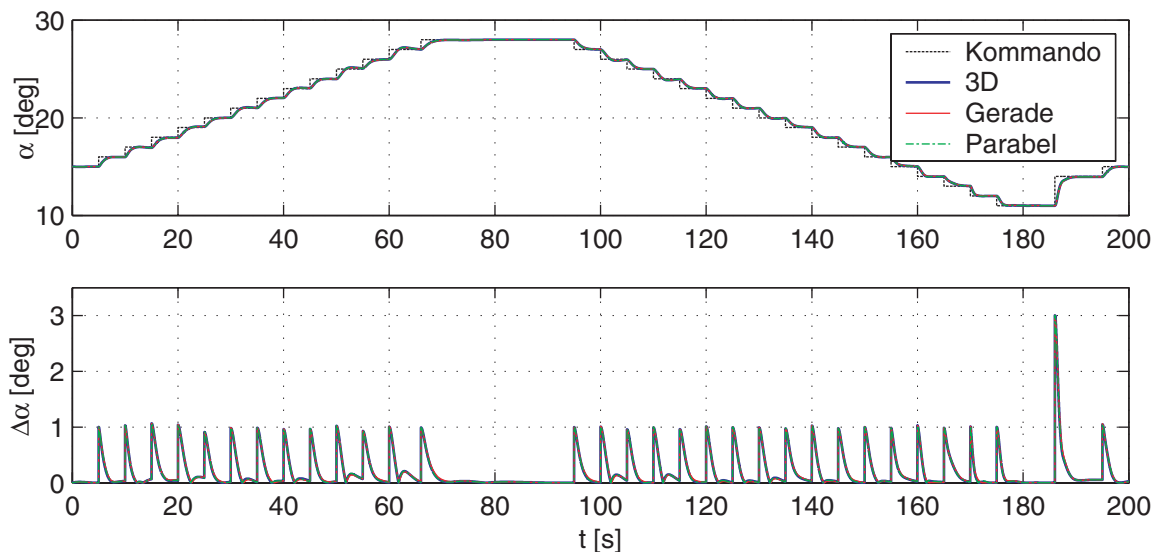


Abb. 5.17: Vergleich verschiedener Scheduling-Methoden für treppenförmigen Eingang

Die entsprechenden Simulationsergebnisse sind in Abbildung 5.18 dargestellt. Auch hier zeigt die obere Hälfte der Abbildung die verschiedenen Anstellwinkelverläufe sowie der untere Teil die Beträge der Abweichungen zwischen kommandiertem und tatsächlichem Anstellwinkel. Es ergibt sich auch in diesem Fall eine sehr gute Übereinstimmung zwischen den einzelnen Scheduling-Methoden.

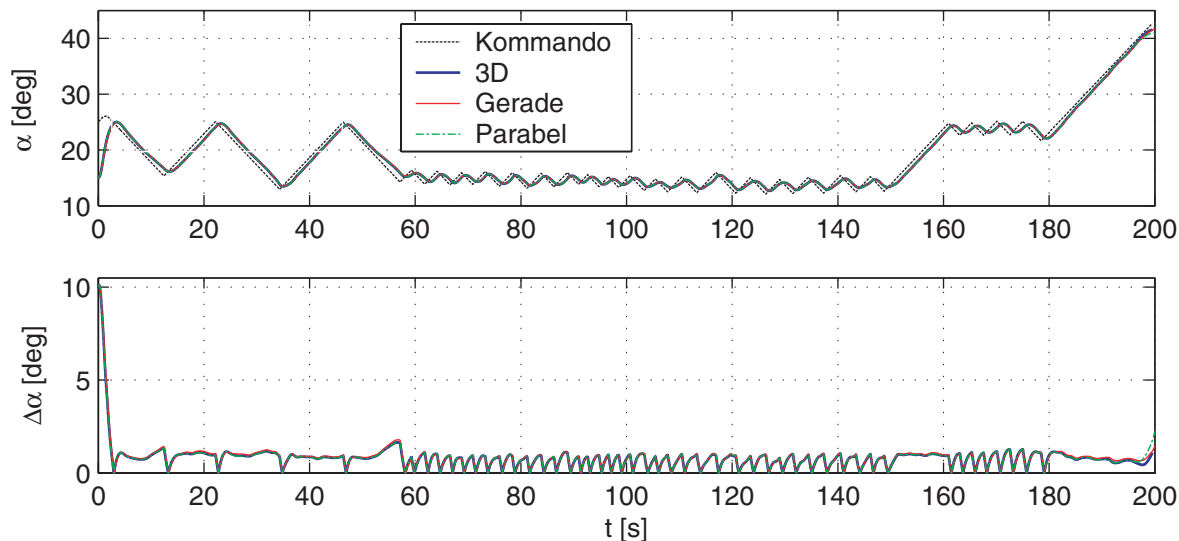


Abb. 5.18: Vergleich verschiedener Scheduling-Methoden für treppenförmigen Eingang

Die Abbildungen 5.17 und 5.18 legen den Schluss nahe, dass sowohl die Geraden- als auch die Parabelregressionsmethode der 3D-Interpolationsmethode gleichwertig sind. Hinsichtlich des Folgeverhaltens des geschlossenen Regelkreises für die im Rahmen der betrachteten nichtlinearen Simulationen durchflogenen Flugzustände ist dies sicherlich richtig. Allerdings kann nicht ausgeschlossen werden, dass es Flugzustände gibt, für die der

Regelkreis bei Bildung der Reglermatrizen über Regressionen instabil wird. Das Folgeverhalten des Regelkreises ist nur ein Kriterium zur Beurteilung des Reglers, ebenso entscheidend sind die Robustheitseigenschaften. Diese sind im Fall der Geraden- und Parabelregressionsmethode schlechter als im Fall der mehrdimensionalen Interpolation. Dies zeigt sich beispielsweise in der Tatsache, dass es, wie in Abschnitt 4.4 erwähnt, für die lineare Simulation zufälliger Flugzustände in mindestens einem Fall zu einem instabilen Regelkreis kommt. Deshalb ist die Verwendung des Interpolationsverfahrens den Regressionsverfahren vorzuziehen.

5.8 Zusammenfassung zum nichtlinearen Regler

Die Auswertung der nichtlinearen Simulationen in Abschnitt 5.5.2 zeigt, dass die Verwendung eines Gain-Scheduling-Reglers zur Realisierung eines stabilitätserhöhenden Anstellwinkelfolgereglers prinzipiell möglich ist. Wie sich in Abschnitt 5.5.1 zeigt, ist es nicht möglich, diesen Regler im gesamten Bereich der Envelope des Jettrainers einzusetzen. In weiten Teilen ergibt sich jedoch ein passables Folgeverhalten. Ausnahmen bilden einzelne Anstellwinkelbereiche, in denen sich eine etwas langsamere Sprungantwort bzw. etwas stärkeres Überschwingen als in der entsprechenden Umgebung ergibt.

Ein direkter Vergleich zwischen nichtlinearer Simulation mit sprungförmigen Eingangskommandos und entsprechenden linearen Simulationen ergibt eine hohe Übereinstimmung und führt zu dem Schluss, dass die gewählte Implementierung des nichtlinearen Reglers mit getaktetem Scheduling bei regelmäßiger Rücksetzung des Regler-Integrators sowie die gewählte Implementierung der Bildung von Referenzwerten eine gültige und funktionsfähige Übersetzung der linearen Einzelregler ist.

Für die durch treppen- bzw. rampenförmige Eingänge getesteten Flugregler ergab sich für verschiedene Schedulingmethoden für die durchflogenen Flugzustände ein stabiles Verhalten des geschlossenen Regelkreises. Diese Beobachtung ist konform zu dem Ergebnis aus Kapitel 4.4: Dort zeigte sich, dass alle getesteten, zufälligen Flugzustände im Fall des linearen Reglers zu einem stabilen Regelkreis führen und sich in nahezu allen Flugzuständen eine Stabilitätsreserve von $\rho \geq 1$ ergibt. Dennoch ist für den nichtlinearen Gain-Scheduling-Regler ein gesonderter Nachweis des Stabilitätsverhaltens erforderlich, da ein Verlust der Stabilitätsreserve durch den Übergang vom linearen zum nichtlinearen Regler nicht ausgeschlossen werden kann (Addition bzw. Subtraktion von Referenzwerten, Rücksetzen des Regler-Integrators).

Kapitel 6

Zusammenfassung

6.1 Durchgeführte Arbeiten und erzielte Resultate

Im Rahmen der vorliegenden Arbeit wurden folgende Teilaufgaben erfolgreich bearbeitet:

- Modellierung der Dynamik des Jettrainers Mako und Bildung eines c-mex s-function basierten Simulinkmodells zur Trimmung, Linearisierung und nichtlinearen Simulation.
- Automatische Trimmung des Jettrainers für ein gegebenes Raster von Trimpunkten und verschiedene Trimmfälle unter besonderer Berücksichtigung der Implizität der Differentialgleichungen.
- Automatische Linearisierung und Bildung der Kurzeitnäherung der Bewegungsgleichungen des Jettrainers basierend auf den Ergebnissen der automatischen Trimmung unter besonderer Berücksichtigung der Implizität der Differentialgleichungen.
- Analyse der ermittelten linearen Modelle.
- Betrachtung des H_∞ -Loop-Shaping-Verfahrens zum Entwurf eines stabilitätserhöhenden Anstellwinkelfolgereglers unter Verwendung des Gain-Scheduling-Verfahrens. Untersuchung verschiedener Implementierungsarten des Reglers.
- Betrachtung und Entwicklung verschiedener mehrdimensionaler Interpolationsverfahren. Diskretisierung der Envelope zur Definition eines Rasters von Reglerentwurfspunkten.
- Entwicklung eines optimierungsbasierten Verfahrens zum automatischen Reglerentwurf für die Kurzeitnäherung beliebiger Flugzustände. Optimierung der Robustheitseigenschaften der Regler unter gleichzeitiger Einhaltung von Forderungen an die Handling-Qualities CAP und ζ_{AS} . Entwicklung zugehöriger Routinen zur Nachbearbeitung und Weiterverarbeitung der Ergebnisse des automatischen Reglerentwurfs.
- Test von linearen Gain-Scheduling-Reglern an zufälligen Flugzuständen hinsichtlich ihres Robustheitsverhaltens und der Einhaltung der Handling-Quality-Vorgaben.
- Überführung des linearen Reglers in einen nichtlinearen Gain-Scheduling-Regler.
- Test des nichtlinearen Reglers an zufälligen Flugzuständen und für treppen- sowie rampenförmige Kommandoeingänge im Betrieb mit dem nichtlinearen Modell des Jettrainers.

Neben den Routinen zur automatischen Trimmung und anschließenden Linearisierung des Jettrainers Mako ist das Verfahren zum automatischen Reglerentwurf eines der wichtigsten Resultate dieser Arbeit. Nach Trimmung, Linearisierung und Generierung eines Satzes von Startwerten gelingt es, für ein sehr heterogenes System mit stark unterschiedlichem dynamischem Verhalten in Abhängigkeit des Flugzustands für nahezu alle Entwurfspunkte einen Flugregler zu entwerfen. Im Fall der Entwurfs-Version *V205* müssen von 1320 Entwurfspunkten weniger als zehn Punkte manuell nachbearbeitet werden. Dies entspricht einem Anteil von rund 0,75%. Die Nachbearbeitung ist dank eines entsprechenden Matlab-Skriptes komfortabel und problemlos möglich. Über dieses Skript besteht außerdem die Möglichkeit, den Verlauf der Reglerparameter in Richtung des Staudrucks bzw. der Höhe an zugehörige Regressionsgeraden bzw. -parabeln anzupassen. Dieser Schritt ist sinnvoll, falls der Einsatz der Geraden- bzw. Parabelregressionsmethode zum Scheduling des Flugreglers in Erwägung gezogen wird.

Die Auswertung der Ergebnisse des automatischen Reglerentwurfes durch die Simulation von linearen Gain-Scheduling-Reglern an zufälligen Flugzuständen zeigt, dass die Robustheitsanforderungen an den Regler in nahezu allen Testpunkten erfüllt werden. In allen Testpunkten ergibt sich ein stabiler Regelkreis. Obwohl die Anforderungen an die Handling-Qualities in allen Entwurfspunkten erfüllt werden, ist es nicht möglich, sie auch in allen Testpunkten zu erfüllen. Vor allem die Forderung eines *CAP*-Wertes nach Level 1* erweist sich als zu anspruchsvoll. Hinsichtlich der Anstellwinkelschwingungsdämpfung sind die Ergebnisse besser, es müssen jedoch ebenfalls Abstriche gemacht werden. Begründet werden diese Schwächen des Verfahrens zum einen durch die problematische und empfindliche Berechnung der AS-Dämpfung und vor allem des *CAP*-Wertes. Zum anderen erweist sich das verwendete Entwurfsraster als stellenweise nicht fein genug. Auf eine weitere Verfeinerung wird jedoch im Hinblick auf den damit verbundenen Mehraufwand bei der Reglerberechnung sowie beim Scheduling im späteren Betrieb des Reglers verzichtet. Die Annahme, durch den Entwurf robuster Flugregler lasse sich die Anzahl der Entwurfspunkte reduzieren, bewahrheitet sich also nicht.

Es gelingt, den linearen Regler in eine nichtlineare Form zu überführen. Der Vergleich von Simulationsergebnissen des nichtlinearen Systems bei einer treppenförmigen Anregung mit Ergebnissen von linearen Simulationen an entsprechenden Flugzuständen mit entsprechenden Eingängen zeigt eine hohe Übereinstimmung.

Die nichtlineare Simulation von Sprungantworten an zufälligen Flugzuständen aus dem gesamten Entwurfsbereich führt aufgrund zu hoher Fluggeschwindigkeitsableitungen teilweise zu schlechten Ergebnissen. Für treppen- und rampenförmige Eingänge ergibt sich innerhalb eines erfliegbaren Bereichs der Envelope jedoch ein stabiles Verhalten des Regelkreises und eine hohe Übereinstimmung zwischen dem gemessenen und dem kommandierten Anstellwinkel. Dies gilt sowohl für die Bestimmung der variablen Reglerparameter über mehrdimensionale Interpolation als auch für eine Bestimmung über Geraden- oder Parabelregression.

6.2 Fazit und Ausblick

Die Ergebnisse der vorliegenden Arbeit zeigen, dass die Verwendung des H_∞ -Loop-Shaping-Verfahrens in Verbindung mit dem Gain-Scheduling-Verfahren zur Realisierung eines stabilitätserhöhenden Anstellwinkelregelers prinzipiell möglich ist.

Ob und wie sich die im linearen Entwurf erzielten Robustheitseigenschaften auf den nichtlinearen Regelkreis und dessen Stabilitätsverhalten übertragen lassen, ist in einer gesonderten Analyse zu untersuchen.

Des Weiteren ist zu überprüfen, ob durch weitere Verbesserungen des Verfahrens zum automatischen Reglerentwurf die Einhaltung der Handling-Qualities an Nicht-Entwurfspunkten auch ohne eine Verfeinerung des Entwurfsrasters verbessert werden kann. Hierzu kommen beispielsweise Verbesserungen der Verfahren zur Berechnung der Kostenfunktion (ρ) sowie der Beschränkungsfunktionen (CAP und ζ_{AS}) des Optimierungsverfahrens in Betracht. Weiteres Verbesserungspotenzial liegt in der Startwertgenerierung des automatischen Reglerentwurfs. Bisher gehen in die Startwertbildung (außer an den Rändern der Envelope) lediglich die Ergebnisse von vorangegangenen Entwürfen benachbarter Staudrücke bzw. Flughöhen der gleichen Machzahl und des gleichen Anstellwinkels ein. Um eine möglichst glatte Reglerparameterfläche in Richtung aller Scheduling-Parameter zu erhalten, ist bei der Startwertgenerierung auch die Berücksichtigung der Berechnungsergebnisse benachbarter Machzahlen und Anstellwinkel sinnvoll. Darüber hinaus ist durch die weitere Auswertung nichtlinearer Simulationen zu überprüfen, ob die teilweise Verletzung der Anforderungen an die Handling-Qualities an Nicht-Entwurfspunkten toleriert werden kann.

Literatur

- [1] Fischer, Balasz: *Entwurf eines Reglers für die Längsbewegung des Trainingsflugzeugs Mako*. Universität Stuttgart, Institut für Flugmechanik und Flugregelung, Diplomarbeit, 1999.
- [2] Stevens, Brian L.; Lewis, Frank L.: *Aircraft Control and Simulation, 2nd Edition*. Hoboken, New Jersey, Wiley, 2003.
- [3] Brockhaus, Rudolf: *Flugregelung. Zweite Auflage*. Berlin, Springer, 2001. ISBN 3-540-41890-3.
- [4] Papageorgiou, George; Glover, Keith; Smerlas, Alex; Postlethwaite, Ian: *H_∞ -Loop-Shaping*. In: Magni, Jean-François (Hrsg.); Bennani, Samir (Hrsg.); Terlouw, Jan (Hrsg.): *Robust Flight Control: A Design Challenge*. Berlin, Springer, 1997 (lecture Notes in Control and Information Sciences 224). Kap. 7, S. 64-80.
- [5] Skogestad, Sigurd; Postlethwaite, Ian: *Multivariable Feedback Control: Analysis and Design*. Chichester, Wiley, 1996.
- [6] Hyde, Richard Alden: *H_∞ Aerospace Control Design: A VSTOL Flight Application*. Berlin, Springer, 1995 (Advances in Industrial Control).
- [7] Levine, William S. (Hrsg.): *The Control Handbook*, Boca Raton (FL,USA), CRC Press, 1996.
- [8] Horn, M.K.: *Subroutines for Handling Tabular Data Used in System Equations*. Oberpfaffenhofen, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt, Institut für Dynamik der Flugsysteme, 1982.
- [9] The MathWorks, Inc. (Hrsg.): *Matlab, The Language of Technical Computing, Matlab Function Reference, Volume 2: F - O, Version 6*. Natick (MA,USA), The MathWorks Inc., 2001.
- [10] Normenstelle Luftfahrt im DIN Deutsches Institut für Normung e.V. Berlin (Hrsg.): *Begriffe, Größen und Formelzeichen der Flugmechanik. Teil 1: Bewegung des Luftfahrzeugs gegenüber der Luft. ISO 1151-1: 1988 modifiziert. DIN 9300, Teil 1*. Berlin, Beuth Verlag GmbH, 1990.
- [11] Normenstelle Luftfahrt im DIN Deutsches Institut für Normung e.V. Berlin (Hrsg.): *Normatmosphäre DIN ISO 2533*. Berlin, Beuth Verlag GmbH, 1979.
- [12] Pertermann, Christian: *Voruntersuchung zum H_∞ -Reglerentwurf für die Längsbewegung eines leichten generischen Kampfflugzeugs*. Stuttgart, Universität Stuttgart, Institut für Flugmechanik und Flugregelung, Technischer Bericht, 2000
- [13] Dr. Heller, Matthias: *Technische Anleitung IFR Datentransfer*, EADS München, Anhang an Mail vom 27.02.2002

- [14] Dr. Heller, Matthias; Paul, Ralph.: *EADS Linearisierungsergebnisse*. EADS München, Mail vom 19.05.2003
- [15] The MathWorks, Inc. (Hrsg.): *Matlab, The Language of Technical Computing, Matlab Function Reference, Volume 1: A - E, Version 6*. Natick (MA,USA), The MathWorks Inc., 2001.
- [16] Grimm, Werner; Well, Klaus H.: *Nichtlineare Optimierung*. Stuttgart, Universität Stuttgart, Institut für Flugmechanik und Flugregelung, Skriptum, 2001.
- [17] The MathWorks, Inc. (Hrsg.): *Optimization Toolbox, For Use with Matlab, User's Guide Version 2*. Natick (MA,USA), The MathWorks Inc., 1999.
- [18] Föllinger, Otto; Dörrscheidt, Frank (mitwirkend): *Regelungstechnik: Einführung in die Methoden und ihre Anwendung*. Heidelberg, Hüthig, 1992.
- [19] Joos, Dieter: *Direkte Anwendung von Flugeigenschaften zum Entwurf von robusten SAS/CAS*. München, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 1993.
- [20] Shamma, Jeff, S.; Athans, Michael: *Gain Scheduling: Potential Hazards and Possible Remedies*. In: IEEE Control Systems Magazine, Juni 1992, S. 101 - 107, IEEE Control Systems Society, 1992.
- [21] Hanel, Martin; Kämpf, Bernhard: *Robuste Regelung*. Stuttgart, Universität Stuttgart, Institut für Flugmechanik und Flugregelung, Skriptum, 1999.
- [22] Raisch, Jörg; Gilles, Ernst Dieter: *Reglerentwurf mittels H_∞ -Minimierung: Eine Einführung. Teil 2*. In: Automatisierungstechnik 40, Nr. 4, S.123-131, 1992.
- [23] The MathWorks, Inc. (Hrsg.): *SIMULINK, Model-Based and System-Based Design, Simulink Reference, Version 5*. Natick (MA,USA), The MathWorks Inc., 2002.

Anhang A

Simulink-Blockdiagramme

A.1 Nichtlineares Simulink-Modell für Trimmung und Linearisierung

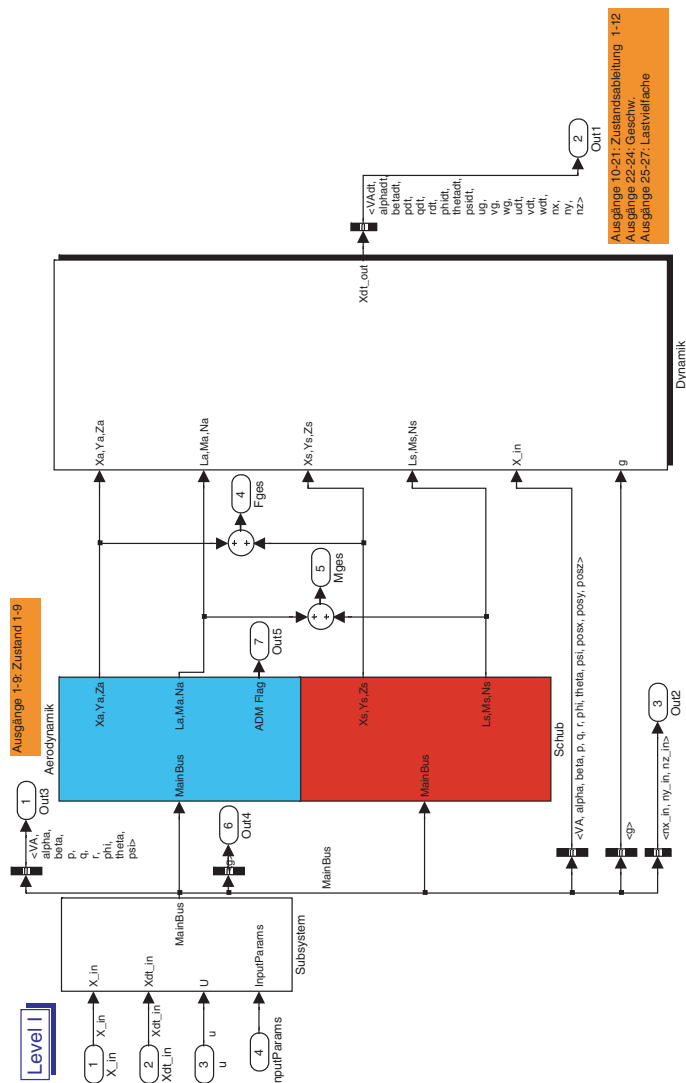


Abb. A.1: Modell der nichtlinearen Modellierung für Trimmung und Linearisierung

A.2 Simulink-Blockdiagramm nichtlineares Simulationsmodell

A.2.1 Gesamtmodell (Ebene I)

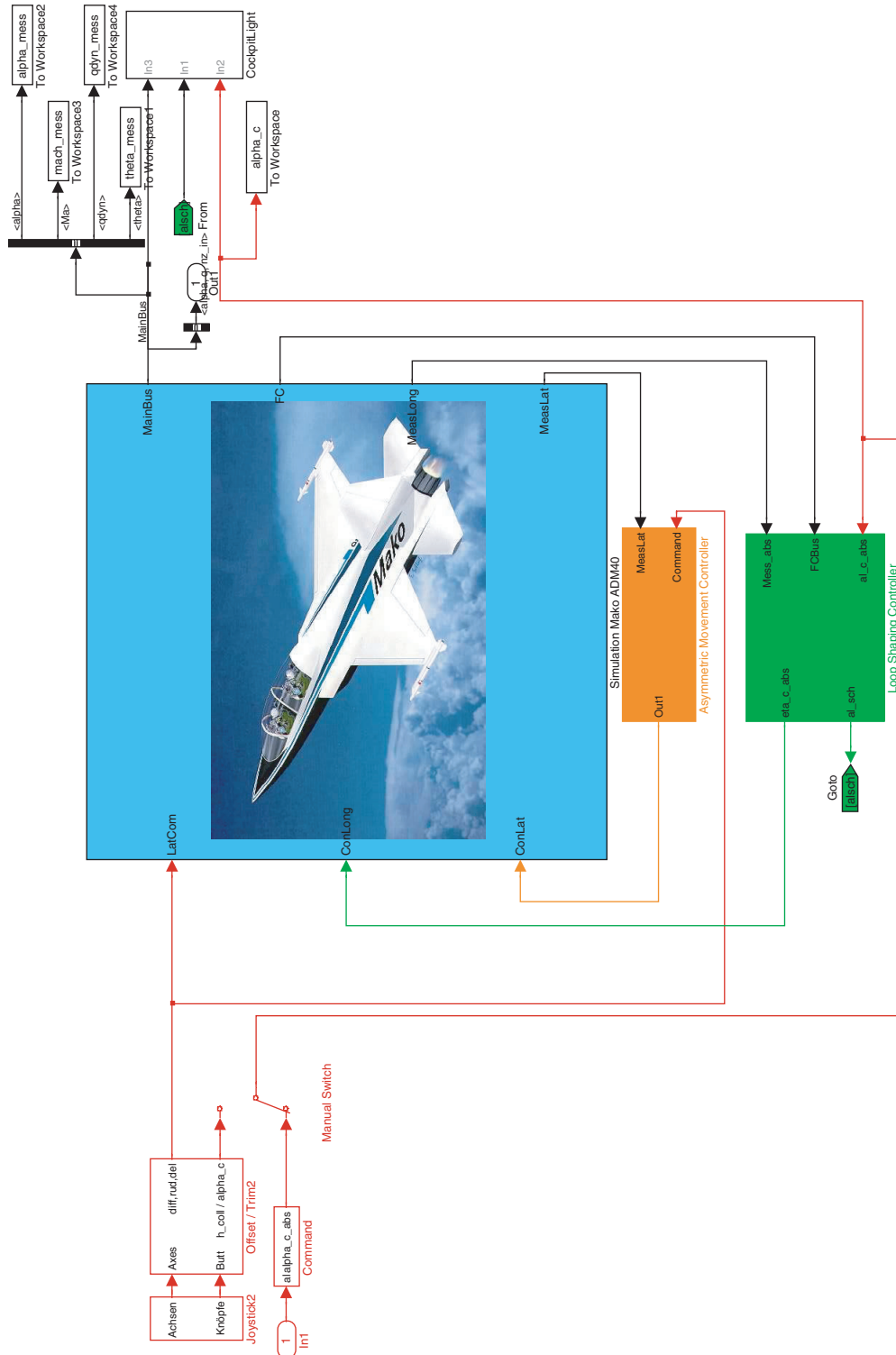


Abb. A.2: Simulink-Blockdiagramm der Ebene I des nichtlinearen Simulationsmodells

A.2.2 Gain-Scheduling-Regler (Ebene II)

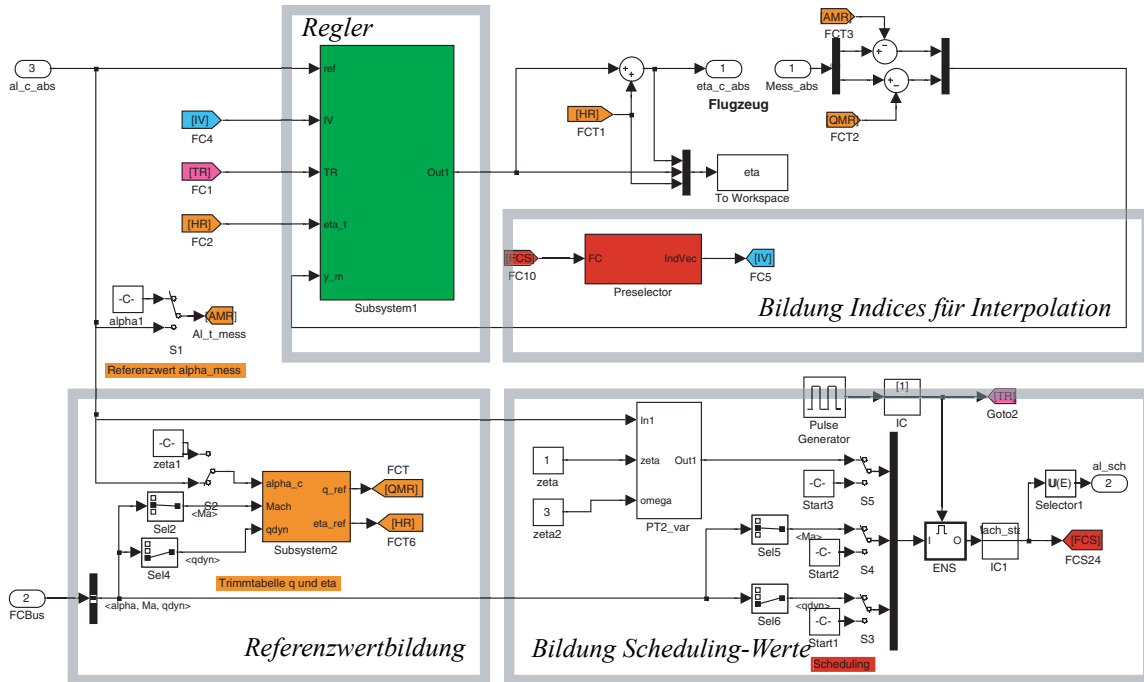


Abb. A.3: Simulink-Blockdiagramm des nichtlinearen Gain-Scheduling-Reglers (Ebene II)

A.2.3 Gain-Scheduling-Regler (Ebene III)

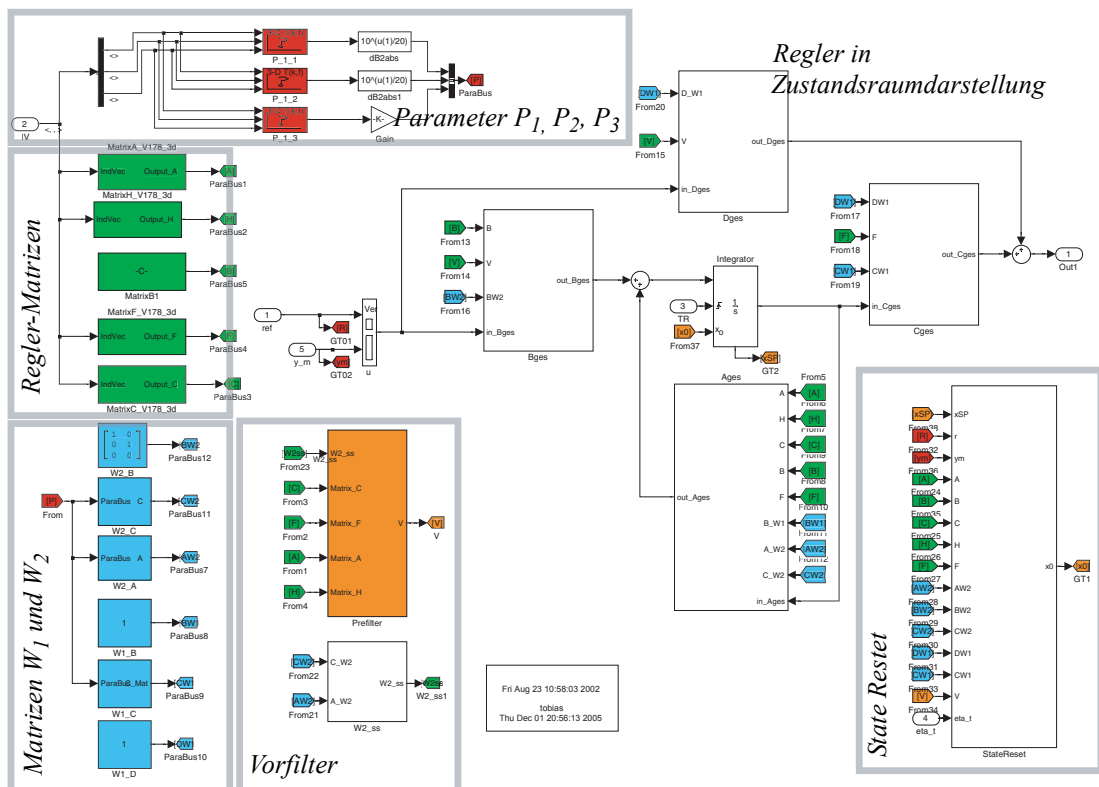


Abb. A.4: Simulink-Blockdiagramm des nichtlinearen Gain-Scheduling-Reglers (Ebene III)

A.2.4 Blockdiagramm Zustandsrücksetzung (State Reset, Ebene IV)

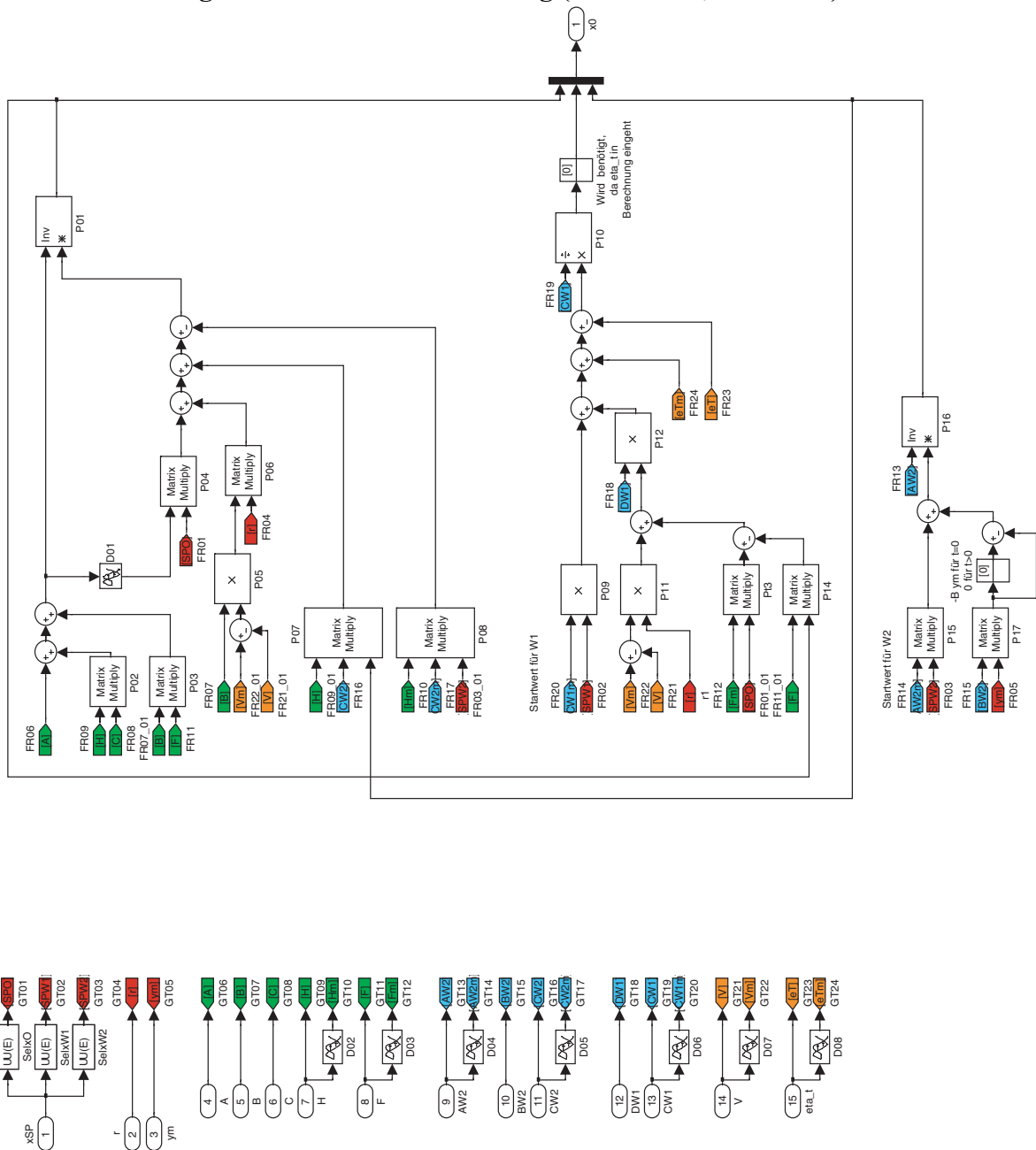


Abb. A.5: Simulink-Blockdiagramm der Zustandsrücksetzung des Gain-Scheduling-Reglers (Ebene IV)

Anhang B

Struktogramme

B.1 Struktogramm der Trimmroutine `trimenvqdyn`

<i>trimcase</i>	Auswahl der Trimm-Art
Setzen verschiedener Einstellungen	
Spezifizierung der zu trimmenden Flugzustände (FZ) über Begrenzung eines zu trimmenden Bereichs.	
Erzeugung der Listen der zu trimmenden Anstellwinkel, Machzahlen etc.	
Initialisierung des Simulink-Modells der Bewegungsgleichungen (liefert Vektor der Zustandsableitungen in Abh. Des Zustandsvektors, des Steuervektors sowie der Zustandsableitung selbst)	
<i>trimplist.m</i>	Sortierung der Flugzustände und Bildung der Matrix <i>tplist_tmp</i> in Abhängigkeit des <i>trimcase</i> . Matrix enthält in ihren Zeilen alle zu trimmenden Flugzustände.
<i>inenv01.m</i>	Überprüfung ob und Markierung der Flugzustände, die innerhalb der Envelope liegen. Hinweis: Lastvielfache im Trimmzustand an dieser Stelle noch nicht bekannt, deshalb Verwendung eines Blanko-Wertes.
Initialisierung diverser Matrizen und Vektoren	
Initialisierung der Ausgabe-Matrix <i>trimlist</i>	
Initialisierung des Ausgabe-Cell-Arrays <i>trimarray</i>	
FOR	Schleife über Zeilen der Flugzustandsmatrix (FZM) <i>tplist_tmp</i> zur Bearbeitung jedes Flugzustands

Fortsetzung auf folgender Seite...

Abb. B.1: Nassi-Shneiderman Struktogramm der Trimmroutine `trimenvqdyn`, Teil 1

...Fortsetzung

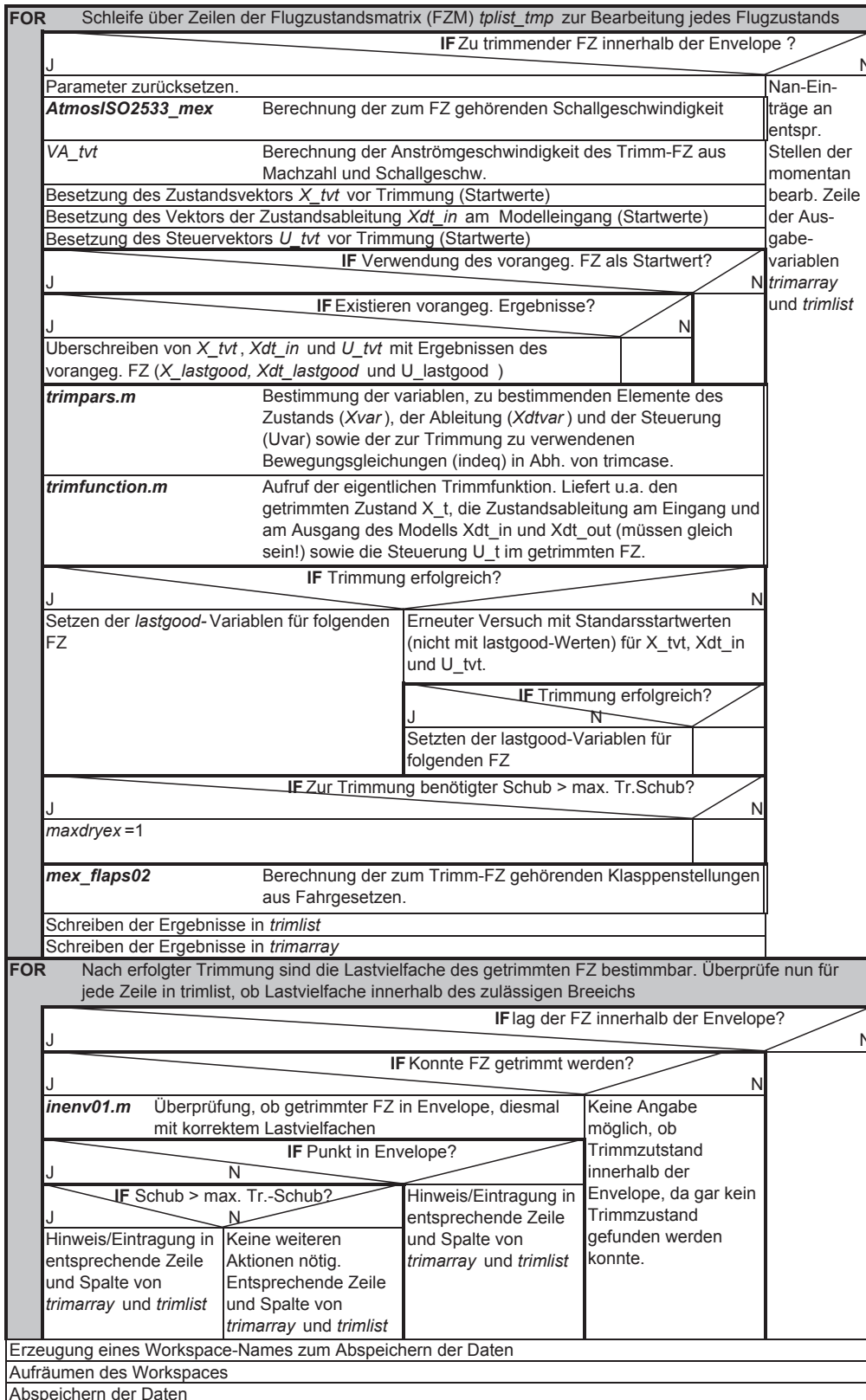
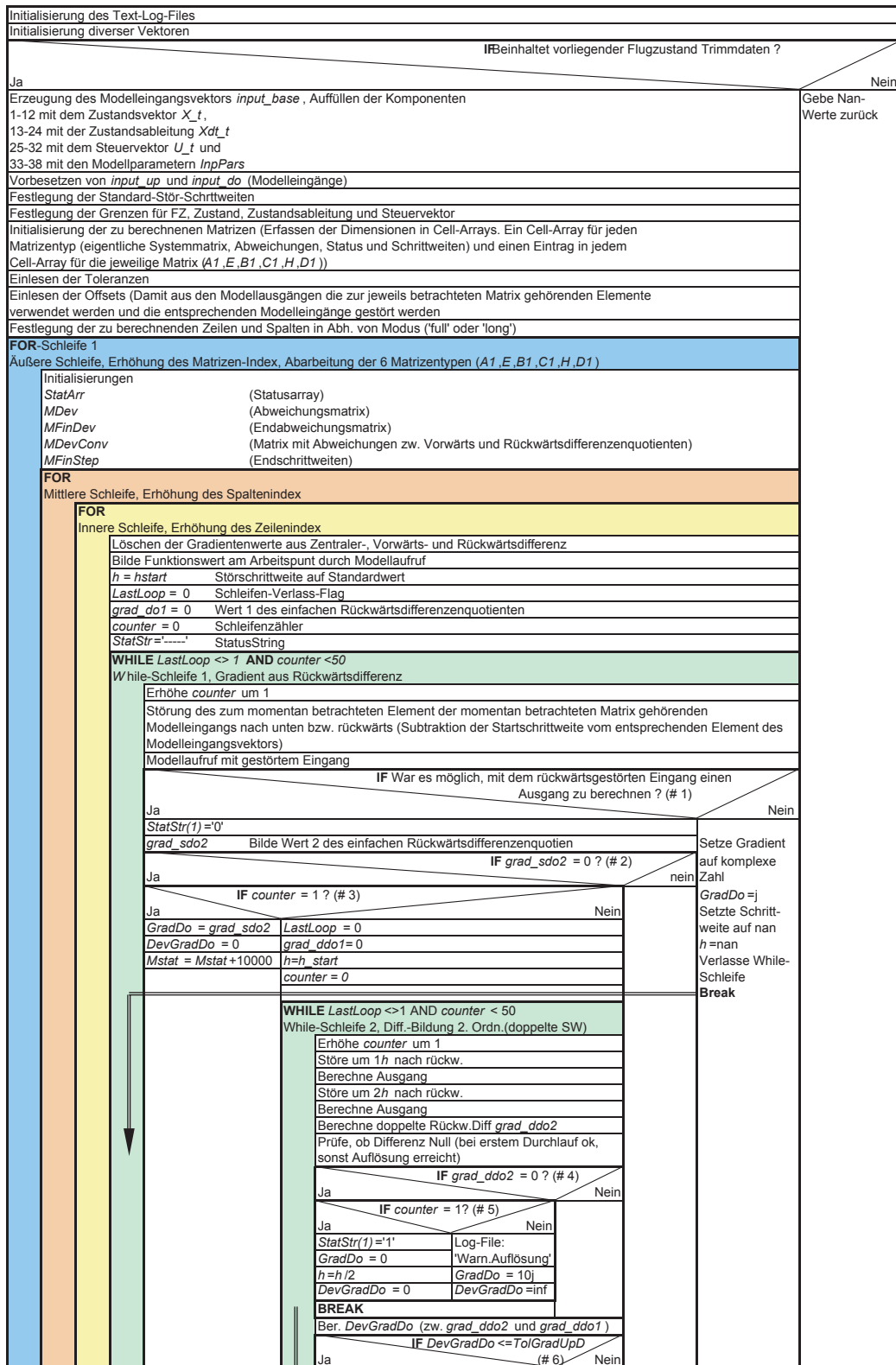


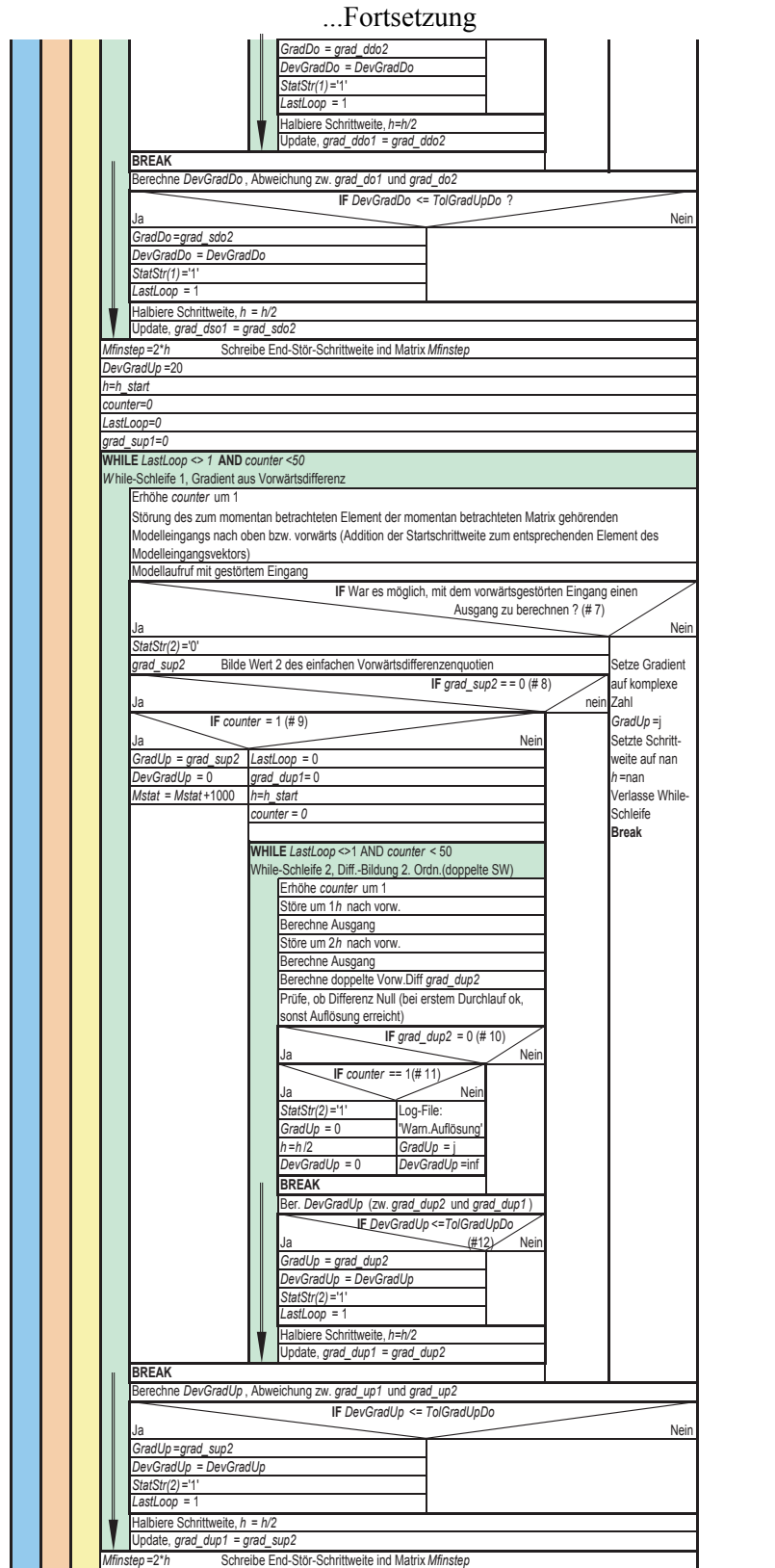
Abb. B.2: Nassi-Shneiderman Struktogramm der Trimmroutine *trimenvqdyn*, Teil 2

B.2 Struktogramm der Linearisierungsfunktion `linfun10`



Fortsetzung auf folgender Seite...

Abb. B.3: Struktogramm der Linearisierungsfunktion `linfun10`, Teil 1



Fortsetzung auf folgender Seite...

Abb. B.4: Struktogramm der Linearisierungsfunktion 1 in fun10, Teil 2

...Fortsetzung

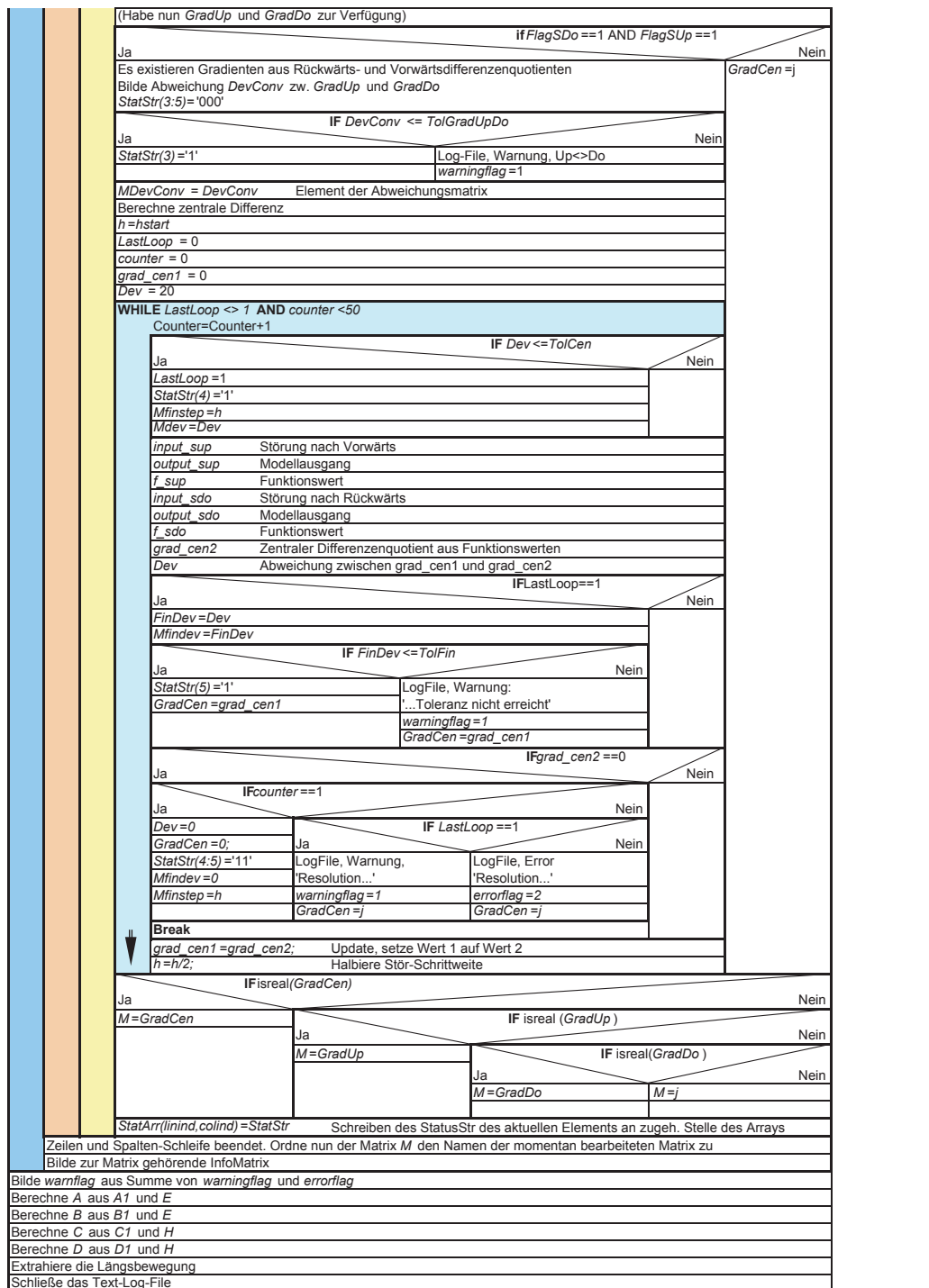


Abb. B.5: Struktogramm der Linearisierungsfunktion linfun10, Teil 3

B.3 Baumstruktur der Wiederversuchslogik

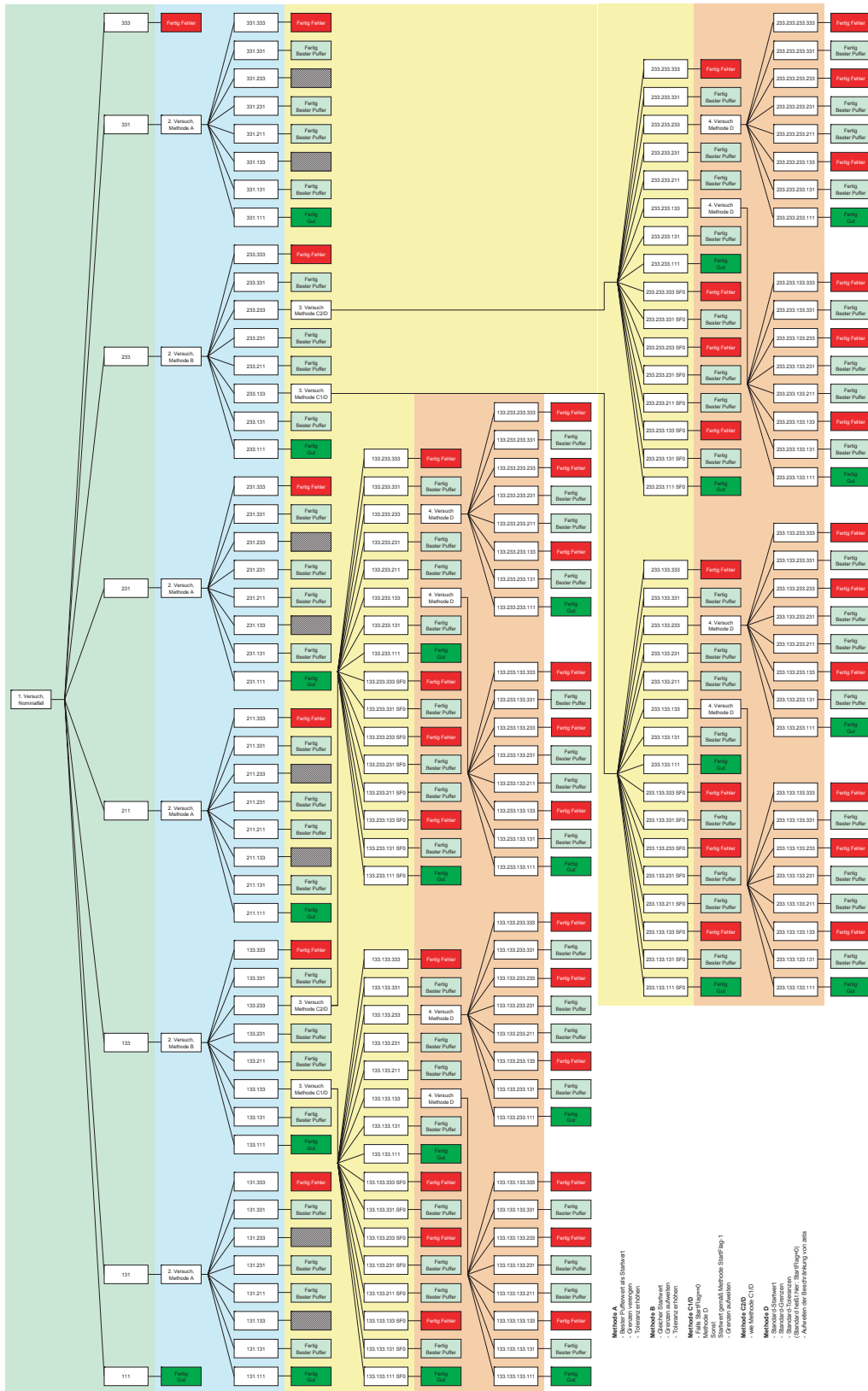


Abb. B.6: Baumstruktur der Wiederversuchslogik des automatischen Reglerentwurfs

Anhang C

CAP-Bestimmung

C.1 Erweiterte MIL-Forderungen für die Auslegung gemäß CAP nach Level 1*

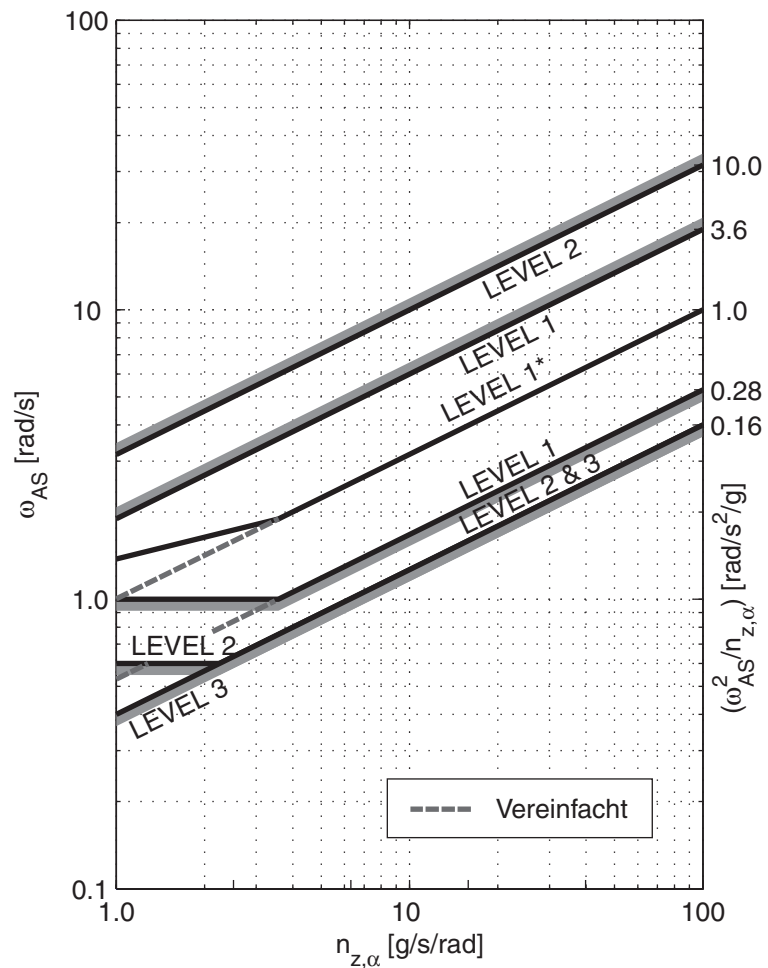


Abb. C.1: Diagramm zur Reglerauslegung gemäß CAP nach Level 1*

Anhang D

Lineares Entwurfsmodell

D.1 Lineares Entwurfsmodell, Regler im Vorwärtszweig

D.1.1 Zustandsraumdarstellung

Systemmatrix und Eingangsmatrix:

$$\begin{bmatrix} \dot{x}_{W1} \\ \dot{x}_E \\ \dot{x}_{W2} \\ \dot{x}_O \end{bmatrix} = \begin{bmatrix} A_{W1} & 0 & 0 & 0B_{W1}F \\ B_EC_{W1} & A_E & 0 & B_ED_{W1}F \\ 0 & B_{W2}C_E & A_{W2} & 0 \\ 0 & 0 & -HC_{W2} & A + BF + HC \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -B_{W2} \\ 0 \end{bmatrix} \cdot r \quad (\text{D.1})$$

Ausgangsmatrix:

$$y = \begin{bmatrix} 0 & C_E & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} = y_E \quad (\text{D.2})$$

D.1.2 Blockdiagramm

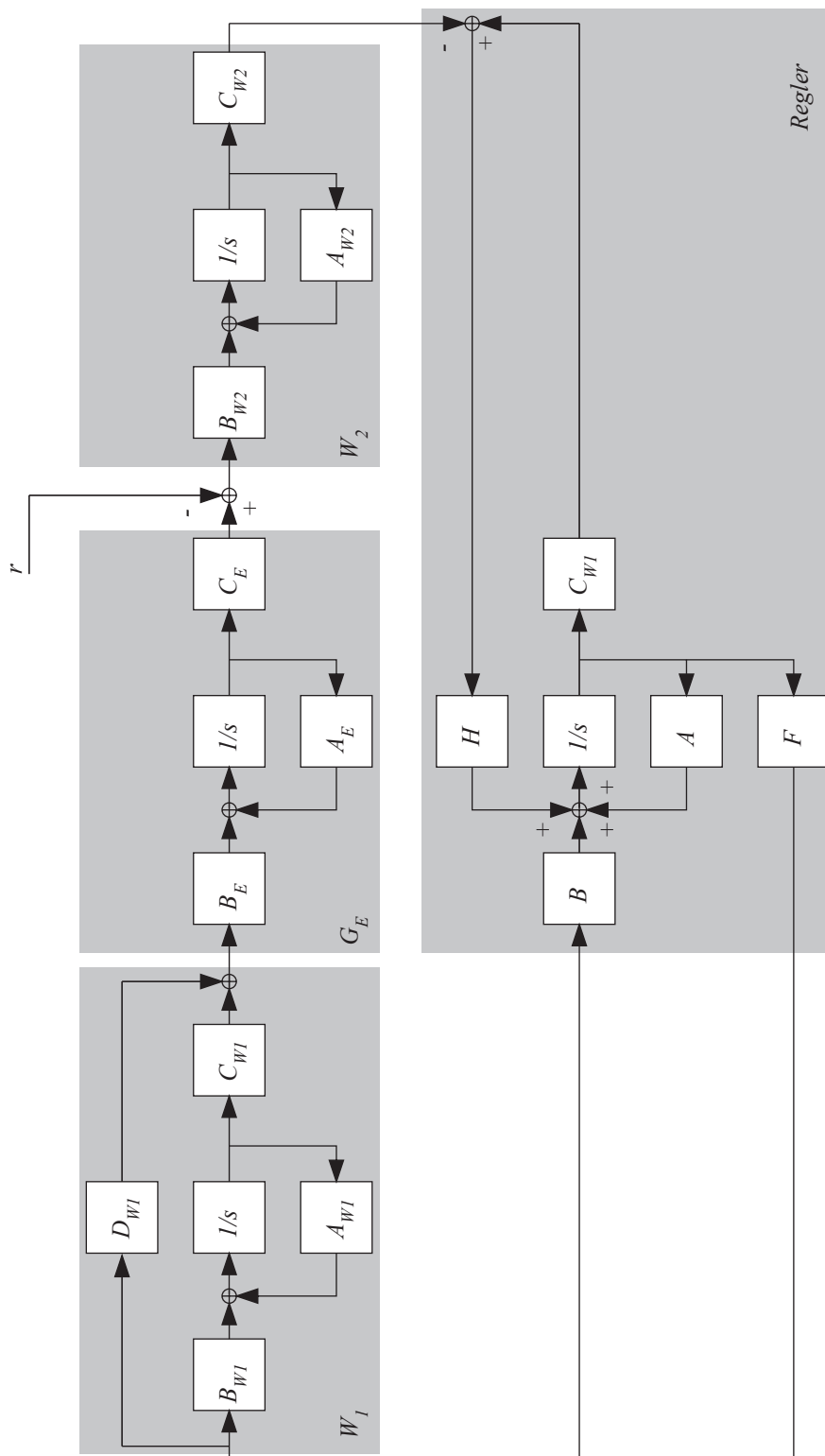


Abb. D.1: Blockdiagramm des linearen Entwurfsmodells, Regler im Vorwärtszweig

D.2 Lineares Entwurfsmodell, Regler in Beobachterdarstellung mit Zustandsvektorrückführung

D.2.1 Zustandsraumdarstellung

Systemmatrix und Eingangsmatrix:

$$\begin{bmatrix} \dot{x}_{W1} \\ \dot{x}_E \\ \dot{x}_{W2} \\ \dot{x}_O \end{bmatrix} = \begin{bmatrix} A_{W1} & 0 & 0 & 0B_{W1}F \\ B_EC_{W1} & A_E & 0 & B_ED_{W1}F \\ 0 & B_{W2}C_E & A_{W2} & 0 \\ 0 & 0 & -HC_{W2} & A + BF + HC \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} + \begin{bmatrix} -B_{W1}V \\ B_ED_{W1}V \\ 0 \\ -BV \end{bmatrix} \cdot r \quad (\text{D.3})$$

Ausgangsmatrix:

$$y = \begin{bmatrix} 0 & C_E & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} = y_E \quad (\text{D.4})$$

D.2.2 Blockdiagramm

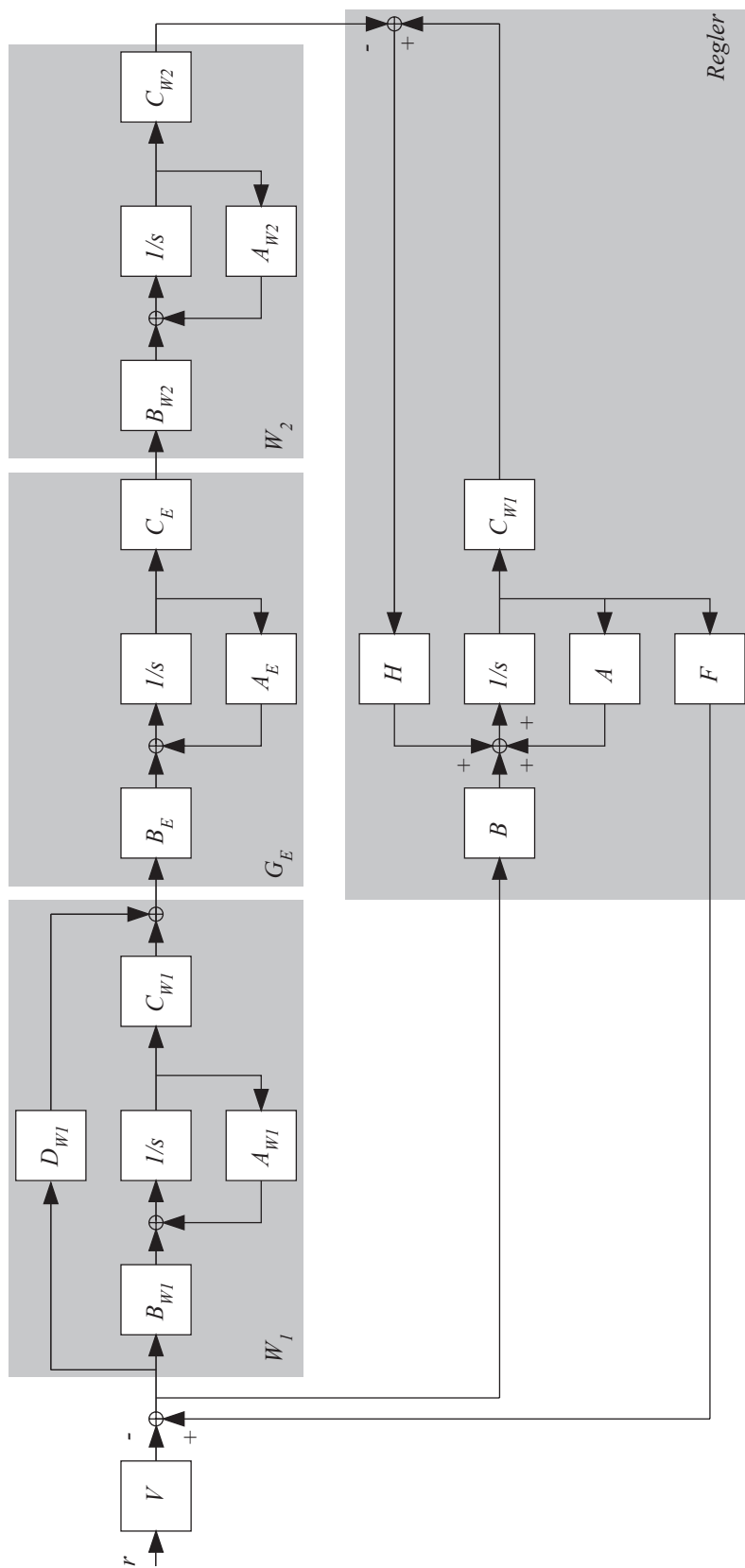


Abb. D.2: Blockdiagramm des linearen Entwurfsmodells, Regler in Beobachterdarstellung

D.3 Lineares Entwurfsmodell, Regler im Rückwärtszweig

D.3.1 Zustandsraumdarstellung

Systemmatrix und Eingangsmatrix:

$$\begin{bmatrix} \dot{x}_{W1} \\ \dot{x}_E \\ \dot{x}_{W2} \\ \dot{x}_O \end{bmatrix} = \begin{bmatrix} A_{W1} & 0 & 0 & 0B_{W1}F \\ B_E C_{W1} & A_E & 0 & B_E D_{W1}F \\ 0 & B_{W2}C_E & A_{W2} & 0 \\ 0 & 0 & -HC_{W2} & A + BF + HC \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} + \begin{bmatrix} -B_{W1}V \\ B_E D_{W1}V \\ 0 \\ 0 \end{bmatrix} \cdot r \quad (\text{D.5})$$

Ausgangsmatrix:

$$y = \begin{bmatrix} 0 & C_E & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_{W1} \\ x_E \\ x_{W2} \\ x_O \end{bmatrix} = y_E \quad (\text{D.6})$$

D.3.2 Blockdiagramm

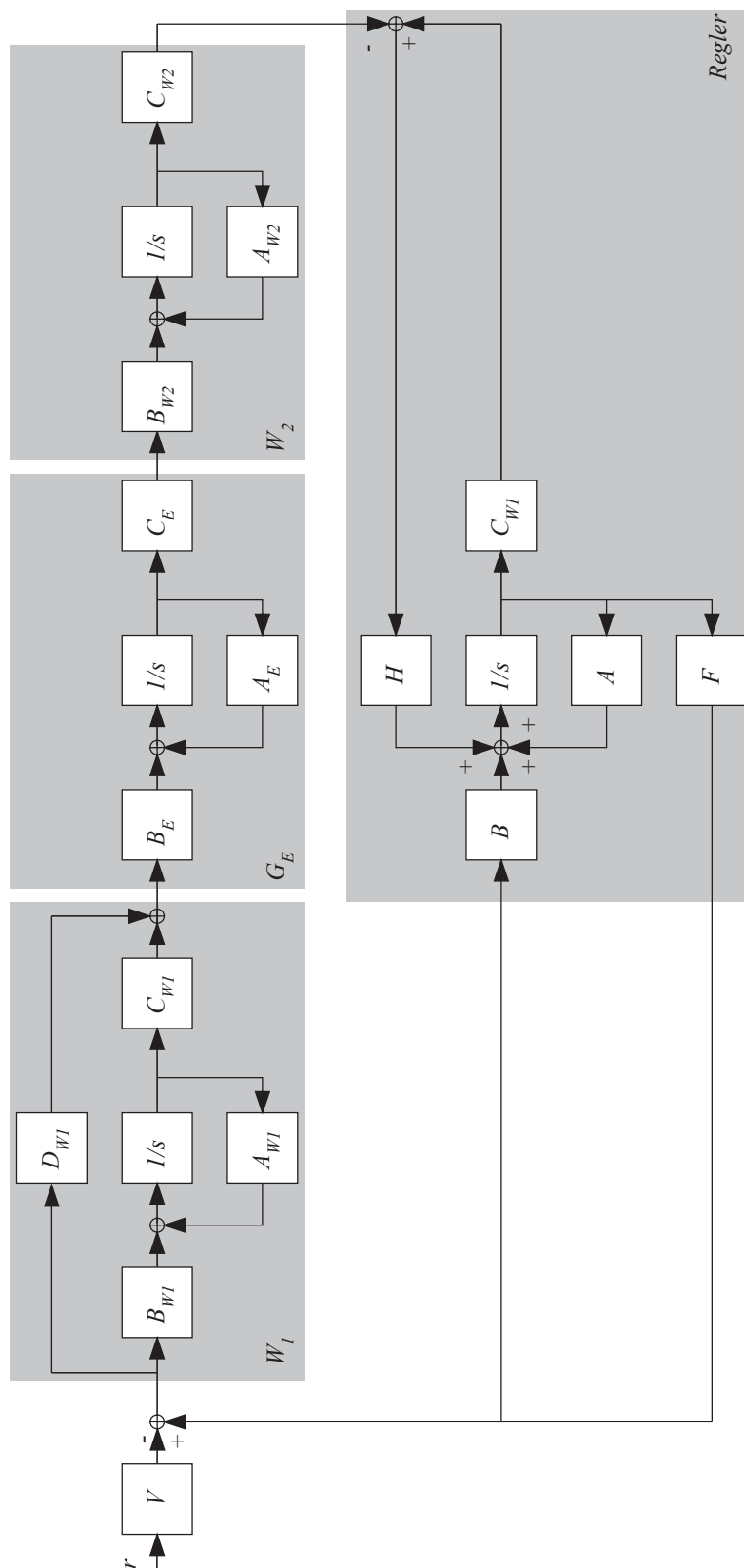


Abb. D.3: Blockdiagramm des linearen Entwurfsmodells, Regler im Rückwärtszweig

