
Eingebettete modellprädiktive Regelung der Abgastemperatur eines Dieselmotors

Von der Fakultät für Elektrotechnik, Informationstechnik und
Medientechnik der Bergischen Universität Wuppertal
genehmigte

Dissertation

zur Erlangung des akademischen Grades
eines Doktors der Ingenieurwissenschaften

von
Michael Thiel, M. Sc.
aus
Steinfurt

Tag der mündlichen Prüfung: 05. November 2021

Referent: Prof. Dr.-Ing. Bernd Tibken

Korreferent: Prof. Dr.-Ing. Dr. h. c. Oliver Sawodny

Die Dissertation kann wie folgt zitiert werden:

urn:nbn:de:hbz:468-20220505-115840-0

[<http://nbn-resolving.de/urn/resolver.pl?urn=urn%3Anbn%3Ade%3Ahbz%3A468-20220505-115840-0>]

DOI: 10.25926/mcra-ka55

[<https://doi.org/10.25926/mcra-ka55>]

Vorwort

Die vorliegende Dissertation entstand im Rahmen meiner Tätigkeit als Entwicklungsingenieur in der Softwareentwicklung bei der Firma *Deutz AG*, einem führenden Hersteller von industriell genutzten Dieselmotoren.

Wissenschaftlich wurde ich dabei von Herrn Prof. Dr.-Ing. Bernd Tibken vom Lehrstuhl für Elektrotechnik, Informationstechnik und Medientechnik an der Bergischen Universität Wuppertal betreut. Ihm danke ich herzlich für die Betreuung meiner Promotion, die Anregungen und Hinweise bei der Erstellung meiner Arbeit und für die Möglichkeit, die Promotion als externer Promovend durchzuführen.

Herrn Prof. Dr.-Ing. Dr. h. c. Oliver Sawodny, Leiter des Instituts für Systemdynamik der Universität Stuttgart, danke ich für das Interesse an meiner Arbeit und für seine freundliche Bereitschaft zur Übernahme des Korreferats.

Mein besonderer Dank gilt der Deutz AG für die Ermöglichung der Arbeit an diesem Projekt und die gewährten Freiheiten während dieser Zeit. Insbesondere danke ich Andreas Friesen, für die Unterstützung und die produktiven Diskussionen sowie für das in mich gesetzte Vertrauen. Darüber hinaus bedanke ich mich bei Matthias Illian und Thorsten Mülders für die Unterstützung bei der Entwicklung des Luftpfadmodells. Ein besonderer Dank gebührt meinen fleißigen Korrekturlesern Volker Smits, Markus Spiegel und Karsten Wimmers.

Meinen Eltern danke ich für ihr Verständnis und den Zuspruch während der anstrengenden Zeit. Zu guter Letzt danke ich meiner Frau Anna, für ihren Rückhalt, ihre Rücksichtnahme und ihre großartige Unterstützung im Rahmen der zahlreichen Korrekturrunden beim Verfassen dieser Dissertation.

Wuppertal, im August 2021

Michael Thiel

Kurzfassung

Moderne Dieselmotoren enthalten Abgasnachbehandlungssysteme, deren Wirksamkeit maßgeblich von der Abgastemperatur abhängt. Ziel dieser Arbeit ist es, ein modellprädiktives Regelungskonzept zu entwickeln, welches die Abgastemperatur parallel zu den eigentlichen Regelungszielen des Motors berücksichtigt. An das Regelungskonzept wird zudem die Anforderung gestellt, auf einem Serienmotorsteuergerät umsetzbar zu sein, sodass eine kommerzielle Nutzung möglich ist. Dementsprechend basieren die gewählten Ansätze maßgeblich auf der Bestrebung, eine rechentechnisch möglichst anspruchslose Umsetzung zu erarbeiten, die für andere auf dem Motorsteuergerät ausgeführte Programme genügend Kapazitäten lässt.

Die Regelung greift motorseitig in den sogenannten Luftpfad ein, wobei als Aktuatoren ein Wastegate und eine Abgasklappe verwendet werden. Die mathematische Modellierung dieses Luftpfades bildet einen Kernpunkt dieser Arbeit. Sie erfolgt in Form von gewöhnlichen, nichtlinearen Differentialgleichungen, die mit Hilfe eines CFD-Simulationsmodells entwickelt werden. Der modellierte Luftpfad wird mit einem im Rahmen dieser Arbeit entwickelten Codegenerator in auf dem Steuergerät lauffähigen C-Code umgesetzt.

Der zweite Kernpunkt liegt in der Entwicklung eines echtzeitfähigen, in einem eingebetteten System umsetzbaren modellprädiktiven Regelungsalgorithmus. Das verwendete Verfahren basiert auf dem Ansatz nach *Englert et al.* [EVM⁺19], welcher ein Gradientenverfahren zur Lösung des zugrunde liegenden dynamischen Optimierungsproblems mit der erweiterten Lagrange-Methode zur Handhabung der Nebenbedingungen kombiniert. Erweitert wird dieser Ansatz um eine Liniensuche zur Steigerung der Robustheit sowie um eine nicht-äquidistante Diskretisierung der prädizierten Trajektorien und einen separaten (kleineren) Kontrollhorizont zur Reduzierung der zu speichernden Variablen.

Die Erprobung des modellprädiktiven Reglers erfolgt sowohl anhand einer Simulation als auch am Motorenprüfstand. Die simulative Erprobung verwendet als Regelstrecke das Luftpfadmodell, um das theoretische Potential der Regelung aufzuzeigen. Es werden verschiedene Reglerkonfigurationen evaluiert und hinsichtlich der beanspruchten Rechenressourcen bewertet. Am Motorenprüfstand erfolgt die Erprobung mit dem Dieselmotor *Deutz TCD 12.0 V6*. Der Regelungsalgorithmus wird dabei parallel zur Seriensoftware auf dem Motorsteuergerät umgesetzt. Das Steuergerät steht der Regelung mithin nicht exklusiv zur Verfügung. Es wird gezeigt, dass der Regler die gestellte Regelungsaufgabe erfüllt und damit das Potential für eine kommerzielle Nutzung gegeben ist.

Abstract

Modern Diesel engines contain exhaust aftertreatment systems that depend to a large extent on the exhaust gas temperature regarding their effectiveness. The objective of this work is to develop a model predictive control concept that takes the exhaust gas temperature into account in parallel with the actual control objectives of the engine. The control concept is also required to be implementable on an engine control unit so that commercial use is possible. Accordingly, the selected approaches are based primarily on the effort to develop an implementation that is as computationally undemanding as possible and leaves sufficient capacity for other programs executed on the engine control unit.

The control system uses the so-called air path to control the engine, with a wastegate and an exhaust throttle valve as actuators. The mathematical description of this air path is one of the main topics of this thesis. It is performed in terms of ordinary nonlinear differential equations, which are developed using a CFD simulation model. The modeled air path is converted into C code executable on the control unit using a code generator developed as part of this work.

The second major topic of this work is the development of a real-time capable model predictive control algorithm that can be implemented in an embedded system. The method that is utilized is based on the approach according to *Englert et al.* [EVM⁺19], which combines a gradient method for solving the underlying dynamic optimization problem with the extended Lagrange method for handling the constraints. This approach is extended by including a line search to increase robustness as well as a non-equidistant discretization of the predicted trajectories and a separate (smaller) control horizon to reduce the variables to be stored.

The model predictive controller is tested both on the basis of a simulation and on the engine test bench. The simulative testing uses the air path model as the controlled system to demonstrate the theoretical potential of the controller. Various controller configurations are evaluated and assessed in terms of the required computing resources. On the engine test bench, testing is carried out with the Diesel engine *Deutz TCD 12.0 V6*. The control algorithm is implemented on the engine control unit in parallel with the series software. The ECU is therefore not exclusively available to the control system. It is shown that the controller fulfills the control task set and thus has the potential for commercial use.

Inhaltsverzeichnis

| | |
|--|--------------|
| Kurzfassung | v |
| Abstract | vii |
| Inhaltsverzeichnis | ix |
| Symbolverzeichnis | xiii |
| Abbildungsverzeichnis | xxi |
| Tabellenverzeichnis | xxiii |
| 1 Einleitung | 1 |
| 1.1 Konzept des emissionsgeführten Motors | 3 |
| 1.1.1 Funktionsprinzip des Dieselpartikelfilters | 4 |
| 1.1.2 Funktionsprinzip des SCR-Katalysators | 5 |
| 1.1.3 Emissionsführung | 7 |
| 1.2 Temperaturgeführter Motor | 9 |
| 1.2.1 Luftpfad | 9 |
| 1.2.2 Modellprädiktive Regelung | 11 |
| 1.2.3 Anforderungen an den Lösungsalgorithmus | 12 |
| 1.3 Gliederung | 15 |
| 2 Grundlagen | 17 |
| 2.1 Nichtlineare Systembeschreibung | 17 |
| 2.1.1 Zustandsraumdarstellung | 17 |
| 2.1.2 Numerische Integration nichtlinearer Differentialgleichungen | 19 |
| 2.1.3 Quadraturformeln | 21 |
| 2.2 Statische Optimierung | 27 |
| 2.2.1 Abstiegsverfahren | 29 |
| 2.2.2 Liniensuche | 33 |
| 2.2.3 Gradientenprojektion | 36 |
| 2.2.4 Erweiterte Lagrange-Methode | 37 |
| 2.3 Dynamische Optimierung | 46 |
| 2.3.1 Optimalitätsbedingungen | 49 |
| 2.3.2 Numerische Lösungsansätze | 55 |
| 3 Luftpfadmodell | 63 |
| 3.1 Modellbildung | 63 |
| 3.1.1 Modellierungsansatz | 64 |
| 3.1.2 Luftfilter | 69 |
| 3.1.3 Kompressor | 70 |
| 3.1.4 Ladeluftkühler | 73 |

| | | |
|----------|--|------------|
| 3.1.5 | Einlassflut | 75 |
| 3.1.6 | Motorblock | 76 |
| 3.1.7 | Auslassflut | 81 |
| 3.1.8 | Wastegate | 82 |
| 3.1.9 | Turbine | 83 |
| 3.1.10 | Abgasklappe | 87 |
| 3.1.11 | Turboladerdrehzahl | 89 |
| 3.2 | Parameteridentifikation und Validierung | 89 |
| 3.2.1 | Datenbasis zur Identifikation | 90 |
| 3.2.2 | Identifikation der Teilmodelle | 90 |
| 3.2.3 | Globale Optimierung | 94 |
| 3.2.4 | Stützung des Luftpfadmodells mit Sensorinformationen | 97 |
| 3.3 | Rechentechische Umsetzung | 99 |
| 3.3.1 | Beschreibung der C-Code-Generierung | 100 |
| 3.3.2 | Eigenschaften des generierten C-Codes | 102 |
| 3.3.3 | Bewertung der Echtzeitfähigkeit | 104 |
| 4 | Modellprädiktiver Regelungsalgorithmus | 107 |
| 4.1 | Modellprädiktive Regelung | 107 |
| 4.2 | Motivation des Gradientenverfahrens | 110 |
| 4.3 | Lösungsansatz | 112 |
| 4.3.1 | Erweiterte Lagrange-Formulierung | 114 |
| 4.3.2 | Diskretisierung und Interpolation | 116 |
| 4.4 | Erweitertes Gradientenverfahren | 117 |
| 4.4.1 | Gradientenverfahren | 119 |
| 4.4.2 | Lösung der kanonischen Differentialgleichungen | 121 |
| 4.4.3 | Liniensuche | 124 |
| 4.4.4 | Gradientenprojektion mit Änderungsratenbeschränkung | 127 |
| 4.4.5 | Aktualisierung der Lagrange-Multiplikatoren sowie der Strafparameter | 129 |
| 4.4.6 | Initialisierung | 131 |
| 4.5 | Bewertung und Möglichkeiten zur Weiterentwicklung | 134 |
| 5 | Temperaturgeführte Motorregelung | 137 |
| 5.1 | Regelungskonzept | 137 |
| 5.1.1 | Motorregelung | 138 |
| 5.1.2 | Beeinflussung der Abgastemperatur | 141 |
| 5.1.3 | Einflüsse von Wastegate und Abgasklappe | 144 |
| 5.1.4 | Regelungstechnische Eigenschaften des Luftpfades | 146 |
| 5.2 | Formulierung des Optimierungsproblems | 150 |
| 5.2.1 | Regelungstechnische Anforderungen | 151 |
| 5.2.2 | Endlicher Horizont ohne Endbedingung | 152 |
| 5.2.3 | Skalierung | 154 |
| 5.2.4 | Integrales Gütefunktional | 155 |
| 5.2.5 | Nebenbedingungen | 156 |

| | | |
|----------|---|------------|
| 5.3 | Konfiguration des Lösungsalgorithmus | 159 |
| 5.3.1 | Grundeinstellungen | 160 |
| 5.3.2 | Konfigurationen | 161 |
| 5.3.3 | Vergleich der Konfigurationen | 166 |
| 5.4 | Validierung | 167 |
| 5.4.1 | Vorsteuerung der Abgasklappe | 168 |
| 5.4.2 | Detailbetrachtung | 169 |
| 5.4.3 | Simulation des NRTC-Testzyklus | 172 |
| 5.4.4 | Prüfstandsmessung des NRTC-Testzyklus | 175 |
| 5.5 | Bewertung und Möglichkeiten zur Weiterentwicklung | 178 |
| 5.5.1 | Bewertung der Regelung | 178 |
| 5.5.2 | Separate Auflösung des Zielkonfliktes | 181 |
| 5.5.3 | Potential der Einbeziehung des Einspritzwinkels | 182 |
| 6 | Zusammenfassung und Ausblick | 183 |
| A | Mathematischer Anhang | 187 |
| A.1 | Lösen nichtlinearer Gleichungssysteme | 187 |
| A.1.1 | Fixpunktiteration | 187 |
| A.1.2 | Newton-Verfahren | 188 |
| A.2 | Fehler und Normen | 189 |
| B | Abbildungen zu Kapitel 5 | 191 |
| C | Emissionsgesetzgebung | 205 |
| D | Deutz TCD 12.0 V6 | 209 |
| | Literaturverzeichnis | 211 |

Symbolverzeichnis

Abkürzungen

| | |
|---------------|---|
| AGN | Abgasnachbehandlung |
| AGR | Abgasrückführung |
| ASC | Ammoniak-Schlupf-Katalysator (engl.: <i>Ammonia Slip Catalyst</i>) |
| AWP | Anfangswertproblem |
| BSR | Drehzahlverhältnis der Turbinenschaufeln (engl.: <i>Blade Speed Ratio</i>) |
| CAD | Rechnergestütztes Konstruieren (engl.: <i>Computer-Aided Design</i>) |
| CFD | Numerische Strömungsmechanik (engl.: <i>Computational Fluid Dynamics</i>) |
| CPU | Zentrale Recheneinheit (engl.: <i>Central Processing Unit</i>) |
| CSF | Katalytischer Rußfilter (engl.: <i>Catalytic Soot Filter</i>) |
| DGL | Differentialgleichung |
| DK | Direktes Kollokationsverfahren |
| DMS | Direktes Mehrfachschießverfahren |
| DOC | Diesel-Oxydationskatalysator (engl.: <i>Diesel Oxidation Catalyst</i>) |
| DoE | Statistische Versuchsplanung (engl.: <i>Design of Experiments</i>). Nennungen im Text beziehen sich auf Tab. 3.1 |
| DP | Dynamisches Optimierungsproblem |
| DPF | Dieselpartikelfilter |
| DS | Direktes Schießverfahren |
| ECU | Motorsteuergerät (engl.: <i>Electronic Control Unit</i>) |
| ETV | Abgasklappe (engl.: <i>Exhaust Throttle Valve</i>) |
| FLOPS | Anzahl der Gleitkommaoperationen pro Sekunde (engl.: <i>Floating Point Operations Per Second</i>) |
| GM | Gesamtmodell |
| GMGO | Global optimiertes Gesamtmodell |
| GV | Gradientenverfahren zur Lösung dynamischer Optimierungsprobleme |
| HJB | Hamilton-Jacobi-Bellman-Gleichung |
| HM | Hesse-Matrix |
| Impl.-Aufwand | Implementierungsaufwand |
| IMS | Indirektes Mehrfachschießverfahren |
| IRK1 | Implizites Euler-Verfahren |
| IRK2 | Implizites Trapezverfahren |
| IS | Indirektes Schießverfahren |
| KNN | Künstliches neuronales Netz |
| MIMO | System mit mehreren Ein- und Ausgängen (engl.: <i>Multiple Input Multiple Output</i>) |
| MPC | Modellprädiktive Regelung (engl.: <i>Model Predictive Control</i>) |

| | |
|---------------|--|
| NRMSE | Normierte Wurzel der mittleren Fehlerquadratsumme (engl.: <i>Normalized Root Mean Square Error</i>) |
| NRTC | Non-Road Transient Cycle |
| PM | Schadstoffpartikel (engl.: <i>Particulate Matter</i>) |
| PS | Prüfstandsmessung |
| QP | Quadratisches Optimierungsproblem |
| RAM | Direktzugriffsspeicher (engl.: <i>Random Access Memory</i>) |
| RK | Klasse der Runge-Kutta-Verfahren |
| RK1 | Explizites Euler-Verfahren |
| RK4 | Klassisches Runge-Kutta-Verfahren |
| RMC | Ramped Mode Cycle |
| RMSE | Wurzel der mittleren Fehlerquadratsumme (engl.: <i>Root Mean Square Error</i>) |
| RRMSE | Wurzel der mittleren Relativfehlerquadratsumme (engl.: <i>Relative Root Mean Square Error</i>) |
| RTF | Echtzeitfaktor (engl.: <i>Real-Time Factor</i>) |
| RWP | Randwertproblem |
| SCR | SCR-Katalysator (engl.: <i>Selective Catalytic Reduction-Catalyst</i>) |
| SISO | System mit einem Eingang und einem Ausgang (engl.: <i>Single Input Single Output</i>) |
| SQP | Sequentiell quadratische Programmierung |
| TM | Teilmodell |
| u. B. v. | Unter Berücksichtigung von |
| VEG | Vollständig erweiterte Gütefunktion |
| Vorz. Abbruch | Vorzeitiger Abbruch |
| WG | Wastegate |
| adj. | adjungiert |
| dünnbes. | dünnbesetzt |
| exp | Exponentialfunktion |
| log | Logarithmusfunktion |
| lut2d | Zweidimensionale Lookup-Tabelle |
| sqrt | Wurzelfunktion |

Notationsvereinbarungen

| | |
|--------------|--|
| x | Skalar |
| \mathbf{x} | Vektor |
| \mathbf{X} | Matrix |
| \hat{x} | Schätzwert |
| x^* | Optimaler Wert |
| x^* | Für die Reglerauslegung relevanter Koeffizient |
| $x^{(i)}$ | Iterationsindex |

| | |
|-----------|---|
| $x_{k i}$ | Diskretisierung i auf dem Prädiktionshorizont ausgehend von dem Zeitpunkt t_k |
| $ x $ | Betrag; Mächtigkeit einer Menge |
| Δ | Differenz zwischen zwei Größen gleichen Typs |
| ∇ | Nabla-Operator; Gradient |

Physikalische Größen

| | | |
|---------------|---|--------------------|
| A | Fläche | $[m^2]$ |
| J | Trägheitsmoment | $[kg\ m^2]$ |
| M | Drehmoment | $[Nm]$ |
| M_e | Effektives Drehmoment | $[Nm]$ |
| M_l | Lastmoment | $[Nm]$ |
| Ma | Mach-Zahl | $[-]$ |
| P | Leistung | $[W]$ |
| \dot{Q} | Wärmeleistung | $[\frac{J}{s}]$ |
| T | Temperatur | $[K]$ |
| U_c | Geschwindigkeit der Kompressorblattspitze | $[\frac{m}{s}]$ |
| V | Volumen | $[m^3]$ |
| W | Massenstrom | $[\frac{kg}{s}]$ |
| X_{O_2} | Sauerstoffmassenanteil | $[-]$ |
| c_p | Isobare spezifische Wärmekapazität | $[\frac{J}{kgK}]$ |
| m | Masse | $[kg]$ |
| n | Drehzahl | $[\frac{1}{s}]$ |
| p | Druck | $[Pa]$ |
| r | Aktuatorsollwert | $[\%]$ |
| r | Radius | $[m]$ |
| t | Zeit | $[s]$ |
| Δp | Druckdifferenz | $[Pa]$ |
| Γ | Dimensionslose Hilfsvariable | $[-]$ |
| Π | Druckverhältnis | $[-]$ |
| Φ | Normalisierter Kompressordurchsatz | $[-]$ |
| Φ_s | Luftzahl | $[-]$ |
| Ψ | Durchflussfunktion | $[-]$ |
| ε | Wirkungsgrad | $[-]$ |
| η | Effizienz; Isentropische Effizienz | $[-]$ |
| η_V | Volumetrische Effizienz | $[-]$ |
| ρ | Dichte | $[\frac{kg}{m^3}]$ |
| τ | Zeitkonstante | $[s]$ |
| ζ | Aktuatorposition | $[\%]$ |
| ω | Winkelgeschwindigkeit | $[\frac{rad}{s}]$ |
| ω | Kreisfrequenz | $[\frac{1}{s}]$ |

Physikalische Konstanten

| | | |
|------------------|------------------------------|---------------------------------------|
| N_{cyl} | Zylinderanzahl | 6 |
| R_a | Gaskonstante von Luft | $287,058 \frac{\text{J}}{\text{kgK}}$ |
| γ | Isentropenexponent | 1,4 |
| Φ_s | Stöchiometrischer Luftbedarf | 14,5 |

Mathematische Größen

| | |
|---------------------------|---|
| A | Systemmatrix |
| B | Eingangsmatrix |
| C | Ausgangsmatrix |
| D | Durchgriffsmatrix |
| \mathcal{E} | Menge der Indizes der Gleichungsbeschränkungen |
| G | Anzahl der Gradienteniterationen |
| $H(\cdot)$ | Hamilton-Funktion |
| I | Einheitsmatrix |
| \mathcal{I} | Menge der Indizes der Ungleichungsbeschränkungen |
| $J(\cdot)$ | Gütefunktion eines dynamischen Optimierungsproblems |
| K | Lineare Zustandsrückführung |
| L | Anzahl der Liniensuche-Iterationen |
| \mathcal{L} | Lagrange-Funktion |
| \mathcal{L}_A | Erweiterte Lagrange-Funktion |
| M | Anzahl der Stellgrößendiskretisierungspunkte |
| N | Anzahl der Diskretisierungspunkte auf dem Prädiktionshorizont |
| $\mathbf{P}(\cdot)$ | (Gradienten-)Projektionsfunktion |
| $\bar{\mathbf{P}}(\cdot)$ | (Gradienten-)Projektionsfunktion mit Änderungsratenbeschränkung |
| $Q(\cdot)$ | Straffunktion |
| Q_p | p-Quantil |
| \mathcal{R} | Menge der Ruhelagen |
| T | Prädiktionshorizont |
| T_C | Kontrollhorizont |
| T_{int} | Integrationsschrittweite |
| T_s | Abtastzeit; Abtastzeit des Reglers |
| $V(\cdot)$ | Endkostengewichtung |
| $\text{diag}(\cdot)$ | Diagonalmatrix aus einem Vektor |
| a | Anstiegsrichtung |
| b_l | Untere Grenze (engl.: <i>lower bound</i>) |
| b_u | Obere Grenze (engl.: <i>upper bound</i>) |
| $\mathbf{c}(\cdot)$ | Nebenbedingung |
| c_1 | Steilheitsfaktor der Armijo-Bedingung |
| c_2 | Steilheitsfaktor der Goldstein-Bedingung |

| | |
|-------------------------------|--|
| \mathbf{d} | Abstiegsrichtung |
| $\bar{\mathbf{d}}$ | Vektor der Abstiegsrichtungsdiskretisierungen |
| $\exp(\cdot)$ | Exponentialfunktion |
| $f(\cdot)$ | Gütefunktion, Kostenfunktion |
| $\mathbf{f}(\cdot)$ | Nichtlineare Differentialgleichung |
| $g(\cdot)$ | Stabilitätsfunktion; Duale Funktion |
| \mathbf{g} | Gradient |
| $\bar{\mathbf{g}}$ | Vektor der Gradientendiskretisierungen |
| $\mathbf{g}(\cdot)$ | Nichtlinearer Ausgangsgleichung |
| h | Integrations-schrittweite |
| i | Iterationsvariable |
| j | Iterationsvariable |
| k | (Polynom-)Koeffizient |
| $\max(\cdot)$ | Maximumsfunktion |
| $\min(\cdot)$ | Minimumsfunktion |
| n | Anzahl der Zustände |
| l | Iterationsvariable der Liniensuche; Anzahl der integralen Nebenbedingungen |
| l_T | Anzahl der terminalen Nebenbedingungen |
| $l(\cdot)$ | Integralkosten |
| $\log(\cdot)$ | Natürlicher Logarithmus |
| m | Anzahl der Eingänge; Iterationsvariable des Gradientenverfahrens |
| p | Anzahl der Ausgänge |
| $\mathbf{q}(\cdot)$ | Nichtlineares Regelungsgesetz |
| \mathbf{r} | Änderungsrate |
| \mathbf{r}_l | Untere Grenze der Ratenlimitierung (engl.: <i>lower rate limit</i>) |
| \mathbf{r}_u | Obere Grenze der Ratenlimitierung (engl.: <i>upper rate limit</i>) |
| \mathbf{s} | Schlupfvariablen |
| t | Laufzeit |
| \mathbf{u} | Eingangsvektor |
| $\bar{\mathbf{u}}$ | Vektor der Eingangsgrößendiskretisierungen |
| $\mathbf{u}(\cdot)$ | Steuertrajektorie |
| \mathbf{w} | Vektor aus den Zuständen sowie der adjungierten Zuständen |
| \mathbf{x} | Zustandsvektor; Optimierungsvariablen |
| $\bar{\mathbf{x}}$ | Vektor der Zustandsgrößendiskretisierungen |
| \mathbf{y} | Ausgangsvektor |
| α | Schrittweite der Liniensuche |
| β | Schrittweite der Anstiegsiteration der Lagrange-Multiplikatoren |
| γ | Faktor der Strafparameteranpassung |
| ε | Toleranz |
| λ | Eigenwert |
| $\boldsymbol{\lambda}$ | Adjungierte Zustände |
| $\bar{\boldsymbol{\lambda}}$ | Vektor der Diskretisierungen der adjungierten Zustände |
| $\boldsymbol{\lambda}(\cdot)$ | Trajektorie der adjungierten Zustände |
| $\boldsymbol{\mu}$ | Lagrange-Multiplikatoren |
| $\bar{\boldsymbol{\mu}}$ | Vektor der Lagrange-Multiplikator-Diskretisierungen |

| | |
|-----------------|--|
| $\mu(\cdot)$ | Lagrange-Multiplikator-Trajektorie |
| ν | Strafparameter |
| $\bar{\nu}$ | Vektor der Strafparameterdiskretisierungen |
| $\nu(\cdot)$ | Strafparametertrajektorie |
| τ | MPC-Zeit |
| $\bar{\tau}$ | Diskretisierungszeitpunkte auf dem Prädiktionshorizont |
| $\psi(\cdot)$ | Lineare Interpolationsfunktion |
| $\psi_s(\cdot)$ | Stufenweise lineare Interpolationsfunktion |

Indizes

| | |
|---------------|---|
| B | Bandbreite |
| \mathcal{E} | Gleichungsbeschränkung |
| \mathcal{I} | Ungleichungsbeschränkung |
| R | Ruhelage |
| T | Terminale Größe |
| a | Umgebung / Umwelt |
| af | Luftfilter |
| ac | Ladeluftkühler |
| c | Kompressor |
| cor | Korrigierter Parameter |
| cyl | Zylinder |
| dec | Dekrement |
| eff | Effektive Größe |
| ei | Einlassseite des Motorblocks |
| eng | Dem Motorblock zugeordnete Quantität |
| eo | Auslassseite des Motorblocks |
| f | Diesekraftstoff |
| fac | Faktor |
| ff | Vorsteuerung (engl.: <i>feed forward</i>) |
| inc | Inkrement |
| ini | Initialwert |
| inj | Einspritzung |
| int | Integration |
| k | Zeitschritt, Zeitindex |
| l | Untere Grenze (engl.: <i>lower bound</i>) |
| m | Mechanischer Wert bspw. mechanischer Wirkungsgrad |
| max | Maximalwert |
| mess | Messwert |
| min | Minimalwert |
| ofs | Versatz (engl.: <i>offset</i>) |
| rail | Kraftstoffversorgung der Einspritzung |
| red | Reduzierter Parameter |

| | |
|-------|--|
| ref | Referenzwert |
| s | Sollwert; isentroper Wert; Abtastung |
| start | Startwert einer Iteration |
| t | Turbine |
| tc | Turbolader |
| tm | Kombination aus einer Turbinen- und einer mechanischen Eigenschaft |
| u | Obere Grenze (engl.: <i>upper bound</i>) |
| wg | Wastegate |
| 0 | Position vor dem Luftfilter; Wert zum Zeitpunkt $t = 0$ bzw. t_0 |
| 1 | Position zwischen Luftfilter und Kompressor |
| 20 | Position zwischen Kompressor und Ladeluftkühler |
| 21 | Position zwischen Ladeluftkühler und dem Ersatzvolumen der Einlassflut(en) |
| 22 | Ersatzvolumen der Einlassflut(en) |
| 3 | Ersatzvolumen der Auslassflut(en) |
| 40 | Position vor der Abgasklappe |
| 41 | Position unmittelbar hinter der Abgasklappe |

Abbildungsverzeichnis

| | | |
|-----|--|-----|
| 1.1 | Entwicklung der gesetzlichen Schadstoffgrenzen. | 2 |
| 1.2 | Anordnung des Motors und der Abgasnachbehandlung. | 4 |
| 1.3 | Funktionsweise des Dieselpartikelfilters. | 5 |
| 1.4 | Funktionsweise des SCR-Katalysators. | 6 |
| 1.5 | Emissionsgeführter Motor. | 9 |
| 1.6 | Systemdarstellung des Deutz TCD 12.0 V6 Dieselmotors. | 10 |
| 1.7 | Symbolbild des im Deutz TCD 12.0 V6 verbauten Motorsteuergeräts. . . . | 13 |
| | | |
| 2.1 | Qualitative Darstellung der Quadratur mit dem expliziten Euler-Verfahren für den eindimensionalen Fall. | 23 |
| 2.2 | Stabilitätsgebiete der betrachteten Integrationsverfahren. | 24 |
| 2.3 | Qualitative Darstellung des Abstiegsverfahrens. | 29 |
| 2.4 | Extremstellen im Rahmen einer Minimierung. | 30 |
| 2.5 | Armijo-Goldstein-Bedingungen. | 34 |
| 2.6 | Veranschaulichung der Gradientenprojektion. | 38 |
| | | |
| 3.1 | Modellvorstellung des <i>Deutz TCD 12.0 V6</i> Dieselmotors. | 65 |
| 3.2 | Modell des WG-Massenstroms sowie der modellierten effektiven Querschnittsfläche. | 83 |
| 3.3 | Modell des Drucks vor der Abgasklappe sowie die modellierte Drosselkonstante. | 88 |
| 3.4 | Simulation des Teilmodells für die Abgastemperatur. | 94 |
| 3.5 | Güte des lokal und des global optimierten Luftpfadmodells im Vergleich zur CFD-Simulation. | 96 |
| 3.6 | Güte des unkorrigierten und des korrigierten Luftpfadmodells im Vergleich mit einer Prüfstandsmessung. | 99 |
| | | |
| 4.1 | Prinzip des bewegten Horizonts. | 108 |
| 4.2 | MPC-Reglerstruktur mit Zustandsrückführung. | 109 |
| | | |
| 5.1 | Drehzahlregelung. | 138 |
| 5.2 | Übersicht der Motorregelung. | 139 |
| 5.3 | Einfluss des Betriebspunktes auf die Abgastemperatur sowie die Einspritzmenge bei konstanter effektiver Leistung. | 142 |
| 5.4 | Einfluss der Ladeluftmenge auf die Abgastemperatur, die Einspritzmenge sowie den Ladedruck bei konstantem Drehmoment. | 143 |
| 5.5 | Einfluss des Differenzdrucks zwischen Abgasgegen- und Ladedruck auf die Abgastemperatur bei konstantem Drehmoment. | 144 |
| 5.6 | Einfluss der Aktuatorpositionen auf die Drücke. | 145 |
| 5.7 | Einfluss der Aktuatorpositionen auf die Abgastemperatur sowie der Zusammenhang zwischen Abgastemperatur und Ladedruck. | 146 |
| 5.8 | Bode-Diagramm der Übertragungsfunktionsmatrix. | 148 |
| 5.9 | Veranschaulichung der abgastemperaturabhängigen ETV-Begrenzung. . . . | 157 |

| | | |
|------|--|-----|
| 5.10 | Verhalten eines PT1-Filters bei ratenlimitierten Sprüngen. | 159 |
| 5.11 | Fehler der Regelgrößen sowie Rechenzeit und Speicherverbrauch der MPC-Konfigurationen. | 166 |
| 5.12 | Abgasklappen-Vorsteuerung. | 169 |
| 5.13 | Verhalten des finalen Reglers unter Verwendung der Abgasklappen-Vorsteuerung. | 171 |
| 5.14 | Simulative Ladedruckregelung ohne gleichzeitige Regelung der Mindestabgastemperatur. | 173 |
| 5.15 | Simulative Regelung der Mindestabgastemperatur und des Ladedrucks im Vergleich mit einer reinen Ladedruckregelung. | 176 |
| 5.16 | Verhalten der MPC am Motorenprüfstand im Vergleich mit der Seriensoftware. | 177 |
| 5.17 | Prüfstandszeichnung der Regelung auf eine Mindestabgastemperatur und auf einen Ladedruck. | 179 |
| A.1 | Fixpunktiteration. | 187 |
| B.1 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration A | 192 |
| B.2 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration B | 193 |
| B.3 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration C | 194 |
| B.4 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration D | 195 |
| B.5 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration E | 196 |
| B.6 | Verhalten des MPC-Lösungsalgorithmus unter Verwendung von Konfiguration F | 197 |
| B.7 | Verhalten des finalen Reglers ohne Abgasklappen-Vorsteuerung. | 198 |
| B.8 | Reglerverhalten im simulierten NRTC ohne Regelung der Mindestabgastemperatur. | 199 |
| B.9 | Reglerverhalten im simulierten NRTC mit Regelung der Mindestabgastemperatur. | 200 |
| B.10 | Veranschaulichung einer bleibenden Regelabweichung bei der Regelung des Ladedrucks am Motorenprüfstand. | 201 |
| B.11 | Prüfstandszeichnung der Regelung der Mindestabgastemperatur sowie des Ladedrucks. | 202 |
| B.12 | Simulative Regelung der Mindestabgastemperatur sowie des Ladedrucks. | 203 |
| C.1 | Normierte Drehzahlen und Drehmomente im NRTC-Testzyklus. | 205 |
| D.1 | CAD-Zeichnung des Deutz TCD 12.0 V6 Dieselmotors. | 209 |

Tabellenverzeichnis

| | | |
|-----|--|-----|
| 1.1 | Eckdaten des in dem MD1-CE101 verbauten Mikrocontrollers. | 14 |
| 1.2 | Anforderungen an die programmiertechnische Umsetzung des Lösungsalgorithmus der MPC. | 15 |
| 3.1 | Statistische Beschreibung des DoE aus [TT21]. | 92 |
| 3.2 | Fehler der autark berechneten Teilmodelle und des Gesamtmodells. | 93 |
| 3.3 | Modellgenauigkeit unter Verwendung von Sensoren zur Modellstützung. | 98 |
| 3.4 | Ausführungszeit und Größe des Funktionsstacks. | 102 |
| 3.5 | Anzahl der durchgeführten Operationen und Funktionsaufrufe im generierten C-Code. | 103 |
| 4.1 | Verfahren der dynamischen Optimierung im qualitativen Vergleich. | 111 |
| 4.2 | Variablenanzahl und die Anzahl der Elemente in der Hesse-Matrix. | 112 |
| 4.3 | Parameter des MPC-Lösungsalgorithmus. | 133 |
| 5.1 | Verwendete Skalierungen. | 155 |
| 5.2 | Grundeinstellungen des MPC-Optimierungsproblems. | 156 |
| 5.3 | Nicht-variierte Einstellungen des MPC-Lösungsalgorithmus. | 161 |
| 5.4 | MPC-Konfigurationen. | 161 |
| 5.5 | Fixierte Modelleingänge des Szenarios zur Reglerevaluierung. | 162 |
| 5.6 | Fehler der Regelgrößen sowie Rechenzeit und Speicherverbrauch der MPC-Konfigurationen. | 167 |
| C.1 | Emissionsgrenzwerte der verschiedenen Emissionsstufen in der Europäischen Union. | 207 |
| D.1 | Technische Daten des Deutz TCD 12.0 V6 für landwirtschaftliche Anwendungen. | 209 |

1 Einleitung

Die Entwicklung von Dieselmotoren war bis in die 90er Jahre hinein durch Verbesserungen der Leistungsdichte sowie von Effizienzsteigerungen geprägt. Um Mensch und Umwelt vor den schädlichen Abgasen zu schützen, wurde im Jahr 1992 die *Euro 1*-Norm als erste Emissionsgesetzgebung in der Europäischen Gemeinschaft festgelegt.¹ Seitdem sind die Emissionsgesetzgebungen ein wesentlicher Entwicklungstreiber in der Motorenentwicklung. Mit der *Richtlinie 97/68/EG* vom 16.12.1997 (ABl. L 59) wurde die ab 1999 gültige (Emissions-) *Stufe I* für *mobile Maschinen und Geräte* – im Weiteren als *Non-Road-Anwendung* bezeichnet – eingeführt. Typische Non-Road-Anwendungen sind Baumaschinen, Kräne, landwirtschaftliche Maschinen oder die Bodengeräte eines Flughafens. Durch diese Richtlinie wurden die zulässigen Emissionen von Kohlenmonoxid (CO), Kohlenwasserstoffen (HC), Stickoxiden ($\text{NO}_x := \text{NO} + \text{NO}_2$) und (Schadstoff-)Partikeln (engl.: *Particulate Matter*, kurz: PM) begrenzt. Von den genannten Schadstoffen stellt die Beseitigung der Stickoxid- und Partikelemissionen die treibende technische Herausforderung dar. Die qualitative Entwicklung der Gesetzgebung bezüglich dieser beiden Schadstoffe ist in Abbildung 1.1 dargestellt. Die konkreten Zahlenwerte der betrachteten Leistungsklasse der Motoren mit einer Leistung von 130 kW bis 560 kW können Tab. C.1 entnommen werden. Bis zur (Emissions-) *Stufe IIIA* war es möglich, diese Grenzen durch innermotorische Maßnahmen, d. h. durch eine Anpassung der Dieseleinspritzung sowie durch eine *Abgasrückführung* (AGR)², zu erreichen. Seit der Einführung der *Stufe IIIB* ist eine zusätzliche *Abgasnachbehandlung* (AGN) notwendig. Es sei angemerkt, dass sich die Notwendigkeit einer AGN rein faktisch aus der Festlegung niedrigerer Grenzwerte ergeben hat. Der Gesetzgeber macht dazu, auf welche Weise die Einhaltung dieser Werte sicherzustellen ist, keine expliziten Vorgaben. Die jüngste und aktuell gültige *Stufe V* definiert eine Partikelobergrenze, die lediglich noch ca. 2,8 % der erlaubten Emissionen von *Stufe I* entspricht. Die Stickoxidobergrenze hat sich auf ca. 4,3 % der erlaubten Grenze von 1999 reduziert. Die Schadstoffemissionen werden als Integralwerte über genormte Prüfzyklen ermittelt. Genauere Informationen hierzu sind in Anhang C zu finden.

Die Leistungsfähigkeit der AGN ist abhängig vom aktuellen Betriebspunkt des Motors, insbesondere von der sich daraus ergebenden Abgastemperatur, sowie dem Volumenstrom des Abgases.³ Letzterer ist ein Indikator für die Verweilzeit des Abgases in der jeweiligen AGN-Komponente.⁴ Aus der momentanen Leistungsfähigkeit der AGN ergeben sich motorseitig einzuhaltende Grenzwerte für die in dem jeweiligen Zeitpunkt maximal zu

¹Die *Euro 1*-Norm wurde in der *Richtlinie 91/441/EWG* vom 26.06.1991 (ABl. L 242) eingeführt.

²Die AGR führt einen Teil des Abgases in den Brennraum zurück. Dadurch sinkt die Verbrennungstemperatur, wodurch eine Reduzierung der Stickoxidrohmissionen erwirkt wird. Im Gegenzug erhöhen sich die Partikelemissionen. [PKS09, S. 241f]

³In Grenzen ebenfalls beeinflussbar ist die Abgaschemie, beispielsweise das NO_2/NO_x -Verhältnis. Dieser Aspekt wird in Abschnitt 1.1 näher ausgeführt.

⁴Üblicherweise wird die sogenannte *Raumgeschwindigkeit* betrachtet, welche über den auf das Volumen der AGN-Komponente normierten Volumenstrom definiert ist. Sie gibt an, mit welcher Rate das Gas in der Komponente ausgetauscht wird.

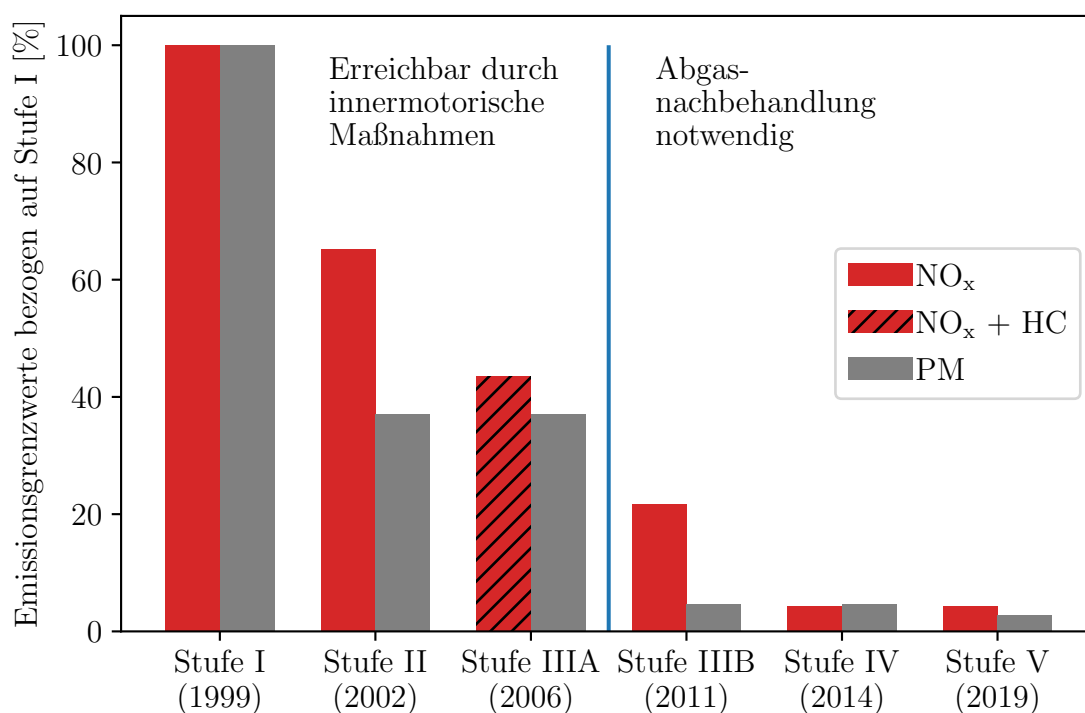


Abb. 1.1: Entwicklung der gesetzlichen Schadstoffgrenzen. Die Abbildung basiert auf den Daten aus Tab. C.1.

emittierenden Rohemissionen.⁵ Diese dürfen nicht überschritten werden, um ein gesetzeskonformes Verhalten hinsichtlich der Schadstoffemissionen zu gewährleisten. Es handelt sich demnach bei dem Gesamtsystem, bestehend aus Motor und AGN, um ein gekoppeltes System. Der Abgasvolumenstrom wird primär durch die Motordrehzahl bestimmt, welche wiederum durch den Nutzer bzw. das Nutzungsprofil des Motors vorgegeben wird.⁶ Deshalb wird der Volumenstrom aus Sicht der AGN als nicht (maßgeblich) beeinflussbar betrachtet, vielmehr als Störgröße. Anders verhält sich dies bei der Abgastemperatur: Die Abgastemperatur kann unabhängig vom Nutzungsprofil durch eine Vertrimmung der internen Motorbetriebsparameter verändert werden. Das Zusammenspiel zwischen Motor und AGN kann regelungstechnisch als kaskadiertes System betrachtet werden, mit dem Motor als innerem (schnellerem) Regelkreis, welchem die Sollwerte vom äußeren (langsameren) Regelkreis, der AGN, vorgegeben werden. Die Aufheizprozesse der AGN bewegen sich im Zeitraum von einer halben Minute, die relevanten Zeitkonstanten des Motors hingegen, beispielsweise die Turboladerbeschleunigung, im Bereich von unter zwei Sekunden.

Ziel dieser Arbeit ist es, ein auf einem Seriensteuergerät implementierfähiges Regelungskonzept für den Dieselmotor *Deutz TCD 12.0 V6* zu erarbeiten, mit dem es möglich ist, die

⁵Mit Rohemissionen werden die Schadstoffemissionen des Abgases vor dem Eintritt in das AGN-System bezeichnet.

⁶Formelmäßig geht der Zusammenhang zwischen Motordrehzahl und Abgasmassenstrom aus den Gleichungen (3.27) und (3.30) hervor.

Abgastemperatur im Rahmen einer Regelung zu berücksichtigen. Auf diesem Weg soll die Leistungsfähigkeit der AGN an die sich durch das Nutzungsprofil des Motors ergebenden Rohemissionen angepasst werden. Dieses Konzept wird in dieser Arbeit als *temperaturgeführter Motor* bzw. *temperaturgeführte Motorregelung* bezeichnet. Die temperaturgeführte Motorregelung stellt einen Teilschritt auf dem Weg zum Fernziel der Entwicklung eines *emissionsgeführten Motors* dar, der neben der Temperatur zusätzlich die Rohemissionen berücksichtigt. Im Hinblick darauf wird mit der *modellprädiktiven Regelung* (engl.: *Model Predictive Control*, kurz: MPC) ein Ansatz gewählt, der in der Lage ist, mehrere, sich unter Umständen widersprechende Regelungsziele zu moderieren und Nebenbedingungen, welche sich durch die zulässigen Betriebsbereiche des Motors ergeben, zu berücksichtigen.

1.1 Konzept des emissionsgeführten Motors

Zur Einordnung der temperaturgeführten Motorregelung soll zunächst das Fernziel eines emissionsgeführten Motors beschrieben und motiviert werden. In diesem Zuge werden die Bestandteile der AGN sowie ihre Funktionsweisen dargestellt.

Die AGN von modernen Dieselmotoren besteht typischerweise aus einem Dieselpartikelfilter (DPF) und einem SCR-Katalysator (engl.: *Selective Catalytic Reaction Catalyst*, kurz: SCR).⁷ Daneben werden teilweise noch weitere Katalysatoren im Zusammenspiel mit einem DPF und einem SCR eingesetzt, auf die im Weiteren nicht eingegangen werden soll. Es sei jedoch angemerkt, dass der Deutz TCD 12.0 V6 solche weiteren Katalysatoren verwendet.⁸ Für die in diesem Abschnitt dargelegte Motivation des emissionsgeführten Motors wird ein System betrachtet, das lediglich aus einem DPF und einem SCR besteht. Hierfür wird zunächst die Funktionsweise der beiden Komponenten erläutert. Ihre Anordnung ist in Abb. 1.2 dargestellt. Das Abgas, welches den Motor verlässt, strömt zunächst durch den DPF und wird dort von den Schadstoffpartikeln befreit. Anschließend wird dem Abgas über eine Harnstoffeinspritzung Ammoniak zugesetzt, welches im SCR-Katalysator die Stickoxide in die unschädlichen Substanzen Wasser und Stickstoff umwandelt.

⁷SCR und SCR-Katalysator werden im Weiteren synonym verwendet.

⁸Der DPF ist in Kombination mit einem *Diesel-Oxidationskatalysator* (engl.: *Diesel Oxidation Catalyst*, kurz: DOC), welcher in den DPF als zusätzliche katalytische Schicht eingebracht wird, zu einem *katalytischen Rußfilter* (engl.: *Catalytic Soot Filter*, kurz: CSF) ausgeführt. Durch die zusätzliche katalytische Schicht wird das NO₂/NO_x-Verhältnis erhöht. Hinter dem SCR befindet sich ein *Ammoniak-Schlupf-Katalysator* (engl.: *Ammonia Slip Catalyst*, kurz: ASC), um unverbrauchtes Ammoniak in weniger schädliche Stickoxide zu konvertieren. Eine Übersicht der verschiedenen Technologien wird in [Joh11] gegeben.

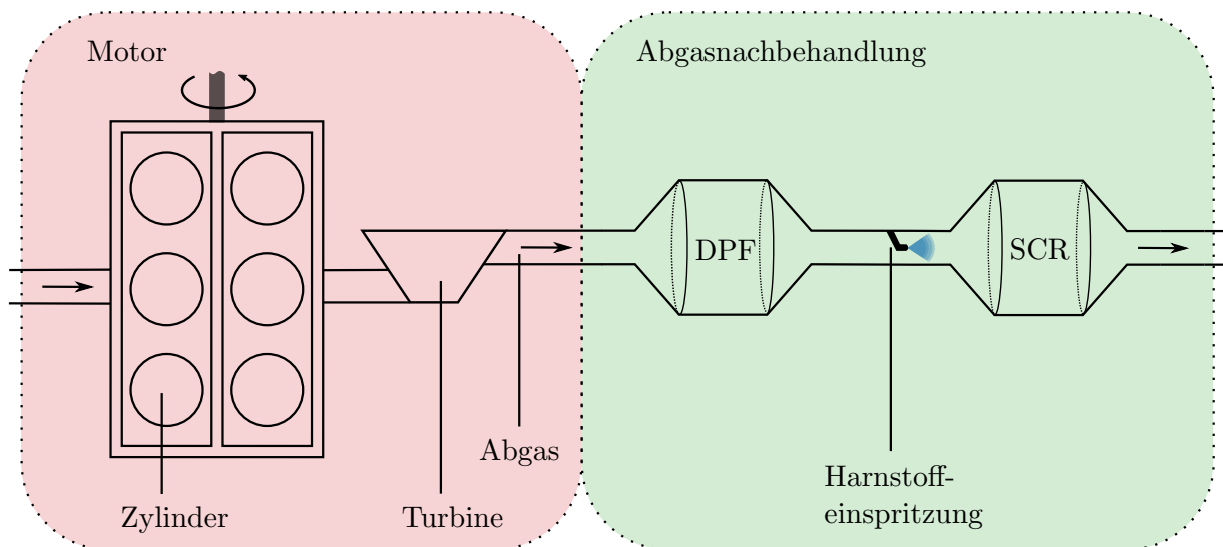
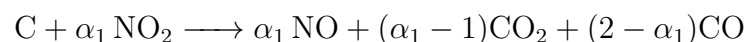


Abb. 1.2: Anordnung des Motors und der Abgasnachbehandlung.

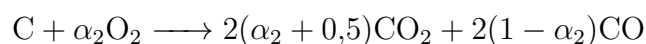
1.1.1 Funktionsprinzip des Dieselpartikelfilters

Der DPF besteht aus einer Keramik, die in einer Wabenstruktur von Kanälen durchzogen ist (vgl. Abb. 1.3). Die Kanäle sind jedoch wechselweise an der Vorder- und Rückseite verschlossen. Das an einer Seite eintretende Gas wird auf diese Weise gezwungen, durch eine poröse Schicht zu strömen, die die benachbarten Kanäle miteinander verbindet. Die Schicht enthält auf mikroskopischer Ebene kleine Poren, welche kleiner als das Gros der Partikel sind. Die gefilterten Schadstoffpartikel sammeln sich vor der porösen Schicht in den Kanälen mit offener Vorderseite sowie in den Poren selbst. Die Filterwirkung basiert auf Diffusion und mechanischem Abfangen der Partikel. Mit der Zeit bildet sich vor der Filterschicht eine Schicht aus herausgefilterten Partikeln, die die Filterwirkung verbessert. Mit diesem Wirkprinzip werden nahezu 100 % der Partikel herausgefiltert [KHDR13]. Durch die zusätzliche Partikelschicht erhöht sich allerdings im Gegenzug der Strömungswiderstand des Partikelfilters. Der damit einhergehende gesteigerte Abgasgegendruck wirkt dem Arbeitsspiel des Motors entgegen und führt damit zu einem erhöhten Kraftstoffverbrauch.

Um eine zu große Partikelbeladung des DPF zu verhindern, wird dieser durch eine aktive Temperaturerhöhung regeneriert. Primärer Bestandteil der aufgefangenen Schadstoffe ist Ruß, welcher hauptsächlich aus elementarem Kohlenstoff (C) besteht. Bei genügend hoher Abgastemperatur reagiert dieser mit Stickstoffdioxid (NO_2) [KHK02]



oder mit Sauerstoff (O_2) [KS96]



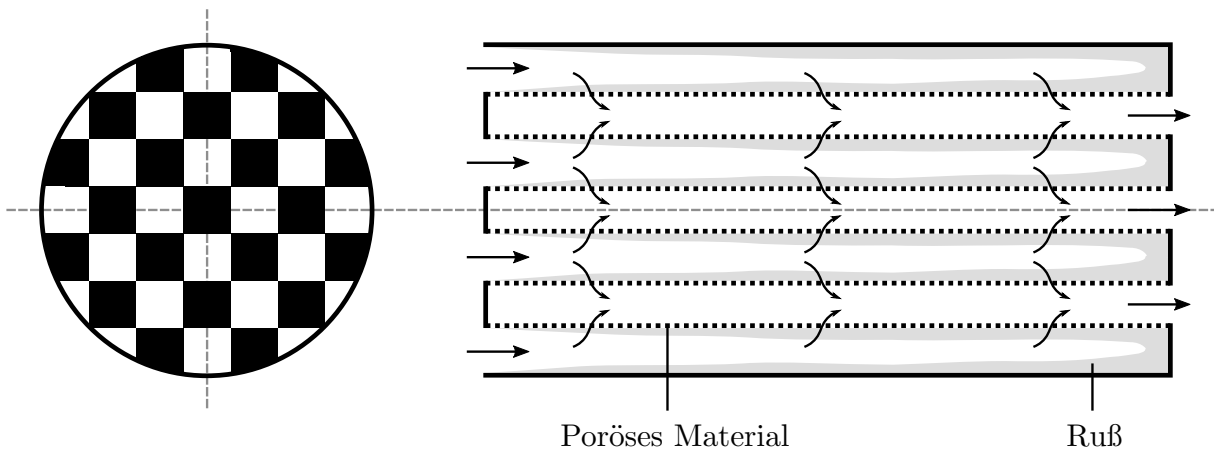


Abb. 1.3: Funktionsweise des DPF. Links: Schnitt entlang der Querachse; Rechts: Schnitt entlang der Längsachse.

zu Kohlenstoffdioxid (CO_2) sowie Kohlenstoffmonoxid (CO) und im Falle der Stickstoffdioxidreaktion zusätzlich zu Stickstoffmonoxid (NO). Die Parameter $\alpha_1 \in [1,65; 1,75]$, $\alpha_2 \in [0,8; 0,9]$ geben das Verhältnis der stattfindenden Reaktionen zueinander an. Die Reaktion mit Stickstoffdioxid⁹ findet bereits ab einer Temperatur von $250\text{ }^\circ\text{C}$ in nennenswerter Weise statt, die Reaktion mit Sauerstoff allerdings erst ab $500\text{ }^\circ\text{C}$ [Gör05, GZLH15]. Die Regenerationsgeschwindigkeit hängt sowohl von der Verfügbarkeit der Reaktanten als auch von der Temperatur ab.¹⁰

Neben dem Ruß ist Asche im Abgas enthalten. Asche besteht hauptsächlich aus anorganischen Stoffen, welche aus mitverbranntem Motoröl stammen [VNS03]. Im Gegensatz zu Ruß können diese nicht oxidiert werden und lagern sich dauerhaft im DPF an. Ab einer bestimmten Aschemenge wird der Partikelfilter daher ersetzt. Den Filter in Verbindung mit einem Aus- und Wiedereinbau zu reinigen bzw. zu waschen ist ebenfalls möglich.

1.1.2 Funktionsprinzip des SCR-Katalysators

Im Gegensatz zum DPF basiert die reinigende Eigenschaft des SCR auf einem chemischen Wirkprinzip. Der dem Abgas zugeführte Harnstoff (siehe Abb. 1.2) zersetzt sich in der Verrohrung bzw. in einem Mischsystem vor dem SCR sowie teilweise im Anfangsbereich des SCR zu Ammoniak (NH_3). Das Ammoniak tritt zusammen mit den Stickoxiden als Reaktanten und dem übrigen Abgas in das SCR-System ein. Der Aufbau des SCR ist grundsätzlich ähnlich zu dem des DPF. Das Abgas strömt ebenfalls durch ein von Kanälen

⁹Stickstoffdioxid reagiert auf unterschiedlichen Wegen mit Kohlenstoff. Eine Besprechung der Reaktionen sowie der zugehörigen Kinetik wird in [JTB09] geliefert.

¹⁰Die meisten chemischen Reaktionsraten können über einen *Arrhenius-Ansatz* beschrieben werden. Die Geschwindigkeit einer chemischen Reaktion hängt damit i. d. R. in exponentieller Weise von der Temperatur ab [Lee99].

durchzogenes Substrat. Jedoch sind diese Kanäle nicht durchlässig und zudem mit einem Katalysator¹¹ beschichtet (siehe Abb. 1.4). Reaktionskinetisch reagiert maßgeblich zuvor

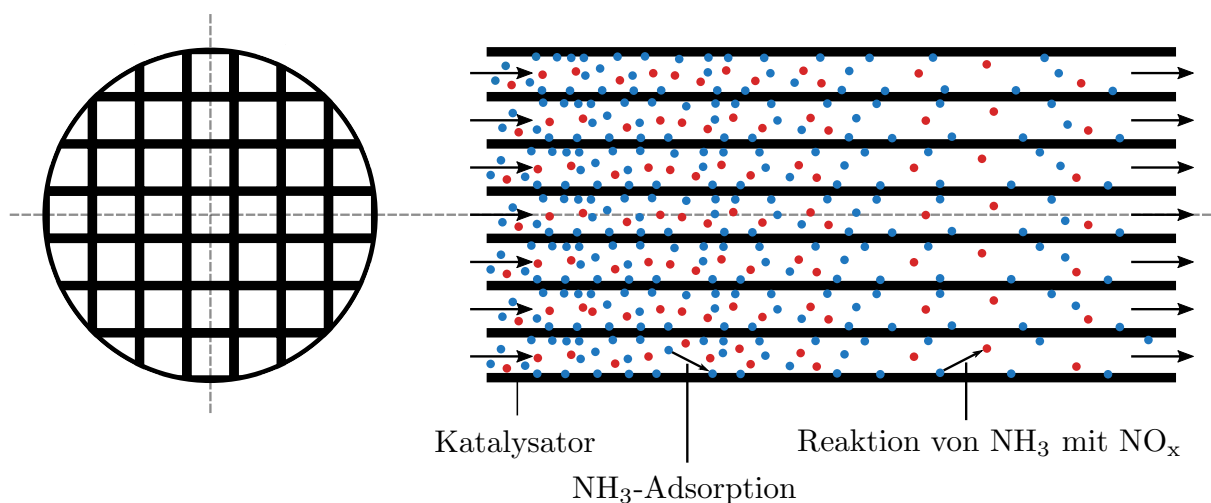
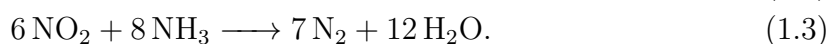
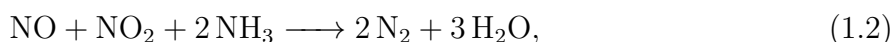
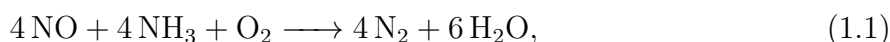


Abb. 1.4: Funktionsweise des SCR-Katalysators. Links: Schnitt entlang der Querachse; Rechts: Schnitt entlang der Längsachse.

adsorbiertes Ammoniak mit den Stickoxiden, also das Ammoniak, das sich zuvor an der Oberfläche des Katalysatormaterials angereichert hat.¹² Das adsorbierte Ammoniak wird als *eingelagertes Ammoniak* bezeichnet. Es ergibt sich für den stationären Fall eine in Strömungsrichtung abnehmende Konzentration der Reaktanten. Im Idealfall wird genau so viel Ammoniak in das System gegeben, dass beide Reaktanten vollständig reagieren und keines der Edukte das System verlässt. Im SCR finden hauptsächlich drei chemische Reaktionen statt:

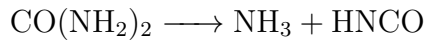


Reaktion (1.1) wird als *Standard-SCR-Reaktion* bezeichnet, die aufgrund des stets in nennenswerter Weise vorhandenen NO in sämtlichen Betriebspunkten stattfindet. Aufgrund ihrer höheren Geschwindigkeit wird die *schnelle SCR-Reaktion* (1.2) bevorzugt, allerdings wird diese Reaktion maßgeblich durch das Vorhandensein von NO₂ begrenzt. Ohne einen gesonderten *Dieseloxydationskatalysator* [Joh11] liegt das NO₂/NO_x-Verhältnis zwischen 5 % und 10 % ([RKV16], zitiert in [RVJ⁺17]). Das bedeutet, dass ohne diesen gesonderten Katalysator weniger als 10 % der Stickoxide über diese Reaktion umgewandelt werden. Reaktion 1.3 wird *langsame SCR-Reaktion* genannt und findet bei einem NO₂/NO_x-Verhältnis größer 0,5 statt. Diese Reaktion wird bei der Auslegung des Systems nicht forciert, da sie langsam abläuft und mehr NH₃ als die beiden anderen Reaktionen verbraucht, was in einem schlechteren Wirkungsgrad resultiert.

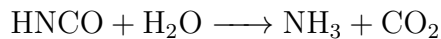
¹¹Übliche Katalysatormaterialien sind Vanadium sowie Eisen- und Kupferzeolith [LZL18].

¹²Es sei angemerkt, dass der Katalysator keine glatte Schicht bildet, sondern vielmehr eine schwammartige Struktur besitzt.

Die Ammoniakbereitstellung über Harnstoffeinspritzung funktioniert über zwei Wege: In der Thermolysereaktion



wird der Harnstoff ($\text{CO}(\text{NH}_2)_2$) ab einer Temperatur von 133 °C in Ammoniak und Isocyanensäure (HNCO) umgewandelt [SCH⁺04]. In der Hydrolysereaktion



zersetzt sich die Isocyanensäure zu Ammoniak und Kohlenstoffdioxid. Ohne Beteiligung des Katalysators findet die Hydrolysereaktion ab einer Temperatur von 350 °C statt, mit seiner Beteiligung bereits ab 150 °C [YKB⁺04]. Die maximale Einspritzmenge ist abhängig von der Abgastemperatur begrenzt. Ist die Einspritzmenge zu hoch, reagiert die Isocyanensäure mit dem noch intakten Harnstoff zu festen Nebenprodukten, die sich im SCR-System ablagern und dieses damit verstopfen können [SCH⁺04]. Ebenfalls kann dann der Fall auftreten, dass sich die abgelagerten Harnstoffkomponenten zu einem (ungewollten) späteren Zeitpunkt auflösen und damit eine nicht vorgesehene Ammoniakbeigabe darstellen, die der SCR u. U. nicht aufnehmen kann. Dies ist wegen der gesetzlichen Begrenzung der Ammoniakemissionen problematisch (vgl. Tab. C.1).

Die Leistungsfähigkeit des SCR-Systems hängt damit von folgenden Einflüssen ab: Aus denselben Gründen, die bereits bei der Beschreibung des DPF angeführt wurden, ist die Temperatur von Bedeutung. Das NO_2/NO_x -Verhältnis liegt wegen der bevorzugten schnellen SCR-Reaktion bei 0,5 und ist deshalb ebenfalls relevant. Neben den zuvor genannten die Reaktionsgeschwindigkeit beeinflussenden Faktoren spielt die Zeit, die für die Reaktionen zur Verfügung steht, eine Rolle. Je langsamer das Gas durch den SCR strömt, desto mehr Zeit steht für die Reaktionen zur Verfügung. Ein Maß für die Geschwindigkeit des Gases ist der Volumenstrom. Eine Voraussetzung dafür, dass die Reaktionen überhaupt stattfinden können, ist die temperaturabhängige Verfügbarkeit von Ammoniak.

1.1.3 Emissionsführung

Das Ziel der Abgasnachbehandlung ist die Reduktion der Schadstoffe, sodass die gesetzlichen Grenzwerte eingehalten werden. Der DPF und das SCR-System sollen gemäß der Betrachtung als äußerer Regler einer Kaskadenregelung im Weiteren als zu regelnde Systeme betrachtet werden. Es soll herausgearbeitet werden, welche Systemeingänge motorseitig beeinflussbar sind und damit potentielle Stellgrößen aus Sicht des AGN-Systems darstellen.

Beim DPF besteht das Regelungsziel darin, die Rußbelastung unterhalb eines bestimmten Grenzwertes zu halten, um eine *aktive Regeneration* zu vermeiden bzw. möglichst lange hinauszuzögern. Bei der aktiven Regeneration wird der Motor in einen bestimmten Betriebs-

zustand gebracht, der dazu dient, den DPF zu leeren. Während der aktiven Regeneration kann das Gerät, in dem der Motor verbaut ist, nicht verwendet werden. Die Rußwerte hinter dem DPF werden aufgrund seines Funktionsprinzips eingehalten und stellen damit kein Regelungsziel dar. Als Stellgrößen stehen die Rußemissionen des Motors, die Abgastemperatur und die Stickoxidemissionen zur Verfügung. Die Rußbelastung kann reduziert werden, indem die Regenerationsrate größer ist als der Rußeintrag. Entsprechend kann dies durch eine Reduktion der Rußrohmissionen oder durch eine Steigerung der Regenerationsrate ermöglicht werden. Die Regenerationsrate lässt sich durch eine höhere Abgastemperatur oder eine höhere Anzahl an Stickoxiden – primär NO_2 – steigern. Denkbar wäre als weitere Stellgröße eine aktive Beeinflussung des NO_2/NO_x -Verhältnisses. Dieses lässt sich jedoch lediglich im unteren einstelligen Prozentbereich motorseitig beeinflussen und wird daher nicht als geeignete Stellgröße betrachtet [RVJ⁺17].

Die Einhaltung der NO_x -Emissionen hinter dem SCR sind das Regelungsziel des SCR-Systems. Als Stellgrößen stehen die NO_x -Emissionen des Motors sowie die Abgastemperatur zur Verfügung. Denkbar wäre ebenfalls eine Beeinflussung des NO_2/NO_x -Verhältnisses, welche jedoch aus den o.g. Gründen ausscheidet. Mit steigender Abgastemperatur lässt sich zum einen mehr Harnstoff einspritzen und zum anderen laufen die chemischen Reaktionen im SCR schneller ab. Ersteres ist wichtig, um überhaupt genügend Reaktionsmittel zur Verfügung zu haben, letzteres erhöht die Umsatzrate. Eine Reduktion der NO_x -Rohmissionen verringert die geforderte Umsatzrate des SCR. Eine SCR-interne Regelung hat zum Ziel, die Menge des eingelagerten Ammoniaks für eine bestimmte Temperatur zu maximieren, da dies ebenfalls den Wirkungsgrad erhöht. Jedoch ist die Einlagerungsfähigkeit im SCR temperaturabhängig. Je geringer die Temperatur ist, desto mehr Ammoniak kann eingelagert werden (vgl. Abbildung in [WCV⁺07]). Bei einer zu schnellen Temperatursteigerung kann das dann überschüssig eingelagerte Ammoniak eventuell nicht schnell genug neutralisiert werden und verlässt den SCR als Schadstoff; es wird von *Ammoniakschlupf* gesprochen.

Aus diesen Zusammenhängen lassen sich seitens der AGN Randbedingungen ableiten. Ziel der *Emissionsführung* ist es, aus diesen Randbedingungen Vorgaben für den Motorbetrieb abzuleiten und dabei eventuelle Widersprüche aufzulösen. Das Konzept der *emissionsgeführten Motors* ist in Abb. 1.5 skizziert. Auffällig ist, dass keine konkreten Sollwerte gefordert sind, sondern Bereiche, in denen sich die Rohmissionen und die Temperatur befinden dürfen. Der maximale Temperaturgradient $\left(\frac{dT_{\text{SCR}}}{dt}\right)_{\text{max}}$ findet sich nicht als motorseitige Nebenbedingung wieder. Die Begrenzung ist primär dafür gedacht, eine für das SCR-System zu hohe Temperaturanforderung seitens des DPF zu begrenzen.¹³ Es ist bei dem Entwurf der SCR-Regelung sicherzustellen, dass Temperaturgradienten durch einen regulären Betriebspunktwechsel durch das Nutzungsprofil des Motors bei der (Harnstoff-)Dosierstrategie berücksichtigt werden. Aus regelungstechnischer Sicht handelt es sich damit nicht um Sollwerte, sondern um Nebenbedingungen. Es ist also ein Regelungskonzept für den Motor gefordert, welches mit Nebenbedingungen umgehen kann.

¹³Es ist jedoch denkbar, diese in Form einer Maximaltemperatur T_{max} ebenfalls zu einer Temperaturvorgabe für den Motor zu machen.

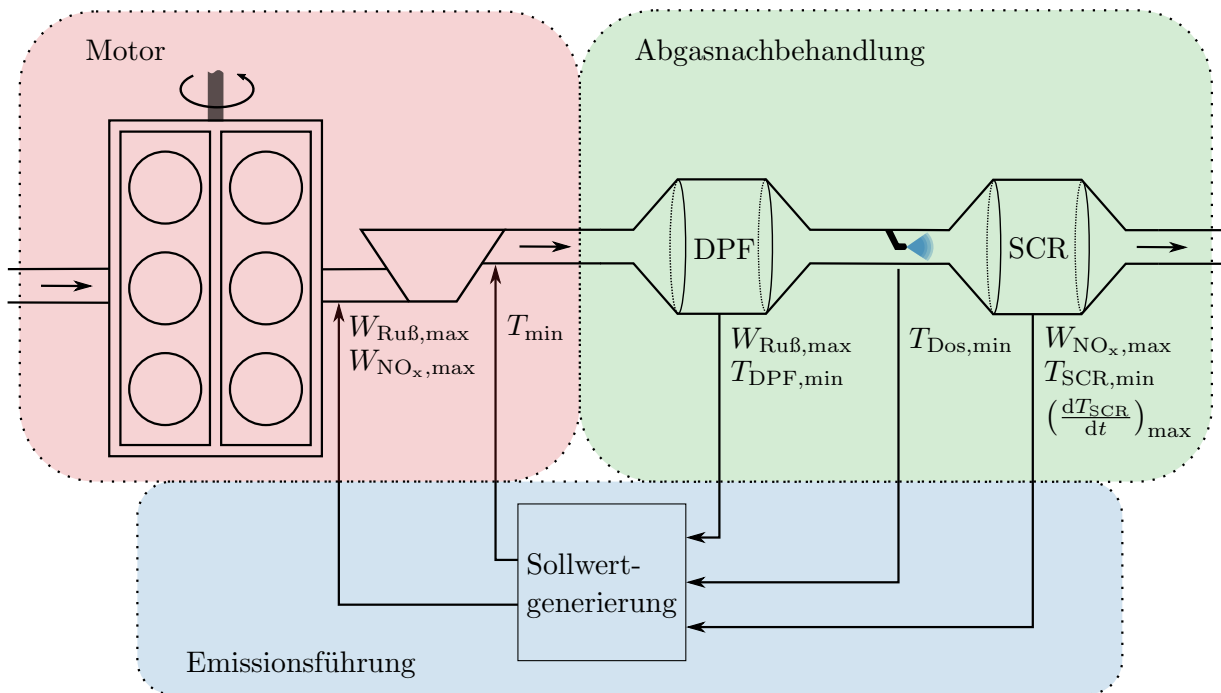


Abb. 1.5: Emissionsgeführter Motor.

1.2 Temperaturgeführter Motor

Ein Teilaspekt des emissionsgeführten Motors ist die motorseitige Berücksichtigung einer AGN-seitig geforderten Mindesttemperatur T_{min} des Abgases. Die alleinige Berücksichtigung der Abgastemperatur unter Außerachtlassung der Rohemissionen wird im Weiteren als *temperaturgeführter Motor* bezeichnet. Grundsätzlich existieren mehrere Möglichkeiten, die Abgastemperatur des Motors zu beeinflussen. Diese hängt vom Betriebspunkt (Drehzahl und Drehmoment), von Einstellungen bezüglich der Kraftstoffeindosierung in den Brennraum und von der Luftversorgung des Motors ab. Genauere Ausführungen werden in Abschnitt 5.1.2 geliefert. Im Rahmen dieser Arbeit erfolgt die Beeinflussung über die Luftversorgung des Motors unter Verwendung eines Wastegates (WG) und einer Abgasklappe (engl.: *Exhaust Throttle Valve*, kurz: ETV) als Aktuatoren.

1.2.1 Luftpfad

Sowohl das WG als auch die ETV sind Bestandteile des in dieser Arbeit betrachteten, motorischen Luftpfades. In Abb. 1.6 ist dieser für den betrachteten Dieselmotor Deutz TCD 12.0 V6 systemisch dargestellt. Der Luftpfad ist für die Luftversorgung des Motors zuständig. Über einen Kompressor wird die einströmende Frischluft verdichtet und in die Zylinder gedrückt, wobei je eine Einlassflut eine Zylinderbank versorgt. Der Ladeluftkühler erhöht durch die Temperaturabsenkung der Ladeluft deren Dichte. Aufgrund

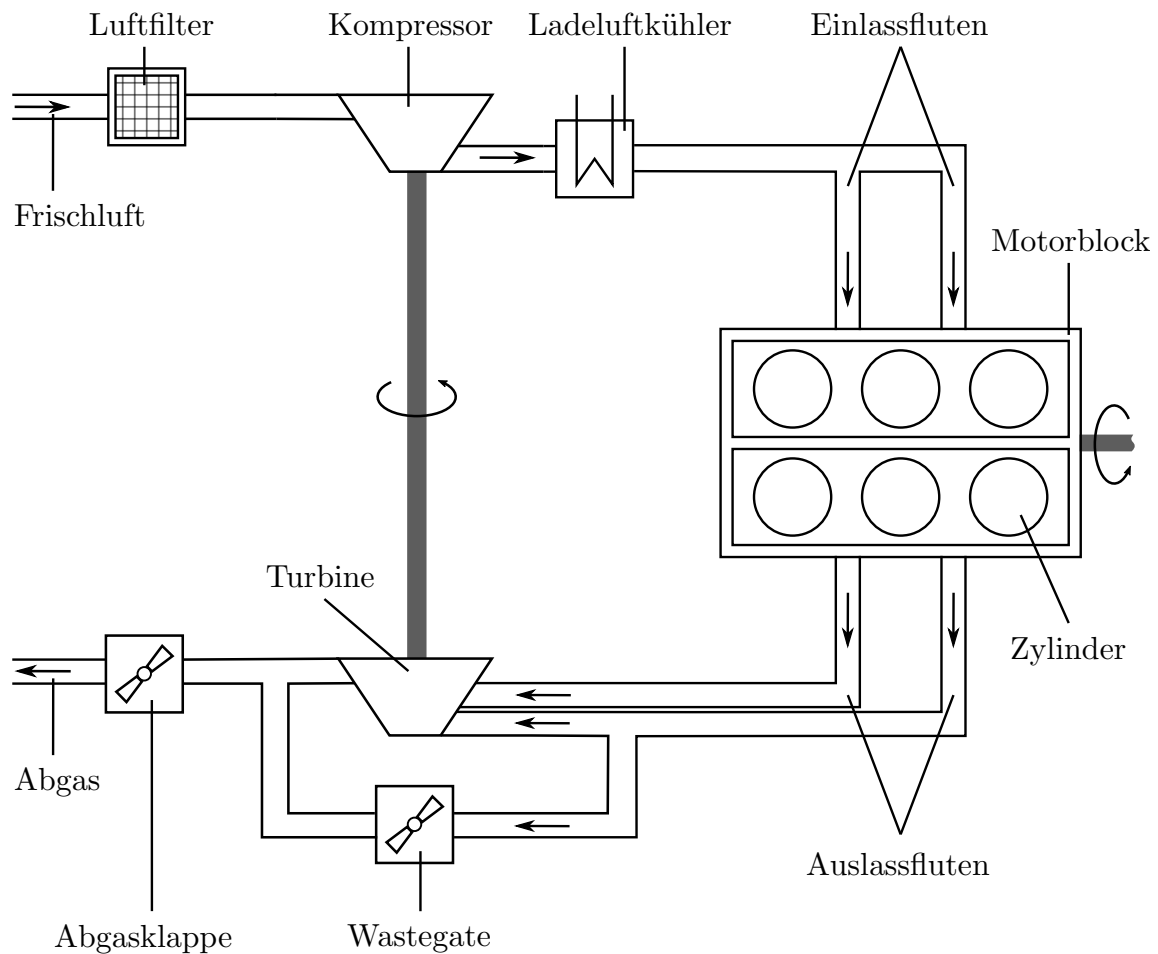


Abb. 1.6: Systemdarstellung des Deutz TCD 12.0 V6 Dieselmotors.

der erhöhten Dichte kann den Zylindern bei gleichbleibendem Druck mehr Luftmasse zugeführt werden.¹⁴ In den Zylindern wird der Ladeluft Dieselkraftstoff zugesetzt, der sich in der hochverdichteten Ansaugluft selbst entzündet und schlussendlich ein Drehmoment erzeugt. Das Abgas verlässt über die Auslassfluten den Motorblock und treibt eine Turbine an, welche über eine Welle mit dem Kompressor verbunden ist und diesen wiederum antreibt. Kompressor und Turbine bilden zusammen den Turbolader. Eine der Auslassfluten kann über das WG an der Turbine vorbeigeführt werden, was eine Steuerung der Leistungsaufnahme des Turboladers ermöglicht und damit über eine Anpassung der Ladeluftmenge auch eine Anpassung der Abgastemperatur bewirken kann.¹⁵ Hinter der Turbine und dem WG befindet sich die ETV. Die ETV erzeugt einen erhöhten Abgasgegendruck, der zum einen ebenfalls die Leistungsaufnahme des Turboladers reduziert, und zum anderen die Ladungswechselverluste des Motors erhöht. Letzteres sorgt dafür, dass zum Aufrechterhalten des gleichen effektiven Drehmoments mehr Kraftstoff pro Arbeitsspiel eingespritzt werden

¹⁴Die geringere Temperatur hat zusätzlich eine Verringerung der entstehenden Stickoxide zur Folge [PKS09, S. 283ff].

¹⁵Wird die gleiche Kraftstoffmenge bei weniger Ladeluft verbrannt, ist entsprechend weniger Luft zu erhitzen, was diese heißer werden lässt.

muss. Dies führt zu einer zusätzlichen Temperaturerhöhung des Abgases. Das WG und die ETV haben folglich grundsätzlich ähnliche Effekte, jedoch liegt der Fokus der ETV durch die Wirkungsgradverschlechterung auf der Steuerung der Abgastemperatur, wohingegen das WG primär zur Beeinflussung des Ladedrucks sowie der Turboladerdrehzahl verwendet wird. Eine tiefergehende Betrachtung der Wirkmechanismen von WG und ETV erfolgt in Abschnitt 5.1.3.

1.2.2 Modellprädiktive Regelung

Das motorinterne Regelungsziel ist die Bereitstellung eines bestimmten Ladedrucks, da dieser maßgeblich die in die Zylinder einströmende Luftmenge beeinflusst. Die Luftmenge ist wiederum für die Leistung des Motors und für die entstehenden Schadstoffe entscheidend. Damit ergibt sich insgesamt ein System mit zwei Regelgrößen (Ladedruck, Mindestabgastemperatur) und zwei Aktuatoren (WG, ETV). Es gibt Betriebspunkte, in denen sich die Regelungsziele widersprechen können und eine Gewichtung zwischen den Zielen gefordert ist. Dies ist beispielsweise der Fall, wenn gleichzeitig ein hoher Ladedruck und eine hohe Abgastemperatur gefordert sind. Neben diesen Regelungszielen gibt es eine Reihe an Nebenbedingungen, die zur Gewährleistung des sicheren Betriebs des Motors zwingend eingehalten werden müssen. Diese sind beispielsweise die maximale Turboladerdrehzahl oder die maximale Abgastemperatur vor der Turbine. Eine Verletzung der Nebenbedingungen führt zu einer Beschädigung des Turboladers.

Ein Regelungskonzept, welches sich dadurch auszeichnet, mehrere Nebenbedingungen beachten und verschiedene Regelungsziele moderieren zu können, ist die MPC. Diese Eigenschaft ist im Hinblick auf die zuvor genannten Ziele gefordert, aber insbesondere auch vor dem Hintergrund einer späteren Weiterentwicklung zur emissionsgeführten Motorregelung relevant. Wie in Abb. 1.5 ersichtlich ist, sind in diesem Fall mehr Vorgaben seitens der AGN zu beachten als bei einer rein temperaturegeführten Motorregelung. Ebenfalls wird dann eine Anpassung der Verbrennung und den daraus resultierenden zusätzlichen Stellgrößen zur gezielten Beeinflussung der Rohemissionen von Bedeutung sein.

Das grundsätzliche Funktionsprinzip der modellprädiktiven Regelung basiert auf der Lösung eines Optimierungsproblems zur Laufzeit des eingebetteten Systems. Unter Zuhilfenahme eines Modells der Regelstrecke werden die optimalen Stellgrößen im Sinne einer Gütefunktion und der einzuhaltenden Nebenbedingungen ermittelt. Die Bezeichnung *prädiktive Regelung* rührt daher, dass die Stellgrößen nicht nur für den aktuellen Zeitpunkt, sondern ebenso für zukünftige Zeitpunkte berechnet werden. Dies geschieht über eine Simulation des Modells in die Zukunft. Jedoch wird lediglich die ermittelte Stellgröße für den aktuellen Zeitpunkt angewendet. Im nächsten Zeitschritt wird die Berechnung erneut durchgeführt. Obwohl die anderen Stellgrößen nicht direkt verwendet werden, ist deren Berechnung prinzipbedingt erforderlich. Grafisch ist das Funktionsprinzip der MPC in Abb. 4.1 dargestellt.

Das sicherlich größte technische Hindernis bei der Umsetzung einer MPC ist die benötigte Rechenleistung. Historisch fand die MPC aus diesem Grund zuerst Anwendung in der Prozesstechnik zur Steuerung von chemischen Anlagen. Dort sind die Zeitkonstanten für eine Regelung in Echtzeit groß genug, und zur Lösung der MPC können stationäre (Groß-)Rechner verwendet werden. Mit der zunehmenden Leistungsfähigkeit der Digitalrechner wurde die MPC jedoch auch in immer schnelleren und mobileren Anwendungen einsetzbar. [MH99], [QB03]

Neben der zur Laufzeit zu lösenden *impliziten MPC* ist für besonders schnell zu regelnde Systeme die *explizite MPC* populär geworden. Bei diesem Ansatz wird das Optimierungsproblem im Vorhinein für die verschiedenen Betriebssituationen gelöst und auf dem Zielsystem in Form einer *Lookup-Tabelle* hinterlegt. Die aktuell gültige Lösung des Optimierungsproblems wird zur Laufzeit über das Durchsuchen dieser Tabelle ermittelt. Der größte Nachteil dieser Methode besteht darin, dass die Anzahl der Einträge der Tabelle in exponentieller Weise von der Länge des Prädiktionshorizonts, der Anzahl der Modelleingänge und Zustände sowie der Anzahl der Nebenbedingungen abhängt und damit der dafür erforderliche Speicherplatz ebenso schnell steigt [BMDP02]. Dieses Problem wird noch zusätzlich dadurch verschärft, dass der in dieser Arbeit betrachtete, zu regelnde Luftpfad ein hochgradig nichtlineares System ist (vgl. Kapitel 3), welches beispielsweise mehrere explizite MPC für unterschiedliche Arbeitspunkte erfordern würde.¹⁶ Selbst für Umsetzungen mit einer praktikablen Größe der Tabelle kann es schneller sein, das Optimierungsproblem zu lösen, anstatt die Tabelle zu durchsuchen [WB10].¹⁷

Anstelle der expliziten MPC wird der Ansatz verfolgt, eine effiziente Implementierung einer nichtlinearen impliziten MPC umzusetzen. Dies erlaubt eine Adaption der Regelstrecke sowie eine Verstellung der Reglereigenschaften zur Laufzeit. Mit letzterer Eigenschaft ist eine manuelle *Feinkalibrierung* am Motorenprüfstand der ansonsten am Computer ausgelegten Regelung praktikabel.¹⁸

1.2.3 Anforderungen an den Lösungsalgorithmus

Die Software auf einem Motorsteuergerät (engl.: *Electronic Control Unit*, kurz: ECU) (vgl. Abb. 1.7) besteht aus zahlreichen, unterschiedlichen Programmen, die zyklisch aufgerufen werden. Organisiert ist das zyklische Aufrufen in sogenannten *Tasks*. Jede Task wird in einem bestimmten Zeitraster aufgerufen, beispielsweise alle 50 ms – dies ist die Aufruf-

¹⁶Es gibt Ansätze der expliziten MPC, die auf ein *künstliches neuronales Netz* setzen, anstatt auf eine *Lookup-Tabelle*. Damit können die Nachteile hinsichtlich des Speicherverbrauchs und der Zeit zum Auffinden der Lösung umgangen werden [CSA⁺18].

¹⁷Die praktische Umsetzbarkeit ist für Systeme mit beispielsweise $n \geq 5$, $m \geq 3$, $l \geq 12$ nicht mehr gegeben [WB10]. n : Anzahl der Zustände, m : Anzahl der Eingänge, l : Anzahl der Nebenbedingungen.

¹⁸Die Feinkalibrierung einer expliziten MPC würde das Kopieren abstrakter, maschinengenerierter Datensätze vom Entwicklungsrechner auf das Motorsteuergerät erfordern. Manuelle Änderungen an einzelnen Parametern sind damit nicht sinnvoll möglich.

zeit, die für den umgesetzten Regelalgorithmus verwendet wird. Sämtliche einer Task zugeordneten Programme müssen entsprechend beendet sein, bevor diese erneut aufgerufen wird; der MPC-Lösungsalgorithmus ist eines dieser Programme. Wenn die Ausführung sämtlicher Programme einer Task genau so lange dauert, bis die Task erneut aufgerufen wird, dann ist die Recheneinheit (engl.: *Central Processing Unit*, kurz: CPU) zu 100 % ausgelastet und es bleibt keine Rechenzeit für die Programme anderer Tasks übrig. Dies bedeutet, dass ein *Echtzeitfaktor* (engl.: *Real-Time Factor*, kurz: RTF)

$$\text{RTF} := \frac{t_{\text{task}}}{t_{\text{CPU}}} \quad (1.4)$$

deutlich größer als eins gefordert ist.

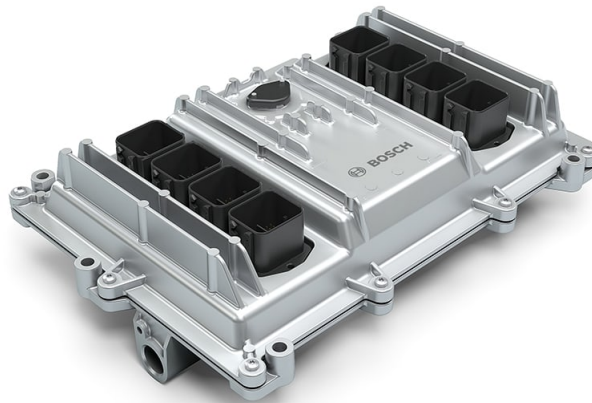


Abb. 1.7: Symbolbild des im Deutz TCD 12.0 V6 verbauten Motorsteuergeräts für Großmotoren MD1-CE101. Quelle: ©Robert Bosch GmbH.

Neben der Rechenkapazität ist auch der zur Verfügung stehende Direktzugriffsspeicher (engl.: *Random Access Memory*, kurz: RAM) begrenzt (vgl. Tab. 1.1). Im RAM werden die Daten gespeichert, die sich zur Laufzeit ändern können. Dies sind zum einen sämtliche internen Variablen, die beim nächsten Programmaufruf wieder benötigt werden, und zum anderen lokale Variablen, die unmittelbar beim Programmaufruf erzeugt und anschließend wieder verworfen werden. Erstere werden auf dem sogenannten *Heap* (Freispeicher) allokiert, letztere auf dem sogenannten *Stack* (Stapelspeicher). Heap und Stack sind virtuelle Bereiche auf dem RAM. Für den Stack gilt – bezogen auf den in dieser Arbeit betrachteten Deutz TCD 12.0 V6 – die firmeninterne Design-Richtlinie, dass ein Programm nicht mehr als 2 kB auf diesem ablegen darf.¹⁹ Der Heap darf hingegen – zumindest theoretisch – voll ausgeschöpft werden. Dieser ist tatsächlich allerdings auf einen Bruchteil des verfügbaren

¹⁹Die Limitierung der Stack-Größe hat sicherheitstechnische Gründe: Der insgesamt verbrauchte Stack hängt davon ab, welche Programme gleichzeitig auf dem RAM liegen. Dies kann sich zur Laufzeit ändern. Wird der maximal verfügbare RAM überschritten, führt dies zu einem Absturz der ECU.

RAM begrenzt, da der RAM zum einen auf die einzelnen Rechenkerne aufgeteilt ist und zum anderen der Rechenkern, auf dem der MPC-Lösungsalgorithmus ausgeführt wird, zu einem signifikanten Anteil dauerhaft vom Stack, Betriebssystem und anderen Programmen reserviert bzw. belegt ist. Aus diesem Grund wird hier eine verfügbare Obergrenze von 5 kB als vertretbar angesehen. Die Allokierung von Variablen auf dem Heap ist insofern kritisch, da dieser Speicher dauerhaft reserviert ist und damit den anderen Programmen nicht zur Verfügung steht. Im Idealfall werden deshalb auf dem Heap ausschließlich Werte abgelegt, die beim erneuten Programmaufruf wieder benötigt werden. Unveränderliche Parameter hingegen werden im *Flash* gespeichert.

Tab. 1.1: Eckdaten des in dem MD1-CE101 verbauten Mikrocontrollers *Infineon TC297 AURIX TriCore*.

| | | |
|------------|-----|-----|
| Rechentakt | 300 | MHz |
| Kerne | 3 | |
| RAM | 512 | kB |
| Flash | 8 | MB |

Abseits der zuvor genannten *harten* Kriterien, ohne deren Erfüllung der hier verfolgte Ansatz nicht funktioniert, gibt es *weiche* Kriterien, die einbezogen werden sollen. Diese beziehen sich auf die Aspekte der Qualitätssicherung, Wartbarkeit sowie Erweiterbarkeit.²⁰ Für die Qualitätssicherung ist ein möglichst *deterministisches* Verhalten der Software wünschenswert. Dies bedeutet, dass die Ausführungszeiten nicht zu stark bzw. idealerweise gar nicht schwanken. Auftreten kann ein solches Verhalten z. B. bei iterativen Lösungsverfahren, die mit einem Abbruchkriterium arbeiten und so je nach Anfangsbedingungen unterschiedlich viele Iterationen durchführen. Stark schwankende Ausführungszeiten erschweren mögliche Stresstests, die dazu dienen, herauszufinden, ob die Rechenleistung der CPU überstrapaziert wird. Bei dynamischer Speicherallokierung wäre ein Stresstest ebenso für den RAM durchzuführen. Dynamische Speicherallokierung ist für die Software der verwendeten Motorsteuergeräte jedoch per Design-Richtlinie verboten, womit eine *Worst-Case-Abschätzung* im Vorhinein durch eine Analyse der statischen Speicherbelegung möglich ist.²¹ Qualitätssichernd wirkt sich ebenfalls ein *robuster* Lösungsalgorithmus aus. Es ist ein Algorithmus wünschenswert, der auch in Grenzbereichen des Motorbetriebs stabile Lösungen liefert. Andernfalls besteht – beispielsweise durch das Verletzen von Nebenbedingungen, wie der maximalen Turboladerdrehzahl – im schlimmsten Fall die Gefahr, dass der Motor beschädigt wird. Auch ist ein *modularer* Aufbau für möglichst granulare Tests der Software wünschenswert. Es lässt sich dann mit weniger Aufwand zielgerichtet testen, was für die Qualität der Tests förderlich ist. Für die Wartbarkeit ist dies ebenfalls interessant, da durch einen modularen Aufbau die Strukturen klarer sind und so mögliche Anpassungen an der richtigen Stelle stattfinden. Dies gilt ebenso für den Faktor der

²⁰Eine Definition und Einordnung der Begriffe ist in [RB09] zu finden.

²¹Dynamische Speicherallokierung bedeutet, dass sich die Programme zur Laufzeit Speicherbereiche von im Vorhinein u. U. unbekannter Größe reservieren.

Erweiterbarkeit. Sämtliche Aspekte profitieren von einem möglichst *unkomplizierten* und damit leichter nachvollziehbaren Ansatz.

Insgesamt ergeben sich die in Tab. 1.2 aufgeführten Rahmenbedingungen für eine programmiertechnische Umsetzung. Diese Bedingungen sind als Richtlinie für bzw. als Zielsetzung in dieser Arbeit zu verstehen.

Tab. 1.2: Anforderungen an die programmiertechnische Umsetzung des Lösungsalgorithmus der MPC.

| Harte Kriterien | |
|--|------------------------------|
| Maximaler Anteil an der CPU-Auslastung | 10 % |
| Maximaler Stack-Speicher | 2 kB |
| Maximaler Heap-Speicher | 5 kB |
| Speicherallokierung | keine dynamische Allokierung |
| Weiche Kriterien | |
| Verhalten | deterministisch, robust |
| Aufbau | modular |

1.3 Gliederung

Die vorliegende Arbeit gliedert sich in die folgenden Abschnitte:

Kapitel 2 beinhaltet die Darstellung der theoretischen Grundlagen für diese Arbeit. Es erfolgt eine Besprechung der Methoden zur Beschreibung nichtlinearer Systeme, auf denen das Luftpfadmodell basiert und die auch zur Berechnung des MPC-Algorithmus verwendet werden. Des Weiteren werden die Grundlagen zur Lösung von statischen und dynamischen Optimierungsproblemen eingeführt, auf denen der entwickelte MPC-Lösungsalgorithmus fußt.

Kapitel 3 widmet sich zunächst der (mathematischen) Beschreibung des Luftpfades des Dieselmotors *Deutz TCD 12.0 V6*. Im Rahmen dessen werden neue Modellierungsansätze für die Beschreibung der Abgastemperatur sowie der Beschreibung der ETV präsentiert. Darüber hinaus wird das Vorgehen zur Identifikation der Modellparameter erläutert. Eine Bewertung des Modells erfolgt anhand einer Simulation und einer Prüfstandsaufnahme. Ebenfalls bewertet wird die Eignung verschiedener Sensoren zur Stützung des Modells im eingebetteten Betrieb. Der zur Generierung von echtzeitfähigem Quellcode entwickelte C-Code-Generator wird im letzten Abschnitt vorgestellt. Das Kapitel schließt mit einer Bewertung des generierten C-Codes im Hinblick auf dessen rechentechnische Eigenschaften.

Kapitel 4 stellt die Entwicklung des nichtlinearen modellprädiktiven Regelungsalgorithmus vor, der zur temperaturgeführten Motorregelung verwendet werden soll. Der Algorithmus setzt auf dem Verfahren nach *Englert et al.* [EVM⁺19] auf, welches im Hinblick auf eine höhere Robustheit und einen geringeren Ressourcenverbrauch erweitert und adaptiert wird. Die Erweiterungen bestehen aus einer auf die MPC zugeschnittenen Liniensuche, einer nicht-äquidistanten Diskretisierung sowie einem separat wählbaren Kontroll-Horizont. Adaptiert wird die Formulierung der Nebenbedingungen hinsichtlich einer geringen zu speichernden Variablenanzahl. Es erfolgt eine qualitative Bewertung des Algorithmus, wobei bereits auf die Ergebnisse aus Kapitel 5 vorgegriffen wird.

Kapitel 5 beinhaltet die regelungstechnische Umsetzung der temperaturgeführten Motorregelung. Dafür wird zunächst die zu lösende Regelungsaufgabe in Form eines dynamischen Optimierungsproblems ausgedrückt. Für dessen Berechnung wird der entwickelte MPC-Lösungsalgorithmus herangezogen. Ausgehend von einer als sinnvoll erachteten initialen Konfiguration des Algorithmus werden Stück für Stück die am Algorithmus gemachten Erweiterungen ausgeschöpft, um die in Tab. 1.2 genannten Anforderungen für eine Umsetzung auf dem Deutz TCD 12.0 V6 zu erfüllen. Eine Bewertung erfolgt anhand synthetischer Beispiele sowie anhand einer Simulation und einer Prüfstandsaufnahme des NRTC unter Verwendung des entwickelten Reglers.

Kapitel 6 bildet schließlich den Abschluss dieser Arbeit mit einer Zusammenfassung der Ergebnisse sowie einem Ausblick auf potentielle Weiterentwicklungen.

2 Grundlagen

In diesem Kapitel werden die Grundlagen für die mathematische Beschreibung des Luftpfades und dessen Simulation sowie für die Entwicklung des MPC-Lösungsalgorithmus gelegt. Die Beschreibung des Luftpfades basiert auf einer nichtlinearen Zustandsraumdarstellung, dessen Simulation erfolgt unter Verwendung von Einschrittverfahren. Beide Aspekte sind der nichtlinearen Systembeschreibung zuzuordnen und werden in Abschnitt 2.1 besprochen. Für die Berechnung der MPC ist die Lösung von Optimierungsproblemen erforderlich. Die diesbezüglich relevanten Grundlagen der statischen und der dynamischen Optimierung sind Inhalt der Abschnitte 2.2 und 2.3. Die Ausführungen zur statischen Optimierung in Abschnitt 2.2 dienen im Wesentlichen der Vorarbeit und stellen die erforderliche Basis für den in dieser Arbeit entwickelten MPC-Algorithmus dar. Das für diesen Algorithmus zu lösende eigentliche Optimierungsproblem wird durch die Heranziehung einer dynamischen Optimierungsmethode gelöst. Die Beschreibung der dynamischen Optimierung und die Präsentation der numerischen Lösungsansätze erfolgen in Abschnitt 2.2.

2.1 Nichtlineare Systembeschreibung

Zur Beschreibung des Luftpfades in Kapitel 3 wird die *nichtlineare Zustandsraumdarstellung* verwendet. Diese Art der Modellbeschreibung ist die in der Regelungstechnik favorisierte Standardbeschreibung nichtlinearer, zeitkontinuierlicher Zusammenhänge. Diese Darstellung sowie die daraus ableitbare *lineare Zustandsraumdarstellung* sollen im Folgenden formal vorgestellt werden. Im Anschluss wird die numerische Integration solch eines Systems beschrieben. Sowohl zur Simulation des Luftpfades als auch für dessen Prädiktion innerhalb des in Kapitel 4 vorgestellten MPC-Algorithmus sind Differentialgleichungen (DGL) numerisch zu lösen.

2.1.1 Zustandsraumdarstellung

Das (zukünftige) zeitliche Verhalten eines Systems hängt von dessen Historie und von den auf das System einwirkenden externen Einflüssen ab. Die Historie geht aus dem Zustand $\mathbf{x}(t) \in \mathbb{R}^n$ hervor, die externen Einflüsse sind in dem Eingang $\mathbf{u}(t) \in \mathbb{R}^m$ enthalten. Soll beispielsweise das Befüllen eines Wassertanks beschrieben werden, ist der Füllstand des Tanks der Zustand des Systems, welcher die bereits in den Tank eingefüllte Flüssigkeitsmenge – die Historie – widerspiegelt. Der Eingang dieses Systems ist die pro Zeiteinheit zugeführte Flüssigkeitsmenge. Die weitere Entwicklung dieses Systems wird über eine gewöhnliche Vektordifferentialgleichung

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$

mit $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ beschrieben. In regelungstechnischen Anwendungen sind oftmals Größen $\mathbf{y}(t) \in \mathbb{R}^p$ von Interesse, die sich aus dem momentanen Eingang und Zustand ergeben. Diese Größen – die Ausgänge – werden über die Ausgangsgleichung

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t))$$

mit $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$ berechnet. Beide Gleichungen zusammengenommen bilden die nichtlineare Zustandsraumdarstellung

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), & \mathbf{x}(0) := \mathbf{x}_0 \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)). \end{cases} \quad \begin{array}{l} (2.1a) \\ (2.1b) \end{array}$$

In einer allgemeineren Schreibweise sind $\mathbf{f}(\cdot)$ sowie $\mathbf{g}(\cdot)$ zusätzlich von der Zeit abhängig. Um die Notation nicht unnötig zu verkomplizieren, wird darauf verzichtet.¹

Zur Abschätzung der Schrittweite zur numerischen Integration des Luftpfades und zur Gewinnung von Einblicken in dessen Eigenschaften werden Linearisierungen der nichtlinearen Zustandsraumbeschreibung an bestimmten Arbeitspunkten betrachtet. Die dafür verwendete lineare Zustandsraumbeschreibung soll im Weiteren eingeführt werden.

Definition 2.1.1 (Ruhelage). Gegeben sei das System

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)).$$

Dieses System befindet sich in einer Ruhelage, wenn

$$\mathbf{0} = \mathbf{f}(\mathbf{x}_R, \mathbf{u}_R)$$

gilt. Es lässt sich damit die Menge der Ruhelagen über

$$\mathcal{R} := \{\mathbf{x}, \mathbf{u} \mid \mathbf{0} = \mathbf{f}(\mathbf{x}, \mathbf{u})\}$$

definieren.

Um eine lineare Modellbeschreibung zu erhalten, welche die nichtlineare Zustandsraumbeschreibung (2.1) in einer Umgebung $(\mathbf{x}, \mathbf{u}) \in U(\mathbf{x}_R, \mathbf{u}_R)$ einer Ruhelage $(\mathbf{x}_R, \mathbf{u}_R) \in \mathcal{R}$ nach Definition 2.1.1 beschreibt, werden die Originalgrößen $\mathbf{x}(t)$, $\mathbf{u}(t)$ und $\mathbf{y}(t)$ zunächst durch die Abweichungen von der betrachteten Ruhelage ersetzt:

$$\begin{aligned} \delta\mathbf{x}(t) &= \mathbf{x}(t) - \mathbf{x}_R, \\ \delta\mathbf{u}(t) &= \mathbf{u}(t) - \mathbf{u}_R, \\ \delta\mathbf{y}(t) &= \mathbf{y}(t) - \mathbf{y}_R. \end{aligned}$$

¹In der vorliegenden Arbeit werden zeitvariante Systeme nicht betrachtet, wengleich der entwickelte MPC-Lösungsalgorithmus dementsprechend umgesetzt wurde.

Es ergibt sich folgende äquivalente Formulierung:

$$\begin{cases} \delta \dot{\mathbf{x}}(t) = \mathbf{f}(\delta \mathbf{x}(t) + \mathbf{x}_R, \delta \mathbf{u}(t) + \mathbf{u}_R), & \delta \mathbf{x}(0) := \mathbf{x}_0 - \mathbf{x}_R \end{cases} \quad (2.2a)$$

$$\begin{cases} \delta \mathbf{y}(t) = \mathbf{g}(\delta \mathbf{x}(t) + \mathbf{x}_R, \delta \mathbf{u}(t) + \mathbf{u}_R) - \mathbf{y}_R. \end{cases} \quad (2.2b)$$

Die Entwicklung einer Taylor-Reihe für (2.2a) bis zum Glied erster Ordnung ergibt

$$\begin{aligned} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) &= \mathbf{f}(\mathbf{x}_R, \mathbf{u}_R) + \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))}{\partial \mathbf{x}(t)} \right|_{\substack{\mathbf{x}(t)=\mathbf{x}_R \\ \mathbf{u}(t)=\mathbf{u}_R}} \delta \mathbf{x}(t) + \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))}{\partial \mathbf{u}} \right|_{\substack{\mathbf{x}(t)=\mathbf{x}_R \\ \mathbf{u}(t)=\mathbf{u}_R}} \delta \mathbf{u}(t) \\ &+ \mathbf{R}\{\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))\}, \end{aligned} \quad (2.3)$$

wobei $\mathbf{R}\{\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))\}$ das Restglied bzw. den Verfahrensfehler darstellt. Der Zusammenhang für die Ausgangsgleichung (2.2b) kann in analoger Vorgehensweise hergestellt werden. Werden die Restglieder weggelassen und die Stationaritätsbedingung $\mathbf{f}(\mathbf{x}_R, \mathbf{u}_R) = \mathbf{0}$ verwendet, ergibt sich die lineare Zustandsraumdarstellung mit

$$\begin{cases} \delta \dot{\mathbf{x}}(t) = \mathbf{A} \delta \mathbf{x}(t) + \mathbf{B} \delta \mathbf{u}(t) \end{cases} \quad (2.4a)$$

$$\begin{cases} \delta \mathbf{y}(t) = \mathbf{C} \delta \mathbf{x}(t) + \mathbf{D} \delta \mathbf{u}(t). \end{cases} \quad (2.4b)$$

Die Differentiale von $\mathbf{f}(\cdot)$ und $\mathbf{g}(\cdot)$ wurden durch die gebräuchlichen Zustandsraummatrizen ersetzt.² Auf die linearisierte Beschreibung können die Werkzeuge der linearen Systemtheorie angewendet werden, um damit beispielsweise Aussagen für das nichtlineare System in der Nähe des Arbeitspunktes $(\mathbf{x}_R, \mathbf{u}_R)$ zu treffen.

2.1.2 Numerische Integration nichtlinearer Differentialgleichungen

Zur Simulation eines mit (2.1) beschriebenen Modells, ausgehend von einem Anfangswert \mathbf{x}_0 , ist das Anfangswertproblem (AWP)

$$\mathbf{x}(T) = \mathbf{x}_0 + \int_{t_0}^T \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt \quad (2.5)$$

zu lösen. Die numerische Berechnung des Integrals wird als *numerische Integration* oder auch *numerische Quadratur* bezeichnet. Um eine iterative Lösungsvorschrift zu entwickeln, wird das Integral zunächst in $N - 1$ Abschnitte aufgeteilt, wodurch sich die Diskretisierung

$$t_0 < t_1 < \dots < t_k < \dots < \underbrace{t_{N-1}}_{:=T}$$

² \mathbf{A} wird Fundamental- oder auch Systemmatrix genannt, \mathbf{B} ist die Eingangsmatrix, \mathbf{C} die Ausgangsmatrix und \mathbf{D} die Durchgriffsmatrix.

ergibt.³ Damit lässt sich Gleichung (2.5) als Summe von Integralen über

$$\begin{aligned} \mathbf{x}(T) &= \mathbf{x}_0 + \sum_{i=1}^{N-1} \int_{t_{k-1}}^{t_k} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt \\ &= \underbrace{\mathbf{x}_0 + \sum_{i=1}^k \int_{t_{i-1}}^{t_i} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt}_{x(t_k)} + \int_{t_k}^{t_{k+1}} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt + \sum_{i=k+1}^{N-1} \int_{t_{i-1}}^{t_i} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt \\ &\quad \underbrace{\hspace{10em}}_{x(t_{k+1})} \end{aligned}$$

darstellen. Der Wert zu einem Zeitpunkt $x_{k+1} := x(t_{k+1})$ kann aus der vorigen Diskretisierung mit

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \int_{t_k}^{t_{k+1}} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt \quad (2.6)$$

ermittelt werden; dies ist die gesuchte Iterationsvorschrift. Die Fläche unter der Funktion $\mathbf{f}(\cdot)$ zwischen zwei benachbarten Zeitpunkten (t_k, t_{k+1}) wird durch eine Approximation, der sogenannten Quadraturformel

$$\Psi(t_k, h_k, \dots) := \int_{t_k}^{t_k+h_k} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt, \quad h_k := t_{k+1} - t_k, \quad (2.7)$$

berechnet. Die Funktion $\Psi(\cdot)$ wird auch *Verfahrensfunktion* genannt. Die offen gelassenen Argumente hängen von der verwendeten Quadraturformel ab.

Es wird im Grundsatz zwischen Ein- und Mehrschrittverfahren unterschieden. Einschrittverfahren zeichnen sich dadurch aus, dass die Quadraturformel lediglich auf dem zuletzt berechneten Diskretisierungspunkt \mathbf{x}_k beruht. Mehrschrittverfahren nutzen zusätzlich vorausgegangene Ergebnisse $\mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \dots, \mathbf{x}_{k-m}$. Für genauere Informationen zu letzteren sei auf [DR08, S. 423ff] verwiesen.⁴ Grundsätzlich wird zwischen *expliziten* und *impliziten* Berechnungsvorschriften unterschieden. Explizite Verfahren können analytisch gelöst werden, weil die Quadraturformel ausschließlich auf bereits zuvor berechnete Informationen zurückgreift und damit kausal ist. Bei impliziten Quadraturmethoden hingegen ist Kausalität nicht gegeben. Diese sind deswegen iterativ zu lösen, was mehrere Auswertungen von $\mathbf{f}(\cdot)$ erfordert.

³Es werden $N - 1$ Abschnitte gewählt, um die Diskretisierung einheitlich mit der im Rahmen des MPC-Algorithmus vorgenommenen Diskretisierung in Abschnitt 4.3.2 zu halten.

⁴Bekannte Mehrschrittverfahren sind die Verfahren nach *Adams-Bashforth* (explizit), *Adams-Moulton* (implizit) sowie die *Backward Differentiation Formulas* (implizit).

2.1.3 Quadraturformeln

Im Folgenden sollen die in dieser Arbeit verwendeten Quadraturformeln vorgestellt werden. Es handelt sich dabei ausschließlich um Einschrittverfahren, die der Familie der Runge-Kutta-Verfahren (RK) zugeordnet werden können. Für theoretische Betrachtungen im Rahmen der Abschätzung einer Mindestschrittweite zur stabilen Integration des Luftpfades wird das *explizite Euler-Verfahren* (RK1) herangezogen. Die in Kapitel 3 gezeigten Simulationen basieren auf dem stabileren und genaueren (expliziten) *klassischen Runge-Kutta-Verfahren* (RK4). Die auf der ECU umgesetzte MPC verwendet das *implizite Trapezverfahren* (IRK2). Um den Unterschied zwischen den expliziten und impliziten Verfahren zu verdeutlichen, wird zusätzlich das *implizite Euler-Verfahren* (IRK1) vorgestellt.

Die beiden ein numerisches Integrationsverfahren auszeichnenden Eigenschaften sind die Genauigkeit und die Stabilität des Verfahrens. Für die Umsetzung der MPC auf der ECU ist insbesondere die zweite Eigenschaft von Bedeutung, da diese die funktionale Sicherheit des Systems beeinflusst. Entsprechend wird darauf der größere Fokus gelegt. Die Genauigkeit wird unter Angabe der *Konsistenzordnung* genannt und die Stabilität unter Betrachtung des *Stabilitätsgebiets* bewertet. Theoretische Details zu den Verfahren, insbesondere bezüglich ihrer Konvergenz- und Konsistenzordnung sowie zur Stabilität, werden in [DR08, S. 376ff] ausgeführt.

Definition 2.1.2 (Konsistenzordnung). Ein Einschrittverfahren heißt mit dem AWP (2.5) konsistent von der Ordnung p (oder hat die Konsistenzordnung p), falls

$$\|e(t_k, h)\|_2 \leq Ch^p = \mathcal{O}(h^p), \quad h \rightarrow 0$$

mit dem *Konsistenzfehler*

$$e(t_k, \mathbf{x}_k, h) := \frac{1}{h} \left(\int_{t_k}^{t_k+h} \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) dt - \Psi(t_k, h, \dots) \right)$$

für alle Punkte $t_k \in [t_0, T]$ in einer Umgebung des Lösungsgraphen $\{(t, \mathbf{x}(t)) \mid t \in [t_0, T]\}$ von (2.5) gilt. [DR08, S. 306]

Die in Definition 2.1.2 eingeführte Konsistenzordnung ist ein Maß für die Größe des lokalen Abbruchfehlers, mithin des Fehlers, der bei der Durchführung eines Integrationsschrittes gemacht wird. Bei Verfahren mit der Konsistenzordnung $p = 1$ halbiert sich der Fehler bei einer Halbierung der Schrittweite. Hat ein Verfahren beispielsweise die Konsistenzordnung $p = 2$, wird hingegen der Fehler bei einer Halbierung der Schrittweite geviertelt. Bei gleicher Schrittweite sind Verfahren höherer Konsistenzordnung i. d. R. genauer, bzw. es sind bei gleicher Genauigkeit weniger Funktionsauswertungen erforderlich, was einen geringeren Rechenaufwand impliziert.⁵

⁵Bei den im Rahmen dieser Arbeit betrachteten Verfahren ist der genannte Zusammenhang zwischen

Zur Definition des Stabilitätsgebiets wird das lineare, autonome System

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{x}(0) = x_0, \quad \lambda_i < 0$$

mit ausschließlich stabilen Eigenwerten λ_i betrachtet. Für dieses System kann eine Diagonalisierung

$$\dot{\tilde{\mathbf{x}}}(t) = \mathbf{\Lambda}\tilde{\mathbf{x}}(t), \quad \tilde{\mathbf{x}}(0) = \tilde{\mathbf{x}}_0$$

mit

$$\mathbf{\Lambda} := \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}, \quad \tilde{\mathbf{x}}(t) := \mathbf{T}^{-1}\mathbf{x}(t)$$

unter Verwendung einer Transformationsmatrix $\mathbf{T} \in \mathbb{R}^{n \times n}$ gefunden werden. Durch die Diagonalisierung sind die Gleichungen entkoppelt und können damit isoliert betrachtet werden. Bei hinreichend langer Integrationszeit unter Verwendung eines stabilen Integrationsverfahrens wird jede der entkoppelten Gleichungen für beliebige Anfangswerte gegen Null konvergieren. Die numerische Integration eines Zustandes \tilde{x}_i kann als Zeitdiskretisierung verstanden werden, welche über

$$\tilde{x}_{i,k+1} = \tilde{x}_{i,k} + \Psi(0, h, \dots) =: g(z_i)\tilde{x}_k, \quad z := h\lambda_i \quad (2.8)$$

gegeben ist. Auf der rechten Seite wird die verfahrensabhängige Stabilitätsfunktion $g(z)$ definiert. Gleichung (2.8) kann als *Folge* aufgefasst werden. Diese konvergiert, wenn $|g(z_i)| < 1$ für $z_i \in \mathbb{C}$ gilt, denn in diesem Fall wird ein $\tilde{x}_{i,k}$ auf ein betragsmäßig kleineres $\tilde{x}_{i,k+1}$ abgebildet. Sämtliche diskreten Eigenwerte z_i müssen damit in dem Stabilitätsgebiet

$$B := \{z \in \mathbb{C} \mid |g(z_i)| < 1\}$$

liegen; es muss gelten:

$$h\lambda_i \in B.$$

Aussagen über die Stabilität nichtlinearer Systeme können anhand des Stabilitätsgebiets nicht getroffen werden. Jedoch ist über eine Linearisierung (2.4) eine tendenzielle Bewertung in der Nähe der betrachteten Ruhelage möglich.

Konsistenzordnung und Genauigkeit gegeben. Implizite Verfahren können in dieser Hinsicht nicht mit expliziten verglichen werden.

Explizites Euler-Verfahren – RK1

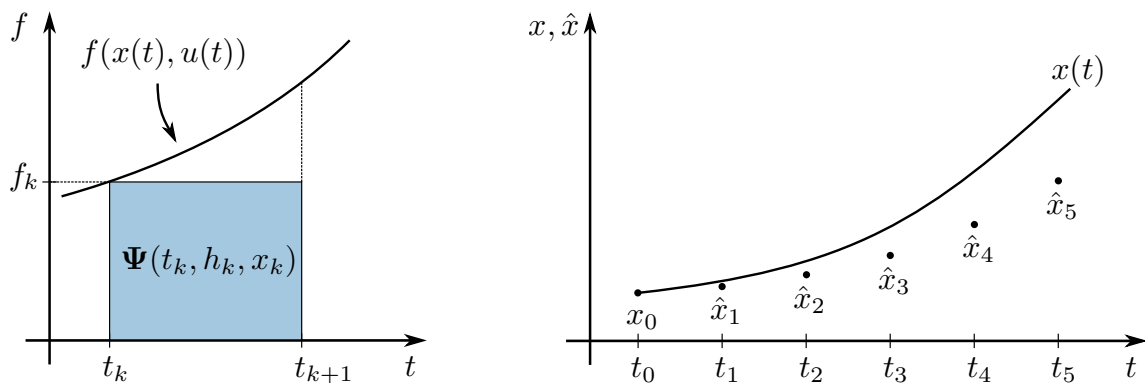
Das RK1-Verfahren⁶ verwendet die Verfahrensfunktion

$$\Psi(t_k, h_k, \mathbf{x}_k) = h_k \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k))$$

mit der Schrittweite h_k . Die Rekursionsformel (2.6) ergibt sich entsprechend zu

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \Psi(t_k, h_k, \mathbf{x}_k) \\ &= \mathbf{x}_k + h_k \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k)). \end{aligned} \quad (2.9)$$

Eine geometrische Interpretation der Approximation ist in Abb. 2.1a dargestellt. Die Fläche der Funktion $f(\cdot)$ wird durch ein Rechteck mit der Breite $h_k := t_{k+1} - t_k$ angenähert. Die Höhe des Rechtecks ist der Funktionswert an der Stelle x_k . Die Fläche, die zwischen der Kurve und der Approximation liegt, stellt den Verfahrens- bzw. Abbruchfehler dar. In diesem eindimensionalen Fall würde die numerische Integration die Fläche und damit den Wert des Zustandes unterschätzen. Qualitativ ist dies in Abb. 2.1b zu sehen.



(a) Approximationsfehler durch die Näherung über ein Rechteck.

(b) Auseinanderdriften des tatsächlichen x und des berechneten Zustandes \hat{x} .

Abb. 2.1: Qualitative Darstellung der Quadratur mit dem expliziten Euler-Verfahren für den eindimensionalen Fall.

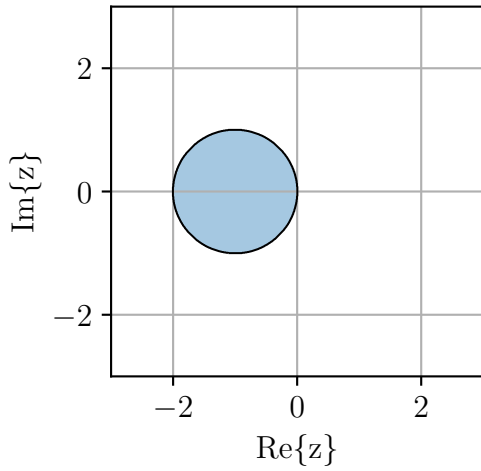
Das RK1-Verfahren hat eine Konsistenzordnung von $p = 1$. Die Stabilitätsfunktion lautet

$$g(z) = 1 + z. \quad (2.10)$$

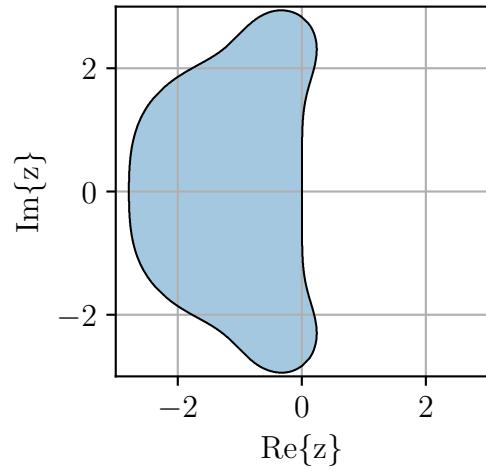
Das zugehörige Stabilitätsgebiet ist in Abb. 2.2a dargestellt. Sämtliche Argumente z , welche im blau hinterlegten Bereich liegen, führen zu einer stabilen Integration. Wird beispielsweise ein lineares System mit einem reellwertigen, stabilen Eigenwert λ betrachtet, welches mit einer zu großen Schrittweite h integriert wird, dann gilt $z = h\lambda < -2$. Der diskrete Ei-

⁶Das explizite Euler-Verfahren ist auch unter dem Namen *Euler-Cauchy-Verfahren* bekannt.

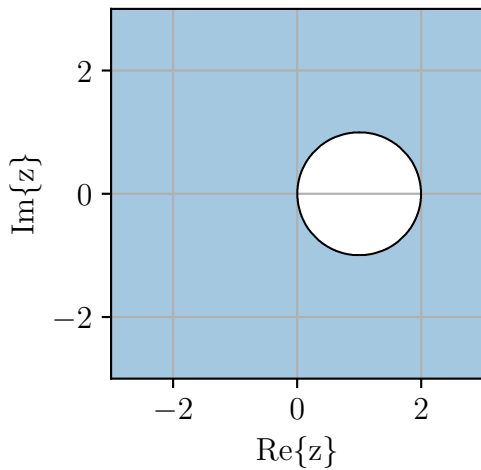
genwert liegt damit links außerhalb des dargestellten Gebietes. Die rechte Halbebene wird bei instabilen Eigenwerten betreten.



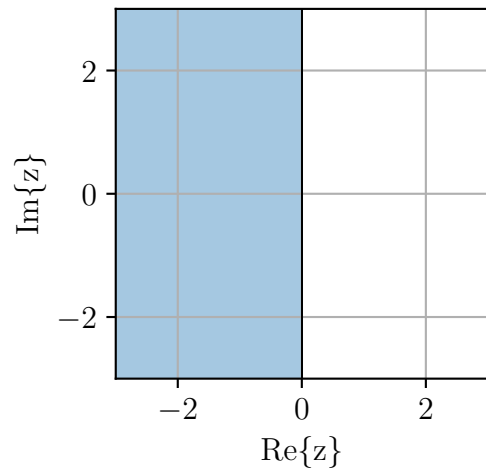
(a) Stabilitätsgebiet des RK1-Verfahrens.



(b) Stabilitätsgebiet des RK4-Verfahrens.



(c) Stabilitätsgebiet des IRK1-Verfahrens.



(d) Stabilitätsgebiet des IRK2-Verfahrens.

Abb. 2.2: Stabilitätsgebiete der betrachteten Integrationsverfahren. Die eingezeichneten Konturen sind nicht Teil der jeweiligen Gebiete; dort gilt $|g(z)| = 1$.

Klassisches Runge-Kutta-Verfahren – RK4

Die Verfahrensfunktion des RK4-Verfahrens ist über

$$\Psi(t_k, h_k, \mathbf{x}_k) = \frac{h_k}{6} (\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \quad (2.11)$$

mit

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k)), \\ \mathbf{k}_2 &= \mathbf{f}\left(\mathbf{x}_k + \frac{h_k}{2}\mathbf{k}_1, \mathbf{u}\left(t_k + \frac{h_k}{2}\right)\right), \\ \mathbf{k}_3 &= \mathbf{f}\left(\mathbf{x}_k + \frac{h_k}{2}\mathbf{k}_2, \mathbf{u}\left(t_k + \frac{h_k}{2}\right)\right), \\ \mathbf{k}_4 &= \mathbf{f}(\mathbf{x}_k + h_k\mathbf{k}_3, \mathbf{u}(t_k + h_k)) \end{aligned}$$

gegeben. Sie hat damit die gleichen Argumente wie das RK1-Verfahren. In einer anschaulichen Betrachtung geschieht die Approximation der DGL über ein Polynom dritter Ordnung. Das RK1-Verfahren approximiert die Funktion in Abb. 2.1a hingegen durch ein Polynom der Ordnung Null.

Das Verfahren hat die Konsistenzordnung $p = 4$ und die Stabilitätsfunktion

$$g(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}.$$

In Abb. 2.2b ist das Stabilitätsgebiet des RK4-Verfahrens dargestellt. Es ist ersichtlich, dass das Verfahren im Vergleich zum RK1-Verfahren (vgl. Abb. 2.2a) neben einer höheren Konsistenzordnung zusätzlich stabiler ist. Das Stabilitätsgebiet des RK1-Verfahrens wird vollständig von dem des RK4-Verfahrens umschlossen. Die reelle Achse wird an den Stellen $(-2,785; 0)$ von der Kontur des Gebiets geschnitten. Zur stabilen Integration eines stabilen, reellwertigen Systems kann die Schrittweite damit ca. 40 % größer als beim RK1-Verfahren gewählt werden.

Implizites Euler-Verfahren – IRK1

Anschaulich wird das IRK1-Verfahren genau so berechnet, wie das RK1-Verfahren, jedoch mit dem Unterschied, dass die Höhe des Rechtecks in Abb. 2.1a dem Funktionswert an der Stelle x_{k+1} entspricht. Die Verfahrensfunktion ist identisch, sie geht allerdings mit anderen Argumenten in die Rekursionsformel ein:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \Psi(t_{k+1}, h_k, \mathbf{x}_{k+1}) \\ &= \mathbf{x}_k + h_k \mathbf{f}(\mathbf{x}_{k+1}, \mathbf{u}(t_{k+1})). \end{aligned}$$

Eine numerische Berechnung ist iterativ über die Fixpunktiteration

$$\mathbf{x}_{k+1}^{(m+1)} = \mathbf{x}_k + h_k \mathbf{f}(\mathbf{x}_{k+1}^{(m)}, \mathbf{u}(t_{k+1})), \quad \mathbf{x}_{k+1}^{(0)} := \mathbf{x}_k, \quad m = 0, 1, \dots$$

möglich, wenn die rechte Seite eine *Kontraktion* ist (vgl. Abschnitt A.1.1). Eine Berechnung mit dem Newton-Verfahren (vgl. Abschnitt A.1.2) ist robuster, allerdings erfordert das Verfahren die Berechnung der Jacobi-Matrix des Gleichungssystems.

Das IRK1-Verfahren hat wie das RK1-Verfahren eine Konsistenzordnung von $p = 1$. Der Unterschied zum RK1-Verfahren wird durch die Betrachtung des Stabilitätsgebiets in Abb. 2.2c sichtbar. Das Stabilitätsgebiet des IRK1-Verfahrens umfasst die gesamte linke Halbebene; das Verfahren ist *A-stabil*. Damit wird ein stabiles lineares System mit einer beliebig großen Integrationsschrittweite in stabiler Weise integriert. Die zugehörige Stabilitätsfunktion ist über

$$g(z) = \frac{1}{1-z}$$

gegeben.

Implizites Trapezverfahren – IRK2

Das IRK2-Verfahren verwendet zwei Stützstellen und approximiert die DGL auf diese Weise anschaulich durch ein Trapez bzw. ein Polynom erster Ordnung. Die zugehörige Rekursionsformel ist über

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \Psi(t_{k+1}, h_k, \mathbf{x}_k, \mathbf{x}_{k+1}) \\ &= \mathbf{x}_k + \frac{h_k}{2} (\mathbf{f}(\mathbf{x}_{k+1}, \mathbf{u}(t_{k+1})) + \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k))) \end{aligned}$$

gegeben. Analog zum IRK1-Verfahren lässt sich die Gleichung ebenfalls über eine Fixpunktiteration

$$\mathbf{x}_{k+1}^{\langle m+1 \rangle} = \mathbf{x}_k + \frac{h_k}{2} (\mathbf{f}(\mathbf{x}_{k+1}^{\langle m \rangle}, \mathbf{u}(t_{k+1})) + \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k))), \quad \mathbf{x}_{k+1}^{\langle 0 \rangle} := \mathbf{x}_k, \quad m = 0, 1, \dots$$

lösen.⁷ Bemerkenswert ist, dass zur iterativen Berechnung des IRK2-Verfahrens bei gleicher Iterationsanzahl insgesamt eine Funktionsauswertung mehr erforderlich ist, als zur iterativen Lösung des IRK1-Verfahrens. Das bedeutet: Je mehr Iterationen zur Berechnung der Verfahren erforderlich sind, desto mehr gleichen sich die Rechenaufwände an. Die Konsistenzordnung des Verfahrens liegt bei $p = 2$.

Im Rahmen der zur temperaturgeführten Motorregelung in Kapitel 5 verwendeten MPC wird das Gleichungssystem über ein Abstiegsverfahren (vgl. Abschnitt 2.2.1) gelöst:

$$\begin{aligned} \mathbf{x}_{k+1}^{\langle m+1 \rangle} &= \mathbf{x}_{k+1}^{\langle m \rangle} - \alpha^{\langle m \rangle} \mathbf{g}^{\langle m \rangle}, \quad \mathbf{x}_{k+1}^{\langle 0 \rangle} := \mathbf{x}_k, \quad \alpha^{\langle m \rangle} > 0, \\ \mathbf{g}_k^{\langle m \rangle} &:= \mathbf{x}_{k+1}^{\langle m \rangle} - \mathbf{x}_k - \frac{h_k}{2} (\mathbf{f}(\mathbf{x}_{k+1}^{\langle m \rangle}, \mathbf{u}(t_{k+1})) + \mathbf{f}(\mathbf{x}_k, \mathbf{u}(t_k))). \end{aligned} \tag{2.12}$$

⁷Eine Lösung über das Newton-Verfahren (vgl. Abschnitt A.1.2) ist ebenfalls möglich.

Die Schrittweite wird unter Verwendung von (2.19), (2.21a), und (2.22b) über

$$\alpha^{(m)} = P(\tilde{\alpha}^{(m)}, \alpha_{\min}, \alpha_{\max}),$$

$$\tilde{\alpha}^{(m)} = \begin{cases} 0,1 & \text{für } m = 0 \\ \frac{(\Delta \mathbf{x}_{k+1}^{(m)})^\top \Delta \mathbf{x}_{k+1}^{(m)}}{(\Delta \mathbf{x}_{k+1}^{(m)})^\top \Delta \mathbf{g}_{k+1}^{(m)}} & \text{für } m > 0 \wedge \frac{1}{2} \|\mathbf{g}_k^{(m)}\|_2 \leq \frac{1}{2} \|\mathbf{g}_k^{(m-1)}\|_2 + c_1 \Delta \mathbf{x}_{k+1}^{(m)} \mathbf{g}_k^{(m-1)} \\ \frac{\alpha^{(m-1)}}{2} & \text{für } m > 0 \wedge \frac{1}{2} \|\mathbf{g}_k^{(m)}\|_2 < 1,1 \frac{1}{2} \|\mathbf{g}_k^{(m-1)}\|_2 \\ \alpha_{\min} & \text{sonst} \end{cases}$$

mit

$$\Delta \mathbf{x}_{k+1}^{(m)} := \mathbf{x}_{k+1}^{(m)} - \mathbf{x}_{k+1}^{(m-1)},$$

$$\Delta \mathbf{g}_k^{(m)} := \mathbf{g}_k^{(m)} - \mathbf{g}_k^{(m-1)}$$

ermittelt.⁸ Die Parameter werden zu $c_1 := 10^{-3}$, $\alpha_{\min} := 10^{-6}$ und $\alpha_{\max} := 0,9$ gewählt.

Die Stabilitätsfunktion des Verfahrens ist über

$$g(z) = \frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z}$$

gegeben und das zugehörige Stabilitätsgebiet ist in Abb. 2.2d dargestellt. Es ist zu sehen, dass das IRK2-Verfahren wie das IRK1-Verfahren A-stabil ist. Allerdings ist das Stabilitätsgebiet kleiner als beim IRK1-Verfahren. Die Konvergenzgeschwindigkeit der Fixpunktiteration ist beim IRK1-Verfahren für betragsgroße Werte $h\lambda$ höher als beim IRK2-Verfahren. Für unendlich große Werte konvergiert $\lim_{z \rightarrow -\infty} |g(z)| = 1$ gegen die Stabilitätsgrenze, wohingegen das IRK1-Verfahren gegen Null konvergiert. Diese Eigenschaft hat Auswirkungen auf die Konditionierung des Iterationsschemas (2.12).⁹

2.2 Statische Optimierung

Ziel der statischen Optimierung ist es, diejenigen Argumente zu ermitteln, die eine Funktion $f(\mathbf{x})$ minimieren.¹⁰ Es werden die optimalen Argumente \mathbf{x}^* gesucht, für die

$$f(\mathbf{x}^*) \leq f(\mathbf{x}), \quad \forall \mathbf{x} \in \mathcal{D}$$

⁸Die Armijo-Bedingung (2.19) wird formal nicht korrekt angewendet, da $\frac{1}{2} \|\mathbf{g}_k^{(m)}\|_2$ nicht die Basisfunktion von $\mathbf{g}_k^{(m)}$ ist; lediglich auf den skalaren Fall trifft dies zu. In numerischen Untersuchungen im Rahmen dieser Arbeit hat sich diese Vorgehensweise dennoch als zielführend herausgestellt.

⁹Auswirkungen sind für Schrittweiten, die mehrere Größenordnungen über der Maximalschrittweite des RK1-Verfahrens für das betrachtete System liegen, zu beobachten.

¹⁰Typischerweise wird nach einem Minimum gesucht. Eine Formulierung hinsichtlich einer Maximierung ist ebenso möglich.

gilt. Die Menge \mathcal{D} enthält die zulässigen Argumente bzw. Lösungen. Die Argumente \mathbf{x} werden fortan als *Optimierungsvariablen* bezeichnet. Ein Optimierungsproblem hat allgemein folgendes Erscheinungsbild:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{u. B. v.} \quad & c_i(\mathbf{x}) = 0, \quad i \in \mathcal{E}, \\ & c_i(\mathbf{x}) \leq 0, \quad i \in \mathcal{I}. \end{aligned} \tag{2.13a}$$

Dieses Problem beschreibt die Minimierung der *Gütefunktion* $f : \mathbb{R}^n \mapsto \mathbb{R}$ in Abhängigkeit von der Optimierungsvariablen $\mathbf{x} \in \mathbb{R}^n$ und unter Berücksichtigung von (u. B. v.) den Nebenbedingungen $\mathbf{c}_i : \mathbb{R}^n \mapsto \mathbb{R}$. Die Nebenbedingungen mit $i \in \mathcal{E}$ werden Gleichungsbeschränkungen (engl.: *equality constraints*) und die Nebenbedingungen mit $i \in \mathcal{I}$ entsprechend Ungleichungsbeschränkungen (engl.: *inequality constraints*) genannt. Die Menge \mathcal{D} zulässiger Lösungen geht demnach aus den Nebenbedingungen hervor. In den nachfolgenden Ausführungen wird an einigen Stellen die kompaktere Formulierung

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{u. B. v.} \quad & \mathbf{c}_{\mathcal{E}}(\mathbf{x}) = \mathbf{0}, \\ & \mathbf{c}_{\mathcal{I}}(\mathbf{x}) \leq \mathbf{0} \end{aligned} \tag{2.13b}$$

verwendet. Es sei angemerkt, dass stets $|\mathcal{E}| < n$ zu gelten hat, um eine sinnvolle Problemstellung zu erhalten. Andernfalls ergibt sich die Lösung direkt aus dem Schnittpunkt der Nebenbedingungen – sofern diese linear unabhängig sind und eine gemeinsame Lösung besitzen. Im Weiteren wird davon ausgegangen, dass dieses Optimierungsproblem lösbar ist. Dies beinhaltet den zuvor genannten Aspekt, sowie, dass $f(\mathbf{x})$ ein begrenztes Minimum besitzt, stetig ist und dass sich die Nebenbedingungen nicht widersprechen. Es wird von einem *wohlgestellten* Optimierungsproblem ausgegangen.

Zur Lösung von nichtlinearen Optimierungsproblemen dieser Art existieren verschiedene Verfahren. Grundsätzlich kann zwischen gradientenbasierten (bspw.: *SQP-Verfahren* [BT95], *Innere-Punkt-Verfahren* [FGW02], *Erweiterte Lagrange-Methode* [Ber96]) und gradientenfreien Verfahren (bspw.: *Nelder-Mead-Verfahren* [NM95], *Evolutionäre Algorithmen* [Vik17]) unterschieden werden. Es existieren jedoch auch Verfahren, die eine Mischung aus beiden Klassen darstellen (bspw.: *Basin-Hopping* [WD97]).

Das eingesetzte Lösungsverfahren für den modellprädiktiven Regler (vgl. Kapitel 4) basiert unter anderem auf der *erweiterten Lagrange-Methode*. Diese Methode spielt in den folgenden Abschnitten die zentrale Rolle und soll stückweise erarbeitet werden. Dazu wird zunächst in Abschnitt 2.2.1 das Abstiegsverfahren eingeführt, die damit im engen Zusammenhang stehende Liniensuche in Abschnitt 2.2.2. Mit dem Abstiegsverfahren lässt sich ein Optimierungsproblem ohne Nebenbedingungen lösen. In dem darauffolgenden Abschnitt 2.2.3 wird mit der *Gradientenprojektion* eine Methode besprochen, die eine bestimmte Art von Nebenbedingung ohne gesonderte iterative Behandlung berücksichtigen kann. Diese Methode ist nicht Teil der Lagrange-Methode, sie wird jedoch für den Lösungsal-

gorithmus des modellprädiktiven Reglers verwendet und daher zusätzlich besprochen. Im letzten Abschnitt 2.2.4 wird die erweiterte Lagrange-Methode zur Lösung des allgemeinen Optimierungsproblems eingeführt.¹¹

2.2.1 Abstiegsverfahren

Ein Optimierungsproblem ohne Nebenbedingungen hat die Gestalt

$$\min_{\mathbf{x}} f(\mathbf{x}). \quad (2.14)$$

Zur numerischen Lösung können *Abstiegsverfahren* verwendet werden. Diese nähern sich iterativ der gesuchten Lösung \mathbf{x}^* des Optimierungsproblems an. Initial wird mit einem Schätzwert $\mathbf{x}^{(0)}$ gestartet, welcher in einem iterativen Vorgehen sukzessive verbessert wird:

$$\|\mathbf{x}^{(i+1)} - \mathbf{x}^*\|_2 \leq \|\mathbf{x}^{(i)} - \mathbf{x}^*\|_2, \quad i = 0, 1, \dots,$$

wobei der Funktionswert $f(\cdot)$ von Iteration zu Iteration kleiner wird. Qualitativ ist die iterative Näherung in Abb. 2.3 dargestellt.

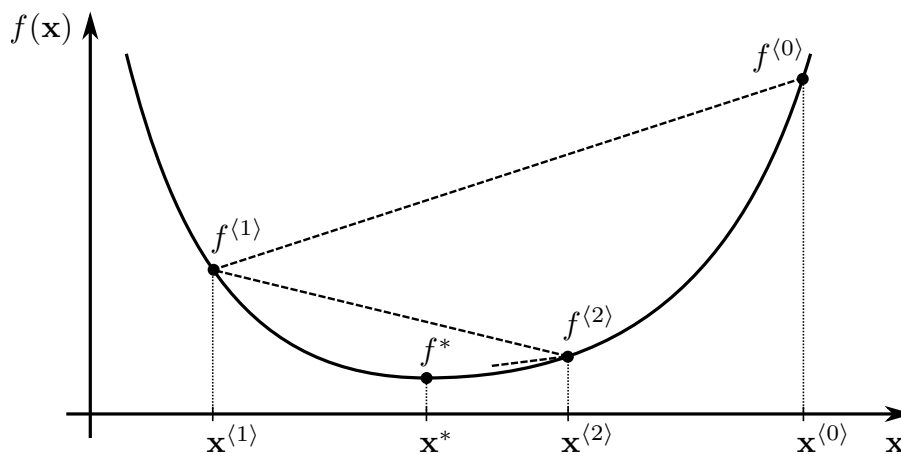


Abb. 2.3: Qualitative Darstellung des Abstiegsverfahrens.

Bevor auf das Verfahren eingegangen wird, soll zunächst die Frage beantwortet werden, unter welchen Bedingungen eine Lösung optimal ist. Aus der Kurvendiskussion ist bekannt, dass eine Extremstelle, d.h. ein Minimum, ein Maximum oder ein Sattelpunkt, über das Verschwinden der ersten Ableitung gekennzeichnet ist. Diese Bedingung 1. Ordnung ist in dem folgenden Satz zusammengefasst:

¹¹Es sei angemerkt, dass die Gradientenprojektion nicht Teil der erweiterten Lagrange-Methode ist.

Satz 2.2.1 (Notwendige Bedingung 1. Ordnung). Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}$ eine stetig differenzierbare Funktion und \mathbf{x}^* ein Minimum. Dann verschwindet der Gradient in diesem Punkt:

$$\nabla f(\mathbf{x}^*) = \mathbf{0}. \quad (2.15)$$

Für eine konvexe Gütefunktion ist diese Bedingung hinreichend. Konvexe Funktionen zeichnen sich dadurch aus, dass sie genau eine Extremstelle besitzen und diese eine Extremstelle ein Minimum ist.¹² Es sei diesbezüglich angemerkt, dass eine konkave Funktion durch Negation in eine konvexe Funktion umgewandelt werden kann. Soll entsprechend eine Funktion maximiert werden, ist dies in äquivalenter Weise über eine Minimierung der Negation zu erreichen. Handelt es sich bei der zu minimierenden Funktion hingegen um eine beliebige nichtlineare Funktion, wird zur Qualifikation eines Minimums die hinreichende Bedingung 2. Ordnung benötigt:

Satz 2.2.2 (Hinreichende Bedingung 2. Ordnung). Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}$ eine zweimal stetig differenzierbare Funktion und \mathbf{x}^* ein Minimum. Dann muss

$$\nabla^2 f(\mathbf{x}^*) \succ \mathbf{0} \quad (2.16)$$

gelten.

Es handelt sich demnach um ein Minimum, wenn die Hesse-Matrix $\nabla^2 f(\mathbf{x}^*)$ positiv definit ist. Bei der Existenz mehrerer Minima garantieren die Bedingungen 1. und 2. Ordnung keine *globale* Optimalität. Es ist möglich, dass ein anderes, besseres Minimum existiert. Entsprechend garantieren die Bedingungen lediglich *lokale* Optimalität. Veranschaulicht ist dieser Zusammenhang in Abb. 2.4. In der Abbildung ist zu sehen, dass neben dem globalen Minimum noch zwei weitere lokale Minima existieren. Diese erfüllen ebenfalls beide Optimalitätsbedingungen. Sattelpunkte werden hingegen durch die Bedingung 2. Ordnung

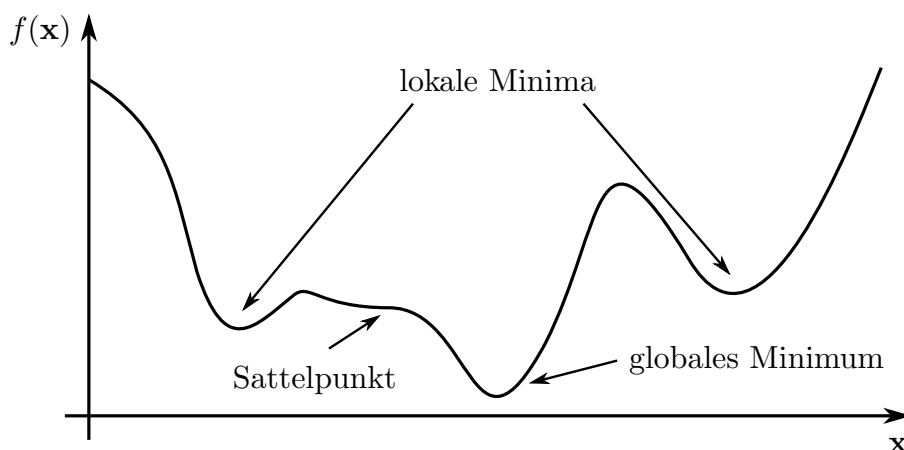


Abb. 2.4: Extremstellen im Rahmen einer Minimierung.

¹²Die in Abb. 2.3 dargestellte Funktion ist eine konvexe Funktion.

ausgeschlossen. Die erste Ableitung ist dort zwar Null, jedoch weist die Kurve keine Krümmung auf.

Um diese mit einzuschließen, kann die in Satz 2.2.3 definierte notwendige Bedingung 2. Ordnung zweckmäßiger sein. Es sind Problemstellungen denkbar, in denen das Minimum gleichzeitig ein Sattelpunkt ist.

Satz 2.2.3 (Notwendige Bedingung 2. Ordnung). *Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}$ eine zweimal stetig differenzierbare Funktion und \mathbf{x}^* ein Minimum oder ein Sattelpunkt. Dann muss*

$$\nabla^2 f(\mathbf{x}^*) \succcurlyeq \mathbf{0}$$

gelten.

Die Iterationsvorschrift für solch ein Verfahren ist über

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \alpha^{(i)} \mathbf{d}^{(i)}, \quad \alpha^{(i)} > 0 \quad (2.17)$$

gegeben. Dabei bezeichnet $\mathbf{x}^{(i)}$ die Lösung des aktuellen Iterationsschrittes und $\mathbf{x}^{(i+1)}$ die (verbesserte) Lösung im nächsten Iterationsschritt. Die Abstiegsrichtung (auch Suchrichtung genannt) ist $\mathbf{d}^{(i)}$. Die skalare Variable $\alpha^{(i)}$ ist die sog. *Schrittweite*, welche typischerweise im Rahmen einer separat durchgeführten *Liniensuche* (siehe Abschnitt 2.2.2) ermittelt wird. Diese gibt an, wie weit der Abstiegsrichtung gefolgt wird. Das Abstiegsverfahren folgt dem Vorgehen in Algorithmus 2.2.1, welches unter Verwendung von Abb. 2.3 nachvollzogen werden kann. Die Suchrichtung ist in der Darstellung ein Skalar, dessen Wert beispielsweise bei der Anwendung eines Gradientenverfahrens der negativen Tangentensteigung an der Stelle $\mathbf{x}^{(i)}$ entspricht.

Algorithmus 2.2.1 (Abstiegsverfahren).

Initialisierung

- Startpunkt festlegen: $\mathbf{x}^{(0)} := \mathbf{x}_{\text{start}}$

for i **in** $0, 1, \dots$ **do**

- Suchrichtung $\mathbf{d}^{(i)}$ festlegen
- Schrittweite $\alpha^{(i)} > 0$ ermitteln
- Optimierungsvariable aktualisieren

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \alpha^{(i)} \mathbf{d}^{(i)}$$

if Konvergenztest für (2.14) erfüllt **then**

- **stop** Beende den Algorithmus

end if

end for

Es sei angemerkt, dass bei dieser Art von Algorithmus der Startwert maßgeblich dafür ist, zu welchem Minimum der Algorithmus konvergiert (vgl. Abb. 2.4). Das Auffinden des globalen Minimums ist damit nicht garantiert.

Die Geschwindigkeit des Abstiegsverfahrens im Sinne der benötigten Rechenleistung hängt davon ab, auf welche Weise die Abstiegsrichtung und die Schrittweite ermittelt werden. Sowohl für die Abstiegsrichtung als auch für die Liniensuche gibt es verschiedene Ansätze, die abhängig von der Problemstellung mehr oder weniger geeignet sind [NW06], [Pap15], [RHG13].

Es gibt Optimierungsverfahren, die die Schritte *Abstiegsrichtung* und *Liniensuche* simultan durchführen: die sogenannten *Trust-Region-Verfahren*. Diese Verfahren approximieren das eigentliche Problem durch eine Näherung, welche zur Ermittlung eines geeigneten Lösungskandidaten herangezogen wird [NW06, S. 66ff].¹³ Auf diese Verfahrensklasse wird nicht weiter eingegangen.

Um die Notation im Folgenden übersichtlich zu halten, werden die Kurzschreibweisen

$$\begin{aligned}f^{(i)} &:= f(\mathbf{x}^{(i)}(t)) \\ \mathbf{g}^{(i)} &:= \nabla f(\mathbf{x}^{(i)}(t))\end{aligned}$$

eingeführt.

Abstiegsrichtung

Die Abstiegsrichtung $\mathbf{d}^{(i)}$ wird bei dem Abstiegsverfahren derart gewählt, dass die Anwendung der Iterationsvorschrift (2.17) zu einer Verkleinerung der Gütefunktion führt – unter der Voraussetzung, dass die Schrittweite geeignet gewählt wird. Eine gültige Abstiegsrichtung gehorcht entsprechend der Bedingung

$$\mathbf{g}^{(i)} \mathbf{d}^{(i)} < 0.$$

In einer bildlichen Vorstellung zeigen der Gradient und die Abstiegsrichtung im Grundsatz in entgegengesetzte Richtung. Beim Gradientenverfahren wird als Suchrichtung der negative Gradient gewählt:

$$\mathbf{d}^{(i)} = -\mathbf{g}^{(i)}.$$

In diesem Fall zeigen die Vektoren exakt in entgegengesetzte Richtung und (2.18a) ist damit stets erfüllt. Im Rahmen dieser Arbeit wird ausschließlich von dieser Abstiegsrichtung Gebrauch gemacht.

¹³Das Newton-Verfahren ohne separate Liniensuche ist ein Trust-Region-Verfahren.

Zeigt die Abstiegsrichtung exakt in Richtung des Minimums, ist mit einer Abstiegsiteration in Kombination mit einer optimal gewählten Schrittweite eine Minimierung eines beliebigen Optimierungsproblems möglich. Zur Verdeutlichung kann Abb. 2.3 als Minimierung einer skalaren Optimierungsvariablen aufgefasst werden: Die aktuelle Optimierungsvariable sei größer als die optimale Lösung. In diesem Fall ist jede negative Abstiegsrichtung in dem zuvor genannten Sinne optimal, weil stets eine Schrittweite gefunden werden kann, die diese Funktion minimiert.

Bei mehrdimensionalen Problemen gestaltet sich das Auffinden einer optimalen Abstiegsrichtung schwieriger. Dazu sei auf Abb. 2.6 verwiesen. Für diese Art von Problemstellung liefert der negative Gradient keine optimale Suchrichtung und es sind auch mit optimaler Schrittweite mehrere Iterationen notwendig.¹⁴ Die Betrachtung macht jedoch auch klar, dass bessere Abstiegsrichtungen als die Richtung des negativen Gradienten existieren. Der Erfolg eines Optimierungsverfahrens hängt damit von der Wahl der Abstiegsrichtung ab. Methoden, die eine bessere Suchrichtung verwenden, sind beispielsweise das *Newton-Verfahren* [BV09, S. 484] (vgl. Abschnitt A.1.2), das *Quasi-Newton-Verfahren* [NW06, S. 135ff] oder auch das *konjugierte Gradientenverfahren* [NW06, S. 121ff]. Diese verbesserten Suchrichtungen sind jedoch mit einem höheren Rechenaufwand verbunden und werden in dieser Arbeit nicht herangezogen.

2.2.2 Liniensuche

Die Liniensuche dient zur Ermittlung der optimalen bzw. einer akzeptablen Schrittweite $\alpha^{(i)}$ für eine gewählte Suchrichtung $\mathbf{d}^{(i)}$. Die optimale Schrittweite ist über das Minimum der univariaten Funktion

$$\phi(\alpha) = f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}), \quad \alpha > 0 \quad (2.18a)$$

mit

$$\alpha^{(i)} = \arg \min_{\alpha > 0} \phi(\alpha) \quad (2.18b)$$

gegeben. Dies ist die sogenannte *exakte Liniensuche* [BV09, S. 464]. Im Regelfall lohnt sich eine exakte Liniensuche nicht, da die Suchrichtung, wie bereits gezeigt, typischerweise nicht exakt in Richtung des Minimums weist und deshalb selbst bei exakter Liniensuche weitere Iterationen des Abstiegsverfahrens erforderlich sind. Der Fortschritt pro Iteration nimmt mit zunehmender Konvergenz ab, weshalb insgesamt unverhältnismäßig viele Auswertungen von $f(\cdot)$ erforderlich wären. Eine Aktualisierung der Abstiegsrichtung und eine erneute Liniensuche minimieren die Gütefunktion i. d. R. schneller. Um mit möglichst wenig Aufwand eine akzeptable Schrittweite zu erhalten, werden inexakte Methoden verwendet, die eine genügend große Reduktion von $f(\cdot)$ sicherstellen [NW06, S. 31]. Es ist im Kontext

¹⁴Der negative Gradient steht orthogonal zu den eingezeichneten Höhenlinien.

der Problemstellung zu bewerten, wie *teuer* jeweils die Berechnung der Gütefunktion und der Abstiegsrichtung ist, und darauf basierend zu entscheiden, wie viele Iterationen in der Liniensuche sich positiv auf die Gesamtlaufzeit auswirken.

Armijo-Goldstein-Bedingungen

Die Qualität einer Liniensuche kann mit den *Armijo-Goldstein-Bedingungen* bewertet werden. Sind diese erfüllt, kann die Liniensuche beendet werden und eine akzeptable, inexacte Lösung ist gefunden. Es handelt sich dabei um zwei Bedingungen: eine zur Begrenzung der maximalen Schrittweite und eine zur Begrenzung der minimalen Schrittweite. Ist die Schrittweite zu groß, konvergiert das Iterationsverfahren möglicherweise nicht. Ist die Schrittweite zu klein, wird pro Abstiegsiteration zu wenig Fortschritt erzielt, was sich negativ auf die benötigte Rechenzeit auswirkt.

Die erste der beiden Bedingungen wird *Armijo-Bedingung* genannt und ist wie folgt definiert:

$$f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}) \leq f(\mathbf{x}^{(i)}) + c_1 \alpha (\mathbf{g}^{(i)})^T \mathbf{d}^{(i)}, \quad c_1 \in (0, 1). \quad (2.19)$$

Ein typischer Wertebereich für die Steigungskonstante ist $c_1 \in (10^{-4}, 10^{-2})$ [NW06, S. 33]. Die Bedingung fordert in Abhängigkeit von der Schrittweite α eine gewisse Reduzierung der Gütefunktion, welche in Abb. 2.5 durch die obere Gerade dargestellt ist. Sämtliche gültigen Lösungen liegen unterhalb dieser Geraden. Ein Algorithmus zum Auffinden einer

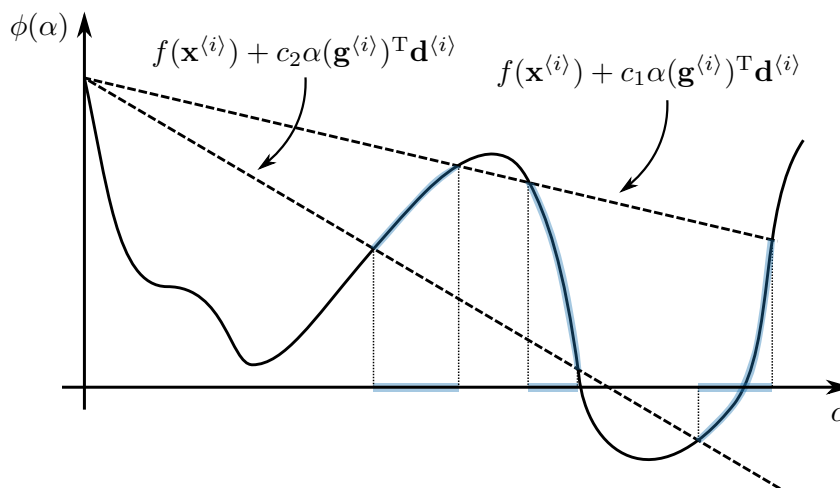


Abb. 2.5: Armijo-Goldstein-Bedingungen. Darstellung nach [NW06, Figure 3.6].

Schrittweite, welche diese Bedingung erfüllt, ist das sogenannte *Backtracking* [NW06, S. 37]. Der Backtracking-Algorithmus (Algorithmus 2.2.2) startet mit einer vorher festgelegten Schrittweite, beispielsweise $\alpha^{(i,0)} = 1$, und reduziert die Schrittweite so lange, bis die Armijo-Bedingung erfüllt ist.

Algorithmus 2.2.2 (Backtracking).**Initialisierung**

- Schrittweite festlegen: $\alpha^{(i,0)} := \alpha_{\text{start}}$

for j **in** $0, 1, \dots$ **do**

if Armijo-Bedingung (2.19) nicht erfüllt **then**

 - Verkleinere die Schrittweite

$$\alpha^{(i,j+1)} = \alpha_{\text{dec}} \alpha^{(i,j)}, \quad 0 < \alpha_{\text{dec}} < 1$$

else

 - **stop** Beende den Algorithmus

end if

end for

In einer konkreten Umsetzung eines Optimierungsalgorithmus könnte es jedoch von Interesse sein, das Backtracking fortzusetzen, bis sich die Gütefunktion nicht mehr verbessert bzw. anfängt sich zu verschlechtern. Dies kann bei nichtlinearen, nicht-konvexen Zusammenhängen dazu führen, dass die Schrittweite zu klein wird oder gar gegen Null konvergiert. Um dies zu verhindern, wird analog zur Armijo-Bedingung eine untere Grenze formuliert; die *Goldstein-Bedingung* [NW06, S. 36]. Beide Bedingungen zusammengenommen werden *Armijo-Goldstein-Bedingungen* genannt:

$$f(\mathbf{x}^{(i)}) + c_2 \alpha (\mathbf{g}^{(i)})^T \mathbf{d}^{(i)} \leq f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}) \leq f(\mathbf{x}^{(i)}) + c_1 \alpha (\mathbf{g}^{(i)})^T \mathbf{d}^{(i)}, \quad c_2 > c_1. \quad (2.20)$$

Die Steigungskonstante der unteren Grenze kann zu $c_2 := (1 - c_1)$ gewählt werden. Sämtliche Punkte, die beide Bedingungen erfüllen, kommen als Lösung in Betracht. In Abb. 2.5 sind die gültigen Schrittweiten sowie die zugehörigen Funktionswerte blau hinterlegt.¹⁵

Schrittweitschätzung nach Barzilai und Borwein

Die Geschwindigkeit einer Liniensuche hängt vom Startwert $\alpha^{(i,0)}$ ab. Je näher dieser an der Lösung liegt, desto weniger Iterationen sind erforderlich. Um einen geeigneten Startwert zu ermitteln, kann die Heuristik von *Barzilai und Borwein* [BB88] verwendet werden. Der Heuristik liegt die Annahme zugrunde, dass sich die optimale Schrittweite durch Mi-

¹⁵Es gibt neben den *Armijo-Goldstein-Bedingungen* die *Wolfe-Bedingungen*. Diese verwenden zur Kontrolle der Mindestschrittweite eine *Krümmungsbedingung* unter Verwendung der Ableitung $\phi'(\alpha)$. Die Wolfe-Bedingungen werden in [NW06, S. 33f] beschrieben.

minimierung von

$$\alpha^{(i,0)} = \arg \min_{\alpha > 0} \|\Delta \mathbf{x}^{(i)} - \alpha \Delta \mathbf{g}^{(i)}\|_2^2$$

mit

$$\begin{aligned} \Delta \mathbf{x}^{(i)} &:= \mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}, \\ \Delta \mathbf{g}^{(i)} &:= \mathbf{g}^{(i)} - \mathbf{g}^{(i-1)} \end{aligned}$$

ergibt. Aus Symmetriegründen sind zwei algebraische Lösungen möglich:

$$\alpha^{(i,0)} = \frac{(\Delta \mathbf{x}^{(i)})^T \Delta \mathbf{x}^{(i)}}{(\Delta \mathbf{x}^{(i)})^T \Delta \mathbf{g}^{(i)}} \quad (2.21a)$$

oder

$$\alpha^{(i,0)} = \frac{(\Delta \mathbf{x}^{(i)})^T \Delta \mathbf{g}^{(i)}}{(\Delta \mathbf{g}^{(i)})^T \Delta \mathbf{g}^{(i)}}. \quad (2.21b)$$

Für den eindimensionalen Fall entspricht das Einsetzen von (2.21a) bzw. (2.21b) in (2.17) bei gleichzeitiger Wahl der Abstiegsrichtung zu $\mathbf{d}^{(i)} := -\mathbf{g}^{(i)}$ dem Sekantenverfahren.¹⁶

Aus numerischer Sicht kann der Fall auftreten, dass der Nenner näherungsweise Null wird, insbesondere dann, wenn das Abstiegsverfahren dem Minimum nahe kommt. Dies beruht darauf, dass es dann beinahe keine Änderung mehr im Gradienten ($\Delta \mathbf{g}^{(i)} \rightarrow 0$) gibt und die berechnete Lösung ebenfalls näherungsweise konstant ($\Delta \mathbf{x}^{(i)} \rightarrow 0$) ist. Eines von beidem genügt bereits, um den Nenner in (2.21a) gegen Null gehen zu lassen. In dieser Gleichung kann zusätzlich die Situation auftreten, dass durch Orthogonalität von $\Delta \mathbf{x}^{(i)}$ und $\mathbf{g}^{(i)}$ der Nenner Null wird. Dieser Fall hat allerdings eher einen zufälligen Charakter als einen systematischen.

Die Orthogonalität ist für Gleichung (2.21b) ebenfalls nicht unkritisch, da in diesem Fall die Schrittweite permanent Null wäre und der Lösungs-Algorithmus die Optimierungsvariablen ab diesem Zeitpunkt nicht mehr verändern würde; der Algorithmus würde feststecken.

2.2.3 Gradientenprojektion

Bisher wurden Optimierungsprobleme ohne Nebenbedingungen (2.14) betrachtet. Auf dem Weg zur Betrachtung eines allgemeinen Optimierungsproblems mit Nebenbedingun-

¹⁶Das Sekantenverfahren konvergiert superlinear, jedoch langsamer als das Newton-Verfahren, welches eine quadratische Konvergenz aufweist [DR08, S. 182, 187].

gen (2.13b) soll zuvor der Sonderfall

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{u. B. v.} \quad & \mathbf{b}_l \leq \mathbf{x} \leq \mathbf{b}_u \end{aligned}$$

betrachtet werden. Dieses Optimierungsproblem definiert eine untere Grenze \mathbf{b}_l sowie eine obere Grenze \mathbf{b}_u für die Optimierungsvariable. Es handelt sich hierbei um isolierte lineare Ungleichungsbeschränkungen (zwei Bedingungen für jedes x_i). Räumlich betrachtet spannen diese Nebenbedingungen eine Box¹⁷ auf, wobei sämtliche in der Box liegenden Punkte zum Lösungsgebiet gehören.

Im Rahmen eines iterativen Verfahrens, wie beispielsweise dem Abstiegsverfahren, können solche Nebenbedingungen über eine sogenannte *Gradientenprojektion* berücksichtigt werden. Dazu wird eine stückweise Projektionsfunktion

$$\mathbf{P}(\mathbf{x}, \mathbf{b}_l, \mathbf{b}_u) = \left[P(x_1, b_{l,1}, b_{u,1}), \dots, P(x_n, b_{l,n}, b_{u,n}) \right]^T \quad (2.22a)$$

mit

$$P(x_i, b_{l,i}, b_{u,i}) = \begin{cases} b_{l,i} & \text{für } x_i < b_{l,i}, \\ b_{u,i} & \text{für } x_i > b_{u,i}, \\ x_i & \text{sonst} \end{cases} \quad (2.22b)$$

definiert. Durch Umformulieren der Iterationsvorschrift (2.17) des Abstiegsverfahrens zu

$$\mathbf{x}^{(i+1)} = \mathbf{P} \left(\mathbf{x}^{(i)} + \alpha^{(i)} \mathbf{d}^{(i)}, \mathbf{b}_l, \mathbf{b}_u \right), \quad \alpha^{(i)} > 0$$

ist sichergestellt, dass das berechnete $\mathbf{x}^{(i+1)}$ innerhalb des Lösungsgebiets liegt. Grafisch ist die Gradientenprojektion in Abb. 2.6 dargestellt. Die konzentrischen Ellipsen stellen die Höhenlinien der Gütefunktion dar, wobei der Funktionswert zur Mitte hin kleiner wird – in einer dreidimensionalen Vorstellung handelt es sich um einen parabolischen Becher. Die blau hinterlegte Box ist der durch die Nebenbedingungen vorgegebene gültige Lösungsbereich. Die ohne Projektion berechneten Optimierungsvariablen liegen außerhalb der blauen Box. Durch die elementweise Begrenzung der Lösung auf den zulässigen Bereich, werden die Optimierungsvariablen auf den Rand des Lösungsgebiets projiziert.

2.2.4 Erweiterte Lagrange-Methode

Um ein allgemeines nichtlineares Optimierungsproblem mit Nebenbedingungen (2.13b) zu lösen, soll die *erweiterte Lagrange-Methode* eingeführt werden. Die erweiterte Lagrange-Methode ändert die Formulierung des Optimierungsproblems in ein unbeschränktes Opti-

¹⁷In der englischen Literatur werden diese Nebenbedingungen als *box constraints* bezeichnet.

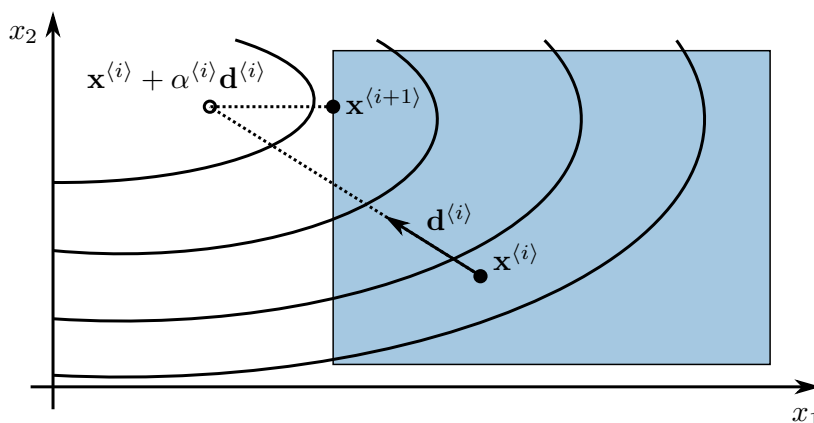


Abb. 2.6: Veranschaulichung der Gradientenprojektion.

mierungsproblem, damit das in Abschnitt 2.2.1 eingeführte Abstiegsverfahren angewendet werden kann. Sie basiert auf der *primal-dualen Lagrange-Methode* und der *quadratischen Strafterm-Methode*. Im Folgenden werden zunächst diese beiden Methoden getrennt eingeführt, bevor im letzten Absatz das kombinierte Verfahren vorgestellt wird.

Lagrange-Methode

Eine Kostenfunktion, welche neben der Gütefunktion $f(\cdot)$ die Nebenbedingungen $\mathbf{c}(\cdot)$ enthält, ist die sogenannte *Lagrange-Funktion*.

Definition 2.2.1 (Lagrange-Funktion). Die Lagrange-Funktion des Optimierungsproblems (2.13b) ist über

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &:= f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{c}(\mathbf{x}) \\ &= f(\mathbf{x}) + \boldsymbol{\lambda}_{\mathcal{E}}^T \mathbf{c}_{\mathcal{E}}(\mathbf{x}) + \boldsymbol{\lambda}_{\mathcal{I}}^T \mathbf{c}_{\mathcal{I}}(\mathbf{x}), \quad \boldsymbol{\lambda}_{\mathcal{I}} \geq \mathbf{0} \end{aligned} \quad (2.23)$$

mit den *Lagrange-Multiplikatoren* $\boldsymbol{\lambda}_{\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}$ für die Gleichungsbeschränkungen und $\boldsymbol{\lambda}_{\mathcal{I}} \in \mathbb{R}_+^{|\mathcal{I}|}$ für die Ungleichungsbeschränkungen definiert.

Die Nebenbedingungen gehen über die *Lagrange-Multiplikatoren* als gewichtete Summe in den Funktionswert ein. Die Lagrange-Multiplikatoren werden auch *duale Variablen* genannt. Entsprechend werden die eigentlich zu optimierenden Variablen \mathbf{x} *primale Variablen* genannt. Auch hier ist das Ziel, eine Minimierung bezüglich \mathbf{x} zu erreichen. Die dualen Variablen können als Gegenspieler der primalen Variablen verstanden werden: Wird eine Nebenbedingung nicht erfüllt, gilt also $c_{\mathcal{E},i}(\mathbf{x}) \neq 0$ bzw. $c_{\mathcal{I},i}(\mathbf{x}) > 0$, dann wird λ_i derart verändert, dass $\mathcal{L}(\cdot)$ größer wird. Dadurch ergibt sich eine andere primale Lösung, die dann idealerweise $c_{\mathcal{E},i}(\mathbf{x}) = 0$ bzw. $c_{\mathcal{I},i}(\mathbf{x}) \leq 0$ erfüllt. Jede Nebenbedingung impliziert eine duale Variable. Entsprechend vergrößert sich mit jeder zusätzlichen Nebenbedingung der Lösungsraum.

Die notwendigen Bedingungen 1. Ordnung sind über Satz 2.2.4, die sogenannten *Karush-Kuhn-Tucker-Bedingungen* (*KKT-Bedingungen*), gegeben.

Satz 2.2.4 (Notwendige Bedingungen 1. Ordnung). *Sei \mathbf{x}^* eine (lokale) optimale Lösung von (2.13b). Wenn die Funktionen f, c_i in (2.13b) stetig differenzierbar sind und wenn die aktiven Beschränkungen c_i linear unabhängig sind, dann existiert ein Vektor von Lagrange-Multiplikatoren $\boldsymbol{\lambda}^*$, der die Bedingungen*

$$\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0, \quad (2.24a)$$

$$c_i(\mathbf{x}^*) = 0, \quad i \in \mathcal{E}, \quad (2.24b)$$

$$c_i(\mathbf{x}^*) \leq 0, \quad i \in \mathcal{I}, \quad (2.24c)$$

$$\lambda_i^* \geq 0, \quad i \in \mathcal{I}, \quad (2.24d)$$

$$\lambda_i^* c_i(\mathbf{x}^*) = 0, \quad i \in \mathcal{I} \quad (2.24e)$$

an der Stelle $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ erfüllt.^a

^aEin Beweis der KKT-Bedingungen ist in [NW06, S. 323ff] zu finden.

Die Bedingung (2.24e) wird *Komplementaritätsbedingung* genannt und bedeutet, dass entweder die Nebenbedingung i aktiv ist oder $\lambda_i = 0$ gilt oder beides gemeinsam zutrifft [NW06, S. 321]. Für konvexe Probleme sind die KKT-Bedingungen auch hinreichend. Konvex bedeutet in diesem Fall, dass neben der Gütefunktion $f(\cdot)$ auch die Nebenbedingungen $c(\cdot)$ konvex sind [BV09, S. 226f].

Um auch bei nicht-konvexen Problemen eine Extremstelle als Minimum zu qualifizieren, werden die hinreichenden Bedingungen 2. Ordnung (2.25) verwendet.

Satz 2.2.5 (Hinreichende Bedingungen 2. Ordnung). *Sei $\boldsymbol{\lambda}^*$ ein Lagrange-Multiplikator, der für einen gültigen Punkt \mathbf{x}^* die KKT-Bedingungen (2.24) erfüllt. Seien ferner die Funktionen f und c_i zweimal stetig differenzierbar. Wenn*

$$\mathbf{w}^T \nabla_{\mathbf{xx}}^2 \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{w} > 0, \quad \forall \mathbf{w} \in \mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \quad (2.25a)$$

mit

$$\mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \{ \mathbf{w} \in \mathbb{R}^n : \nabla c_i(\mathbf{x})^T \mathbf{w} = 0, \quad i \in \mathcal{E}, \quad (2.25b)$$

$$\nabla c_i(\mathbf{x})^T \mathbf{w} = 0, \quad i \in \{i \mid \lambda_{\mathbf{h},i} > 0, i \in \mathcal{I}\}, \quad (2.25c)$$

$$\nabla c_i(\mathbf{x})^T \mathbf{w} \leq 0, \quad i \in \{i \mid \lambda_{\mathbf{h},i} = 0, i \in \mathcal{I}\} \quad (2.25d)$$

gilt, dann ist \mathbf{x}^* eine (lokale) Lösung von (2.13b).^a

^aEin Beweis der Optimalitätsbedingungen 2. Ordnung ist in [NW06, S. 333ff] zu finden.

Für eine numerische Lösung bietet es sich an, die Ungleichungsbeschränkung $\mathbf{c}_{\mathcal{I}}(\cdot)$ in eine Ungleichungsbeschränkung unter Verwendung der *Schlupfvariablen* \mathbf{s} umzuwandeln. Das

zu lösende Optimierungsproblem hat damit die Gestalt

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{s}} \quad & f(\mathbf{x}) \\ \text{u. B. v.} \quad & \mathbf{c}_{\mathcal{E}}(\mathbf{x}) = \mathbf{0}, \\ & \mathbf{c}_{\mathcal{I}}(\mathbf{x}) + \mathbf{s} = \mathbf{0}, \\ & \mathbf{s} \geq \mathbf{0}. \end{aligned} \tag{2.26}$$

Die Schlupfvariable ist Teil der primalen Variablen. Formal wird jedoch im Weiteren zwischen diesen unterschieden. Die Lagrange-Funktion kann mit der Kombination eines Abstiegsverfahrens und eines Anstiegsverfahrens über die Iterationsvorschriften

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} - \alpha^{(i)} \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^{(i)}, \mathbf{s}^{(i)}, \boldsymbol{\lambda}^{(i)}), \tag{2.27a}$$

$$\mathbf{s}^{(i+1)} = \mathbf{P}(\mathbf{s}^{(i)} - \alpha^{(i)} \nabla_{\mathbf{s}} \mathcal{L}(\mathbf{x}^{(i)}, \mathbf{s}^{(i)}, \boldsymbol{\lambda}^{(i)}), \mathbf{0}, \infty), \tag{2.27b}$$

$$\boldsymbol{\lambda}^{(i+1)} = \boldsymbol{\lambda}^{(i)} + \alpha^{(i)} \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}^{(i)}, \mathbf{s}^{(i)}, \boldsymbol{\lambda}^{(i)}) \tag{2.27c}$$

gelöst werden. Die primalen Variablen werden über ein Abstiegsverfahren und die dualen Variablen über ein Anstiegsverfahren iteriert. Über die Projektionsfunktion $\mathbf{P}(\cdot)$ (vgl. Gleichung (2.22a)) wird $\mathbf{s} \geq \mathbf{0}$ sichergestellt. Ansätze dieser Art werden in [AS58] (zitiert in [Ber96], [Pol70]) besprochen, worauf an dieser Stelle verwiesen sei.

Für konvexe Optimierungsprobleme ist eine Lösung über ein nichtlineares Gleichungssystem

$$\mathbf{F}(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}) = \begin{bmatrix} \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}) \\ \nabla_{\mathbf{s}} \mathcal{L}(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}) \\ \mathbf{c}_{\mathcal{E}}(\mathbf{x}) \\ \mathbf{c}_{\mathcal{I}}(\mathbf{x}) + \mathbf{s} \end{bmatrix} = \mathbf{0}, \quad \mathbf{s} \geq \mathbf{0} \tag{2.28}$$

denkbar. Für nicht-konvexe Probleme ist dieser Ansatz ungeeignet, da er lediglich die notwendigen Bedingungen erfüllt, welche insbesondere Maxima nicht ausschließen; es wird lediglich eine Extremstelle gefunden. In (2.27) ist über die Suchrichtung das Auffinden eines Minimums bzw. eines Sattelpunktes sichergestellt.¹⁸ In einer rechentechnischen Umsetzung ist auf den Wertebereich der Schlupfvariablen zu achten.

Verfahren nach der Lagrange-Methode haben den Nachteil, dass sie für von der Lösung weit entfernte Startwerte langsam konvergieren [Ber96, S. 3].

¹⁸Dies gilt mit der Ausnahme eines genau auf einem Maximum liegenden Startwertes. An dieser Stelle verschwindet der Gradient und der Lösungsalgorithmus wäre sofort beendet.

Primal-duale Lagrange-Methode

Die primal-duale Lagrange-Methode basiert auf den gleichen Überlegungen wie die zuvor dargestellte Lagrange-Methode, mit dem Unterschied, dass \mathbf{x} und $\boldsymbol{\lambda}$ getrennt voneinander berechnet werden.

Definition 2.2.2 (Duale Funktion). Die Lagrange-duale Funktion (kurz: duale Funktion) $g : \mathbb{R}^{|\mathcal{E}|+|\mathcal{I}|} \mapsto \mathbb{R}$ ist das Minimum der Lagrange-Funktion bezüglich der primalen Variablen \mathbf{x} . Sie ist durch

$$g(\boldsymbol{\lambda}) := \inf_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) \quad (2.29)$$

definiert.^a

^aDie duale Funktion wird in [BV09, S. 215ff] besprochen.

Die Grundlage dieses Ansatzes bildet die *duale Funktion* aus Definition 2.2.2. Die duale Funktion ist konkav und gibt für die optimale Lösung von $f(\cdot)$ eine untere Grenze an:

$$g(\boldsymbol{\lambda}) \leq f(\mathbf{x}^*).$$

Der Abstand wird für den maximalen Wert der dualen Funktion minimal. Es ist folglich das stets konkave Optimierungsproblem

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \quad & g(\boldsymbol{\lambda}) \\ \text{u. B. v.} \quad & \boldsymbol{\lambda}_{\mathcal{I}} \geq \mathbf{0} \end{aligned} \quad (2.30)$$

zu lösen, um in diesem Sinne zu einer optimalen Lösung zu gelangen. Dieses Problem wird das zu dem *primalen Optimierungsproblem* (2.13b) *duale Optimierungsproblem* genannt. Für ein konvexes primales Optimierungsproblem sind die primale und die duale Lösung identisch; die *Dualitätslücke* verschwindet [BV09, S. 226f]:

$$g(\boldsymbol{\lambda}^*) = f(\mathbf{x}^*) \stackrel{(2.24b)}{=} \stackrel{(2.24e)}{=} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*). \quad (2.31)$$

Gemäß (2.31) kann für die optimale Lösung des einen Problems die Lösung des jeweils anderen nach

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}^*), \quad (2.32a)$$

$$\boldsymbol{\lambda}^* = \arg \max_{\boldsymbol{\lambda}_{\mathcal{E}}, \boldsymbol{\lambda}_{\mathcal{I}} \geq \mathbf{0}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}) \quad (2.32b)$$

berechnet werden.

Wie auch bei der Lagrange-Methode sind die Ungleichungsbeschränkungen in einer algorithmischen Struktur gesondert zu behandeln, beispielsweise unter Verwendung einer

Schlupfvariablen (vgl. Gleichung (2.26)). Das prinzipielle Vorgehen ist in Algorithmus 2.2.3 dargestellt. In einem ersten Schritt wird das *innere* Optimierungsproblem gelöst; es wird das Minimum der Lagrange-Funktion für ein gegebenes $\boldsymbol{\lambda}^{(i)}$ ermittelt. Der Funktionswert der Lagrange-Funktion entspricht nach dieser Optimierung dem der dualen Funktion (2.29). Anschließend wird $\boldsymbol{\lambda}^{(i)}$ in Richtung des Maximums der dualen Funktion verändert. Die Maximierung hat die Lösung des dualen Optimierungsproblems (2.30) zum Ziel.

Algorithmus 2.2.3 (Primal-duale Lagrange-Methode).

Initialisierung

- Startpunkt festlegen: $\mathbf{x}^{(0)} := \mathbf{x}_{\text{start}}, \mathbf{s}^{(0)} := \mathbf{0}, \boldsymbol{\lambda}^{(0)} := \mathbf{0}$

for i **in** $0, 1, \dots$ **do**

- Bestimme die primalen Variablen für ein gegebenes $\boldsymbol{\lambda}^{(i)}$

$$\left[\mathbf{x}^{(i+1)}, \mathbf{s}^{(i+1)} \right] = \arg \min_{\mathbf{x}, \mathbf{s} \geq \mathbf{0}} \mathcal{L}(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}^{(i)})$$

if Konvergenztest für (2.26) erfüllt **then**

- **stop** Beende den Alorithmus

end if

- Gehe in Richtung des Maximums der dualen Funktion

$$\boldsymbol{\lambda}^{(i+1)} = \boldsymbol{\lambda}^{(i)} + \alpha^{(i)} \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}^{(i+1)}, \mathbf{s}^{(i+1)}, \boldsymbol{\lambda}^{(i)})$$

end for

Wenn strenge Dualität nicht gegeben ist, liefert das duale Problem zumindest eine lokal minimale Lösung, die die Nebenbedingungen erfüllt; Lösbarkeit vorausgesetzt. Die primal-duale Lagrange-Methode lässt sich lediglich bei lokaler Konvexität ($\nabla_{\mathbf{xx}}^2 \mathcal{L} > 0$) sinnvoll anwenden [Ber96, S. 156].

Quadratische Strafterm-Methode

Ein vom Grundgedanken her ähnlicher Ansatz zu den zuvor genannten Lagrange-Ansätzen ist die *quadratische Strafterm-Methode* nach [FM68] (zitiert in [Ber96, S. 3]). Auch hier wird eine neue Funktion definiert, die neben der Gütefunktion zusätzlich die Nebenbedingungen enthält. Jedoch gehen die Nebenbedingungen als quadratische Summe in die *Straffunktion*

$$Q(\mathbf{x}; \nu) := f(\mathbf{x}) + \frac{\nu}{2} \sum_{i \in \mathcal{E}} c_i^2(\mathbf{x}) + \frac{\nu}{2} \sum_{i \in \mathcal{I}} ([c_i(\mathbf{x})]^+)^2 = f(\mathbf{x}) + \frac{\nu}{2} \|\mathbf{c}_{\mathcal{E}}(\mathbf{x})\|_2^2 + \frac{\nu}{2} \|[\mathbf{c}_{\mathcal{I}}(\mathbf{x})]^+\|_2^2$$

unter Verwendung der Notation

$$[\mathbf{y}]^+ := [\max\{y_1, 0\}, \max\{y_2, 0\}, \dots]^T$$

ein. Die Nebenbedingungen werden über den *Strafparameter* $\nu > 0$ gewichtet. Der Ablauf der zugehörigen Optimierung ist in Algorithmus 2.2.4 aufgezeigt.

Algorithmus 2.2.4 (Quadratische Strafterm-Methode).

Initialisierung

- Startpunkt festlegen: $\mathbf{x}^{(0)} := \mathbf{x}_{\text{start}}, \nu^{(0)} := \nu_{\text{start}} > 0$

for i **in** $0, 1, \dots$ **do**

- Bestimme die Optimierungsvariablen für ein gegebenes $\nu^{(i)}$

$$\mathbf{x}^{(i+1)} = \arg \min_{\mathbf{x}} Q(\mathbf{x}; \nu^{(i)})$$

if Konvergenztest für (2.13b) erfüllt **then**

- **stop** Beende den Algorithmus

end if

- Wähle einen neuen Strafparameter

$$\nu^{(i+1)} \leftarrow \nu^{(i)}$$

end for

Die Methode basiert darauf, den Strafparameter von Iteration zu Iteration zu vergrößern und so die Bedeutung der (nicht eingehaltenen) Nebenbedingungen zu erhöhen. Auf welche Weise der Strafparameter aktualisiert wird, ist nicht festgelegt. In [NW06, S. 501] wird vorgeschlagen, den Strafparameter, wenn die Bestimmung von $\mathbf{x}^{(i+1)}$ *vielen* Iterationen in Anspruch genommen hat, lediglich etwas größer zu wählen, beispielsweise $\nu^{(i+1)} = 1,5 \nu^{(i)}$. Andernfalls wird mit $\nu^{(i+1)} = 10 \nu^{(i)}$ der Strafparameter stärker vergrößert, um die Konvergenzgeschwindigkeit zu erhöhen.

Im Gegensatz zu der Lagrange-Methode, bei der die dualen Variablen am Ende des Algorithmus konkrete Werte annehmen, geht der Strafparameter für die optimale Lösung gegen unendlich. Es ist mit diesem Ansatz dementsprechend nicht möglich, die Nebenbedingungen ohne bleibenden Fehler zu erfüllen.¹⁹ Des Weiteren führt der skalare Strafterm, der gleichzeitig auf alle Nebenbedingungen angewendet wird, zu einer schlechten Konditionierung der Hesse-Matrix $\nabla_{\mathbf{xx}}Q$ für große Strafparameter (die Singulärwerte sind stark unterschiedlich). Dies hat zur Folge, dass zum einen bei einem Gradientenverfahren der Gradient stark von der optimalen Suchrichtung abweicht, und zum anderen selbst bei

¹⁹Es gibt Strafterm-Methoden mit *exakten* Straftermen [NW06, S. 507ff].

einem Newton-Verfahren, wo

$$\nabla_{\mathbf{xx}}Q(\mathbf{x}; \nu)\mathbf{d} = \nabla_{\mathbf{x}}Q(\mathbf{x}; \nu)$$

gelöst wird, die berechnete Schrittweite unter der schlechten Konditionierung leidet [Ber96, S. 103].

Dagegen bietet die Strafterm-Methode den Vorteil einer geringen Komplexität bei der rechen-technischen Implementierung, insbesondere bezüglich der Ungleichungsbeschränkungen; eine gesonderte Behandlung über eine Schlupfvariable ist nicht erforderlich. Ein weiterer Vorteil besteht in der lokalen Konvexität ($\nabla_{\mathbf{xx}}Q > 0$) für einen genügend großen Strafterm [Ber96, S. 102].

Erweiterte Lagrange-Methode

Die primal-duale Lagrange-Methode hat den Nachteil, dass sie lokale Konvexität verlangt, und die quadratische Strafterm-Methode den der schlechten Konditionierung. Jedoch ist die Schwäche der einen Methode jeweils die Stärke der anderen. Die Idee hinter der *erweiterten Lagrange-Methode* ist, beide Ansätze in der Hoffnung zu kombinieren, dass sich die Schwächen und Stärken der Ansätze ausgleichen. Für den Fall eines Optimierungsproblems der Form (2.13b), allerdings ohne Ungleichungsbeschränkungen, lautet die Kostenfunktion²⁰

$$\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}_{\mathcal{E}}; \nu) := f(\mathbf{x}) + \boldsymbol{\lambda}_{\mathcal{E}}^T \mathbf{c}_{\mathcal{E}}(\mathbf{x}) + \frac{\nu}{2} \|\mathbf{c}_{\mathcal{E}}(\mathbf{x})\|_2^2. \quad (2.33)$$

Die Ungleichungsbeschränkungen können über die Umformulierung (2.26) unter der Verwendung von Schlupfvariablen \mathbf{s} berücksichtigt werden:

$$\bar{\mathcal{L}}_A(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}; \nu) = f(\mathbf{x}) + \boldsymbol{\lambda}_{\mathcal{E}}^T \mathbf{c}_{\mathcal{E}}(\mathbf{x}) + \frac{\nu}{2} \|\mathbf{c}_{\mathcal{E}}(\mathbf{x})\|_2^2 + \sum_{i \in \mathcal{I}} \lambda_i (c_i(\mathbf{x}) + s_i) + \frac{\nu}{2} (c_i(\mathbf{x}) + s_i)^2.$$

Nach [Ber96, S. 159] können die Schlupfvariablen explizit über

$$s_i = \arg \min_{s_i \geq 0} \bar{\mathcal{L}}_A(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}; \nu) = \arg \min_{s_i \geq 0} \lambda_i (c_i(\mathbf{x}) + s_i) + \frac{\nu}{2} (c_i(\mathbf{x}) + s_i)^2$$

berechnet werden.²¹ Die unbeschränkte Lösung ergibt sich über die Optimalitätsbedin-

²⁰Die Methode wurde von [Hes69] und [Pow69] (zitiert in [Ber96, S. 4] und [NW06, S. 526]) für Gleichungsbeschränkungen vorgeschlagen.

²¹Für die Lagrange-Methode funktioniert dieser Ansatz nicht.

gung 1. Ordnung (2.15)

$$\begin{aligned} \lambda_i + \nu(c_i(\mathbf{x}) + s_i) &= 0 \\ \Leftrightarrow s_i &= -\left(\frac{\lambda_i}{\nu} + c_i(\mathbf{x})\right). \end{aligned}$$

Aufgrund der Konvexität des zugrunde liegenden Optimierungsproblems ist diese Lösung global. Unter Berücksichtigung der Nebenbedingung $s_i \geq 0$ folgt

$$s_i^* = \max \left\{ 0, -\left(\frac{\lambda_i}{\nu} + c_i(\mathbf{x})\right) \right\}$$

als Lösung für die Schlupfvariable s_i . Unter Verwendung der Notation

$$\begin{aligned} c_i^+(\mathbf{x}, \lambda_i; \nu) &:= c_i(\mathbf{x}) + s_i^* = \max \left\{ c_i(\mathbf{x}), -\frac{\lambda_i}{\nu} \right\}, \\ \mathbf{c}^+(\mathbf{x}, \boldsymbol{\lambda}_{\mathcal{I}}; \nu) &:= \left[\dots, c_i(\mathbf{x}, \lambda_i; \nu)^+, \dots \right]^T \end{aligned} \quad (2.34)$$

ergibt sich die vollständige Formulierung der erweiterten Lagrange-Funktion

$$\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}; \nu) := f(\mathbf{x}) + \boldsymbol{\lambda}^T \tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\lambda}; \nu) + \frac{\nu}{2} \|\tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\lambda}; \nu)\|_2^2 \quad (2.35)$$

mit

$$\tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\lambda}, \nu) := \left[\mathbf{c}_{\mathcal{E}}(\mathbf{x}), \mathbf{c}_{\mathcal{I}}^+(\mathbf{x}, \boldsymbol{\lambda}_{\mathcal{I}}; \nu) \right]^T.$$

Es sei darauf hingewiesen, dass $\boldsymbol{\lambda}_{\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}|}$ gilt, also nicht $\boldsymbol{\lambda}_{\mathcal{I}} \in \mathbb{R}_+^{|\mathcal{I}|}$, da diese dualen Variablen formal Gleichungsbeschränkungen zugeordnet sind. In den vorigen Abschnitten waren sie Ungleichungsbeschränkungen zugeordnet (vgl. Gleichung (2.24d)), sodass dort lediglich positive Werte zulässig waren. Neben der Formulierung (2.35) gibt es weitere Möglichkeiten, die Ungleichungsbeschränkungen zu formulieren. Einige sind in [NW06, S. 519ff] zu finden.

Zur Anwendung der erweiterten Lagrange-Methode ist das Problem

$$\max_{\boldsymbol{\lambda}} \min_{\mathbf{x}} \mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}; \nu)$$

zu lösen. Der Strafparameter wird über eine Heuristik derart gewählt, dass die Nebenbedingungen ein Genauigkeitskriterium erfüllen. Ein numerisches Vorgehen ist in Algorithmus 2.2.5 skizziert. Grundsätzlich wird wie in Algorithmus 2.2.3 vorgegangen, jedoch wird zusätzlich nach der Bestimmung der aktualisierten dualen Variablen der Strafparameter aktualisiert.

Wenn der Strafparameter ν groß genug ist, dann ist die erweiterte Lagrange-Methode lokal konvex, womit die unterlagerte primal-duale Lagrange-Methode anwendbar ist [Ber96].

Algorithmus 2.2.5 (Erweiterte Lagrange-Methode).**Initialisierung**

- Startpunkt festlegen: $\mathbf{x}^{(0)} := \mathbf{x}_{\text{start}}, \nu^{(0)} := \nu_{\text{start}} > 0$

for i **in** $0, 1, \dots$ **do**

- Bestimme die primalen Variablen für gegebenes $\boldsymbol{\lambda}^{(i)}$ und $\mu^{(i)}$

$$\mathbf{x}^{(i+1)} = \arg \min_{\mathbf{x}} \mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}^{(i)}; \nu^{(i)})$$

if Konvergenztest für (2.26) erfüllt **then**

- **stop** Beende den Algorithmus

end if

- Aktualisiere die dualen Variablen^a

$$\boldsymbol{\lambda}^{(i+1)} = \boldsymbol{\lambda}^{(i)} + \alpha^{(i)} \nu^{(i)} \nabla_{\boldsymbol{\lambda}} \mathcal{L}_A(\mathbf{x}^{(i+1)}, \boldsymbol{\lambda}^{(i)}; \nu^{(i)}) \quad (2.36)$$

- Wähle einen neuen Strafparameter

$$\nu^{(i+1)} \leftarrow \nu^{(i)}$$

end for

^aDie Aktualisierungsvorschrift der dualen Variablen wird in [NW06, S. 515] motiviert.

Da der Strafparameter nicht mehr beliebig groß werden muss, ist die algorithmusbedingte schlechte Konditionierung entsprechend schwächer ausgeprägt. Diese Methode schafft es, die Vor- und Nachteile der Methoden zu moderieren.²² Das Verfahren wurde unabhängig voneinander von *Hestens* [Hes69] und *Powell* [Pow69] entwickelt. Die Behandlung von Ungleichungsbeschränkungen geht auf *Rockafellar* [Roc73] zurück.[NW06]

2.3 Dynamische Optimierung

Im vorigen Abschnitt 2.2 wurde eine Funktion bezüglich einer Optimierungsvariablen $\mathbf{x} \in \mathbb{R}^n$ aus dem Euklidischen Raum minimiert. Ein *dynamisches Optimierungsproblem* (DP) ergibt sich, wenn die Optimierungsvariable eine Parametrierung einer unabhängigen Variablen t ist. Bei *Optimalsteuerungsproblemen* wird eine über die Zeit parametrisierte Steuertrajektorie $\mathbf{u}(t)$, welche eine zeitabhängige Funktion aus dem Hilbertraum

²²Es gibt Softwarepakete, wie beispielsweise MINOS [MS83] und LANCELOT [CGT92], die auf dieses Lösungsverfahren setzen.

darstellt, ermittelt.²³

Das in dieser Arbeit betrachtete Optimalsteuerungsproblem hat die Form

$$\min_{\mathbf{u}(\cdot)} \quad J(\mathbf{u}(\cdot)) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\mathbf{x}(t), \mathbf{u}(t), t) dt \quad (2.37a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (2.37b)$$

$$\mathbf{b}_l(t) \leq \mathbf{u}(t) \leq \mathbf{b}_u(t), \quad (2.37c)$$

$$\mathbf{c}_{\mathcal{E}}(\mathbf{x}(t), \mathbf{u}(t), t) = 0, \quad (2.37d)$$

$$\mathbf{c}_{\mathcal{I}}(\mathbf{x}(t), \mathbf{u}(t), t) \leq 0, \quad (2.37e)$$

$$\mathbf{c}_{T, \mathcal{E}}(\mathbf{x}(t_T), t_T) = 0, \quad (2.37f)$$

$$\mathbf{c}_{T, \mathcal{I}}(\mathbf{x}(t_T), t_T) \leq 0. \quad (2.37g)$$

mit $\mathbf{u}(t), \mathbf{b}_l(t), \mathbf{b}_u(t) \in \mathbb{R}^m$, $\mathbf{x}(t), f(\cdot) \in \mathbb{R}^n$, $V(\cdot), l(\cdot) \in \mathbb{R}$, $\mathbf{c}_{\mathcal{E}}(\cdot) \in \mathbb{R}^{|\mathcal{E}|}$, $\mathbf{c}_{\mathcal{I}}(\cdot) \in \mathbb{R}^{|\mathcal{I}|}$, $\mathbf{c}_{T, \mathcal{E}}(\cdot) \in \mathbb{R}^{|\mathcal{E}_T|}$ und $\mathbf{c}_{T, \mathcal{I}}(\cdot) \in \mathbb{R}^{|\mathcal{I}_T|}$. Zur Notation sei angemerkt, dass die Nebenbedingungen analog zur Äquivalenz von (2.13a) zu (2.13b) über

$$\begin{aligned} c_i(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) &= 0, & i \in \mathcal{E}, \\ c_i(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) &\leq 0, & i \in \mathcal{I}, \\ c_{T, i}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) &= 0, & i \in \mathcal{E}_T, \\ c_{T, i}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) &\leq 0, & i \in \mathcal{I}_T \end{aligned}$$

kompakt dargestellt werden können. Die Mengen \mathcal{E} , \mathcal{I} , \mathcal{E}_T , \mathcal{I}_T enthalten die Indizes der zugehörigen Nebenbedingungen. Ziel der Optimierung ist es, die optimale Steuerungstrajektorie $\mathbf{u}^*(t)$ zu ermitteln, welche das Kostenfunktional $J(\cdot)$ minimiert, unter Verwendung des Modells (2.37b) und gleichzeitiger Einhaltung der Nebenbedingungen (2.37c-2.37g). Die optimale Steuerungstrajektorie wird auch *optimale Steuerung* genannt.

Kostenfunktional

Das Kostenfunktional (2.37a) setzt sich aus einer *Endkostengewichtung*²⁴ $V(\cdot)$ und der *Integralekostengewichtung* $l(\cdot)$ zusammen. Diese Formulierung wird *Bolza-Form* genannt [Pap15, S. 207]. Die aus dem Entwurf von (linearen) Optimalreglern bekannten Integralekriterien

²³Ein Beispiel für eine andere Parametrierung als über die Zeit wird in [Pap15, S. 3f] gegeben. Dort wird ermittelt, welchen geometrischen Verlauf ein an zwei Punkten aufgehängtes Seil einnimmt. Die Parametrierung erfolgt über den Ort.

²⁴Für die Endkostengewichtung ist in der Literatur auch der Name *Mayer-Term* zu finden.

sind:

$$l(\mathbf{x}, \mathbf{u}, t) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} \quad (\text{quadratische Zustandsgewichtung,} \\ \mathbf{Q} \text{ symmetrisch und } \mathbf{Q} \succ 0), \quad (2.38a)$$

$$l(\mathbf{x}, \mathbf{u}, t) = \frac{1}{2} \mathbf{u}^T \mathbf{R} \mathbf{u} \quad (\text{quadratische Eingangsgewichtung,} \\ \mathbf{R} \text{ symmetrisch und } \mathbf{R} \succ 0), \quad (2.38b)$$

$$l(\mathbf{x}, \mathbf{u}, t) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u} \quad (\text{allgemeines quadratisches Kriterium,} \\ \mathbf{Q}, \mathbf{R} \text{ symmetrisch und } \mathbf{Q} \succ 0, \mathbf{R} \succ 0). \quad (2.38c)$$

Typischerweise werden die Wichtungsmatrizen \mathbf{Q} und \mathbf{R} als Diagonalmatrizen ausgelegt. Jedes Diagonalelement ist dann einem Zustand bzw. einem Eingang zuordenbar. Wird eine Zustandsgewichtung vergrößert, so wird der korrespondierende Zustand schneller eingeregelt. Wird hingegen ein Diagonalelement in der Eingangsgewichtung vergrößert, so wird die Nutzung des jeweiligen Aktuators bestraft und der Regler wird diesen behutsamer verwenden, was den Regler gemeinhin verlangsamt.[Lun13, S. 304ff]

Der Vollständigkeit halber soll ein weiteres relevantes Integalkriterium, das zeitoptimale Kriterium, aufgeführt werden:

$$l(\mathbf{x}, \mathbf{u}, t) = 1 \Rightarrow \int_{t_0}^{t_T} 1 \, dt = t_T - t_0.$$

Dieses Gütekriterium ist sinnvoll, wenn neben der Steuerungstrajektorie die Endzeit t_T eine Optimierungsvariable ist. Damit ist es möglich, in Kombination mit einer geeigneten Endkostengewichtung bzw. geeigneten Beschränkungen des Zielzustandes (2.37g) einen Endzustand möglichst schnell zu erreichen. Im weiteren Verlauf wird dieses Kriterium nicht vertieft. Genauere Ausführungen sind in [Pap15, S. 266ff] zu finden.

Nebenbedingungen

Die Nebenbedingungen können grob in drei Gruppen aufgeteilt werden. Die erste Gruppe besteht aus dem zu regelnden System (2.37b). Diese Nebenbedingung gibt die Trajektorie des Zustandsvektors $\mathbf{x}(t)$ vor und ist Bestandteil eines jeden dynamischen Steuerungsproblems.²⁵

Die zweite Gruppe (2.37c) bis (2.37e) enthält die Beschränkungen, die im Intervall $[t_0, t_T]$, also während der Integration, aktiv werden können. Diese Nebenbedingungen schränken den Verlauf der Trajektorie ein und werden daher *integrale Nebenbedingungen* genannt.

²⁵Dynamisch ist an dieser Stelle so zu verstehen, dass das gesteuerte System ein Zeitverhalten und damit Zustände aufweist.

Die erste integrale Nebenbedingung (2.37c) beschränkt die Stellgrößen auf ein zulässiges Gebiet

$$\mathbf{u}(t) \in U \subseteq \mathbb{R}^m, \quad \forall t \in [t_0, t_T],$$

und trägt damit den technischen Grenzen der verwendeten Aktuatoren Rechnung. Formal stellt diese Nebenbedingung einen Sonderfall von (2.37e) dar. Aufgrund der gesonderten numerischen Behandlung in Abschnitt 4.4.4 wird sie separat ausgewiesen. Mit den Nebenbedingungen (2.37d) sowie (2.37e) können beliebige nichtlineare Zusammenhänge formuliert werden. Herauszuheben ist die Möglichkeit, die Systemzustände auf ein zulässiges Gebiet

$$\mathbf{x}(t) \in X \subseteq \mathbb{R}^n, \quad \forall t \in [t_0, t_T],$$

begrenzen zu können.

Die letzte Gruppe, bestehend aus (2.37f) und (2.37g), ermöglicht die Festlegung eines Endgebietes. Die Zustände müssen sich zur Endzeit t_T in einem Zielgebiet

$$\mathbf{x}(t_T) \in X_T \subseteq \mathbb{R}^n$$

befinden. Diese werden *Endnebenbedingungen* oder *terminale Nebenbedingungen* genannt.

2.3.1 Optimalitätsbedingungen

Analog zu den Optimalitätsbedingungen der statischen Optimierung (vgl. Satz 2.2.4, Satz 2.2.5) sollen diese im Folgenden für die dynamische Optimierung angegeben werden. Es werden drei aufeinander aufbauende Fälle betrachtet: Zuerst wird der unbeschränkte Fall betrachtet, welcher im darauf betrachteten Fall um Eingangsbeschränkungen erweitert wird. Im letzten Fall werden die Optimalitätsbedingungen des vollständigen Optimierungsproblems (2.37) angegeben.

Unbeschränktes Optimalsteuerungsproblem

Es wird das unbeschränkte Optimalsteuerungsproblem

$$\min_{\mathbf{u}(\cdot)} \quad J(\mathbf{u}(\cdot)) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\mathbf{x}(t), \mathbf{u}(t), t) dt \quad (2.39a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (2.39b)$$

betrachtet. Das Modell 2.37b ist formal eine Beschränkung, die jedoch inhärenter Bestandteil des Steuerungsproblems und daher im Folgenden mit *Beschränkung* nicht gemeint ist.

Analog zum Vorgehen bei der Formulierung der Lagrange-Funktion (2.23) können die Nebenbedingungen mit in das Kostenfunktional aufgenommen werden:

$$\bar{J}(\mathbf{u}(\cdot)) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T(t)(\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) - \dot{\mathbf{x}}(t)) dt. \quad (2.40)$$

Aus den Lagrange-Multiplikatoren in (2.23) ist eine zeitabhängige Funktion $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ geworden. An dieser Stelle könnte die notwendige Bedingung 1. Ordnung analog zu den KKT-Bedingungen (2.24) und die Bedingung 2. Ordnung analog zu (2.25) formuliert werden. Jedoch wäre $\boldsymbol{\lambda}^*(t)$ dann ebenfalls zu ermitteln.

Weniger allgemeine Bedingungen können über eine Variationsrechnung hergestellt werden. Das Funktional $\bar{J}(\cdot)$ soll minimal werden. Damit wird als notwendige Bedingung 1. Ordnung entsprechend

$$\nabla_{\mathbf{u}} \bar{J}(\mathbf{u}(\cdot)) = 0$$

gefordert (vgl. Satz 2.2.1). Wird nun die Steuerungstrajektorie durch eine zulässige Variation $\delta \mathbf{u}(t)$ von der optimalen Steuerung $\mathbf{u}^*(t)$

$$\mathbf{u}(t) := \mathbf{u}^*(t) + \epsilon \delta \mathbf{u}(t), \quad \forall t \in [t_0, t_T],$$

unter Verwendung der skalaren Variablen ϵ ersetzt, ergibt sich für die Variation des Kostenfunktionals die Forderung

$$\delta \bar{J}(\epsilon) := \frac{\partial \bar{J}(\mathbf{u}^*(t) + \epsilon \delta \mathbf{u}(t))}{\partial \epsilon} = 0.$$

Mit der zusätzlichen Variation $\delta \mathbf{x}(t)$ von der optimalen Zustandstrajektorie $\mathbf{x}^*(t)$ sowie dem Einsetzen von (2.40) folgt

$$\begin{aligned} 0 &= \nabla_{\mathbf{x}} V(\mathbf{x}^*(t_T), t_T)^T \delta \mathbf{x}(t_T) \\ &+ \int_{t_0}^{t_T} \left[\nabla_{\mathbf{x}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) + \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^T \boldsymbol{\lambda}(t) \right]^T \delta \mathbf{x}(t) \\ &+ \left[\nabla_{\mathbf{u}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) + \nabla_{\mathbf{u}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^T \boldsymbol{\lambda}(t) \right]^T \delta \mathbf{u}(t) - \boldsymbol{\lambda}(t)^T \delta \dot{\mathbf{x}}(t) dt. \end{aligned}$$

Eine anschließende partielle Integration liefert

$$\begin{aligned} 0 &= \nabla_{\mathbf{x}} V(\mathbf{x}^*(t_T), t_T)^T \delta \mathbf{x}(t_T) - \left[\boldsymbol{\lambda}(t)^T \delta \mathbf{x}(t) \right]_{t_0}^{t_T} \\ &+ \int_{t_0}^{t_T} \left[\nabla_{\mathbf{x}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) + \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^T \boldsymbol{\lambda}(t) + \dot{\boldsymbol{\lambda}}(t) \right]^T \delta \mathbf{x}(t) \\ &+ \left[\nabla_{\mathbf{u}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) + \nabla_{\mathbf{u}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^T \boldsymbol{\lambda}(t) \right]^T \delta \mathbf{u}(t) dt, \end{aligned}$$

wobei der Restterm durch

$$\left[\boldsymbol{\lambda}(t)^\top \delta \mathbf{x}(t) \right]_{t_0}^{t_T} = \boldsymbol{\lambda}(t_T)^\top \delta \mathbf{x}(t_T) - \boldsymbol{\lambda}(t_0)^\top \delta \mathbf{x}(t_0) = \boldsymbol{\lambda}(t_T)^\top \delta \mathbf{x}(t_T)$$

vereinfacht werden kann. Durch den bekannten Anfangszustand gilt die Randbedingung $\delta \mathbf{x}(t_0) = 0$. An diesem Zeitpunkt ist eine Variation nicht möglich.

Satz 2.3.1 (Fundamentallemma der Variationsrechnung). Sei $\mathbf{h} : [t_0, t_T] \mapsto \mathbb{R}^n$ eine stetige Funktion. Falls für alle stetigen Funktionen $\delta \mathbf{x} : [t_0, t_T] \mapsto \mathbb{R}^n$ die Bedingung

$$\int_{t_0}^{t_T} \mathbf{h}(t)^\top \delta \mathbf{x}(t) dt = 0$$

gilt, so ist $\mathbf{h}(t) = 0$ für alle $t \in [t_0, t_T]$.^a

^aEin Beweis ist in [AF66] (zitiert in [Gra18]) zu finden.

Mit dem *Fundamentallemma der Variationsrechnung* (Satz 2.3.1) entsteht die Forderung

$$\begin{aligned} \dot{\boldsymbol{\lambda}}(t) &= -\nabla_{\mathbf{x}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) - \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^\top \boldsymbol{\lambda}, & \forall t \in [t_0, t_T], \\ \mathbf{0} &= -\nabla_{\mathbf{u}} l(\mathbf{x}^*(t), \mathbf{u}^*(t), t) - \nabla_{\mathbf{u}} \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t))^\top \boldsymbol{\lambda}, & \forall t \in [t_0, t_T], \\ \boldsymbol{\lambda}(t_T) &= \nabla_{\mathbf{x}} V(\mathbf{x}^*(t_T), t_T), \end{aligned}$$

damit für alle zulässigen Variationen $\delta \tilde{J}(\epsilon, t) = 0$ gilt. Entsprechend muss die optimale Steuerung $\mathbf{u}^*(t)$ diese Bedingungen ebenfalls erfüllen. Im Gegensatz zur eingangs erwähnten Vorgehensweise über die KKT-Bedingungen ergibt sich der Lagrange-Multiplikator $\boldsymbol{\lambda}(t)$ mit diesem Ansatz über eine DGL und ist nicht über eine zusätzliche Optimierung zu ermitteln. Für Anwendungen mit variabler Endzeit ist (2.40) zusätzlich nach δt_T zu variieren. Dieser Aspekt wird nicht vertieft, es sei jedoch auf [Pap15, S. 187ff] verwiesen.

Definition 2.3.1 (Hamilton-Funktion). Die Hamilton-Funktion des unbeschränkten Optimierungsproblems (2.39) ist über

$$H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) := l(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^\top \mathbf{f}(\mathbf{x}, \mathbf{u})$$

unter Verwendung der Integralkostengewichtung $l(\cdot)$, des Modells (2.37b) und des Lagrange-Multiplikators $\boldsymbol{\lambda} \in \mathbb{R}^n$ definiert.

Mit der Definition der *Hamilton-Funktion* ergeben sich die notwendigen Optimalitätsbe-

dingungen 1. Ordnung in kompakterer Schreibweise zu

$$\dot{\mathbf{x}}^*(t) = \nabla_{\lambda} H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t), \quad (\text{Modell}) \quad (2.41a)$$

$$\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t), \quad (\text{adj. Modell}) \quad (2.41b)$$

$$\mathbf{0} = \nabla_{\mathbf{u}} H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t), \quad \forall t \in [t_0, t_T], \quad (\text{notw. Bedingung}) \quad (2.41c)$$

$$\mathbf{x}^*(t_0) = \mathbf{x}_0, \quad (\text{Anfangswert}) \quad (2.41d)$$

$$\boldsymbol{\lambda}(t_T) = \nabla_{\mathbf{x}} V(\mathbf{x}^*(t_T), t_T). \quad (\text{adj. Endwert}) \quad (2.41e)$$

Die Randbedingungen der adjungierten Zustände werden *Transversalitätsbedingungen* genannt. Das hier beschriebene Optimalsteuerungsproblem hat mit (2.41e) entsprechend eine Transversalitätsbedingung.

Durch zweifache Variation können die notwendigen Optimalitätsbedingungen 2. Ordnung zu

$$\nabla_{\mathbf{u}}^2 H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t) \succcurlyeq 0, \quad \forall t \in [t_0, t_T], \quad (2.42)$$

der sogenannten *Legendreschen Bedingung*, ermittelt werden. Genauere Ausführungen sind in [Pap15, S. 195] zu finden.

Neben den das Modell beschreibenden DGLs (2.41a) enthalten die Bedingungen einen zweiten Satz DGLs (2.41b), die als *adjungierte Differentialgleichungen* bezeichnet werden; die Lagrange-Multiplikatoren konsequenterweise als *adjungierte Zustände*. Die $2n$ DGLs können nicht unabhängig voneinander ausgewertet werden und bilden zusammen die *kanonischen Differentialgleichungen* der Problemstellung. Die Steuertrajektorie $\mathbf{u}(t)$ geht aus den m *Koppelgleichungen* (2.41c) hervor. Zusammen mit den Randwerten (2.41d) und (2.41e) bilden die kanonischen DGLs und die Koppelgleichungen ein Randwertproblem (RWP).

Es sei angemerkt, dass, wie bei den statischen Optimierungsproblemen (vgl. Abschnitt 2.2), die notwendigen Optimalitätsbedingungen 1. und 2. Ordnung eine (lokale) Lösung qualifizieren.²⁶ Bei einer nichtlinearen Problemstellung ist damit nicht sichergestellt, dass eine gefundene lokale Lösung die optimale (globale) Lösung darstellt (vgl. Abb. 2.4).

²⁶Für stückweise stetige Zusammenhänge sind weitere Bedingungen zu erfüllen [Pap15, S. 199ff].

Eingangsbeschränktes Optimalsteuerungsproblem

Wird dem unbeschränkten Optimalsteuerungsproblem die Nebenbedingung (2.37c) hinzugefügt, ergibt sich das eingangsbeschränkte Optimalsteuerungsproblem

$$\begin{aligned} \min_{\mathbf{u}(\cdot)} \quad & J(\mathbf{u}(\cdot)) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\mathbf{x}(t), \mathbf{u}(t), t) dt \\ \text{u. B. v.} \quad & \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \\ & \mathbf{b}_l(t) \leq \mathbf{u}(t) \leq \mathbf{b}_u(t). \end{aligned}$$

Durch die Beschränkung verliert die Koppelgleichung (2.41c) ihre Gültigkeit. Ist nämlich eine Stellgrößenbeschränkung aktiv, so kann die Ableitung der Hamiltonfunktion nach der Stellgröße nicht verschwinden.

Satz 2.3.2 (Pontrjaginsches Minimumprinzip). Sei $\{\mathbf{u}^*(t) \mid t \in [t_0, t_T]\}$ eine optimale Kontrolltrajektorie und sei $\{\mathbf{x}^*(t) \mid t \in [t_0, t_T]\}$ die durch

$$\dot{\mathbf{x}}^*(t) = \mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t)), \quad \mathbf{x}^*(t_0) = \mathbf{x}_0$$

gegebene Zustandstrajektorie. Ferner sei die Trajektorie $\{\boldsymbol{\lambda}(t) \mid t \in [t_0, t_T]\}$ der adjungierten Zustände über

$$\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t), \quad \boldsymbol{\lambda}(t_T) = \nabla_{\mathbf{x}} V(\mathbf{x}^*(t_T), t_T)$$

mit den Endkosten $V(\cdot)$ gegeben. Dann gilt für die optimale Kontrolltrajektorie

$$\mathbf{u}^*(t) = \arg \min_{\mathbf{u}} H(\mathbf{x}^*(t), \mathbf{u}, \boldsymbol{\lambda}(t), t), \quad \forall t \in [t_0, t_T].$$

Des Weiteren existiert eine Konstante C , mit der

$$H(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), t) = C, \quad \forall t \in [t_0, t_T]$$

gilt, sofern die Integralkosten $l(\cdot)$ sowie das System $\mathbf{f}(\cdot)$ zeitunabhängig sind.^a

^aEine Herleitung des Minimumprinzips über Variationsrechnung ist in [Kir04, S. 227ff] aufgeführt. In [Ber17, S. 435ff] wird das Minimumprinzip über *dynamische Programmierung* motiviert.

Die notwendigen Bedingungen 1. Ordnung für diesen Fall sind über das *Pontrjaginsche Minimumprinzip* in Satz 2.3.2 gegeben. Die Koppelgleichung wird demgemäß durch

$$\mathbf{u}^*(t) = \arg \min_{\mathbf{b}_l(t) \leq \mathbf{u} \leq \mathbf{b}_u(t)} H(\mathbf{x}^*(t), \mathbf{u}, \boldsymbol{\lambda}^*(t), t), \quad \forall t \in [t_0, t_T]$$

ersetzt. Die Legendresche Bedingung (2.42) behält ihre Gültigkeit.

Beschränktes Optimalsteuerungsproblem

Die Optimalitätsbedingungen für das eingangs aufgeführte Optimalsteuerungsproblem (2.37) werden über eine Kombination des Vorgehens für das unbeschränkte Optimalsteuerungsproblem mit den KKT-Bedingungen (2.24) hergestellt. Wie in (2.40) lassen sich die Nebenbedingungen mit in das Kostenfunktional aufnehmen, welches sich damit zu

$$\begin{aligned} \tilde{J}(\mathbf{u}(\cdot), \boldsymbol{\mu}(\cdot), \boldsymbol{\mu}_T) &= \tilde{V}(\mathbf{x}(t_T), \boldsymbol{\mu}_T, t_T) \\ &+ \int_{t_0}^{t_T} \tilde{l}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\mu}(t), t) + \boldsymbol{\lambda}^T(t)(\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) - \dot{\mathbf{x}}(t)) dt \end{aligned} \quad (2.44)$$

mit

$$\tilde{V}(\mathbf{x}, \boldsymbol{\mu}_T, t) := V(\mathbf{x}, t) + \boldsymbol{\mu}_{T,\mathcal{E}} \mathbf{c}_{T,\mathcal{E}}(\mathbf{x}, t) + \boldsymbol{\mu}_{T,\mathcal{I}} \mathbf{c}_{T,\mathcal{I}}(\mathbf{x}, t), \quad \boldsymbol{\mu}_{T,\mathcal{I}} \geq \mathbf{0} \quad (2.45)$$

und

$$\tilde{l}(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}, t) := l(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\mu}_{\mathcal{E}} \mathbf{c}_{\mathcal{E}}(\mathbf{x}, t) + \boldsymbol{\mu}_{\mathcal{I}} \mathbf{c}_{\mathcal{I}}(\mathbf{x}, t), \quad \boldsymbol{\mu}_{\mathcal{I}} \geq \mathbf{0} \quad (2.46)$$

ergibt. Neben den adjungierten Zuständen $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ enthält diese Formulierung die zusätzlichen Lagrange-Multiplikatoren $\boldsymbol{\mu}(t) := [\boldsymbol{\mu}_{\mathcal{E}}(t)^T, \boldsymbol{\mu}_{\mathcal{I}}(t)^T]^T$, $\boldsymbol{\mu}_{\mathcal{E}}(t) \in \mathbb{R}^{|\mathcal{E}|}$, $\boldsymbol{\mu}_{\mathcal{I}}(t) \in \mathbb{R}_+^{|\mathcal{I}|}$ für die integralen und $\boldsymbol{\mu}_T := [\boldsymbol{\mu}_{T,\mathcal{E}}^T, \boldsymbol{\mu}_{T,\mathcal{I}}^T]^T$, $\boldsymbol{\mu}_{T,\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}$, $\boldsymbol{\mu}_{T,\mathcal{I}} \in \mathbb{R}_+^{|\mathcal{I}|}$ für die terminalen Nebenbedingungen.

Definition 2.3.2 (Hamilton-Funktion mit Nebenbedingungen). Die Hamilton-Funktion des beschränkten Optimierungsproblems (2.37) ist über

$$\tilde{H}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, \boldsymbol{\mu}, t) := \tilde{l}(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}, t) + \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \mathbf{u})$$

unter Verwendung der Integralkostengewichtung (2.46), des Modells (2.37b), der adjungierten Zustände $\boldsymbol{\lambda} \in \mathbb{R}^n$ und der Lagrange-Multiplikatoren $\boldsymbol{\mu} := [\boldsymbol{\mu}_{\mathcal{E}}^T, \boldsymbol{\mu}_{\mathcal{I}}^T]^T$, $\boldsymbol{\mu}_{\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}$, $\boldsymbol{\mu}_{\mathcal{I}} \in \mathbb{R}_+^{|\mathcal{I}|}$ definiert.

Mit der modifizierten Hamilton-Funktion aus Definition 2.3.2 können die adjungierten Zustände wie beim eingangsbeschränkten Optimalsteuerungsproblem gehandhabt werden. Der erste Teil der Optimalitätsbedingungen lautet entsprechend

$$\begin{aligned} \dot{\mathbf{x}}^*(t) &= \nabla_{\boldsymbol{\lambda}} \tilde{H}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), \boldsymbol{\mu}^*(t), t), & (\text{Modell}) \\ \dot{\boldsymbol{\lambda}}(t) &= -\nabla_{\mathbf{x}} \tilde{H}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), \boldsymbol{\mu}^*(t), t), & (\text{adj. Modell}) \\ \mathbf{u}^*(t) &= \arg \min_{\mathbf{b}_l(t) \leq \mathbf{u} \leq \mathbf{b}_u(t)} \tilde{H}(\mathbf{x}^*(t), \mathbf{u}, \boldsymbol{\lambda}(t), \boldsymbol{\mu}^*(t), t), \quad \forall t \in [t_0, t_T], & (\text{Minimum-Prinzip}) \\ \mathbf{x}^*(t_0) &= \mathbf{x}_0, & (\text{Anfangswert}) \\ \boldsymbol{\lambda}(t_T) &= \nabla_{\mathbf{x}} \tilde{V}(\mathbf{x}^*(t_T), \boldsymbol{\mu}^* T, t_T). & (\text{adj. Endwert}) \end{aligned}$$

Die übrigen Lagrange-Multiplikatoren, welche im Zusammenhang mit den integralen und terminalen Nebenbedingungen eingeführt wurden, werden über die KKT-Bedingungen gehandhabt, womit sich der zweite Teil der Optimalitätsbedingungen zu

$$\begin{aligned}
 c_i(\mathbf{x}^*(t), \mathbf{u}^*(t), t) &= 0, & \forall t \in [t_0, t_T], & \quad i \in \mathcal{E}, \\
 c_i(\mathbf{x}^*(t), \mathbf{u}^*(t), t) &\leq 0, & \forall t \in [t_0, t_T], & \quad i \in \mathcal{I}, \\
 \mu_i^*(t) &\geq 0, & \forall t \in [t_0, t_T], & \quad i \in \mathcal{I}, \\
 \mu_i^*(t)c_i(\mathbf{x}^*(t), \mathbf{u}^*(t), t) &= 0, & \forall t \in [t_0, t_T], & \quad i \in \mathcal{I}, \\
 c_i(\mathbf{x}^*(t_T), t_T) &= 0, & & \quad i \in \mathcal{E}_T, \\
 c_i(\mathbf{x}^*(t_T), t_T) &\leq 0, & & \quad i \in \mathcal{I}_T, \\
 \mu_{T,i}^* &\geq 0, & & \quad i \in \mathcal{I}_T, \\
 \mu_{T,i}^*c_{T,i}(\mathbf{x}^*(t_T), t_T) &= 0, & & \quad i \in \mathcal{I}_T,
 \end{aligned}$$

ergibt [GD20, S. 206].

Die Legendresche Bedingung wird entsprechend zu

$$\nabla_{\mathbf{u}}^2 \tilde{H}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}(t), \boldsymbol{\mu}(t), t) \succcurlyeq 0, \quad \forall t \in [t_0, t_T]$$

angepasst.

2.3.2 Numerische Lösungsansätze (Übersicht)

Zur Lösung des Optimalsteuerungsproblems (2.37) gibt es grundsätzlich drei verschiedene Arten von Verfahrensfamilien: die *zustandsraumbasierten*, die *indirekten* und die *direkten Verfahren*. In den folgenden Ausführungen wird der unbeschränkte Fall (2.39) betrachtet. Auf die zusätzliche Berücksichtigung von Nebenbedingungen wird an dieser Stelle nicht eingegangen, da deren Behandlung Teil einer programmiertechnischen Umsetzung ist und separat vom gewählten Verfahren zu betrachten ist. Dasselbe gilt für andere algorithmische Feinheiten, wie beispielsweise eine unterschiedliche Diskretisierung von Steuer- und Zustandsvariablen.

Eine Übersicht sowie eine Bewertung der indirekten und direkten Verfahren wird in dem Übersichtsartikel [Rao10] gegeben. In den Vorlesungsskripten [GD20, GK12] werden Einblicke in mögliche rechentechnische Herangehensweisen gegeben.

Zustandsraumbasierte Verfahren

Die zustandsraumbasierten Verfahren beruhen auf der Lösung der *Hamilton-Jacobi-Bellman-Gleichung* (HJB), welche über

$$-\frac{\partial \tilde{J}(\mathbf{x}(t), t)}{\partial t} = \min_{\mathbf{u}} H(\mathbf{x}(t), \mathbf{u}, \boldsymbol{\lambda}(t), t) = \min_{\mathbf{u}} H(\mathbf{x}(t), \mathbf{u}, \nabla_{\mathbf{x}} \tilde{J}(\mathbf{x}(t), t), t), \quad (2.48)$$

$$\mathbf{x}(t_0) := \mathbf{x}_0, \quad t \in [t_0, T_T], \quad (2.49)$$

mit

$$\tilde{J}(\mathbf{x}, t) := \min_{\mathbf{u}(\cdot)} J(\mathbf{u}(\cdot); \mathbf{x}_0 := \mathbf{x}, t_0 := t)$$

gegeben ist. Es handelt sich hierbei um eine partielle DGL. In einer numerischen Umsetzung wäre diese zu diskretisieren und über ein geeignetes Optimierungsverfahren in zeitlicher Rückwärtsrichtung zu lösen. Für bestimmte Fälle ist eine analytische Lösung möglich. Das Verfahren hat seinen Ursprung in der *dynamischen Programmierung* von *Richard Bellman* und leidet aufgrund des rekursiven Charakters ebendieser unter dem *Fluch der Dimensionalität*. Je höher die Dimension des Problems ist (Anzahl der Zustände und Diskretisierungen), desto mehr Speicherplätze werden benötigt. Beispielsweise erfordert ein System mit drei Zuständen und hundert Diskretisierungen $10^2 \times 10^2 \times 10^2 = 10^6$ Speicherplätze [Kir04, S. 78]. Dies ist exemplarisch zu verstehen, die genaue Anzahl hängt von der rechentechnischen Umsetzung ab. Des Weiteren ist die Differenzierbarkeit der Kostenfunktion nicht garantiert.²⁷

Indirekte Verfahren

Die indirekten Verfahren arbeiten mit den im vorigen Abschnitt hergeleiteten notwendigen Optimalitätsbedingungen (2.15) und lösen das dazugehörige RWP numerisch. Lösungsansätze, die den indirekten Verfahren zugeordnet werden können, sind unter anderem das *indirekte Schießverfahren*, das *indirekte Mehrfachschießverfahren* [DBDW05] und das *Gradientenverfahren* [Kel62, S. 216ff], [Kir04, S. 331ff].²⁸ Diese Verfahren lösen die DGLs (2.41a) sowie (2.41b) zeitkontinuierlich durch ein Integrationsverfahren (vgl. Abschnitt 2.1.2) und leiten daraus eine geeignete Steuertrajektorie ab. Dieses Vorgehen wird iterativ bis zur Konvergenz wiederholt.

Beim *indirekten Schießverfahren* wird die Steuertrajektorie $\mathbf{u}(t)$ über das Minimum-Prinzip (vgl. Satz 2.3.2) ausgedrückt und stellt damit eine Parametrierung von $\mathbf{x}(t)$ sowie $\boldsymbol{\lambda}(t)$ dar. Unter Verwendung der übrigen Optimalitätsbedingungen (vgl. (2.15)) ist

²⁷Genauere Ausführungen finden sich in [Kir04, S. 86ff] sowie in [Ber17, S. 429ff]. Es existieren numerische Lösungsmethoden, z. B. [Ben83] (zitiert in [DBDW05]).

²⁸Weitere Verfahren sind die *Variation der Sensitivitäten* [Kir04, S. 343] sowie *indirekte Kollokationsverfahren*, z. B. [DW75].

folgendes Gleichungssystem

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (2.50a)$$

$$\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} H(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t), \quad \boldsymbol{\lambda}(t_0) = \boldsymbol{\lambda}_0, \quad (2.50b)$$

$$\mathbf{u}(t) = \arg \min_{\mathbf{u}} H(\mathbf{x}(t), \mathbf{u}, \boldsymbol{\lambda}(t), t), \quad \forall t \in [t_0, t_T], \quad (2.50c)$$

$$\mathbf{0} = \underbrace{\boldsymbol{\lambda}(t_T) - \nabla_{\mathbf{x}} V(\mathbf{x}(t_T), t_T)}_{=: \mathbf{R}(\mathbf{x}(t_T), \boldsymbol{\lambda}(t_T), t_T)} \quad (2.50d)$$

unter Variation von $\boldsymbol{\lambda}_0$ zu lösen. Eine numerische Umsetzung des Verfahrens besteht aus zwei sich iterativ wiederholenden Schritten. Im ersten Schritt wird das AWP der kanonischen DGLs (2.50a) und (2.50b) unter Verwendung des Minimum-Prinzips (2.50c) gelöst. Diese Gleichungen sind damit per Konstruktion des Algorithmus stets erfüllt. Im zweiten Schritt wird unter Verwendung der Transversalitätsbedingung (2.50d) ein neuer Anfangswert der adjungierten DGL ermittelt, beispielsweise über die Iterationsvorschrift

$$\boldsymbol{\lambda}_0^{(i+1)} = \boldsymbol{\lambda}_0^{(i)} - \alpha^{(i)} \left(\frac{\partial \mathbf{R}(\mathbf{x}^{(i)}(t_T), \boldsymbol{\lambda}^{(i)}(t_T), t_T)}{\partial \boldsymbol{\lambda}_0} \right)^{-1} \mathbf{R}(\mathbf{x}^{(i)}(t_T), \boldsymbol{\lambda}^{(i)}(t_T), t_T).$$

Die Schrittweite α^i kann über eine Liniensuche (vgl. Abschnitt 2.2.2) mit dem Ziel, $J(\cdot)$ zu minimieren, bestimmt werden. Das Verfahren ist in Algorithmus 2.3.1 skizziert.

Das Schießverfahren kommt entsprechend mit n zu optimierenden Variablen aus, da $\boldsymbol{\lambda}_0 \in \mathbb{R}^n$ gilt. Aus numerischer Sicht besteht jedoch die Schwierigkeit, eine geeignete Startschätzung für $\boldsymbol{\lambda}_0$ bereitzustellen. Die Vorwärtsintegration der kanonischen Gleichungen ist i. d. R. instabil und zeigt damit divergierendes Verhalten, welches umso stärker ausgeprägt ist, je weiter die Startschätzung vom optimalen Wert entfernt ist und je größer das Schießintervall $[t_0, t_T]$ in Relation zu den Zeitkonstanten des zu regelnden Systems ist [Rao10].²⁹

Abhilfe schafft das *indirekte Mehrfachschießverfahren* [Os69]. Bei diesem Verfahren wird das Schießintervall unter Verwendung von N Stützstellen in $N - 1$ Teilintervalle

$$t_0 \leq t_1 \leq \dots \leq t_k \leq \dots \leq t_{N-1} := t_T \quad (2.51)$$

mit $[t_k, t_{k+1}] \subset [t_0, t_T]$ unterteilt, auf die das zuvor beschriebene (Einfach-)Schießverfahren

²⁹Die Instabilität kann anhand eines linearen Systems nachvollzogen werden: Die DGL der adjungierten Zustände lautet dann $\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} l(\mathbf{x}(t), \mathbf{u}(t), t) - \mathbf{A}^T \boldsymbol{\lambda}(t)$ und hat damit dieselben Eigenwerte wie das lineare System, jedoch mit negiertem Vorzeichen. Damit enthält zwangsweise eines der beiden Systeme instabile Moden (es sei denn, sämtliche Eigenwerte liegen auf der imaginären Achse).

Algorithmus 2.3.1 (Indirektes Schießverfahren).

Initialisierung

- Startwert festlegen: $\boldsymbol{\lambda}_0^{(i)} := \boldsymbol{\lambda}_{\text{start}}$

for j **in** $0, 1, \dots$ **do**

- Integriere die kanonischen Gleichungen für $t \in [t_0, t_T]$ unter Verwendung des Minimum-Prinzips

$$\begin{aligned} \dot{\mathbf{x}}^{(i)}(t) &= \nabla_{\boldsymbol{\lambda}} H(\mathbf{x}^{(i)}(t), \mathbf{u}^{(i)}(t), \boldsymbol{\lambda}^{(i)}(t), t), & \mathbf{x}^{(i)}(t_0) &= \mathbf{x}_0, \\ \dot{\boldsymbol{\lambda}}^{(i)}(t) &= -\nabla_{\mathbf{x}} H(\mathbf{x}^{(i)}(t), \mathbf{u}^{(i)}(t), \boldsymbol{\lambda}^{(i)}(t), t), & \boldsymbol{\lambda}^{(i)}(t_0) &= \boldsymbol{\lambda}_0^{(i)}, \\ \mathbf{u}^{(i)}(t) &= \min_{\mathbf{u}} H(\mathbf{x}^{(i)}(t), \mathbf{u}, \boldsymbol{\lambda}^{(i)}(t), t) \end{aligned}$$

- Verbessere den Initialwert mit dem Newton-Verfahren. Suche die Nullstelle von (2.50d)

$$\boldsymbol{\lambda}_0^{(i+1)} = \boldsymbol{\lambda}_0^{(i)} - \alpha^{(i)} \left(\frac{\partial \mathbf{R}(\mathbf{x}^{(i)}(t_T), \boldsymbol{\lambda}^{(i)}(t_T), t_T)}{\partial \boldsymbol{\lambda}_0} \right)^{-1} \mathbf{R}(\mathbf{x}^{(i)}(t_T), \boldsymbol{\lambda}^{(i)}(t_T), t_T)$$

if Konvergenztest erfüllt **then**

- **stop** Beende den Algorithmus

end if

end for

angewendet wird. Dies führt zu dem Gleichungssystem

$$\dot{\mathbf{x}}_k(t) = \mathbf{f}(\mathbf{x}_k(t), \mathbf{u}_k(t)), \quad \mathbf{x}_k(t_k) = \mathbf{x}_k, \quad k = 0, 1, \dots, N-2, \quad (2.52a)$$

$$\dot{\boldsymbol{\lambda}}_k(t) = -\nabla_{\mathbf{x}} H(\mathbf{x}_k(t), \mathbf{u}_k(t), \boldsymbol{\lambda}_k(t), t), \quad \boldsymbol{\lambda}_k(t_k) = \boldsymbol{\lambda}_k, \quad k = 0, 1, \dots, N-2, \quad (2.52b)$$

$$\mathbf{u}_k(t) = \arg \min_{\mathbf{u}} H(\mathbf{x}(t), \mathbf{u}, \boldsymbol{\lambda}(t), t), \quad \forall t \in [t_0, t_T], \quad k = 0, 1, \dots, N-2, \quad (2.52c)$$

$$\mathbf{0} = \mathbf{x}_k(t_k) - \mathbf{x}_{k-1}(t_k), \quad k = 1, 2, \dots, N-2, \quad (2.52d)$$

$$\mathbf{0} = \boldsymbol{\lambda}_k(t_k) - \boldsymbol{\lambda}_{k+1}(t_k), \quad k = 0, 1, \dots, N-3, \quad (2.52e)$$

$$\mathbf{0} = \boldsymbol{\lambda}_{N-2}(t_T) - \nabla_{\mathbf{x}} V(\mathbf{x}_{N-2}(t_T), t_T). \quad (2.52f)$$

Neben dem Startwert $\boldsymbol{\lambda}_0$ der adjungierten DGL sind zusätzlich die Startwerte der kanonischen Gleichungen an den Intervallgrenzen zu ermitteln. Diese stellen sicher, dass die Trajektorien $\mathbf{x}(t)$ und $\boldsymbol{\lambda}(t)$, welche aus den Intervallen $\mathbf{x}_k(t)$ und $\boldsymbol{\lambda}_k(t)$ zusammengesetzt sind, einen stetigen Verlauf annehmen. Mathematisch geht diese Bedingung aus den zusätzlichen Kontinuitätsbedingungen (2.52d) und (2.52e) hervor. Die neuen Optimierungsvariablen haben zur Folge, dass für einen nicht-konvergierten Zustand die Trajektorien unstetig sind. Die Lösung ist damit erst verwendbar, wenn die zusätzlichen Randbedingungen so weit konvergiert sind, dass die Trajektorien einen (quasi-)stetigen Verlauf annehmen. Mit

dieser Methode sind $2Nn - 3n$ Variablen zu optimieren. Eine numerische Lösung kann über die Iterationsvorschrift

$$\mathbf{s}^{(i+1)} = \mathbf{s}^{(i)} - \alpha^{(i)} \left(\frac{\partial \mathbf{R}_{\text{MS}}(\mathbf{s}^{(i)}, t_T)}{\partial \mathbf{s}} \right)^{-1} \mathbf{R}_{\text{MS}}(\mathbf{s}^{(i)}, t_T) \quad (2.53)$$

erzielt werden. Dabei werden die Gleichungen (2.52d), (2.52e) und (2.52f) zu

$$\mathbf{0} = \mathbf{R}_{\text{MS}}(\mathbf{s}, t_T)$$

mit

$$\mathbf{s} := [\mathbf{x}_1, \dots, \mathbf{x}_{N-2}, \boldsymbol{\lambda}_0, \dots, \boldsymbol{\lambda}_{N-2}]^T$$

zusammengefasst. Eine Lösungsvorschrift ist in Algorithmus 2.3.2 skizziert.

Dem *Gradientenverfahren* liegt im Grundsatz dasselbe Gleichungssystem wie dem indirekten Schießverfahren zugrunde, mit dem Unterschied, dass die Transversalitätsbedingung per Konstruktion eingehalten wird, dafür aber die Steuertrajektorie nicht direkt aus dem Minimum-Prinzip hervorgeht. Da die Steuertrajektorie eine zeitkontinuierliche Funktion ist, ist ihre Bestimmung ein unendlich-dimensionales Optimierungsproblem. Durch Zeitdiskretisierung auf dem Intervall $[t_0, t_T]$ wird daraus eine endlich-dimensionale Problemstellung. Für jeden Zeitpunkt t_k wird eine Optimierungsvariable \mathbf{u}_k definiert. Die Steuertrajektorie ergibt sich durch eine Parametrierung

$$\mathbf{u}(t) = \boldsymbol{\psi}(t, \bar{\mathbf{u}}), \quad t \in [t_0, t_T] \quad \text{mit} \quad \bar{\mathbf{u}} = \{\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1}\}. \quad (2.54)$$

Beispielsweise könnte $\mathbf{u}(t)$ als lineare Interpolation der Optimierungsvariablen über die Zeit gewählt werden. Es ist das Gleichungssystem

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\psi}(t, \bar{\mathbf{u}})), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (2.55a)$$

$$\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} H(\mathbf{x}(t), \boldsymbol{\psi}(t, \bar{\mathbf{u}}), \boldsymbol{\lambda}(t), t), \quad \boldsymbol{\lambda}(t_T) = \nabla_{\mathbf{x}} V(\mathbf{x}(t_T), t_T), \quad (2.55b)$$

$$\mathbf{0} = \nabla_{\mathbf{u}} H(\mathbf{x}(t_k), \mathbf{u}_k, \boldsymbol{\lambda}(t_k), t_k), \quad k = 0, 1, \dots, N-1 \quad (2.55c)$$

zu lösen. Das Gradientenverfahren integriert zuerst (2.55a) in Vorwärtsrichtung, anschließend (2.55b) in Rückwärtsrichtung. Mit den dadurch berechneten Zuständen sowie den adjungierten Zuständen wird unter Verwendung von (2.55c) eine neue Schätzung der Steuertrajektorie

$$\mathbf{u}_k^{(i+1)} = \mathbf{u}_k^{(i)} - \alpha^{(i)} \nabla_{\mathbf{u}} H(\mathbf{x}^{(i)}(t_k), \mathbf{u}_k^{(i)}, \boldsymbol{\lambda}^{(i)}(t_k), t_k), \quad \alpha^{(i)} > 0, \quad k = 0, 1, \dots, N-1 \quad (2.56)$$

erzeugt, die die Hamilton-Funktion minimiert bzw. dafür sorgt, dass diese in der nächsten Iteration kleiner wird. Die Hamilton-Funktion ist bezüglich \mathbf{u} konvex; es wird in Richtung des negativen Gradienten iteriert.³⁰ Die Anzahl der Optimierungsvariablen ergibt sich aus

³⁰Dies gilt unter der Bedingung, dass die integralen Kosten $l(\cdot)$ konvex sind.

Algorithmus 2.3.2 (Indirektes Mehrfachschießverfahren).

Initialisierung

- Startwerte festlegen:

$$\begin{aligned}\mathbf{x}_k^{(i)} &:= \mathbf{x}_{k,\text{start}}, & \forall k \in [1, N - 2], \\ \boldsymbol{\lambda}_k^{(i)} &:= \boldsymbol{\lambda}_{k,\text{start}}, & \forall k \in [0, N - 2]\end{aligned}$$

- Es gilt:

$$\mathbf{x}_0^{(i)} := \mathbf{x}_0$$

for j **in** $0, 1, \dots$ **do**

- Integriere die kanonischen Gleichungen für die Intervalle $t \in [t_k, t_{k+1}]$ und $k \in [0, N - 2]$ unter Verwendung des Minimum-Prinzips

$$\begin{aligned}\dot{\mathbf{x}}_k^{(i)}(t) &= \nabla_{\boldsymbol{\lambda}} H(\mathbf{x}_k^{(i)}(t), \mathbf{u}^{(i)}(t), \boldsymbol{\lambda}_k^{(i)}(t), t), & \mathbf{x}_k^{(i)}(t_k) &= \mathbf{x}_k^{(i)}, \\ \dot{\boldsymbol{\lambda}}_k^{(i)}(t) &= -\nabla_{\mathbf{x}} H(\mathbf{x}_k^{(i)}(t), \mathbf{u}^{(i)}(t), \boldsymbol{\lambda}_k^{(i)}(t), t), & \boldsymbol{\lambda}_k^{(i)}(t_k) &= \boldsymbol{\lambda}_k^{(i)}, \\ \mathbf{u}^{(i)}(t) &= \min_{\mathbf{u}} H(\mathbf{x}_k^{(i)}(t), \mathbf{u}, \boldsymbol{\lambda}_k^{(i)}(t), t)\end{aligned}$$

- Verbessere die Lösung mit (2.53)

$$\mathbf{s}^{(i+1)} = \mathbf{s}^{(i)} - \alpha^{(i)} \left(\frac{\partial \mathbf{R}_{\text{MS}}(\mathbf{s}^{(i)}, t_T)}{\partial \mathbf{s}} \right)^{-1} \mathbf{R}_{\text{MS}}(\mathbf{s}^{(i)}, t_T)$$

if Konvergenztest erfüllt **then**

- **stop** Beende den Algorithmus

end if

end for

der Anzahl der Stützpunkte der Interpolation und der Länge des Eingangsvektors \mathbf{u} zu Nm . Dieses Verfahren bildet die Basis des in Kapitel 4 präsentierten MPC-Algorithmus und wird dort eingehend vorgestellt.

Direkte Verfahren

Die direkten Verfahren basieren auf der Umwandlung des (unendlich dimensionalen) DP (2.37) in ein (endlich dimensionales) statisches Optimierungsproblem durch Diskretisierung. Gelöst wird das umgewandelte Problem mit Ansätzen aus der statischen Optimierung (vgl. Abschnitt 2.2). In Abgrenzung zu den vorherigen Verfahren wird von *direkter* Lösung

gesprochen, da die Lösung des Optimierungsproblems ohne den Umweg über die Optimalitätsbedingungen erfolgt. Lösungsansätze sind unter anderem das *direkte Schießverfahren*, das *direkte Mehrfachschießverfahren* und das *direkte Kollokationsverfahren* [DBDW05].

Beim *direkten Schießverfahren* wird auf dem Zeitintervall $[t_0, t_T]$ wie in (2.51) diskretisiert. Für jeden Zeitpunkt t_k wird eine Optimierungsvariable \mathbf{u}_k für die Steuergröße definiert. Die Steuertrajektorie geht wie beim Gradientenverfahren aus einer Interpolation (2.54) hervor. Es ergibt sich das statische Optimierungsproblem

$$\begin{aligned} \min_{\bar{\mathbf{u}}} \quad & J(\bar{\mathbf{u}}, t) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\mathbf{x}(t), \boldsymbol{\psi}(t, \bar{\mathbf{u}}), t) dt \\ \text{u. B. v.} \quad & \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\psi}(t, \bar{\mathbf{u}})), \quad \mathbf{x}(t_0) = \mathbf{x}_0. \end{aligned}$$

Der Gradient der Kostenfunktion kann beispielsweise über Sensitivitäts-DGLs erhalten werden.³¹ Im unbeschränkten Fall entsprechen diese der adjungierten DGL (2.41b), wodurch sich eine Iterationsvorschrift analog zu (2.56) ergibt. Das direkte Schießverfahren lässt sich entsprechend identisch zum Gradientenverfahren umsetzen. Es gibt daneben noch andere Formulierungen, dazu sei auf [GD20, S. 230ff] verwiesen. Grundsätzlich ist das direkte Schießverfahren ein *sequenzielles Verfahren*, in dem abwechselnd das AWP gelöst und die Steuertrajektorie verbessert werden.

Das *direkte Mehrfachschießverfahren* diskretisiert neben der Steuertrajektorie die Zustandstrajektorie und nimmt zusätzlich die Intervallgrenzen mit in das Optimierungsproblem auf. Die zu optimierenden Intervallgrenzen seien über

$$\bar{\mathbf{x}} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N-1}\}$$

gegeben. Der Anfangszustand \mathbf{x}_0 ist bekannt und wird nicht optimiert. Die Lösung des statischen Optimierungsproblems

$$\min_{\bar{\mathbf{u}}, \bar{\mathbf{x}}} \quad J(\bar{\mathbf{u}}, \bar{\mathbf{x}}, t) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\boldsymbol{\psi}(t, \bar{\mathbf{x}}), \boldsymbol{\psi}(t, \bar{\mathbf{u}}), t) dt \quad (2.58a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}_k(t) = \mathbf{f}(\mathbf{x}_k(t), \boldsymbol{\psi}(t, \bar{\mathbf{u}})), \quad \mathbf{x}(t_k) = \mathbf{x}_k, \quad (2.58b)$$

$$\mathbf{x}_k = \mathbf{x}_{k-1}(t_k), \quad k = 1, 2, \dots, N - 1 \quad (2.58c)$$

liefert die Stützstellen der Eingangsdiskretisierung $\bar{\mathbf{u}}$ sowie der Zustandsdiskretisierung $\bar{\mathbf{x}}$. Wie beim indirekten Mehrfachschießverfahren enthält dieses Optimierungsproblem ebenfalls eine Gleichungsnebenbedingung (2.58c), welche die Kontinuität an den Grenzen der Teilintervalle erzwingt. Die Lösung der DGL ist Teil des Optimierungsverfahrens; es wird von einem *simultanen Verfahren* gesprochen. Sie geschieht analog zum direkten (Einfach-)Schießverfahren über numerische Integrationstechniken.

Anders ist dies bei den *direkten Kollokationsverfahren*³². Diese überführen die Lösung

³¹Eine Einführung zu Sensitivitäts-DGLs wird in [GD20, S. 187ff] gegeben.

³²In der Literatur ist für das direkte Kollokationsverfahren auch die Bezeichnung *Volldiskretisierung* zu

der DGL an sich ebenfalls in das Optimierungsproblem. Exemplarisch sei dies anhand der Quadratur mit dem expliziten Euler-Verfahren veranschaulicht: Das Optimierungsproblem ergibt sich zu

$$\begin{aligned} \min_{\bar{\mathbf{u}}, \bar{\mathbf{x}}} \quad & J(\bar{\mathbf{u}}, \bar{\mathbf{x}}, t) = V(\mathbf{x}(t_T), t_T) + \int_{t_0}^{t_T} l(\boldsymbol{\psi}(t, \bar{\mathbf{x}}), \boldsymbol{\psi}(t, \bar{\mathbf{u}}), t) dt \\ \text{u. B. v.} \quad & \mathbf{x}_{k+1} = \mathbf{x}_k + (t_{k+1} - t_k) \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), \quad k = 0, 1, \dots, N - 2 \end{aligned}$$

und enthält als eine Nebenbedingung die Quadraturformel des expliziten Euler-Verfahrens (2.9). Die Kontinuität der Teilintervalle ist in dieser Formulierung implizit enthalten und muss nicht separat gefordert werden. Typischerweise sind mit *Kollokationsverfahren* jedoch Verfahren gemeint, die die DGL stückweise über ein Polynom approximieren.³³ Die Bestimmung der Polynomkoeffizienten $\bar{\mathbf{v}}$ wird ebenfalls Teil des Optimierungsproblems. Durch diesen Ansatz wird der Verfahrensfehler der Diskretisierung kleiner. Des Weiteren vereinfacht sich dadurch die Berechnung der Ableitungen erster und zweiter Ordnung. Letztere werden für die gängigen Lösungsansätze für Probleme dieser Art benötigt. Im Vergleich zum direkten Mehrfachschießverfahren ist die Hesse-Matrix des Optimierungsproblems zwar größer, dafür jedoch dünner besetzt (vgl. [GD20, S. 238] vs. [GD20, S. 242]).

finden [MHM09].

³³Eine Approximation über gewöhnliche Polynome ist in [THE75] zu finden. In [SB92] werden *Hermite-Polynome* und in [DBDW05] *Gauß-Legendre-Polynome* zur Kollokation verwendet.

3 Luftpfadmodell

Das Konzept der temperaturgeführten Motorregelung in Kapitel 5 besteht darin, unter Verwendung der Aktuatoren des Luftpfades (vgl. Abb. 1.6) – des WG und der ETV – die Abgastemperatur zu beeinflussen. Zur Regelung wird eine MPC (vgl. Abschnitt 4) eingesetzt, welche prinzipbedingt ein Modell der Regelstrecke erfordert. An dieses Streckenmodell werden mehrere Anforderungen gestellt. Zum einen müssen die von dem Regler geforderten mathematischen Voraussetzungen eingehalten werden: Es ist ein stetig differenzierbares (nichtlineares) Zustandsraummodell (2.1a) gefordert. Die stetige Differenzierbarkeit ist insofern wichtig, weil die modellprädiktive Regelung über ein gradientenbasiertes Optimierungsverfahren berechnet wird und diese Verfahren stetige Gradienten benötigen. Zum anderen ist eine aus rechentechnischer Sicht möglichst anspruchslose Formulierung von zentraler Bedeutung. Dies gilt insbesondere vor dem Hintergrund, dass die numerische Lösung der Zustandsraumbeschreibung der Regelstrecke den maßgeblichen Anteil an der von der MPC beanspruchten Rechenzeit inne hat, wie es im Rahmen der Validierung der Regelung in Abschnitt 5.3.2 deutlich wird. Für den beanspruchten RAM gilt dies ebenso, jedoch in abgeschwächter Form. Neben diesen Kriterien, deren Nichteinhaltung eine praktische Umsetzung der MPC verhindern würde, ist schließlich ein Modell mit möglichst hoher Genauigkeit wünschenswert, um eine hohe Regelgüte zu erhalten. Das Streckenmodell sollte mindestens die relevanten Dynamiken beschreiben, sodass das qualitative Verhalten dem der Regelstrecke entspricht. Zusammengefasst ergeben sich an die Modellbeschreibung mithin die folgenden vier Anforderungen: stetige Differenzierbarkeit, geringer rechentechnischer Aufwand, niedriger Speicherverbrauch und eine möglichst hohe Genauigkeit.

Inhalt dieses Kapitels ist die in [TT22] präsentierte Entwicklung und rechentechnische Umsetzung eines Luftpfadmodells, welches die genannten Anforderungen in adäquater Weise moderiert. In Abschnitt 3.1 wird der Luftpfad mathematisch beschrieben und es werden die zur Bestimmung der Parameter zu lösenden Optimierungsprobleme erläutert. Die Parameterbestimmung geschieht unter Verwendung von stationären sowie dynamischen Messungen (vgl. Abschnitt 3.2.1), die aus einem CFD-Simulationsmodell (engl.: *Computational Fluid Dynamics*) des Motors gewonnen werden. In Abschnitt 3.2 folgen die Beschreibung der numerische Parameterbestimmung und die Beurteilung der erhaltenen Modellgüte. Des Weiteren wird untersucht, welche Sensoren zur Unterstützung des Modells geeignet sind. Im letzten Abschnitt 3.3 wird die Erzeugung von für die ECU geeignetem C-Code beschrieben sowie dieser Code hinsichtlich dessen Echtzeitfähigkeit bewertet.

3.1 Modellbildung

Ziel der Modellbildung ist es, den in Abb. 1.6 dargestellten Luftpfad des Deutz TCD 12.0 V6 mathematisch in Form einer nichtlinearen Zustandsraumdarstel-

lung

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), & \mathbf{x}(0) := \mathbf{x}_0 \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \end{cases}$$

mit $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ und $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$ zu beschreiben (vgl. Abschnitt 2.1.1). Es wird eine Beschreibung von $\mathbf{f}(\cdot)$ sowie $\mathbf{g}(\cdot)$ gesucht, die den eingangs erwähnten Anforderungen gerecht wird.

Im folgenden Abschnitt werden zunächst die allgemeinen Modellierungsansätze genannt, welche insbesondere vor dem Hintergrund der eingangs formulierten Anforderungen motiviert werden. In den nachfolgenden Abschnitten werden die einzelnen Teilmodelle beschrieben. Inhaltlich sortiert sind diese Teilmodelle nach den Komponenten bzw. Bauteilen des Luftpfades.

3.1.1 Modellierungsansatz

Um der Rechenzeitanforderung gerecht zu werden, werden sogenannte *Mittelwertmodelle* verwendet. Die Alternative wäre eine arbeitsspielaufgelöste Modellierung: Bei einer zeitlich hoch aufgelösten Betrachtung der Drücke und Temperaturen sind die Pulsationen durch das Arbeitsspiel des Motors sichtbar. Der Motor arbeitet nach dem Viertaktprinzip, was bei den vorhandenen sechs Zylindern bedeutet, dass drei mal pro Motorumdrehung Frischluft aus den Einlassfluten in die Zylinder strömt bzw. Abgas aus den Zylindern in die Auslassfluten gedrückt wird. Bei der niedrigsten Leerlaufdrehzahl von 600 U/min haben diese Pulsationen demgemäß eine Frequenz von 30 Hz bzw. eine Periodendauer von 33 ms. Für eine numerische Beschreibung über eine DGL wäre mit einer Integrationsschrittweite von unter 3,3 ms zur Lösung ebenjener zu rechnen. Mittelwertmodelle ignorieren hingegen das Arbeitsspiel des Motors und betrachten die darin stattfindenden Prozesse als kontinuierlich, wodurch größere Integrationsschrittweiten ermöglicht werden, welche wiederum einen geringeren Rechenaufwand implizieren. Es wird demgemäß mit effektiven Massenströmen, Drücken, Temperaturen und Emissionen gerechnet. Ebenfalls rechenaufwandsreduzierend wirkt sich eine möglichst geringe Komplexität der zur Modellbeschreibung verwendeten mathematischen Zusammenhänge aus. Rechentechnisch aufwendig zu berechnen – und demnach weniger geeignet – sind die Exponentialfunktion, der Logarithmus, die Potenzfunktion (mit nicht-ganzzahligem Exponenten) und Lookup-Tabellen. Für nähere Informationen sei auf Tab. 3.4 verwiesen. Entsprechend sind polynomiale Zusammenhänge möglichst geringer Ordnung der präferierte Modellierungsansatz. Polynome sind mathematisch glatte Funktionen und bieten sich damit ebenfalls als Beschreibungsform zur Gewährleistung der geforderten stetigen Differenzierbarkeit an.

Die Beanspruchung des RAM hängt maßgeblich mit der Anzahl der Modellzustände und damit mit der Anzahl der DGLs zusammen. Der modellprädiktive Regler berechnet zur Bestimmung der *adjungierten Zustände* in Gleichung (4.16c) die Gradienten der DGLs (2.1a) nach den Zuständen und damit die Systemmatrix \mathbf{A} (vgl. lineare Zustandsraumbescrei-

bung (2.4)). Die Dimension der Systemmatrix entspricht der Anzahl der Zustände. Dazu sei folgende Überlegung angestellt: Wenn das entwickelte Modell zehn Differentialgleichungen aufwiese, hätte die Systemmatrix einhundert Elemente. Allein diese mit einfacher Genauigkeit¹ zu speichern, würde 0,4 kB Speicherplatz beanspruchen und damit bereits 20 % des für die MPC verfügbaren Stacks (vgl. Tab. 1.2). Eine geringe Anzahl an Zuständen wirkt sich ebenfalls förderlich auf die benötigte Rechenleistung aus, da die DLGs zur Berechnung der Regelung mehrfach numerisch zu integrieren sind (vgl. Abschnitt 4.4.2). Um die Anzahl der Zustände möglichst gering zu halten, wird auf eine ortsabhängige Beschreibung verzichtet und lediglich mit konzentrierten Parametern gearbeitet. Die Drücke und Temperaturen in den einzelnen Komponenten werden als homogen angenommen. Unter diesen Maßgaben ergibt sich die in Abb. 3.1 dargestellte Modellvorstellung. Die in Abb. 1.6 dargestellten

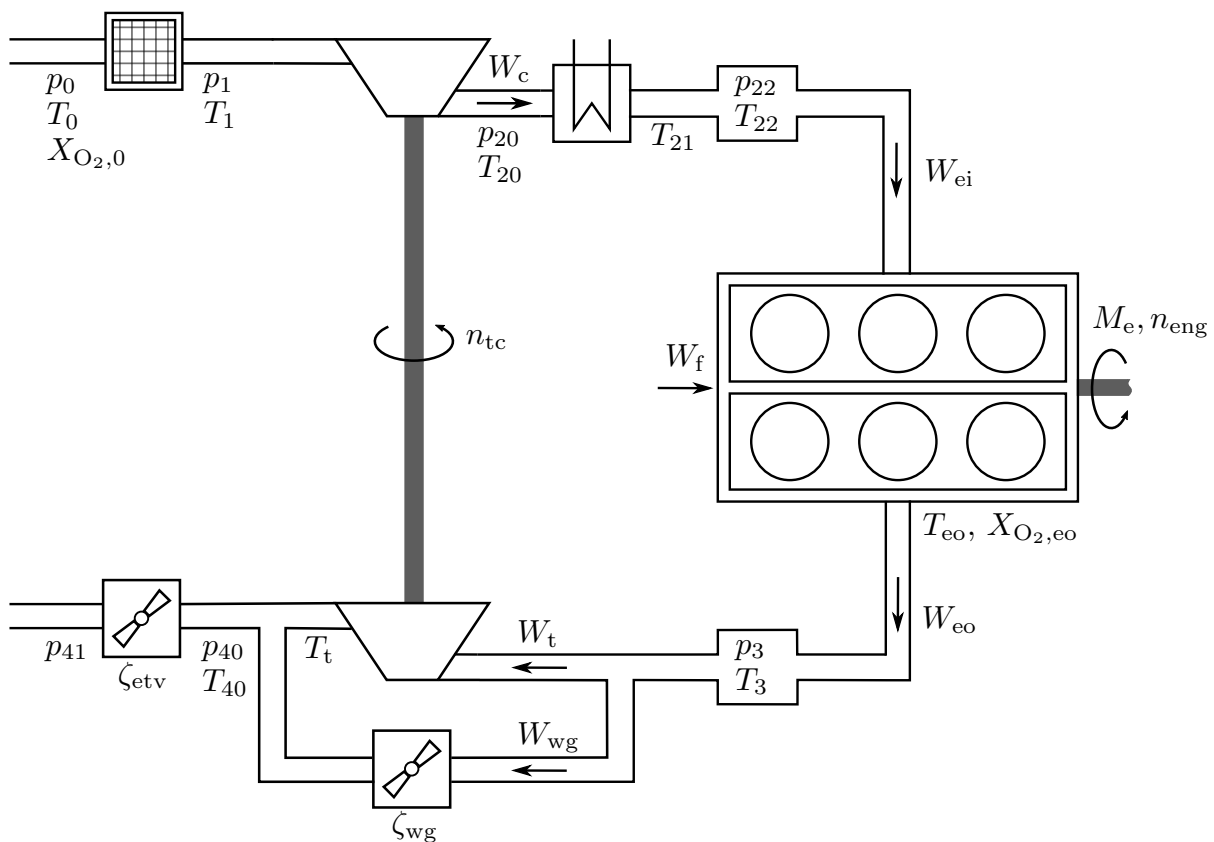


Abb. 3.1: Modellvorstellung des *Deutz TCD 12.0 V6* Dieselmotors.

Komponenten / Bauteile bleiben erhalten. Zwischen dem Ladeluftkühler und dem Motorblock sowie zwischen dem Motorblock und der Turbine werden Volumina eingefügt. Die Volumina dienen dazu, das Befüllen / Entleeren der Fluten zu beschreiben. Des Weiteren wird auf die getrennte Beschreibung ebenjener verzichtet; die Einlass- und Auslassfluten werden zu jeweils einer Flut zusammengefasst. Das WG wirkt sich durch diese Änderung formal auf beide Zuführungen der Zwillingstromturbine aus.

¹Eine Fließkommazahl einfacher Genauigkeit nach dem *IEEE Standard for Floating-Point Arithmetic (IEEE 754)* besteht aus 32 Bit.

Die DGLs und damit die Zustände werden über die in der Modellvorstellung enthaltenen *Speicher* identifiziert. Ein Speicher ist ein Bauelement, welches Energie oder Materie aufnehmen kann.² Im Kontext der Modellvorstellung des Luftpfades sind derartige Speicher die Aktuatorpositionen $(\zeta_{wg}, \zeta_{etv})$, die Turboladerdrehzahl (n_{tc}) und der Gaszustand der Einlassflut (p_{22}, T_{22}) und der Auslassflut (p_3, T_3) . Die Fluten werden über zwei Zustände beschrieben. Über die ideale Gasgleichung ist folgender Zusammenhang gegeben:

$$pV = mRT. \quad (3.1)$$

Die Gaskonstante R ist eine Stoffeigenschaft. Diese Gleichung hat drei konzentrierte Parameter: den Druck p , die Masse m und die Temperatur T . Mit zweien dieser Parameter ist der Gaszustand vollständig beschrieben, die dritte (extensive) Größe ergibt sich über die Gasgleichung [BK16, S. 22]. Das hier präsentierte Modell verwendet den Druck und die Temperatur aufgrund ihrer messtechnisch direkten Zugänglichkeit. Dies vereinfacht die Parameteridentifikation anhand von Messdaten, da keine Umrechnung über (3.1) erforderlich ist. Auch Zustandsbeobachter³ profitieren von dieser Wahl, da dort ebenfalls der Messwert zur Korrektur der Zustände umzurechnen wäre. Die Speicher werden mit anderen Bauteilen, im Folgenden *Koppler* genannt, zu einem Gesamtsystem verknüpft.⁴ Der Luftpfad beherbergt folgende Koppler (in der Reihenfolge der Luftströmung aufgezählt): Luftfilter, Kompressor, Ladeluftkühler, Motorblock, Turbine, Wastegate und Abgasklappe. Formal befände sich zwischen jedem Koppler ein Speicher. Um die Anzahl der Zustände klein zu halten, wird für die Massenströme stets Stationarität angenommen, was bedeutet, dass diese folgender Beziehung gehorchen:

$$W_c = W_{ei} = W_{eo} - W_f = W_t + W_{wg} - W_f.$$

Entsprechend wird für die Bauteile vor dem Motorblock W_{ei} angenommen und für die Bauteile dahinter W_{eo} . Unter Berücksichtigung dieser Annahme kann beispielsweise der Druck hinter dem Kompressor, der eigentlich über eine eigene DGL beschrieben würde, unter Verwendung des Massenstroms W_{ei} und des Drucks p_{22} aus dem Strömungswiderstand des Ladeluftkühlers berechnet werden (vgl. Gleichung (3.21)). Die Massenströme W_c, W_t, W_{wg} werden dennoch modelliert, da diese zur Beschreibung der DGLs der eingefügten Speicher erforderlich sind.⁵ Zur Notation sei angemerkt, dass die Massenströme das Formelzeichen W anstatt \dot{m} verwenden, um eine Verwechslung mit dem zeitlichen Differential der Gasmasse in einem Volumenelement auszuschließen. Die Vernachlässigung der Volumina zwischen den Kopplern hat aus rechentechnischer Sicht einen positiven Seiteneffekt: Die Volumina sind als verhältnismäßig klein anzunehmen. Kleine Speicher führen zu

²In [Ise08] ist ein Speicher weiter gefasst. Dort handelt es sich um ein Bauteil, welches eine *Quantität* aufnehmen kann, womit die Größen Energie, Materie oder Information gemeint sind.

³Das Konzept eines Zustandsbeobachters wird in [Lue71] vorgestellt. In der Praxis sind das Kálmán-Filter und dessen Derivate verbreitet. Hierzu sei auf [Wen11] verwiesen.

⁴In [Ise08] sind mit *Kopplern* lediglich Bauelemente gemeint, die eine Quantität (bspw. einen Massenstrom) aufnehmen und ihn in gleicher Weise wieder abgeben. Daneben existieren noch Wandler, die eine Energieform in eine andere umwandeln. Im Luftpfad ist die Turbine ein solcher Wandler; es wird thermische Energie aus dem Abgas in Bewegungsenergie umgewandelt.

⁵Andernfalls wären diese stets Null; vgl. (3.26) sowie (3.25).

betragsmäßig großen Eigenwerten, welche sich nach negativ auf die benötigte Abtastzeit zur Lösung der DGLs auswirken (vgl. Abschnitt 2.1.3). Dass die Eigenwerte betragsmäßig groß werden, wird durch Vergleich von (3.26) mit der als PT1-Filter modellierten WGDynamik (3.38) ersichtlich. Das Speichervolumen V_{22} kann als Zeitkonstante der DGL interpretiert werden. Eine weitere modellierungstechnische Vereinfachung wird erreicht, indem die Temperaturen der Volumina vor und hinter dem Motor mit der Temperatur des in das jeweilige Volumen einströmenden Gases gleichgesetzt werden. Die Motivation hinter dieser Entscheidung ist folgende: Selbst im Leerlauf beträgt der Volumenstrom des Motors ca. 50 Liter/s. Bei einem Volumen der Fluten in einer Größenordnung von 5 Litern bedeutet dies, dass das in den Fluten enthaltene Gas alle 100 ms vollständig ausgetauscht wird und damit die Temperatur in den Volumina ebenso schnell der Temperatur des einströmenden Gases entspricht.⁶ Bei höheren Drehzahlen geschieht dies entsprechend schneller. Mit derselben Argumentation wie für die Mittelwertmodellierung werden diese Effekte vernachlässigt. Die Anzahl der zur Beschreibung des Modells erforderlichen Zustände reduziert sich durch diese Maßnahme allerdings gegenüber der zur Beschreibung der Speicher erforderlichen Anzahl insgesamt nicht, da im Gegenzug die artifiziellen Zustände T_{21} , T_{40} und T_{cyl} eingefügt werden. Zum Beseitigen von algebraischen Schleifen werden die als PT1-Filter beschriebenen Temperaturen T_{21} sowie T_{40} eingeführt. Deren Zeitkonstanten werden so gewählt, dass sie einerseits keine sichtbaren Auswirkungen auf das Gesamtsystem haben und andererseits die numerische Integration nicht dominieren.⁷ Der Zustand T_{cyl} ist zur Beschreibung der Temperatur des aus dem Motorblock austretenden Gases erforderlich.⁸ Damit ergibt sich insgesamt der Zustandsvektor

$$\mathbf{x} = \left[p_{22} \quad p_3 \quad T_{21} \quad T_{\text{cyl}} \quad T_{40} \quad n_{\text{tc}} \quad \zeta_{\text{wg}} \quad \zeta_{\text{etv}} \right]^T. \quad (3.2)$$

Die Eingänge des Modells sind die Aktuatorsollwerte $(r_{\text{wg}}, r_{\text{etv}})$, welche die regelungstechnischen Eingänge des Systems darstellen, und die Randbedingungen. Letztere sind über die Systemgrenzen des Luftpfades – den Umgebungsdruck (p_0), die Umgebungstemperatur (T_0) und die Motordrehzahl (n_{eng}) – sowie die zeitvariablen Stützstellen \dot{Q}_f^* , q_{inj}^* und $\Delta p_{3,22}^*$ gegeben.⁹ Die Stützstellen haben ihren Ursprung in der Beschreibung der stationären Abgastemperatur. Die bei der Verbrennung des Kraftstoffs in den Zylindern freiwerdende Energie wird zu ca. 45 % in Bewegungsenergie umgewandelt.¹⁰ Die verbliebene Energie wird über die Zylinderwände oder das Abgas abgeführt. Zur Bestimmung der Abgastemperatur ist der im Abgas verbleibende Anteil von Bedeutung. Die geforderten Genauigkeiten werden über dessen Beschreibung mithilfe eines außerhalb des eigentlichen Luftpfadmo-

⁶In dieser vereinfachten Betrachtung werden die thermischen Kopplungen zu der Verrohrung vernachlässigt.

⁷In Abschnitt 3.3.3 wird gezeigt, dass die dominanten Eigenwerte – und damit die dominanten Zeitkonstanten – für die Abtastrate der numerischen Integration entscheidend sind.

⁸Die Temperatur ist in Abbildung 3.1 nicht dargestellt. Sie kann am ehesten in den Zylindern lokalisiert werden.

⁹Der massenbezogene Sauerstoffanteil der Umgebungsluft ($X_{\text{O}_2,0}$) ist konstant und wird daher als fester Modellparameter betrachtet.

¹⁰Der Deutz TCD 12.0 V6 hat einen effektiven Bestpunktwirkungsgrad von 43,4 % (vgl. Tab. D.1).

dells evaluierten *künstlichen neuronalen Netzes* (KNN) erreicht.¹¹ Es wird jedoch anstelle der Abgastemperatur die dem Abgas zugeführte Heizleistung in das Modell gegeben. Auf diese Weise sind über geeignete Linearisierungen die Kopplungen des Luftpfades auf die Abgastemperatur in dem Luftpfadmodell enthalten. Dies ist insofern wichtig, als die Abgastemperatur eine Regelgröße des temperaturgeführten Motors ist. Der Eingangsvektor ist über

$$\mathbf{u} = \left[n_e \quad q_{inj} \quad \dot{Q}_f^* \quad q_{inj}^* \quad \Delta p_{3,22}^* \quad T_0 \quad p_0 \quad p_{41} \quad r_{wg} \quad r_{etv} \right]^T \quad (3.3)$$

gegeben.

In den folgenden Abschnitten wird der Luftpfad komponentenweise beschrieben. Die Reihenfolge der Beschreibung orientiert sich dabei an der Strömungsrichtung des Gases; es wird beim Luftfilter begonnen. Die Gleichungen der Komponenten werden derart formuliert, dass die von den DGLs geforderten Größen berechnet werden. Die Beschreibung einer jeden Komponente besteht aus zwei Teilen. Im ersten Teil erfolgt jeweils die mathematische Beschreibung, im zweiten Teil wird das zu lösende Optimierungsproblem zur Parameterermittlung beschrieben. Für die Parameterermittlung werden ausschließlich messtechnisch zugängliche Größen verwendet. Das bedeutet, dass, obwohl CFD-Simulationsdaten verwendet werden, dasselbe Vorgehen ebenso anhand von Prüfstandsmessungen möglich ist. Die in den Optimierungsproblemen verwendeten Fehlernormen sind in Abschnitt A.2 aufgeführt. Die Durchführung der Identifikation wird in Abschnitt 3.2 beschrieben.

Um die Notation der Gleichungen möglichst kompakt zu gestalten, werden die Gleichungen soweit möglich auf die *korrigierten* und *reduzierten* Größen

$$\begin{aligned} n_{t,c,red} &= n_{tc} \sqrt{\frac{1}{T_3}}, & W_{t,red} &= W_t \frac{\sqrt{T_3}}{p_3}, \\ n_{t,cor} &= n_{tc} \sqrt{\frac{T_{3,ref}}{T_3}}, & W_{t,cor} &= W_t \frac{\sqrt{T_3/T_{3,ref}}}{p_3/p_{3,ref}}, \\ n_{c,cor} &= n_{tc} \sqrt{\frac{T_{1,ref}}{T_1}}, & W_{c,cor} &= W_c \frac{\sqrt{T_1/T_{1,ref}}}{p_1/p_{1,ref}} \end{aligned} \quad (3.4)$$

zurückgeführt. Des Weiteren werden für die Druckverhältnisse die Abkürzungen

$$\Pi_c := \frac{p_1}{p_{20}}, \quad (3.5)$$

$$\Pi_t := \frac{p_{40}}{p_3} \quad (3.6)$$

eingeführt. Ebenfalls zur Vereinfachung der Notation wird die Abhängigkeit von der Zeit weggelassen. Es besteht kein Mehrwert, physikalische Größen als zeitabhängig zu kennzeichnen; diese Eigenschaft ist immanent. Das Symbol k wird für Polynomkoeffizienten

¹¹Es existieren polynomiale Ansätze zur Beschreibung der Abgastemperatur, beispielsweise in [GOPB17]. Polynomiale Ansätze wurden bei der Entwicklung des Modells als zu ungenau eingeschätzt.

verwendet und entsprechend mehrfach genutzt. Die Bedeutung der Polynomkoeffizienten ergibt sich daher aus dem Kontext des Textabschnitts. Die Koeffizienten k^* sind ausschließlich für die Reglerauslegung verwendete Parameter. Soll das Modell ausschließlich für Simulationszwecke verwendet werden, können diese zu Null gewählt werden.

3.1.2 Luftfilter: p_1

Begründet durch die Annahme der Stationarität der Massenströme wird eine Beschreibung des Luftfilters gesucht, die den Druck hinter dem Luftfilter p_1 in Abhängigkeit vom Massenstrom W_{ei} berechnet. Der Druck vor dem Luftfilter entspricht dem Umgebungsdruck p_0 und ist damit bekannt. Für das durchströmende Gas stellt der Filter einen Widerstand dar, welcher für einen Druckabfall über diese Komponente sorgt. Der Druck p_1 ist dementsprechend geringer als p_0 . Systemisch wird der Luftfilter als Düse betrachtet, welche mathematisch über die Düsengleichung eines idealen Gases

$$W_{ei} = \frac{A_{af,eff} p_0}{\sqrt{R_a T_0}} \Psi(\Pi_c) \quad (3.7)$$

mit der Definition von Π_c nach (3.5) sowie der stückweise stetig differenzierbaren Durchflussfunktion

$$\Psi(\Pi) = \begin{cases} \tilde{\Psi}(\Pi) & \text{für } \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} < \Pi < 1 \\ \tilde{\Psi}\left(\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}}\right) & \text{sonst,} \end{cases}$$

$$\tilde{\Psi}(\Pi) = \sqrt{\frac{2\gamma}{\gamma-1} \left(\Pi^{\frac{2}{\gamma}} - \Pi^{\frac{\gamma+1}{\gamma}}\right)}.$$

gegeben ist [Wat07, S.120]. $A_{af,eff}$ bezeichnet dabei die effektive Querschnittsfläche und $\gamma \approx 1.4$ ist der Isentropenexponent der Frischluft. Es sei angemerkt, dass für das Abgas, welches eine andere chemische Zusammensetzung besitzt, der gleiche Wert verwendet wird. Mathematisch ist Gleichung (3.7) für die gesuchte Beschreibung ungeeignet, da sich diese Gleichung algebraisch nicht nach p_1 auflösen lässt. Unter der Annahme eines inkompressiblen¹² Gases kann die Drosselgleichung der Form

$$\Delta p = p_0 - p_1 = k_{af} W_{ei} \dot{V}_{af,0} = \frac{k_{af} R_a T_0}{p_0} W_{ei}^2 \quad (3.8)$$

verwendet werden. Der Parameter k_{af} beschreibt dabei den Luftwiderstand des Luftfilters und ist der zu bestimmende Parameter dieser Gleichung. Die Umstellung nach p_1 liefert

¹²Bei kompressiblen Flüssigkeiten bewirkt eine Druckänderung eine Änderung der Dichte. Flüssigkeiten sind in guter Näherung inkompressible Fluide. Für Gase kann eine Abschätzung in Abhängigkeit von der Machzahl vorgenommen werden [BB12].

den gesuchten Zusammenhang:

$$p_1 = p_0 - \frac{k_{\text{af}} R_a T_0}{p_0} W_{\text{ei}}^2.$$

Parameteridentifikation

Die Parameter werden durch die Lösung des Optimierungsproblems

$$\begin{aligned} \min \quad & \|e_{\text{rel}}(p_1)\|_2 \\ \text{u. B. v.} \quad & k_{\text{af}} \in \mathbb{R}^+ \end{aligned}$$

bestimmt. Der relative Fehler $e_{\text{rel}}(\cdot)$ ist in Anhang A.2 definiert. Für sämtliche anderen verwendeten (Fehler-)Normen sei an dieser Stelle ebenfalls auf den genannten Anhang verwiesen.

3.1.3 Kompressor: W_c, T_{20}, P_c

Die Beschreibung des Kompressors besteht aus drei Teilmodellen: ein Modell für den Kompressormassenstrom W_c , eines für die Gastemperatur T_{20} hinter dem Kompressor und eines für die aufgenommene Kompressorleistung P_c . Die ersten beiden Teilmodelle können losgelöst voneinander berechnet werden. Die Leistungsberechnung hängt jedoch von den beiden anderen Modellen ab.

Die Berechnung des Kompressormassenstroms W_c basiert auf den Ausführungen in [JKS⁺91] (zitiert in [MK99]), jedoch in abgewandelter Form. Die Formeln verwenden einige Parameter physikalischen Ursprungs, welche durch beliebige, aber feste Ersatzwerte ersetzt werden, mit denen die Gleichungen zur numerischen Vereinfachung normiert werden können. Dies ist durch den gewählten polynomialen Ansatz möglich, da bei der Parameteroptimierung der dadurch entstandene Fehler mit in die Optimierungsparameter aufgenommen wird. Nachteilig ist, dass ein Teil der physikalischen Einsicht verloren geht, jedoch sind die physikalischen Parameter ebenfalls eine Approximation. Beispielsweise ist nicht klar, welcher Radius des Kompressors verwendet werden muss; der Kompressor weist durch seine Form mehrere Radien auf. Neben der Abstraktion der physikalischen Parameter werden die Formeln derart umgestellt, dass die reduzierten Parameter (3.4) verwendet werden.

Gesucht ist eine Modellbeschreibung, die den Kompressormassenstrom durch die Abhängigkeiten

$$\Pi_c, n_{\text{tc}} \mapsto W_c$$

beschreibt. Dazu wird die dimensionslose Hilfsvariable

$$\Gamma := \frac{c_{p,1} T_1 \left(\Pi_c^{\frac{\gamma-1}{\gamma}} - 1 \right)}{\frac{1}{2} U_c^2} \stackrel{(3.4)}{=} \tilde{k}_1 \frac{c_{p,1} \left(\Pi_c^{\frac{\gamma-1}{\gamma}} - 1 \right)}{n_{c,\text{cor}}^2} \quad (3.9)$$

unter Verwendung der Geschwindigkeit der Kompressorblattspitze

$$U_c = 2\pi r_c n_{tc}$$

eingeführt. Die isobare spezifische Wärmekapazität $c_{p,1}$ hängt an dieser Stelle von der Temperatur T_1 ab. Die Hilfsvariable Γ wird über den polynomialen Ansatz

$$\Gamma = \frac{k_1 + k_2 \Phi}{k_3 - \Phi}, \quad (3.10)$$

$$k_i = k_{i1} + k_{i2} \tilde{n}_{c,\text{cor}} + k_{i3} \tilde{n}_{c,\text{cor}}^2, \quad i = 1, 2, 3,$$

$$\tilde{n}_{c,\text{cor}} = \tilde{k}_3 \min\{n_{c,\text{cor}}, n_{c,\text{cor},\text{max}}\} \quad (3.11)$$

mit dem normalisierten Kompressordurchsatz

$$\Phi = \frac{W_c}{\rho_1 \pi r_c^2 U_c} \stackrel{(3.4)}{=} \tilde{k}_2 \frac{W_{c,\text{cor}}}{n_{c,\text{cor}}} \quad (3.12)$$

verknüpft. Im Gegensatz zu [JKS⁺91] hängen die Polynome k_i nicht von der Mach-Zahl, sondern von der auf den Kompressor bezogenen korrigierten Turboladergeschwindigkeit $n_{c,\text{cor}}$ ab. Der Zusammenhang zur Mach-Zahl ist über

$$Ma = \frac{U_c}{\sqrt{\gamma R_a T_1}} \stackrel{(3.4)}{=} \tilde{k}_3 n_{c,\text{cor}} \quad (3.13)$$

gegeben. Ein weiterer Unterschied ist der Polynomgrad. Wie von [WE10] vorgeschlagen, werden diese als Polynome zweiter Ordnung gewählt, da Untersuchungen eine Verbesserung der Modellgüte gezeigt haben. Die Limitierung (3.11) verhindert die Extrapolation der Polynome. Andernfalls könnte der bei der Parameteridentifikation berücksichtigte Wertebereich des Modells verlassen werden, was unter Umständen zur Abbildung eines unphysikalischen Verhaltens führt. Durch Einsetzen von (3.10) in (3.12) und unter Verwendung von (3.4) kann der Kompressormassenstrom durch

$$W_c = \frac{1}{\tilde{k}_2} \frac{k_3 \Gamma - k_1}{\Gamma + k_2} \frac{p_1 \sqrt{T_{1,\text{ref}}}}{p_{1,\text{ref}} \sqrt{T_1}} n_{tc} \quad (3.14)$$

beschrieben werden. Die Hilfsvariable Γ und der normalisierte Kompressordurchsatz sind über

$$\Phi : [0, 1] \mapsto [0, 1]; \quad \Gamma \mapsto \frac{k_3 \Gamma - k_1}{\Gamma + k_2}.$$

miteinander verknüpft. Der angegebene Definitions- und Wertebereich ergibt sich bei geeigneter Wahl der Parameter \tilde{k}_1 , \tilde{k}_2 und \tilde{k}_3 . Diese werden derart gewählt, dass (3.9), (3.12) und (3.13) auf $[0, 1]$ abbilden. Entsprechend liegen die Koeffizienten k_i ebenfalls in diesem Bereich.

Die Temperatur T_{20} wird unter der Annahme, dass es sich um einen isentropischen Prozess handelt, modelliert. Unter dieser Bedingung gilt

$$T_{20} = T_1 \left(1 + \frac{1}{\eta_c} \left(\Pi_c^{\frac{\gamma-1}{\gamma}} - 1 \right) \right). \quad (3.15)$$

Die isentropische Effizienz η_c wird als nach unten geöffnete Parabel modelliert:

$$\eta_c = \max\{\tilde{\eta}_c, \eta_{c,\min}\}, \quad (3.16a)$$

$$\tilde{\eta}_c = \eta_{c,\max}(1 - a_{c,\text{prbl}}(W_{c,\text{cor}} - W_{c,\text{cor,ref}})^2). \quad (3.16b)$$

Die Parameter der Parabel werden über Polynome in Abhängigkeit der korrigierten Kompressordrehzahl ausgedrückt:

$$\begin{aligned} \eta_{c,\max} &= k_{11} + k_{12}(n_{c,\text{cor}} - k_{14})^2 \\ &\quad + k_{13}(n_{c,\text{cor}} - k_{14})^4, \\ a_{c,\text{prbl}} &= k_{21} + \frac{k_{22}}{(n_{c,\text{cor}} - k_{23})^2}, \\ W_{c,\text{cor,ref}} &= k_{31} + k_{32}n_{c,\text{cor}}. \end{aligned}$$

Die Kompressorleistung P_c ergibt sich über

$$P_c = W_c c_{p,1} (T_{20} - T_1). \quad (3.17)$$

Parameteridentifikation

Die Parameter des Kompressormassenstrommodells werden über die Lösung des Optimierungsproblems

$$\begin{aligned} \min \quad & \|\mathbf{e}_{\text{rel}}(W_c)\|_2 \\ \text{u. B. v.} \quad & k_{11}, k_{13}, k_{22}, k_{31}, k_{33} \in \mathbb{R}^+, \\ & k_{12}, k_{21}, k_{23}, k_{32} \in \mathbb{R}^-, \\ & \Gamma + k_2 > 0, \\ & k_3 - \Phi > 0 \end{aligned}$$

bestimmt. Die letzten beiden Nebenbedingungen des Optimierungsproblems berücksichtigen die Singularitäten der Polynome (3.10) und (3.14). Die Gütefunktion enthält den

relativen Fehler anstelle des absoluten. Dies liegt darin begründet, dass das Modell auch für kleine Massenströme möglichst genau sein soll, da der Massenstrom direkt mit der Turbinenleistung verknüpft ist. Die Turbinenleistung hat einen direkten Einfluss auf die Turboladerdrehzahl, welche auf das Kompressorstrommodell zurückkoppelt und einen Einfluss auf das Turbinenmodell (vgl. Abschnitt 3.1.9) hat.

Die Koeffizienten der isentropischen Effizienz werden durch die Minimierung des absoluten Fehlers der Kompressorleistung mit

$$\begin{aligned} \min \quad & \| \mathbf{e}_{\text{abs}}(P_c) \|_2 \\ \text{u. B. v.} \quad & k_{11}, k_{14}, k_{21}, k_{22}, k_{23} \in \mathbb{R}^+, \\ & k_{12}, k_{13}, k_{31}, k_{32} \in \mathbb{R}^- \end{aligned} \quad (3.18)$$

bestimmt. Die Temperatur T_{20} wird nicht in der Gütefunktion verwendet, weil die Kompressorleistung für die Güte des Gesamtmodells aus den o. g. Gründen relevanter ist. Der Referenzwert für die Kompressorleistung wird mit (3.17) aus den am Prüfstand messbaren Temperaturen und Drücken berechnet. Das verwendete CFD-Modell stellt zwar die Leistung direkt bereit, jedoch soll dieser Ansatz auch unter alleiniger Verwendung von Prüfstandsdaten funktionieren. Nach der vorigen Argumentation müsste hier ebenfalls der relative Fehler verwendet werden, allerdings berücksichtigt das CFD-Modell die Temperaturverluste an die Umgebung nicht. Der dadurch induzierte Fehler ist für kleine Leistungen stärker ausgeprägt als für große. Durch die Verwendung des absoluten Fehlers werden die großen Leistungen stärker gewichtet als die kleinen. Da das Modell relativ viele Koeffizienten aufweist, besteht ein sinnvolles Vorgehen für den Fit darin, zunächst $\eta_{c,\text{max}}$, $a_{c,\text{prbl}}$ und $W_{c,\text{cor,ref}}$ für feste korrigierte Turboladerdrehzahlen zu berechnen und anschließend die korrespondierenden Polynome zu bestimmen. Die auf diese Weise berechneten Koeffizienten können zur Verfeinerung des Suchraums von (3.18) verwendet werden. Die Effizienz kann mit (3.15) berechnet werden.

3.1.4 Ladeluftkühler: T_{21}, p_{20}

Der Ladeluftkühler besteht aus zwei Teilmodellen: zum einen der Beschreibung der Temperatur T_{21} hinter dem Kühler und zum anderen des Drucks p_{20} vor dem Kühler. Das Druckmodell hängt von der berechneten Temperatur hinter dem Kühler ab.

Für die Temperatur wird ein stationärer Ansatz gewählt. Der Kühler hat einen bestimmten Wirkungsgrad ϵ_{ac} , mit dem sich die gesuchte Temperatur über

$$\tilde{T}_{21} = T_{20} - \epsilon_{ac} (T_{20} - T_0) \quad (3.19)$$

ergibt. Die Effizienz des Kühlers hängt maßgeblich davon ab, wie lange sich das durchströmende Gas in dem Kühler aufhält und wie viel Energie es aufnehmen kann. Die Auswertung

von Messdaten hat eine Beschreibung in Abhängigkeit des Massenstroms nahegelegt:

$$\varepsilon_{ac} = k_1 + k_2 W_{ei}.$$

Da \tilde{T}_{21} in dem Modell für den Druck p_{20} enthalten ist und dieser wiederum über (3.15) T_{20} beeinflusst, ergibt sich eine algebraische Schleife. Die Schleife wird durch das Tiefpassfilter

$$\frac{dT_{21}}{dt} = \frac{1}{\tau_{21}}(\tilde{T}_{21} - T_{21}) \quad (3.20)$$

gebrochen. Die Zeitkonstante τ_{21} ist entsprechend klein zu wählen, so dass das Filter keinen nennenswerten Einfluss auf die Dynamik hat. Andererseits darf es wiederum nicht so klein gewählt werden, dass es den dominanten Eigenwert im Sinne der Schrittweitenabschätzung (3.57) bildet, da dieser dann die nötige Abtastzeit vorgeben und damit die benötigte Rechenleistung zur Berechnung des Modells erhöhen würde (vgl. Abschnitt 3.3.3). Das Modell berücksichtigt keine Wärmekapazität des Kühlers, welche sich als tiefpassfilternde Wirkung bemerkbar machen würde. Die Zeitkonstante τ_{21} kann zur Beschreibung eines solchen Verhaltens zweckentfremdet werden. Von einer expliziten Formulierung eines dynamischen Verhaltens des Kühlers wird aufgrund einiger Unsicherheiten abgesehen. Zum einen ist der Kühler eine aktive Komponente. Er ist geregelt und tritt damit als nicht-deterministische, nichtlineare Komponente auf. Zum anderen ist der Kühler ein Kundenbauteil, welches nicht von der *Deutz AG* vorgegeben wird. Daher wird davon ausgegangen, dass es vielversprechender ist, ein vergleichsweise simples Modell mit wenigen Parametern zu definieren, welche über einen geeigneten Beobachteransatz zur Laufzeit angepasst werden. Dies ist möglich, da die Verbrennungsmotoren der *Deutz AG* über einen Ladeluft-TemperaturSENSOR verfügen.

Der Druckabfall über den Ladeluftkühler wird analog zum Druckabfall über den Luftfilter modelliert. Auch hier müsste formal mit der Düsengleichung (3.7) gearbeitet werden. Mit dem Drosselansatz ergibt sich

$$\Delta p = p_{20} - p_{22} = k_{ac} W_{ei} \dot{V}_{ei,21} = \frac{k_{ac} R_a T_{21}}{p_{20}} W_{ei}^2.$$

Durch Auflösen nach p_{20} folgt der gesuchte Zusammenhang

$$p_{20} = \frac{p_{22}}{2} + \sqrt{\left(\frac{p_{22}}{2}\right)^2 + k_{ac} R_a T_{21} W_{ei}^2}. \quad (3.21)$$

Der Druck p_{22} wird in (3.26) über eine Differentialgleichung beschrieben und ist damit, als Zustand des Modells, bekannt. p_{20} ist größer als p_{22} . Formal müsste der Volumenstrom $\dot{V}_{ei,20}$ verwendet werden. Durch die Auswertung von Messdaten hat sich jedoch $\dot{V}_{ei,21}$ als zweckmäßiger erwiesen – beide Volumenströme liefern akzeptable Genauigkeiten.

Parameteridentifikation

Der Kühlerwirkungsgrad wird bezüglich des relativen Fehlers mit

$$\begin{aligned} \min & \quad \|\mathbf{e}_{\text{rel}}(T_{21})\|_2 \\ \text{u. B. v.} & \quad k_1 \in \mathbb{R}^+, \\ & \quad k_2 \in \mathbb{R}^- \end{aligned} \quad (3.22)$$

bestimmt.

Der Koeffizient k_{ac} für den Druckverlust über den Ladeluftkühler wird ebenfalls derart bestimmt, dass der relative Fehler minimal wird:

$$\begin{aligned} \min & \quad \|\mathbf{e}_{\text{rel}}(p_{20})\|_2 \\ \text{u. B. v.} & \quad k_{\text{ac}} \in \mathbb{R}^+. \end{aligned} \quad (3.23)$$

3.1.5 Einlassflut: T_{22}, p_{22}

Die Einlassflut beherbergt die Modelle für die Ladelufttemperatur T_{22} sowie für den Ladedruck p_{22} . Beide Größen gehen aus einer DGL hervor.

Für die Ladelufttemperatur wird jedoch die Vereinfachung

$$T_{22} := T_{21} \quad (3.24)$$

verwendet. Die Begründung liegt darin, dass das Volumen der Einlassflut so klein ist, dass selbst im Leerlauf die Luft einmal alle 40 ms vollständig ausgetauscht wird und damit identisch zu T_{21} ist. In anderen Betriebszuständen mit höherer Drehzahl ist diese Zeit entsprechend kleiner. Aus einer praktischen Perspektive müsste die Abtastzeit des Integrators mindestens eine Größenordnung kleiner sein, um diesen Verlauf mit hinreichender Präzision abzubilden. Im Kontext der anderen Systemdynamiken ist diese Zeit klein. Diese Annahme kann formal durch Einsetzen der Stationaritätsbedingung $W_c = W_{\text{ei}}$ und $\frac{dT_{22}}{dt} = 0$ in die korrespondierende DGL [MHS98] (zitiert in [CMH00])

$$\frac{dT_{22}}{dt} = \frac{R_a T_{22}}{V_{22} p_{22}} \left(T_{21} - T_{22} + \frac{\gamma}{c_{p,1}} R_a (W_c T_{21} - W_{\text{ei}} T_{22}) \right) \quad (3.25)$$

begründet werden. V_{22} bezeichnet das Volumen der Einlassflut.

Der Druck wird durch die DGL

$$\frac{dp_{22}}{dt} = \frac{\gamma R_a}{V_{22}} T_{22} (W_c - W_{\text{ei}}) \quad (3.26)$$

nach [CVC01]¹³ beschrieben. Die Formel ergibt sich aus einer Massen- und Energiebilanz. Es empfiehlt sich in einer praktischen Umsetzung, das Volumen V_{22} größer zu wählen als es eigentlich ist. Es ist möglich, dass ohne diese Anpassung diese Gleichung den dominanten Eigenwert stellt und damit maßgeblich für die nötige Abtastrate ist. Das makroskopische Zeitverhalten des Gesamtsystems wird jedoch durch andere Komponenten, beispielsweise den Turbolader, dominiert. Das Volumen V_{22} kann als Zeitkonstante, analog zu τ_{21} in (3.20), interpretiert werden. Je größer das Volumen ist, desto *langsamer* wird das zeitliche Verhalten von p_{22} .

3.1.6 Motorblock: $W_{ei}, W_f, W_{eo}, X_{O_2, eo}, T_{eo}$

Das Motorblockmodell beinhaltet einerseits die Berechnung der Luft- und Kraftstoffmassenströme W_{ei}, W_f, W_{eo} und andererseits den Sauerstoffmassenanteil $X_{O_2, eo}$ sowie die verbrennungsabhängige Auslasstemperatur T_{eo} .

Der Ladeluftmassenstrom geht unter Kenntnis des Hubraumes V_d sowie der Luftdichte ρ_{22} aus der Motordrehzahl n_{eng} über

$$W_{ei} = \frac{1}{2} \eta_V \rho_{22} \frac{V_d}{2} n_{eng} = \frac{1}{2} \eta_V \frac{p_{22}}{R_a T_{22}} V_d n_{eng} \quad (3.27)$$

hervor. Die Dichte wurde auf der rechten Seite durch die Modellzustände ersetzt. Noch zu bestimmen ist ein Modell für die volumetrische Effizienz η_V . Eine Analyse von Messdaten legt nahe, dass die Effizienz maßgeblich vom Druck in der Einlassflut, vom Druck in der Auslassflut p_3 und von der Motordrehzahl beeinflusst wird. Dieser Zusammenhang wird über das Polynom

$$\begin{aligned} \eta_V &= k_V \eta_{V, p_{22}} \eta_{V, p_3} \eta_{V, n_{eng}}, \\ \eta_{V, p_{22}} &= 1 + k_{11} p_{22} + k_{12} p_{22}^2, \\ \eta_{V, p_3} &= 1 + k_{21} p_3, \\ \eta_{V, n_{eng}} &= 1 + k_{31} n_{eng} + k_{32} n_{eng}^2 \end{aligned}$$

abgebildet.

Unter Verwendung der Motordrehzahl, der Zylinderzahl $N_{cyl} := 6$ und der Einspritzmenge kann der Kraftstoffmassenstrom über

$$W_f = \frac{1}{2} N_{cyl} n_{eng} \tilde{q}_{inj} \quad (3.29)$$

¹³[CVC01] verweist auf [GA98], verwendet jedoch eine andere Schreibweise.

berechnet werden. Als Einspritzmenge wird der Zusammenhang

$$\tilde{q}_{\text{inj}} = q_{\text{inj}} + \underbrace{\frac{\partial q_{\text{inj}}}{\partial \Delta p_{3,22}}}_{k_{\text{pump}}^*} (\Delta p_{3,22} - \Delta p_{3,22}^*), \quad \Delta p_{3,22} := p_3 - p_{22}$$

verwendet. Der Ausdruck stellt eine Taylor-Approximation erster Ordnung nach dem Differenzdruck $\Delta p_{3,22}$ dar. Der Druck wirkt dem Arbeitsspiel des Motors entgegen und bewirkt eine höhere Einspritzmenge zur Erzielung des gleichen effektiven Drehmoments M_e . Die durch den Differenzdruck hervorgerufenen Leistungseinbußen werden *Ladungswechsel* oder auch *Pumpverluste* genannt. Für die Reglerentwicklung können durch diese Art der Darstellung die Pumpverluste des Motors berücksichtigt werden, die im Verbund mit einem Drehmomentregler unmittelbar ausgeglichen würden. Für Simulationsanwendungen kann k_{pump}^* zu Null gewählt werden.

Der Abgasmassenstrom ergibt sich über die Massenbilanz des Ladeluftmassenstroms und des Kraftstoffmassenstroms durch

$$W_{\text{eo}} = W_{\text{ei}} + W_{\text{f}}. \quad (3.30)$$

Die zur Verbrennung einer bestimmten Menge Kraftstoff nötige Menge Luft ist über das stöchiometrische Verhältnis $\Phi_s \approx 14,5$ bekannt.¹⁴ Das bedeutet, wenn dieses Verhältnis exakt eingehalten wird, befindet sich näherungsweise kein Sauerstoff mehr im Abgas. Formal entspricht der Sauerstoffmassenstrom im Abgas dem Sauerstoffmassenstrom, welcher in der verbliebenen Frischluft enthalten ist:¹⁵

$$X_{\text{O}_2,\text{eo}} W_{\text{eo}} = X_{\text{O}_2,0} (W_{\text{ei}} - \Phi_s W_{\text{f}}).$$

Die Frischluft hat einen Sauerstoffanteil von $X_{\text{O}_2,0} \approx 0,23$.¹⁶ Der gesuchte Sauerstoffanteil ergibt sich damit über

$$X_{\text{O}_2,\text{eo}} = \frac{X_{\text{O}_2,0} (W_{\text{ei}} - \Phi_s W_{\text{f}})}{W_{\text{eo}}}.$$

Die Abgastemperatur direkt hinter dem Motorblock wird durch eine Kombination eines Stationärmodells \tilde{T}_{eo} und eines Rohrmodells abgebildet, welches das zeitabhängige Verhalten durch das Aufheizen der Zylinderwände wiedergibt.

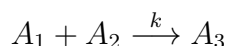
Das Stationärmodell besteht aus einem KNN und wird nicht als Teil des eigentlichen Luft-

¹⁴Das stöchiometrische Sauerstoff/Kraftstoff-Verhältnis beträgt $0,23\Phi_s \approx 3,36$.

¹⁵Dieser Motor hat keine Abgasrückführung. Damit ist der Frischluftmassenstrom identisch zum Ladeluftmassenstrom.

¹⁶Bei einem Motor mit Abgasrückführung ist die rechte Seite durch $X_{\text{O}_2,\text{ei}} W_{\text{ei}} - X_{\text{O}_2,0} \Phi_s W_{\text{f}}$ zu ersetzen. Die Sauerstoffkonzentration $X_{\text{O}_2,\text{ei}}$ in der Einlassflut geht in diesem Fall aus einer DGL hervor.

pfadmodells betrachtet. Das KNN wird außerhalb des Modells berechnet und stellt daher formal lediglich einen zusätzlichen Modelleingang bereit. Die stationäre Beschreibung beruht auf der Tatsache, dass die Verbrennung im Vergleich zu den restlichen Prozessen schnell stattfindet. Ein KNN wird aufgrund der Nichtlinearität der Verbrennung und der hohen Anzahl an Abhängigkeiten verwendet. Um dies zu verdeutlichen, wird eine chemische Reaktion zweiter Ordnung betrachtet:¹⁷ Die Kinetik der Reaktion zweier Edukte A_1 und A_2 zum Produkt A_3



wird über die DGL der Form

$$r = \frac{dc_3}{dt} = -\frac{dc_1}{dt} = -\frac{dc_2}{dt} = c_1 c_2 \cdot k = c_1 c_2 \cdot k_0 \exp\left(-\frac{E}{RT}\right) \quad (3.31)$$

beschrieben [BBB⁺13, S. 76]. Hierin ist k die Geschwindigkeitskonstante der Reaktion, deren Temperaturabhängigkeit, wie auf der rechten Seite dargestellt, über einen Arrhenius-Ansatz approximiert werden kann. Die Reaktionsgeschwindigkeit r hängt entsprechend von der Geschwindigkeitskonstante sowie von den Konzentrationen der Edukte c_1 und c_2 ab. Folgende Überlegung macht klar, warum eine dynamische Beschreibung der Verbrennung in einem eingebetteten Umfeld problematisch ist: Wenn sich ein Viertakt-Motor im Leerlauf bei einer angenommenen Drehzahl von 600 U/min befindet, dauert ein Takt ca. 50 ms. Die eigentliche Verbrennung dauert lediglich einen Bruchteil davon. Um eine DGL der Art (3.31) zu integrieren, wäre eine deutlich kleinere Abtastrate vonnöten. Bei höheren Drehzahlen werden die Zeiten entsprechend kleiner. Auf die Verbrennung übertragen korrespondieren die Konzentrationen zum Luftsauerstoff und zum Kraftstoff. Die Konzentration des Luftsauerstoffs hängt vom Ladeluftmassenstrom ab. Nach (3.27) ist dieser abhängig von der Motordrehzahl, dem Ladedruck, der Ladelufttemperatur und dem Druck in der Auslassflut. Da sich die Drücke und Temperaturen während des Brennverlaufs abhängig vom Kurbelwinkel ϕ ändern, spielt auch der Einspritzwinkel φ_{inj} eine Rolle [PKS09, S. 111]. Bei mehrfachen Einspritzungen sind dies entsprechend mehrere Winkel. Für das KNN in dieser Arbeit werden die Abhängigkeiten von den Einspritzwinkeln außer Acht gelassen, da diese begründet durch die Betriebsstrategie des Motors über ein Kennfeld von n_{eng} und q_{inj} abhängen und damit indirekt über das KNN abgebildet werden können.¹⁸ Damit ergibt sich folgender funktionaler Zusammenhang:¹⁹

$$\text{KNN} : (q_{inj}, T_{22}, n_{eng}, p_{22}, p_3) \mapsto \tilde{T}_{eo,ANN}. \quad (3.32)$$

¹⁷Exemplarisch wird eine chemische Reaktion zweiter Ordnung gewählt. Der dargestellte Zusammenhang gilt für chemische Reaktionen anderer Ordnungen in ähnlicher Weise. Weitere Ausführungen zur Verbrennung sind in [PKS09, S. 68ff] zu finden. Laut [PKS09, S. 102] würde eine vollständige chemische Darstellung eine Modellierung mehrerer tausend Elementarreaktionen bedingen.

¹⁸Es ist möglich, dass in unterschiedlichen Betriebssituationen unterschiedliche Datensätze für die Kennfelder verwendet werden. Um ein zu den Kennfeldern passendes Verhalten des KNN sicherzustellen, bietet es sich an, für dieses ebenfalls unterschiedliche Datensätze zu verwenden.

¹⁹Bei einem Motor mit Abgasrückführung wäre der Sauerstoffmassenanteil des Ladeluftmassenstroms ein weiterer Parameter.

Durch (3.32) wird ersichtlich, dass die Eingänge des KNN bis auf die externen Größen q_{inj} und n_{eng} aus modellierten Größen bestehen. Um die Kopplungen auf den Ladeluftmassenstrom W_{ei} zu erhalten, wird statt der Temperatur die dem Abgas hinzugefügte Heizleistung in das Modell gegeben:

$$\begin{aligned}\dot{Q}_f^* &= \dot{Q}_{eo}^* - \dot{Q}_{ei}^* \\ &= c_p W_{eo}^* \tilde{T}_{eo,ANN}^* - c_{p,22}^* W_{ei}^* T_{22}^*.\end{aligned}$$

Um zu kennzeichnen, dass es sich aus Sicht des Modells um eine externe Größen handelt, sind diese im Exponenten mit einem * gekennzeichnet. Im Modell wird eine inverse Berechnung durchgeführt, um aus \dot{Q}_f^* die Abgastemperatur T_{eo} zu berechnen:

$$\tilde{T}_{eo} = \frac{c_{p,22} W_{ei} T_{22} + \dot{Q}_f}{c_p W_{eo}} \quad (3.33)$$

mit

$$\dot{Q}_f = \dot{Q}_f^* + \underbrace{\frac{\partial \dot{Q}_f}{\partial W_f}}_{k_f^*} (W_f - W_f^*). \quad (3.34)$$

Anstelle \dot{Q}_f^* direkt zu verwenden, wird eine Taylor-Approximation erster Ordnung (3.34) nach dem Kraftstoffmassenstrom herangezogen. Dadurch enthält das Modell diese Abhängigkeit ebenfalls. Wenn im Rahmen einer Simulation das KNN mit der gleichen Abtastrate wie das Luftpfadmodell berechnet wird, entfällt der letzte Term, da dann $W_f = W_f^*$ gilt. Der Parameter k_f^* kann in diesem Fall beliebig gewählt werden, es sei denn das Modell soll zum Reglerentwurf verwendet werden. Es sei angemerkt, dass eine feste aber beliebige Wärmekapazität c_p verwendet wird. Dies ist durch die Abhängigkeit von der Temperatur \tilde{T}_{eo} begründet, deren Verwendung zu einer algebraischen Schleife führen würde bzw. eine iterative Berechnung von (3.33) erfordern würde. Ebenfalls sei angemerkt, dass im Eingangsvektor (3.3) q_{inj}^* anstelle von W_f^* verwendet wird.

Das dynamische Verhalten wird nach einem angepassten Ansatz von [GOPB17] dargestellt:²⁰

$$\frac{dT_{cyl}}{dt} = W_{eo}^{0.8} k_1 (\tilde{T}_{eo} - T_{cyl}), \quad (3.35a)$$

$$T_{eo} = T_{cyl} + (\tilde{T}_{eo} - T_{cyl}) e^{\frac{k_2}{W_{eo}^{0.2} \tilde{c}_{p,eo}}}. \quad (3.35b)$$

Die Bezeichnung T_{cyl} meint nicht die Zylindertemperatur, sondern ist durch die zugrundeliegende Modellvorstellung motiviert. Die Temperatur kann als Indikation verstanden werden.

²⁰In [GOPB17] wird mit dem Ansatz die Dynamik der Auslassflut beschrieben.

Parameteridentifikation

Die Parameter des Ladeluftmassenstrommodells werden durch die Lösung des folgenden Optimierungsproblems ermittelt:

$$\begin{aligned} \min \quad & \|\mathbf{e}_{\text{rel}}(W_{\text{ei}})\|_2 \\ \text{u. B. v.} \quad & k_{\text{V}}, k_{11}, k_{31} \in \mathbb{R}^+, \\ & k_{21}, k_{12}, k_{32} \in \mathbb{R}^-, \\ & \eta_{\text{V},p22} > 0, \\ & \eta_{\text{V},p3} > 0, \\ & \eta_{\text{V},n_{\text{eng}}} > 0. \end{aligned}$$

Während der Optimierung ist darauf zu achten, dass die Polynome $\eta_{\text{V},p22}$, $\eta_{\text{V},p3}$, $\eta_{\text{V},n_{\text{eng}}}$ stets positiv sind, andernfalls kann der Optimierer zu einer ungewollten lokalen Lösung konvergieren.

Für die Pumpverluste wird eine Ersatzoptimierung herangezogen. Das Drehmoment ist näherungsweise proportional zur Einspritzmenge (vgl. Abb. 5.3), wenn keine (hohen) Pumpverluste vorhanden sind. Das effektive Drehmoment kann daher in einer einfachen Näherung über

$$M_e = k_1 (q_{\text{inj}} - k_{\text{pump}}^* \Delta p_{3,22})$$

beschrieben werden. Die Pumpverluste wirken sich drehmomentmindernd aus. Es ist das Optimierungsproblem

$$\begin{aligned} \min \quad & \|\mathbf{e}_{\text{abs}}(M_e)\|_2 \\ \text{u. B. v.} \quad & k_1, k_{\text{pump}}^* \in \mathbb{R}^+ \end{aligned}$$

zu lösen. An bestimmten Arbeitspunkten wird das effektive Drehmoment klein bzw. sogar negativ, wenn der Motor geschleppt wird. Aus diesem Grund wird der absolute Fehler gewählt. Der Parameter k_1 wird lediglich im Rahmen der Optimierung benötigt und im Weiteren nicht verwendet.

Das statische Abgastemperaturmodell erfordert zwei unterlagerte Optimierungsschritte. Der erste ist das Training des KNN. Das in dieser Arbeit verwendete Netz ist als dichtes *Multilayer Perceptron* realisiert und wird mit dem *Deep Learning* Framework *keras* [Cho15] trainiert. Es wird nach

$$\min \quad \|\mathbf{e}_{\text{abs}}(\tilde{T}_{\text{eo,ANN}})\|_2$$

optimiert, mit dem Ziel, den absoluten Fehler zu minimieren. Der absolute Fehler wird gewählt, um den sicheren Betrieb des Motors zu gewährleisten. Besonders hohe Abgastemperaturen sind aufgrund der negativen Auswirkungen auf die Turbine kritisch. Durch

den absoluten Fehler werden die hohen Temperaturen stärker betont. Der zweite Optimierungsschritt betrifft die Kraftstoffabhängigkeit der Heizleistung \dot{Q}_f aus Gleichung (3.34):

$$\begin{aligned} \min & \quad \|e_{\text{abs}}(\dot{Q}_f - \dot{Q}_f^* - k_f^*(W_f - W_f^*))\|_2 \\ \text{u. B. v.} & \quad k_f^* \in \mathbb{R}^+. \end{aligned}$$

Das Rohrmodell (3.35) wird – im Kontrast zu den anderen Modellen – anhand einer dynamischen Messung bestimmt. Um diese Optimierung durchführen zu können, muss zuvor das statische Temperaturmodell bestimmt worden sein. Da das dynamische Modell lediglich den zeitlichen Verlauf abbildet und keinen Einfluss auf den stationären Temperaturwert hat, kann der relative Fehler verwendet werden:

$$\begin{aligned} \min & \quad \frac{1}{T} \int_0^T e_{\text{rel}}(T_{\text{eo}})^2 dt \\ \text{u. B. v.} & \quad \frac{dT_{\text{cyl}}}{dt} = W_{\text{eo}}^{0.8} k_1 (\tilde{T}_{\text{eo}} - T_{\text{cyl}}), \quad T_{\text{cyl}}(0) := T_{\text{eo},0}, \\ & \quad k_1 \in \mathbb{R}^+, \\ & \quad k_2 \in \mathbb{R}^-. \end{aligned}$$

Es genügt, das statische Modell zuvor einmal separat zu evaluieren und die berechnete Temperatur als Eingang von (3.35) zu betrachten.

Es empfiehlt sich, die Messdaten für T_3 zu verwenden, anstatt auf T_{eo} zu optimieren. Begründet wird dies im folgenden Abschnitt.

3.1.7 Auslassflut: T_3, p_3

Die Auslassflut wird analog zur Einlassflut in Abschnitt 3.1.5 modelliert und beschreibt entsprechend die Temperatur T_3 sowie den Druck p_3 .

Wie bei der Einlassflut wird auf eine DGL für T_3 verzichtet und stattdessen

$$T_3 := T_{\text{eo}} \tag{3.36}$$

definiert. In der Realität stimmen die Temperaturen in transienten Vorgängen wegen stattfindender Transport- und Einschwingprozesse nicht überein. Aber auch im Stationären gibt es Abweichungen: Einerseits wird das Abgas gerade bei großen Massenströmen durch Rohrreibung weiter aufgeheizt. Andererseits finden Wärmeverluste an die Umgebung statt, die einen gegenteiligen Effekt auf die Temperatur haben. Letzterer Effekt kann mit dem Ansatz von [GOPB17] abgebildet werden. Allerdings wird dafür die Temperatur außerhalb der Flut benötigt, welche je nach Platzierung des Motors im Motorraum von der Umgebungstemperatur abweichen kann und damit eine Unsicherheit darstellt. Um diese Effekte

nicht unberücksichtigt zu lassen, wird das Abgastemperaturmodell aus dem vorigen Abschnitt 3.1.6 idealerweise auf T_3 optimiert.

Der Druck wird ebenfalls über eine DGL dargestellt:

$$\frac{dp_3}{dt} = \frac{\gamma R_a}{V_3} T_3 (W_{eo} - W_t - W_{wg}). \quad (3.37)$$

Es sei angemerkt, dass auch hier mit der gleichen Argumentation wie in Abschnitt 3.1.5 eine Vergrößerung des Volumens V_3 im Sinne der Rechenzeit sinnvoll sein kann, wenn die DGL den dominanten Eigenwert stellt.

3.1.8 Wastegate: ζ_{wg} , W_{wg}

Das WG ist eine aktive Komponente. Es regelt die Klappenposition ζ_{wg} auf den einen Sollwert r_{wg} ein. Für die Transienten wird ein PT1-Verhalten angenommen:

$$\frac{d\zeta_{wg}}{dt} = \frac{1}{\tau_{wg}} (r_{wg} - \zeta_{wg}). \quad (3.38)$$

Realitätsnäher ist ein PT2-Verhalten, da die Klappe aufgrund ihrer Trägheit erst beschleunigt werden muss. Dieses Verhalten wird durch ein PT2-Glied besser nachgeahmt. Eine solche Beschreibung benötigt jedoch einen weiteren Zustand, weswegen darauf verzichtet wird. Um die maximale Klappengeschwindigkeit zu berücksichtigen, ist es denkbar, die DGL durch eine obere und untere Grenze zu erweitern. Die Zeitkonstante kann dem Datenblatt entnommen oder experimentell durch Bewertung der Sprungantwort ermittelt werden.

Der Massenstrom wird durch die Düsgleichung (3.7) beschrieben:

$$W_{wg} = \frac{A_{wg,eff} p_3}{\sqrt{R_a T_3}} \Psi(\Pi_t), \quad (3.39)$$

unter Verwendung des Druckverhältnisses Π_t nach (3.6). Der Massenstrom erhält die Abhängigkeit zur Klappenposition über die Beschreibung der effektiven Querschnittsfläche $A_{wg,eff}$:

$$A_{wg,eff} = \max(\tilde{A}_{wg,eff}, A_{wg,eff}^*), \quad (3.40a)$$

$$\tilde{A}_{wg,eff} = k_1 + k_2 (\min(\zeta_{wg}, k_4))^2 + k_3 \max(0, \zeta_{wg} - k_4), \quad (3.40b)$$

$$A_{wg,eff}^* = k_1 + k_5^* \zeta_{wg}. \quad (3.40c)$$

Das eigentliche Modell ist das stückweise Polynom $\tilde{A}_{wg,eff}$. Der Bereich für $\zeta_{wg} \leq k_4$ wird über einen quadratischen Zusammenhang dargestellt. Für größere Klappenpositionen wird ein linearer Zusammenhang angenommen. Die Ableitung des quadratischen Teils nach der

Klappenposition kann für $\zeta_{wg} \mapsto 0$ Null bzw. für einen Reglerentwurf zu klein werden. In diesem Fall würde der entsprechende Eintrag der Eingangsmatrix \mathbf{B} (siehe (2.4)) verschwinden und das System wäre mit dem WG an dieser Stelle nicht steuerbar. Aus diesem Grund wird das Design-Polynom $A_{wg,eff}^*$ eingeführt. Dadurch kann die Ableitung auch für kleine Positionen zu größer als Null erzwungen werden. Der berechnete Massenstrom sowie die effektive Querschnittsfläche sind in Abb. 3.2 dargestellt. Das Design-Polynom ist bis 20 % Aktuierung dominant.

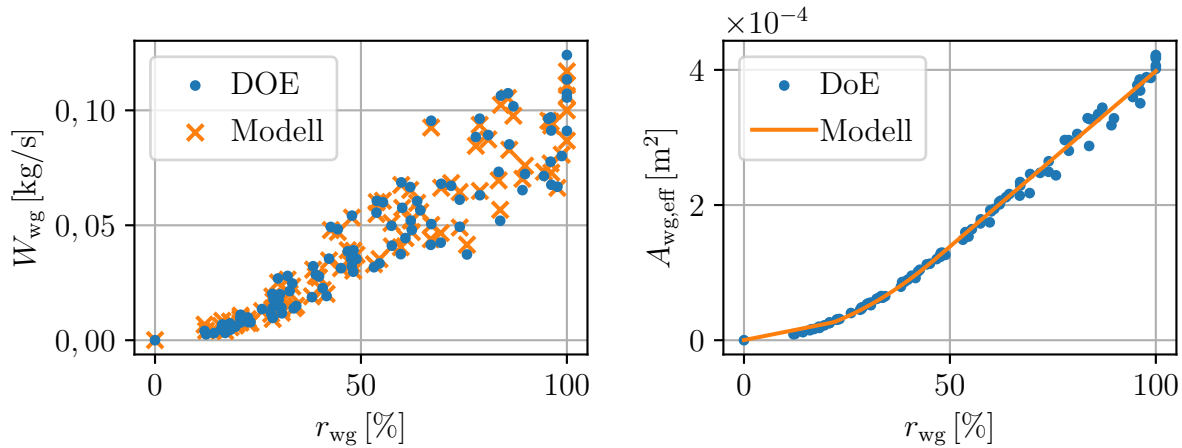


Abb. 3.2: Modell des WG-Massenstroms sowie der modellierten effektiven Querschnittsfläche im Vergleich mit dem DoE (vgl. Abschnitt 3.2.1).

Parameteridentifikation

Der Massenstrom wird dahingehend optimiert, dass der absolute Fehler minimal wird:

$$\begin{aligned} \min \quad & \|e_{abs}(W_{wg})\|_2 \\ \text{u. B. v.} \quad & k_1, k_2, k_3, k_4 \in \mathbb{R}^+. \end{aligned}$$

Der absolute Fehler wird aus numerischen Gründen verwendet. Es gibt Arbeitspunkte, an denen das WG geschlossen ist und damit der Massenstrom Null wird. Der eigentlich aufgrund der Gleichbehandlung der verschiedenen Arbeitspunkte zu bevorzugende relative Fehler ist an dieser Stelle nicht definiert.

3.1.9 Turbine: W_t, P_{tm}, T_t, T_{40}

Der Turbinenmassenstrom wird ebenfalls mit der Düsgleichung beschrieben. Die Düsgleichung reicht jedoch alleine nicht aus, um die Physik vollständig abzubilden. Ein vielfach

verwendeter Ansatz in der Literatur ist, die effektive Fläche durch eine von der korrigierten Turbinendrehzahl $n_{t,\text{cor}}$ abhängige Funktion zu ersetzen [MK99, ECTC12]. Besonders der Ansatz von [ECTC12] verspricht eine hohe Modellqualität, allerdings wird dort eine Formulierung mit einer Exponentialfunktion verwendet, welche aufgrund ihrer Gestalt die Parameteridentifikation verkompliziert. In [Eri07] wird ein anderer Ansatz motiviert: Bei radialen Turbinen – welche die gebräuchlichste Bauweise für Turbolader ist – expandiert das Gas auch im Rotor, weswegen angenommen wird, dass lediglich die Hälfte im Stator expandiert:

$$\Pi_t = \underbrace{\sqrt{\Pi_t}}_{\text{Stator}} \underbrace{\sqrt{\Pi_t}}_{\text{Rotor}}.$$

Da die Aufteilung auf Rotor und Stator lediglich eine Annahme darstellt, wird in diesem Modell die Aufteilung selbst zu einem Optimierungsparameter:

$$\Pi_t = \underbrace{\Pi_t^{k_1}}_{\text{Stator}} \underbrace{\Pi_t^{(1-k_1)}}_{\text{Rotor}}.$$

Ein weiterer Unterschied zu [Eri07] ist die Beschreibung der effektiven Fläche durch eine lineare Funktion in Abhängigkeit zur korrigierten Turbinendrehzahl. Zusammenfassend wird der Turbinenmassenstrom über

$$W_t = \frac{A_{t,\text{eff}} p_3}{\sqrt{R_a T_3}} \Psi(\Pi_t^{k_1}),$$

$$A_{t,\text{eff}} = k_2 + k_3 n_{t,\text{cor}},$$

beschrieben.

Zur Beschreibung der Turbinenleistung wird eine andere Formulierung als für die Kompressorleistung (3.17) verwendet. Prinzipiell könnte das Modell identisch formuliert werden: eine Berechnung von T_{40} analog zu (3.15) unter Verwendung eines Modells für die isentrope Effizienz η_t . Allerdings ist die etablierte Praxis der Turboladerhersteller die Bereitstellung der Effizienz η_{tm} , welche neben dem isentropen Wirkungsgrad den mechanischen enthält. Aus diesem Grund stellt das verwendete CFD-Modell keine akkurate Temperatur T_t bereit.²¹ Diese Ungenauigkeit schließt die Verwendung dieser Temperatur zur Leistungsbeschreibung aus. Statt die Turbinenleistung zu beschreiben, soll die kombinierte Leistung aus Turbinen- und mechanischer Verlustleistung beschrieben werden:

$$P_{tm} := \eta_m P_t = \underbrace{\eta_t \eta_m}_{\eta_{tm}} P_{t,s}, \quad \text{mit} \quad \eta_t = \frac{P_t}{P_{t,s}}. \quad (3.41)$$

²¹Es ist möglich, die mechanische Effizienz mit dem verwendeten CFD-Modell darzustellen, sodass T_t akkurat wird, jedoch wurde dieser Aufwand nicht investiert, da T_t für die Motorkalibrierung auch ohne Berücksichtigung des mechanischen Wirkungsgrades genau genug ist.

Die Leistung ohne die isentropischen Verluste ist über

$$P_{t,s} = W_t c_{p,3} (T_3 - T_{t,s}) = W_t c_{p,3} T_3 \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right) \quad (3.42)$$

gegeben.

Eine Beschreibung der Effizienz η_{tm} wird über das Drehzahlverhältnis der Turbinenschaufeln (engl.: *Blade Speed Ratio*, kurz: BSR) hergestellt. Die Effizienz zeigt eine parabolische Abhängigkeit zum BSR [WJ82] (zitiert in [WE11]). Das Polynom hat die von [MK99] vorgeschlagene Form

$$\eta_{tm} = k_1 + k_2 \text{BSR} + k_3 \text{BSR}^2. \quad (3.43)$$

Das Drehzahlverhältnis selbst ist über

$$\text{BSR} = \frac{r_t \omega_t}{\sqrt{2c_{p,3} T_3 \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right)}} \stackrel{(3.4)}{=} \tilde{k}_1 \frac{n_{t,\text{cor}}}{\sqrt{c_{p,3} \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right)}} \quad (3.44)$$

in reduzierten Parametern unter Verwendung des Turbinenblattradius r_t definiert. Der Ansatz über das Polynom (3.43) liefert vergleichbare Ergebnisse zum Ansatz von [WE11]. Allerdings ist er aus rechentechnischer Sicht genügsamer, da lediglich ganzzahlige Exponenten verwendet werden (vgl. Abschnitt 3.3.2). Der Koeffizient \tilde{k}_1 kann beliebig gewählt werden. Abweichungen werden durch die Parameter von η_{tm} ausgeglichen. Wie bei den freien Parametern im Kompressormodell bietet es sich auch hier an, \tilde{k}_1 so zu wählen, dass das BSR einen Wertebereich von $[0, 1]$ hat. Die Koeffizienten k_1, k_2, k_3 von (3.43) sind – im Gegensatz zur Wahl von [MK99] – ihrerseits wieder Polynome, welche über empirische Untersuchungen zu

$$\begin{aligned} k_1 &= k_{11} + k_{12} n_{t,\text{cor}} + k_{13} n_{\text{eng}}, \\ k_2 &= k_{21} + k_{22} n_{t,\text{cor}} + k_{23} n_{\text{eng}}, \\ k_3 &= k_{31} + k_{32} n_{t,\text{cor}} \end{aligned}$$

definiert werden. Die Abhängigkeit von der Motordrehzahl n_{eng} soll Pulsationseffekte im Abstrakt berücksichtigen, die durch den gewählten Mittelwertansatz nicht explizit modelliert werden können. Im Rahmen einer Simulation mit dem Gesamtmodell ist es vorteilhaft, das BSR nach unten zu begrenzen. Andernfalls kann ein Zustand mit zu kleiner Turbinendrehzahl erreicht werden, aus dem sich das Modell nicht mehr selbst befreien kann. Dies ist durch das Verlassen des gültigen Wertebereichs der Polynome begründet.

Die Temperatur des Turbinenabstroms wird über einen isentropischen Prozess beschrieben:

$$T_t = T_3 \left(1 - \eta_t \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right)\right).$$

Die isentropische Effizienz wird durch die arbiträre Festlegung der mechanischen Effizienz auf 0,9 approximiert:

$$\eta_t = \frac{\eta_{tm}}{\eta_m} = \frac{\eta_{tm}}{0.9}. \quad (3.45)$$

Die Approximation könnte durch eine drehzahlabhängige Beschreibung verbessert werden. Auch eine Einbeziehung einer Turboladertemperatur, welche die Viskosität des Schmiermittels beeinflusst, ist denkbar. In dieser Arbeit wird darauf allerdings verzichtet. Das grundsätzliche Systemverhalten wird auch durch diese Approximation wiedergegeben.

Die Temperatur des vereinigten Turbinen- und WG-Abgasstroms wird über eine gewichtete Mittelung anhand der Massenströme berechnet:

$$\tilde{T}_{40} = \frac{W_t T_t + W_{wg} T_3}{W_t + W_{wg}}.$$

Analog zum Vorgehen beim Ladeluftkühler (Abschnitt 3.1.4) wird für diese Größe ein Tiefpass

$$\frac{dT_{40}}{dt} = \frac{1}{\tau_{40}} (\tilde{T}_{40} - T_{40}) \quad (3.46)$$

eingeführt, um eine andernfalls entstehende algebraische Schleife zu brechen. Diese tritt in Kombination mit der ETV (Abschnitt 3.1.10) auf. Die ETV beschreibt über (3.49) den Druck p_{40} . Für dessen Berechnung wird die Temperatur T_{40} benötigt, welche über T_t wiederum von p_{40} abhängt. Die Zeitkonstante τ_{40} ist möglichst klein zu wählen, damit ihr Einfluss möglichst gering wird, aber nicht so klein, dass (3.46) den dominanten Eigenwert stellt.

Parameteridentifikation

Die Parameter des Turbinenmassenstroms werden durch Lösen des Optimierungsproblems

$$\begin{aligned} \min \quad & \|\mathbf{e}_{\text{rel}}(W_t)\|_2 \\ \text{u. B. v.} \quad & k_1, k_2, k_3 \in \mathbb{R}^+ \end{aligned}$$

bestimmt.

Um die Referenzwerte zur Bestimmung der Effizienz η_{tm} zu erhalten, können die vom Turbinenhersteller bereitgestellten Daten verwendet werden. Sollen die Referenzwerte hingegen aus Prüfstandsmessungen gewonnen werden, ist eine Berechnung über die Stationaritäts-

bedingung

$$P_c \stackrel{!}{=} \eta_m P_t \quad (3.47)$$

möglich. Mit Gleichung (3.41) gilt die Beziehung

$$\eta_{tm} = \frac{P_c}{P_{t,s}},$$

welche mit (3.17) und (3.42) berechnet werden kann. Unter stationären Bedingungen und bei geschlossenem WG gilt $W_c = W_t = W_{ei}$, womit folgende Werte zu messen sind: $W_{ei}, p_3, p_{40}, T_1, T_{20}, T_3$.

Eine direkte Bestimmung der Effizienz hat den Nachteil, dass die Berechnung der Effizienzen in den Randbereichen, beispielsweise für $\Pi_t \rightarrow 1$, anfällig für Messungenauigkeiten ist. In dem korrespondierenden Optimierungsproblem müssten diese Bereiche entsprechend bewertet werden. Ein anderer Ansatz ist, direkt auf das Leistungsgleichgewicht (3.47) zu optimieren. Mit gleicher Argumentation wie beim Kompressor (Abschnitt 3.1.3) wird der absolute Fehler gewählt:

$$\begin{aligned} \min \quad & \|\mathbf{e}_{\text{abs}}(\eta_{tm} P_{t,s})\|_2 \\ \text{u. B. v.} \quad & k_{13}, k_{21}, k_{22} \in \mathbb{R}^+, \\ & k_{11}, k_{12}, k_{23}, k_{31}, k_{32} \in \mathbb{R}^-. \end{aligned}$$

3.1.10 Abgasklappe: $\zeta_{\text{etv}}, p_{40}, T_{41}$

Die Dynamik der ETV wird wie das WG mit einem PT1-Glied modelliert:

$$\frac{d\zeta_{\text{etv}}}{dt} = \frac{1}{\tau_{\text{etv}}}(r_{\text{etv}} - \zeta_{\text{etv}}). \quad (3.48)$$

Die ETV wird über den Druck p_{40} mit dem Modell verbunden. Analog zur Modellierung des WG (Abschnitt 3.1.8) müsste hier die Düsgleichung Verwendung finden. Wie bereits bei der Modellierung des Luftfilters (Abschnitt 3.1.2) festgestellt wurde, lässt sich die Gleichung nicht nach p_{40} auflösen. Deswegen wird auch hier ein inkompressibles Gas angenommen und ein Drosselansatz verwendet:

$$p_{40} = \frac{p_{41}}{2} + \sqrt{\frac{p_{41}^2}{2} + k_{\text{etv}} R_a T_{40} W_{\text{eo}}^2}. \quad (3.49)$$

Im Gegensatz zum Luftfilter ist die ETV ein Aktuator, daher fließt dessen Position mit in die Gleichung ein, indem die Beschreibung von k_{etv} in Abhängigkeit von der Klappenposition erfolgt. Eine Analyse der Messdaten hat ergeben, dass ein Polynom relativ hoher

Ordnung zur Modellierung nötig wäre, da die Abhängigkeit ein exponentielles Verhalten zeigt. Illustriert ist der Zusammenhang im rechten Diagramm in Abb. 3.3; dort ist der natürliche Logarithmus von k_{etv} aufgetragen. Es ist zu sehen, dass ab ca 35% Aktuierung ein linearer Zusammenhang besteht. Es wird entsprechend stattdessen der natürliche

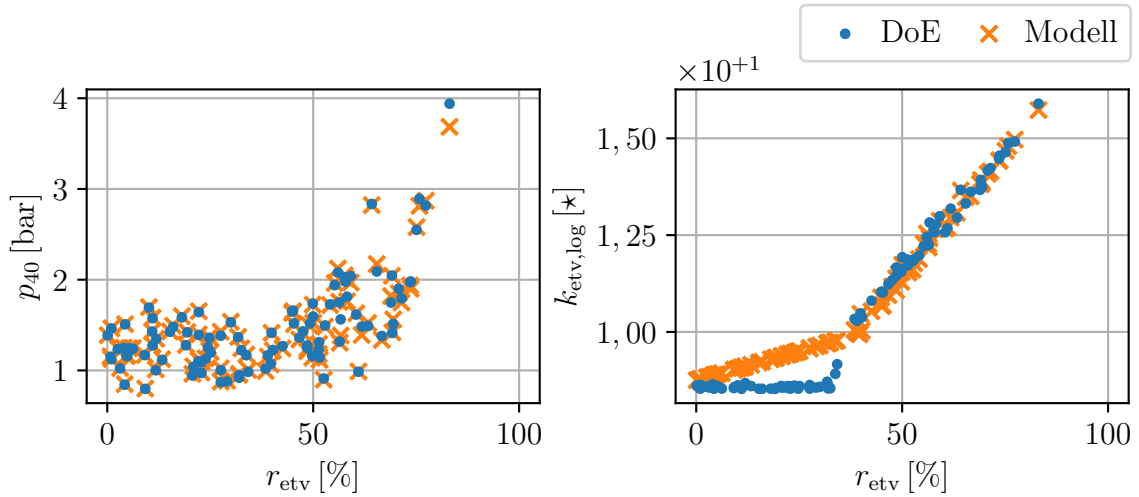


Abb. 3.3: Modell des Drucks vor der Abgasklappe sowie die modellierte Drosselkonstante im Vergleich mit dem DoE (vgl. Abschnitt 3.2.1).

$$\star [k_{\text{etv,log}}] = \log \left(1 / \sqrt{\text{kg m K}^2} \right).$$

Logarithmus $k_{\text{etv,log}}$ der Drosselkonstanten über ein Polynom approximiert:

$$k_{\text{etv}} = e^{\max(\tilde{k}_{\text{etv,log}}, k_{\text{etv,log,min}})},$$

$$\tilde{k}_{\text{etv,log}} = (k_1 + k_2 \zeta_{\text{etv}})(1 + k_3 T_{40} W_{\text{etv}}), \quad (3.50a)$$

$$k_{\text{etv,log,min}} = k_4 + k_5^* \zeta_{\text{etv}}. \quad (3.50b)$$

Der erste Faktor in (3.50a) beschreibt die Abhängigkeit zur Aktuatorposition und ist der dominante von beiden. Der zweite Faktor dient der Kompensation des durch die fehlerhafte Annahme eines inkompressiblen Gases verursachten Fehlers. Über Gleichung (3.50b) wird Gleichung (3.50a) nach unten begrenzt. Zum einen, um einen negativen Drosselwiderstand zu verhindern, und zum anderen, um über k_{etv}^* eine positive Ableitung nach ζ_{etv} zu erzwingen. Wie schon beim WG dient letzteres dazu, die Steuerbarkeit zu erhalten, welche für den Reglerentwurf relevant ist.

Parameteridentifikation

Die Parameter werden über die Lösung von

$$\begin{aligned} \min \quad & \| \mathbf{e}_{\text{abs}}(p_{40}) \|_2 \\ \text{u. B. v.} \quad & k_1, k_2, k_3, k_4 \in \mathbb{R}^+ \end{aligned}$$

ermittelt.

3.1.11 Turboladerdrehzahl: n_{tc}

Die Turboladerdrehzahl wird über die Drehbeschleunigung einer Masse beschrieben:²²

$$\frac{dn_{tc}}{dt} = \frac{1}{2\pi J_{tc}} (M_{tc,t} - M_{tc,m} - M_{tc,c}) = \frac{1}{4\pi^2 J_{tc}} \frac{P_{tm} - P_c}{n_{tc}}. \quad (3.51)$$

Der Turbolader beschleunigt, wenn die Turbine mehr Leistung erbringt, als der Kompressor und die Reibeffekte beanspruchen. Das Trägheitsmoment J_{tc} ist durch die Kenntnis der Materialien sowie der Geometrie der bewegten Teile bekannt. Eine Feintuning durch Optimierung über einen dynamischen Zyklus ist sinnvoll, um andere Effekte, die die Drehzahl beeinflussen, mit in das Trägheitsmoment aufzunehmen.

3.2 Parameteridentifikation und Validierung

In diesem Abschnitt wird der Vorgang der Parameteridentifikation näher ausgeführt. Zuerst werden die Teilmodelle anhand der im vorigen Abschnitt angegebenen Optimierungsprobleme unter Verwendung von stationären Messdaten identifiziert und auf dieser Grundlage bewertet. Es wird gezeigt, wie die Verknüpfung der Teilmodelle die Genauigkeit insgesamt negativ beeinflusst. Darauf aufbauend werden in einer globalen Optimierung unter Verwendung einer dynamischen Messung ausgewählte Teilmodelle vertrimmt, um die Kopplungseffekte in diese aufzunehmen und auf diese Weise das Gesamtmodell zu verbessern. Im letzten Teilabschnitt wird das Modell mit der zur CFD-Simulation korrespondierenden Prüfstandsmessung verglichen und darauf aufbauend die Nutzung von Sensoren motiviert. Es wird eine Aussage getroffen, welche Sensoren zur Stützung des Modells geeignet sind.

Sowohl die zur Identifizierung der Parameter durchgeführten Optimierungen als auch die gezeigten Simulationen basieren auf dem aus der Modellbeschreibung generierten C-Code,

²²Es sei darauf hingewiesen, dass die Drehzahl in Umdrehungen pro Minute gegeben ist.

dessen Generierung in Abschnitt 3.3.1 näher beschrieben wird. Folglich sind damit sämtliche Ungenauigkeiten bedingt durch die numerische Umsetzung der Modelle enthalten. Dieselben Eigenschaften des Modells sind dementsprechend ebenso bei einer Umsetzung auf der ECU zu erwarten. Die Modelle werden anhand der für den temperaturgeführten Motor relevanten Größen bewertet. Diese sind der Ladedruck p_{22} , der Frischluftmassenstrom W_{ei} , die Temperatur in den Auslassfluten T_3 und die Abgastemperatur T_{40} .

3.2.1 Datenbasis zur Identifikation

Zur Bestimmung der Parameter des Modells werden sowohl stationäre als auch dynamische Messungen herangezogen. Beide Messungen wurden unter Verwendung eines bereits existierenden CFD-Simulationsmodells erstellt. Das Simulationsmodell wurde mit der proprietären Software *GT-POWER* von der Firma *Gamma Technologies LLC* erstellt und mit Prüfstandsmessungen des in dieser Arbeit verwendeten Deutz TCD 12.0 V6 abgeglichen.

Die stationären Messungen basieren auf aus einer statistischen Versuchsplanung (engl.: *Design of Experiments*, kurz: DoE) gewonnenen Messdaten. Die DoE besteht aus 8 597 Datenpunkten, welche unter Verwendung eines *Latin-Hypercube-Samplings* [Loh96] durch Variation von n_{eng} , q_{inj} , r_{etv} , p_0 , p_{41} sowie T_0 gewonnen wurden. Der WG-Sollwert wurde über einen im Simulationsmodell enthaltenen Regler vorgegeben und ist daher nicht Teil der aktiv variierten Modelleingänge (vgl. Gleichung 3.3). Die statistischen Eckdaten der DoE sind in Tab. 3.1 aufgeführt.

Die dynamischen Messwerte stammen aus einer Simulation des *Non-Road Transient Cycle* (NRTC). Der NRTC ist ein genormter Zertifizierungs-Testzyklus; nähere Informationen sind in Anhang C zu finden. Für die Simulation des NRTC werden als Modelleingänge Aufzeichnungen einer Prüfstandsmessung verwendet. Auf diese Weise wird die Vergleichbarkeit zwischen dem simulierten und dem am Prüfstand gemessenen Zyklus für die in Abschnitt 3.2.4 besprochene Stützung des Modells durch Sensorinformationen gewährleistet. Der Umgebungsdruck und die Umgebungstemperatur sind konstant und liegen bei 1 bar sowie 32 °C.

3.2.2 Identifikation der Teilmodelle

Die Teilmodelle (TM) werden unabhängig voneinander, d. h. ohne Verwendung der Resultate anderer TM, identifiziert. Auf diese Weise wird sichergestellt, dass die Fehler des einen Modells nicht in die Parameter des anderen einfließen. Die zur Identifikation zu lösenden Optimierungsprobleme wurden bereits im Rahmen der Herleitung der Modelle aufgestellt. Gelöst werden die Optimierungen unter Verwendung des in [SP97] präsentier-

ten *genetischen Algorithmus* für globale Optimierungsprobleme.²³ Der Algorithmus ist für kontinuierliche, nicht differenzierbare, nichtlineare Optimierungsprobleme geeignet, was insbesondere die im Rahmen der Identifikation durchgeführten Optimierungen einschließt. Der Algorithmus umfasst eine größere Klasse von Problemstellungen als gradientenbasierte Algorithmen und stellt damit geringere Ansprüche an die Formulierung der Optimierungsprobleme, wodurch die praktische Nutzung erleichtert wird. Demgegenüber steht allerdings der Nachteil einer geringeren Konvergenzgeschwindigkeit. Die formulierten Optimierungsprobleme sind jedoch klein genug, sodass die Vorteile überwiegen.²⁴ Die einzige Herausforderung besteht darin, geeignete Wertebereiche für die zu optimierenden Parameter zu definieren, sodass der Algorithmus in akzeptabler Zeit zu einer akzeptablen Lösung konvergiert.²⁵ Die zur Parameterbestimmung verwendete Datenbasis ist in [TT21] zu finden.

Die erzielte Genauigkeit der identifizierten TM ist in Tab. 3.2 in den Spalten *TM* aufgeführt.²⁶ Es sei angemerkt, dass sich die in der Tabelle angegebenen Einheiten nicht auf die Spalten RRMSE sowie NRMSE beziehen; diese Fehler sind einheitenlos. Die Spalten $Q_{5\%}$ und $Q_{95\%}$ können als Minimal- und Maximalfehler interpretiert werden. Diese finden Verwendung, um die statistischen Ausreißer zu entfernen. Die Notation Q_p bezeichnet das p -Quantil. Dies bedeutet, dass p Prozent der Daten in dieser Größe kleiner oder gleich dem in der Tabelle angegebenen Wert sind. Bei einer näheren Betrachtung des RRMSE sind die Turbinenleistung und der WG-Massenstrom auffällig. Diese Größen weisen einen Fehler von 8,2 % bzw. 10,5 % Prozent auf, die übrigen Werte liegen unter 3,4 %. Im Falle des WG ist dies durch die Divergenz des relativen Fehlers für kleine Massenströme begründet; der Fehler ist numerisch begründbar. Im Falle der Turbine liegt eine solche Begründung nicht vor, sondern der Fehler liegt an der Modellqualität, insbesondere im Vergleich mit der Genauigkeit der modellierten Kompressorleistung, deren RRMSE bei 0,7 % liegt. Der Fehler hat seine Ursache in der vorgenommenen Modellvereinfachung (vgl. Abb. 1.6 und Abb. 3.1): Es wird vernachlässigt, dass es sich um eine Zwillingstromturbine handelt und das WG lediglich eine der beiden Fluten beeinflusst. In der vorgenommenen Modellvereinfachung (vgl. Abb. 1.2.1 und Abb. 3) wird zwischen den Fluten nicht unterschieden.

²³Es wird die Implementierung `differential_evolution` aus dem *Python*-Paket `SciPy` [VGO⁺20] verwendet.

²⁴Die Anwendung anderer Optimierungsalgorithmen, welche die Differenzierbarkeit ausnutzen, ist zur Beschleunigung der Optimierung denkbar.

²⁵Durch die in den Modellen vorgenommenen Normierungen sind die Wertebereiche ähnlich und damit in der Praxis mit geringem Aufwand festzulegen.

²⁶Die verwendeten Fehlernormen, u.a. RRMSE und NRMSE, werden in Abschnitt A.2 definiert.

Tab. 3.1: Statistische Beschreibung des DoE aus [TT21].

| | | Mittelw. | Min. | Q _{25%} | Q _{50%} | Q _{75%} | Max. |
|----------------|----------|-----------|---------|------------------|------------------|------------------|-----------|
| M_e | [Nm] | 996,890 | 10,266 | 636,817 | 978,250 | 1 354,296 | 2 238,963 |
| M_i | [Nm] | 1 117,119 | 105,254 | 753,666 | 1 093,601 | 1 478,605 | 2 378,217 |
| P_c | [kW] | 12,650 | 0,715 | 3,019 | 6,247 | 15,799 | 109,226 |
| P_e | [kW] | 144,200 | 1,281 | 78,630 | 126,273 | 197,116 | 446,849 |
| P_{tm} | [kW] | 12,637 | 0,714 | 3,015 | 6,244 | 15,778 | 109,244 |
| T_0 | [K] | 294,378 | 249,136 | 279,530 | 294,387 | 309,423 | 333,034 |
| T_1 | [K] | 294,374 | 249,135 | 279,529 | 294,380 | 309,416 | 333,034 |
| T_{20} | [K] | 341,408 | 258,955 | 314,537 | 336,243 | 361,233 | 491,371 |
| T_{21} | [K] | 300,594 | 251,675 | 285,798 | 300,528 | 315,527 | 353,310 |
| T_{22} | [K] | 300,606 | 252,099 | 285,882 | 300,488 | 315,360 | 353,055 |
| T_3 | [K] | 840,145 | 376,949 | 709,364 | 855,900 | 969,265 | 1 273,301 |
| T_{40} | [K] | 805,597 | 365,313 | 671,441 | 814,209 | 934,826 | 1 227,864 |
| T_t | [K] | 805,608 | 365,312 | 671,452 | 814,207 | 934,846 | 1 227,874 |
| W_c | [kg/h] | 723,284 | 180,412 | 448,557 | 635,173 | 911,051 | 2 448,848 |
| W_{ei} | [kg/h] | 723,673 | 180,436 | 448,840 | 635,509 | 911,637 | 2 450,143 |
| W_{eo} | [kg/h] | 756,631 | 193,515 | 470,867 | 665,448 | 952,225 | 2 535,647 |
| W_f | [g/h] | 32,859 | 3,175 | 18,989 | 29,590 | 43,911 | 93,312 |
| W_t | [kg/h] | 717,865 | 169,832 | 455,918 | 638,430 | 893,281 | 2 429,563 |
| W_{wg} | [kg/h] | 38,768 | -0,000 | 0,000 | 0,000 | 0,000 | 539,189 |
| $X_{O_2, eo}$ | [-] | 0,064 | 0,000 | 0,012 | 0,061 | 0,105 | 0,186 |
| m_{cyl} | [g] | 2,795 | 1,275 | 2,149 | 2,566 | 3,237 | 6,970 |
| n_{eng} | [kU/min] | 1,399 | 0,600 | 1,077 | 1,407 | 1,738 | 2,100 |
| n_{tc} | [kU/min] | 45,995 | 20,005 | 32,480 | 42,332 | 57,669 | 99,220 |
| p_0 | [bar] | 0,950 | 0,700 | 0,827 | 0,952 | 1,072 | 1,200 |
| p_1 | [bar] | 0,948 | 0,693 | 0,825 | 0,950 | 1,070 | 1,199 |
| p_{20} | [bar] | 1,417 | 0,738 | 1,110 | 1,309 | 1,622 | 3,476 |
| p_{21} | [bar] | 1,432 | 0,746 | 1,119 | 1,319 | 1,642 | 3,564 |
| p_{22} | [bar] | 1,387 | 0,718 | 1,088 | 1,284 | 1,586 | 3,372 |
| p_3 | [bar] | 1,898 | 0,808 | 1,477 | 1,749 | 2,177 | 4,648 |
| p_{40} | [bar] | 1,477 | 0,724 | 1,171 | 1,349 | 1,588 | 4,371 |
| p_{41} | [bar] | 1,219 | 0,711 | 1,077 | 1,220 | 1,363 | 1,741 |
| $p_3 - p_{22}$ | [bar] | 0,511 | -0,500 | 0,188 | 0,380 | 0,615 | 3,314 |
| q_{inj} | [mg] | 131,969 | 15,300 | 88,100 | 133,148 | 174,600 | 249,900 |
| r_{etv} | [%] | 43,051 | 0,000 | 21,556 | 43,000 | 64,333 | 88,889 |
| r_{wg} | [%] | 15,208 | 0,000 | 0,000 | 0,000 | 0,000 | 100,000 |
| Π_c | [-] | 0,712 | 0,272 | 0,593 | 0,754 | 0,845 | 0,977 |
| Π_t | [-] | 1,313 | 1,050 | 1,130 | 1,210 | 1,385 | 3,457 |
| η_c | [-] | 0,688 | 0,413 | 0,641 | 0,705 | 0,749 | 0,771 |
| η_{tm} | [-] | 0,691 | 0,478 | 0,655 | 0,688 | 0,731 | 0,831 |
| λ | [-] | 1,805 | 0,952 | 1,065 | 1,434 | 2,063 | 9,468 |

Die Notation Q_p bezeichnet das p -Quantil; p Prozent der Datenpunkte sind in dieser Größe kleiner oder gleich dem angegebenen Wert.

Tab. 3.2: Fehler der autark berechneten Teilmodelle (TM) und des Gesamtmodells (GM) an den stationären DoE-Punkten [TT22].

| | RRMSE | | NRMSE | | RMSE | | $Q_{5\%}$ | | $Q_{95\%}$ | |
|-----------------|-------|-------|-------|-------|--------|----------|-----------|-----------|------------|----------|
| | TM | GM | TM | GM | TM | GM | TM | GM | TM | GM |
| n_{tc} [rpm] | - | 0,059 | - | 0,070 | - | 3741,188 | - | -5390,102 | - | 6836,059 |
| p_1 [hPa] | 0,000 | 0,000 | 0,000 | 0,001 | 0,014 | 0,345 | -0,026 | -0,626 | 0,018 | 0,388 |
| p_{20} [hPa] | 0,033 | 0,054 | 0,019 | 0,035 | 55,674 | 93,705 | 8,556 | -134,812 | 113,778 | 165,942 |
| p_{22} [hPa] | - | 0,053 | - | 0,034 | - | 90,372 | - | -134,674 | - | 155,375 |
| p_3 [hPa] | - | 0,049 | - | 0,030 | - | 107,644 | - | -212,416 | - | 102,949 |
| p_{40} [hPa] | 0,032 | 0,059 | 0,021 | 0,031 | 70,724 | 105,310 | -70,938 | -250,661 | 118,349 | 84,069 |
| P_c [kW] | 0,007 | 0,202 | 0,002 | 0,062 | 0,177 | 5,858 | -0,266 | -6,352 | 0,272 | 12,655 |
| P_{tm} [kW] | 0,082 | 0,339 | 0,025 | 0,115 | 2,428 | 10,928 | -5,645 | -8,400 | 1,211 | 25,237 |
| T_{20} [K] | 0,002 | 0,024 | 0,003 | 0,049 | 0,578 | 9,287 | -0,830 | -13,088 | 0,897 | 18,628 |
| T_{22} [K] | 0,002 | 0,004 | 0,004 | 0,011 | 0,463 | 1,164 | -0,790 | -1,798 | 0,706 | 1,984 |
| T_3 [K] | 0,009 | 0,029 | 0,010 | 0,029 | 8,072 | 26,147 | -14,130 | -45,223 | 12,078 | 37,068 |
| T_{40} [K] | 0,012 | 0,048 | 0,013 | 0,051 | 10,494 | 39,628 | -22,662 | -80,564 | 3,720 | 52,108 |
| T_t [K] | 0,017 | 0,046 | 0,020 | 0,049 | 15,816 | 38,168 | -33,422 | -82,832 | 1,810 | 39,846 |
| W_c [kg/h] | 0,030 | 0,082 | 0,012 | 0,047 | 25,338 | 92,915 | -37,890 | -106,338 | 39,354 | 174,831 |
| W_{ei} [kg/h] | 0,025 | 0,062 | 0,007 | 0,026 | 15,943 | 58,294 | -25,690 | -78,278 | 24,511 | 108,717 |
| W_t [kg/h] | 0,022 | 0,163 | 0,012 | 0,084 | 23,527 | 165,213 | -42,691 | -62,794 | 28,964 | 370,822 |
| W_{wg} [kg/h] | 0,105 | 0,052 | 0,031 | 0,020 | 15,581 | 10,990 | -14,201 | -12,956 | 34,658 | 20,587 |

Die Notation Q_p bezeichnet das p -Quantil; p Prozent der Datenpunkte sind in dieser Größe kleiner oder gleich dem angegebenen Wert.

Es ist hervorzuheben, dass im Besonderen die für den temperaturgeführten Motor relevanten Temperaturen T_3 sowie T_{40} einen RMSE von $8,1^\circ\text{C}$ respektive $10,7^\circ\text{C}$ aufweisen. Der Fehler von T_3 spiegelt die Genauigkeit des verwendeten KNN wider, da im stationären Fall T_3 der stationären, vom KNN berechneten Temperatur \tilde{T}_{e0} entspricht (vgl. Gleichungen (3.35) und (3.36)). Die Temperatur T_{40} weist einen geringfügig größeren Fehler auf, mutmaßlich wegen der Annahme einer konstanten mechanischen Effizienz (3.45). Eine Begründung über die Qualität des Turbineneffizienzmodells ist ebenso denkbar. In Abb. 3.4 ist das Abgastemperaturmodell für einen Ausschnitt aus dem NRTC dargestellt. Es wird klar, dass ein stationäres Modell zur zeitlichen Beschreibung der Abgastemperatur nicht ausreichend ist. Je nach Betriebspunktwechsel zeigt sich eine deutliche Über- bzw. Unterschätzung der Abgastemperatur. Die zugehörige Einspritzmenge sowie die zugehörige Drehzahl können Abb. 3.5 entnommen werden. Das dynamische Temperaturmodell T_{e0} zeigt hingegen fast keine sichtbare Abweichung zur Referenz $T_{e0,\text{ref}}$ aus der CFD-Simulation.

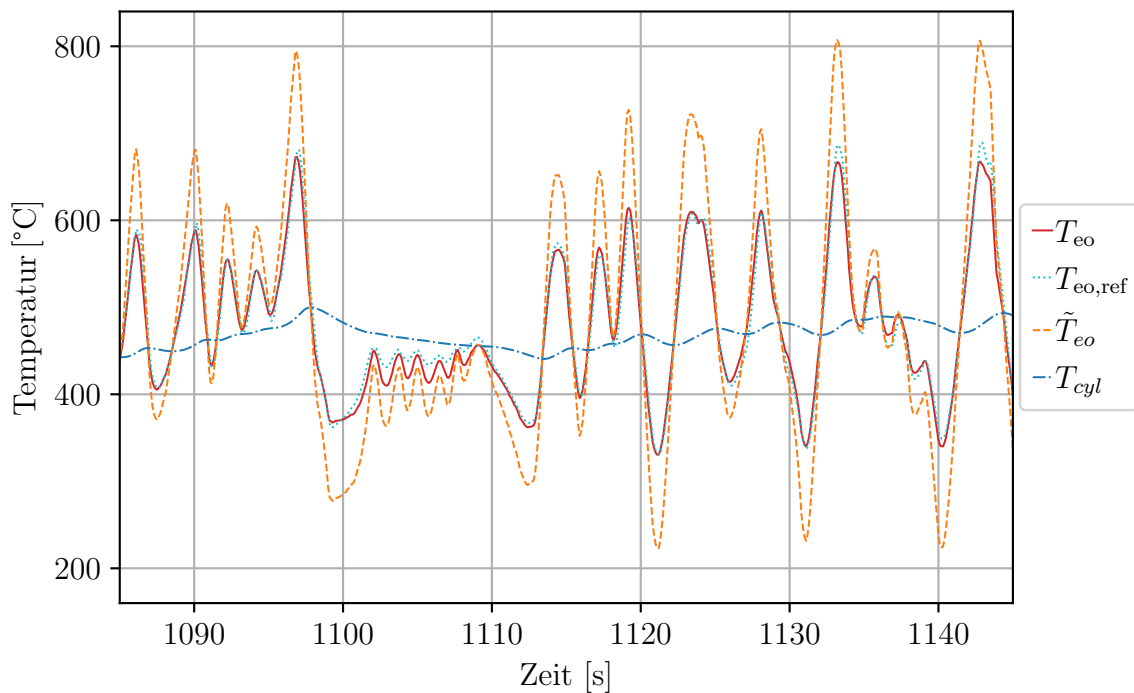


Abb. 3.4: Simulation des Teilmodells für die Abgastemperatur. Die korrespondierende Motordrehzahl sowie das korrespondierende Drehmoment sind in Abb. 3.5 zu finden.

3.2.3 Globale Optimierung

Werden die TM zu einem Gesamtmodell (GM) zusammengefügt und mit den Modelleingängen (3.3) stimuliert, vergrößern sich die Fehler in den einzelnen Größen signifikant. Dies kann durch einen Vergleich der Fehler für das GM in Tab. 3.2 mit den Fehlern der TM

nachvollzogen werden. Die Verschlechterung der Modellgüte kommt durch negative Queinflüsse der TM zustande. Beispielsweise führt ein Unterschätzen der Turbinenleistung zu den folgenden Konsequenzen:

$$P_{\text{tm}} \downarrow \Rightarrow n_{\text{tc}} \downarrow, W_c \downarrow, p_{22} \downarrow, W_{\text{ei}} \downarrow, T_{\text{eo}} \uparrow. \quad (3.52)$$

In Abb. 3.5 ist ein simulierter Ausschnitt des NRTC für die im Rahmen des temperatgeführten Motors relevanten Größen dargestellt. Es ist ersichtlich, dass das qualitative Verhalten des Luftpfadmodells dem der CFD-Simulation entspricht. Insbesondere sind keine gegenläufigen Bewegungen zu beobachten. Gerade dies ist im Hinblick auf einen Reglerentwurf von Bedeutung, da andernfalls der Regler in bestimmten Situationen u.U. in eine falsche Richtung aktuiert. Nichtsdestotrotz ist eine Verschiebung der Konturen zu beobachten, welche im ersten Drittel der dargestellten Zeitspanne besonders ausgeprägt ist. Die Art der Verschiebung entspricht den in (3.52) angestellten Überlegungen.

Um den Fehler zu reduzieren, wird bei einer Adaption der Parameter des Turbinen-Effizienzmodells (3.43) angesetzt, da dieses mit der Turbinenleistung (3.41) zusammenhängt, welche als Ausgangspunkt des Modellfehlers angenommen wird. Die Annahme beruht auf der Tatsache, dass bei der Betrachtung der Modellfehler im vorigen Abschnitt dieses Modell den größten Fehler aufwies. Die Parameter des Effizienzmodells werden im Rahmen einer globalen, dynamischen Optimierung durch Lösen des Optimierungsproblems

$$\begin{aligned} \min \quad & \frac{1}{T} \int_0^T e_{\text{rel}}(W_{\text{ei}}(\mathbf{x}(t), \mathbf{u}(t)))^2 dt \\ \text{u. B. v} \quad & \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) := \mathbf{x}_0, \\ & k_{13}, k_{21}, k_{22} \in \mathbb{R}^+, \\ & k_{11}, k_{12}, k_{23}, k_{31}, k_{32} \in \mathbb{R}^- \end{aligned}$$

verbessert. Die Referenzwerte stammen aus dem NRTC. Gelöst wird dieses Problem mit dem im vorigen Abschnitt verwendeten genetischen Algorithmus. In der Optimierung wird der relative Fehler des Frischluftmassenstroms minimiert, da sich dieser in numerischen Experimenten als für die gesamte Modellqualität besonders relevant herausgestellt hat. Eine mehrkriterielle Optimierung ist ebenso denkbar, beispielsweise unter zusätzlicher Berücksichtigung der Abgastemperatur. Die höhere Güte des global optimierten GM (GMGO) ist ebenfalls in Abb. 3.5 dargestellt. Es ist ersichtlich, wie sich das Modell in jeder Größe verbessert, und zwar ohne sich an anderen Stellen zu verschlechtern. Die CFD-Simulation und das Luftpfadmodell können als näherungsweise identisch angesehen werden. Es sei angemerkt, dass die Einbeziehung weiterer Parameter in das Optimierungsproblem lediglich in begrenztem Maße möglich ist, da für jeden vom Optimierer zu evaluierenden Parametersatz das Luftpfadmodell für den gesamten NRTC zu simulieren ist. Entsprechend steigt die Anzahl der erforderlichen NRTC-Simulationen mit der Anzahl der zu optimierenden Parameter überproportional. Ab einer gewissen Parameteranzahl ist es zielführender, dedizierte Algorithmen für die dynamische Optimierung (vgl. Abschnitt 2.3) zu verwenden.

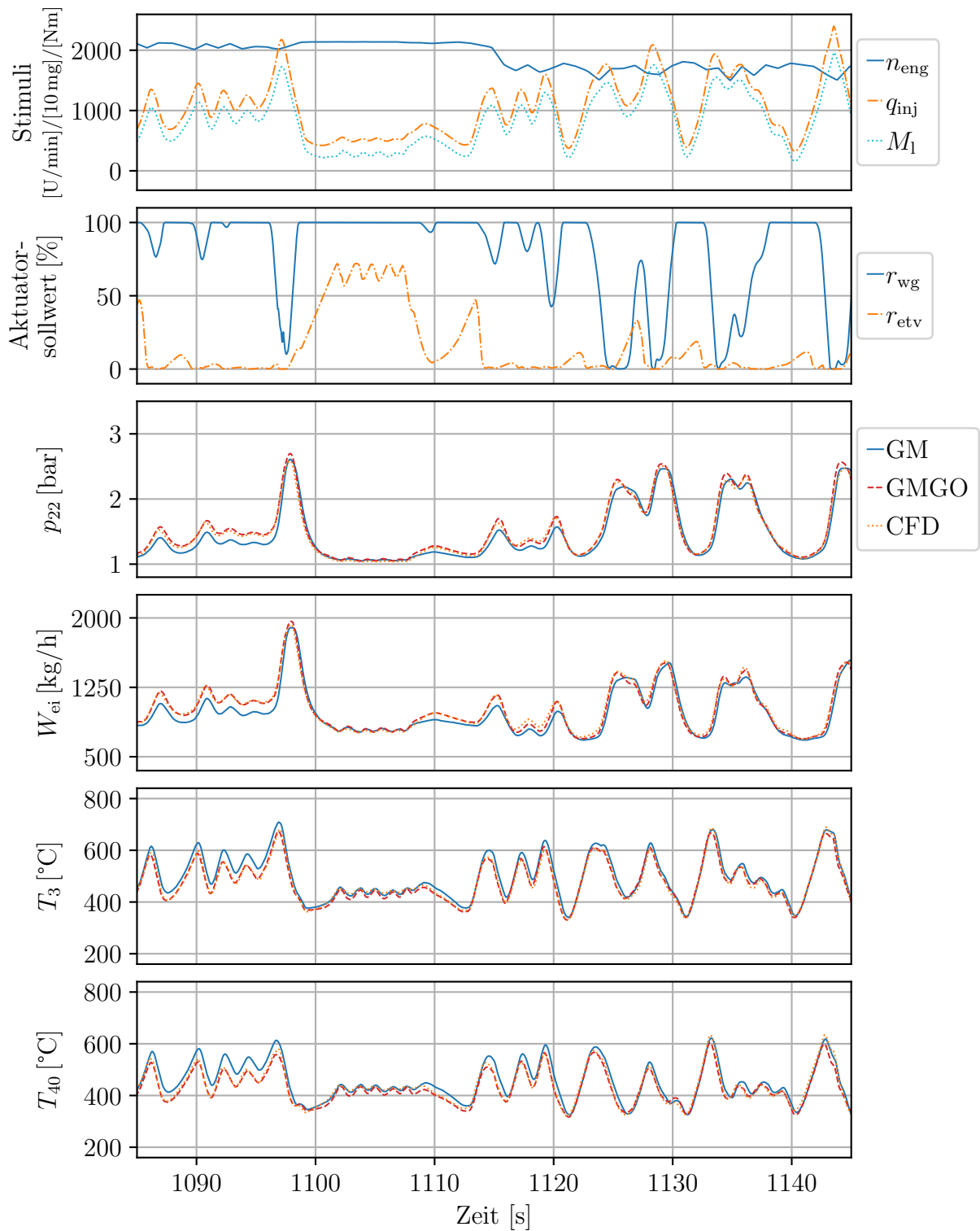


Abb. 3.5: Güte des lokal (GM) und des global (GMGO) optimierten Luftpfadmodells im Vergleich zur CFD-Simulation.

3.2.4 Stützung des Luftpfadmodells mit Sensorinformationen

Bis jetzt wurde das Luftpfadmodell mit einer CFD-Simulation verglichen, also ein Modell mit einem anderen Modell. Reale Systeme sind jedoch modelltechnisch nicht oder lediglich mit erheblichem Aufwand erfassbaren Einflüssen bzw. Veränderungen unterworfen. Beispiele hierfür sind produktionsbedingte Unterschiede zwischen den einzelnen Motoren oder ein durch Alterungseffekte bzw. Verschleiß verändertes Systemverhalten. Um diesen Effekten Rechnung zu tragen, bietet sich eine Stützung des Modells durch geeignete Sensoren an. Es soll im Weiteren die Frage beantwortet werden, welche Verbesserungen durch die Nutzung von Sensoren zu erwarten sind und welcher Sensor am besten geeignet ist.

Grundsätzlich geschieht eine Modellkorrektur mittels Sensoren über einen Zustandsbeobachter [Lue71], beispielsweise über ein Kálmán-Filter [WB95], welcher den Fehler zwischen dem Sensormesswert und dem korrespondierenden Modellwert auf die Modellzustände zurückkoppelt und diese dadurch korrigiert.²⁷ Für solch eine Korrektur bietet der Deutz TCD 12.0 V6 einen Druck- und einen Temperatursensor in der Einlassflut sowie einen Drucksensor in der Auslassflut.²⁸ Um den Entwurf eines Beobachters zu umgehen, aber dennoch eine Aussage über die Stützung des Modells durch die Sensoren treffen zu können, wird die zum jeweiligen Sensor korrespondierende DGL durch den Sensormesswert ersetzt. Zur Auswertung wird der DoE mit dem GM nachsimuliert, wobei jeweils der zu dem zu untersuchenden Sensor korrespondierende Zustand durch den Wert der CFD-Simulation ersetzt wird; es wird mit einem perfekten Sensor ohne Messverzögerung gearbeitet.²⁹ Aus diesem Grund wird auf eine Bewertung des Temperatursensors verzichtet. Der Temperatursensor zeigt ein Tiefpassverhalten mit einer Zeitkonstanten in der Größenordnung von zehn Sekunden. Die Ergebnisse wären damit nicht auf eine praktische Anwendung übertragbar. Die Drucksensoren zeigen hingegen nahezu kein verzögerndes Verhalten. Zusätzlich wird die Nutzung eines Turboladerdrehzahlsensors betrachtet, dessen Einsatz ebenfalls als direkter Zustandsersatz denkbar ist.

In Tab. 3.3 sind in den Spalten p_{22} , p_3 und n_{tc} die Fehler unter Verwendung des jeweiligen Sensors aufgeführt. Die Spalte p_{22}/p_3 betrachtet den Fall, dass sowohl der Ladedruck- als auch der Auslassflutdrucksensor verwendet werden. Zur einfacheren Vergleichbarkeit sind in der unbeschrifteten Spalte die Werte des GM aus Tab. 3.2 aufgeführt. Prinzipiell kann beobachtet werden, dass durch den p_{22} -Sensor sowie durch den n_{tc} -Sensor eine Verbesserung der Modellgüte eintritt. Insbesondere die für den temperaturgeführten Motor als wichtig empfundenen Größen p_{22} , W_{ei} , T_3 sowie T_{40} werden durch die Sensornutzung genauer. Lediglich W_{wg} und W_t und für p_{22} zusätzlich W_c und p_{40} verschlechtern sich. Die Verwendung des p_3 -Sensors führt zu einer Verschlechterung des Modells. Ursächlich könnte die geringe Qualität des Turbinenmodells sein, die durch die Verwendung des Auslassflut-

²⁷Eine umfassende Betrachtung verschiedener Beobachteransätze ist in der Dissertation [Irl16] zu finden.

²⁸Vor dem Luftfilter ist ein Druckschalter mit einer Schaltschwelle von ca. 50 mbar installiert, dieser ist für eine fortlaufende Modellkorrektur ungeeignet.

²⁹Es wird das GM anstelle des akkurateren GMGO verwendet, um die Effekte durch die verwendeten Sensoren stärker herauszustellen.

drucks als Sensor stärker betont wird. Offen bleibt die Frage, ob der Drehzahlsensor in Kombination mit einem Kompressormodell geringerer Güte ähnliche Konsequenzen hätte. Bemerkenswert ist zudem die Erkenntnis, dass eine gemeinsame Verwendung der beiden Drucksensoren zu dem größten Fehler der Turboladerdrehzahl führt. Bei Verwendung eines Sensors ist das System selbststabilisierend: Wird die Turboladerdrehzahl über- bzw. unterschätzt, verändern sich die Drücke und Massenströme derart, dass sie dem Fehler entgegenwirken. Bei der Verwendung beider Drücke bleibt diese Wirkung aus. Bei der alleinigen Verwendung von p_3 kann sich beispielsweise bei einer Überschätzung der Turboladerdrehzahl ein höherer Ladedruck p_{22} aufbauen, der der erhöhten Drehzahl entgegen wirkt.

Tab. 3.3: Modellgenauigkeit unter Verwendung von Sensoren zur Modellstützung [TT22].

| | RRMSE | | | | | NRMSE | | | | |
|----------|-------|----------|-------|----------|--------------|-------|----------|-------|----------|--------------|
| | - | p_{22} | p_3 | n_{tc} | p_{22}/p_3 | - | p_{22} | p_3 | n_{tc} | p_{22}/p_3 |
| n_{tc} | 0,059 | 0,020 | 0,058 | 0,000 | 0,140 | 0,070 | 0,023 | 0,065 | 0,000 | 0,139 |
| p_1 | 0,000 | 0,000 | 0,000 | 0,000 | 0,000 | 0,001 | 0,000 | 0,001 | 0,000 | 0,000 |
| p_{20} | 0,054 | 0,003 | 0,101 | 0,006 | 0,003 | 0,035 | 0,002 | 0,053 | 0,003 | 0,002 |
| p_{22} | 0,053 | 0,000 | 0,102 | 0,006 | 0,000 | 0,034 | 0,000 | 0,054 | 0,004 | 0,000 |
| p_3 | 0,049 | 0,049 | 0,000 | 0,049 | 0,000 | 0,030 | 0,028 | 0,000 | 0,027 | 0,000 |
| p_{40} | 0,059 | 0,060 | 0,055 | 0,061 | 0,058 | 0,031 | 0,029 | 0,025 | 0,030 | 0,030 |
| P_c | 0,202 | 0,092 | 0,223 | 0,017 | 0,326 | 0,062 | 0,026 | 0,057 | 0,004 | 0,079 |
| P_{tm} | 0,339 | 0,346 | 0,352 | 0,347 | 0,376 | 0,115 | 0,119 | 0,101 | 0,111 | 0,106 |
| T_{20} | 0,024 | 0,005 | 0,022 | 0,003 | 0,007 | 0,049 | 0,011 | 0,044 | 0,006 | 0,014 |
| T_{22} | 0,004 | 0,002 | 0,006 | 0,002 | 0,002 | 0,011 | 0,005 | 0,017 | 0,005 | 0,007 |
| T_3 | 0,029 | 0,013 | 0,048 | 0,014 | 0,009 | 0,029 | 0,012 | 0,045 | 0,013 | 0,008 |
| T_{40} | 0,048 | 0,016 | 0,049 | 0,017 | 0,015 | 0,051 | 0,018 | 0,052 | 0,019 | 0,016 |
| T_t | 0,046 | 0,016 | 0,047 | 0,016 | 0,016 | 0,049 | 0,018 | 0,050 | 0,018 | 0,017 |
| W_c | 0,082 | 0,101 | 0,109 | 0,021 | 0,302 | 0,047 | 0,052 | 0,049 | 0,009 | 0,162 |
| W_{ei} | 0,062 | 0,023 | 0,111 | 0,023 | 0,026 | 0,026 | 0,007 | 0,030 | 0,006 | 0,008 |
| W_t | 0,163 | 0,173 | 0,167 | 0,168 | 0,203 | 0,084 | 0,083 | 0,075 | 0,078 | 0,090 |
| W_{wg} | 0,052 | 0,077 | 0,203 | 0,149 | 0,193 | 0,020 | 0,039 | 0,057 | 0,041 | 0,052 |

Insgesamt kann festgehalten werden, dass sowohl der Ladedrucksensor als auch ein Drehzahlsensor zur Modellstützung geeignet sind. Obwohl der Drehzahlsensor insgesamt eine stärkere Verbesserung zeigt, wird die Nutzung des Ladedrucksensors wegen der zuvor aufgeworfenen Fragestellung empfohlen. Des Weiteren hängt der Ladedruck direkt mit dem Frischluftmassenstrom zusammen (vgl. Gleichung (3.27) und (3.28)), welcher für das Verbrennungsluftverhältnis entscheidend ist. Dieses Verhältnis stellt eine zentrale Größe bei der Kalibrierung eines Dieselmotors dar.

In Abb. 3.6 sind das GMGO, dasselbe Modell unter Verwendung des Ladedrucksensors (GMGO + p_{22}) sowie eine Prüfstandsmessung (PS) für einen Ausschnitt aus dem

NRTC dargestellt. Es ist zu sehen, wie sowohl beim Ladedruck p_{22} als auch beim Frischluftmassenstrom W_{ei} durch die Sensorkorrektur eine Verbesserung eintritt. Das sich näherungsweise gleich verhaltende GMGO und das CFD-Modell weichen in diesen Größen sichtbar von der PS ab. Beim Druck in der Auslassflut führt die Sensorverwendung zu einer leichten Verbesserung, jedoch weniger deutlich als bei den beiden anderen Größen. Im Bereich $1100\text{ s} < t < 1110\text{ s}$ ist zu beobachten, wie sämtliche Modelle ein qualitativ anderes Verhalten zeigen als die PS. Die dynamische Simulation spiegelt damit die in Tab. 3.2 anhand von stationären Betriebspunkten evaluierten Verbesserungen wider.

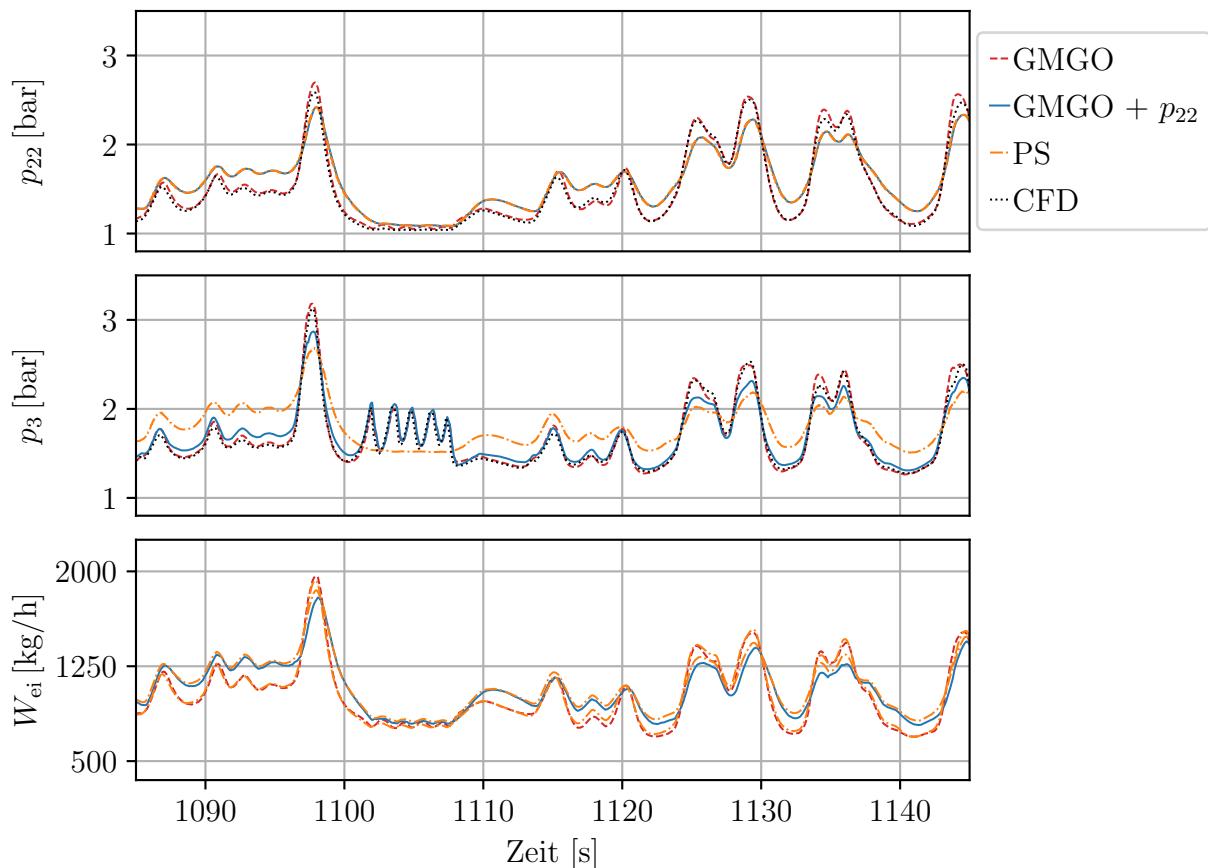


Abb. 3.6: Güte des unkorrigierten (GMGO) und des korrigierten (GMGO + p_{22}) Luftpfadmodells sowie der CFD-Simulation im Vergleich mit einer Prüfstandsmessung (PS). Die zugehörige Drehzahl und das zugehörige Drehmoment können Abb. 3.5 entnommen werden.

3.3 Rechentechnische Umsetzung

Das Luftpfadmodell soll im Rahmen einer MPC (vgl. Abschnitt 4.1) verwendet werden. Die MPC erfordert pro Zeitschritt – der Algorithmus soll alle 50 ms ausgeführt werden –

mehrere Simulationen des Modells für einen Zeitraum von über einer Sekunde in die Zukunft. Um dies unter den begrenzten hardwaretechnischen Voraussetzungen der ECU (vgl. Tab. 1.1) zu ermöglichen, ist eine performante rechen-technische Umsetzung des Luftpfadmodells vonnöten. Die ECU wird unter Verwendung der Programmiersprache *C* programmiert. Im Folgenden wird die Erzeugung von C-Code aus der mathematischen Beschreibung und dessen Eignung für eine echtzeitfähige Umsetzung der MPC besprochen.

3.3.1 Beschreibung der C-Code-Generierung

Erzeugt wird der C-Code unter Verwendung eines im Rahmen dieser Arbeit entstandenen Codegenerators. Im Folgenden sollen die wesentlichen Prozessierungsschritte erklärt werden. Für das Verständnis der übrigen Abschnitte ist dieser Abschnitt nicht erforderlich und kann beim Lesen übersprungen werden.

Der Codegenerator setzt auf dem Python-Paket `SymPy` [MSP⁺17] auf. `SymPy` ist ein Paket, welches die symbolische Beschreibung mathematischer Ausdrücke ermöglicht. Dieses Paket wird genutzt, um die in Abschnitt 3.1 beschriebenen Gleichungen zu erfassen. Das Resultat ist eine Liste an Ausdrücken, die ineinander eingesetzt werden können. Diese Liste lässt sich als gerichteter Graph interpretieren. Jeder Knoten des Graphen ist gemäß dieser Betrachtung eine Gleichung bzw. eine Zuweisung. Die *Eltern* eines solchen Knotens sind die Gleichungen (Symbole), die zur Auswertung des Knotens benötigt werden. Die *Kinder* eines Knotens sind demgemäß die Knoten, die von der Berechnung des betrachteten Knotens abhängen.

In einem ersten Prozessierungsschritt werden dem Graphen sämtliche erforderlichen Ableitungen hinzugefügt. Dies geschieht unter einer Mischung von algorithmischer Differentiation (AD) [Gri08] und symbolischer Differentiation. Die einzelnen Gleichungen werden symbolisch differenziert, die Gleichung als Ganzes wird als Element einer AD interpretiert. Über letztere wird ermittelt, nach welchen Symbolen die jeweilige Gleichung zu differenzieren ist – es wird der *Backward-Mode* genutzt.

Im nächsten Schritt wird in sämtlichen Gleichungen nach gemeinsamen Teilausdrücken gesucht und diese werden durch eigene Symbole (Variablen) ersetzt. Die Ausdrücke hinter den Symbolen werden dem Graphen hinzugefügt. Durch diesen Schritt wird eine Mehrfachberechnung gleicher Teilausdrücke vermieden.

Der letzte Prozessierungsschritt des Graphen beinhaltet eine Abspaltung sämtlicher Summanden der Gleichungen. Auf diese Weise werden die einzelnen Gleichungen kürzer und enthalten lediglich Funktionsaufrufe oder Multiplikationen. Kürzere Gleichungen weisen i. d. R. weniger Abhängigkeiten auf, was in einer programmiertechnischen Umsetzung eine geringere Anzahl an zu speichernden Variablen zur Folge hat.³⁰

³⁰Eine Aufspaltung der Multiplikationen wurde untersucht, diese hat jedoch zu einem größeren und lang-

Der Graph enthält zu diesem Zeitpunkt sämtliche erforderlichen Berechnungen. Zur Erzeugung von ausführbarem Programmcode, beispielsweise zur Auswertung von $\partial \mathbf{f}(\mathbf{x}, \mathbf{u}) / \partial \mathbf{x}$, werden aus dem Graphen die für diese Berechnung nicht benötigten Knoten entfernt. Die übrigen Knoten werden in eine Prozessierungsreihenfolge gebracht; es wird ein Programm erzeugt. Ein Programm geht aus einer topologischen Reihenfolge hervor. Jedoch ist eine solche topologische Reihenfolge nicht eindeutig und führt zu unterschiedlich speicherintensivem Programmcode. Eine schlechte Reihenfolge ist beispielsweise eine solche, in der ein bestimmter Knoten zu Anfang berechnet, aber erst am Ende des Programms benötigt wird. Dieser Knoten beansprucht über die gesamte Programmlaufzeit einen Speicherplatz. Eine in diesem Sinne *gute* Reihenfolge zeichnet aus, dass ein berechneter Knoten des Graphen möglichst schnell das letzte Mal verwendet wird und der zum Speichern des Knotens verwendete Speicher frühzeitig erneut zum Speichern eines anderen Knotens verwendet werden kann. Das zugrundeliegende Optimierungsproblem ist das sogenannte *Linear Arrangement Problem*, welches NP-schwer ist [Sei10]. Entsprechend wird sich dessen Lösung über eine Heuristik genähert. In dem Graphen werden sämtliche Teilbäume identifiziert. Die Knoten eines Baumes zeichnen sich dadurch aus, dass sie genau einen Kindknoten besitzen. Als Wurzelknoten für die Teilbäume werden die Knoten ausgewählt, die mehrere Kindknoten haben. Die Teilbäume an sich werden über eine Tiefensuche in eine topologische Reihenfolge gebracht, welche im Sinne des Speichers optimale Ergebnisse liefert. Die Teilbäume untereinander werden ebenfalls über eine Tiefensuche in eine Prozessierungsreihenfolge gebracht, diese Reihenfolge ist hingegen nicht optimal.

Das auf diese Weise erhaltene Programm wird in einem letzten Schritt knotenweise in Zeilen C-Code übertragen. Bei der Übertragung wird der Berechnungsaufwand bestimmter mathematischer Operatoren/Funktionen berücksichtigt. Es werden die Standardoperatoren (+, −, *, /), die Quadratwurzel, die Exponentialfunktion, der Logarithmus und die Potenzfunktion verwendet.³¹ Die bedeutendste vorgenommene Ersetzung ist die der Potenzfunktion, da diese die höchsten Rechenansprüche stellt (vgl. Tab. 3.4). Die Potenzfunktion mit der Basis $b \in \mathbb{R}^+$ und einem Exponenten $a \in \mathbb{R}$ wird über

$$\text{pow}(b, a) := b^a = e^{\log(b^a)} = e^{a \log(b)} \quad (3.53)$$

berechnet. Es sind entsprechend ein Logarithmus und eine Exponentialfunktion auszuwerten sowie eine Multiplikation durchzuführen. Ganzzahlige Potenzen, insbesondere ganzzahlige Potenzen kleiner Ordnung, lassen sich über wenige Multiplikationen durch Ausnutzen des Potenzgesetzes

$$b^{a_1+a_2} = b^{a_1} b^{a_2} \quad (3.54)$$

schneller berechnen. Daher werden Potenzfunktionen mit ganzzahligem Exponenten durch *geschickte* Multiplikationen ersetzt. Ein Nebeneffekt dieser Ersetzungen ist, dass im Falle

sameren Programmcode geführt. Ursächlich könnten compilerseitige Optimierungen sein, die bei einer solchen Aufteilung schlechter greifen.

³¹Die trigonometrischen Funktionen und die Hyperbelfunktionen werden nicht betrachtet; diese kommen in der mathematischen Beschreibung des Luftpfades nicht vor.

Tab. 3.4: Ausführungszeit und Größe des Funktionsstacks. Der Code wurde mit einem GCC unter Verwendung der Optimierungseinstellung `-O2` kompiliert und auf einem *Intel i5-8250U*-System ausgewertet. Die Funktionen mit angefügtem `math.h` basieren auf der C-Standardbibliothek.

| Funktion | Ausführungszeit [ns] | Funktionsstack [Byte] |
|--|-------------------------|--------------------------|
| $f(\mathbf{x}, \mathbf{u})$ | 272,3 | 336 |
| $f(\mathbf{x}, \mathbf{u})$ (<code>math.h</code>) | 1 676,0 | 336 |
| $df(\mathbf{x}, \mathbf{u})/d\mathbf{x}$ | 504,4 | 496 |
| $df(\mathbf{x}, \mathbf{u})/d\mathbf{x}$ (<code>math.h</code>) | 2 888,5 | 496 |
| <code>exp</code> | 2,7 | 8 |
| <code>expf</code> (<code>math.h</code>) | 42,8 | - |
| <code>log</code> | 2,7 | 8 |
| <code>logf</code> (<code>math.h</code>) | 29,6 | - |
| <code>lut2d</code> | 17,7 | 160 |
| <code>pow</code> | 5,5 | 8 |
| <code>powf</code> (<code>math.h</code>) | 104,3 | - |
| <code>sqrt</code> | 10,0 | 8 |
| <code>sqrtf</code> (<code>math.h</code>) | 29,7 | - |

von Polynomen der Compiler die Möglichkeit hat, Zwischenergebnisse der Multiplikationen auszunutzen. Neben der Ersetzung der Potenzfunktion werden sämtliche auswertbaren Ausdrücke ausgewertet; beispielsweise $\sqrt{2}$.

3.3.2 Eigenschaften des generierten C-Codes

In Tab. 3.5 ist die Anzahl der Operationen und Funktionsaufrufe der beiden zur Integration der *kanonischen DGLs* (4.16) verwendeten C-Funktionen aus dem Luftpfadmodell dargestellt. Es sollen die Unterschiede zur mathematischen Beschreibung in Abschnitt 3.1 erklärt werden. Die Anzahl der min-/max-Operationen ist größer als in den angegebenen Berechnungen. Ursächlich hierfür sind zusätzlich eingefügte Polynombegrenzungen, um mathematisch undefinierte Zustände zu vermeiden, wie beispielsweise eine Division durch Null. Die Anzahl an Aufrufen der Potenzfunktion (`pow`) ist, bedingt durch die zuvor besprochenen rechentechnischen Umformulierungen, geringer und dementsprechend die Anzahl der Multiplikationen größer. Eine weitere Besonderheit ist die Verwendung der zweidimensionalen Lookup-Tabelle (`lut2d`), welche zur Umsetzung der spezifischen Wärmekapazität c_p verwendet wird. Die Lookup-Tabelle wird sowohl zur Umsetzung der DGLs ($f(\cdot)$) als auch zur Berechnung der Systemmatrix $\partial f(\cdot)/\partial \mathbf{x}$ fünf mal verwendet. Erwartungsgemäß

Tab. 3.5: Anzahl der durchgeführten Operationen und Funktionsaufrufe im generierten C-Code.

| Operation / Funktion | $f(\mathbf{x}, \mathbf{u})$ | $\frac{df(\mathbf{x}, \mathbf{u})}{d\mathbf{x}}$ |
|----------------------|-----------------------------|--|
| + | 38 | 143 |
| - | 40 | 116 |
| * | 145 | 605 |
| / | 25 | 73 |
| exp | 2 | 2 |
| log | - | 1 |
| pow | 12 | 24 |
| sqrt | 7 | 7 |
| max | 28 | 26 |
| min | 18 | 16 |
| heaviside | - | 23 |
| lut2d | 5 | 5 |

müsste zur Berechnung letzterer die Anzahl der Aufrufe höher sein.³² Dass dies nicht so ist, beruht auf einer Design-Entscheidung: Die Implementierung der Lookup-Tabelle berechnet neben dem Wert zusätzlich die Ableitungen, weil der rechentechnische Mehraufwand geringer ist, als die Tabellenposition an sich zu ermitteln. Dies führt dazu, dass die Systemmatrix weniger, aber dafür die DGLs einen höheren Rechenaufwand erfordern. In Tab. 3.4 ist ersichtlich, dass die DGLs trotz dieser Entscheidung etwa halb so viel Rechenaufwand erfordern.

Durch eine Neuimplementierung der transzendenten Funktionen (exp, log), von der die Potenzfunktion (3.53) unmittelbar profitiert, sowie der Neuimplementierung der Wurzelfunktion, wird die erforderliche Rechenzeit zur Auswertung der DGLs sowie der Systemmatrix um jeweils ca. 83 % reduziert. Die unterschiedlichen Ausführungszeiten sind in Tab. 3.4 aufgeführt. Die Neuimplementierung der Exponentialfunktion sowie des Logarithmus basiert auf dem in [Sch99] präsentierten Ansatz zur schnellen Berechnung der Exponentialfunktion. Dieser Ansatz wird mit einer Lookup-Tabelle zur Verbesserung der Genauigkeit kombiniert. Der relative Fehler sowohl für die Exponentialfunktion als auch für den Logarithmus liegt bei etwa 0,1 % für den kompletten Wertebereich. Der Fehler wirkt sich nicht maßgeblich auf die erreichte Modellgüte aus, da die Modellfehler mindestens eine Größenordnung darüber liegen (vgl. Tab. 3.2). Die Quadratwurzel kann ebenfalls mit dem Ansatz nach [Sch99] verbessert werden, wobei hier keine Lookup-Tabelle zur Steigerung der Genauigkeit verwendet wird, sondern ein Newton-Verfahren. Die verwendete Umsetzung hat einen Fehler, der unter 0,176 % liegt.

³²Die Wärmekapazität $c_{p,3}$ hängt von der Temperatur T_3 ab, welche sich aus dem Zustand T_{cyl} berechnet (vgl. Gleichungen (3.36) und (3.35)). Durch die Differentiation nach T_{cyl} sind mehrere Auswertungen zu erwarten.

3.3.3 Bewertung der Echtzeitfähigkeit

Es soll die Echtzeitfähigkeit des Luftpfadmodells unter Verwendung des generierten C-Codes bewertet werden. Dazu wird zunächst ermittelt, mit welcher Schrittweite die numerische Integration durchzuführen ist, damit diese in numerisch stabiler Weise erfolgt (vgl. Abschnitt 2.1.2). Anschließend wird unter Verwendung der ermittelten Schrittweite sowie der zuvor genannten Ausführungszeiten der Echtzeitfaktor (1.4) angegeben.

Um die Zustände zu einem bestimmten Zeitpunkt zu berechnen, ist die nichtlineare DGL (2.1a) numerisch zu integrieren. Zur Abschätzung der erforderlichen Schrittweite soll das *explizite Euler-Verfahren* betrachtet werden. Die Quadraturformel ist nach Gleichung (2.9) über

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \quad (3.55)$$

mit der Schrittweite $h > 0$ gegeben. Es wird die Notation $\mathbf{x}_k := \mathbf{x}(hk)$ verwendet, wobei k den Zeitschritt bezeichnet. Diese Formel stellt eine Diskretisierung des zeitkontinuierlichen Systems (2.1a) dar. Es soll ein Maß zur Bewertung der erforderlichen Abtastzeit zur numerischen Integration hergeleitet werden. Im Weiteren sollen die Stabilitätsaspekte betrachtet werden, daher wird mit dem autonomen Anteil des Systems fortgefahren. Durch eine *Taylor-Approximation* an einer Ruhelage $(\mathbf{x}_R, \mathbf{u}_R) \in \mathcal{R}$ nach Definition 2.1.1 kann Gleichung (3.55) zu

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h\mathbf{A}(\mathbf{x}_R, \mathbf{u}_R)(\mathbf{x}_k - \mathbf{x}_R) \quad (3.56)$$

unter Verwendung der Jakobi-Matrix (vgl. Gleichung (2.3))

$$\mathbf{A}(\mathbf{x}, \mathbf{u}) := \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}, \mathbf{u}}$$

umformuliert werden. Um Aussagen über die Stabilität der auf diese Weise gewonnenen Linearisierung machen zu können, wird die Stabilitätsfunktion (2.10) des expliziten Euler-Verfahrens, welche über

$$g(z_i) = 1 + z_i, \quad z_i := h\lambda_i$$

gegeben ist, herangezogen. Die Integrationsschrittweite ist derart zu wählen, dass für sämtliche Eigenwerte λ_i

$$|g(z_i)| = |1 + h\lambda_i| < 1$$

gilt (vgl. Abschnitt 2.1.3). Dies ist der Fall, wenn

$$h < \min_i \left\{ \frac{-2\operatorname{Re}\{\lambda_i\}}{\operatorname{Re}\{\lambda_i\}^2 + \operatorname{Im}\{\lambda_i\}^2} \right\} \quad (3.57)$$

erfüllt ist.³³ Es wird ersichtlich, dass für instabile Systeme mit $\text{Re}\{\lambda_i\} > 0$ die berechnete Schrittweite stets negativ ist, wodurch die Forderung einer positiven Schrittweite verletzt wird. Ebenso kann festgehalten werden: Je kleiner der kleinste Realteil aller Eigenwerte λ_i ist, desto kleiner ist die nötige Schrittweite zur stabilen Integration des Systems.³⁴

Die Auswertung der Systemmatrix $\mathbf{A}(\cdot)$ an den DoE-Punkten ergibt, dass sämtliche Eigenwerte negativ sind und das System damit an den Linearisierungen stabil ist. Dies ist das zu erwartende Resultat, da die DoE-Punkte unter Verwendung konstanter Systemeingänge ermittelt werden und das System unter diesen Umständen nicht divergiert. Es sei angemerkt, dass dies kein Stabilitätsbeweis ist, sondern lediglich eine Aussage über die Linearisierungen ist. Allerdings kann dieses Resultat als Indikation für die Stabilität des nichtlinearen Systems verstanden werden. Unter Verwendung von (3.57) kann die nötige Abtastzeit unter Einbeziehung sämtlicher Linearisierungen zu

$$h \stackrel{!}{<} 12,9 \text{ ms}$$

bestimmt werden. Zur stabilen Integration des Modells ist mindestens eine kleinere Schrittweite zu wählen. Die tatsächlich erforderliche Schrittweite liegt möglicherweise unter der angegebenen, da es sich um ein nichtlineares System handelt; die Bedingung kann als Abschätzung einer oberen Grenze aufgefasst werden. Es sei angemerkt, dass die Zeitkonstanten der Aktuatoren bei $\tau_{\text{wg}} = 21 \text{ ms}$ und $\tau_{\text{wg}} = 78 \text{ ms}$ liegen und die Abschätzung der Schrittweite deswegen nicht durch die Aktuatoren dominiert wird. Die Zeitkonstanten der Aktuatoren korrespondieren zu Schrittweiten von 42 ms bzw. 156 ms.

Unter Verwendung der ermittelten Mindestschrittweite soll nun die Echtzeitfähigkeit des Luftpfades bewertet werden. Die MPC wird alle 50 ms auf dem Steuergerät ausgeführt. Daraus ergibt sich eine Abtastzeit T_s des Reglers von ebenjener Zeitspanne. Für diesen Zeitraum sind die seitens des Reglers vorgegebenen Eingänge des Luftpfades konstant. Wird die Quadratur für ebenjenen Zeitraum durchgeführt, entsteht kein zusätzlicher Fehler durch vernachlässigte Änderungen der Modelleingänge. Für die Simulation des Systems innerhalb des MPC-Algorithmus (vgl. Abschnitt 4.4.2) bietet sich diese Schrittweite ebenfalls an.

Es wird das *klassische Runge-Kutta-Verfahren* (2.11) betrachtet. Dieses Verfahren benötigt vier Auswertungen der DGLs, es ist jedoch genauer als das zur Abschätzung der Schrittweite verwendete explizite Euler-Verfahren. Des Weiteren ist das Stabilitätsgebiet größer (vgl. Abb. 2.2a und Abb. 2.2b). Dieser Aspekt ist von Bedeutung, da es sich bei der Mindestschrittweite lediglich um eine Abschätzung handelt. Zur Integration eines Zeitraumes von 50 ms werden fünf Integrationsschritte à 10 ms Schrittweite durchgeführt.³⁵ Unter Verwen-

³³Die Bedingung für die Schrittweite kann über $|1 + h\lambda_i| = (1 + \text{Re}\{h\lambda_i\})^2 + \text{Im}\{h\lambda_i\}^2$ hergeleitet werden.

³⁴Die Verwendung von Integrationsverfahren höherer Ordnung erlaubt die Verwendung größerer Schrittweiten (vgl. Abschnitt 2.1.3).

³⁵Eine Integration mit vier Integrationsschritten à 12,5 ms würde die Mindestschrittweite ebenfalls erfüllen. Aus Gründen der numerischen Robustheit werden jedoch 10 ms für die ersten Untersuchungen des

derung der angegebenen Ausführungszeit in Tab. 3.4 ergibt sich eine Rechenzeit von $5,446 \mu\text{s}$, unter Vernachlässigung der Rechenzeit zur Auswertung der Quadraturformel. Neben der Integration des Luftpfadmodells ist das KNN auszuwerten. Das KNN besteht insgesamt aus 25 Neuronen. Zur Auswertung eines jeden Neurons ist eine Exponentialfunktion zu evaluieren sowie eine bestimmte Anzahl an Multiplikationen und Additionen durchzuführen, deren konkrete Anzahl von der Architektur des KNN abhängt. Die Evaluierung der Neuronen summiert sich zu $67,5 \text{ ns}$. Da die Rechenzeit der Additionen und Multiplikationen nicht ermittelt wurde, wird die Rechenzeit des KNN zu $0,1 \mu\text{s}$ abgeschätzt. Damit sind insgesamt $5,546 \mu\text{s}$ Rechenzeit zur Propagation des Systems um 50 ms Echtzeit erforderlich, welche mehrere Größenordnungen unter ebenjener liegt.³⁶ Das Zielsystem (*Infineon Aurix TriCore*, 300 MHz) ist langsamer als der zur Berechnung der Ausführungszeiten (vgl. Tab. 3.4) verwendete PC (*Intel i5-8250U*, $3,4 \text{ GHz}$).³⁷ Um die Ausführungszeiten auf der ECU abzuschätzen, werden die Zeiten um den Faktor 12 vergrößert. Unter diesen Bedingungen ergibt sich ein RTF (1.4) von $752,958$. Die Auswertung des Luftpfades würde damit $0,132 \%$ der verfügbaren Rechenleistung beanspruchen. Zur Bewertung der Genauigkeit können die Simulationen in Abb. 3.5 sowie in Abb. 3.6 herangezogen werden; diese wurden mit denselben Parametern berechnet.

Regelungsalgorithmus gewählt und deshalb an dieser Stelle betrachtet.

³⁶Das KNN wird einmalig zur Integration von 50 ms berechnet.

³⁷Die technischen Daten des in der ECU verbauten Prozessors sind in Tab. 1.1 aufgeführt.

4 Modellprädiktiver Regelungsalgorithmus

Zur Regelung des temperaturgeführten Motors in Kapitel 5 wird ein nichtlinearer modellprädiktiver Regelungsalgorithmus eingesetzt. Inhalt dieses Kapitels ist die Entwicklung eines solchen Algorithmus. Unter den gegebenen rechentechnischen Voraussetzungen liegt der Fokus darauf, ein Verfahren zu entwerfen, welches auf der ECU umsetzbar ist, und zwar derart, dass die Möglichkeit für eine kommerzielle Verwendung des Verfahrens besteht. Dies bedeutet, dass die in Tab. 1.2 genannten Voraussetzungen einzuhalten sind. Die dort aufgeführten Rahmenbedingungen liegen hinsichtlich der Rechenzeit sowie der Beanspruchung mindestens eine Größenordnung unter den technischen Möglichkeiten der ECU (vgl. Tab. 1.1). Des Weiteren sind ein deterministisches Verhalten des Algorithmus sowie ein Lösungsansatz mit möglichst geringer Komplexität gefordert. Letztere Eigenschaft ist insbesondere für die Qualitätssicherung maßgeblich. Die wichtigste qualitätssichernde Eigenschaft ist ein robuster Algorithmus, der hohe numerische Stabilitätsreserven aufweist.

4.1 Modellprädiktive Regelung

Einer Regelung liegt typischerweise das Ziel zugrunde, ein Gütekriterium zu minimieren, unter gleichzeitiger Einhaltung bestimmter Zustands- und Eingangsbeschränkungen. Bei einem Folgeregler besteht das Gütekriterium beispielsweise darin, den Sollwert möglichst genau und schnell einzuregeln, ohne dabei zu stark überzuschwingen. Die Stellgröße – der Eingang der Regelstrecke – einer solchen Regelung ist in der Praxis i. d. R. begrenzt. Bezogen auf die in Kapitel 5 entwickelte temperaturgeführte Motorregelung können das WG und die ETV lediglich Positionen zwischen vollständig geöffnet und vollständig geschlossen annehmen. Neben den Eingängen können auch interne Größen der Regelstrecke Limitierungen unterworfen sein, deren Überschreitung unerwünscht oder sicherheitsrelevant ist. Insbesondere letztere lassen sich lediglich durch eine vorausschauende Regelungsstrategie einhalten. Wenn der Regler erst Maßnahmen ergreift, sobald eine solche Bedingung verletzt wird, ist eine weitere, stärkere Verletzung wahrscheinlich, da die Maßnahmen aufgrund der Trägheit des Systems nicht direkt eine Wirkung zeigen.

Diese Art der Problemstellung kann als DP (siehe Optimierungsproblem (2.37)) formuliert werden. Eine analytische Lösung ist über die Hamilton-Jacobi-Bellman Gleichung (2.48) zu erreichen, was allerdings selbst für den unbeschränkten nichtlinearen Fall lediglich in Sonderfällen gelingt [AFN04]. Die analytische Lösung auf numerischem Wege umgangen: Das DP wird stattdessen wiederholt zur Laufzeit auf dem Zielsystem – im Rahmen dieser Arbeit die ECU – ausgewertet. Dies führt zu der abgewandelten Formulierung (4.1). Was diese Formulierung von einem gewöhnlichen DP unterscheidet, sind die Integralgrenzen. Diese basieren auf dem aktuellen Zeitpunkt t_k . Das Funktionsprinzip ist in Abb. 4.1 grafisch dargestellt. Der grau hinterlegte Bereich entspricht dem Prädiktionshorizont der Länge T , dem Bereich, der von den Integralgrenzen abgedeckt wird. In der Grafik soll die Regelgrö-

ße $\mathbf{y}(t)$ auf den Sollwert $\mathbf{y}_s(t)$ unter Verwendung der Stellgröße $\mathbf{u}(t)$ eingeregelt werden. Die blau gestrichelten Linien sind die im Rahmen der Optimierung berechneten *optimalen* Trajektorien. Durch Modellungenauigkeiten oder durch ein – zur Einhaltung rechentechnischer Grenzen gegebenenfalls erforderliches – verfrühtes Abbrechen der Optimierung, der sogenannten *suboptimalen MPC*, unterscheiden sich diese Trajektorien von den tatsächlichen, in schwarz dargestellten Trajektorien. Nach Beendigung der Optimierung wird die berechnete Stellgröße $\mathbf{u}_{k|0}$ auf das System angewendet. Beim nächsten Aufruf des Regelalgorithmus, T_s Zeiteinheiten später, wird der Prädiktionshorizont weiterschieben und die Optimierung erneut ausgeführt. Die Berechnungen für die nachfolgenden Stellgrö-

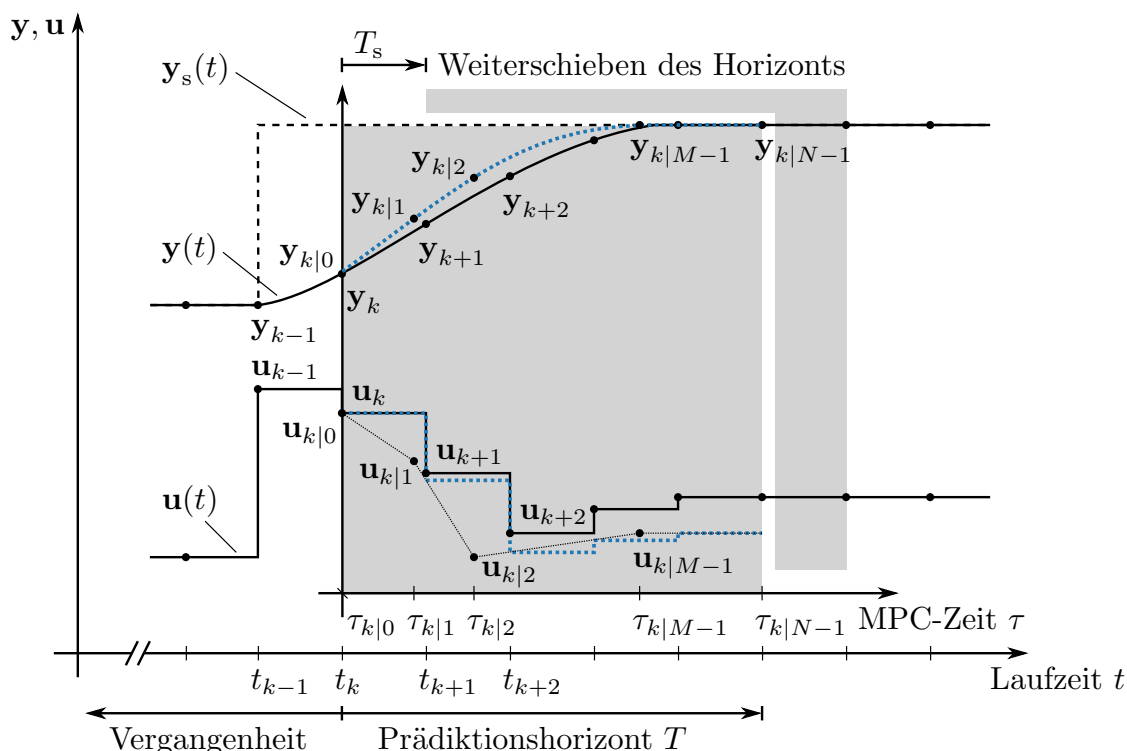


Abb. 4.1: Prinzip des bewegten Horizonts.

ßen werden rein formal nicht benötigt, jedoch werden diese in einer Implementierung zur Initialisierung der nächsten Optimierung verwendet, um die Startschätzung zu verbessern und damit der Lösung näher zu sein als bei einer beliebigen aber festen Startschätzung. Die fortlaufende Lösung eines Optimierungsproblems hat einen nicht unerheblichen numerischen Aufwand zur Folge.

Die Berechnung des Reglergesetzes zur Laufzeit durchzuführen ist der Hauptunterschied zu konventionellen Reglern mit festem (rückführungsbasiertem) Reglergesetz. In einer systemischen Betrachtung der MPC sind die Sollwerte die Eingänge in das System und die Stellgrößen die Ausgänge. Gemäß dieser Betrachtung kann die MPC als *Vorsteuerung* eingestuft werden. Der Regelkreis kann jedoch durch einen Beobachter geschlossen werden (vgl. Abb. 4.2), um ein Auseinanderdriften der MPC und der Regelstrecke zu verhindern. Dieser führt die aktuellen Systemzustände als zusätzlichen Eingang in die MPC zurück,

welche als Startbedingungen des zu lösenden DP dienen; es wird in Gleichung (4.1b) der geschätzte Zustand $\mathbf{x}(t_k) = \hat{\mathbf{x}}_k$ verwendet.

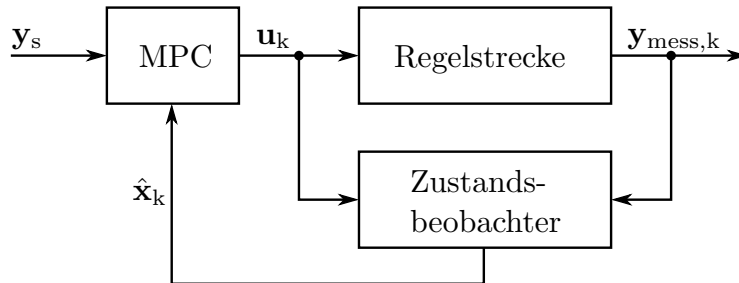


Abb. 4.2: MPC-Reglerstruktur mit Zustandsrückführung.

Unterschieden wird zwischen *linearer* und *nichtlinearer* MPC. Bei der linearen MPC sind das zugrundeliegende System – zumindest dessen mathematische Beschreibung –, die Nebenbedingungen sowie das Gütekriterium linear. Bei der nichtlinearen MPC ist zumindest eines von den dreien nichtlinear. Numerisch resultiert eine lineare MPC in der Lösung eines *linearen* oder *quadratischen Optimierungsproblems* (QP), für welche effiziente Lösungsverfahren existieren.¹ Im Falle der nichtlinearen MPC sind aufwendigere numerische Lösungsverfahren erforderlich. Die gängigen Lösungsverfahren sind gradientenbasiert (vgl. Abschnitt 2.2.1) und konvergieren nicht zwangsläufig zur globalen (besten) Lösung. Es ist möglich, dass der Algorithmus lediglich eine lokale Lösung findet (vgl. Abb. 2.4). Aus theoretischer Perspektive sind die Stabilitätseigenschaften der linearen MPC besser erforscht und leichter nachzuweisen als bei der nichtlinearen. Einen Überblick zur Stabilität der linearen und nichtlinearen MPC sowie verschiedener Formulierungen liefert [MRRS00].

Dem genannten Nachteil des erhöhten Rechenaufwands im Vergleich mit festen Reglergesetzen stehen mehrere Vorteile gegenüber. Der Hauptvorteil ist die inhärente Möglichkeit, Nebenbedingungen berücksichtigen zu können. Eingangsnebenbedingungen können mit festen Reglergesetzen bei Systemen mit einer Stellgröße über Saturierungen in Kombination mit *Anti-Windup-Mechanismen* für Regler mit integralem Anteil praktikabel gehandhabt werden [Ada14, S. 223ff]. Für Systeme mit mehreren Eingängen und einem integrierenden Regler können Kompensationsregelungen verwendet werden, um dies zu gewährleisten, jedoch sind diese relativ aufwendig zu bestimmen [KCMN93], [SP07, S. 484ff]. Die Berücksichtigung von Zustands- bzw. Ausgangsbeschränkungen ist ebenfalls über einen Kompensationsregler denkbar. Ein weiterer Vorteil ist die Möglichkeit, die Gütefunktion während des Betriebs anpassen zu können, was bei einem Zielkonflikt verschiedener Regelungsziele zusätzliche Eingriffsmöglichkeiten offen hält (vgl. Gleichung (5.8a)). Neben der Adaption der Gütefunktion ist eine Adaption des Streckenmodells ebenfalls möglich, um gegebenenfalls einem sich ändernden Systemverhalten Rechnung zu tragen. Zuletzt bietet die MPC den Vorteil, für den Anwender leicht verständliche Eingriffsmöglichkeiten bereit

¹Konvexe Optimierungsprobleme werden in [BV09] besprochen. Von diesen sind die *linearen* und *quadratischen Optimierungsprobleme* umfasst.

zu halten. Dies sind beispielsweise ein Grenzwert für eine bestimmte Größe oder ein Gewichtungsfaktor in der Gütefunktion. Eine allgemeine Einführung in die MPC wird in den Übersichtsartikeln [Raw00], [MH99], [AFN04] gegeben.

4.2 Motivation des Gradientenverfahrens

Zur Lösung des einer nichtlinearen MPC unterlagerten DP können grundsätzlich die in Abschnitt 2.3.2 vorgestellten Verfahren verwendet werden. Für die Implementierung auf der ECU mit ihren begrenzten rechentechnischen Kapazitäten (vgl. Tab. 1.1) sind die Verfahren jedoch unterschiedlich gut geeignet. Im Folgenden sollen die Verfahren, bis auf die Lösung mit der HJB-Gleichung (2.48), bezüglich ihrer Eignung bewertet werden.² Eine Zusammenfassung der Bewertung ist in Tab. 4.1 zu finden.

Das indirekte Schießverfahren (IS) zeichnet sich durch seine hohe Genauigkeit aus, die im Wesentlichen durch die Qualität der numerischen Integration begrenzt wird. Des Weiteren kommt es von allen betrachteten Verfahren mit der geringsten Anzahl an Optimierungsvariablen aus. Es ist lediglich der Initialwert der adjungierten Variablen λ_0 zu ermitteln (vgl. Algorithmus 2.3.1). Demgegenüber steht die hohe Empfindlichkeit gegenüber der Startschätzung für λ_0 (vgl. Abschnitt 2.3.2). Der Implementierungsaufwand ist von den betrachteten Verfahren am geringsten. Es besteht aus einer Vorwärtsintegration und einem Iterationsschritt.

Beim indirekten Mehrschießverfahren (IMS) wird der Aspekt der Empfindlichkeit gegenüber der Startschätzung durch die Einführung mehrerer Intervalle entschärft, wodurch die Robustheit dieses Verfahrens höher ist. Allerdings steigen dadurch die Anzahl der zu optimierenden Variablen sowie der Implementierungsaufwand. Wenn ein Lösungsalgorithmus verwendet wird, der den Gebrauch von der Hesse-Matrix (HM) macht, wäre zusätzlich deren dünnbesetzte Gestalt auszunutzen. Ein weiterer Nachteil besteht darin, dass die Lösung erst valide ist, wenn die zu optimierenden Intervallgrenzen einen stetigen Gesamtverlauf ergeben.

Das Gradientenverfahren (GV) bzw. das direkte Schießverfahren (DS) zeichnen sich durch eine höhere Robustheit als die bereits erwähnten Verfahren aus.³ Diese kommt primär dadurch zustande, dass durch die Rückwärtsintegration der adjungierten Zustände keine Startschätzung ebenjener erforderlich ist und dass die Rückwärtsintegration bessere Stabilitätseigenschaften hat (vgl. Abschnitt 2.3.2). Im Gegenzug ist die Genauigkeit aufgrund der Diskretisierung der Steuertrajektorie und des dadurch bedingten Integrationsfehlers geringer. Der Implementierungsaufwand ist wegen der Vorwärts- und Rückwärtsintegra-

²Die HJB-Gleichung kann wegen des in Abschnitt 2.3.2 erläuterten *Fluches der Dimensionalität* als möglicher Lösungsansatz im Vorhinein ausgeschlossen werden.

³Das GV und das DS werden aufgrund ihrer umsetzungstechnischen Ähnlichkeit gemeinsam betrachtet. In Abschnitt 2.3.2 wird erläutert, dass sich das DS identisch zum GV umsetzen lässt.

tion sowie der durch die Diskretisierung erforderlichen Interpolationsroutinen als höher einzuschätzen.

Analog zum IMS wird die Robustheit beim direkten Mehrfachschießverfahren (DMS) gegenüber dem DS durch die Aufteilung des Horizonts in Intervalle erhöht, mit denselben Nachteilen wie beim IMS.

Das direkte Kollokationsverfahren (DK) hat prinzipiell ähnliche Eigenschaften wie das DMS. Jedoch sind aufgrund der Tatsache, dass die im Optimierungsproblem enthaltenen DGLs i. d. R. über Polynome approximiert werden, mehr Variablen zu optimieren. Des Weiteren ist die Genauigkeit der Approximation über ein Polynom geringer als die eines dedizierten DGL-Lösers. Andererseits wird durch diesen Ansatz die Robustheit erhöht, was das Verfahren besonders attraktiv für instabile Systeme macht [DBDW05].

Tab. 4.1: Verfahren der dynamischen Optimierung im qualitativen Vergleich.

| Kriterium | GV / DS | IS | IMS | DMS | DK |
|----------------------------------|---------|-------------|------------|-----------------|----------------|
| Genauigkeit | hoch | sehr hoch | sehr hoch | hoch | mittel |
| Robustheit | mittel | sehr gering | gering | hoch | sehr hoch |
| Variablenanz.¹ | Nm | n | $2Nn - 3n$ | $(N - 2)n + Nm$ | $> \text{DMS}$ |
| Dünnbes. HM | nein | nein | ja | ja | ja |
| Impl.-Aufwand | gering | sehr gering | mittel | hoch | sehr hoch |
| Vorz. Abbruch | ja | ja | nein | nein | ja |

¹ Anzahl der zu optimierenden Variablen für ein Optimierungsproblem ohne Nebenbedingungen. n : Anzahl der Zustände, m : Anzahl der Eingänge / Stellgrößen, N : Anzahl der Diskretisierungspunkte.

In Tab. 4.1 sind die Eigenschaften der Verfahren qualitativ gegenübergestellt. Es sei angemerkt, dass es sich um eine allgemeine Bewertung handelt. Im Detail kann diese abhängig von der Problemstellung sowie der konkreten Umsetzung anders ausfallen. Die obersten Ziele für die Umsetzung auf der ECU sind die Einhaltung der in Tab. 1.2 aufgeführten Anforderungen bezüglich der CPU-Auslastung und des beanspruchten RAM. Letzterer wird umso mehr beansprucht, je größer die Anzahl an Optimierungsvariablen ist. Auf dieser Basis sollen das IMS, das DMS sowie das DK als Lösungsstrategien ausgeschlossen werden. Das IMS sowie das DMS werden oftmals mit einem *SQP-Verfahren* gelöst, für das DK finden auch *Innere-Punkte-Verfahren* Anwendung [DFH09].⁴ Bei beiden Verfahren wird unterlagert ein Newton-Verfahren (vgl. Abschnitt A.1.2) angewendet, welches die Hesse-Matrix des Optimierungsproblems erfordert. Ohne Nebenbedingungen entspricht deren Dimension der Variablenanzahl der Problemstellung. Auf den in Kapitel 3 vorgestellten Luftpfad mit $n = 8$ Zuständen und $m = 2$ zu regelnden Eingängen übertragen und unter

⁴Das SQP- und das Innere-Punkte-Verfahren werden in [NW06, S. 539ff u. S. 563ff] beschrieben.

der Annahme, dass $N = 7$ Diskretisierungen verwendet werden – dies ist die später verwendete Einstellung für den Regler des temperaturgeführten Motors –, ergeben sich die in Tab. 4.2 dargestellten Speicherverbräuche. Es sei darauf hingewiesen, dass lediglich die Speicherverbräuche der Optimierungsvariablen aufgeführt sind, wodurch der tatsächlich beanspruchte RAM jedes der aufgeführten Verfahren unterschätzt wird. Das DMS kommt von den drei auszuschließenden Verfahren mit dem geringsten Speicherverbrauch aus, doch selbst in diesem Falle werden ca. 10 % des verfügbaren RAM zur Speicherung der HM benötigt. Für den Stack, der per Design-Richtlinie auf 2 kB begrenzt ist, ist die HM zu groß, und für den Heap ist diese Größe im Kontext der dauerhaften Belegung nicht akzeptabel. Durch das Ausnutzen der *dünnbesetzten* Struktur der Matrizen ließe sich der Speicherverbrauch reduzieren, jedoch würde eine rechentechnische Umsetzung einen zusätzlichen Implementierungsaufwand bedeuten. Des Weiteren ist die statische Reservierung von Speicher für dünnbesetzte Matrizen nicht trivial; es müsste abhängig vom zu regelnden System die Struktur der HM sowie die Struktur der darauf basierenden Berechnungen analysiert werden.

Tab. 4.2: Variablenanzahl und die Anzahl der Elemente in der Hesse-Matrix bezogen auf das Luftpfadmodell aus Kapitel 3 mit $N = 7$ Diskretisierungspunkten. Es werden keine Nebenbedingungen berücksichtigt.

| Kriterium | GV / DS | IS | IMS | DMS | DK |
|----------------------------|---------|-------|--------|--------|----------|
| Variablenanzahl | 14 | 8 | 88 | 54 | > 54 |
| RAM [kB] ¹ | 0,056 | 0,032 | 0,352 | 0,216 | > 0,216 |
| Anz. Elemente in HM | 196 | 64 | 7 744 | 2 916 | > 2 916 |
| RAM [kB] ¹ | 0,784 | 0,256 | 30,976 | 11,664 | > 11,664 |

¹ Die Speicherverbräuche ergeben sich unter Verwendung von Fließkommazahlen mit einfacher Genauigkeit nach dem IEEE-754 Format. Pro zu speichernder Zahl werden 4 B RAM benötigt.

Mit dieser Argumentation bleiben als Lösungskandidaten das GV / DS und das IS übrig. Wie eingangs erwähnt, ist das IS von sämtlichen betrachteten Verfahren das am wenigsten robuste (vgl. Tab. 4.1). Für den Einsatz in einem Gerät, welches im schlimmsten Fall durch eine fehlerhafte Regelung beschädigt werden kann, ist allerdings ein zuverlässiges Verfahren gefordert (vgl. Tab. 1.2). Aus diesem Grund fällt die Wahl auf das GV / DS.

4.3 Lösungsansatz

Konzeptionell greift der in diesem Kapitel vorgestellte Lösungsalgorithmus auf die Ansätze des Softwareframeworks GRAMPC [EVM⁺19] zurück, welches vor dem Hintergrund eines

Einsatzes auf einem eingebetteten System entworfen wurde.⁵ Der Algorithmus kombiniert das zuvor motivierte GV (vgl. Abschnitt 2.3.2) zur Lösung des zugrundeliegenden dynamischen Optimierungsproblems mit einer Gradientenprojektion (vgl. Abschnitt 2.2.3) sowie der erweiterten Lagrange-Methode (vgl. Abschnitt 2.2.4). Die Gradientenprojektion und die erweiterte Lagrange-Methode werden zur Handhabung von Nebenbedingungen eingesetzt, das GV löst das eigentliche dynamische Optimierungsproblem. Es werden alle drei genannten Verfahren in abgewandelter Form umgesetzt, um die Robustheit zu steigern und den erforderlichen RAM zu reduzieren. Die Gradientenprojektion wird dahingehend erweitert, dass Änderungsraten der Aktuatoren berücksichtigt werden können. Diese können beim Einsatz der Regelung zur Verbesserung der Robustheit herangezogen werden. Je genauer der Lösungsraum eingegrenzt wird – die Änderungsratenbeschränkung ist eine solche Eingrenzung – desto *einfacher* ist es, das Problem zu lösen [AEH14]. Des Weiteren kann über diese Begrenzung eine stärkere Kontrolle auf die Geschwindigkeit des Reglers ausgeübt werden. Ebenfalls zur Steigerung der Robustheit wird eine iterative Liniensuche eingeführt (vgl. Abschnitt 2.2.2), welche die über eine Heuristik initial abgeschätzte Schrittweite verbessert. Die erweiterte Lagrange-Methode wird dahingehend verändert, dass weniger zu speichernde Variablen erforderlich sind und somit weniger Speicherplatz im RAM zu allokiert ist.⁶ Die Berechnung der Gradienten beider Methoden wird hinsichtlich einer nicht-äquidistanten Diskretisierung des Optimierungsproblems angepasst. Durch eine nicht-äquidistante Diskretisierung ist es möglich, gezielt an den Stellen, wo eine geringe Aktivität des Reglers erwartet wird, weniger Stützstellen platzieren zu können und auf diese Weise die Anzahl der zu speichernden Variablen zu reduzieren. Eine weitere Verringerung des Speicherplatzes wird durch die Einführung eines separat wählbaren *Kontrollhorizonts* erzielt.

Der präsentierte Regelungsalgorithmus löst im Grundsatz das in Abschnitt 2.3 besprochene DP (2.37). Die einzigen Änderungen bestehen darin, dass statt der Startzeit t_0 der aktuelle Abtastschritt t_k verwendet wird, dass an Stelle von \mathbf{x}_0 von \mathbf{x}_k ausgehend integriert wird und dass zusätzlich eine Änderungsratenbeschränkung berücksichtigt werden kann. Die Eingangs- (4.1c) und die Änderungsratenbeschränkung (4.1d) sind in der Umsetzung

⁵Frühe Versionen von GRAMPC basierten auf den Inhalten von [GK12] und [KG13].

⁶Durch diese Änderung liegt die Umsetzung näher an der Definition des Lagrange-Ansatzes für statische Optimierungsprobleme (vgl. Abschnitt 2.2.4) als GRAMPC.

zeitunabhängig. Der Algorithmus löst damit ein Optimierungsproblem der Form

$$\min_{\mathbf{u}(\cdot)} \quad J(\mathbf{u}(\cdot); \mathbf{x}_k, t_k) = V(\mathbf{x}(t_k + T), t_k + T) + \int_{t_k}^{t_k + T} l(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) d\tau \quad (4.1a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)), \quad \mathbf{x}(t_k) = \mathbf{x}_k, \quad (4.1b)$$

$$\mathbf{b}_l \leq \mathbf{u}(\tau) \leq \mathbf{b}_u, \quad (4.1c)$$

$$\mathbf{r}_l \leq \frac{d\mathbf{u}(\tau)}{dt} \leq \mathbf{r}_u, \quad (4.1d)$$

$$\mathbf{c}_\mathcal{E}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) = 0, \quad (4.1e)$$

$$\mathbf{c}_\mathcal{I}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) \leq 0, \quad (4.1f)$$

$$\mathbf{c}_{\mathcal{T}, \mathcal{E}}(\mathbf{x}(t_k + T), t_k + T) = 0, \quad (4.1g)$$

$$\mathbf{c}_{\mathcal{T}, \mathcal{I}}(\mathbf{x}(t_k + T), t_k + T) \leq 0. \quad (4.1h)$$

Die Definitionen in (2.37) werden um die hinzugekommene Änderungsratenbeschränkung mit $\mathbf{b}_l, \mathbf{b}_u \in \{\mathbb{R}^m \mid \mathbf{b}_l \leq \mathbf{b}_u\}$ erweitert. Als Integrationsvariable wird die MPC-Zeit τ verwendet. Auf diese Weise tritt der Unterschied zur Laufzeit t deutlicher hervor. Diese Definition des Optimierungsproblems umfasst die in der Literatur gebräuchlichen MPC-Formulierungen (vgl. [MRRS00], [AFN04], [GP17]), insbesondere: *Endlicher Horizont mit Endbeschränkung* (durch Nebenbedingung (4.1g)), *Endlicher Horizont mit Endregion* (durch Nebenbedingung (4.1h)), *Endlicher Horizont mit Endkostengewichtung* (unter Verwendung von $V(\cdot)$) und *Endlicher Horizont ohne Endbedingung*.

4.3.1 Erweiterte Lagrange-Formulierung

Analog zur erweiterten Lagrange-Funktion (2.35) werden die Nebenbedingungen (4.1e - 4.1h) mit in die Gütefunktion aufgenommen. Die Nebenbedingungen (4.1e) und (4.1f), welche während der Integration aktiv sind, werden den Integralkosten $l(\cdot)$ zugeordnet, die anderen beiden Nebenbedingungen den Endkosten $V(\cdot)$. Es ergibt sich die erweiterte Gütefunktion

$$\begin{aligned} \tilde{J}(\mathbf{u}(\cdot), \mathbf{x}(\cdot), \tilde{\boldsymbol{\mu}}(\cdot), t_k; \tilde{\nu}(\cdot)) &= \tilde{V}(\mathbf{x}(t_k + T), \boldsymbol{\mu}_T, t_k + T; \nu_T) \\ &+ \int_{t_k}^{t_k + T} \tilde{l}(\mathbf{x}(\tau), \mathbf{u}(\tau), \boldsymbol{\mu}(\tau), \tau; \nu(\tau)) d\tau \end{aligned} \quad (4.2)$$

mit $\tilde{\boldsymbol{\mu}}(\tau) = [\boldsymbol{\mu}(\tau)^T, \boldsymbol{\mu}_T^T]^T$ und $\tilde{\nu}(\tau) = [\tilde{\nu}(\tau), \tilde{\nu}_T]^T \in \mathbb{R}^2$ unter Verwendung von

$$\begin{aligned} \tilde{V}(\mathbf{x}, \boldsymbol{\mu}_T, \tau; \nu_T) &= V(\mathbf{x}, \tau) \\ &+ \boldsymbol{\mu}_{T, \mathcal{E}} \mathbf{c}_{T, \mathcal{E}}(\mathbf{x}, \tau) + \frac{\nu_T}{2} \|\mathbf{c}_{T, \mathcal{E}}(\mathbf{x}, \tau)\|^2 \\ &+ \boldsymbol{\mu}_{T, \mathcal{I}} \mathbf{c}_{T, \mathcal{I}}^+(\mathbf{x}, \boldsymbol{\mu}_{T, \mathcal{I}}, \tau; \nu_T) + \frac{\nu_T}{2} \|\mathbf{c}_{T, \mathcal{I}}^+(\mathbf{x}, \boldsymbol{\mu}_{T, \mathcal{I}}, \tau; \nu_T)\|^2 \\ &= V(\mathbf{x}, \tau) + \boldsymbol{\mu}_T \tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\mu}_T, \tau; \nu_T) + \frac{\nu_T}{2} \|\tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\mu}_T, \tau; \nu_T)\|^2 \end{aligned} \quad (4.3)$$

mit $\boldsymbol{\mu}_T = [\boldsymbol{\mu}_{T,\mathcal{E}}^T, \boldsymbol{\mu}_{T,\mathcal{I}}^T]^T$, $\tilde{\mathbf{c}}_T = [\mathbf{c}_{T,\mathcal{E}}^T, \mathbf{c}_{T,\mathcal{I}}^{+T}]^T$, $\boldsymbol{\mu}_{T,\mathcal{E}}, c_{T,\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}_T|}$, $\boldsymbol{\mu}_{T,\mathcal{I}}, c_{T,\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}_T|}$, sowie

$$\begin{aligned} \tilde{l}(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}, \tau; \nu) &= l(\mathbf{x}, \mathbf{u}, \tau) \\ &+ \boldsymbol{\mu}_{\mathcal{E}} \mathbf{c}_{\mathcal{E}}(\mathbf{x}, \mathbf{u}, \tau) + \frac{\nu}{2} \|\mathbf{c}_{\mathcal{E}}(\mathbf{x}, \mathbf{u}, \tau)\|^2 \\ &+ \boldsymbol{\mu}_{\mathcal{I}} \mathbf{c}_{\mathcal{I}}^+(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}_{\mathcal{I}}, \tau; \nu) + \frac{\nu}{2} \|\mathbf{c}_{\mathcal{I}}^+(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}_{\mathcal{I}}, \tau; \nu)\|^2 \\ &= l(\mathbf{x}, \mathbf{u}, \tau) + \boldsymbol{\mu}_T \tilde{\mathbf{c}}(\mathbf{x}, \boldsymbol{\mu}_T, \tau; \nu_T) + \frac{\nu_T}{2} \|\tilde{\mathbf{c}}_T(\mathbf{x}, \boldsymbol{\mu}_T, \tau; \nu_T)\|^2 \end{aligned} \quad (4.4)$$

mit $\boldsymbol{\mu} = [\boldsymbol{\mu}_{\mathcal{E}}^T, \boldsymbol{\mu}_{\mathcal{I}}^T]^T$, $\tilde{\mathbf{c}}_T = [\mathbf{c}_{\mathcal{E}}^T, \mathbf{c}_{\mathcal{I}}^{+T}]^T$, $\boldsymbol{\mu}_{\mathcal{E}}, c_{\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}$, $\boldsymbol{\mu}_{\mathcal{I}}, c_{\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}|}$. Wie in Abschnitt 2.2.4 näher ausgeführt, sind die Lagrange-Multiplikatoren der Ungleichungsbeschränkungen auf \mathbb{R} und nicht lediglich auf \mathbb{R}_+ definiert, da sie im erweiterten Lagrange-Ansatz formal zu Gleichungsbeschränkungen korrespondieren. Damit lässt sich der erweiterte Lagrange-Ansatz zur Lösung von (4.1) formulieren:

$$\max_{\tilde{\boldsymbol{\mu}}(\cdot)} \min_{\mathbf{u}(\cdot)} \quad \tilde{J}(\mathbf{u}(\cdot), \mathbf{x}(\cdot), \tilde{\boldsymbol{\mu}}(\cdot); \tilde{\nu}(\cdot)) \quad (4.5a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)), \quad \mathbf{x}(t_k) = \mathbf{x}_k, \quad (4.5b)$$

$$\mathbf{b}_l(\tau) \leq \mathbf{u}(\tau) \leq \mathbf{b}_u(\tau), \quad \forall \tau \in [t_k, t_k + T], \quad (4.5c)$$

$$\mathbf{r}_l(\tau) \leq \frac{d\mathbf{u}(\tau)}{d\tau} \leq \mathbf{r}_u(\tau), \quad \forall \tau \in [t_k, t_k + T]. \quad (4.5d)$$

Die (innere) Minimierung löst das eigentliche dynamische Optimierungsproblem zur Bestimmung der Steuertrajektorie $\mathbf{u}(\tau)$. Die Lagrange-Multiplikatoren $\tilde{\boldsymbol{\mu}}(\tau)$ gehen aus der (äußeren) Maximierung hervor, welche in einer algorithmischen Umsetzung im Zusammenspiel mit der Wahl der Strafparameter $\tilde{\nu}(\tau)$ erfolgt.

Zur Berechnung der kanonischen DGLs (4.16) und zur Berechnung der Abstiegsrichtung (4.11) wird die Hamilton-Funktion dieser Problemformulierung benötigt. Durch analoges Vorgehen zu Abschnitt 2.3.1 kann diese durch die Definition 4.3.1 angegeben werden.

Definition 4.3.1 (Erweiterte Hamilton-Funktion). Die Hamilton-Funktion des beschränkten Optimierungsproblems (4.5) ist über

$$\tilde{H}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, \boldsymbol{\mu}, t; \nu) := \tilde{l}(\mathbf{x}, \mathbf{u}, \boldsymbol{\mu}, t; \nu) + \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (4.6)$$

unter Verwendung der Integralkosten (4.4), des Modells (4.5b), der adjungierten Variablen $\boldsymbol{\lambda} \in \mathbb{R}^n$ und der Lagrange-Multiplikatoren $\boldsymbol{\mu} := [\boldsymbol{\mu}_{\mathcal{E}}^T, \boldsymbol{\mu}_{\mathcal{I}}^T]^T$, $\boldsymbol{\mu}_{\mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}$, $\boldsymbol{\mu}_{\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}|}$ definiert.

4.3.2 Diskretisierung und Interpolation

Aufgrund der zeitkontinuierlichen Trajektorien handelt es sich um ein unendlich-dimensionales Optimierungsproblem. Durch eine Zeitdiskretisierung auf dem Intervall $[t_k, t_k + T]$ durch

$$\underbrace{\tau_{k|0}}_{:=t_k} < \tau_{k|1} < \dots < \underbrace{\tau_{k|M-1}}_{:=t_k+T_C} < \dots < \underbrace{\tau_{k|N-1}}_{:=t_k+T}$$

wird dieses in ein endlich-dimensionales umgewandelt. Die Diskretisierungspunkte werden durch

$$\bar{\tau}_k = [\tau_{k|0}, \tau_{k|1}, \dots, \tau_{k|N-1}]^T := t_k + \bar{\tau}$$

zu einem Vektor zusammengefasst. Die Steuergrößen werden für einen u. U. kleineren Zeitraum $t_k + T_C$ diskretisiert, was die Wahl eines kleineren Kontrollhorizonts T_C gegenüber dem Prädiktionshorizont T ermöglicht; die Wahl von $M = N$ bzw. $T_C = T$ ist dadurch nicht ausgeschlossen. Denkbar wäre es, für die Steuergrößen ein eigenständiges Zeitraster zu wählen. Davon wird jedoch abgesehen, um die gleichen Zwischenvariablen verwenden zu können. Sowohl die adjungierte DGL (2.55b) als auch die Suchrichtung (4.11) basieren auf einer Ableitung der Hamilton-Funktion, für deren Auswertung die zuvor berechneten Zustände benötigt werden. Durch die Diskretisierung wird die Steuertrajektorie über

$$\mathbf{u}(\tau) = \boldsymbol{\psi}_s(\tau, \bar{\mathbf{u}}_k; T_s), \quad \forall \tau \in [t_k, t_k + T], \quad \text{mit} \quad \bar{\mathbf{u}}_k = [\mathbf{u}_{k|0}^T, \mathbf{u}_{k|1}^T, \dots, \mathbf{u}_{k|M-1}^T]^T$$

parametriert und die Zustandstrajektorie, die Trajektorie der adjungierten Zustände, die Lagrange-Multiplikator-Trajektorie sowie die Strafterm-Trajektorie über

$$\begin{aligned} \mathbf{x}(\tau) &= \boldsymbol{\psi}(\tau, \bar{\mathbf{x}}_k), & \forall \tau \in [t_k, t_k + T], & \quad \text{mit} \quad \bar{\mathbf{x}}_k = [\mathbf{x}_{k|0}^T, \mathbf{x}_{k|1}^T, \dots, \mathbf{x}_{k|N-1}^T]^T, \\ \boldsymbol{\lambda}(\tau) &= \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\lambda}}_k), & \forall \tau \in [t_k, t_k + T], & \quad \text{mit} \quad \bar{\boldsymbol{\lambda}}_k = [\boldsymbol{\lambda}_{k|0}^T, \boldsymbol{\lambda}_{k|1}^T, \dots, \boldsymbol{\lambda}_{k|M-1}^T]^T, \\ \boldsymbol{\mu}(\tau) &= \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\mu}}_k), & \forall \tau \in [t_k, t_k + T], & \quad \text{mit} \quad \bar{\boldsymbol{\mu}}_k = [\boldsymbol{\mu}_{k|0}^T, \boldsymbol{\mu}_{k|1}^T, \dots, \boldsymbol{\mu}_{k|N-1}^T]^T, \\ \nu(\tau) &= \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\nu}}_k), & \forall \tau \in [t_k, t_k + T], & \quad \text{mit} \quad \bar{\boldsymbol{\nu}}_k = [\nu_{k|0}^T, \nu_{k|1}^T, \dots, \nu_{k|N-1}^T]^T. \end{aligned}$$

Die Funktion $\boldsymbol{\psi}(\cdot)$ stellt eine lineare Interpolation dar, die Funktion $\boldsymbol{\psi}_s(\cdot)$ eine stufenweise lineare Interpolation. Letztere wird für die Steuertrajektorie gewählt, um den Aufruf des Reglers mit der Abtastzeit T_s abzubilden sowie die dadurch bedingte Änderung der Stellgrößen zu diskreten Zeitpunkten. Für Argumente τ außerhalb des von den Koeffizienten abgedeckten Bereichs wird der nächste Randwert verwendet. In Abb. 4.1 ist das qualitative Verhalten der stufenweisen linearen Interpolation dargestellt: Die schwarz gestrichelte Kurve stellt die lineare Interpolation der Stellgrößen dar, die zugehörige blaue Kurve ist die stufenweise lineare Interpolation durch eine Abtastung der linearen Interpolation mit T_s . Für $\tau > t_k + T_C$ wird $\mathbf{u}(\tau)$ konstant gehalten.

Durch die Zeitdiskretisierung ändert sich das zeitkontinuierliche Optimierungsproblem (4.5) zu

$$\max_{\bar{\boldsymbol{\mu}}_k} \min_{\bar{\mathbf{u}}_k} \tilde{J}(\bar{\mathbf{u}}_k, \bar{\mathbf{x}}_k, \bar{\boldsymbol{\mu}}_k, \boldsymbol{\mu}_{T,k}, t_k; \bar{\boldsymbol{\nu}}_k, \boldsymbol{\nu}_{T,k}) \quad (4.8a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}(\tau), \boldsymbol{\psi}_s(\tau, \bar{\mathbf{u}}_k)), \quad \mathbf{x}(t_k) = \mathbf{x}_k, \quad (4.8b)$$

$$- \mathbf{u}_{k|0} + \max \left\{ \mathbf{b}_l, \mathbf{u}_{k-1|0} + T_s \mathbf{r}_l \right\} \leq \mathbf{0}, \quad (4.8c)$$

$$- \mathbf{u}_{k|i} + \max \left\{ \mathbf{b}_l, \mathbf{u}_{k|i-1} + (\tau_{k|i} - \tau_{k|i-1}) \mathbf{r}_l \right\} \leq \mathbf{0}, \quad i = 1, 2, \dots, M-1, \quad (4.8d)$$

$$\mathbf{u}_{k|0} - \min \left\{ \mathbf{b}_u, \mathbf{u}_{k-1|0} + T_s \mathbf{r}_u \right\} \leq \mathbf{0}, \quad (4.8e)$$

$$\mathbf{u}_{k|i} - \min \left\{ \mathbf{b}_u, \mathbf{u}_{k|i-1} + (\tau_{k|i} - \tau_{k|i-1}) \mathbf{r}_u \right\} \leq \mathbf{0}, \quad i = 1, 2, \dots, M-1. \quad (4.8f)$$

Die Nebenbedingungen (4.8c) und (4.8d) enthalten die unteren Aktuatorbeschränkungen aus (4.5c) und (4.5d), die Nebenbedingungen (4.8e) und (4.8f) entsprechend die oberen. Zur Umwandlung von (4.5d) wurde die lineare Interpolation der Stellgrößen zwischen den Diskretisierungspunkten ausgenutzt. An einem Diskretisierungspunkt i darf die Stellgröße lediglich so groß bzw. klein gewählt werden, dass die Steigung der Interpolation – die Gerade, die beide Punkte verbindet – die Grenzwerte nicht verletzt. Für den ersten Diskretisierungspunkt wird jeweils auf die im vorigen Zeitschritt angewendeten Stellgrößen $\mathbf{u}_{k-1|0}$ zurückgegriffen.

4.4 Erweitertes Gradientenverfahren

Zur Lösung des Optimierungsproblems (4.8) wird ein *erweitertes Gradientenverfahren* eingesetzt. Die Bezeichnung des Verfahrens stammt nicht aus der Literatur. Sie wird an dieser Stelle eingeführt, um die Kombination des GV mit der erweiterten Lagrange-Methode kenntlich zu machen und auf diese Weise die Bezeichnung des Ansatzes als Ganzes von dem GV abzuheben.

Der Algorithmus ist im Grundsatz analog zur erweiterten Lagrange-Methode (vgl. Algorithmus 2.2.5) aufgebaut, mit dem Unterschied, dass die Maximierung in die Minimierung eingebettet und nicht über zwei ineinander geschachtelte Iterationsschleifen gelöst wird. Bei der in Algorithmus 4.4.1 dargestellten erweiterten Lagrange-Methode wird lediglich eine Iterationsschleife verwendet. Die Aktualisierungsfunktionen der Lagrange-Multiplikatoren und der Strafparameter enthalten jeweils eine Konvergenzbedingung, über die deren Aktualisierung gesteuert wird. Der Algorithmus insgesamt vollführt G Iterationen, ohne eine gesonderte Abbruchbedingung. Auf die gesonderte Abbruchbedingung wird verzichtet, um eine deterministische Ausführungszeit zu erhalten (vgl. Tab. 1.2).

Die Ausführungsschritte bis einschließlich der Liniensuche sind der Minimierung und damit

Algorithmus 4.4.1 (MPC Lösungsalgorithmus).

Initialisierung

- Zeitverschiebung und Initialisierung (siehe Abschnitt 4.4.6)

$$\begin{aligned}\bar{\mathbf{u}}_k^{(0)} &\leftarrow \bar{\mathbf{u}}_{k-1}^{(G)}, & \bar{\mathbf{u}}_k^{(1)} &\leftarrow \bar{\mathbf{u}}_{k-1}^{(G+1)}, & \bar{\mathbf{g}}_k^{(0)} &\leftarrow \bar{\mathbf{g}}_{k-1}^{(G)}, \\ \bar{\boldsymbol{\mu}}_k^{(1)} &\leftarrow \bar{\boldsymbol{\mu}}_{k-1}^{(G+1)}, & \boldsymbol{\mu}_{T,k}^{(1)} &\leftarrow \boldsymbol{\mu}_{T,k-1}^{(G+1)}, & \bar{\boldsymbol{\nu}}_k^{(1)} &\leftarrow \bar{\boldsymbol{\nu}}_{k-1}^{(G+1)}, \\ \nu_{T,k}^{(1)} &\leftarrow \nu_{T,k-1}^{(G+1)}\end{aligned}$$

- Vorwärtsintegration mit Algorithmus 4.4.2

$$\bar{\mathbf{x}}_k^{(1)} \leftarrow \bar{\mathbf{u}}_k^{(1)}, \mathbf{x}_k$$

for m **in** $1, 2, \dots, G$ **do**

- Rückwärtsintegration mit Algorithmus 4.4.3

$$\bar{\boldsymbol{\lambda}}_k^{(m)} \leftarrow \bar{\mathbf{u}}_k^{(m)}, \bar{\mathbf{x}}_k^{(m)}, \bar{\boldsymbol{\mu}}_k^{(m)}, \bar{\boldsymbol{\mu}}_{T,k}^{(m)}, \bar{\boldsymbol{\nu}}_k^{(m)}, \nu_{T,k}^{(m)}$$

- Abstiegsrichtung bestimmen über (4.11)

$$\mathbf{d}^{(m)}, \mathbf{g}^{(m)} \leftarrow \bar{\mathbf{u}}^{(m)}, \bar{\mathbf{x}}^{(m)}, \bar{\boldsymbol{\lambda}}^{(m)}, \bar{\boldsymbol{\mu}}^{(m)}, \bar{\boldsymbol{\nu}}^{(m)}$$

- Liniensuche mit Algorithmus 4.4.4

$$\begin{aligned}\bar{\mathbf{u}}_k^{(m+1)} &\leftarrow \bar{\mathbf{u}}_k^{(m)}, \bar{\mathbf{u}}_k^{(m-1)}, \mathbf{d}^{(m)}, \mathbf{g}^{(m)}, \mathbf{g}^{(m-1)}, \mathbf{u}_{k-1|0} \\ \bar{\mathbf{x}}_k^{(m+1)} &\leftarrow \bar{\mathbf{u}}_k^{(m+1)}, \mathbf{x}_k \\ \tilde{\mathbf{J}}_k^{(m+1)} &\leftarrow \bar{\mathbf{u}}_k^{(m+1)}, \bar{\mathbf{x}}_k^{(m+1)}, \bar{\boldsymbol{\mu}}_k^{(m)}, \bar{\boldsymbol{\mu}}_{T,k}^{(m)}, \bar{\boldsymbol{\nu}}_k^{(m)}, \nu_{T,k}^{(m)}\end{aligned}$$

- Lagrange-Multiplikatoren aktualisieren über (4.26) sowie (4.27)

$$\begin{aligned}\bar{\boldsymbol{\mu}}_k^{(m+1)} &\leftarrow \bar{\mathbf{u}}_k^{(m+1)}, \bar{\mathbf{x}}_k^{(m+1)}, \bar{\boldsymbol{\mu}}_k^{(m)}, \bar{\boldsymbol{\nu}}_k^{(m)} \\ \boldsymbol{\mu}_{T,k}^{(m+1)} &\leftarrow \mathbf{x}_{k|N-1}^{(m+1)}, \boldsymbol{\mu}_{T,k}^{(m)}, \nu_{T,k}^{(m)}\end{aligned}$$

- Strafparameter aktualisieren über (4.28) sowie (4.29)

$$\begin{aligned}\bar{\boldsymbol{\nu}}_k^{(m+1)} &\leftarrow \bar{\mathbf{u}}_k^{(m+1)}, \bar{\mathbf{x}}_k^{(m+1)}, \bar{\boldsymbol{\mu}}_k^{(m)}, \bar{\boldsymbol{\nu}}_k^{(m)} \\ \nu_{T,k}^{(m+1)} &\leftarrow \mathbf{x}_{k|N-1}^{(m+1)}, \boldsymbol{\mu}_{T,k}^{(m)}, \nu_{T,k}^{(m)}\end{aligned}$$

end for

dem GV sowie der Gradientenprojektion zuzuordnen.⁷ Die verbleibenden Ausführungsschritte stehen im Zusammenhang mit der Maximierung.

4.4.1 Gradientenverfahren

Das GV für dynamische Optimierungsprobleme ist algorithmisch analog zum Abstiegsverfahren (vgl. Algorithmus 2.2.1) aufgebaut. Über eine Gradientenprojektion $\bar{\mathbf{P}}(\cdot)$ (vgl. Abschnitt 4.4.4) werden die Nebenbedingungen (4.8c - 4.8f) sichergestellt, so dass sich die Iterationsvorschrift

$$\bar{\mathbf{u}}_k^{(m+1)} = \bar{\mathbf{P}}(\bar{\mathbf{u}}_k^{(m)} + \alpha_k^{(m)} \bar{\mathbf{d}}_k^{(m)}, \mathbf{u}_{k-1|0}), \quad \alpha_k^{(m)} > 0 \quad (4.9)$$

ergibt. Die Schrittweite $\alpha_k^{(m)}$ wird in einer separaten Liniensuche (vgl. Abschnitt 4.4.3) bestimmt.

Zur Herleitung der Abstiegsrichtung wird die Analogie zum erweiterten Lagrange-Ansatz gesucht. Bei diesem Ansatz geht die Lösung der primalen Variablen – der Stellgrößen – aus der Minimierung der erweiterten Lagrange-Funktion (2.35) hervor (vgl. Algorithmus 2.2.5). Das Pendant der erweiterten Lagrange-Funktion bezogen auf das Optimierungsproblem (4.5) ist die erweiterte Gütefunktion, in die zusätzlich das Modell (4.5b) aufgenommen wird. Die fortan als *vollständig erweiterte Gütefunktion* (VEG) bezeichnete Funktion ist über

$$\begin{aligned} \bar{J}(\mathbf{u}(\cdot), \tilde{\boldsymbol{\mu}}(\cdot); \tilde{\nu}(\cdot)) &= \tilde{V}(\mathbf{x}(t_k + T), \boldsymbol{\mu}_T, t_k + T; \nu_T) \\ &+ \int_{t_k}^{t_k+T} \tilde{H}(\mathbf{x}(\tau), \mathbf{u}(\tau), \boldsymbol{\lambda}(\tau), \boldsymbol{\mu}(\tau), \tau; \nu(\tau)) - \boldsymbol{\lambda}(\tau)^\top \dot{\mathbf{x}}(\tau) \, d\tau \end{aligned} \quad (4.10)$$

gegeben. Zur Verkürzung der Notation wurde die Hamilton-Funktion (4.6) bereits eingesetzt.⁸ Die VEG enthält sämtliche Nebenbedingungen des Optimierungsproblems, die nicht über die Gradientenprojektion (vgl. Abschnitt 4.4.4) behandelt werden. Die notwendige Bedingung zur Minimierung der VEG in Abhängigkeit von den Stellgrößen und damit ebenso zur Lösung des Minimierungsproblems in (4.5) fordert, dass deren Gradient bezüglich der Stellgrößen verschwindet (vgl. Satz 2.2.1). Im Rahmen des GV wird dies über eine Iteration in Richtung des negativen Gradienten

$$\bar{\mathbf{d}}_k^{(m)} := -\bar{\mathbf{g}}_k^{(m)} \quad (4.11)$$

⁷Die grundsätzlichen Züge des GV werden in Abschnitt 2.3.2 durch die Gleichungen (2.55) und (2.56) skizziert.

⁸Streng genommen müssten die durch das Modell bedingten Nebenbedingungen ebenfalls durch den Strafparameter bewertet werden, um dem erweiterten Lagrange-Ansatz zu genügen. Die präsentierte Funktion stellt in diesem Sinne eine Kombination aus der Lagrange- (2.23) und der erweiterten Lagrange-Funktion (2.35) dar.

erreicht. Der Gradient der VEG ist über

$$\nabla_{\mathbf{u}(t)} \bar{J}(\mathbf{u}(\cdot), \mathbf{x}(\cdot), \tilde{\boldsymbol{\mu}}(\cdot), t_k; \tilde{\nu}(\cdot)) = \int_{t_k}^{t_k+T} \nabla_{\mathbf{u}(\tau)} \tilde{H}(\mathbf{u}(\tau), \mathbf{x}(\tau), \boldsymbol{\mu}(\tau), \tau; \nu(\tau)) \, d\tau \quad (4.12)$$

gegeben. Es ist jedoch der Gradient bezüglich des zeitdiskreten Problems (4.8) gesucht. Unter Verwendung der Diskretisierung (4.7) kann der Zusammenhang

$$\begin{aligned} \nabla_{\bar{\mathbf{u}}_k} \bar{J}(\bar{\mathbf{u}}_k, \bar{\mathbf{x}}_k, \bar{\boldsymbol{\mu}}_k, t_k; \bar{\boldsymbol{\nu}}_k) &= \int_{t_k}^{t_k+T} \nabla_{\bar{\mathbf{u}}_k} \tilde{H}(\boldsymbol{\psi}_s(\tau, \bar{\mathbf{u}}_k), \boldsymbol{\psi}(\tau, \bar{\mathbf{x}}_k), \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\mu}}_k), \tau_k; \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\nu}}_k)) \, d\tau \\ &\approx \frac{1}{2} \sum_{i=0}^{M-2} (\nabla_{\bar{\mathbf{u}}_k} \tilde{H}_{k|i+1} + \nabla_{\bar{\mathbf{u}}_k} \tilde{H}_{k|i}) (\tau_{k|i+1} - \tau_{k|i}) + \nabla_{\bar{\mathbf{u}}_k} \tilde{H}_{k|M-1} (\tau_{k|N-1} - \tau_{k|M-1}) \end{aligned} \quad (4.13)$$

mit

$$\tilde{H}_{k|i} := \begin{cases} \tilde{H}(\mathbf{x}_{k|i}, \mathbf{u}_{k|i}, \boldsymbol{\lambda}_{k|i}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \nu_{k|i}) & \text{für } 0 \leq i \leq M-1 \\ \tilde{H}(\mathbf{x}_{k|i}, \mathbf{u}_{k|M-1}, \boldsymbol{\lambda}_{k|i}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \nu_{k|i}) & \text{für } M-1 < i \leq N-1 \end{cases} \quad (4.14)$$

hergestellt werden. Der vom zeitkontinuierlichen Integral abgedeckte Kontrollhorizont T_C wird durch die Trapezregel approximiert, der verbleibende Teil über die linksseitige Boxregel (Untersummenregel) unter Verwendung der letzten Diskretisierung innerhalb des Kontrollhorizonts.⁹ Unter Ausnutzung der Integralabschätzung kann der Gradient über

$$\bar{\mathbf{g}}_k^{(m)} = \left[\left(\mathbf{g}_{k|0}^{(m)} \right)^T, \left(\mathbf{g}_{k|1}^{(m)} \right)^T, \dots, \left(\mathbf{g}_{k|M-1}^{(m)} \right)^T \right]^T := \nabla_{\bar{\mathbf{u}}_k} \bar{J}(\mathbf{u}_k^{(m)}, \mathbf{x}_k^{(m)}, \tilde{\boldsymbol{\mu}}_k^{(m)}, t_k; \tilde{\boldsymbol{\nu}}_k^{(m)}) \quad (4.15)$$

mit

$$\mathbf{g}_{k|i} = \begin{cases} \frac{1}{2} \nabla_{\mathbf{u}_{k|0}} \tilde{H}_{k|0} (\tau_{k|1} - \tau_{k|0}) & \text{für } i = 0 \\ \frac{1}{2} \nabla_{\mathbf{u}_{k|i}} \tilde{H}_{k|i} (\tau_{k|i+1} - \tau_{k|i-1}) & \text{für } 0 < i < M-2 \\ \frac{1}{2} \nabla_{\mathbf{u}_{k|M-1}} \tilde{H}_{k|M-1} (\tau_{k|N-1} - \tau_{k|M-2}) & \text{für } i = M-1 \end{cases}$$

kompakt angegeben werden.¹⁰ Würde die Trapezregel auf den gesamten Prädiktionshorizont angewendet, dann wäre das letzte Block-Element $\mathbf{g}_{k|M-1}^{(m)}$ des Gradienten von mehreren Diskretisierungen der Hamilton-Funktion abhängig. In diesem Fall würden die folgenden Ausführungen zur Optimalität nicht gelten.

Die gewählte Abstiegsrichtung (4.11) minimiert $\nabla_{\bar{\mathbf{u}}_k} \bar{J}(\cdot)$, sodass bei Erreichen des Minimums – ohne aktive Steuergrößenbeschränkungen – der Gradient $\bar{\mathbf{g}}_k^{(m)} = \mathbf{0}$ erfüllt. Damit gilt ebenso $\nabla_{\mathbf{u}_{k|i}} \tilde{H}_{k|i} = \mathbf{0}$, womit Pontryagin's Minimumprinzip (vgl. Satz 2.3.2) an den Diskretisierungspunkten erfüllt wird; unter Einbeziehung der im folgenden Abschnitt

⁹Die Untersummenregel approximiert eine Funktion durch ein Rechteck analog zur Approximation einer DGL durch das RK1-Verfahren (vgl. Abb. 2.1a).

¹⁰Informationen zur Bestimmung des Gradienten für den zeitkontinuierlichen Fall sind in [Kir04, S. 334f] und in [Pap15, S. 339] zu finden.

definierten kanonischen DGLs (4.16).¹¹

Zur algorithmischen Auswertung des Gradienten sind neben den Stellgrößen $\bar{\mathbf{u}}_k^{(m)}$ die Zustände $\bar{\mathbf{x}}_k^{(m)}$ und die adjungierten Zustände $\bar{\boldsymbol{\lambda}}_k^{(m)}$ erforderlich. Die übrigen Parameter sind im Rahmen des GV als beliebig aber fest anzusehen. Die Zustände sowie die adjungierten Zustände gehen aus einer separat durchzuführenden Lösung der kanonischen DGLs hervor.

4.4.2 Lösung der kanonischen Differentialgleichungen

Die kanonischen DGLs (vgl. Abschnitt 2.3.1) des Optimierungsproblems (4.8) sind unter Verwendung von Definition 4.3.1 über

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\psi}_s(t, \bar{\mathbf{u}}_k)), \quad (4.16a)$$

$$\mathbf{x}(t_k) = \mathbf{x}_k, \quad (4.16b)$$

$$\dot{\boldsymbol{\lambda}}(\tau) = -\nabla_{\mathbf{x}} \tilde{H}(\boldsymbol{\psi}(\tau, \bar{\mathbf{x}}_k), \boldsymbol{\psi}_s(\tau, \bar{\mathbf{u}}_k), \boldsymbol{\lambda}(\tau), \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\mu}}_k), \tau; \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\nu}}_k)), \quad (4.16c)$$

$$\boldsymbol{\lambda}(t_k + T) = \nabla_{\mathbf{x}} \tilde{V}(\boldsymbol{\psi}(t_k + T, \bar{\mathbf{x}}_k), \boldsymbol{\mu}_{T,k}, t_k + T, \nu_{T,k}) \quad (4.16d)$$

gegeben. Ziel ist es, die Zustände $\bar{\mathbf{x}}_k$ und die adjungierten Zustände $\bar{\boldsymbol{\lambda}}_k$ an den Diskretisierungspunkten $\bar{\tau}_k$ zu ermitteln. Dazu wird unter Kenntnis von $\bar{\mathbf{u}}_k$ die DGL (4.16a) durch Vorwärtsintegration mit dem Startwert (4.16b) gelöst. Anschließend wird unter Kenntnis von $\bar{\mathbf{u}}_k$ sowie $\bar{\mathbf{x}}_k$ die DGL (4.16c) durch Rückwärtsintegration mit dem Startwert (4.16d) ermittelt. Die numerische Integration erfolgt über ein Einschrittverfahren nach dem Schema

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \boldsymbol{\Psi}_{\mathbf{x}}(t_i, h, \dots)$$

zur Lösung von (4.16a), bzw.

$$\boldsymbol{\lambda}_i = \boldsymbol{\lambda}_{i+1} + \boldsymbol{\Psi}_{\boldsymbol{\lambda}}(t_i, -h, \dots)$$

zur Lösung von (4.16c), wobei $\boldsymbol{\Psi}_{\mathbf{x}}(\cdot)$ und $\boldsymbol{\Psi}_{\boldsymbol{\lambda}}(\cdot)$ die zugehörigen Verfahrensfunktionen (vgl. Gleichung (2.7)) sind. Die offen gelassenen Argumente hängen von dem gewählten Integrationschema ab.¹² Die numerische Integration wird in Abschnitt 2.1.2 vorgestellt. Die zwischen zwei Diskretisierungspunkten $\tau_{k|i}, \tau_{k|i+1}$ verwendete Schrittweite h wird zu

$$h = \frac{\tau_{k|i+1} - \tau_{k|i}}{n} \quad \text{mit} \quad n = \left\lceil \frac{\tau_{k|i+1} - \tau_{k|i}}{T_{\text{int}}} \right\rceil$$

¹¹In GRAMPC wird $\mathbf{g}_{k|i} := \nabla_{\mathbf{u}_{k|i}} \tilde{H}_{k|i}$ verwendet.

¹²Bei den expliziten Einschrittverfahren ist das zusätzliche Argument der Verfahrensfunktion der aktuelle Zustand \mathbf{x}_i , bei den impliziten Einschrittverfahren der zu berechnende Zustand \mathbf{x}_{i+1} (vgl. Abschnitt 2.1.3).

gewählt. Das bedeutet, dass zwischen zwei benachbarten Diskretisierungspunkten n äquidistante Integrationsschritte durchgeführt werden, deren Schrittweite einer vorgegebenen Schrittweite T_{int} möglichst nahe kommt, diese aber nicht überschreitet. Ist der Abstand zwischen den Diskretisierungspunkten ein ganzzahliges Vielfaches von der vorgegebenen Schrittweite T_{int} , dann wird ebjenese verwendet. Auf diese Weise ist sichergestellt, dass die Diskretisierungspunkte durch die Integration exakt getroffen werden, aber trotzdem die Kontrolle der numerischen Stabilität durch die Nicht-Überschreitung einer vorgebbaren Schrittweite erhalten bleibt. Die Verwendung einer Schrittweite, die einer festen Rechenvorschrift gehorcht, sorgt für eine deterministische Integrationszeit.¹³ Die algorithmische Umsetzung der Vorwärtsintegration wird in Algorithmus 4.4.2 präsentiert, die Umsetzung der Rückwärtsintegration in Algorithmus 4.4.3.

Algorithmus 4.4.2 (Vorwärtsintegration).

Initialisierung

- Startpunkt durch (4.16b) festlegen: $\mathbf{x}_{k|0} := \mathbf{x}_k$; $\mathbf{x}^{(0)} := \mathbf{x}_k$

for i **in** $0, 1, \dots, N - 2$ **do**

- Bestimme die Anzahl der Integrationsschritte sowie die Schrittweite

$$h = \frac{\tau_{k|i+1} - \tau_{k|i}}{n}, \quad n = \left\lceil \frac{\tau_{k|i+1} - \tau_{k|i}}{T_{\text{int}}} \right\rceil$$

for j **in** $0, 1, \dots, n - 1$ **do**

- Propagiere (4.16a) um einen Zeitschritt

$$\mathbf{x}^{(j+1)} = \mathbf{x}^{(j)} + \Psi_{\mathbf{x}}(t, h, \dots)$$

end for

- Zuweisen der Lösung

$$\mathbf{x}_{k|i+1} = \mathbf{x}^{(n)}$$

- Rücksetzen der Integrationsvariablen

$$\mathbf{x}^{(0)} = \mathbf{x}^{(n)}$$

end for

Eine denkbare Alternative zum gewählten Verfahren ist die Integration mit fester Schrittweite T_{int} und einer Interpolation der Diskretisierungspunkte unter Verwendung der den jeweiligen Diskretisierungspunkt einschließenden Integrationsschritte. Rechentechnisch ver-

¹³Der Gegenentwurf in diesem Sinne wäre die Verwendung eines Integrationsverfahrens mit variabler Schrittweite.

Algorithmus 4.4.3 (Rückwärtsintegration).**Initialisierung**

- Startpunkt durch (4.16d) festlegen: $\boldsymbol{\lambda}^{(0)} := \nabla_{\mathbf{x}} \tilde{V}(\boldsymbol{\psi}(t_k + T, \bar{\mathbf{x}}), t_k + T)$

if $N = M$ **then**

- Initialisiere letzte Diskretisierung: $\boldsymbol{\lambda}_{k|M-1} := \boldsymbol{\lambda}^{(0)}$

end if**for** i **in** $N - 1, N - 2, \dots, 1$ **do**

- Bestimme die Anzahl der Integrationsschritte sowie die Schrittweite

$$h = \frac{\tau_{k|i} - \tau_{k|i-1}}{n}, \quad n = \left\lceil \frac{\tau_{k|i} - \tau_{k|i-1}}{T_{\text{int}}} \right\rceil$$

for j **in** $0, 1, \dots, n - 1$ **do**

- Propagiere (4.16c) um einen Zeitschritt

$$\boldsymbol{\lambda}^{(j+1)} = \boldsymbol{\lambda}^{(j)} + \boldsymbol{\Psi}_{\boldsymbol{\lambda}}(t, -h, \dots)$$

end for**if** $i \leq M$ **then**

- Zuweisen der Lösung

$$\boldsymbol{\lambda}_{k|i-1} = \boldsymbol{\lambda}^{(n)}$$

end if

- Rücksetzen der Integrationsvariablen

$$\boldsymbol{\lambda}^{(0)} = \boldsymbol{\lambda}^{(n)}$$

end for

spricht diese Methode eine Entlastung, sobald die Diskretisierungspunkte kein ganzzahliges Vielfaches von T_{int} voneinander entfernt sind. In diesen Fällen sind weniger Integrationschritte erforderlich. Jedoch ist dann nicht die Möglichkeit gegeben, über die Platzierung der Diskretisierungspunkte indirekt eine Obergrenze der Integrationsschrittweite in bestimmten Regionen des Prädiktionshorizonts festzulegen.¹⁴

¹⁴Diese Möglichkeit wird in Abschnitt 5.3.2 (vgl. *Konfiguration D*) ausgenutzt.

4.4.3 Liniensuche

Durch die Liniensuche wird eine akzeptable Schrittweite $\alpha_k^{(m)}$ ermittelt, die einen genügend großen Fortschritt hinsichtlich der Minimierung der Kostenfunktion sicherstellt. Es wird das Minimierungsproblem

$$\alpha_k^{(m)} = \arg \min_{\alpha > 0} \tilde{J}^{(m+1)}(\alpha)$$

gelöst. Die Kostenfunktion wird formal als univariate Funktion der Schrittweite über

$$\tilde{J}_k^{(m+1)}(\alpha) := \tilde{J}(\bar{\mathbf{u}}_k^{(m+1)}(\alpha), \bar{\mathbf{x}}(\bar{\mathbf{u}}_k^{(m+1)}(\alpha)), \bar{\boldsymbol{\mu}}_k^{(m)}, \boldsymbol{\mu}_{T,k}^{(m)}; \bar{\boldsymbol{\nu}}_k^{(m)}, \boldsymbol{\nu}_{T,k}^{(m)}) \quad (4.17)$$

unter Verwendung von $\bar{\mathbf{u}}_k^{(m+1)}(\alpha)$ aus (4.9) definiert. Die Zustandsdiskretisierungen sind als zusätzliches Argument in Form einer Funktion der Stellgrößen aufgeführt. Damit wird formal der Tatsache Rechnung getragen, dass die Zustandsdiskretisierungen vor der Auswertung der Kostenfunktion über eine Vorwärtsintegration von (4.16a) zu berechnen sind. Die Kostenfunktion selbst wird, wie bereits das Integral in (4.13), über die Trapezregel

$$\tilde{J}(\bar{\mathbf{u}}_k, \bar{\boldsymbol{\mu}}_k, \boldsymbol{\mu}_{T,k}, \bar{\boldsymbol{\nu}}_k, \boldsymbol{\nu}_{T,k}) \approx V(\mathbf{x}_{k|N-1}, \boldsymbol{\mu}_{T,k}, \tau_{k|N-1}; \boldsymbol{\nu}_{T,k}) + \frac{1}{2} \sum_{i=0}^{N-2} (\tilde{l}_{k|i+1} + \tilde{l}_{k|i})(\tau_{k|i+1} - \tau_{k|i}) \quad (4.18)$$

mit

$$\tilde{l}_{k|i} := \begin{cases} \tilde{l}(\mathbf{x}_{k|i}, \mathbf{u}_{k|i}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \boldsymbol{\nu}_{k|i}) & \text{für } 0 \leq i \leq M-1 \\ \tilde{l}(\mathbf{x}_{k|i}, \mathbf{u}_{k|M-1}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \boldsymbol{\nu}_{k|i}) & \text{für } M-1 < i \leq N-1 \end{cases}$$

näherungsweise berechnet.¹⁵

Es soll ein iteratives Schema zur inexakten Bestimmung der Schrittweite entwickelt werden. Ausgangspunkt ist die Bestimmung der Schrittweite basierend auf drei Stützstellen nach [GK12]. Die Gütefunktion wird durch ein Polynom

$$\tilde{J}(\alpha) \approx \phi(\alpha) := k_0 + k_1\alpha + k_2\alpha^2$$

approximiert. Für deren Minimum gilt

$$\alpha^*(\alpha_1, \alpha_2, \alpha_3) := \arg \min_{\alpha > 0} \phi(\alpha) = -\frac{k_1}{2k_2}, \quad \text{wenn } k_2 > 0. \quad (4.19)$$

¹⁵Formal könnte ebenso die zeitdiskrete Version der VEG (4.10) verwendet werden, welche allerdings durch die per Konstruktion des Algorithmus eingehaltene DGL (4.16a) identisch zu $\tilde{J}(\cdot)$ ist.

Die Koeffizienten k_1, k_2 werden unter Verwendung der Stützstellen $\alpha_1 < \alpha_2 < \alpha_3$ durch

$$k_1 = \frac{(\alpha_2^2 - \alpha_1^2)\tilde{J}(\alpha_3) + (\alpha_1^2 - \alpha_3^2)\tilde{J}(\alpha_2) + (\alpha_3^2 - \alpha_2^2)\tilde{J}(\alpha_1)}{(\alpha_1 - \alpha_2)(\alpha_1 - \alpha_3)(\alpha_2 - \alpha_3)},$$

$$k_2 = \frac{(\alpha_1 - \alpha_2)\tilde{J}(\alpha_3) + (\alpha_3 - \alpha_1)\tilde{J}(\alpha_2) + (\alpha_2 - \alpha_3)\tilde{J}(\alpha_1)}{(\alpha_1 - \alpha_2)(\alpha_1 - \alpha_3)(\alpha_2 - \alpha_3)}$$

berechnet. Dieses Schema ist erst anwendbar, sobald drei Stützstellen berechnet wurden. Die Polynom-Schrittweitenbestimmung (4.19) wird in eine Heuristik eingebettet, welche mit einem Backtracking (vgl. Algorithmus 2.2.2) mit dem Verkleinerungsfaktor $\alpha_{\text{dec}} \in \mathbb{R}_{>0, <1}$ startet und auf diese Weise eine untere Grenze $\alpha_{1,k}^{(l)}$ für die Schrittweite ermittelt. Die untere Grenze wird aktualisiert, wenn die Goldstein-Bedingung (vgl. linke Seite in (2.20)) verletzt wird, wenn eine Verkleinerung der Schrittweite zu einer Verschlechterung geführt hat (Vergrößerung der Gütefunktion) oder wenn eine Vergrößerung der Schrittweite eine Verbesserung erzielt hat. Es gilt:

$$\alpha_{1,k}^{(l)} = \begin{cases} \alpha_k^{(m,l)} & \text{für } \begin{aligned} &\tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) < \tilde{J}^{(m+1)}(0) + c_2 \alpha_k^{(m,l)} (\bar{\mathbf{g}}^{(m)})^T \bar{\mathbf{d}}^{(m)} \\ &\vee (\tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) \geq \tilde{J}^{(m+1)}(\alpha_k^{(m,l-1)}) \wedge \alpha_k^{(m,l)} \leq \alpha_k^{(m,l-1)}) \end{aligned} \\ \alpha_k^{(m,l-1)} & \text{für } \tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) < \tilde{J}^{(m+1)}(\alpha_k^{(m,l-1)}) \wedge \alpha_k^{(m,l)} > \alpha_k^{(m,l-1)} \\ \alpha_{1,k}^{(l-1)} & \text{sonst} \end{cases} \quad (4.20)$$

mit

$$\alpha_{1,k}^{(0)} = \emptyset.$$

Die leere Menge wird verwendet, um einen ungültigen und damit nicht nutzbaren Wert zu kennzeichnen. Initial ist $\alpha_{1,k}^{(0)}$ nicht definiert und darf daher nicht berücksichtigt werden. Eine Vergrößerung der Schrittweite findet statt, wenn die untere Grenze erstmalig aktualisiert wurde. Analog zur unteren Grenze existiert eine obere Grenze $\alpha_{u,k}^{(l)}$, welche in ähnlicher Weise berechnet wird. Statt der Goldstein-Bedingung wird die Armijo-Bedingung (2.19) verwendet, um eine unzulässig große Schrittweite zu erkennen. Die obere Grenze ist über

$$\alpha_{u,k}^{(l)} = \begin{cases} \alpha_k^{(m,l)} & \text{für } \begin{aligned} &\tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) > \tilde{J}^{(m+1)}(0) + c_1 \alpha_k^{(m,l)} (\bar{\mathbf{g}}^{(m)})^T \bar{\mathbf{d}}^{(m)} \\ &\vee (\tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) \geq \tilde{J}^{(m+1)}(\alpha_k^{(m,l-1)}) \wedge \alpha_k^{(m,l)} > \alpha_k^{(m,l-1)}) \end{aligned} \\ \alpha_k^{(m,l-1)} & \text{für } \tilde{J}^{(m+1)}(\alpha_k^{(m,l)}) < \tilde{J}^{(m+1)}(\alpha_k^{(m,l-1)}) \wedge \alpha_k^{(m,l)} \leq \alpha_k^{(m,l-1)} \\ \alpha_{u,k}^{(l-1)} & \text{sonst} \end{cases} \quad (4.21)$$

mit

$$\alpha_{u,k}^{(0)} = \emptyset$$

gegeben. Solange die obere Grenze nicht bekannt ist, wird in einer inversen Vorgehensweise zum Backtracking die Schrittweite um einen festen Faktor $\alpha_{\text{inc}} \in \mathbb{R}_{>1}$ vergrößert. Insgesamt

folgt die Schrittweitenberechnung unter Nutzung von (2.22b) der Vorgehensweise

$$\alpha_k^{\langle m, l+1 \rangle} = P(\tilde{\alpha}_k^{\langle m, l+1 \rangle}, \alpha_{\min}, \alpha_{\max}), \quad (4.22)$$

$$\tilde{\alpha}_k^{\langle m, l+1 \rangle} = \begin{cases} \alpha_{\text{dec}} \alpha_k^{\langle m, l \rangle} & \text{für } \alpha_{1,k}^{\langle l \rangle} = \emptyset \\ \alpha_{\text{inc}} \alpha_k^{\langle m, l \rangle} & \text{für } \alpha_{u,k}^{\langle l \rangle} = \emptyset \wedge \alpha_{1,k}^{\langle l \rangle} \neq \emptyset \\ \alpha_k^*(\alpha_{1,k}^{\langle l \rangle}, \alpha_2, \alpha_{u,k}^{\langle l \rangle}) & \text{für } \alpha_{1,k}^{\langle l \rangle} < \alpha^*(\alpha_{1,k}^{\langle l \rangle}, \alpha_2, \alpha_{u,k}^{\langle l \rangle}) < \alpha_{u,k}^{\langle l \rangle} \wedge \alpha_{1,k}^{\langle l \rangle}, \alpha_2, \alpha_{u,k}^{\langle l \rangle} \neq \emptyset \\ (\alpha_{1,k}^{\langle l \rangle} + \alpha_{u,k}^{\langle l \rangle}) / 2 & \text{sonst} \end{cases}$$

mit

$$\alpha_2 = \begin{cases} \alpha_k^{\langle m, l \rangle} & \text{für } \alpha_{1,k}^{\langle l \rangle} < \alpha_k^{\langle m, l \rangle} < \alpha_{u,k}^{\langle l \rangle} \\ \alpha_k^{\langle m, l-1 \rangle} & \text{für } \alpha_{1,k}^{\langle l \rangle} < \alpha_k^{\langle m, l-1 \rangle} < \alpha_{u,k}^{\langle l \rangle} \\ \emptyset & \text{sonst.} \end{cases}$$

Es sei angemerkt, dass $\alpha_k^{\langle m, l \rangle}$ und $\alpha_k^{\langle m, l-1 \rangle}$ nicht gleichzeitig zwischen der unteren und der oberen Grenze liegen können.

Die initiale Schrittweite geht aus der Schrittweitenbestimmung nach *Barzilai und Borwein* hervor.¹⁶ Es folgt gemäß (2.21a)

$$\alpha_{\text{bb},k}^{\langle m, 0 \rangle} = \frac{(\Delta \bar{\mathbf{u}}_k^{\langle m \rangle})^\top \Delta \bar{\mathbf{u}}_k^{\langle m \rangle}}{(\Delta \bar{\mathbf{u}}_k^{\langle m \rangle})^\top \Delta \bar{\mathbf{g}}_k^{\langle m \rangle}} \quad (4.23)$$

mit

$$\begin{aligned} \Delta \bar{\mathbf{u}}_k^{\langle m \rangle} &= \bar{\mathbf{u}}_k^{\langle m \rangle} - \bar{\mathbf{u}}_k^{\langle m-1 \rangle}, \\ \Delta \bar{\mathbf{g}}_k^{\langle m \rangle} &= \bar{\mathbf{g}}_k^{\langle m \rangle} - \bar{\mathbf{g}}_k^{\langle m-1 \rangle}. \end{aligned}$$

Wie in Abschnitt 2.2.2 ausgeführt wird, können Situationen auftreten, in denen in (4.23) eine Null oder eine negative Zahl im Nenner steht. Eine Division durch Null ist mathematisch nicht zulässig und ein negativer Nenner führt zu einer negativen Schrittweite, welche ebenfalls unzulässig ist. In diesen Situationen wird die bereits in GRAMPC genutzte Ersatzberechnung verwendet. Die Idee ist, in diesen Fällen die Schrittweite derart zu wählen, dass sich bei Anwendung der Schrittweite die Position des am stärksten bewegten Aktuators um 1 % seines Wertebereichs verändert:

$$\alpha_{1\%,k}^{\langle m, 0 \rangle} = \frac{1}{100} \cdot \min_{j=1,2,\dots,m} \left\{ \frac{\mathbf{b}_{u,j} - \mathbf{b}_{l,j}}{\mathbf{d}_{\max,j}} \right\}, \quad \mathbf{d}_{\max,j} = \|\bar{\mathbf{d}}_{k,j}^{\langle m \rangle}\|_\infty, \quad \mathbf{b}_u, \mathbf{b}_l, \mathbf{d}_{\max,j} \in \mathbb{R}^m.$$

¹⁶Die Verwendung der Schrittweitenbestimmung nach *Barzilai und Borwein* wurde in [KG13] für das GV im Kontext der dynamischen Optimierung eingeführt. Das Verfahren wird abseits von GRAMPC in dem MPC-Framework VIATOC [KBV15] eingesetzt.

Insgesamt folgt die Bestimmung der initialen Schrittweite der Vorschrift

$$\alpha_k^{\langle m,0 \rangle} = \begin{cases} \alpha_{\text{fac}} \alpha_{\text{bb},k}^{\langle m,0 \rangle} & \text{für } (\Delta \bar{\mathbf{u}}_k^{\langle m \rangle})^T \Delta \bar{\mathbf{g}}_k^{\langle m \rangle} > 0 \\ \alpha_{1\%,k}^{\langle m,0 \rangle} & \text{sonst.} \end{cases} \quad (4.24)$$

Die Schrittweitschätzung (4.23) wird mit dem Skalierungsfaktor $\alpha_{\text{fac}} \in \mathbb{R}_{>0, <1}$ versehen, um die initiale Schrittweite zur Steigerung der Robustheit kleiner wählen zu können.

Die Liniensuche ist in Algorithmus 4.4.4 zusammengefasst.

4.4.4 Gradientenprojektion mit Änderungsratenbeschränkung

Mithilfe einer Gradientenprojektion werden die Nebenbedingungen (4.8c - 4.8f) berücksichtigt. Die Wirkungsweise der Gradientenprojektion ist eine Begrenzung der Optimierungsvariablen derart, dass die an der Gradientenprojektion beteiligten Nebenbedingungen unmittelbar, ohne ein iteratives Näherungsverfahren eingehalten werden. Veranschaulicht ist das Vorgehen in Abb. 2.6.

Bedingt durch die Änderungsratenbegrenzung bauen die Nebenbedingungen aufeinander auf, und zwar in einer Weise, dass für die Anwendung der Projektion auf eine Stellgröße $\mathbf{u}_{k|i}$ die Projektion der vorausgehenden $\mathbf{u}_{k|i-1}$ bekannt sein muss. Die erste Projektion in dieser Kaskade wird auf die Nebenbedingungen (4.8c) sowie (4.8e) angewendet. Diese Bedingungen hängen von der im vorigen Zeitschritt angewendeten Steuerung $\mathbf{u}_{k-1|0}$ ab, welche bekannt ist. Es folgt für die Projektion bezüglich dieser Nebenbedingungen

$$\mathbf{P}_0 = \mathbf{P} \left(\mathbf{u}_{k|0}, \max \{ \mathbf{b}_l, \mathbf{u}_{k-1|0} + T_s \mathbf{r}_l \}, \min \{ \mathbf{b}_u, \mathbf{u}_{k-1|0} + T_s \mathbf{r}_u \} \right).$$

Die übrigen Nebenbedingungen (4.8d) sowie (4.8f) werden durch die Vorschrift

$$\mathbf{P}_i = \mathbf{P} \left(\mathbf{u}_{k|i}, \max \{ \mathbf{b}_l, \mathbf{P}_{i-1} + T_s \mathbf{r}_l \}, \min \{ \mathbf{b}_u, \mathbf{P}_{i-1} + T_s \mathbf{r}_u \} \right), \quad i = 1, 2, \dots, M - 1$$

berücksichtigt. Insgesamt folgt damit die Projektionsvorschrift

$$\bar{\mathbf{P}}(\bar{\mathbf{u}}_k, \mathbf{u}_{k-1|0}) := \left[\mathbf{P}_0^T, \mathbf{P}_1^T, \dots, \mathbf{P}_{M-1}^T \right]^T. \quad (4.25)$$

Die Projektion ist in einer rechentechnischen Umsetzung durch ein iteratives Schema, beginnend mit \mathbf{P}_0 , zu berechnen.

Algorithmus 4.4.4 (Liniensuche).

Initialisierung

- Grenzen initialisieren: $\alpha_{1,k}^{(0)} := 0, \alpha_{u,k}^{(l)} := 0$
- Initiale Schrittweite nach (4.24) berechnen

$$\alpha_k^{(m,0)} \leftarrow \bar{\mathbf{u}}_k^{(m)}, \bar{\mathbf{u}}_k^{(m-1)}, \bar{\mathbf{g}}_k^{(m)}, \bar{\mathbf{g}}_k^{(m-1)}$$

for l **in** $0, 1, \dots, L - 1$ **do**

- Steuertrajektorie über (4.9) berechnen

$$\bar{\mathbf{u}}_k^{(m+1,l)} \leftarrow \bar{\mathbf{u}}_k^{(m)}, \alpha_k^{(m,l)}, \bar{\mathbf{d}}_k^{(m)}, \mathbf{u}_{k-1|0}$$

- Vorwärtsintegration mit Algorithmus 4.4.2

$$\bar{\mathbf{x}}_k^{(m+1,i)} \leftarrow \bar{\mathbf{u}}_k^{(m+1,l)}, \mathbf{x}_k$$

- Gütefunktion über (4.18) auswerten

$$\tilde{J}_k^{(m+1,l)} \leftarrow \bar{\mathbf{u}}_k^{(m+1,l)}, \bar{\mathbf{x}}_k^{(m+1,l)}, \bar{\boldsymbol{\mu}}_k^{(m)}, \bar{\boldsymbol{\mu}}_{T,k}^{(m)}, \bar{\boldsymbol{\nu}}_k^{(m)}, \bar{\boldsymbol{\nu}}_{T,k}^{(m)}$$

if $l = 0 \vee \tilde{J}_k^{(m+1,l)} < \tilde{J}_k^{(m+1)}$ **then**

- Lösung aktualisieren

$$\tilde{J}_k^{(m+1)} = \tilde{J}_k^{(m+1,l)}, \quad \bar{\mathbf{u}}_k^{(m+1)} = \bar{\mathbf{u}}_k^{(m+1,l)}, \quad \bar{\mathbf{x}}_k^{(m+1)} = \bar{\mathbf{x}}_k^{(m+1,l)}$$

end if

if $l < L - 1$ **then**

- Grenzen mit (4.20), (4.21) aktualisieren und Gütefunktionen hinterlegen

$$\begin{aligned} \alpha_{1,k}^{(l)} &\leftarrow \alpha_k^{(m,l)}, \alpha_k^{(m,l-1)}, \bar{\mathbf{g}}_k^{(m)}, \bar{\mathbf{d}}_k^{(m)}, \tilde{J}_k^{(m,l)}, \tilde{J}_k^{(m,l-1)}, \\ \tilde{J}_{l,k}^{(l)} &\leftarrow \tilde{J}_k^{(m+1,l)} \quad \text{oder} \quad \tilde{J}_{l,k}^{(m+1,l-1)} \quad \text{oder} \quad \tilde{J}_{l,k}^{(l-1)}, \\ \alpha_{u,k}^{(l)} &\leftarrow \alpha_k^{(m,l)}, \alpha_k^{(m,l-1)}, \bar{\mathbf{g}}_k^{(m)}, \bar{\mathbf{d}}_k^{(m)}, \tilde{J}_k^{(m,l)}, \tilde{J}_k^{(m,l-1)}, \\ \tilde{J}_{u,k}^{(l)} &\leftarrow \tilde{J}_k^{(m+1,l)} \quad \text{oder} \quad \tilde{J}_k^{(m+1,l-1)} \quad \text{oder} \quad \tilde{J}_{u,k}^{(l-1)} \end{aligned}$$

- Schrittweite für die nächste Iteration berechnen mit (4.22)

$$\tilde{\alpha}_k^{(m,l+1)} \leftarrow \alpha_{1,k}^{(l)}, \alpha_{u,k}^{(l)}, \alpha_k^{(m,l)}, \alpha_k^{(m,l-1)}$$

end if

end for

4.4.5 Aktualisierung der Lagrange-Multiplikatoren sowie der Strafparameter

Bisher wurde die Lösung des Minimierungsproblems in (4.8) über ein GV besprochen. Noch ausstehend ist die Lösung des Maximierungsproblems. Bevor im Anschluss die Bestimmung der Strafparameter besprochen wird, soll zunächst die eigentliche Lösung des Maximierungsproblems über eine Anstiegsiteration der Lagrange-Multiplikatoren besprochen werden.

Wie bereits bei der Bestimmung der Abstiegsrichtung in Abschnitt 4.4, wird die VEG (4.10) ebenso zur Bestimmung der Anstiegsrichtung herangezogen. Die notwendige Bedingung für einen Extrempunkt ist das Verschwinden des Gradienten der VEG bezüglich der Lagrange-Multiplikatoren. Es gilt

$$\begin{aligned} \nabla_{\tilde{\boldsymbol{\mu}}(t)} \bar{J}(\mathbf{u}(\cdot), \mathbf{x}(\cdot), \tilde{\boldsymbol{\mu}}(\cdot), t_k; \tilde{\nu}(\cdot)) &= \nabla_{\tilde{\boldsymbol{\mu}}(t)} \tilde{V}(\mathbf{x}(t_k + T), \boldsymbol{\mu}_T, t_k + T; \nu_T) \\ &+ \int_{t_k}^{t_k+T} \nabla_{\tilde{\boldsymbol{\mu}}(t)} \tilde{H}(\mathbf{u}(\tau), \mathbf{x}(\tau), \boldsymbol{\mu}(\tau), \tau; \nu(\tau)) d\tau. \end{aligned}$$

Analog zu (4.13) geht der Gradient bezüglich des zu lösenden diskreten Optimierungsproblems (4.8) aus

$$\begin{aligned} \nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \bar{J}(\bar{\mathbf{u}}_k, \bar{\mathbf{x}}_k, \bar{\boldsymbol{\mu}}_k, t_k; \bar{\boldsymbol{\nu}}_k) &= \nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \tilde{V}(x_{k|N-1}, \boldsymbol{\mu}_{T,k}, \tau_{k|N-1}; \boldsymbol{\nu}_{T,k}) \\ &+ \int_{t_k}^{t_k+T} \nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \tilde{H}(\boldsymbol{\psi}_s(\tau, \bar{\mathbf{u}}_k), \boldsymbol{\psi}(\tau, \bar{\mathbf{x}}_k), \boldsymbol{\psi}(\tau, \bar{\boldsymbol{\mu}}_k), \tau_k; \boldsymbol{\psi}(\tau, \bar{\nu}_k)) d\tau \\ &\approx \nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \tilde{V}(x_{k|N-1}, \boldsymbol{\mu}_{T,k}, \tau_{k|N-1}; \boldsymbol{\nu}_{T,k}) \\ &+ \frac{1}{2} \sum_{i=0}^{N-2} (\nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \tilde{H}_{k|i+1} + \nabla_{\bar{\boldsymbol{\lambda}}_k, \boldsymbol{\lambda}_{T,k}} \tilde{H}_{k|i}) (\tau_{k|i+1} - \tau_{k|i}) \end{aligned}$$

unter Verwendung von (4.14) durch die Anwendung der Trapezregel hervor. Die Anstiegsrichtung zur Bestimmung der Lagrange-Multiplikatoren $\bar{\boldsymbol{\mu}}_k, \boldsymbol{\mu}_{T,k}$ ist damit über

$$\begin{aligned} \bar{\mathbf{a}}_k^{(m+1)} &= \left[\left(\mathbf{a}_{k|0}^{(m+1)} \right)^T, \left(\mathbf{a}_{k|1}^{(m+1)} \right)^T, \dots, \left(\mathbf{a}_{k|N-1}^{(m+1)} \right)^T \right]^T := \nabla_{\bar{\boldsymbol{\lambda}}_k} \bar{J}(\mathbf{u}_k^{(m+1)}, \mathbf{x}_k^{(m+1)}, \tilde{\boldsymbol{\mu}}_k^{(m)}, t_k; \tilde{\nu}_k^{(m)}), \\ \mathbf{a}_{T,k}^{(m+1)} &:= \nabla_{\boldsymbol{\lambda}_{T,k}} \bar{J}(\mathbf{u}_k^{(m+1)}, \mathbf{x}_k^{(m+1)}, \tilde{\boldsymbol{\mu}}_k^{(m)}, t_k; \tilde{\nu}_k^{(m)}) \end{aligned}$$

mit

$$\mathbf{a}_{k|i} = \begin{cases} \frac{1}{2} \tilde{\mathbf{c}}_{k|0} (\tau_{k|1} - \tau_{k|0}) & \text{für } i = 0 \\ \frac{1}{2} \tilde{\mathbf{c}}_{k|i} (\tau_{k|i+1} - \tau_{k|i-1}) & \text{für } 0 < i < N - 2, \\ \frac{1}{2} \tilde{\mathbf{c}}_{k|N-1} (\tau_{k|N-1} - \tau_{k|N-2}) & \text{für } i = N - 1 \end{cases}, \quad \mathbf{a}_{T,k} = \tilde{\mathbf{c}}_{T,k}$$

sowie

$$\begin{aligned}\tilde{\mathbf{c}}_{k|i} &= \begin{cases} \tilde{\mathbf{c}}(\mathbf{x}_{k|i}, \mathbf{u}_{k|i}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \nu_{k|i}) & \text{für } 0 \leq i \leq M-1 \\ \tilde{\mathbf{c}}(\mathbf{x}_{k|i}, \mathbf{u}_{k|M-1}, \boldsymbol{\mu}_{k|i}, \tau_{k|i}; \nu_{k|i}) & \text{für } M-1 < i \leq N-1, \end{cases} \\ \tilde{\mathbf{c}}_{T,k} &= \tilde{\mathbf{c}}_{T,k}(\mathbf{x}_{k|N-1}, \boldsymbol{\mu}_{T,k}, \tau_{k|N-1}; \nu_{T,k})\end{aligned}$$

gegeben.¹⁷

Die eigentliche Anstiegsiteration erfolgt analog zu (4.9) unter Verwendung einer Gradientenprojektion (2.22a), um etwaige numerische Probleme durch zu große Lagrange-Multiplikatoren auszuschließen. Die Multiplikatoren werden unter Verwendung von (2.36) gemäß

$$\begin{aligned}\boldsymbol{\mu}_{k|i}^{(m+1)} &= \mathbf{P}(\boldsymbol{\mu}_{k|i}^{(m+1)}, -\mu_{\max}, \mu_{\max}), \\ \tilde{\boldsymbol{\mu}}_{k|i}^{(m+1)} &= \begin{cases} \boldsymbol{\mu}_{k|i}^{(m)} + \beta \nu_{k|i} \mathbf{a}_{k|i}^{(m+1)} & \text{für } \forall \tilde{c}_{k|i,j} (\tilde{c}_{k|i,j} < \varepsilon_{\tilde{c},j}) \wedge \nu_{k|i}^{(m)} > \nu_{\min} \wedge \eta_k^{(m)} < \varepsilon_{\mathbf{u}} \\ \boldsymbol{\mu}_{k|i}^{(m)} & \text{sonst} \end{cases}\end{aligned}\quad (4.26)$$

sowie

$$\begin{aligned}\boldsymbol{\mu}_{T,k}^{(m+1)} &= \mathbf{P}(\boldsymbol{\mu}_{T,k}^{(m+1)}, -\mu_{\max}, \mu_{\max}), \\ \boldsymbol{\mu}_{T,k}^{(m+1)} &= \begin{cases} \boldsymbol{\mu}_{T,k}^{(m)} + \beta \nu_{T,k} \mathbf{a}_{T,k}^{(m+1)} & \text{für } \forall \tilde{c}_{T,k,j} (\tilde{c}_{T,k,j} < \varepsilon_{\tilde{c},j}) \wedge \nu_{T,k}^{(m)} > \nu_{\min} \wedge \eta_k^{(m)} < \varepsilon_{\mathbf{u}} \\ \boldsymbol{\mu}_{T,k}^{(m)} & \text{sonst} \end{cases}\end{aligned}\quad (4.27)$$

mit einer festen Schrittweite $\beta \in \mathbb{R}_+$ aktualisiert. Die Aktualisierung ist an Bedingungen geknüpft: Sämtliche Nebenbedingungen befinden sich innerhalb der Toleranzen $\varepsilon_{\tilde{\mathbf{c}}} \in \mathbb{R}_+^{|\mathcal{E}|+|\mathcal{I}|}$ bzw. $\varepsilon_{\tilde{\mathbf{c}}_T} \in \mathbb{R}_+^{|\mathcal{E}_T|+|\mathcal{I}_T|}$, die Strafparameter $\nu_{k|i}$ bzw. $\nu_{T,k}$ sind größer als deren Minimalwert $\nu_{\min} \in \mathbb{R}_+$ und die Gradienteniteration (4.9) des GV ist konvergiert.

Die Konvergenz des GV wird über die Metrik

$$\eta_k^{(m)} = \max_{j=1,2,\dots,m} \left\{ \frac{1}{\sqrt{M}} \frac{\|\bar{\mathbf{u}}_{k,j}^{(m)} - \bar{\mathbf{u}}_{k,j}^{(m-1)}\|_2}{\mathbf{u}_{\max,j} - \mathbf{u}_{\min,j}} \right\}, \quad \mathbf{u}_{\min}, \mathbf{u}_{\max} \in \mathbb{R}^m$$

bewertet. Das GV gilt als konvergiert, wenn das Maximum der NRMSEs $\eta_k^{(m)}$ der einzelnen Stellgrößen unterhalb des Schwellwertes $\varepsilon_{\mathbf{u}} \in \mathbb{R}_+$ liegt.¹⁸

Das Update der Strafparameter erfolgt ebenfalls unter der Voraussetzung, dass das GV konvergiert ist. Eine Erhöhung um den Faktor $\gamma_{\text{inc}} \in \mathbb{R}_{>1}$ findet statt, wenn mindestens eine

¹⁷Bei der Differentiation von $\tilde{H}(\cdot)$ sind bei der Ableitung der Ungleichungsbedingungen die beiden Fälle $c_j^+(\cdot) = c_j(\cdot)$ sowie $c_j^+(\cdot) = \mu_j/\nu$ zu unterscheiden (vgl. Gleichung (2.34)). Die Ableitung von $\tilde{H}(\cdot)$ ist für beide Fälle identisch, weswegen $\tilde{\mathbf{c}}(\cdot)$ bei der Herleitung der Ableitung formal als unabhängig von $\boldsymbol{\mu}$ betrachtet werden kann.

¹⁸Der NRMSE wird in Gleichung (A.6) definiert.

Nebenbedingung den korrespondierenden Schwellwert in $\varepsilon_{\tilde{\mathbf{c}}} \in \mathbb{R}_+$ bzw. $\varepsilon_{\tilde{\mathbf{c}}_T} \in \mathbb{R}_+$ verletzt. Verkleinert wird der Strafparameter um einen Faktor $\gamma_{\text{dec}} \in \mathbb{R}_{>0, <1}$, wenn keine Nebenbedingung ein Hundertstel des Schwellwertes überschreitet. Die Strafparameteraktualisierungen sind über

$$\begin{aligned} \nu_{k|i}^{\langle m+1 \rangle} &= P(\tilde{\nu}_{k|i}^{\langle m+1 \rangle}, \nu_{\min}, \nu_{\max}), \\ \tilde{\nu}_{k|i}^{\langle m+1 \rangle} &= \begin{cases} \gamma_{\text{inc}} \nu_{k|i}^{\langle m \rangle} & \text{für } \exists \tilde{\mathbf{c}}_{k|i,j} (\tilde{\mathbf{c}}_{k|i,j} \geq \varepsilon_{\tilde{\mathbf{c},j}}) \wedge \eta_k^{\langle m \rangle} < \varepsilon_{\mathbf{u}} \\ \gamma_{\text{dec}} \nu_{k|i}^{\langle m \rangle} & \text{für } \forall \tilde{\mathbf{c}}_{k|i,j} (\tilde{\mathbf{c}}_{k|i,j} < 0,01 \cdot \varepsilon_{\tilde{\mathbf{c},j}}) \wedge \eta_k^{\langle m \rangle} < \varepsilon_{\mathbf{u}} \\ \nu_{k|i}^{\langle m \rangle} & \text{sonst} \end{cases} \end{aligned} \quad (4.28)$$

für die integralen Strafparameter und über

$$\begin{aligned} \nu_{T,k}^{\langle m+1 \rangle} &= P(\tilde{\nu}_{T,k}^{\langle m+1 \rangle}, \nu_{\min}, \nu_{\max}), \\ \tilde{\nu}_{T,k}^{\langle m+1 \rangle} &= \begin{cases} \gamma_{\text{inc}} \nu_{T,k}^{\langle m \rangle} & \text{für } \exists \tilde{\mathbf{c}}_{T,k,j} (\tilde{\mathbf{c}}_{T,k,j} \geq \varepsilon_{\tilde{\mathbf{c},j}}) \wedge \eta_k^{\langle m \rangle} < \varepsilon_{\mathbf{u}} \\ \gamma_{\text{dec}} \nu_{T,k}^{\langle m \rangle} & \text{für } \forall \tilde{\mathbf{c}}_{T,k,j} (\tilde{\mathbf{c}}_{T,k,j} < 0,01 \cdot \varepsilon_{\tilde{\mathbf{c},j}}) \wedge \eta_k^{\langle m \rangle} < \varepsilon_{\mathbf{u}} \\ \nu_{T,k}^{\langle m \rangle} & \text{sonst} \end{cases} \end{aligned} \quad (4.29)$$

für den terminalen Strafparameter gegeben.

Das Zusammenspiel zwischen Multiplikator- und Strafparameterupdate ist dem Softwarepaket LANCELOT [CGT92], [NW06, S. 520f] entnommen worden. Von dort stammt die Idee, bei einer Vergrößerung der Strafparameter die Lagrange-Multiplikatoren nicht gleichzeitig zu verstellen. Aus GRAMPC wurde der Ansatz übernommen, zwei unterschiedliche Schwellwerte für das Erhöhen bzw. Verringern der Strafparameter zu verwenden.¹⁹

4.4.6 Initialisierung

Wie in Algorithmus 4.4.1 deutlich wird, sind zur Ausführung des erweiterten GV Startwerte vorzugeben. Ziel ist es, diese aus der vorigen MPC-Ausführung abzuleiten. Auf diese Weise profitiert die aktuelle Ausführung von den Berechnungen im vorigen Zeitschritt, was sich in einer höheren Konvergenzgeschwindigkeit niederschlägt. Tendenziell ist davon auszugehen, dass die berechneten Stellgrößen für den vorigen Zeitschritt näherungsweise ebenso auf den aktuellen angewendet werden können. Die zeitabhängigen Größen werden aus einer Interpolation der Diskretisierungen (vgl. Abschnitt 4.3.2) der vorigen Ausführung gewonnen. Die Stellgrößen sowie der Gradient werden entsprechend über

$$\begin{aligned} \mathbf{u}_{k|i}^{\langle 1 \rangle} &= \bar{\mathbf{P}} \left(\boldsymbol{\psi} \left(\tau_{k|i}, \bar{\mathbf{u}}_{k-1}^{\langle G+1 \rangle} \right), \mathbf{u}_{k|-1} \right), & \mathbf{u}_{k|i}^{\langle 0 \rangle} &= \bar{\mathbf{P}} \left(\boldsymbol{\psi} \left(\tau_{k|i}, \bar{\mathbf{u}}_{k-1}^{\langle G \rangle} \right), \mathbf{u}_{k|-1} \right), \\ \mathbf{g}_{k|i}^{\langle 0 \rangle} &= \boldsymbol{\psi} \left(\tau_{k|i}, \bar{\mathbf{g}}_{k-1}^{\langle G \rangle} \right), & i &= 0, 1, \dots, M-1 \end{aligned}$$

¹⁹In GRAMPC besitzt jede Nebenbedingung einen Strafparameter, entsprechend ist die Umsetzung dort anders gestaltet.

aktualisiert. Für die Stellgrößen wird die lineare Interpolation anstelle der sonst verwendeten stufenweisen Interpolation eingesetzt und es wird anschließend die Gradientenprojektion (4.25) angewendet. Die Projektion ist primär aus dem Grunde erforderlich, weil $\mathbf{u}_{k|i-1}$ nicht zwangsweise der im vorigen Zeitschritt angewendeten Stellgröße entsprechen muss. Durch externe Eingriffe wegen der Umsetzung im eingebetteten System sind davon abweichende Werte denkbar. Des Weiteren werden mithilfe der Projektion durch die Interpolation hervorgerufene Verletzungen der Änderungsratenbeschränkungen \mathbf{r}_l und \mathbf{r}_u in (4.8c - 4.8f) abgefangen; dieser Aspekt ist von hintergründiger Bedeutung. Die übrigen zeitabhängigen Größen werden über

$$\boldsymbol{\mu}_{k|i}^{\langle 1 \rangle} = \boldsymbol{\psi} \left(\tau_{k|i}, \bar{\boldsymbol{\mu}}_{k-1}^{\langle G+1 \rangle} \right), \quad \nu_{k|i}^{\langle 1 \rangle} = \boldsymbol{\psi} \left(\tau_{k|i}, \bar{\nu}_{k-1}^{\langle G+1 \rangle} \right), \quad i = 1, 2, \dots, N - 1$$

interpoliert.

Die zeitunabhängigen terminalen Lagrange-Multiplikatoren sowie die terminalen Strafparameter werden unverändert übernommen:

$$\boldsymbol{\mu}_{T,k}^{\langle 1 \rangle} = \boldsymbol{\mu}_{T,k-1}^{\langle G+1 \rangle}, \quad \nu_{T,k}^{\langle 1 \rangle} = \nu_{T,k-1}^{\langle G+1 \rangle}.$$

Neben dem zuvor besprochenen *Warmstart* des Algorithmus tritt ebenso die Situation auf, dass keine nutzbaren vorausgegangen Berechnungen zur Verfügung stehen. Dies ist i. d. R. beim Einschalten des zu regelnden Systems der Fall; es wird von einem *Kaltstart* gesprochen. In solchen Situationen werden die Stellgrößen mit einem vorgebbaren Wert \mathbf{u}_{ini} initialisiert und der Gradient mit Null:

$$\mathbf{u}_{k|i}^{\langle 1 \rangle} = \mathbf{u}_{k|i}^{\langle 0 \rangle} = \mathbf{P}(\mathbf{u}_{\text{ini}}, \mathbf{b}_l, \mathbf{b}_u), \quad \mathbf{g}_{k|i}^{\langle 0 \rangle} = \mathbf{0}, \quad i = 0, 1, \dots, M - 1.$$

Des Weiteren wird davon ausgegangen, dass keine Nebenbedingungen verletzt werden. Entsprechend sind die Lagrange-Multiplikatoren Null und die Strafparameter befinden sich an deren unterer Grenze:

$$\begin{aligned} \boldsymbol{\mu}_{k|i}^{\langle 1 \rangle} &= \mathbf{0}, & \nu_{k|i}^{\langle 1 \rangle} &= \nu_{\min}, & i &= 0, 1, \dots, N - 1, \\ \boldsymbol{\mu}_{T,k}^{\langle 1 \rangle} &= \mathbf{0}, & \nu_{T,k}^{\langle 1 \rangle} &= \nu_{\min}. \end{aligned}$$

Die zur Konfiguration des MPC-Lösungsalgorithmus veränderbaren Parameter sind in Tab. 4.3 zusammengefasst.

Tab. 4.3: Parameter des MPC-Lösungsalgorithmus.

| Parameter | Zulässige Werte | Beschreibung |
|--|--|---|
| Allgemeine Parameter | | |
| G | $[1, \infty)$ | Anzahl der Gradienteniterationen |
| M | $[2, N]$ | Anzahl der Stellgrößendiskretisierungspunkte |
| N | $[2, \infty)$ | Anzahl der Diskretisierungspunkte auf dem Prädiktionshorizont |
| T_{int} | $(0, \infty)$ | Integrationsschrittweite |
| T_s | $(0, \infty)$ | Abtastzeit des Reglers |
| $\bar{\tau}$ | $[0, \dots, T] \in \{R^N \mid \tau_i > \tau_{i-1}\}$ | Diskretisierungszeitpunkte |
| Liniensuche | | |
| L | $[1, \infty)$ | Anzahl der Liniensuche-Iterationen |
| c_1 | $[0; 0,5)$ | Steilheitsfaktor der Armijo-Bedingung |
| c_2 | $(0,5; 1]$ | Steilheitsfaktor der Goldstein-Bedingung |
| α_{dec} | $(0, 1)$ | Verkleinerungsfaktor |
| α_{fac} | $(0, 1)$ | Anpassungsfaktor der initialen Schrittw. |
| α_{inc} | $(1, \infty)$ | Vergrößerungsfaktor |
| α_{max} | $(\alpha_{\text{min}}, \infty)$ | Maximalschrittweite |
| α_{min} | $(0, \infty)$ | Mindestschrittweite |
| Lagrange-Multiplikatoren und Strafparameter | | |
| \mathbf{u}_{ini} | \mathbb{R}^m | Initiale Stellgrößen; Kaltstart |
| μ_{max} | $(0, \infty)$ | Lagrange-Multiplikator: Maximalwert |
| β | $(0, \infty)$ | Schrittweite der Anstiegsiteration |
| γ_{inc} | $(1, \infty)$ | Strafparameter: Vergrößerungsfaktor |
| γ_{dec} | $(0, 1)$ | Strafparameter: Verkleinerungsfaktor |
| $\epsilon_{\bar{\mathbf{c}}}$ | $\mathbb{R}_+^{ \mathcal{E} + \mathcal{I} }$ | Toleranzen der integralen Nebenb. |
| $\epsilon_{\bar{\mathbf{c}}_T}$ | $\mathbb{R}_+^{ \mathcal{E}_T + \mathcal{I}_T }$ | Toleranzen der terminalen Nebenb. |
| $\epsilon_{\mathbf{u}}$ | $(0, \infty)$ | Relative Änderung der Stellgrößen |
| ν_{max} | $(\nu_{\text{min}}, \infty)$ | Strafparameter: Maximalwert |
| ν_{min} | $(0, \infty)$ | Strafparameter: Minimalwert |

4.5 Bewertung und Möglichkeiten zur Weiterentwicklung

Eingangs wurden die wesentlichen von dem Regelungsalgorithmus zu erfüllenden Eigenschaften genannt: eine geringe Rechenzeit, ein geringer RAM-Verbrauch sowie ein numerisch robustes Verhalten. Der Algorithmus soll im Folgenden auf diese Eigenschaften hin bewertet werden; bis auf den Aspekt der Rechenzeit, da sich diese ohne den Vergleich zu anderen Methoden nicht sinnvoll bewerten lässt. Eine numerische Bewertung des Algorithmus erfolgt im Rahmen der in Kapitel 5 vorgestellten *temperaturgeführten Motorregelung* anhand der in Abschnitt 5.3.2 evaluierten Reglerkonfigurationen. Auf die Ergebnisse soll an dieser Stelle Bezug genommen werden.

Hinsichtlich des RAM-Verbrauchs können theoretische Untergrenzen angegeben werden. Bezüglich des beanspruchten Speichers auf dem Heap ergibt sich diese aus dem benötigten Speicher für die zu initialisierenden Variablen in Abschnitt 4.4.6, da diese Variablen zwischen den Ausführungen des Algorithmus hinterlegt werden und deshalb über den Programmaufruf hinweg zu speichern sind. Es ergibt sich für die Anzahl der zu speichernden Variablen die Abschätzung

$$\#\text{Heap-Variablen} \geq 3 \cdot Mm + N \cdot (l + 1) + l_T + 1,$$

wobei l die Anzahl der integralen und l_T die Anzahl der terminalen Nebenbedingungen bezeichnen. Übertragen auf die verwendeten Einstellungen des im Rahmen des temperaturgeführten Motors eingesetzten Reglers (vgl. *Konfiguration F* in Abschnitt 5.3.2), lässt sich eine theoretische Untergrenze von 0,34 kB unter der Annahme, dass Fließkommazahlen einfacher Genauigkeit²⁰ verwendet werden, angeben. Bezüglich des beanspruchten Speichers auf dem Stack sind zeitgleich mindestens die Zustände sowie die adjungierten Zustände zu speichern.²¹ Des Weiteren sind für die Auswertung der Ableitungen Matrizen zu evaluieren, insbesondere die Ableitungen des Modells (4.16a) im Rahmen der Auswertung der Hamilton-Funktion (4.14), welche als die größten vorkommenden Matrizen angenommen werden. Es ergibt sich für den Stack die Abschätzung

$$\#\text{Stack-Variablen} \geq 2 \cdot Nn + \max \{n^2, nm\},$$

welche für den eingesetzten Regler auf eine theoretische Untergrenze von 0,176 kB schließen lässt.²² Einen signifikanten Beitrag zur Reduzierung des RAM leistet die Möglichkeit, nicht-äquidistant zu diskretisieren. Durch den Sprung von *Konfiguration B* auf *Konfiguration C* wird der benötigte RAM mehr als halbiert, ohne die Leistungsfähigkeit des Algorithmus maßgeblich zu verschlechtern. Damit ist der Algorithmus ohne die Betrachtung

²⁰Eine Fließkommazahl einfacher Genauigkeit nach dem *IEEE Standard for Floating-Point Arithmetic (IEEE 754)* besteht aus 32 Bit.

²¹Der Begriff *zeitlich* wird verwendet, um die Mehrfachnutzung des reservierten Speichers anzuzeigen.

²²In *Konfiguration F* in Abschnitt 5.3.2 ist ersichtlich, dass die Untergrenze für den Stack um ein Vielfaches übertroffen wird.

implementierungstechnischer Details geeignet, die in Tab. 1.2 dargestellten Anforderungen zu erfüllen.

Im Vergleich der Algorithmuseinstellung ohne Liniensuche in *Konfiguration A* und der Einstellung mit Liniensuche in *Konfiguration B* wird deutlich, dass die Liniensuche als Instrument zur Steigerung der Robustheit geeignet ist (vgl. Abb. B.1 und Abb. B.2). Daneben hat die Liniensuche das Potential, die Rechenzeit zu verkürzen, bzw. die Verbesserung pro investierter Rechenzeit zu erhöhen. Der Vergleich von *Konfiguration C* mit *Konfiguration D* lässt darauf schließen, dass die Vor- und Rückwärtsintegration (vgl. Algorithmus 4.4.2 und Algorithmus 4.4.3) die rechentechnisch aufwendigsten Bestandteile des Algorithmus sind. Die Rückwärtsintegration ist jedoch aufgrund der Auswertung der Ableitung der Hamilton-Funktion rechentechnisch aufwendiger als die Vorwärtsintegration. Pro Iteration der Liniensuche wird eine Vorwärtsintegration durchgeführt, eine vollständige Iteration des GV erfordert jedoch eine Rückwärtsintegration und mindestens eine Vorwärtsintegration. Das bedeutet, dass durch eine Ausweitung der Liniensuche möglicherweise weniger Rückwärtsintegrationen zur Erreichung derselben Verbesserung nötig sind und damit insgesamt Rechenzeit eingespart wird.

Eine Möglichkeit zur Weiterentwicklung greift die unterschiedlichen Kosten der Vor- und Rückwärtsintegration auf: Denkbar ist eine Metrik, die den Fortschritt durch die Liniensuche bewertet, diese vorzeitig abbricht und stattdessen zusätzliche Iterationen des GV durchführt. Dem Algorithmus wäre zu hinterlegen, wie *teuer* jeweils die Vorwärts- und Rückwärtsintegrationen sind. Mit diesen Informationen wäre der Algorithmus dann in der Lage zu entscheiden, ob eine zusätzliche Gradienteniteration durch einen Abbruch der Liniensuche unter gleichzeitiger Einhaltung des Kontingents an verfügbarer Rechenzeit möglich ist.

Die Suchrichtung (4.11) ist ebenfalls ein Ansatzpunkt für eine Weiterentwicklung. Statt in Richtung des negativen Gradienten zu suchen, ist ebenso die Newton-Richtung unter Verwendung eines *BFGS-Verfahrens* [NW06, S. 136ff] denkbar. Allerdings wäre zur Umsetzung dieses Verfahrens eine Matrix zu speichern, deren Dimension der Anzahl der Optimierungsvariablen entspricht. Vor dem Hintergrund des Einsatzgebietes des entwickelten Regelungsalgorithmus würde dieses Verfahren die gewählten Ansätze durch seine erhöhte Speicherbeanspruchung konterkarieren. Als in dieser Hinsicht günstigere Alternative bietet sich ein *konjugiertes Gradientenverfahren* [NW06, S. 121ff] an, welches zwar langsamer als das BFGS-Verfahren konvergiert, dafür aber ohne eine speicherintensive Hesse-Matrix auskommt.

Festzuhalten bleibt, dass der Algorithmus auch ohne die genannten Weiterentwicklungspotentiale zur Lösung dynamischer Optimierungsprobleme mit Nebenbedingungen in robuster Weise geeignet ist.

5 Temperaturgeführte Motorregelung

Der *temperaturgeführte Motor* deckt einen Teilaspekt des in Abschnitt 1.1 eingeführten *emissionsgeführten Motors* ab: Der emissionsgeführte Motor erhält seitens der AGN-Systeme Vorgaben im Hinblick auf die Schadstoffemissionen sowie hinsichtlich der Abgastemperatur. Inhalt dieses Kapitels ist es, den letztgenannten Aspekt mit regelungstechnischen Mitteln sicherzustellen. Es geht dabei nicht darum, die Vorgaben für die Abgastemperatur zu ermitteln, sondern um die motorseitige Berücksichtigung ebenjener. Zur technischen Umsetzung der Regelung kommt der in Kapitel 4 entwickelte MPC-Lösungsalgorithmus zum Einsatz.

Für die Regelung werden als Aktuatoren ein WG und eine ETV verwendet, welche einen Teil der Möglichkeiten zur Beeinflussung der Abgastemperatur darstellen. In Abschnitt 5.1 wird eine Einführung in die Motorregelung gegeben und die Einflüsse der beiden verwendeten Aktuatoren auf den Motor werden besprochen. Im darauffolgenden Abschnitt 5.2 wird das MPC-Optimierungsproblem formuliert, was insbesondere die Formulierung der *integralen Gütefunktion* und der *Nebenbedingungen* beinhaltet. Die integrale Gütefunktion sowie die Nebenbedingungen bestimmen das Verhalten des Reglers. In Abschnitt 5.3 erfolgt die Konfiguration des verwendeten MPC-Lösungsalgorithmus. Diese ist darauf ausgerichtet, eine auf der ECU (vgl. Abb. 1.7) des Deutz TCD 12.0 V6 umsetzbare Einstellung des Lösungsalgorithmus zu erarbeiten, und zwar derart, dass die in Tab. 1.2 genannten Anforderungen erfüllt werden. Validiert wird die temperaturgeführte Motorregelung in Abschnitt 5.4 anhand synthetischer Simulationsbeispiele sowie anhand einer Simulation und einer Prüfstandsaufnahme des NRTC. Im letzten Abschnitt 5.5 erfolgt eine Bewertung der Regelung und es werden Weiterentwicklungspotentiale aufgezeigt.

5.1 Regelungskonzept

Ziel der bereits in Abschnitt 1.2 eingeführten temperaturgeführten Motorregelung ist die Beeinflussung der Abgastemperatur des Motors dahingehend, dass eine vom AGN-System geforderte Mindesttemperatur bereitgestellt wird (vgl. Abb. 1.5). Auf diese Weise stellt der Motor die ordnungsgemäße Funktionsweise der AGN sicher. Das zur Beeinflussung der Abgastemperatur eingesetzte WG sowie die eingesetzte ETV sind dem Luftpfad zuzuordnen (vgl. Abb. 1.6). Die Abgastemperatur kann unabhängig von den beiden genannten Aktuatoren ebenfalls durch eine Manipulation der Verbrennung, beispielsweise durch eine Verschiebung der Kraftstoffeinspritzung bezüglich der Kolbenposition, beeinflusst werden. Auf diesen Eingriff wird jedoch verzichtet, da die Demonstration der technischen Umsetzbarkeit einer MPC auf der ECU im Vordergrund steht. Somit kann diese Umsetzung der temperaturgeführten Motorregelung als *temperaturgeführte Luftpfadregelung* betrachtet werden.

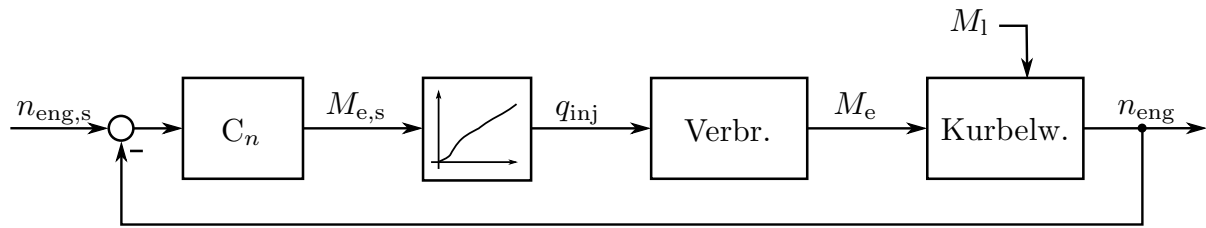


Abb. 5.1: Drehzahlregelung.

Die Regelung der Kraftstoffeinspritzung und der damit verbundenen Drehzahl- bzw. Drehmomentregelung ist ebenfalls nicht Teil der temperaturgeführten Motorregelung. Vor allem die Drehzahlregelung ist für Aspekte der funktionalen Sicherheit besonders relevant, da durch eine fehlerhafte Regelung bzw. ungewollte Drehmomenterzeugung Personenschäden verursacht werden können.

5.1.1 Motorregelung (Übersicht)

Die Kernaufgabe eines Motors ist die Bereitstellung eines (effektiven) Drehmoments M_e , beziehungsweise, im Zusammenspiel mit einer bestimmten Drehzahl n_{eng} , die Bereitstellung einer Leistung

$$P_e = 2\pi n_{\text{eng}} M_e. \quad (5.1)$$

Aus einer regelungstechnischen Perspektive ist je nach Betriebsart des Motors das Drehmoment oder die Drehzahl die Regelgröße; es wird von einem *drehmoment-* oder einem *drehzahlgeregelten Motor* gesprochen. Doch selbst bei einem drehmomentgeregelten Motor wird unter Einbeziehung der Anwendung, in der der Motor verbaut ist, letztendlich eine Drehzahl geregelt, womit sich für beide Fälle der in Abb. 5.1 dargestellte Zusammenhang ergibt.¹ Der Drehzahlregler C_n hat als Stellgröße ein effektives Soll-Drehmoment $M_{e,s}$. Beim drehmomentgeregelten Motor wäre dies der Eingriffspunkt der Führungsgröße. Über ein Kennfeld wird das Soll-Drehmoment in eine Einspritzmenge q_{inj} übersetzt. Dabei wird ausgenutzt, dass das Drehmoment näherungsweise proportional zur Einspritzmenge ist (vgl. Abb. 5.3).² Das durch die Verbrennung erzeugte Drehmoment beschleunigt die Kurbelwelle auf eine Drehzahl n_{eng} . Die Kurbelwellendynamik kann über

$$\frac{dn_{\text{eng}}}{dt} = \frac{1}{2\pi J_{\text{eng}}} (M_e - M_l)$$

¹Der Fahrer, der über die Gaspedalstellung sein Fahrzeug auf eine bestimmte Geschwindigkeit einstellt, variiert die Einspritzmenge q_{inj} . In diesem Fall übernimmt der Fahrer die Aufgabe des Drehzahlreglers.

²Analog zur Proportionalität von Drehmoment und Einspritzmenge ist der Kraftstoffmassenstrom W_f proportional zur effektiven Leistung (vgl. (5.1) und (3.29)).

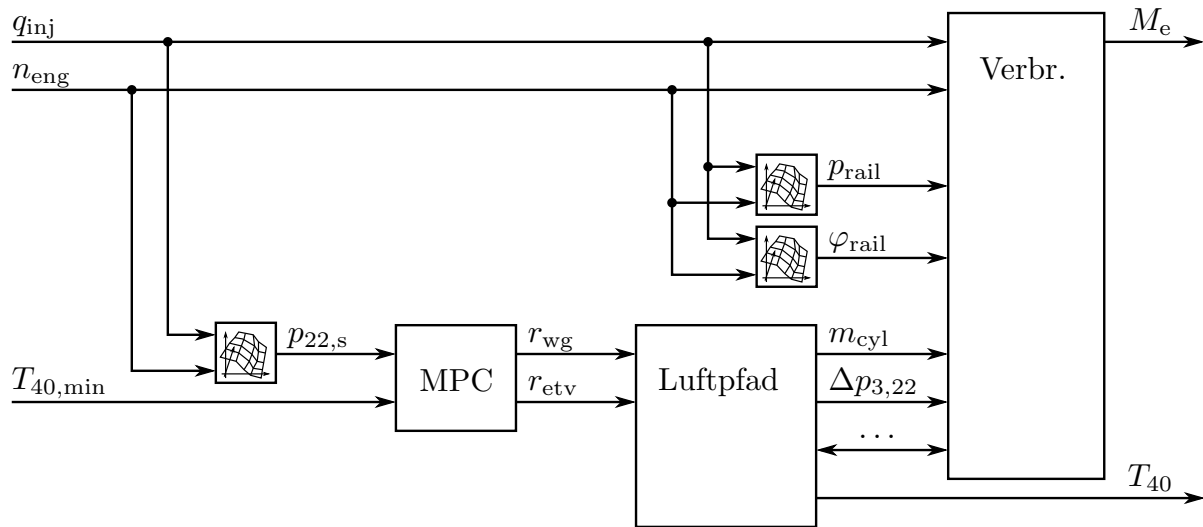


Abb. 5.2: Übersicht der Motorregelung.

unter Verwendung des Trägheitsmoments J_{eng} beschrieben werden. Das Lastmoment M_l berücksichtigt die Effekte des an den Motor angekoppelten Systems, insbesondere auch dessen Trägheit. Im stationären Fall gilt $M_e = M_l$. Es sei angemerkt, dass die Reibungsverluste³ in der Abbildung bereits im Drehmoment M_e enthalten sind.

Bei einer näheren Betrachtung der Drehmomenterzeugung wird deutlich, dass die Vorgabe der Einspritzmenge durch die Drehmoment- / Drehzahlregelung lediglich einen Teilaspekt der Motorregelung darstellt. In Abb. 5.2 werden die übrigen Einflüsse skizziert, inklusive der MPC. Die Darstellung hat keinen Anspruch auf Vollständigkeit; etwaige Rückkopplungen sowie Quereinflüsse werden aus Gründen der Übersichtlichkeit außen vor gelassen. Neben der bereits erwähnten Einspritzmenge sind insbesondere der Raildruck p_{rail} , der Einspritzwinkel φ_{rail} und die Ladeluftmenge m_{cyl} von Bedeutung. Bevor die regelungstechnische Beeinflussung dieser Größen erklärt wird, sollen diese zunächst eingeführt werden.

Der Deutz TCD 12.0 V6 arbeitet nach dem Viertaktprinzip, wobei die Kurbelwelle pro Takt einen Winkel von 180° überstreicht: Im ersten Takt befindet sich der Kolben am oberen Totpunkt – in maximaler Entfernung zur Kurbelwelle – und bewegt sich auf die Kurbelwelle zu. Durch das geöffnete Einlassventil strömt währenddessen Frischluft in den Zylinder. Am Ende des Ansaugtaktes, wenn das Einlassventil geschlossen wird, befindet sich die Ladeluftmenge m_{cyl} im Zylinder. Im zweiten Takt bewegt sich der Kolben zum oberen Totpunkt zurück. Wegen des nun geschlossenen Einlassventils verdichtet sich die Luft, wobei die Temperatur und der Druck steigen. Im dritten Takt erfolgt die wesentliche Arbeit des Motors: Bei Erreichen eines bestimmten Winkels bezüglich des oberen Totpunktes,

³Die Reibungsverluste sind in diesem Kontext ein Sammelbegriff für sämtliche Verluste, die das *innere Moment* des Motors schmälern. Dies ist beispielsweise die Reibung in den Lagern, die Reibung der Kolbenringe oder auch der Leistungsbedarf etwaiger Nebenaggregate.

des Einspritzwinkels, wird die Einspritzmenge unter einem bestimmten Raildruck in den Brennraum gegeben. Der Kraftstoff entzündet sich dort wegen der hohen Temperatur und des hohen Drucks von selbst. Durch die Verbrennung wird der Druck im Brennraum weiter gesteigert, wodurch der Kolben dem unteren Totpunkt entgegen getrieben wird und auf diese Weise die Kurbelwelle antreibt. Im vierten und letzten Takt wird das Abgas durch das nun geöffnete Auslassventil aus dem Brennraum gedrückt. Es sei angemerkt, dass die Einspritzung vereinfacht dargestellt wurde. Die Einspritzung findet tatsächlich nicht zu einem bestimmten Zeitpunkt statt, sondern erstreckt sich über mehrere Grad-Kurbelwinkel. Des Weiteren wird bei modernen Dieselmotoren mehrfach zu unterschiedlichen Zeitpunkten Kraftstoff in den Brennraum gegeben. Dadurch vervielfältigt sich die Zahl der genannten Verbrennungsparameter entsprechend.⁴

Über den Einspritzwinkel und den Raildruck wird der Ablauf der Verbrennung im Sinne der erreichten Spitzendrücke, Spitzentemperaturen und der entstehenden Schadstoffe beeinflusst. Die Ladeluft ist indirekt über den darin enthaltenen Sauerstoff der Reaktionspartner des Kraftstoffs und ist für die Beeinflussung der Rußemissionen sowie insbesondere für die Bereitstellung der Motorleistung von besonderer Bedeutung. Die Einspritzmenge ist durch das Verbrennungsluftverhältnis

$$\lambda = \frac{m_{\text{cyl}}}{\Phi_s q_{\text{inj}}}, \quad \Phi_s := 14,5, \quad (5.2)$$

welches stets $\lambda > 1$ erfüllen muss, begrenzt. Ein Verbrennungsluftverhältnis größer als eins bedeutet, dass im Brennraum mehr Sauerstoff vorhanden ist, als zur stöchiometrischen Umsetzung erforderlich ist. Das stöchiometrische Verhältnis für Dieselkraftstoff ist über Φ_s gegeben. Durch die bei einem Dieselmotor verwendete Direkteinspritzung sind Ladeluft und Kraftstoff inhomogen durchmischt, weshalb in der Praxis eine größere Ladeluftmenge von etwa $\lambda > 1,2$ zur Vermeidung von Sauerstoffmangelzonen⁵ erforderlich ist. Eine unterstöchiometrische Einspritzmenge ist technisch möglich, jedoch verbrennt dann ein gewisser Teil des Kraftstoffs nicht und trägt damit nicht zur Drehmomentenerzeugung bei. Für weitere Informationen sei an dieser Stelle auf [PKS09] verwiesen.

Der Raildruck und der Einspritzwinkel, im Weiteren *Verbrennungsparameter* genannt, werden über ein Kennfeld vorgegeben. Das Kennfeld hängt von der Einspritzmenge und der aktuellen Motordrehzahl ab. Die Ladeluftmenge kann nicht direkt vorgegeben und mit den verfügbaren Sensoren des Motors auch nicht gemessen werden. Sie wird deshalb indirekt über den messbaren Ladedruck p_{22} eingeregelt. Der Ladedruck selbst entspringt, wie die Verbrennungsparameter, einem Kennfeld mit denselben Abhängigkeiten. Die Regelung des Ladedrucks erfolgt im Rahmen dieser Arbeit über eine MPC und ist das *motorinterne* Regelungsziel.

⁴Eine Einführung zur Kraftstoffverbrennung wird in [PKS09, S. 173ff] gegeben.

⁵Sauerstoffmangelzonen sind Bereiche im Brennraum, in denen zu wenig Sauerstoff für eine vollständige Verbrennung des Dieselkraftstoffs vorhanden ist. Diese fördern die Entstehung von Ruß bzw. Rauch. [PKS09, S. 282]

Das *motorexterne* Regelungsziel ist die Einhaltung der Temperaturvorgaben seitens der AGN. Dadurch bedingt ist die Abgastemperatur T_{40} eine weitere Regelgröße. Diese wird derart geregelt, dass sie größer als eine Mindesttemperatur $T_{40,\min}$ ist. In Abb. 5.2 wird suggeriert, dass die Abgastemperatur durch den Luftpfad vorgegeben wird. Durch die Systemübersicht des Luftpfades in Abb. 3.1 wird hingegen deutlich, dass T_{40} maßgeblich von der Temperatur T_{eo} des den Brennraum verlassenden Gases abhängt.

5.1.2 Beeinflussung der Abgastemperatur

Die Abgastemperatur kann motorseitig im Grundsatz auf vier Arten beeinflusst werden: über einen *Betriebspunktwechsel*, über eine *Vertrimmung der Kraftstoffverbrennung*, über eine Anpassung der *Ladeluftmenge* und über eine Veränderung der *Pumpverluste*. In dieser Arbeit werden die letzten beiden Wege verfolgt, deren Wirkung vom Luftpfad (vgl. Abschnitt 3.1.2) ausgeht. Die ersten beiden Wege seien jedoch der Vollständigkeit halber genannt, um eine Perspektive für mögliche Weiterentwicklungen aufzuzeigen. Es sei angemerkt, dass in den weiteren Ausführungen vereinfachte Zusammenhänge herangezogen werden, um eine qualitative Beschreibung zu liefern. Der Einfluss des Luftpfades, insbesondere die Rückkopplungen über den Turbolader, wird nicht beachtet.

Betriebspunktwechsel

Unter dem Begriff *Betriebspunkt* wird eine Kombination aus einer Motordrehzahl und einem effektiven Drehmoment verstanden. Für leistungsgeführte Anwendungen, beispielsweise drehzahlvariable Generatoren zur Stromerzeugung oder Hydraulikanwendungen, ist der tatsächliche Betriebspunkt von untergeordneter Bedeutung. Die effektive Leistung ergibt sich aus der Drehzahl und dem Drehmoment nach (5.1). Somit stehen unterschiedliche Kombinationen von Drehzahlen und Drehmomenten zum Erzielen derselben Leistung zur Verfügung. Jedoch unterscheiden sich diese Kombinationen hinsichtlich der Abgastemperatur. Ohne die Berücksichtigung von Wirkungsgraden wird für dieselbe Leistung dieselbe Menge an Kraftstoff verbrannt und damit dem Abgas dieselbe Wärmeleistung \dot{Q}_f hinzugefügt. Abhängig von der Drehzahl unterscheidet sich nach (3.27) der Luftmassenstrom W_{ei} . Der Einfluss des veränderten Luftmassenstroms auf die Abgastemperatur ist über

$$\Delta T = \frac{\dot{Q}_f}{c_p W_{ei}} \quad (5.3)$$

gegeben. Bei gleicher zugeführter Wärmeleistung besteht ein antiproportionaler Zusammenhang. Veranschaulicht ist dies anhand des zur Kalibrierung des Luftpfadmodells verwendeten DoE (vgl. Abschnitt 3.2.1) in Abb. 5.3. Es ist zu beobachten, dass mit steigendem Drehmoment – entsprechend mit fallender Drehzahl und folglich mit fallendem Luftmassenstrom – die Abgastemperatur steigt.

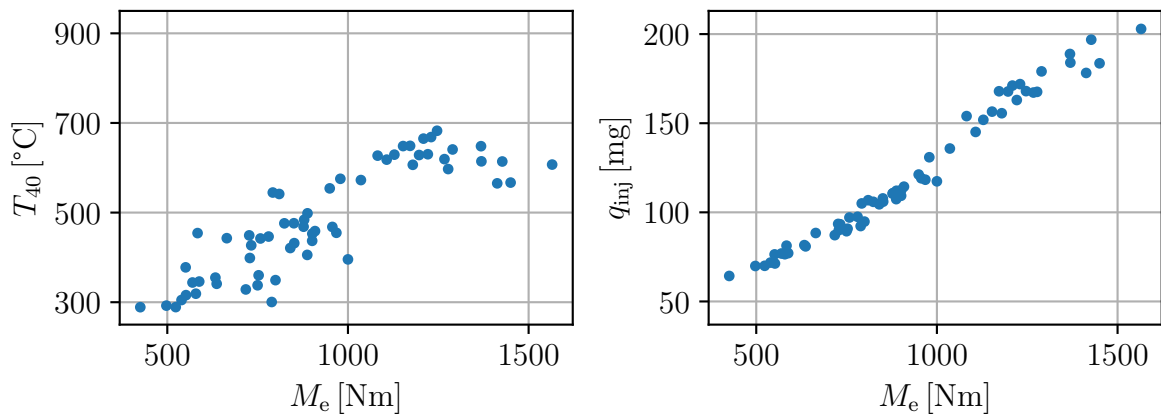


Abb. 5.3: Einfluss des Betriebspunktes auf die Abgastemperatur sowie die Einspritzmenge bei konstanter effektiver Leistung. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $P_e = 100 \pm 10$ kW, $p_0 = 1 \pm 0,1$ bar, $r_{wg} = 0$ %, $r_{etv} < 20$ %.

Vertrimmung der Kraftstoffverbrennung

Die Beeinflussung der Abgastemperatur ist seitens der in den Zylindern stattfindenden Verbrennung über eine Modulation der Einspritzung, namentlich der Einspritzmenge, des Einspritzdrucks oder des Einspritzwinkels, möglich. Durch eine Erhöhung der Einspritzmenge wird die freiwerdende Wärmeenergie im Brennraum gesteigert, was sich in einer höheren Abgastemperatur äußert. Diese Größe eignet sich jedoch nicht zur Regelung der Abgastemperatur, da sie - wie zuvor erwähnt - in erster Linie zur Drehmoment- / Drehzahlregelung verwendet wird und demnach als Stellgröße bereits dafür reserviert ist. Der Einspritzdruck wirkt sich maßgeblich auf die Ruß- und Stickoxidentstehung aus [SGD⁺15] und wird primär zu deren Beeinflussung verwendet, womit er zur Temperaturregelung ebenfalls ausscheidet. Durch eine Verschiebung des Einspritzwinkels kann der Wirkungsgrad des Motors beeinflusst und damit künstlich verschlechtert werden [PKS09, S. 180f]. Ein reduzierter Wirkungsgrad geht zwangsläufig mit einem höheren Kraftstoffverbrauch zum Erzielen desselben Drehmoments einher, was wiederum eine höhere Abgastemperatur zur Folge hat. Der Einspritzwinkel ist dementsprechend die maßgebliche verbrennungsseitige Stellgröße zur Beeinflussung der Abgastemperatur.

Ladeluftmenge

Die luftpfadseitige Beeinflussung der Abgastemperatur ist über zwei Wirkmechanismen möglich. Der eine Wirkmechanismus ist die Reduzierung des Luftmassenstroms W_{ei} bei gleicher Drehzahl, bzw. die Reduzierung der Ladeluftmenge m_{cyl} . Die Ladeluftmenge kann

über

$$m_{\text{cyl}} = \frac{2W_{\text{ei}}}{N_{\text{cyl}} n_{\text{eng}}}, \quad N_{\text{cyl}} := 6 \quad (5.4)$$

aus dem Luftmassenstrom berechnet werden und ist auf diese Weise mit dem modellierten Luftmassenstrom (3.27) verknüpft. Die Temperaturabhängigkeit ist wie bei dem Betriebspunktwechsel über (5.3) gegeben. Entsprechend steigt die Abgastemperatur mit sinkender Ladeluftmenge. In dem oberen linken Diagramm in Abb. 5.4 ist dieser Zusammenhang anhand der DoE-Daten dargestellt. In dem rechten Diagramm wird deutlich, dass die Einspritzmenge näherungsweise konstant bleibt und damit nicht für die Temperaturerhöhung verantwortlich ist. Im Vergleich mit Abb. 5.3 wird ersichtlich, dass über diesen Mechanismus eine weniger starke Temperaturerhöhung als über den Betriebspunktwechsel möglich ist. Der primäre Einflussfaktor auf die Ladeluftmenge ist – wie bereits in Abschnitt 5.1.1 durch die indirekte Regelung der Ladeluftmenge über den Ladedruck angedeutet – über den Ladedruck p_{22} gegeben (vgl. Diagramm unten links in Abb. 5.4).

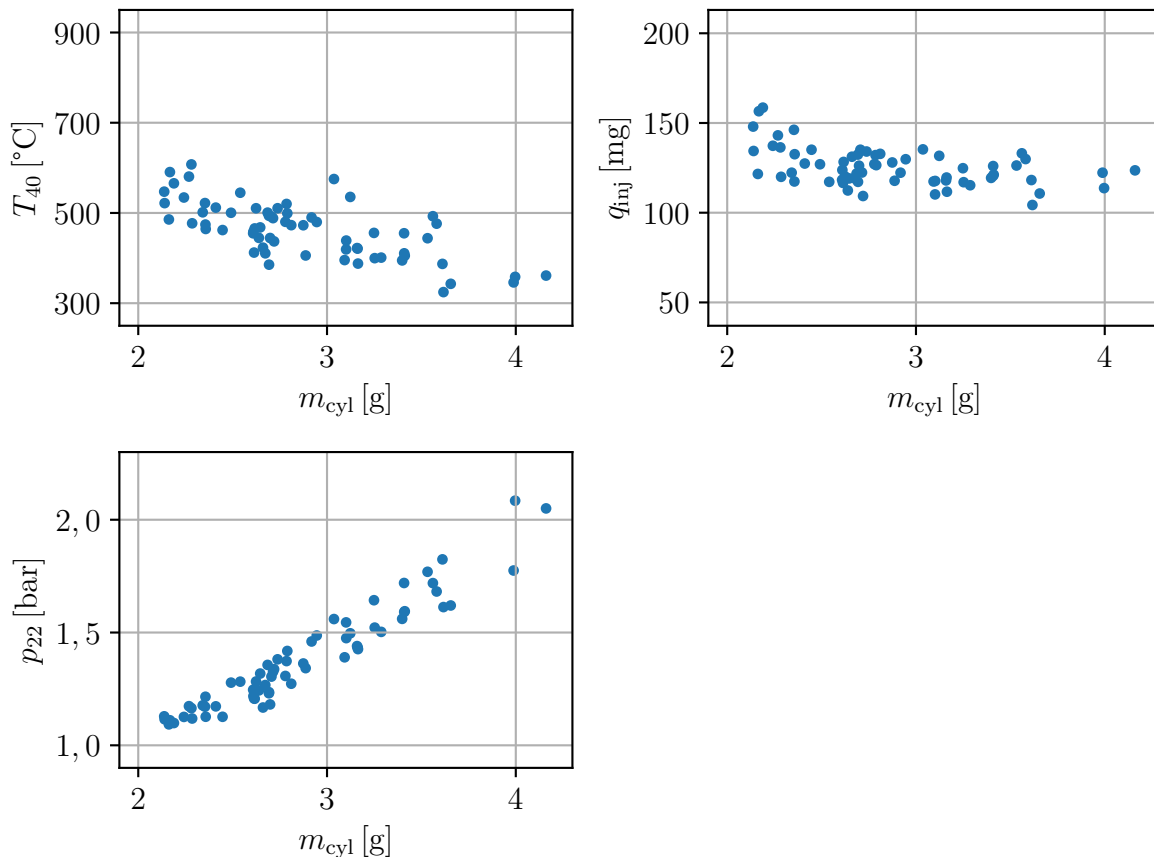


Abb. 5.4: Einfluss der Ladeluftmenge auf die Abgastemperatur, die Einspritzmenge sowie den Ladedruck bei konstantem Drehmoment. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $M_e = 1000 \pm 100$ Nm, $p_0 = 1 \pm 0,1$ bar, $p_3 - p_{22} < 0,15$ bar.

Pumpverluste

Der vierte Wirkmechanismus funktioniert über eine künstliche Erhöhung der Pumpverluste (vgl. Gleichung (3.29)). Steigen die Pumpverluste, ist für dasselbe effektive Drehmoment eine höhere Einspritzmenge erforderlich. Eine höhere Einspritzmenge hat eine höhere Wärmeleistung \dot{Q}_f zur Folge und erhöht über (5.3) die Abgastemperatur. In Abb. 5.5 ist zu sehen, wie mit zunehmendem Differenzdruck $\Delta p_{3,22} := p_3 - p_{22}$ die Einspritzmenge steigt. Das Drehmoment ist im Rahmen der angewendeten Datenfilterung konstant. Mit der Einspritzmengenerhöhung geht eine Temperaturerhöhung einher, was im linken Diagramm zu sehen ist. Im Vergleich mit Abb. 5.4 ist über diesen Wirkmechanismus eine stärkere Temperaturerhöhung zu erreichen.

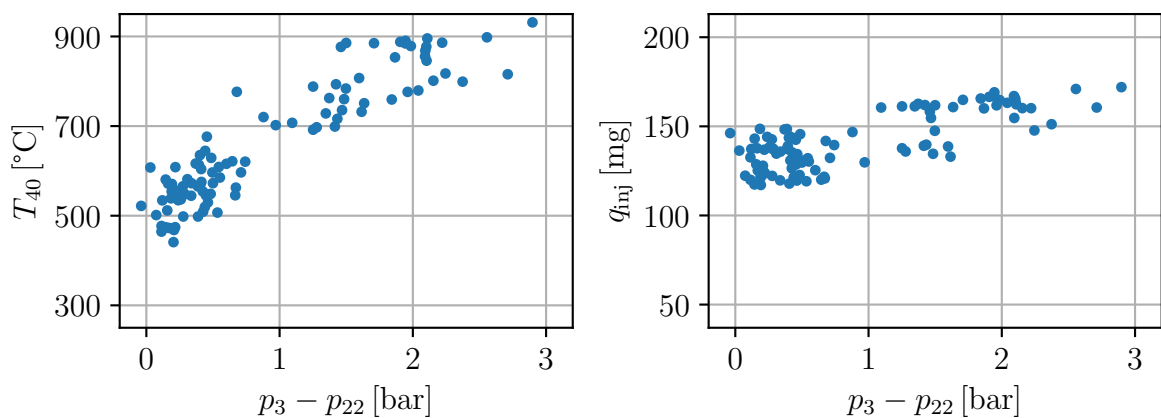


Abb. 5.5: Einfluss des Differenzdrucks zwischen Abgasgegen- und Ladedruck auf die Abgastemperatur bei konstantem Drehmoment. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $M_e = 1000 \pm 100$ Nm, $p_0 = 1 \pm 0,1$ bar, $m_{cyl} = 2,3 \pm 0,1$ g/s.

5.1.3 Einflüsse von Wastegate und Abgasklappe

Es stellt sich die Frage, welcher der beiden Aktuatoren WG und ETV für welche Regelaufgabe geeignet ist. Dazu sollen deren Einflüsse auf den Ladedruck sowie auf den Druck in der Auslassflut betrachtet werden. Der Ladedruck ist einerseits eine direkte Regelgröße, andererseits aber auch über seinen Einfluss auf die Ladeluftmenge für die Abgastemperaturregelung relevant. Der Druck in der Auslassflut spielt für die Pumpverluste eine Rolle und ist damit für die Abgastemperaturregelung ebenfalls von Bedeutung.

In Abb. 5.6 sind die Effekte der Aktuatoren auf die Drücke dargestellt, wobei das WG bei 0% Aktuierung geschlossen und die ETV geöffnet ist. Im linken Diagramm ist zu sehen, wie mit zunehmender Öffnung des WG der Ladedruck sowie der Druck in der Auslassflut abnehmen. Das Wastegate führt einen Teil des Abgasmassenstroms an der Tur-

bine vorbei, wodurch sich zum einen die Auslassflut schneller entleert und deren Druck abfällt (vgl. Gleichung (3.37)) und zum anderen aufgrund der deshalb verringerten Turbinenleistung (vgl. Gleichung (3.41) und Gleichung (3.42)) weniger Ladedruck generiert wird (vgl. Gleichung (3.17) sowie (3.26)). Damit ist das WG zur direkten Beeinflussung des Ladedrucks und darüber zur Beeinflussung der Abgastemperatur über die Ladeluftmenge geeignet. Die Pumpverluste werden über das WG hingegen nicht gesteigert. Der Einfluss der ETV ist im rechten Diagramm zu sehen. Mit zunehmender Schließung der ETV sinkt der Ladedruck aufgrund der ebenfalls verminderten Turbinenleistung ab. Begründet ist dies in dem durch die Schließung hervorgerufenen Druckanstieg hinter der Turbine. Damit eignet sich die ETV ebenfalls zur Ladedruck- und zur Abgastemperaturregelung über die Ladeluftmenge. Der Ladedruck kann in dem betrachteten Betriebspunkt sogar weiter abgesenkt werden als mit dem WG. Im Unterschied zu dem Ergebnis der WG-Öffnung nimmt der Druck in der Auslassflut mit zunehmender Aktuierung der ETV zu. Es ist zu sehen, wie sich p_{22} und p_3 immer weiter voneinander entfernen; der Differenzdruck steigt. Damit können mit der ETV zusätzlich die Pumpverluste beeinflusst werden. Dass bis ca. 50 % Klappenauslenkung kein signifikanter Effekt zu beobachten ist, hängt mit der Nichtlinearität des Aktuators zusammen (vgl. Abb. 3.3).

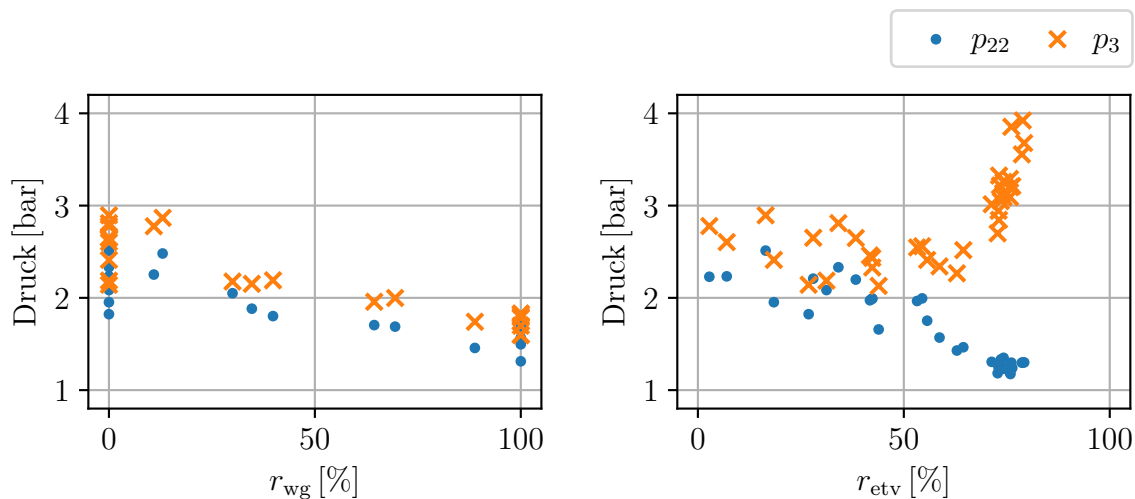


Abb. 5.6: Einfluss der Aktuatorpositionen auf die Drücke. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $p_0 = 1 \pm 0,1$ bar, $p_{41} < 1,3$ bar, $n_{eng} = 1950 \pm 150$ U/min, $M_e = 1000 \pm 100$ Nm.

Bezüglich der Beeinflussung der Abgastemperatur hat die ETV eine höhere Wirkung als das WG. Diese Tatsache liegt darin begründet, dass die ETV beide die Abgastemperatur beeinflussenden Effekte anspricht. Das linke Diagramm in Abb. 5.7 zeigt, wie mit dem Einsatz der ETV eine höhere Temperatur erzielt wird als mit dem WG. Dadurch, dass beide Aktuatoren den Ladedruck signifikant beeinflussen, zeichnet sich eine regelungstechnische Problemstellung ab: Die beiden Regelgrößen weisen eine nicht unerhebliche Abhängigkeit voneinander auf. Es ist damit nicht möglich, gleichzeitig einen hohen Ladedruck und eine hohe Abgastemperatur einzuregeln. Der Zusammenhang ist im rechten Diagramm in

Abb. 5.7 dargestellt. Es ist zu beobachten, dass die Abgastemperatur und der Ladedruck eine hyperbelförmige Abhängigkeit aufweisen. Auf die Regelaufgabe des Luftpfades bezogen bedeutet dies: Wenn eine Mindesttemperatur $T_{40,\min}$ eingehalten werden soll, können die Ladedrücke, deren zugehörige Abgastemperatur unter diesem Wert liegt, nicht eingeregelt werden. Es liegt ein Zielkonflikt vor. Durch die starke Verknüpfung beider Größen kann der Luftpfad im Sinne der Regelgrößen als *überaktuiertes* System aufgefasst werden. Es sind entsprechend mit beiden Aktuatoren – in gewissen Grenzen – die gleichen Regelungsziele zu erreichen.

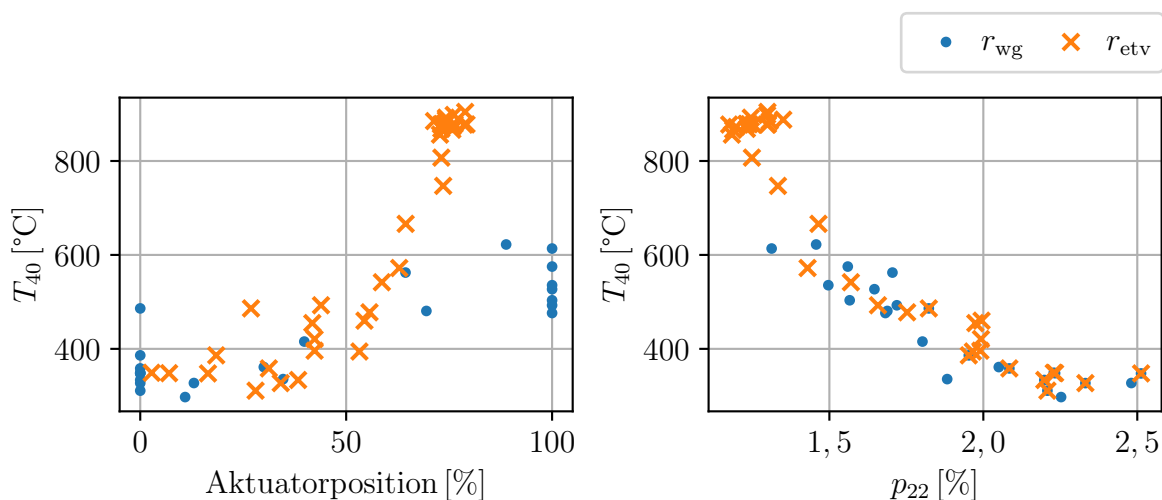


Abb. 5.7: Einfluss der Aktuatorpositionen auf die Abgastemperatur sowie der Zusammenhang zwischen Abgastemperatur und Ladedruck. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $p_0 = 1 \pm 0,1$ bar, $p_{41} < 1,3$ bar, $n_{\text{eng}} = 1950 \pm 150$ U/min, $M_e = 1000 \pm 100$ Nm.

5.1.4 Regelungstechnische Eigenschaften des Luftpfades

Bisher wurden lediglich stationäre Zusammenhänge betrachtet. Zur Regelung eines Systems sind die dynamischen Eigenschaften jedoch ebenso relevant. Diese sollen anhand von Linearisierungen an den DoE-Punkten gemäß (2.4) bewertet werden. Es sei angemerkt, dass sämtliche der folgenden Aussagen einen indikativen Charakter haben und lediglich an den Linearisierungspunkten und in deren näherer Umgebung gelten. Auf eine direkte Betrachtung des nichtlinearen Systems wird aufgrund der hohen Systemkomplexität und der in Teilen nicht-algebraischen Modellierung verzichtet.

Stabilität

Wie bereits in Abschnitt 3.3.3 zur Beurteilung der Abtastrate für eine numerische Integration gezeigt wurde, besitzt die Systemmatrix $\mathbf{A}(\mathbf{x}_R, \mathbf{u}_R)$ für keine Linearisierung aus $\mathbf{x}_R, \mathbf{u}_R \in \mathcal{DOE}$ – der Menge der DoE-Punkte – Eigenwerte mit positivem Realteil. Damit sind sämtliche Linearisierungen stabil.⁶

Steuerbarkeit

Satz 5.1.1 (Steuerbarkeitskriterium von Hautus). *Das System (\mathbf{A}, \mathbf{B}) ist genau dann vollständig steuerbar, wenn die Bedingung*

$$\text{Rang} \left[\lambda_i \mathbf{I} - \mathbf{A} \mid \mathbf{B} \right] = n$$

für alle Eigenwerte λ_i , ($i = 1, 2, \dots, n$) der Systemmatrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ erfüllt ist [Lun13, S. 81].

Für sämtliche Linearisierungen ist das *Kriterium von Hautus* (vgl. Satz 5.1.1) erfüllt. Damit ist das System an diesen Punkten steuerbar.⁷ Ohne die Hilfspolynome (3.40c) sowie (3.50b) wäre das Systemmodell für kleine Aktuierungen von r_{wg} und r_{etv} aufgrund der verschwindenden Aktuatoreinflüsse nicht steuerbar.

Dynamisches Verhalten

Für eine qualitative Beurteilung der Systemdynamik wird der Luftpfad als MIMO-System (engl.: *Multiple Input Multiple Output*) mit den beiden Aktuatoren als Eingängen und den beiden Regelgrößen als Ausgängen betrachtet. Die Übertragungsfunktionsmatrix des Systems ist über

$$\mathbf{G}(s) = \begin{bmatrix} G_{p22, r_{wg}}(s) & G_{p22, r_{etv}}(s) \\ G_{T40, r_{wg}}(s) & G_{T40, r_{etv}}(s) \end{bmatrix} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (5.5)$$

durch eine Linearisierung gegeben. In Abb. 5.8 sind die Bode-Diagramme der vier Übertragungsfunktionen des Luftpfades aufgetragen. Die blau hinterlegten Bereiche bilden den Wertebereich sämtlicher aus den DoE-Punkten gewonnenen Übertragungsfunktionen ab. Farbig dargestellt ist eine Auswahl an Linearisierungen.

⁶Die Stabilität nichtlinearer Systeme wird korrekterweise unter Verwendung der Stabilitätstheorie von *Ljapunov* beurteilt [Ada14, S. 83].

⁷Die Steuerbarkeit des nichtlinearen Systems kann über den Nachweis der *Flachheit* des Systems gezeigt werden [Ada14, S. 187ff].

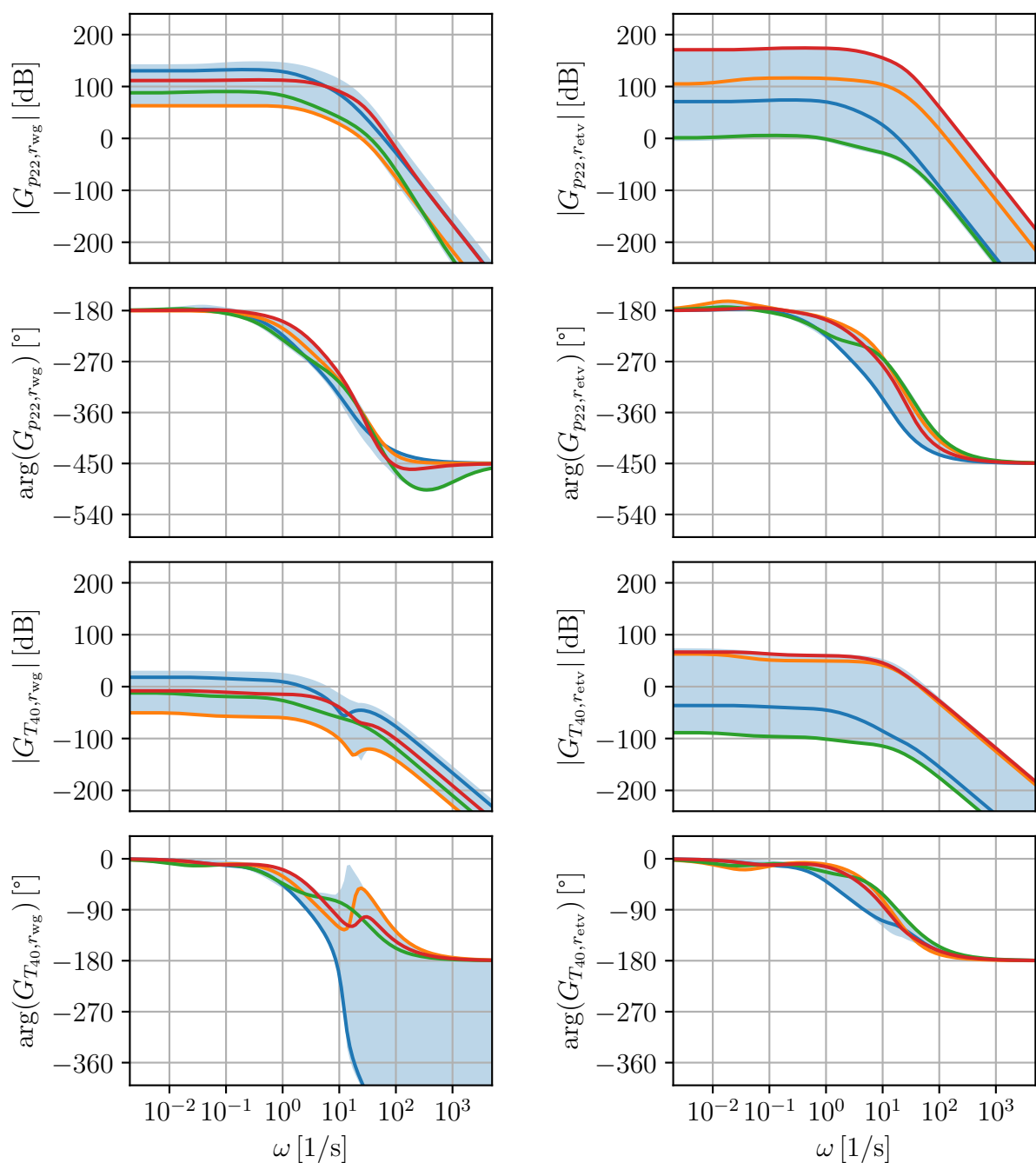


Abb. 5.8: Bode-Diagramm der Übertragungsfunktionsmatrix.

Die Amplitudengänge sollen näher betrachtet werden. Auffällig ist, dass die einzelnen Übertragungsfunktionen um Größenordnungen unterschiedliche Verstärkungen aufweisen. Dies ist eine direkte Folge der Nichtlinearitäten des WG (vgl. Abb. 3.2) und der ETV (vgl. Abb. 3.3). Die ETV zeigt eine exponentielle Abhängigkeit bezüglich der Aktuatorposition, was in der im Vergleich zum WG größeren Spreizung des Amplitudenganges resultiert. Ohne die Hilfspolynome (3.40c) und (3.50b) wären die blauen Bereiche der Amplitudengänge

nach unten unbegrenzt. Jedoch gehen die unterschiedlichen Verstärkungen nicht ausschließlich auf die Aktuatornichtlinearitäten zurück. Ebenso spielt der Arbeitspunkt des Motors eine Rolle – die Effekte sind in der Abbildung nicht unterscheidbar. Wie bereits aus der Untersuchung der Aktuatoreinflüsse im vorigen Abschnitt bekannt ist, weisen sämtliche Übertragungsfunktionen einen nennenswerten Einfluss auf die zugehörige Regelgröße auf. Damit bestätigt sich, dass sich mit jeweils beiden Aktuatoren beide Regelgrößen beeinflussen lassen. Ebenso wird bestätigt, dass mit der ETV stärkere Eingriffe möglich sind; die maximale Verstärkung ist größer. Auffällig ist, dass die Bandbreite der ETV in allen Übertragungsfunktionen etwa eine Größenordnung über der des WG liegt – mit der ETV wird sowohl bezüglich des Ladedrucks als auch bezüglich der Abgastemperatur schneller eine Wirkung erzielt. Physikalisch lässt sich dies über die unmittelbare Wirkung der ETV auf die Turbinenleistung begründen. Beim WG ist der Wirkungspfad länger: Das WG senkt – wie zuvor erwähnt – den Druck p_3 ab, was zu einer geringeren Turbinenleistung und in weiterer Folge zu einem geringeren Ladedruck führt. Dieser hat seinerseits einen geringeren Frischluftmassenstrom zur Folge (vgl. Gleichung (3.27)). Letzterer senkt p_3 weiter herab und verstärkt den Effekt des WG (vgl. Gleichung (3.37)). Bezüglich der Kontur der Übertragungsfunktionen sticht $G_{T_{40}, r_{wg}}$ heraus. Dort ist eine Anhebung der Verstärkung zwischen $10^1 \text{ rad/s} < \omega < 10^2 \text{ rad/s}$ zu beobachten (vgl. orangene und blaue Kurve). Ursächlich dafür sind komplexe Nullstellen.⁸

Bei der Betrachtung des zugehörigen Phasenganges fällt auf, dass dieser im Vergleich zu den anderen Phasengängen den größten Wertebereich abdeckt. Dies beruht auf dem Vorhandensein von Nullstellen mit positivem Realteil einiger Linearisierungen. Bei einem Vergleich der orangenen und blauen Kurve wird ersichtlich, dass die Amplitudengänge eine ähnliche Kontur aufweisen, die Phasengänge jedoch nicht. Beide Kurven weisen komplexe Nullstellen auf, wenngleich die der blauen Kurve positiv sind. In diesem Fall wird die Phase nicht angehoben, sondern weiter abgesenkt. Die Phasenverläufe der übrigen Übertragungsfunktionen enthalten ebenfalls Nullstellen, jedoch keine positiven; dies gilt für sämtliche DoE-Punkte. Damit ist der Zusammenhang zwischen WG und Abgastemperatur der einzige nicht-minimalphasige Zusammenhang. Sollte einzig das WG zur Temperaturregelung verwendet werden, wäre somit die maximale Bandbreite des Reglers durch die kleinste positive Nullstelle begrenzt [SP07, S. 183ff].

Definition 5.1.1 (Übertragungsnullstelle). Die Übertragungsnullstellen des Systems $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ sind Frequenzen s_0 , für die die symmetrische Übertragungsfunktionsmatrix $\mathbf{G}(s)$ eine der folgenden Bedingungen erfüllt [Lun13, S. 49]:

$$\begin{aligned} \det \mathbf{G}(s_0) &= 0, \\ \text{Rang } \mathbf{G}(s_0) &< \max_s \text{Rang } \mathbf{G}(s). \end{aligned}$$

Das gesamte System weist in über 99,5% der Linearisierungen Übertragungsnullstellen (vgl. Definition 5.1.1) mit positivem Realteil auf. Somit kann das System qualitativ als nicht-minimalphasig betrachtet werden. Die für die maximale Bandbreite eines Reglers

⁸Komplexe Nullstellen treten stets paarweise auf [SP07, S. 183].

interessante kleinste positive Nullstelle liegt im Wertebereich (9,633; 5420,439). Die untere Grenze des Wertebereichs wird für kleine Auslenkungen der ETV erreicht. Um in einer praktischen Umsetzung etwaige Reglerbeschränkungen zu entschärfen, ist daher eine Begrenzung der ETV-Position auf genügend große Auslenkungen ratsam. In Abb. 3.3 ist ersichtlich, dass die ETV erst ab ca. 50 % Auslenkung wirksam wird. Entsprechend hat eine Begrenzung der ETV-Position auf den Wertebereich (50 %; 100 %) keine nennenswerten Nachteile. Allerdings ist nach [SP07, S. 235ff] die Beschränkung durch Übertragungsnullstellen bei einem MIMO-System nicht so strikt wie bei einem SISO-System (engl.: *Single Input Single Output*), da die Nullstellen an eine bestimmte Richtung des Eingangsvektors \mathbf{u} gebunden sind. Auf eine tiefergehende Betrachtung wird an dieser Stelle verzichtet.

Wenn das Regelungsziel entweder die Regelung des Ladedrucks oder die Regelung der Abgastemperatur ist, dann ist das System minimalphasig und aus systemtheoretischer Sicht beliebig schnell zu regeln. Keine der Zeilen der Übertragungsfunktionsmatrix verliert ihren Rang. In der Praxis sind der realisierbaren Regelungsgeschwindigkeit Grenzen gesetzt beispielsweise durch die Leistungsfähigkeit der Aktuatoren oder auch der verwendeten Abtastzeit des Reglers. Unter Verwendung eines modellbasierten Regelungsansatzes, wie in dieser Arbeit, führen die gewählten Modellierungsansätze (vgl. Abschnitt 3.1.1) zu Fehlern in den hohen Dynamikanteilen der Systembeschreibung, wodurch der Regler in diesen Bereichen ebenfalls an Güte verliert.⁹

Abseits der Nullstellen ist auffällig, dass die Phasengänge von $G_{p_{22},*}$ im Gegensatz zu denen von $G_{T_{40},*}$ bei -180° Phasenwinkel beginnen. Dieser Winkel ist durch die negierte Wirkrichtung begründet: Durch das Öffnen des WG ($r_{wg} \uparrow$) sowie durch das Schließen der ETV ($r_{etv} \uparrow$) wird der Ladedruck gesenkt ($p_{22} \downarrow$), wohingegen die Abgastemperatur steigt ($T_{40} \uparrow$).

Sollte ein dezentraler Regler – das WG zur Regelung des Ladedrucks und die ETV zur Regelung der Abgastemperatur – angestrebt werden, kann für den gesamten Arbeitsbereich der gleiche Regler verwendet werden. Lediglich die Verstärkungen wären arbeitspunktabhängig zu adaptieren. Dies ist möglich, weil die Phasengänge einen kleinen Wertebereich aufweisen und damit näherungsweise gleich sind und die Amplitudengänge lediglich zueinander verschoben sind. Durch die Kopplung wären die Regler jedoch derart zu entwerfen, dass einer deutlich schneller ist als der andere.

5.2 Formulierung des Optimierungsproblems

Für die Regelung wird der in Kapitel 4 entwickelte MPC-Lösungsalgorithmus unter Verwendung des in Kapitel 3 beschriebenen Luftpfades verwendet. Inhalt dieses Abschnitts ist die Formulierung der in (4.1) geforderten Funktionen.

⁹Es werden insbesondere die Transportprozesse sowie die durch das Arbeitsspiel des Motors bedingten Pulsationen der Massenströme, der Drücke und der Temperaturen vernachlässigt.

5.2.1 Regelungstechnische Anforderungen

Die obersten Ziele der Regelung sind die Einregelung des Ladedruck-Sollwertes $p_{22,s}$ bei gleichzeitiger Einhaltung der Mindesttemperatur $T_{40,\min}$. Wie im rechten Diagramm in Abb. 5.7 ersichtlich ist, existiert lediglich ein schmales Band, in dessen Bereich der Ladedruck unabhängig von der Abgastemperatur geregelt werden kann. Es ist daher davon auszugehen, dass ein Zielkonflikt zwischen den beiden Regelgrößen bestehen wird. Entsprechend ist ein Mechanismus gefordert, der diesen Zielkonflikt auflöst.

Abseits der reinen Regelungsziele gibt es Nebenbedingungen, die eingehalten werden müssen. Zum Schutz des Turboladers darf dessen Drehzahl einen Wert von $n_{tc,\max} := 100 \text{ kU/min}$ nicht überschreiten, andernfalls droht eine mechanische Beschädigung bzw. eine Verkürzung der Lebenserwartung. Neben der Drehzahl wirkt sich eine zu hohe Abgastemperatur negativ auf den Turbolader aus. Diese darf nicht höher als $T_{3,\max} := 700 \text{ °C}$ werden.¹⁰ Auf den Motor an sich wirkt sich ein zu hoher Abgasgedruck negativ aus. Aus mechanischen Gründen darf dieser einen Grenzwert von $p_{3,\max} := 4 \text{ bar}$ nicht überschreiten. Die Dynamik des Motors wird in erster Linie durch zu wenig verfügbaren Sauerstoff eingeschränkt. Aus diesem Grund muss das Verbrennungsluftverhältnis (5.2) stets oberhalb eines Wertes von $\lambda_{\min} := 1,2$ liegen, um zumindest eine geringe Drehmomentsteigerung bei aktiver Nebenbedingung zu ermöglichen. Selbst bei korrekt gewähltem Ladedruck-Sollwert ist es denkbar, dass während schneller Betriebspunktwechsel das Verbrennungsluftverhältnis zu klein wird. Der Hauptgrund, weswegen mit einem Unterschreiten des kritischen Wertes zu rechnen ist, ist allerdings die Auflösung des Zielkonfliktes zwischen den beiden Regelgrößen. Der Zusammenhang ist derart gegeben, dass im Falle eines Konfliktes der Ladedruck zugunsten einer höheren Abgastemperatur reduziert werden kann.

In Abschnitt 5.1.3 wurde gezeigt, dass bei Aktuierung der ETV Pumpverluste entstehen. Diese haben eine Erhöhung des Kraftstoffverbrauchs zur Folge (vgl. rechtes Diagramm in Abb. 5.5). Wenn mehrere Kombinationen von Aktuatorstellungen die Regelungsaufgabe erfüllen, sind die Stellungen mit minimaler ETV-Aktuierung aufgrund der dann höheren Kraftstoffeffizienz zu bevorzugen, im Sinne eines optimalen Kraftstoffverbrauchs.

Neben diesen quantifizierbaren Zielen soll die Regelung hinsichtlich ihrer Geschwindigkeit einen Kompromiss aus höchstmöglicher Motordynamik und gleichzeitigem Schutz der Aktuatoren vor zu hohen Stellgeschwindigkeiten finden.

Die verwendeten Parameter sind am Ende des Abschnitts in Tab. 5.2 zusammengefasst.

¹⁰Laut Datenblatt des verbauten Turboladers ist eine kurzzeitige Überschreitung des Temperatur-Grenzwertes um 15 % gestattet (vgl. Tab. D.1). Regelungstechnisch wird diesem Umstand keine Rechnung getragen.

5.2.2 Endlicher Horizont ohne Endbedingung

Wie in der Einführung von Kapitel 4 bereits erwähnt wurde, sind in der Literatur vier grundlegende MPC-Formulierungen gebräuchlich: *Endlicher Horizont mit Endbeschränkung*, *Endlicher Horizont mit Endregion*, *Endlicher Horizont mit Endkostengewichtung* und *Endlicher Horizont ohne Endbedingung*. Es soll die letzte Formulierung motiviert werden.

Die Formulierung *Endlicher Horizont mit Endbeschränkung* fordert $\mathbf{x}(t_k + T) = \mathbf{x}_T$; es soll ein bestimmter Endzustand exakt erreicht werden. In einer dynamischen Regelung mit unterschiedlichen, sich zeitlich verändernden Sollwerten ist die Bestimmung der Zielwerte am Ende des Horizonts nicht ohne zusätzlichen Aufwand möglich. Übertragen auf den Luftpfad wäre über den einzuregelnden Sollwert für den Ladedruck $p_{22,s}$ lediglich einer der acht Zielzustände bekannt. Für die Abgastemperatur gilt lediglich ein Minimalwert. Die übrigen Zielzustände bzw. erreichbaren stationären Endwerte wären in einer separaten Berechnung zu ermitteln. Aus optimierungstechnischer Sicht ist die exakte Einhaltung von Nebenbedingungen mit einem erhöhten Rechenaufwand verbunden. Um Stabilität zu gewährleisten, ist dies jedoch zwingend erforderlich [MH99]. Aus diesem Grund wird diese Formulierung nicht weiter verfolgt.

Die Formulierungen *Endlicher Horizont mit Endregion* und *Endlicher Horizont mit Endkostengewichtung* werfen die Fragestellung auf, wie die Endkostengewichtung $V(\cdot)$ in (4.1a) zu bestimmen ist.¹¹ Die MPC arbeitet aus Gründen der rechentechnischen Realisierbarkeit mit einem endlichen Prädiktionshorizont T . Die Endkostengewichtung stellt eine Abschätzung des vernachlässigten Teils der Gütefunktion mit unendlichem Prädiktionshorizont dar:

$$J_{\infty}^*(\mathbf{x}_k) := \int_{t_k}^{\infty} l(\mathbf{x}^*(\tau), \mathbf{q}(\mathbf{x}^*(\tau)), \tau) d\tau \approx V(\mathbf{x}^*(t_k + T)) + \int_{t_k}^{t_k + T} l(\mathbf{x}^*(\tau), \mathbf{q}(\mathbf{x}^*(\tau)), \tau) d\tau \quad (5.6)$$

mit dem Regelungsgesetz

$$\mathbf{u}(t) = \mathbf{q}(\mathbf{x}(t)).$$

Das Regelungsgesetz ergibt sich durch den MPC-Lösungsalgorithmus.

¹¹Beim Ansatz *Endlicher Horizont mit Endbeschränkung* werden keine Endkosten verwendet; $\mathbf{V}(\cdot) = 0$.

Definition 5.2.1 (Control-Ljapunov-Funktion(CLF)). Gegeben sei das System

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x} \in \mathbb{R}^n, \quad \mathbf{u} \in \mathbb{R}^m$$

mit der Ruhelage $\mathbf{x}_R = \mathbf{0}$ für $\mathbf{u} = \mathbf{0}$. Dann heißt eine stetig differenzierbare Funktion $V(\mathbf{x})$ CLF, wenn sie die folgenden Bedingungen erfüllt:

$$\begin{aligned} V(\mathbf{0}) &= 0, \\ V(\mathbf{x}) &> 0, \quad \forall \mathbf{x} \neq \mathbf{0}, \\ V(\mathbf{x}) &\rightarrow \infty, \quad \forall |\mathbf{x}| \rightarrow \infty. \end{aligned}$$

Es gibt ein $\mathbf{u}(\mathbf{x})$, so dass $\dot{V}(\mathbf{x}) < 0 \forall \mathbf{x} \neq \mathbf{0}$ gilt.

[Ada14, S. 389]

Eine Voraussetzung für Stabilität ist, dass $J_\infty(\mathbf{x}_k)$ eine CLF (vgl. Definition 5.2.1) ist [GP17, S. 79]. Um dies für die rechte Seite von (5.6) bzw. für (4.1a) sicherzustellen, besteht eine Möglichkeit darin, die Endkosten $V(\cdot)$ als CLF zu wählen und das Reglergesetz derart zu entwerfen, dass die gesamte rechte Seite ebenfalls eine CLF ist [JYH01]. Diesen Ansatz verfolgen die technisch realisierbaren Umsetzungen [MM90, CA98]. Beide Ansätze wählen als Lösungskandidaten die Ljapunov-Funktion

$$V(\mathbf{x}) := \mathbf{x}^T \mathbf{P} \mathbf{x}, \quad \mathbf{P} \succ 0, \quad \mathbf{P}^T = \mathbf{P}.$$

Unter Verwendung einer lokalen linearen Zustandsrückführung \mathbf{K}_T für den Arbeitspunkt $(\mathbf{x}_T, \mathbf{u}_T)$, wird die Matrix \mathbf{P} durch Lösen der Ljapunov-Gleichung

$$\mathbf{A}_T^T \mathbf{P} + \mathbf{P} \mathbf{A}_T = - \left. \frac{\partial^2 l(\mathbf{x}, \mathbf{K}_T \mathbf{x}, t_k + T)}{\partial^2 \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_T}$$

ermittelt.¹² Die Matrix \mathbf{A}_T ist die aus einer Linearisierung an der Stelle $(\mathbf{x}_T, \mathbf{u}_T)$ hervorgehende Systemmatrix (vgl. Gleichung (2.4)). Als Voraussetzung für die Berechnung muss die rechte Seite symmetrisch positiv definit sein. Für das quadratische Gütefunktional $l(\cdot)$ (vgl. (2.38)) ist dies der Fall. In [HRFA06] wird basierend auf [CA98] eine MPC für Dieselmotoren mit AGR entworfen.¹³ Die Autoren bemerken, dass das Verfahren für eine praktische Implementierung mit der verfügbaren Rechentechnik (Stand: 2006) nicht umsetzbar sei. Mit den beiden Formulierungen wird zwar die numerische Lösung durch das Fehlen einer Endbeschränkung vereinfacht, jedoch bleibt das grundlegende Problem der Ermittlung der Endzustände \mathbf{x}_T erhalten. Erschwerend kommt die gesonderte Berechnung eines lokalen Reglergesetzes sowie die Lösung der Ljapunov-Gleichung hinzu, insbesondere derart, dass für sämtliche Situationen eine numerische Lösung mit akzeptabler numerischer

¹²In [CA98] wird eine leicht abgewandelte Ljapunov-Gleichung gelöst, um sicherzustellen, dass \mathbf{P} positiv definit ist.

¹³In [HRFA06] wird mit einem Modell bestehend aus den drei Zuständen p_{22} , p_3 sowie der Kompressorleistung P_c gearbeitet. Die Effizienz des Kompressors (3.16b) sowie die Effizienz der Turbine (3.43) werden als konstant angenommen.

Konditionierung gefunden wird.

Aufgrund der beschriebenen Nachteile der übrigen Formulierungen wird im Folgenden die MPC-Formulierung *Endlicher Horizont ohne Endbedingung* verwendet. Das zu lösende Optimierungsproblem hat damit die Form (vgl. Gleichung (4.1))

$$\min_{\mathbf{u}(\cdot)} \quad J(\mathbf{u}(\cdot); \mathbf{x}_k, t_k) = \int_{t_k}^{t_k+T} l(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) d\tau \quad (5.7a)$$

$$\text{u. B. v.} \quad \dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)), \quad \mathbf{x}(t_k) = \mathbf{x}_k, \quad (5.7b)$$

$$\mathbf{b}_l \leq \mathbf{u}(\tau) \leq \mathbf{b}_u, \quad (5.7c)$$

$$\mathbf{r}_l \leq \frac{d\mathbf{u}(\tau)}{dt} \leq \mathbf{r}_u, \quad (5.7d)$$

$$\mathbf{c}_{\mathcal{E}}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) = 0, \quad (5.7e)$$

$$\mathbf{c}_{\mathcal{I}}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) \leq 0. \quad (5.7f)$$

Diese Formulierung ist ebenfalls mit Nachteilen behaftet. So erfordert das Fehlen der Endkosten eine gesonderte Ermittlung der Länge des Prädiktionshorizonts, um Stabilität zu gewährleisten. Es existieren jedoch Abschätzungen für dessen erforderliche Länge [GP17, S. 135ff]. In der Praxis ist die Länge des Horizonts über numerische Experimente zu bestimmen.

5.2.3 Skalierung

Bevor in den folgenden Abschnitten das MPC-Problem (5.7) präzisiert wird, sollen zunächst geeignete Skalierungen eingeführt werden. Die Wertebereiche der einzelnen Größen unterscheiden sich um Größenordnungen (vgl. Tab. 3.1). Bei der Definition eines Gütefunktional und der zugehörigen Gewichtungen wären diese Größenunterschiede entsprechend zu berücksichtigen. Daneben wirken sich stark unterschiedliche Größenordnungen negativ auf die numerische Konditionierung aus. Über die Rechenvorschrift

$$\tilde{d} = \frac{d - d_{\text{Offset}}}{d_{\text{Faktor}}}$$

werden die Größen auf einen einheitlichen Wertebereich normiert. Die Skalierungen $(d_{\text{Faktor}}, d_{\text{Offset}})$ werden derart gewählt, dass idealerweise $\tilde{d} \in [-1, 1]$ gilt. Die gewählten Skalierungen sind in Tab. 5.1 aufgeführt, wobei der regelungstechnisch als relevant angesehene Wertebereich berücksichtigt wird. Um den Parameterumfang einer rechentechnischen Umsetzung nicht über Gebühr zu erhöhen, werden die Drücke sowie die Temperaturen einheitlich skaliert.¹⁴ Aus Übersichtlichkeitsgründen werden die Skalierungen im Weiteren formeltechnisch nicht kenntlich gemacht.

¹⁴Eine weitere Strategie zur Skalierung besteht darin, die maximal zu erwartenden Fehler zwischen Soll- und Istwert als Maßgabe für eine Skalierung heranzuziehen.

Tab. 5.1: Verwendete Skalierungen.

| Größe | Relevanter Wertebereich | d_{Faktor} | d_{Offset} |
|-------------------|-------------------------|---------------------|---------------------|
| n_{tc} [kU/min] | [30; 100] | 35,00 | 65,00 |
| p_* [bar] | [1,00; 4] | 1,65 | 2,35 |
| r_* [%] | [0,00; 100] | 50,00 | 50,00 |
| T_* [K] | [373,14; 973,14] | 300,00 | 673,14 |
| λ [-] | [1; 6] | 2,50 | 3,50 |

5.2.4 Integrales Gütefunktional

Das integrale Gütefunktional $l(\cdot)$ bestimmt das Verhalten des eigentlichen Reglers, ohne Berücksichtigung etwaiger Nebenbedingungen. Ausgangspunkt für dessen Definition ist ein linear-quadratischer Ansatz nach (2.38). Es wird die Formulierung

$$l(\mathbf{x}(t), \mathbf{u}(t), t) := q_p \alpha^2 \Delta p_{22}^2 + q_T (1 - \alpha)^2 \Delta T_{40}^2 + q_{\dot{\zeta}_{wg}} \dot{\zeta}_{wg}^2 + q_{\dot{\zeta}_{etv}} \dot{\zeta}_{etv}^2 \quad (5.8a)$$

mit

$$\Delta p_{22} = p_{22} - p_{22,s}, \quad (5.8b)$$

$$\Delta T_{40} = \min \{T_{40} - T_{40,\min}, 0\} \quad (5.8c)$$

und $\alpha \in [0, 1]$ verwendet. Aus Gründen der Übersichtlichkeit wird die Abhängigkeit zur Zeit weggelassen; dies gilt ebenso für die weiteren Ausführungen. Über q_p sowie q_T werden die Regelgrößen gewichtet. Mit $q_{\dot{\zeta}_{wg}}$ und $q_{\dot{\zeta}_{etv}}$ wird die Änderungsrate der Aktuatoren bestraft; der Regler wird durch eine Erhöhung langsamer. Durch die Berücksichtigung der Änderungsraten, welche über (3.38) und (3.48) gegeben sind, werden die zu optimierenden Stellgrößen in die Gütefunktion eingebracht. Dies ist für die verwendete Formulierung ohne Endbedingung (5.7) zwingend erforderlich, da ansonsten der Gradient der Hamiltonfunktion (4.6) am Ende des Prädiktionshorizonts dem Nullvektor entspricht und die Gradientenberechnung (4.15) für die letzte Diskretisierung nicht sinnvoll möglich ist; die korrespondierende Stellgröße wird stets auf dem Anfangswert verharren.¹⁵ Der Mechanismus zum Auflösen des Zielkonfliktes wird über einen nichtlinearen Faktor

$$\alpha = \frac{\gamma \Delta p_{22} + 10^{-9}}{\gamma \Delta p_{22} + (1 - \gamma) \Delta T_{40} + 10^{-9}}$$

geliefert. In (5.8a) ist α als Interpolationsfaktor anzusehen. Für $\alpha = 1$ wird ausschließlich der Druck geregelt und für $\alpha = 0$ ausschließlich die Temperatur. Welchen Wert der Zusammenhang annimmt, hängt von der aktuellen Regelabweichung ab. Es wird die Regelgröße bevorzugt, deren (skaliertes) Fehler größer ist. Über den Parameter $\gamma \in [0, 1]$ werden die Fehler und damit letztendlich der Ladedruck gegenüber der Abgastemperatur gewichtet; eine großes γ bevorzugt den Ladedruck und ein kleines die Abgastemperatur. Die kleinen

¹⁵Wird der Kontrollhorizont kleiner als der Prädiktionshorizont gewählt, tritt dieses Problem nicht auf.

additiven Beiträge sind zur numerischen Absicherung enthalten. Ohne den zusätzlichen Beitrag im Nenner würde bei verschwindenden Fehlern durch Null geteilt.¹⁶

Ein im Sinne des Kraftstoffverbrauchs vorteilhaftes Verhalten könnte beispielsweise über eine Bestrafung der Pumpverluste realisiert werden – der Differenzdruck $p_3 - p_{22}$ ist möglichst klein. Eine Gewichtung der Pumpverluste hätte jedoch ungewollte Seiteneffekte: Zum einen können auch ohne Verwendung der ETV Pumpverluste vorhanden sein. Eine Bestrafung würde entsprechend einen Zielkonflikt zwischen der Einregelung des Ladedrucks und der Reduktion der Pumpverluste erzeugen. Zum anderen würde eine Erhöhung der Abgastemperatur über die ETV mit der Bestrafung der Pumpverluste ebenfalls in Konflikt geraten – die ETV erhöht den Differenzdruck. Beide Konflikte würden eine bleibende Regelabweichung nach sich ziehen.¹⁷ Gleiches gilt für die Aktuatorsollwerte: Jeder beliebige aber feste Aktuatorsollwert würde an den meisten Betriebspunkten den Regelungszielen entgegen arbeiten.

Die verwendeten Zahlenwerte der Parameter sind in Tab. 5.2 aufgeführt.

Tab. 5.2: Grundeinstellungen des MPC-Optimierungsproblems.

| Parameter | Wert | Parameter | Wert | | |
|--------------------------------|----------|-----------|----------------------------------|-----------|-------|
| $k_{\text{etv,dyn}}$ | [-] | 0.180 | $r_{\text{etv,min}}$ [%] | 0,000 | |
| $n_{\text{tc,max}}$ | [kU/min] | 100,000 | $r_{\text{wg,max}}$ [%] | 100,000 | |
| $p_{3,\text{max}}$ | [bar] | 4,000 | $r_{\text{wg,min}}$ [%] | 0,000 | |
| q_p | [-] | 10,000 | $\dot{r}_{\text{etv,max}}$ [%/s] | 2289,000 | |
| q_T | [-] | 1,000 | $\dot{r}_{\text{etv,min}}$ [%/s] | -2289,000 | |
| $q_{\dot{\zeta}_{\text{wg}}}$ | [-] | 0,010 | $\dot{r}_{\text{wg,max}}$ [%/s] | 652,000 | |
| $q_{\dot{\zeta}_{\text{etv}}}$ | [-] | 0,500 | $\dot{r}_{\text{wg,min}}$ [%/s] | 652,000 | |
| $r_{\text{etv,max}}$ | [%] | 95,000 | γ | [-] | 0,001 |
| $r_{\text{etv,min/max}}^*$ | [%] | 0,000 | λ_{min} | [-] | 1,200 |

5.2.5 Nebenbedingungen

Die sich unmittelbar aus den Anforderungen an die Regelung (vgl. Abschnitt 5.2.1) ergebenden Nebenbedingungen werden über Ungleichungsbeschränkungen (5.7f) in den Algo-

¹⁶Trotz der Min-Verknüpfung in (5.8c) ist die Gütefunktion (5.8a) nach T_{40} stetig differenzierbar. Damit erfüllt die Gütefunktion die KKT-Bedingungen (vgl. Satz 2.2.4).

¹⁷Um den Konflikt mit dem Ladedruck aufzulösen, ist die Tolerierung eines bestimmten Differenzdrucks analog zur Umsetzung der Mindestabgastemperatur denkbar. Es werden dann lediglich die Differenzdrücke bestraft, die über einem Schwellwert liegen.

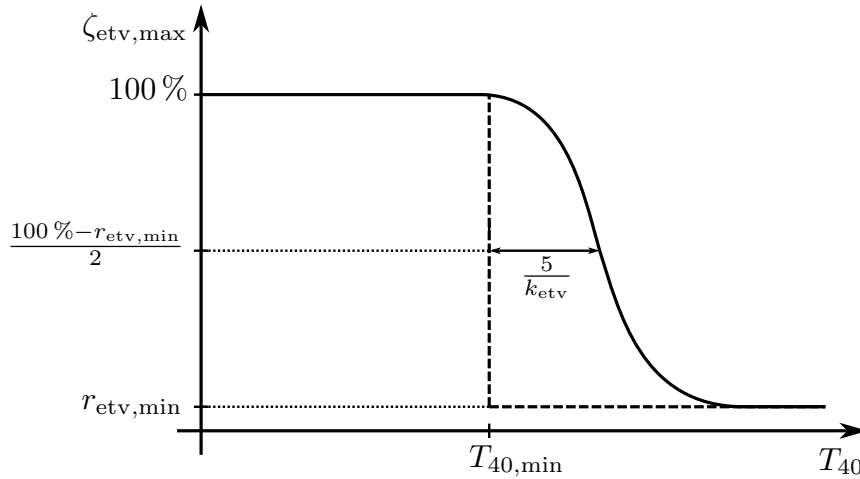


Abb. 5.9: Veranschaulichung der abgastemperaturabhängigen ETV-Begrenzung.

rithmus eingebracht. Es wird

$$\mathbf{c}_{\mathcal{I}}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) = \begin{bmatrix} n_{tc} - n_{tc,\max} \\ p_3 - p_{3,\max} \\ T_3 - T_{3,\max} \\ -\lambda + \lambda_{\min} \\ \zeta_{\text{etv}} - \zeta_{\text{etv},\max} \end{bmatrix} \leq \mathbf{0} \quad (5.9)$$

verwendet. Die letzte Gleichung begrenzt die ETV-Position durch

$$\zeta_{\text{etv}} \leq \zeta_{\text{etv},\max} = r_{\text{etv},\min} + (100\% - r_{\text{etv},\min}) \cdot \text{sig}(-k_{\text{etv}}(T_{40} - T_{40,\min}) + 5)$$

unter Verwendung der *Sigmoid-Funktion*

$$\text{sig}(x) := \frac{1}{1 + \exp(-x)}.$$

Überschreitet die Abgastemperatur T_{40} die Mindesttemperatur $T_{40,\min}$, wird die maximal erlaubte Klappenstellung in Abhängigkeit von der aktuellen Abgastemperatur bis maximal zur unteren Aktuatorgrenze $r_{\text{etv},\min}$ reduziert (vgl. Abb. 5.9).¹⁸ Über den Parameter $k_{\text{etv}} \in (0,05; 10]$ wird die Steilheit des Übergangs eingestellt. Auf diese Weise wird indirekt ein erhöhter Kraftstoffverbrauch bestraft. Die ETV wirkt durch die Pumpverluste verbrauchssteigernd. Die Verluste werden nur dann in Kauf genommen, wenn die Abgastemperatur über das WG alleine nicht dargestellt werden kann. Ein diskretes Umschalten der Limitierung (gestrichelter Verlauf in Abb. 5.9) ist wegen der daraus folgenden Verletzung der KKT-Bedingungen (vgl. Satz 2.2.4) nicht möglich, welche zur Verwendung des Lösungsalgorithmus zu erfüllen sind. Die Bedingungen fordern u. a. die stetige Differenzierbarkeit der Nebenbedingungen.

¹⁸Eine Umsetzung unter Verwendung des ETV-Sollwertes r_{etv} ist ebenfalls möglich.

Gleichungsbeschränkungen (5.7e) werden zur Formulierung der MPC nicht verwendet, damit gilt:

$$\mathbf{c}_{\mathcal{E}}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) = \mathbf{0}.$$

Die Aktuatorbeschränkungen werden über (5.7c) und (5.7d) berücksichtigt. Die Aussteuerungsgrenzen sind durch

$$\mathbf{b}_l = \begin{bmatrix} r_{\text{wg,min}} & r_{\text{etv,min}} \end{bmatrix}^T, \\ \mathbf{b}_u = \begin{bmatrix} r_{\text{wg,max}} & r_{\text{etv,max}} \end{bmatrix}^T$$

gegeben und die maximale sinnvolle Änderungsrate durch

$$\mathbf{r}_l = \begin{bmatrix} \dot{r}_{\text{wg,min}} & \dot{r}_{\text{etv,min}} \end{bmatrix}^T, \\ \mathbf{r}_u = \begin{bmatrix} \dot{r}_{\text{wg,max}} & \dot{r}_{\text{etv,max}} \end{bmatrix}^T.$$

Letztere dient dazu, den Suchraum des Optimierungsproblems einzugrenzen. Des Weiteren kann über eine Begrenzung der Änderungsrate der Regler künstlich verlangsamt werden, um beispielsweise an besonders kritischen Betriebspunkten – im Sinne der technischen Belastungsgrenzen – ein behutsames Reglerverhalten zu erzwingen. Bevor im Folgenden eine Abschätzung für die Begrenzungen gegeben wird, sei angemerkt, dass sämtliche untersuchten Linearisierungen des Luftpfades stabil sind. Deshalb ist davon auszugehen, dass durch eine Beschränkung der Aktuatorgeschwindigkeit kein instabiles Verhalten erzeugt wird; der Regler wird lediglich langsamer. Eine Abschätzung soll anhand der Bandbreiten¹⁹ der Übertragungsfunktionen (5.5) ermittelt werden. Eine Analyse der maximalen Bandbreite der einzelnen Übertragungsfunktionen für sämtliche Linearisierungen ergibt die Frequenzen

$$\omega_{\text{B},p_{22},r_{\text{wg}}} = 3,260 \text{ 1/s}, \quad \omega_{\text{B},T_{40},r_{\text{wg}}} = 1,374 \text{ 1/s}, \\ \omega_{\text{B},p_{22},r_{\text{etv}}} = 11,443 \text{ 1/s}, \quad \omega_{\text{B},T_{40},r_{\text{etv}}} = 1,456 \text{ 1/s}.$$

Sämtliche Frequenzen liegen erwartungsgemäß unter der Bandbreite der Aktuatoren, welche bei $\omega_{\text{B},\text{wg}} = 1/\tau_{\text{wg}} = 48,08 \text{ 1/s}$ sowie $\omega_{\text{B},\text{etv}} = 1/\tau_{\text{etv}} = 12,82 \text{ 1/s}$ liegen. Auffällig ist, dass die Regelung des Ladedrucks über die ETV fast mit voller Aktuatorbandbreite erfolgen kann. Die maximal sinnvollen Änderungsraten sollen über die Betrachtung eines PT1-Filters abgeschätzt werden. In Abb. 5.10 ist der Zusammenhang zwischen der Sprungantwort eines PT1-Filters mit unterschiedlichen vorgeschalteten Ratenlimitierungen dargestellt. Es ist zu sehen, wie die Verzögerung der Sprungantwort mit steigender Ratenlimitierung zunimmt und im Falle der stärksten Limitierung das Gesamtverhalten durch diese dominiert wird. Eine durch die Rate von $2/\tau$ eingefügte Verzögerung wird als vertretbar

¹⁹Die Bandbreite ist als die Kreisfrequenz ω definiert, bei der der Amplitudengang gegenüber dem Wert für $\omega \rightarrow 0$ um 3 dB abgefallen ist und für alle höheren Frequenzen unter diesem Wert bleibt.

eingeschätzt. In diesem Fall bleibt der Aktuator maximal ca. 20 % hinter der möglichen Auslenkung zurück. Daraus ergibt sich die Heuristik

$$\dot{r}_{\max} = 2\omega_B,$$

aus der die Maximalsteigungen abgeleitet werden. Es folgen die Grenzen

$$\begin{aligned} \dot{r}_{\text{wg,max}} = -\dot{r}_{\text{wg,min}} &:= 100\% \cdot 2 \max\{\omega_{B,p22,r_{\text{wg}}}, \omega_{B,T40,r_{\text{wg}}}\} = 652\%/s, \\ \dot{r}_{\text{etv,max}} = -\dot{r}_{\text{etv,min}} &:= 100\% \cdot 2 \max\{\omega_{B,p22,r_{\text{etv}}}, \omega_{B,T40,r_{\text{etv}}}\} = 2289\%/s. \end{aligned}$$

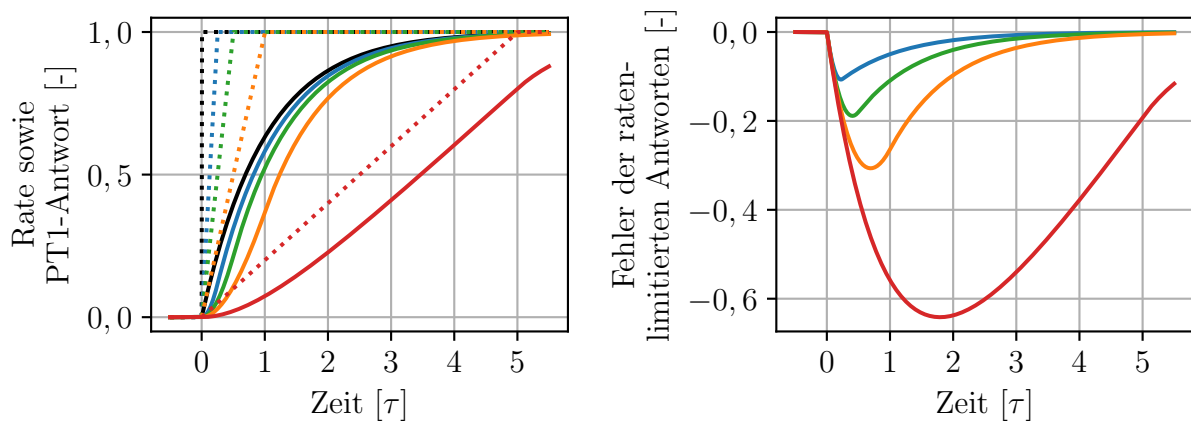


Abb. 5.10: Verhalten eines PT1-Filters mit der Zeitkonstanten τ bei ratenlimitierten Sprüngen. Es sind die Ratenlimitierungen (in abnehmender Steilheit) $\frac{\infty}{\tau}$, $\frac{4}{\tau}$, $\frac{2}{\tau}$, $\frac{1}{\tau}$, $\frac{0,2}{\tau}$ dargestellt.

5.3 Konfiguration des Lösungsalgorithmus

In diesem Abschnitt wird anhand einer synthetischen Regelungssituation der MPC-Lösungsalgorithmus aus Kapitel 4 eingestellt. Die Reihenfolge der Ausführungen ist der chronologischen Reihenfolge bei dessen Entwicklung nachempfunden. Die Haupttriebfeder war dabei die Reduktion der Rechenleistung sowie des Speicherverbrauchs durch sukzessive Vereinfachungen, so dass der Algorithmus auf der ECU lauffähig ist. Im Einzelnen zielte die Entwicklung darauf ab, eine Ausführungszeit auf dem Entwicklungsrechner von 0,41 ms nicht zu überschreiten, den beanspruchten *Stack*-Speicher unter 2 kB zu halten und weniger als 5 kB Speicher auf dem *Heap* zu beanspruchen. Die Anforderungen sind in Tab. 1.2 aufgeführt. Die Ausführungszeit entstammt folgender Abschätzung: Auf der ECU soll der Algorithmus alle 50 ms ausgeführt werden. Formal steht damit ebendiese Zeit zur Verfügung. Jedoch wäre die ECU in diesem Fall zu 100 % ausgelastet. Nach den Anforderungen in Tab. 1.2 wird eine maximale CPU-Auslastung von 10 % durch den Regelalgo-

rithmus akzeptiert; es stehen mithin lediglich 5 ms Rechenzeit zur Verfügung. Bezüglich des verwendeten Entwicklungsrechners (*Intel i5-8250U*, 3,4 GHz) hat die ECU (*Infineon Aurix TriCore*, 300 Mhz) eine etwa eine Größenordnung kleinere Taktrate. Überschlägig ergibt sich damit die um einen Faktor 12 kleinere auf dem Entwicklungsrechner angestrebte Ausführungszeit von 0,41 ms. Es handelt sich hierbei um eine grobe Abschätzung, da die Taktrate lediglich einer von mehreren die Ausführungszeit bestimmenden Aspekten ist.²⁰

5.3.1 Grundeinstellungen

Es wird mit den im Vorhinein als vertretbar angesehenen Reglereinstellungen begonnen und im Weiteren die Komplexität so weit reduziert, bis die Ausführungszeit und die Speicherverbräuche innerhalb der Zielparameter liegen. Die zur Verfügung stehenden Reglereinstellungen sind in Tab. 4.3 aufgeführt. Die nicht-variierten Einstellungen des MPC-Lösungsalgorithmus sind in Tab. 5.3 zusammengefasst und wurden durch numerische Untersuchungen ermittelt.

Die initiale Wahl der im Anschluss veränderten Parameter soll im Folgenden dargelegt werden. Ebenfalls durch numerische Untersuchungen motiviert, wird der Prädiktionshorizont im Vorhinein auf $T := 1,5$ s und der Kontrollhorizont auf $T_C := 0,5$ s festgelegt. Unter Berücksichtigung der Abtastrate des Reglers T_s wird ein äquidistantes Zeitraster von

$$\bar{\tau} = [0, T_s, 2T_s, \dots, 29T_s, 30T_s] \quad (5.10)$$

gewählt. Um den gewünschten Prädiktions- und Kontrollhorizont zu erreichen, sind $N := 31$ Stützstellen und $M := 11$ zu optimierende Kontrollzeitpunkte erforderlich. Die numerische Integration erfolgt mit einer Integratorschrittweite von $T_{\text{int}} := 12$ ms mit dem *klassischen Runge-Kutta-Verfahren* (RK4). In Abschnitt 3.3.3 wurde gezeigt, dass sämtliche Linearisierungen des Luftpfades mit dem *Euler-Verfahren* mit einer Integrationschrittweite von 12,9 ms stabil integriert werden können.²¹ Es wird das RK4-Verfahren gewählt, um zum einen die Genauigkeit der Integration zu erhöhen und um zum anderen eine größere Stabilitätsreserve zu haben (vgl. Abschnitt 2.1.3). Die ermittelte Schrittweite für das Euler-Verfahren gilt lediglich für die Linearisierungen und stellt damit eine obere Grenze dar. Es wird zunächst keine iterative Liniensuche $L := 1$ durchgeführt. Die Anzahl der Gradienteniterationen $G := 7$ wird so gewählt, dass deren Rechenzeit der bevorzugten Konfiguration E (vgl. Abschnitt 5.3.2) von $G = 4$ und $L = 4$ des entwickelten Regelungsalgorithmus entspricht.

Die Grundeinstellungen entsprechen den Möglichkeiten des Regelungsalgorithmus ohne die

²⁰Eine Einführung zur Abschätzung der FLOPS (engl.: *Floating Point Operations Per Second*) einer CPU basierend auf deren technischen Eckdaten ist in [Dol18] zu finden.

²¹Die Wahl fällt auf 12 ms anstelle von 12,9 ms, weil die Umsetzung der Vor- und Rückwärtsintegration implizit diese Schrittweite wählen würde (vgl. Abschnitt 4.4.2).

in dieser Arbeit vorgenommenen Erweiterungen. Die Wahl eines im Vergleich zum Prädiktionshorizont verkürzten Kontrollhorizonts liegt allerdings bereits außerhalb der Möglichkeiten des Referenzalgorithmus.

Tab. 5.3: Nicht-varierte Einstellungen des MPC-Lösungsalgorithmus. Die Bedeutung der Parameter kann Tab. 4.3 entnommen werden.

| Parameter | Wert | Parameter | Wert |
|--|-----------------------|--|---|
| T_s [s] | 0,05 | β [-] | 1 |
| c_1 [-] | 0,01 | γ_{inc} [-] | 1,5 |
| c_2 [-] | 0,99 | γ_{dec} [-] | 0,9 |
| \mathbf{u}_{ini} [% , %] ^T | [100, 0] ^T | $\boldsymbol{\varepsilon}_{\tilde{\mathbf{c}}}$ [U/s, bar, %, K, -] ^T | [10 ³ ; 0,1; 1; 10; 0,01] ^T |
| α_{dec} [-] | 0,2 | $\boldsymbol{\varepsilon}_{\tilde{\mathbf{c}}_T}$ | nicht verwendet |
| α_{fac} [-] | 0,8 | $\varepsilon_{\mathbf{u}}$ [-] | 0,01 |
| α_{inc} [-] | 1,2 | μ_{max} [-] | 10 ³ |
| α_{max} [-] | 10 | ν_{max} [-] | 10 ⁵ |
| α_{min} [-] | 10 ⁻⁸ | ν_{min} [-] | 0,1 |

5.3.2 Konfigurationen

Es werden die in Tab. 5.4 aufgeführten MPC-Konfigurationen evaluiert, wobei sich ausschließlich die in der Tabelle aufgeführten Einstellungen ändern; die übrigen bleiben konstant.

Tab. 5.4: MPC-Konfigurationen. Die übrigen Parameter sind in Tab. 5.3 aufgeführt.

| Konfiguration | N | M | $\bar{\tau}$ | Integrator ¹ | T_{int} [ms] | G | L | IL ² |
|---------------|----|----|--------------|-------------------------|--------------------------|---|---|-----------------|
| A | 31 | 11 | siehe (5.10) | RK4 | 12 | 7 | 1 | nein |
| B | 31 | 11 | siehe (5.10) | RK4 | 12 | 4 | 4 | nein |
| C | 7 | 6 | siehe (5.11) | RK4 | 12 | 4 | 4 | nein |
| D | 7 | 6 | siehe (5.11) | IRK2 | 200 | 4 | 4 | nein |
| E | 7 | 6 | siehe (5.11) | IRK2 | 200 | 4 | 4 | ja |
| F | 7 | 6 | siehe (5.11) | IRK2 | 200 | 3 | 4 | ja |

¹ RK4: *Runge-Kutta-4-Verfahren*, IRK2: *Implizites Runge-Kutta-2-Verfahren* bzw. *implizite Trapezmethode*.

² IL: *Initiale Linearisierung des Modells*.

Evaluiert werden die verschiedenen Konfigurationen anhand eines synthetischen Testzyklus. Sämtliche Modelleingänge (3.3) sind fest (vgl. Tab. 5.5), bis auf die Einspritzmenge q_{inj} , welche als Störgröße zeitlich variiert wird, sowie die vom Regler berechneten Stellgrößen r_{wg} und r_{etv} . Auf diese Weise ist sichergestellt, dass sämtliche zu beobachtenden

Effekte ausschließlich auf die MPC zurückzuführen sind. Das Verhalten der Entwürfe soll lediglich qualitativ beschrieben werden. Eine detailliertere Betrachtung folgt in Abschnitt 5.4. Die Sollwerte des Reglers werden derart variiert, dass sämtliche Nebenbedingungen aktiv werden.

Tab. 5.5: Fixierte Modelleingänge des Szenarios zur Reglerevaluierung.

| Parameter | | Wert |
|---------------------|---------|--------|
| \dot{Q}_f^* | [kW] | 237,0 |
| T_0 | [°C] | 25,0 |
| n_{eng} | [U/min] | 1900,0 |
| p_0 | [bar] | 1,0 |
| p_{41} | [bar] | 1,0 |
| q_{inj}^* | [mg] | 189,0 |
| $\Delta p_{3,22}^*$ | [bar] | 1,6 |

Konfiguration A – Ausschließliche Gradienteniteration

Unter Verwendung der initialen Einstellungen des Algorithmus ergeben sich die in Abb. B.1 dargestellten Verläufe. Es ist zu beobachten, wie der Regler an mehreren Stellen ein instabiles Verhalten zeigt. Die erste Instabilität tritt im Bereich $100\text{ s} < t < 120\text{ s}$ auf. In dieser Region wird die ETV besonders stark aktuiert. Es liegt die Vermutung nahe, dass dort die Nichtlinearität des Aktuators (vgl. Abb. 3.3) besonders zum Tragen kommt und große, destabilisierende Gradienten erzeugt. Der zweite instabile Bereich befindet sich im Bereich $t > 140\text{ s}$, in dem ausschließlich auf den Druck geregelt wird. Bei $t = 140\text{ s}$ wird die Abgasmindesttemperatur auf 200 °C gesetzt und hat damit keinen Einfluss auf die Regelung. Aus der Wahl der Parameter (vgl. Tab. 5.2) geht hervor, dass die Druckregelung über das WG die größte Dynamik aufweist. Diese ist für diesen Regler zu anspruchsvoll und führt zur Destabilisierung. Im Anfangsbereich für $t < 40\text{ s}$ ist zu sehen, wie ein Kompromiss zwischen dem Ladedruck und der Abgastemperatur gefunden wird. Gemäß der Wahl von $\gamma = 0,001$ liegt die Priorität auf der Abgastemperatur. Es werden sowohl das WG als auch die ETV verwendet. Bei $t = 40\text{ s}$ erfolgt ein Sprung des Abgastemperatursollwertes. Diesem folgt der Regler mit dem WG unmittelbar, die ETV zeigt hingegen eine vergleichsweise langsame Bewegung, die sich mit steigender Aktuierung beschleunigt. Dieses Verhalten ist ebenfalls durch die Nichtlinearität des Aktuators zu erklären. Der Sprung der Mindesttemperatur bei $t = 80\text{ s}$ resultiert in einer deutlich schnelleren ETV-Bewegung als bei $t = 40\text{ s}$.

Konfiguration B – Iterative Liniensuche

Der Lösungsalgorithmus wird dahingehend abgeändert, dass weniger Gradienteniterationen, im Gegenzug aber mehr Liniensuche-Iterationen durchgeführt werden. Die Parameter werden derart gewählt, dass die benötigte Rechenzeit näherungsweise identisch zur vorigen Konfiguration ist.

Durch die Verwendung der Liniensuche (vgl. Abschnitt 4.4.3) können die Instabilitäten vollständig beseitigt werden, wie in Abb. B.2 zu sehen ist. Die ETV bewegt sich im Vergleich zu Abb. B.1 qualitativ identisch, wohingegen beim Verhalten des WG Unterschiede festzustellen sind – es werden die Bereiche für $t \leq 100$ s verglichen, bevor die vorige Konfiguration instabil wird. Die geringere Anzahl an Gradienteniterationen hat ein langsames Konvergenzverhalten zur Folge. Die ETV erreicht erst bei $t \approx 80$ s die 50 %-Schwelle, wohingegen in der vorigen Konfiguration dieser Wert bei $t \approx 62$ s erreicht wird. Eine erste Schließbewegung des WG ist in dieser Konfiguration andeutungsweise ab $t = 80$ s zu sehen.

Stationär werden sämtliche Nebenbedingungen eingehalten. In der Dynamik ist die Nichteinhaltung der dynamischen ETV-Begrenzung sowie der Maximaldrehzahl des Turboladers bei $t \approx 180$ s für etwa zwei Sekunden auffällig. Zu begründen ist dies durch die Absenkung der Maximaldrehzahl. Der Regler versucht dies durch ein Öffnen des WG und ein Schließen der ETV zu erreichen. Letztere Aktion erzeugt die Verletzung der ETV-Begrenzung. An dieser Stelle stehen sich zwei Nebenbedingungen gegenüber und der Regler muss zwangsläufig gegen mindestens eine von beiden verstoßen. Durch die Trägheit des Turboladers kann dessen Drehzahl nicht in Nullzeit abgesenkt werden.

Konfiguration C – Reduktion der Stützstellen

Konfiguration B konsumiert 3,284 kB Speicher auf dem Stack. Für eine Implementierung auf der ECU ist jedoch eine Maximalgrenze von 2 kB einzuhalten (vgl. Tab. 1.2). Erreicht werden soll dies über eine Reduktion der Stützstellen. Diese werden anstelle der bisher verwendeten äquidistanten Verteilung mit der Schrittweite T_s bei

$$\bar{\tau} = [0, T_s, 2 T_s, 3 T_s, 4 T_s, 10 T_s, 30 T_s] \quad (5.11)$$

platziert. Die ersten 5 Stützstellen werden passend zur Abtastrate des Reglers gewählt. Dieser Entscheidung liegt die Beobachtung zugrunde, dass ein Regler unmittelbar nach einer Sollwertänderung die meiste Aktivität zeigt. Die vorletzte Stützstelle ist so gewählt, dass der Kontrollhorizont weiterhin bei $T_C = 0,5$ s liegt. Die letzte Stützstelle liegt am Ende des Prädiktionshorizonts bei $T = 1,5$ s. Durch diese Maßnahme wird der Speicher auf dem Stack auf 1,668 kB reduziert. Dieser Wert liegt um ein Vielfaches über der theoretischen Untergrenze von 0,17 kB (vgl. Abschnitt 4.5). Ursächlich sind zusätzliche Zwischenvariablen durch die programmiertechnische Umsetzung sowie die Berechnung der Systemmatrix, wel-

che 0,496 kB zum Stackverbrauch beiträgt (vgl. Tab. 3.4). Der benötigte Speicher auf dem Heap reduziert sich auf 0,372 kB, welcher geringfügig über dem theoretischen Minimum für den Algorithmus von 0,340 kB liegt (vgl. Abschnitt 4.5). Die größere Speicherbelegung kommt durch die zu speichernden Sollwerte und Nebenbedingungen zustande.

Der zugehörige Verlauf ist in Abb. B.3 dargestellt. Im Vergleich mit Konfiguration B ist eine leichte Verschlechterung des Konvergenzverhaltens zu bemerken. Die ETV erreicht eine Aktuierung von 50 % bei $t \approx 82$ s, wohingegen Konfiguration B diese etwa zwei Sekunden früher erreicht. Des Weiteren ist im Bereich $100\text{ s} \leq t \leq 120\text{ s}$ ein unruhigeres Reglerverhalten festzustellen. Die ETV führt in diesem Bereich periodische Bewegungen mit einer Amplitude von weniger als 1 % aus. Durch die hohe Verstärkung der ETV in diesem Bereich bewegt sich der Druck p_3 in einer relativen Betrachtung stärker.

Konfiguration D – Reduktion der Abtastzeit

Die benötigte Rechenzeit von Konfiguration C liegt mit ca. 7 ms über den zu erreichenden 0,41 ms. Die bisher betrachteten Konfigurationen werden bei einer festen Schrittweite von 12 ms mit einem RK4-Verfahren numerisch integriert. Es soll die Abtastzeit auf 200 ms erhöht werden. Selbst das stabiler gewählte RK4-Verfahren ist für eine fast zwanzigfach vergrößerte Schrittweite nicht geeignet. Dies wird durch den Vergleich der Stabilitätsgebiete beider Verfahren klar (vgl. Abb. 2.2c und 2.2b). Stattdessen wird die *implizite Trapezmethode* (RK2) verwendet, welche für beliebige Abtastzeiten stabil ist – sofern das zu integrierende System ebenfalls stabil ist (vgl. Abb. 2.2d). Jedoch hat das Verfahren im Gegensatz zum RK4-Verfahren die Konsistenzordnung $p = 2$, und es wird bezüglich der im System enthaltenen Dynamiken unter-abgetastet. Beides führt zu einer Verschlechterung der Genauigkeit. Es ist insbesondere mit einer Glättung der schnellen Moden zu rechnen. Die Änderung der Schrittweite reduziert die Rechenzeit für die MPC-Ausführung auf ca. 0,7 ms. Ein größerer Zuwachs wird durch die Umsetzung des IRK2-Verfahrens (vgl. Abschnitt 2.1.3), welches im Gegensatz zum RK4-Verfahren fünf anstelle von vier Auswertungen der Modell-DGL erfordert, und durch die Wahl der ersten vier Stützstellen in (5.11) verhindert. Die Stützstellen erzwingen eine Schrittweite von 50 ms, begründet durch die Umsetzung der Vor- und Rückwärtsintegration (vgl. Abschnitt 4.4.2). Dieses implementierungstechnische Detail führt dazu, dass in den Bereichen besonders hoher Regleraktivität der Fehler der numerischen Integration durch die Unter-Abtastung geringer ausfällt.

In Abb. B.4 ist das Verhalten im synthetischen Testzyklus zu sehen. Es ist zu beobachten, dass sich das Konvergenzverhalten weiter verschlechtert. Jedoch wirkt sich die Verschlechterung lediglich auf die Bereiche aus, in denen die ETV verwendet wird und deren aktuelle Position unter 45 % liegt. Die übrigen Bereiche sind optisch nicht zu unterscheiden. Die Verschlechterung des Konvergenzverhaltens wirkt sich positiv auf die Einhaltung der dynamischen ETV-Begrenzung bei $t \approx 180$ s aus. In Abb. B.3 liegt eine stärkere Grenzverletzung vor. Neben der langsameren Konvergenzgeschwindigkeit ist eine stärkere Ausprägung des unruhigen Reglerverhaltens im Bereich $100\text{ s} \leq t \leq 120\text{ s}$ zu beobachten.

Konfiguration E – Initiale Linearisierung

In Konfiguration B wurde gegenüber Konfiguration A die Anzahl der Iterationen verändert. In der ersten Konfiguration werden sieben Gradienteniterationen durchgeführt, was ebenso vielen Vorwärts- und Rückwärtsintegrationen entspricht. In Konfiguration B werden vier Gradienteniterationen und pro Gradienteniteration vier Liniensuche-Iterationen durchgeführt, dies entspricht sechzehn Vorwärts- und vier Rückwärtsintegrationen. Dies lässt darauf schließen, dass die Rückwärtsintegrationen rechentechnisch aufwendiger sind. Zur Rückwärtsintegration wird die Systemmatrix \mathbf{A} verwendet. Der Rechenaufwand im Vergleich zur Auswertung der DGL $\mathbf{f}(\cdot)$ ist in Tab. 3.4 gegeben. Die Tabelle bestätigt den höheren Rechenaufwand zur Berechnung der Systemmatrix. Der Gedanke, den Rechenaufwand weiter zu reduzieren, beruht darauf, die Linearisierung des Systems ($\mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k, \mathbf{D}_k$) am aktuellen Zeitpunkt t_k zu ermitteln und abzuspeichern. Die Rückwärtsintegration löst damit

$$\dot{\boldsymbol{\lambda}}(t) = -\nabla_{\mathbf{x}} l(\boldsymbol{\psi}(t, \bar{\mathbf{x}}_k), \boldsymbol{\psi}_s(t, \bar{\mathbf{u}}_k), \boldsymbol{\psi}(t, \bar{\boldsymbol{\mu}}_k), t; \boldsymbol{\psi}(t, \bar{\boldsymbol{\nu}}_k)) - \mathbf{A}_k^T \boldsymbol{\lambda}(t)$$

im Gegensatz zu (4.16c). Durch diese Maßnahme reduziert sich die Rechenzeit auf 0,51 ms, wohingegen der benötigte Stack-Speicher von 1,668 kB auf 1,892 kB steigt.

Im Vergleich zur vorherigen Konfiguration sind wenige Unterschiede zu erkennen: Das Erreichen der 50 %-Schwelle verschiebt sich um etwa 1,5 s nach hinten, was auf eine Verschlechterung der Konvergenzgeschwindigkeit hindeutet. Allerdings ist bei $t = 140$ s ein gegenteiliger Effekt zu beobachten (vgl. Abb. B.5). Auf die Reglerruhe wirkt sich die Linearisierung förderlich aus; es sind weniger Schwingungen bei den großen ETV-Aktuierungen zu beobachten.

Konfiguration F – Reduktion der Gradienteniterationen

Um die Ausführungsdauer unter die eingangs definierten 0,41 ms zu senken, wird die Anzahl der Gradienteniterationen auf drei reduziert, womit sich eine Ausführungsdauer von ca. 0,39 ms ergibt.

Der zugehörige Verlauf ist in Abb. B.6 dargestellt. Wie zu erwarten benötigt der Regler mehr Zeit, um die optimalen Aktuatorpositionen zu ermitteln, was sich in langsameren Bewegungen äußert. Beispielsweise dauert es im Vergleich zur vorigen Konfiguration länger, bis das WG ab $t = 120$ s eine Aktuierung von 100 % erreicht.

5.3.3 Vergleich der Konfigurationen

In Abb. 5.11 sind die Fehler sowie die Entwicklung der Rechenzeit und des RAM-Verbrauchs der zuvor besprochenen Konfigurationen dargestellt.²² Zuvor sei angemerkt, dass sich der RAM-Verbrauch aus der Summe des beanspruchten Heap- und Stack-Speichers ergibt. Die Darstellung ist auf Konfiguration B normiert. Konfiguration A ist für eine Regelung aufgrund des instabilen Verhaltens ungeeignet und wird nicht näher betrachtet. Jedoch sei angemerkt, dass das Regelverhalten durch Anpassungen des Algorithmus stabilisiert werden kann, ohne die Rechenzeit zu steigern oder den verbrauchten RAM zu vergrößern.

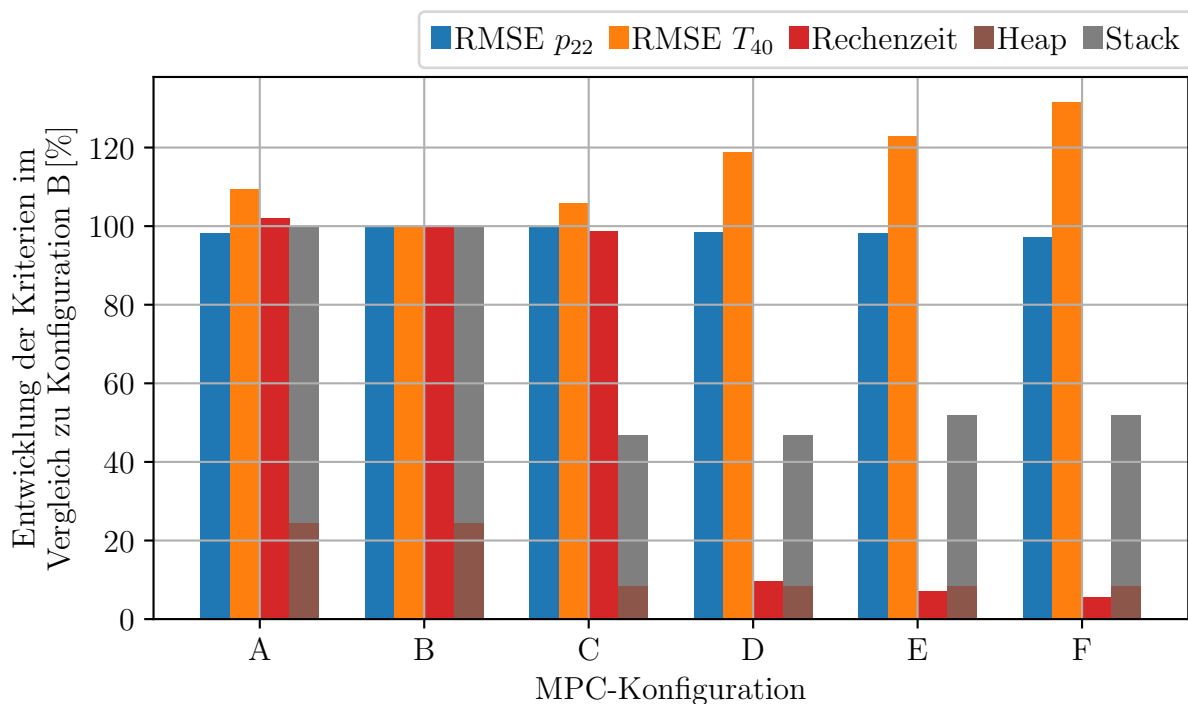


Abb. 5.11: Fehler der Regelgrößen sowie Rechenzeit und Speicherverbrauch der MPC-Konfigurationen. Die Darstellung basiert auf den Daten aus Tab. 5.6.

Auffällig ist, dass sich der Regelfehler der Abgastemperatur mit zunehmender algorithmischer Vereinfachung verschlechtert. Die stärkste Verschlechterung wird durch die Änderung der Abtastrate von Konfiguration C nach Konfiguration D verursacht. Maßgeblich geht der zunehmende Fehler auf die Verschlechterung des Konvergenzverhaltens zurück. Dies führt dazu, dass die ETV immer länger braucht, bis sie eine kritische Aktuierung erreicht, aber die ETV vom Algorithmus mit höherer Geschwindigkeit in die Regelung eingebunden wird. Die kritische Aktuierung ist die Stelle, ab der der Einfluss der ETV auf den Druck hinter dem Turbolader hinreichend groß wird (vgl. Abb. 3.3); die Steigung des Zusammenhangs

²²Die verwendeten Fehlermaße können Anhang A.2 entnommen werden.

wird hinreichend groß. Ab dem Überschreiten der kritischen Schwelle wirken sich die algorithmischen Vereinfachungen weniger gravierend aus und das Regelverhalten gleicht sich an. Auf die Regelung des Ladedrucks wirkt sich die schlechtere Konvergenz hingegen förderlich aus. Im ersten Teil $t \leq 140$ s stehen Druck und Temperatur in einem Zielkonflikt; entsprechend profitiert der Ladedruck von einer schlechteren Regelung der Abgastemperatur. Die Unterschiede der Regelgüte bezüglich des Ladedrucks fallen jedoch weniger stark aus. Nach dem Überschreiten der kritischen Schwelle sind die Konfigurationen B bis F für die Regelung geeignet.

Die Entwicklung der Rechenzeit erfährt, analog zur Veränderung der Regelgüte der Abgastemperatur, von Konfiguration C nach Konfiguration D die größte relative Veränderung. Zwischen diesen beiden Konfigurationen wurde die Integratorschrittweite vergrößert. Es kann gefolgert werden, dass die Vor- und Rückwärtsintegrationen maßgeblich für die Rechenzeit sind. Der RAM-Verbrauch kann durch die Reduktion der Stützstellen von Konfiguration B nach Konfiguration C nahezu halbiert werden, ohne das Verhalten der Regelung maßgeblich zu verschlechtern.

Insgesamt stellt Konfiguration D den besten Kompromiss aus Verschlechterung der Güte und Reduktion der rechentechnischen Ressourcen dar. Die Kriterien für eine Implementierung auf der ECU erfüllt jedoch lediglich Konfiguration F. Diese Konfiguration stellt daher die Basis der weiteren Untersuchungen dar.

Tab. 5.6: Fehler der Regelgrößen sowie Rechenzeit und Speicherverbrauch der MPC-Konfigurationen.

| Konfiguration | RMSE | | Rechenzeit [ms] | RAM | |
|---------------|----------|------------------------|--------------------|-------|------|
| | [bar] | [°C] | | [kB] | |
| | p_{22} | $T_{40} > T_{40,\min}$ | Heap | Stack | |
| A | 0,89 | 111,04 | 7,34 | 1,07 | 3,28 |
| B | 0,91 | 101,53 | 7,19 | 1,07 | 3,28 |
| C | 0,91 | 107,33 | 7,10 | 0,37 | 1,67 |
| D | 0,89 | 120,55 | 0,69 | 0,37 | 1,67 |
| E | 0,89 | 124,82 | 0,51 | 0,37 | 1,89 |
| F | 0,88 | 133,40 | 0,39 | 0,37 | 1,89 |

5.4 Validierung

Zur Validierung der MPC werden drei verschiedene Untersuchungen vorgenommen: eine Detailbetrachtung anhand von Drehzahl- und Drehmomentsprüngen/-rampen, eine Simulation des NRTC und schließlich die Begutachtung einer Prüfstandsaufzeichnung des NRTC. In allen drei Szenarien wird die MPC im geschlossenen Regelkreis nach dem in

Abb. 5.2 dargestellten Schema eingesetzt. Bei den simulativen Untersuchungen wird ein selbstentworfenen PID-Regler zur Regelung der Drehzahl verwendet. Am Prüfstand kommt der Regler des Serienmotors zum Einsatz. Der Ladedruck-Sollwert (vgl. Abb. 5.2) wird bei den simulativen Betrachtungen derart gewählt, dass stationär näherungsweise ein Verbrennungsluftverhältnis von $\lambda = 2$ erreicht wird. Auf der ECU wird diesbezüglich der Seriensatz verwendet. Ein weiterer Unterschied zwischen Simulation und Prüfstand liegt darin, dass in der Simulation dem Lastmoment fiktive Reibungsverluste und die Pumpverluste zugeschlagen werden. Zusätzlich wird eine Vorsteuerung für die ETV verwendet. Das Ziel dieser Maßnahme ist das *Überspringen* der Phase mit geringer Konvergenzgeschwindigkeit für kleine ETV-Aktuierungen (vgl. Abb. B.6).

5.4.1 Vorsteuerung der Abgasklappe

Die Vorsteuerung der ETV verfolgt die Idee, die Mindestaktuierung $r_{\text{etv},\min}$ bei ausreichender Unterschreitung der Abgasmindesttemperatur zu erhöhen. In diesem Fall wird die Position gewählt, ab der die ETV gerade noch keinen (nennenswerten) Einfluss auf das Gesamtsystem hat. Diese Grenze wird auf 0,1 bar festgelegt. Die Position, bei der die ETV diese Schwelle erreicht, ist in der linken Abbildung in Abb. 5.12 über die Drehzahl aufgetragen. Farblich ist zusätzlich die Abhängigkeit zur Einspritzmenge kenntlich gemacht. Es ist ein linearer Zusammenhang zwischen ETV-Position und Motordrehzahl bei fester Einspritzmenge ersichtlich. Für andere Einspritzmengen gilt derselbe, verschobene Zusammenhang. Es ist damit möglich, eine geeignete Lookup-Tabelle mit diesen beiden Abhängigkeiten zu finden. Statt des tabellenbasierten Ansatzes wird zur Vereinfachung eine Parametrierung über den Abgasmassenstrom W_{eo} gewählt. Der Zusammenhang ist in der rechten Abbildung aufgetragen; es besteht eine näherungsweise univariate Abhängigkeit zum Abgasmassenstrom.²³ Aufgrund der Tatsache, dass von der Vorsteuerung keine hohe Genauigkeit gefordert ist, wird eine lineare Approximation gewählt. Eine höhere Genauigkeit ist durch eine Approximation höherer Ordnung zu erzielen.

Die Vorsteuerung soll allerdings nur aktiviert werden, wenn die Abgastemperatur signifikant unter ihrem Mindestwert liegt, wenn das Verbrennungsluftverhältnis über der Untergrenze liegt und wenn das WG im linearen Teil des Arbeitsbereiches liegt (vgl. Abb. 3.2). Es ergibt sich für die Vorsteuerung folgender allgemeiner Zusammenhang:

$$r_{\text{etv},\min}^* = \begin{cases} \max\{k_0, k_1 + k_2 W_{\text{eo}}\} & T_{40} < T_{40,\min} - T_{\text{ff},\text{ofs}} \wedge r_{\text{wg}} > r_{\text{ff},\text{act}} \wedge \lambda > \lambda_{\min} \\ r_{\text{etv},\min} & \text{sonst.} \end{cases} \quad (5.12)$$

Die Schaltschwelle für die Temperatur wird zu $T_{\text{ff},\text{ofs}} := 30 \text{ K}$ und die Mindestaktuierung des WG zu $r_{\text{ff},\text{act}} := 30 \%$ gewählt.

²³Korrektweise müsste eine Abhängigkeit zum Volumenstrom gewählt werden. Bei der Berechnung des Volumenstroms aus dem Abgasmassenstrom ist jedoch durch den Druck zu dividieren, was diesen Ausdruck *empfindlich* gegenüber Druckschwankungen werden lässt.

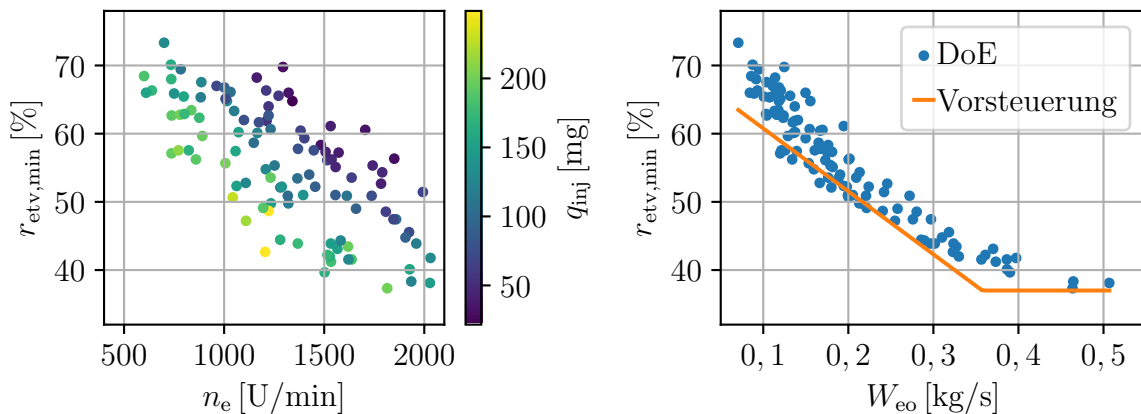


Abb. 5.12: ETV-Vorsteuerung. Die Abbildungen basieren auf der DoE (vgl. Abschnitt 3.2.1), gefiltert mit $p_0 = 1 \pm 0,1$ bar, $r_{\text{wg}} = 0\%$ und $0,07$ bar $< p_{40} - p_{41} < 0,1$ bar.

5.4.2 Detailbetrachtung

Das Verhalten der MPC im geschlossenen Regelkreis ist in Abb. 5.13 dargestellt. Der Drehzahlregler stellt die Sollzahl $n_{\text{eng,s}}$ unter Verwendung der Stellgröße q_{inj} ein. Aus Sicht des Drehzahlreglers nimmt das Lastmoment M_1 die Rolle einer Störgröße ein (vgl. Abb. 5.1). Die Drehzahl ist anfangs konstant bei 1200 U/min, das Lastmoment liegt bei 800 Nm. Die Mindesttemperatur wird in diesem Bereich eingehalten und der Ladedrucksollwert liegt über dem Istwert. Entsprechend ist das WG geschlossen und die ETV geöffnet. Auf diese Weise wird der maximale Ladedruck in dem Betriebspunkt erreicht. Ab $t = 10$ s setzt eine lineare Reduzierung des Lastmoments ein, was dazu führt, dass die Einspritzmenge q_{inj} und damit auch der nötige Ladedruck $p_{22,s}$ sinken, um ein Verbrennungsluftverhältnis $\lambda = 2$ zu erreichen. Bei $t \approx 13,4$ s ist der Ladedrucksollwert schließlich so weit abgesunken, dass er den Istwert schneidet; das gewünschte Verbrennungsluftverhältnis wird erreicht. Dies ist der Punkt, ab dem der Regler das WG öffnet, um den Istwert dem Sollwert nachzuführen. Der Öffnungsvorgang geschieht allerdings zu langsam, als dass der Istwert ohne Regelabweichung nachgeführt werden könnte. Dafür sind zwei Faktoren verantwortlich: Ein Grund besteht darin, dass das Drehmoment sukzessive reduziert wird, der Regler die Stellgröße aber so wählt, als bliebe das Lastmoment konstant. Der zweite Grund ist, dass die ETV ebenfalls zur Absenkung des Ladedrucks geeignet wäre. Die Nutzung der ETV für diesen Zweck wird durch die dynamische Nebenbedingung $\zeta_{\text{etv}} \leq \zeta_{\text{etv,max}}$ (vgl. Gleichung (5.9)) unterbunden. Als ungewollter Nebeneffekt der aktiven ETV-Nebenbedingung wird dadurch die WG-Bewegung verlangsamt. Ab $t \approx 21$ s kommt die Abgastemperatur T_{40} der Mindesttemperatur so nahe, dass die dynamische ETV-Begrenzung gelockert wird. Die Nebenbedingung ist beim Durchtritt der Mindesttemperatur $T_{40,\text{min}}$ vollständig aufgehoben. Dies wäre eigentlich der Zeitpunkt, ab dem die ETV geschlossen werden müsste, um die Mindesttemperatur einzuhalten. Aufgrund der in Abschnitt 5.3.2 bereits festgestellten geringen Konvergenzgeschwindigkeit für kleine ETV-Aktuierungen ist jedoch keine signifikante ETV-Bewegung sichtbar. Dies ändert sich bei Unterschreiten der Mindesttemperatur um 30°C bei $t \approx 26,8$ s. Zu diesem Zeitpunkt

wird die ETV-Vorsteuerung aktiv und hebt die ETV-Position schlagartig auf 56 % an. Ausgehend von dieser Aktuierung führt der Regler die Abgastemperatur in Richtung des Sollwertes, bis dieser bei $t \approx 30$ s etwa vier Sekunden nach Aktivwerden der Vorsteuerung erreicht wird. Es tritt ab diesem Zeitpunkt die gleiche Situation wie zu Anfang der Simulation auf: Die Mindestabgastemperatur ist überschritten, der Ladedruck liegt über dem Sollwert und das WG ist vollständig geöffnet. Einem weiteren Schließen der ETV wird durch die dynamische ETV-Begrenzung entgegengewirkt. Durch diesen Mechanismus wird die ETV bei dem ab $t = 30$ s steigenden Lastmoment wieder geöffnet, da durch die steigende Abgastemperatur weniger ETV-Aktuierung erforderlich ist.

Bei $t = 50$ s wird das Lastmoment schlagartig auf 1200 Nm angehoben. Um die Drehzahl zu halten, wird die Einspritzmenge auf die stöchiometrisch mögliche Menge erhöht. Dies führt zu einem Unterschreiten der unteren Grenze des Verbrennungsluftverhältnisses λ_{\min} . Um diese Grenzverletzung zu beheben, muss Ladedruck aufgebaut werden, was der Regler durch sofortiges Schließen des WG bewirkt. Bei $t = 60$ s kommt es durch einen Sprung des Drehzahl-Sollwertes auf $n_{\text{eng,s}} = 2000$ U/min zu derselben Situation. Der Drehzahlregler dosiert die maximale Kraftstoffmenge ein, was erneut zu einem Unterschreiten des Verbrennungsluftverhältnisses führt. Das WG ist zu diesem Zeitpunkt bereits geschlossen, weswegen reglerseitig keine Reaktion zu beobachten ist. Anders ist dies bei Erreichen der Solldrehzahl bei $t \approx 62,3$ s. Zu diesem Zeitpunkt fällt die Einspritzmenge ab und der Ladedrucksollwert fällt unter den Istwert. Als Reaktion darauf wird das WG maximal geöffnet und anschließend auf den stationär erforderlichen Wert eingestellt. An dieser Stelle ist keine Verzögerung des WG wie bei $13 \text{ s} \leq t \leq 20 \text{ s}$ zu beobachten. Die Ursache liegt mutmaßlich in dem deutlich höheren Druck in der Auslassflut p_3 . Dadurch hat das WG einen größeren Einfluss auf das Gesamtsystem und die Optimierung gelangt einfacher zu einer Lösung. Bei $t = 80$ s springt der Drehzahlsollwert auf 1500 U/min zurück. Durch das sofortige Ausbleiben der Einspritzung fällt die Abgastemperatur unter den Mindestwert und die ETV wird durch die Vorsteuerung direkt auf 37,3 % Auslenkung angehoben. Das WG zeigt keine Reaktion, obwohl der Ladedrucksollwert unter dem Istwert liegt. Dies ist durch die dynamische Gewichtung zwischen Abgastemperatur und Ladedruckregelung begründet (vgl. Gleichung (5.8a)). Durch diese liegt zusammen mit der Wahl des Gewichtungsfaktors $\gamma = 10^{-3}$ der Fokus kurzzeitig fast vollständig auf der Regelung der Abgastemperatur. Mit beginnendem Einsetzen der Einspritzung durch das Erreichen der Solldrehzahl bei $t \approx 80,7$ s ist die Mindesttemperatur wieder erfüllt und der Regler stellt durch kurzzeitiges Schließen des WG mit anschließendem Einstellen des Stationärwertes den Ladedruck auf den Sollwert ein.

Das Verhalten des Reglers ohne eine Vorsteuerung der ETV ist in Abb. B.7 dargestellt. Ohne die Vorsteuerung kommt die ETV nicht zum Einsatz. Die Zeiträume sind zu klein, um die ETV auch ohne die Vorsteuerung in einen Bereich zu bringen, ab dem diese schnell eingesetzt wird. In den übrigen Regionen ist das Verhalten identisch zu dem Regler mit ETV-Vorsteuerung.

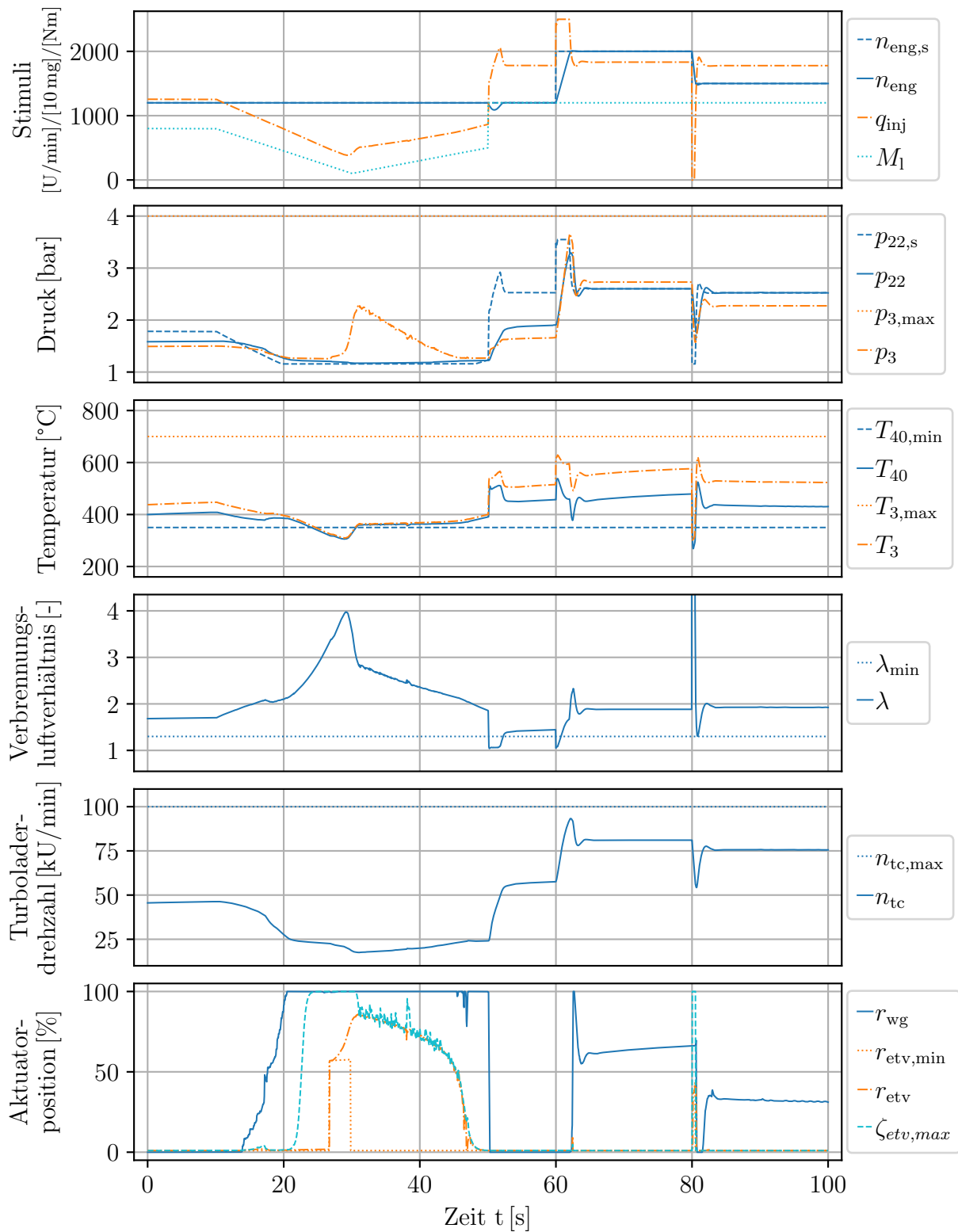


Abb. 5.13: Verhalten des finalen Reglers unter Verwendung der ETV-Vorsteuerung.

5.4.3 Simulation des NRTC-Testzyklus

Anhand des NRTC (vgl. Anhang C) soll der Regler simulativ erprobt werden. Dabei wird zwischen zwei verschiedenen Situationen differenziert: Im ersten Fall wird die Regelung ohne Temperaturziel betrachtet. Dieser spiegelt die Situation eines heißen AGN-Systems wider, welches keine Temperaturanforderung an die Motorregelung stellt und damit keinen Einfluss auf die motorinternen Regelungsziele ausübt.²⁴ Der zweite Fall behandelt schließlich die Situation mit aktiver Temperaturanforderung an den Motor. Dieser Fall liegt der zuvor durchgeführten Detailbetrachtung zugrunde.

Ladedruckregelung ohne gleichzeitige Regelung der Mindestabgastemperatur

In Abb. 5.14 ist ein Ausschnitt aus dem NRTC dargestellt.²⁵ Dieser ist so gewählt, dass sich die Motordrehzahlwerte darin im oberen Bereich befinden und das Drehmoment den gesamten Wertebereich abdeckt. In der ruhigeren rechten Hälfte des Ausschnitts liegt das Drehmoment im Bereich um 1450 Nm. Übertragen auf die Detailbetrachtung in Abschnitt 5.4.2 handelt es sich folglich um die Bereiche, in denen die Regelung besser funktioniert und das WG kein gedämpftes Verhalten zeigt.

In der ersten Hälfte der Darstellung pendelt das Drehmoment zwischen großen und kleinen Werten. Dieses Pendeln resultiert zur Einregelung von $\lambda = 2$ in entsprechend schwankenden Ladedrucksollwerten. Zu Anfang liegt der Ladedrucksollwert $p_{22,s}$ unter dem Istwert p_{22} . Durch maximale Öffnung des WG wird der Druck so weit wie möglich abgesenkt. Eine weitere Absenkung wäre durch ein Schließen der ETV möglich, dies wird jedoch durch die dynamische ETV-Limitierung verhindert. Bei $t = 596$ s steigt der Sollwert über den Istwert, bei gleichzeitiger Verletzung der unteren Grenze des Verbrennungsluftverhältnisses. Das WG wird als Reaktion darauf innerhalb von 150 ms vollständig geschlossen. Direkt im Anschluss fällt der Sollwert wieder unter den Istwert und das WG wird erneut vollständig geöffnet. Zwischen $600 \text{ s} \leq t \leq 605 \text{ s}$ befindet sich das WG im aktiven Bereich. Es ist zu sehen, wie der Ladedruck dem Sollwert in verschobener Weise folgt. Die Verschiebung beträgt ca. 600 ms und ist maßgeblich durch die Trägheit des Systems vorgegeben. Dies wird durch einen Vergleich der Bereiche mit Vollaktuierung deutlich.

Die zweite Hälfte ist durch näherungsweise stationäre Drehzahl- und Drehmomentvorgaben geprägt. Das Drehmoment schwingt mit einer Periodendauer von ca. zwei Sekunden. Dieselbe Kontur vollführt der Ladedrucksollwert. Es ist zu sehen, wie die Regelung dem Sollwert nahezu ohne Abweichung folgen kann und das WG zu der Kontur passende Bewegungen vollführt.

Es werden während des gesamten Zyklus keine Nebenbedingungen verletzt, bis auf λ_{\min}

²⁴Das Zusammenspiel zwischen AGN und Motor wurde in Abschnitt 1.1 skizziert.

²⁵Der vollständige Zyklus ist in Abb. B.8 zu finden.

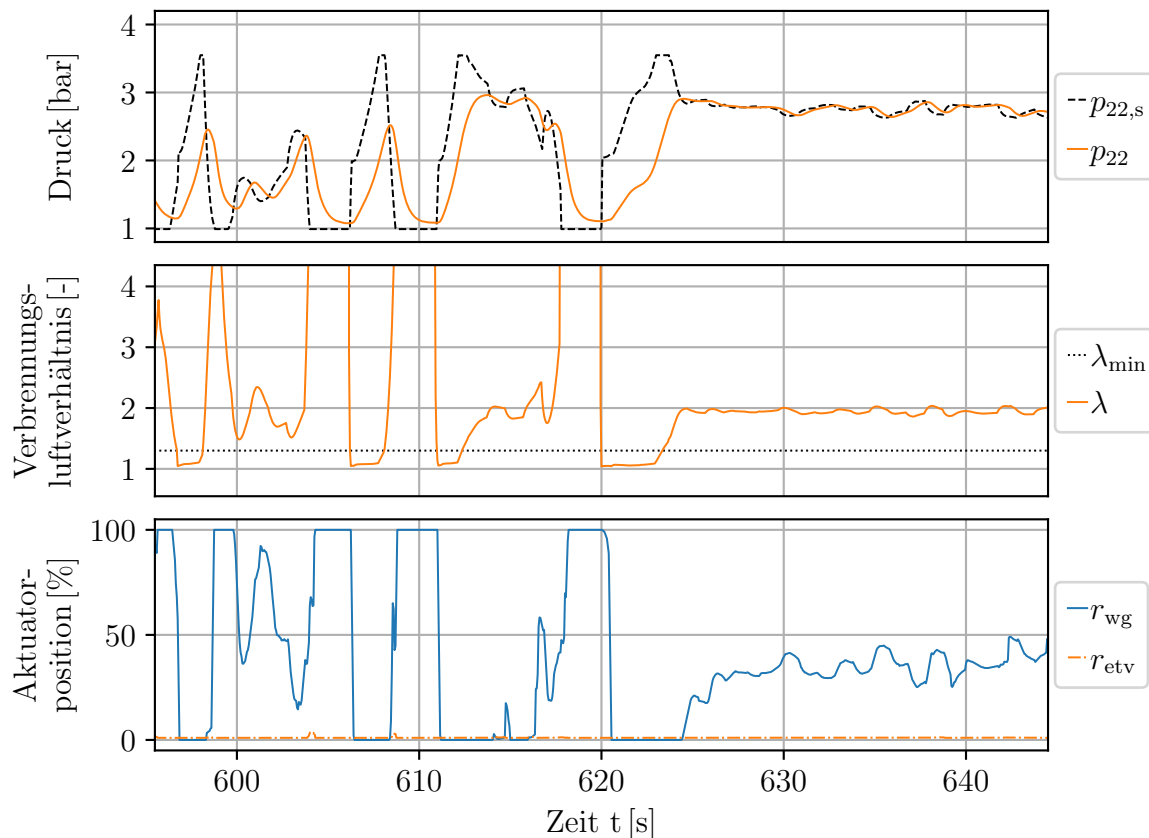


Abb. 5.14: Simulative Ladedruckregelung ohne gleichzeitige Regelung der Mindestabgastemperatur. Der vollständige simulierte NRTC ist in Abb. B.8 dargestellt.

und $\zeta_{\text{etv},\text{min}}$ (vgl. Abb. B.8). Die Verletzung der ersten Bedingung geht auf das dynamische Verhalten des Drehzahlreglers zurück. Phanomenologisch liegt zu diesen Zeitpunkten ein Lastsprung bzw. Drehzahlsprung vor, der in einer Maximalaussteuerung der Einspritzmenge resultiert. Der Drehzahlregler ist in der Simulation so konfiguriert, dass eine Kraftstoffeinspritzung fur $\lambda \geq 1,05$ erfolgen darf. Die zweite Bedingung ist zeitvariabel und wird deshalb im Rahmen der Konvergenzgeschwindigkeit des Algorithmus und der maximalen Bewegungsgeschwindigkeit der ETV verletzt. Es liegt damit keine Verletzung von solchen Nebenbedingungen vor, die Auswirkungen auf die Lebenserwartung des Motors haben.

Regelung der Mindestabgastemperatur und des Ladedrucks

Anders sieht dies bei zusatzlicher Regelung der Mindesttemperatur $T_{40,\text{min}} = 420^\circ\text{C}$ aus. Der gesamte NRTC ist in Abb. B.9 dargestellt. Der zulassige Druck in der Auslassflut p_3 wird bei $t \approx 270\text{ s}$ und bei $t \approx 929\text{ s}$ aufgrund der Dynamik des Profils im Zusammenspiel mit der ETV-Nutzung kurzzeitig verletzt. Es handelt sich dabei jeweils um Ereignisse von weniger als einer halben Sekunde Dauer. Die maximale Abgastemperatur $T_{3,\text{max}}$ sowie

die maximale Turboladerdrehzahl $n_{tc,max}$ werden zu keinem Zeitpunkt überschritten. Wie bei der alleinigen Ladedruckregelung wird das Verbrennungsluftverhältnis auch hier bei Laständerungen verletzt.

In einem qualitativen Vergleich mit der Regelung ohne Temperaturziel (vgl. Abb. B.8) fällt zunächst bezüglich der Einhaltung der Mindesttemperatur auf, dass gerade im zweiten Teil des NRTC mit hoher Motordrehzahl die Temperatureinbrüche gegenüber der Regelung ohne Temperaturziel durch das Schließen der ETV vermieden werden können. Im Bereich $800\text{ s} \leq t \leq 940\text{ s}$ ist kein Absacken der Abgastemperatur zu beobachten. In den Bereichen mit niedriger Motordrehzahl $160\text{ s} \leq t \leq 230\text{ s}$ kann der Temperaturabfall selbst bei einer Aussteuerung der ETV von 98 % nicht verhindert werden. Jedoch fällt der Abfall geringer aus als bei ausschließlicher Ladedruckregelung.

Die mittlere Aktuierung der ETV liegt bei 34,4 %, die des WG in einem ähnlichen Bereich von 38,2 %. Bei einer reinen Ladedruckregelung beträgt letztere 62,4 %. In den Bereichen, in denen im NRTC ohne Temperaturziel die Mindesttemperatur von 420 °C für einen längeren Bereich überschritten wird, findet primär das WG Anwendung. Dies ist beispielsweise für $650\text{ s} \leq t \leq 720\text{ s}$ der Fall. In anderen Bereichen, in denen das Temperaturziel ebenfalls erreicht wird, aber lediglich für einen kurzen Zeitraum, findet kein Wechsel vom ETV auf das WG statt; beispielsweise im Bereich $240\text{ s} \leq t \leq 290\text{ s}$.

In Abb. 5.15 sind die zuvor besprochene Ladedruckregelung und die hier besprochene Regelung auf eine Mindesttemperatur für einen Ausschnitt des NRTC gegenübergestellt. Im Bereich $t < 770\text{ s}$ erreicht die Ladedruckregelung durch den Betriebspunkt bereits die Zieltemperatur der Temperaturregelung. Es wäre zu erwarten, dass der Temperaturregler ein identisches Verhalten zeigt. Näherungsweise ist dies der Fall, jedoch zeigt der Temperaturregler eine unruhigere Nutzung des WG und eine Aktuierung der ETV im unteren Bereich. Die Nutzung der ETV hat keine relevante Auswirkung auf das System; die Einspritzmengen sind identisch. Bei $t = 700\text{ s}$ findet ein Drehmomentabfall statt, der zu einem Einbruch der Einspritzmenge und damit zu einem Einbruch der Abgastemperatur (im unregelmäßigen Fall) um ca. 100 °C führt. Sowohl die Ladedruck- als auch die Temperaturregelung öffnen das WG. Letztere schließt, beschleunigt durch die Vorsteuerung, zusätzlich die ETV, um die Abgastemperatur zu erhöhen. Die Abgastemperatur sackt für ca. $2,5\text{ s}$ unter die Zieltemperatur von 420 °C ab, danach wird der Zielwert erreicht. Bei $t = 777\text{ s}$ steigen sowohl die Drehzahl als auch das Drehmoment. Die Steigerung führt zu einer Erhöhung der Motorlast, welche eine höhere Abgastemperatur und eine Erhöhung des Ladedrucksollwertes zur Folge hat. In diesem Bereich ist die Mindesttemperatur ohne Zutun der ETV erfüllt und der Regler schließt das WG, um dem Ladedruck zu folgen. Bei $t = 779\text{ s}$ erreichen die Stimuli die Endwerte, die den Werten am Anfang des betrachteten Ausschnitts entsprechen. Es wäre zu erwarten, dass das WG und die ETV ähnliche Stellungen einnehmen. Das WG und die ETV verharren stattdessen in einer mittleren Stellung, was ein näherungsweise Erreichen der Mindesttemperatur zur Folge hat, sowie ein Überschreiten des Ladedrucks. Es wäre jedoch ein Öffnen des WG zu erwarten, um den Ladedruck zu senken. Diesbezüglich kann gemutmaßt werden, dass an dieser Stelle die dynamische Gewichtung der Gütefunktion 5.8a aufgrund der vorhandenen Regelabweichung der Abgastemperatur

zu einer Ausblendung des Druck-Regelziels führt. Im weiteren Verlauf des Ausschnitts tritt dieser Fall nicht erneut auf. Durch einen Vergleich der Einspritzmengen beider Regelungen wird deutlich, dass die Temperaturerhöhung auf Kosten des Kraftstoffverbrauchs erreicht wird.

5.4.4 Prüfstandsmessung des NRTC-Testzyklus

Der auf dem Steuergerät implementierte Regler folgt dem Schema von Abb. 5.2. Um die Zustände des Modells in die MPC zurückzuführen und auf diese Weise den Regelkreis zu schließen, wird typischerweise ein Zustandsbeobachter (vgl. Abb. 4.2) verwendet. Die Prüfstandsmessungen dienen jedoch lediglich dem Ziel, die Umsetzungsfähigkeit des Reglers auf der ECU zu zeigen. In diesem Sinne wurde auf den Entwurf eines Zustandsbeobachters verzichtet. Anstelle eines Beobachters wird ausschließlich der Messwert des Ladedrucksensors zurückgeführt. Auf dem Motorsteuergerät stehen neben dem Ladedrucksensor ein Drucksensor in der Auslassflut sowie ein Temperatursensor in der Einlassflut zur Verfügung, die zur Stützung der MPC infrage kommen. Die Temperatur scheidet aufgrund der im Vergleich zur Modelldynamik großen Trägheit des Temperatursensors aus.²⁶ Der Drucksensor in der Auslassflut wird aufgrund der besseren Korrektoreigenschaften einer alleinigen Stützung mit dem Ladedrucksensor nicht verwendet. In Abschnitt 3.2.4 wird die Modellstützung mit unterschiedlichen Sensoren besprochen. Von den verfügbaren Sensoren schneidet der Ladedrucksensor am besten ab, besser als eine Kombination beider Sensoren.

Implementierungstechnische Unterschiede zum simulativ erprobten Regler liegen in dem Verzicht auf eine ETV-Vorsteuerung sowie dem Fehlen der dynamischen Gewichtung der Regelungsziele in (5.8a). Davon abgesehen wird der Regler zudem langsamer ausgelegt, mit einer zusätzlichen Begrenzung der maximalen Änderungsrate der ETV auf 20 %. Die Änderungen dienen dazu, sicherheitskritische Situationen durch ein Versagen des Regelungsalgorithmus zu vermeiden. Neben der genannten, den Regler betreffenden Unterschiede wird außerdem ein Luftpfadmodell ohne globale Optimierung verwendet (vgl. Abb. 3.5), mit der Folge einer damit verbundenen geringeren Genauigkeit und einer folglich zu erwartenden geringeren Regelgüte.

Der Ladedrucksollwert wird der Seriensoftware entnommen. Dort nimmt dieser in Situationen, in denen zur Temperatursteigerung ein geöffnetes WG verlangt wird, unerreichbar kleine Werte an. An den übrigen Stellen werden Drücke verwendet, die im Kontext anderer Zielsetzungen als sinnvoll erachtet wurden. In der Simulation wurde der Ladedruck stets so gewählt, dass stationär näherungsweise ein Verbrennungsluftverhältnis von $\lambda = 2$ erreicht wird.

Im Folgenden soll analog zu dem vorigen Abschnitt 5.4.3 zunächst die Regelung des La-

²⁶Die Zeitkonstante des Temperatursensors kann mit $\tau = 10$ s abgeschätzt werden.

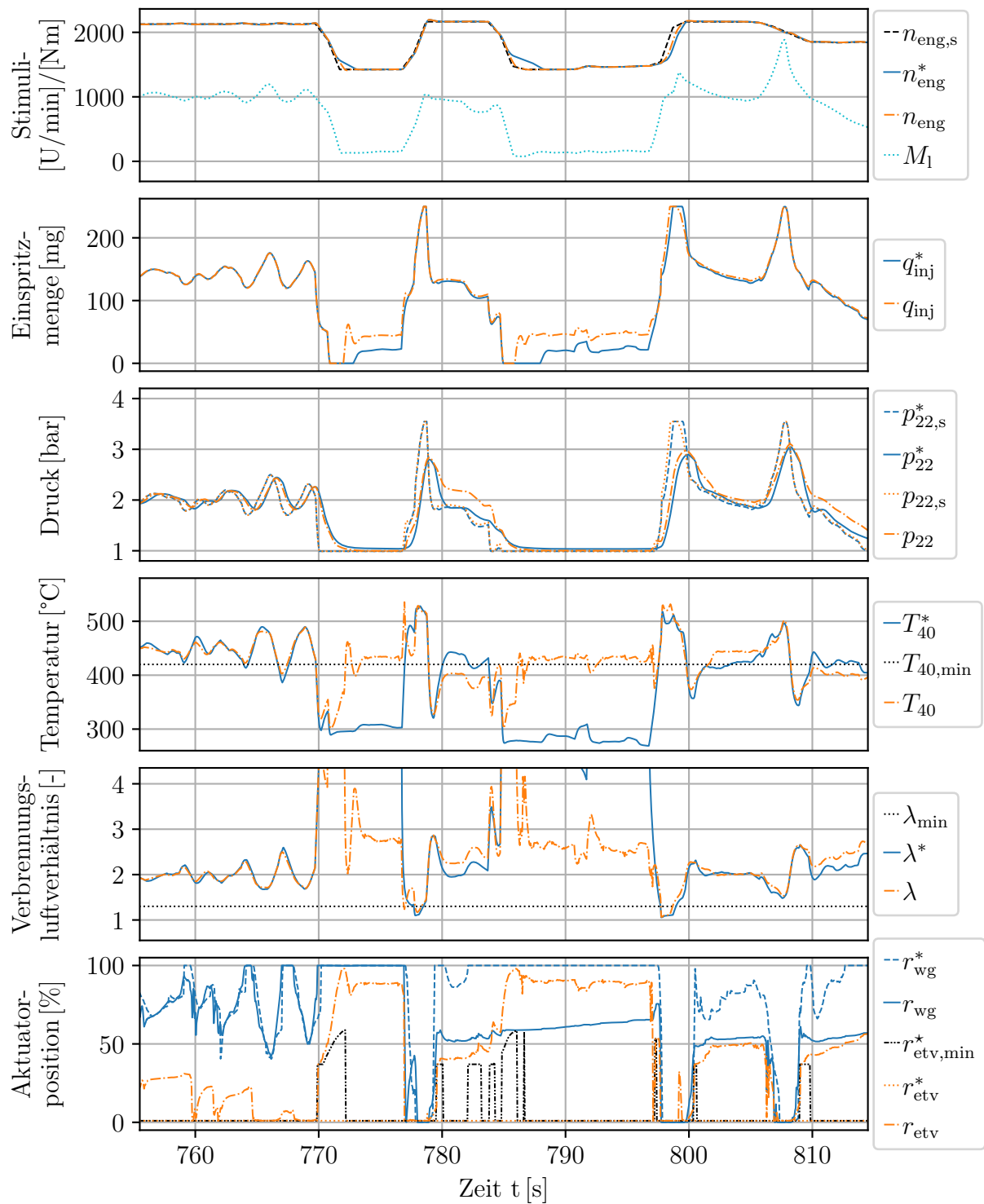


Abb. 5.15: Simulative Regelung der Mindestabgastemperatur und des Ladedrucks im Vergleich mit einer reinen Ladedruckregelung. Die Größen ohne Temperaturziel sind mit einem * gekennzeichnet.

dedrucks betrachtet werden. Anschließend wird eine Situation mit aktiver Nutzung der ETV besprochen. Beide Darstellungen entstammen derselben Prüfstandsmessung. In beiden Darstellungen wird zur besseren Beurteilung das Verhalten der Seriensoftware ohne Nutzung der ETV hinzugenommen.

Ladedruckregelung ohne gleichzeitige Regelung der Mindestabgastemperatur

Zur Beurteilung der Ladedruckregelung wird der gleiche Ausschnitt des NRTC betrachtet, der im vorigen Abschnitt bereits in einer Simulation beurteilt wurde (vgl. Abb. 5.14). In Abb. 5.16 ist zu sehen, wie sich die MPC und die Seriensoftware im Grundsatz ähnlich verhalten, jedoch ist die MPC insgesamt langsamer, was auf die konservativere Parametrierung der Regelung zurückzuführen ist. Insbesondere die von der Seriensoftware bei $t = 597$ s durchgeführte Öffnung des WG erfolgt seitens der MPC zwei Sekunden später. Die in diesem Ausschnitt durchgeführten Bewegungen der ETV befinden sich in einem Bereich, der keine relevanten Auswirkungen auf das Systemverhalten hat. Es kann festgestellt werden, dass die MPC das WG qualitativ identisch zur Seriensoftware aktuiert.

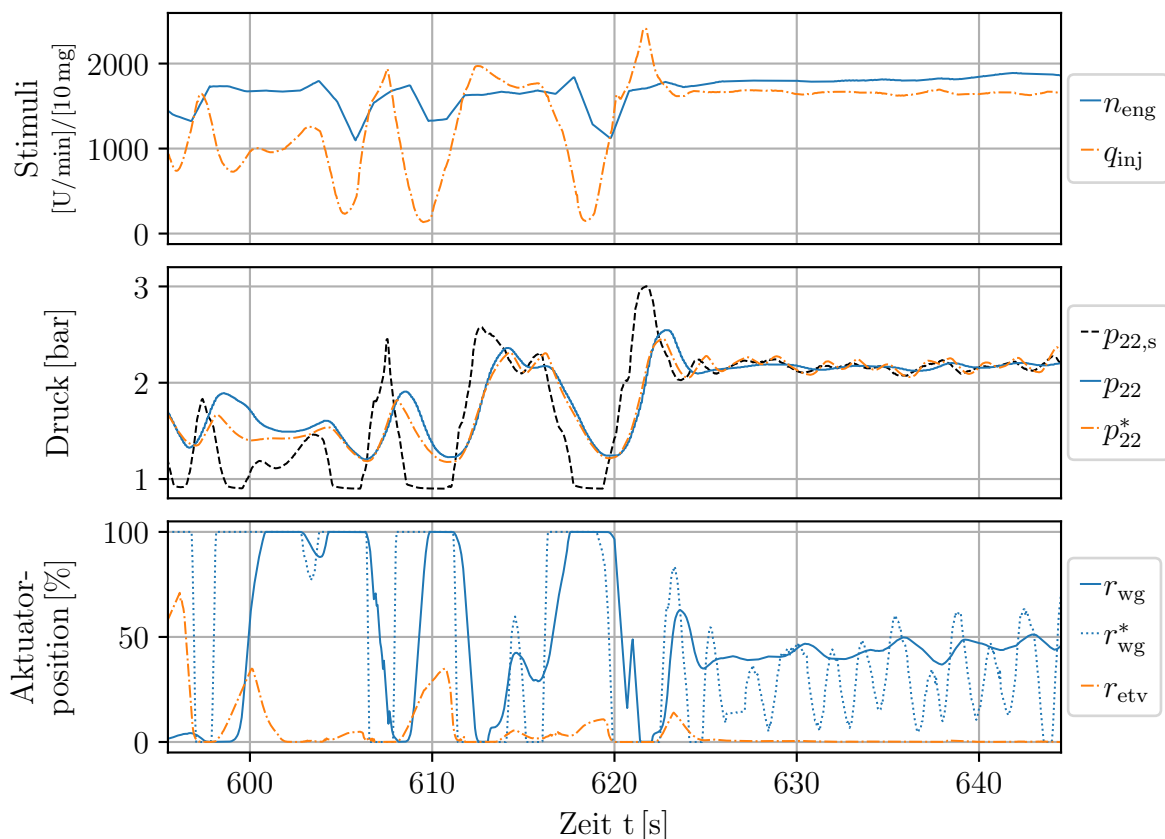


Abb. 5.16: Verhalten der MPC am Motorenprüfstand im Vergleich mit der Seriensoftware. Die mit der Seriensoftware gemessenen Größen sind mit einem * im Exponenten gekennzeichnet.

Ein Ausschnitt, der einen Zeitbereich 40 s später abbildet, ist in Abb. B.10 gegeben. Es ist zu sehen, wie sich eine bleibende Regelabweichung zwischen dem Ladedrucksollwert und dem Istwert herausbildet. Technisch ist dieser durch das Erreichen der Aussteuerungsgrenze des WG begründet. Jedoch führt die MPC den Aktuator nicht vollständig an diese Grenze heran. Begründen lässt sich diese Abweichung mit einem Modellfehler bzw. dem Nichtvorhandensein eines Fehlermodells zu dessen Kompensation.

Regelung der Mindestabgastemperatur und des Ladedrucks

Die Temperaturregelung kann nicht anhand des gleichen Zeitabschnitts des NRTC beurteilt werden wie die Simulation (vgl. Abb. 5.15), weil die MPC in diesem Bereich zu lange braucht, um die ETV in einen relevanten Wirkungsbereich zu führen (vgl. Abb. B.11). Stattdessen wird in Abb. 5.17 ein anderer Zeitabschnitt betrachtet. Es ist zu sehen, wie ab $t = 1097$ s die ETV mit der Maximalrate von 20 % geschlossen wird, bis die Abgastemperatur in einem Bereich unterhalb des Sollwertes liegt. Die Schwingbewegung entsteht durch Drehmomentänderungen derselben Kontur. Dies kann anhand einer Simulation der gleichen Stelle in Abb. B.12 nachvollzogen werden. Dort ist das Schwingen ebenfalls zu beobachten. Es sei darauf hingewiesen, dass es sich in der Darstellung bei T_{40} um eine modellierte Temperatur handelt. Der Vollständigkeit halber ist in der Abbildung die unmittelbar vor den AGN-Systemen gemessene Temperatur T_5 dargestellt. Durch die Sensorträgheit kann diese nicht zur Beurteilung der Dynamik herangezogen werden.

5.5 Bewertung und Möglichkeiten zur Weiterentwicklung

Wie in den vorausgegangenen Abschnitten sowohl simulativ als auch anhand einer Prüfstandsaufzeichnung gezeigt wurde, ist es möglich, den Deutz TCD 12.0 V6 Dieselmotor hinsichtlich der Einhaltung einer Mindesttemperatur zu regeln. Auf dem Motor selbst wurde der Regler parallel zur existierenden Seriensoftware umgesetzt. Die ECU stand mit ihren Ressourcen folglich nicht exklusiv der MPC zur Verfügung. Den daraus resultierenden Anforderungen ist die MPC gerecht geworden, womit die Voraussetzungen für eine kommerzielle Umsetzung gegeben sind.

5.5.1 Bewertung der Regelung

In Abschnitt 5.1.3 und 5.1.4 wurde gezeigt, dass es sich bei der Regelung von Ladedruck und Abgastemperatur über ein WG und eine ETV um ein stark gekoppeltes System handelt. Es ist damit nicht möglich, die Größen getrennt voneinander zu regeln – beide Aktuatoren

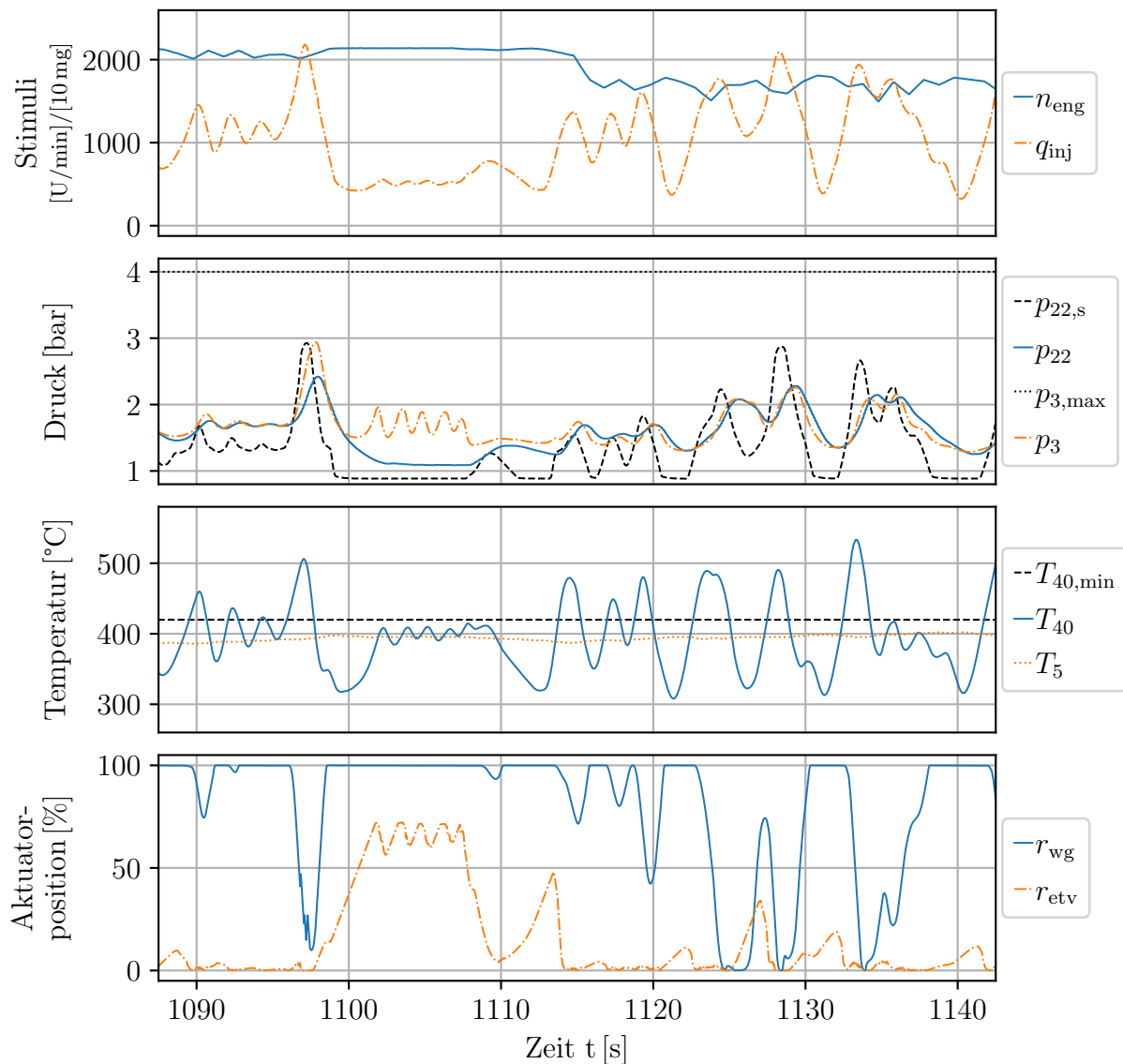


Abb. 5.17: Prüfstandsaufzeichnung der Regelung auf eine Mindestabgastemperatur und auf einen Ladedruck.

haben auf beide Größen dieselbe Wirkung, allerdings in unterschiedlich starker Weise. Gemäß dieser Eigenschaft ist durch den Regler eine Entscheidung zu treffen, welche Größe die *Hauptregelgröße* ist und welcher Aktuator dafür verwendet werden soll. Es wurde gezeigt, dass die ETV den Kraftstoffverbrauch erhöht und daher soweit wie möglich das WG der bevorzugte Aktuator ist.

Die Entscheidung, welches die Hauptregelgröße ist, wird in der Gütefunktion (5.8a) über eine dynamische Gewichtung der Regelabweichungen in Kombination mit einem Gewichtungsparemeter umgesetzt. In den Bereichen, in denen das Temperaturziel unterschritten wird, nehmen die Aktuatoren Positionen ein, die zu einer Erhöhung der Abgastemperatur

führen. Der Ladedruck wird umso stärker vernachlässigt, je weiter die Abgastemperatur von dem Sollwert entfernt ist. Über diese Formulierung der Gütefunktion ist eine Moderation der Regelungsziele möglich. Nachteilig an dieser Formulierung sind Situationen, in denen gleichzeitig die Abgastemperatur und der Ladedruck Regelabweichungen aufweisen. In diesen Situationen wird vorrangig die Abgastemperatur berücksichtigt. Erst wenn deren Regelabweichung klein genug ist, wird zusätzlich der Ladedruck berücksichtigt. In Abb. 5.15 im Bereich $780\text{ s} \leq t \leq 785\text{ s}$ ist zu sehen, wie aufgrund einer nicht vollständigen Erreichung des Abgastemperatursollwertes das Ladedruckregelungsziel vernachlässigt wird, obwohl dieses durch ein Öffnen des WG erreichbar wäre. Das Öffnen des WG wäre dem Erreichen der Zielabgastemperatur ebenfalls förderlich.

Die Präferenzierung des WG gegenüber der ETV wird indirekt über eine Nebenbedingung (5.7f) umgesetzt. Die maximale Aktuierung wird bei Überschreiten der Mindesttemperatur eingeschränkt. Über diesen Mechanismus kann das Ziel jedoch lediglich teilweise erreicht werden. Zwar wird die ETV wie beabsichtigt nicht genutzt, wenn die Zieltemperatur überschritten wird (vgl. Abb. 5.13 u. Abb. 5.14). Problematisch sind jedoch Situationen, in denen die Zieltemperatur knapp erreicht wird und die ETV-Aktuierung gegen eine WG-Aktuierung ausgetauscht werden könnte. Dies ist in Abb. 5.15 bei $785\text{ s} \leq t \leq 797\text{ s}$ zu sehen: Die ETV wird mit geringer Geschwindigkeit gegen das WG ausgetauscht. Zu dieser Situation trägt die Tatsache bei, dass bei Aktuierung der ETV das Druckgefälle über die Turbine abnimmt und das WG infolgedessen an Wirksamkeit verliert.²⁷ Dies resultiert seitens des Lösungsalgorithmus in einer geringen Konvergenzgeschwindigkeit. Eine Einbeziehung der Aktuatorpräferenz in das Optimierungsproblem oder eine Bestrafung der Pumpverluste stellt keine geeignete Lösung dar. Dadurch sind bleibende Regelabweichungen zu erwarten, die vermieden werden sollen.

Ein weiterer Mechanismus, der das WG gegenüber der ETV bevorzugt, ist die Gestaltung der Vorsteuerung. In Abschnitt 5.3.2 wurde gezeigt, dass die ETV in Kombination mit dem entwickelten Lösungsalgorithmus für kleine Aktuierungen unter einer geringen Konvergenzgeschwindigkeit leidet. Durch die Vorsteuerung (vgl. Abschnitt 5.4.1) wird die ETV auf eine Mindestauslenkung gebracht, ab welcher der Algorithmus eine höhere Konvergenzgeschwindigkeit zeigt. Die geringere Konvergenzgeschwindigkeit wird im Sinne einer Bevorzugung des WG ausgenutzt: Die Vorsteuerung wird erst dann aktiviert, wenn sich das WG in einem Bereich mit erhöhter Verstärkung befindet. Zu beobachten ist die Schaltschwelle in Abb. 5.13 und in Abb. 5.15 bei $t = 779\text{ s}$. In beiden Darstellungen würde bei einer direkten Vorsteuerung ohne die Bedingung einer Mindestauslenkung des WG (vgl. Gleichung (5.12)) das WG bei geringer Aktuierung verharren.

Trotz der genannten Schwächen zeigt die Regelung insgesamt ein akzeptables Verhalten. Insbesondere etwaige Nachteile in der Dynamik durch zu wenig zur Verfügung stehende Ladeluft aufgrund einer Nichtbeachtung des Ladedruckregelungsziels können durch eine

²⁷Der WG-Massenstrom W_{wg} (3.39) hängt von dem Druckgefälle über der Turbine bzw. dem WG ab. Je geringer dieses Gefälle ist, desto geringer ist der Massenstrom und damit die Wirkung des WG auf das Gesamtsystem.

geeignete Wahl der unteren Grenze des Verbrennungsluftverhältnisses abgemildert bzw. vermieden werden. Beispielsweise werden in Abb. 5.15 bei $t = 797$ s bei Verletzung von λ_{\min} unmittelbar die ETV geöffnet und das WG geschlossen. Auch die übrigen Nebenbedingungen werden vom Regler berücksichtigt. Lediglich in Abb. B.9 wird in zwei, durch Änderung der Motorlast hervorgerufenen Situationen der Grenzwert für den Druck in der Auslassflut für ca. eine halbe Sekunde überschritten.

Ein direkter Vergleich der MPC mit der Seriensoftware zeigt, dass diese für die Ladedruckregelung ebenso gut geeignet ist. Beim Vergleich der Prüfstandsmessung (Abb. B.11) mit der zugehörigen Simulation (Abb. 5.15) wird deutlich, dass die MPC das Potential hat, die gleiche Regelgüte mit weniger Aktuatoreinsatz zu erreichen. Des Weiteren müssen die Sollwerte für den Ladedruck nicht defensiv bezüglich der Vermeidung von Drehzahl-grenzverletzungen gewählt werden, da die MPC prinzipbedingt solche Nebenbedingungen berücksichtigen kann.

5.5.2 Separate Auflösung des Zielkonfliktes

Die Güte der MPC leidet unter der Moderierung der Regelungsziele sowie der Festlegung auf einen Aktuator zur Laufzeit. Wünschenswert wäre es, über einen Mechanismus zu verfügen, der im Vorhinein erreichbare Sollwerte $\tilde{\mathbf{y}}_s$ und dazugehörige Stellgrößen $\tilde{\mathbf{u}}_s$ berechnet, in welche die Moderierung der Regelungsziele einfließt. In [MR93] wird vorgeschlagen, aus den Referenzwerten für die zu regelnden Größen \mathbf{y}_s unter Verwendung des Modells (vgl. (2.1)) die korrespondierenden Zustände und Eingänge des Systems für den stationären Fall zu berechnen. Eine Adaption dieses Ansatzes führt zu dem Optimierungsproblem

$$\min_{\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s} l(\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s, t_k; \mathbf{y}_s) \quad (5.13a)$$

$$\text{u. B. v. } \mathbf{f}(\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s) = \mathbf{0}, \quad (5.13b)$$

$$\mathbf{b}_l(t_k) \leq \tilde{\mathbf{u}}_s \leq \mathbf{b}_u(t_k), \quad (5.13c)$$

$$\mathbf{c}_{\mathcal{E}}(\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s, t_k) = 0, \quad (5.13d)$$

$$\mathbf{c}_{\mathcal{I}}(\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s, t_k) \leq 0. \quad (5.13e)$$

$$(5.13f)$$

Unter Verwendung der Ausgangsgleichung (2.1b) lassen sich die zugehörigen Referenzwerte

$$\tilde{\mathbf{y}}_s = \mathbf{g}(\tilde{\mathbf{x}}_s, \tilde{\mathbf{u}}_s)$$

für die MPC berechnen. Gleichung (5.13b) fordert stationäres Verhalten. Als Gütefunktion kommt das verwendete Gütefunktional (5.8a) zum Einsatz, wobei die Anteile zur Bestrafung der Aktuatorbewegung weggelassen werden können. Diese ist aufgrund der zuvor genannten Stationaritätsbedingung bereits Teil des Optimierungsproblems. Die Nebenbedingungen (5.13c - 5.13e) sind, wie das Gütefunktional, identisch zur MPC-Formulierung.

In einer Implementierung ist es sinnvoll, die Optimierung (5.13) vor jeder MPC-Berechnung durchzuführen und diese mit den aktuellen Zuständen der MPC zu initialisieren, damit sichergestellt ist, dass diese Lösung und die MPC gegen die gleiche Lösung konvergieren; die Abbildung der Ausgangsgleichung (2.1b) ist nicht bijektiv.

Dieses Vorgehen bietet wegen der Trennung zwischen Sollwertbildung und Reglerauslegung mehrere Vorteile: Bei der Reglerauslegung kann davon ausgegangen werden, dass sämtliche Sollwerte erreichbar sind. Entsprechend ist dort – wie beabsichtigt – kein Zielkonflikt mehr aufzulösen. Ebenfalls ist bei der Parameterwahl kein Kompromiss zwischen der Zielgewichtung und dem dynamischen Verhalten zu suchen, was einen leistungsfähigeren Regler verspricht. Im Hinblick auf den emissionsgeführten Motor (vgl. Abschnitt 1.1) treten die zuvor genannten Vorteile aufgrund der zusätzlichen Regelungsziele noch deutlicher hervor. Rechentechnisch ist (5.13) aufgrund der geringeren Größe des Problems leichter zu lösen als (5.7). Das Problem enthält weniger zu optimierende Variablen, was die Verwendung des Newton-Verfahrens mit seiner dem GV überlegenen Konvergenzgeschwindigkeit ermöglicht.

Daneben kann in diese Formulierung der Ansatz einer *offset-freien* MPC aus [PB07] eingearbeitet werden. In dem Ansatz wird parallel zur MPC ein Fehlermodell durch einen Beobachter geschätzt. Das Fehlermodell wird, analog zu dem präsentierten Schema, verwendet, um korrigierte Ersatzsollwerte bereitzustellen, welche eine Regelung ohne stationäre Abweichung ergeben. Es ist im Sinne der Betriebssicherheit ratsam, lediglich Störungen in der Ausgangsgleichung (2.1b), nicht jedoch in der Systemdynamik (2.1a) zu modellieren und wie in dem Beitrag vorgeschlagen zu schätzen.²⁸

5.5.3 Potential der Einbeziehung des Einspritzwinkels

Abseits der technischen Umsetzung der Regelung ist die Einbeziehung weiterer Stellgrößen förderlich. Beispielsweise kann durch Verstellung des in Abschnitt 5.1.2 besprochenen *Einspritzwinkels* die Abgastemperatur ebenfalls beeinflusst werden. Dieser Aktuator wäre in seiner Wirkung von den anderen Aktuatoren entkoppelt und würde teilweise sogar gegenteilige Effekte hervorrufen. Eine Verstellung des Einspritzwinkels derart, dass sich der Verbrennungswirkungsgrad verschlechtert, würde einerseits die Abgastemperatur erhöhen und andererseits nach (3.41) die Turbinenleistung und damit letztendlich den Ladedruck steigern. Sowohl beim WG als auch bei der ETV geht eine Steigerung der Abgastemperatur stets mit einer Absenkung des Ladedrucks einher. Der Einspritzwinkel bietet daher die Möglichkeit, durch gemeinsame Verstellung des Winkels und beispielsweise des WG den Ladedruck konstant zu halten und gleichzeitig die Abgastemperatur zu steigern. Allerdings wäre zu erforschen, inwiefern über eine Verstellung des Einspritzwinkels die Turbinenleistung gesteigert werden kann.

²⁸Die Stabilität der Kombination von MPC und Beobachter ist im nichtlinearen Fall nicht garantiert, selbst wenn beide Systeme für sich genommen stabil sind [AFN04].

6 Zusammenfassung und Ausblick

Der wesentliche Entwicklungstreiber moderner Dieselmotoren ist die Einhaltung der regulatorischen Grenzwerte hinsichtlich der Schadstoffemissionen. Waren die ersten gesetzlichen Grenzwerte noch durch innermotorische Maßnahmen realisierbar, sind die jüngeren Emissionsgrenzen lediglich unter Verwendung gesonderter Abgasnachbehandlungssysteme einzuhalten (vgl. Abb. 1.5). In diesen Nachbehandlungssystemen finden chemische Reaktionen statt, die von einer möglichst hohen Abgastemperatur profitieren. An dieser Stelle setzt das Konzept der in dieser Arbeit entwickelten temperaturgeführten Motorregelung an. In einer regelungstechnischen Umsetzung übermitteln die Abgasnachbehandlungssysteme eine Temperaturanforderung an den Dieselmotor in Form einer Mindesttemperatur. Die internen Regelungen des Dieselmotors berücksichtigen die Vorgabe und leisten auf diese Weise ihren Beitrag zur Gewährleistung der Leistungsfähigkeit der Abgasnachbehandlungssysteme und somit zur Einhaltung der Schadstoffemissionsgrenzen.

Eine der internen Regelungen des Motors ist die Einstellung der dem Motor zugeführten Ladeluftmenge über den Ladedruck. Zur Bereitstellung der Ladeluft dient der sogenannte Luftpfad, welcher als Stellgrößen ein Wastegate und eine Abgasklappe beinhaltet. Durch eine Aktuierung der Abgasklappe ist eine künstliche Verschlechterung des Motorwirkungsgrades möglich, welche zu einer Erhöhung der Abgastemperatur führt. Auch der Ladedruck wird hierdurch beeinflusst. Das Wastegate dient primär der Regelung des Ladedrucks, allerdings geht auch mit seiner Aktuierung eine Beeinflussung der Abgastemperatur einher. Es handelt sich mithin um ein gekoppeltes Mehrgrößensystem mit den zwei Regelgrößen Abgastemperatur und Ladedruck, wobei die Mindestabgastemperatur den Charakter einer Nebenbedingung hat. Neben dieser weichen Nebenbedingung sind systembedingte harte Nebenbedingungen zu beachten, bei deren Verletzung Systemschäden zu erwarten sind, beispielsweise bei einer Überschreitung der maximalen Turboladerdrehzahl. Abseits der genannten Aspekte ist ein möglichst geringer Kraftstoffverbrauch sowie die Sicherstellung einer ausreichenden Motordynamik von Bedeutung.

Zur Regelung des Luftpfades wird eine modellprädiktive Regelung eingesetzt. Dieses Regelungskonzept zeichnet sich durch seine Fähigkeit zur Mehrgrößenregelung und zur Berücksichtigung von Nebenbedingungen aus. Die umgesetzte Regelung ist eine sogenannte nichtlineare modellprädiktive Regelung, welche im Vergleich zu anderen Konzepten nicht auf einem Entwicklungsrechner vorberechnet werden kann, sondern zur Laufzeit auf dem Zielsystem zu berechnen ist. Erprobt werden soll sie an dem Deutz TCD 12.0 V6 Dieselmotor. Aus diesem Grund besteht eine Randbedingung bei der Umsetzung darin, diese Regelung auf einem gängigen von der Firma *Deutz AG* verbauten Motorsteuergerät betreiben zu können. Die Entwicklung einer auf dem Zielsystem umsetzbaren und rechentechnisch robusten Realisierung stellt demnach einen Schwerpunkt der Arbeit dar. Inhaltlich ergeben sich daraus zwei Themenkomplexe: zum einen die Entwicklung eines Luftpfadmodells, welches einerseits möglichst genau, aber andererseits hinsichtlich der Rechenanforderungen möglichst effizient ist, und zum anderen die Entwicklung eines robusten Lösungsalgorithmus.

mus, der die Ressourcen des Motorsteuergerätes nicht überfordert.

Der Luftpfad wird als sogenanntes Mittelwertmodell in Form einer nichtlinearen Zustandsraumbeschreibung formuliert. Bei seiner Beschreibung wird mithin nicht das Arbeitsspiel des Motors abgebildet. Im Sinne einer geringen Modellkomplexität wird mit konzentrierten Parametern gearbeitet, sodass der Luftpfad mit acht Zuständen beschrieben werden kann, wovon zwei Zustände zum Brechen algebraischer Schleifen benötigt werden. Im Grundsatz handelt es sich um eine physikalische Modellierung, jedoch werden zahlreiche Effekte durch anhand von CFD-Simulationen motivierte Polynome beschrieben. Zur Berechnung der stationären Abgastemperatur wird ein außerhalb des eigentlichen Luftpfadmodells ausgewertetes künstliches neuronales Netz verwendet. Aus dem auf diese Weise über Formeln beschriebenen Modell wird mit einem im Rahmen dieser Arbeit entwickelten Codegenerator auf dem Steuergerät lauffähiger C-Code erzeugt. Teil des Generierungsprozesses ist die automatische Berechnung der für den prädiktiven Lösungsalgorithmus erforderlichen Ableitungen für eine Linearisierung des Modells. Unter Verwendung eines *Runge-Kutta-4* Integrationsverfahrens wird auf dem Entwicklungsrechner ein Echtzeitfaktor von etwa 9035 bei einer Schrittweite von 10 ms erreicht; auf dem Zielsystem liegt dieser bei schätzungsweise 753.¹ Der unter Verwendung des generierten C-Codes erreichte RRMSE (engl.: *Relative Root Mean Square Error*) liegt stationär für sämtliche Drücke und Temperaturen sowie für die Turboladerdrehzahl bei weniger als 6 %, ausgewertet anhand der zur Ermittlung der Modellparameter verwendeten stationären Trainingsdaten. Die Trainingsdaten entstammen einer statistischen Versuchsplanung und bilden sämtliche Betriebspunkte des Motors ab. Das qualitative dynamische Verhalten stimmt mit der CFD-Simulation überein.

Der Regelungsalgorithmus löst das zur Berechnung der prädiktiven Regelung auszuwertende dynamische Optimierungsproblem unter Verwendung eines Gradientenverfahrens in Kombination mit der *erweiterten Lagrange-Methode*. Das Gradientenverfahren wird zur Lösung des eigentlichen dynamischen Optimierungsproblems verwendet, die erweiterte Lagrange-Methode dient zur Berücksichtigung von Nebenbedingungen. Hinsichtlich der Anforderungen eines eingebetteten Systems zeichnet sich dieser Ansatz durch eine geringe Speichernutzung aus, da keine Ableitungen zweiter Ordnung verwendet und somit die damit einhergehenden speicherintensiven Matrizen vermieden werden. Diese Lösungsstrategie stammt von *Englert et al.* [EVM⁺19]. Ohne entsprechende Erweiterungen ist das Verfahren betreffend der Entwicklungsziele (vgl. Tab. 1.2) zu speicherintensiv und nicht robust genug. Eine solche Erweiterung ist eine nicht-äquidistante Diskretisierung des Prädiktionshorizontes. Auf diese Weise lässt sich die Anzahl der Stützstellen gegen Ende des Horizonts ausdünnen, da dort weniger Regleraktivität zu erwarten ist, wodurch eine Reduzierung der Beanspruchung des RAM erreicht wird. Ebenfalls RAM-reduzierend wirkt sich die Einführung eines separaten Kontroll-Horizontes aus. Ein kürzerer Kontrollhorizont hat weniger zu optimierende und damit zu speichernde Variablen zur Folge. Zur Steigerung der Robustheit wird das Verfahren um eine iterative Liniensuche erweitert, die auf einem *Backtracking-*

¹Der Echtzeitfaktor gibt an, wie viel schneller als die Echtzeit simuliert werden kann. Beispielsweise bedeutet ein Faktor von zehn, dass unter Aufwendung einer Sekunde Rechenzeit zehn Sekunden Echtzeit simuliert werden.

Algorithmus aufsetzt. Es wird gezeigt, dass erst durch die Liniensuche eine zuverlässige Regelung möglich ist. Hinsichtlich der erforderlichen Rechenzeit wird bei einem Aufruf des Reglers alle 50 ms und einem Prädiktionshorizont von 1,5 s ein Echtzeitfaktor von 128 auf dem Entwicklungsrechner bzw. geschätzten 10,7 auf dem Motorsteuergerät erreicht, was einer CPU-Auslastung von etwa 9,3 % entspricht. Bezüglich des RAM werden 1,89 kB auf dem Stack und 0,37 kB auf dem Heap beansprucht. Damit werden sowohl hinsichtlich der Rechenzeit als auch hinsichtlich des beanspruchten RAM die Entwicklungsziele erfüllt.

Simulativ wird gezeigt, dass der Regler zur Einregelung des Ladedrucks bei gleichzeitiger Beachtung einer Mindesttemperatur geeignet ist. Der Regler verwendet beide Aktuatoren und bevorzugt – wenn möglich – eine Aktuierung des Wastegates, da dieses im Gegensatz zur Abgasklappe keine negativen Auswirkungen auf den Kraftstoffverbrauch hat. Des Weiteren werden sämtliche harten Nebenbedingungen eingehalten. Der Vergleich der Simulation des NRTC-Testzyklus mit einer Prüfstandsaufnahme desselben legt nahe, dass die Leistungsfähigkeit der prädiktiven Regelung hinsichtlich der Ladedruckregelung vergleichbar mit der Seriensoftware ist, dies jedoch unter reduziertem Einsatz der Aktuatoren. Es besteht damit das Potential, bei gleicher Regelgüte die Lebenserwartung der Aktuatoren zu verlängern. Der Einsatz des Reglers am Prüfstand erfolgte unter Berücksichtigung einer defensiveren Auslegung der Reglergeschwindigkeit, um mögliche Schäden durch eine Fehlfunktion zu vermeiden.² Dennoch kann gezeigt werden, dass der entwickelte Regler auch am Prüfstand plausible Ergebnisse liefert und demnach für den kommerziellen Einsatz infrage kommt.

Aus regelungstechnischer Sicht anspruchsvoll ist die Tatsache, dass beide Aktuatoren näherungsweise dieselben Folgen nach sich ziehen: Sowohl eine Aktuierung des Wastegates als auch eine Betätigung der Abgasklappe führen jeweils zu einer Absenkung des Ladedrucks und zu einer Erhöhung der Abgastemperatur. Somit handelt es sich in gewisser Weise um ein unteraktuiertes System hinsichtlich der getrennten Regelung des Ladedrucks und der Abgastemperatur. Daraus ergibt sich ein permanenter Zielkonflikt zwischen den beiden Regelgrößen, sofern die Mindestabgastemperatur nicht erreicht wird. Dieser Konflikt wird durch die Formulierung des Optimierungsproblems dahingehend gelöst, dass die Abgastemperatur bei seinem Auftreten eine höhere Priorität genießt. Diese Formulierung hat jedoch den Nachteil, dass in einer dynamischen Situation bei einer Verletzung der Mindesttemperatur zuerst diese eingeregelt wird und erst im Anschluss daran der Ladedruck Berücksichtigung findet. Es bietet sich daher an, im Rahmen einer möglichen Weiterentwicklung des Regelungskonzepts auf eine separate Sollwertgenerierung zurückzugreifen, und zwar derart, dass die Sollwerte von der Regelung erreichbar sind. Unter diesen Umständen entfielen die Notwendigkeit, die Regelungsziele innerhalb der prädiktiven Regelung moderieren zu müssen, und es könnten beide Regelgrößen gleichzeitig berücksichtigt werden.

Neben dem Wastegate und der Abgasklappe gibt es zudem weitere Stellgrößen, die im

²Insbesondere bei einer zu starken Aktuierung der Abgasklappe können hohe Drücke auftreten, die die Systemgrenzen überschreiten.

Rahmen einer Weiterentwicklung der in dieser Arbeit entwickelten temperaturgeführten Motorregelung zur Beeinflussung der Abgastemperatur herangezogen werden könnten. Dazu zählt insbesondere der Einspritzwinkel. Dieser gibt an, zu welchem Zeitpunkt bezüglich der Kolbenposition der Kraftstoff in den Brennraum gegeben wird. Der Zeitpunkt kann – aus der Perspektive des Motorwirkungsgrades – bewusst ungünstig gewählt werden, um so die Abgastemperatur zu steigern. Aus regelungstechnischer Sicht bringt diese Stellgröße den Vorteil mit sich, von den übrigen Aktuatoren entkoppelt zu sein. Eine Erhöhung der Abgastemperatur durch eine Verstellung des Einspritzwinkels führt zu keiner Reduzierung des Ladedrucks, sondern durch die gesteigerte Enthalpie im Abgas zu einer (leichten) Erhöhung.

Der *temperaturgeführte Motor* ist – über die zuvor genannten Weiterentwicklungsmöglichkeiten hinaus – zudem lediglich ein Zwischenschritt auf dem Weg zu einem *emissionsgeführten Motor*. Dieses weitergehende Regelungskonzept enthält zusätzlich zu der Abgastemperatur die Rohemissionen des Motors als weitere Regelgrößen. Im Rahmen dieses Konzeptes kämen weitere Stellgrößen zum Einsatz, wie beispielsweise der zuvor genannte Einspritzwinkel. Für den prädiktiven Regler hätte dies die Konsequenz, dass das zu lösende Optimierungsproblem durch die höhere Anzahl an Optimierungsvariablen an Komplexität gewinnt. Denkbar wäre es, dieser Problematik durch eine weitergehende Verbesserung der Suchrichtung des Lösungsalgorithmus zu begegnen. Dies könnte mit Hilfe des *konjugierten Gradientenverfahrens* erfolgen. Das Verfahren zeichnet sich im Vergleich zu anderen Methoden dadurch aus, dass es ohne zu speichernde Matrizen auskommt und deshalb für ein eingebettetes System mit seinen begrenzten Ressourcen eine Alternative darstellt.

Die entwickelte modellprädiktive Regelung ist nach alledem ein geeigneter Ansatz, die Temperaturanforderungen seitens der Abgasnachbehandlung bei gleichzeitiger Beachtung von motorinternen Regelungszielen zu erfüllen und auf diese Weise den strengen gesetzlichen Vorgaben hinsichtlich der Schadstoffemissionen gerecht zu werden. Darüber hinaus hält der Ansatz Potential für Weiterentwicklungen bereit, um auch künftig die Einhaltung der gesetzlichen Emissionsgrenzen zu gewährleisten.

A Mathematischer Anhang

A.1 Lösen nichtlinearer Gleichungssysteme

A.1.1 Fixpunktiteration

Es soll das nichtlineare Gleichungssystem

$$\mathbf{g}(\mathbf{x}) = 0 \quad (\text{A.1})$$

gelöst werden. Algebraisch ist dies im Allgemeinen nicht möglich. Eine Alternative ist das iterative Annähern, bis das gewünschte Ergebnis die geforderten Toleranzen erfüllt. Solch ein Iterationsverfahren kann durch Überführen in eine Fixpunktgleichung erzeugt werden.

Eine Fixpunktiteration

$$\mathbf{x}^{(i+1)} = \phi(\mathbf{x}^{(i)}), \quad i = 0, 1, 2, \dots, \quad \mathbf{x}^{(0)} := \mathbf{x}_0 \quad (\text{A.2})$$

konvergiert gegen einen Fixpunkt \mathbf{x}^* , wenn ϕ

$$\|\phi(\mathbf{x} + \delta) - \phi(\mathbf{x})\| \leq L\|\delta\|, \quad L < 1 \quad (\text{A.3})$$

genügt und damit eine *Kontraktion* ist. Die genannte Bedingung ist Teil des *Banach'schen* Fixpunktsatzes. Dargestellt ist die Fixpunktiteration für den skalaren Fall in Abb. A.1. Die Diagonale hat die Steigung 1. Die Darstellung entspricht qualitativ der Lösung des als skalare Funktion interpretierten Optimierungsproblems in Abb. 2.3 über eine Fixpunktiteration; es wird die Nullstelle des Gradienten über eine Fixpunktiteration gesucht.

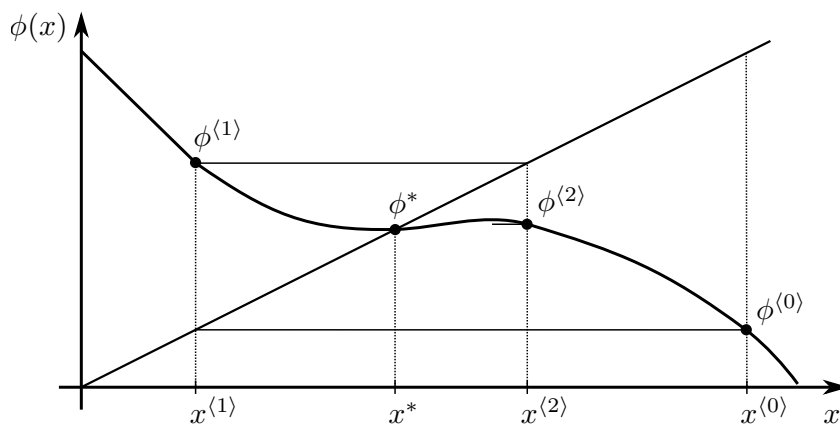


Abb. A.1: Fixpunktiteration.

Beispiel: Löse das Gleichungssystem:

$$g(x) = x^6 - x - 1, \quad x \in [0, 2].$$

Die Lösung ist $x^* \approx 1,135$.

1. Fixpunktiteration:

$$x^{(i+1)} = (x^{(i)})^6 - 1 =: \phi_1(x^{(i)})$$

Diese Iteration ist ungeeignet, da

$$|\phi_1'(x^*)| = |6(x^*)^5| > 1$$

gilt und der Zusammenhang damit keine Kontraktion ist.

2. Fixpunktiteration:

$$x^{(i+1)} = x^{(i)} - 0.01 ((x^{(i)})^6 - x^{(i)} - 1) =: \phi_2(x^{(i)})$$

Diese Iteration stellt wegen

$$|\phi_2'(x^*)| = |1 - 0.01 (6x^5 - 1)| < 1$$

eine Kontraktion dar und ist damit zur Lösung des Gleichungssystems geeignet.

A.1.2 Newton-Verfahren

Der Funktionswert der Funktion $\mathbf{g}(\mathbf{x})$ kann durch eine Taylor-Reihe approximiert werden:

$$\mathbf{g}(\mathbf{x}) = \mathbf{g}(\mathbf{x}^{(i)}) + \nabla \mathbf{g}(\mathbf{x}^{(i)})(\mathbf{x} - \mathbf{x}^{(i)}) + \mathcal{O}(\|\mathbf{x} - \mathbf{x}^{(i)}\|_2^2).$$

Statt nach einer Nullstelle von \mathbf{g} zu suchen, soll die Nullstelle der linearen Näherung von \mathbf{g} in $\mathbf{x}^{(i)}$ gesucht werden:

$$\mathbf{g}(\mathbf{x}^{(i+1)}) \stackrel{!}{=} \mathbf{0} = \mathbf{g}(\mathbf{x}^{(i)}) + \nabla \mathbf{g}(\mathbf{x}^{(i)})(\mathbf{x}^{(i+1)} - \mathbf{x}^{(i)}).$$

Daraus ergibt sich bei hinreichender Nähe zur Lösung die Fixpunktiteration

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} - (\nabla \mathbf{g}(\mathbf{x}^{(i)}))^{-1} \mathbf{g}(\mathbf{x}^{(i)}). \quad (\text{A.4})$$

Statt die Inverse zu berechnen ist es sinnvoller, dem Vorgehen in Algorithmus A.1.1 zu folgen, da dort die Berechnung der Inversen vermieden wird, welche numerisch schlechtere Eigenschaften hat, als das Gleichungssystem (A.5) zu lösen. Dies kann beispielsweise über

eine *LR-Zerlegung* [DR08, S. 68ff] geschehen.

Das Newton-Verfahren zur Lösung von Gleichungssystemen kann als Sonderfall der Lösung eines Optimierungsproblems aufgefasst werden. Insbesondere wenn (A.4) keine Fixpunktiteration ist, bietet sich die Lösung des Gleichungssystems über ein Optimierungsverfahren der Form (2.17) an. Die Abstiegsrichtung wird in diesem Fall zu

$$\mathbf{d}^{(i)} = -(\nabla \mathbf{g}^{(i)})^{-1} \mathbf{g}^{(i)}$$

gewählt.

Algorithmus A.1.1 (Newton-Verfahren).

Initialisierung

- Startpunkt festlegen: $\mathbf{x}^{(0)} := \mathbf{x}_{\text{start}}$

for i in $0, 1, 2, \dots$ do

- Berechne $\mathbf{g}(\mathbf{x}^{(i)})$, $\nabla \mathbf{g}(\mathbf{x}^{(i)})$
- Löse das lineare Gleichungssystem

$$\nabla \mathbf{g}(\mathbf{x}^{(i)}) \Delta \mathbf{x}^{(i)} = -\mathbf{g}(\mathbf{x}^{(i)}) \quad (\text{A.5})$$

- Setze (Newton-Korrektur)

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \Delta \mathbf{x}^{(i)}$$

if Konvergenztest für (A.2) erfüllt **then**

- **stop** Beende den Algorithmus

end if

end for

A.2 Fehler und Normen

Im Rahmen dieser Arbeit werden verschiedene Normen verwendet, um Fehler zwischen Mess- und Modelldaten zu bewerten. Auch zur Formulierung von in Optimierungsproblemen auftretenden Gütefunktionen werden Normen verwendet.

Für die im Folgenden vorgestellten Fehler und Normen wird eine vereinfachende Schreibweise verwendet: Als Argument wird der zu vergleichende Wert \mathbf{y} oder der Bezeichner der physikalischen Quantität y verwendet. Der zur Berechnung des Fehlers / der Norm

erforderliche Vergleichswert \mathbf{y}_{ref} geht aus dem Kontext hervor. Wenn ein Bezeichner übergeben wird, gilt dies für den zu vergleichenden Wert ebenso. Beispielsweise wird bei der Verwendung von $\mathbf{e}_{\text{abs}}(p_{22})$ unterstellt, dass dieser Fehler vor dem Hintergrund einer durchgeführten Messung bzw. eines bekannten Referenzwertes genannt wird. Zur Berechnung sind demgemäß diese Werte heranzuziehen.

Definition des absoluten Fehlers:

$$\mathbf{e}_{\text{abs}} : \mathbb{R}^n \mapsto \mathbb{R}^n; \mathbf{y} \mapsto \mathbf{y} - \mathbf{y}_{\text{ref}}.$$

Definition des relativen Fehlers:

$$\mathbf{e}_{\text{rel}} : \mathbb{R}^n \mapsto \mathbb{R}^n; \mathbf{y} \mapsto \text{diag}(\mathbf{y}_{\text{ref}})^{-1} \mathbf{y} - \mathbf{1}.$$

Definition der Euklidischen Norm (auch: 2-Norm):

$$\|\mathbf{y}\|_2 : \mathbb{R}^n \mapsto \mathbb{R}; \mathbf{y} \mapsto \sqrt{\sum_{i=1}^N y_i^2}.$$

Definition des RMSE (engl.: *Root Mean Square Error*):

$$\text{RMSE} : \mathbb{R}^n \mapsto \mathbb{R}; \mathbf{y} \mapsto \frac{1}{\sqrt{N}} \|\mathbf{e}_{\text{abs}}(\mathbf{y})\|_2.$$

Definition des NRMSE (engl.: *Normalized Root Mean Square Error*): ist über

$$\text{NRMSE} : \mathbb{R}^n \mapsto \mathbb{R}; \mathbf{y} \mapsto \frac{\text{RMSE}(\mathbf{y})}{y_{\text{min}} - y_{\text{max}}}. \quad (\text{A.6})$$

Definition des RRMSE (engl.: *Relative Root Mean Square Error*):

$$\text{RRMSE} : \mathbb{R}^n \mapsto \mathbb{R}; \mathbf{y} \mapsto \frac{1}{\sqrt{N}} \|\mathbf{e}_{\text{rel}}(\mathbf{y})\|_2.$$

B Abbildungen zu Kapitel 5

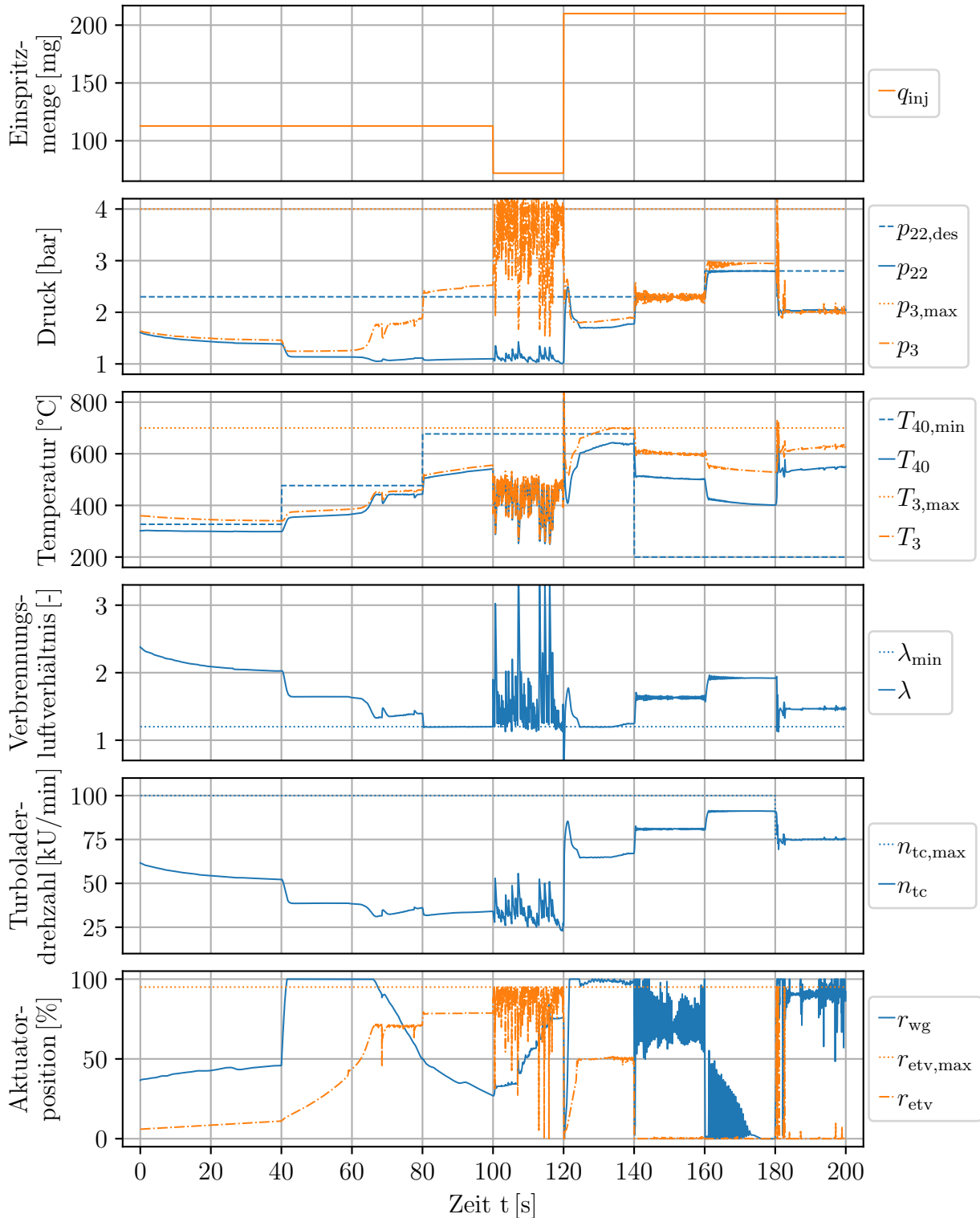


Abb. B.1: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration A (vgl. Abschnitt 5.3.2).

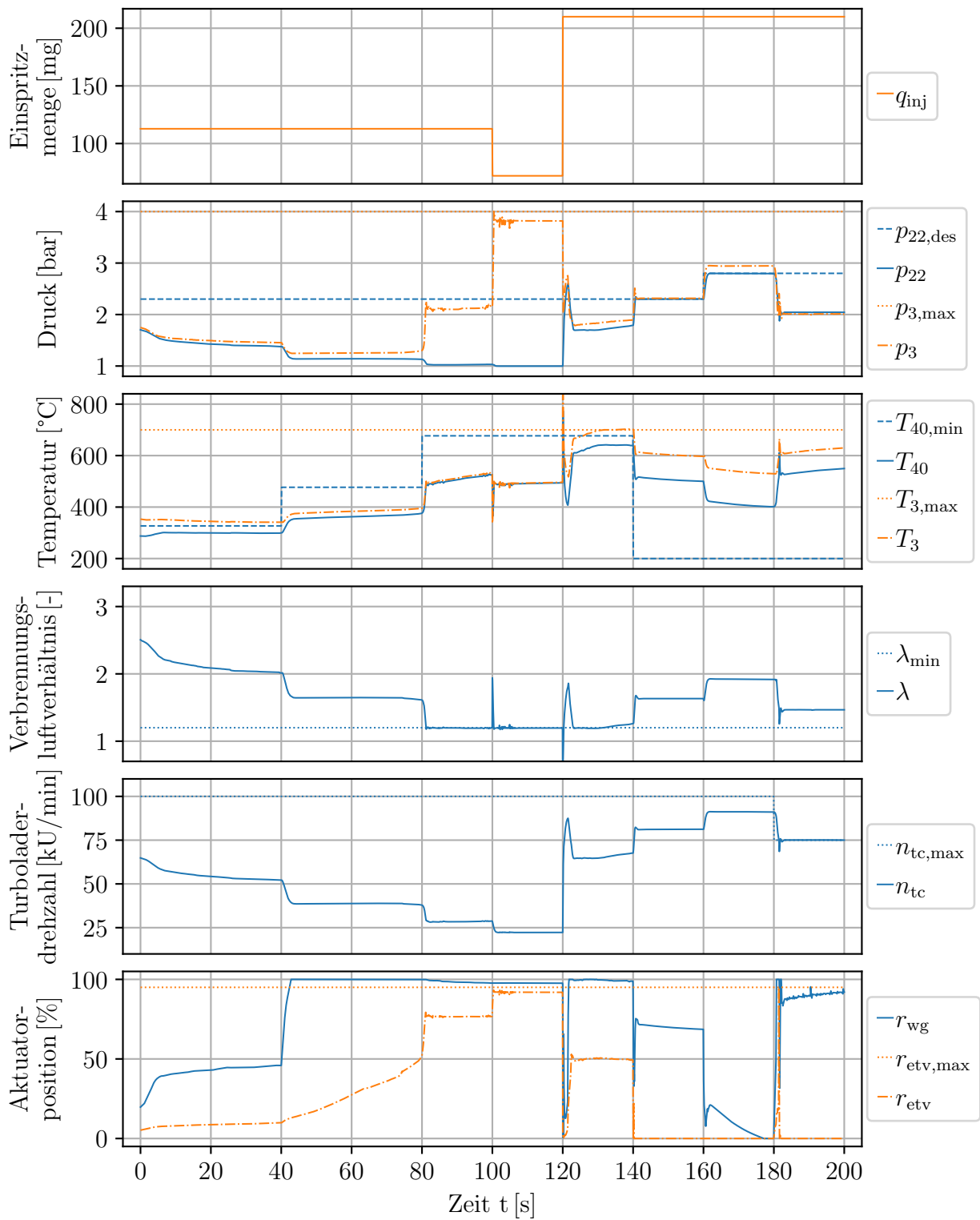


Abb. B.2: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration B (vgl. Abschnitt 5.3.2).

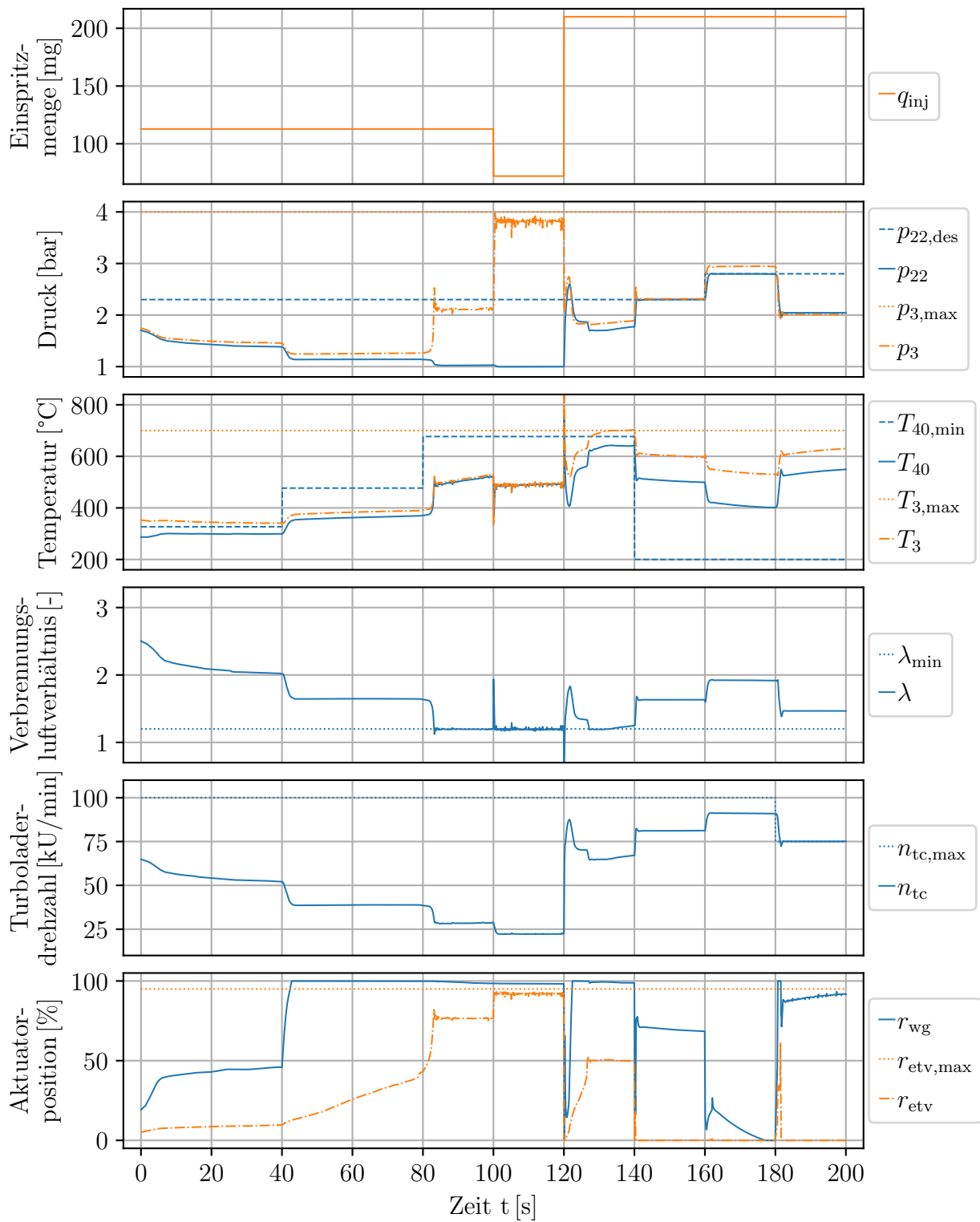


Abb. B.3: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration C (vgl. Abschnitt 5.3.2).

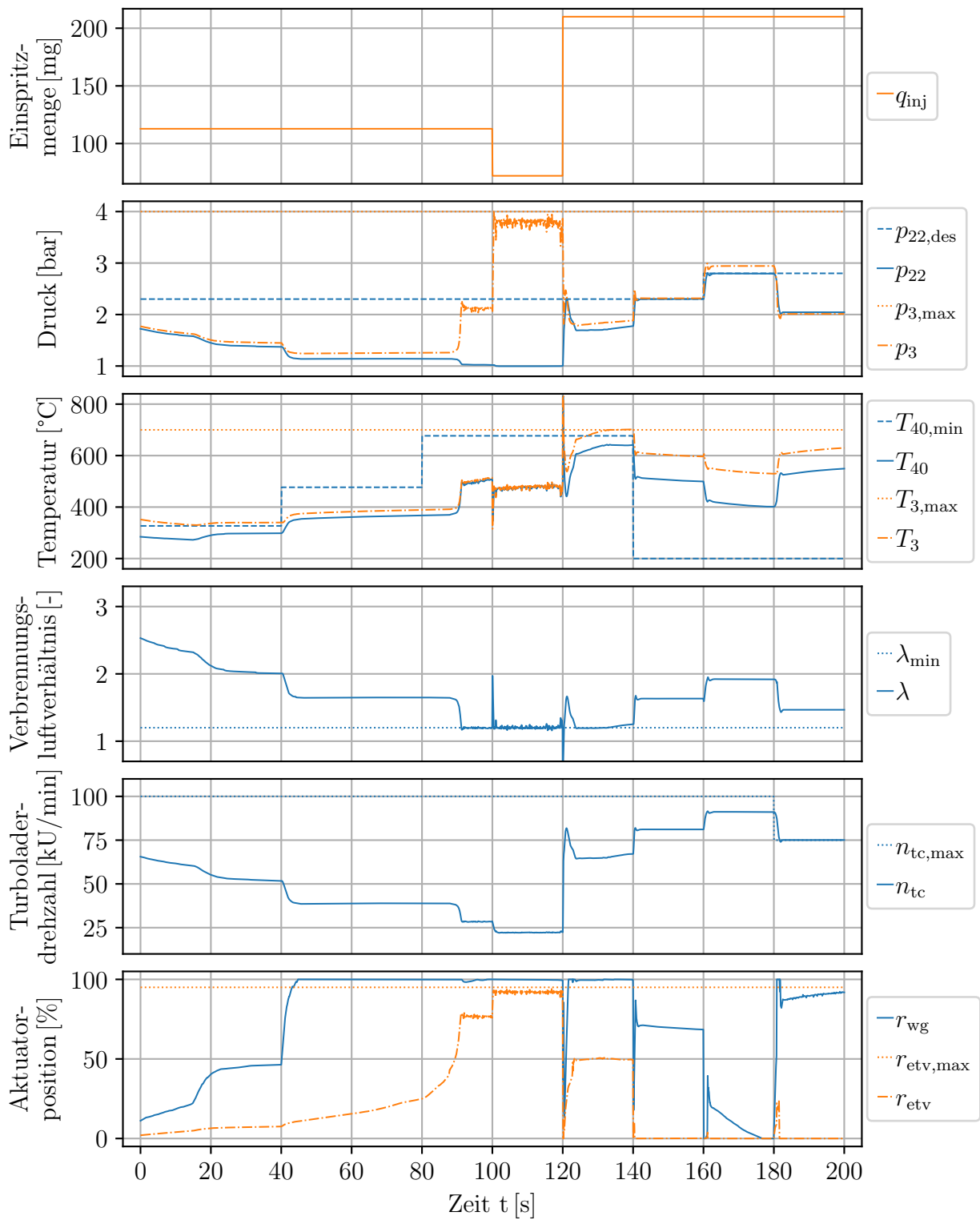


Abb. B.4: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration D (vgl. Abschnitt 5.3.2).

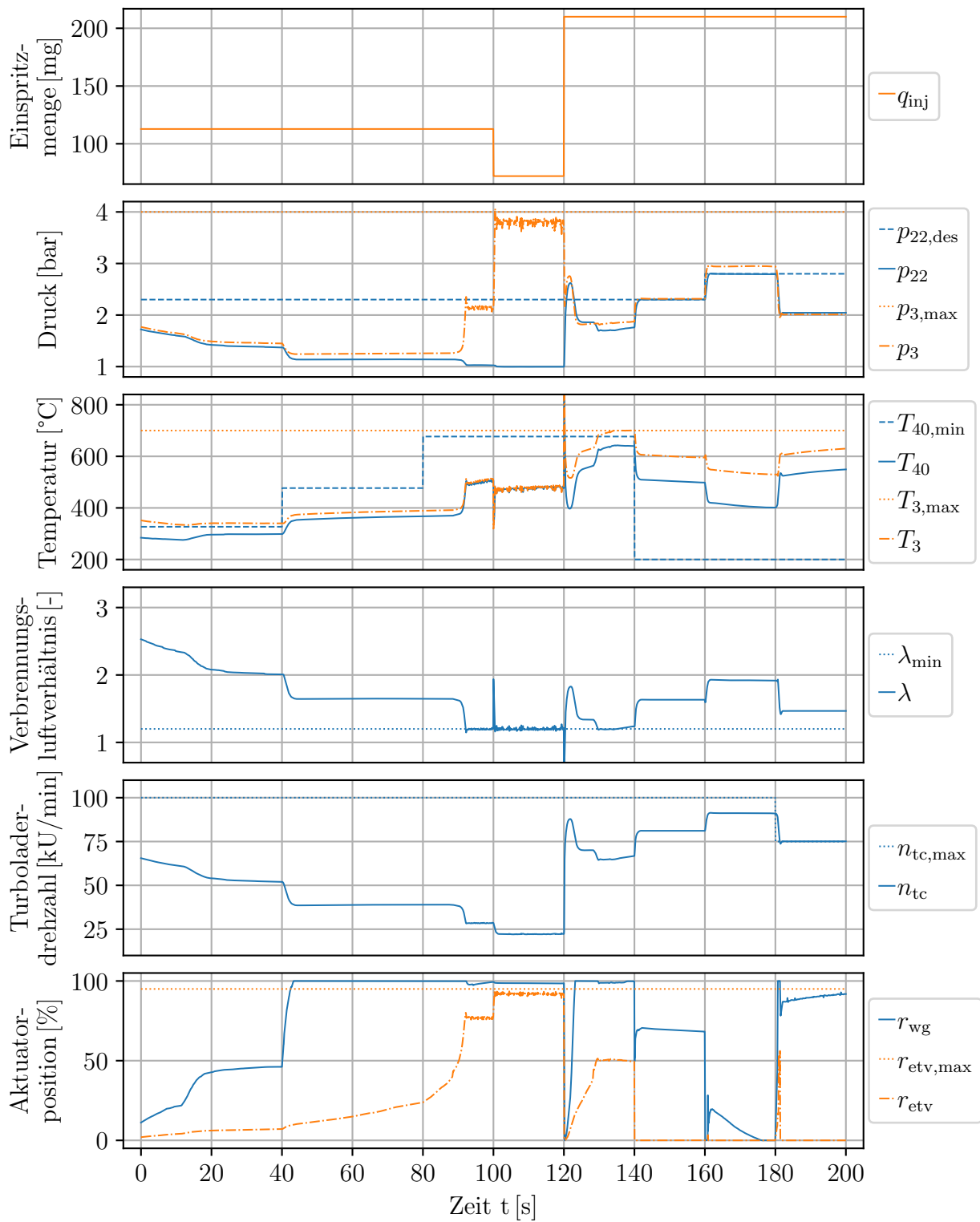


Abb. B.5: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration E (vgl. Abschnitt 5.3.2).

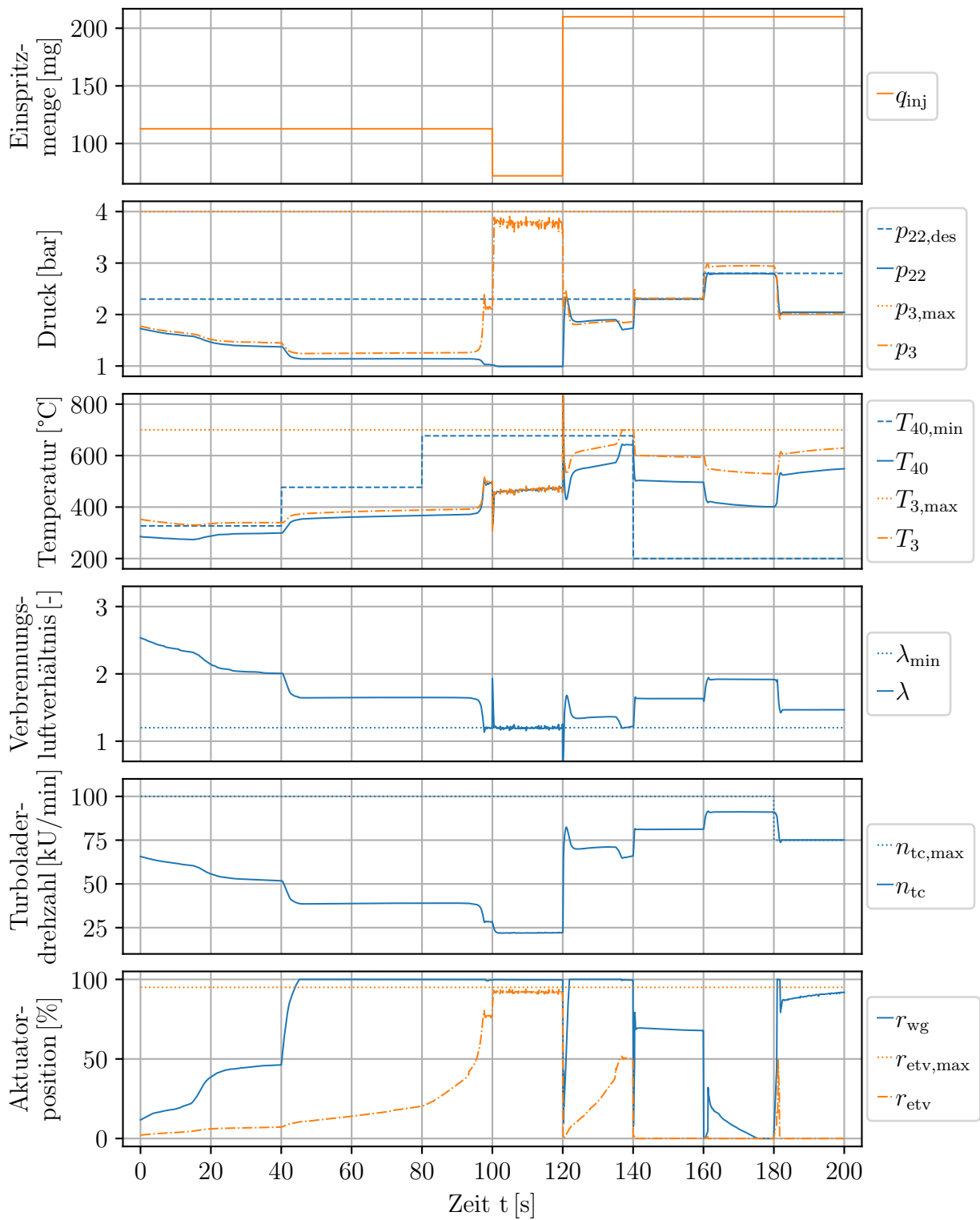


Abb. B.6: Verhalten des MPC-Lösungsalgorithmus (vgl. Abschnitt 4.4) zur Lösung der MPC-Problemformulierung (vgl. Abschnitt 5.2) unter Verwendung von Konfiguration F (vgl. Abschnitt 5.3.2).

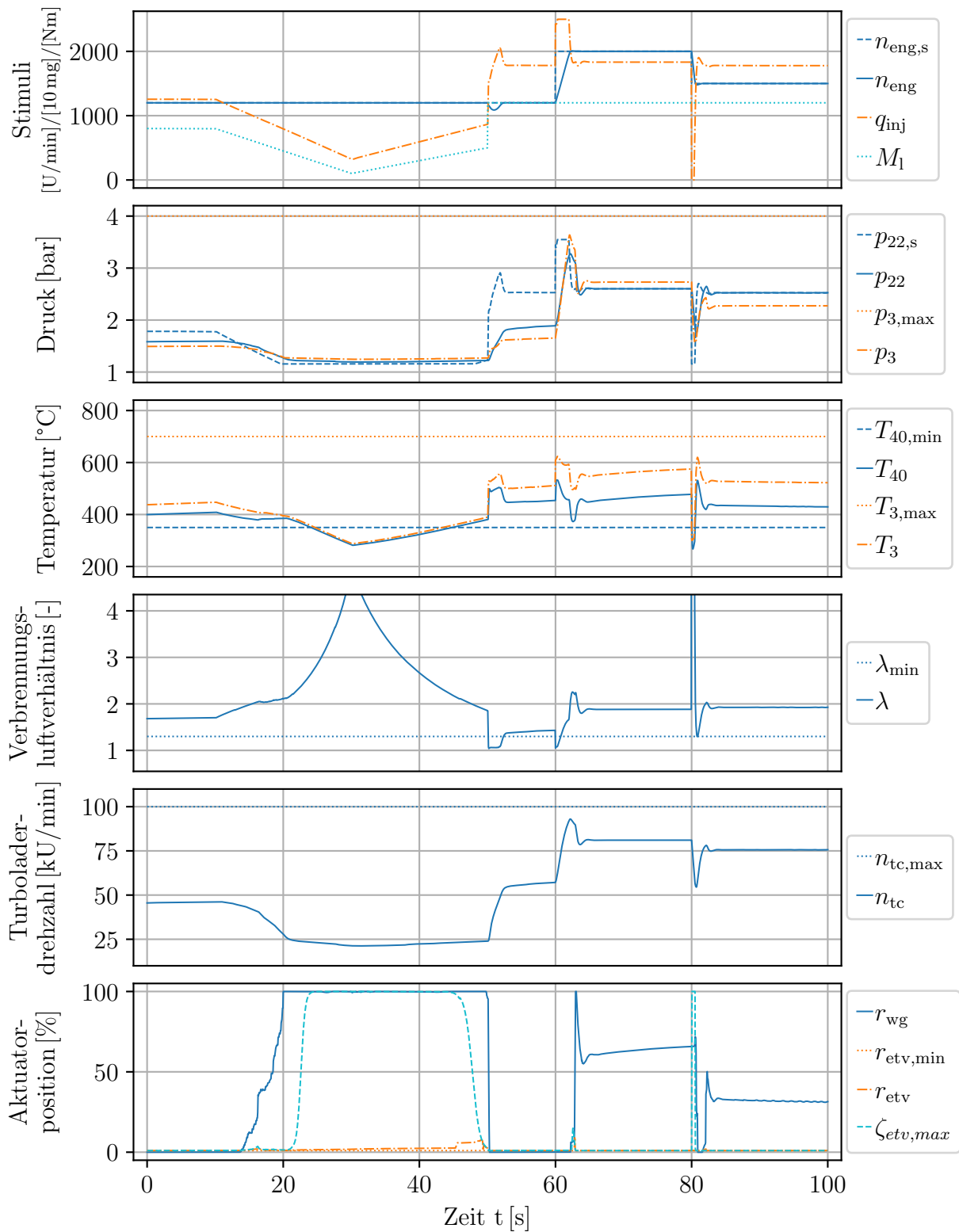


Abb. B.7: Verhalten des finalen Reglers ohne ETV-Vorsteuerung. Abb. 5.13 zeigt dasselbe Szenario mit ETV-Vorsteuerung.

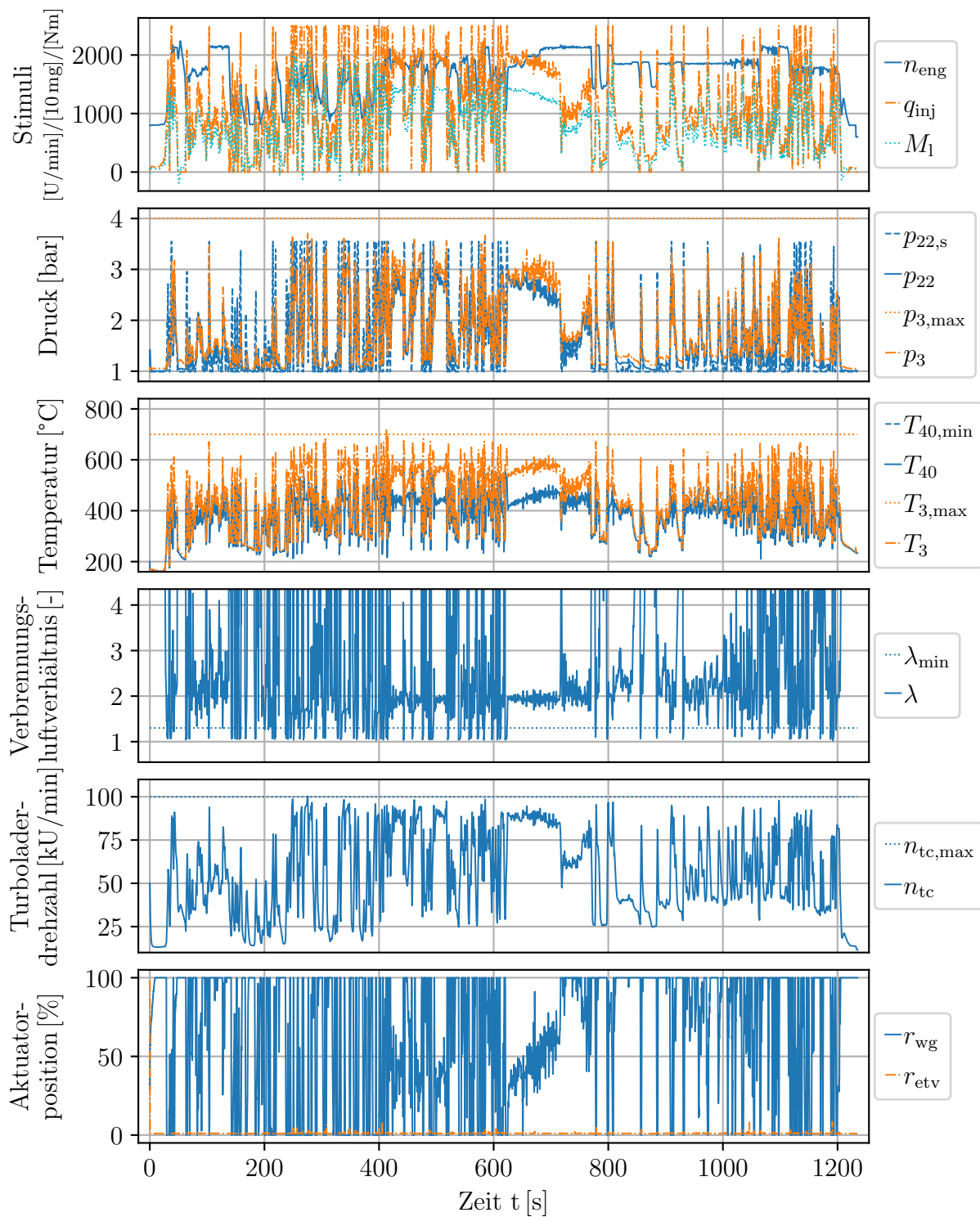


Abb. B.8: Reglerverhalten im simulierten NRTC ohne Regelung der Mindestabgastemperatur.

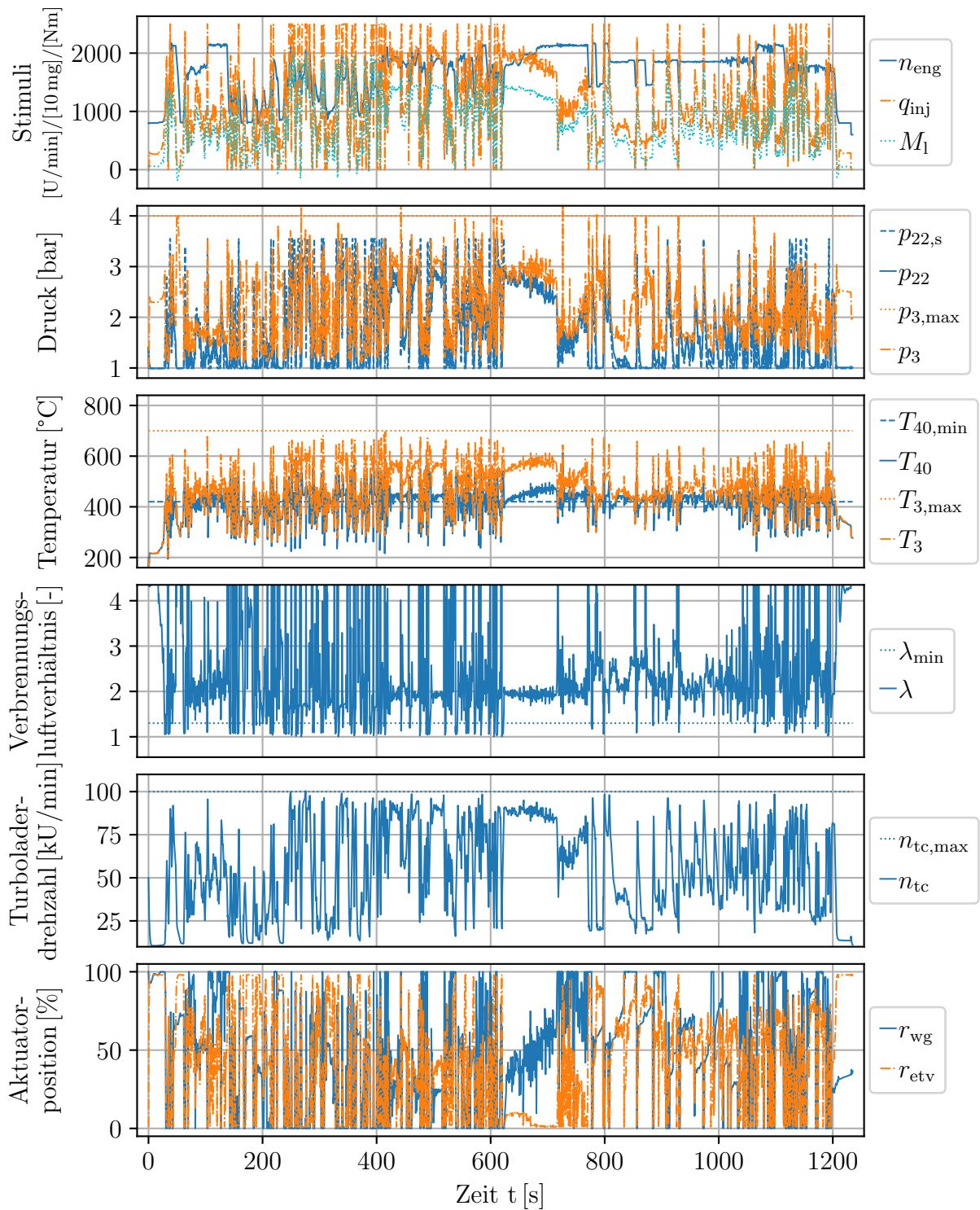


Abb. B.9: Reglerverhalten im simulierten NRTC mit Regelung der Mindestabgastemperatur.

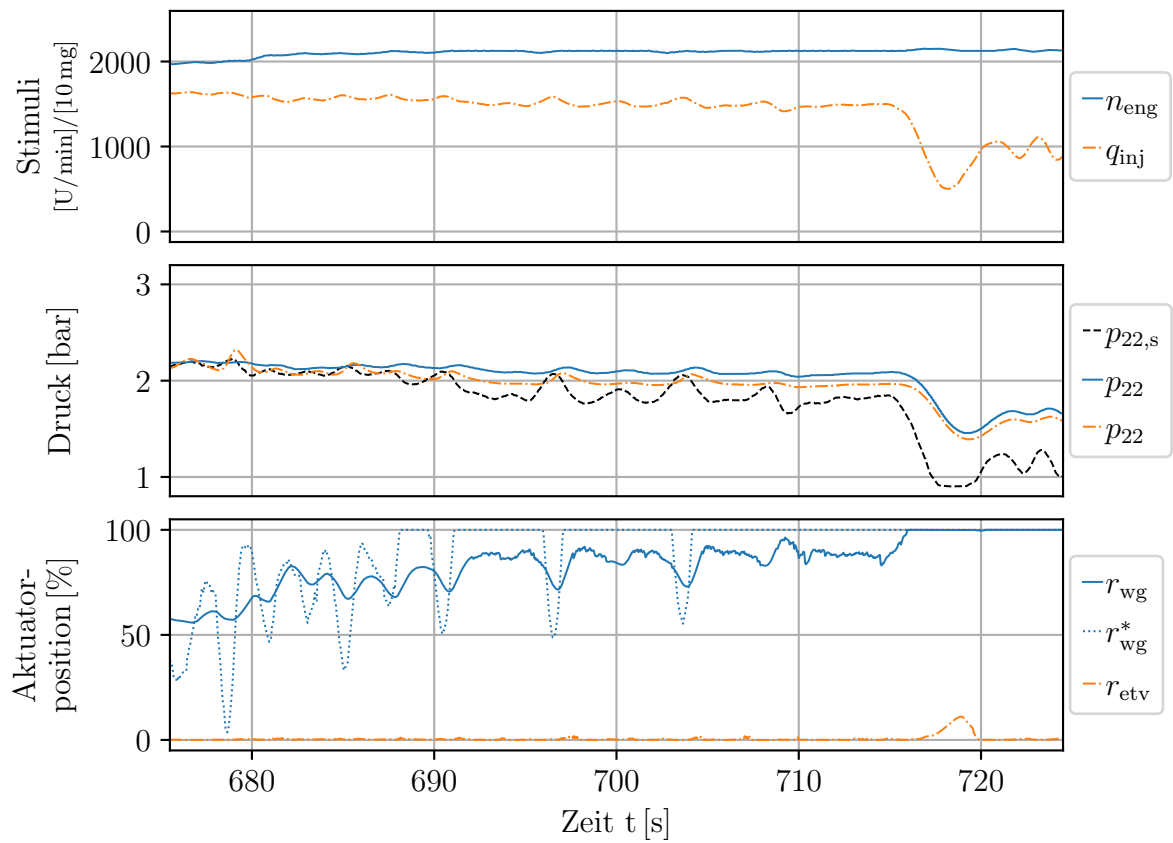


Abb. B.10: Veranschaulichung einer bleibenden Regelabweichung bei der Regelung des Ladedrucks am Motorenprüfstand.

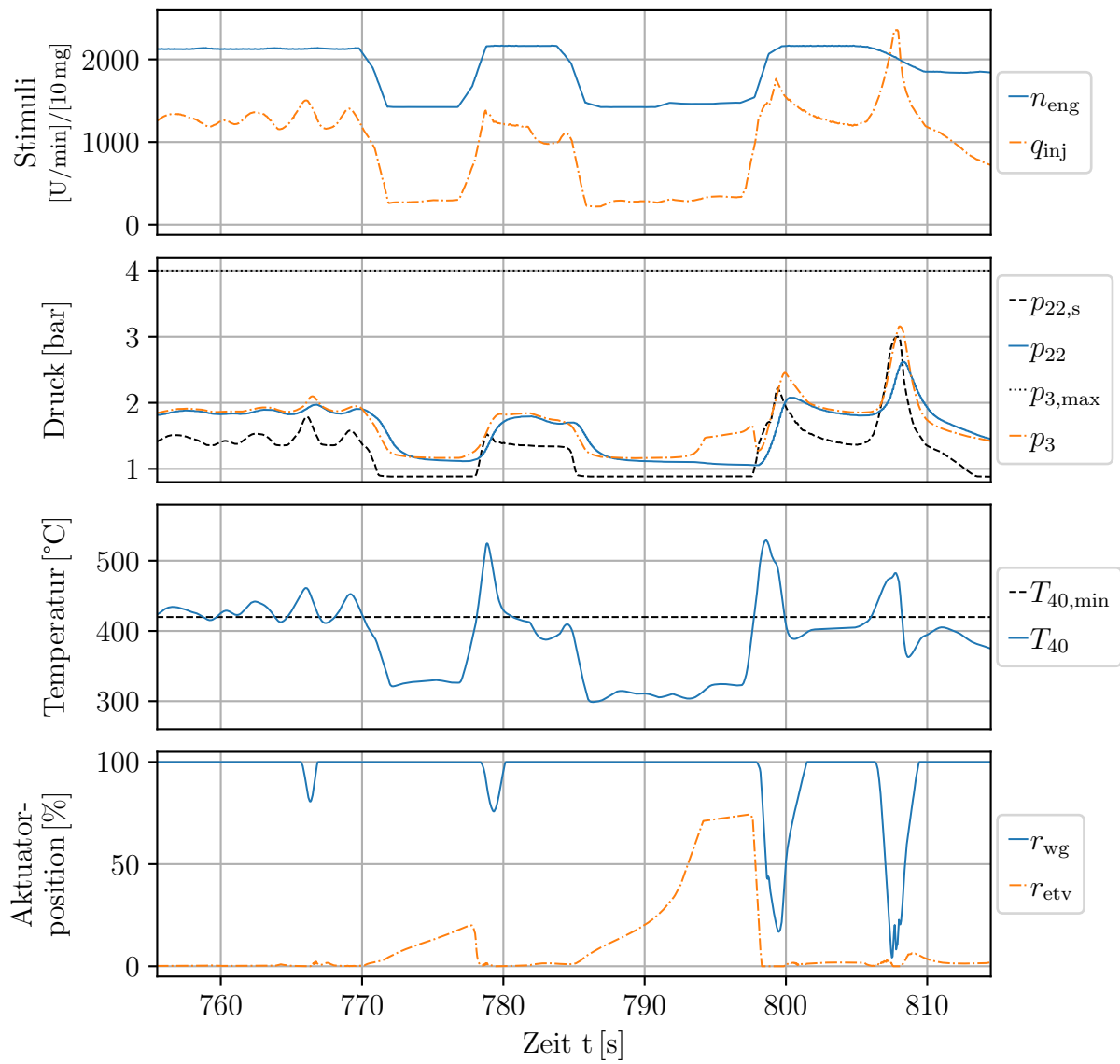


Abb. B.11: Prüfstandsauzeichnung der Regelung der Mindestabgastemperatur sowie des Ladedrucks. Korrespondierende Prüfstandsmessung zur Simulation in Abb. 5.15.

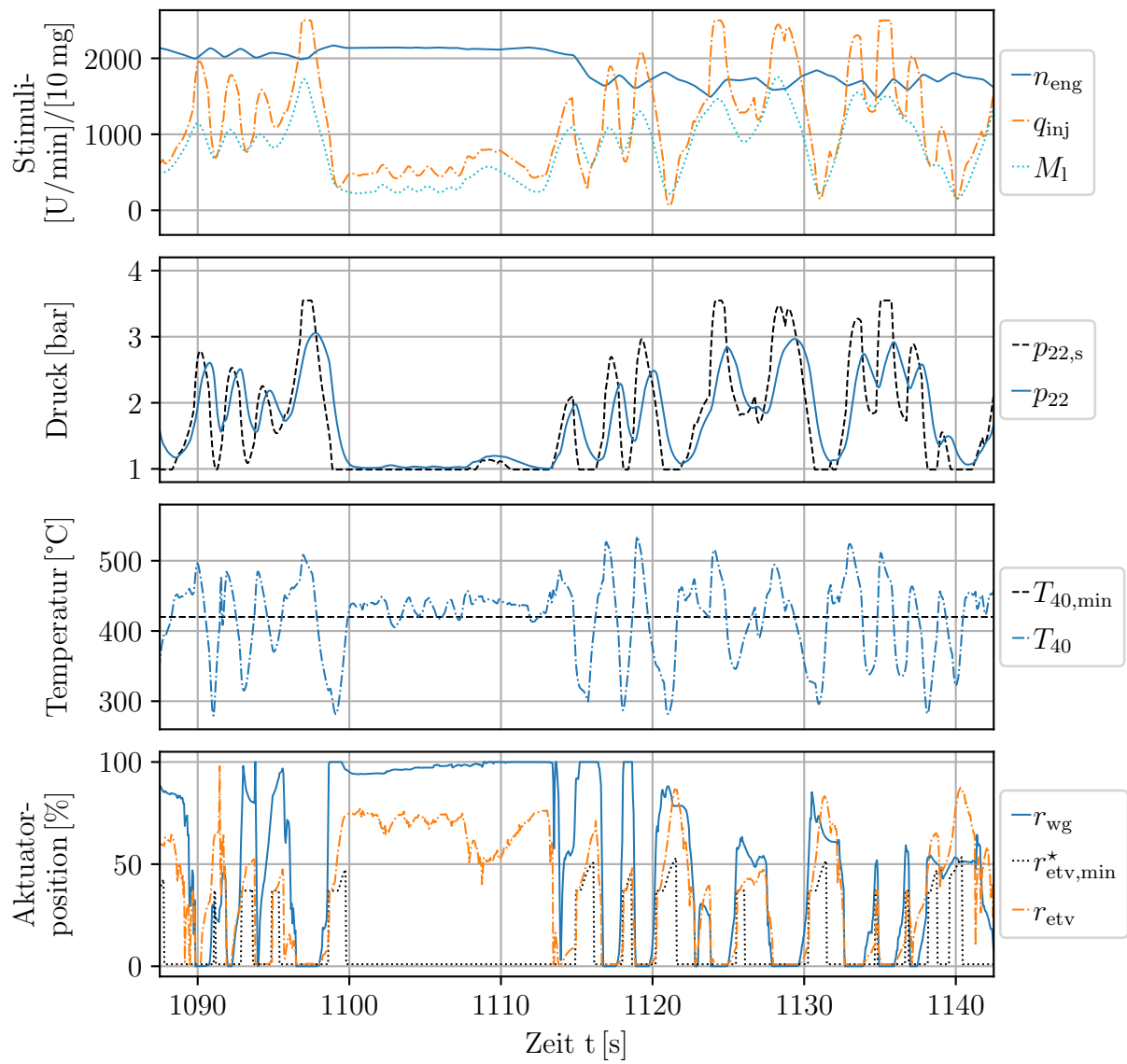


Abb. B.12: Simulative Regelung der Mindestabgastemperatur sowie des Ladedrucks. Korrespondierende Simulation zur Prüfstandsmessung in Abb. 5.17.

C Emissionsgesetzgebung

Dieser Abschnitt erhebt keinen Anspruch auf Vollständigkeit. Er soll lediglich einen Eindruck davon vermitteln, wie Dieselmotoren für *mobile Maschinen und Geräte* auf die Einhaltung der Emissionsgrenzwerte hin geprüft werden. Die genauen Prüfabläufe für die Emissionsstufen *Stufe I* bis *Stufe IV* können im Wesentlichen *Anhang III der Richtlinie 97/68/EG* vom 16.12.1997 (ABl. L 59) und für *Stufe V* der *Delegierten Verordnung (EU) 2017/654 zur Ergänzung der Verordnung 2016/1628* vom 19.12.2016 (ABl. L 102) entnommen werden.

Die in Tab. C.1 aufgeführten Grenzwerte der spezifischen Emissionen sind von Motoren der Leistungsklasse $130 \text{ kW} < P < 560 \text{ kW}$ einzuhalten, die zur Kategorie *mobile Maschinen und Geräte* gehören. Die spezifischen Emissionen gehen aus dem Quotienten der integralen Emissionen und der Gesamtarbeit hervor. Ermittelt werden diese Werte anhand gesetzlich definierter Prüfzyklen. Der NRSC (siehe *Richtlinie 97/68/EG* vom 16.12.1997, ABl. L 59/29) bildet ein stationäres Profil ab, der NRTC (siehe *Richtlinie 2004/26/EG* vom 21.04.2004, ABl. L 225/69ff.) ein dynamisches. Für *Stufe V* wurde ein weiterer stationärer Zyklus, der RMC (siehe *Delegierte Verordnung (EU) 2017/654* vom 19.12.2016, ABl. L 102/307ff.), eingeführt.

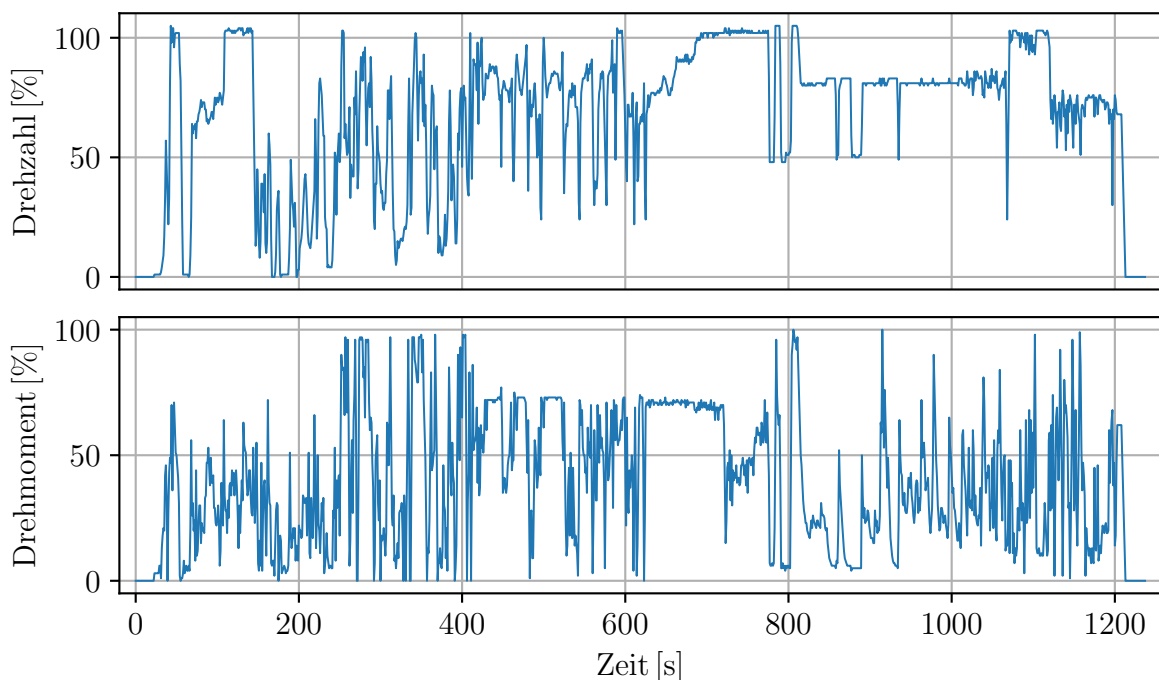


Abb. C.1: Normierte Drehzahlen und Drehmomente im NRTC-Testzyklus. Die Daten stammen aus der *Delegierten Verordnung (EU) 2017/654* vom 19.12.2016 (ABl. L 102/313ff). Anmerkung: Bei den normierten Drehzahlen größer 100 % liegt kein Übertragungsfehler vor.

Der NRTC wird im Rahmen der Prüfung zweimal durchgeführt: einmal als sog. *kalt-NRTC* und ein weiteres Mal als sog. *warm-NRTC*. Der kalt-NRTC wird aus einem Kaltstart heraus durchlaufen. Der Motor ist auf Raumtemperatur abgekühlt, die Temperaturen des Motorkühlmittels und Motoröls, der Abgasnachbehandlungseinrichtungen und sämtlicher Motorsteuerungshilfsmittel haben sich zwischen 20 °C bis 30 °C stabilisiert. Im Anschluss wird mit dem aufgewärmten Motor ein weiterer NRTC durchlaufen. Die Integralwerte beider Zyklen werden mit 10 % (kalt-NRTC) und 90 % (warm-NRTC) gewichtet. Die gewichteten Integralwerte beider Zyklen dürfen die Grenzwerte nicht überschreiten. Das Profil des NRTC-Zyklus ist in Abb. C.1 dargestellt. Der kleinste Drehzahlwert entspricht der Leerlaufdrehzahl und der größte Wert der maximalen Prüfdrehzahl. Das Drehmoment ist bezüglich des bei der zugehörigen Drehzahl maximal erreichbaren Drehmoments normiert.

Tab. C.1: Emissionsgrenzwerte der verschiedenen Emissionsstufen in der Europäischen Union. Es sind die Werte für solche Motoren aufgeführt, die für *mobile Maschinen und Geräte* eingesetzt werden und sich im Leistungsbereich von 130 kW bis 560 kW befinden. Diese dürfen, über den Testzyklus gemittelt, nicht überschritten werden.

| Stufe | Gültig ab | Testzyklen | CO [g/kWh] | HC [g/kWh] | HC + NO _x [g/kWh] | NO _x [g/kWh] | NH ₃ [ppm] | PM [g/kWh] | PN [1/kWh] |
|-------------------|-----------|-----------------|---------------|---------------|---------------------------------|----------------------------|--------------------------|---------------|------------------|
| I ¹ | 1999 | NRSC | 5,0 | 1,30 | - | 9,2 | - | 0,540 | - |
| II ¹ | 2002 | NRSC | 3,5 | 1,00 | - | 6,0 | - | 0,200 | - |
| IIIA ² | 2006 | NRSC / NRTC | 3,5 | - | 4,0 | - | - | 0,200 | - |
| IIIB ³ | 2011 | NRSC, NRTC | 3,5 | 0,19 | - | 2,0 | 25 | 0,025 | - |
| IV ⁴ | 2014 | NRSC, NRTC | 3,5 | 0,19 | - | 0,4 | 10 | 0,025 | - |
| V ⁵ | 2019 | NRSC, NRTC, RMC | 3,5 | 0,19 | - | 0,4 | 10 | 0,015 | 10 ¹² |

¹ Vgl. *Richtlinie 97/68/EG* vom 16.12.1997, ABl. L 59/15,59.

² Vgl. *Richtlinie 2004/26/EG* vom 21.04.2004, ABl. L 225/15,17,69ff.

³ Vgl. *Richtlinie 2004/26/EG* vom 21.04.2004, ABl. L 225/16,17f und *Richtlinie 2010/26/EU* vom 31.03.2010, ABl. L 86/32. Anmerkung: Die deutsche Fassung von *Richtlinie 2004/26/EG* ist bezüglich der HC-Grenzwerte auf S. 16 unvollständig. Der angegebene Wert entstammt daher der englischen Originalfassung.

⁴ Vgl. *Richtlinie 2004/26/EG* vom 21.04.2004, ABl. L 225/17 und *Richtlinie 2012/46/EU* vom 06.12.2012, ABl. L 353/84.

⁵ Vgl. *Verordnung (EU) 2016/1628* vom 14.09.2016, ABl. L 252/104 und *Delegierte Verordnung (EU) 2017/654* vom 19.12.2016, ABl. L 102/33,67.

D Deutz TCD 12.0 V6

Tab. D.1: Technische Daten des Deutz TCD 12.0 V6 für landwirtschaftliche Anwendungen.

| | | |
|---|---------|-------------|
| Zylinderzahl | [-] | 6 |
| Bohrung / Hub | [mm] | 132 / 145 |
| Hubvolumen | [Liter] | 11,9 |
| Maximale Nenndrehzahl | [U/min] | 1800 – 2100 |
| Leistung nach ISO 14396 ¹ | [kW] | 390 |
| (bei Drehzahl) | [U/min] | 2 100 |
| Maximales Drehmoment | [Nm] | 2 130 |
| (bei Drehzahl) | [U/min] | 1 400 |
| Niedrigste Leerlaufdrehzahl | [U/min] | 600 |
| Spezifischer Kraftstoffverbrauch ² | [g/kWh] | 195 |
| Gewicht nach DIN 70020 Teil 7A ³ | [kg] | 995 |

¹ Leistungsangaben ohne Abzug der Lüfterleistung.

² Bestpunktverbrauch, bezogen auf Dieselkraftstoff mit der Dichte 0,835 kg/dm³ bei 15 °C.

³ Ohne Anlasser / Lichtmaschine, Kühler und Flüssigkeiten, jedoch mit Schwungrad und Schwungradgehäuse.

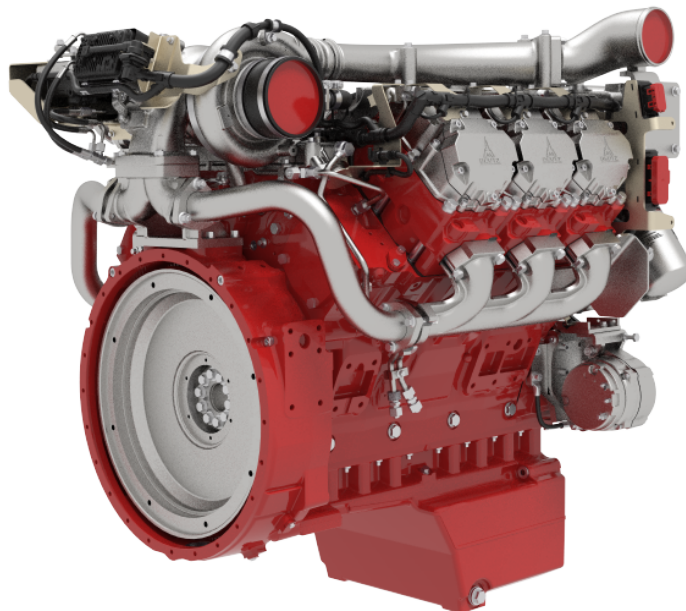


Abb. D.1: CAD-Zeichnung des Deutz TCD 12.0 V6 Dieselmotors in einer Betrachtung von vorne rechts. Quelle: ©Deutz AG.

Literaturverzeichnis

- [Ada14] ADAMY, Jürgen: *Nichtlineare Systeme und Regelungen*. Berlin u.a. : Springer Vieweg, 2014
- [AEH14] ALPCAN, Tansu ; EVERITT, Tom ; HUTTER, Marcus: Can we measure the difficulty of an optimization problem? In: *2014 IEEE Information Theory Workshop (ITW 2014)* (2014), S. 356–360
- [AF66] ATHANS, Michael ; FALB, Peter L.: *Optimal Control*. New York : McGraw-Hill, 1966
- [AFN04] ALLGÖWER, Frank ; FINDEISEN, Rolf ; NAGY, Zoltan K.: Nonlinear model predictive control: From theory to application. In: *Journal of the Chinese Institute of Chemical Engineers* 35 (2004), Nr. 3, S. 299–315
- [AS58] ARROW, K J. ; SOLOW, R M.: Gradient Methods for Constrained Maxima with Weakened Assumptions. In: *Studies in Linear and Nonlinear Programming* (1958), S. 166–176
- [BB88] BARZILAI, Jonathan ; BORWEIN, Jonathan M.: Two-point step size gradient methods. In: *IMA Journal of Numerical Analysis* 8 (1988), Nr. 1, S. 141–148
- [BB12] BÖSWIRTH, Leopold ; BSCHORER, Sabine: *Technische Strömungslehre*. 10. Aufl. Wiesbaden : Vieweg+Teubner Verlag, 2012
- [BBB⁺13] BAERNS, Manfred ; BEHR, Arno ; BREHM, Axel ; GMEHLING, Jürgen ; HINRICHSSEN, Kai-Olaf ; HOFMANN, Hanns ; ONKEN, Ulfert ; PALKOVITS, Regina ; RENKEN, Albert: *Technische Chemie*. 2. Auflage. Weinheim : Wiley-VCH, 2013
- [Ben83] BENTON, Stanley: PL Lions, Generalized solutions of Hamilton-Jacobi equations. In: *Bulletin (New Series) of the American Mathematical Society* 9 (1983), Nr. 2, S. 252–256
- [Ber96] BERTSEKAS, Dimitri P.: *The Method of Multipliers for Equality Constrained Problems*. Belmont : Athena Scientific, 1996
- [Ber17] BERTSEKAS, Dimitri P.: *Dynamic Programming and Optimal Control*. 4. Auflage. Belmont : Athena Scientific, 2017
- [BK16] BAEHR, Hans D. ; KABELAC, Stephan: *Thermodynamik Grundlagen und tech-*

- nische Anwendungen*. 16. Auflag. Berlin u.a. : Springer Vieweg, 2016. – 671 S.
- [BMDP02] BEMPORAD, Alberto ; MORARI, Manfred ; DUA, Vivek ; PISTIKOPOULOS, Efstratios N.: The explicit linear quadratic regulator for constrained systems. In: *Automatica* 38 (2002), Nr. 1, S. 3–20
- [BT95] BOGGS, Paul T. ; TOLLE, Jon W.: Sequential Quadratic Programming. In: *Acta Numerica* 4 (1995), S. 1–51
- [BV09] BOYD, Stephen ; VANDENBERGHE, Lieven: *Convex Optimization*. Cambridge u.a. : Cambridge University Press, 2009
- [CA98] CHEN, Hong ; ALLGÖWER, Frank: A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. In: *ECC 1997 - European Control Conference* 34 (1998), Nr. 10, S. 1205–1217
- [CGT92] CONN, Andrew ; GOULD, Nick ; TOINT, Philippe: *LANCELOT: a Fortran package for large-scale nonlinear optimization (Release A)*. Berlin u.a. : Springer Verlag, 1992
- [Cho15] CHOLLET, François: Keras: The Python Deep Learning library. 2015. – Forschungsbericht
- [CMH00] CHEVALIER, Alain ; MÜLLER, Martin ; HENDRICKS, Elbert: On the validity of mean value engine models during transient operation. In: *SAE Transactions* 109 (2000), Nr. 724, S. 1571–1592
- [CSA⁺18] CHEN, Steven ; SAULNIER, Kelsey ; ATANASOV, Nikolay ; LEE, Daniel D. ; KUMAR, Vijay ; PAPPAS, George J. ; MORARI, Manfred: Approximating Explicit Model Predictive Control Using Constrained Neural Networks. In: *Proceedings of the American Control Conference 2018-June* (2018), S. 1520–1527
- [CVC01] CHRISTEN, Urs ; VANTINE, Katie J. ; COLLINGS, Nick: Event-Based Mean-Value Modeling of DI Diesel Engines for Controller Design. In: *SAE Technical Paper Series* 1 (2001), Nr. 724
- [DBDW05] DIEHL, Moritz ; BOCK, Hans G. ; DIEDAM, Holger ; WIEBER, Pierre-brice: Fast Direct Multiple Shooting Algorithms for Optimal Robot Control. In: *Fast Motions in Biomechanics and Robotics* (2005)
- [DFH09] DIEHL, Moritz ; FERREAU, Hans J. ; HAVERBEKE, Niels: Efficient numerical methods for nonlinear MPC and moving horizon estimation. In: *Nonlinear model predictive control* (2009), S. 391–417

- [Dol18] DOLBEAU, Romain: Theoretical peak FLOPS per instruction set: a tutorial. In: *Journal of Supercomputing* 74 (2018), Nr. 3, S. 1341–1377
- [DR08] DAHMEN, Wolfgang ; REUSKEN, Arnold: *Numerik für Ingenieure und Naturwissenschaftler*. 2. Auflage. Berlin u.a. : Springer Verlag, 2008
- [DW75] DICKMANN, Ernst D. ; WELL, Klaus H.: Approximate solution of optimal control problems using third order hermite polynomial functions. In: *Optimization Techniques IFIP Technical Conference*. Berlin u.a. : Springer, 1975, S. 158–166
- [ECTC12] EL HADEF, Jamil ; COLIN, Guillaume ; TALON, Vincent ; CHAMAILLARD, Yann: New physics-based turbocharger data-maps extrapolation algorithms: Validation on a spark-ignited engine. In: *IFAC Proceedings Volumes (IFAC PapersOnline)* (2012), S. 213–220
- [Eri07] ERIKSSON, Lars: Modeling and control of turbocharged SI and DI engines. In: *Oil & Gas Science and Technology-Revue de l'IFP* 62 (2007), S. 523–538
- [EVM⁺19] ENGLERT, Tobias ; VÖLZ, Andreas ; MESMER, Felix ; RHEIN, Sönke ; GRAICHEN, Knut: *A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC)*. Springer US, 2019
- [FGW02] FORSGREN, Anders ; GILL, Philip E. ; WRIGHT, Margaret H.: Interior methods for nonlinear optimization. In: *SIAM Review* 44 (2002), Nr. 4, S. 525–597
- [FM68] FIACCO, Anthony V. ; MCCORMICK, Garth P.: *Nonlinear programming: sequential unconstrained minimization techniques*. New York u.a. : Wiley, 1968
- [GA98] GUZZELLA, Lino ; AMSTUTZ, Alois: Control of Diesel Engines. In: *IEEE Control Systems* 18 (1998), Nr. 5, S. 53–71
- [GD20] GROS, Sébastien ; DIEHL, Moritz: *Numerical Optimal Control*. 2020
- [GK12] GRAICHEN, Knut ; KÄPERNICK, Bartosz: A Real-Time Gradient Method for Nonlinear Model Predictive Control. In: *IntechOpen* (2012)
- [GOPB17] GUARDIOLA, Carlos ; OLMEDA, Pablo ; PLA, Benjamin ; BARES, Pau: In-cylinder pressure based model for exhaust temperature estimation in internal combustion engines. In: *Applied Thermal Engineering* 115 (2017), S. 212–220
- [Gör05] GÖRSMANN, Claus: Catalytic coatings for active and passive diesel particulate filter regeneration. In: *Monatshefte für Chemie* 136 (2005), Nr. 1, S. 91–105

- [GP17] GRÜNE, Lars ; PANNEK, Jürgen: *Nonlinear Model Predictive Control: Theory and Algorithms*. 2. Auflage. Springer Verlag, 2017
- [Gra18] GRAICHEN, Knut: *Methoden der Optimierung und optimalen Steuerung*. 2018
- [Gri08] GRIEWANK, Andreas: *Evaluating derivatives : principles and techniques of algorithmic differentiation*. Philadelphia : SIAM, 2008
- [GZLH15] GUAN, Bin ; ZHAN, Reggie ; LIN, He ; HUANG, Zhen: Review of the state-of-the-art of exhaust particulate filter technology in internal combustion engines. In: *Journal of Environmental Management* 154 (2015), S. 225–258
- [Hes69] HESTENS, Magnus R.: Multiplier and gradient methods. In: *Journal of Optimization Theory and Applications* 4 (1969), Nr. 5, S. 303–320
- [HRFA06] HERCEG, Martin ; RAFF, Tobias ; FINDEISEN, Rolf ; ALLGÖWER, Frank: Non-linear model predictive control of a turbocharged diesel engine. In: *Proceedings of the IEEE International Conference on Control Applications*, 2006, S. 2766–2771
- [Irl16] IRLE, Jan P.: *Analyse und Entwurf von Beobachtern mit unbekanntem Signalen und Parametern*, Karlsruher Institut für Technologie (KIT), Diss., 2016
- [Ise08] ISERMANN, Rolf: *Mechatronische Systeme Grundlagen*. 2. Auflage. Berlin u.a. : Springer Verlag, 2008
- [JKS⁺91] JENSEN, J. P. ; KRISTENSEN, A. F. ; SPENCER, C. ; HOUBAK, N. ; HENDRICKS, E.: Mean value modeling of a small turbocharged diesel engine. In: *SAE Technical Paper* (1991)
- [Joh11] JOHNSON, Timothy V.: Diesel Emissions in Review. In: *SAE International Journal of Engines* 4 (2011), Nr. 1, S. 143–157
- [JTB09] JEGUIRIM, Mejdí ; TSCHAMBER, Valérie ; BRILHAC, Jean F.: Kinetics of catalyzed and non-catalyzed soot oxidation with nitrogen dioxide under regeneration particle trap conditions. In: *Journal of Chemical Technology and Biotechnology* 84 (2009), Nr. 5, S. 770–776
- [JYH01] JADBABAIE, Ali ; YU, Jie ; HAUSER, John: Unconstrained Receding-Horizon Control of Nonlinear Systems. In: *IEEE Transactions on Automatic Control* 46 (2001), Nr. 5, S. 776–783
- [KBV15] KALMARI, Jouko ; BACKMAN, Juha ; VISALA, Arto: A toolkit for nonlinear model predictive control using gradient projection and code generation. In: *Control Engineering Practice* 39 (2015), S. 56–66

- [KCMN93] KOTHARE, Mayuresh V. ; CAMPO, Peter J. ; MORARI, Manfred ; NETT, Carl N.: A Unified Framework for the Study of Anti-Windup Designs. In: *Automatica* 30 (1993), Nr. 12, S. 1869–1883
- [Kel62] KELLEY, Henry J.: Method of Gradients. In: LEITMANN, George (Hrsg.): *Optimization techniques: with applications to aerospace systems*. New York : Academic Press Inc., 1962, Kapitel 6, S. 205–254
- [KG13] KÄPERNICK, Bartosz ; GRAICHEN, Knut: Model predictive control of an overhead crane using constraint substitution. In: *Proceedings of the American Control Conference* (2013), S. 3973–3978
- [KHDR13] KOLTSAKIS, Grigorios ; HARALAMPOUS, Onoufrios ; DEPCIK, Christopher ; RAGONE, J. C.: Catalyzed diesel particulate filter modeling. In: *Reviews in Chemical Engineering* 29 (2013), Nr. 1, S. 1–61
- [KHK02] KANDYLAS, Ioannis P. ; HARALAMPOUS, Onoufrios A. ; KOLTSAKIS, Grigorios C.: Diesel soot oxidation with NO₂: Engine experiments and simulations. In: *Industrial and Engineering Chemistry Research* 41 (2002), Nr. 22, S. 5372–5384
- [Kir04] KIRK, Donald E.: *Optimal Control Theory: An Introduction*. Mineola, NY : Dover Publications, 2004
- [KS96] KOLTSAKIS, Grigorios C. ; STAMATELOS, Anastasios M.: Modeling Catalytic Regeneration of Wall-Flow Particulate Filters. In: *Industrial and Engineering Chemistry Research* 35 (1996), Nr. 1, S. 2–13
- [Lee99] LEENSON, Ilya A.: Old Rule of Thumb and the Arrhenius Equation. In: *Journal of Chemical Education* 76 (1999), Nr. 10, S. 1459–1460
- [Loh96] LOH, Wei L.: On latin hypercube sampling. In: *Annals of Statistics* 24 (1996), Nr. 5, S. 2058–2080
- [Lue71] LUENBERGER, David G.: An Introduction to Observers. In: *IEEE Transactions on Automatic Control* 16 (1971), Nr. 6, S. 596–602
- [Lun13] LUNZE, Jan: Einführung in die Mehrgrößenregelung. In: *Regelungstechnik 2*. Berlin u.a. : Springer Verlag, 2013
- [LZL18] LIU, Yu ; ZHAO, Jun ; LEE, Jong M.: Conventional and New Materials for Selective Catalytic Reduction (SCR) of NO_x. In: *ChemCatChem* 10 (2018), Nr. 7, S. 1499–1511
- [MH99] MORARI, Manfred ; H. LEE, Jay: Model predictive control: Past, present and future. In: *Computers and Chemical Engineering* 23 (1999), Nr. 4-5, S. 667–682

- [MHM09] MICHALIK, Claas ; HANNEMANN, Ralf ; MARQUARDT, Wolfgang: Incremental single shooting-A robust method for the estimation of parameters in dynamical systems. In: *Computers and Chemical Engineering* 33 (2009), Nr. 7, S. 1298–1305
- [MHS98] MÜLLER, Martin ; HENDRICKS, Elbert ; SORENSON, Spencer C.: Mean value modelling of turbocharged spark ignition engines. In: *SAE Technical Paper*, SAE International, 1998
- [MK99] MORAAL, Paul ; KOLMANOVSKY, Ilya: Turbocharger Modeling for Automotive Control Applications. In: *SAE Technical Paper*, 1999
- [MM90] MAYNE, David. Q. ; MICHALSKA, Hannah: An Implementable Receding Horizon Controller for Stabilization of Nonlinear Systems. In: *Proceedings of the IEEE Conference on Decision and Control*, 1990, S. 3396–3397
- [MR93] MUSKE, Kenneth R. ; RAWLINGS, James B.: Model predictive control with linear models. In: *AIChE Journal* 39 (1993), Nr. 2, S. 262–287
- [MRRS00] MAYNE, David ; RAWLINGS, James B. ; RAO, Christopher V. ; SCOKAERT, Pierre: Constrained model predictive control: Stability and optimality. In: *Automatica* 36 (2000), S. 789–814
- [MS83] MURTAGH, Bruce A. ; SAUNDERS, Michael A.: *MINOS 5.0 User's Guide*. Stanford : Systems Optimization Laboratory, Stanford University, 1983
- [MSP⁺17] MEURER, Aaron ; SMITH, Christopher P. ; PAPROCKI, Mateusz ; ČERTÍK, Ondřej ; KIRPICHEV, Sergey B. ; ROCKLIN, Matthew ; KUMAR, AMiT ; IVANOV, Sergiu ; MOORE, Jason K. ; SINGH, Sartaj ; RATHNAYAKE, Thilina ; VIG, Sean ; GRANGER, Brian E. ; MULLER, Richard P. ; BONAZZI, Francesco ; GUPTA, Harsh ; VATS, Shivam ; JOHANSSON, Fredrik ; PEDREGOSA, Fabian ; CURRY, Matthew J. ; TERREL, Andy R. ; ROUČKA, Štěpán ; SABOO, Ashutosh ; FERNANDO, Isuru ; KULAL, Sumith ; CIMRMAN, Robert ; SCOPATZ, Anthony: SymPy: symbolic computing in Python. In: *PeerJ Computer Science* 3 (2017), S. e103
- [NM95] NELDER, John A. ; MEAD, Roger: A simplex method for function minimization. In: *The Computer Journal* 7 (1995), Nr. 4, S. 308–313
- [NW06] NOCEDAL, Jorge ; WRIGHT, Stephen: *Numerical optimization, series in operations research and financial engineering*. New York : Springer Verlag, 2006
- [Os69] OSBORNE, Mike R.: On Shooting Methods for Boundary Value Problems. In: *Journal of Mathematical Analysis and Applications* 27 (1969), Nr. 2, S. 417–433

- [Pap15] PAPAGEORGIOU, Markos: *Optimierung: Statische, Dynamische, Stochastische Verfahren*. 4. Auflage. Berlin u.a. : Springer Verlag, 2015
- [PB07] PANNOCCHIA, Gabriele ; BEMPORAD, Alberto: Combined Design of Disturbance Model and Observer for Offset-Free Model Predictive Control. In: *IEEE Transactions on Automatic Control* 52 (2007), Nr. 6, S. 1048–1053
- [PKS09] PISCHINGER, Rudolf ; KLELL, Manfred ; SAMS, Theodor: *Thermodynamik der Verbrennungskraftmaschine*. 3. Auflage. Wien, New York : Springer Verlag, 2009
- [Pol70] POLYAK, Boris T.: Iterative methods using lagrange multipliers for solving extremal problems with constraints of the equation type. In: *USSR Computational Mathematics and Mathematical Physics* 10 (1970), Nr. 5, S. 42–52
- [Pow69] POWELL, Michael J.: A method for nonlinear constraints in minimization problems. In: *Optimization, Academic Press* (1969), S. 283–298
- [QB03] QIN, S J. ; BADGWELL, Thomas A.: A survey of industrial model predictive control technology. In: *Control Engineering Practice* 11 (2003), Nr. 7, S. 733–764
- [Rao10] RAO, Anil V.: A survey of numerical methods for optimal control. In: *Advances in the Astronautical Sciences*, 2010
- [Raw00] RAWLINGS, James B.: Tutorial Overview of Model Predictive Control. In: *IEEE Control Systems Magazine* 20 (2000), Nr. 3, S. 38–52
- [RB09] RIEBISCH, Matthias ; BODE, Stephan: Software-Evolvability. In: *Informatik-Spektrum* 32 (2009), Nr. 4, S. 339–343
- [RHG13] REINHARDT, Rüdiger ; HOFFMANN, Armin ; GERLACH, Tobias: *Nichtlineare Optimierung*. Berlin u.a. : Springer Spektrum, 2013
- [RKV16] RÖSSLER, Michael ; KOCH, Thomas ; VELJI, Amin: Increasing the NO₂/NO_x ratio in Diesel engines to enhance exhaust aftertreatment. In: *International Congress : SIA Powertrain*. Rouen : SIA, 2016
- [Roc73] ROCKAFELLAR, R. T.: The multiplier method of Hestenes and Powell applied to convex programming. In: *Journal of Optimization Theory and Applications* 12 (1973), Nr. 6, S. 555–562
- [RVJ⁺17] RÖSSLER, Michael ; VELJI, Amin ; JANZER, Corina ; KOCH, Thomas ; OLMANN, Matthias: Formation of Engine Internal NO₂: Measures to Control the

- NO₂/NO_x Ratio for Enhanced Exhaust After Treatment. In: *SAE International Journal of Engines* 10 (2017), Nr. 4
- [SB92] STRYK, Oskar von ; BULIRSCH, Roland: Direct and indirect methods for trajectory optimization. In: *Annals of Operations Research* 37 (1992), Nr. 1, S. 357–373
- [Sch99] SCHRAUDOLPH, Nicol N.: A fast, compact approximation of the exponential function. In: *Neural Computation* 11 (1999), Nr. 4, S. 853–862
- [SCH⁺04] SCHABER, Peter M. ; COLSON, James ; HIGGINS, Steven ; THIELEN, Daniel ; ANSPACH, Bill ; BRAUER, Jonathan: Thermal decomposition (pyrolysis) of urea in an open reaction vessel. In: *Thermochimica Acta* 424 (2004), Nr. 1-2, S. 131–142
- [Sei10] SEITZ, Hanna: *Contributions to the Minimum Linear Arrangement Problem*, Ruprecht-Karls-Universität Heidelberg, Diss., 2010
- [SGD⁺15] SHATROV, Mikhail G. ; GOLUBKOV, Leonid N. ; DUNIN, Andrey U. ; YAKOVENKO, Andrey L. ; DUSHKIN, Pavel V.: Influence of high injection pressure on fuel injection performances and diesel engine working process. In: *Thermal Science* 19 (2015), Nr. 6, S. 2245–2253
- [SP97] STORN, Rainer ; PRICE, Kenneth: Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. In: *Journal of Global Optimization* 4 (1997), S. 341–359
- [SP07] SKOGESTAD, Sigurd ; POSTLETHWAITE, Ian: *Multivariable feedback control: analysis and design*. 2. Auflage. New York : Wiley, 2007
- [THE75] TSANG, T. H. ; HIMMELBLAU, D. M. ; EDGAR, T. F.: Optimal control via collocation and non-linear programming. In: *International Journal of Control* 21 (1975), Nr. 5, S. 763–768
- [TT21] THIEL, Michael ; TIBKEN, Bernd: CFD and Testbench Measurements of the Deutz TCD12 V6 Diesel Engine. In: *Zenodo* (2021)
- [TT22] THIEL, Michael ; TIBKEN, Bernd: A Diesel Engine Air Path Model with a WG and an ETV for Exhaust Temperature Controller Design and Embedded Control. In: *Part D: SAGE Journal of Automobile Engineering* (2022)
- [VGO⁺20] VIRTANEN, Pauli ; GOMMERS, Ralf ; OLIPHANT, Travis E. ; HABERLAND, Matt ; REDDY, Tyler ; COURNAPEAU, David ; BUROVSKI, Evgeni ; PETERSON, Pearu ; WECKESSER, Warren ; BRIGHT, Jonathan ; OTHERS: SciPy 1.0:

- fundamental algorithms for scientific computing in Python. In: *Nature Methods* 17 (2020), Nr. 3, S. 261–272
- [Vik17] VIKHAR, Pradnya A.: Evolutionary algorithms: A critical review and its future prospects. In: *Proceedings - International Conference on Global Trends in Signal Processing, Information Computing and Communication, ICGTSPICC 2016* (2017), S. 261–265
- [VNS03] VOITSIS, Elias ; NTZIACHRISTOS, Leonidas ; SAMARAS, Zissis: Particulate matter mass measurements for low emitting diesel powered vehicles: What's next? In: *Progress in Energy and Combustion Science* 29 (2003), Nr. 6, S. 635–672
- [Wat07] WATTER, Holger: *Hydraulik und Pneumatik*. Wiesbaden : Vieweg+Teubner Verlag, 2007
- [WB95] WELCH, Greg ; BISHOP, Gary: *An Introduction to the Kalman Filter*. Chapel Hill, 1995
- [WB10] WANG, Yang ; BOYD, Stephen: Fast model predictive control using online optimization. In: *IEEE Transactions on Control Systems Technology* 18 (2010), Nr. 2, S. 267–278
- [WCV⁺07] WILLEMS, Frank ; CLOUDT, Robert ; VAN DEN EIJNDEN, Edwin ; VAN GENDEREN, Marcel ; VERBEEK, Ruud ; DE JAGER, Bram ; BOOMSMA, Wiebe ; VAN DEN HEUVEL, Ignace: Is closed-loop SCR control required to meet future emission targets? In: *SAE Technical Papers* (2007)
- [WD97] WALES, David J. ; DOYE, Jonathan P.: Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. In: *Journal of Physical Chemistry A* 101 (1997), Nr. 28, S. 5111–5116
- [WE10] WAHLSTRÖM, Johan ; ERIKSSON, Lars: *Modeling of a Diesel Engine with Intake Throttle, VGT and EGR*. 2010
- [WE11] WAHLSTRÖM, Johan ; ERIKSSON, Lars: Modelling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 225 (2011), Nr. 7, S. 960–986
- [Wen11] WENDEL, Jan: *Integrierte Navigationssysteme*. 2. Auflage. München : Oldenbourg Verlag, 2011

- [WJ82] WATSON, Neil ; JANOTA, Marian: *Turbocharging the Internal Combustion Engine*. London u.a. : The Macmillan Press, 1982
- [YKB⁺04] YIM, Sung D. ; KIM, Soo J. ; BAIK, Joon H. ; NAM, In S. ; MOK, Young S. ; LEE, Jong H. ; CHO, Byong K. ; OH, Se H.: Decomposition of urea into NH₃ for the SCR process. In: *Industrial and Engineering Chemistry Research* 43 (2004), Nr. 16, S. 4856–4863