TECHNISCHE UNIVERSITÄT WIEN

DIPLOMARBEIT

Optimierungsbasierte Generierung von Testsignalen für die Analyse von Regelkreisen

ausgeführt am

Institut für
Analysis und Scientific Computing
TU Wien

unter der Anleitung von

Assistant Prof. Dr. Andreas Körner

und

Dipl.-Ing. Markus Gurtner

durch

Paul Westhauser
Matrikelnummer: 11778877

Wien, am 2. Januar 2025

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Kurzfassung

In der vorliegenden Arbeit wird ein optimierungsbasierter Signalgenerator entwickelt, der Worst-Case Testsignale zur Analyse von Regelkreisen erzeugt. Dabei verfolgt das zugrundeliegende Optimierungsproblem das Ziel, die Abweichung zwischen Soll- und Ausgangssignal eines Regelkreises zu maximieren, um herausfordernde Testsignale für nichtlineare Regelkreise zu generieren. Die Struktur des Signalgenerators wurde so gestaltet, dass die erzeugten Testsignale physikalisch sinnvoll und realisierbar sind. Da die Bestimmung geeigneter Testsignale ohne Systemwissen eine besondere Herausforderung darstellt, ermöglicht der entwickelte Signalgenerator die Bestimmung von herausfordernden Testsignalen für nichtlinearen Regelkreise auch ohne Kenntnis des zugrunde liegenden Systems.

Für den optimierungsbasierter Signalgenerator wurden geeignete Optimierungsverfahren ausgewählt und es wurden Simulationsstudien durchgeführt, um diese Algorithmen hinsichtlich absoluter Genauigkeit, Wiederholgenauigkeit und Konvergenzverhalten zu untersuchen. Zur Verbesserung der Optimierung wurden Startwerte auf Basis schon durchgeführter Simulationsstudien gewählt und deren Effektivität experimentell geprüft. Der anwendungsorientierte Teil dieser Arbeit umfasst, neben ausführlichen Simulationsstudien auch die Validierung des Signalgenerators an einem realen Regelkreis. Die durchgeführten Experimente zeigen, dass der entwickelte Ansatz eine Bestimmung von Schwachstellen ermöglicht und somit die Effektivität des Verfahrens bestätigt. Damit leistet diese Arbeit einen vielversprechenden Beitrag zur Erzeugung von herausfordernden Testsignalen für nichtlineare Regelkreise.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Abstract

This work introduces an optimization-based signal generator for the systematic analysis of closed-loop control systems. The generator is designed to produce worst-case test signals by formulating an optimization problem that maximizes the deviation between the desired input signal and the system's output signal. The generated signals are not only challenging but also physically meaningful, ensuring their applicability to real-world scenarios. Since identifying suitable test signals without prior system knowledge poses a significant challenge, the developed signal generator enables the determination of such signals for nonlinear control systems without requiring knowledge of the underlying system.

To identify the most suitable optimization algorithms, we conducted comprehensive simulation studies. These algorithms were evaluated based on criteria such as absolute accuracy, repeatability, and convergence behavior. Additionally, initial values for the optimization process were determined using a clustering method and their performance was experimentally tested. The application-oriented part of this work also includes the validation of the signal generator on a real control system. The conducted experiments demonstrate that the developed approach enables a systematic identification of weaknesses, thereby confirming the effectiveness of the method. Thus, this work contributes to the model-free evaluation of closed-loop control systems.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Danksagung

Ich möchte mich bei meinen Betreuern Markus Gurtner und Andreas Pfeffer bedanken, die mich mit ihren wertvollen Anregungen und konstruktiven Diskussionen bei der Ausarbeitung dieser Arbeit unterstützt haben. Mein Dank gilt auch meinem Kollegen Paul Strasser für seine hilfreichen Beiträge und die angenehme Zusammenarbeit. Ein besonderer Dank geht an meine Eltern, meine Freundinnen und Freunde sowie meine Freundin, die mich während dieser Arbeit und meinem gesamten Studium stets ermutigt und unterstützt haben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am 2. Januar 2025

Name des_r Autors_in

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Inhaltsverzeichnis

1. Einleitung **1**

2. Grundlagen dynamischer Systeme und globaler Optimierungsverfahren **4**
2.1. Dynamische Systeme 4
2.2. Globale Optimierungsverfahren 7
2.2.1. Genetic Algorithm 8
2.2.2. Differential evolution algorithm 9
2.2.3. Particle swarm optimization 10
2.2.4. Covariance Matrix Adaptation Evolution Strategy 12
2.2.5. Surrogate Algorithmus 12
2.2.6. Patternsearch 15
2.2.7. Genetic algorithm-Patternsearch 16
2.2.8. Monte-Carlo Optimierung 17

3. Optimierungsbasierter Signalgenerator **20**
3.1. Deterministischer Signalgenerator 20
3.1.1. Signalerzeugung 21
3.1.2. Aufbau des deterministischen Signalgenerators 22
3.2. Formulierung des Optimierungsproblems 27
3.3. Wahl der Startwerte durch Clustering 36

4. Fallstudien geschlossener Regelkreise **40**
4.1. Feder-Masse System 40
4.2. Pneumatisches Ventil 42

5. Experimentelle Bewertung **47**
5.1. Beschreibung des Bewertungssystems 47
5.1.1. Bewertungskriterien 48
5.1.2. Reihung der Bewertungsvektoren 51
5.2. Ergebnisse der Simulationsstudien 53
5.2.1. Gesamtbewertung 53
5.2.2. lineares Feder-Masse System 56
5.2.3. nichtlineares Feder-Masse System 59
5.2.4. Pneumatisches Ventil mit Druck als Eingangsgröße 61
5.2.5. Pneumatisches Ventil mit Massenstrom als Eingangsgröße 64
5.2.6. Fazit zur Auswahl eines Optimierungsalgorithmus 66
5.3. Simulationsergebnisse zu den gewählten Startwerten 68
5.4. Bewertung am Prüfstandsaufbau 71

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

6. Zusammenfassung und Ausblick **74**

A. Anhang **76**
A.1. Verwendung der Optimierer 76
A.2. Zusätzliche Tabelle und Grafiken zu den Experimenten 80

Abbildungsverzeichnis **84**

Tabellenverzeichnis **87**

Literaturverzeichnis **89**

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Abkürzungsverzeichnis

CMAES *Covariance Matrix Adaptation Evolution Strategy*

DE *Differential evolution algorithm*

EOP *Expensiv Optimization Problem*

FMSL *Feder-Masse System linear*

FMSN *Feder-Masse System nichtlinear*

GA *Genetic algorithm*

GA-PS *Genetic algorithm-Patternsearch*

MAE *Mean Absolute Error*

MC *Monte-Carlo*

PS *Patternsearch*

PSO *Particle swarm optimization*

PVED *Pneumatisches Ventil Eingangsgröße Druck*

PVEM *Pneumatisches Ventil Eingangsgröße Massenstrom*

RMSE *Root Mean Square Error*

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

1. Einleitung

Während der Entwicklung werden nichtlinearer Regelkreise üblicherweise in Simulationsumgebungen getestet und bewertet. Diese numerischen Untersuchungen sind vorteilhaft, da im nichtlinearen Fall nur bei speziellen Gegebenheiten allein an dem mathematischen Model des Regelkreises die Stabilität und Robustheit nachgewiesen werden kann. Bei numerischen Robustheitsuntersuchungen sind vor allem die Berücksichtigung von Parameterschwankungen und variierenden externen Einflussgrößen etabliert und ausführlich diskutiert. Nichtsdestotrotz stellt sich die Frage, welches Sollsignal für die Bewertung des Regelkreises herangezogen werden sollte. In der linearen Systemtheorie dienen häufig der Sprung oder die Rampe als geeignete, universell einsetzbare Testsignale. Im nichtlinearen Fall lässt sich aus diesen einzelnen Testsignale aber nicht auf das Gesamtverhalten des Regelkreises zurückschließen. In der Praxis ist die Wahl eines geeigneten Testsignals eine anspruchsvolle Aufgabe, welche detailliertes Systemwissen voraussetzt, um tatsächlich die potentiellen Schwächen des Regelkreises aufzuzeigen.

Regelkreise sind dynamische Systeme, die aus einem Regler und einer Strecke bestehen, siehe Abbildung 1.1. Die Aufgabe eines Reglers $\mathcal{R}$ ist typischerweise den Streckeneingang u so vorzugeben, dass der Streckenausgang y_{sys} mit möglichst geringer Abweichung dem Soll- oder Referenzsignal y^d folgt. Wir werden in Abschnitt 2.1 detaillierter darauf eingehen.

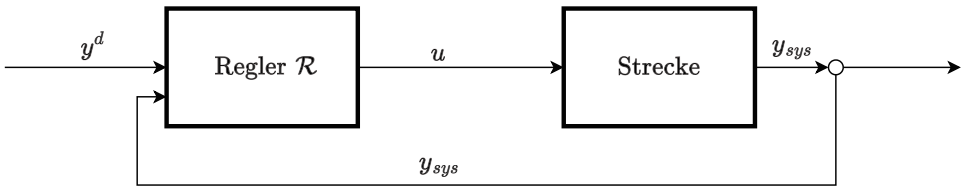


Abbildung 1.1.: Blockschaltbild eines Regelkreises.

Ziel dieser Arbeit ist die Entwicklung eines Ansatzes zur Bestimmung von Testsignalen für nichtlineare Regelkreise, der sowohl auf in der Praxis existierende Systeme als auch auf Modelle von Regelkreisen anwendbar ist. Traditionell erfordert die Analyse und Bewertung nichtlinearer Regelkreise spezifisches Systemwissen, um geeignete Testsignale zu identifizieren. Diese Arbeit strebt an, die Generierung solcher Testsignale auch ohne systembezogenes Vorwissen zu ermöglichen. Die Herausforderung dieser Zielsetzung liegt in der Vielzahl potenziell möglicher Testsignale, deren Auswahl ohne spezifisches Fachwissen kaum eingeschränkt werden kann. Daher soll die Bestimmung von Worst-Case Testsignalen fokussiert werden, die gezielt Schwachstellen in Regelkreisen identifizieren. Solche Signale dienen der effizienten Bewertung der Leistungsfähigkeit und Robustheit der Regelkreise.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Der Schwerpunkt liegt insbesondere auf Anwendungen, bei denen Simulationen und umfangreiche Messreihen mit hohem Zeit- und Kostenaufwand verbunden sind. Ziel ist es, Schwachstellen der Regelung präzise zu bestimmen und damit eine effiziente Bewertung der Regelkreise zu ermöglichen.

Ausgehend von diesem Ziel wird ein optimierungsbasierter Signalgenerator entwickelt, der sowohl universell einsetzbar ist als auch möglichst wenige Simulationsaufrufe oder Messvorgänge benötigt. Zudem wird er die Erzeugung physikalisch sinnvoller Testsignale sicher stellen. Der Signalgenerator G soll ein stetiges Sollsignal y^d auf Basis eines Parametervektors $\boldsymbol{\xi}$ erzeugen, welches durch Diskretisierung in $\boldsymbol{y}^d$ überführt wird. Eine detaillierte Beschreibung dieses Prozesses findet sich in Abschnitt 3.1. Um damit ein Worst-Case Testsignal für einen beliebigen Regelkreis Σ zu bestimmen, wird das

$$\begin{aligned} \max_{\boldsymbol{\xi}\in\Xi_i} f(\boldsymbol{\xi}, \Sigma, \|\cdot\|_p) &= \|\boldsymbol{y}^d - \boldsymbol{y}_{sys}\|_p, \quad p \in \{1,2\}\,, \\ \boldsymbol{y}_{sys} &= \Sigma(\boldsymbol{y}^d), \\ \boldsymbol{y}^d &= G(\boldsymbol{\xi}), \end{aligned}$$

durch Gleichung (3.14) formulierte Optimierungsproblem gelöst. Die Kostenfunktion f beschreibt die Abweichung zwischen dem generierten Sollsignal $\boldsymbol{y}^d$ und dem Ausgangssignal $\boldsymbol{y}_{sys}$ des Regelkreises Σ, welches durch $\Sigma(\boldsymbol{y}^d)$ gegeben wird. Die Menge Ξ_i umfasst die möglichen Parameter des Signalgenerators. Ziel der Optimierung wird es sein, die Differenz zwischen dem Soll- und dem Ausgangssignal zu maximieren. Dadurch soll ein Testsignal gefunden werden, das eine maximale Regelabweichung hervorruft, ohne spezifisches Systemwissen zu erfordern.

Aufgrund der Formulierung des Optimierungsproblems wird die Frage aufgeworfen, welcher Optimierungsalgorithmus die Anforderungen dieser Arbeit am besten erfüllt. Zur Beantwortung dieser Frage wird eine umfassende Literaturrecherche durchgeführt [1, 2, 3, 4], um vielversprechende Optimierungsalgorithmen zu identifizieren. Die Auswahl der in Abschnitt 2.2 beschriebenen Algorithmen, die auf Grundlage dieser Recherche getroffen wird, soll anschließend experimentell bewertet werden, um deren Eignung systematisch zu analysieren. Die Bewertung wird anhand der Kriterien *absolute Genauigkeit*, *Wiederholgenauigkeit* und *Konvergenzverhalten* erfolgen, um den geeignetsten Algorithmus für die betrachtete Problemstellung zu bestimmen. Während der Simulationsstudien zeigen sich Ähnlichkeiten in den Signalstrukturen, die zur Entwicklung von Startwerten mithilfe einer Clustermethode führen. Dieser Ansatz soll dazu beitragen, die Effizienz des optimierungsbasierten Signalgenerators weiter zu verbessern. Abschließend wird in einem Experiment an einem realen Regelkreis untersucht, ob der entwickelte optimierungsbasierte Signalgenerator in der Lage sein wird, Testsignale zu generieren, die Schwachstellen des Systems zuverlässig identifizieren können.

Im Folgenden wird die Gliederung dieser Arbeit vorgestellt. In Kapitel 2 werden die relevanten Grundlagen und Methoden erläutert. Im speziellen sind dies die Beschreibung dynamischer Systeme, die Struktur von Regelkreisen sowie die verwendeten globalen Optimierungsalgorithmen. Kapitel 3 behandelt die Entwicklung des optimierungsbasierten Si-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

gnalgenerators. Es werden die Struktur des Generators und das zugrunde liegende Optimierungsproblem beschrieben. Außerdem werden die Simulationsstudien formuliert und darüber hinaus wird ein Ansatz zur Bestimmung geeigneter Startwerte basierend auf ersten Simulationsergebnissen mithilfe einer Clustering-Methode vorgestellt. Dem folgt Kapitel 4, das mit einem Feder-Masse System und einem pneumatischen Ventil insgesamt 4 Simulationsstudien für die Durchführung der Experimente vorstellt. Der experimentelle Nachweis findet sich schlussendlich in Kapitel 5. Es erfolgt eine Bewertung der Optimierungsalgorithmen, anhand eines Bewertungssystems, eine Untersuchung der Regelkreise mit Worst-Case Testsignalen und die Überprüfung der Startwerte. Abschließend wird der optimierungsbasierte Signalgenerator am Prüfstand validiert.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

2. Grundlagen dynamischer Systeme und globaler Optimierungsverfahren

Das folgende Kapitel definiert die wesentlichen Begriffe und Methoden, die in dieser Arbeit verwendet werden. Es liefert die notwendigen Konzepte, um dynamische Systeme zu beschreiben und stellt globale Optimierungsverfahren vor.

Im ersten Abschnitt wird der Begriff des dynamischen Systems eingeführt. Im Gegensatz zu statischen Systemen, deren Ausgänge nur vom aktuellen Eingang abhängen, haben bei dynamischen Systemen auch vergangene Systemeingänge einen Einfluss auf das momentane Verhalten. Unter den vielzähligen Modellierungsmöglichkeiten für dynamische Systeme nimmt in der Regelungstechnik die Zustandsraumdarstellung eine zentrale Rolle ein. Sie ermöglicht es, die Entwicklung des Systems über die Zeit hinweg anhand von Differentialgleichungssystemen erster Ordnung zu beschreiben. Auf dieser Basis werden wir uns mit Regelkreisen befassen und dabei auch einfache, aber sehr häufig anzutreffende, Regler kurz vorstellen. Abschließend wird beschrieben, welche Art von Regelkreis in dieser Arbeit untersucht wird.

Der zweite Abschnitt widmet jenen sich globalen Optimierungsverfahren, die in dieser Arbeit als zentraler Ansatz zur Lösung des Optimierungsproblems dienen. Globale Optimierungsverfahren dienen der Suche nach Lösungen in komplexen, multidimensionalen Suchräumen, in denen die Kostenfunktion nichtlinear, nicht differenzierbar oder unstetig sein kann. Ein Auszug verschiedener globalen Optimierungsalgorithmen wird hier vorgestellt und im Hinblick auf die potenzielle Eignung für unsere Optimierungsaufgabe bewertet.

2.1. Dynamische Systeme

Dynamische Systeme sind ein grundlegendes Konzept das in verschiedensten wissenschaftlichen und technischen Disziplinen Anwendung findet. In diesem Abschnitt geben wir einen Überblick über die grundlegenden Beschreibungen und Eigenschaften dynamischer Systeme, wobei wir uns an [5, 6, 7] orientieren.

Ein *System* ist die Verbindung verschiedener Komponenten, die miteinander interagieren, um bestimmte Aufgaben zu erfüllen [5]. Wesentlich sind dabei die Eingangs- und Ausgangsgrößen, welche die Wechselwirkung zwischen System und Umgebung beschreiben. Dabei beschreiben die *Eingangsgrößen* das Wirken der Systemumgebung auf das System. Bezeichnet werden die Eingangsgrößen meist mit $u_1, \ldots, u_r$, wobei $r \in \mathbb{N}$ die Anzahl der Eingangsgrößen beschreibt. Wir unterscheiden bei den Eingangsgrößen zwischen Stellgrößen, welche gezielt gewählt werden können um das System zu beeinflussen, und Störgrößen, welche nicht unserer Kontrolle obliegen. *Ausgangsgrößen*, $y_1, \ldots, y_s$, wobei $s \in \mathbb{N}$ die Anzahl der Ausgangsgrößen beschreibt, folgen aus dem den Eigenschaften und Verhalten des Systems und wirken auf die Umgebung. Messtechnisch erfassbare Ausgangsgrößen werden auch als

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

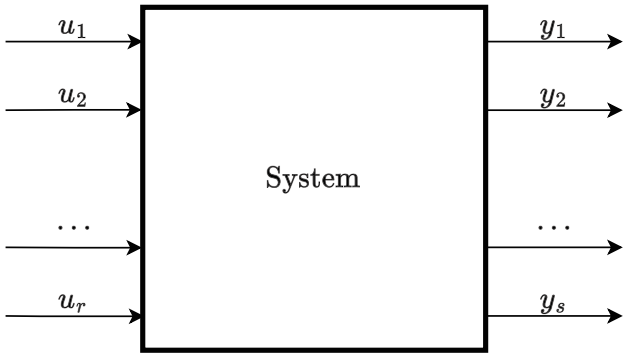


Abbildung 2.1.: Darstellung eines dynamischen Systems.

Messgrößen bezeichnet. Oft werden Systeme mithilfe von Blockschaltbildern dargestellt, siehe Abbildung 2.1.

Ein System wird als *statisch* bezeichnet, wenn sein Verhalten ausschließlich von den Eingangsgrößen zum aktuellen Zeitpunkt abhängt. Wenn das Verhalten jedoch zusätzlich von früheren Eingangsgrößen beeinflusst wird, spricht man von einem *dynamischen* System. In dynamischen Systemen beschreiben wir mit $x_1, \ldots, x_n$ die *Zustandsgrößen*. Diese besitzen die Eigenschaft, dass die Ausgangsgrößen $y_1(t), \ldots, y_s(t)$ des Systems zu einem beliebigen Zeitpunkt t durch den Verlauf der Eingangsgrößen des Systems $u_1(\tau), \ldots, u_r(\tau)$ auf dem Intervall $t_0 \leq \tau \leq t$ und die Werte von $x_1(t_0), \ldots, x_n(t_0)$ für ein beliebiges t_0 festgelegt sind.

Mathematisch kann ein dynamisches System, als System gewöhnlicher Differentialgleichungen dargestellt werden. Dabei gehen wir davon aus, dass das System eine endlichen Anzahl an Zustandsgrößen $n \in \mathbb{N}$ hat. Dies führt mit

$$\begin{aligned}
\boldsymbol{u} &= u(t) = [u_1(t), \ldots, u_r(t)]^{\mathrm{T}}, \quad u(t) \in C^1(I, \mathbb{R}^r), \\
\boldsymbol{y} &= y(t) = [y_1(t), \ldots, y_s(t)]^{\mathrm{T}}, \quad y(t) \in C^1(I, \mathbb{R}^s), \\
\boldsymbol{x} &= x(t) = [x_1(t), \ldots, x_n(t)]^{\mathrm{T}}, \quad x(t) \in C^1(I, \mathbb{R}^n),
\end{aligned}$$

wobei $t \in I$ und $I = [a, b] \subseteq \mathbb{R}$, zu der flexiblen und kompakten Schreibweise

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{2.1a}$$

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, t). \tag{2.1b}$$

Dabei bezeichnet man (2.1a) als Zustandsdifferentialgleichungen mit Anfangsbedingungen und (2.1b) als Ausgangsgleichungen. Generell bezeichnet $\boldsymbol{u}$ den Eingang, $\boldsymbol{y}$ den Ausgang und $\boldsymbol{x}$ den Zustand des dynamischen Systems. Wird der Zustand $\boldsymbol{x}$ als Element eines Vektorraumes betrachtet, so nennt man diesen Vektorraum auch *Zustandsraum*. Der Zustand eines Systems zum Zeitpunkt t kann dann als Element des Zustandsraumes dargestellt werden. Die Kurve all dieser Punkte im Zustandsraum für veränderliche Zeit t in einem Zeitintervall wird als *Trajektorie* des Systems bezeichnet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Im Fokus dieser Arbeit steht eine spezifische Klasse dynamischer Systeme, der sogenannte *Regelkreis*. Ein Regelkreis Σ ist ein dynamisches System, das aus einem *Regler* $\mathcal{R}$ und einer *Strecke* besteht. Abbildung 1.1 zeigt ein Blockschaltbild eines geschlossenen Regelkreises. Der Regler ermittelt auf Basis des Sollsignals y^d und des gemessenen Signals y_{sys}, welches die Messgrößen der Strecke repräsentiert, die Stellgröße u so, dass die *Regelabweichung* $e = y_{sys} - y^d$ minimiert wird. Das Ziel ist, das Ausgangssignal y_{sys} möglichst exakt an das Sollsignal y^d heran zu führen. Die Strecke stellt wiederum ein dynamisches System dar, dessen Eingang u vom Regler bestimmt wird. Die Regelkreise, welche wir betrachten werden, werden immer nur eine skalare Messgröße haben, also $y_{sys} \in C(I, \mathbb{R})$. Je nach Komplexität der zu regelnden Strecke und den spezifischen Anforderungen an den geschlossenen Regelkreis kann die Komplexität des Reglers deutlich variieren. Die Bandbreite reicht dabei von sehr einfachen, statischen Reglern bis hin zu komplexen, dynamischen Regelstrukturen. Im Folgenden werden drei verschiedene Regler vorgestellt, die im weiteren Verlauf der Arbeit Anwendung finden.

P-Regler: Einfache Regler, wie der Proportionalregler (P-Regler), stellen eine der einfachsten Regelungsstrukturen dar. Diese Regler können mithilfe statischer Abbildungen beschrieben werden und kommen häufig in Systemen zum Einsatz, bei denen einfache, proportionale Korrekturen genügen, um eine stabile Regelung zu erzielen. Sie können mit der Gleichung

$$u(t) = k_P e(t) \tag{2.2}$$

dargestellt werden, wobei k_P den Verstärkungsfaktor des P-Anteils beschreibt

PI-Regler: Ein PI-Regler kombiniert eine proportionale Komponente (2.2) mit einer integrierenden Komponente und kann als

$$u(t) = k_P e(t) + k_I \int_0^t e(\tau)\ \mathrm{d}\tau \tag{2.3}$$

darstellt werden. Die proportionale Komponente mit Verstärkungsfaktor k_P reagiert auf den aktuellen Fehler und sorgt für eine unmittelbare Korrektur. Die integrierende Komponente mit Verstärkungsfaktor k_I hingegen summiert die Fehler über die Zeit, um verbleibende systematische Abweichungen auszugleichen.

PID-Regler: In der Praxis sind PID-Regler besonders verbreitet, da sie neben proportionalen und integrierenden auch differenzierende Anteile enthalten. Strukturell werden sie mittels

$$u(t) = k_P e(t) + k_I \int_0^t e(\tau)\ \mathrm{d}\tau + k_D \frac{\mathrm{d}}{\mathrm{d}t} e(t) \tag{2.4}$$

beschrieben. Die Differenzialkomponente mit Verstärkungsfaktor k_D reagiert auf die Änderungsrate des Fehlers und trägt dazu bei, Schwingungen zu dämpfen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Diese Regler können in der vorhin beschriebenen Zustandsdarstellung (2.1) beschrieben werden und sind damit selbst dynamische Systeme. Da ein Gesamtsystem auch aus gekoppelten Teilsystemen zusammengesetzt werden kann, gilt der gesamte Regelkreis Σ wieder als ein dynamisches System. Wir fassen den Zusammenhang zwischen dem Sollsignal y^d und der Messgröße des gesamten Systems, welche wir im Weiteren auch als *Ausgangssignal* y_{sys} bezeichnen werden, in der kompakten Schreibweise

$$y_{sys} = \Sigma(y^d) \tag{2.5}$$

zusammen. Diese Schreibweise werden wir auch verwenden, um die Messungen von realen Regelkreisen darzustellen, da diese auch vom Sollsignal abhängen.

2.2. Globale Optimierungsverfahren

In diesem Kapitel widmen wir uns verschiedenen globalen Optimierungsverfahren, die sich zur Lösung komplexer Optimierungsprobleme eignen. Der Fokus liegt auf Methoden, die in der Lage sind, nichtlineare, nicht differenzierbare und mehrdimensionale Probleme zu bewältigen und dabei ein globales Optimum finden. Ziel dieser Verfahren ist es, in möglichst wenigen Iterationen eine optimale oder bei vorzeitigem Abbruch suboptimale Lösung zu erreichen.

Zunächst klären wir die Anforderungen, die wir an globale Optimierungsverfahren stellen und welche Erwartungen an die ausgewählten Algorithmen bestehen. Ein Optimierungsproblem ist im Allgemeinen definiert als

$$\min_{x \in M} \quad f(x), \tag{2.6a}$$

$$g_i(x) \leq 0, \quad i = 1, \ldots, m, \tag{2.6b}$$

$$h_j(x) = 0, \quad j = 1, \ldots, p. \tag{2.6c}$$

Dabei bezeichnet $f : M \mapsto \mathbb{R}$ die Kostenfunktion, M die zulässige Menge und g_i und h_j die Nebenbedingungen. Motiviert durch die Eigenschaften der Kostenfunktion, welche in Abschnitt 3.2 detailliert diskutiert werden, ordnen wir das Optimierungsproblem anhand der folgenden Kategorien ein.

Rechenaufwand: Die Berechnung der Kostenfunktion ist als rechenintensiv einzustufen, da diese dynamische Systeme auswerten wird.

Diffenzierbarkeit: Die Kostenfunktion ist nicht differenzierbar, sodass gradientenbasierte Methoden auszuschließen sind.

Stetigkeit: Die Kostenfunktion weist Unstetigkeiten auf.

Parameterraum: Die Kostenfunktion hängt mehreren verschiedenen Parametern ab

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

In der relevanten Literatur wie [1, 2, 3, 4, 8, 9, 10] werden Optimierungsprobleme mit dieser Kategorisierung als *Expensiv Optimization Problem* (EOP) klassifiziert. Im Folgenden betrachten wir nun sieben vielversprechende Algorithmen für Probleme dieser Art. Jeder Algorithmus wird mithilfe eines Flussdiagramms dargestellt, sowie die entsprechenden Vor- und Nachteile diskutiert. Wenn im Folgenden vom *Lösungsraum* die Rede ist, wird damit ein Vektorraum $\mathbb{R}^n$ bezeichnet, wobei n die Anzahl der Parameter darstellt. Die *Lösungsvektoren*, manchmal auch nur Vektoren, sind dabei Vektoren in diesem Raum, deren Einträge jeweils einen einzelnen Parameter repräsentieren. Damit stellt der Lösungsraum den Parameterraum als Vektorraum dar.

2.2.1. Genetic Algorithm

Der *Genetic algorithm* (GA), auch genetischer Algorithmus, wurde ursprünglich von John Holland [11] entwickelt und später von David E. Goldberg angewendet und popularisiert [12]. GA ist ein evolutionäres Optimierungsverfahren, das von den Mechanismen der natürlichen Selektion inspiriert ist. Ziel ist es, eine Population von Individuen, welche hier einzelne Lösungen darstellen, iterativ zu verbessern, sodass eine optimale Lösung im Lösungsraum gefunden wird. Jedes Individuum wird durch einen Vektor, das sogenannte Chromosom, repräsentiert. Die Elemente des Chromosoms stellen die Gene dar und kodieren damit die Lösung des Optimierungsproblems. Die Bewertung der Eignung einzelner Individuen erfolgt über eine *Fitnessfunktion*, welche eine Kostenfunktionsauswertung beinhaltet.

Der GA beginnt mit der Initialisierung einer Population, wobei jedes Individuum eine potenzielle Lösung für das Optimierungsproblem darstellt. Die Initialisierung erfolgt entweder zufällig durch spezielle stochastische Funktionen oder durch eine im Vorhinein getroffene Vorgabe. Die Fitness der Individuen wird über eine deterministische Kostenfunktion bewertet, wobei fittere Individuen höhere Überlebens- und Reproduktionschancen haben. Die Nachkommen der Individuen werden durch einen von 3 sogenannten *genetische Operatoren*, das sind Selektion, Crossover oder Mutation, erzeugt. Bei der Selektion werden die Nachkommen basierend auf der besten Fitness der Elterngeneration ausgewählt, beim Crossover (Rekombination) werden Gene von zwei Elternindividuen kombiniert, um neue Lösungen zu generieren. Durch die Mutation werden Gene zufällig verändert, um die Vielfalt der Population zu bewahren und ein Verharren in lokalen Minima zu vermeiden. Diese genetischen Operatoren verleihen dem Algorithmus seine stochastische Natur, da sowohl die Auswahl der Operatoren als auch die Prozesse selbst zufallsbedingte Elemente enthalten. Dieser Prozess wird über mehrere Generationen fortgeführt, bis ein Abbruchkriterium erfüllt ist. Typische Abbruchkriterien sind eine maximale Anzahl von Generationen oder das Erreichen einer bestimmten Fitnessschwelle. In Abbildung 2.2 ist ein Flussdiagramm zum GA zu sehen.

Vorteile: Der GA eignet sich zur Optimierung von nichtlinearen, nicht differenzierbaren und unstetigen Problemen in größeren Parameterräumen. Ein wesentlicher Vorteil liegt in seiner Fähigkeit, simultan mehrere Regionen des Parameterraums zu explorieren, wodurch die Gefahr des Verbleiben in lokalen Minima reduziert wird. Diese Eigenschaft macht den GA besonders geeignet für globale Optimierungsaufgaben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

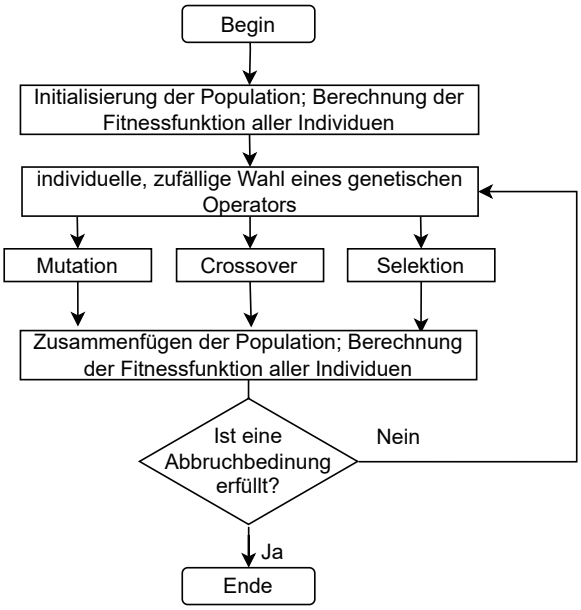


Abbildung 2.2.: Flussdiagramm des Genetic algorithm.

Nachteile: Der Algorithmus ist rechenintensiv, da für jede Generation zahlreiche Fitnessbewertungen und damit Kostenfunktionsaufrufe notwendig sind. Er kann in einigen Fällen langsam konvergieren. Außerdem liegt bei GA eine signifikante Varianz der Ergebnisse vor, da er viele stochastische Prozesse beinhaltet.

2.2.2. Differential evolution algorithm

Der *Differential evolution algorithm* (DE) ist ein evolutionäres, stochastisches Optimierungsverfahren, das in den 1990er Jahren von Rainer Storn und Kenneth Price entwickelt wurde [13]. DE ist speziell für stetige Optimierungsprobleme ausgelegt. Ähnlich wie zuvor GA, verwendet auch DE eine Population von potenziellen Lösungsvektoren, genannt Individuen, die potenzielle Lösungen darstellen, und verbessert diese durch die wiederholte Anwendung von Mutation, Crossover und Selektion.

Zu Beginn des Algorithmus wird eine Population von Individuen zufällig im *Lösungsraum* verteilt. Danach gliedert sich jede Iteration in 3 Phasen. In der ersten Phase, der *Mutationsphase*, wird für jedes Individuum ein zufälliges Individuum aus der Population ausgewählt. In der zweiten Phase, der *Rekombinationsphase*, wird die Differenz der beiden Individuen skaliert und zu einem dritten Individuum addiert, um eine neue Position im Lösungsraum zu erzeugen. Dieses neue Individuum kann außerhalb der aktuellen Population liegen und somit neue Lösungsmöglichkeiten erschließen. In der dritten Phase, der *Selektionsphase*, wird die Fitness der rekombinierten Lösung mit der des ursprünglichen Individuums verglichen, um zu entscheiden, ob das neue Individuum das ursprüngliche ersetzt. Dieser ite-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

rative Prozess wird wiederholt, bis ein Abbruchkriterium, wie eine maximale Anzahl an Generationen oder eine zufriedenstellende Fitnessschwelle, erreicht ist. Die beste Lösung der finalen Population stellt das Ergebnis des Algorithmus dar. Damit unterscheidet sich dieser Algorithmus vom GA, da er immer dieselben 3 Phasen hintereinander ausgeführt werden. In GA hingegen wird stochastisch gewählt, welche der 3 genetischen Operationen ausgeführt wird. Obwohl der Algorithmus für stetige Probleme ausgelegt ist, empfiehlt auf Grund empirischer Benchmarks die Literatur [1, 2] DE für EOPs. Ein Flussdiagramm ist in Abbildung 2.3 dargestellt.

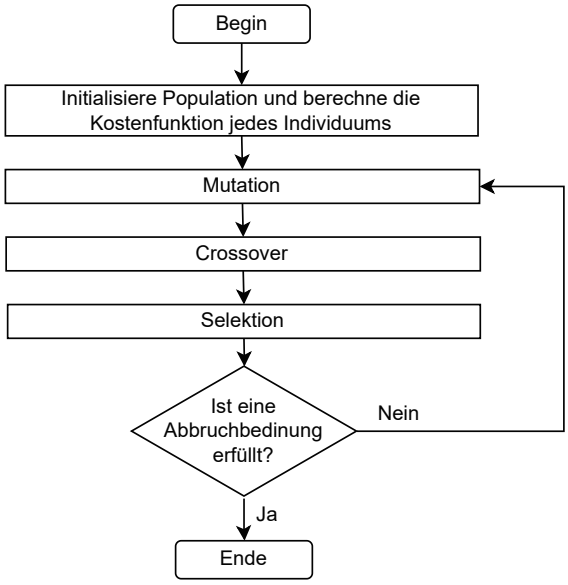


Abbildung 2.3.: Flussdiagramm des Differential evolution algorithm.

Vorteile: DE zeichnet sich durch eine hohe Effizienz in stetigen, nichtlinearen Problemen mit einer großen Anzahl an Parametern aus. Dank seiner globalen Suchstrategie neigt er nicht dazu, in lokalen Minima zu verharren.

Nachteile: Die Konvergenzgeschwindigkeit kann bei einigen komplexen Problemen langsam sein. DE weist eine hohe Varianz der Ergebnisse auf, auch wenn sie niedriger als in GA ausfällt. Zudem ist DE für stetige Kostenfunktionen konzipiert, was die Fähigkeit des Algorithmus, ein Optimum zu finden, beeinträchtigen könnte.

2.2.3. Particle swarm optimization

Der *Particle swarm optimization* (PSO) Algorithmus wurde 1995 von James Kennedy und Russell Eberhart entwickelt. Der PSO Algorithmus ist inspiriert von Schwarmverhalten in der Natur, wie etwa dem Verhalten von Vogel- oder Fischschwärmen [14]. PSO ist ein

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

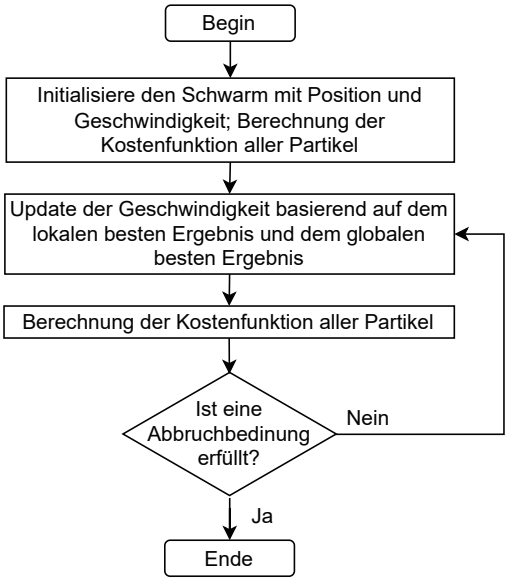


Abbildung 2.4.: Flussdiagramm zur Particle swarm optimization.

populationsbasiertes, stochastisches Optimierungsverfahren, das zur Lösung kontinuierlicher, nichtlinearer Optimierungsprobleme verwendet wird. Das Ziel des PSO Algorithmus besteht darin, durch die iterative Anpassung der Positionen und Geschwindigkeiten der Partikel eine optimale oder zumindest näherungsweise optimale Lösung für das gegebene Optimierungsproblem zu ermitteln.

Jeder Partikel wird durch einen Positions- und einen Geschwindigkeitsvektor beschrieben. Der Schwarm wird, entweder durch eine Vorgabe oder durch eine stochastisch Funktion initialisiert. Die Geschwindigkeit eines Partikels wird in jeder Iteration so aktualisiert, dass sich der Partikel sowohl in Richtung der besten Position in seiner individuellen Historie als auch zur besten Position des Schwarms bewegt. Somit übernimmt die Geschwindigkeit hier die Rolle eines Gradienten. Dies ermöglicht es dem Algorithmus, den Lösungsraum systematisch zu durchsuchen und die global beste Lösung zu finden.

Der Algorithmus läuft iterativ ab, bis ein Abbruchkriterium, wie die Erreichung einer bestimmten Fitness oder eine festgelegte Anzahl an Iterationen, erfüllt ist. Der Partikel mit der besten Lösung stellt das Ergebnis des Algorithmus dar. Der PSO Algorithmus wurde gewählt, da er in der Literatur [2] als ein guter Algorithmus für das Lösen von EOPs gilt. In der Abbildung 2.4 ist ein Flussdiagramm dazu dargestellt.

Vorteile: Der PSO Algorithmus ist einfach zu implementieren, bietet eine gute Konvergenzgeschwindigkeit und eignet sich gut für große Parametermengen, da er mehrere Positionen simultan bewertet. Vorteilig für die Varianz ist, dass der Algorithmus nur in der Initialisierung stochastische Prozesse verwendet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Nachteile: Der PSO Algorithmus weist die Gefahr auf, suboptimale Lösungen zu finden, und ist stark von den initialen Startwerten abhängig, insbesondere bei kleinen Populationen. Dies kann nachteilig für die Varianz der Ergebnisse sein. Darüber hinaus ist der Algorithmus primär für stetige Kostenfunktionen ausgelegt, wodurch seine Leistung bei diskreten oder nicht stetigen Problemen eingeschränkt sein kann.

2.2.4. Covariance Matrix Adaptation Evolution Strategy

Die *Covariance Matrix Adaptation Evolution Strategy* (CMAES) ist eine evolutionäre Optimierungsstrategie, die in den 1990er Jahren von Nikolaus Hansen und Andreas Ostermeier entwickelt wurde [15]. CMAES ist speziell für nichtlineare, multidimensionale als auch stetige Optimierungsprobleme geeignet und gehört zu den flexibelsten evolutionären Algorithmen. Der Algorithmus passt iterativ die Kovarianzmatrix der Verteilung der Kandidatenlösungen an, um eine optimale Suchrichtung zu finden und den Lösungsraum effektiv zu durchsuchen. Man kann sich diesen Algorithmus als eine Newton-Methode ohne Gradienteninformationen vorstellen. Da dieser Algorithmus ziemlich komplex ist, werden wir einen kurzen Überblick über die Funktionsweise geben, wobei mehr Details in [16] zu finden sind.

Zu Beginn wird eine Population von Kandidatenlösungen zufällig generiert und die Schrittweite sowie die Kovarianzmatrix werden initialisiert. Die Kovarianzmatrix, welche in jeder Iteration aktualisiert wird, modelliert die Form der Verteilung. Sie wird in weiterer Folge verwendet, um die Varianz der Suchrichtung an die Struktur des Lösungsraums anzupassen. In jeder Iteration werden die besten Individuen ausgewählt und der Erwartungswert sowie die Kovarianz der Verteilung entsprechend angepasst. Die Schrittweite wird dynamisch geändert, um die Annäherung an das Optimum zu beschleunigen oder zu verlangsamen, je nachdem, wie sich die Population im Lösungsraum verteilt.

Die Iterationen setzen sich fort, bis ein Abbruchkriterium erreicht ist. Obwohl der Algorithmus ursprünglich für stetige Funktionen entwickelt wurde, ist er auch für unstetige, multidimensionale Optimierungsprobleme geeignet. CMAES eignet sich sowohl für lokale [15] als auch globale Optimierungen, [8, 17, 18]. Ein Flussdiagramm zu diesem Algorithmus ist in Abbildung 2.5 dargestellt.

Vorteile: CMAES ist besonders leistungsfähig in hochdimensionalen, nichtlinearen und multidimensionalen Lösungsräumen. Er benötigt eine geringe Startpopulation, womit er für teure Kostenfunktion gut geeignet ist. Ähnlich wie der PSO Algorithmus verwendet CMAES weniger stochastische Prozesse als GA oder DE und zeigt damit eine niedrigere Varianz der Ergebnisse.

Nachteile: CMAES ist für stetige Kostenfunktionen ausgelegt. CMAES hat selbst hohe Berechnungskosten und ist damit für einfache Kostenfunktionen nur eingeschränkt zu empfehlen. Außerdem ist die Implementierung komplex.

2.2.5. Surrogate Algorithmus

Der Surrogate Algorithmus ist ein Optimierungsverfahren, das sich insbesondere für kostspielige oder zeitintensive Optimierungsprobleme eignet. Er ist ausgelegt für Black-Box Modelle, also Kostenfunktionen, von denen kaum Informationen bekannt sind [19]. Der

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

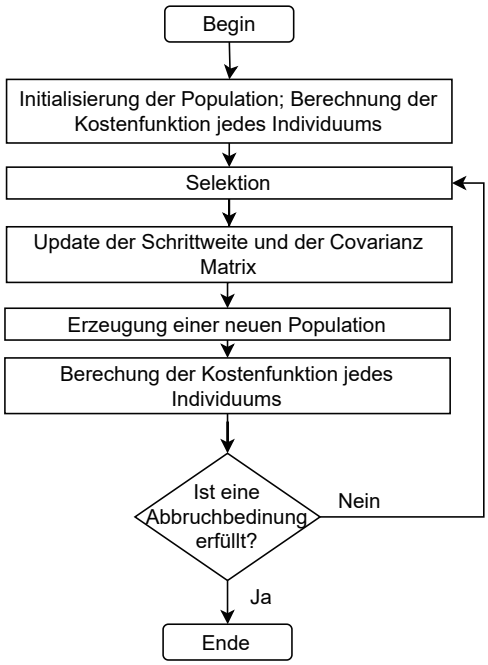


Abbildung 2.5.: Flussdiagramm der Covariance Matrix Adaptation Evolution Strategy.

Surrogate Algorithmus verwendet ein Ersatzmodell, genannt *Surrogat*, das die tatsächliche Kostenfunktion approximiert und somit weniger Auswertungen der eigentlichen Black-Box Funktion benötigt. Der Surrogate Algorithmus kommt häufig in der Optimierung von Computational Fluid Dynamics und der Regelungstechnik zum Einsatz [10].

Der Surrogate Algorithmus ist mehr eine Klasse von Algorithmen als ein spezieller Algorithmus. Die Unterschiede der Algorithmen liegen in der Erstellung des Surrogate Modells und an den Optimierungsmethoden des Modells. Alle Surrogate Algorithmen beginnen mit der Auswahl einer initialen Menge von Lösungen (Evaluationspunkten), für welche die exakte Kostenfunktion berechnet wird. Diese Punkte dienen als Grundlage für die Erstellung eines Surrogate Modells, das die Beziehungen zwischen den Optimierungsparametern und der Kostenfunktion approximiert. Diese Startwerte werden entweder stochastisch gewählt oder gezielt vorgegeben. Das Surrogate Modell kann mithilfe unterschiedlicher Methoden erstellt werden, die gängigsten sind die Folgenden.

Kriging, auch Gaussian Process Regression: Ein probabilistisches Modell, das die Unsicherheit in der Approximation der Kostenfunktion modelliert und als Gauß-Prozess Modell fungiert.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Radiale Basisfunktionen, auch Radial Basis Function (RBF): Eine Interpolationsmethode, bei der die Gewichtung der Basisfunktionen vom Abstand zu bestimmten Stützpunkten abhängt [20].

Polynome: Ein Ansatz, der Polynome zur Annäherung der Kostenfunktion verwendet. Dieser ist simpel, aber meist auf Probleme niedriger Dimension beschränkt [19].

Der iterative Prozess startet mit dem Aufbau eines Surrogate Modells, das die Kostenfunktion basierend auf den bereits ausgewerteten Punkten approximiert. Das Surrogate Modell wird genutzt, um einen neuen Evaluationspunkt zu bestimmen. Hier werden verschiedene Optimierer verwendet, als auch verschiedene Optimierungskriterien, um eine Balance zwischen der Exploration neuer Regionen und der Exploitation bereits vielversprechender Regionen zu finden. Der ausgewählte Punkt wird mithilfe der Kostenfunktion evaluiert, und der Datensatz wird aktualisiert, indem der neue Punkt und sein Ergebnis dem Modell hinzugefügt werden. Diese Schritte werden iterativ wiederholt, bis ein Abbruchkriterium erreicht ist. Ein Flussdiagramm, welches die Funktionsweise der Surrogate Methode skizziert, ist in Abbildung 2.6 zu finden.

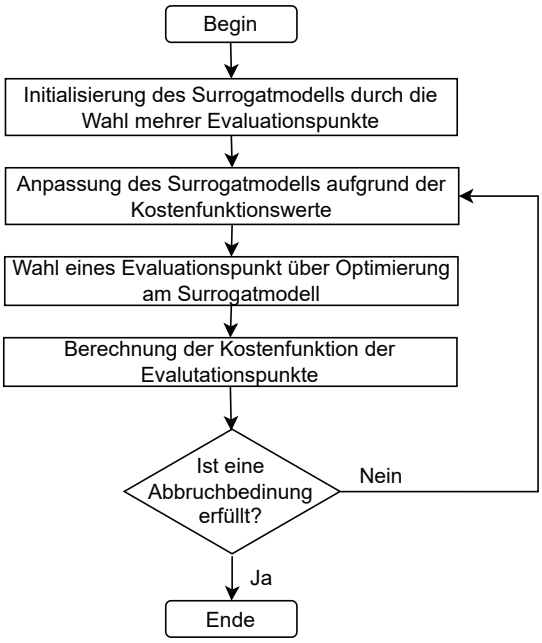


Abbildung 2.6.: Flussdiagramm des Surrogate Algorithmus.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Vorteile: Der Surrogate Algorithmus ist besonders effizient bei rechenintensiven Optimierungsproblemen, da er die Anzahl der Auswertungen der tatsächlichen Kostenfunktion drastisch reduziert. Außerdem funktioniert der Surrogate Algorithmus auf unstetigen, nichtlinearen, komplexen, hochdimensionalen Kostenfunktionen. Ausgehend davon welches zugrunde liegende Surrogate Modell gewählt wurde, verwendet der Algorithmus wenige stochastische Prozesse und liefert damit Ergebnisse mit geringer Varianz.

Nachteile: Die Genauigkeit des Surrogate Modells ist stark von der initialen Datenmenge und der gewählten Modellierungsstrategie abhängig. Zudem kann das Modell bei sehr komplexen oder hochdimensionalen Suchräumen erhebliche Rechenressourcen benötigen.

2.2.6. Patternsearch

Der *Patternsearch* (PS) Algorithmus ist ein deterministisches, gradientenfreies Optimierungsverfahren. Der Algorithmus ist besonders gut geeignet für Optimierungsprobleme, bei denen die Zielfunktion unstetig, nicht differenzierbar ist. Der PS Algorithmus wurde ursprünglich in [21] vorgestellt und später in [22] weiterentwickelt.

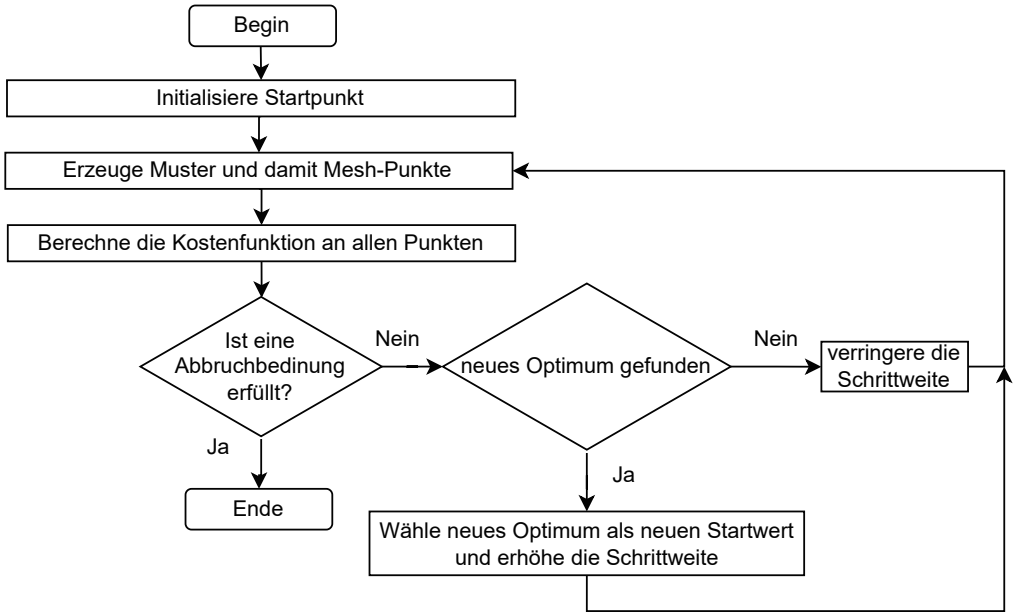


Abbildung 2.7.: Flussdiagramm des PS Algorithmus.

Die Iterationsvorschrift des PS Algorithmus beginnt mit der Auswahl eines Startpunkts. Ausgehend von diesem Punkt werden in der Umgebung neue Punkte gemäß vordefinierter Muster generiert. An diesen Mesh-Punkten wird der Wert der Kostenfunktion berechnet. Zeigt der Kostenfunktionswert eines Mesh-Punkts eine Verbesserung im Vergleich zu den

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

anderen, wird dieser Punkt zum Ausgangspunkt der nächsten Iteration. Gleichzeitig passt der Algorithmus die Schrittweite an, um die Annäherung an das Optimum zu beschleunigen oder gegebenenfalls zu verlangsamen. Falls keiner der Mesh-Punkte eine Verbesserung liefert, wird die Schrittweite reduziert, sodass sich die Suche auf einen kleineren Bereich konzentriert. Dieser iterative Prozess wird fortgesetzt, bis ein Abbruchkriterium, wie etwa eine festgelegte Anzahl von Iterationen, erreicht ist. Ein Flussdiagramm des PS Algorithmus ist in Abbildung 2.7 dargestellt.

Vorteile: Patternsearch ist für unstetige, nicht differenzierbare und nichtlineare Funktionen ausgelegt, was ihn für komplexe Optimierungsprobleme nützlich macht.

Nachteile: Der Algorithmus kann bei höheren Dimensionen ineffizient sein und langsam konvergieren. Die Wahl des Startwertes wirkt sich deutlich auf das Ergebnis aus woraus bei einer zufälligen Wahl des Startwertes eine hohe Varianz folgt.

2.2.7. Genetic algorithm-Patternsearch

Genetic algorithm-Patternsearch (GA-PS) ist ein hybrider Algorithmus, er kombiniert den *Genetic algorithm* GA mit dem *Patternsearch* Algorithmus, um die Vorteile beider Optimierungsmethoden zu nutzen. Diese sequentielle Anwendung ist besonders geeignet für die globale Optimierung komplexer und nichtlinearer Probleme, da der GA eine breit angelegte Exploration des Lösungsraums ermöglicht, während der PS Algorithmus in einem zweiten Schritt eine präzisere lokale Suche durchführt.

Der GA dient als erster Schritt dieses hybriden Algorithmus, indem er eine Population von Kandidatenlösungen generiert und diese über mehrere Generationen hinweg verbessert. Dabei nutzt der GA die Mechanismen der Selektion, des Crossovers und der Mutation, um neue Individuen zu erzeugen und die Population auf optimale Bereiche des Lösungsraums zu lenken. Nach einer festgelegten Anzahl von Generationen wird der PS-Algorithmus auf das aktuell beste Individuum angewendet. Dabei überprüft der PS Algorithmus in der Umgebung der aktuell besten Lösung verschiedene Richtungen in dem Lösungsraum, um eine lokale Verbesserung zu finden. Dies ermöglicht eine präzisere Anpassung der Lösung. Der hybride Algorithmus endet, wenn ein Abbruchkriterium für PS erfüllt ist, beispielsweise eine maximale Anzahl von Kostenfunktionsaufrufen. Ein Flussdiagramm des GA-PS Algorithmus ist in Abbildung 2.8 dargestellt.

Vorteile: Die hybride Struktur verbindet die breite, explorative Suche des GA mit der schnellen, lokalen Optimierung des PS-Algorithmus. Dadurch eignet sich der hybride Algorithmus gut für komplexe und große Parametermengen.

Nachteile: Aufgrund der hybriden Struktur sind auch die Nachteile der beiden Algorithmen vorhanden. Zum Beispiel die Schwachstellen der hohen Varianz der Ergebnisse von GA.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

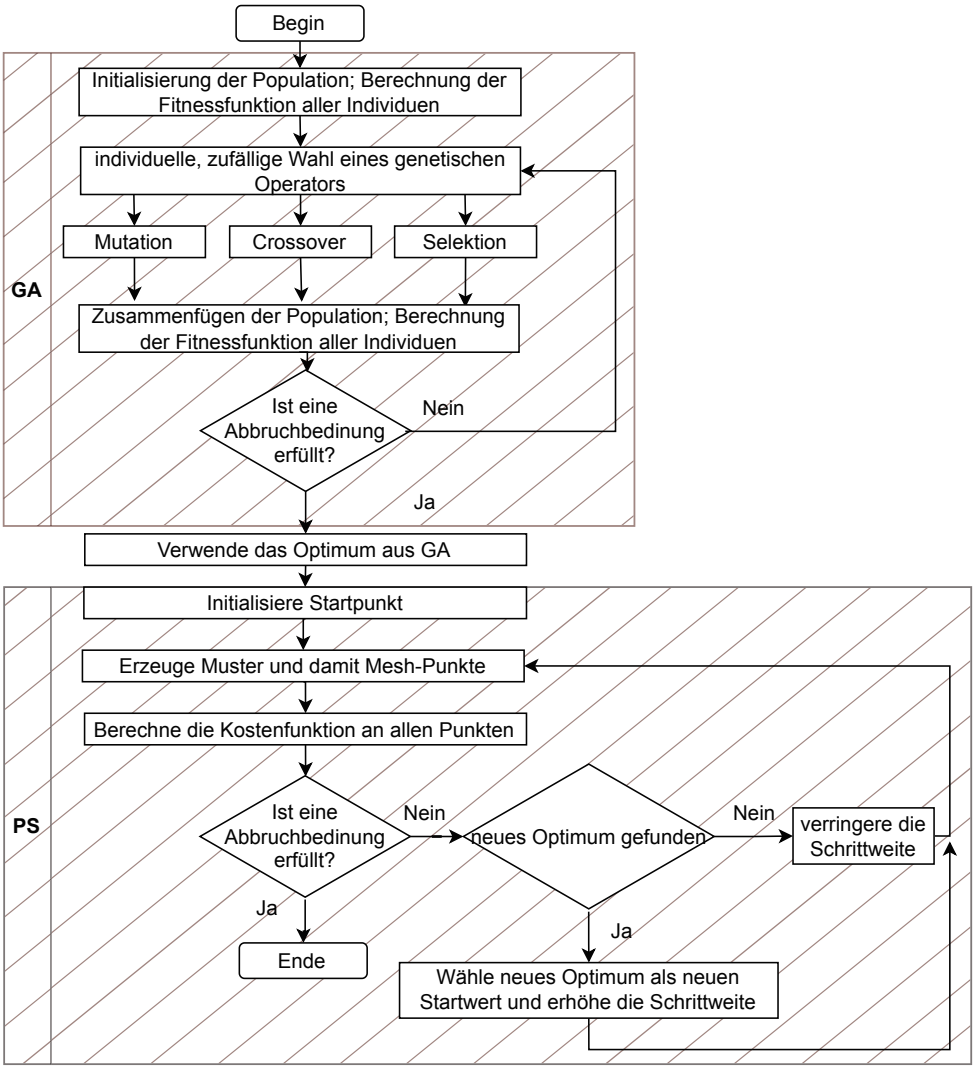


Abbildung 2.8.: Flussdiagramm des GA-PS-Algorithmus.

2.2.8. Monte-Carlo Optimierung

Monte-Carlo (MC) basierte Optimierungsmethoden sind sehr einfache und robuste Verfahren. Hier werden in jeder Iteration zufällig Optimierungsparameter bestimmt und die Kostenfunktion ausgewertet. Somit werden sukzessive neue Auswertungen der Kostenfunktion erstellt und jener Parameter mit den niedrigsten Kosten als potentielles Optimum verwenden. Wir beschreiben dieses globale Optimierungsverfahren, da wir es zur Verifikation und als Vergleichsmethode für andere globale Optimierungsmethoden verwenden wollen. Das sehr überschaubare Flussdiagramm zur MC Methode ist in Abbildung 2.9 dargestellt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Vorteile: Der MC Algorithmus ist universell einsetzbar und benötigt weder Gradienten noch spezifische Voraussetzungen an die Kostenfunktion. Durch die zufällige Natur des Algorithmus ist die Varianz der Ergebnisse niedrig.

Nachteile: Der Algorithmus konvergiert außerordentlich langsam und bietet bei einer beschränkten Anzahl an Kostenfunktionsaufrufen keine Garantie, eine Lösung in der Nähe des tatsächlichen Optimums zu finden.

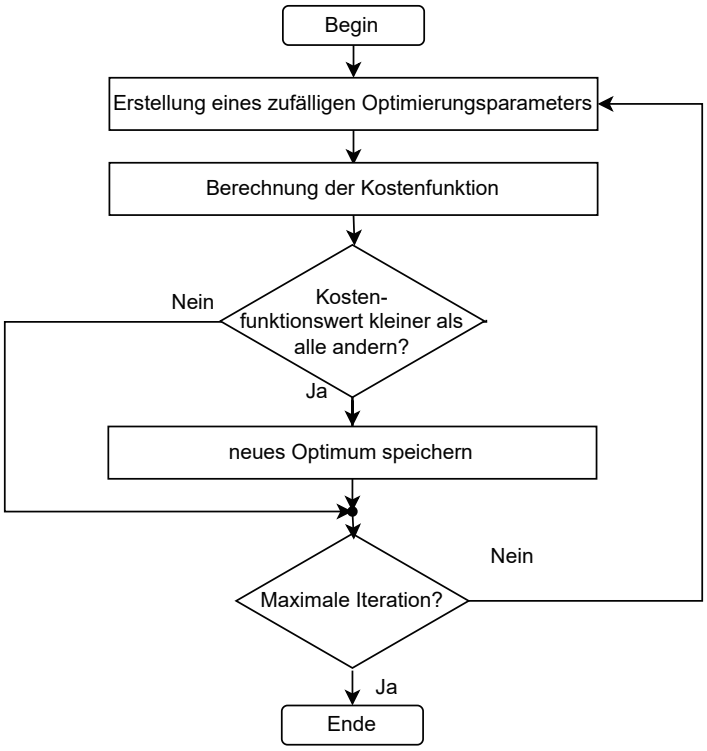


Abbildung 2.9.: Flussdiagramm des MC Algorithmus.

Abschließend haben wir in Tabelle 2.1 einen Vergleich der Algorithmen basierend auf der Literatur vorgenommen. Beim GA gehen wir von einer guten Effizienz aus, da die Größe der Parametermenge und die Unstetigkeit der Kostenfunktion keine wesentlichen Probleme darstellen sollten. Allerdings wird die hohe Varianz der Ergebnisse als eine Schwäche des GA betrachtet. DE und PSO könnten Schwierigkeiten mit der Unstetigkeit der Zielfunktion haben, jedoch erwarten wir eine bessere Varianz der Ergebnisse im Vergleich zum GA. CMAES erweist sich laut der Literatur als ein vielversprechender Kandidat, da er eine hohe Recheneffizienz aufweist, wobei die Unstetigkeit ebenfalls eine Herausforderung darstellen könnte. Der Surrogate Algorithmus bietet großes Potenzial, insbesondere wenn das geeignete Surrogate Modell gewählt wird, da hier eine bessere Varianz erwartet werden kann. Vom PS Algorithmus erwarten wir aufgrund der Analyse der Literatur keine großen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Vorteile, da die Größe der Parametermenge problematisch werden könnte. Der kombinierte GA-PS Ansatz erscheint vielversprechend, solange die Varianzproblematik des GA nicht dominiert. Vom MC-Algorithmus erwarten wir aufgrund der begrenzten Effizienz keine herausragenden Ergebnisse. Dennoch betrachten wir den MC-Algorithmus als einen nützlichen Referenzalgorithmus, um die Ergebnisse der anderen Algorithmen zu bewerten.

Kategorie	**GA**	**DE**	**PSO**	**CMAES**	**Surrogate**	**PS**	**GA-PS**	**MC**
Dimensionalität	$+$	$+\sim$	$+$	$+$	$+$	$-$	$+$	$-$
Recheneffizienz	$\sim$	$+\sim$	$+\sim$	$+$	$++$	$-\sim$	$+$	$--$
Varianz	$-\sim$	$\sim$	$+\sim$	$+\sim$	$\sim$	$-\sim$	$-\sim$	$+$
Unstetigkeit	$+$	$-\sim$	$-\sim$	$\sim$	$+$	$+$	$+$	$+$

Tabelle 2.1.: Vergleich globaler Optimierungsalgorithmen anhand der Literaturrecherche. Die verwendeten Symbole repräsentieren Bewertungsstufen, die von höchster Qualität, $++$, bis niedrigster Qualität, $--$, reichen. Die Symbole $+$ und $-$ kennzeichnen Zwischenstufen, während $\sim$, $+\sim$ und $-\sim$ Übergangsbewertungen darstellen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

3. Optimierungsbasierter Signalgenerator

Das vorliegende Kapitel stellt den optimierungsbasierten Signalgenerator, als wesentliches Herzstück dieser Arbeit, vor. Dieser wurde entwickelt, um Worst-Case Signale zum Testen geschlossener Regelkreise zu finden. Der Signalgenerator G generiert Sollsignale y^d, die als Eingangsgrößen für einen geschlossenen Regelkreis Σ dienen sollen. Dabei ist es das Ziel, ein Sollsignal y^d zu finden, bei dem der Ausgang des Regelkreises y_{sys} signifikante Abweichungen zum Sollsignal aufweist.

Der erste Abschnitt dieses Kapitels widmet sich der Vorstellung eines deterministischen Signalgenerators G welcher auf Basis eines Parametervektors $\boldsymbol{\xi}$ einen Signalverlauf $\chi(t)$ liefert. Bei der Erzeugung der Signale werden verschiedene mathematische Aspekte wie Stetigkeit und Änderungsrate berücksichtigt, um eine realistische Eingangsgröße zu erstellen. Der entwickelte Generator ermöglicht eine flexible und gezielte Anpassung der Signale, sodass diese optimal auf die spezifischen Eigenschaften des jeweiligen Regelkreises abgestimmt werden können.

Im Anschluss erfolgt die Formulierung eines Optimierungsproblems, um mit dem diskutierten Generator G Worst-Case Signale zu finden. Eine angepasste Kostenfunktion bewertet dabei den Fehler zwischen dem generierten Sollsignal und dem Ausgang des Regelkreises y_{sys}. Der Abschnitt umfasst zudem die Auswahl geeigneter Parametermengen sowie die Diskussion verschiedener Fehlermaße für den Optimierungsprozess. Nach einer Diskussion der wesentliche Charakteristika der Optimierungsaufgabe werden geeignete Experimente für die spätere Analyse des optimierungsbasierten Signalgenerators definiert.

Im letzten Abschnitt wird ein Ansatz zur gezielten Wahl der Startwerte erläutert. Basierend auf den bereits durchgeführten Experimenten werden mit Hilfe einer Clustering Methode ähnliche Parameter gruppiert, welche in den vergangenen Experimenten zu Worst-Case Signalen führten. Abschließend erfolgt die Formulierung eines Experiments, um die Eignung der so gewählten Startwerte zu bewerten.

3.1. Deterministischer Signalgenerator

Der deterministische Signalgenerator ist das Fundament des Optimierungsverfahrens, mit dem Worst-Case Signale produziert werden. Der Signalgenerator liefert Sollsignale für geschlossene Regelkreise. Ziel ist es, reproduzierbar ein Signal zu erzeugen, das stationäre und dynamische Bereiche aneinander reiht und damit ein realistisches Sollsignal für einen geschlossenen Regelkreis darstellt.

Der Generator basiert auf dem in [23] vorgestellten stochastischen Signalgenerator. Dieser erfüllte bereits die Anforderungen an ein realitätsnahes Sollsignal, operierte jedoch mit einem stochastischen Ansatz, der zufällige Signale erzeugt und daher keine Wiederholbarkeit gewährleistet. Zwar lässt sich mit einem solchen Ansatz über Monte-Carlo Simulationen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

nach Worst-Case Signalen suchen, doch ist dieser Vorgang sehr zeitaufwendig. Wir wollen diesen Prozess optimierungsbasiert gestalten, was eine deterministische Formulierung erfordert. So soll ein Sollsignal durch einen Parametervektor erzeugt werden, wodurch das Signal reproduzierbar wird. Diese deterministische Formulierung ist besonders wichtig, da die meisten globalen Optimierungsverfahren auf deterministischen Kostenfunktionen beruhen.

Zuerst werden die Signalanforderungen des Sollsignals behandelt, gefolgt von einer Betrachtung des Aufbaus des Signalgenerators. Dabei wird zunächst das grundlegende Konzept beschrieben, das die meisten Signalanforderungen erfüllt. Dem folgt die Definition des Parametervektors $\boldsymbol{\xi}$, über den das Signal des Generators deterministisch erzeugt werden kann. Abschließend wird die maximale Änderungsrate diskutiert und beschrieben, wie diese bestimmt werden kann.

3.1.1. Signalerzeugung

Das vom Signalgenerator erzeugte Signal soll ein Sollsignal für einen geschlossenen nichtlinearen Regelkreis sein. Anhand der geplanten Verwendung des Generators formulieren wir nun Anforderungen an das zu generierende Signal. Bei den ersten 4 Signalanforderungen orientieren wir uns an [23].

Definitions- und Zielbereich: Für unser Signal $\chi(\tau)$ gilt

$$\chi(\tau) : [0, 1] \longrightarrow [0, 1].$$

Damit ist $\chi(\tau)$ skalar. Die Festlegung auf diesen fixen Definitions- und Zielbereichs ist keine Einschränkung, da durch Transformationen ein beliebiger Zeitbereich $t \in [0, t_E]$

$$\tilde{\chi}(t) = \chi\left(\frac{t}{t_E}\right),$$

möglich wird, als auch mit

$$\tilde{\chi}(t) = x_1 + (x_2 - x_1)\chi(\tau),$$

ein beliebiger Zielbereich von $x \in [x_1, x_2]$ möglich ist.

Stetigkeit: Wir fordern vom Signal, dass es stetig ist. Diese Forderung ergibt sich, da sich Ausgänge von geschlossenen Regelkreisen üblicherweise nicht sprungartig ändern können.

Signalabfolge: Das Signal setzt sich aus einer Abfolge von stationären und dynamischen Bereichen zusammen. Ein dynamischer Bereich dient als Verbindung zwischen zwei stationären Phasen, wobei diese Übergänge, wie im letzten Punkt gefordert, stetig verlaufen müssen.

Anfangszustand: Das Signal startet stets in einem stationären Zustand. Dies entspricht nicht nur einem typischen Testszenario, bei dem der Ausgangspunkt ein konstanter Arbeitspunkt ist, sondern vereinfacht auch die präzise Initialisierung komplexer Simulationen mit mehreren dynamischen Subsystemen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Reproduzierbarkeit: Das Signal muss deterministisch erzeugbar sein, sodass für einen gegebenen Parametersatz stets dasselbe Signal erzeugt wird. Diese Eigenschaft ist entscheidend für die Optimierungsprozesse, da sie die Reproduzierbarkeit der Ergebnisse sicherstellt. Durch das deterministische Verhalten des Generators können Optimierungsalgorithmen zuverlässig und auf vorhersehbare Daten angewendet werden.

Änderungsrate: Es muss eine Möglichkeit bestehen, die maximale Änderungsrate der Signalabfolge zu begrenzen. Dies ist notwendig, um sicherzustellen, dass der Optimierungsprozess realistische Signale erzeugt, die im praktischen Kontext umsetzbar sind und innerhalb der Entwicklungsanforderungen an den geschlossenen Regelkreis liegen.

3.1.2. Aufbau des deterministischen Signalgenerators

Der Generator baut auf einem bestehenden stochastischen Ansatz auf, der für diese Arbeit jedoch erweitert und angepasst wurde, um die deterministischen Anforderungen zu erfüllen. Dies gewährleistet, dass der Generator ein Signal erzeugt, das für Optimierungsprozesse geeignet ist und reproduzierbare Signale liefert. Die Adaptionen hinsichtlich der Formulierung eines deterministischen Generators wurden dabei so gewählt, dass eine stochastische Erstellung von Signalen weiterhin möglich ist.

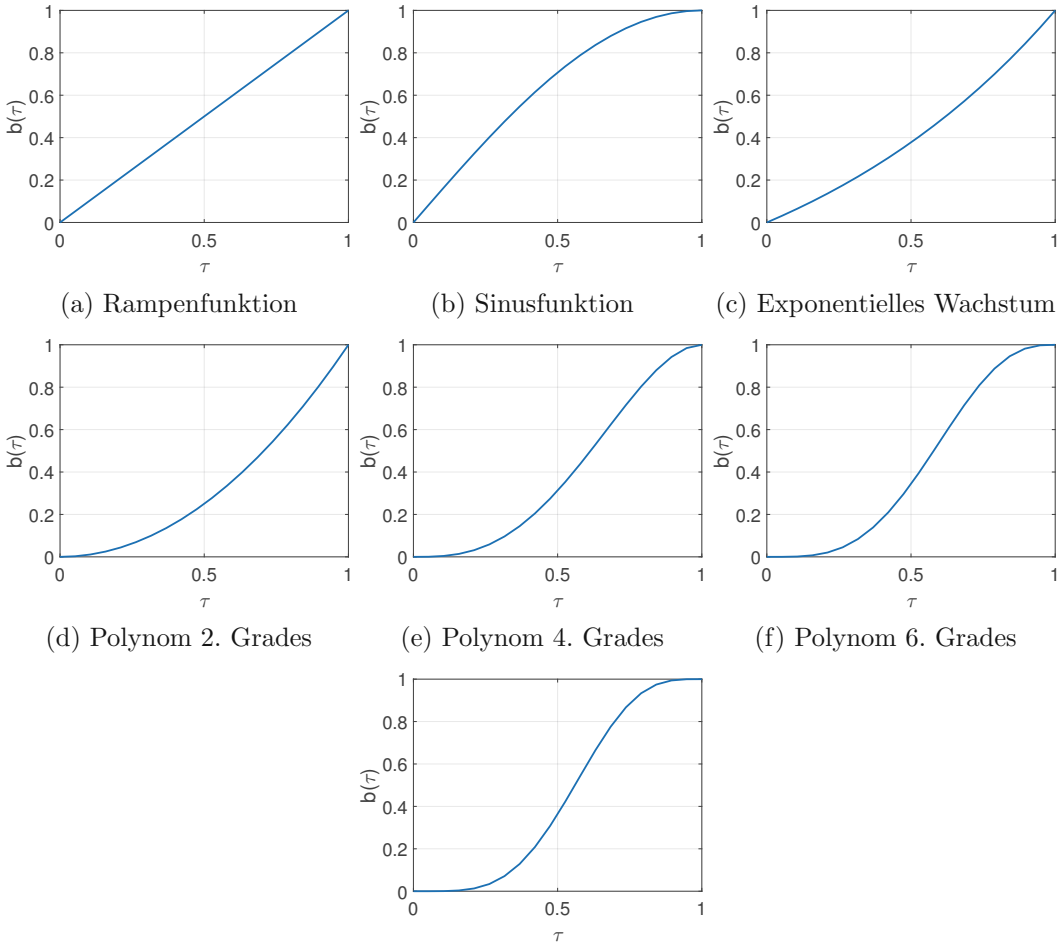
Wir beginnen mit der Definition des stationären Anteils. In stationären Bereichen gilt $\frac{\mathrm{d}\chi}{\mathrm{d}t} = 0$ bzw. $\chi = c$, wobei $c \in \mathbb{R}$. Damit ist der erste stationäre Teil des Signals über den Startwert χ_0 und die Dauer Δt_i vollständig definiert. Wir nennen den Zeitpunkt, an welchen sich stationärer und dynamischer Bereich ändern *Umschaltstelle*.

Um das Signal des dynamischen Abschnitts zu definieren, benötigen wir neben dem Startwert χ_0 und der Dauer Δt_i die Höhe $\Delta\chi_j$ und die Verbindungsfunktion $b_i(\tau)$. Durch $\chi_0 + \Delta\chi_i$ wird das Ende des dynamischen Abschnittes festgelegt. Die Funktion $b_i(\tau)$ wird so gewählt, dass die Randbedingungen $b(0) = 0$ und $b(1) = 1$ gelten. Tabelle 3.1 und Abbildung 3.1, beide übernommen aus [23], stellen 7 mögliche Funktionen dar. Der Index i von $b_i(\tau)$ gibt an, welche der Funktionen gewählt wird. Eine Ergänzung zusätzlicher dynamischen Bausteine ist somit jederzeit möglich. Anschließend wird $b_i(\tau)$ so skaliert, dass die Übergänge an den Rändern eines dynamischen Abschnitts kontinuierlich sind.

Baustein	$\boldsymbol{b(\tau)}$
Rampenfunktion	τ
Sinusfunktion	$\sin(\pi/2\tau)$
Exponentielles Wachstum	$(1-\mathrm{e}^{\tau})/(1-\mathrm{e}^{1})$
Polynom 2. Grades	τ^2
Polynom 4. Grades	$-3\tau^4 + 4\tau^3$
Polynom 6. Grades	$10\tau^6 - 24\tau^5 + 15\tau^4$
Polynom 8. Grades	$-35\tau^8 + 120\tau^7 - 140\tau^6 + 56\tau^5$

Tabelle 3.1.: Auflistung der Bausteine zur Konstruktion eines dynamischen Bereichs.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN



(a) Rampenfunktion (b) Sinusfunktion (c) Exponentielles Wachstum

(d) Polynom 2. Grades (e) Polynom 4. Grades (f) Polynom 6. Grades

(g) Polynom 8. Grades

Abbildung 3.1.: Elementare Bausteine b für die dynamischen Bereiche. Aus [23].

Mathematisch lässt sich dies durch

$$\chi(t) = \chi_0 + \Delta\chi_j \, b_k \left(\frac{t - t_i}{\Delta t_i} \right),$$

mit $t \in [t_i, t_i + \Delta t]$, erreichen.

In [23] wird für die zufällige Generierung von Testsignalen eine zufällige Wahl von $\Delta\chi, \Delta t$ und $b_i(\tau)$ vorgeschlagen. Für den optimierungsbasierten Ansatz ist hingegen eine Formulierung sinnvoll, bei der der Ausgang über einen Parametervektor $\boldsymbol{\xi}$ definiert wird.

Definition 1 (Parametervektor)**.** *Wir bezeichnen mit $\boldsymbol{\xi} \in \Xi_n$ den Parametervektor, aus dem der Generator G_{cont} ein Signal $G_{cont}(\boldsymbol{\xi}) = \chi_{\boldsymbol{\xi}}(t)$ erzeugt. Dieser besteht aus 3 Komponenten,*

$$\boldsymbol{\xi} := \begin{pmatrix} \boldsymbol{T} \\ \boldsymbol{B} \\ \boldsymbol{X} \end{pmatrix}.$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Dabei gilt:

(a) *Der Vektor* $\boldsymbol{T} \in [0,1]^{T_N}$ *wird als Umschaltstellenvektor bezeichnet. Er enthält Werte im Bereich* $[0,1]$*, die entweder bei 0 oder bei einer bestimmten Umschaltstelle liegen. Diese geben an, wann die nächste Umschaltstelle auftritt, und definieren damit die Länge jedes Bereichs. Dabei entspricht* Δt_i *dem* i*-ten Eintrag von* $\boldsymbol{T}$*, also* $\Delta t_i = T(i)$*.*

(b) *Der Vektor* $\boldsymbol{B} \in \{1,\dots,7\}^{B_N}$ *ist der Bausteinvektor. Er bestimmt, welche Bausteinfunktion für den dynamischen Abschnitt* j *gewählt wird, siehe Tabelle 3.1. Dabei entspricht das* k *von* $b_k(\tau)$ *dem* j*-ten Eintrag von* $\boldsymbol{B}$*, also* $b_k(\tau) = b_{B(j)}(\tau)$*. Aufgrund der Auswahl der Bausteine sind Werte von 1 bis 7 möglich.*

(c) *Wir nennen* $\boldsymbol{X} \in [0,1]^{X_N}$ *den Wertevektor. Er gibt mit* $X(1)$ *den Startwert des Signals vor.* $X(j)$ *beschreibt jeweils den Wert am Ende des* j*-ten dynamischen Bereichs. Die Werte von* $\boldsymbol{X}$ *liegen aufgrund der Signalvoraussetzungen zwischen* $[0,1]$*. Damit entspricht* $\Delta\chi_j$ *der Differenz aus* $X(j+1)$ *und* $X(j)$*, also* $\Delta\chi_j = X(j) - X(j+1)$ *und der Startwert* χ_0 *ist über* $X(1)$ *gegeben,* $\chi_0 = X(1)$*.*

Außerdem definieren wir

$$T_\Sigma(i) = \sum_{k=1}^{i} T(k), \quad \textit{mit} \quad T_\Sigma(0) = 0 \quad \textit{und} \quad T_\Sigma = \sum_{k=1}^{T_N} T(k).$$

Die Länge T_N von $\boldsymbol{T}$ bestimmt die Anzahl der dynamischen Teile eines Signals. Dabei ist T_N die Anzahl der Umschaltstellen. T_Σ beschreibt die letzte Umschaltstelle, damit muss diese, um den Signalanforderungen zu entsprechen, $T_\Sigma < 1$ erfüllen. Die Länge X_N von $\boldsymbol{X}$ als auch die Länge B_N von $\boldsymbol{B}$ hängt von der Anzahl der dynamischen Bereiche ab. Damit folgt für die Länge der Vektoren

$$B_N = \left\lceil \frac{T_N}{2} \right\rceil \text{ und } X_N = B_N + 1.$$

Definition 2 (Parametermenge)**.** *Sei* $N \in \mathbb{N}$*. Für* N *Umschaltstellen ist die Parametermenge* Ξ_N *durch,*

$$\Xi_N := \{\boldsymbol{\xi} \in M_N : T_\Sigma < 1\}$$

definiert. Dabei ist

$$M_N := [0,1]^N \times \{1,\dots,7\}^{\lceil \frac{N}{2} \rceil} \times [0,1]^{\lceil \frac{N}{2} \rceil + 1}.$$

Jeder Parametervektor $\boldsymbol{\xi} \in \Xi_N$ erzeugt ein Signal. Mit $c_i(\boldsymbol{\xi}, t)$ gemäß

$$c_i(\boldsymbol{\xi}, t) = \begin{cases} X(j), & i \in \{1,3,\dots\} \\ X(j) + (X(j+1) - X(j))\, b_{B(j)}\left(\frac{t - T_\Sigma(i)}{T(i)}\right), & i \in \{2,4,\dots\} \end{cases}, \tag{3.1}$$

wobei $j = \lceil i/2 \rceil$ gilt, lassen sich stationäre und dynamische Bereiche eines Signals beschreiben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die Gleichung des gesamten Signals $\chi(\boldsymbol{\xi}, t)$ kann man nun vollständig beschreiben als

$$\chi(\boldsymbol{\xi}, t) = \begin{cases} c_i(\boldsymbol{\xi}, t) & t \in [T_\Sigma(i-1), T_\Sigma(i)] \\ c_{\dim(\boldsymbol{T})+1}(\boldsymbol{\xi}, t) & t \in [T_\Sigma, 1]. \end{cases}$$

Dabei wird i über t festgelegt. Für einen geraden Index i in Ξ_i gilt, dass $G_{cont}(\boldsymbol{\xi})$ mit $\boldsymbol{\xi} \in \Xi_i$ mit einem stationären Teil endet. Für einen ungeraden Index i in Ξ_N endet das Signal in einem dynamischen Teil. Ein Beispiel für ein solches Signal $\chi(\boldsymbol{\xi}, t)$, $\boldsymbol{\xi} \in \Xi_4$ ist in Abbildung 3.2 dargestellt. Dabei wurden die Komponenten des Parametervektors $\boldsymbol{\xi}$ in der Grafik gekennzeichnet.

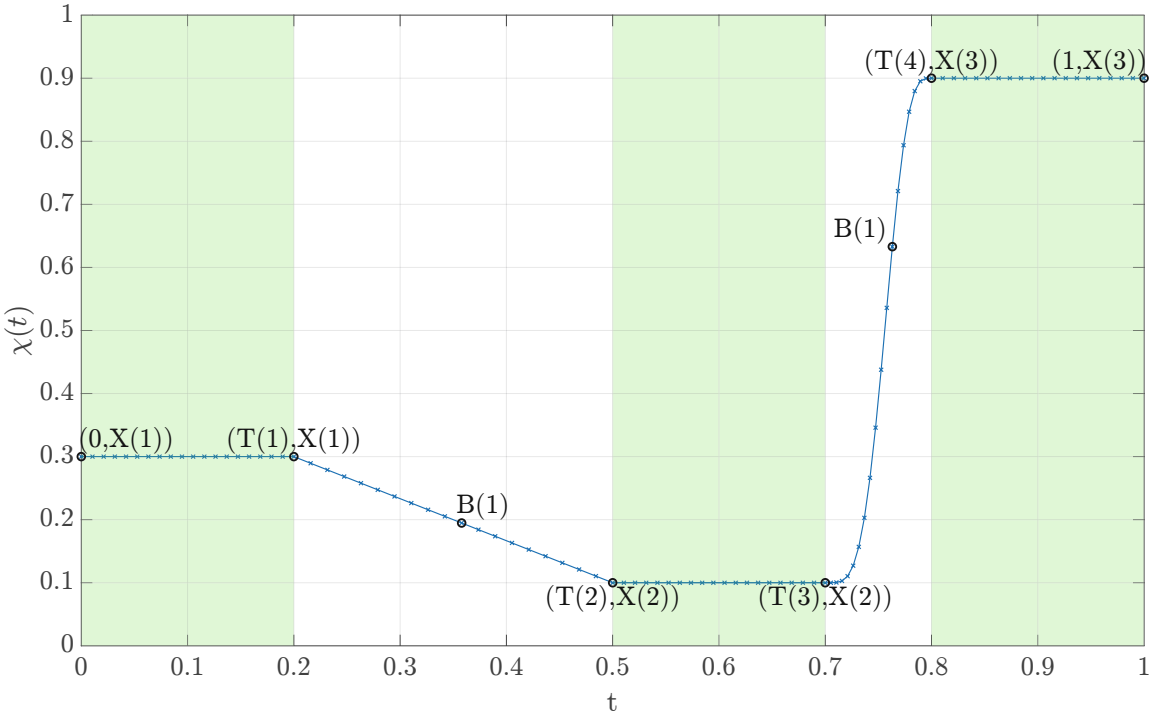


Abbildung 3.2.: Beispiel eines Signals mit $\boldsymbol{\xi} = [[0.2, 0.3, 0.2, 0.1]^{\mathrm{T}}, [1, 7]^{\mathrm{T}}, [0.3, 0.1, 0.9]^{\mathrm{T}}]^{\mathrm{T}}$.

Über diese Definitionen ist der Signalgenerator G_{cont} nun deterministisch und das Signal $\chi(\boldsymbol{\xi}, t)$ ist vollständig bestimmt über $\boldsymbol{\xi}$. Die stochastischen Eigenschaften des Generators [23], sind auch mit unserem Generator G_{cont} erhalten geblieben. Dafür muss lediglich ein zufälliger Parametervektor $\boldsymbol{\xi}$ aus einer Parametermenge Ξ_i mit zufälligem Index i gewählen werden.

Um mit dem optimierungsbasierten Signalgenerator aussagekräftige Worst-Case Signale zu erstellen, muss die maximale Änderungsrate der generierten Signale sorgfältig gewählt werden. Da der Regelkreis typischerweise nur einer begrenzten Änderungsrate folgen kann und oft Zielvorgaben zur maximalen Änderungsrate existieren, soll das Sollsignal an diese Grenzen angepasst werden. Durch die Einhaltung der maximalen Änderungsrate wird sichergestellt, dass die erzeugten Sollsignale die Grenzen des Regelkreises auf praxisgerechte Weise testen und so potenzielle Schwächen aufdecken.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Wir wissen, dass

$$c_i(\boldsymbol{\xi}, t) = X(j) + (X(j+1) - X(j))b_{B(j)}\left(\frac{t - T_\Sigma(i)}{T(i)}\right). \tag{3.2}$$

einen dynamischen Teil beschreibt. Wir können uns hier auf einen beliebigen dynamischen Teil einschränken, da die stationären Teile eine Änderungsrate von 0 haben. Die maximale Änderungsrate wird über das Maximieren der Zeitableitung, bezüglich $\boldsymbol{\xi}$ und t, von $c_i(\boldsymbol{\xi}, t)$ gebildet,

$$\dot{c}_i(\boldsymbol{\xi}, t) = (X(j+1) - X(j))\left(b'_{B(j)}\left(\frac{t - T_\Sigma(i)}{T(i)}\right)\right)\frac{1}{T(i)}. \tag{3.3}$$

O.B.d.A nehmen wir an, dass der beliebige dynamische Bereich zwischen $[T_\Sigma(i-1), T_\Sigma(i)] = [a, b]$ liegt. Somit folgt aus (3.3)

$$\max_{\substack{t\in[a,b]\\ \boldsymbol{\xi}\in\Xi_n}} \dot{c}_i(\boldsymbol{\xi}, t) = \max_{\substack{t\in[a,b]\\ \boldsymbol{\xi}\in\Xi_n}} (X(j+1) - X(j))\left(b'_{B(j)}\left(\frac{t - T_\Sigma(i)}{T(i)}\right)\right)\frac{1}{T(i)} \tag{3.4}$$

$$= \max_{\substack{t\in[a,b]\\ \boldsymbol{\xi}\in\Xi_n}} 1\left(b'_{B(j)}\left(\frac{t - T_\Sigma(i)}{T(i)}\right)\right)\frac{1}{T(i)} \tag{3.5}$$

$$= \max_{\substack{t\in[0,1]\\ \boldsymbol{\xi}\in\Xi_n}} b'_{B(j)}(t)\frac{1}{T(i)}. \tag{3.6}$$

Damit müssen wir die größte Änderungsrate der Bausteinfunktionen finden. Die maximalen Änderungsraten der Bausteinfunktionen für $\tau \in [0, 1]$ ist in Tabelle 3.2 gegeben. Es ist klar

Ableitung von $b(\tau)$	**Maximale Änderungsrate**
1	1
$1/2\pi\cos(\pi\tau/2)$	$\pi/2$
$-e^\tau/(1-e^1)$	$(e^1)/(e^1-1)$
2τ	2
$-12\tau^3 + 12\tau^2$	$16/9 \approx 1.78$
$60\tau^5 - 120\tau^4 + 60\tau^3$	$1296/625 \approx 2.07$
$-280\tau^7 + 840\tau^6 - 840\tau^5 + 280\tau^4$	$276480/117649 \approx 2.350$

Tabelle 3.2.: Auflistung der maximalen Änderungsrate der Bausteine zur Konstruktion eines dynamischen Bereichs.

erkennbar, dass $b_7(t)$ die maximale Änderungsrate aller Bausteinfunktionen aufweist. Damit folgt für die Maximale Änderungsrate

$$\delta_{max} = \max_{\substack{t\in[a,b]\\ \boldsymbol{\xi}\in\Xi_n}} \dot{c}(\boldsymbol{\xi}, t) = \max_{\substack{t\in[0,1]\\ \boldsymbol{\xi}\in\Xi_n}} b'_{B(j)}(t)\frac{1}{T(i)} = \frac{1}{\Delta_{min}}\frac{276480}{117649}. \tag{3.7}$$

Dabei bezeichnen wir $\min_{T(i)\in[0,1]}$ als Δ_{min}. Dadruch ist die maximale Änderungsrate von Δ_{min} abhänig. Damit können wir über die Wahl eines fixen Minimums Δ_{min} für den Umschaltstellenvektor $\boldsymbol{T}$ die maximale Änderungsrate bestimmen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Damit haben wir nun einen deterministischen Signalgenerator G_{cont}, welcher zu einem Parametervektor $\boldsymbol{\xi}$ ein Signal $\chi(\boldsymbol{\xi}, t)$ erzeugt. Da sowohl Simulationsmodelle geschlossener Regelkreise, als auch deren Messungen typischerweise mit einer festen Abtastzeit T_s arbeiten, diskretisieren wir $\chi(\boldsymbol{\xi}, t)$ gemäß T_s und erhalten damit direkt eine Abfolge von Datenpunkte für das Sollsignal. Diese Datenpunkte fassen wir in dem Vektor

$$\boldsymbol{y}^d = \begin{bmatrix} \chi(\boldsymbol{\xi}, 0) \\ \chi(\boldsymbol{\xi}, T_s) \\ \vdots \\ \chi(\boldsymbol{\xi}, 1) \end{bmatrix} \tag{3.8}$$

zusammenfassen. Wir identifizieren das in der Definition 1 definierte Signal $\chi(\boldsymbol{\xi}, t) = G_{cont}(\boldsymbol{\xi})$ mit $y^d = \chi(\boldsymbol{\xi}, t)$ und mit der oben beschriebenen Diskretisierung schreiben wir $\boldsymbol{y}^d = G_{disc}(\boldsymbol{\xi})$. Da wir stets mit diskreten Signalen weiterarbeiten werden, schreiben wir für G_{disc} nur G, also $\boldsymbol{y}^d = G(\boldsymbol{\xi})$. Wir haben somit einen Generator G eingeführt, den wir in weiterer Folge verwenden können, um optimierungsbasiert Wort-Case Testsignale zu finden.

3.2. Formulierung des Optimierungsproblems

In diesem Abschnitt formulieren wir ein Optimierungsproblem, mit dem Ziel jene Signale zu finden, denen der geschlossene Regelkreis nur unzureichend folgen kann. Solche Signale bezeichnen wir auch als Worst-Case Testsignale. Damit wir das Folgeverhalten bewerten können, ist ein geeignetes Fehlermaß zu wählen. Dieses wird in weiterer Folge als Kostenfunktion in der Optimierungsaufgabe verwendet. Da das Sollsignal mit dem in Abschnitt 3.1 eingeführten Signalgenerator erstellt werden soll, muss ein beschränktes Optimierungsproblem gelöst werden. Dabei wird ein Regelkreis Σ betrachtet, der versucht dem Sollsignal $\boldsymbol{y}^d$ möglichst genau zu folgen. Die konkrete Optimierungsaufgabe besteht darin, den Parameter $\boldsymbol{\xi}$ derart zu wählen, dass der Abstand zwischen Sollsignal $\boldsymbol{y}^d = G(\boldsymbol{\xi})$ und Systemausgang $\boldsymbol{y}_{sys}$ maximiert wird.

Der Abschnitt beginnt mit der detallierten Formulierung eines allgemeinen Optimierungsproblems. Anschließend wird die zu maximierende Kostenfunktion, in Abhängigkeit vom Fehlermaß und der zu optimierenden Parametermenge, aufgestellt. Danach formulieren wir das Optimierungsproblem einschließlich aller Nebenbedingungen. Es folgt eine Klassifikation des Problems auf Grundlage der Literatur, sowie eine Diskussion der Anforderungen für globale Optimierungsverfahren. Aufgrund dieser Klassifikation endet dieser Abschnitt mit der Definition von gezielten Experimenten in Form von Simulationsstudien, welche die Grundlage für eine detaillierte Untersuchung und Bewertung der in Abschnitt 2.2 gesammelten Optimierungsverfahren bilden.

Allgemein ist ein skalares Optimierungsproblem durch gegeben,

$$\max_{x \in M} \quad f(x), \tag{3.9a}$$

$$g_i(x) \leq 0, \quad i = 1, \ldots, m, \tag{3.9b}$$

$$h_j(x) = 0, \quad j = 1, \ldots, p, . \tag{3.9c}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Dabei bezeichnet $f : M \mapsto \mathbb{R}$ die Kostenfunktion, M die zulässige Menge, g_i (3.9b) die Ungleichungsnebenbedingungen und h_j (3.9c) die Gleichungsnebenbedingungen. Dieses Problem lässt sich einfach in die in der Literatur typischerweise anzutreffenden Minimierungsprobleme überführen, indem $-f(x)$ minimiert wird. Auf dieser Basis wird im Folgenden das anwendungsspezifische Optimierungsproblem formuliert, wobei wir mit der Formulierung der Kostenfunktion beginnen.

Das allgemeine Ziel eines Regelkreises Σ ist, mit dem Ausgang $\boldsymbol{y}_{sys}$ der Sollgröße $\boldsymbol{y}^d$ hinreichend genau zu folgen. Daraus folgt das Ziel der Kostenfunktion, den Fehler zwischen der Sollgröße $\boldsymbol{y}^d$ und dem Systemausgang $\boldsymbol{y}_{sys}$ zu bestimmen. Diese Abweichung wird durch ein Fehlermaß $d(\cdot,\cdot)$ zwischen den beiden Signalen gemessen. Für das Fehlermaß $d(\cdot,\cdot)$ verwenden wir sowohl den *Mean Absolute Error* (MAE) als auch den *Root Mean Square Error* (RMSE). Diese sind definiert durch

$$d_{mae}(\boldsymbol{y}^d, \boldsymbol{y}_{sys}) = \frac{1}{N}\sum_{i=1}^{N} |y^d(i) - y_{sys}(i)| = \frac{1}{N}\|\boldsymbol{y}^d - \boldsymbol{y}_{sys}\|_1 \tag{3.10}$$

$$d_{rmse}(\boldsymbol{y}^d, \boldsymbol{y}_{sys}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y^d(i) - y_{sys}(i))^2} = \frac{1}{\sqrt{N}}\|\boldsymbol{y}^d - \boldsymbol{y}_{sys}\|_2. \tag{3.11}$$

Anhand von (3.10) und (3.11) erkennt man, dass beide Kostenfunktionen bis auf einen verbleibenden Skalierungsfaktor in die 1-Norm und die 2-Norm übergeführt werden können. Da die Lage des Optimums von dieser Skalierung unbeeinflusst ist, nutzen wird dies in weiterer Folge für eine kompakte Formulierung des Optimierungsproblems. Diese Fehlermaße sind damit Metriken, da jede Norm eine Metrik induziert und wir bezeichnen diese ab jetzt mit d. Der Grund für die Wahl von MAE und RMSE liegt in ihrer unterschiedlichen Gewichtung von Abweichungen. MAE verwendet lineare Gewichtungen und betont daher vor allem anhaltende Abweichungen. RMSE hingegen skaliert Abweichungen quadratisch, wodurch extreme Werte stärker berücksichtigt werden.

Das Sollsignal $\boldsymbol{y}^d$ wird vom Generator G mit dem Parametervektor $\boldsymbol{\xi} \in \Xi_i$, siehe Definition 1, erzeugt und es gilt $\boldsymbol{y}^d = G(\boldsymbol{\xi})$. Gemeinsam mit dem Fehlermaß (3.10) und (3.11) gilt somit für den Definitions- und Wertebereich unserer Kostenfunktion $f : \Xi_i \mapsto \mathbb{R}^+$. Die Parametermenge Ξ_i, siehe Definition 3.1.2, hat verschiedene Größen, je nachdem wie viele Umschaltstellen das Signal aufweisen soll. Da wir nicht in beliebiger Parametermenge optimieren wollen, werden vier Varianten von Ξ_i festlegen.

- Ξ_3: 2 Umschaltstellen, dynamisches Ende
- Ξ_4: 2 Umschaltstellen, stationäres Ende
- Ξ_5: 3 Umschaltstellen, dynamisches Ende
- Ξ_6: 3 Umschaltstellen, stationäres Ende

Wir haben uns für diese 4 verschiedenen Ξ_i entschieden, da diese eine repräsentative Teilmenge möglicher Aneinanderreihungen dynamischer und stationärer Teilbereiche darstellen. Die möglichen Ξ_i kleiner als Ξ_3 sind aufgrund von nur einer Umschaltstelle von geringer

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Bedeutung. Eine Erweiterung über Ξ_6 hinaus wurde aufgrund des steigenden Rechenaufwands nicht weiterverfolgt.

Das Ausgangssignal $y_{sys}(t)$ wird durch den geschlossenen Regelkreis Σ erzeugt. Da das Sollsignal y^d gemäß (3.8) durch Diskretisierung mit der Abtastzeit T_s in $\boldsymbol{y}^d$ überführt wurde, soll nun auch das Ausgangssignal $y_{sys}(t)$ diskretisiert werden. Hierfür verwenden wir dieselbe feste Abtastzeit T_s, und erhalten die Folge von diskreten Datenpunkten des Ausgangssignals

$$\boldsymbol{y}_{sys} = \begin{bmatrix} y_{sys}(0) \\ y_{sys}(T_s) \\ \vdots \\ y_{sys}(1) \end{bmatrix}.$$

Zukünftig wird hierfür die verkürzte Schreibweise

$$\boldsymbol{y}_{sys} = \Sigma(\boldsymbol{y}^d) \tag{3.12}$$

verwendet, wobei diese Notation die oben beschriebene Diskretisierung von $y_{sys}(t)$ mittels T_s impliziert. Die Auswahl repräsentativer Regelkreise Σ wird in Kapitel 4 diskutiert.

Die Kostenfunktion f hängt daher unmittelbar vom gewählten Regelkreis Σ, von der Menge Ξ mit $\boldsymbol{\xi} \in \Xi$ und vom Fehlermaß d ab. Die Kostenfunktion ist schematisch in Abbildung 3.3 dargestellt und lässt sich formulieren als

$$\begin{aligned} f &= \|\boldsymbol{y}^d - \boldsymbol{y}_{sys}\|_p, \quad p \in \{1, 2\}, \\ \boldsymbol{y}^d &= G(\boldsymbol{\xi}), \\ \boldsymbol{y}_{sys} &= \Sigma(\boldsymbol{y}^d). \end{aligned} \tag{3.13}$$

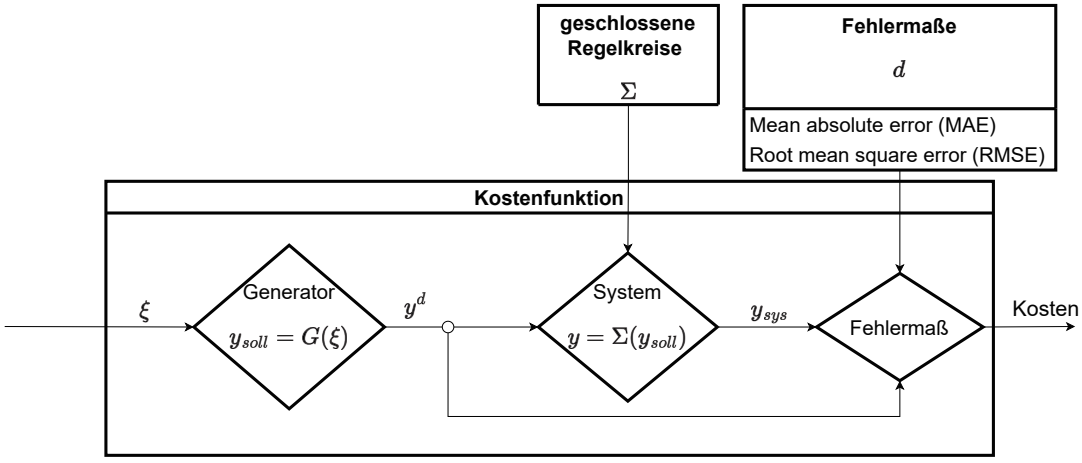


Abbildung 3.3.: Darstellung der Kostenfunktion, mit dem gewählte Fehlermaß und Σ.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Auf Grundlage der formulierten Kostenfunktion ist es nun möglich, das Optimierungsproblem zu definieren. Wir optimieren über die zulässigen Mengen Ξ_i, die in Abschnitt 3.1 definiert wurden. Das Optimierungsproblem wird für beliebige Ξ_i und Σ formuliert, indem die Kostenfunktion $f(\boldsymbol{\xi}, \Sigma, \|\cdot\|_p)$ bezüglich $\boldsymbol{\xi} \in \Xi_i$ maximiert wird. Das Problem ist formuliert als

$$\max_{\boldsymbol{\xi}\in\Xi_i} f(\boldsymbol{\xi}, \Sigma, \|\cdot\|_p) = \|\boldsymbol{y}^d - \boldsymbol{y}_{sys}\|_p, \quad p \in \{1, 2\}\,, \tag{3.14a}$$

$$\boldsymbol{y}_{sys} = \Sigma(\boldsymbol{y}^d), \tag{3.14b}$$

$$\boldsymbol{y}^d = G(\boldsymbol{\xi}), \quad \text{für } \boldsymbol{\xi} = [\boldsymbol{T}, \boldsymbol{B}, \boldsymbol{X}]^{\mathrm{T}} \in \Xi_i, \tag{3.14c}$$

$$\Delta_{min} \leq 1 - \sum_{i=0}^{T_N} T(i), \tag{3.14d}$$

$$\Delta_{min} \leq T(i) \quad \forall i \in \{1, \ldots, T_N\}. \tag{3.14e}$$

T_N beschreibt die Länge von $\boldsymbol{T}$. Die maximale Änderungsrate des Sollsignals δ_{max} wird durch den geschlossenen Regelkreis Σ vorgegeben und beeinflusst die Wahl von Δ_{min}. Wie in (3.7) beschrieben, wird durch die Auswahl von Δ_{min} sichergestellt, dass das Signal die Begrenzung von δ_{max} nicht überschreitet, siehe Gleichung (3.7).

Die Ungleichungsnebenbedingung (3.14d) stellt sicher, dass die Länge des letzten Segmentes des Signals mindestens Δ_{min} beträgt. Somit ist gewährleistet, dass der letzte Teil des Signals die maximale Änderungsrate von δ_{max} nicht überschreitet, siehe Abschnitt 3.1. Die Ungleichungsnebenbedingung (3.14e) begrenzt die Änderungsrate des Signals, indem sie eine Mindestlänge Δ_{min} für alle verbleibenden Abschnitte vorschreibt. Indirekt sind auch mehrere Nebenbedingungen über die zulässige Menge Ξ_i gegeben. Diese betreffen, welche Werte für $\boldsymbol{X}$ und $\boldsymbol{B}$ gewählt werden können.

Nun wollen wir das Problem unter der Zuhilfenahme der reichlich vorhandenen Literatur, wie [1, 2, 3, 4], klassifizieren, um geeignete globale Optimierungs-Algorithmen zu wählen, siehe Abschnitt 2.2.

Von besonderer Bedeutung bei der Einordnung in die Literatur sind die Art der Kostenfunktion und das Vorhandensein von Nebenbedingungen. Die Kostenfunktion ist skalar, nicht differenzierbar und nicht stetig, aufgrund der diskreten Wahl der unterschiedlichen Bausteinfunktionen. Damit ist es nicht möglich, einen Gradienten der Kostenfunktion zu bilden, was die Auswahl an Optimierungsverfahren einschränkt. Die Auswertung der Kostenfunktion setzt dabei die Simulation eines dynamischen Systems oder Messungen an einem realen System voraus. Simulationen oder Testfahrten an realen Systemen sind aber kostspielig und zeitaufwendig. Die Auswertung der Kostenfunktion kann von wenigen Sekunden bis zu mehreren Tagen dauern. Daher gilt die Kostenfunktion unseres Optimierungsproblems als *expensive*, siehe [1]. Zudem sind Nebenbedingungen direkt (3.14e), als auch indirekt über Ξ_i zu berücksichtigen.

Aufgrund dieser Eigenschaften ordnen [1] und [2] das Optimierungsproblem (3.14) als ein *expensive optimization problem* EOP ein, genauer gesagt als ein *single-objective constrained expensive optimization problem*. Die Literatur [1] und [2] empfiehlt für EOPs insbesondere den Einsatz von genetischen Algorithmen, siehe Abschnitt 2.2.1, sowie Surrogate-Algorithmen, siehe Abschnitt 2.2.5. Es wird jedoch betont, dass es keinen allgemein besten

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmus für EOPs gibt, und der passende Algorithmus nur durch Experimente identifiziert werden kann. Dabei sollen folgende Anforderungen betrachtet werden.

Konvergenz: Weil die Auswertung der Kostenfunktion teuer ist, wollen wir die Anzahl an notwendigen Funktionsauswertungen möglichst gering halten wollen. Deswegen setzen wir ein maximales Budget von 100 Auswertungen fest.

Wiederholgenauigkeit: Die Ergebnisse der Optimierung sollten konsistent und reproduzierbar sein, sodass bei einer Wiederholung des Experiments keine signifikanten Abweichungen auftreten. Im Abschnitt über die Optimierungsverfahren wird darauf eingegangen, dass verschiedene zufällige Elemente, wie etwa die Initialisierung der Startwerte, die Wahl von Parametern oder Mutationen bei genetischen Algorithmen (GA), das Ergebnis beeinflussen können, siehe Abschnitt 2.2.1. Es ist jedoch entscheidend, dass diese zufälligen Variationen nicht zu einer zu großen Streuung der Resultate führen. Da die Optimierung nur einmal durchgeführt werden soll, muss sichergestellt werden, dass das Ergebnis zuverlässig und stabil ist.

Absolute Genauigkeit: Das Optimierungsverfahren sollte in der Lage sein, die Kostenfunktion möglichst gut zu maximieren.

Rechenkosten: Die Rechenkosten des Optimierungsverfahrens selbst sind vernachlässigbar, da die Auswertungskosten um ein Vielfaches höher anzunehmen sind.

Nach der Formulierung der Anforderungen an einen geeigneten Optimierungsalgorithmus, widmen wir uns nun der Planung geeigneter Experimente, um die gewünschten Eigenschaften empirisch nachzuweisen bzw. zu widerlegen. Zum Einen wollen wir anhand der folgenden Experimente demonstrieren, dass die optimierungsbasierte Ermittlung von Wort-Case Testsignalen für mehrere komplexe geschlossene Regelkreise funktioniert. Zum Anderen ist es notwendig, anhand nachvollziehbarer Untersuchungen einen geeigneten Optimierungsansatz aus der Vielzahl der in Abschnitt 2.2 diskutierten Möglichkeiten auszuwählen. Bevor wir uns der Beschreibung der eigentlichen Experimente widmen, werden vorab wiederkehrende Begriffe eingeführt.

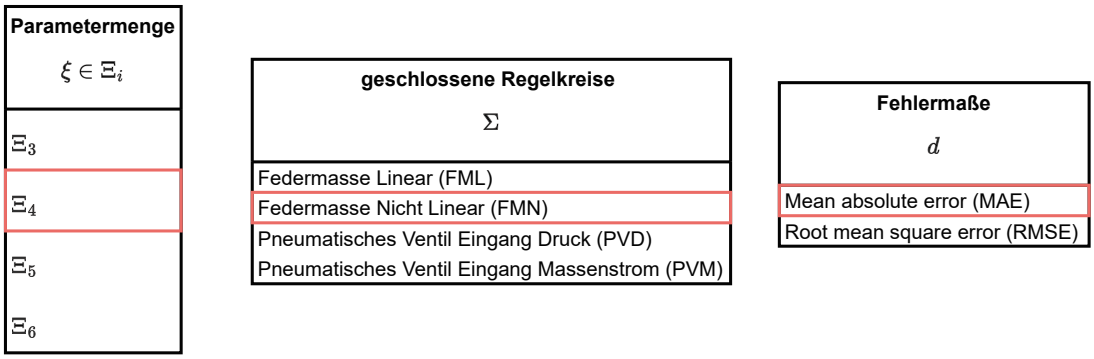


Abbildung 3.4.: Darstellung eines Szenarios s. Dabei sieht man die 4 Parametermengen, 4 Regelkreise und 2 Fehlermaße. Es ergeben sich 32 mögliche Szenarien.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Wir bezeichnen die maximale Anzahl der Auswertung der Kostenfunktion f gemäß (3.13) als N_{iter}. Wird ein Optimierungsproblem wiederholt gelöst, so passiert das N_{rep}-mal. Wird von einem *Szenario* s gesprochen, so ist eine konkrete Realisierung des in (3.14) beschriebenen Optimierungsproblem gemeint. Das bedeutet, dass die Wahl des Fehlermaßes getroffen ist, ein konkreter Regelkreis gewählt ist und, dass die Charakteristik des Sollsignals über die Wahl der Parametermenge gewählt ist. Ein Beispiel für solch ein Szenario wäre das Feder-Masse-System nichtlinear, siehe Abschnitt 4.1, mit Ξ_4 und d_{mae}, siehe Abbildung 3.4. Wir definieren die Menge der in den Experimenten betrachteten Szenarien als $\mathcal{S}$ und bezeichnen die Kardinalität dieser Menge mit $|\mathcal{S}|$. Die Menge der in den folgenden Experimenten untersuchten Optimierer ist durch

$$\mathcal{O} = \{GA,\ Surrogate,\ PS,\ GA\text{-}PS\ ,\ PSO,\ MC,\ DE,\ CMAES\} \tag{3.15}$$

gegeben. Für eine übersichtliche Referenzierung und Zusammenfassung der danach diskutierten Simulationsstudien sind die wesentlichen Parameter und Begriffe in Tabelle 3.3 gelistet.

Begriff	**Beschreibung**
N_{iter}	Anzahl der maximalen Kostenfunktionsaufrufe
N_{rep}	Anzahl der Wiederholungen eines Durchlaufs
$\mathcal{O}$	Menge der betrachteten Optimierer
$\mathcal{S}$	Menge der betrachteten Szenarien
$\lvert\mathcal{S}\rvert$	Anzahl der Szenarien
Szenario $s \in \mathcal{S}$	Wahl eines festen Σ, d und Ξ
Experiment 1	Optimierung eines Szenarios mit N_{iter} Kostenfunktionsauswertungen
Experiment 2	N_{rep}-fache Wiederholung des Experiments 1
Experiment 3	Durchführung von Experiment 2 mit jedem Optimierer
Experiment 4	Analyse eines Σ, durch mehrerer Experimente 3
Experiment 5	Durchführung des Experimentes 3 für alle möglichen Szenarien

Tabelle 3.3.: Übersicht zu den Begriffen der Simulationsstudien.

Experiment 1: Einzelszenario

Es soll in diesem Experiment geprüft werden, ob und wie rasch ein Worst-Case Testsignal gefunden wird. Das Experiment dient zugleich als Machbarkeitsnachweis für die Funktionalität des optimierungsbasierten Signalgenerators.
Dazu wird der zu testende Algorithmus an einem zuvor festgelegten Szenario s erprobt. Ein Szenario besteht aus einem geschlossenen Regelkreis, einem fixen Fehlermaß und einer fixen Ξ. Der gewählte Algorithmus darf maximal N_{iter} Auswertungen der Kostenfunktion durchführen. Die Startwerte $\boldsymbol{\xi}_0$ für die Optimierung, werden entweder anhand einer internen Logik des verwendeten Optimierers oder zufällig gewählt.
Ein Blockdiagramm findet sich in Abbildung 3.5.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

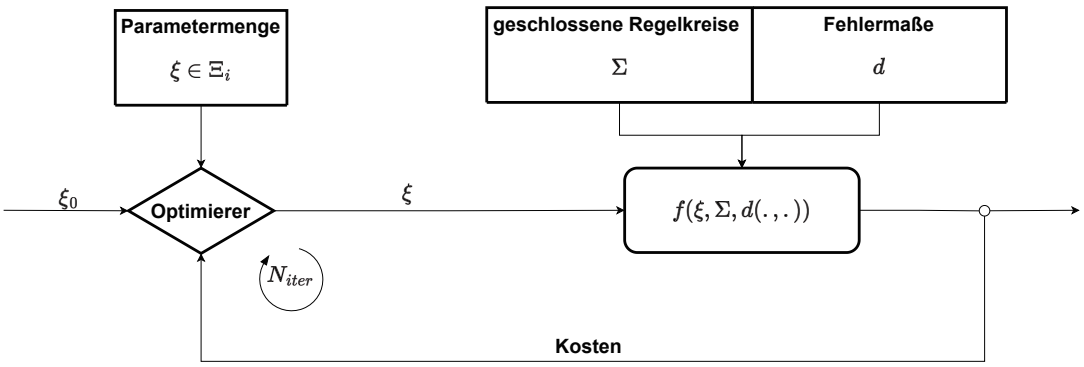


Abbildung 3.5.: Blockdiagramm zu Experiment 1. Dargestellt ist ein Einzelexperiment, wobei die Parametermenge Ξ_i, das System Σ und das Fehlermaß d fest gewählt ist. Die Kostenfunktion $f(\boldsymbol{\xi}, \Sigma, d)$ wird maximal N_{iter}-mal ausgewertet. $\boldsymbol{\xi}_0$ beschreibt die Startwerte für den Optimierer.

Für die spätere systematische Analyse werden die Kostenfunktionswerte während der Optimierung eines Szenarios im Vektor

$$\boldsymbol{j} = [f(\boldsymbol{\xi}_1, \Sigma, \|\cdot\|_p), (f(\boldsymbol{\xi}_2, \Sigma, \|\cdot\|_p), \ldots, f(\boldsymbol{\xi}_{N_{iter}}, \Sigma, \|\cdot\|_p)] \tag{3.16}$$

gesammelt, dabei bezeichnet $\boldsymbol{\xi}_i$ den Parametervektor in der i-ten Iteration.

Experiment 2: Wiederholbarkeit

Das Experiment der Wiederholbarkeit prüft die Stabilität und globale Konvergenz des Optimierers. Die meisten der in Abschnitt 2.2 beschriebenen globalen Optimierungsverfahren, enthalten stochastische Einflussgrößen, wie zum Beispiel den Startwert $\boldsymbol{\xi}_0$, oder die genetischen Operatoren wie Mutation oder Crossover. Ziel dieses Experiments ist es, den Einfluss dieser zufälligen Faktoren zu analysieren.

In Anlehnung an Experiment 1 wird dieses Experiment N_{rep}-mal durchgeführt. Ein globaler Optimierungsalgorithmus erhält erneut ein Budget von N_{iter} Iterationen und löst das Problem nun N_{rep}-fach. Grafisch ist das Experiment in Abbildung 3.6 dargestellt.

Die Ergebnisse zur Wiederholbarkeit lassen sich in Form der $N_{rep} \times N_{iter}$ Matrix

$$\boldsymbol{J}_s^o = \begin{bmatrix} \boldsymbol{j}_1 \\ \boldsymbol{j}_2 \\ \vdots \\ \boldsymbol{j}_{N_{rep}} \end{bmatrix} \tag{3.17}$$

darstellen, wobei $o \in \mathcal{O}$, mit $\mathcal{O}$ gemäß (3.15), den verwendeten Optimierer bezeichnet und s das verwendete Szenario. In dieser Matrix entspricht $\boldsymbol{j}_i$ dem Ergebnisvektor aus Experiment 1 (3.16). Das heißt, jede Zeile beschreibt die Entwicklung der Kostenfunktion eines Einzeltests über N_{iter} Iterationen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

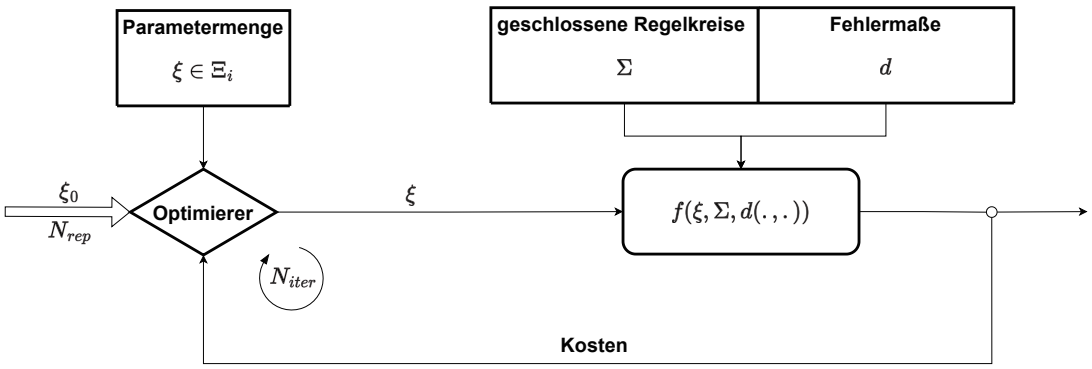


Abbildung 3.6.: Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung. Die Verbreiterung des Signalpfades bei $\boldsymbol{\xi}_0$ verdeutlicht die Verwendung von N_{rep} Startwerten.

Experiment 3: Optimierervergleich an einem Szenario

In diesem Experiment werden verschiedene globale Optimierungsalgorithmen anhand eines Szenarios s verglichen. Ziel ist es, den besten Algorithmus für ein gegebenes Szenario zu identifizieren.

Basierend auf Experiment 2 wird ein Szenario s mit jeweils N_{iter} Funktionsaufrufen und N_{rep} Wiederholungen von jedem Optimierer in $\mathcal{O}$ getestet. Das Experiment liefert auch wertvolle Daten für die Analyse von Σ. Grafisch ist das Experiment 3 im Blockdiagramm in Abbildung 3.7 dargestellt.

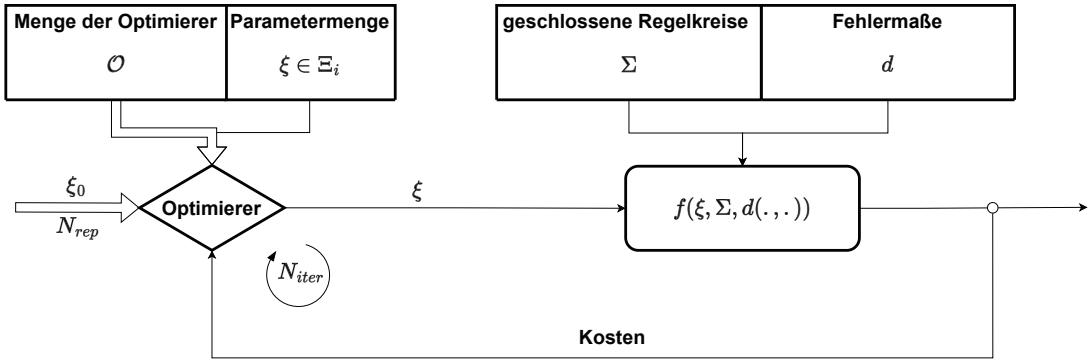


Abbildung 3.7.: Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren.

Die Daten aus Experiment 3 lassen sich als Bild einer Funktion F_s aus der Menge der Optimierer $\mathcal{O}$ betrachten, also $F_s(\mathcal{O})$. Dabei bildet diese Funktion in den Raum der $N_{rep} \times N_{iter}$-Matrizen ab. Die Funktion wird durch

$$F_s(o) = \boldsymbol{J}_s^o \tag{3.18}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

beschrieben, mit dem Szenario s und $\boldsymbol{J}_s^o$ gemäß (3.17). Diese Formulierung wird uns auch in den folgenden Experimenten nützlich sein.

Experiment 4: Optimierervergleich an einem Modell

Dieses Experiment vergleicht für einen gegebenen Regelkreis Σ die verschiedenen globalen Optimierungsverfahren. Ziel dieses Experiments ist es, den Optimierer zu identifizieren, der auf einem gegebenen geschlossenen Regelkreis Σ die besten Ergebnisse liefert.

Für das gewählte Σ testen wir in Experiment 4 jede Kombination von Fehlermaßen und Optimierungsmengen, indem Experiment 3 mit verschiedenen Ξ und d durchlaufen wird. Dies bedeutet, dass für einen festen Regelkreis Σ vier verschiedene Ξ und zwei Fehlermaße getestet werden, sodass Experiment 3 insgesamt achtmal ausgeführt wird. Wir beschreiben die Menge der Szenarien hierfür mit $\mathcal{S}_4$ Die grafische Darstellung dieses Experiments ist im Blockdiagramm Abbildung 3.8 zu finden.

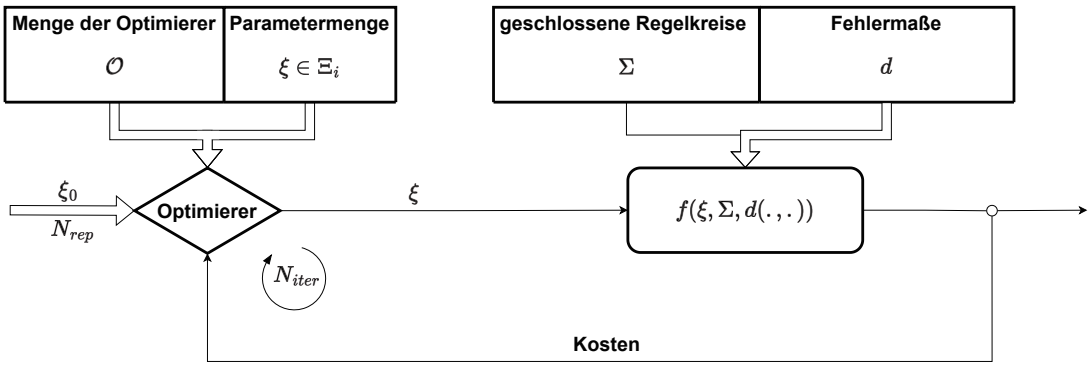


Abbildung 3.8.: Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren, jeder Ξ als auch jedem Fehlermaß d. Dies haben wir durch die Verbreiterung des Signalpfades bei diesen hervorgehoben.

Die Ergebnisse von Experiment 4 werden erneut als Bild einer Funktion betrachtet. Diese Funktion stellt eine Erweiterung der in (3.18) definierten Funktion dar und bildet aus der Menge der betrachteten Szenarien $\mathcal{S}_4$ sowie der Menge der Optimierer $\mathcal{O}$ in den Raum der $N_{rep} \times N_{iter}$-Matrizen ab. Die Funktion ist durch

$$F(s, o) = \boldsymbol{J}_s^o \tag{3.19}$$

beschrieben, mit $\boldsymbol{J}_s^o$ gemäß (3.17).

Experiment 5: Gesamtbewertung

Die abschließende, gesamtheitliche Bewertung der zu testenden Optimierungsmethoden wird mithilfe dieses Experimentes durchgeführt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Im wesentlich ist dies eine Erweiterung von Experiment 3 bzw. 4, wobei nun alle möglichen Szenarien, beschreiben als $\mathcal{S}_5$ untersucht werden. Das bedeutet, dass mit jeder Optimierungsmethode $|\mathcal{S}_5| = 32$ Szenarien optimiert werden. Dieses umfassende Experiment liefert die Grundlage, um am Ende dieser Arbeit einen oder mehrere Algorithmen für die optimierungsbasierte Bestimmung von Worst-Cast Testsignalen zu empfehlen. Aufgrund der hohen Anzahl an Datenpunkten werden die Untersuchungen in Kapitel 5 um ein Bewertungssystem ergänzt. Das Experiment ist in Abbildung 3.9 dargestellt.

Die Ergebnisse des Experiments 5 werden erneut unter Verwendung von (3.19) beschrieben, wobei nun mit $F(\mathcal{S}_5, \mathcal{O})$ alle Szenarien betrachtet werden.

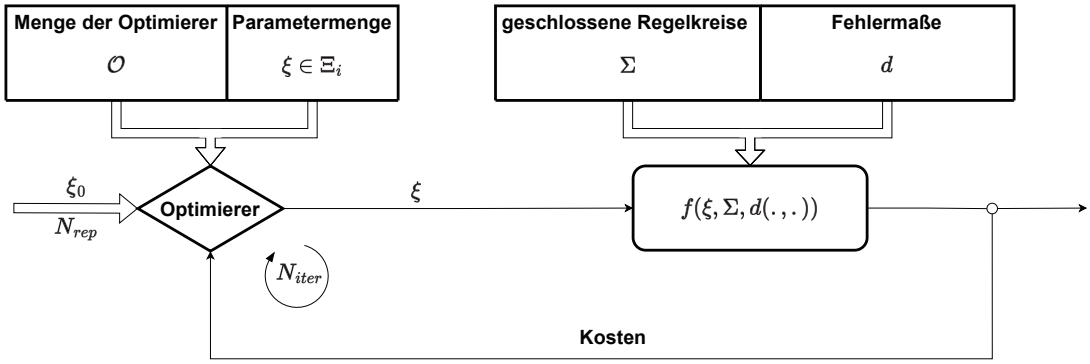


Abbildung 3.9.: Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren, jeder Ξ, jedes betrachteten Systems Σ und jedes Fehlermaßes d. Dies haben wir durch die Verbreiterung des Signalpfades bei diesen hervorgehoben.

3.3. Wahl der Startwerte durch Clustering

Dieser Abschnitt beschreibt eine mögliche Vorauswahl geeignet Startwerte um die optimierungsbasierte Suche nach Worst-Case Testsignalen zu verbessern. Die Simulationsstudie, welche im praktischen Teil in Abschnitt 5.2 ausführlich diskutiert wird, hat gezeigt, dass typische Schwachstellen der betrachteten Regelkreise in verschiedenen Szenarien ähnliche Muster aufweisen. Beispielsweise führen schnelle Änderungen in den dynamischen Bereichen, oder auch nicht differenzierbare Arbeitspunktwechsel, häufig zu hohen Kostenfunktionswerten. Letztere resultieren aus manchen Bausteinfunktionen, bei denen die Übergänge zu den stationären Abschnitten nicht differenzierbar sind. Basierend auf diesen Erkenntnissen sollen die Startwerte für weitere Analysen systematisch gewählt werden. Hierzu wird ein Clustering der Ergebnisdaten der Experimente durchgeführt. Die Verifizierung dieses Ansatzes erfolgt auf ergänzenden Experimenten, deren Ergebnisse im Abschnitt 5.3 beschrieben werden.

Zunächst wird der Prozess des *Clusterings* kurz beschrieben. Clustering ist eine grundlegende Methode der Datenanalyse, die darauf abzielt, Datensätze in Gruppen, sogenannte Cluster, zu unterteilen. Dabei werden Datenpunkte so gruppiert, dass innerhalb eines Clusters eine hohe Ähnlichkeit besteht, während die Ähnlichkeit zu anderen Clustern möglichst

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

gering gehalten wird. Diese Technik wird häufig verwendet, um in komplexen Datenstrukturen Muster zu erkennen und Zusammenhänge zu identifizieren [24].

Im Rahmen unserer Arbeit verwenden wir Clustering, um geeignete Startwerte für die Optimierung zu ermitteln. Hierzu wurden die Ergebnisse einer umfassenden Bewertung in mehreren Schritten verarbeitet. Zunächst wurden die Daten basierend auf den zugrunde liegenden Parametermengen, den Fehlermaßen und den verwendeten Modellen aufgeteilt. Da die Startwerte unabhängig vom gewählten Optimierungsverfahren sein sollen, werden alle Optimierungsverfahren gemeinsam betrachtet und in der Datenmatrix

$$D_i^{\Sigma} = \begin{bmatrix} f(\boldsymbol{\xi}_1, \Sigma, d_{mae}) & \boldsymbol{\xi}_1 \\ f(\boldsymbol{\xi}_2, \Sigma, d_{mae}) & \boldsymbol{\xi}_2 \\ \vdots & \vdots \\ f(\boldsymbol{\xi}_n, \Sigma, d_{mae}) & \boldsymbol{\xi}_n, \end{bmatrix} \in \mathbb{R}^{m \times n} \tag{3.20}$$

zusammengefasst. Der Index i entspricht hierbei der jeweiligen Parametermenge Ξ_i gemäß 3.1.2, Σ steht für einen Regelkreis, m entspricht der Länge von $\boldsymbol{\xi} + 1$ und n beschreibt die Anzahl an verschiedenen Kostenfunktionswerten und Parametern in einer Matrix D_i^{Σ}. Da das Clustering mit einer festen Datenmenge durchgeführt wird, wird im Weiteren lediglich D zur Bezeichnung der Daten verwendet. Auf die Matrix D wird hierarchisches Clustering mit einer euklidischen Distanzmetrik angewandt, mit einer Zerlegung in 150 Cluster. Nach Abschluss der Clustering-Analyse wurden D in die Menge $C = \{c_1, c_2, \ldots, c_{150}\}$ zerlegt, wobei die Elemente von C Matrizen der Form (3.20) sind. Grafisch wurden die Ergebnisse dieses Clustering in Abbildung 3.10 dargestellt.

Aus den erstellten Clustern, die sowohl die Kostenfunktionswerte als auch die zugehörigen Parameter gemeinsam gruppieren, sollen geeignete Startwerte extrahiert werden. Würden Startwerte ausschließlich aus dem besten Cluster entnommen werden, wären die Parameter und das aus diesen daraus gebildeten Signale sehr ähnlich. Daher werden aus den drei Clustern mit den höchsten durchschnittlichen Kostenfunktionswerten die besten Parameterwerte als Startwerte ausgewählt. Zur Bestimmung der besten Cluster wird die Funktion $\bar{f}(c)$ verwendet, die den durchschnittlichen Kostenfunktionswert eines Clusters $c \in C$ berechnet. Wir wählen nun die jeweils 3 Cluster c_0^1, c_0^2, c_0^3 aus, welche den höchsten durchschnittlichen Kostenfunktionswert haben, also

$$\begin{aligned} c_0^1 &= \arg\max_{c \in C} \bar{f}(c), \\ c_0^2 &= \arg\max_{c \in C \setminus c_0^1} \bar{f}(c) \quad \text{und} \\ c_0^3 &= \arg\max_{c \in C \setminus \{c_0^1, c_0^2\}} \bar{f}(c). \end{aligned}$$

Aus diesen drei Clustern mit den höchsten Durchschnittswerten wurde jeweils der Parametervektor $\boldsymbol{\xi}_0$ ausgewählt, der innerhalb des Clusters c_0^i den höchsten Kostenfunktionswert liefert. Wir bezeichnen die Menge der Startwerte als $S_0(D)$, da sie auf den Daten D basiert. Die Startwerte eines spezifischen Σ und einer Parametermenge Ξ_i können dementsprechend als $S_0\left(D_i^{\Sigma}\right)$ beschrieben werden. Wird das Clustering für alle Σ und jede Parametermenge

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

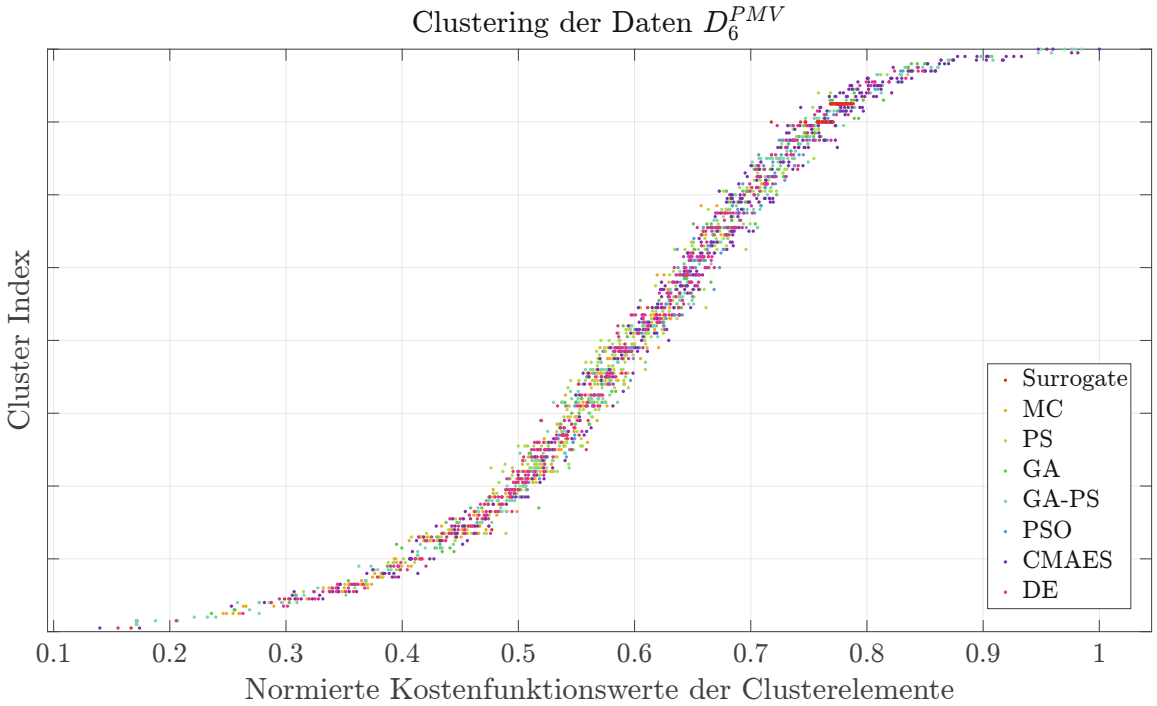


Abbildung 3.10.: Darstellung der Cluster von D_6^{PMV}, nach den durchschnittlichen Kostenfunktionswerten sortierten. Die Farbmarkierungen zeigen die Herkunft der Daten in Bezug auf die jeweiligen Optimierer.

Ξ_i durchgeführt, ergibt sich die Menge

$$\Xi_{0,i} = \bigcup_{\Sigma} S_0\left(D_i^{\Sigma}\right), \quad \Sigma \in S, \tag{3.21}$$

wobei in S alle betrachteten Σ beinhaltet sind. Um die Effektivität und Übertragbarkeit der gewählten Startwerte zu demonstrieren und deren Einfluss auf die Leistung der Optimierungsalgorithmen zu analysieren, wird im Folgenden ein weiteres Experiment beschrieben.

Experiment 6: Startwerte

Das Experiment untersucht den Einfluss der gewählten Startwerte auf die Ergebnisse der Optimierung. Hierfür wird ein *leave one out cross validation*-Ansatz [25] verwendet, bei dem Startwerte genutzt werden, die aus den Clustering-Ergebnissen anderer Modelle stammen. Zusätzlich wird jeder Optimierungsalgorithmus auch ohne Startwerte ausgeführt, um einen klaren Vergleich zu ermöglichen. Die Menge der betrachteten Optimierer wird in diesem Kontext als $\mathcal{O}_6$ bezeichnet, wobei

$$\mathcal{O}_6 = \mathcal{O} \cup \{GA_0, GA\text{-}PS_0, PSO_0, Surrogate_0, MC_0, CMAES_0, PS_0\} \tag{3.22}$$

mit $\mathcal{O}$ gemäß (3.15). Dabei werden nicht alle möglichen Szenarien betrachtet, sondern ausschließlich jene Szenarien, die das Fehlermaß MAE verwenden. Diese Menge wird als

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\mathcal{S}_6$ bezeichnet. Für jedes Szenario $s \in \mathcal{S}_6$ wird eine spezifische Menge von Startwerten ausgewählt, welche durch

$$\Xi_{0,i}^{s} = \bigcup_{\Sigma} S_0\left(D_i^{\Sigma}\right) \tag{3.23}$$

für alle Betrachteten Σ, die nicht in s enthalten sind, beschrieben wird. Das heißt, wir bilden die Startwerte für ein Modell Σ und einer Parametermenge i mit dem Clustering der anderen Modelle mit derselben Parametermenge. Grafisch ist die Wahl der Startwerte in Abbildung 3.11 dargestellt.

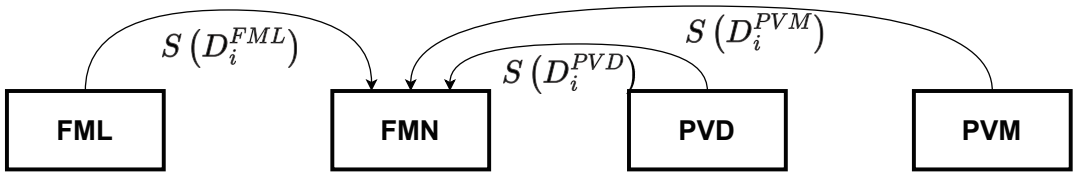


Abbildung 3.11.: Darstellung vom Ursprung der Daten für $\Xi_{0,i}^{s}$ für ein s mit dem Modell FMN im Experiment 6.

Ziel des Experiments ist es, den Einfluss der Startwertwahl auf die Optimierungsergebnisse zu analysieren. Insbesondere wird untersucht, ob durch diese gezielte Wahl der Startwerte signifikant höhere Werte der Kostenfunktion erreicht werden können. Zusätzlich wird evaluiert, welcher Algorithmus am stärksten von den gewählten Startwerten profitiert. Die Ergebnisse dieses Experiments sind vergleichbar mit denen aus den Experimenten 4 und 5. Sie können als Bild der Funktion F gemäß Gleichung (3.19) interpretiert werden. Damit werden die Ergebnisse beschrieben als $F(\mathcal{O}_6, \mathcal{S}_6)$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4. Fallstudien geschlossener Regelkreise

In diesem Kapitel wird ein Überblick über die in dieser Arbeit betrachteten Fallbeispiele von Regelkreise gegeben. Dabei stellen wir zwei dynamische Systeme vor, das Feder-Masse System und das pneumatische Ventil. Beide Systeme dienen als Grundlage für die Entwicklung mehrerer Regelkreise.

Im ersten Abschnitt liegt der Schwerpunkt auf der Positionierung eines Feder-Masse Systems mit der Kraft als Eingang. Das wesentliche Charakteristikum ist die lineare bzw. nichtlineare Formulierung des Reibmodells. Im zweiten Abschnitt wird das pneumatische Ventil behandelt, das anhand zweier Simulationsmodelle und eines Prüfstandaufbaus untersucht wird. In der Simulation wird entweder der Druck oder der Massenstrom als Eingangsgröße für das Ventil gewählt. Der Prüfstand dient als abschließendes Validierungsszenario und wird nicht für den Optimierervergleich verwendet. Für sämtliche Varianten werden sowohl die Funktionsweise des Grundsystems als auch die spezifischen Regelkreise beschrieben, die durch unterschiedliche Modellannahmen und Regelungskonzepte entstehen.

4.1. Feder-Masse System

In diesem Abschnitt wird ein positionsgeregeltes Feder-Masse System vorgestellt. Dabei entstehen zwei Varianten der Positionieraufgabe welche sich anhand der linearen und der nichtlinearen Modellierung der Reibung unterscheiden. Darüber hinaus wird der zugehörige Regler, ein PI-Regler, erläutert. Die Darstellung beginnt mit der Modellierung der Strecke und schließt mit einer Beschreibung der jeweiligen Regler für den linearen und den nichtlinearen Fall ab.

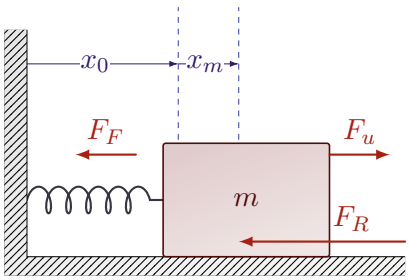


Abbildung 4.1.: Feder-Masse System mit Haftreibung.

Das zu regelnde Feder-Masse System ist in Abbildung 4.1 schematisch dargestellt. Die Masse m ist über eine Feder mit einer starren Wand verbunden. Die Federkraft F_F wird durch das lineare Modell $F_F = cx_m$ beschrieben, wobei c die Federkonstante ist. Eine äußere Eingangskraft F_u wirkt auf die Masse m, um deren Position x_m zu verändern. Diese Eingangskraft entspricht der Stellgröße der Strecke, also $u = F_u$. Zusätzlich wirkt auf die

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

bewegliche Masse m die Reibkraft F_R. In diesem Abschnitt wird zwischen der nichtlinearen Modellierung der Reibkraft mit Haftbedingung und einer linearen Approximation von F_R unterschieden.

Wir können dieses Modell in beiden Fällen als Differentialgleichung darstellen. Die Bewegung der Masse wird durch

$$\frac{\mathrm{d}}{\mathrm{d}t}x_m = v, \tag{4.1a}$$

$$m\frac{\mathrm{d}}{\mathrm{d}t}v = F_u - F_R - cx_m, \tag{4.1b}$$

beschrieben, wobei v die Geschwindigkeit der bewegten Masse beschreibt. Die Messgröße der Strecke wird durch

$$y_{sys}(t) = x_m(t) \tag{4.2}$$

dargestellt. Im linearen Fall ist die Reibkraft F_R als $r_v v$ mit der Geschwindigkeit der Masse v gegeben als

$$F_R = r_v v. \tag{4.3}$$

Somit ist die Strecke im linearen Fall, *Feder-Masse System linear* (FMSL), vollständig beschrieben.

Im nichtlinearen Fall, *Feder-Masse System nichtlinear* (FMSN), wird die Reibkraft F_R gemäß (4.3) um eine Coulombsche Komponente und Haftreibung erweitert. Die Reibkraft lässt sich anhand von

$$F_R = r_v v + r_C \operatorname{sgn}(v) + (r_H - r_C) \exp\left(-\left(\frac{v}{v_0}\right)^2\right) \operatorname{sgn}(v) \tag{4.4}$$

beschreiben. Darin entspricht r_C dem Coulombschen Reibungskoeffizienten, r_h der Haftkraft und v_0 einer Referenzgeschwindigkeit. Außerdem wird eine Systemumschaltung benötigt, welche durch die Haftbedingung gegeben ist. Ist die gesamt wirkende Kraft geringer als die Haftkraft r_H und die Geschwindigkeit $v = 0$, dann bleibt die Masse m in Ruhe. Als Ungleichung ist dies durch

$$|F_u - cx_m| \leq r_H \tag{4.5}$$

beschrieben. Falls nun diese Bedingungen erfüllt sind, gilt für das System

$$\frac{\mathrm{d}}{\mathrm{d}t}x_m = 0, \tag{4.6a}$$

$$m\frac{\mathrm{d}}{\mathrm{d}t}v = 0. \tag{4.6b}$$

Die Haftphase des nichtlinearen Systems, $|F_u - cx_m| \leq r_H$ und $v = 0$, ist somit beschrieben durch (4.6) und während der Gleichphase, $v \neq 0$ oder $|F_u - cx_m| > r_H$, gilt (4.1). Die Parameter des soeben diskutierten mechanischen Systems sind in Tabelle 4.1 zusammengefasst.

Um den geschlossenen Regelkreis aus Abbildung 1.1 vollständig zu charakterisieren, fehlt nun die Darstellung des verwendeten Reglers. Dieser soll die Eingangskraft gemäß einer

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Parameter	Formelzeichen	Wert linear	Wert nichtlinear
Masse	m	1	
Federkonstante	c	2	
Viskose Reibungskomponente	r_v	4	
Bezugsgeschwindigkeit	v_0	/	0.01
Coulombsche Reibungskomponente	r_C	0	1
Haftreibungskomponente	r_H	0	2

Tabelle 4.1.: Parameter des mechanischen Systems.

Positionsvorgabe für die Masse bestimmen. Dazu kommt ein Proportional-Integral Regler (PI-Regler) der Form

$$F_u(t) = u(t) = -k_P e(t) - k_I \int_0^t e(\tau)\,\mathrm{d}\tau \tag{4.7}$$

zum Einsatz. Dabei ist die Eingangskraft F_u die Stellgröße u der Strecke und e der Regelfehler, also $e(t) = y_{sys}(t) - y^d(t)$. Die Verstärkungsfaktoren des P- und I-Anteils sind k_P und k_I. Im FMSL wird als Regler (4.7) mit den Verstärkungsfaktoren $k_P = 20$ und $k_I = 30$ verwendet.

Bei FMSN wird aufgrund der Haftreibung ein leicht veränderter Regler verwendet. Um störende Effekte wie Grenzzyklen zu vermeiden, wird der Integrator des Reglers modifiziert. Das sogenannte *Einfrieren* verhindert die weitere Integration kleiner Fehler innerhalb einer Toleranz d. Dies wird berücksichtigt durch eine Anpassung im Integralterm gemäß

$$F_u(t) = -k_P e(t) - k_I \int_0^t e_{fr}(\tau)\,\mathrm{d}\tau,$$

wobei der manipulierte Fehler e_{fr} für jedes τ durch

$$e_{fr}(\tau) = \begin{cases} 0, & |e(\tau)| \leq d, \\ e(\tau), & |e(\tau)| > d, \end{cases}$$

definiert ist. Für die Verstärkungsfaktoren wurde $k_P = 20$ und $k_I = 30$ gewählt. Für die Toleranz wurde $d = 0.02$ gewählt.

4.2. Pneumatisches Ventil

Wir werden in diesem Abschnitt den Regelkreis des pneumatischen Ventils erläutern. Die Strecke, welche wir dafür betrachten, ist ein Pneumatisches-Ventil vom Typ VEVM [26] der Firma Festo SE & Co. KG. Für eine detaillierte mathematische Beschreibung sei auf [27, 28, 29] verwiesen. In den beiden Simulationsmodellen werden Teilmodelle des Ventils mit geeigneten Reglern abgebildet. Das gesamte Ventil wird anhand des Prüfstandsaufbaus

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

untersucht und wird zur abschließenden Validierung herangezogen. Ausgehend von der Beschreibung des Gesamtventils folgt die mathematische Formulierung der Teilmodelle und die Diskussion der verwendeten Regler.

Das pneumatische Ventil ist in eine Vorstufe und Hauptstufe eingeteilt, siehe Abbildung 4.2. Die Vorstufe besteht aus 8 Ventilen, 4 Belüftungs- und 4 Entlüftungsventilen. Die Hauptstufe besteht aus 4 einfachwirkenden Zylindern mit Rückstellfedern und fungiert damit ebenso als Ventil. Jeweils ein Belüftungs- und Entlüftungsventil der Vorstufe führen Massenstrom der Hauptstufe zu. Diese Massenströme sind also die Stellgrößen für die Zylinder der Hauptstufe.

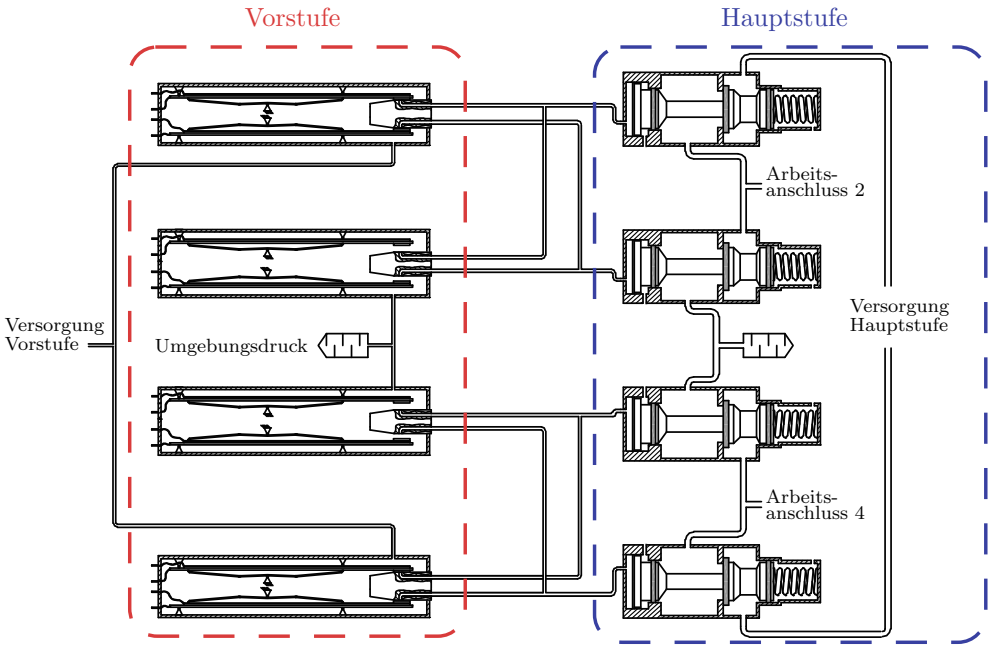


Abbildung 4.2.: Übersicht von Vor- und Hauptstufe des Pneumatikventils, siehe [28, Abbildung 2.1]. Links ist die Vorstufe mit den acht piezoelektrisch aktuierten Balken und rechts die Hauptstufe mit vier Ventilstößeln dargestellt.

Die Ventile der Vorstufe werden mittel piezoelektrischer Balken aktuiert. Der Balken ist mit einer Blattfeder fixiert. Es wird bei den Ventilen der Vorstufe zwischen *Normally Open* (NO) und *Normally Closed* (NC) unterschieden. In der NO-Bauweise ist das Ventil offen und schließt sich, wenn durch das Anlegen einer ausreichend hohen elektrischen Spannung die Vorspannkraft der Feder überwunden wird. Bei NC ist das Ventil offen, bis eine ausreichend hohe Spannung vorhanden ist. Die Belüftungsventile sind alle in NC-Bauweise, die Entlüftungsventile in NO-Bauweise. Damit ist sichergestellt, dass im spannungsfreien Zustand, kein Druck in der Hauptstufe entstehen kann.

Die einfachwirkenden Zylinder der Hauptstufe, auch *Booster* genannt, bilden eine pneumatische Vollbrücke, das heißt, dass an den Arbeitsanschlüssen sowohl belüftet als auch entlüftet werden kann, siehe Abbildung 4.3. Die Funktionsweise aller Booster ist identisch. Durch das Befüllen der Boostervorkammer über die Vorstufe kommt es zu einem Druckanstieg in der Vorkammer. Der Druck p bringt über die Membran des Stößels mit der Fläche

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A eine Kraft auf, die der Federkraft entgegen wirkt. Ausreichend hohe Drücke führen zu einer Bewegung des Stößels und ermöglichen schlussendlich einen Durchfluss der Luft in der Hauptstufe. Das von uns untersuchte positionsgeregelte pneumatische Ventil bezieht

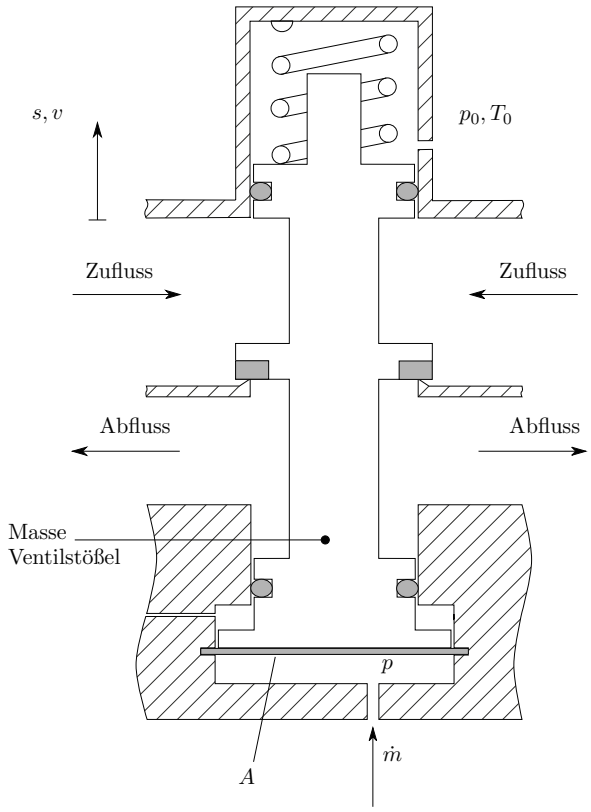


Abbildung 4.3.: Darstellung eines Ventils der Hauptstufe, mit Rückstellfeder. Der Massenstrom $\dot{m}$ in die Vorkammer, sowie der, durch Auslenkung des Ventilstößels, resultierende Luftstrom der Hauptstufe sind durch Pfeile angedeutet. Diese Darstellung ist aus [30] entnommen.

sich auf die Hauptstufe, deren Stößelposition als Regelgröße dient. Diese Hauptstufe werden wir entweder in Kombination mit zwei zugehörigen Vorstufenventilen oder eigenständig betrachten.

Die erste Variante, *Pneumatisches Ventil Eingangsgröße Druck* (PVED), benötigt keine Vorstufe und die Stellgröße ist direkt der Druck in der Boostervorkammer. Dieses System kann mathematisch durch Differentialgleichungen beschrieben werden, wobei die relevanten Größen der Druck p als Eingang sowie die Position s und die Geschwindigkeit v des Ventilstößels sind.
Auf Basis der Impulserhaltung lässt sich die Zustandsraumdarstellung des Systems durch

$$\frac{\mathrm{d}}{\mathrm{d}t}s = v, \tag{4.8a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}v = \frac{1}{m}(A(p - p_0) - c(s - s_0) + f_r(v) + f_k(s, v)), \tag{4.8b}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

mit der Masse m, der Membranfläche A, der entspannten Federlänge s_0 und dem Umgebungsdruck p_0 beschrieben. Die Reibkraft und die Kontaktkraft, die die Bewegung des Ventilstößels beeinflussen, werden durch die nichtlinearen Funktionen $f_r(v)$ und $f_k(s, v)$ abgebildet. Als Messgröße

$$y_{sys}(t) = s(t), \tag{4.9}$$

dient stets die Position des Stößels.

Die zweite Variante, *Pneumatisches Ventil Eingangsgröße Massenstrom* (PVEM), verwendet den Massenstrom $\dot{m}$ als Eingangsgröße, während der Druck als Zustandsgröße durch eine Differentialgleichung beschrieben wird. Das System umfasst insgesamt drei Differentialgleichungen, die die die Zustandsänderung des Systems definieren. Diese Gleichungen beschreiben die zeitliche Entwicklung des Drucks p in der Boostervorkammer, der Position s und der Geschwindigkeit v des Ventilstößels. Die Gleichungen sind durch

$$\frac{\mathrm{d}}{\mathrm{d}t}s = v, \tag{4.10a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}v = \frac{1}{m}(A(p - p_0) - c(s - s_0) + f_r(v) + f_k(s, v)), \tag{4.10b}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}p = \frac{\kappa}{As + V_0}(-Avp + R_s T_0 \dot{m}), \tag{4.10c}$$

mit dem Wärmekapazitätsverhältnis κ, der Gaskonstante R_s und der Temperatur des Gases T_0, gegeben. V_0 beschreibt das Totvolumen der Boostervorkammer und die Reib- und Kontaktkraft sind wieder durch $f_r(v)$ und $f_k(s, v)$ abgebildet.

In der dritten Variante wird das vollständige System eines Boosters auf einem Prüfstand betrachtet. Hierbei wird die Dynamik des Massenstroms durch folgende Gleichung

$$\dot{m} = \dot{m}_{in}(u_{nc}) - \dot{m}_{out}(u_{no}) \tag{4.11}$$

beschrieben, wobei die Änderung des Massenstroms durch die Differenz zwischen dem einströmenden Massenstrom $\dot{m}_{in}(u_{nc})$ und dem ausströmenden Massenstrom $\dot{m}_{out}(u_{no})$ der Vorstufenventile bestimmt wird. Dieses Modell beschreibt den Prüfstandsaufbau und wird nicht für die Entwicklung und Bewertung des optimierungsbasierten Signalgenerators verwendet, sondern dient als abschließendes Validierungsszenario. Auf die genauen Beschreibungen der Formeln für $\dot{m}_{in}(u_{nc})$, $\dot{m}_{out}(u_{no})$, $f_r(v)$ und $f_k(s, v)$ werden wir nicht eingehen und verweisen auf [27, 28, 29]. Alle übrigen Parameter sind in Tabelle 4.2 zusammengefasst.

Nachdem die Strecke der Regelkreise beschrieben wurde, widmen wir uns nun den Reglern. Im Folgenden werden die Regler der ersten beiden Varianten detailliert betrachtet. Das PVEM verwendet einen PI-Regler. Dieser ist durch

$$\dot{m} = u(t) = k_P e(t) + k_I \int_0^t e(\tau)\,\mathrm{d}\tau \tag{4.12}$$

gegeben, wobei $u(t)$ als Massenstrom $\dot{m}$ die Stellgröße der Strecke beschreibt. Der normierte Regelfehler $e(t)$ wird durch

$$e(t) = \frac{y^d(t) - y_{sys}(t)}{s_n} \tag{4.13}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Parameter	Beschreibung
κ	Isentropenexponent (Wärmekapazitätsverhältnis)
A	Fläche über welche der Vorkammerdruck auf den Ventilstößel drückt
R_s	Gaskonstante
T_0	Temperatur des Gases (konstant)
p_0	Umgebungsdruck
c	Federkonstante
s_0	entspannte Position der Feder
V_0	Totvolumen der Vorkammer

Tabelle 4.2.: Beschreibung physikalischer Parameter des pneumatischen Ventils.

mit der Sollposition y^d, der Istposition y_{sys} und dem Normierungsfaktor s_n beschrieben. Als Verstärkungsfaktoren des P- und I-Anteils des Reglers werden $k_P = 3 \cdot 10^{-4}$ und $k_I = 10^{-7}$ festgelegt.

Im PVED kommt ein PID-Regler zum Einsatz. Dieser ist im Allgemeinen durch (2.4) gegeben. Die allgemeine Form des PID-Reglers ist jedoch nicht realisierbar. Damit wird der verwendete Regler durch

$$p(t) = u(t) = k_P e(t) + k_I \int_0^t e(\tau)\ \mathrm{d}\tau + k_D \frac{\mathrm{d}}{\mathrm{d}t} e(t) + R(k_D, e, k_t) \tag{4.14}$$

beschrieben. Dabei stellt $u(t)$ die Stellgröße dar, $e(t)$ den normierten Regelfehler (4.13), k_t die Filterzeitkonstante, und k_P, k_I sowie k_D sind die Verstärkungsfaktoren des P-, I- und D-Anteils. Der Restterm $R(k_D, e, k_t)$ wird durch

$$R(k_D, e, k_t) = k_D \left(e(0) - \dot{e}(0)\right) \exp(-k_t^{-1}) - k_D \int_0^t \exp\left(-k_t^{-1}(t-\tau)\right) \ddot{e}(\tau)\ \mathrm{d}\tau \tag{4.15}$$

dargestellt, und es gilt

$$\lim_{k_t \to 0} R(k_D, e, k_t) = 0.$$

Die Verstärkungsfaktoren sind mit $k_P = 3 \cdot 10^7$, $k_I = 6 \cdot 10^8$, $k_t = 0.01\,\mathrm{s}$ und $k_D = 3 \cdot 10^4$ gegeben.

Der Regler des Prüfstands, der das Gesamtsystem steuert, basiert auf einem speziell entwickelten, nichtlinearen Regelungskonzept. Eine ausführliche Darstellung dieses Reglers wird in der vorliegenden Arbeit nicht behandelt, da dies den Rahmen sprengen würde. Eine detaillierte Beschreibung einer früheren Version dieses Reglers findet sich jedoch in [27]. Für die Validierung des optimierungsbasierten Signalgenerators stellt es keine Einschränkung dar, dass lediglich Informationen über den Ausgang des Regelkreises verfügbar sind. Denn entsprechend der Entwurfskriterien benötigt der Generator ausschließlich das Ausgangssignal y_{sys}, um Worst-Case-Testsignale zu generieren.

Zusammenfassend wurde in diesem Kapitel ein umfassender Überblick über zwei zentrale dynamische Systeme, das Feder-Masse-System und das pneumatische Ventil, gegeben. Für beide Systeme wurden unter verschiedenen Modellannahmen geschlossene Regelkreise entworfen. Diese soeben diskutierten Regelkreise stellen die Grundlage für die durchgeführten Experimente dar, welche wir im nächsten Kapitel betrachten werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

5. Experimentelle Bewertung

Das vorliegende Kapitel beschreibt die Ergebnisse der im Abschnitt 3.2 spezifizierten Experimente und analysiert die Regelkreise auf Basis der identifizierten Worst-Case Testsignale. Hierfür wird anhand der Merkmale aus Abschnitt 3.2 systematisch ein Bewertungssystem entwickelt, um die Leistungsfähigkeit der globalen Optimierungsverfahren aus Abschnitt 2.2 zu vergleichen.

Im ersten Teil dieses Kapitels wird das Bewertungssystem umfassend beschrieben. Es setzt sich aus zwei zentralen Komponenten zusammen. Diese sind die Bewertungskriterien, die als Grundlage für die Analyse der Ergebnisse dienen und der Reihung, die eine geordnete Bewertung der Optimierungsverfahren anhand dieser Kriterien ermöglicht. Zunächst werden die Anforderungen an das Bewertungssystem definiert und anschließend die Kriterien und die Art der Reihung erläutert.

Im zweiten Teil dieses Kapitels werden die Ergebnisse aus allen Szenarien bewertet und analysiert. Anschließend erfolgen in einzelnen Unterabschnitten die Bewertungen der Optimierer anhand der verschiedenen Regelkreise. In diesen Abschnitten werden die Regelkreise zusätzlich auf Grundlage der identifizierten Worst-Case Testsignale detailliert untersucht. Am Ende des zweiten Teils geben wir eine finale Bewertung zu den Optimierern ab.

Der dritte Teil dieses Kapitels befasst sich mit den Ergebnissen des Experiments 6, in dem die Startwerte aus Abschnitt 3.3 auf ihre Eignung untersucht werden. Ziel ist es, die Auswirkungen dieser Startwerte auf die Optimierungsergebnisse zu bewerten, indem diese anhand des entwickelten Bewertungssystems analysiert werden.

Abschließend wird im vierten Teil des Kapitels untersucht, ob die grundlegende Zielsetzung dieser Arbeit erreicht wurde. Hierzu wird eine Optimierung direkt am Prüfstand durchgeführt, die Messergebnisse analysiert und überprüft, ob der optimierungsbasierte Signalgenerator erfolgreich ein Worst-Case Testsignal identifizieren und Schwachstellen in der Regelung aufzeigen konnte.

5.1. Beschreibung des Bewertungssystems

Wie in Abschnitt 3.2 beschrieben, wurde das Bewertungssystem entwickelt, um die große Menge an Daten aus Experiment 5, der Gesamtbewertung, effizient zu verarbeiten. Im Verlauf der Arbeit zeigte sich jedoch, dass das Bewertungssystem auch in den Experimenten 3, 4 und 6 unverzichtbar ist, da es eine strukturierte und vergleichbare Auswertung der umfangreichen Simulationsergebnisse ermöglicht. Unser Bewertungssystem wurde so konzipiert, dass es folgende Bedingungen erfüllt.

1. Das Bewertungssystem in der Lage, mit Daten aus unterschiedlichen Szenarien umzugehen und diese unabhängig zu bewerten.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

2. Die Kriterien basieren auf den Anforderungen an die Optimierer, welche in Abschnitt 3.2 formuliert wurden.

3. Ähnliche Resultate in der Optimierung führen zu ähnlichen Bewertungen.

Die Bewertungskriterien werden über Bewertungsfunktionen, die auf die Ergebnisse eines Szenarios angewandt werden, realisiert. Die Reihung bewertet anschließend diese Funktionen, wobei ein Vergleich der Ergebnisse vorgenommen wird. Da das Bewertungssystem für vergleichende Analysen konzipiert ist, wird ein Referenzalgorithmus benötigt. In dieser Arbeit wird die Monte-Carlo Optimierung (MC) als der Referenzalgorithmus verwendet, weil dieser nur zufällige Signale erzeugt und damit eine gute Referenz zu den Optimieren bildet, welche eine Optimierungsstrategie verfolgen. Das System wurde so konzipiert, dass es flexibel und skalierbar ist, da die Bewertung jeweils auf einem einzelnen Szenario durchgeführt wird. Dadurch kann das Bewertungssystem sowohl mit einer beliebigen Anzahl von Szenarien, einschließlich der Experimente 3, 4, 5 und 6 eingesetzt, als auch bei Bedarf um zusätzliche Bewertungskriterien erweitert werden.

5.1.1. Bewertungskriterien

Die Bewertungskriterien stellen Funktionen dar, welche die Ergebnismatrizen $\boldsymbol{J}_s^o$, aus jeweils einem Szenario resultieren, hinsichtlich der Optimierer bewerten und dafür teilweise Vergleiche mit dem Referenzalgorithmus durchführen. Dabei werden die Ergebnismatrizen für alle Optimierer aus der Menge $\mathcal{O}$ der Optimierer zu einem Szenario s berücksichtigt. In der weiteren Beschreibung dieses Kapitels verzichten wir in einer reduzierten Schreibweise auf die Hervorhebung des Szenarios, also $\boldsymbol{J}^o = \boldsymbol{J}_s^o$. Eine grafische Darstellung der Bewertungskriterien ist in Abbildung 5.1 zu finden.

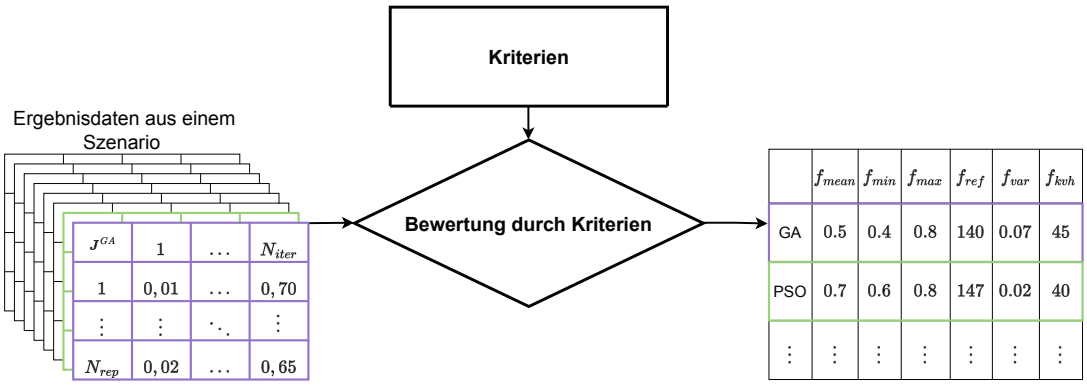


Abbildung 5.1.: Grafische Darstellung der Bewertung anhand der festgelegten Kriterien. Als Eingabedaten dienen $|\mathcal{O}|$ verschiedene $\boldsymbol{J}^o$-Matrizen, die mittels Bewertungsfunktionen in skalare Werte überführt werden. Exemplarisch wird hier die Matrix $\boldsymbol{J}^{GA}$ (rot) anhand der Kriterien bewertet. Die dargestellten Werte dienen ausschließlich der Veranschaulichung und sind beispielhaft.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\boldsymbol{J}^o$	$\boldsymbol{J}^o(\cdot, 1)$	...	...	$\boldsymbol{J}^o(\cdot, N_{iter})$
$\boldsymbol{J}^o(1, \cdot)$	0.02	...	...	0.034
$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\vdots$
$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\vdots$
$\boldsymbol{J}^o(N_{rep}, \cdot)$	0.01	...	...	0.045

Tabelle 5.1.: Ergebnisse eines Experiments zur Wiederholbarkeit $\boldsymbol{J}^o$ mit exemplarischen Zahlenwerten. Dabei beschreibt die letzte Spalte den Vektor $\bar{\boldsymbol{J}}^o$, gemäß Gleichung (5.1), welcher in jedem Eintrag $\bar{\boldsymbol{J}}^o(i)$ den letzten Kostenfunktionswert der Optimierung darstellt.

Die Bewertungskriterien analysieren die Optimierer hinsichtlich der Eigenschaften *Konvergenz*, *Wiederholgenauigkeit* und *absolute Genauigkeit*, welche bereits in Abschnitt 3.2 definiert und beschrieben wurden. Die Ergebnismatrix $\boldsymbol{J}^o$ wird gemäß (3.17) beschrieben. Dabei repräsentiert jede Zeile, bestehend aus N_{rep} Einträgen, ein Einzelszenario. Die Einträge dieser Zeilen entsprechen den Kostenfunktionswerten über die Iterationen $k = 1, \ldots, N_{iter}$ der Optimierung. Eine solche Matrix ist als Übersicht in Tabelle 5.1 dargestellt. In der weiteren Analyse werden häufig ausschließlich die Endwerte der Einzelszenarios betrachtet. Wir wählen dafür die kompakte Schreibweise

$$\bar{\boldsymbol{J}}^o(i) = \boldsymbol{J}^o(i, N_{iter}). \tag{5.1}$$

$\bar{\boldsymbol{J}}^o(i)$ beschreibt damit den finalen Kostenfunktionswert in der i-ten Wiederholung der Optimierung. Im Folgenden werden die sechs Bewertungskriterien als Funktionen vorgestellt und beschrieben. Dabei wird aufgezeigt, welche der genannten Eigenschaften von den jeweiligen Funktionen überprüft werden.

Durchschnitt

Dieses Bewertungskriterium wurde gewählt, um die mittlere, absolute Genauigkeit, als auch die Wiederholgenauigkeit des globalen Optimierungsverfahrens zu bewerten. Es basiert auf der Berechnung des arithmetischen Mittelwerts der Ergebnisse der Einzelszenarios. Das Kriterium Durchschnitt wird durch die Funktion

$$f_{mean}\left(\boldsymbol{J}^o\right) = \frac{1}{N_{rep}} \sum_{i=1}^{N_{rep}} \bar{\boldsymbol{J}}^o(i) \tag{5.2}$$

beschrieben, mit $\bar{\boldsymbol{J}}^o(i)$ gemäß (5.1) und der Anzahl der Wiederholungen N_{rep}.

Minimum

Dieses Kriterium identifiziert den niedrigsten Kostenfunktionswert der letzten Iteration, der über mehrere Wiederholungen hinweg durch den Optimierer generiert wurde. Dabei werden sowohl die Wiederholgenauigkeit als auch die Konvergenz des Optimierers überprüft. Es ist von entscheidender Bedeutung, dass der Optimierungsalgorithmus selbst im ungünstigsten

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Szenario ein akzeptables Resultat liefert, welches potenzielle Schwachstellen des Regelkreises aufdecken kann. Dieses Bewertungskriterium wird anhand der Funktion

$$f_{min}\left(\boldsymbol{J}^o\right) = \min_{i\in\{1,\ldots,N_{rep}\}} \bar{\boldsymbol{J}}^o(i), \tag{5.3}$$

mit $\bar{\boldsymbol{J}}^o$ gemäß (5.1) beschrieben.

Maximum

Dieses Kriterium bestimmt die maximal erreichten Kosten über mehrere Wiederholungen. Damit können wir ermitteln, welcher Optimierungsalgorithmus das Signal mit dem höchsten Kostenfunktionswert findet. Damit wird die absolute Genauigkeit eines Optimierers o geprüft. Die Funktion f_{max} ist gegeben durch

$$f_{max}\left(\boldsymbol{J}^o\right) = \max_{i\in\{1,\ldots,N_{rep}\}} \bar{\boldsymbol{J}}^o(i), \tag{5.4}$$

mit $\bar{\boldsymbol{J}}^o$ laut (5.1).

Referenzverhalten

Dieses Kriterium bewertet, ob der betrachtete Optimierer in den N_{rep} durchgeführten Einzelszenarios bessere, gleiche oder schlechtere Kostenfunktionswerte als der MC Algorithmus erzielt hat. Dabei wird nicht nur die Wiederholgenauigkeit untersucht, vielmehr liegt der Fokus auf einer Kombination aus absoluter Genauigkeit und Wiederholgenauigkeit. Damit wird sichergestellt, dass der Optimierungsalgorithmus nicht nur eine geringe Streuung aufweist, sondern in den Einzelszenarios auch durchgehend bessere Ergebnisse erzielt als der Referenzalgorithmus MC.

Die Bewertungsfunktion f_{ref} beschreibt das Kriterium anhand der Hilfsfunktion f^h_{ref}. Dabei werden beide Funktionen durch

$$f^h_{ref}(i) = \begin{cases} 1 & \text{falls } \bar{\boldsymbol{J}}^o(i) - \bar{\boldsymbol{J}}^{MC}(i) > 0, \\ 0 & \text{falls } \bar{\boldsymbol{J}}^o(i) - \bar{\boldsymbol{J}}^{MC}(i) = 0, \\ -1 & \text{falls } \bar{\boldsymbol{J}}^o(i) - \bar{\boldsymbol{J}}^{MC}(i) < 0, \end{cases} \tag{5.5}$$

und

$$f_{ref}\left(\boldsymbol{J}^o\right) = N_{rep} + \sum_{i=1}^{N_{rep}} f^h_{ref}(i) \tag{5.6}$$

beschrieben. Hierbei gilt wieder $\bar{\boldsymbol{J}}^o(i)$ also auch $\bar{\boldsymbol{J}}^{MC}(i)$ gemäß (5.1). Um zu verhindern, dass f_{ref} negative Werte annimmt, wird N_{rep} addiert, da aufgrund der verwendeten Normierung Probleme mit negativen Werten entstehen könnten.

Varianz

Dieses Bewertungskriterium dient zur Analyse der Streuung der Ergebnisse, die aus den N_{rep} durchgeführten Einzelszenarios hervorgehen. Es wird verwendet, um die Wiederholgenauigkeit der Optimierer zu bewerten. Ein niedriger Wert der Bewertungsfunktion f_{var}

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

weist auf eine geringe Streuung der Ergebnisse hin. Dies ist wünschenswert, da es auf eine höhere Konsistenz und Zuverlässigkeit des Optimierungsverfahrens hindeutet. Die Funktion wird durch

$$f_{var}\left(\boldsymbol{J}^o\right) = \frac{1}{N_{rep}-1}\sum_{i=1}^{N_{rep}}\left(\bar{\boldsymbol{J}}^o(i) - f_{mean}\left(\boldsymbol{J}^o\right)\right)^2. \tag{5.7}$$

beschrieben. Dabei wird $\bar{\boldsymbol{J}}^o(i)$ gemäß (5.1) und $f_{mean}\left(\boldsymbol{J}^o\right)$ gemäß (5.2) verwendet.

Konvergenzverhalten

Mit diesem Kriterium wird das Konvergenzverhalten der Optimierer untersucht. Die Bewertung erfolgt anhand eines Vergleichs zum mittleren Kostenfunktionswert des MC Algorithmus. Es soll ermittelt werden, ab welcher Iteration ein Optimierer in einem Einzelszenario $i \in \{1, \ldots, N_{rep}\}$ bessere Ergebnisse liefert als der Mittelwert von MC. Konkret wird dazu das minimale j gesucht, für das die Bedingung $f_{mean}\left(\boldsymbol{J}^{MC}\right) < \boldsymbol{J}^o(i,j)$ gilt. Dazu erstellen wir einen Vektor $\boldsymbol{k}$ der Länge N_{rep}, wobei die Einträge mit

$$k(i) = \begin{cases} N_{iter}, & f_{mean}\left(\boldsymbol{J}^{MC}\right) \geq \bar{\boldsymbol{J}}^o(i) \\ \min\{j \in \{1, \ldots, N_{iter}\} \colon f_{mean}\left(\boldsymbol{J}^{MC}\right) < \boldsymbol{J}^o(i,j)\}, & \text{sonst} \end{cases} \tag{5.8}$$

gegeben sind. Die Einträge $k(i)$ beschreiben den kleinsten Iterationsindex, ab welchem der Kostenfunktionswert des Optimierers besser ist als der Durchschnittswert $f_{mean}\left(\boldsymbol{J}^{MC}\right)$ von MC. Falls dies nie gilt, wird N_{iter} eingetragen. Über diesen Vektor $\boldsymbol{k}$ können wir die Bewertungsfunktion als

$$f_{kvh}(\boldsymbol{J}^o) = \text{Median}(\boldsymbol{k}) \tag{5.9}$$

beschreiben. Hierbei wird der Median von $\boldsymbol{k}$ berechnet, um den Einfluss von Ausreißern zu minimieren. Einige Optimierungsalgorithmen können in der Regel erst nach einer bestimmten Anzahl von Iterationen ausgewertet werden. In solchen Fällen wird der Zeitpunkt berücksichtigt, ab dem eine Auswertung möglich ist. Für GA mit einer Population von 20 Individuen würde dies in jeder 20. Iteration der Fall sein. Daher kann GA im besten Fall einen Wert von 20 erreichen.

5.1.2. Reihung der Bewertungsvektoren

Die Reihung ist essenziell für das Bewertungssystem. Sie stellt nicht nur die verschiedenen Optimierer in Relation und ermöglicht somit eine vergleichende Bewertung, sondern erlaubt auch die Analyse mehrerer Szenarien. Eine Reihung ist möglich, indem wir die vorher betrachteten Bewertungsfunktionen über einen Bewertungsvektor beschreiben. Um diesen Vektor festzulegen, erinnern wir an die Funktion F aus Gleichung (3.19), deren Bild die Bewertung der Experimente 3, 4, 5 und 6 beschreibt. Damit können wir die Bewertungsvektoren durch

$$\boldsymbol{v}_s^K = \begin{bmatrix} f_K(F(s,o_1)) \\ \vdots \\ f_K(F(s,o_n)) \end{bmatrix} \tag{5.10}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

beschreiben, wobei $K \in \mathcal{K}$ ein Kriterium bezeichnet, $o_1, \ldots, o_n \in \mathcal{O}$ die betrachteten Optimierer darstellen und s das bewertete Szenario repräsentiert. Die Menge der Kriterien $\mathcal{K}$ ist definiert als $\mathcal{K} = \{mean, min, max, ref, var, kvh\}$.

Die Reihung kann eine beliebige Anzahl an Bewertungsvektoren $\boldsymbol{v}_s^K$ verarbeiten, je nachdem wie groß die Menge der betrachteten Szenarien $\mathcal{S}$ des zu bewerteten Experiments ist. Um die Vergleichbarkeit der Bewertungen sicherzustellen, werden die Vektoren normiert und gegebenenfalls durch geeignete Transformationen angepasst, dass der höhere Wert immer der Wert ist, welcher am besten bewertet werden soll. Für Bewertungsvektoren, bei denen niedrigere Werte besser sind, wird eine Transformation des Vektors durchgeführt. Für die Varianz erfolgt die Transformation von $\boldsymbol{v}_s^{var}$ gemäß

$$\tilde{\boldsymbol{v}}_s^{var} = 1 - \frac{\boldsymbol{v}_s^{var}}{\max(\boldsymbol{v}_s^{var})}. \tag{5.11}$$

Anschließend wird der Vektor $\boldsymbol{v}_s^{var}$ durch den transformierten Vektor ersetzt. Für das Konvergenzverhalten mit $\boldsymbol{v}_s^{kvh}$ erfolgt die Transformation analog. Nach diesen Anpassungen kann die Normierung der Bewertungsvektoren durchgeführt werden. Die normierten Ergebnisvektoren werden mit $\overline{\boldsymbol{v}}_s^K$ bezeichnet und sind durch

$$\overline{\boldsymbol{v}}_s^K = \frac{\boldsymbol{v}_s^K - \min(\boldsymbol{v}_s^K)}{\max(\boldsymbol{v}_s^K) - \min(\boldsymbol{v}_s^K)} \tag{5.12}$$

festgelegt, wobei $K \in \mathcal{K}$ ein Bewertungskriterium beschreibt. Ein Vektor $\overline{\boldsymbol{v}}_s^K$ enthält nun Werte, die im Intervall zwischen 0 und 1 liegen. Die Normierung bewahrt die relativen Verhältnisse zwischen den Werten, führt jedoch dazu, dass in jeder durchgeführten Bewertung jeweils ein Optimierer 0 Punkte und ein anderer 1 Punkt erhält. Um zu vermeiden, dass ein Optimierer stets 0 Punkte erhält, und gleichzeitig eine aussagekräftigere Skalierung zu erreichen, wird empfohlen, einen Referenzalgorithmus wie MC zu verwenden. Dieser Ansatz ermöglicht eine sinnvolle Interpretation der relativen Verhältnisse und stellt sicher, dass die Unterschiede zwischen den Bewertungsergebnissen hinreichend deutlich hervortreten. Der normierte Vektor $\overline{\boldsymbol{v}}_s^K$ stellt ausschließlich die Reihung der verschiedenen Algorithmen in Bezug auf ein einzelnes Szenario dar. Um ein Gesamtergebnis für alle bewerteten Szenarien zu erhalten, wird der Mittelwert über alle betrachteten Szenarien $\mathcal{S}$ gebildet. Auf dieser Grundlage wird die endgültige Bewertung durchgeführt. Der Ergebnisvektor, der die Bewertung über alle Szenarien repräsentiert, ergibt sich gemäß

$$\boldsymbol{P}_K = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \overline{\boldsymbol{v}}_s^K \tag{5.13}$$

wobei $\mathcal{S}$ die Menge der betrachteten Szenarien bezeichnet, die durch das Bewertungssystem analysiert werden sollen, und $|\mathcal{S}|$ die Kardinalität dieser Menge beschreibt. Die Vektoren $\boldsymbol{P}_K$ dienen als Grundlage zur Analyse der Algorithmen. Insbesondere wird dabei auf die Prozent Bezug genommen, die ein Algorithmus o in einem bestimmten Kriterium K erreicht hat. Um eine gesamtheitliche Bewertung der Algorithmen zu ermöglichen, wird zusätzlich der Mittelwert über alle Bewertungskriterien gebildet. Der Gesamtbewertungsvektor ergibt sich zu

$$\boldsymbol{P}_{all} = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \boldsymbol{P}_k, \tag{5.14}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

wobei $|\mathcal{K}|$ die Anzahl der Bewertungskriterien bezeichnet. Der Vektor $\boldsymbol{P}_{all}$ beschreibt somit das Ergebnis einer vollständigen Bewertung in Prozent.

5.2. Ergebnisse der Simulationsstudien

In diesem Abschnitt werden die Ergebnisse der Simulationsstudie analysiert, wobei wir die Experimente 4 und 5 bewerten. Zunächst wird das Experiment 5, das die Gesamtbewertung umfasst und in Abschnitt 3.2 beschrieben ist, anhand des entwickelten Bewertungssystems untersucht. Anschließend wird der Vergleich der verschiedenen Optimierer anhand der vier verwendeten Regelkreise diskutiert. Dies umfasst eine Analyse der Regelkreise, die auf den identifizierten Worst-Case Sollsignalen basiert. Ergänzende Informationen zur Implementierung und Wahl der Hyperparameter der globalen Optimierungsverfahren sind im Anhang A.1 dokumentiert.

5.2.1. Gesamtbewertung

Die Gesamtbewertung $\boldsymbol{P}_{all}$ sowie die einzelnen Kriterien $\boldsymbol{P}_K$ sind in Tabelle 5.2 zu finden. Zusätzlich ist die Gesamtbewertung $\boldsymbol{P}_{all}$ in Abbildung 5.2 visualisiert.

Hierbei erzielt der Surrogate Algorithmus mit $88,22\,\%$ die höchste Gesamtbewertung. Besonders positiv fällt auf, dass der Algorithmus in den Kriterien, welche die Wiederholgenauigkeit und damit die Streuung der Ergebnisse bewerten, sehr hohe Prozentpunkte erreicht. Im Minimum-Kriterium werden $96,84\,\%$ erreicht, im Referenzverhalten-Kriterium

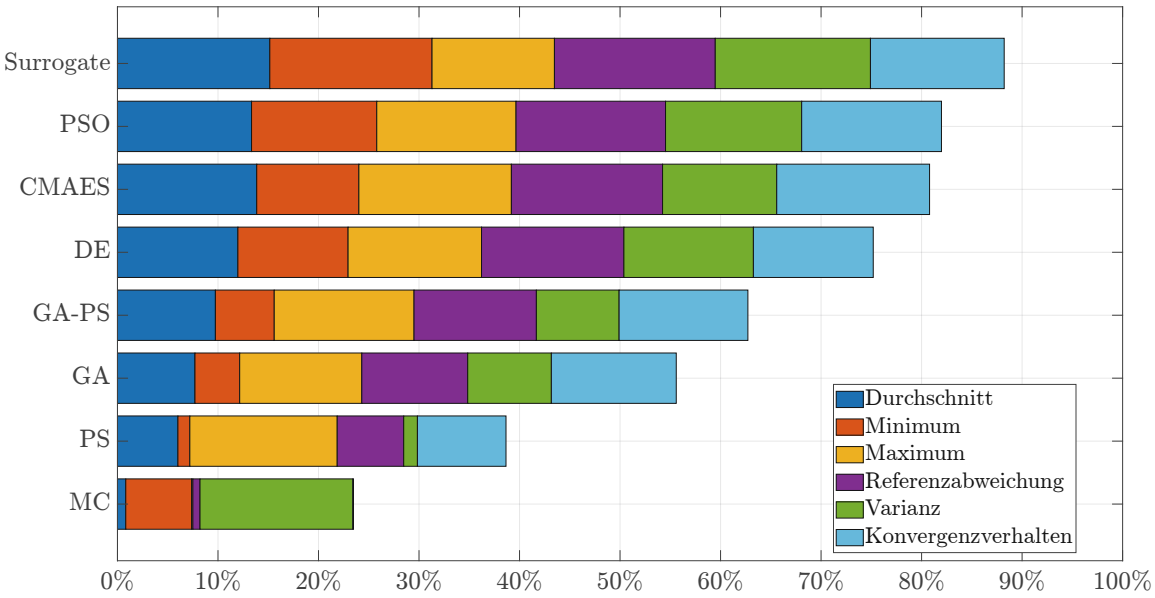


Abbildung 5.2.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 5. $\boldsymbol{P}_{all}$ ist in Prozent angeben und setzt sich aus den Bewertungen der 6 Kriterien zusammen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{ref}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
Surrogate	88,22 %	90,93 %	96,84 %	73,06 %	96,00 %	92,63 %	79,84 %
PSO	81,98 %	80,04 %	74,74 %	83,14 %	89,25 %	81,26 %	83,45 %
CMAES	80,79 %	83,13 %	60,98 %	91,03 %	90,25 %	68,15 %	91,23 %
DE	75,19 %	71,88 %	65,78 %	79,65 %	84,99 %	77,30 %	71,54 %
GA-PS	62,72 %	58,45 %	35,12 %	83,45 %	73,03 %	49,39 %	76,87 %
GA	55,59 %	46,26 %	26,69 %	72,90 %	63,23 %	49,93 %	74,56 %
PS	38,65 %	36,09 %	7,11 %	87,98 %	39,70 %	8,18 %	52,88 %
MC	23,48 %	4,95 %	39,27 %	0,79 %	4,27 %	91,25 %	0,35 %

Tabelle 5.2.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 5. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet und beziehen sich auf den Mittelwert der jeweiligen normierten Punkte in jedem Kriterium, bis auf $\boldsymbol{P}_{all}$, welches den Mittelwert über alle Kriterien entspricht.

werden 96 % erreicht. Auch in der Bewertung der Varianz erzielt der Surrogate Algorithmus mit 92,63 % hohe Ergebnisse. Diese Beobachtungen stimmen mit den Erkenntnissen aus der Literatur überein, da der Surrogate Algorithmus mit den gewählten Eigenschaften eine geringe Streuung der Ergebnisse und somit eine hohe Wiederholgenauigkeit aufweist. Die durchschnittlichen Werte der Kostenfunktion am Ende der Optimierung, bewertet anhand des Durchschnittskriteriums, weisen ebenfalls eine hohe Bewertung von 90,93 % auf. Das im Vergleich niedrigere Maximum der Kostenfunktionsbewertung mit 73,06 % ist auf die hohe Varianz anderer Algorithmen zurückzuführen, welche zu einzelnen sehr hohen Auswertungen führen. Damit zeigt der Surrogate Algorithmus eine gute absolute Genauigkeit. Die Konvergenzgeschwindigkeit ist mit 79,8 % im guten Mittelfeld. Diese Bewertung ist auf die lange Anlaufphase des hier verwendeten Surrogate Algorithmus zurückzuführen. Dabei wurde eine Konfiguration gewählt, bei der zunächst ein Surrogatmodell basierend auf 30 ausgewählten Punkten aufgebaut wird. Erst nach der 30. Iteration nimmt die Konvergenzgeschwindigkeit signifikant zu, was im Vergleich zu anderen Algorithmen, welche spätestens nach der 20. Iteration, mit der adaptiven Suche beginnen, ein Nachteil ist.

Die Particle Swarm Optimization (PSO) erreicht eine Gesamtbewertung von 81,98 % und belegt damit den zweiten Platz in der Rangfolge. Entgegen den Erwartungen aus der Literaturrecherche liefert PSO sehr gute Ergebnisse, wobei die bewerteten Punkte auf die Kriterien gleichmäßig verteilt sind. PSO erreicht 80,04 % im Durchschnitt, 74,74 % im Minimum, 83,14 % im Maximum, 89,25 % in dem Referenzverhalten, 81,26 % in der Varianz und 83,45 % in der Konvergenzgeschwindigkeit. Die anfängliche Vermutung, dass PSO Probleme mit der Unstetigkeit der Kostenfunktion haben könnte, hat sich nicht bestätigt. Dass PSO je nach getestetem Regelkreis unterschiedliche Ergebnisse liefert, werden wir in den folgenden Unterabschnitten beobachten können.

Die Covariance Matrix Adaptation Evolution Strategy (CMAES) erreicht mit einer Gesamtbewertung von 80,79 % den dritten Platz und liegt damit nur knapp hinter PSO. Im Gegensatz zu PSO zeigt sich beim CMAES ein deutlich ausgeprägteres Profil. Hervorzuheben sind die sehr guten Ergebnisse im Durchschnitts-Kriterium mit 83,13 %, in der Kon-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

vergenzgeschwindigkeit mit 91,23 % sowie im Maximum-Kriterium mit 91,03 %. Demgegenüber fallen die Bewertungen im Minimum-Kriterium mit 60,98 % und in der Varianz mit 68,15 % vergleichsweise gering aus. Dies lässt sich darauf zurückführen, dass der CMAES eine niedrigere Wiederholgenauigkeit und somit eine hohe Streuung der Ergebnisse zeigt. Die hohe absolute Genauigkeit des CMAES kompensiert die geringere Wiederholgenauigkeit weitgehend. Diese Einschätzung wird durch den hohen Wert im Referenzverhalten-Kriterium mit 90,8 % untermauert. Dieses Kriterium bewertet, wie häufig der Optimierer in den Wiederholungen ein besseres Ergebnis als MC erzielt hat, siehe Abschnitt 5.1.1. Die Ergebnisse des CMAES bestätigen die in Abschnitt 2.2 diskutierten Eigenschaften aus der Literatur.

Der Differential Evolution Algorithm (DE) erzielt eine Gesamtbewertung von 75,19 % und erreicht damit etwas weniger Prozentpunkte. Das Profil des DE ähnelt dem von PSO, wobei er in allen Kriterien mittelmäßige Ergebnisse erzielt. Eine Ausnahme bildet $\boldsymbol{P}_{ref}$, bei dem DE 84,99 % erreicht. Die Ergebnisse in den weiteren Bewertungskriterien sind wie folgt: 71,88 % im Durchschnitt, 65,78 % im Minimum, 79,65 % im Maximum, 77,3 % in der Varianz und 71,54 % in der Konvergenzgeschwindigkeit. Damit erreicht DE mehr Prozent als ursprünglich erwartet, was darauf hinweist, dass die Unstetigkeit der Kostenfunktion für diesen Optimierer kein Problem darstellt. Im Gegensatz zu PSO bleibt DE jedoch konsistent im Mittelfeld der Bewertungen und zeigt keine ausgeprägten Stärken an einzelnen Modellen.

Der Genetic Algorithm (GA) erzielt eine Gesamtbewertung von 55,59% und zeigt dabei deutliche Schwächen. Das in der Literatur bekannte Problem der hohen Streuung der Ergebnisse spiegelt sich in der niedrigen Bewertung von $\boldsymbol{P}_{min}$ mit 26,69% und $\boldsymbol{P}_{var}$ mit 49,93% wieder. Auch die absolute Genauigkeit überzeugt nicht mit einem Wert von 46,26% im Durchschnitts-Kriterium und einer Bewertung von 63,23% im Referenzverhalten-Kriterium. Die Ergebnisse im Maximum mit 72,9% und in der Konvergenzgeschwindigkeit mit 74,56% sind hingegen vergleichsweise gut. Ein Hauptgrund für die schwache Leistung von GA ist die unpassende Wahl der Startwerte. In Abschnitt 5.3 wird gezeigt, wie eine Wahl der Startwerte die Leistung des GA deutlich verbessert.

Der Patternsearch-Algorithmus (PS) belegt mit einer Gesamtbewertung von 38,65% den vorletzten Platz. Positiv hervorzuheben ist lediglich die Bewertung im Maximum-Kriterium, in dem PS mit 87,98% gut abschneidet, da er in vielen Szenarien ein besonders anspruchsvolles Testsignal findet. Jedoch ist die Wiederholgenauigkeit außergewöhnlich niedrig. Im Minimum-Kriterium erzielt PS lediglich 7,1%, und auch die Varianz wird mit 8,1% äußerst schlecht bewertet. Die hohe Streuung der Ergebnisse führt dazu, dass PS in diesen beiden Kriterien häufig eine Bewertung von 0 erhält, was sich negativ auf $\boldsymbol{P}_{mean}$ mit 36,09%, $\boldsymbol{P}_{ref}$ mit 39,7% und $\boldsymbol{P}_{kvh}$ mit 52,88% auswirkt. In kleineren Parametermengen zeigt PS eine bessere Bewertung, erreicht jedoch auch dort maximal eine Gesamtbewertung von 50%. Der Hauptgrund für das schwache Abschneiden liegt in der Abhängigkeit von einem einzelnen Startwert. Ist dieser ungünstig gewählt, führt PS in der Regel zu keinem brauchbaren Ergebnis.

Der hybride Algorithmus Genetic Algorithm - Patternsearch (GA-PS) erreicht eine Gesamtbewertung von 62,72% und liegt damit deutlich hinter DE. Besondere Schwierigkeiten hat GA-PS im Minimum-Kriterium und in der Varianz, mit Bewertungen von lediglich 35,12% bzw. 49,39%. Auch die Bewertung des Durchschnitts-Kriteriums mit 58,45% ist niedrig. Bessere Ergebnisse erzielt der Algorithmus im Maximum mit 83,45% und in der

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Konvergenzgeschwindigkeit mit 74, 56%. Trotzdem erreicht GA-PS einen Wert von 73, 03% im Referenzverhalten-Kriterium. Dies deutet darauf hin, dass GA-PS trotz seiner niedrigen Wiederholgenauigkeit eine ausreichend hohe absolute Genauigkeit aufweist, um in den meisten Fällen zumindest eine suboptimale Lösung zu finden, welche besser als die von MC war. GA-PS zeigt über verschiedene Modelle hinweg eine ähnliche Performance, wobei er bei kleineren Parametermengen bessere Ergebnisse liefert. Dennoch erreicht er auch in diesen Szenarien maximal den dritten Platz. Die vergleichsweise niedrige Bewertung von GA-PS war jedoch zu erwarten, da die Ergebnisse von GA und PS ähnliche Schwierigkeiten bei der Optimierung zeigen.

Der Algorithmus der Monte-Carlo Optimierung (MC) erreicht eine Gesamtbewertung von 23, 48 % und belegt damit, wie zu erwarten, den letzten Platz. Die gute Bewertung der Varianz mit 91, 25 % war vorhersehbar, da die zufällige Natur des MC Algorithmus zu einer gleichmäßigen Verteilung der Ergebnisse führt. Diese Gleichmäßigkeit resultiert aus dem Fehlen gezielter Optimierungsschritte. Statt systematisch bessere Lösungen anzustreben, generiert der MC-Algorithmus Ergebnisse rein zufällig über den gesamten Suchraum hinweg. Dadurch entstehen keine systematischen Abweichungen oder Schwerpunkte in den Resultaten, was die Streuung der Werte minimiert und zu einer niedrigen Varianz führt. Interessant zu beobachten ist, dass die Bewertung des Minimum-Kriteriums mit 39, 27 % nicht die niedrigste unter den betrachteten Algorithmen ist. MC erreicht damit deutlich mehr Prozent als PS, GA und etwas mehr als GA-PS. Auffällig ist zudem, dass der MC-Algorithmus mit 4, 95 % im Durchschnitts-Kriterium hin und wieder nicht der schlechteste Optimierer war.

Abseits der Einzelbewertung ist besonders hervorzuheben, dass MC im Referenzabweichungs-Kriterium einen Wert von 4, 27 % aufweist. Dies ist überraschend, da es darauf hindeutet, dass ein Optimierer in bestimmten Szenarien häufiger schlechtere Kostenfunktionswerte als MC hatte. Konkret handelt es sich dabei um den PS Algorithmus, der in einigen Szenarien während der Wiederholungstests in mehr als $N_{rep}/2$ Fällen niedrigere Werte der Kostenfunktion erzielte als der MC Algorithmus. Dieses Ergebnis deutet darauf hin, dass der PS Algorithmus in diesen spezifischen Fällen eine geringere Stabilität aufwies und trotz seines algorithmischen Ansatzes in der Optimierung keine signifikanten Vorteile gegenüber dem rein zufälligen Ansatz des MC Algorithmus zeigte.

Um ein genaueres Bild der Leistungsfähigkeit der Algorithmen zu erhalten, werden im Folgenden die Ergebnisse an den einzelnen Regelkreisen detailliert untersucht, da davon auszugehen ist, dass diese einen wesentlichen Einfluss auf die Komplexität der Aufgabe haben. Dabei liegt ein besonderes Augenmerk auf der Analyse der Regelkreise anhand der identifizierten der Worst-Case Testsignale.

5.2.2. lineares Feder-Masse System

In diesem Abschnitt wird das Modell des linearen Feder-Masse Systems (FMSL), siehe Abschnitt 4.1, detaillierter anhand der Konfiguration laut Experiment 4 untersucht. Zunächst werden die Algorithmen anhand des Bewertungssystems analysiert, siehe Abbildung 5.3 und Tabelle A.1. Anschließend wird das Regelverhalten von FMSL diskutiert. Aus der Bewertung geht hervor, dass der Algorithmus PSO mit einem Wert von 93, 96 % die höchste Bewertung erreicht und im Vergleich zur Gesamtbewertung eine bessere Leistung zeigt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

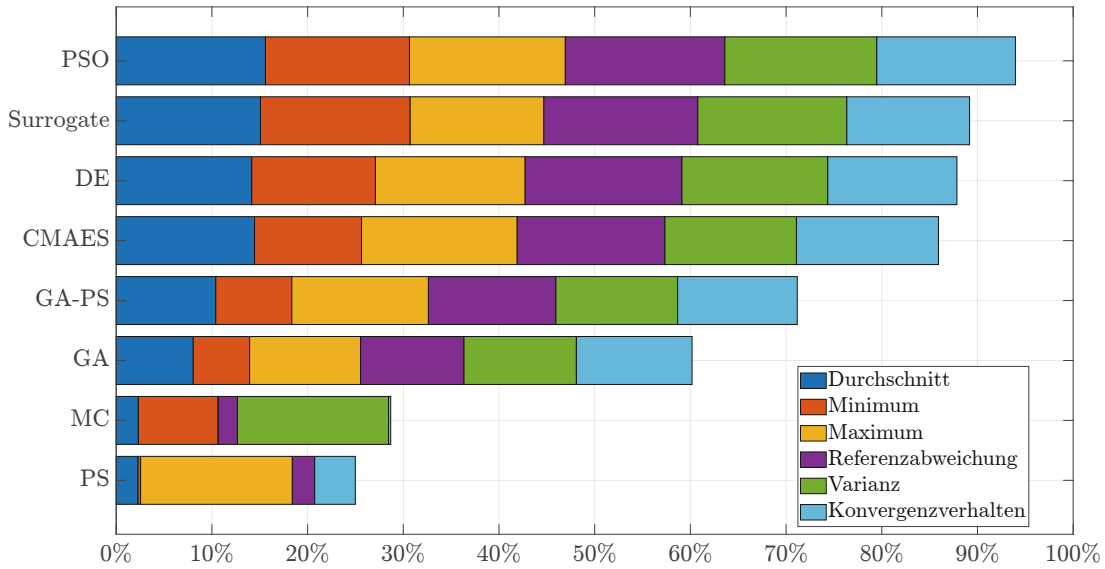


Abbildung 5.3.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell FMSL. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen.

Der Surrogate-Algorithmus folgt mit $89,18\,\%$. Knapp dahinter liegen DE mit $87,85\,\%$ und CMAES mit $85,92\,\%$. Alle genannten Algorithmen überschreiten die Marke von $85\,\%$. Der Algorithmus GA-PS erreicht $71,17\,\%$, während GA mit $60,18\,\%$ bewertet wird. Beide Algorithmen zeigen jedoch eine bessere Leistung im Vergleich zur Gesamtbewertung. Diese Ergebnisse deuten darauf hin, dass das Worst-Case Sollsignal für FMSL bei diesen Optimierern relativ leicht zu identifizieren war, während der Referenzalgorithmus MC Schwierigkeiten hatte, dieses Signal zufällig zu finden. Diese Beobachtung wird durch die Bewertung des Kriteriums $\boldsymbol{P}_{ref}$ gestützt, da die meisten Optimierer in diesem Kriterium mehr als $80\,\%$ erreichen. Der Algorithmus PS erreicht an diesem Modell lediglich eine Gesamtbewertung von $24,99\,\%$ und bleibt damit hinter dem Referenzalgorithmus MC zurück.

Die relativen Stärken und Schwächen der Algorithmen decken sich weitgehend mit der Gesamtbewertung. PSO und DE erzielen durchweg gute Ergebnisse in allen Kriterien, wobei PSO in $\boldsymbol{P}_{ref}$ $100\,\%$ erreicht und damit stets besser als MC abschneidet. Der Surrogate-Algorithmus zeichnet sich durch eine geringe Streuung der Ergebnisse aus, während CMAES eine hohe absolute Genauigkeit erreicht. GA und GA-PS zeigen hingegen Schwierigkeiten mit der Wiederholgenauigkeit und eine mittelmäßige absolute Genauigkeit. PS weist eine äußerst geringe Wiederholgenauigkeit auf, mit $0\,\%$ in der Varianz. Das heißt, dass PS in jedem der 8 Szenarien die höchste Varianz erzielt hatte und damit immer 0 Punkte erreicht hat. Dennoch scheint PS bei geeigneter Initialisierung häufig das Maximum zu finden.

Nachdem wir nun das Gesamtverhalten der Optimierer betrachtet haben, stellt sich die Frage, wie ein tatsächliches Worst-Case Signal für FMSL aussieht. Zunächst muss jedoch geklärt werden, welche Beispiele in die Analyse einbezogen werden. Es werden ausschließlich Beispiele mit den Parametermengen Ξ_6 oder Ξ_4 berücksichtigt, da diese sich aufgrund des

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

stationären Endabschnitts des Signals am besten für die Analyse eignen. Außerdem wurde die Position als auch der Eingang in allen verwendeten Abbildungen normiert, bzw. skaliert, damit eine Vergleichbarkeit zwischen den verschiedenen Regelkreisen hergestellt werden kann.

Eine Simulation des Regelkreises FMSL ist in Abbildung 5.4 dargestellt. Diese zeigt das Worst-Case Sollsignal y^d, das mit CMAES unter Verwendung der Parametermenge Ξ_6, erkennbar an den drei Umschaltstellen, und MAE als Kostenfunktion ermittelt wurde, siehe erste Grafik von Abbildung 5.4. Der Ausgang des Regelkreises y_{sys} scheint dem Sollsignal hinreichend gut zu folgen. Man erkennt, dass der Regelfehler, dessen Betrag $|y^d - y_{sys}|$ im 3. Abschnitt von Abbildung 5.4 dargestellt ist, vor allem am Ende der dynamischen

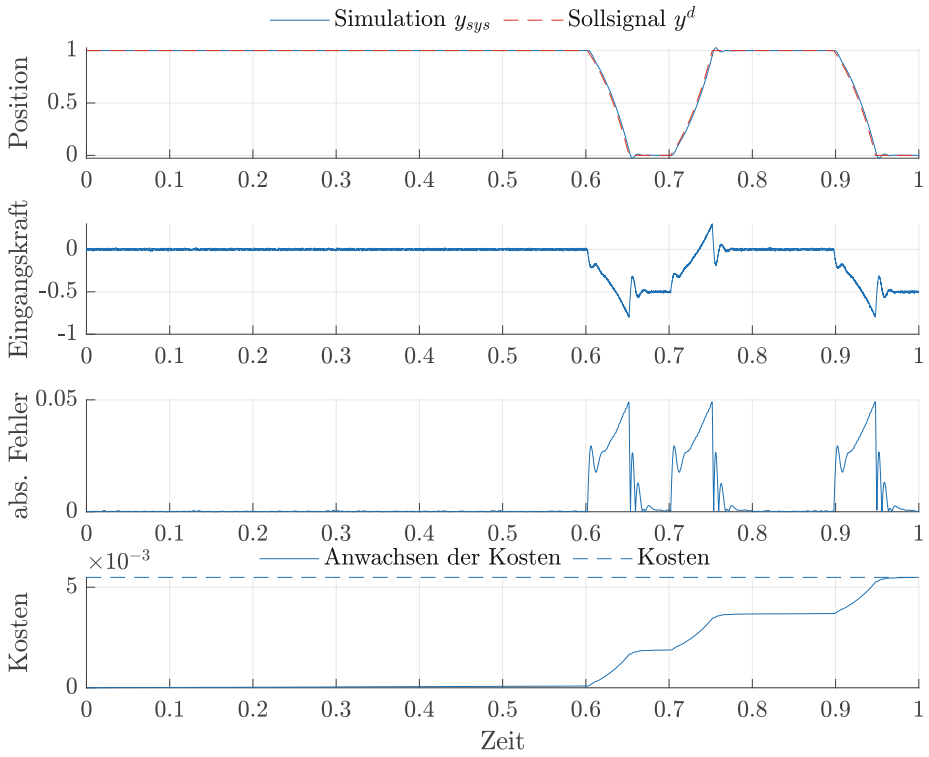


Abbildung 5.4.: Simulation eines Worst-Case Signals an dem Modell FMSL, das durch die Optimierung mittels CMAES unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes MAE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von FMSL. In der zweiten Grafik ist die Eingangskraft F_u darstellt. Darunter ist der absolute Fehler in jedem Abtastschritt eingetragen. In der letzten Grafik ist das Anwachsen der Kostenfunktion und die Kostenfunktion dargestellt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Bereiche zunimmt. Zu Beginn der stationären Abschnitte ist ein Überschwingen erkennbar, welche auch deutlich in der Eingangskraft zu erkennen ist, siehe 2.te Grafik. Der Optimierer scheint dieses Verhalten gezielt zu provozieren, da die Arbeitspunktwechsel nahe den Rändern des Arbeitsbereichs mit minimalem Abstand zueinander erzeugt wurden, um die möglichst hohe Änderungen im Sollsignal zu erreichen. Zusätzlich lässt sich erkennen, dass hier Bausteine mit nicht differenzierbaren Übergängen an den Enden ausgewählt wurden. Dadurch nähert sich das Verhalten des Systems der Analyse mit mehreren Sprungantworten an, was eine gängige Methode zur Untersuchung linearer Systeme darstellt. Der Verlauf der Kostenfunktion, zu sehen in der untersten Grafik von Abbildung 5.4, zeigt deutlich, dass die Arbeitspunktwechsel in den dynamischen Bereichen den wesentlichen Beitrag liefern.

Zusammenfassend lässt sich feststellen, dass der optimierungsbasierte Signalgenerator erfolgreich ein Testsignal erzeugt hat das gezielt eine Schwäche im Regelkreis hervorruft. Für ein lineares System handelt es sich dabei jedoch um ein typisches Testszenario, das häufig zur Validierung von Reglern eingesetzt wird. Das beobachtete Verhalten entspricht daher durchaus der vorherrschenden Erwartungshaltung.

5.2.3. nichtlineares Feder-Masse System

In diesem Abschnitt betrachten wir nun das nichtlinearen Feder-Masse System (FMSN) aus Abschnitt 4.1. In Abbildung 5.5 und Tabelle A.2 ist die Bewertung für das Modell FMSN zu finden. Die Bewertungen der Optimierer zeigen deutliche Unterschiede, insbesondere im Vergleich zur Gesamtbewertung, siehe Abbildung 5.2.

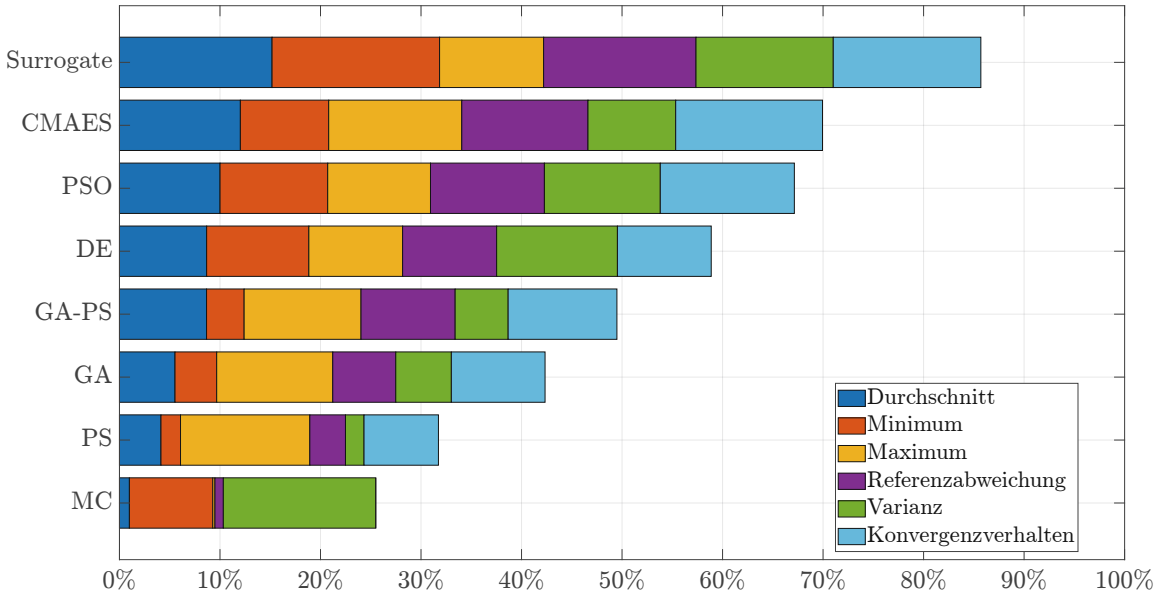


Abbildung 5.5.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell FMSN. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Der Surrogate-Algorithmus erreicht mit $85,7\,\%$ den höchsten Wert und belegt damit den ersten Platz. Auf dem zweiten Platz folgt CMAES mit $69,94\,\%$, was einem Rückstand von knapp $15\,\%$ entspricht. PSO erreicht $67,14\,\%$ und belegt damit den dritten Platz, während DE mit $58,87\,\%$ an vierter Stelle steht. Alle weiteren Optimierer liegen unterhalb der $50\,\%$-Marke.

Das Profil der Bewertung der Algorithmen im FMSN zeigt ähnliche Tendenzen wie die Gesamtbewertung. Besonders hervorzuheben ist, dass der Surrogate Algorithmus im Minimum-Kriterium $100\,\%$ erreicht und somit in allen acht Szenarien stets den besten Minimalwert erzielt, damit zeigt Surrogate eine hervorragende Wiederholgenauigkeit. PSO und DE erreichen in allen Kriterien eine überdurchschnittliche Bewertung. PSO weist hier eine bessere Konvergenzgeschwindigkeit als in der Gesamtbewertung auf, während DE in $\boldsymbol{P}_{ref}$ eine höhere Bewertung erreicht. CMAES zeigt geringe Prozente in der Varianz mit $53,8\,\%$, was auf die hohe Streuung der Ergebnisse zurückzuführen ist. Nichtsdestotrotz fällt die Bewertung der absoluten Genauigkeit durch die Kriterien Maximum, $79,44\,\%$ und Durchschnitt, $73,64\,\%$ hoch aus. Für GA, GA-PS und PS gilt ein ähnliches Profil, wobei diese Algorithmen eine niedrige absolute Genauigkeit und eine noch niedrigere Wiederholgenauigkeit als CMAES aufweisen.

Die Schwierigkeit dieses Systems wird besonders im Referenzverhalten-Kriterium deutlich. Während bei FMSL fast alle Algorithmen Werte über $70\,\%$ erzielten, überschreiten bei FMSN nur der Surrogate-Algorithmus mit $91\,\%$ und der CMAES mit $76,8\,\%$ die $70\,\%$-Marke. Dies bedeutet, dass in einigen Einzelszenarios der Referenzalgorithmus, welcher zufällige Signale generiert, bessere Ergebnisse erzielt hat. Daraus kann geschlossen werden, dass die Identifikation der Worst-Case Signale für die Optimierer eine erhebliche Herausforderung darstellte.

Eines dieser Worst-Case Testsignale werden wir anhand der Simulation des Regelkreises FMSN analysieren. Diese Simulation ist in Abbildung 5.6 dargestellt. Das mit dem Optimierer Surrogate, Ξ_4 und d_{rmse} ermittelte Sollsignal y^d ist gemeinsam mit dem Systemausgang y_{sys} dargestellt, siehe oberste Grafik. Dabei erkannt man, dass in den stationären Bereichen das Ausgangssignal y_{sys} Grenzzyklen aufweist, welche auch im quadratischen Fehler, siehe 3. Grafik, deutlich sichtbar sind. Dieses Schwanken liegt an der Haftreibung, bei welcher die Masse zur Ruhe kommt, falls die darauf wirkende Kraft zu niedrig ist. Aufgrund des Integralanteils des Reglers, baut dieser über eine gewisse Zeit die Kraft auf, welche wir im 2. Abschnitt von Abbildung 5.6 betrachten können. Diese Kraft führt schlussendlich zu einem Losbrechen der Masse, ehe sie wieder starr verharrt. Wesentliche Beiträge zur Kostenfunktion sind insbesondere im stationären Bereich des Signals zu beobachten, was durch die Grafik des akkumulierten Fehlers verdeutlicht wird, siehe letzte Grafik von Abbildung 5.6.

Zusammenfassend konnte ein Sollsignal identifiziert werden, das eine Schwäche im Regler offenbart. Insbesondere zeigte sich, dass die erwartete Einfrier-Eigenschaft des Reglers nicht wie vorgesehen wirksam war. Entgegen den Erwartungen führte diese Eigenschaft nicht dazu, dass der Integralanteil während der Haftreibungsphase in der Nähe des Sollsignals nicht weiter anstieg. Dies hatte zur Folge, dass das System in Grenzzyklen überging, anstatt eine stabile stationäre Lage zu erreichen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

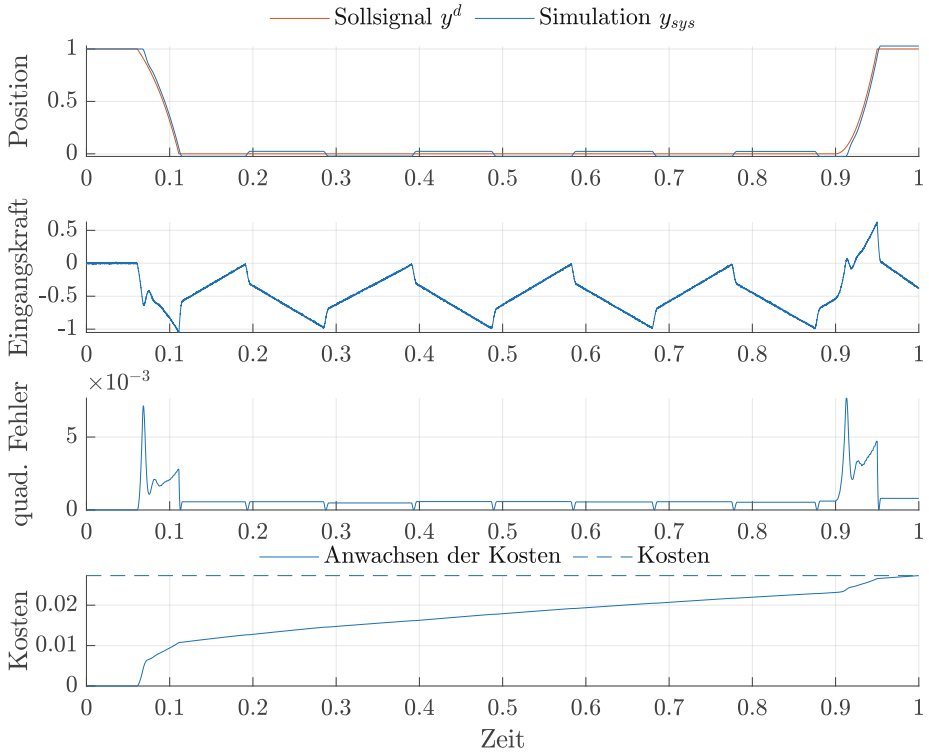


Abbildung 5.6.: Simulation eines Worst-Case Signals für FMSN, das durch die Optimierung mittels Surrogate unter Verwendung der Parametermenge Ξ_4 und des Fehlermaßes RMSE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von FMSN. In der zweiten Grafik ist die Eingangskraft F_u darstellt. Darunter ist der quadratische Fehler in jedem Abtastschritt eingetragen. In der letzten Grafik ist das Anwachsen der Kostenfunktion und die Kostenfunktion dargestellt.

5.2.4. Pneumatisches Ventil mit Druck als Eingangsgröße

In diesem Abschnitt wird das pneumatische Ventil mit der Eingangsgröße Druck (PVED) analysiert. Im Vergleich zum pneumatischen Ventil mit der Eingangsgröße Massenstrom (PVEM) zeichnet sich das PVED durch eine geringere Modellkomplexität aus. Dennoch weist der Regler des PVED eine höhere Regelungskomplexität auf, da ein PID-Regler verwendet wird, während beim PVEM lediglich ein PI-Regler zum Einsatz kommt, genaueres dazu in Abschnitt 4.2.

Wie in Abbildung 5.7 erkennbar, liegen die prozentualen Werte der Optimierer in diesem Modell deutlich höher als in der Gesamtbewertung, Abbildung 5.2, als auch im Modell FMSN, Abbildung 5.5. Dies deutet darauf hin, dass es an diesem Modell vergleichsweise einfacher war, ein Worst-Case Testsignal zu identifizieren. Diese Schlussfolgerung wird

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

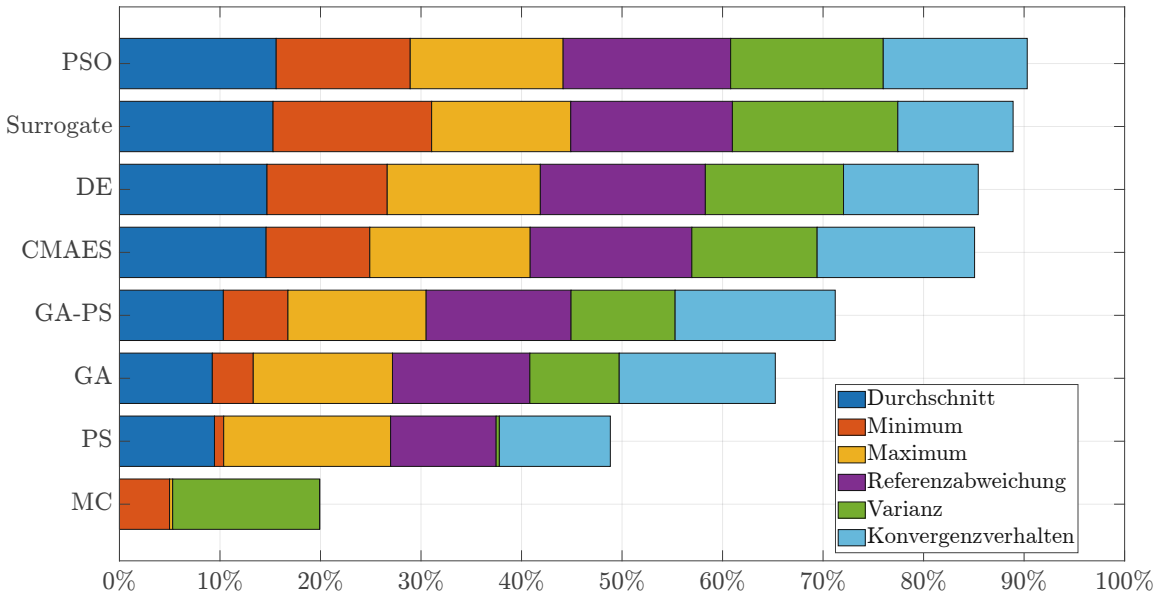


Abbildung 5.7.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell PVED. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen.

durch die Werte von $\boldsymbol{P}_{ref}$ gestützt, bei dem alle Optimierer, mit Ausnahme von PS, Werte von über 80 % erreichen. Auffällig ist, dass PS in diesem Modell mit einem Wert von 63 % seine beste Leistung im Vergleich zu den anderen Modellen erzielt. Die detaillierten Daten zu den einzelnen Kriterien sind im Anhang als Tabelle dargestellt, siehe Tabelle A.2.

PSO erreicht mit 90, 31 % die höchste Bewertung, dicht gefolgt von Surrogate mit 88, 91 %. DE und CMAES erzielen mit 85, 44 % bzw. 85, 44 % nahezu gleichwertige Ergebnisse. GA-PS und GA liegen mit 71, 21 % beziehungsweise 65, 25 % deutlich darunter. PS weist mit 48, 84 % im Vergleich zu seinen Ergebnissen an anderen Modellen eine höhere Leistung auf, bleibt jedoch hinter den anderen Optimierern zurück und belegt den vorletzten Platz. Wenig überraschend erreicht der Referenzalgorithmus MC den letzten Platz mit 19, 02 %.

Die Bewertung der Algorithmen zeigt ähnliche Stärken und Schwächen wie in der Gesamtbewertung. Die Algorithmen PS, GA und GA-PS zeichnen sich durch eine moderate absolute Genauigkeit und eine niedrige Wiederholgenauigkeit aus. Besonders PS weist mit einer Varianzbewertung von 1, 92 % und 99, 63 % im Maximal-Kriterium extreme Schwankungen auf. DE erzielt insgesamt solide Ergebnisse, zeigt jedoch eine Diskrepanz zwischen dem Referenzverhalten-Kriterium, 98, 5 % und dem Minimal-Kriterium mit 71, 8 %. CMAES ist erneut durch eine höhere Streuung der Ergebnisse charakterisiert, wobei die absolute Genauigkeit in diesem Fall nicht signifikant besser als bei vergleichbaren Algorithmen ist. Der Surrogate Algorithmus überzeugt durch eine geringe Varianz in den Ergebnissen, 98, 7 %, weist jedoch Schwächen in der Konvergenzgeschwindigkeit auf, 68, 8 %. Der PSO-Algorithmus zeigt in allen Bewertungskriterien überzeugende Leistungen mit Werten über 80 % und erreicht im Referenzverhalten-Kriterium sogar die Maximalbewertung von 100 %.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Damit war PSO in jedem der insgesamt 400 durchgeführten Einzelszenarien, 8 Szenarien mit jeweils 50 Einzeltests, immer besser als der Referenzalgorithmus MC.

Abbildung 5.8 zeigt eine Simulation des Systems PVED basierend auf einem Worst-Case Sollsignal y^d. Dieses Signal wurde durch Optimierung mit dem Surrogate-Algorithmus unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes d_{rmse} generiert. In der oberen Grafik, inwelcher y^d und y_{sys} dargestellt sind, ist ein sehr gutes Folgeverhalten des Regelkreises erkennbar. In der Darstellung des quadratischen Fehlers, dritte Grafik, wird deutlich, dass die Fehler nahezu ausschließlich in den dynamischen Abschnitten des Signals auftreten. Obwohl das System eine Kontaktkraft f_k aufweist, die in den oberen und unteren Positionen potenziell problematisch sein kann, kompensiert der Regler dies durch den gezielten direkten Aufbau und Abbau des Drucks. Dieses Verhalten lässt sich insbesondere

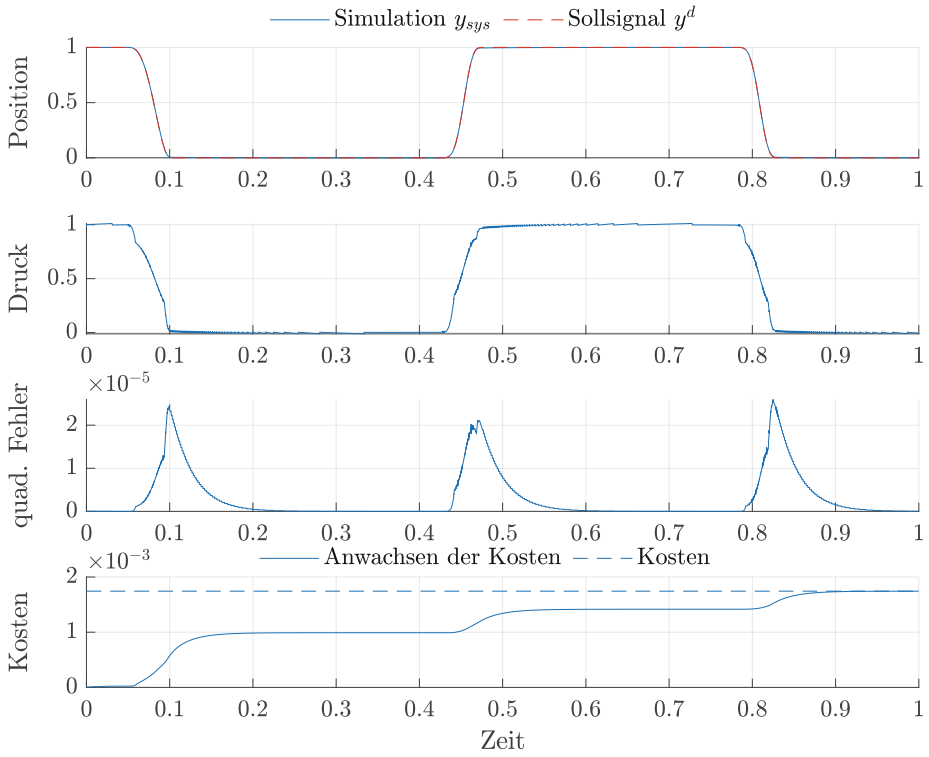


Abbildung 5.8.: Simulation eines Worst-Case Signals für PVED, das durch die Optimierung mittels Surrogate unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes RMSE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von PVED, als auch die Eingangsgröße Druck, welche mit dem maximal möglichen Druck skaliert wurde. Darunter ist der quadratische Fehler in jedem Zeitschritt abgebildet. In der vierten Grafik sind die Kosten zusehen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

im Bereich zwischen 0.45 und 0.8 im Druckverlauf beobachten. Nach den dynamischen Abschnitten zeigt das Ausgangssignal eine leichte Verzögerung beim vollständigen Nachfahren des Sollsignals. Diese Verzögerung, die aufgrund ihrer geringen Größe nur in den unteren beiden Grafiken erkennbar ist, könnte auf eine zu niedrig gewählte Integralkonstante des Reglers hinweisen. Um diesen Effekt maximal auszureizen, wurde die Änderungsrate des Signals so groß wie möglich gewählt, kombiniert mit einer minimalen Länge der dynamischen Bereiche, einem Übergang von 0 auf 1 und der Verwendung der Bausteinfunktion mit der höchsten Änderungsrate. Die Kostenfunktion besteht hauptsächlich aus den kleinen Fehlern, die in den dynamischen Bereichen auftreten, wie in der untersten Grafik von Abbildung 5.8 dargestellt.

Zusammenfassend lässt sich feststellen, dass in diesem Regelkreis keine signifikante Schwachstelle identifiziert werden konnte. Die Unterscheidung zwischen Soll- und Ausgangssignal ist in der oberen Grafik von Abbildung 5.8 aufgrund der geringen Fehlergröße kaum erkennbar. Dennoch konnte das generierte Worst-Case Signal durch seine schnellen dynamischen Umschwünge, einen hohen Kostenfunktionswert erzeugen, zumindest im Vergleich zum Referenzalgorithmus.

5.2.5. Pneumatisches Ventil mit Massenstrom als Eingangsgröße

In diesem Abschnitt wird das pneumatische Ventil mit der Eingangsgröße Massenstrom (PVEM) analysiert. Trotz der erhöhten Komplexität von PVEM im Vergleich zu PVED wird ein einfacher PI-Regler in der Regelstrecke verwendet. Wie in Abbildung 5.9 und Tabelle A.4 zu erkennen ist, zeigen sich die Unterschiede zwischen den Optimierern in diesem Fall wieder deutlicher als bei PVED.

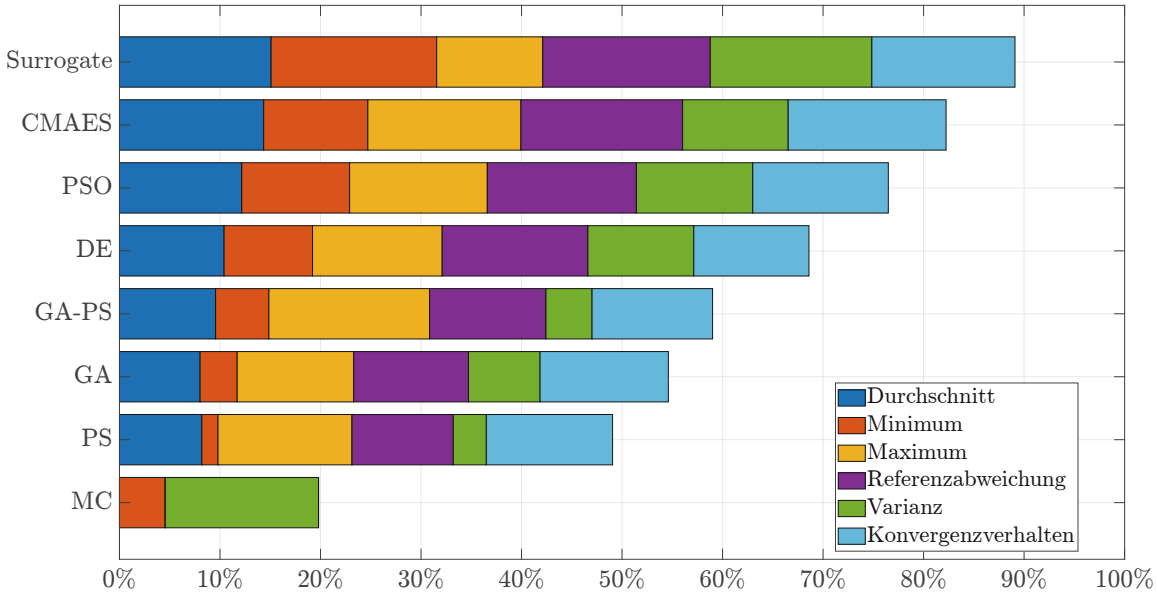


Abbildung 5.9.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell PVEM. Die Bewertung ist in Prozent angeben, und stellt sich aus den Bewertungen der 6 Kriterien zusammen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die erzielten Prozentwerte liegen in $\boldsymbol{P}_{mean}$ jedoch höher als im Fall von FMSN, und auch in $\boldsymbol{P}_{ref}$ erreichen die Algorithmen eine hohe Prozentzahl, wobei jeder Optimierer mehr als 68% erreicht, mit Ausnahme von PS, das einen Wert von 60% erzielt. Diese Beobachtung führt zu der Schlussfolgerung, dass es in diesem Modell leicht war Signale zu finden, welche besser als die des Referenzalgorithmus MC bewertet wurden. Allerdings stellte sich die Identifikation von Worst-Case Signalen als eine deutlich anspruchsvollere Aufgabe heraus.

Der Surrogate-Algorithmus erreicht mit einer Gesamtbewertung von $89,10\,\%$ den ersten Platz. In keinem Einzelszenario und in keiner der betrachteten Szenarien war er schlechter als MC, wodurch er im Referenzverhalten-Kriterium die maximale Bewertung von $100\,\%$ erzielt. Ähnlich wie andere Modelle zeichnet sich Surrogate durch eine sehr geringe Streuung der Ergebnisse aus, kombiniert mit einer guten absoluten Genauigkeit. Allerdings konvergiert er langsamer als die übrigen Optimierungsverfahren und erreicht keine besonders hohen Maximalwerte. CMAES weist mit einer Gesamtbewertung von $82,24\,\%$ ein entgegengesetztes Bewertungsprofil auf. Es überzeugt durch hohe absolute Genauigkeit und ein schnelles Konvergenzverhalten, zeigt jedoch eine vergleichsweise hohe Streuung der Ergebnisse. PSO mit $76,49\,\%$ und DE mit $68,6\,\%$ zeigen fast dasselbe Bewertungsprofil. Beide erzielen durchgehend solide Ergebnisse, wobei sie im Referenzverhalten-Kriterium besonders gut abschneiden. GA-PS mit $59,01\,\%$, GA mit $54,61\,\%$ sowie PS mit $49,06\,\%$ erreichen vergleichsweise niedrige Bewertungen. Diese Optimierer zeichnen sich erneut durch eine niedrige Wiederholgenauigkeit aus, insbesondere PS. Darüber hinaus weisen sie lediglich eine durchschnittliche bis niedrige absolute Genauigkeit auf.

Eine Simulation des Systems PVEM ist in Abbildung 5.10 dargestellt. In der obersten Grafik sind das Worst-Case Sollsignal y^d sowie das Ausgangssignal y_{sys} des Systems PVEM abgebildet. Das Sollsignal wurde mit dem optimierungsbasierten Signalgenerator erzeugt, unter Verwendung des Optimierers CMAES, der Parametermenge Ξ_6 und des Fehlermaßes d_{mae}. Zu Beginn der Simulation, in den dynamischen Bereichen, hat y_{sys} Schwierigkeiten y^d zu folgen, was auf die Kontaktkraft zurückzuführen ist. In der zweiten Grafik, die den Massenstrom als Eingangssignal des Systems darstellt, wird ersichtlich, dass der Regler nur langsam und kontinuierlich auf die durch die Kontaktkraft veränderten Bedingungen reagiert. Dies steht im Gegensatz zu den Reaktionen im System PVED, bei dem der Regler durch direkten Zugriff auf den Eingangsdruck unstetige, schnelle Anpassungen zeigen konnte. Neben den Fehler in den dynamischen Bereichen können wir beobachten, dass nach einem nicht differenzierbaren Arbeitspunktwechsel Grenzzyklen auftreten, die durch die Haftreibung verursacht werden. Diese sind in der dritten Grafik von Abbildung 5.10 deutlich zu erkennen. Die Haftreibung verhindert, dass der Regler die entstehenden Schwankungen vollständig kompensieren kann, was zu einer periodischen Abweichung vom Sollsignal führt. In der letzten Grafik, welche das Anwachsen der Kostenfunktion darstellt, wird deutlich, dass sowohl die Fehler in den dynamischen Bereichen als auch die durch die Grenzzyklen bedingten Fehler in ähnlicher Weise zur Kostenfunktion beitragen.

Damit konnte ein Worst-Case Signal identifiziert werden, das zwei unterschiedliche Schwachstellen des Reglers aufzeigt. Zum einen hat das System PVEM Schwierigkeiten, die Kontaktkraft an den Randpositionen effektiv zu bewältigen. Zum anderen können wir beobachten, dass ein nicht-differenzierbarer Arbeitspunktwechsel dazu führt, dass das System in Grenzzyklen verfällt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

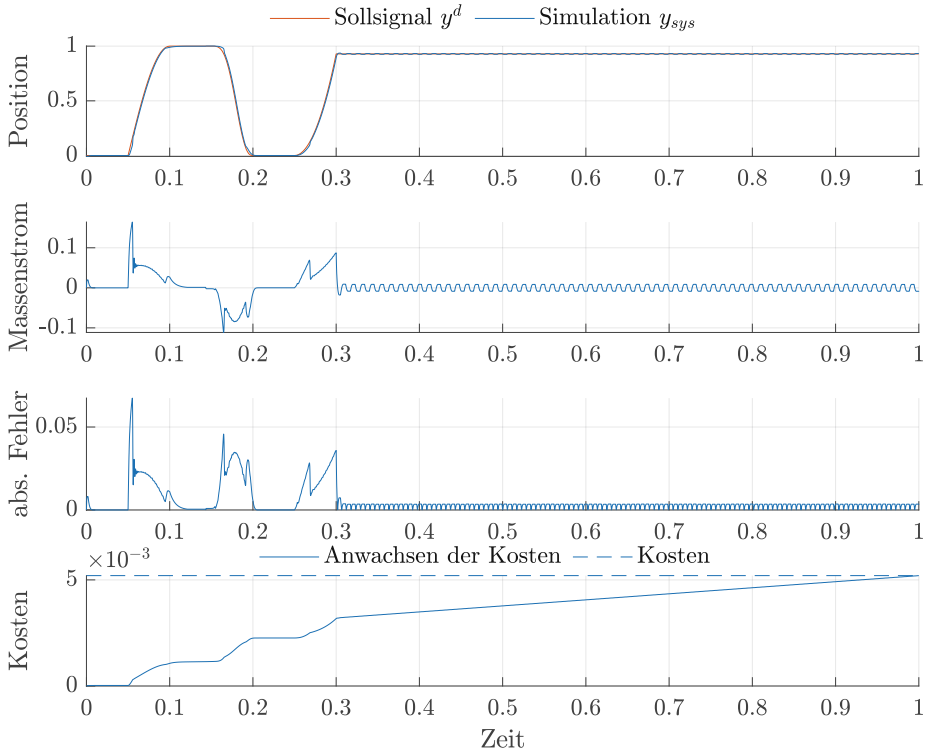


Abbildung 5.10.: Simulation eines Worst-Case Signals für PVEM, das durch die Optimierung mittels CMAES unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes MAE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von PVEM, als auch die Eingangsgröße Massenstrom. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der vierten Grafik sind die Kosten zusehen.

5.2.6. Fazit zur Auswahl eines Optimierungsalgorithmus

Nach der detaillierten Untersuchung aller Regelkreise erfolgt sowohl eine abschließende Beurteilung der untersuchten Optimierungsalgorithmen, als auch eine Interpretation der durchgeführten Simulationsstudien, siehe Tabelle 5.3 und Abbildung 5.11.
Es lässt sich feststellen, dass kein einzelner Optimierer als universell beste Wahl identifiziert werden konnte. Während der Surrogate-Algorithmus durch seine hohe Wiederholgenauigkeit überzeugt, zeigt der CMAES Algorithmus die besten Ergebnisse in Bezug auf die absolute Genauigkeit. Darüber hinaus wurde deutlich, dass die Leistungsfähigkeit der Optimierer stark vom betrachteten Modell abhängt. PSO erzielt die besten Ergebnisse bei Modellen, in denen das Auffinden eines Worst-Case Signals keine größeren Schwierigkeiten bereitet. Im Gegensatz dazu zeigt der Surrogate Algorithmus seine Stärken besonders bei komplexeren Systemen. Auf Basis der durchgeführten Analysen lassen sich folgende Emp-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

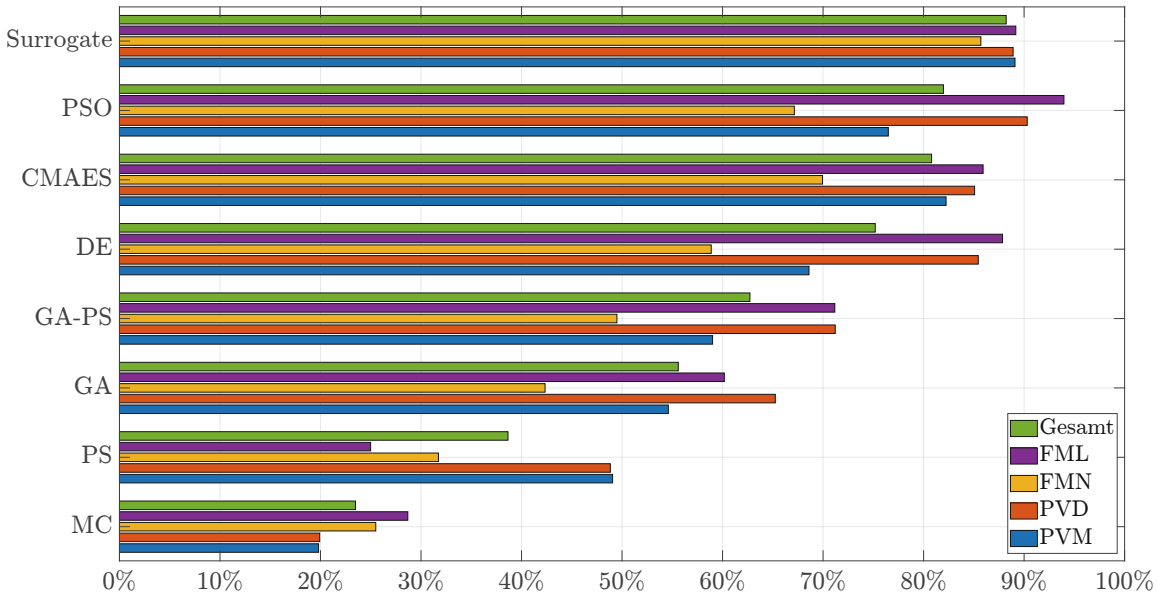


Abbildung 5.11.: Visualisierung von $\boldsymbol{P}_{all}$ mit allen Modellen. Die Bewertung ist in Prozent angeben.

fehlungen für den Einsatz globaler Optimierungsverfahren zur Ermittlung von Worst-Case Testsignalen ableiten. In hochkomplexen Systemen mit langen Simulationszeiten empfiehlt sich die Verwendung des Surrogate Algorithmus, da die hohe Wiederholgenauigkeit sicherstellt, dass bereits eine einzelne Optimierung zuverlässige Ergebnisse liefert. Damit eignet sich Surrogate für die Zielsetzung der Arbeit am besten. Für Systeme mit kürzeren Rechenzeiten bietet sich der Einsatz von CMAES an, da durch mehrere Optimierungsdurchläufe eine höhere Wahrscheinlichkeit besteht, möglichst nahe am globalen Optimum zu liegen. Bei einfachen Systemen ist PSO eine geeignete Wahl.

Die durchgeführten Simulationsstudien offenbarten eine Schwachstelle. In einigen Einzelszenarien erzielte der Referenzalgorithmus bessere Kostenfunktionen als die Optimierer. Dieses Verhalten wurde bei allen Algorithmen, mit Ausnahme des Surrogate Algorithmus, beobachtet und war hauptsächlich auf stochastische Faktoren zurückzuführen. Die hohe

Kategorie	Surrogate	PSO	CMAES	DE	GA-PS	GA	PS	MC
Absolute Genauigkeit	+	+	++	+	+ ∼	+	∼	−−
Wiederholgenauigkeit	++	+	∼	+	− ∼	− ∼	−	++
Konvergenzverhalten	+ ∼	+	+	+ ∼	+ ∼	∼	∼	−

Tabelle 5.3.: Vergleich der Algorithmen hinsichtlich der in Abschnitt 3.2 betrachteten Kategorien. Dabei ist die Reihenfolge der Bewertungen folgende, ++, +, + ∼, ∼, − ∼, −, −−.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Streuung der Ergebnisse verhinderte in diesen Fällen die Identifikation von Worst-Case Sollsignalen.

Um dieses Problem zu beheben und gleichzeitig die Konvergenz der Optimierer zu beschleunigen, wodurch der Rechenaufwand reduziert werden kann, wurden geeignete Startwerte gefunden, siehe Abschnitt 3.3. Im folgenden Abschnitt wird gezeigt, dass die Verwendung dieser Startwerte zu einer signifikanten Verbesserung der Schwachstelle führt.

5.3. Simulationsergebnisse zu den gewählten Startwerten

In diesem Abschnitt wird die Bewertung von Experiment 6, siehe Abschnitt 3.3, behandelt. Mit diesem Experiment wollen wir den Einfluss der von uns gewählten Startwerte untersuchen. Die Algorithmen werden anhand des Bewertungssystems analysiert.

Für jedes Szenario $s \in \mathcal{S}_6$ wurden spezifische Startwerte verwendet, die durch Clustering der Ergebnisse anderer Modelle mit derselben Parametermenge erzeugt wurden. Die Startwerte sind definiert durch die Menge

$$\Xi_{0,i}^{s} = \bigcup_{\Sigma} S_0\left(D_i^{\Sigma}\right), \tag{5.15}$$

für alle betrachteten Σ, die nicht in s enthalten sind. Die genaue Auswahl der Startwerte wurde ausführlich in Abschnitt 3.3 beschrieben. Zur klaren Unterscheidung wird der Index 0 verwendet, um anzugeben, dass ein Optimierer Startwerte nutzt. Die folgenden Optimierer konnten die bereitgestellten Startwerte

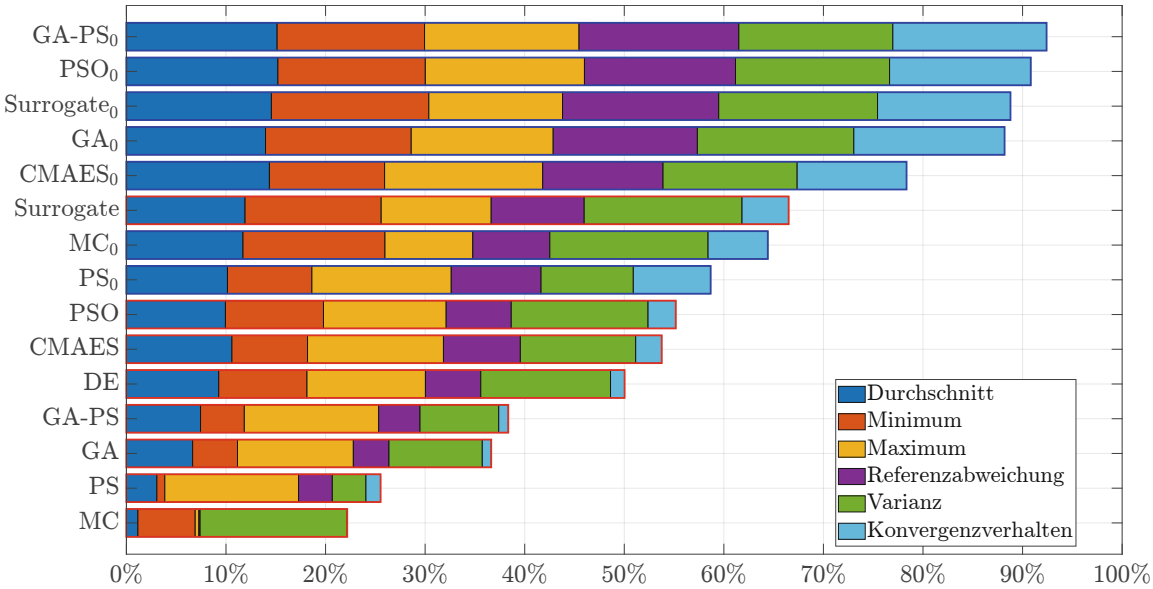


Abbildung 5.12.: Visualisierung der Ergebnisse des Bewertungssystems mit den Daten von Experiment 6. Dabei wird $\boldsymbol{P}_{all}$ dargestellt mit den 6 Kriterien über welche der Mittelwert gebildet wurde. Blaue Umrandungen repräsentieren Algorithmen mit Startwerten, rote ohne Startwerte.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{ref}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
GA-PS$_0$	92,41 %	90,82 %	88,84 %	93,08 %	96,29 %	92,81 %	92,62 %
PSO$_0$	90,82 %	91,30 %	88,81 %	95,89 %	91,02 %	92,84 %	85,05 %
Surrogate$_0$	88,79 %	87,36 %	94,88 %	80,55 %	94,12 %	95,69 %	80,12 %
GA$_0$	88,19 %	83,83 %	87,78 %	85,54 %	86,89 %	94,28 %	90,81 %
CMAES$_0$	78,35 %	86,17 %	69,41 %	95,28 %	72,50 %	80,82 %	65,91 %
Surrogate	66,51 %	71,48 %	82,05 %	66,28 %	56,04 %	95,11 %	28,08 %
MC$_0$	64,41 %	70,25 %	85,54 %	52,94 %	46,41 %	95,35 %	35,98 %
PS$_0$	58,67 %	60,89 %	50,89 %	83,96 %	54,13 %	55,58 %	46,60 %
PSO	55,17 %	59,66 %	59,08 %	73,93 %	39,22 %	82,42 %	16,68 %
CMAES	53,76 %	63,58 %	45,56 %	81,90 %	46,33 %	69,50 %	15,66 %
DE	50,03 %	55,63 %	53,18 %	71,36 %	33,37 %	78,17 %	8,48 %
GA-PS	38,35 %	44,67 %	26,39 %	80,96 %	24,90 %	47,45 %	5,75 %
GA	36,64 %	39,91 %	27,03 %	69,80 %	21,48 %	56,22 %	5,39 %
PS	25,52 %	18,36 %	4,78 %	80,59 %	20,34 %	20,27 %	8,81 %
MC	22,18 %	6,93 %	34,52 %	2,21 %	0,75 %	88,65 %	0,00 %

Tabelle 5.4.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 6. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

vollständig nutzen GA-PS$_0$, GA$_0$, Surrogate$_0$, PSO$_0$ und MC$_0$. PS$_0$ und CMAES$_0$ konnten jeweils nur einen der neun Startwerte verwenden. Diese Auswahl erfolgte abhängig von der jeweiligen Wiederholung, wobei jeweils ein einzelner Startwert aus $\Xi^s_{0,i}$ ausgewählt wurde. Der Referenzalgorithmus, der für die Bewertungskriterien herangezogen wurde, ist MC$_0$.

Die Ergebnisse der Bewertung sind in Tabelle 5.4 und Abbildung 5.12 dargestellt. Der Algorithmus GA-PS$_0$ erreicht eine Gesamtbewertung von 92,41 % und belegt damit den ersten Platz. Der hybride Optimierer erzielt in den meisten Kriterien eine Bewertung von über 90 %, mit Ausnahme des Kriteriums Minimum, in dem er 88,84 % erreicht. Das Konvergenzverhalten zeigt, dass der Optimierer schnell bessere Lösungen als die initialen Startwerte gefunden hat. Insgesamt konnte sich GA-PS$_0$ um mehr 50 % durch die Startwerte verbessern. Insbesondere in den Kriterien, welche die Wiederholgenauigkeit bewerten, also Varianz und Minimum, zeigt der Algorithmus mit Startwerten eine deutliche Verbesserung im Vergleich zur Leistung ohne Startwerte.

Der Algorithmus PSO$_0$ erreicht eine Gesamtbewertung von 90,82 % und belegt damit den zweiten Platz. Das Bewertungsprofil von PSO$_0$ ähnelt den bisherigen Bewertungen, zeigt jedoch insgesamt höhere Prozentwerte. PSO$_0$ erzielt die höchste Bewertung im Kriterium Maximum mit 95,89 %. Der Algorithmus PSO$_0$ zeigt eine deutlich bessere Leistung im Vergleich zu PSO ohne Startwerte, mit einem Unterschied von mehr als 30 %.

Der Algorithmus Surrogate$_0$ erreicht den dritten Platz mit einer Gesamtbewertung von 88,79 %. Dabei überzeugt Surrogate$_0$ durch eine hohe Wiederholgenauigkeit, erkennbar an der Varianzbewertung von 95,69 % sowie einer Bewertung von 94,88 % im Kriterium

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Minimum. Beide Werte stellen die höchsten Bewertungen in den jeweiligen Kriterien dar. Im Vergleich zu Surrogate ohne Startwerte ist die absolute Genauigkeit um einiges besser.

GA_0 liegt knapp hinter $Surrogate_0$ mit $88,19\,\%$. GA_0 mit den gewählten Startwerten zeigt ein anderes Verhalten als GA ohne Startwerte. Im Gegensatz zu GA ist die Streuung der Ergebnisse bei GA_0 sehr gering, mit einer Varianz von $94,28\,\%$ und einer Bewertung in $\boldsymbol{P}_{min}$ von $87,78\,\%$. GA_0 kann sich damit durch die Startwerte deutlich verbessern, der Algorithmus erreicht $60\,\%$ mehr als GA ohne Startwerte. In der absoluten Genauigkeit ist GA_0 mit $83,83\,\%$ im Durchschnitt am schlechtesten von den Optimierern, welche 9 Startwerte bekommen haben.

$CMAES_0$ erreicht $78,35\,\%$ und belegt damit den fünften Platz. Es ist jedoch zu berücksichtigen, dass $CMAES_0$ nur einen Startwert erhalten hat, da es in der verwendeten Implementierung nicht möglich war, mehrere Startwerte zu spezifizieren. Theoretisch ist dies jedoch möglich. Bemerkenswert ist in diesem Zusammenhang, dass $CMAES_0$ einen Maximalwert von $95,28\,\%$ erzielt hat. Insgesamt ist die Leistung von $CMAES_0$ unerwartet hoch, obwohl nur ein Startwert verwendet wurde. Das Profil von $CMAES_0$ ist stimmig mit den Beobachtungen aus anderen Experimenten. Der Algorithmus weist eine hohe absolute Genauigkeit auf, zeigt jedoch eine mittelmäßige Wiederholgenauigkeit. Die Werte von $\boldsymbol{P}_{ref}$ und $\boldsymbol{P}_{kvh}$ fallen in diesem Fall vergleichsweise niedrig aus, da diese sich auf den Bezugsalgorithmus MC_0 beziehen. Im Gegensatz zu $CMAES_0$ wurde MC_0 mit neun Startwerten getestet. Nichtsdestotrotz zeigt bereits die Verwendung eines einzelnen Startwertes eine signifikante Verbesserung von $CMAES_0$ gegenüber CMAES von etwa $25\,\%$.

PS_0, welches ebenfalls nur einen Startwert erhalten hat, belegt den achten Platz mit $58,67\,\%$. Der Algorithmus PS kann, im Gegensatz zu CMAES, auch theoretisch nur einen Startwert verwenden. Wie in den anderen Experimenten zeigt PS_0 eine hohe Bewertung in $\boldsymbol{P}_{max}$, schneidet jedoch in den anderen Kriterien mit den Startwerten nur mittelmäßig ab. Im Vergleich zu PS ohne Startwerte zeigt sich jedoch eine deutliche Verbesserung.

Der Referenzalgorithmus MC_0 erzielt eine Bewertung von $64,41\,\%$. Die hohe Bewertung in $\boldsymbol{P}_{var}$ von $95,69\,\%$ deutet darauf hin, dass MC_0 nahezu keine Verbesserungen gegenüber den neun initialen Startwerten erreicht hat. Die niedrige Bewertung im Konvergenzkriterium ist darauf zurückzuführen, dass MC_0 keine Ergebnisse erzielt, die besser als die Startwerte sind, und folglich in der Regel auch nicht besser als der Durchschnitt der eigenen Endwerte abschneidet.

Die Analyse zeigt, dass die Startwerte in diesem Experiment eine entscheidende Rolle für den Erfolg der Optimierungsalgorithmen gespielt haben. Die besten Ergebnisse wurden ausschließlich von Algorithmen erzielt, die Zugriff auf alle neun Startwerte hatten. Dies unterstreicht die hohe Qualität der Startwerte. Algorithmen, die keine Startwerte nutzten, zeigten deutliche Schwächen, insbesondere bei der Bewertung $\boldsymbol{P}_{ref}$ und $\boldsymbol{P}_{kvh}$. Diese Defizite sind darauf zurückzuführen, dass der Referenzalgorithmus MC_0, der alle neun Startwerte verwendete, durchgehend hohe Kostenfunktionswerte erzielte. Die Leistungsfähigkeit der Startwerte wird auch durch die Ergebnisse von MC_0 verdeutlicht. Lediglich Surrogat-Methoden ohne Startwerte erzielten höhere Prozentwerte. Dies zeigt, dass die Startwerte allein sehr hohe Kostenfunktionswerte verursachten, da MC_0 diese kaum verbessert.

Gleichzeitig zeigen die hohen Werte in $\boldsymbol{P}_{kvh}$ der Algorithmen mit Startwerten, dass diese in der Lage waren, schnell Verbesserungen gegenüber den Startwerten zu finden, auch wenn die erreichten Werte im Durchschnitt nicht signifikant höher waren. Dies wird daran

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

deutlich, dass MC_0 im Durchschnitts-Kriterium eine Bewertung von 70 % erreicht, was deutlich näher an den etwa 90 % der besten Algorithmen liegt. Zum Vergleich erreicht MC_0 in Abschnitt 5.2 lediglich 23,48 %, was in manchen Fällen einen Unterschied von über 60 % zu den besten Algorithmen darstellt. Die durchwegs hohen Maximalbewertungswerte der Optimierer ohne Startwerte deuten darauf hin, dass keine neuen Schwachstellen der Optimierer aufgrund der Startwerte gefunden wurden.

Außerdem zeigt Tabelle A.5, in der MC ohne Startwerte als Referenzalgorithmus verwendet wurde, dass nahezu alle Optimierer, die Startwerte nutzen konnten, im Referenzverhalten-Kriterium 99 % erreichten. Daraus lässt sich schließen, dass diese Optimierer in jeder Wiederholung der Optimierung zuverlässig ein Worst-Case Testsignal gefunden haben.

Es lässt sich zusammenfassen, dass das Experiment äußerst erfolgreich zeigt, dass die Wahl der Startwerte durch die Clustering-Methode einen signifikanten positiven Einfluss auf die Leistung der Optimierer hat. Insbesondere im Hinblick auf die Wiederholgenauigkeit kann durch die Verwendung dieser Startwerte eine erhebliche Verbesserung erzielt werden. Dadurch werden die im vorherigen Abschnitt identifizierte Schwachstelle des optimierungsbasierten Signalgenerators erfolgreich behoben. Basierend auf den Ergebnissen der Simulationsstudien mit diesen Startwerten können wir nun die Aussage treffen, dass der optimierungsbasierte Signalgenerator Schwachstellen in nichtlinearen Regelkreisen innerhalb eines einzigen Optimierungsdurchlaufs zuverlässig identifizieren kann.

Im nächsten Abschnitt wollen wir diese Aussage untermauern, indem wir ein praktisches Experiment an einem Regelkreis durchführen.

5.4. Bewertung am Prüfstandsaufbau

Im abschließenden Abschnitt dieses Kapitels wird Experiment 1 am Prüfstand durchgeführt. Ziel ist es zu überprüfen, ob der optimierungsbasierte Signalgenerator in der Lage ist, Schwachstellen in der Regelung des Prüfstandes zu identifizieren.

Da wir nur einen Optimierungsdurchgang verwenden werden, wollen wir bei der Auswahl des Optimierers auf eine möglichst hohe Wiederholgenauigkeit achten. Da in den vorherigen Experimenten Surrogate stets die höchste Wiederholgenauigkeit gezeigt hat, werden wir diesen Optimierer für den Prüfstandsversuch einsetzen. Wir führen die Optimierung auf dem Prüfstand mit der Parametermenge Ξ_4 und dem Fehlermaß MAE durch. Um einen Vergleich möglich zu machen, führten wir den Test auch mit MC durch. Die Ergebnisse der Optimierung sind in Abbildung 5.13 und Abbildung 5.14 dargestellt. In diesem Fall wurden die Eingänge des Reglers nicht abgebildet, da in Kapitel 4 nicht auf die Stellgrößen des Reglers eingegangen wurde.

In Abbildung 5.13 ist eine Messung des pneumatischen Ventils dargestellt. Im oberen Diagramm sind das Sollsignal y^d sowie das Ausgangssignal y_{sys} abgebildet. Am rechten Ende des Plots sowie im gesamten ersten stationären Abschnitt des Signals ist erkennbar, dass das System Schwierigkeiten hat, die normierte Position 1 zu erreichen. Dieser Effekt ist auf das fehlende Anschlagsmodell im Vorsteuermodell des Reglers zurückzuführen und führt bereits zu einem signifikanten Fehler. Das ist in den unteren Diagrammen der Abbildung 5.13 ersichtlich. Der schnelle Übergang des Sollsignals von 1 auf 0 bewirkt ein längeres Verharren des Systems.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

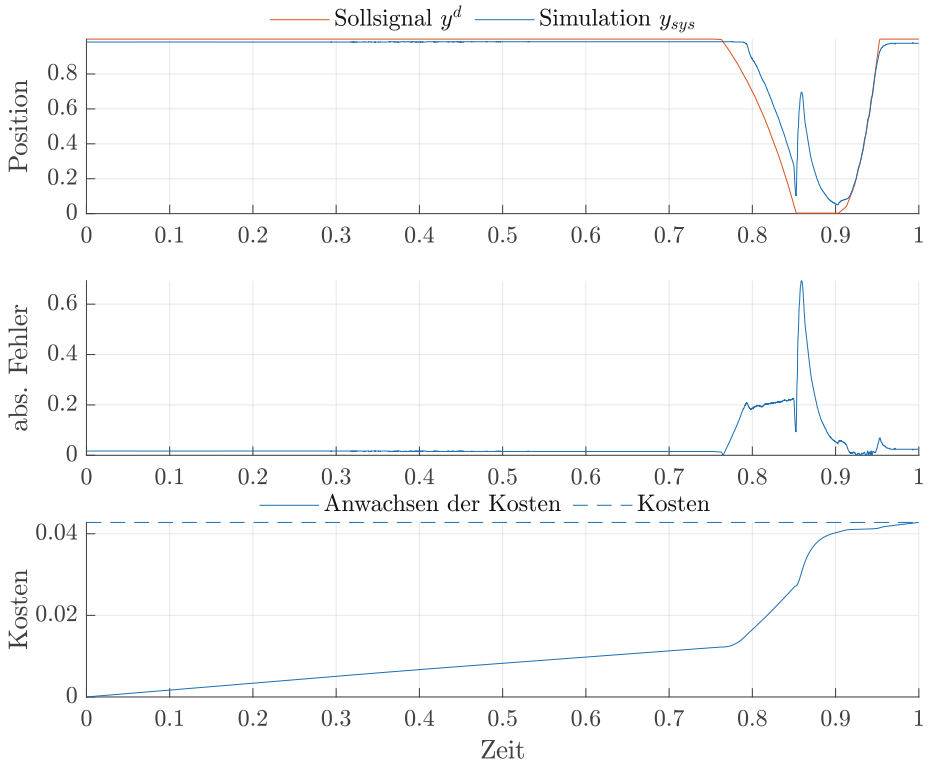


Abbildung 5.13.: Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist des Ergebnisses des optimierungsbasierten Signalgenerator mit dem Optimierer Surrogate, der Parametermenge Ξ_4 und des Fehlermaßes MAE. Im obersten Verlauf ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen.

Dies ist vermutlich darauf zurückzuführen, dass der I-Anteil des Reglers, bedingt durch das fehlende Anschlagsmodell, zuvor stark angewachsen ist und vor einer erneuten Systemanpassung abgebaut werden muss. Anschließend führt ein Fehler in der Bestimmung der Bremswirkung zu einem Überschwingen in Richtung der oberen Position, was durch den nicht differenzierbaren Arbeitspunktwechsel zusätzlich verstärkt werden könnte. In den beiden unteren Diagrammen der Abbildung 5.13, welche den absoluten Fehler sowie die Kosten darstellen, wird deutlich, dass der größte Anteil der Kosten durch den vermuteten Bremsfehler verursacht wird.

Das beste Ergebnis aus 100 zufälligen Versuchen ist in Abbildung 5.14 dargestellt. Auch hier zeigt sich in der oberen Grafik das gleiche Fehlverhalten wie im vorherigen Beispiel, das auf das fehlende Anschlagsmodell in der Vorsteuerung zurückzuführen ist. Nach dem kurzen Verharren, das hier den größten Fehler verursacht hat, siehe beide untere Grafiken

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

von Abbildung 5.14, ist der Regelkreis jedoch in der Lage, dem Sollsignal sehr präzise zu folgen.

Als weiteren Versuch haben wir die Startwerte $\Xi_{0,4}$ gemäß Gleichung (3.21) überprüft und diese auf dem Prüfstand getestet. Die Ergebnisse eines ausgewählten Signals sind in Abbildung A.1 dargestellt. Es ist erkennbar, dass der Bremsfehler bereits in den Startwerten zu sehen ist, was erneut die Qualität der gewählten Startwerte unterstreicht.

Zusammenfassend kann festgestellt werden, dass die Bewertung am Prüfstand äußerst erfolgreich war. Es konnte nachgewiesen werden, dass der Signalgenerator in der Lage ist, ein Worst-Case Signal zu erzeugen, die gezielt Schwächen des Regelkreises aufdeckt.

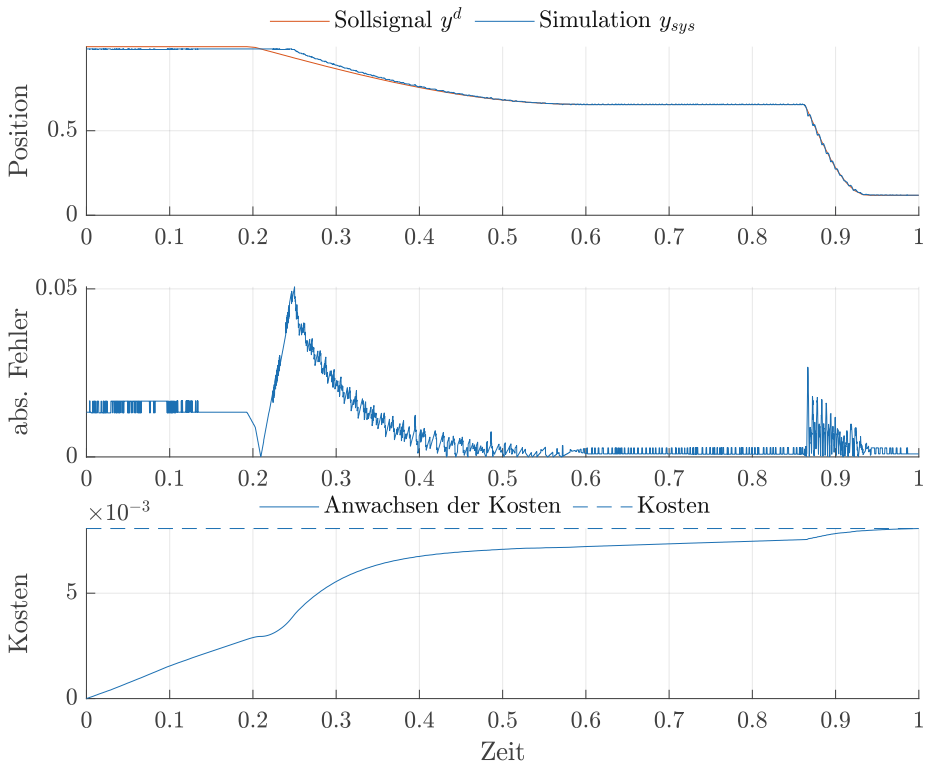


Abbildung 5.14.: Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist des Ergebnisses des optimierungsbasierten Signalgenerator mit dem Optimierer MC, der Parametermenge Ξ_4 und des Fehlermaßes MAE. Es soll als Vergleich zu Abbildung 5.13 dienen. Im obersten Verlauf ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

6. Zusammenfassung und Ausblick

In dieser Arbeit wurde ein optimierungsbasierter Signalgenerator entwickelt und angewendet, um Worst-Case Testsignale für geschlossene Regelkreise zu bestimmen. Zunächst wurden die theoretischen Grundlagen dynamischer Systeme sowie die Struktur von Regelkreisen erläutert. Anschließend erfolgte eine Einführung in die verwendeten Optimierungsalgorithmen. Im Rahmen der Entwicklung des Signalgenerators wurde ein deterministischer Signalgenerator entworfen, der als Grundlage für die Formulierung eines Optimierungsproblems diente. Zur Initialisierung der Optimierung wurden geeignete Startwerte auf Basis der durchgeführten Simulationstudien, mithilfe einer Clustering-Methode, bestimmt.

Zur Überprüfung der Funktionsfähigkeit des optimierungsbasierten Signalgenerators wurden umfassende Simulationsstudien konzipiert. Anhand von vier verschiedenen Regelkreisen wurde untersucht welches der betrachteten globalen Optimierungsverfahren die gestellte Aufgabe am zuverlässigsten löst. Dabei hatte jeder Optimierungsalgorithmus ein Budget von 100 Kostenfunktionsauswertungen. Um die Auswirkungen stochastischer Einflussfaktoren zu berücksichtigen, wurden die Optimierungen jeweils 50-mal für jeden Algorithmus ausgeführt. Eine ergänzende Simulationsstudie untersucht, ob die Testsignale aus anderen Regelkreisen als geeignete Startwerte dienen und letztlich zu einer schnelleren Konvergenz der Optimierung beitragen. Schließlich erfolgte eine experimentelle Validierung des entwickelten Signalgenerators an einem Prüfstand, um dessen Leistungsfähigkeit unter realen Bedingungen zu überprüfen.

In den Simulationsstudien wurde kein universell einsetzbarer Optimierungsalgorithmus gefunden. Einige Algorithmen zeigten vielversprechende Ergebnisse, wobei deren Leistungsfähigkeit jedoch stark vom betrachteten Regelkreis abhängig war. Der Surrogate Algorithmus erzielte die besten Ergebnisse im Hinblick auf das zu Beginn der Arbeit formulierte Ziel. Dank seiner hohen Wiederholgenauigkeit eignet sich dieser Algorithmus besonders für den Einsatz an Regelkreisen mit aufwändigen Kostenfunktionsauswertungen. Er konnte aufgrund der niedrigen Varianz zusammen einer guten absoluten Genauigkeit Worst-Case Signale zuverlässig bestimmen. Der CMAES Algorithmus liefert häufig Testsignale mit hohen Kostenfunktionswerten und weist prinzipiell eine hohe absolute Genauigkeit auf. Allerdings gelingt es ihm in einigen Fällen nicht, das gewünschte Ergebnis zu erzielen, was auf eine hohe Varianz und eine insgesamt moderate Wiederholgenauigkeit hinweist. Daher empfiehlt sich der Einsatz von CMAES insbesondere dann, wenn die Optimierung mehrfach durchgeführt werden kann.

Zudem wurde festgestellt, dass viele Optimierungsalgorithmen aufgrund der begrenzten Anzahl an Iterationen in manchen Wiederholungen schlechtere Ergebnisse erzielten als eine zufällige Generierung von Testsignalen. Dieses Verhalten konnte häufig auf unzureichende Startwerte zurückgeführt werden. Zur Lösung dieses Problems wurden Startwerte auf Grundlage bereits durchgeführter Simulationsstudien unter Anwendung einer Clustering-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Methode bestimmt. Die Verwendung dieser Startwerte zeigte einen äußerst positiven Einfluss auf die Performance der Optimierungsalgorithmen. Insbesondere führte sie zu einer signifikanten Verbesserung der Ergebnisse hinsichtlich der Wiederholgenauigkeit. Dadurch konnte in jeder Optimierungswiederholung ein Testsignal generiert werden, das einen hohen Kostenfunktionswert erreichte. Somit wurde die in vorherigen Simulationsstudien beobachtete Schwachstelle behoben.

Schließlich konnte am Prüfstand eines pneumatischen Ventils nachgewiesen werden, dass der optimierungsbasierte Signalgenerator erfolgreich funktioniert. Nach 100 Messvorgängen wurde eine Schwachstelle des betrachteten Regelkreises gefunden, wodurch die Effizienz und der Nutzen des entwickelten Ansatzes eindrucksvoll bestätigt wurden.

Im Verlauf dieser Arbeit eröffneten sich neue Fragestellungen, welche als Ausgangspunkt für weiterführende Forschungsarbeiten dienen können.

Obwohl im Rahmen dieser Arbeit die Surrogate Methode als sehr vielversprechender Ansatz zur optimierungsbasierten Generierung von Testsignalen für die betrachteten geschlossene Regelkreise gefunden wurde, verbleibt die Frage, ob diese Methode auch für andere Regelkreise zuverlässige Ergebnisse liefern kann. Da in den durchgeführten Experimenten deutliche Unterschiede auftauchten, lässt sich hier keine Vorhersage treffen. Deshalb wäre es von besonderem Interesse, wie sich diese Methode an anderen Regelkreisen verhält.

Es wurde demonstriert, dass sich die vorgestellte Methode eignet um ein Worst-Case Testsignal für einen geschlossenen Regelkreis zu finden. Damit kann das System im Hinblick auf das schlechtest mögliche Verhalten untersucht werden. Für eine ausführliche Bewertung geschlossener Regelkreise wäre es hilfreich über eine Schar an Testsignal zu verfügen. Diese könnten, ähnlich der Startwerte, aus vergangenen Experimenten stammen oder mit lokalen Optimierungsverfahren und variierenden Startwerten gefunden werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A. Anhang

A.1. Verwendung der Optimierer

Alle Optimierungen und Auswertungen wurden in MATLAB 2023b durchgeführt. Mit Ausnahme der Algorithmen MC, DE und CMAES stammen alle verwendeten Optimierer aus der Global Optimization Toolbox von MATLAB. Für diese 5 Algorithmen (GA, GA-PS, PS, PSO, Surrogate) kam zusätzlich die Parallel Computing Toolbox zum Einsatz, obwohl diese nicht notwendig ist. Zu diesen 5 Optimierern werden wir exemplarisch kurze Codeausschnitte angeben, die die gewählten Einstellungen veranschaulichen. Falls Einstellungen nicht erwähnt werden, gelten die Standardeinstellungen. Die Nebenbedingungen und Grenzen für den Parametervektor $\boldsymbol{\xi}$ wurden allen Optimierern übergeben.

Algorithm 1: Allgemeine Einstellungen

```
fun = @(x)objgeneratorfnc(x,NT,Costfunction,Sim,Ts);
len_V = NT + NB + NX;
A = [ones(1, NT), zeros(1, NX + NB)];
b = 0.95;
lb = [zeros(1, NT) + 0.05, ones(1, NB), zeros(1, NX)];
ub = [zeros(1, NT) + 0.95, ones(1, NB) * 7, ones(1, NX)];
```

Algorithmus 1 enthält sechs Zeilen, die von allen in MATLAB implementierten Algorithmen gemeinsam genutzt werden. Die erste Zeile definiert den `function handle` der Kostenfunktion, der auch eine Funktion enthält, um sicherzustellen, dass die Nebenbedingungen eingehalten werden, selbst wenn der Optimierer keine direkte Unterstützung für lineare Nebenbedingungen bietet. Der Parameter `NT` gibt die Größe T_n an, also die Länge des Vektors $\boldsymbol{T}$. `Costfunktion` repräsentiert die definierte Kostenfunktion, `Sim` beschreibt den gewählten Regelkreis Σ, und T_s entspricht der Schrittweite, die am Ende von Kapitel 3 beschrieben wurde. Die Variablen `A` und `b` implementieren die Nebenbedingungen als lineares Gleichungssystem. `lb` und `ub` definieren die unteren und oberen Grenzen für die Vektoren $\boldsymbol{T}$, $\boldsymbol{B}$ und $\boldsymbol{X}$. Wie in allen anderen Implementierungen werden die Einträge von $\boldsymbol{B}$ als `double` behandelt und in der Baustein-Funktion auf ganze Zahlen gerundet. Diese Vorgehensweise verbessert die Performance der Implementierung.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Algorithm 2: Einstellungen zu GA

```
options = optimoptions('ga', ...
    'MaxGenerations', 4, ...
    'UseParallel', true, ...
    'PopulationSize', 20, ...
    'CrossoverFraction', 0.8);
ga(fun, len_V, A, b, [], [], lb, ub, [], options);
```

In Algorithmus 2 sind die wichtigsten Aspekte der Implementierung von GA in MATLAB 2023b dargestellt. Die Population wurde auf 20 Individuen und die Anzahl der Generationen auf 4 festgelegt, da auch die 0. Generation berücksichtigt wird und die Anzahl der Funktionsaufrufe 100 nicht überschreiten sollte. Der Parameter `CrossoverFraction` wurde auf 0.8 gesetzt. `UseParallel` ermöglicht die parallele Ausführung der Optimierung, um die Laufzeit zu verkürzen. Für GA_0 wurde die Einstellung `InitialPopulationMatrix` verwendet, um die vorgegebenen Startwerte zu übergeben. Weitere Details zur Implementierung des GA sind auf der MathWorks-Website verfügbar[1].

Algorithm 3: Einstellungen zu PS

```
x0=randomvalues(T,B,X);
options = optimoptions("patternsearch", ...
    "Algorithm", "nups", ...
    "MaxFunctionEvaluations", 100, ...
    "UseParallel", true, ...
    'MaxTime',1000, ...
    'MaxIterations',1000 ...);
patternsearch(fun,x0,A,b,[],[],lb,ub,[],options);
```

In Algorithmus 3 sind die wichtigsten Aspekte der Implementierung des PS in MATLAB 2023b dargestellt. Die maximale Anzahl von Iterationen N_{iter} wurde über den Parameter `MaxFunctionEvaluations` festgelegt. Der Startwert `x0` wurde zufällig initialisiert, während für PS_0 ein Element aus $\Xi_{0,i}$ als Startwert verwendet wurde. Weitere Details zur Implementierung von PS sind auf der MathWorks-Website verfügbar[2].

[1] Mathworksseite zur Implementierung von GA: https://de.mathworks.com/help/gads/ga.html
[2] Mathworksseite zur Implementierung von PS: https://de.mathworks.com/help/gads/patternsearch.html

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithm 4: Einstellungen zu GA-PS

```
hybridopts = optimoptions("patternsearch", ...
    'MaxFunctionEvaluations', 20, ...
    'UseParallel', true, ...
    'Display', 'iter', ...
    'OutputFcn', @myfun);

options = optimoptions('ga', ...
    'MaxGenerations', 3, ...
    'UseParallel', true, ...
    'PopulationSize', 20, ...
    'CrossoverFraction', 0.8, ...
    'HybridFcn', {@patternsearch, hybridopts});

ga(fun, len_V, A, b, [], [], lb, ub, [], options);
```

In Algorithmus 4 sind die Einstellungen des Algorithmus GA-PS in MATLAB 2023b dargestellt. Für den hybriden Algorithmus wurden die gleichen Einstellungen wie für GA und PS übernommen. Die letzte Generation von GA wurde durch 20 Funktionsaufrufe von PS ersetzt, wodurch insgesamt 100 Funktionsaufrufe möglich sind. Weitere Details zu GA-PS sind auf der MathWorks-Website verfügbar[3].

Algorithm 5: Einstellungen zu PSO

```
options = optimoptions('particleswarm', ...
    'MaxIterations', 4, ...
    'UseParallel', true, ...
    'SwarmSize', 20, ...);

particleswarm(fun, len_V, lb, ub, options);
```

Auch in Algorithmus 5 sind die Einstellungen für PSO in MATLAB 2023b dargestellt. PSO in MATLAB bietet keine direkte Möglichkeit, lineare Nebenbedingungen zu berücksichtigen, wie man am Fehlenden `A` und `b` feststellen kann. Dies ist aber dank der angepassten Kostenfunktion kein Problem, siehe Beschreibung von Algorithmus 1. Die maximale Anzahl von Iterationen wurde über `MaxIterations` auf 4 festgelegt, während die `SwarmSize` auf 20 gesetzt wurde, da auch hier eine 0. Iteration berücksichtigt wird. `UseParallel` ermöglicht wieder die parallele Ausführung der Optimierung. Für PSO_0 wurde die Einstellung `InitialPoints` verwendet, um spezifische Startwerte zu übergeben. Weitere Details zu PSO sind auf der MathWorks-Website verfügbar[4].

[3] Mathworksseite zur Implementierung von GA-PS: https://de.mathworks.com/help/gads/using-a-hybrid-function.html

[4] Mathworksseite zur Implementierung von PSO: https://de.mathworks.com/help/gads/particleswarm.html

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithm 6: Einstellungen zu Surrogate

```
options = optimoptions('surrogateopt', ...
    'MaxFunctionEvaluations', 100, ...
    'UseParallel', true, ...
    'MinSurrogatePoints', 30);

surrogateopt(fun, lb, ub, [], A, b, [], [], options);
```

Die Einstellungen für den Surrogate Algorithmus sind in Algorithmus 6 dargestellt. Die maximale Anzahl von Iterationen N_{iter} wurde über den Parameter `MaxFunctionEvaluations` definiert. Für Surrogate$_0$ wurde die Einstellung `InitialPoints` verwendet, um spezifische Startwerte zu übergeben. Der Parameter `MinSurrogatePoints` wurde auf 30 gesetzt, wodurch zunächst 30 Punkte zur Erstellung des Surrogate Modells verwendet wurden, bevor der Optimierungsprozess begann. MATLAB nutzt standardmäßig radiale Basisfunktionen (RBF) als Surrogat-Modell. Weitere Details zur Implementierung des Surrogat-Algorithmus sind auf der MathWorks-Website verfügbar[5].

Der Monte-Carlo-Algorithmus wurde eigenständig implementiert, indem die Simulation N_{iter}-mal mit zufällig gewählten Werten für $\boldsymbol{\xi}$ durchgeführt wurde. Für MC$_0$ wurden einfach die Startwerte als die ersten $\boldsymbol{\xi}$ der Simulation gewählt.

Es wurde eine Implementierung von Markus Buehren für MATLAB verwendet [31]. Dabei wurden die Population auf 20 und die Anzahl der Generationen auf 4 festgelegt. Um sicherzustellen, dass $\boldsymbol{\xi}$ die gewünschten Nebenbedingungen erfüllt, wurde auch hier die angepasste Kostenfunktion aus Algorithmus 1 verwendet.

Der CMA-ES-Algorithmus wurde durch eine Implementierung von Nikolaus Hansen in MATLAB verwendet [32], `cmaVersion = '3.61.beta'`. Die Größe der Population wurde auf 10 festgelegt und die maximale Anzahl der Funktionsauswertungen über `MaxFunEvals` begrenzt. Da diese Implementierung keine Möglichkeit bot, die Summennebenbedingung (3.14e) direkt zu erfüllen, wurde dies wie auch bei PSO und DE über die Kostenfunktion gehandhabt. Als Startwert wurde ein zufälliger Wert gewählt, für CMAES$_0$ war dieser ein Element von $\Xi_{0,i}$. Die Standardabweichungen σ wurden für die Komponenten $\boldsymbol{T}$, $\boldsymbol{B}$ und $\boldsymbol{X}$ auf 0.3, 1.3 bzw. 0.3 festgelegt.

[5] Mathworksseite zur Implementierung von Surrogate: https://de.mathworks.com/help/gads/surrogateoptimization-algorithm.html

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A.2. Zusätzliche Tabelle und Grafiken zu den Experimenten

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{vgl}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
PSO	93,96 %	93,57 %	90,29 %	97,72 %	100,00 %	95,30 %	86,90 %
Surrogate	89,18 %	90,46 %	93,84 %	83,81 %	96,54 %	93,42 %	76,98 %
DE	87,85 %	84,98 %	77,56 %	93,86 %	98,33 %	91,42 %	80,97 %
CMAES	85,92 %	86,73 %	67,10 %	97,57 %	92,65 %	82,44 %	89,04 %
GA-PS	71,17 %	62,43 %	47,67 %	85,64 %	79,94 %	76,31 %	75,02 %
GA	60,18 %	48,32 %	35,34 %	69,61 %	64,79 %	70,46 %	72,54 %
MC	28,69 %	13,88 %	50,07 %	0,00 %	12,05 %	94,72 %	1,39 %
PS	24,99 %	13,69 %	1,59 %	95,16 %	14,00 %	0,00 %	25,49 %

Tabelle A.1.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell FML. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{vgl}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
Surrogate	85,70 %	91,08 %	100,00 %	62,10 %	90,98 %	81,93 %	88,09 %
CMAES	69,94 %	72,17 %	52,72 %	79,44 %	75,34 %	52,35 %	87,64 %
PSO	67,14 %	60,04 %	64,27 %	61,35 %	68,00 %	69,17 %	80,03 %
DE	58,87 %	52,03 %	61,01 %	55,97 %	56,13 %	72,10 %	56,00 %
GA-PS	49,49 %	51,98 %	22,43 %	69,77 %	56,20 %	31,63 %	64,93 %
GA	42,34 %	33,15 %	24,88 %	69,28 %	37,61 %	33,18 %	55,94 %
PS	31,73 %	24,77 %	11,67 %	77,19 %	21,28 %	11,07 %	44,42 %
MC	25,50 %	5,93 %	49,79 %	1,28 %	5,01 %	90,99 %	0,00 %

Tabelle A.2.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell FMN. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{vgl}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
PSO	90,31 %	93,56 %	80,01 %	91,27 %	100,00 %	91,09 %	85,95 %
Surrogate	88,91 %	91,68 %	94,66 %	83,04 %	96,50 %	98,74 %	68,82 %
DE	85,44 %	88,06 %	71,76 %	91,40 %	98,50 %	82,48 %	80,44 %
CMAES	85,07 %	87,53 %	61,90 %	95,74 %	96,50 %	74,80 %	93,98 %
GA-PS	71,21 %	62,01 %	38,56 %	82,49 %	86,50 %	62,12 %	95,58 %
GA	65,25 %	55,43 %	24,42 %	83,11 %	82,00 %	53,34 %	93,18 %
PS	48,84 %	56,72 %	5,50 %	99,63 %	63,00 %	1,92 %	66,25 %
MC	19,92 %	0,00 %	29,90 %	1,86 %	0,00 %	87,75 %	0,00 %

Tabelle A.3.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell PVD. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{vgl}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
Surrogate	89,10 %	90,51 %	98,85 %	63,31 %	100,00 %	96,44 %	85,46 %
CMAES	82,24 %	86,08 %	62,19 %	91,38 %	96,50 %	63,02 %	94,26 %
PSO	76,49 %	72,99 %	64,38 %	82,20 %	89,00 %	69,46 %	80,91 %
DE	68,60 %	62,45 %	52,79 %	77,39 %	87,00 %	63,20 %	68,76 %
GA-PS	59,01 %	57,38 %	31,83 %	95,90 %	69,50 %	27,49 %	71,96 %
GA	54,61 %	48,13 %	22,13 %	69,57 %	68,50 %	42,72 %	76,58 %
PS	49,06 %	49,17 %	9,66 %	79,94 %	60,50 %	19,73 %	75,36 %
MC	19,81 %	0,00 %	27,33 %	0,00 %	0,00 %	91,52 %	0,00 %

Tabelle A.4.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell PVM. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

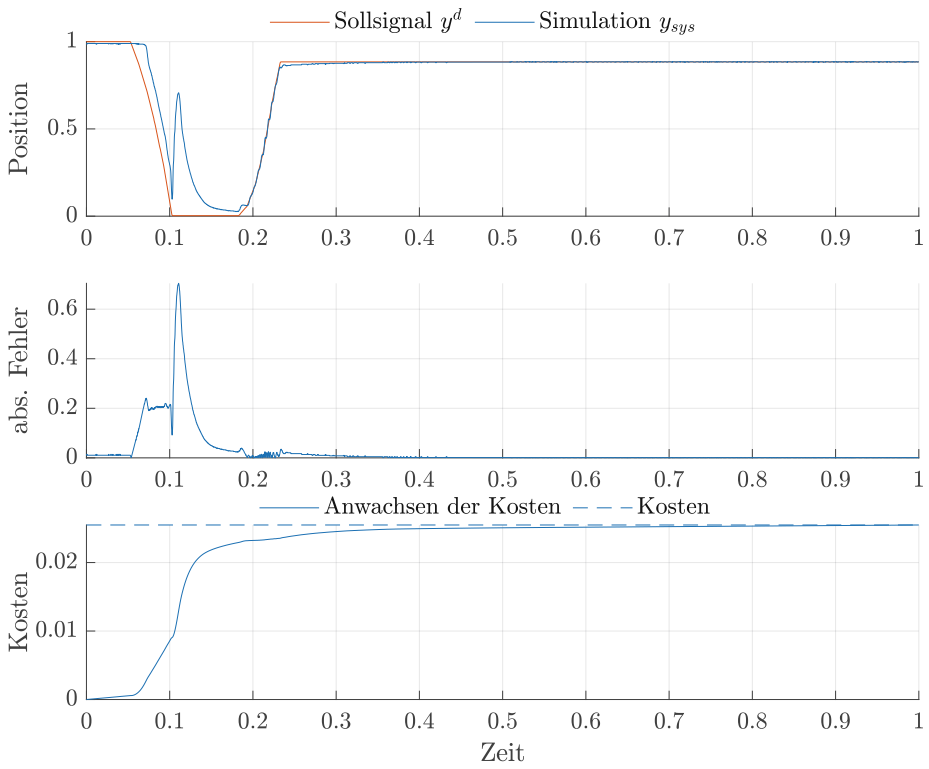


Abbildung A.1.: Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist ein Element aus $\Xi_{0,4}$. In der obersten Abbildung ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\boldsymbol{P}$	$\boldsymbol{P}_{all}$	$\boldsymbol{P}_{mean}$	$\boldsymbol{P}_{min}$	$\boldsymbol{P}_{max}$	$\boldsymbol{P}_{rab}$	$\boldsymbol{P}_{var}$	$\boldsymbol{P}_{kvh}$
PSO_0	94,66 %	91,30 %	88,81 %	95,89 %	99,78 %	92,84 %	99,31 %
$GA\text{-}PS_0$	94,03 %	90,82 %	88,84 %	93,08 %	99,35 %	92,81 %	99,31 %
GA_0	91,61 %	83,83 %	87,78 %	85,54 %	99,57 %	94,28 %	98,65 %
$Surrogate_0$	90,82 %	87,36 %	94,88 %	80,55 %	93,75 %	95,69 %	92,67 %
$CMAES_0$	88,10 %	86,17 %	69,41 %	95,28 %	98,12 %	80,82 %	98,83 %
MC_0	81,17 %	70,25 %	85,54 %	52,94 %	90,47 %	95,35 %	92,48 %
Surrogate	79,08 %	71,48 %	82,05 %	66,28 %	92,52 %	95,11 %	67,07 %
PSO	71,53 %	59,66 %	59,08 %	73,93 %	83,09 %	82,42 %	71,00 %
CMAES	70,08 %	63,58 %	45,56 %	81,90 %	83,47 %	69,50 %	76,45 %
PS_0	68,22 %	60,89 %	50,89 %	83,96 %	73,41 %	55,58 %	84,59 %
DE	66,05 %	55,63 %	53,18 %	71,36 %	78,51 %	78,17 %	59,47 %
GA-PS	54,22 %	44,67 %	26,39 %	80,96 %	64,49 %	47,45 %	61,37 %
GA	52,99 %	39,91 %	27,03 %	69,80 %	61,60 %	56,22 %	63,39 %
PS	32,02 %	18,36 %	4,78 %	80,59 %	27,89 %	20,27 %	40,23 %
MC	23,56 %	6,93 %	34,52 %	2,21 %	8,26 %	88,65 %	0,82 %

Tabelle A.5.: Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 6, Dabei wurde hier MC als der Referenzalgorithmus gewählt. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abbildungsverzeichnis

1.1. Blockschaltbild eines Regelkreises. 1

2.1. Darstellung eines dynamischen Systems. 5
2.2. Flussdiagramm des Genetic algorithm. 9
2.3. Flussdiagramm des Differential evolution algorithm. 10
2.4. Flussdiagramm zur Particle swarm optimization. 11
2.5. Flussdiagramm der Covariance Matrix Adaptation Evolution Strategy. . . . 13
2.6. Flussdiagramm des Surrogate Algorithmus. 14
2.7. Flussdiagramm des PS Algorithmus. 15
2.8. Flussdiagramm des GA-PS-Algorithmus. 17
2.9. Flussdiagramm des MC Algorithmus. 18

3.1. Elementare Bausteine b für die dynamischen Bereiche. Aus [23]. 23
3.2. Beispiel eines Signals mit $\boldsymbol{\xi} = [[0.2, 0.3, 0.2, 0.1]^{\mathrm{T}}, [1, 7]^{\mathrm{T}}, [0.3, 0.1, 0.9]^{\mathrm{T}}]^{\mathrm{T}}$. . . 25
3.3. Darstellung der Kostenfunktion, mit dem gewählte Fehlermaß und Σ. . . . 29
3.4. Darstellung eines Szenarios s. Dabei sieht man die 4 Parametermengen, 4 Regelkreise und 2 Fehlermaße. Es ergeben sich 32 mögliche Szenarien. . . . 31
3.5. Blockdiagramm zu Experiment 1. Dargestellt ist ein Einzelexperiment, wobei die Parametermenge Ξ_i, das System Σ und das Fehlermaß d fest gewählt ist. Die Kostenfunktion $f(\boldsymbol{\xi}, \Sigma, d)$ wird maximal N_{iter}-mal ausgewertet. $\boldsymbol{\xi}_0$ beschreibt die Startwerte für den Optimierer. 33
3.6. Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung. Die Verbreiterung des Signalpfades bei $\boldsymbol{\xi}_0$ verdeutlicht die Verwendung von N_{rep} Startwerten. 34
3.7. Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren. 34
3.8. Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren, jeder Ξ als auch jedem Fehlermaß d. Dies haben wir durch die Verbreiterung des Signalpfades bei diesen hervorgehoben. 35
3.9. Dargestellt ist der N_{rep}-fache Durchlauf der Optimierung, mit jedem globalen Optimierungsverfahren, jeder Ξ, jedes betrachteten Systems Σ und jedes Fehlermaßes d. Dies haben wir durch die Verbreiterung des Signalpfades bei diesen hervorgehoben. 36
3.10. Darstellung der Cluster von D_6^{PMV}, nach den durchschnittlichen Kostenfunktionswerten sortierten. Die Farbmarkierungen zeigen die Herkunft der Daten in Bezug auf die jeweiligen Optimierer. 38
3.11. Darstellung vom Ursprung der Daten für $\Xi_{0,i}^s$ für ein s mit dem Modell FMN im Experiment 6. 39

4.1. Feder-Masse System mit Haftreibung. 40

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4.2. Übersicht von Vor- und Hauptstufe des Pneumatikventils, siehe [28, Abbildung 2.1]. Links ist die Vorstufe mit den acht piezoelektrisch aktuierten Balken und rechts die Hauptstufe mit vier Ventilstößeln dargestellt. 43
4.3. Darstellung eines Ventils der Hauptstufe, mit Rückstellfeder. Der Massenstrom $\dot{m}$ in die Vorkammer, sowie der, durch Auslenkung des Ventilstößels, resultierende Luftstrom der Hauptstufe sind durch Pfeile angedeutet. Diese Darstellung ist aus [30] entnommen. 44

5.1. Grafische Darstellung der Bewertung anhand der festgelegten Kriterien. Als Eingabedaten dienen $|\mathcal{O}|$ verschiedene $\boldsymbol{J}^o$-Matrizen, die mittels Bewertungsfunktionen in skalare Werte überführt werden. Exemplarisch wird hier die Matrix $\boldsymbol{J}^{GA}$ (rot) anhand der Kriterien bewertet. Die dargestellten Werte dienen ausschließlich der Veranschaulichung und sind beispielhaft. 48
5.2. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 5. $\boldsymbol{P}_{all}$ ist in Prozent angeben und setzt sich aus den Bewertungen der 6 Kriterien zusammen. 53
5.3. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell FMSL. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen. 57
5.4. Simulation eines Worst-Case Signals an dem Modell FMSL, das durch die Optimierung mittels CMAES unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes MAE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von FMSL. In der zweiten Grafik ist die Eingangskraft F_u darstellt. Darunter ist der absolute Fehler in jedem Abtastschritt eingetragen. In der letzten Grafik ist das Anwachsen der Kostenfunktion und die Kostenfunktion dargestellt. 58
5.5. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell FMSN. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen. 59
5.6. Simulation eines Worst-Case Signals für FMSN, das durch die Optimierung mittels Surrogate unter Verwendung der Parametermenge Ξ_4 und des Fehlermaßes RMSE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von FMSN. In der zweiten Grafik ist die Eingangskraft F_u darstellt. Darunter ist der quadratische Fehler in jedem Abtastschritt eingetragen. In der letzten Grafik ist das Anwachsen der Kostenfunktion und die Kostenfunktion dargestellt. 61
5.7. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell PVED. Die Bewertung ist in Prozent angeben, und setzt sich aus den Bewertungen der 6 Kriterien zusammen. 62

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

5.8. Simulation eines Worst-Case Signals für PVED, das durch die Optimierung mittels Surrogate unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes RMSE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von PVED, als auch die Eingangsgröße Druck, welche mit dem maximal möglichen Druck skaliert wurde. Darunter ist der quadratische Fehler in jedem Zeitschritt abgebildet. In der vierten Grafik sind die Kosten zusehen. 63
5.9. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten des Experiments 4, mit dem Modell PVEM. Die Bewertung ist in Prozent angeben, und stellt sich aus den Bewertungen der 6 Kriterien zusammen. 64
5.10. Simulation eines Worst-Case Signals für PVEM, das durch die Optimierung mittels CMAES unter Verwendung der Parametermenge Ξ_6 und des Fehlermaßes MAE erzeugt wurde. Dargestellt ist das Sollsignal y^d und das Ausgangssignal y_{sys} von PVEM, als auch die Eingangsgröße Massenstrom. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der vierten Grafik sind die Kosten zusehen. 66
5.11. Visualisierung von $\boldsymbol{P}_{all}$ mit allen Modellen. Die Bewertung ist in Prozent angeben. 67
5.12. Visualisierung der Ergebnisse des Bewertungssystems mit den Daten von Experiment 6. Dabei wird $\boldsymbol{P}_{all}$ dargestellt mit den 6 Kriterien über welche der Mittelwert gebildet wurde. Blaue Umrandungen repräsentieren Algorithmen mit Startwerten, rote ohne Startwerte. 68
5.13. Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist des Ergebnisses des optimierungsbasierten Signalgenerator mit dem Optimierer Surrogate, der Parametermenge Ξ_4 und des Fehlermaßes MAE. Im obersten Verlauf ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen. 72
5.14. Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist des Ergebnisses des optimierungsbasierten Signalgenerator mit dem Optimierer MC, der Parametermenge Ξ_4 und des Fehlermaßes MAE. Es soll als Vergleich zu Abbildung 5.13 dienen. Im obersten Verlauf ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen. 73

A.1. Ergebnisse der Messung auf dem Prüfstand des pneumatischen Ventils. Das Sollsignal ist ein Element aus $\Xi_{0,4}$. In der obersten Abbildung ist das Sollsignal y^d und das Ausgangssignal y_{sys} der Messung zu sehen. Darunter ist der absolute Fehler in jedem Zeitschritt abgebildet. In der letzten Grafik sind die Kosten zusehen. 82

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Tabellenverzeichnis

2.1. Vergleich globaler Optimierungsalgorithmen anhand der Literaturrecherche. Die verwendeten Symbole repräsentieren Bewertungsstufen, die von höchster Qualität, ++, bis niedrigster Qualität, −−, reichen. Die Symbole + und − kennzeichnen Zwischenstufen, während ∼, + ∼ und − ∼ Übergangsbewertungen darstellen. 19

3.1. Auflistung der Bausteine zur Konstruktion eines dynamischen Bereichs. . . 22
3.2. Auflistung der maximalen Änderungsrate der Bausteine zur Konstruktion eines dynamischen Bereichs. 26
3.3. Übersicht zu den Begriffen der Simulationsstudien. 32

4.1. Parameter des mechanischen Systems. 42
4.2. Beschreibung physikalischer Parameter des pneumatischen Ventils. 46

5.1. Ergebnisse eines Experiments zur Wiederholbarkeit $\boldsymbol{J}^o$ mit exemplarischen Zahlenwerten. Dabei beschreibt die letzte Spalte den Vektor $\bar{\boldsymbol{J}}^o$, gemäß Gleichung (5.1), welcher in jedem Eintrag $\bar{\boldsymbol{J}}^o(i)$ den letzten Kostenfunktionswert der Optimierung darstellt. 49
5.2. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 5. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet und beziehen sich auf den Mittelwert der jeweiligen normierten Punkte in jedem Kriterium, bis auf $\boldsymbol{P}_{all}$, welches den Mittelwert über alle Kriterien entspricht. 54
5.3. Vergleich der Algorithmen hinsichtlich der in Abschnitt 3.2 betrachteten Kategorien. Dabei ist die Reihenfolge der Bewertungen folgende, ++, +, + ∼, ∼, − ∼, −, −−. 67
5.4. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 6. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 69

A.1. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell FML. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 80
A.2. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell FMN. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 80
A.3. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell PVD. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 81

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A.4. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 4 mit Modell PVM. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 81
A.5. Ergebnisse des Bewertungssystems, basierend auf den Daten aus Experiment 6, Dabei wurde hier MC als der Referenzalgorithmus gewählt. Die angegebenen Prozentwerte sind auf die letzten beiden Nachkommastellen gerundet. 83

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Literaturverzeichnis

[1] C. He, Y. Zhang, D. Gong, and X. Ji, "A review of surrogate-assisted evolutionary algorithms for expensive optimization problems," *Expert Systems with Applications*, vol. 217, p. 119495, 2023. [Online]. Available: https://doi.org/10.1016/j.eswa.2022.119495

[2] J.-Y. Li, Z.-H. Zhan, and J. Zhang, "Evolutionary computation for expensive optimization: A survey," *Machine Intelligence Research*, vol. 19, no. 1, pp. 3–23, Februar 2022. [Online]. Available: https://doi.org/10.1007/s11633-022-1317-4

[3] M. Zhou, M. Cui, D. Xu, S. Zhu, Z. Zhao, and A. Abusorrah, "Evolutionary optimization methods for high-dimensional expensive problems: A survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 5, pp. 1092–1105, 2024. [Online]. Available: https://doi.org/10.1109/JAS.2024.124320

[4] X. Wang, G. G. Wang, B. Song, P. Wang, and Y. Wang, "A novel evolutionary sampling assisted optimization method for high-dimensional expensive problems," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 815–827, 2019. [Online]. Available: https://doi.org/10.1109/TEVC.2019.2890818

[5] T. Frey and M. Bossert, *Signal- und Systemtheorie*. Vieweg+Teubner, 2009. [Online]. Available: https://doi.org/10.1007/978-3-8348-9292-8

[6] O. Föllinger, "Einführung in die Zustandsbeschreibung dynamischer Systeme," *at - Automatisierungstechnik*, vol. 28, no. 1-12, pp. 429–432, 1980. [Online]. Available: https://doi.org/10.1524/auto.1980.28.112.429

[7] M. Cremer, *Regelungstechnik: Eine Einführung*, ser. Springer-Lehrbuch. Springer Berlin Heidelberg, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-76994-8

[8] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the cec'2005 special session on real parameter optimization," *Journal of Heuristics*, vol. 15, no. 6, pp. 617–644, 2009. [Online]. Available: https://doi.org/10.1007/s10732-008-9080-4

[9] L. Bliek, A. Guijt, R. Karlsson, S. Verwer, and M. de Weerdt, "Benchmarking surrogate-based optimisation algorithms on expensive black-box functions," *Applied Soft Computing*, vol. 147, p. 110744, 2023. [Online]. Available: https://doi.org/10.1016/j.asoc.2023.110744

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[10] X. Cai, L. Gao, and X. Li, "Efficient generalized surrogate-assisted evolutionary algorithm for high-dimensional expensive problems," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, pp. 365–379, 2020. [Online]. Available: https://doi.org/10.1109/TEVC.2019.2919762

[11] J. H. Holland, *Genetic Algorithms and Adaptation*. Boston, MA: Springer US, 1984, pp. 317–333. [Online]. Available: https://doi.org/10.1007/978-1-4684-8941-5_21

[12] D. Goldberg, *Genetic Algorithms*. Pearson Education India, 2013.

[13] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dezember 1997. [Online]. Available: https://doi.org/10.1023/A:1008202821328

[14] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948 vol.4. [Online]. Available: https://doi.org/10.1109/ICNN.1995.488968

[15] N. Hansen and A. Ostermeier, "Completely Derandomized Self-Adaptation in Evolution Strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, Juni 2001. [Online]. Available: https://doi.org/10.1162/106365601750190398

[16] N. Hansen, "The CMA evolution strategy: A tutorial," *CoRR*, vol. abs/1604.00772, 2016. [Online]. Available: https://doi.org/10.48550/arXiv.1604.00772

[17] N. Hansen and S. Kern, "Evaluating the CMA evolution strategy on multimodal test functions," in *Parallel Problem Solving from Nature PPSN VIII*, ser. LNCS, X. Yao *et al.*, Eds., vol. 3242. Springer, 2004, pp. 282–291. [Online]. Available: https://doi.org/10.1007/978-3-540-30217-9_29

[18] N. Hansen, "Benchmarking a BI-population CMA-ES on the BBOB-2009 function testbed," in *Workshop Proceedings of the GECCO Genetic and Evolutionary Computation Conference*. ACM, Juli 2009, pp. 2389–2402. [Online]. Available: https://doi.org/10.1145/1570256.1570333

[19] K. K. Vu, C. D'ambrosio, Y. Hamadi, and L. Liberti, "Surrogate-based methods for black-box optimization," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 393–424, Mai 2017. [Online]. Available: https://doi.org/10.1111/itor.12292

[20] H.-M. Gutmann, "A radial basis function method for global optimization," *Journal of Global Optimization*, vol. 19, no. 3, pp. 201–227, März 2001. [Online]. Available: https://doi.org/10.1023/A:1011255519438

[21] R. Hooke and T. A. Jeeves, ""direct search"solution of numerical and statistical problems," *Journal of the ACM (JACM)*, vol. 8, no. 2, pp. 212–229, 1961. [Online]. Available: https://doi.org/10.1145/321062.321069

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[22] T. M. Alkhamis and M. A. Ahmed, "A modified hooke and jeeves algorithm with likelihood ratio performance extrapolation for simulation optimization," *European Journal of Operational Research*, vol. 174, no. 3, pp. 1802–1815, 2006. [Online]. Available: https://doi.org/10.1016/j.ejor.2005.04.032

[23] C. Geischläger, "Analyse nichtlinearer Regelkreise mittels computerunterstützter Methoden," Diplomarbeit, Technische Universität Wien, Wien, Februar 2023. [Online]. Available: https://doi.org/10.34726/hss.2023.100620

[24] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999. [Online]. Available: https://doi.org/10.1145/331499.331504

[25] T.-T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognition*, vol. 48, no. 9, pp. 2839–2846, 2015. [Online]. Available: https://doi.org/10.1016/j.patcog.2015.03.009

[26] Produktbeschreibung Festo Ventil VEVM. Zuletzt aufgerufen: 12.10.2024. [Online]. Available: https://www.festo.com/at/de/p/ventil-id_VEVM/?q=%7E%3AsortByCoreRangeAndNewProduct

[27] T. Glück, D. Büchl, C. Krämer, A. Pfeffer, A. Risle, L. Hägele, and A. Kugi, "Modeling and control of a novel pneumatic two-stage piezoelectric-actuated valve," *Mechatronics*, vol. 75, pp. 102 529 : 1–11, Mai 2021. [Online]. Available: https://doi.org/10.1016/j.mechatronics.2021.102529

[28] C. Krämer, "Regelung von pneumatischen Lasten mithilfe eines piezoelektrischen Stetigventils," Diplomarbeit, Technische Universität Wien, Wien, Juni 2016. [Online]. Available: http://hdl.handle.net/20.500.12708/158627

[29] D. Büchl, "Modellierung und Regelung eines pneumatischen, piezoelektrischen Stetigventils," Diplomarbeit, Technische Universität Wien, Wien, Januar 2015. [Online]. Available: http://hdl.handle.net/20.500.12708/158627

[30] P. Strasser, "Simulation dynamischer Systeme mit neuronalen Netzen," Diplomarbeit, Technische Universität Wien, Wien, November 2024. [Online]. Available: https://doi.org/10.34726/hss.2024.120720

[31] M. Buehren. Differential evolution. Zuletzt aufgerufen: 8.12.2024. [Online]. Available: https://de.mathworks.com/help/gads/ga.html

[32] N. Hansen. CMAES. Zuletzt aufgerufen: 8.12.2024. [Online]. Available: https://github.com/CMA-ES/CMA-ES.github.io/tree/master

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN