

Robuste optimale Prozessplanung und -führung nichtlinearer Systeme unter Verwendung der Unscented-Transformation

vorgelegt von
Dipl.-Ing.
Michael Kawohl
aus Berlin

von der Fakultät III - Prozesswissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr.-Ing. habil. Prof. h.c. Dr. h.c. Günter Wozny

Gutachter: Prof. Dr.-Ing. habil. Rudibert King

Gutachter: Prof. Dr.-Ing. habil. Pu Li

Tag der wissenschaftlichen Aussprache: 13. Dezember 2013

Berlin 2014

D 83

Abstract

Die Optimierung verfahrenstechnischer Prozesse erfolgt in vielen Fällen klassisch durch reaktionstechnische, fluiddynamische oder andere verfahrenstechnische Eingriffe. In den letzten Jahren wird jedoch auch immer mehr versucht, bestehende Anlagen mit modernen modellbasierten Automatisierungskonzepten optimaler zu betreiben. Insbesondere chemische und biologische Batch und Semi-/Fedbatch Prozesse zeichnen sich jedoch meist durch eine ausgeprägt nichtlineare, zu stark unterschiedlichem Verhalten führende Eigendynamik aus.

Während sich die Strukturen der zu Grunde gelegten Modelle des Prozessverhaltens auf Basis chemisch-physikalischer Zusammenhänge herleiten lassen, sind diverse Modellparameter a priori unbekannt. Diese Parameter müssen aus experimentellen Daten identifiziert werden. Um auch mit wenigen Versuchen gute Identifizierbarkeit zu erreichen, werden Verfahren wie die Optimale-Versuchsplanung angewendet. Die hierbei verwendeten Ansätze fußen auf einer Analyse des verbleibenden Parameterfehlers mit Hilfe der Fisher'schen-Informationsmatrix. In dieser Arbeit wird gezeigt, dass die Berechnung der Fisher'schen-Informationsmatrix intern einer Approximation der Parametervarianz mit einem Approximationsansatz erster Ordnung entspricht. Davon ausgehend wird ein neues Verfahren zur Optimalen-Versuchsplanung vorgestellt, welches die Möglichkeit bietet, Approximationen beliebiger Ordnung zur Berechnung der verbleibenden Parameterfehler zu verwenden und im Falle der Nutzung einer Approximation 1. Ordnung äquivalent zur Fisher'schen-Informationsmatrix ist.

Ausgehend von einem mathematischen Prozessmodell können Trajektorien geplant werden, die maximale Produktivität versprechen. Im Rahmen der klassischen Prozessplanung und -führung werden die genauen Auswirkungen von Modellfehlern auf die geplante Trajektorie nicht beachtet. Robuste Ansätze zur modellbasierten Planung und Regelung von Prozessen, die explizit die Unsicherheiten beim Prozessstart und in den Modellparametern berücksichtigen, können Schwankungen der Produktivität stark reduzieren. Während bisher die Auswirkungen von unsicheren Größen auf den Prozessverlauf meist nur durch eine Approximation erster Ordnung prädiziert werden, stellt die vorliegende Arbeit einen neuen robusten Ansatz zur Prozessplanung und -führung vor, der auf der Unscented Transformation basiert. Die Unscented Transformation erlaubt die näherungsweise Berechnung der Verteilungsfunktion des Prozessausgangs basierend auf einer Approximation zweiter oder höherer Ordnung. Die Ergebnisse dieser Prädiktion werden unter Verwendung einer „Value@Risk“-Formulierung zur robusten Prozessoptimierung verwendet. Umfangreiche Simulationsstudien untermauern die Effizienz dieses robusten Ansatzes.

Abstract

Typically, optimization of chemical and biochemical processes is performed by classic process engineering, e.g. reaction engineering or fluid dynamics. In recent years, however, more and more modern model-based automation concepts have been used to optimize the operation of existing facilities. On the downside, chemical and biological batch and semi-/fedbatch processes are usually characterized by strong non-linear behavior.

While the structures of the underlying process models can be derived from physical and chemical relationships, various model parameters are unknown, a priori. These parameters have to be identified from experimental data. In order to achieve good identifiability even after few experiments, methods for Optimal-Experimental-Design are applied. These methods are based on an analysis of the remaining parameter error evaluated by the Fisher-Information-Matrix. In this work, it is shown that the calculation of the Fisher-Information-Matrix corresponds to a first order approximation of the remaining parameter variance. From this conclusion, a new method for Optimal-Experimental-Design is derived, which offers the possibility to use approximations of arbitrary order to calculate the remaining parameter error. When using a first order approximation, it is equivalent to Fisher-Information-Matrix.

Based on a mathematical process model, trajectories promising maximum productivity can be planned. In traditional process planning and execution, the exact impact of model errors on the planned trajectory is ignored. Robust approaches to model-based planning and control of processes explicitly take into account uncertainties of the initial process state and the model parameters. These approaches can greatly reduce fluctuations of productivity. So far, the effects of uncertain quantities are usually predicted by a first order approximation. This work presents a new robust approach to process planning and control based on the Unscented-Transformation. The Unscented-Transformation approximates the distribution function of the process output utilizing an approximation of second or higher order. The results of the prediction can be integrated in a "Value@Risk" approach for robust process optimization and control. Extensive simulation studies substantiate the efficiency of this robust approach.

Inhaltsverzeichnis

Abstract	i
Symbolverzeichnis	v
1. Einleitung	1
2. Mathematische Methoden	7
2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen	8
2.1.1. Monte-Carlo-Simulation	8
2.1.2. Approximation durch Taylor-Reihenentwicklung	10
2.1.3. Unscented-Transformation	15
2.2. Analyse der Unsicherheiten nichtlinearer Modelle	25
2.2.1. Die Bootstrap-Methode	27
2.2.2. Die Fisher'sche-Informationsmatrix	28
2.3. Optimale-Versuchsplanung	29
2.3.1. Optimale-Versuchsplanung mit der Fishermatrix	30
2.3.2. Ein Maximum-a-Posteriori-Ansatz zur Optimalen-Versuchsplanung	33
2.4. Modellbasierte Prozessplanung und -regelung	38
2.4.1. Trajektorienplanung	38
2.4.2. Nichtlineare-Modellprädiktive-Folgeregelung	39
2.4.3. Online-Trajektorienplanung	41
2.4.4. Diskretisierung der Güte- und Beschränkungsfunktion	42
2.5. Robuste Modellbasierte Prozessplanung und -führung	43
2.6. Numerische Methoden zur nichtlinearen Optimierung	46
2.6.1. Numerische Optimierungsverfahren	47
2.6.2. Gradientenberechnung bei unterliegender numerischer Integration .	50
2.6.3. Multiple-Shooting-Optimierung	51
3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung	59
3.1. Parameteranalyse	60
3.2. Optimale-Versuchsplanung	62
4. Anwendung robuster Prozessführungsverfahren	69
4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners	69
4.1.1. Trajektorienplanung	71
4.1.2. Robuste Trajektorienplanung	73

4.1.3.	Nichtlineare-Modellprädiktive-Folgeregelung	78
4.1.4.	Online-Trajektorienplanung	80
4.1.5.	Robuste Nichtlineare-Modellprädiktive-Folgeregelung	82
4.1.6.	Robuste Online-Trajektorienplanung	85
4.1.7.	Zusammenfassung	85
4.2.	Sulfonierung von Parachlornitrobenzol	86
4.2.1.	Trajektorienplanung	88
4.2.2.	Robuste Trajektorienplanung	90
4.2.3.	Nichtlineare-Modellprädiktive-Folgeregelung	96
4.2.4.	Robuste Nichtlineare-Modellprädiktive-Folgeregelung	98
4.3.	Anwendung auf die Produktion von Nikkomycin durch <i>Streptomyces tendae</i>	100
4.3.1.	Simulationsstudie	100
4.3.2.	Umsetzung einer robusten Nichtlinearen-Modellprädiktiven-Folge- regelung und einer robusten Online-Trajektorienplanung	107
5.	Zusammenfassung und Ausblick	115
A.	Ableitungsregeln für Vektoren	119
B.	Constrained-Unscented-Kalman-Filter	121
B.1.	Time-Update	121
B.2.	Measurement-Update	123
C.	Kompartimentmodell des Stammes <i>S. tendae</i>	127

Symbolverzeichnis

Lateinische Buchstaben

A	Mantelfläche
\underline{a}	Allgemeiner Eingangsvektor
\underline{b}	Allgemeiner Ausgangsvektor
\mathbf{C}	(Kreuz-)Kovarianzmatrix
c	Konzentration Federkonstante
c_P	Spezifische Wärmekapazität
D	Differenzenoperator
d	Dämpfungskonstante
\underline{d}	Partieller Differenzenoperator
\underline{e}	Einheitsvektor Endzustandsvektor eines Multiple Shooting Intervalls
E	Erwartungswert Aktivierungsenergie
\mathbf{F}	Fisher'sche-Informationsmatrix
\underline{f}	Rechte Seite des gewöhnlichen Differentialgleichungssystems
g	Konzentration
\underline{g}	Lösung eines Differentialgleichungssystems
\mathbf{G}	Sensitivitätsmatrix

\underline{g}	Allgemeine vektorielle Funktion Lösung eines Differentialgleichungssystems
$\{H\}$	Schrittfolge
h	Schrittweite
\underline{h}	Messgleichung
ΔH_R	Reaktionsenthalpie
(i)	Zählvariable
\mathbf{I}	Einheitsmatrix
K	(Monod-)Konstante
k	Zählvariable Zeitpunkt t_k Reaktionsgeschwindigkeit / Stoßzahl
L	Dimension
M	Molare Masse
m	Masse
N	Anzahl
\mathcal{N}	Normalverteilung
n	Anzahl Molzahl
P	Wahrscheinlichkeit
\underline{p}	Optimierungsparametervektor
p	Optimierungsparameter Wahrscheinlichkeitsdichte
R	Allgemeine Gaskonstante

\mathbb{R}	Menge der reellen Zahlen
r	Beschränkungsfunktion
\mathbf{S}	Matrixwurzel einer Kovarianzmatrix \mathbf{C}
\underline{s}	Spaltenvektor von \mathbf{S}
	Startzustandsvektor eines Multiple Shooting Intervalls
t	Zeit
T	Zeit
	Temperatur
T_C	Rechenzeit
T_P	Prädiktionshorizont
T_S	Stellhorizont
U	Energie
u	Stellgröße
\mathbf{U}	Menge der Stellgrößen
\underline{u}	Stellgrößenvektor
V	Volumen
\mathbf{W}	Gewichtungsmatrix
x	Zustandsgröße
\underline{x}	Zustandsvektor
K	Ausbeutekoeffizient
y	Messgröße
\underline{y}	Messgrößenvektor
\underline{y}	Messgrößenvektor
\mathbf{Z}	Sensitivitätsmatrix

\underline{z}	Transformierter allgemeiner Eingangsvektor
(\cdot)	Platzhalter für verschiedene Größen

Griechische Buchstaben

α	Wärmedurchgangskoeffizient
δ	Dirac-Impuls
$\underline{\eta}$	Messrauschen (Vektor)
$\underline{\mathfrak{N}}$	Messrauschen (Vektor)
γ	Gewichtung des Risikos
λ	Abstand zur Prozessgrenze
μ	Wachstumsrate
Φ	Gütefunktion
ϕ	Aktivität
ρ	Dichte
σ	Standardabweichung
θ	Parameter
$\underline{\theta}$	Parametervektor
$\underline{\xi}$	Systemrauschen (Vektor)

Indizes

0	Anfangswert Wert zum Zeitpunkt $t = 0$
Am	Ammonium
As	Aminosäuren
C	Glukose
D	DNA
E	Experimente Erhaltungstoffwechsel
Ende	Endwert
krit	kritisch
max	Maximalwert
min	Minimalwert
M	Mantel
Ni	Nikkomycin
Nu	Nukleotide
Ph	Phosphat
Pr	Proteine
R	RNA
SIM	Simulation
U	Strukturelemente
zu	Zulauf

Hochgestellte Größen

$\hat{}$	Geschätzter Wert
(i)	Zählvariable
$-$	Mittelwert
$'$	Neuer Wert
(SP)	Sigmapunkt
$*$	Wahrer Wert
	Optimaler Wert
\sim	Transformierte Größe

Abkürzungen

COM	Component-Object-Model
DAE	Differential-Algebra-Gleichung (Differential algebraic equation)
DCOM	Distributed-Component-Object-Model
FIM	Fisher'sche-Informationsmatrix
MAP	Maximum-a-Posteriori
MC	Monte-Carlo-Simulation
MLE	Maximum-Likelihood-Identifikation (Maximum Likelihood Estimation)
MS	Multiple-Shooting
NMPC	Nichtlineare-Modellprädiktive-Folgeregelung
ODE	Gewöhnliche-Differentialgleichung (Ordinary Differential Equation)
OT	Online-Trajektorienplanung
OVP	Optimale-Versuchsplanung
PCNB	Parachlornitrobenzol
PCNBS ₂	Parachlornitrobenzoldisulfonsäure
PCNBS	Parachlornitrobenzolsulfonsäure
PI	Parameteridentifikation
P	Produkt
RNMPC	Robuste Nichtlineare-Modellprädiktive-Folgeregelung
ROT	Robuste Online-Trajektorienplanung
RTP	Robuste Trajektorienplanung
SISO	Single-Input-Single-Output
SQP	Sequentielle-Quadratische-Programmierung
S	Substrat

TP	Trajektorienplanung
UT	Unscented-Transformation
X	Biotrockenmasse

Alle lateinischen und griechischen Buchstaben und alle Abkürzungen können auch als Indizes anderer Variablen auftreten.

1. Einleitung

Die Optimierung verfahrenstechnischer Prozesse erfolgt in den meisten Fällen klassisch durch reaktionstechnische, fluiddynamische oder andere verfahrenstechnische Eingriffe, die oft mit einem Um- oder Neubau von Anlagen verbunden sind. In den letzten Jahren wird, u.a. bedingt durch die jetzt verfügbare Rechenleistung, jedoch auch immer mehr versucht, bestehende Anlagen mit modernen Automatisierungskonzepten optimaler zu betreiben. Das Potential, welches auf diese Art und Weise gehoben werden kann, ist zwar geringer als bei klassischer verfahrenstechnischer Optimierung, die Kosten einer Automatisierungslösung jedoch auch um Größenordnungen geringer [QB03], als ein Anlagenumbau. Für lineare dynamische Systemmodelle und stationäre physikalische Modelle stehen mit Software von z.B. AspenTech [CR80] oder Honeywell [NA00] auch leistungsfähige kommerzielle Programme für die Lösung dieser Aufgaben zur Verfügung.

Insbesondere chemische und biologische Batch und Semi-/Fedbatch Prozesse zeichnen sich jedoch meist durch eine ausgeprägt nichtlineare, zu stark unterschiedlichem Verhalten führende Eigendynamik aus. Um Anforderungen an die Qualität und Quantität der dabei herzustellenden Produkte zu erfüllen, wird oft eine modellgestützte Prozessauslegung auf Basis chemisch-/physikalisch fundierter Modelle angestrebt. In den vergangenen Jahren wird daher vermehrt der Versuch unternommen, mit analytisch erstellten Zustandsraummodellen die komplexen Mechanismen in chemischen und biologischen Systemen zu beschreiben [Bai98, SK00, Lud99, LK04, KSK00, Far92]. Die hierfür verwendeten Modelle dienen im nächsten Schritt auch online für eine modellgestützte Zustandsschätzung und -regelung, so dass Störungen des geplanten Betriebs durch leistungsfähige modellbasierte Verfahren ausgeglichen werden können. Die Güte der Prozessführung ist in diesem Fall direkt mit der Modellgüte, insbesondere mit der Prädiktionsgenauigkeit des Modells gekoppelt. Um eine möglichst exakte Wiedergabe des realen Systemverhaltens zu erreichen, kommen immer komplexere, hoch parametrisierte Systembeschreibungen zum Einsatz, die im Bereich der Chemie und Biologie fast ausnahmslos nichtlinear sind.

Während sich die Modellstrukturen auf Basis chemisch-physikalischer Zusammenhänge herleiten lassen, sind diverse Modellparameter a priori unbekannt. Diese Parameter müssen aus experimentellen Daten identifiziert werden. Die zur Parameteridentifikation verwendeten Messdaten basieren oft auf einfachen Puls- oder Sprungversuchen. Nicht selten stehen dabei nur wenige Messdaten zur Verfügung. Auch können nicht alle Modellparameter mit gleich hoher Genauigkeit identifiziert werden.

1. Einleitung

Um auch mit wenigen Versuchen gute Identifizierbarkeit zu erreichen, ist eine optimale Anregung des biologischen Systems nötig, wie sie mit Hilfe der Optimalen-Versuchsplanung [BSSR94, BBKS00, CV95, GP77, Kör02, MP89a, MP89b] berechnet werden kann. Die hierbei verwendeten Ansätze fußen auf einer Parameteridentifikation mit Hilfe eines Maximum-Likelihood Ansatzes [Ise88] und einer Analyse des verbleibenden Parameterfehlers mit Hilfe der Fisher'schen-Informationsmatrix [Fis71, MP89a]. In dieser Arbeit wird gezeigt, dass die Berechnung der Fisher'schen-Informationsmatrix intern einer Approximation der Parametervarianz mit einem Approximationsansatz erster Ordnung entspricht. Davon ausgehend wird ein neues Verfahren zur Optimalen Versuchsplanung vorgestellt, welches die Möglichkeit bietet, Approximationen beliebiger Ordnung zur Berechnung der verbleibenden Parameterfehler zu verwenden und im Falle der Nutzung einer Approximation 1. Ordnung äquivalent zur Fisher'sche-Informationsmatrix ist.

Ausgehend von einem mathematischen Prozessmodell können Trajektorien geplant werden, die maximale Produktivität versprechen. Die für die Planung optimaler Prozessläufe verwendeten mathematischen Modelle stellen in der Regel nur eine Approximation der Realität dar. Ursachen sind nicht modellierte Effekte und nicht genau bekannte Modellparameter, die aus verrauschten Messdaten identifiziert werden müssen. Die so berechneten Parameter und der mit dem Modell prädizierte Prozessverlauf sind als Folge der zufällig verteilten Messfehler ebenfalls zufällig verteilt. Im Rahmen der klassischen Prozessplanung und -führung werden die genauen Auswirkungen von Modellfehlern auf die geplante Trajektorie nicht beachtet.

Durch die Aufschaltung von leistungsfähigen, modellbasierten Regelungsalgorithmen, wie zum Beispiel der Nichtlinearen-Modellprädiktiven-Folgeregelung [AZ00, CB99, DFS⁺02, Eng02, DFS⁺03, Hei04] oder adaptiven Reglern [Est95, WKG97], wird versucht, die Abweichungen des Prozessverlaufes von der optimalen Trajektorie zu minimieren und den Prozess innerhalb der Beschränkungen zu halten. Auswirkungen der Unsicherheiten können dennoch den Prozessverlauf, wenn auch verringert, negativ beeinflussen. Ohne einen regelnden Eingriff wird zum einen die Produktivität variieren, zum anderen kann es beim Überschreiten von sicherheitskritischen Grenzen sogar zum Produktionsausfall oder zu Unfällen kommen. Dies ist zu beachten, da sich als Ergebnis einer Trajektorienplanung oft Verläufe ergeben, die nah an den physikalischen und/oder sicherheitsrelevanten Grenzen entlang führen.

Robuste Ansätze zur modellbasierten Planung und Regelung von Prozessen, die explizit die Unsicherheiten beim Prozessstart und in den Modellparametern in Form von Intervallen, Sensitivitäten oder Wahrscheinlichkeitsdichtefunktionen berücksichtigen, können Schwankungen der Produktivität stark reduzieren. Insbesondere kann mit diesen Verfahren aber auch das Risiko minimiert werden, dass Auswirkungen der o.g. Unsicherheiten zu Verletzungen sicherheitskritischer Grenzen führen. Für solche robusten Betrachtungen nichtlinearer Systeme lassen sich, wie erwähnt, verschiedene Beschreibungsformen der Unsicherheiten unterscheiden. Zum einen ist eine Darstellung mit mathematischen Intervallen, zum anderen mit stochastischen Größen möglich.

Im ersten Fall können mit der Intervallmathematik [Moo79, Neu90] die Auswirkungen von unsicheren Anfangsbedingungen und Modellparametern auf die Lösung des in der Prozesstechnik meist nichtlinearen Differentialgleichungssystems berechnet werden. Diese Verfahren zeigen bedingt durch die Methoden der Intervallarithmetik im Allgemeinen eine starke Überschätzung der tatsächlichen Lösungsintervalle. Aktuell werden zwar Rückprojektionsverfahren untersucht, die diese Fehler verringern [DJH02]. Der Aufwand zur Berechnung minimaler einhüllender Intervalle steigt jedoch exponentiell mit der Anzahl der unsicheren Größen, wie auch eigene explorative Arbeiten auf diesem Gebiet bestätigen. Dieser sehr hohe Rechenaufwand bei den verwendeten Rückprojektionsverfahren ermöglicht bis dato aber noch nicht den Einsatz im Rahmen der optimierungsbasierten Prozessplanung [KRAH05]. Solange hier nicht neue Methoden mit logarithmischer oder polynomialer Größenordnung gefunden werden, ist die Anwendung zwar für kleine Beispielsysteme nicht jedoch für komplexe Modelle realer Systeme denkbar.

Bei der Verwendung von Wahrscheinlichkeitsdichtefunktionen zur Beschreibung der Unsicherheiten ist bei nichtlinearen Systemen die exakte Berechnung der Verteilungsfunktion des Prozessergebnisses nur durch umfangreiche Monte-Carlo-Simulationen möglich [Sha00]. Auch solche Ansätze zeichnen sich durch einen sehr hohen Rechenaufwand aus. Zusätzlich beinhalten Monte-Carlo-Simulationen eine zufällige Komponente, was den Einsatz moderner Optimierungsverfahren nur eingeschränkt ermöglicht.

In [NB03, NB04, KDM⁺05] und [DBK06] ist ein Ansatz beschrieben, der es ermöglicht, die Auswirkungen von Unsicherheiten auf die zu planende Trajektorie approximativ zu berechnen und im Rahmen einer robusten Prozessplanung und -führung zu berücksichtigen. Andere ähnliche Ansätze wie z.B. [CSA97, CA98] nutzen Methoden aus der Spieltheorie um eine robuste nichtlineare Regelung zu entwerfen. Auch lassen sich unter dem Begriff robuste Regelung \mathbf{H}_∞ -Methoden [Kwa93] finden, die im klassischen Sinn Robustheit bezüglich der Stabilität und der Performance erzielen wollen.

Die Definition des Begriffes „robust“ beinhaltet in dieser Arbeit nicht das in der Regelungstechnik übliche robust bezüglich der Stabilität des Verfahrens, sondern bezieht sich darauf, dass ein Prozesslauf so geplant wird, dass sich zufällige Störungen auf Einflussgrößen möglichst wenig auf den Prozessverlauf auswirken. Bei reinen Steuerungen wäre sonst eine Benennung robust nicht möglich. Der Beweis der Stabilität nichtlinearer Regelungsverfahren ist schon für klassische Verfahren nur schwer möglich und mit vielen Einschränkungen verbunden. Ein Stabilitätsbeweis für die vorgestellten Regelungsverfahren wird hier daher nicht angestrebt. Vielmehr wird an einigen Beispielen mit sehr umfangreichen Simulationsstudien gezeigt, wie solche robusten Ansätze die Schwankungen der Prozessverläufe bei zufälligen Störungen stark verringern und damit hohe Sicherheit bezüglich Produktivität und Einhaltung von Prozessgrenzen liefern. Verringert man die Auswirkungen von Störungen auf den Prozessverlauf ist jedoch zusätzlich auch eine erhöhte Robustheit im regelungstechnischen Sinne zu erwarten.

In [Li07] wird ein robustes Prozessführungsverfahren vorgestellt, welches auf einer Inversion des Prozessmodells beruht. Während erste Arbeiten ausschließlich für stationäre

1. Einleitung

Modelle galten, ist aktuell auch deren Erweiterung auf dynamische Modelle vorgestellt worden. Diese Methodik fokussiert auf Robustheit bezüglich von Prozessgrenzen. Hierbei werden die Prozessgrenzen invers auf die Verteilung der unsicheren Größen projiziert. Somit können die Wahrscheinlichkeiten einer Grenzverletzung gut vorhergesagt und im Rahmen der Prozessführung verwendet werden. Die ursprüngliche Methode beinhaltet einige Einschränkungen bezüglich der Monotonie der Prozessmodelle. In [AG06] wird eine Erweiterung vorgestellt, die diese Einschränkungen behandelt. Anders als in der hier vorliegenden Arbeit muss zwar eine Verteilungsfunktion für die unsicheren Größen angenommen werden, nicht jedoch für den daraus resultierenden Prozessverlauf. Die Entwicklung insbesondere der Erweiterung auf dynamische Probleme erfolgte zeitlich parallel zu den hier vorgestellten Arbeiten. Ein direkter Vergleich mit dieser Methodik ist hier daher nicht möglich.

Während [NB04, KDM⁺05] und [DBK06] und andere, die vorwärts die Auswirkungen von unsicheren Größen auf den Prozessverlauf berechnen, meist nur eine Approximation erster Ordnung zur Prädiktion der statistischen Momente verwenden, stellt die vorliegende Arbeit einen neuen robusten Ansatz zur Prozessplanung und -führung vor, der auf der Unscented-Transformation [JU96] basiert. Die Unscented-Transformation ist im regelungstechnischen Umfeld vor allem durch Zustandsschätzverfahren auf Basis dieser Approximationsmethode bekannt geworden [JUDW95, NPR00a, NPR00b, JU04, Mer04]. Die Unscented-Transformation erlaubt die näherungsweise Berechnung der Verteilungsfunktion des Prozessausgangs basierend auf einer Approximation zweiter oder höherer Ordnung [NPR00a].

Ausgehend von der Beschreibung der Unsicherheiten existieren verschiedene Optimierungsstrategien zur Berechnung optimaler Prozessverläufe. Als MinMax-Optimierung wird die Maximierung der Produktivität für das Worst-Case-Szenario bezeichnet [NB04]. Andere Ansätze erweitern das nominelle Gütefunktional um mindestens einen Term, der die Robustheit repräsentiert. Diese Funktionale sind unter dem Oberbegriff „Value@Risk“ zusammengefasst [RS05]. Zur Risikobewertung dienen hier meist die Standardabweichung oder die Varianz des Gütefunktional. Diese Methoden sind insbesondere aus der Finanzmathematik bekannt. Um Systembeschränkungen in die robuste Betrachtung einzubinden, können basierend auf der stochastischen Betrachtung Konfidenzintervalle der Systemgrößen angegeben werden. Eine Übersicht ist z.B. [SBVP03], [Sah04] zu entnehmen.

Diese Arbeit gliedert sich wie folgt. Das Kapitel 2 befasst sich mit dem theoretischen Hintergrund der verwendeten Verfahren und stellt eigene Erweiterungen vor. Zunächst werden in Abschnitt 2.1 die mathematischen Grundlagen zur approximativen Berechnung der statistischen Verteilung von Prozessverläufen mit Hilfe der Monte-Carlo-Simulation, einer Approximation erster Ordnung und einer Approximation zweiter Ordnung vorgestellt und in ein gemeinsames Framework integriert. Da die statistische Analyse der verbleibenden Modellunsicherheiten von signifikanter Bedeutung für robuste Planungsverfahren ist, erfolgt in Abschnitt 2.2 eine kurze theoretische Wiederholung mathematischer Verfahren, die eine solche Analyse ermöglichen. In Abschnitt 2.3 wird

ein neues Verfahren zur Optimalen-Versuchsplanung vorgestellt, welches im Gegensatz zu bisherigen Ansätzen, die üblicherweise auf der Fisher'sche-Informationsmatrix beruhen, die Möglichkeit bietet, die benötigten statistischen Momente mit beliebigen Approximationsverfahren zu berechnen. Im Abschnitt 2.4 sind klassische modellbasierte Prozessplanungs- und -regelungsalgorithmen zusammengefasst. Die Erweiterung der klassischen Verfahren auf einen robusten Prozessführungsansatz ist in Abschnitt 2.5 dargestellt.

Neben der methodischen Verbesserung der Prozessführungsalgorithmen spielen die unterlagerten numerischen Optimierungsverfahren eine wesentliche Rolle. In der Literatur finden sich viele Veröffentlichungen zum Thema Optimierung. So seien z.B. [Pap96, Poh99, MD03, KW03] genannt. Für die Optimierung dynamischer Systeme wird in den meisten Fällen das Single-Shoot-Verfahren angewendet. Signifikante Vorteile sind mit modernen Multiple-Shoot-Ansätzen [SB92, BDLS00, LSBS03] möglich. Solche Algorithmen werden in dieser Arbeit verwendet und in Abschnitt 2.6.3 näher vorgestellt.

Das Kapitel 3 zeigt die Anwendung des in Abschnitt 2.3 vorgestellten neuen Versuchsplanungsverfahrens auf ein einfaches Simulationsbeispiel. Im Kapitel 4 werden die in der Theorie eingeführten robusten Prozessführungsverfahren in umfangreichen Simulationsstudien mit klassischen modellbasierten Verfahren verglichen. Abschnitt 4.3 zeigt am Beispiel der Fermentation des Antibiotikaproduzenten *Streptomyces tendae* die Umsetzung robuster Prozessführungsansätze an einem realen System und Kapitel 5 fasst die Ergebnisse zusammen.

1. *Einleitung*

2. Mathematische Methoden

Ausgangspunkt für die vorgestellten Methoden bildet ein mathematisches kontinuierlich-diskretes Systemmodell der Form

$$\dot{\underline{x}}(t) = \underline{f}(t, \underline{x}(t), \underline{\theta}, \underline{u}(t)), \quad \underline{x}(t=0) = \underline{x}_0 \quad (2.1a)$$

$$\underline{y}(t_k) = \underline{h}(t_k, \underline{x}(t_k), \underline{\theta}, \underline{u}(t_k)). \quad (2.1b)$$

Hier beschreiben $\underline{x} \in \mathbb{R}^{L_x}$ den Systemzustand, $\underline{\theta} \in \mathbb{R}^{L_\theta}$ die Modellparameter, $\underline{u} \in \mathbb{R}^{L_u}$ die Stellgrößen und $\underline{y} \in \mathbb{R}^{L_y}$ den Messwertvektor an den diskreten Messzeitpunkten t_k . Durch diese Form eines nichtlinearen Zustandsraummodells zusammengesetzt aus einem gewöhnlichen Differentialgleichungssystem erster Ordnung (ODE), Gl. (2.1a), und einer Messgleichung, Gl. (2.1b), können viele chemische und biochemische Produktionsprozesse mathematisch wiedergegeben werden. Viele Differentialalgebrasysteme (DAE), die ebenfalls oft zur Beschreibung solcher Systeme herangezogen werden, lassen sich in ODE-Systeme umformen. Zusätzlich sind die in dieser Arbeit vorgestellten Verfahren mit kleinen Adaptionen auch für DAE-Systeme anwendbar. Somit soll hier ausschließlich auf ODE-Systeme eingegangen werden. Die Darstellung des Modells als rein deterministisches System stellt jedoch nur eine grobe Näherung des realen Systems dar. Eine Möglichkeit, zu erwartende Unsicherheiten im Anfangszustand \underline{x}_0 und den Modellparametern $\underline{\theta}$, sowie weitere Modellunsicherheiten zu berücksichtigen, ist die Beschreibung dieser Größen durch Wahrscheinlichkeitsdichtefunktionen. Ausgangspunkt für Methoden, die auf Wahrscheinlichkeitsdichtefunktionen beruhen, bildet ein erweitertes mathematisches Systemmodell der Form

$$\dot{\underline{x}}(t) = \underline{f}(t, \underline{x}(t), \underline{\theta}, \underline{u}(t)) + \underline{\xi}, \quad \underline{x}(t=0) = \underline{x}_0 \quad (2.2a)$$

$$\underline{y}(t_k) = \underline{h}(t_k, \underline{x}(t_k), \underline{\theta}, \underline{u}(t_k)) + \underline{\eta}(t_k). \quad (2.2b)$$

Hier beschreiben wie im klassischen Systemmodell (2.1) \underline{x} den Systemzustand, $\underline{\theta}$ die Modellparameter, \underline{u} die Stellgrößen und \underline{y} den Messwertvektor. Sowohl der Parametersatz $\underline{\theta}$ als auch der Startzustand \underline{x}_0 oder zumindest Teile des Startzustandes sind üblicherweise jedoch nur ungenau bekannt und werden hier als zufällige Größen angenommen. Basierend auf der vereinfachenden Annahme, dass sowohl die Unsicherheiten der Modellparameter als auch die des Startzustandes normalverteilt sind, können diese mit

$$\underline{\theta} \sim \mathcal{N}(\bar{\underline{\theta}}, \mathbf{C}_{\underline{\theta}}) \quad \underline{x}_0 \sim \mathcal{N}(\bar{\underline{x}}_0, \mathbf{C}_{\underline{x}_0}) \quad (2.3)$$

2. Mathematische Methoden

beschrieben werden. Der zusätzliche Term $\dot{\underline{\xi}}$ in (2.2) beschreibt das so genannte Systemrauschen, einen als mittelwertfreien normalverteilten und weiß angenommenen Rauschprozess

$$\dot{\underline{\xi}} \sim \mathcal{N}(\underline{0}, \mathbf{C}_{\dot{\underline{\xi}}}), \quad (2.4)$$

der strukturelle Unsicherheiten des Modells wiedergibt. Der hier ebenfalls als normalverteilt weiß beschriebene Rauschprozess

$$\underline{\eta}(t_k) \sim \mathcal{N}(\underline{0}, \mathbf{C}_{\underline{\eta}}), \quad (2.5)$$

beinhaltet das Messrauschen.

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Ausgehend von den unsicheren Einflussgrößen \underline{x}_0 , $\underline{\theta}$, $\dot{\underline{\xi}}$ und $\underline{\eta}$ ist es für später beschriebene Prozessplanungs- und -führungsverfahren notwendig, die statistischen Momente der Zustandstrajektorien $\underline{x}_k = \underline{x}(t_k)$, der Messgrößentrajektorien $\underline{y}_k = \underline{y}(t_k)$ und weiterer daraus abgeleiteter Funktionen an einem oder mehreren betrachteten Zeitpunkten t_k , $k = 1, \dots, N$ zu berechnen. Zur besseren Veranschaulichung sollen die verschiedenen Verfahren jeweils zunächst mit Hilfe einer allgemeinen Funktion

$$\underline{b} = g(\underline{a}), \quad \underline{a} \in \mathbb{R}^{L_a}, \quad \underline{b} \in \mathbb{R}^{L_b}, \quad \underline{a} \sim \mathcal{N}(\bar{\underline{a}}, \mathbf{C}_a), \quad (2.6)$$

deren unabhängige Größe \underline{a} normalverteilt ist, dargestellt und die vorgestellten Methoden anschließend auf die Lösung des Zustandsraummodells (2.2) angewendet werden.

Im Rahmen dieser Arbeit werden dabei nur Verfahren verwendet, die auf der Beschreibung der Verteilungen mit zwei statistischen Momenten beruhen. Erweiterungen der beschriebenen Verfahren auf nicht Gauß'sche Verteilungsfunktionen sind möglich und Teil aktueller Forschung. Einige der beschriebenen Verfahren lassen sich dabei direkt auf höhere statistische Momente erweitern, andere benötigen Beschreibungen, die auf der Superposition von z.B. Gaußverteilungen bestehen.

2.1.1. Monte-Carlo-Simulation

Durch den Einsatz von Monte-Carlo-Verfahren ist im Grenzwert für unendlich viele Simulationen eine erwartungstreue Schätzung aller statistischen Momente möglich [Sha00].

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Ausgehend von $N_{\text{MC}} \gg 1$ Auswertungen der Funktion g mit N_{MC} zufällig aus der Verteilung von \underline{a} gezogenen Werten $\underline{a}^{(i)}, i = 1, \dots, N_{\text{MC}}$, können der Mittelwert $\bar{\underline{b}} = E\{\underline{b}\}$ und die Kovarianzmatrix $\mathbf{C}_{\underline{b}} = E\{(\underline{b} - \bar{\underline{b}})(\underline{b} - \bar{\underline{b}})^T\}$ von \underline{b} durch

$$\begin{aligned} \bar{\underline{b}} &\approx \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} g(\underline{a}^{(i)}) = \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} \underline{b}^{(i)} \quad \text{und} \\ \mathbf{C}_{\underline{b}} &\approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} (\underline{b}^{(i)} - \bar{\underline{b}}) (\underline{b}^{(i)} - \bar{\underline{b}})^T \end{aligned} \quad (2.7)$$

geschätzt werden. Für die Kreuzkovarianz $\mathbf{C}_{\underline{a}\underline{b}} = E\{(\underline{a} - \bar{\underline{a}})(\underline{b} - \bar{\underline{b}})^T\}$ gilt

$$\mathbf{C}_{\underline{a}\underline{b}} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} (\underline{a}^{(i)} - \bar{\underline{a}}) (\underline{b}^{(i)} - \bar{\underline{b}})^T. \quad (2.8)$$

Angewendet auf das Systemmodell (2.2) ergibt sich für die Berechnung der statistischen Momente einer Prozesstrajektorie folgendes Vorgehen.

1. Zunächst werden zufällig jeweils N_{MC} Werte $\underline{x}_0^{(i)}, \underline{\theta}^{(i)}, \underline{\xi}^{(i)}$ und $\underline{\eta}^{(i)}, i = 1, \dots, N_{\text{MC}}$ aus den statistischen Verteilungen des Anfangszustandes, des Parametersatzes, des Systemrauschens und des Messrauschens gezogen.
2. Durch jeweiliges Einsetzen der Werte $(\cdot)^{(i)}$ und Lösung der Modellgleichung werden die Zustandsverläufe $\underline{x}_k^{(i)}$ und die Messgrößenverläufe $\underline{y}_k^{(i)}$ an den betrachteten Zeitpunkten $t_k, k = 1, \dots, N$ berechnet.
3. Anwendung der Approximationsformeln (2.7) liefert die statistischen Momente

$$\bar{\underline{x}}_k \approx \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} \underline{x}_k^{(i)} \quad \mathbf{C}_{\underline{x}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} (\underline{x}_k^{(i)} - \bar{\underline{x}}_k) (\underline{x}_k^{(i)} - \bar{\underline{x}}_k)^T \quad (2.9a)$$

$$\bar{\underline{y}}_k \approx \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} \underline{y}_k^{(i)} \quad \mathbf{C}_{\underline{y}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} (\underline{y}_k^{(i)} - \bar{\underline{y}}_k) (\underline{y}_k^{(i)} - \bar{\underline{y}}_k)^T \quad (2.9b)$$

2. Mathematische Methoden

und die Kreuzkovarianzen

$$\mathbf{C}_{\underline{x}_0 \underline{x}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \left(\underline{x}_0^{(i)} - \bar{\underline{x}}_0 \right) \left(\underline{x}_k^{(i)} - \bar{\underline{x}}_k \right)^{\text{T}} \quad (2.9\text{c})$$

$$\mathbf{C}_{\underline{\theta} \underline{x}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \left(\underline{\theta}^{(i)} - \bar{\underline{\theta}} \right) \left(\underline{x}_k^{(i)} - \bar{\underline{x}}_k \right)^{\text{T}} \quad (2.9\text{d})$$

$$\mathbf{C}_{\underline{\xi} \underline{x}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \underline{\xi}^{(i)} \left(\underline{x}_k^{(i)} - \bar{\underline{x}}_k \right)^{\text{T}} \quad (2.9\text{e})$$

$$\mathbf{C}_{\underline{x}_0 \underline{y}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \left(\underline{x}_0^{(i)} - \bar{\underline{x}}_0 \right) \left(\underline{y}_k^{(i)} - \bar{\underline{y}}_k \right)^{\text{T}} \quad (2.9\text{f})$$

$$\mathbf{C}_{\underline{\theta} \underline{y}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \left(\underline{\theta}^{(i)} - \bar{\underline{\theta}} \right) \left(\underline{y}_k^{(i)} - \bar{\underline{y}}_k \right)^{\text{T}} \quad (2.9\text{g})$$

$$\mathbf{C}_{\underline{\xi} \underline{y}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \underline{\xi}^{(i)} \left(\underline{y}_k^{(i)} - \bar{\underline{y}}_k \right)^{\text{T}} \quad (2.9\text{h})$$

$$\mathbf{C}_{\underline{\eta} \underline{y}_k} \approx \frac{1}{N_{\text{MC}} - 1} \sum_{i=1}^{N_{\text{MC}}} \underline{\eta}^{(i)} \left(\underline{y}_k^{(i)} - \bar{\underline{y}}_k \right)^{\text{T}}. \quad (2.9\text{i})$$

Für eine hinreichend genaue Schätzung der statistischen Eigenschaften ist eine sehr große Anzahl von Simulationen notwendig. Vor allem schon der daraus resultierende hohe numerische Aufwand der Monte Carlo Methode verbietet den Einsatz im Rahmen der Prozessführung. Zusätzlich enthält der Algorithmus einen zufälligen Anteil, das zufällige Ziehen der Eingangsgrößen aus deren jeweiliger Verteilung, was den Einsatz leistungsfähiger gradientenbasierter Optimierungsverfahren stark erschwert.

2.1.2. Approximation durch Taylor-Reihenentwicklung

Da, wie beschrieben, der Einsatz der Monte-Carlo-Methode sehr aufwändig ist, ist es sinnvoll, Approximationsverfahren zur vereinfachten Darstellung der Abbildungsfunktion \underline{g} bei der Berechnung der statistischen Momente zu verwenden. Das bekannteste Verfahren zur Approximation nichtlinearer Abbildungen ist die Taylor-Reihenentwicklung. Entwickelt man die allgemeine Funktion \underline{g} in eine Taylor-Reihe erster Ordnung um $\bar{\underline{a}}$

$$\underline{b} \approx \underline{g}(\bar{\underline{a}}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{\underline{a}}} (\underline{a} - \bar{\underline{a}}), \quad (2.10)$$

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

kann damit eine näherungsweise Berechnung der ersten beiden Momente erfolgen [KDM⁺05]. Der Mittelwert von \underline{b} ergibt sich zu

$$\bar{b} = E\{b\} \quad (2.11a)$$

$$\approx E \left\{ \underline{g}(\bar{a}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} (\underline{a} - \bar{a}) \right\} \quad (2.11b)$$

$$= \underline{g}(\bar{a}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} E\{(\underline{a} - \bar{a})\} \quad (2.11c)$$

$$= \underline{g}(\bar{a}). \quad (2.11d)$$

Die approximative Berechnung der Kovarianzmatrix ist mit

$$\mathbf{C}_b = E\{(\underline{b} - \bar{b})(\underline{b} - \bar{b})^T\} \quad (2.12a)$$

$$\approx E \left\{ \left(\underline{g}(\bar{a}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} (\underline{a} - \bar{a}) - \underline{g}(\bar{a}) \right) \right. \quad (2.12b)$$

$$\left. \left(\underline{g}(\bar{a}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} (\underline{a} - \bar{a}) - \underline{g}(\bar{a}) \right)^T \right\}$$

$$= \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} E\{(\underline{a} - \bar{a})(\underline{a} - \bar{a})^T\} \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}}^T \quad (2.12c)$$

$$= \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} \mathbf{C}_a \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}}^T \quad (2.12d)$$

möglich. Die Kreuzkovarianz zwischen Eingangs- und Ausgangsgröße lässt sich mit

$$\mathbf{C}_{ab} = E\{(\underline{a} - \bar{a})(\underline{b} - \bar{b})^T\} \quad (2.13a)$$

$$\approx E \left\{ (\underline{a} - \bar{a}) \left(\underline{g}(\bar{a}) + \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}} (\underline{a} - \bar{a}) - \underline{g}(\bar{a}) \right)^T \right\} \quad (2.13b)$$

$$= E\{(\underline{a} - \bar{a})(\underline{a} - \bar{a})^T\} \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}}^T \quad (2.13c)$$

$$= \mathbf{C}_a \left. \frac{d\underline{g}(\underline{a})}{d\underline{a}} \right|_{\bar{a}}^T \quad (2.13d)$$

approximativ berechnen. Zur Berechnung des Mittelwertes einer Trajektorie sind somit nur die entsprechenden Gleichungen mit den Mittelwerten der unsicheren Eingangsgrößen $\underline{x}_0 = \bar{\underline{x}}_0$, $\underline{\theta} = \bar{\underline{\theta}}$, $\underline{\xi} = \underline{0}$ und $\underline{\eta} = \underline{0}$ auszuwerten. Für die Berechnung der verschiedenen Kovarianzmatrizen müssen zunächst die Sensitivitäten \mathbf{Z} der verschiedenen Ausgangsgrößen bezüglich der unsicheren Eingangsgrößen berechnet werden. Für die Zustandstrajektorie sind an allen betrachteten Abtastzeitpunkten t_k , $k = 1, \dots, N$ die

2. Mathematische Methoden

Ableitungen

$$\begin{aligned} \mathbf{Z}_{\underline{x}_k \underline{x}_0} &= \left. \frac{d\underline{x}}{d\underline{x}_0} \right|_{t_k, (\bar{\cdot})}, & \mathbf{Z}_{\underline{x}_k \underline{\theta}} &= \left. \frac{d\underline{x}}{d\underline{\theta}} \right|_{t_k, (\bar{\cdot})}, \\ \mathbf{Z}_{\underline{x}_k \underline{\xi}} &= \left. \frac{d\underline{x}}{d\underline{\xi}} \right|_{t_k, (\bar{\cdot})} \quad \text{und} & \mathbf{Z}_{\underline{x}_k \underline{\eta}} &= \mathbf{0} \end{aligned} \quad (2.14)$$

zu bestimmen, wobei der Auswertungspunkt $(\bar{\cdot})$ die Mittelwerte der Verteilungen der unsicheren Eingangsgrößen \underline{x}_0 , $\underline{\theta}$, $\underline{\xi}$ und $\underline{\eta}$ markiert. Die Stellgröße $\underline{u}(t)$ wird als deterministischer Wert angenommen, weswegen sie hier nicht explizit angeführt wird. Da \underline{x}_k bereits die üblicherweise numerisch berechnete Lösung des Differentialgleichungssystems (2.2a) darstellt, ist die Berechnung dieser Ableitungen nicht ohne weiteres möglich. Grundsätzlich besteht die Möglichkeit, diese Ableitungen approximativ mit Hilfe von Differenzenquotienten zu berechnen. Eine solche numerische Differentiation beinhaltet jedoch aufgrund der unterliegenden numerischen Integration weitere Fehlerquellen, die in Abschnitt 2.6.2 noch genauer erklärt werden. Ein anderer Ansatz zur Berechnung der Sensitivitäten $\mathbf{Z}_{\underline{x}_k \underline{a}}$ des Zustandes bezüglich einer Einflussgröße \underline{a} , die für \underline{x}_0 , $\underline{\theta}$ oder $\underline{\xi}$ steht, lässt sich durch Ableitung der Modellgleichungen (2.2a) nach der entsprechenden Einflussgröße

$$\left. \frac{d\underline{\dot{x}}}{d\underline{a}} \right|_{t, (\bar{\cdot})} = \left. \frac{d}{d\underline{a}} \left(\underline{f}(t, \underline{x}, \underline{\theta}, \underline{u}) + \underline{\dot{\xi}} \right) \right|_{t, (\bar{\cdot})} \quad (2.15)$$

herleiten. Vertauschen der Differentiationsreihenfolge und Anwendung der Kettenregel liefern die Matrixdifferentialgleichung

$$\underbrace{\frac{d}{dt} \frac{d\underline{x}}{d\underline{a}} \Big|_{t, (\bar{\cdot})}}_{\dot{\mathbf{Z}}_{\underline{x} \underline{a}}} = \frac{\partial \underline{f}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{x}} \Big|_{t, (\bar{\cdot})} \underbrace{\frac{d\underline{x}}{d\underline{a}} \Big|_{t, (\bar{\cdot})}}_{\mathbf{Z}_{\underline{x} \underline{a}}} + \frac{\partial \left(\underline{f}(t, \underline{x}, \underline{\theta}, \underline{u}) + \underline{\dot{\xi}}(t) \right)}{\partial \underline{a}} \Big|_{t, (\bar{\cdot})}, \quad (2.16)$$

mit der Anfangsbedingung

$$\mathbf{Z}_{\underline{x} \underline{a}}(t=0) = \mathbf{Z}_{\underline{x}_0 \underline{a}} = \left. \frac{d\underline{x}_0}{d\underline{a}} \right|_{(\bar{\cdot})}, \quad (2.17)$$

wobei $\mathbf{Z}_{\underline{x}_k \underline{a}}$ die Lösung der Matrixdifferentialgleichung (2.16) zum Zeitpunkt $t = t_k$ bezeichnet. Substitution der hier verwendeten Hilfsvariable \underline{a} durch die verschiedenen unsicheren Eingangsgrößen liefert die parallel zu den Modellgleichungen (2.2a) zu lösenden Matrixdifferentialgleichungssysteme

$$\dot{\mathbf{Z}}_{\underline{x} \underline{x}_0} = \left. \frac{\partial \underline{f}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{x}} \right|_{t, (\bar{\cdot})} \mathbf{Z}_{\underline{x} \underline{x}_0}, \quad \mathbf{Z}_{\underline{x} \underline{x}_0}(t=0) = \mathbf{I} \quad (2.18a)$$

$$\dot{\mathbf{Z}}_{\underline{x} \underline{\theta}} = \left. \frac{\partial \underline{f}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{x}} \right|_{t, (\bar{\cdot})} \mathbf{Z}_{\underline{x} \underline{\theta}} + \left. \frac{\partial \underline{f}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{\theta}} \right|_{t, (\bar{\cdot})}, \quad \mathbf{Z}_{\underline{x} \underline{\theta}}(t=0) = \left. \frac{d\underline{x}_0}{d\underline{\theta}} \right|_{(\bar{\cdot})} \quad (2.18b)$$

$$\dot{\mathbf{Z}}_{\underline{x} \underline{\xi}} = \left. \frac{\partial \underline{f}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{x}} \right|_{t, (\bar{\cdot})} \mathbf{Z}_{\underline{x} \underline{\xi}} + \mathbf{I}, \quad \mathbf{Z}_{\underline{x} \underline{\xi}}(t=0) = \mathbf{0}. \quad (2.18c)$$

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Die Kovarianzmatrix der Zustandstrajektorie an den Abtastzeitpunkten t_k berechnet sich folglich als

$$\mathbf{C}_{\underline{x}_k} \approx \mathbf{Z}_{\underline{x}_k \underline{x}_0} \cdot \mathbf{C}_{\underline{x}_0} \cdot \mathbf{Z}_{\underline{x}_k \underline{x}_0}^\top + \mathbf{Z}_{\underline{x}_k \underline{\theta}} \cdot \mathbf{C}_{\underline{\theta}} \cdot \mathbf{Z}_{\underline{x}_k \underline{\theta}}^\top + \mathbf{Z}_{\underline{x}_k \underline{\xi}} \cdot \mathbf{C}_{\underline{\xi}} \cdot \mathbf{Z}_{\underline{x}_k \underline{\xi}}^\top. \quad (2.19)$$

Die Kreuzkovarianzen des Zustandes bezüglich der unsicheren Größen ergeben sich zu

$$\mathbf{C}_{\underline{x}_0 \underline{x}_k} \approx \mathbf{C}_{\underline{x}_0} \cdot \mathbf{Z}_{\underline{x}_k \underline{x}_0}^\top \quad \mathbf{C}_{\underline{\theta} \underline{x}_k} \approx \mathbf{C}_{\underline{\theta}} \cdot \mathbf{Z}_{\underline{x}_k \underline{\theta}}^\top \quad \mathbf{C}_{\underline{\xi} \underline{x}_k} \approx \mathbf{C}_{\underline{\xi}} \cdot \mathbf{Z}_{\underline{x}_k \underline{\xi}}^\top. \quad (2.20)$$

Die Sensitivitäten anderer aus dem Zustand abgeleiteter Ausgangsgrößen können mit Hilfe der Kettenregel aus den mit (2.18) berechneten Werten ermittelt werden. Mit (2.2b) folgen für die Messgrößentrajektorie

$$\mathbf{Z}_{\underline{y}_k \underline{x}_0} = \left. \frac{d\underline{y}}{d\underline{x}_0} \right|_{t_k, (\bar{\cdot})} = \left. \frac{\partial \underline{h}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{x}} \right|_{t_k, (\bar{\cdot})} \mathbf{Z}_{\underline{x}_k \underline{x}_0} \quad (2.21a)$$

$$\mathbf{Z}_{\underline{y}_k \underline{\theta}} = \left. \frac{d\underline{y}}{d\underline{\theta}} \right|_{t_k, (\bar{\cdot})} = \left. \frac{\partial \underline{h}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{\theta}} \right|_{t_k, (\bar{\cdot})} \mathbf{Z}_{\underline{x}_k \underline{\theta}} + \left. \frac{\partial \underline{h}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{\theta}} \right|_{t_k, (\bar{\cdot})} \quad (2.21b)$$

$$\mathbf{Z}_{\underline{y}_k \underline{\xi}} = \left. \frac{d\underline{y}}{d\underline{\xi}} \right|_{t_k, (\bar{\cdot})} = \left. \frac{\partial \underline{h}(t, \underline{x}, \underline{\theta}, \underline{u})}{\partial \underline{\xi}} \right|_{t_k, (\bar{\cdot})} \mathbf{Z}_{\underline{x}_k \underline{\xi}} \quad (2.21c)$$

$$\mathbf{Z}_{\underline{y}_k \underline{\eta}} = \mathbf{I} \quad (2.21d)$$

und somit deren Kovarianzmatrix

$$\mathbf{C}_{\underline{y}_k} \approx \mathbf{Z}_{\underline{y}_k \underline{x}_0} \cdot \mathbf{C}_{\underline{x}_0} \cdot \mathbf{Z}_{\underline{y}_k \underline{x}_0}^\top + \mathbf{Z}_{\underline{y}_k \underline{\theta}} \cdot \mathbf{C}_{\underline{\theta}} \cdot \mathbf{Z}_{\underline{y}_k \underline{\theta}}^\top + \mathbf{Z}_{\underline{y}_k \underline{\xi}} \cdot \mathbf{C}_{\underline{\xi}} \cdot \mathbf{Z}_{\underline{y}_k \underline{\xi}}^\top + \mathbf{C}_{\underline{\eta}} \quad (2.22)$$

und die Kreuzkovarianzen

$$\begin{aligned} \mathbf{C}_{\underline{x}_0 \underline{y}_k} &\approx \mathbf{C}_{\underline{x}_0} \cdot \mathbf{Z}_{\underline{y}_k \underline{x}_0}^\top & \mathbf{C}_{\underline{\theta} \underline{y}_k} &\approx \mathbf{C}_{\underline{\theta}} \cdot \mathbf{Z}_{\underline{y}_k \underline{\theta}}^\top \\ \mathbf{C}_{\underline{\xi} \underline{y}_k} &\approx \mathbf{C}_{\underline{\xi}} \cdot \mathbf{Z}_{\underline{y}_k \underline{\xi}}^\top & \mathbf{C}_{\underline{\eta} \underline{y}_k} &= \mathbf{C}_{\underline{\eta}}. \end{aligned} \quad (2.23)$$

Die Gleichheit in der Kreuzkovarianz der Messgrößen bezüglich des Messrauschens ergibt sich aus der Annahme des additiven Rauschens in (2.2b), wodurch die Approximation erster Ordnung exakt ist.

Es sei darauf hingewiesen, dass diese Berechnungen auf Basis der Kettenregel nicht uneingeschränkt anwendbar sind, wenn eine der beteiligten Funktionen nicht stetig nach dem Zustand \underline{x} differenzierbar ist, wie es zum Beispiel für eine Modellgleichung der Form $\dot{x} = \sqrt{x}$ der Fall wäre. Die resultierenden Matrix-Differentialgleichungssysteme sind zusätzlich sehr groß, oft steif und somit schwierig numerisch zu lösen, jedoch üblicherweise nur schwach gekoppelt. Hier ist darauf zu achten, ein numerisches Lösungsverfahren zu verwenden, welches für solche Systeme geeignet ist.

Das Bild 2.1 zeigt die Anwendung der Monte-Carlo-Simulation und der Taylor-Approximation erster Ordnung für die Abbildung einer Eingangsgröße $\underline{x} \in \mathbb{R}^2$ durch die nicht-lineare Funktion $\underline{y} = (\sin(x_1 x_2), (x_1 - x_2)^3)^\top$. Veranschaulicht ist oben die Verteilung

2. Mathematische Methoden

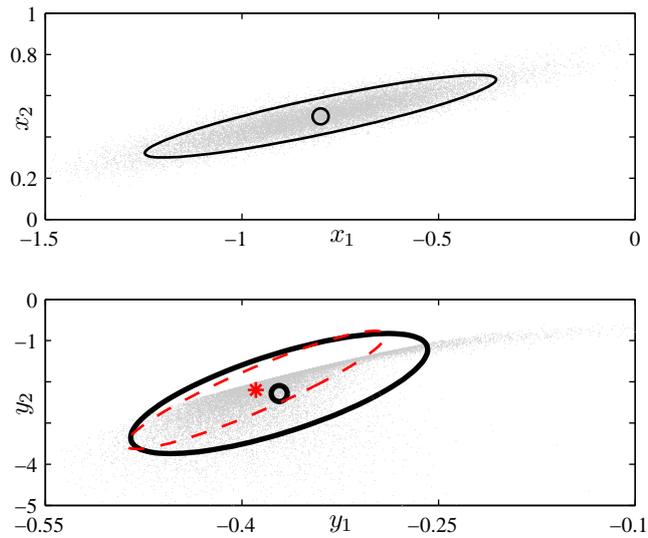


Abbildung 2.1.: Ergebnis der Monte-Carlo-Simulation (graue Punkte) für die einfache Funktion mit Mittelwert und 2σ -Grenze in schwarz (o Mittelwert, - 2σ -Grenze) im Vergleich zum Ergebnis der Approximation erster Ordnung in rot (* Mittelwert, - - - 2σ -Grenze).

der unabhängigen Größe \underline{x} über eine Punktwolke, den Mittelwert und die 2σ -Grenze. Abbildung dieser Eingangsgröße durch die nichtlineare Funktion liefert die unten dargestellte Punktwolke. Die Ellipsen ergeben sich aus der 2σ -Grenze der Verteilung von \underline{y} , berechnet aus den ersten zwei statistischen Momenten resultierend aus der Monte Carlo Simulation mit $N_{MC} = 20000$ Einzelwerten und der Approximation erster Ordnung. Aus der Punktwolke, welche die Ausgangsgrößen darstellt, ist ersichtlich, dass die resultierende Verteilung keine Normalverteilung ist. Hier sind jedoch nur die ersten zwei Momente berechnet. Aufgrund der hohen Anzahl Einzelberechnungen in der Monte Carlo Simulation ist zu erwarten, dass die dargestellte 2σ -Grenze nahezu der wahren Lösung für die Kovarianzmatrix entspricht. Die Ergebnisse der Approximation erster Ordnung zeigen Fehler sowohl im Mittelwert, als auch bei der berechneten Kovarianzmatrix.

2.1.3. Unscented-Transformation

Im Allgemeinen ist davon auszugehen, dass bei Verwendung einer Approximation zweiter oder höherer Ordnung die Approximationsgenauigkeit gegenüber der beschriebenen Approximation erster Ordnung steigt. Bei Verwendung einer Taylorreihenentwicklung ist in diesen Fällen jedoch eine zum Teil extrem aufwändige analytische Berechnung von Funktionsableitungen nötig. Zusätzlich zu den ersten Ableitungen, die angewendet auf das Differentialgleichungssystem (2.2a) bereits in Matrixdifferentialgleichungen resultierten, wäre die Berechnung höherer Ableitungen nötig.

Die Unscented-Transformation [JU04, Mer04] basiert auf der Auswertung einer festen Anzahl von Punkten und benötigt keine Ableitungen. Diese klassische Unscented-Transformation approximiert dazu die Eingangsverteilung mit Hilfe von Sigma-Punkten, bildet diese ab und kombiniert die Resultate zu einer Ausgangsverteilung.

In letzter Zeit wird vorgeschlagen, anstelle des Tayloransatzes die Stirling-Polynom-Interpolationsformel [Sti30, Haz02, NPR00a, Mer04] anzuwenden. Hierbei werden die Funktionsableitungen durch eine begrenzte Anzahl von Funktionsberechnungen ersetzt. Als Ergebnis erhält man die Central-Difference-Form der Unscented-Transformation, die für eine Approximation 2. Ordnung im Folgenden eingeführt wird.

Im skalaren Fall der oben eingeführten allgemeinen Funktion (Gl. 2.6) $b = g(a)$ ergibt sich für eine Beschreibung zweiter Ordnung mit einer Taylorreihe

$$g(a) \approx g(\bar{a}) + \left. \frac{dg}{da} \right|_{\bar{a}} (a - \bar{a}) + \frac{1}{2!} \left. \frac{d^2g}{da^2} \right|_{\bar{a}} (a - \bar{a})^2 \quad (2.24)$$

und mit einem Stirling-Polynom

$$\approx g(\bar{a}) + D_{\Delta a} g + \frac{1}{2!} D_{\Delta a}^2 g, \quad (2.25)$$

wobei $D_{\Delta a}$ und $D_{\Delta a}^2$ Differenzenoperatoren darstellen. Bei Verwendung von zentralen Differenzenquotienten können diese Operatoren durch

$$D_{\Delta a} g = \frac{g(\bar{a} + h) - g(\bar{a} - h)}{2h} (a - \bar{a}) \quad (2.26)$$

$$D_{\Delta a}^2 g = \frac{g(\bar{a} + h) + g(\bar{a} - h) - 2g(\bar{a})}{h^2} (a - \bar{a})^2 \quad (2.27)$$

definiert werden. h stellt hier die Schrittweite für die Differenzenquotienten dar. Zur Erweiterung der Ausdrücke auf den mehrdimensionalen Fall ist es zunächst sinnvoll, mit Hilfe der Transformation

$$\underline{z} = \mathbf{S}_a^{-1} \underline{a} \quad (2.28)$$

2. Mathematische Methoden

eine statistische Entkopplung des Systemeingangs \underline{a} durchzuführen. $\mathbf{S}_{\underline{a}}$ ist hierbei die Cholesky-Zerlegung der Kovarianzmatrix $\mathbf{C}_{\underline{a}}$, so dass folgt $\mathbf{C}_{\underline{a}} = \mathbf{S}_{\underline{a}}\mathbf{S}_{\underline{a}}^T$. Hierdurch gilt im entkoppelten transformierten System

$$\mathbf{C}_{\underline{z}} = E \{ (\underline{z} - \bar{\underline{z}})(\underline{z} - \bar{\underline{z}})^T \} = \mathbf{I}. \quad (2.29)$$

Anstelle der Cholesky-Zerlegung wären auch andere Matrixwurzeln möglich. Einsetzen der Koordinatentransformation von \underline{a} in \underline{z} in die Abbildung \underline{g} liefert

$$\tilde{\underline{g}}(\underline{z}) = \underline{g}(\mathbf{S}_{\underline{a}}\underline{z}) = \underline{g}(\underline{a}). \quad (2.30)$$

Anwendung der skalaren Differenzenquotienten für die jetzt statistisch unabhängigen Komponenten in \underline{z} ergibt

$$\underline{b} = \tilde{\underline{g}}(\underline{z}) \approx \tilde{\underline{g}}(\bar{\underline{z}}) + \mathbf{D}_{\Delta\underline{z}}\tilde{\underline{g}} + \frac{1}{2!}\mathbf{D}_{\Delta\underline{z}}^2\tilde{\underline{g}} \quad (2.31)$$

mit den mehrdimensionalen Differenzenoperatoren

$$\begin{aligned} \mathbf{D}_{\Delta\underline{z}}\tilde{\underline{g}} &= \left(d_1\tilde{\underline{g}} \cdots d_{L_a}\tilde{\underline{g}} \right) \Delta\underline{z} \\ &= \sum_{i=1}^{L_a} d_i\tilde{\underline{g}} \Delta z_i \end{aligned} \quad (2.32a)$$

$$\begin{aligned} \mathbf{D}_{\Delta\underline{z}}^2\tilde{\underline{g}} &= \mathbf{D}_{\Delta\underline{z}} \left(\mathbf{D}_{\Delta\underline{z}}\tilde{\underline{g}} \right) \\ &= \sum_{i=1}^{L_a} d_i^2\tilde{\underline{g}} \Delta z_i^2 + \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} d_i d_j \tilde{\underline{g}} \Delta z_i \Delta z_j \end{aligned} \quad (2.32b)$$

und den partiellen Differenzenoperatoren

$$d_i\tilde{\underline{g}} = \frac{1}{2h_i} \left[\tilde{\underline{g}}(\bar{\underline{z}} + h_i\mathbf{e}_i) - \tilde{\underline{g}}(\bar{\underline{z}} - h_i\mathbf{e}_i) \right] \quad (2.33a)$$

$$d_i^2\tilde{\underline{g}} = \frac{1}{h_i^2} \left[\tilde{\underline{g}}(\bar{\underline{z}} + h_i\mathbf{e}_i) + \tilde{\underline{g}}(\bar{\underline{z}} - h_i\mathbf{e}_i) - 2\tilde{\underline{g}}(\bar{\underline{z}}) \right]. \quad (2.33b)$$

\mathbf{e}_i stellt den i -ten Einheitsvektor dar, h_i die Schrittweite des Differenzenquotienten in der i -ten Dimension.

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Ausgehend von diesen Vorbetrachtungen kann der Mittelwert

$$\bar{b} = E \{b\} \quad (2.34a)$$

von \underline{b} berechnet werden. Einsetzen von (2.31) liefert

$$\bar{b} \approx E \left\{ \tilde{g}(\underline{z}) + \mathbf{D}_{\Delta \underline{z}} \tilde{g} + \frac{1}{2} \mathbf{D}_{\Delta \underline{z}}^2 \tilde{g} \right\}. \quad (2.34b)$$

Aus $E\{z - \bar{z}\} = 0$ folgt $E\{\mathbf{D}_{\Delta \underline{z}} \tilde{g}\} = 0$. Somit gilt

$$= \tilde{g}(\bar{z}) + \frac{1}{2} E \left\{ \mathbf{D}_{\Delta \underline{z}}^2 \tilde{g} \right\}. \quad (2.34c)$$

Substitution von $\mathbf{D}_{\Delta \underline{z}}^2 \tilde{g}$ mit (2.32b) unter Beachtung der statistischen Entkopplung der Elemente von \underline{z} ergibt

$$= \tilde{g}(\bar{z}) + \frac{1}{2} E \left\{ \sum_{i=1}^{L_a} d_i^2 \tilde{g} \Delta z_i^2 \right\}. \quad (2.34d)$$

Mit $E\{\Delta z_i^2\} = 1$ wegen Gl. (2.29) und (2.33b) folgt schließlich

$$= \tilde{g}(\bar{z}) + \frac{1}{2} \sum_{i=1}^{L_a} \frac{\tilde{g}(\bar{z} + h_i \underline{e}_i) + \tilde{g}(\bar{z} - h_i \underline{e}_i) - 2\tilde{g}(\bar{z})}{h_i^2}. \quad (2.34e)$$

Die Rücktransformation nach Gl. (2.28) liefert

$$\tilde{g}(\bar{z}) = \underline{g}(\mathbf{S}_a \bar{z}) = \underline{g}(\bar{a}) \quad (2.35a)$$

$$\tilde{g}(\bar{z} \pm h_i \underline{e}_i) = \underline{g}(\mathbf{S}_a (\bar{z} \pm h_i \underline{e}_i)) = \underline{g}(\bar{a} \pm h_i \overbrace{\mathbf{S}_a \underline{e}_i}^{s_a^{(i)}}). \quad (2.35b)$$

Die Argumente $\bar{a} \pm h_i s_{a_i}$ werden als Sigmapunkte bezeichnet [JU04]. Anwendung der Rücktransformation (2.28) auf Gl. (2.34e) führt zu

$$\bar{b} \approx \underline{g}(\bar{a}) + \frac{1}{2} \sum_{i=1}^{L_a} \frac{\underline{g}(\bar{a} + h_i s_a^{(i)}) + \underline{g}(\bar{a} - h_i s_a^{(i)}) - 2\underline{g}(\bar{a})}{h_i^2}, \quad (2.36)$$

das sich im Falle von identischer Schrittweite in allen Dimensionen ($h_i = h \forall i$) zu

$$\bar{b} \approx \frac{h^2 - L_a}{h^2} \underline{g}(\bar{a}) + \frac{1}{2h^2} \sum_{i=1}^{L_a} \left(\underline{g}(\bar{a} + h s_a^{(i)}) + \underline{g}(\bar{a} - h s_a^{(i)}) \right) \quad (2.37)$$

2. Mathematische Methoden

vereinfachen lässt. Analog kann die Kovarianzmatrix $\mathbf{C}_{\underline{b}} = E \{ (\underline{b} - \bar{\underline{b}})(\underline{b} - \bar{\underline{b}})^T \}$ gebildet werden. Mit der Identität

$$\bar{\underline{b}} = E\{\underline{b}\} = \tilde{\underline{g}}(\bar{\underline{z}}) + E\{\underline{b} - \tilde{\underline{g}}(\bar{\underline{z}})\} \quad (2.38)$$

folgt

$$\mathbf{C}_{\underline{b}} = E \left\{ (\underline{b} - \tilde{\underline{g}}(\bar{\underline{z}})) (\underline{b} - \tilde{\underline{g}}(\bar{\underline{z}}))^T \right\} - E \left\{ (\underline{b} - \tilde{\underline{g}}(\bar{\underline{z}})) \right\} E \left\{ (\underline{b} - \tilde{\underline{g}}(\bar{\underline{z}}))^T \right\} \quad (2.39a)$$

$$\begin{aligned} &\approx E \left\{ \left(\mathbf{D}_{\Delta z} \tilde{\underline{g}} + \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right) \left(\mathbf{D}_{\Delta z} \tilde{\underline{g}} + \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right)^T \right\} \\ &- E \left\{ \mathbf{D}_{\Delta z} \tilde{\underline{g}} + \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right\} E \left\{ \mathbf{D}_{\Delta z} \tilde{\underline{g}} + \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right\}^T \end{aligned} \quad (2.39b)$$

Ausmultiplizieren und Anwendung von $E\{\mathbf{D}_{\Delta z} \tilde{\underline{g}}\} = \underline{0}$ liefert

$$\begin{aligned} &= E \left\{ \mathbf{D}_{\Delta z} \tilde{\underline{g}} (\mathbf{D}_{\Delta z} \tilde{\underline{g}})^T \right\} + E \left\{ \mathbf{D}_{\Delta z} \tilde{\underline{g}} \left(\frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right)^T \right\} + E \left\{ \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} (\mathbf{D}_{\Delta z} \tilde{\underline{g}})^T \right\} \\ &+ E \left\{ \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \left(\frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right)^T \right\} - E \left\{ \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right\} E \left\{ \frac{1}{2} \mathbf{D}_{\Delta z}^2 \tilde{\underline{g}} \right\}^T. \end{aligned} \quad (2.39c)$$

Da alle ungeraden Momente einer Normalverteilung null sind, entfallen die Summanden 2 und 3. Einsetzen von (2.32) unter Beachtung der statistischen Entkopplung der Elemente von \underline{z} ergibt

$$\begin{aligned} &= E \left\{ \left(\sum_{i=1}^{L_a} \underline{d}_i \tilde{\underline{g}} \Delta z_i \right) \left(\sum_{i=1}^{L_a} \underline{d}_i \tilde{\underline{g}} \Delta z_i \right)^T \right\} \\ &+ \frac{1}{4} E \left\{ \left(\sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{\underline{g}} \Delta z_i^2 + \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} \underline{d}_i \underline{d}_j \tilde{\underline{g}} \Delta z_i \Delta z_j \right) \right. \\ &\quad \cdot \left. \left(\sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{\underline{g}} \Delta z_i^2 + \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} \underline{d}_i \underline{d}_j \tilde{\underline{g}} \Delta z_i \Delta z_j \right)^T \right\} \\ &- E \left\{ \sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{\underline{g}} \Delta z_i^2 \right\} \cdot E \left\{ \sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{\underline{g}} \Delta z_i^2 \right\}^T. \end{aligned} \quad (2.39d)$$

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Ausmultiplizieren und Anwendung des Erwartungswertoperators auf die einzelnen Summanden liefert unter Beachtung, dass sich alle ungeraden Momente zu Null ergeben

$$\begin{aligned}
&= \sum_{i=1}^{L_a} \underline{d}_i \tilde{g}(\underline{d}_i \tilde{g})^\top + \frac{1}{4} \sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{g} E\{\Delta z_i^4\} (\underline{d}_i^2 \tilde{g})^\top + \frac{1}{4} \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} \underline{d}_i^2 \tilde{g} (\underline{d}_j^2 \tilde{g})^\top \\
&+ \frac{1}{4} \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} \underline{d}_i \underline{d}_j \tilde{g} (\underline{d}_i \underline{d}_j \tilde{g})^\top - \frac{1}{4} \sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{g} (\underline{d}_i^2 \tilde{g})^\top - \frac{1}{4} \sum_{i=1}^{L_a} \sum_{\substack{j=1 \\ j \neq i}}^{L_a} \underline{d}_i^2 \tilde{g} (\underline{d}_j^2 \tilde{g})^\top .
\end{aligned} \tag{2.39e}$$

Die Terme 3 und 6 sind identisch bis auf das Vorzeichen und entfallen damit. Das vierte Moment $\sigma_4 = E\{\Delta z_i^4\}$ ist für die angenommene Normalverteilung des Eingangs gleich 3. Entsprechend [NPR00b] werden die Kreuzdifferenzen im Term 4 vernachlässigt, da deren Berechnung den numerischen Aufwand signifikant erhöhen würde. Zur Berechnung aller statistischen Momente vierter Ordnung wäre, wie in [NPR00a] beschrieben, letztendlich eine Approximation von \underline{g} mit einem Polynom 3. Ordnung notwendig. Zusammengefasst folgt

$$\mathbf{C}_b \approx \sum_{i=1}^{L_a} \underline{d}_i \tilde{g} (\underline{d}_i \tilde{g})^\top + \frac{\sigma_4 - 1}{4} \sum_{i=1}^{L_a} \underline{d}_i^2 \tilde{g} (\underline{d}_i^2 \tilde{g})^\top . \tag{2.39f}$$

Einsetzen von (2.33) und Rücktransformation zum gekoppelten System liefert

$$\begin{aligned}
\mathbf{C}_b &\approx \sum_{i=1}^{L_a} \frac{1}{4h_i^2} \left(\underline{g}(\bar{\underline{a}} + h_i \underline{s}_a^{(i)}) - \underline{g}(\bar{\underline{a}} - h_i \underline{s}_a^{(i)}) \right) \left(\underline{g}(\bar{\underline{a}} + h_i \underline{s}_a^{(i)}) - \underline{g}(\bar{\underline{a}} - h_i \underline{s}_a^{(i)}) \right)^\top \\
&+ \sum_{i=1}^{L_a} \frac{\sigma_4 - 1}{4h_i^4} \left(\underline{g}(\bar{\underline{a}} + h_i \underline{s}_a^{(i)}) + \underline{g}(\bar{\underline{a}} - h_i \underline{s}_a^{(i)}) - 2\underline{g}(\bar{\underline{a}}) \right) \\
&\cdot \left(\underline{g}(\bar{\underline{a}} + h_i \underline{s}_a^{(i)}) + \underline{g}(\bar{\underline{a}} - h_i \underline{s}_a^{(i)}) - 2\underline{g}(\bar{\underline{a}}) \right)^\top .
\end{aligned} \tag{2.40}$$

Wie in [NPR00a] gezeigt, ist der optimale Wert für die Schrittweite bei Annahme von Normalverteilungen $h = h_i = \sqrt{\sigma_4} = \sqrt{3}$. Einsetzen der optimalen Schrittweite ergibt schließlich

$$\begin{aligned}
\mathbf{C}_b &\approx \frac{1}{4h^2} \sum_{i=1}^{L_a} \left(\underline{g}(\bar{\underline{a}} + h \underline{s}_a^{(i)}) - \underline{g}(\bar{\underline{a}} - h \underline{s}_a^{(i)}) \right) \left(\underline{g}(\bar{\underline{a}} + h \underline{s}_a^{(i)}) - \underline{g}(\bar{\underline{a}} - h \underline{s}_a^{(i)}) \right)^\top \\
&+ \frac{h^2 - 1}{4h^4} \sum_{i=1}^{L_a} \left(\underline{g}(\bar{\underline{a}} + h \underline{s}_a^{(i)}) + \underline{g}(\bar{\underline{a}} - h \underline{s}_a^{(i)}) - 2\underline{g}(\bar{\underline{a}}) \right) \left(\underline{g}(\bar{\underline{a}} + h \underline{s}_a^{(i)}) + \underline{g}(\bar{\underline{a}} - h \underline{s}_a^{(i)}) - 2\underline{g}(\bar{\underline{a}}) \right)^\top .
\end{aligned} \tag{2.41}$$

In [NPR00a] wird aufbauend auf dieser Gleichung eine minimal zulässige Schrittweite von $h = 1$ angegeben, da sonst in (2.41) negative Summanden auftreten könnten.

2. Mathematische Methoden

Dies ist jedoch ein Trugschluss, da in (2.41) ausgehend von der optimalen Schrittweihenwahl der Wert σ_4 durch h^2 ersetzt wurde. Wird $h \neq \sqrt{3}$ gewählt, muss Gleichung (2.40) verwendet werden, die auch bei kleinen Schrittweiten keine negative Summanden enthält.

Für die Kreuzkovarianz zwischen der Eingangsgröße \underline{a} und der resultierenden Größe \underline{b} folgt

$$\mathbf{C}_{\underline{ab}} = E \left\{ (\underline{a} - \bar{\underline{a}}) (\underline{b} - \bar{\underline{b}})^T \right\} \quad (2.42a)$$

$$\approx E \left\{ \mathbf{S}_{\underline{a}} \Delta \underline{z} \left(\tilde{\underline{g}}(\bar{\underline{z}}) + \mathbf{D}_{\Delta \underline{z}} \tilde{\underline{g}} + \frac{1}{2} \mathbf{D}_{\Delta \underline{z}}^2 \tilde{\underline{g}} - \tilde{\underline{g}}(\bar{\underline{z}}) \right)^T \right\} \quad (2.42b)$$

$$= \mathbf{S}_{\underline{a}} E \left\{ \Delta \underline{z} \mathbf{D}_{\Delta \underline{z}} \tilde{\underline{g}}^T + \Delta \underline{z} \frac{1}{2} \mathbf{D}_{\Delta \underline{z}}^2 \tilde{\underline{g}}^T \right\}. \quad (2.42c)$$

Ungerade Momente von \underline{z} sind Null. Damit entfällt der zweite Summand. Einsetzen von (2.32a) liefert

$$= \mathbf{S}_{\underline{a}} E \left\{ \Delta \underline{z} \left(\sum_{i=1}^{L_{\underline{a}}} \underline{d}_i \tilde{\underline{g}} \Delta z_i \right)^T \right\} \quad (2.42d)$$

$$= \sum_{i=1}^{L_{\underline{a}}} \underline{s}_{\underline{a}}^{(i)} \underline{d}_i \tilde{\underline{g}}^T, \quad (2.42e)$$

wobei $\underline{s}_{\underline{a}}^{(i)}$ wiederum die i-te Spalte in $\mathbf{S}_{\underline{a}}$ darstellt. Einsetzen von (2.33a) und Rücktransformation zum gekoppelten System liefert

$$\mathbf{C}_{\underline{ab}} \approx \sum_{i=1}^{L_{\underline{a}}} \underline{s}_{\underline{a}}^{(i)} \frac{\left(\underline{g}(\bar{\underline{a}} + h_i \underline{s}_{\underline{a}}^{(i)}) - \underline{g}(\bar{\underline{a}} - h_i \underline{s}_{\underline{a}}^{(i)}) \right)^T}{2h_i}. \quad (2.43)$$

Wie in [JU04] gezeigt wird, kann der beschriebene Algorithmus zur Berechnung von höheren statistischen Momenten erweitert werden. In dieser Arbeit wird ausschließlich die Approximation zweiter Ordnung verwendet.

Ein Vergleich der approximativ berechneten Momente mit den Werten aus der Monte-Carlo-Berechnung ist in Bild 2.2 gezeigt. Die grauen Punkte markieren wiederum die Samples der Monte-Carlo-Simulation, mit in schwarz eingezeichnetem Mittelwert (o) und als Konfidenzellipse eingezeichneter 2σ -Grenze bei Approximation der Verteilungen durch zwei statistische Momente. In rot ist zum Vergleich das Ergebnis der Approximation erster Ordnung eingezeichnet. Der blaue Stern (Mittelwert) und die blaue Ellipse (2σ -Grenze) geben das Ergebnis der Unscented-Transformation bei Anwendung auf die oben beschriebene Funktion wieder. Dies zeigt die hohe Approximationsgenauigkeit dieses Verfahrens. Zusätzlich sind als blaue Punkte die Positionen markiert, an

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

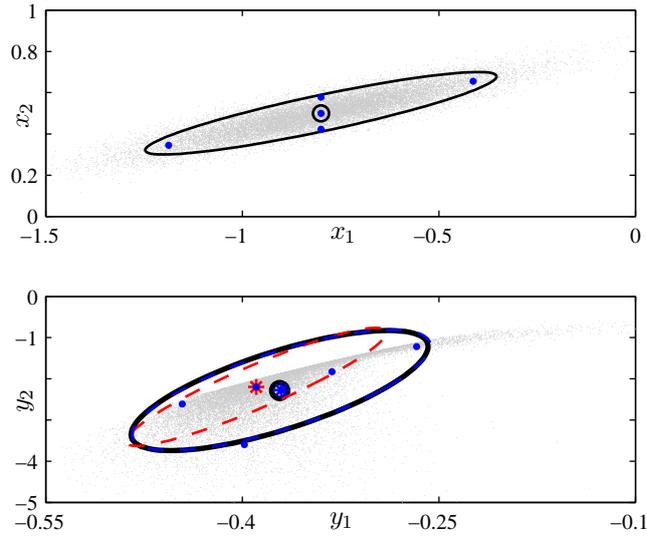


Abbildung 2.2.: Ergebnis der Monte-Carlo-Simulation (graue Punkte) für die einfache Funktion mit Mittelwert und 2σ -Grenze in schwarz (o Mittelwert, - 2σ -Grenze) im Vergleich zum Ergebnis der Approximation erster Ordnung in rot (* Mittelwert, - - - 2σ -Grenze) und der Unscented-Transformation in blau (o Mittelwert, - - - 2σ -Grenze, • Sigmapunkte).

denen die Funktion für die Berechnung der Unscented-Transformation ausgewertet wurde.

Zur Berechnung der statistischen Momente mit der Unscented-Transformation sind $2 \cdot L_a + 1$ Auswertungen der Funktion $g(\underline{a})$ nötig. Der numerische Aufwand skaliert somit trotz der Verwendung einer Approximation zweiter Ordnung nur linear mit der Anzahl der unsicheren Eingangsgrößen, bei der Anwendung auf nichtlineare Funktionen sind jedoch wesentlich genauere Ergebnisse zu erwarten, als bei der Approximation erster Ordnung.

Angewendet auf das Systemmodell (2.2) ergibt sich für die Berechnung der statistischen Momente einer Trajektorie folgendes Vorgehen.

1. Zunächst sind die Cholesky-Zerlegungen der Kovarianzmatrizen der zufälligen Einflussgrößen \underline{x}_0 , $\underline{\theta}$ und $\underline{\xi}$

$$\mathbf{S}_{\underline{x}_0} = \text{chol}(\mathbf{C}_{\underline{x}_0}) \quad \mathbf{S}_{\underline{\theta}} = \text{chol}(\mathbf{C}_{\underline{\theta}}) \quad \text{und} \quad \mathbf{S}_{\underline{\xi}} = \text{chol}(\mathbf{C}_{\underline{\xi}}) \quad (2.44)$$

zu berechnen. Die i -te Spalte in einer Matrix $\mathbf{S}_{(\cdot)}$ sei hier analog zu der oben eingeführten Nomenklatur mit $\underline{s}_{(\cdot)}^{(i)}$ bezeichnet. Für das Messrauschen ist eine solche Berechnung nicht notwendig, da es üblicherweise nur in der Berechnung der

2. Mathematische Methoden

Unsicherheiten der Messgrößentrajektorie Anwendung findet und hier wie im vorigen Abschnitt begründet die Kovarianzmatrix des Messrauschens \mathbf{C}_η additiv dem Resultat hinzugefügt werden kann.

2. Aus den Verteilungsfunktionen der unsicheren Einflussgrößen sind die Sigmapunkte

$$\begin{aligned} \underline{x}_0^{(SP_i+)} &= \bar{x}_0 + h s_{\underline{x}_0}^{(i)} \\ \underline{x}_0^{(SP_i-)} &= \bar{x}_0 - h s_{\underline{x}_0}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{x}} \quad (2.45a)$$

$$\begin{aligned} \underline{\theta}^{(SP_i+)} &= \bar{\theta} + h s_{\underline{\theta}}^{(i)} \\ \underline{\theta}^{(SP_i-)} &= \bar{\theta} - h s_{\underline{\theta}}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{\theta}} \quad (2.45b)$$

$$\begin{aligned} \underline{\dot{\xi}}^{(SP_i+)} &= +h s_{\underline{\dot{\xi}}}^{(i)} \\ \underline{\dot{\xi}}^{(SP_i-)} &= -h s_{\underline{\dot{\xi}}}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{\dot{\xi}}} \quad (2.45c)$$

zu bestimmen.

3. Durch Einsetzen der Werte \bar{x}_0 , $\bar{\theta}$ und $\bar{\dot{\xi}} = \underline{0}$ und Lösung der Modellgleichung werden zunächst die Zustandsverläufe $\underline{x}_k^{(\overline{SP})}$ und die Messgrößenverläufe $\underline{y}_k^{(\overline{SP})}$ berechnet.
4. Durch jeweiliges Einsetzen der Werte $\underline{x}_0^{(SP_i\pm)}$, $\bar{\theta}$ und $\bar{\dot{\xi}} = \underline{0}$ und Lösung der Modellgleichung werden die Zustandsverläufe $\underline{x}_{k,\underline{x}_0}^{(SP_i\pm)}$ und die Messgrößenverläufe $\underline{y}_{k,\underline{x}_0}^{(SP_i\pm)}$ berechnet. Mit den Eingangsgrößen \bar{x}_0 , $\underline{\theta}^{(SP_i\pm)}$ und $\bar{\dot{\xi}} = \underline{0}$ ergeben sich die korrespondierenden Werte $(\cdot)_{\underline{\theta}}^{(SP_i\pm)}$ und mit \bar{x}_0 , $\bar{\theta}$ und mit $\underline{\dot{\xi}}^{(SP_i\pm)}$ die Größen $(\cdot)_{\underline{\dot{\xi}}}^{(SP_i\pm)}$.
5. Mit (2.36) oder (2.37) lässt sich der Mittelwert \bar{x}_k der Zustandstrajektorie berechnen. Hier sei das Ergebnis nach (2.36) angegeben.

$$\begin{aligned} \bar{x}_k &\approx \underline{x}_k^{(\overline{SP})} + \frac{1}{2} \sum_{i=1}^{L_{\underline{x}}} \frac{\underline{x}_{k,\underline{x}_0}^{(SP_i+)} + \underline{x}_{k,\underline{x}_0}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})}}{h_i^2} \\ &\quad + \frac{1}{2} \sum_{i=1}^{L_{\underline{\theta}}} \frac{\underline{x}_{k,\underline{\theta}}^{(SP_i+)} + \underline{x}_{k,\underline{\theta}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})}}{h_i^2} \\ &\quad + \frac{1}{2} \sum_{i=1}^{L_{\underline{\dot{\xi}}}} \frac{\underline{x}_{k,\underline{\dot{\xi}}}^{(SP_i+)} + \underline{x}_{k,\underline{\dot{\xi}}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})}}{h_i^2} \end{aligned} \quad (2.46)$$

2.1. Stochastische Simulation mit Wahrscheinlichkeitsdichtefunktionen

Die Kovarianzmatrix $\mathbf{C}_{\underline{x}_k}$ lässt sich mit (2.40) oder (2.41) schätzen. Mit (2.40) und $\sigma_4 = 3$ ergibt sich

$$\begin{aligned}
\mathbf{C}_{\underline{x}_k} = & \sum_{i=1}^{L_{\underline{x}}} \frac{1}{4h_i^2} \left(\underline{x}_{k,\underline{x}_0}^{(SP_i+)} - \underline{x}_{k,\underline{x}_0}^{(SP_i-)} \right) \left(\underline{x}_{k,\underline{x}_0}^{(SP_i+)} - \underline{x}_{k,\underline{x}_0}^{(SP_i-)} \right)^{\text{T}} \\
& + \sum_{i=1}^{L_{\underline{x}}} \frac{1}{2h_i^4} \left(\underline{x}_{k,\underline{x}_0}^{(SP_i+)} + \underline{x}_{k,\underline{x}_0}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right) \cdot \left(\underline{x}_{k,\underline{x}_0}^{(SP_i+)} + \underline{x}_{k,\underline{x}_0}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right)^{\text{T}} \\
& + \sum_{i=1}^{L_{\underline{\theta}}} \frac{1}{4h_i^2} \left(\underline{x}_{k,\underline{\theta}}^{(SP_i+)} - \underline{x}_{k,\underline{\theta}}^{(SP_i-)} \right) \left(\underline{x}_{k,\underline{\theta}}^{(SP_i+)} - \underline{x}_{k,\underline{\theta}}^{(SP_i-)} \right)^{\text{T}} \\
& + \sum_{i=1}^{L_{\underline{\theta}}} \frac{1}{2h_i^4} \left(\underline{x}_{k,\underline{\theta}}^{(SP_i+)} + \underline{x}_{k,\underline{\theta}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right) \cdot \left(\underline{x}_{k,\underline{\theta}}^{(SP_i+)} + \underline{x}_{k,\underline{\theta}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right)^{\text{T}} \\
& + \sum_{i=1}^{L_{\underline{x}}} \frac{1}{4h_i^2} \left(\underline{x}_{k,\underline{\xi}}^{(SP_i+)} - \underline{x}_{k,\underline{\xi}}^{(SP_i-)} \right) \left(\underline{x}_{k,\underline{\xi}}^{(SP_i+)} - \underline{x}_{k,\underline{\xi}}^{(SP_i-)} \right)^{\text{T}} \\
& + \sum_{i=1}^{L_{\underline{x}}} \frac{1}{2h_i^4} \left(\underline{x}_{k,\underline{\xi}}^{(SP_i+)} + \underline{x}_{k,\underline{\xi}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right) \cdot \left(\underline{x}_{k,\underline{\xi}}^{(SP_i+)} + \underline{x}_{k,\underline{\xi}}^{(SP_i-)} - 2\underline{x}_k^{(\overline{SP})} \right)^{\text{T}}.
\end{aligned} \tag{2.47}$$

Die Kovarianzmatrizen $\mathbf{C}_{\underline{x}_0\underline{x}_k}$, $\mathbf{C}_{\underline{\theta}\underline{x}_k}$ und $\mathbf{C}_{\underline{\xi}\underline{x}_k}$ folgen aus (2.43).

$$\mathbf{C}_{\underline{x}_0\underline{x}_k} \approx \sum_{i=1}^{L_{\underline{x}}} \frac{s_{\underline{x}_0}^{(i)} \left(\underline{x}_{k,\underline{x}_0}^{(SP_i+)} - \underline{x}_{k,\underline{x}_0}^{(SP_i-)} \right)^{\text{T}}}{2h_i} \tag{2.48}$$

$$\mathbf{C}_{\underline{\theta}\underline{x}_k} \approx \sum_{i=1}^{L_{\underline{\theta}}} \frac{s_{\underline{\theta}}^{(i)} \left(\underline{x}_{k,\underline{\theta}}^{(SP_i+)} - \underline{x}_{k,\underline{\theta}}^{(SP_i-)} \right)^{\text{T}}}{2h_i} \tag{2.49}$$

$$\mathbf{C}_{\underline{\xi}\underline{x}_k} \approx \sum_{i=1}^{L_{\underline{x}}} \frac{s_{\underline{\xi}}^{(i)} \left(\underline{x}_{k,\underline{\xi}}^{(SP_i+)} - \underline{x}_{k,\underline{\xi}}^{(SP_i-)} \right)^{\text{T}}}{2h_i} \tag{2.50}$$

2. Mathematische Methoden

6. Analog berechnen sich die statistischen Momente der Messgrößentrajektorie.

$$\begin{aligned}
\bar{y}_k &\approx \underline{y}_k^{(\overline{SP})} + \frac{1}{2} \sum_{i=1}^{L_x} \frac{y_{k,x_0}^{(SP_i+)} + y_{k,x_0}^{(SP_i-)} - 2y_k^{(\overline{SP})}}{h_i^2} \\
&+ \frac{1}{2} \sum_{i=1}^{L_\theta} \frac{y_{k,\theta}^{(SP_i+)} + y_{k,\theta}^{(SP_i-)} - 2y_k^{(\overline{SP})}}{h_i^2} \\
&+ \frac{1}{2} \sum_{i=1}^{L_x} \frac{y_{k,\xi}^{(SP_i+)} + y_{k,\xi}^{(SP_i-)} - 2y_k^{(\overline{SP})}}{h_i^2}
\end{aligned} \tag{2.51}$$

$$\begin{aligned}
\mathbf{C}_{y_k} &= \mathbf{C}_\eta + \sum_{i=1}^{L_x} \frac{1}{4h_i^2} \left(\underline{y}_{k,x_0}^{(SP_i+)} - \underline{y}_{k,x_0}^{(SP_i-)} \right) \left(\underline{y}_{k,x_0}^{(SP_i+)} - \underline{y}_{k,x_0}^{(SP_i-)} \right)^T \\
&+ \sum_{i=1}^{L_x} \frac{1}{2h_i^4} \left(\underline{y}_{k,x_0}^{(SP_i+)} + \underline{y}_{k,x_0}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right) \cdot \left(\underline{y}_{k,x_0}^{(SP_i+)} + \underline{y}_{k,x_0}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right)^T \\
&+ \sum_{i=1}^{L_\theta} \frac{1}{4h_i^2} \left(\underline{y}_{k,\theta}^{(SP_i+)} - \underline{y}_{k,\theta}^{(SP_i-)} \right) \left(\underline{y}_{k,\theta}^{(SP_i+)} - \underline{y}_{k,\theta}^{(SP_i-)} \right)^T \\
&+ \sum_{i=1}^{L_\theta} \frac{1}{2h_i^4} \left(\underline{y}_{k,\theta}^{(SP_i+)} + \underline{y}_{k,\theta}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right) \cdot \left(\underline{y}_{k,\theta}^{(SP_i+)} + \underline{y}_{k,\theta}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right)^T \\
&+ \sum_{i=1}^{L_x} \frac{1}{4h_i^2} \left(\underline{y}_{k,\xi}^{(SP_i+)} - \underline{y}_{k,\xi}^{(SP_i-)} \right) \left(\underline{y}_{k,\xi}^{(SP_i+)} - \underline{y}_{k,\xi}^{(SP_i-)} \right)^T \\
&+ \sum_{i=1}^{L_x} \frac{1}{2h_i^4} \left(\underline{y}_{k,\xi}^{(SP_i+)} + \underline{y}_{k,\xi}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right) \cdot \left(\underline{y}_{k,\xi}^{(SP_i+)} + \underline{y}_{k,\xi}^{(SP_i-)} - 2\underline{y}_k^{(\overline{SP})} \right)^T
\end{aligned} \tag{2.52}$$

$$\mathbf{C}_{x_0 y_k} \approx \sum_{i=1}^{L_x} \frac{s_{x_0}^{(i)} \left(\underline{y}_{k,x_0}^{(SP_i+)} - \underline{y}_{k,x_0}^{(SP_i-)} \right)^T}{2h_i} \tag{2.53}$$

$$\mathbf{C}_{\theta y_k} \approx \sum_{i=1}^{L_\theta} \frac{s_\theta^{(i)} \left(\underline{y}_{k,\theta}^{(SP_i+)} - \underline{y}_{k,\theta}^{(SP_i-)} \right)^T}{2h_i} \tag{2.54}$$

$$\mathbf{C}_{\xi y_k} \approx \sum_{i=1}^{L_x} \frac{s_\xi^{(i)} \left(\underline{y}_{k,\xi}^{(SP_i+)} - \underline{y}_{k,\xi}^{(SP_i-)} \right)^T}{2h_i} \tag{2.55}$$

$$\mathbf{C}_{\eta y_k} = \mathbf{C}_\eta \tag{2.56}$$

2.2. Analyse der Unsicherheiten nichtlinearer Modelle

Um die Darstellung des Systemmodells als stochastisches Modell nutzen zu können, müssen die unsicheren Größen über ihre statistischen Eigenschaften beschrieben werden. Die Eigenschaften des Messrauschens η können oft direkt aus den Datenblättern der verwendeten Messgeräte entnommen werden. Bei aufwändigen Analysemethoden kann eine umfassende Analyse der vorhandenen Messdaten Aufschlüsse über das Messrauschen liefern. In [HKK03] ist ein solches Verfahren näher beschrieben. Die Variationen des Startzustandes \underline{x}_0 können ebenfalls recht einfach analysiert werden. Schwieriger gestaltet sich die Beschreibung des Prozessrauschens und des Parameterfehlers.

Für die systematische Analyse des Prozessrauschens $\dot{\xi}$ sind nur wenige Methoden bekannt. Dies ist vor allem auch dadurch bedingt, dass es sich hier um einen unstrukturierten Unsicherheitsansatz handelt, der viele verbleibende Unsicherheiten des Systems zusammenfasst. Ansätze zur Identifikation des Prozessrauschens finden sich hauptsächlich in Quellen zu modellgestützten Messverfahren, da im Rahmen der Zustandsschätzung diese unstrukturierte Unsicherheitsannahme weit verbreitet ist. Viele Methoden verwenden manuelle oder heuristische Ansätze. Systematische Methoden zur Schätzung der Matrix \mathbf{C}_{ξ} basieren entweder auf deren optimaler Einstellung mittels nichtlinearer Optimierung der Schätzgüte eines gewählten Zustandsschätzverfahrens [Hei04] oder auf Approximation mit Hilfe der Parameterunsicherheit [VG00, KHK07]. Ersteres Verfahren ist sehr aufwändig und liefert teilweise Ergebnisse, die nicht von einem auf ein anderes Zustandsschätzverfahren überführt werden können. Im Rahmen der Optimierung werden hier auch Unzulänglichkeiten des Schätzverfahrens in das Prozessrauschen eingebunden. Die Parameterunsicherheit wird in dieser Arbeit bereits explizit berücksichtigt, so dass Verfahren, die das Prozessrauschen aus den Parameterunsicherheiten approximieren hier nicht anwendbar sind. Diese Unsicherheit wird daher oft bei der Prozessplanung vernachlässigt.

Der zu erwartende Parameterfehler \mathbf{C}_{θ} eines Modells lässt sich aus einer Analyse der Parameteridentifikation ableiten. In der Literatur sind verschiedene Verfahren zur Analyse der verbleibenden Parameterfehler beschrieben. Hier sollen exemplarisch die Bootstrap-Analyse und die Fisher'sche-Informationmatrix vorgestellt werden.

Die Bootstrap-Methode wurde zuerst von [Efr79] vorgestellt. Neben der Abschätzung des zweiten statistischen Moments des zu erwartenden Parameterfehlers können mit dieser Methode auch weitere statistische Momente berechnet werden. In den meisten Fällen werden Bias und Varianz oder Konfidenzintervalle des zu erwartenden Fehlers geschätzt. Die breite Anwendbarkeit der Bootstrap Analyse bedingt eine große Menge an Literatur zu dem Verfahren. Einen Überblick zur Methode und dessen Erweiterungen und Anwendung kann z.B. [DR88, Hal88, Hal92, LPR92, ET93, JKSM06] und [JSMK06] entnommen werden.

2. Mathematische Methoden

Die Analyse der Fisher'sche-Informationsmatrix basiert auf der von [Fis71] vorgestellten Fisher-Information. In [MP89a, MP89b, Mun91] wird das mathematische Verfahren zur Berechnung der Fisher'sche-Informationsmatrix für dynamische Systeme vorgestellt. Neben der Analyse des zu erwartenden Parameterfehlers, dient dieses Verfahren auch als Grundlage zur Optimalen-Versuchsplanung. Anwendungen der Fisher'schen-Informationsmatrix sind u.a. in [BSSR94, BBKS00, Hei04, Maj98] vorgestellt.

Ausgangspunkt einer Parameteranalyse bildet ein reduziertes stochastisches Systemmodell (2.2) bei dem der Anfangszustand als deterministisch $\underline{x}_0 = \bar{x}_0$ angenommen und das Systemrauschen vernachlässigt wird. Sollen die Unsicherheiten des Anfangszustandes mit untersucht werden, können die unsicheren Anteile in \underline{x}_0 durch einen erweiterten Parametervektor dargestellt werden, so dass $\underline{x}_0 = \underline{x}_0(\underline{\theta})$ sich als deterministischer Zusammenhang ergibt. Im Folgenden wird der einfacheren Lesbarkeit halber, auf die explizite Angabe dieser Abhängigkeit verzichtet. Der Messvektor $\underline{y}_k^{(i)}$ des i -ten Versuches zum Zeitpunkt t_k ergibt sich dann zu

$$\underline{y}_k^{(i)} = \underline{g}(t_k, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}^*) + \underline{\eta}^{(i)}(t_k), \quad (2.57)$$

wobei $\mathbf{U}^{(i)}$ den Stellgrößenverlauf repräsentiert. Die Funktion \underline{g} stellt hier die Lösung des Differentialgleichungssystems (2.1a) zum Zeitpunkt t_k dar. Verwendet man, wie hier, den wahren aber unbekanntem Parametersatz $\underline{\theta}^*$, setzt sich der Messwert $\underline{y}_k^{(i)}$ aus einem deterministischen Anteil $\underline{g}(t_k, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}^*)$ und einem stochastischen Anteil $\underline{\eta}^{(i)}(t_k)$ zusammen. Ziel der Parameteridentifikation ist es, einen geschätzten Parametersatz $\underline{\hat{\theta}}$ zu finden, so dass für alle Messzeitpunkte aller Versuche $\underline{g}(\underline{\theta}^*) \approx \underline{g}(\underline{\hat{\theta}})$ und somit $\underline{\theta}^* \approx \underline{\hat{\theta}}$ gilt. Ein effektives Verfahren zur Parameteridentifikation ist die Maximum-Likelihood-Identifikation. Eine Parameterschätzung nach N_E Experimenten, mit jeweils n_i , $i = 1, \dots, N_E$ Messzeitpunkten erhält man dabei durch Lösung des Minimierungsproblems

$$\underline{\hat{\theta}} = \arg \min_{\underline{\theta}} \sum_{i=1}^{N_E} \sum_{k=1}^{n_i} \left(\underline{y}_k^{(i)} - \underline{g}(t_k, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \right)^T \mathbf{C}_{\eta,k}^{(i)-1} \left(\underline{y}_k^{(i)} - \underline{g}(t_k, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \right). \quad (2.58)$$

Die zusätzliche Indizierung der Kovarianzmatrix des Messrauschens sei hier der Vollständigkeit halber eingeführt, da die Kovarianz des Messrauschens von Messzeitpunkt zu Messzeitpunkt variieren kann. Ist $\mathbf{C}_{\eta,k}$ die verwendete Kovarianzmatrix des Messrauschens und geht das Messrauschen additiv in die Messung ein, liefert dieses Verfahren eine biasfreie Schätzung mit der geringst möglichen Varianz. Eine kompaktere Darstellung ergibt sich, wenn alle Messungen eines Versuches und die Lösung der Modellgleichungen jeweils in einem Vektor

$$\underline{y}^{(i)} = \begin{pmatrix} \underline{y}_1^{(i)} \\ \underline{y}_2^{(i)} \\ \vdots \\ \underline{y}_{n_i}^{(i)} \end{pmatrix} \quad \text{und} \quad \underline{g}(\underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) = \begin{pmatrix} \underline{g}(t_1, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \\ \underline{g}(t_2, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \\ \vdots \\ \underline{g}(t_{n_i}, \underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \end{pmatrix} \quad (2.59a)$$

zusammengefasst werden. Das Messrauschen und die Kovarianzmatrix des Messrauschens des i -ten Versuches sind dann

$$\underline{\boldsymbol{\eta}}^{(i)} = \begin{pmatrix} \underline{\eta}^{(i)}(t_1) \\ \underline{\eta}^{(i)}(t_2) \\ \vdots \\ \underline{\eta}^{(i)}(t_{n_i}) \end{pmatrix} \quad \text{und} \quad \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{(i)} = \begin{pmatrix} \mathbf{C}_{\underline{\eta},1}^{(i)} & 0 & \cdots & 0 \\ 0 & \mathbf{C}_{\underline{\eta},2}^{(i)} & 0 & \vdots \\ \vdots & 0 & \ddots & \\ 0 & \cdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{C}_{\underline{\eta},n_i}^{(i)} \end{pmatrix}. \quad (2.59b)$$

Mit diesen Substitutionen entfällt eine Summe in (2.58). Zusammengefasst liest sich das Gütefunktional der Maximum-Likelihood-Identifikation dann wie folgt.

$$\hat{\underline{\theta}} = \arg \min_{\underline{\theta}} \sum_{i=1}^{N_E} \left(\underline{\boldsymbol{y}}^{(i)} - \underline{\boldsymbol{g}}(\underline{\boldsymbol{x}}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \right)^T \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{(i)-1} \left(\underline{\boldsymbol{y}}^{(i)} - \underline{\boldsymbol{y}}(\underline{\boldsymbol{x}}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}) \right) \quad (2.60)$$

Diese kompaktere Schreibweise wird nachfolgend bei der Schätzung des zu erwartenden Parameterfehlers und im Rahmen der Optimalen-Versuchsplanung Verwendung finden. Im Folgenden sollen zunächst drei Verfahren zur Analyse der verbleibenden Parameterfehler nach einer Identifikation basierend auf N_E Experimenten vorgestellt werden.

2.2.1. Die Bootstrap-Methode

Die Bootstrap-Methode [Efr79] zur Analyse des verbleibenden Parameterfehlers basiert auf einer Monte-Carlo-Analyse des Identifikationsverfahrens. Ausgehend von der Tatsache, dass die Parameteridentifikation nach (2.60) als eine funktionale Abbildung

$$\hat{\underline{\theta}} = \underline{PI} \left(\underline{\boldsymbol{y}}^{(1)}, \underline{\boldsymbol{y}}^{(2)}, \dots, \underline{\boldsymbol{y}}^{(N_E)} \right) \quad (2.61)$$

mit zufälligen Eingangsgrößen $\underline{\boldsymbol{y}}^{(i)}$ angesehen werden kann, lassen sich bei gegebener Verteilungsfunktion der Eingangsgrößen die statistischen Momente des Parametervektors mit Hilfe einer Monte-Carlo-Simulation abschätzen. Für die Messwerte gilt nach (2.57)

$$\underline{\boldsymbol{y}}^{(i)} = \underline{\boldsymbol{g}}(\underline{\boldsymbol{x}}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}^*) + \underline{\boldsymbol{\eta}}^{(i)} \quad (2.62)$$

und somit für die Verteilungsfunktion der Eingangsgrößen

$$\underline{\boldsymbol{y}}^{(i)} \sim \mathcal{N} \left(\underline{\boldsymbol{g}}(\underline{\boldsymbol{x}}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}^*), \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{(i)} \right). \quad (2.63)$$

Ausgehend von der gegebenen Verteilung der Eingangsgrößen, können also Samples aus dieser Verteilung gezogen, für jede solche Realisierung einer Parameteridentifikation

2. Mathematische Methoden

durchgeführt und aus den resultierenden Parametervektoren die statistischen Momente der Verteilungsfunktion des Parameterfehlers bestimmt werden. Einzig offen bleibt die Frage nach dem wahren Parametervektor, der für die Berechnung des Mittelwertes der Eingangsverteilung benötigt wird. Basierend auf der Annahme, dass bei Einsatz eines konsistenten Schätzverfahrens, wie der Maximum-Likelihood Methode, $\underline{\theta}^* \approx \hat{\underline{\theta}}$ gilt, kann auch $\underline{\mathcal{G}}(\underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta}^*) \approx \underline{\mathcal{G}}(\underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \hat{\underline{\theta}})$ gefolgert werden.

Mit der Bootstrap-Analyse können bei Verwendung einer hohen Anzahl von Samples die statistischen Momente der Verteilung des Parameterfehlers mit hoher Genauigkeit approximiert werden. Nutzt man als Startwert für die im Rahmen der Monte-Carlo-Simulation durchzuführenden Parameteridentifikationen den geschätzten Parametersatz $\hat{\underline{\theta}}$ aus der Identifikation mit den realen Messdaten, kann ein effizientes lokales Optimierungsverfahren, wie z.B. ein SQP-Verfahren, mit wenigen Iterationen konvergieren. Dennoch bedingt die hohe Anzahl von Einzelidentifikationen einen sehr hohen Rechenaufwand.

Anstelle der Monte-Carlo-Simulation wäre es auch möglich, eine der in Abschnitt 2.1 verwendeten anderen Approximationsverfahren zur Berechnung der Kovarianzmatrix des Parameterfehlers zu verwenden. Insbesondere die Verwendung der Unscented-Transformation, also einer Unscented-Bootstrap-Analyse, scheint hier aussichtsreich. Dieser Ansatz wurde im Rahmen dieser Arbeit jedoch nur ansatzweise untersucht und ist daher hier nicht näher dargestellt.

2.2.2. Die Fisher'sche-Informationsmatrix

Die Herleitung der Fisher'schen-Informationsmatrix basiert auf einer Taylorreihenentwicklung 2. Ordnung des Gütefunktional der Parameteridentifikation um den wahren Parametersatz $\underline{\theta}^*$ [GP77, Mun91, Hei04]. Die untere Cramer-Rao-Grenze des zu erwartenden Parameterfehlers nach N_E Experimenten lässt sich danach mit

$$\begin{aligned} \mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E) &\leq \mathbf{F}(1, \dots, N_E) \\ &= \sum_{i=1}^{N_E} \left(\left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} \right)^T \mathbf{C}_{\underline{\mathfrak{y}}}^{(i)-1} \left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(i)}, \mathbf{U}^{(i)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} \end{aligned} \quad (2.64)$$

angeben. Die Matrix $\mathbf{F}(1, \dots, N_E)$ ist die Fisher'sche-Informationsmatrix (FIM) basierend auf einer Analyse der Experimente 1 bis N_E . Daraus resultierend kann die Verteilung der Parameterwerte mit $\underline{\theta} \sim \mathcal{N}(\underline{\theta}^*, \mathbf{C}_{\underline{\theta}})$ approximiert werden.

Zur numerischen Berechnung der Fisher'schen-Informationsmatrix für Zustandsraummodelle werden im Allgemeinen zwei Methoden vorgeschlagen. Zum einen kann bei Verwendung eines SQP-Verfahrens zur Lösung des Minimierungsproblems (2.60) die vom Optimierer berechnete Hessematrix des Gütefunktional benutzt werden. In den meist verwendeten Quasi-Newton-Verfahren wird die Hessematrix jedoch nicht direkt

berechnet, sondern aus den numerisch berechneten Gradienten im Rahmen der Optimierung geschätzt. Diese Hessematrix liefert daher nur eine grobe Schätzung der Fisher'schen Informationsmatrix. Zum anderen können die in Gleichung (2.64) benötigten Sensitivitäten analog zur Approximation erster Ordnung in Abschnitt 2.1.2 mit Hilfe von Differenzenquotienten oder über analytische Berechnung der Ableitungen errechnet werden.

Die angegebene Berechnungsvorschrift enthält letztendlich noch den unbekanntem wahren Parametersatz $\underline{\theta}^*$. Ausgehend von einem effektiven Identifikationsverfahren, wie zum Beispiel der Maximum-Likelihood-Identifikation, kann die Annahme getroffen werden, dass $\hat{\underline{\theta}} \approx \underline{\theta}^*$ gilt. Somit kann die Fisher'sche-Informationsmatrix näherungsweise mit dem identifizierten Parametersatz berechnet werden. Es ergibt sich die statistische Verteilung der Parameterwerte zu $\underline{\theta} \sim \mathcal{N}(\hat{\underline{\theta}}, \mathbf{C}_{\underline{\theta}})$.

2.3. Optimale-Versuchsplanung

Ausgehend von den im letzten Abschnitt beschriebenen Verfahren, ist eine Abschätzung des verbleibenden Parameterfehlers nach der Parameteridentifikation möglich. Sind die zu erwartenden Fehler noch sehr hoch, ist es nötig, bevor das mathematische Modell im Rahmen der Prozessführung eingesetzt werden soll, den zu erwartenden Parameterfehler noch zu senken und somit die Prädiktionsgüte des Modells zu verbessern. Wie aus dem vergangenen Abschnitt, insbesondere der Fisher'schen-Informationsmatrix, hervor geht, verringert sich der zu erwartende Parameterfehler, je mehr Versuche und somit Messwerte für die Identifikation zur Verfügung stehen. Zusätzlich ist der Informationsgehalt eines Experiments jedoch auch signifikant von der Prozessführung, also der Anfangsbedingung, der Stellgrößentrajektorie und den Messzeitpunkten abhängig. Insbesondere lässt sich der zu erwartende Parameterfehler nach Hinzunahme eines weiteren Experimentes mit den beschriebenen Methoden abschätzen, bevor der Versuch durchgeführt ist, da die beschriebenen Methoden die Messwerte im Rahmen der Analyse nicht benötigen. Hier setzen Verfahren zur Optimalen-Versuchsplanung an, deren Ziel es ist, einen Versuch zu planen, der möglichst viele neue Informationen für die Parameteridentifikation liefert.

Ausgangspunkt einer solchen Optimalen-Versuchsplanung ist jeweils ein zu erwartender Parameterfehler beschrieben durch die Kovarianzmatrix $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E)$ nach der durchgeführten Parameteridentifikation mit N_E Experimenten. Dieser kann wie beschrieben entweder aus einer (Unscented-) Bootstrap-Analyse gewonnen oder aus der Fisher'schen-Informationsmatrix der Identifikationsversuche mit Hilfe der Cramer-Rao-Beziehung abgeschätzt werden.

2.3.1. Optimale-Versuchsplanung mit der Fishermatrix

Die bekannteste Methode der dynamischen Optimalen-Versuchsplanung basiert auf der in Abschnitt 2.2.2 eingeführten Fisher'schen-Informationsmatrix. Die Fisher'sche-Informationsmatrix eines geplanten Versuches $N_E + 1$ kann mit

$$\mathbf{F}(N_E + 1) = \left(\left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} \right)^T \mathbf{C}_{\underline{\mathfrak{U}}}^{(N_E+1)-1} \left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} \quad (2.65)$$

bestimmt werden. Dabei bezeichnet $\underline{\mathcal{G}}$ wiederum die Lösung des deterministischen Anteils der Modellgleichungen (2.2) an allen Messzeitpunkten, $\underline{x}_0^{(N_E+1)}$ die Anfangsbedingung des geplanten Versuches und $\mathbf{U}^{(N_E+1)} = \{\underline{u}^{(N_E+1)}(t), t \in [t_0, t_{\text{Ende}}]\}$, den Stellgrößenverlauf.

Damit lässt sich der zu erwartende Parameterfehler nach Durchführung des geplanten Versuches und anschließender Parameteridentifikation mit allen Versuchen mit

$$\mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E, N_E + 1) \leq \mathbf{F}(1, \dots, N_E) + \mathbf{F}(N_E + 1) \quad (2.66)$$

angeben. Einsetzen von (2.65) und Anwendung der Substitution

$$\mathbf{G}^{(N_E+1)} = \left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} \quad (2.67)$$

liefert

$$\mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E, N_E + 1) \leq \mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E) + \mathbf{G}^{(N_E+1)T} \mathbf{C}_{\underline{\mathfrak{U}}}^{(N_E+1)-1} \mathbf{G}^{(N_E+1)}. \quad (2.68)$$

Für Systeme, die linear in den Parametern sind, stellt diese Cramer-Rao-Grenze sogar eine exakte Berechnung des zu erwartenden Parameterfehlers dar.

$$\mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E, N_E + 1) = \mathbf{F}(1, \dots, N_E) + \mathbf{G}^{(N_E+1)T} \mathbf{C}_{\underline{\mathfrak{U}}}^{(N_E+1)-1} \mathbf{G}^{(N_E+1)}. \quad (2.69)$$

Auf Basis dieses prädierten zu erwartenden Parameterfehlers, kann anschließend eine Optimierung des geplanten Versuches erfolgen. Übliche Freiheitsgrade bei der Planung eines optimal für die Parameteridentifikation geeigneten Versuches sind die Anfangsbedingung \underline{x}_0 und die Stellgrößentrajektorie \mathbf{U} . Zusätzlich könnten auch die Anzahl der Messzeitpunkte und die Messzeitpunkte selbst variabel sein. Aufgrund von Beschränkungen im realen Betrieb, wie z.B. der Kosten der Analyse und der Arbeitszeiten der den Versuch betreuenden Personen, ist dies jedoch eher selten der Fall. Schließlich muss die Planung mit dem geschätzten Parametersatz $\hat{\underline{\theta}}$ erfolgen, da der reale Parametersatz $\underline{\theta}^*$ unbekannt ist. Mathematisch ergibt sich somit das zu lösende Minimierungsproblem

$$\left\{ \underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)} \right\} = \arg \min_{\{\underline{x}_0, \mathbf{U}\}} \Phi_{\text{OVP}} \left(\underline{x}_0, \mathbf{U}, \hat{\underline{\theta}}, \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \right) \quad (2.70a)$$

mit den Nebenbedingungen

$$\text{Modellgleichungen:} \quad \dot{\underline{x}} = \underline{f}(t, \underline{x}(t), \underline{u}(t), \hat{\underline{\theta}}) \quad (2.70b)$$

$$\text{Anfangsbedingung:} \quad \underline{x}(t=0) = \underline{x}_0 \quad (2.70c)$$

$$\text{Versuchslaufzeit:} \quad t_0 \leq t \leq t_{\text{Ende}} \quad (2.70d)$$

$$\text{Beschränkung der Anfangsbedingung:} \quad \underline{x}_{0,\min} \leq \underline{x}_0 \leq \underline{x}_{0,\max} \quad (2.70e)$$

$$\text{Stellgrößenbeschränkungen:} \quad \underline{u}_{\min} \leq \underline{u}(t) \leq \underline{u}_{\max} \quad (2.70f)$$

$$\text{Systembeschränkungen:} \quad \underline{r}(\underline{x}(t)) \leq \underline{0}, \quad \underline{r} \in \mathbb{R}^{L_r}. \quad (2.70g)$$

Für den Einsatz von Optimierungsverfahren muss die Eignung eines geplanten Versuches zur Parameteridentifikation durch einen skalaren Wert Φ_{OVP} beschrieben werden. Einige bekannte Funktionale hierfür sind im Folgenden aufgeführt.

- Das E-Kriterium berechnet sich mit

$$\Phi_{\text{OVP}} = \sqrt{\frac{1}{\max(\text{eig}(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)))}}. \quad (2.71a)$$

$\text{eig}(\mathbf{C})$ steht hier für die Eigenwerte der Matrix \mathbf{C} . Betrachtet man ausschließlich zwei statistische Momente des zu erwartenden Parameterfehlers, also $\underline{\theta} \sim \mathcal{N}(\hat{\underline{\theta}}, \mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1))$, ist dieser Wert proportional zur Länge der längsten Hauptachse eines Konfidenzellipsoiden dieser Verteilung, wie es in Bild 2.3.1 für eine zweiparametrische Verteilung skizziert ist.

- Das modifizierte E-Kriterium

$$\Phi_{\text{OVP}} = \sqrt{\frac{\min(\text{eig}(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)))}{\max(\text{eig}(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)))}}. \quad (2.71b)$$

bewertet den Quotienten aus der kürzesten und der längsten Hauptachse des Konfidenzellipsoiden.

- Das A-Kriterium

$$\Phi_{\text{OVP}} = \text{spur}(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)) \quad (2.71c)$$

bewertet die Varianz der einzelnen Parameter im arithmetischen Mittel.

- Das D-Kriterium

$$\Phi_{\text{OVP}} = \det(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)) \quad (2.71d)$$

ist ein Maß für die Fläche/das Volumen des Konfidenzellipsoiden.

2. Mathematische Methoden

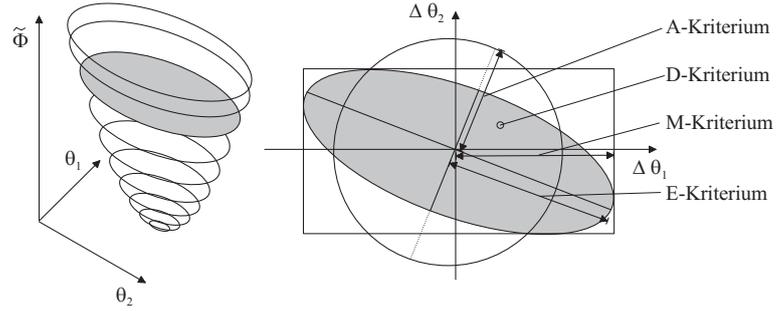


Abbildung 2.3.: Anschauliche Interpretation der Kriterien der Optimalen-Versuchsplanung.

- Das M-Kriterium berechnet sich mit

$$\Phi_{\text{OVP}} = \sqrt{\max_{i \in \{1, \dots, L_\theta\}} (\sigma_{\theta_i}(1, \dots, N_E, N_E + 1))}, \quad (2.71e)$$

wobei $\sigma_{\theta_i}(1, \dots, N_E, N_E + 1)$ das i -te Hauptdiagonalelement in $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)$ darstellt. Es bewertet damit die größte verbleibende Standardabweichung eines Parameters.

Wenn die Modellparameter unterschiedliche Größenordnungen haben, ist es sinnvoll, nicht die absolute Kovarianzmatrix $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)$ zu verwenden, sondern eine auf die Parameterwerte normierte Matrix $\tilde{\mathbf{C}}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)$. Die Berechnung dieser relativen Kovarianzmatrix erfolgt mit

$$\tilde{\mathbf{C}}_{\underline{\theta}}(1, \dots, N_E, N_E + 1) = \Lambda_{\hat{\underline{\theta}}}^{-1} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1) \Lambda_{\hat{\underline{\theta}}}^{-1}$$

$$\text{mit: } \Lambda_{\hat{\underline{\theta}}} = \begin{pmatrix} \hat{\theta}_1 & 0 & \dots & 0 \\ 0 & \hat{\theta}_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \hat{\theta}_{L_\theta} \end{pmatrix}. \quad (2.72)$$

$\tilde{\mathbf{C}}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)$ kann dann analog zu $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1)$ zur Berechnung des Gütefunktions der OVP verwendet werden.

Für nichtlineare Systemmodelle, die bei chemischen und biochemischen Prozessen meist auftreten, ist eine direkte analytische Lösung des Optimierungsproblems, z.B. über den Hamilton-Ansatz, in den seltensten Fällen möglich. Daher kommen hier üblicherweise numerische Verfahren zur nichtlinearen Optimierung, wie SQP-Verfahren, zum Einsatz. Für die Anwendung dieser Verfahren, ist es notwendig die Stellgrößentrajektorie in parametrischer Form anzugeben. Dies kann, wie z.B. in [Hei04] beschrieben, über die stückweise konstante, stückweise lineare oder andere polygone Ansätze erfolgen. Eine stückweise

konstante Darstellung der Steuerfunktion mit $N_{\underline{u}}$ Stützstellen ist

$$\underline{u}(t) = \begin{cases} \underline{u}_0 & , \quad t_0 \leq t < t_1 \\ \underline{u}_1 & , \quad t_1 \leq t < t_2 \\ \vdots & \\ \underline{u}_{N_{\underline{u}}-1} & , \quad t_{N_{\underline{u}}-1} \leq t < t_{N_{\underline{u}}} . \end{cases} \quad (2.73)$$

Weitere Parametrisierungen sind z.B. [Hei04] zu entnehmen. Durch den Optimierer können dann die Stellgrößenwerte $\underline{u}_i, i = 0, \dots, N_{\underline{u}} - 1$ und/oder die Stellzeitpunkte $t_i, i = 1, \dots, N_{\underline{u}}$ variiert werden.

2.3.2. Ein Maximum-a-Posteriori-Ansatz zur Optimalen-Versuchsplanung

Das Hauptproblem des Fisher-Ansatzes ist die komplexe Berechnung der Fisher'schen-Informationsmatrix. Für die analytische Berechnung der benötigten Sensitivitäten in (2.64) ist parallel zur Lösung der Modellgleichungen die Lösung des Matrix-Differentialgleichungssystems (2.16) mit $\underline{a} = \underline{\theta}$ notwendig. Die daraus resultierenden Probleme sind im Abschnitt 2.1.2 beschrieben. [CV95] beschreiben verschiedene Verfahren zur Optimalen -Versuchsplanung, die auf Bayes'schen Ansätzen beruhen. Dennoch werden auch in diesem sehr umfassenden Review nur Anwendungen der verschiedenen Ansätze auf statische Probleme gezeigt. Die vorgestellten Ansätze verwenden dabei in allen Fällen Ableitungen des zu identifizierenden Modells. Somit treten auch hier die gleichen Probleme zu Tage, wie Sie in Abschnitt 2.1.2 beschrieben sind.

Um die Probleme der Berechnung der Sensitivitäten zu umgehen, soll hier ein neuer Ansatz zur Optimalen-Versuchsplanung vorgeschlagen werden. Grundidee des neuen Ansatzes ist, anstelle einer Maximum-Likelihood-Identifikation, eine Maximum-a-Posteriori-Identifikation zu verwenden. Während bei der Maximum-Likelihood-Identifikation

$$\hat{\underline{\theta}}_{\text{MLE}}(1, \dots, N_E) = \arg \max_{\underline{\theta}} p(\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E)} | \underline{\theta}) \quad (2.74)$$

gelöst, also die Likelihood maximiert wird, basiert eine Maximum-a-Posteriori-Schätzung auf der Maximierung von

$$\hat{\underline{\theta}}_{\text{MAP}}(1, \dots, N_E) = \arg \max_{\underline{\theta}} p(\underline{\theta} | \underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(n)}) . \quad (2.75)$$

Der Name dieses Schätzverfahrens resultiert aus der Tatsache, dass die benötigte Verteilungsfunktion über den Satz von Bayes berechnet werden muss und dort die Bezeichnung Posterior trägt.

Satz 1

Iterative Anwendung der Maximum-a-Posteriori-Schätzung für alle Experimente $i = 1, \dots, N_E$ ergibt die gleiche Parameterschätzung, wie die Anwendung der Maximum-Likelihood-Identifikation.

2. Mathematische Methoden

Beweis 1

Induktionsanfang: $N_E = 1$

$$\hat{\theta}_{MAP}(1) = \arg \max_{\theta} p(\theta | \underline{\mathcal{Y}}^{(1)}) \quad (2.76a)$$

Nach dem Satz von Bayes folgt

$$= \arg \max_{\theta} \frac{p(\underline{\mathcal{Y}}^{(1)} | \theta) p(\theta)}{p(\underline{\mathcal{Y}}^{(1)})}. \quad (2.76b)$$

Da der Nenner des Bruches keine Funktion von θ ist, resultiert

$$= \arg \max_{\theta} p(\underline{\mathcal{Y}}^{(1)} | \theta) p(\theta). \quad (2.76c)$$

Ohne Vorwissen bezüglich der Verteilung von θ , muss hierfür eine Gleichverteilung $p(\theta) = \text{konst.}$ angenommen werden.

$$= \arg \max_{\theta} p(\underline{\mathcal{Y}}^{(1)} | \theta) = \hat{\theta}_{MLE}(1) \quad (2.76d)$$

Induktionsschritt: $N_E = n$

$$\hat{\theta}_{MAP}(1, \dots, n) = \arg \max_{\theta} p(\theta | \underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(n)}) \quad (2.77a)$$

Nach dem Satz von Bayes folgt wiederum

$$= \arg \max_{\theta} \frac{p(\underline{\mathcal{Y}}^{(n)} | \theta) p(\theta | \underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(n-1)})}{p(\underline{\mathcal{Y}}^{(n)})}. \quad (2.77b)$$

Da der Nenner des Bruches wiederum keine Funktion von θ ist, resultiert

$$= \arg \max_{\theta} p(\underline{\mathcal{Y}}^{(n)} | \theta) p(\theta | \underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(n-1)}), \quad (2.77c)$$

wobei der zweite Faktor, der Prior der n -ten Schätzung, der Posterior der vorangegangenen $(n - 1)$ -ten Schätzung ist. Rekursive Anwendung des Satzes von Bayes liefert

$$= \arg \max_{\theta} \prod_{i=1}^n p(\underline{\mathcal{Y}}^{(i)} | \theta). \quad (2.77d)$$

Da das Messrauschen zu den Ereignissen $\underline{\mathcal{Y}}^{(i)}$ unabhängig ist, resultiert

$$\begin{aligned} \hat{\theta}_{MAP}(1, \dots, n) &= \arg \max_{\theta} p(\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E)} | \theta) \\ &= \hat{\theta}_{MLE}(1, \dots, n). \end{aligned} \quad (2.77e) \quad \text{q.e.d.}$$

2.3. Optimale-Versuchsplanung

Mit Hilfe des Satzes von Bayes folgt für die Verteilungsdichte der Parameter nach Hinzunahme eines geplanten Experiments $N_E + 1$

$$p(\underline{\theta}|\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E+1)}) = \frac{p(\underline{\mathcal{Y}}^{(N_E+1)}|\underline{\theta})p(\underline{\theta}|\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E)})}{p(\underline{\mathcal{Y}}^{(N_E+1)})}. \quad (2.78)$$

Unter der Voraussetzung, dass der Prior $p(\underline{\theta}|\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E)})$, die Likelihood $p(\underline{\mathcal{Y}}^{(N_E+1)}|\underline{\theta})$ und der Posterior $p(\underline{\theta}|\underline{\mathcal{Y}}^{(1)}, \underline{\mathcal{Y}}^{(2)}, \dots, \underline{\mathcal{Y}}^{(N_E+1)})$ normalverteilt sind, lässt sich die Kovarianzmatrix des verbleibenden Parameterfehlers ($\hat{=}$ die Posteriorverteilung) mit

$$\mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1) = \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) - \mathbf{K} \mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)} \mathbf{K}^T \quad (2.79a)$$

$$\text{mit : } \mathbf{K} = \mathbf{C}_{\underline{\theta}\underline{\mathcal{Y}}}^{(N_E+1)} \mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)-1} \quad (2.79b)$$

$$= \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) - \mathbf{C}_{\underline{\theta}\underline{\mathcal{Y}}}^{(N_E+1)} \left(\mathbf{C}_{\underline{\theta}\underline{\mathcal{Y}}}^{(N_E+1)} \mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)-1} \right)^T \quad (2.79c)$$

approximieren, wobei $\mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)}$ die Kovarianzmatrix der Messgrößen des geplanten Experiments, resultierend aus dem Messrauschen und den Auswirkungen der Parameterunsicherheiten ist. Die Matrix $\mathbf{C}_{\underline{\theta}\underline{\mathcal{Y}}}^{(N_E+1)}$ ist die Kreuzkovarianz zwischen den Modellparameterfehlern und dem Systemausgang.

Satz 2

Für den Fall, dass $\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta})$ linear in den Parametern ist, und den linearisierten Fall

$$\begin{aligned} \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}) &\approx \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) \\ &+ \left. \frac{d\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta})}{d\underline{\theta}} \right|_{\underline{\theta}^*} (\underline{\theta} - \underline{\theta}^*) \end{aligned} \quad (2.80a)$$

$$= \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) + \mathbf{G}^{(N_E+1)} (\underline{\theta} - \underline{\theta}^*) \quad (2.80b)$$

ist die Formulierung (2.79) äquivalent zur Cramer-Rao-Grenze im Fisher-Ansatz (2.69).

Beweis 2

Die Kovarianzmatrix $\mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)}$ ist definiert durch

$$\mathbf{C}_{\underline{\mathcal{Y}}}^{(N_E+1)} = E \left\{ \left(\underline{\mathcal{Y}}^{(N_E+1)} - \underline{\mathcal{Y}}^{(N_E+1)*} \right) \left(\underline{\mathcal{Y}}^{(N_E+1)} - \underline{\mathcal{Y}}^{(N_E+1)*} \right)^T \right\}. \quad (2.81a)$$

2. Mathematische Methoden

Mit (2.57), (2.59) und (2.80) folgt

$$\begin{aligned} &\approx E \left\{ \left(\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) + \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) + \underline{\mathfrak{N}}^{(N_E+1)} \right. \right. \\ &\quad \left. \left. - \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) \right) \right. \\ &\quad \left. \left(\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) + \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) + \underline{\mathfrak{N}}^{(N_E+1)} \right. \right. \\ &\quad \left. \left. - \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) \right)^T \right\} \end{aligned} \quad (2.81b)$$

$$\begin{aligned} &= E \left\{ \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \left(\mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \right)^T + \underline{\mathfrak{N}}^{(N_E+1)} \underline{\mathfrak{N}}^{(N_E+1)T} \right. \\ &\quad \left. + \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \underline{\mathfrak{N}}^{(N_E+1)T} + \underline{\mathfrak{N}}^{(N_E+1)} \left(\mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \right)^T \right\}. \end{aligned} \quad (2.81c)$$

Da das Messrauschen unabhängig vom Parameterfehler ist und sowohl $E \left\{ \underline{\mathfrak{N}}^{(N_E+1)} \right\} = \underline{0}$ als auch $E \left\{ (\underline{\theta} - \underline{\theta}^*) \right\} = \underline{0}$ gelten, entfallen die letzten beiden Summanden.

$$= E \left\{ \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) (\underline{\theta} - \underline{\theta}^*)^T \mathbf{G}^{(N_E+1)T} + \underline{\mathfrak{N}}^{(N_E+1)} \underline{\mathfrak{N}}^{(N_E+1)T} \right\} \quad (2.81d)$$

Auswertung des Erwartungswertes mit $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E) = E \left\{ (\underline{\theta} - \underline{\theta}^*) (\underline{\theta} - \underline{\theta}^*)^T \right\}$ ergibt

$$= \mathbf{G}^{(N_E+1)} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_E+1)T} + \mathbf{C}_{\underline{\mathfrak{N}}}^{(N_E+1)}. \quad (2.81e)$$

Die Kreuzkovarianz $\mathbf{C}_{\underline{\theta}\underline{y}}^{(N_E+1)}$ ist definiert als

$$\mathbf{C}_{\underline{\theta}\underline{y}}^{(N_E+1)} = E \left\{ (\underline{\theta} - \underline{\theta}^*) \left(\underline{y}^{(N_E+1)} - \underline{y}^{(N_E+1)*} \right)^T \right\} \quad (2.82a)$$

Mit (2.57), (2.59) und (2.80) folgt

$$\approx E \left\{ (\underline{\theta} - \underline{\theta}^*) \left(\underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) + \mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \right) \right\} \quad (2.82b)$$

$$+ \underline{\mathfrak{N}}^{(N_E+1)} - \underline{\mathcal{G}}(\underline{x}_0^{(N_E+1)}, \mathbf{U}^{(N_E+1)}, \underline{\theta}^*) \right)^T \left. \right\} \quad (2.82c)$$

$$= E \left\{ (\underline{\theta} - \underline{\theta}^*) \left(\mathbf{G}^{(N_E+1)}(\underline{\theta} - \underline{\theta}^*) \right)^T \right\} \quad (2.82d)$$

Auswertung des Erwartungswertes mit $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E) = E \left\{ (\underline{\theta} - \underline{\theta}^*) (\underline{\theta} - \underline{\theta}^*)^T \right\}$ ergibt

$$= \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_E+1)T}. \quad (2.82e)$$

2.3. Optimale-Versuchsplanung

Substitution von $\mathbf{C}_{\underline{y}}^{(N_{E+1})}$ und $\mathbf{C}_{\underline{\theta}}^{(N_{E+1})}$ in (2.79) mit (2.82) und (2.81) liefert

$$\begin{aligned} & \mathbf{C}_{\underline{\theta}}(1, \dots, N_E, N_E + 1) \\ &= \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) - \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} \end{aligned} \quad (2.83a)$$

$$\cdot \left(\mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} \cdot \left(\mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} + \mathbf{C}_{\underline{\mathfrak{y}}}^{(N_{E+1})} \right)^{-1} \right)^T$$

$$\begin{aligned} &= \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) - \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} \\ &\cdot \left(\left(\mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} + \mathbf{C}_{\underline{\mathfrak{y}}}^{(N_{E+1})} \right)^{-1} \right)^T \cdot \mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}^T(1, \dots, N_E). \end{aligned} \quad (2.83b)$$

Weil $\left(\mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} + \mathbf{C}_{\underline{\mathfrak{y}}}^{(N_{E+1})} \right)^{-1}$ und $\mathbf{C}_{\underline{\theta}}(1, \dots, N_E)$ symmetrisch sind, folgt

$$\begin{aligned} &= \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) - \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} \\ &\cdot \left(\mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E) \mathbf{G}^{(N_{E+1})T} + \mathbf{C}_{\underline{\mathfrak{y}}}^{(N_{E+1})} \right)^{-1} \cdot \mathbf{G}^{(N_{E+1})} \mathbf{C}_{\underline{\theta}}(1, \dots, N_E). \end{aligned} \quad (2.83c)$$

Anwendung des Matrix Inversions Lemmas $(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{B} (\mathbf{C}^{-1} + \mathbf{D} \mathbf{A}^{-1} \mathbf{B})^{-1} \mathbf{D} \mathbf{A}^{-1}$ ergibt

$$= \left(\mathbf{C}_{\underline{\theta}}^{-1}(1, \dots, N_E) + \mathbf{G}^{(N_{E+1})T} \mathbf{C}_{\underline{\mathfrak{y}}}^{(N_{E+1})-1} \mathbf{G}^{(N_{E+1})} \right)^{-1}, \quad (2.83d)$$

was äquivalent zu (2.69) aus dem Fisher-Ansatz ist. Auch hier ergibt sich für ein System, welches linear in den Parametern ist, eine exakte Berechnung der Kovarianzmatrix des Parameterfehlers.

q. e. d.

Dieses Verfahren ist somit äquivalent zum Fisher-Ansatz, wenn \mathbf{G} linear bezüglich $\underline{\theta}$ ist oder bei nichtlinearen Modellen eine Approximation erster Ordnung zur Berechnung der benötigten Kovarianzen verwendet wird. Damit verbleiben jedoch die numerischen Probleme, wie sie im Abschnitt 2.1.2 beschrieben sind. Das vorgestellte Verfahren ist jedoch nicht auf die Approximation erster Ordnung zur Berechnung der benötigten statistischen Momente beschränkt. Es können somit auch die anderen in Abschnitt 2.1 vorgestellten Methoden der stochastischen Simulation verwendet werden. Insbesondere die Verwendung der Unscented-Transformation bietet sich hier an, da hier für die Berechnung der statistischen Momente einer geplanten Versuchstrajektorie keine Berechnung von Ableitungen nötig sind und somit Probleme der Fisher'schen-Informationsmatrix umgangen werden können. Ausgehend von einer Berechnung der benötigten statistischen Momente, kann dann mit dem Optimierungsproblem (2.70) und einem der möglichen Kriterien (2.71) ein optimaler Versuch geplant werden.

2.4. Modellbasierte Prozessplanung und -regelung

Für die Planung und Regelung dynamischer Prozesse existieren viele mathematische Methoden. Einige nutzen beispielsweise ausschließlich statistische Zusammenhänge, andere linearisierte Prozessbeschreibungen. Die mächtigsten Methoden zur Prozesskontrolle basieren bis heute auf chemisch, physikalisch fundierten First-Principle-Modellen. Auf diese Verfahren und darauf aufbauende Erweiterungen soll im Folgenden näher eingegangen werden.

2.4.1. Trajektorienplanung

Einfache modellbasierte Prozessführungskonzepte basieren auf der Steuerung entlang einer offline geplanten Trajektorie. Die Planung von optimalen Prozessverläufen ist vor allem für Semi-/Fed-Batchprozesse bzw. An-/Abfahrprozesse von kontinuierlich betriebenen Anlagen interessant. Aufgrund der hierbei meist resultierenden dynamischen Trajektorie ist eine Beschreibung der Prozesse mit nichtlinearen Modellen unumgänglich und eine manuelle Planung schwierig. Den optimalen Stellgrößenverlauf $\mathbf{U}^* = \{\underline{u}^*(t), t \in [t_0, t_E]\}$ und die optimale Anfangsbedingung \underline{x}_0^* einer Trajektorie erhält man im Allgemeinen durch Lösung eines nichtlinearen Optimierungsproblems der Form

$$\begin{aligned} \{\underline{x}_0^*, \mathbf{U}^*\} &= \arg \min_{\underline{x}_0, \mathbf{U}} (\Phi_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})) \\ \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\ t_0 &\leq t \leq t_{\text{Ende}} \\ \underline{x}_{0,\min} &\leq \underline{x}_0 \leq \underline{x}_{0,\max} \\ \underline{u}_{\min} &\leq \underline{u}(t) \leq \underline{u}_{\max} \\ \underline{r}(\underline{x}(t)) &\leq \underline{0}, \quad \underline{r} \in \mathbb{R}^{L_r}. \end{aligned} \tag{2.84}$$

$\Phi_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})$ beschreibt das Gütefunktional der Trajektorienplanung, $t_0 \leq t \leq t_{\text{Ende}}$ den Zeithorizont der Simulation, \underline{u}_{\min} und \underline{u}_{\max} Stellgrößenbeschränkungen und $\underline{r}(\underline{x}(t))$ die Systembeschränkungen als Funktion des Zustandes. Je nach Anwendungsfall können bei der Trajektorienplanung auch nur Teile der Anfangsbedingung beziehungsweise Teile der Stellgrößentrajektorie optimal eingestellt werden.

Das Gütefunktional ergibt sich üblicherweise durch eine Gegenüberstellung der zu erwartenden Erlöse und der zu erwartenden Kosten einer Trajektorie. Dies wird z.B. mit

$$\Phi_{\text{TP}} = \frac{\sum \text{Kosten}}{\sum \text{Erlös}} \quad \text{oder} \quad \Phi_{\text{TP}} = \sum \text{Kosten} - \sum \text{Erlös} \tag{2.85}$$

erreicht. Die Kosten setzen sich aus Edukt bzw. Fütterungskosten, Reinigungsaufwand, Personal- und Anlagenbelegungskosten und je nach Umfang der Betrachtung aus vielen

weiteren Einflussgrößen zusammen. Wird die Produktionszeit explizit in Form von Kosten für die Anlagenbelegung beachtet, ergibt sich eine so genannte Raum-Zeit-Ausbeute. Der Erlös eines Prozesses ergibt sich aus dem Preis und der Menge des produzierten Produktes. Somit resultiert die mathematische Darstellung

$$\sum \text{Erlös} \sim \text{Menge}_{\text{Produkt}}(t_{\text{Ende}}). \quad (2.86)$$

Auch ohne explizite Betrachtung der Prozesslaufzeit im Rahmen der Kosten, kann mit leichten Veränderungen der Bewertung des Erlöses eine schnelle und stabile Umsetzung forciert werden. Es ergeben sich dann Integralschreibweisen

$$\sum \text{Erlös} \sim \int_{t_0}^{t_{\text{Ende}}} \text{Menge}_{\text{Produkt}}(t) dt, \quad (2.87)$$

oder Time-Integralschreibweisen

$$\sum \text{Erlös} \sim \int_{t_0}^{t_{\text{Ende}}} t \cdot \text{Menge}_{\text{Produkt}}(t) dt. \quad (2.88)$$

für das Gütefunktional.

2.4.2. Nichtlineare-Modellprädiktive-Folgeregelung

Erweist sich die Strategie der optimalen Steuerung als nicht ausreichend, ist die Anwendung von Regelungsverfahren notwendig. Aufgrund der aus der Trajektorienplanung resultierenden dynamischen Sollwerttrajektorie und des insbesondere bei Semi-/Fed-Batch-Verfahren ausgeprägten nichtlinearen Prozessverhaltens ist die Anwendung klassischer linearer Regelungsansätze nur bedingt Erfolg versprechend. Die Nichtlineare-Modellprädiktive-Folgeregelung (NMPC) entlang einer zuvor berechneten Trajektorie stellt eines der leistungsfähigsten Verfahren dar, mit dem die Auswirkungen von Prozessstörungen und Modellfehlern minimiert werden können.

Ziel dieses Regelungsverfahrens ist es, einer offline geplanten Sollwerttrajektorie zu folgen. Die Abbildung 2.4 zeigt prinzipiell die Arbeitsweise der Nichtlinearen-Modellprädiktiven-Folgeregelung für ein SISO-System. Ausgehend von einem am Zeitpunkt t_k geschätzten Zustand \hat{x}_k , wird eine Prädiktion des zukünftigen Prozessverhaltens mit dem nichtlinearen Systemmodell berechnet. Mit Hilfe nichtlinearer Optimierung werden nun die zukünftigen Stellgrößen innerhalb des Stellhorizontes T_S so variiert, dass sich die Prädiktion des Systemverhaltens innerhalb des Prädiktionshorizontes T_P in einem optimalen Sinn der offline geplanten Trajektorie annähert. Da die nichtlineare Optimierung eine gewisse Rechenzeit T_C benötigt, ist der erste Stelleingriff erst nach dieser Rechenzeit möglich. Der optimale Stellgrößenverlauf $\mathbf{U}_k^* = \{\underline{u}^*(t), t \in [t_k + T_C, t_k + T_S]\}$ resultiert aus der Lösung des Minimierungsproblems

2. Mathematische Methoden

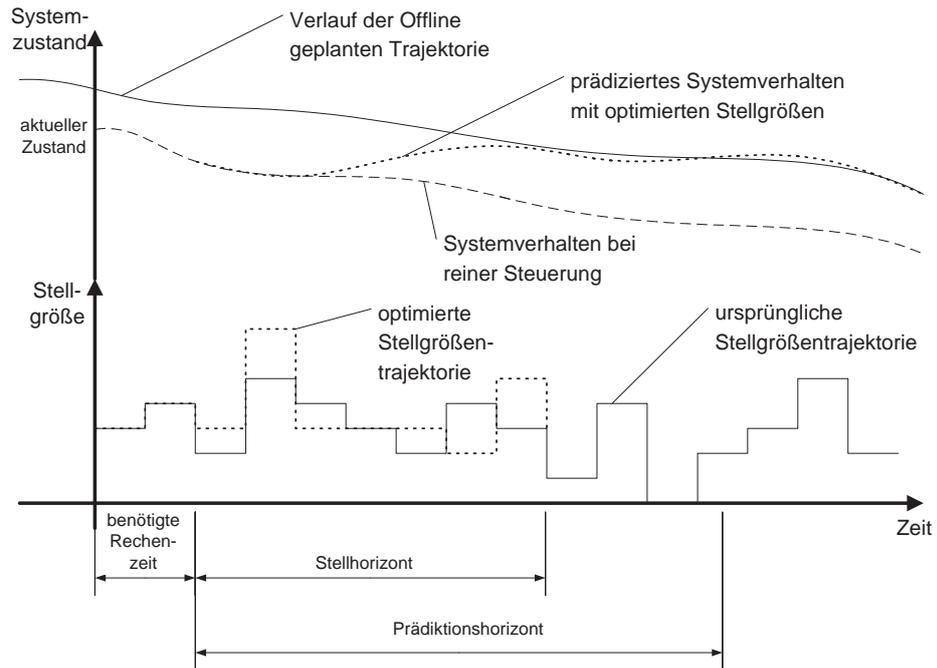


Abbildung 2.4.: Schematische Darstellung der Funktionsweise der Nichtlinearen-Modellprädiktiven-Folgeregelung für ein SISO-System.

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\Phi_{\text{NMPC}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \theta)) \\
 \text{mit: } \dot{\mathbf{x}} &= \mathbf{f}(t, \mathbf{x}(t), \mathbf{u}(t), \theta), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_k + T_P \\
 \mathbf{u}_{\min} &\leq \mathbf{u}(t) \leq \mathbf{u}_{\max} \\
 \mathbf{r}(\mathbf{x}(t)) &\leq \mathbf{0}, \quad \mathbf{r} \in \mathbb{R}^{L_r}.
 \end{aligned} \tag{2.89}$$

Bei der Problemstellung der Modellprädiktiven-Folgeregelung handelt es sich um ein Trajektorienfolgeproblem. Als Gütefunktional Φ_{NMPC} für solche Probleme werden meist Formulierungen verwendet, die zum einen den Abstand von der Trajektorie, zum anderen den Energieaufwand zum Erreichen des Ziels enthalten. Eine oft verwendete Formulie-

zung des NMPC-Gütefunktionalis ist

$$\begin{aligned}
 \Phi_{\text{NMPC}} = & \int_{t_k+T_C}^{t_k+T_P} \left(\underline{x}(t|\hat{\underline{x}}_k) - \underline{x}^{(\text{TP})}(t) \right)^T \mathbf{W}_{\underline{x}} \left(\underline{x}(t|\hat{\underline{x}}_k) - \underline{x}^{(\text{TP})}(t) \right) dt \\
 & + \int_{t_k+T_C}^{t_k+T_P} \left(\underline{y}(t|\hat{\underline{x}}_k) - \underline{y}^{(\text{TP})}(t) \right)^T \mathbf{W}_{\underline{y}} \left(\underline{y}(t|\hat{\underline{x}}_k) - \underline{y}^{(\text{TP})}(t) \right) dt \\
 & + \int_{t_k+T_C}^{t_k+T_S} \left(\underline{u}(t) - \underline{u}^{(\text{TP})}(t) \right)^T \mathbf{W}_{\underline{u}} \left(\underline{u}(t) - \underline{u}^{(\text{TP})}(t) \right) dt.
 \end{aligned} \tag{2.90}$$

Die ersten zwei Integrale beschreiben den Abstand von der Solltrajektorie $(\cdot)^{(\text{TP})}$, das dritte Integral den benötigten Stellgrößenaufwand. Die enthaltenen Gewichtungsmatrizen $\mathbf{W}_{(\cdot)}$ bewerten die einzelnen Terme gegeneinander. Meist werden hierfür Diagonalmatrizen verwendet. Insbesondere für kontinuierliche Prozesse finden sich in der Literatur weitere Summanden, wie z.B.

$$\int_{t_k+T_C}^{t_k+T_S} \dot{\underline{u}}^T(t) \mathbf{W}_{\dot{\underline{u}}} \dot{\underline{u}}(t) dt. \tag{2.91}$$

Die Werte in den Matrizen können sowohl konstant als auch vom Zustand, Messvektor oder Stellvektor abhängen. Letzteres ist insbesondere nötig, wenn sich die Größenordnungen der beteiligten Größen stark unterscheiden oder innerhalb des Prädiktionshorizontes stark variieren. Hier ist eine Normierung der beteiligten Größen notwendig. Eine normierende Form für $\mathbf{W}_{\underline{x}}$ ist beispielsweise

$$\mathbf{W}_{\underline{x}} = \begin{pmatrix} \frac{w_{11}}{x_1^{(\text{TP})}(t)} & 0 & \cdots & 0 \\ 0 & \frac{w_{22}}{x_2^{(\text{TP})}(t)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & \frac{w_{L_x L_x}}{x_{L_x}^{(\text{TP})}(t)} \end{pmatrix} \tag{2.92}$$

mit den Gewichtungsfaktoren w_{ii} für die einzelnen Zustände.

2.4.3. Online-Trajektorienplanung

Ein weiteres Verfahren zur modellbasierten Prozessregelung ist die Online-Trajektorienplanung [Hei04]. Im Gegensatz zur Nichtlinearen-Modellprädiktiven-Folgerregelung, die zu jedem Zeitpunkt einen optimalen Weg zurück zur offline geplanten Trajektorie sucht,

2. Mathematische Methoden

löst die Online-Trajektorienplanung das Trajektorienplanungsproblem wiederholt online. Dazu müssen sowohl der Stell-, als auch der Prädiktionshorizont jeweils bis zum Ende des Prozesses erweitert werden. Der optimale zukünftige Stellgrößenverlauf zum Zeitpunkt t_k , $\mathbf{U}_k^* = \{\underline{u}^*(t), t \in [t_k + T_C, t_{\text{Ende}}]\}$, resultiert dann aus der Lösung des Minimierungsproblems

$$\begin{aligned} \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\Phi_{\text{OT}}(\hat{x}_k, \mathbf{U}_k, \theta)) \\ \text{mit: } \dot{x} &= f(t, x(t), u(t), \theta), \quad x(t = t_k) = \hat{x}_k \\ t_k &\leq t \leq t_{\text{Ende}} \\ \underline{u}_{\min} &\leq u(t) \leq \underline{u}_{\max} \\ \underline{r}(x(t)) &\leq \underline{0}, \quad r \in \mathbb{R}^{L_r}. \end{aligned} \tag{2.93}$$

Als Gütefunktion Φ_{OT} kann in den meisten Fällen direkt das Funktional der Trajektorienplanung Φ_{TP} verwendet werden. Die Online-Trajektorienplanung sucht damit zu jedem Zeitpunkt des Prozesses einen neuen optimalen Weg zu z.B. maximaler Produktivität. Aufgrund des längeren Prädiktions- und Stellhorizontes ist der numerische Aufwand zur Lösung des Online-Trajektorienplanungsproblems jedoch weitaus größer als bei der Nichtlinearen-Modellprädiktiven-Folgeregelung.

Die Güte und Robustheit modellbasierter Regelungsverfahren ist von verschiedenen Faktoren abhängig. Je häufiger die Messgrößen bzw. Zustandsschätzungen zugänglich sind und je weniger Zeit für die Optimierung benötigt wird, desto besser ist das Prozessergebnis, welches in einem geregelten Prozesslauf erwartet werden kann. Viele Prozesse zeichnen sich allerdings dadurch aus, dass aufgrund komplexer Mess- und Zustandsschätzverfahren die Frequenz, in der die Stellgrößen neu berechnet werden können, sich nicht beliebig steigern lässt [Hei04, KHK07].

2.4.4. Diskretisierung der Güte- und Beschränkungsfunktion

Da die analytische Lösung des Differentialgleichungssystems (2.1a) nur in seltenen Fällen möglich ist, werden für diese Lösung numerische Lösungsverfahren verwendet. Als Folge ist es nicht möglich, einen kontinuierlichen Verlauf der Zustands- bzw. Messgrößen-trajektorie zu berechnen. In diesen Fällen werden die Integrale (2.87), (2.88) bzw. (2.90) ausgehend von einer Abtastung des Zeithorizonts mit einer Abtastzeit Δt durch Summen approximiert. Verwendung einer Streifenapproximation für (2.90) liefert beispielels-

weise

$$\begin{aligned}
 \Phi_{\text{NMPC}} = & \sum_{i=N_C+1}^{N_P} \left(\underline{x}(t_{k+i}|\hat{\underline{x}}_k) - \underline{x}^{(\text{TP})}(t_{k+i}) \right)^T \mathbf{W}_x \left(\underline{x}(t_{k+i}|\hat{\underline{x}}_k) - \underline{x}^{(\text{TP})}(t_{k+i}) \right) \Delta t \\
 & + \sum_{i=N_C+1}^{N_P} \left(\underline{y}(t_{k+i}|\hat{\underline{x}}_k) - \underline{y}^{(\text{TP})}(t_{k+i}) \right)^T \mathbf{W}_y \left(\underline{y}(t_{k+i}|\hat{\underline{x}}_k) - \underline{y}^{(\text{TP})}(t_{k+i}) \right) \Delta t \\
 & + \sum_{i=N_C}^{N_S-1} \left(\underline{u}(t_{k+i}) - \underline{u}^{(\text{TP})}(t_{k+i}) \right)^T \mathbf{W}_u \left(\underline{u}(t_{k+i}) - \underline{u}^{(\text{TP})}(t_{k+i}) \right) \Delta t. \quad (2.94)
 \end{aligned}$$

Aufgrund der zeitlichen Abfolge, also \underline{u}_k beeinflusst nur \underline{x}_{k+i} mit $i > 0$ und der Annahme, dass kein direkter Durchgriff bezüglich \underline{y} besteht, wurde bei dieser Diskretisierung die Streifenapproximation für \underline{x} und \underline{y} rückwärts, also bezogen auf den Endwert eines jeweiligen Diskretisierungsabschnitts, für \underline{u} jedoch vorwärts durchgeführt.

Aus den o.g. Gründen kann auch die Beschränkungsfunktion $r(\underline{x}(t))$ nicht kontinuierlich ausgewertet werden. Diese wird in dieser Arbeit daher an diskreten Zeitpunkten $t_k = k \cdot \Delta t$, mit $k = 1, 2, \dots$ ausgewertet. Jeder dieser Abtastpunkte liefert dann eine für den Optimierer zu implementierende Beschränkungsfunktion. Bei N Abtastpunkten ergäbe sich zum Beispiel eine Beschränkungsfunktion

$$\underline{r}_{\text{Optimierer}} = \begin{pmatrix} r(t_0) \\ r(t_1) \\ r(t_2) \\ \vdots \\ r(t_N) \end{pmatrix}. \quad (2.95)$$

In allen im Rahmen dieser Arbeit implementierten Funktionen ist die Diskretisierung wie hier beschrieben umgesetzt. Im Folgenden wird ausschließlich die Abtastzeit Δt angegeben, nicht aber jedes mal explizit die Diskretisierung.

2.5. Robuste Modellbasierte Prozessplanung und -führung

Im Rahmen der klassischen Prozessplanung und -führung werden die genauen Auswirkungen von Modellfehlern auf die geplante Trajektorie nicht beachtet. Durch die Aufschaltung von leistungsfähigen, modellbasierten Regelungsalgorithmen, wie der Nichtlinearen-Modellprädiktiven-Folgeregelung oder der Online-Trajektorienplanung, wird versucht, die Abweichungen von der optimalen Trajektorie zu minimieren und den Prozess innerhalb der Beschränkungen zu halten. Auswirkungen der Unsicherheiten können dennoch den Prozessverlauf, wenn auch verringert, negativ beeinflussen. Zusätzlich bedingen solche aufwändigen Regelungsverfahren einen großen Aufwand beim

2. Mathematische Methoden

Aufbau der hierfür benötigten Infrastruktur, vom Zustandsschätzverfahren bis zur im online Betrieb applizierbaren nichtlinearen Optimierung. Es ist somit sinnvoll, die zu erwartenden Schwankungen eines Prozesses nicht erst im Betrieb, sondern bereits bei der Planung zu berücksichtigen und somit u.U. sogar die Notwendigkeit des Einsatzes eines Regelungsverfahrens zu eliminieren.

Ein mathematischer Ansatz, der die beschriebenen Unsicherheiten im Rahmen einer robusten Trajektorienplanung explizit berücksichtigt, basiert auf einer Erweiterung des klassischen Gütefunctionals (2.84) in eine Value@Risk-Formulierung [RU00]

$$\begin{aligned} \{\underline{x}_0^*, \mathbf{U}^*\} &= \arg \min_{\underline{x}_0, \mathbf{U}} (\overline{\Phi}_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta}) + \gamma \sigma_{\Phi_{\text{TP}}}^2) \\ \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}) + \underline{\xi}, \quad \underline{x}(t=0) = \underline{x}_0 \\ t_0 &\leq t \leq t_{\text{Ende}} \\ \underline{x}_{0,\min} &\leq \underline{x}_0 \leq \underline{x}_{0,\max} \\ \underline{u}_{\min} &\leq \underline{u}(t) \leq \underline{u}_{\max} \\ \bar{r}_i(\underline{x}(t)) + \lambda \sigma_{r_i} &\leq 0, \quad i = 1, \dots, L_r. \end{aligned} \tag{2.96}$$

Sowohl die Anfangsbedingung \underline{x}_0 , als auch der Parametervektor $\underline{\theta}$ und das Systemrauschen $\underline{\xi}$ sind dabei zufällige Größen und werden als Normalverteilungen angenommen. Der „Value“ der Trajektorie ist hier der Mittelwert des Gütefunctionals $\overline{\Phi}_{\text{TP}}$, das Risiko ist durch die Varianz des Gütefunctionals $\sigma_{\Phi_{\text{TP}}}^2$ beschrieben. Anstelle der Varianz $\sigma_{\Phi_{\text{TP}}}^2$ wäre auch die Standardabweichung $\sigma_{\Phi_{\text{TP}}}$ als Risikoterm verwendbar. In dieser Arbeit wird ausschließlich die Varianz als Risikoterm verwendet, da nach [RU00] beide Formulierungen ineinander umrechenbar sind.

Eine Gewichtung zwischen „Value“ und „Risk“ ist durch den Faktor γ möglich. Durch die Erweiterung der Beschränkungsfunktion $r(\underline{x}(t))$ um den Term $\lambda \sigma_{r_i}$ wird zusätzlich ein Sicherheitsabstand eingeführt, bevor eine Grenze verletzt wird. Die Werte $\sigma_{r_i}^2$ sind dabei die Hauptdiagonalelemente der Kovarianzmatrix \mathbf{C}_r der Beschränkungsfunktion $r(\underline{x}(t))$. Mit diesem Abstand in Abhängigkeit von der Standardabweichung der Elemente der Beschränkungsfunktion wird somit kein fester, sondern ein dynamischer Abstand zu den Systemgrenzen festgelegt, der je nachdem, ob sich Unsicherheiten an einem Punkt der Trajektorie stark oder gering auswirken, größer oder kleiner eingestellt wird. Nimmt man eine Normalverteilung für $r(\underline{x}(t))$ an und setzt λ beispielsweise auf 3, so ergibt sich die Wahrscheinlichkeit, dass die Beschränkung i nicht verletzt wird, zu

$$P(r_i(\underline{x}(t)) \leq 0) \geq \text{erf}(3/\sqrt{2}) + \frac{1 - \text{erf}(3/\sqrt{2})}{2} = 99,87\%. \tag{2.97}$$

Zur Berechnung der hier benötigten statistischen Momente des Gütefunctionals und der Beschränkungsfunktion ist die Verwendung des stochastischen Systemmodells (2.2) notwendig. Zur Approximation der statistischen Eigenschaften kann eine beliebige der in Abschnitt 2.1 eingeführten Methoden verwendet werden. Der Einsatz der Monte-Carlo-Simulation ist jedoch aufgrund des hohen numerischen Aufwands und des zufälligen

2.5. Robuste Modellbasierte Prozessplanung und -führung

Anteils im Algorithmus nicht zu empfehlen. Die stochastische Simulation auf Basis einer Approximation erster Ordnung verursacht starke Approximationsfehler bei der Berechnung der benötigten statistischen Momente. Da der numerische Aufwand zur Berechnung der Approximation zweiter Ordnung mit der Unscented-Transformation von der gleichen Größenordnung ist wie der Aufwand zur Berechnung der Approximation erster Ordnung, ist die Verwendung der Unscented-Transformation besonders effektiv.

Die gleiche Erweiterung des Gütefunktional in eine Value@Risk-Schreibweise kann auch für die online Prozessführungsverfahren Nichtlineare-Modellprädiktive-Folgerregelung und Online-Trajektorienplanung eingeführt werden. Es ergeben sich die robuste Nichtlineare-Modellprädiktive-Folgerregelung mit dem zu lösenden Problem

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} \left(\bar{\Phi}_{\text{NMPC}}(t_k, \hat{\mathbf{x}}_k, \mathbf{U}, \theta) + \gamma \sigma_{\Phi_{\text{NMPC}}}^2 \right) \\
 \text{mit: } \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \underline{\mathbf{u}}(t), \theta) + \dot{\underline{\xi}}, \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_k + T_P \\
 \underline{\mathbf{u}}_{\min} &\leq \underline{\mathbf{u}}(t) \leq \underline{\mathbf{u}}_{\max} \\
 \bar{r}_i(\underline{\mathbf{x}}(t)) + \lambda \sigma_{r_i} &\leq 0, \quad i = 1, \dots, L_r
 \end{aligned} \tag{2.98}$$

und die robuste Online-Trajektorienplanung

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} \left(\bar{\Phi}_{\text{OT}}(t_k, \hat{\mathbf{x}}_k, \mathbf{U}, \theta) + \gamma \sigma_{\Phi_{\text{OT}}}^2 \right) \\
 \text{mit: } \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \underline{\mathbf{u}}(t), \theta) + \dot{\underline{\xi}}, \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_{\text{Ende}} \\
 \underline{\mathbf{u}}_{\min} &\leq \underline{\mathbf{u}}(t) \leq \underline{\mathbf{u}}_{\max} \\
 \bar{r}_i(\underline{\mathbf{x}}(t)) + \lambda \sigma_{r_i} &\leq 0, \quad i = 1, \dots, L_r.
 \end{aligned} \tag{2.99}$$

Die Anfangsbedingung $\hat{\mathbf{x}}_k$ resultiert z.B. aus der Anwendung eines modellbasierten Messverfahrens und ist, da hierbei zufällig gestörte Messwerte die Grundlage bilden, ebenfalls eine zufällige Größe.

Insbesondere das Gütefunktional der Nichtlinearen-Modellprädiktiven-Folgerregelung setzt sich nach Diskretisierung des Integrals meist aus einer Summe von Teilfunktionalen, die jeweils zu einem Zeitpunkt im Prädiktionshorizont ausgewertet werden, zusammen

$$\Phi_{\text{NMPC}}(t_k, \hat{\mathbf{x}}_k, \mathbf{U}, \theta) = \sum_{i=N_C}^{N_P} \phi_{\text{NMPC}}^{(i)}(t_{k+i}, \hat{\mathbf{x}}_k, \mathbf{U}, \theta). \tag{2.100}$$

Damit kann das Optimierungsproblem der robusten Nichtlinearen-Modellprädiktiven-

2. Mathematische Methoden

Folgeregelung auch als

$$\begin{aligned} \mathbf{U}^* &= \arg \min_{\mathbf{U}} \sum_{i=N_C}^{N_P} \left(\bar{\phi}_{\text{NMPC}}^{(i)}(t_{k+i}, \hat{\mathbf{x}}_k, \mathbf{U}, \underline{\theta}) + \gamma_i \sigma_{\phi_{\text{NMPC}}^{(i)}}^2 \right) \\ \text{mit: } \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \mathbf{u}(t), \underline{\theta}) + \underline{\xi}, \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\ t_k &\leq t \leq t_k + T_P \\ \underline{\mathbf{u}}_{\min} &\leq \mathbf{u}(t) \leq \underline{\mathbf{u}}_{\max} \\ \bar{r}_i(\mathbf{x}(t)) + \lambda \sigma_{r_i} &\leq 0, \quad i = 1, \dots, L_r \end{aligned} \quad (2.101)$$

umgeschrieben werden, wobei hier die Value@Risk-Formulierung für jeden Summanden des Gütefunktional einzeln angewendet wird. Zusätzlich wird der Faktor γ , der Wert und Risiko gegeneinander gewichtet, auch für jeden Summanden unterschiedlich angenommen. Obwohl diese Schreibweise der ursprünglichen Darstellung stark ähnelt, kann sie nur daraus abgeleitet werden, wenn man die Kreuzkovarianzen der einzelnen Summanden des Gütefunktional vernachlässigt. Diese Formulierung des Gütefunktional der robusten Nichtlinearen-Modellprädiktiven-Folgeregelung bietet nun die Möglichkeit, Variationen des Gütefunktional Φ_{NMPC} über den Prädiktionshorizont unterschiedlich zu gewichten. Beispielsweise durch Wahl von

$$\gamma_i \sim (t_{k+i} - t_k)^{-1} \quad \text{oder} \quad \gamma_i \sim e^{-(t_{k+i} - t_k)} \quad (2.102)$$

werden Variationen in naher Zukunft stärker gewichtet als Variationen am Ende des Prädiktionshorizontes. Dies dürfte eine der meist gewählten Einstellungen sein, da damit dem wiederholten Eingreifen des Reglers Rechnung getragen werden kann. Ebenfalls wäre es denkbar, die Ausdehnung λ des Konfidenzintervalls der Beschränkungsfunktion $\underline{r}(\mathbf{x}(t))$ ebenfalls variabel zu wählen. Dies wurde jedoch im Rahmen dieser Arbeit nicht untersucht.

2.6. Numerische Methoden zur nichtlinearen Optimierung

Die Lösung der in den vorangegangenen Abschnitten angegebenen Optimierungsprobleme ist in den seltensten Fällen analytisch möglich. Daher ist der Einsatz numerischer nichtlinearer Optimierungsverfahren notwendig. In der Literatur finden sich viele Veröffentlichungen zum Thema Optimierung. So seien z.B. [Pap96, Poh99, MD03, KW03] genannt. Allein eine vollständige Literaturliste würde den Rahmen dieser Arbeit sprengen. Im Folgenden soll somit zunächst ein kurzer Überblick über numerische Optimierungsverfahren gegeben werden, die sich für die genannten Probleme eignen.

2.6.1. Numerische Optimierungsverfahren

Methoden zur Lösung von Optimierungsproblemen lassen sich in verschiedene Kategorien einordnen. So existieren Verfahren zur linearen und zur nichtlinearen Optimierung, Verfahren für kontinuierliche Eingangsgrößen und diskrete Eingangsgrößen, Verfahren für beschränkte und unbeschränkte Probleme, globale und lokale Optimierungsverfahren. Im Rahmen dieser Arbeit sind beschränkte, nichtlineare Optimierungsprobleme mit kontinuierlichen Eingangsgrößen zu lösen, weshalb hier der Fokus auf Verfahren zur Lösung solcher Probleme liegt.

Ein beschränktes, nichtlineares Optimierungsproblem hat die allgemeine Form

$$\underline{p}^* = \arg \min_{\underline{p}} \Phi(\underline{p}) \quad \underline{p} \in \mathbb{R}^{L_{\underline{p}}} \quad (2.103a)$$

$$\text{mit: } \underline{p}_{\min} \leq \underline{p} \leq \underline{p}_{\max} \quad \text{Eingangsbeschränkungen} \quad (2.103b)$$

$$\underline{r}_{\text{eq}}(\underline{p}) = \underline{0} \quad \text{Gleichungsnebenbedingungen} \quad (2.103c)$$

$$\underline{r}_{\text{ineq}}(\underline{p}) \leq \underline{0} \quad \text{Ungleichungsnebenbedingungen} \quad (2.103d)$$

Numerische Methoden zur Lösung solcher kontinuierlicher, nichtlinearer Probleme lassen sich zunächst in zwei verschiedene Ansätze aufteilen. Zum einen existieren gradientenbasierte Methoden, die neben dem Wert des Gütefunktional Φ auch dessen Ableitung $\frac{d\Phi}{d\underline{p}}$ nach den Optimierungsvariablen \underline{p} verwenden. Zum anderen nutzen gradientenfreie Verfahren ausschließlich Auswertungen des Gütefunktional Φ an verschiedenen Punkten. Hier sollen zunächst einige Verfahren zur Lösung des unbeschränkten Optimierungsproblems und anschließend einige Erweiterungen der Methoden für das beschränkte Problem kurz beschrieben werden.

Gradientenfreie Optimierungsverfahren

Eines der bekanntesten gradientenfreien Optimierungsverfahren ist die Nealder-Mead-Simplex Methode [NM65]. Dies ist eine Methode zur lokalen nichtlinearen Optimierung.

Das grundsätzliche Vorgehen bei diesem Verfahren ist wie folgt: Ausgehend von $L_{\underline{p}} + 1$ Auswertungen des Gütefunktional $\Phi^{(i)}$ an den Punkten $\underline{p}^{(i)}$, mit $i = 1, \dots, L_{\underline{p}} + 1$, wird der Punkt $\underline{p}^{(k)}$ mit dem größten Gütefunktionalwert durch den Mittelpunkt der von den anderen Punkten $\underline{p}^{(i)}$; $i = 1, \dots, L_{\underline{p}} + 1, i \neq k$ aufgespannten Hyperebene projiziert. Der so entstehende neue Punkt $\underline{p}'^{(k)}$ ersetzt dann den Punkt $\underline{p}^{(k)}$ und das Vorgehen wird wiederholt. Zu diesem grundsätzlichen Vorgehen existieren verschiedene Erweiterungen. Abbildung 2.5 zeigt die in der MatlabTMfunktion „fminsearch“ implementierte Methodik am Beispiel eines zweidimensionalen Problems.

Obwohl die Nealder-Simplex Methode sehr einfach ist, zeigt sie auch bei hochdimensionalen stark nichtlinearen Problemen gute Ergebnisse. Sie kann allerdings mit Ausnahme

2. Mathematische Methoden

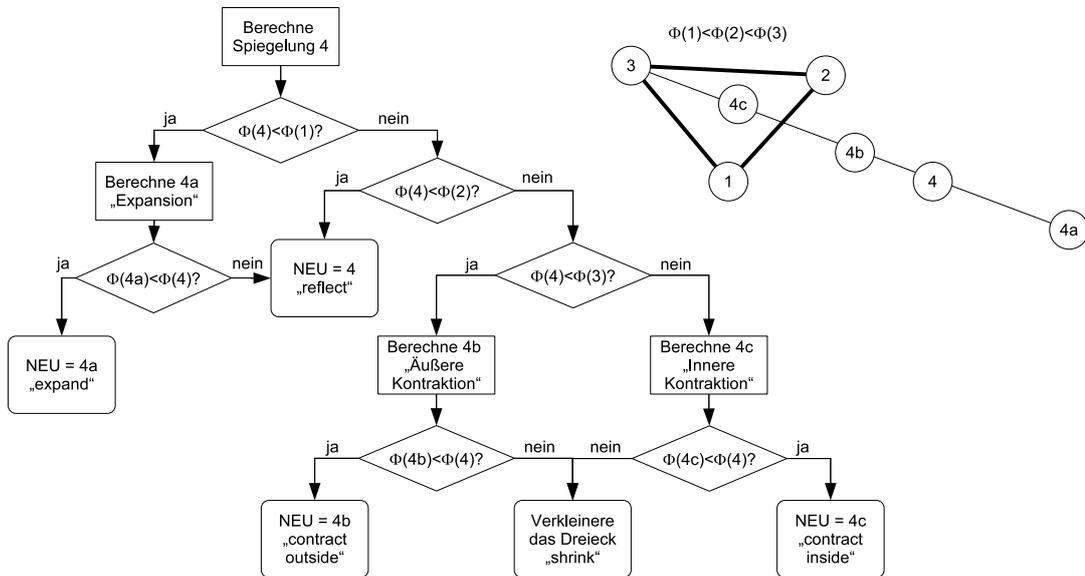


Abbildung 2.5.: Grafische Darstellung des Simplex-Verfahrens, wie es in der MatlabTM-funktion „fminsearch“ implementiert ist.

von Eingangsbeschränkungen nicht direkt zur Lösung beschränkter Probleme erweitert werden.

Eine weitere Gruppe von gradientenfreien Optimierungsverfahren sind evolutionäre Strategien. Die Grundidee solcher Verfahren ist recht einfach. Ausgehend von einem oder mehreren bekannten Punkten, dem/n Elter/n, werden durch zufällige Variation der Optimierungsparameter \underline{p} mehrere Kinder erzeugt. Aus den besten Kinder wird die nächste Elterngeneration gebildet.

Diese Verfahren garantieren keine Verbesserung des Ergebnisses von Iteration zu Iteration. Dadurch können allerdings auch lokale Minima wieder verlassen werden. Im Rahmen dieser Arbeit wurde eine Implementierung nach [Poh99] verwendet.

Gradientenbasierte Optimierungsverfahren

Gradientenbasierte Optimierungsverfahren nutzen neben dem Wert des Gütefunktionals Φ auch dessen Ableitung $\frac{d\Phi}{d\underline{p}}$ nach den Optimierungsvariablen \underline{p} . Im einfachsten Fall wird

2.6. Numerische Methoden zur nichtlinearen Optimierung

in jeder Iteration ein Schritt in Richtung des negativen Gradienten, also des steilsten Abstiegs durchgeführt. Erweiterungen dieser Verfahren verändern die Gradientenrichtung aufgrund bereits durchgeführter Schritte und auch die Schrittweite.

Zu den mächtigsten Optimierungsverfahren zählen in diesem Rahmen die SQP- oder Quasi-Newton-Verfahren, wie z.B. in [Sch82] beschrieben. Anschaulich erklärt, verwenden diese Verfahren neben der ersten Ableitung des Gütefunktional auch dessen Hessematrix. Mit diesen Informationen wird das Gütefunktional durch einen Paraboloiden approximiert und der Optimierungsschritt führt in jeder Iteration direkt in das Minimum des approximierenden Paraboloiden.

Eine kurze mathematische Herleitung sieht wie folgt aus. Um ein lokales Extremum des Problems

$$\underline{p}^* = \arg \min_{\underline{p}} \Phi(\underline{p}) \quad \underline{p} \in \mathbb{R}^{L_p} \quad (2.104)$$

zu finden, muss gelten

$$\left. \frac{d\Phi}{d\underline{p}} \right|_{\underline{p}^*} = \underline{0}^T. \quad (2.105)$$

Verwendet man zur Lösung des Gleichungssystems (2.105) ein Newton-Raphson-Verfahren, so folgt die iterative Lösungsvorschrift

$$\underline{p}^{(i+1)} = \underline{p}^{(i)} - \left(\left. \frac{d^2\Phi}{d\underline{p}^2} \right|_{\underline{p}^{(i)}} \right)^{-1} \left(\left. \frac{d\Phi}{d\underline{p}} \right|_{\underline{p}^{(i)}} \right)^T \quad (2.106)$$

Im Allgemeinen wird diese Lösungsvorschrift noch um eine Schrittweite α erweitert, so dass

$$\underline{p}^{(i+1)} = \underline{p}^{(i)} - \alpha \left(\left. \frac{d^2\Phi}{d\underline{p}^2} \right|_{\underline{p}^{(i)}} \right)^{-1} \left(\left. \frac{d\Phi}{d\underline{p}} \right|_{\underline{p}^{(i)}} \right)^T \quad (2.107)$$

folgt. Diese Schrittweite wird im Rahmen eines Line-Search-Verfahrens eingestellt.

Da bereits die Berechnung des Gradienten des Gütefunktional, insbesondere, wenn unterliegend die numerische Lösung eines Differentialgleichungssystems berechnet werden muss, sehr aufwändig ist, wird die Hessematrix oder deren Inverse üblicherweise aus vergangenen Iterationen geschätzt. Daher ergibt sich auch der Name Quasi-Newton.

Gradientenbasierte Verfahren können direkt zur beschränkten Optimierung erweitert werden. Gleichungsnebenbedingungen werden dafür mit Hilfe von Lagrange-Multiplikatoren eingebunden. Ungleichungsnebenbedingungen werden oft mit Hilfe von Active-Sets behandelt. Das grundlegende Prinzip von Active-Sets besteht darin, verletzte Ungleichungsnebenbedingungen als Gleichungsnebenbedingungen zu behandeln und erfüllte

2. Mathematische Methoden

Bedingungen zu ignorieren. Für detailliertere Beschreibungen sei hier auf die entsprechende Fachliteratur, wie z.B. [Sto85], verwiesen.

Der wesentliche Nachteil gradientenbasierter Verfahren besteht darin, dass sie ausschließlich lokale Extrema finden. Eine erfolgreiche Optimierung mit einem solchen Verfahren garantiert nicht, das globale Optimum gefunden zu haben.

2.6.2. Gradientenberechnung bei unterliegender numerischer Integration

Bei der Anwendung von Optimierungsaufgaben in der Prozessführung ist zur Berechnung des Gütefunktional zusätzlich das den Prozess beschreibende mathematische Modell zu lösen. Da die unterliegenden Prozessmodelle in vielen Fällen nichtlineare Differentialgleichungen sind, ist auch zu deren Lösung der Einsatz numerischer Lösungsverfahren notwendig. Damit können die für die Optimierung benötigten Gradienten des Gütefunktional ebenfalls nicht analytisch berechnet werden.

Eine Möglichkeit zur Berechnung des Gradienten ist in Abschnitt 2.1.2 vorgestellt. Das dort beschriebene Prinzip zur Berechnung der Sensitivitäten \mathbf{Z} kann analog zur Gradientenberechnung im Rahmen der Optimierung eingesetzt werden. Der numerische Aufwand zur Lösung des resultierenden Matrix-Differentialgleichungssystems ist jedoch sehr hoch. In den meisten Fällen wird daher eine Approximation des gesuchten Gradienten mit Hilfe von finiten Differenzenquotienten favorisiert. Das i -te Element des Gradientenvektors berechnet sich dann mit

$$\left. \frac{d\Phi}{dp_i} \right|_p = \frac{\Phi(\underline{p} + h e_i) - \Phi(\underline{p})}{h}, \quad (2.108)$$

wobei \underline{e}_i den i -ten Einheitsvektor bezeichnet.

Zur Auswertung dieser Differenzgleichung ist somit zweimal eine Berechnung des Gütefunktional Φ und somit zweimal eine numerische Lösung des unterliegenden Differentialgleichungssystems notwendig. Aufgrund der dynamischen Schrittweitenanpassung der numerischen Lösungsverfahren können sich bei der Lösung des Differentialgleichungssystems unterschiedliche Schrittfolgen $\{H\}$ und somit unterschiedliche Diskretisierungsfehler der beiden benötigten Lösungen ergeben. Der Differenzenquotient wird dann fehlerhafterweise durch zwei verschiedene Funktionen bestimmt

$$\left. \frac{d\Phi}{dp_i} \right|_p = \frac{\Phi_{\{H\}_2}(\underline{p} + h e_i) - \Phi_{\{H\}_1}(\underline{p})}{h}. \quad (2.109)$$

Aufgrund der unterschiedlichen Schrittfolgen $\{H\}_1$ und $\{H\}_2$ sind $\Phi_{\{H\}_1}$ und $\Phi_{\{H\}_2}$ jedoch bei Betrachtung aller durchgeführten Grundoperationen zwei unterschiedliche Funktionen, so dass selbst bei $h \rightarrow 0$

$$\Phi_{\{H\}_2}(\underline{p} + h e_i) - \Phi_{\{H\}_1}(\underline{p}) \neq 0 \quad (2.110)$$

folgt. Die Größenordnung dieses Fehlers kann durch verringerte Toleranzen des Differentialgleichungslösers verringert werden, was jedoch wiederum die benötigte Rechenkapazität erhöht.

Für die Bestimmung der im Rahmen der Optimierung benötigten Richtungsableitungen ist daher eine Adaption der numerischen Differentialgleichungslöser sinnvoll. Das Verfahren der „Internen Gradienten“ wurde ursprünglich von [Bau90] vorgestellt. Es verwendet zur Berechnung des Funktionswertes $\Phi(\underline{p})$ eine unterliegende numerische Simulation mit Schrittweitensteuerung. Die resultierende Schrittkette $\{H\}$ wird dabei aufgezeichnet. Zur Berechnung der weiteren für die Bestimmung des Differenzenquotienten benötigten Werte $\Phi(\underline{p} + h e_i)$ wird die automatische Schrittweitenanpassung deaktiviert und stattdessen die Schrittkette $\{H\}$ aus der ersten Auswertung des Gütefunktional verwendet. Auch wenn bei diesen Berechnungen die vorgegebenen Toleranzen für die Lösung des Differentialgleichungssystems unter Umständen nicht eingehalten werden, bleibt damit die Folge der durchgeführten Grundoperationen identisch und die Differenz zwischen $\Phi(\underline{p})$ und $\Phi(\underline{p} + h e_i)$ resultiert ausschließlich aus der Variation der Argumente des Funktional Φ .

Das Verfahren der „Internen Gradienten“ wurde für die in Matlab implementierten Lösungsverfahren ODE45 (ein explizites Runge-Kutta-Verfahren 4./5. Ordnung) und ODE15s (ein implizites Gear-Verfahren mit variabler Ordnung) implementiert. Für letzteres Verfahren sind neben den verwendeten Schrittweiten auch die jeweils verwendete Verfahrensordnung und die Anzahl der Sub-Iterationen während der Newton-Raphson basierten Lösung des impliziten Problems festzuhalten.

2.6.3. Multiple-Shooting-Optimierung

Im Allgemeinen wird zur Lösung von Optimierungsproblemen für dynamische Systeme die Single-Shoot-Methode eingesetzt. Für die Lösung des Optimierungsproblems

$$\begin{aligned}
 \underline{p}^* &= \arg \min_{\underline{p}} (\Phi(\underline{p})) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t, \underline{p}), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0(\underline{p}) \\
 t_0 &\leq t \leq t_{\text{Ende}} \\
 \underline{p}_{\min} &\leq \underline{p} \leq \underline{p}_{\max} \\
 \underline{r}(\underline{x}(t), \underline{u}(t), \underline{\theta}) &\leq \underline{0}, \quad \underline{r} \in \mathbb{R}^{L_r}.
 \end{aligned}
 \tag{2.111}$$

wird in jedem Optimierungsschritt ausgehend von der Wahl eines Parametersatzes \underline{p} der Simulationsverlauf des Zustandsvektors \underline{x} über den gesamten Zeitraum $[t_0, t_{\text{Ende}}]$, also mit einem „Schuss“, berechnet. Beim Multiple-Shooting wird der Simulationszeitraum in N_{MS} einzelne Teilabschnitte $T_i = \{t, t_{i-1} \leq t \leq t_i\}$ unterteilt. Der Startzustand $\underline{s}_{i-1} := \underline{x}(t_{i-1})$ eines jedes Teilabschnittes, sowie $\underline{s}_{N_{\text{MS}}} := \underline{x}(t_{\text{Ende}})$ werden als zusätzlicher Freiheitsgrad in das Optimierungsproblem aufgenommen. Um eine konsistente Lösung zu erhalten, muss der sich aus der Simulation des Prozessmodells über einen

2. Mathematische Methoden

Zeitraum T_i ergebende Zustand $e_i := \underline{x}(t_i | \underline{s}_{i-1})$ dem Startzustand \underline{s}_i des nächsten Simulationszeitraumes entsprechen. Dies ist durch Gleichungsnebenbedingungen zusätzlich im Optimierungsproblem zu verankern. Als erweitertes Optimierungsproblem ergibt sich somit

$$\begin{aligned} \{\underline{p}^*, \underline{s}_i^*\} &= \arg \min_{\underline{p}, \underline{s}_i} (\Phi(\underline{p})) \\ \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t, \underline{p}), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0(\underline{p}) \\ t_0 &\leq t \leq t_{\text{Ende}} \\ \underline{p}_{\min} &\leq \underline{p} \leq \underline{p}_{\max} \\ \underline{r}(\underline{x}(t), \underline{u}(t), \underline{\theta}) &\leq \underline{0}, \quad \underline{r} \in \mathbb{R}^{L_r} \\ \underline{s}_0 - \underline{x}_0(\underline{p}) &= 0 \\ \underline{s}_i - \underline{e}_i &= 0, \quad i = 1, \dots, N_{\text{MS}}. \end{aligned} \tag{2.112}$$

Abbildung 2.6 zeigt exemplarisch den in verschiedene Intervalle aufgeteilten Simulationsverlauf eines Systemzustandes x . Die einzelnen Anfangs- und Endpunkte der Simulationsintervalle sind markiert. Da bei der optimalen Lösung des Problems alle Nebenbedingungen erfüllt sein müssen, ergibt sich dann wieder ein kontinuierlicher Zustandsverlauf.

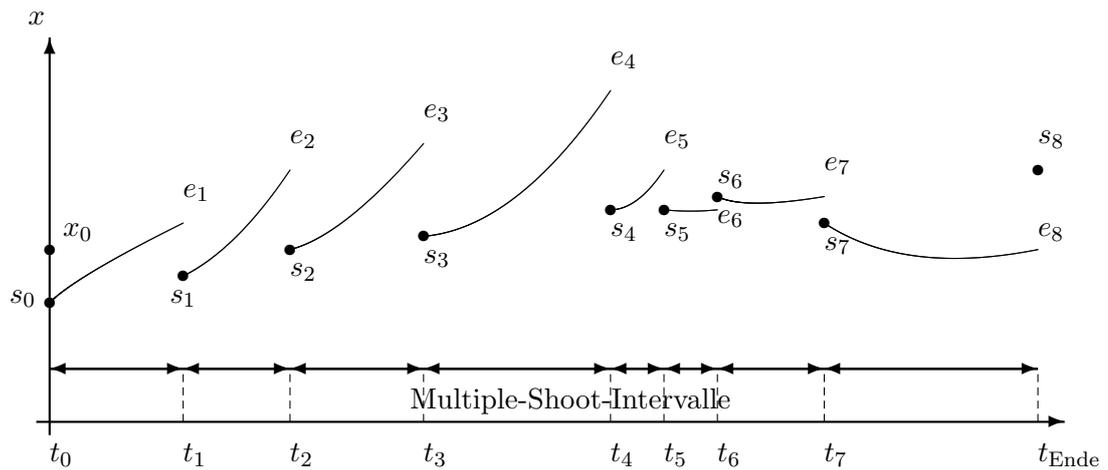


Abbildung 2.6.: Übergangsbedingungen beim Multiple-Shooting mit $N_{\text{MS}} = 8$.

Eine solche Erweiterung des Optimierungsproblems um weitere Freiheitsgrade erscheint zunächst wenig sinnvoll, da das Optimierungsproblem augenscheinlich weitaus komplexer wird. Der Augenschein trägt jedoch aus mehreren Gründen.

Durch die Aufteilung des Simulationsverlaufes in mehrere Teilstücke, wird das Problem entkoppelt. Hatte eine Stellgröße, die im Zeitintervall $[t_i, t_{i+1}]$ wirkt, beim Single-Shooting eine direkte Auswirkung auf den gesamten zukünftigen Simulationsverlauf, beschränkt sich diese nun auf die Multiple-Shooting-Intervalle, die diesen Zeitraum beinhalten.

2.6. Numerische Methoden zur nichtlinearen Optimierung

Insbesondere, wenn die Multiple-Shooting Intervalle sehr kurz sind, kann das Prozessverhalten in diesen Zeitbereichen auch noch als nahezu linear bezüglich der Optimierungsvariablen p angenommen werden. Ausgehend von einem quadratischen Gütefunktional und der Verwendung eines SQP-Verfahrens, ist dann die im Optimierer vorgenommene Approximation des Gütefunktionals durch einen Paraboloiden besonders gut.

Dennoch wird die vom Optimierer zu verarbeitende Hessematrix des Problems sehr groß. Hier ist auf die bereits beschriebene Entkopplung des Problems zu achten. Durch diese Entkopplung ist die Hessematrix nur sehr gering besetzt. Moderne Optimierungsverfahren für solch große Probleme, wie das im Rahmen dieser Arbeit verwendete SNOPT aus dem Paket Tomlab, beinhalten daher die Möglichkeit, ein Füllmuster dieser Matrix vorzugeben. Intern werden dann spezielle Algorithmen für dünn besetzte (Sparse-)Matrizen verwendet.

Schließlich sind im Rahmen der numerischen Gradientenberechnung weitaus mehr Auswertungen des Gütefunktionals notwendig, da neben $\frac{d\Phi}{dp}$ auch alle $\frac{d\Phi}{ds_i}$, $i = 0, \dots, N_{\text{MS}}$ berechnet werden müssen. Damit hier die Vorteile des Multiple-Shootings genutzt werden können, ist eine sorgfältige Programmierung der Gradientenberechnung notwendig, damit ausschließlich die Zeitbereiche simuliert werden, die sich durch Änderung eines Elements im Vektor p verändern. Andere Zeitbereiche müssen konsequent wiederverwendet werden. Dies soll im Folgenden etwas näher betrachtet werden.

Ausgangspunkt sei ein Optimierungsproblem, bei dem ausschließlich eine Stellgrößentrajektorie $u(t)$ optimiert werden soll. Eine stückweise konstante Darstellung der Steuerungsfunktion mit N_u äquidistant verteilten Stützstellen ist

$$u(t) = \begin{cases} p_1 = u_0 & , \quad t_0 \leq t < t_1 \\ p_2 = u_1 & , \quad t_1 \leq t < t_2 \\ \vdots & \\ p_{N_u} = u_{N_u-1} & , \quad t_{N_u-1} \leq t < t_{\text{Ende}} . \end{cases} \quad (2.113)$$

Zur Berechnung des Gradienten des Gütefunktionals ist zunächst eine initiale Simulation über den gesamten Zeithorizont $[t_0, t_{\text{Ende}}]$ notwendig. Zur Berechnung der Ableitung nach p_1 muss wiederum der gesamte Zeitraum simuliert werden. Für die Ableitung nach p_2 muss nur das Intervall $[t_1, t_{\text{Ende}}]$ simuliert werden, da sich der Simulationsverlauf im Intervall $[t_0, t_1]$ nicht verändert. Die Bestimmung der Ableitung nach p_2 verlangt nur die Neuberechnung des Zeitbereiches $[t_2, t_{\text{Ende}}]$, usw.. Abbildung 2.7 zeigt dies anschaulich für ein Problem mit 4 Stellgrößeneingriffen. Durchgezogene Linien markieren Zeitbereiche, die im Rahmen der Gradientenberechnung zu simulieren sind, gestrichelte Linien Zeitbereiche, für die Ergebnisse aus der initialen Simulation wiederverwendet werden können. Geht man von einer über den Integrationshorizont gleichverteilten Rechenzeit T_{Sim} für die Berechnung der Simulation über den gesamten Zeitraum aus, resultiert

2. Mathematische Methoden

damit ein Aufwand zur Berechnung des Gradienten von

$$\begin{aligned} T_{\text{SingleShoot}} &= T_{\text{Sim}} + \sum_{i=1}^{N_u} i \cdot \frac{T_{\text{Sim}}}{N_u} \\ &= \left(1 + \frac{N_u + 1}{2}\right) \cdot T_{\text{Sim}}. \end{aligned} \tag{2.114}$$

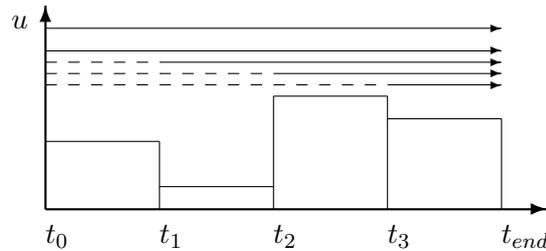


Abbildung 2.7.: Anschauliche Darstellung der zur Gradientenberechnung im Single-Shoot-Verfahren benötigten Simulationen. Durchgezogene Linien markieren Ergebnisse, die durch eine eigene Simulation ermittelt werden müssen. Gestrichelte Bereiche können bei effizienter Implementierung aus anderen Simulationsläufen kopiert werden.

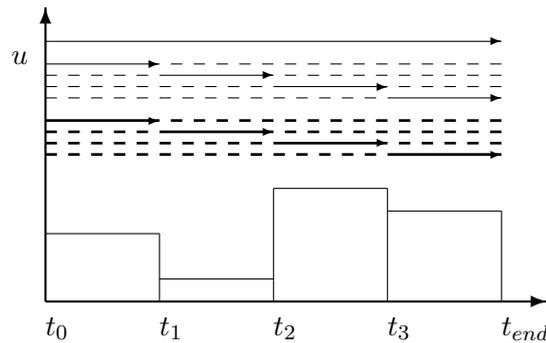


Abbildung 2.8.: Anschauliche Darstellung der zur Gradientenberechnung im Multiple-Shoot-Verfahren benötigten Simulationen.

Im besten Fall des Multiple-Shootings sind die Multiple-Shoot-Intervalle identisch zu den Intervallen der Stellgrößendiskretisierung gewählt, somit sei $N_{\text{MS}} = N_u$. Zur Berechnung des Gradienten des Gütefunktional ist auch hier zunächst eine initiale Simulation über den gesamten Zeithorizont $[t_0, t_{\text{Ende}}]$ notwendig. Zur Berechnung der Ableitung des Gütefunktional nach p_1 muss nun allerdings nur der Zeitraum $[t_0, t_1]$ neu berechnet werden, für die Ableitung nach p_2 nur $[t_1, t_2]$, usw.. Zusätzlich sind jedoch auch die Ableitungen nach den zusätzlichen Optimierungsvariablen s_i , $i = 0, \dots, N_{\text{MS}}$ zu bestimmen. Sei der Zustandsvektor \underline{x} des unterliegenden Systemmodells aus \mathbb{R}^{L_x} , dann sind L_x Simulationen aller Zeitintevalle $[t_{i-1}, t_i]$, $i = 1, \dots, N_{\text{MS}}$ zu berechnen. Abbildung 2.8 zeigt dies

2.6. Numerische Methoden zur nichtlinearen Optimierung

anschaulich für das o.g. Problem mit 4 Stellgrößeneingriffen. Durchgezogene Linien markieren wieder Zeitbereiche, die im Rahmen der Gradientenberechnung zu simulieren sind, gestrichelte Linien Zeitbereiche, für die Ergebnisse aus der initialen Simulation wiederverwendet werden können. Die oberste dünne durchgezogene Linie steht für die initiale Simulation. Die darunter liegende dünne Liniengruppe markiert die zur Berechnung der Ableitungen nach \underline{p} nötigen Simulationen, die unterste dickere Liniengruppe die Ableitungen nach \underline{s}_i . Berechnet man auch für diesen Fall den benötigten Simulationsaufwand, so ergibt sich

$$\begin{aligned} T_{\text{MultipleShoot}} &= T_{\text{Sim}} + T_{\text{Sim}} + L_{\underline{x}} T_{\text{Sim}} \\ &= (2 + L_{\underline{x}}) \cdot T_{\text{Sim}}. \end{aligned} \quad (2.115)$$

Während der Aufwand der Gradientenberechnung im Single-Shoot-Fall eine Funktion der Anzahl der Freiheitsgrade der Optimierung ist, findet sich im Multiple-Shoot-Fall ausschließlich eine Abhängigkeit von der Anzahl der Zustandsgrößen des verwendeten dynamischen Modells. Damit ergibt insbesondere bei niederdimensionalen Modellen sogar ein Rechenzeitvorteil.

Die Leistungsfähigkeit des Multiple-Shooting-Ansatzes soll im Folgenden an einem kleinen Beispiel demonstriert werden. Gegeben sei das dynamische Modell eines Feder-Masse-Dämpfer-Systems mit nichtlinearer Federkennlinie und Anregung durch eine auf die Masse wirkende Kraft F .

$$\begin{aligned} \dot{x} &= v & \text{mit: } m &= 50 \text{ kg} \\ \dot{v} &= \frac{dv + cx^2x + F}{m} & d &= 0.5 \text{ N/(m/s)} \\ & & c &= 0.1 \text{ N/m/m}^2, \end{aligned} \quad (2.116)$$

mit dem Systemzustand $\underline{x} = (x_1, x_2)^T = (x, v)^T$ und der Stellgröße $u = F$.

Ziel der Optimierung ist, die ausgelenkte bewegte Masse ($\underline{x}_0 = (1 \text{ m}, 1 \text{ m/s})^T$) über einen Zeithorizont von 30 s mit geringem Energieaufwand in die Ruhelage ($x_1 = 0, x_2 = 0$) zu bringen. Eine maximale Auslenkung von $\pm 3 \text{ m}$ soll dabei nicht überschritten werden. Damit ergibt sich das Optimierungsproblem

$$\begin{aligned} \mathbf{U} &= \arg \min_{\mathbf{U}} \int_0^{30} (x_1(t)^2 + 0.01u(t)^2) dt \\ \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), u(t), m, d, c), & x_0 &= (1, 1)^T \\ & - 100 \leq u(t) \leq 100 \\ & - 3 \leq x_1(t) \leq 3. \end{aligned} \quad (2.117)$$

Für die numerische Optimierung wird das System mit einer Abtastzeit von einer Sekunde abgetastet und die Stellgröße als stückweise konstant mit Umschaltzeitpunkten ebenfalls zu jeder Sekunde angenommen. Die Abbildungen 2.9 und 2.10 zeigen die Verläufe der Optimierung im Single-Shoot- bzw. im Multiple-Shoot-Verfahren. Die Optimierungen

2. Mathematische Methoden

wurden mit dem Optimierer SNOPT aus dem Paket Tomlab durchgeführt. Durchgezogene Linien markieren die Simulationsverläufe. Die Beschränkung der Auslenkung der Masse ist durch gestrichelte Linien dargestellt.

Beim Single-Shoot verletzt bereits die initiale Simulation (Iteration 0) die Zustandsbeschränkungen über weite Bereiche. Der Optimierer benötigt dann über 30 Iterationen, um zunächst diese Grenzverletzungen zu beseitigen. Dabei sorgen leichte Änderungen an Stellgrößen zu Beginn des Zeithorizontes immer wieder starke Auslenkungen zu späteren Zeitpunkten. In der 100. Iteration ist nahezu das Optimum erreicht. Es werden jedoch noch weitere 83 Iterationen benötigt, um den optimalen Verlauf einzustellen.

Für die Multiple-Shoot Optimierung ist die optimale Anzahl an Schnitten gewählt. Der Zustandsverlauf ist somit in eine Sekunde lange Einzelstücke aufgeteilt. Die zusätzlich einzustellenden Zustände an diesen Schnittpunkten sind in den Abbildungen als + markiert. Schon in der initialen Simulation zeigt sich ein Vorteil des Multiple-Shootings. Ausgehend von der Simulation mit der initialen Stellgröße würden einige Schnittpunkte außerhalb der erlaubten Beschränkungen liegen. Diese werden hier bereits vor Beginn der Optimierung auf die Beschränkung zurück gesetzt. Als Folge ergeben sich nur geringe Verletzungen der Zustandsbeschränkungen, die bereits in der 4. Iteration beseitigt sind. Nach 13 Iterationen ist nahezu der optimale Verlauf gefunden und die Übergangsbedingungen zwischen den einzelnen Teilstücken der Simulation sind ebenfalls erfüllt. Bereits nach 30 Iterationen und mit weniger als der Hälfte an Rechenzeit gegenüber dem Single-Shoot-Verfahren ist der optimale Verlauf erhalten.

Den Vorteilen des Multiple-Shootings stehen jedoch auch einige Nachteile gegenüber. So ist der Implementierungsaufwand einer Multiple-Shoot-Optimierung weitaus höher, als im Single-Shoot-Fall. Zusätzlich steigt bei hochdimensionalen Modellen und vielen Schnitten die Dimension des Optimierungsproblems sehr stark an, so dass trotz Verwendung aktuellster Methoden numerische und speichertechnische Probleme bei der Verarbeitung der im Optimierer benötigten Hessematrix des Optimierungsproblems auftreten.

2.6. Numerische Methoden zur nichtlinearen Optimierung

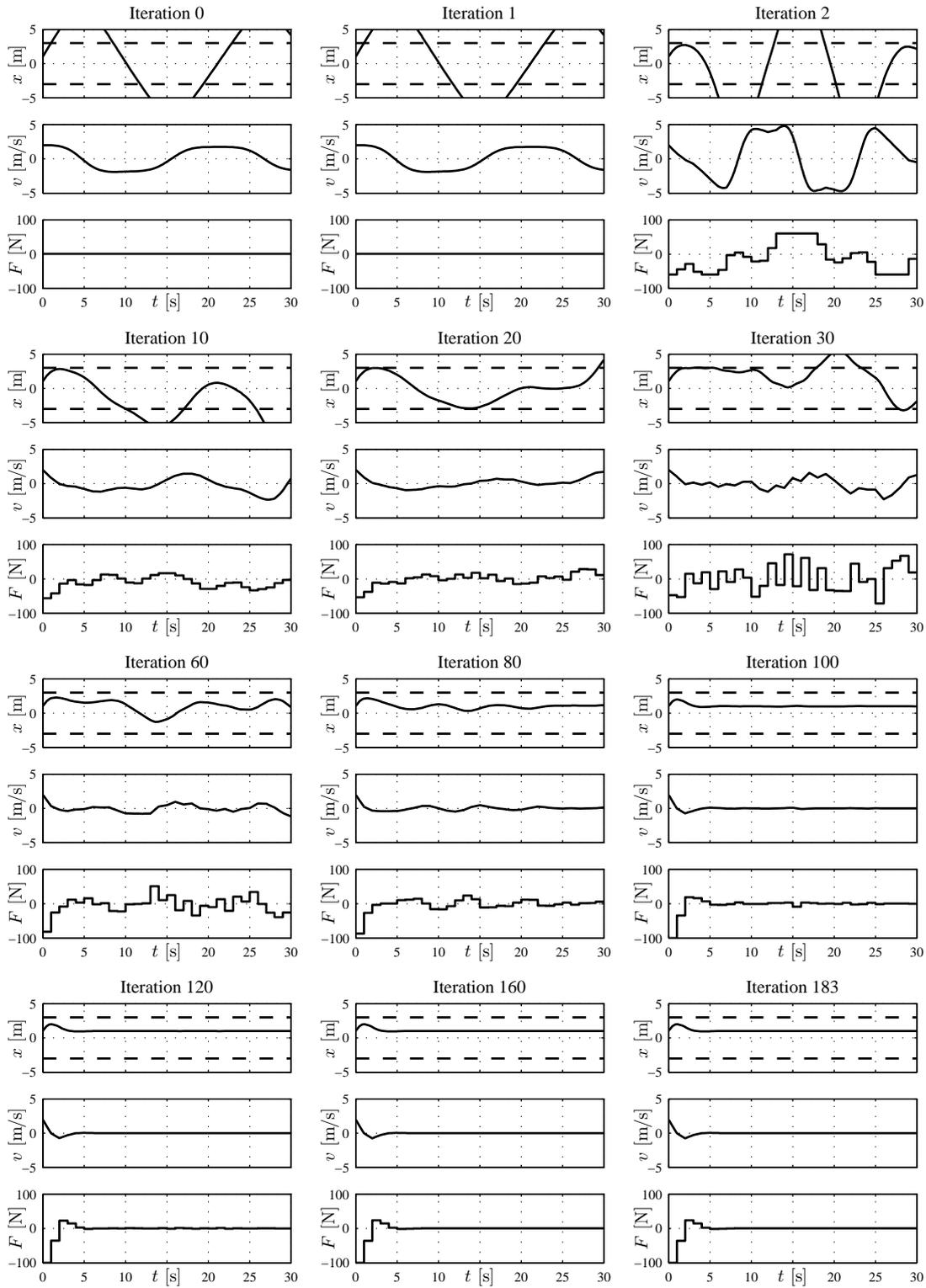


Abbildung 2.9.: Verlauf der Optimierung mit dem Single-Shoot Verfahren.

2. Mathematische Methoden

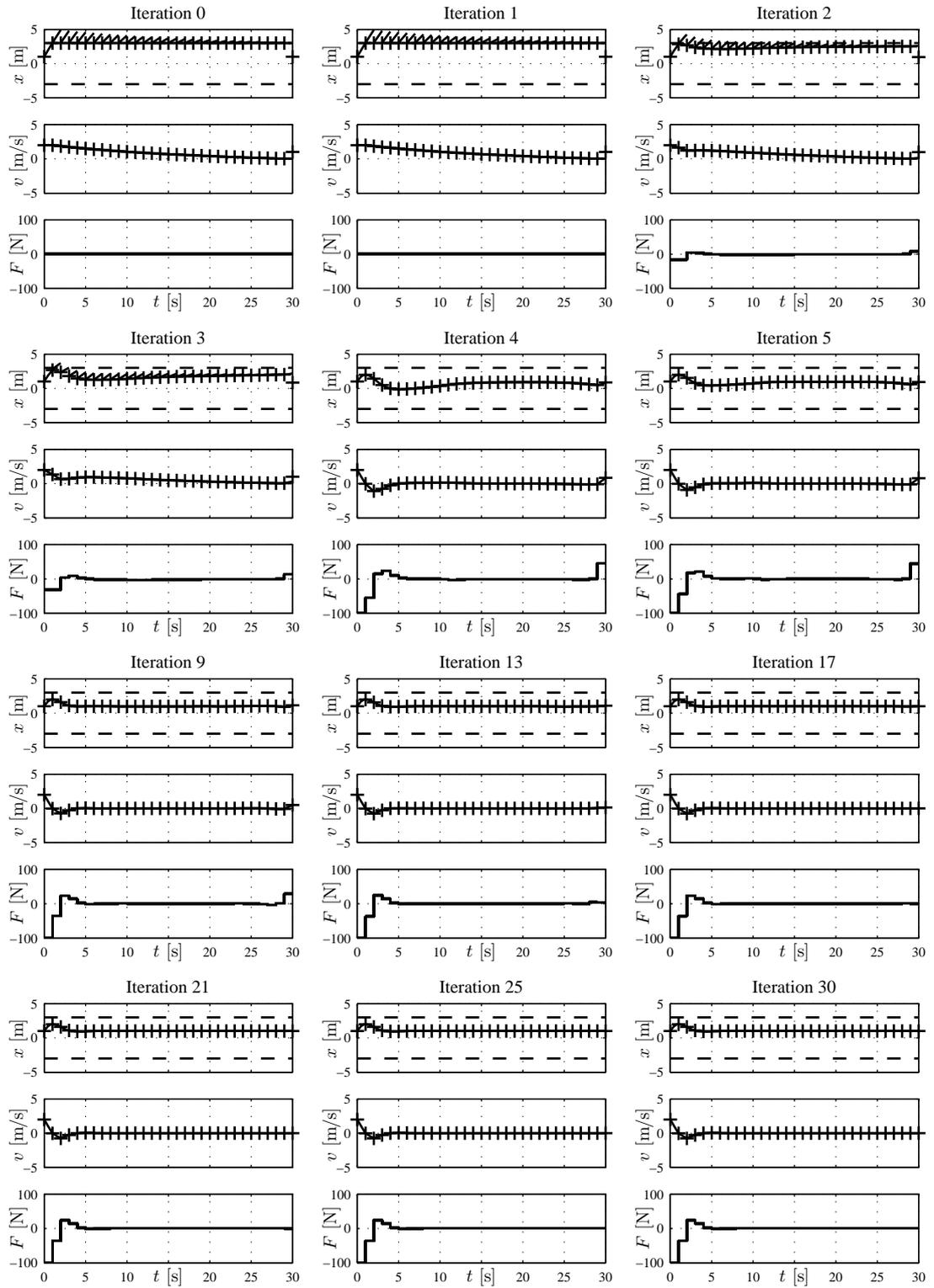


Abbildung 2.10.: Verlauf der Optimierung mit dem Multiple-Shoot Verfahren.

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

In diesem Kapitel sollen die Verfahren aus den Abschnitten 2.2 und 2.3 an einem kleinen dynamischen Beispielsystem veranschaulicht werden. Das verwendete dynamische Modell

$$\dot{m}_X = \mu_X \cdot m_X \quad (3.1a)$$

$$\dot{m}_S = -\frac{1}{Y_{X/S}} \mu_X \cdot m_X + c_{S,Feed} u(t) \quad (3.1b)$$

$$\dot{V} = u(t) \quad (3.1c)$$

gibt das Wachstum eines biologischen Organismus unter Verwendung eines einfachen unstrukturierten Modellansatzes wieder. Die Zustandsgrößen m_X und m_S beschreiben die Massen der im Reaktionsvolumen V enthaltenen Biotrockenmasse und des Substrates. Die Wachstumsrate

$$\mu_X = \mu_{X,max} \frac{c_S}{c_S + K} \quad (3.1d)$$

der Organismen ist durch eine Monod-Kinetik modelliert, wobei $\mu_{X,max}$ die maximale Wachstumsrate, K die Monodkonstante und $c_S = m_S/V$ die volumetrische Substratkonzentration markieren. Über die Ausbeute $Y_{X/S}$ ergibt sich ein Substratverbrauch für den Aufbau von Biomasse. Das Substrat ist in Lösung mit einer Konzentration $c_{S,Feed}$ und dem Volumstrom $u(t)$ zudosierbar. Als messbare Größen stehen die Konzentration der Biotrockenmasse und des Substrates zur Verfügung. Damit ergibt sich die Messgleichung

$$\begin{aligned} c_X &= \frac{m_X}{V} \\ c_S &= \frac{m_S}{V}. \end{aligned} \quad (3.1e)$$

Die als Realität zugrundeliegenden Modellparameter sind

$$\mu_{X,max} = 0,1 \text{ 1/s} \quad K = 0,2 \text{ g/l} \quad Y_{X/S} = 0,5 \text{ g/g}. \quad (3.1f)$$

Grundlage für eine Optimale-Versuchsplanung bilden zunächst eines oder mehrere initiale Experimente, die eine erste Parameteridentifikation ermöglichen. Abbildung 3.1 zeigt ein hierfür üblicherweise verwendetes sogenanntes Pulsexperiment. Dargestellt sind die

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

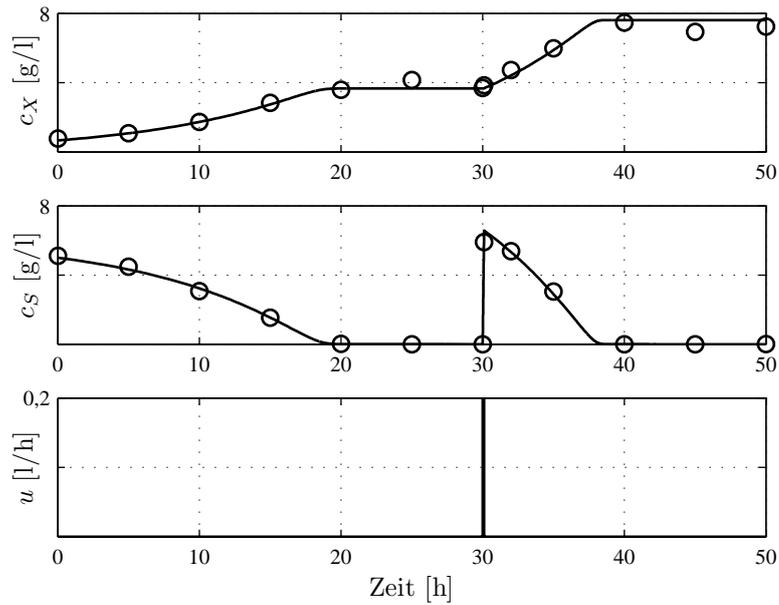


Abbildung 3.1.: Simulation (-) und eine Realisierung zur Verfügung stehender Messdaten bei Durchführung eines Pulsexperiments.

Verläufe der Messgrößen c_X und c_S und der Stellgröße $u(t)$ über der Zeit als durchgezogene Linien. Aufgrund der hohen Substratkonzentration in den ersten 15 Stunden des Versuches wachsen die Organismen mit nahezu der maximalen Wachstumsrate $\mu_{X,\max}$. Nach etwa 20 Stunden ist das Substrat verbraucht und das Wachstum der Organismen stoppt, bis bei $t = 30$ h pulsartig Substrat zudosiert wird.

Die eingezeichneten Kreise markieren die Messungen, die von einem virtuellen Experiment zur Verfügung stehen könnten. Diese wurden aus den simulierten Werten durch Addition eines normalverteilten Rauschens erzeugt. Das Messrauschen ist mit 5% relativer Standardabweichung für die Messung der Biotrockenmassekonzentration und 3% relativer Standardabweichung für die Messung der Substratkonzentration mit einer minimalen Standardabweichung von 0,02 g/l bzw. 0,01 g/l angenommen.

3.1. Parameteranalyse

Zunächst soll der verbleibende Parameterfehler nach einer Parameteridentifikation mit der Maximum-Likelihood-Methode auf Basis eines und zweier solcher Pulsexperimente berechnet werden. Als unbekannte Parameter sind in dieser Studie ausschließlich die maximale Wachstumsrate $\mu_{X,\max}$ und die Monodkonstante K angenommen. Somit gilt $\underline{\theta} = (\mu_{X,\max}, K)^T$. Der Einfachheit halber werden als Modellparameter immer die wahren oben genannten Werte $\hat{\underline{\theta}} = \underline{\theta}^*$ verwendet.

Tabelle 3.1.: Ergebnisse der Berechnung des verbleibenden Parameterfehlers für das dynamische Beispielmodell.

	$\mathbf{C}_{\underline{\theta}}(1)$	$\mathbf{C}_{\underline{\theta}}(2)$
Bootstrap	$\begin{pmatrix} 7,5226 \cdot 10^{-6} & 2,5669 \cdot 10^{-4} \\ 2,5669 \cdot 10^{-4} & 9,2917 \cdot 10^{-3} \end{pmatrix}$	$\begin{pmatrix} 3,7613 \cdot 10^{-6} & 1,2835 \cdot 10^{-4} \\ 1,2835 \cdot 10^{-4} & 4,6459 \cdot 10^{-3} \end{pmatrix}$
FIM	$\begin{pmatrix} 8,3097 \cdot 10^{-6} & 2,8213 \cdot 10^{-4} \\ 2,8213 \cdot 10^{-4} & 1,0122 \cdot 10^{-2} \end{pmatrix}$	$\begin{pmatrix} 4,1549 \cdot 10^{-6} & 1,4106 \cdot 10^{-4} \\ 1,4106 \cdot 10^{-4} & 5,0608 \cdot 10^{-3} \end{pmatrix}$
MAP		$\begin{pmatrix} 3,9136 \cdot 10^{-6} & 1,3351 \cdot 10^{-4} \\ 1,3351 \cdot 10^{-4} & 4,8237 \cdot 10^{-3} \end{pmatrix}$

Im Rahmen einer Bootstrap-Analyse (Monte-Carlo-Identifikation) werden ausgehend von der Simulation des identifizierten Modells durch Addition von zufälligem Messrauschen virtuelle verrauschte Messwerte erschaffen und mit jedem Satz von Messwerten eine Parameteridentifikation durchgeführt. Die in Abbildung 3.1 gezeigten Kreise stellen somit eine im Rahmen dieser Monte-Carlo-Analyse verwendete Realisierung dar. Die aus einer Bootstrap Analyse mit 1000 Identifikationen nach Gleichung (2.7) berechnete Kovarianzmatrix $\mathbf{C}_{\underline{\theta}} = E \{(\underline{\theta} - \underline{\theta}^*)(\underline{\theta} - \underline{\theta}^*)^T\}$ des zu erwartenden Parameterfehlers auf Basis eines und zweier Pulsexperimente ist in Tabelle 3.1 angegeben. Aufgrund der hohen Anzahl benutzter Samples ist der zu erwartende Approximationsfehler sehr gering. Somit soll im Folgenden die Bootstrap-Analyse jeweils als Referenz dienen.

Eine Approximation des verbleibenden Parameterfehlers durch die inverse Fisher'sche-Informationsmatrix(FIM) (Gleichung 2.64) ist ebenfalls in Tabelle 3.1 gegeben. Die Fisher'sche-Informationsmatrix liefert etwas höhere Werte in der Kovarianzmatrix $\mathbf{C}_{\underline{\theta}}$, was (2.64) nicht gerecht wird. Die Ursache hierfür ist in numerischen Ungenauigkeiten der Bootstrap-Analyse zu suchen. Die Ergebnisse liegen jedoch sehr nahe beieinander.

Eine Parameteranalyse mit Hilfe des Posteriors aus der A-Posteriori-Analyse (MAP) nach (2.79) ist für das erste Experiment nicht möglich, da hier eine initiale Parameterkovarianz benötigt wird. Verwendet man das Ergebnis der Bootstrap-Analyse des ersten Pulsexperiments als Prior $\mathbf{C}_{\underline{\theta}}(1)$, so ergibt sich bei zwei verwendeten Experimenten der in der letzten Zeile der Tabelle 3.1 angegebene Parameterfehler nach zwei Experimenten. Die durch die A-Posteriori-Analyse berechneten Werte liegen etwas näher am Ergebnis der Bootstrap-Analyse als die Fisher'sche-Informationsmatrix. Die Abweichungen sind jedoch auch hier sehr gering.

Abbildung 3.2 zeigt die oben angegebenen Zahlenwerte nochmals als Histogramm der Ergebnisse der Monte-Carlo-Simulation und überlagert die sich aus der Fisher'schen-Informationsmatrix (rot) und dem Posterior (blau) ergebenden Verteilungsfunktionen der

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

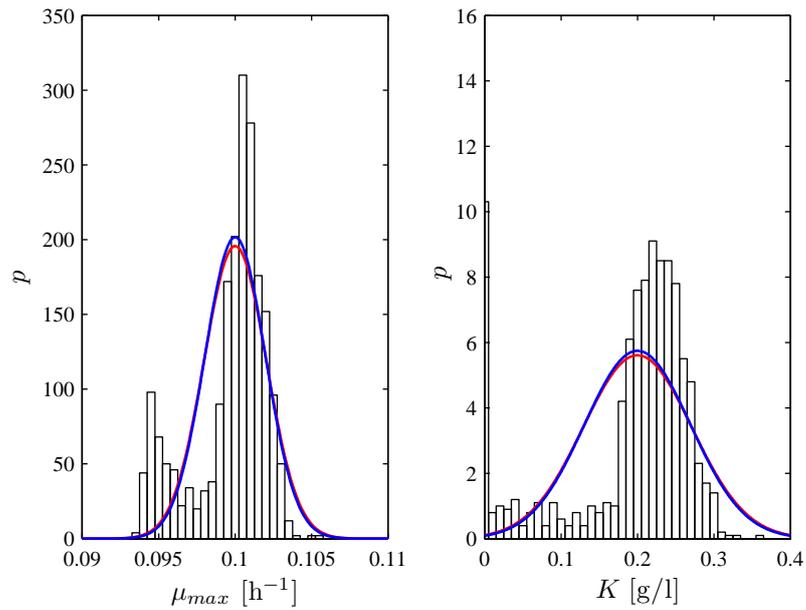


Abbildung 3.2.: Vergleich der Ergebnisse der Parameteranalyse mit der Bootstrap-Methode (schwarz) der Fisher'schen-Informationsmatrix (rot) und der Berechnung der Posteriorverteilung mit Hilfe der Unscented-Transformation (blau).

einzelnen Parameter. Beide Approximationsverfahren zeigen einen Approximationsfehler, da die wahre Verteilung des Parameterfehlers nicht gauß'sch ist. Die Ergebnisse sind jedoch nahezu identisch.

3.2. Optimale-Versuchsplanung

In diesem Abschnitt sollen der klassische Fisher-Ansatz und das Maximum-a-Posteriori-Verfahren am oben eingeführten Beispiel miteinander verglichen werden. Dazu wird mit beiden Verfahren jeweils ein optimales Experiment geplant. Der Einfachheit halber verwenden beide Planungsverfahren das Modell mit dem wahren Parametersatz. Als initiale Parameterunsicherheit verwenden beide Planungsverfahren das Ergebnis der Bootstrap-Analyse. Aus den verschiedenen Kriterien ist hier das E-Kriterium (2.71a) ausgewählt worden.

Ausgehend von einem initialen Reaktionsvolumen von 3 l und 2 g Biotrockenmasse, sollen beide Verfahren eine optimale initiale Substratmenge sowie ein optimales Zudosierprofil

liefern. Das zu lösende Optimierungsproblem liest sich wie folgt:

$$\begin{aligned}
 \{\underline{x}_0^*, \mathbf{U}^*\} &= \arg \min_{\underline{x}_0, \mathbf{U}} (\Phi_{\text{OVP}}(\underline{x}_0, \mathbf{U}, \theta)) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \theta), \quad \underline{x}(t=0) = \underline{x}_0 \\
 t_0 &\leq t \leq t_{\text{Ende}} \\
 \underline{x}_{0,\min} &\leq \underline{x}_0 \leq \underline{x}_{0,\max} \\
 \underline{u}_{\min} &\leq \underline{u}(t) \leq \underline{u}_{\max} \\
 \underline{r}(\underline{x}(t)) &\leq \underline{0}, \quad \underline{r} \in \mathbb{R}^{L_r}.
 \end{aligned} \tag{3.2}$$

Um das Ergebnis nicht zu verfälschen, werden keine Beschränkungen angenommen. Das Ergebnis der Optimierung soll ausschließlich vom Gütefunktional und nicht von zusätzlichen Beschränkungsfunktionen beeinflusst sein. Die Konzentration des Substrates im Feed beträgt 100 g/l. Messungen finden zu den gleichen Zeitpunkten statt, wie beim Puls-experiment. Die Lösung des Optimierungsproblems erfolgt mit einem SQP-Verfahren (SNOPT).

Abbildung 3.3 zeigt das Ergebnis der Planung mit der Fisher'schen-Informationsmatrix in rot und mit dem Maximum-a-Posteriori-Ansatz in blau. Beide Fermentationsverläufe zeigen nur minimale Abweichungen voneinander. Beide Trajektorien starten zunächst

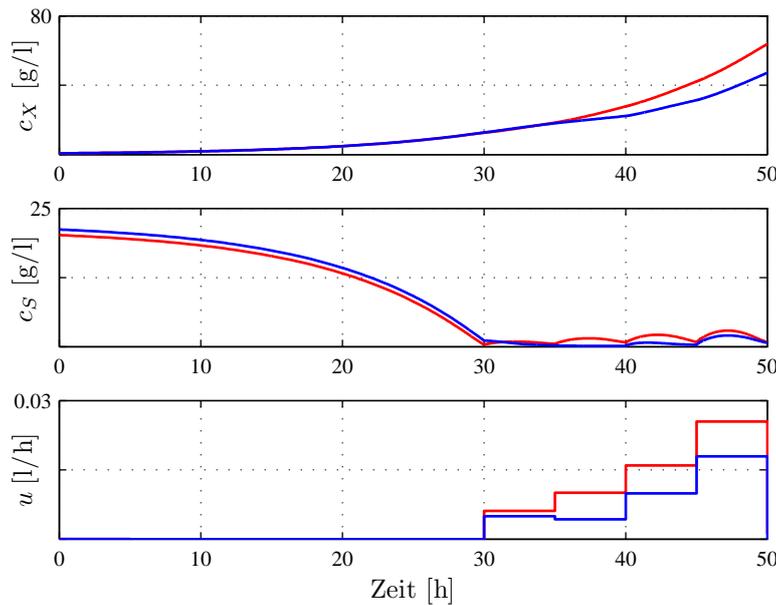


Abbildung 3.3.: Ergebnisse der Optimalen-Versuchsplanung mit dem Fisher-Ansatz (rot) und dem Maximum-a-Posteriori-Ansatz (blau) bei Verwendung der Unscented-Transformation zur Berechnung der benötigten statistischen Momente.

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

mit einer Batchphase. In dieser Zeit ist eine genaue Identifikation der maximalen Wachstumsrate μ_{\max} möglich. Anschließend ist das Fütterprofil so eingestellt, dass sich geringe Substratkonzentrationen einstellen und folglich eine genaue Identifikation des Monod-Parameters K möglich ist.

Um genauere Informationen über den Erfolg der optimalen Versuchsplanungsverfahren zu erhalten, ist es sinnvoll, die Form des Gütefunktional der Parameteridentifikation,

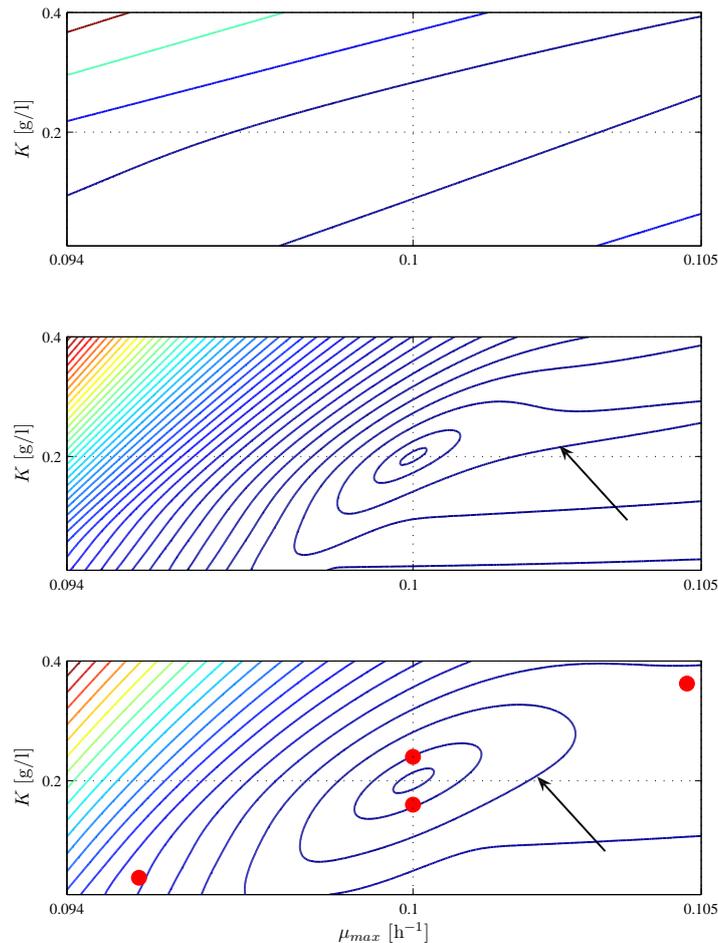


Abbildung 3.4.: Konditionierung des Gütefunktional der Parameteridentifikation nach eine Pulsexperiment (oben), einem Pulsexperiment und einem optimalen Experiment (Mitte und unten) bei Verwendung der Fisher'schen-Informationsmatrix (Mitte) und des Maximum-a-Posteriori-Ansatzes mit unterlagelter Unscented-Transformation (unten). Gleiche Farben markieren gleiche Höhenlinien in allen Darstellungen.

also der Likelihood, näher zu betrachten. Dies ist in Abbildung 3.4 als Höhenlinien-darstellung wiedergegeben. Zunächst ist zu erkennen, dass das Minimum des Gütefunktional der Parameteridentifikation bei Anwendung der Verfahren zur optimalen Versuchsplanung (Mitte und unten) weitaus besser ausgeprägt ist, als bei Durchführung eines Pulsexperiments (oben).

Das Minimum des Gütefunktional ist bei dem mit der Fisher'schen-Informationsmatrix geplanten Experiment am stärksten ausgeprägt. Die Form der umliegenden Höhenlinien zeigt jedoch etwas eher die Neigung der Ausprägung eines langgestreckten Tals (siehe Pfeilmarkierung) als es bei der Planung mit dem A-Posteriori-Ansatz der Fall ist. Der A-Posteriori-Ansatz nutzt bei Verwendung der Unscented-Transformation Simulationen des Modells an den Punkten, die in der Kontourdarstellung als Punkte markiert sind und approximiert die Form des Gütefunktional durch Interpolation, während die Fisher'sche-Informationsmatrix-Methode nur eine Simulation und die Sensitivität der Simulation bezüglich der Parameter an diesem Punkt verwendet. Dies legt zunächst die Vermutung nahe, dass durch die Verwendung der Unscented-Transformation zwar ein nominell etwas schlechteres Ergebnis resultiert, dieses dafür aber etwas robuster gegen Parameterfehler sein könnte. Dies konnte mit verschiedenen Monte-Carlo-Simulationen jedoch nicht untermauert werden. In Abbildung 3.5 ist das Resultat einer Bootstrap-

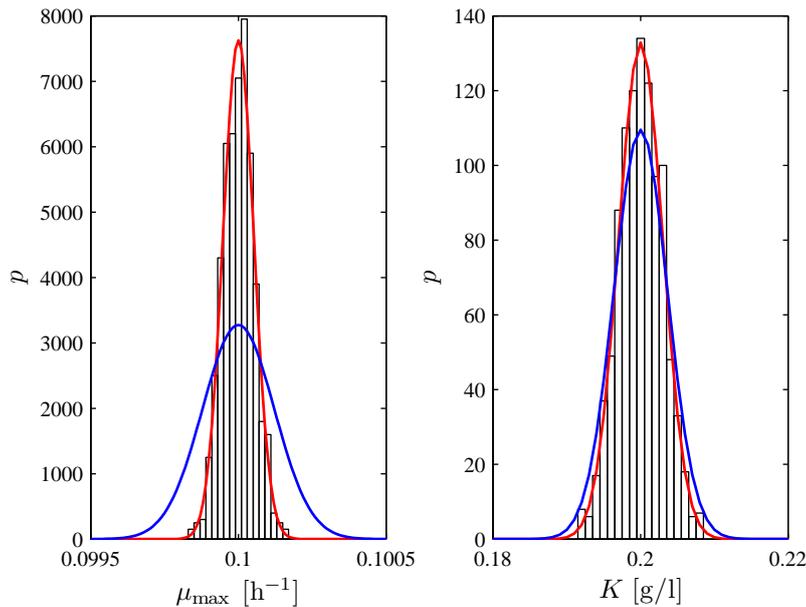


Abbildung 3.5.: Vergleich der Ergebnisse der Parameteranalyse bei Verwendung eines Pulsexperiments und dem mit dem A-Posteriori-Ansatz geplanten optimalen Versuch mit der Bootstrap-Methode (schwarz) der Fisher'schen-Informationsmatrix (rot) und der Berechnung der Posteriorverteilung mit Hilfe der Unscented-Transformation (blau).

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

Analyse bei Verwendung des Pulsexperiments und des mit dem A-Posteriori-Ansatz geplanten Experiments als Histogramm (schwarz) dargestellt. Im Vergleich dazu sind in rot die Approximation des Parameterfehlers mit der Fisher'sche-Informationsmatrix und in blau die Approximation mit dem A-Posteriori-Ansatz angegeben. Hier erkennt man die Ursache für die unterschiedlichen Ergebnisse. Die Unscented-Transformation liefert hier eine signifikante Überschätzung des verbleibenden Fehlers.

Dies steht scheinbar im Gegensatz zu der guten Approximation der Kovarianzmatrizen durch den Ansatz zweiter Ordnung. Ein elementares Element der A-Posteriori-Analyse ist jedoch die Kreuzkovarianzmatrix $C_{\theta y}$. Auch bei Verwendung der Unscented-Transformation ergibt sich zur Berechnung diese Matrix nur die gleiche Approximationsformel (Gleichung (2.43)), die auch bei Verwendung eines Ansatzes erster Ordnung resultiert [JU04]. Während für die Berechnung der Kovarianzmatrizen innerhalb der Unscented-Transformation eine Schrittweite von $h = \sqrt{3}$ optimal ist, ist für Ansätze erster Ordnung eine minimale Schrittweite effektiv.

In Abbildung 3.6 ist in blau die Approximation des resultierenden Parameterfehlers mit der Unscented-Transformation bei Verwendung von $h = 0.001$ gezeigt. Diese Approximation ist weitaus genauer als bei Verwendung des eigentlich als optimal eingeführten

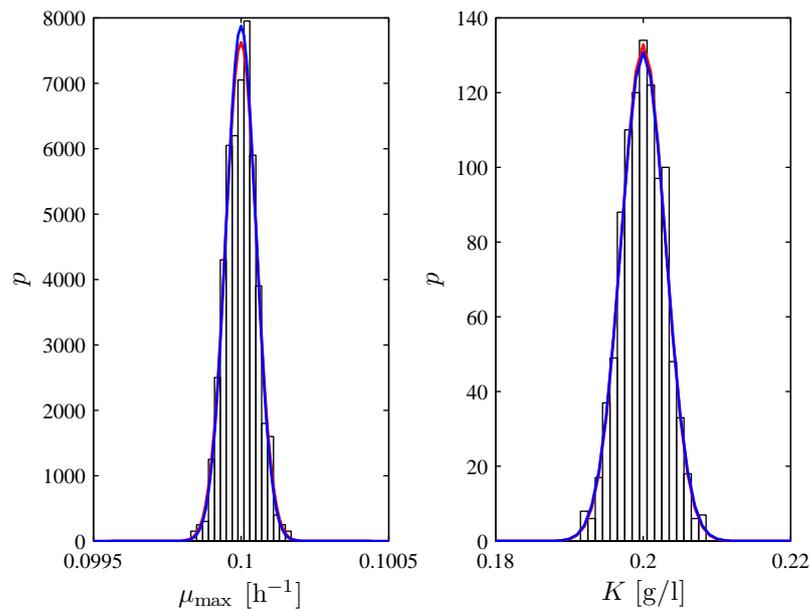


Abbildung 3.6.: Vergleich der Ergebnisse der Parameteranalyse bei Verwendung eines Pulsexperiments und dem mit dem A-Posteriori-Ansatz geplanten optimalen Versuch mit der Bootstrap-Methode (schwarz) der Fisher'schen-Informationsmatrix (rot) und der Berechnung der Posteriorverteilung mit Hilfe der Unscented-Transformation bei Verwendung von $h = 0,001$ (blau).

3.2. Optimale-Versuchsplanung

$h = \sqrt{3}$, da hier die benötigte Kreuzkovarianzmatrix weitaus besser approximiert wird. Verwendet man diese geringere Schrittweite zur Planung eines optimalen Experiments, ergibt sich nahezu der gleiche Versuch, wie bei Verwendung der Fisher'schen-Informationsmatrix.

Eine genauere Untersuchung dieses Phänomens würde den Rahmen dieser Arbeit, die hauptsächlich auf robuste Prozessführung fokussiert, sprengen. Es ist gezeigt, dass mit Hilfe der Unscented-Transformation eine Optimale-Versuchsplanung ohne Verwendung aufwändiger Ableitungen möglich ist. Die Probleme der Überschätzung des verbleibenden Parameterfehlers sind wahrscheinlich auf die in der Unscented-Transformation verwendete Schrittweite zurückzuführen. Während für die Berechnung von Kovarianzmatrizen $h = \sqrt{3}$ optimal ist, ist zur Berechnung von Kreuzkovarianzen eine kleinere Schrittweite zu verwenden.

Dies genauer zu untersuchen, obliegt folgenden Arbeiten. Es ist jedoch dringend nötig diese Problematik zu bearbeiten, da sich hier auch signifikante Auswirkungen auf die Unscented-Kalman-Filter ergeben, deren Measurement-Update ebenfalls auf Kreuzkovarianzen basiert.

3. Anwendung der Maximum-a-Posteriori-Optimalen-Versuchsplanung

4. Anwendung robuster Prozessführungsverfahren

In diesem Kapitel soll die Anwendung der beschriebenen Verfahren zur robusten Prozessführung dargestellt werden. Zunächst werden die klassischen und robusten Methoden anhand eines biotechnologischen und eines chemischen Prozessmodells in umfangreichen Simulationsstudien miteinander verglichen. Zusätzlich werden Ergebnisse realer Versuche eines biotechnologischen Produktionsprozesses vorgestellt.

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

In diesem Abschnitt wird die Anwendung der robusten Verfahren an einem einfachen Prozessmodell eines biotechnologischen Prozesses veranschaulicht. Als Beispielsystem dient hier ein Sekundärmetabolitbildner. Diese zeichnen sich dadurch aus, dass sie bei hohen Nährstoffkonzentrationen hauptsächlich wachsen und wenig bis gar nichts produzieren. Die Produktausschüttung wird durch geringe Nährstoffkonzentrationen ange-regt.

Ein einfaches unstrukturiertes Modell, welches das Wachstums- und Produktionsverhalten eines Sekundärmetabolitbildners wiedergibt, basiert auf dem Pseudoreaktionsschema



wobei X für Biotrockenmasse, S für Substrat und P für Produkt steht. Es finden zwei jeweils durch die Biomasse katalysierte Reaktionen statt. Zum einen bildet sich autokatalytisch neue Biomasse, die Organismen vermehren sich, wenn genügend Nährmedien (S) vorhanden sind. Ist nur wenig Substrat verfügbar, wird dieses nicht in weitere Biomasse, sondern in das Produkt umgesetzt. Die Organismen betreiben ihren Sekundärmetabolismus. Der Erhaltungsstoffwechsel der Organismen ist in diesem Schema nicht dargestellt.

4. Anwendung robuster Prozessführungsverfahren

Die mathematische Modellierung dieses Reaktionsschemas in einem unstrukturierten Modell ist

$$\dot{m}_X = \mu_X \cdot m_X \quad (4.3a)$$

$$\dot{m}_P = \mu_P \cdot m_X \quad (4.3b)$$

$$\dot{m}_S = -\frac{1}{Y_{X/S}}\mu_X \cdot m_X - \frac{1}{Y_{P/S}}\mu_P \cdot m_X - \mu_E \cdot m_X + c_{S,\text{Feed}}u_S(t) \quad (4.3c)$$

$$\dot{V} = u_S(t). \quad (4.3d)$$

Die erste Gleichung (4.3a) beschreibt das Wachstum der Organismen. Die Organismen wachsen, wenn viel Substrat zur Verfügung steht. In vielen Fällen handelt es sich bei Sekundärmetaboliten um Antibiotika. Auch wenn die Organismen diese selbst produzieren, können diese auch das Wachstum der Organismen hemmen. Die spezifische Wachstumsrate der Organismen μ_X setzt sich in diesem Beispiel somit aus einer Monod-Kinetik bezüglich der Nährstoffkonzentration $c_S = m_S/V$ und einer Inhibierungskinetik bezüglich der Produktkonzentration $c_P = m_P/V$ zusammen

$$\mu_X = \frac{c_S}{c_S + K_{XS}} \frac{K_{XP}}{c_P + K_{XP}}. \quad (4.3e)$$

Die Gleichung (4.3b) gibt die Produktbildung wieder. Die Produktbildung erfolgt, wenn wenig Substrat zur Verfügung steht, wird aber wieder unterbrochen, wenn sehr wenig oder gar kein Substrat mehr vorhanden ist, da das Substrat auch als Edukt für die Produktbildung benötigt wird. Dieses Verhalten ist hier durch eine Haldane-Kinetik bezüglich der Substratkonzentration

$$\mu_P = \frac{c_S}{c_S + K_{PS_1} + c_S^2 K_{PS_2}} \quad (4.3f)$$

wiedergegeben.

In der dritten Zeile (4.3c) ist die Massenbilanz des Substrates beschrieben. Dieses wird für Wachstum μ_X mit der Ausbeute $Y_{X/S}$, Produktbildung μ_P mit der Ausbeute $Y_{X/P}$ und Erhaltungsstoffwechsel der Organismen μ_E verbraucht und kann zusätzlich mit der Konzentration $c_{S,\text{Feed}}$ und dem Volumenstrom $u_S(t)$ zudosiert werden. Der Erhaltungsstoffwechsel ist nicht möglich, wenn kein Substrat vorhanden ist, dazu ist in der Implementierung eine Schaltkinetik hinzugefügt.

Ein Absterben der Organismen ist nicht modelliert. Die Beschreibung des Füllvolumens des Fermenters in der vierten Gleichung (4.3d) folgt aus der Gesamtmassenbilanz unter der Annahme einer konstanten Dichte der Brühe.

Der Systemzustand des Modells setzt sich als

$$\underline{x} = (m_X, m_P, m_S, V)^T \quad (4.3g)$$

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

zusammen. Als Messgröße stehen die Konzentrationen der Biotrockenmasse, des Produktes und des Substrates zur Verfügung. Es ergibt sich die Messgleichung

$$\underline{y} = (c_X, c_P, c_S)^T = (m_X/V, m_P/V, m_S/V)^T . \quad (4.3h)$$

Einzigste Stellgröße ist die dosierte Menge an Substrat $u(t) = u_S(t)$. Die Zahlenwerte sämtlicher auftretenden Modellparameter $\underline{\theta}$ können der Tabelle 4.1 entnommen werden.

Tabelle 4.1.: Parameter des unstrukturierten Modells

Paramtere	Wert	Einheit
$\mu_{X,\max}$	0,2	1/h
$\mu_{P,\max}$	0,01	1/h
K_{XS}	6	g/l
K_{XP}	0,1	g/l
K_{PS_1}	1	g/l
K_{PS_2}	1	l/g
$Y_{X/S}$	5	g/g
$Y_{P/S}$	1	g/g
μ_E	0,1	1/h

4.1.1. Trajektorienplanung

Biotechnologische Produktionsprozesse werden meist im Batch oder Fed-Batch betrieben. Zunächst soll hier eine optimale Steuertrajektorie für eine maximale Produktbildung berechnet werden. Ziel der Planung ist eine Steuertrajektorie für die Substratzufuhr $u(t)$ zu finden, so dass nach einer festen Batchlaufzeit von $t_{\text{Ende}} = 100$ h eine maximale Menge Produkt m_P produziert ist. Dazu kann das Substrat mit der Konzentration $c_{S,\text{Feed}} = 400$ g/l zudosiert werden. Dabei darf eine maximale Biomassekonzentration von $c_{X,\max} = 10$ g/l nicht überschritten werden, da sonst keine ausreichende Sauerstoffversorgung der Organismen möglich ist. Zusätzlich sind sehr hohe Substratkonzentrationen toxisch. Da dies im Modell nicht wiedergegeben ist, darf im Rahmen der Trajektorienplanung eine maximale Substratkonzentration von $c_{S,\max} = 10$ g/l nicht überschritten werden. Schließlich darf das maximale Reaktionsvolumen $V_{\max} = 101$ des Fermenters nicht überschritten werden. Initial befindet sich im Fermenter 8 l Brühe mit 0,5 g Biotrockenmasse und 0 g Produkt. Somit ergibt sich für die Trajektorienplanung das zu lösende

4. Anwendung robuster Prozessführungsverfahren

Optimierungsproblem

$$\begin{aligned}
 \{m_{S,0}, \mathbf{U}\}^* &= \arg \min_{m_{S,0}, \mathbf{U}} (\Phi_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})) \\
 \Phi_{\text{TP}} &= -m_{\text{P}}(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), u_{\text{S}}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u_{\text{S}}(t) \leq 0,051/\text{h} \\
 \underline{r}(\underline{x}(t)) &\leq \underline{0} \\
 \underline{r} &= (c_{\text{X}}(t) - c_{\text{X,max}}, c_{\text{S}}(t) - c_{\text{S,max}}, V - V_{\text{max}})^{\text{T}} .
 \end{aligned} \tag{4.4}$$

Da Antibiotika sehr hochpreisige Produkte sind, wird im Gütefunktional ausschließlich die produzierte Produktmenge am Prozessende betrachtet und Kosten für z.B. das Substrat ignoriert. Als Freiheitsgrad für die Optimierung dienen die initiale Masse an Substrat und der zudosierte Volumenstrom des Substrates. Dieser wird für die numerische Optimierung in stückweise konstante Abschnitte von jeweils 5 h unterteilt. Die Beschränkungsfunktion \underline{r} wird ebenfalls mit einer Abtastung von 5 h diskretisiert. Die Lösung des Optimierungsproblems (4.4) erfolgte mit Hilfe des Multiple-Shooting-Verfahrens und des nichtlinearen Optimierers SNOPT aus dem Softwarepaket Tomlab.

Die resultierende Füttertrajektorie und die Verläufe der Biomassekonzentration, der Substratkonzentration und der Produktmenge sind in Abbildung 4.1 dargestellt. Der Prozessverlauf ist in zwei Phasen unterteilt. Zunächst wird Substrat so zudosiert, dass sich die maximal zulässige Substratkonzentration und somit eine hohe Wachstumsrate der Organismen einstellt. In dieser ersten Phase findet fast ausschließlich Reproduktion statt. Etwa zur 42. Stunde wird die Substratdosierung signifikant verringert. In der folgenden zweiten Prozessphase wird über die gedrosselte Dosierung eine Substratkonzentration in der Brühe eingestellt, die genau dem lokalen Maximum der für die Beschreibung der Produktbildung verwendeten Haldane-Kinetik entspricht. In dieser Phase findet fast ausschließlich Produktion und nur minimale Reproduktion statt. Als Resultat ergibt sich eine erwartete Produktmenge von 32,4 g.

Wie bereits beschrieben, stellt das mathematische Modell nur eine Näherung der Realität, bei biotechnologischen Problemen sogar nur eine grobe Näherung, dar. Daher wird im Folgenden der Einfluss von Modellfehlern auf den Produktionsprozess untersucht. Dazu sei angenommen, dass die Modellstruktur korrekt, die Modellparameter jedoch fehlerbehaftet sind. Ersteres stellt eine vereinfachende Annahme dar. Auswirkungen einer fehlerhaften Modellstruktur werden im Rahmen von Untersuchungen mit verschiedenen Modellansätzen untersucht, würden jedoch den Rahmen dieser Arbeit überschreiten. Die Parameter des mathematischen Modells werden üblicherweise aus verrauschten Messdaten gewonnen. Als Folge der verrauschten Messdaten in endlicher Anzahl, sind die identifizierten Modellparameter auch zufällige Größen.

Zur Untersuchung der Auswirkungen von Parameterfehlern wird im Folgenden angenommen, dass die benannten Modellparameter jeweils einen normalverteilten Fehler mit 5%

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

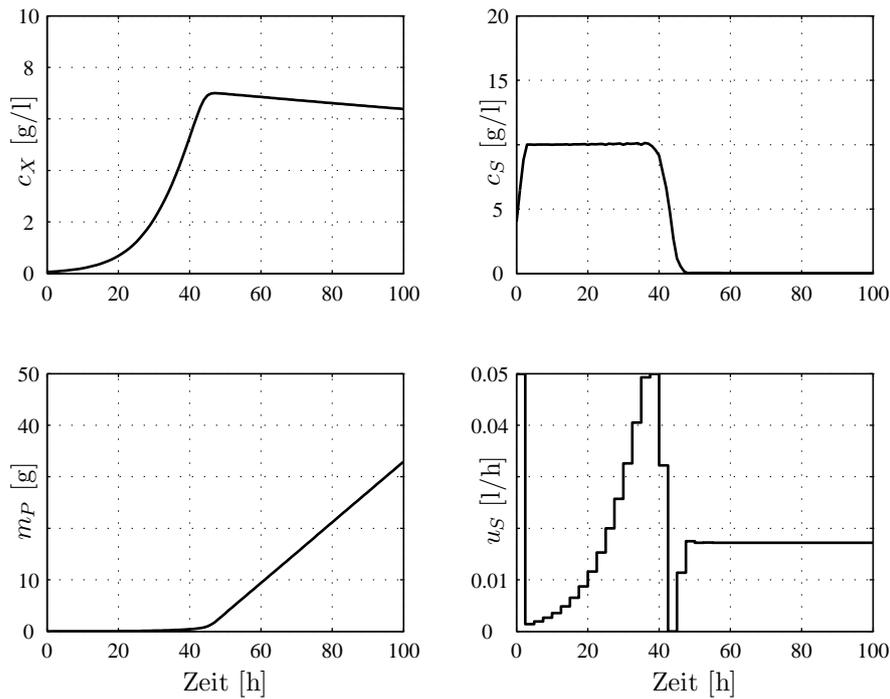


Abbildung 4.1.: Ergebnis der Trajektorienplanung für das unstrukturierte Modell (X-Biotrockenmasse, S-Substrat, P-Produkt).

Standardabweichung beinhalten. Im Rahmen einer Monte-Carlo-Simulationsstudie sollen die Auswirkungen eines solchen Parameterfehlers untersucht werden. Dazu wird aus der Verteilung der Parameter für jeden Simulationslauf ein Sample gezogen, welches jeweils eine mögliche - vom Modell abweichende - Realität darstellt.

Die Auswirkungen auf die produzierte Menge an Produkt aus 10000 simulierten Prozessverläufen sind im Histogramm in Abbildung 4.2 wiedergegeben. Es ist zu erkennen, dass bei dem angenommenen Modellfehler nur eine geringe Wahrscheinlichkeit besteht, die vorausgesagte Produktmenge von 32,4 g (gestrichelte Linie) zu erreichen. Im Mittel ist nach dieser Studie nur mit einer produzierten Menge von 24,8 g zu rechnen. Zusätzlich resultieren einige simulierte Prozessverläufe in einer sehr geringen Produktivität. Gerade in biotechnologischen Prozessen mit langer Laufzeit stellt nur ein mit einer sehr geringen Produktmenge durchgeführter Produktionsprozess einen massiven Verlust dar.

4.1.2. Robuste Trajektorienplanung

Um auch bei vorhandenen Modellfehlern einen Produktionsprozess so zu planen, dass eine vorhergesagte Produktionsmenge mit hoher Wahrscheinlichkeit erreicht wird, können

4. Anwendung robuster Prozessführungsverfahren

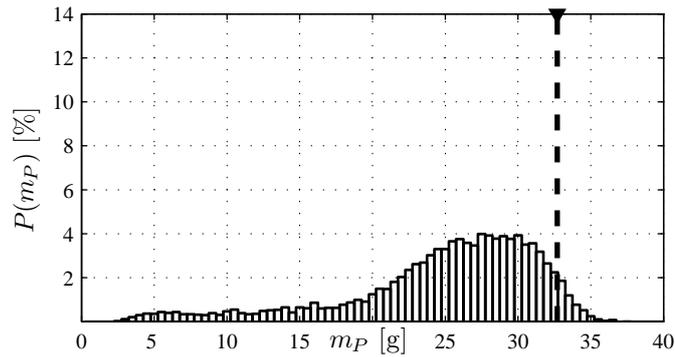


Abbildung 4.2.: Histogramm der Produktivität des geplanten Prozesses bei Annahme von Modellunsicherheiten.

die zu erwartenden Modellfehler im Rahmen einer robusten Trajektorienplanung beachtet werden. Zunächst ist dazu zu prüfen, inwiefern die in Abschnitt 2.1 vorgestellten Verfahren geeignet sind, die Auswirkungen von Modellfehlern auf die Prozess trajektorie zu berechnen. Als Referenz dient die im letzten Abschnitt durchgeführte Monte-Carlo-Simulation mit 10000 Samples. Die aus dieser Monte-Carlo-Simulation resultierenden ersten zwei statistischen Momente sind in Abbildung 4.3 in schwarz (durchgezogen - Mittelwert, gestrichelt - 3σ -Grenze) dargestellt.

Approximiert man diese zwei statistischen Momente mit dem Ansatz erster Ordnung, wie in Abschnitt 2.1.2 beschrieben, ergeben sich die in rot dargestellten Werte. Da der mittlere Prozessverlauf von der Simulation des Modells mit den nominellen Parametern abweicht, ergibt sich hier schon ein signifikanter Fehler. Zusätzlich ist die Standardabweichung des Produktverlaufes auch massiv unterschätzt. In blau sind die mit Hilfe der Unscented-Transformation (Abschnitt 2.1.3) berechneten statistischen Momente des Prozessverlaufes dargestellt. Auch hier zeigen sich Mängel in der Approximation. Die stark asymmetrische Verteilung der Produktmenge resultiert in einer Überschätzung der Varianz und einer zu gering approximierten mittleren Produktionsmenge.

Im Rahmen eines Vorabscreenings wurde daher überprüft, inwiefern die beschriebenen Approximationsverfahren in der Lage sind, Änderungen der Produktivität und dessen Varianz als Folge von veränderten Stellgrößen wiedergeben zu können. Bei Verwendung einer Approximation 1. Ordnung zur Berechnung der Varianz, zeigte sich, dass die Varianz der Produktmenge nicht nur stark unterschätzt wird, sondern sich Auswirkungen veränderter Fütterstrategien zwar auf die berechnete Produktionsmenge, aber kaum auf die approximierte Verteilung niederschlagen.

Bei Verwendung der Approximation 2. Ordnung ist der Einfluss auf beide Größen ausgeprägt. Ergänzende Monte-Carlo-Simulationen zeigen auch, dass die von der Approximation 2. Ordnung errechneten Auswirkungen variierter Fütterstrategien auch den real

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

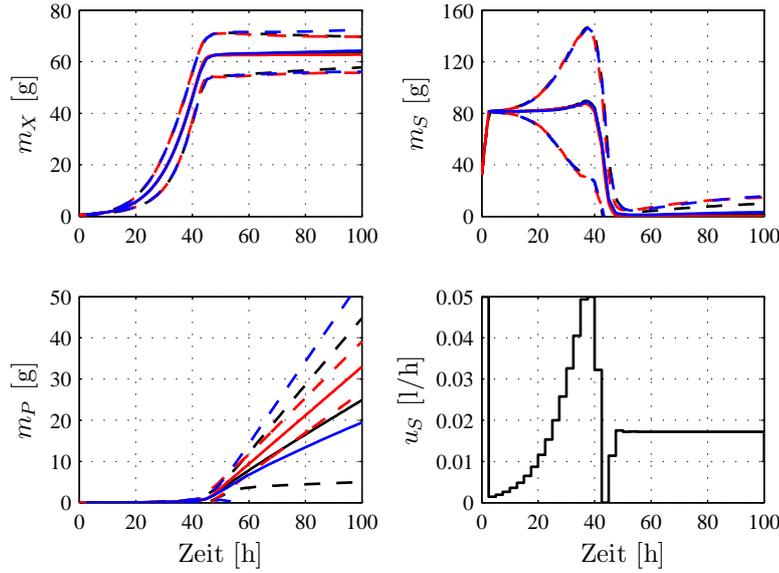


Abbildung 4.3.: Vergleich der Approximation von Mittelwert und 3σ -Grenze der Zustände mit der Approximation 1. Ordnung (rot) und der Unscented-Transformation (blau) mit dem Ergebnis der Monte-Carlo-Simulation.

zu erwartenden Veränderungen entsprechen. Daher wird hier zur robusten Trajektorienplanung die Unscented-Transformation verwendet.

Um eine robuste Prozessplanung durchführen zu können, ist das Gütefunktional der Trajektorienplanung wie in Abschnitt 2.5 in eine Value@Risk Form zu erweitern. Für das beschriebene System ergibt sich das Optimierungsproblem

$$\begin{aligned}
 \{m_{S,0}, \mathbf{U}\}^* &= \arg \min_{m_{S,0}, \mathbf{U}} (\bar{\Phi}_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta}) + \gamma \sigma_{\Phi_{\text{TP}}}^2) \\
 \Phi_{\text{TP}} &= -m_P(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), u_S(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u_S(t) \leq 0,05 \text{ l/h} \\
 \bar{r}_i(\underline{x}(t)) + \lambda \sigma_{r_i} &\leq 0, \quad i = 1, 2, 3 \\
 \underline{r} &= (c_X(t) - c_{X,\text{max}}, c_S(t) - c_{S,\text{max}}, V - V_{\text{max}})^T.
 \end{aligned} \tag{4.5}$$

Im vorliegenden Fall ist vor allem die Verringerung der Variabilität des Prozessergebnisses relevant. Leichte Überschreitungen von Grenzen können dabei toleriert werden. Daher sind $\lambda = 0$ und $\gamma = 1$ gewählt.

Die Lösung des Optimierungsproblems erfolgte mit der gleichen Diskretisierung wie bei der klassischen Trajektorienplanung und den im Rahmen der Monte-Carlo-Simulation

4. Anwendung robuster Prozessführungsverfahren

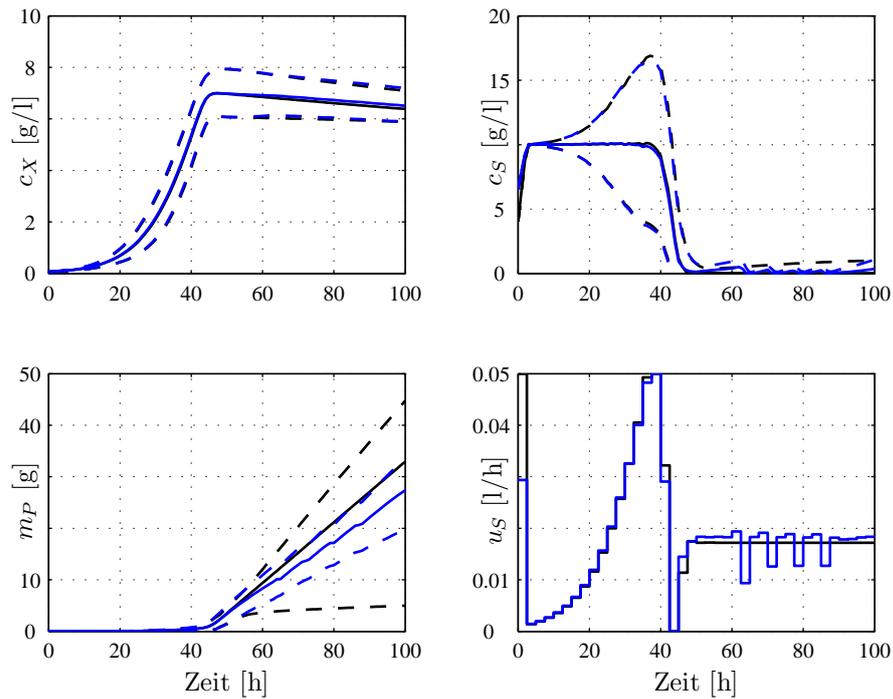


Abbildung 4.4.: Ergebnis der robusten Trajektorienplanung (blau) im Vergleich zur klassischen Trajektorienplanung (schwarz).

vorgestellten Parameterunsicherheiten, allerdings im Single-Shoot-Verfahren, da die hohe Anzahl an Übergangsbedingungen pro Schnitt den Vorteil des Multiple-Shootings aufwiegt.

Das Modell hat vier Zustände. Bei 9 unsicheren Parametern sind 19 Simulationen des Modells zur Berechnung der statistischen Momente notwendig. Das ergibt pro Schnitt $4 \cdot 19$ Gleichungsnebenbedingungen. Schneidet man bei jedem Stellgrößenumschaltzeitpunkt ergeben sich 51 Schnitte und somit in Summe 3876 zusätzliche Freiheitsgrade und Gleichungsnebenbedingungen. Aufgrund der Größe der dann im Rahmen der Optimierung zu verarbeitenden Matrizen ergeben sich somit massive numerische und speichertechnische Probleme. Als numerisches Verfahren wurde ebenfalls der Optimierer SNOPT verwendet.

Das Ergebnis der Planung ist in Abbildung 4.4 in blau im Vergleich zur klassisch geplanten Trajektorie in schwarz dargestellt. Die durchgezogenen Linien markieren den berechneten Prozessverlauf. Zur Verifikation des Ergebnisses wird zunächst die Monte-Carlo-Simulationsstudie aus dem letzten Abschnitt für diese robust geplante Trajektorie wiederholt. Um einen direkten Vergleich zu ermöglichen, werden dazu dieselben 10000 Samples wie in der ersten Simulationsstudie verwendet. Die aus der Monte-Carlo-Simulation resultierenden 3σ -Grenzen sind zusätzlich als gestrichelte Linien dargestellt.

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

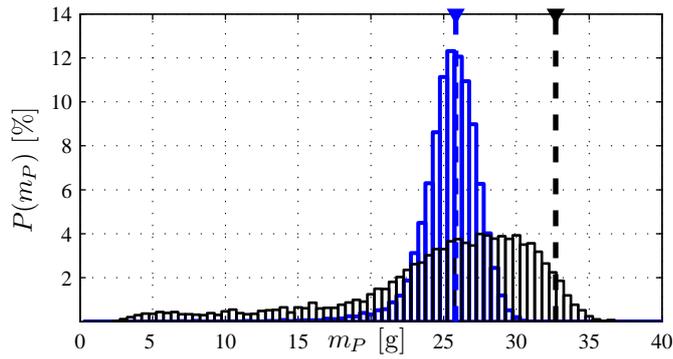


Abbildung 4.5.: Histogramm der Produktivität des klassisch geplanten Prozesses (schwarz) im Vergleich zur robust geplanten Trajektorie (blau) bei Annahme von Modellunsicherheiten.

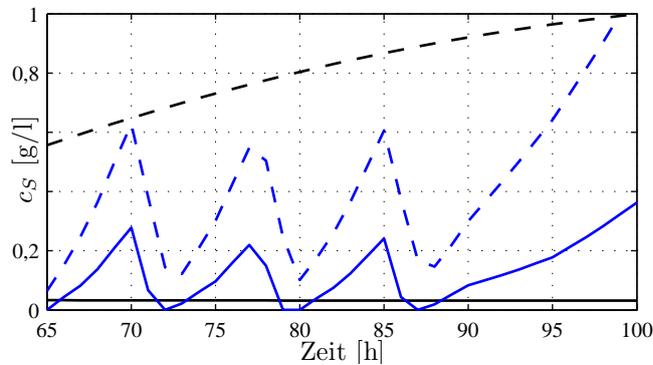


Abbildung 4.6.: Detaildarstellung des Ergebnisses der robusten Trajektorienplanung (blau) im Vergleich zur klassischen Trajektorienplanung (schwarz).

Im Vergleich zur Abbildung 4.3 erkennt man eine minimal verringerte mittlere Produktivität bei einer signifikant kleineren Streuung. Die Verteilung der resultierenden Produktivität ist wiederum als Histogramm in Abbildung 4.5 in blau im Vergleich zum Ergebnis der ersten Studie in schwarz dargestellt. Die gestrichelten Linien markieren die im Rahmen der Optimierung errechnete Produktmenge. Während als Ergebnis der Trajektorienplanung eine Produktmenge am oberen Ende der Verteilung resultierte, liefert die robuste Trajektorienplanung mit der Unscented-Transformation nahezu den statistischen Mittelwert der Produktivität des nicht robust geplanten Falles (24,8 g, siehe Abbildung 4.5). Es ist zu erkennen, dass bei fast identischer mittlerer Produktivität eine weitaus geringere Varianz erreicht ist.

Es bleibt zu klären, warum die neu geplante Trajektorie in Abbildung 4.4 die Varianz so stark verringern kann. Auch der robust geplante Prozessverlauf teilt sich in zwei Phasen. Zuerst wird ebenfalls Substrat so zudosiert, dass sich die maximal zulässige Substratkon-

4. Anwendung robuster Prozessführungsverfahren

zentration und somit eine hohe Wachstumsrate bei geringer Produktionsrate einstellt. Die zweite Phase unterscheidet sich jedoch im Zudosierprofil signifikant von der klassischen Planung. Anstatt eine bestimmte Substratkonzentration in der Brühe einzustellen, wird abwechselnd eine hohe und niedrige Zudosiermenge verwendet.

Zur genaueren Klärung dieses Ergebnisses ist in Abbildung 4.6 die Substratkonzentration in der Brühe für die zweite Prozessphase vergrößert dargestellt. In schwarz ist der Verlauf der klassischen Trajektorie durchgezogen - Mittelwert, gestrichelt - 3σ -Grenze) eingezeichnet. Man erkennt, dass bei der nahezu konstanten Zudosierung zwar im Mittel eine konstante Substratkonzentration zu erwarten ist, die Varianz der Konzentration über die Zeit aber signifikant steigt. Mit sehr hoher Wahrscheinlichkeit wird also nach einiger Zeit die Substratkonzentration neben der optimalen Konzentration liegen.

Die robuste Planung (blau) liefert ein Dosierprofil, welches abwechselnd eine hohe Menge und eine geringe Menge Substrat zudosiert. Während die hohe Menge dosiert wird, steigt die Substratkonzentration in der Brühe und wie bei der klassischen Planung die Varianz dieser Konzentration. In Zeiträumen niedriger Dosierung sinkt die Substratkonzentration nahezu zu null. Dies stellt im Prozessverlauf einen sicheren Punkt dar. Auch bei variablen Modellparametern wird diese geringe Konzentration mit hoher Wahrscheinlichkeit erreicht, die Varianz verringert sich somit ebenfalls auf sehr geringe Werte. Obwohl dadurch die Substratkonzentration nicht immer am optimalen Wert liegt, wird auch bei unsicheren Modellparametern dauerhaft eine Substratkonzentration nahe dem optimalen Punkt eingestellt.

4.1.3. Nichtlineare-Modellprädiktive-Folgeregelung

Mit der robusten Trajektorienplanung ist eine Steuerung gefunden, die mit hoher Sicherheit die vorhergesagte Produktionsmenge erreicht. Dennoch können weitere Modellunsicherheiten und Störungen den Prozessverlauf beeinflussen, was einen korrigierenden Stelleingriff bedingt. Da es sich im vorliegenden Fall um ein nichtlineares System im Fedbatchbetrieb handelt, ist der Einsatz linearer Regelungsverfahren nicht erfolgversprechend. Hier kommt daher die Nichtlineare-Modellprädiktive-Folgeregelung zum Einsatz und wird in einer Simulationsstudie mit den Ergebnissen aus der optimalen Steuerung verglichen.

Als Szenario für dieses Regelungsproblem seien die gleichen Rahmenbedingungen wie für die optimalen Steuerungen verwendet. Die Regelung greift alle 5 h ohne Totzeit ($T_C = 0$) ein. Die Stellgrößentrajektorie ist wie bei der Planung als stückweise konstant mit einer Abtastzeit von ebenfalls 5 h vorgesehen.

Ziel der Regelung ist, der klassisch geplanten Trajektorie zu folgen. Die robust geplante Trajektorie beachtet bereits alle zu erwartenden Unsicherheiten über den gesamten Prozessverlauf und stellt damit eine sehr konservative Planung dar. Durch den

4.1. Unstrukturiertes Modell eines Sekundärmetabolitbildners

Einsatz der Regelung soll jedoch u.U. eine höhere Produktivität erreicht werden. Der Prädiktionshorizont wird zu $T_P = 40$ h der Stellhorizont zu 30 h gewählt.

Die Nichtlineare-Modellprädiktive-Folgeregelung löst in jedem Abtastschritt das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\Phi_{\text{NMPC}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \underline{\theta})) \\
 \text{mit: } \Phi_{\text{NMPC}} &= \int_{t_k}^{t_k+T_P} \left(m_X(t|\hat{\mathbf{x}}_k) - m_X^{(\text{TP})}(t) \right)^2 \\
 &\quad + \left(m_P(t|\hat{\mathbf{x}}_k) - m_P^{(\text{TP})}(t) \right)^2 dt \\
 \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), u_S(t), \underline{\theta}), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_k + T_P \\
 0 &\leq \underline{u}_S(t) \leq 0,051/\text{h} \\
 \underline{r}(\mathbf{x}(t)) &\leq \mathbf{0} \\
 \underline{r} &= (c_X(t) - c_{X,\max}, c_S(t) - c_{S,\max}, V - V_{\max})^T .
 \end{aligned} \tag{4.6}$$

Das Gütefunktional der NMPC verwendet hier ausschließlich die Biotrockenmasse und die Produktmenge als Vorgabe. Dem Regler wird damit die Möglichkeit eingeräumt, insbesondere durch Variation der Substratkonzentration die gewünschte Produktivität einzustellen.

Um die Performance der Regelung zu prüfen, wird eine Monte-Carlo-Simulation mit den gleichen Voraussetzungen wie bei der Prüfung der optimalen Steuerungen durchgeführt. Aus der Verteilung der Modellparameter wird jeweils zufällig ein Parametersatz generiert. Das somit erzeugte Systemmodell simuliert das reale Prozessverhalten. Der Regler arbeitet mit dem nominalen Modell, kennt also die Störung nicht. Für den Einsatz des Regelungsverfahrens ist jedoch zusätzlich die Schätzung des Systemzustandes im online Betrieb notwendig. Hierzu soll ein asymptotisch effizienter Zustandsschätzer mit einer Varianz von 1% verwendet werden. Im Rahmen der Simulationsstudie wird dies dadurch erreicht, dass als Ergebnis der Zustandsschätzung der vom gestörten Modell gelieferte Systemzustand plus einem mittelwertfreien, weißen, normalverteilten Rauschen mit der o.g. Varianz verwendet wird.

Die Lösung des Optimierungsproblems (4.7) erfolgt mit dem Optimierer SNOPT im Multiple-Shooting-Verfahren. Die Optimierung benötigt auf einem aktuellen PC nur wenige Sekunden. Auf eine höhere Abtastrate wird hier jedoch verzichtet, um der in der Biotechnologie üblichen schlechten Messsituation Rechnung zu tragen. In Abbildung 4.7 ist das Ergebnis der Simulationsstudie mit 5347 modellprädiktiv geregelten Fermentationen im Vergleich zu den optimalen Steuerungen als Summenhistogramm dargestellt. Die Nichtlineare-Modellprädiktive-Folgeregelung (schwarz durchgezogen) erreicht in der Produktivität nur den gleichen Mittelwert, wie die optimalen Steuerungen. Die Variabilität ist geringer als bei der klassisch geplanten Trajektorie (schwarz gestrichelt), der die

4. Anwendung robuster Prozessführungsverfahren

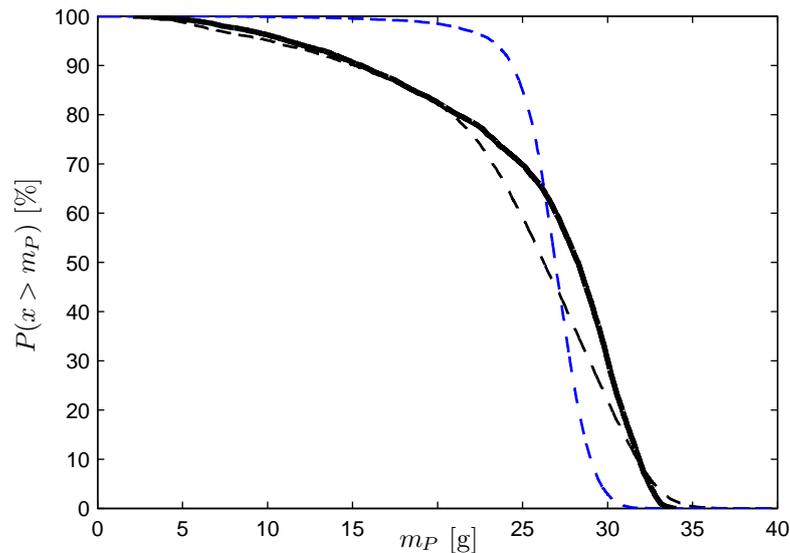


Abbildung 4.7.: Summenhistogramm der Produktionswahrscheinlichkeiten der robusten Trajektorienplanung (blau gestrichelt) und der klassischen Trajektorienplanung (schwarz gestrichelt) im Vergleich zur Nichtlinearen-Modellprädiktiven-Folgeregelung (schwarz durchgezogen).

Nichtlineare-Modellprädiktive-Folgeregelung folgen soll, jedoch höher als bei der Umsetzung der robust geplanten Trajektorie.

Bei den gewählten Einstellungen für die Nichtlineare-Modellprädiktive-Folgeregelung ist aufgrund der geringen Abtastzeit und der Unsicherheit bei der Zustandsschätzung somit nicht genug regelnder Eingriff möglich, um die Variabilität des Prozessergebnisses so stark zu reduzieren, wie es durch eine Planung unter Berücksichtigung zu erwartender Unsicherheiten erreicht wurde. Jede Verringerung der Abtastzeit und jede Verbesserung der Zustandsschätzung verbessern jedoch das Ergebnis der Nichtlinearen-Modellprädiktiven-Folgeregelung signifikant.

4.1.4. Online-Trajektorienplanung

In [Hei04] wird die These aufgestellt, dass eine Neuplanung der gesamten Trajektorie im online Betrieb bessere Ergebnisse erzielen kann, als die Nichtlineare-Modellprädiktive-Folgeregelung. Dies basiert auf der Annahme, dass im Batchbetrieb der optimale Weg zurück zur offline geplanten Trajektorie nicht zwangsläufig optimale Produktivität garantiert. Daher soll hier die Online-Trajektorienplanung ebenfalls untersucht werden.

Der Prädiktionshorizont der Online-Trajektorienplanung reicht immer bis zum Ende des Batches. Als Gütefunktional dient die gleiche Funktion, die für die Offline-Planung

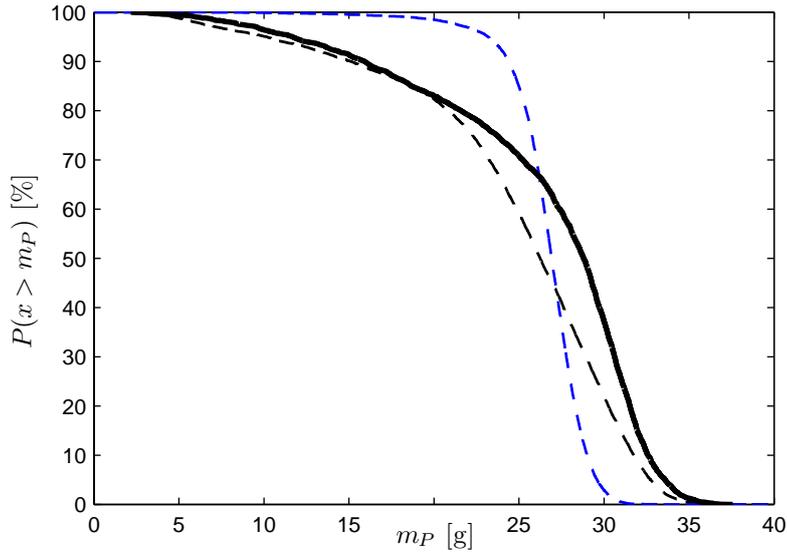


Abbildung 4.8.: Summenhistogramm der Produktionswahrscheinlichkeiten der robusten Trajektorienplanung (blau gestrichelt) und der klassischen Trajektorienplanung (schwarz gestrichelt) im Vergleich zur Online-Trajektorienplanung (schwarz durchgezogen).

der optimalen Steuertrajektorie verwendet wurde. Die Online-Trajektorienplanung löst somit zu jedem Abtastschritt das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\Phi_{\text{OT}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \theta)) \\
 \text{mit: } \Phi_{\text{OT}} &= -m_P(t_{\text{Ende}}) \\
 \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), u_S(t), \theta), \quad \underline{x}(t = t_k) = \hat{\underline{x}}_k \\
 t_k &\leq t \leq T_{\text{Ende}} \\
 0 &\leq u_S(t) \leq 0,051/\text{h} \\
 \underline{r}(\underline{x}(t)) &\leq \underline{0} \\
 \underline{r} &= (c_X(t) - c_{X,\text{max}}, c_S(t) - c_{S,\text{max}}, V - V_{\text{max}})^T.
 \end{aligned} \tag{4.7}$$

Im Rahmen einer Simulationsstudie mit den gleichen Einstellungen wie bei der Nichtlinearen-Modellprädiktiven-Folgeregelung wurde die Regelgüte der Online-Trajektorienplanung bei Unsicherheiten in der Zustandschätzung und in den Modellparametern untersucht.

Abbildung 4.8 zeigt das Ergebnis dieser Studie. In schwarz und blau gestrichelt sind nochmals die Ergebnisse der optimalen Steuerung dargestellt. Die Online-Trajektorienplanung (schwarz durchgezogen) liefert ein etwas besseres Ergebnis als die Nichtlineare-Modellprädiktive-Folgeregelung, kann aber, wie diese, das Ergebnis der robusten Trajektorienplanung nicht erreichen.

4.1.5. Robuste Nichtlineare-Modellprädiktive-Folgeregelung

Das Ergebnis der klassischen Nichtlinearen-Modellprädiktiven-Folgeregelung bleibt im Vergleich hinter der robusten Planung zurück. Es ist also sinnvoll, auch im Rahmen der Regelung die zu erwartenden Unsicherheiten zu berücksichtigen. Zunächst wird daher die robuste Nichtlineare-Modellprädiktive-Folgeregelung zur Regelung des Fermentationsprozesses verwendet. Wie die robuste Trajektorienplanung verwendet die robuste Nichtlineare-Modellprädiktive-Folgeregelung intern einen Value@Risk Ansatz zur robusten Lösung des Prozessführungsproblems. Neben den Unsicherheiten der Modellparameter kann hier auch der aus der Zustandrekonstruktion resultierende zu erwartende Schätzfehler in der Optimierung beachtet werden.

Das Szenario entspricht dem der Nichtlinearen-Modellprädiktiven-Folgeregelung. Auch das robuste Verfahren hat als Ziel das Ergebnis der klassischen Trajektorienplanung, da die robuste Planung bereits alle zu erwartenden Unsicherheiten enthält und somit einen sehr konservativen Prozessverlauf beschreibt. Die robuste Nichtlineare-Modellprädiktive-Folgeregelung löst in jedem Abtastschritt das folgende Gütefunktional

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} \left(\bar{\Phi}_{\text{NMPC}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \underline{\theta}) + \gamma \sigma_{\Phi_{\text{NMPC}}}^2 \right) \\
 \text{mit: } \Phi_{\text{NMPC}} &= \int_{t_k}^{t_k+T_P} \left(m_X(t|\hat{\mathbf{x}}_k) - m_X^{(\text{TP})}(t) \right)^2 \\
 &\quad + \left(m_P(t|\hat{\mathbf{x}}_k) - m_P^{(\text{TP})}(t) \right)^2 dt \\
 \dot{\mathbf{x}} &= \mathbf{f}(t, \mathbf{x}(t), \mathbf{u}(t), \underline{\theta}), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u(t) \leq 0,051/\text{h} \\
 \bar{r}_i(\mathbf{x}(t)) + \lambda \sigma_{r_i} &\leq \underline{0}, \quad i = 1, 2, 3 \\
 r_1 &= c_X(t) - c_{X,\max}, r_2 = c_S(t) - c_{S,\max}, r_3 = V - V_{\max}.
 \end{aligned} \tag{4.8}$$

Wie bereits bei der robusten Trajektorienplanung beschrieben, sind leichte Verletzungen von Grenzwerten tolerierbar, weswegen auch hier $\lambda = 0$ verwendet wird. Die Unsicherheit der Gütefunktion Φ steigt signifikant über den Prädiktionshorizont. Eine gleichmäßige Gewichtung der Varianz über den Prädiktionshorizont bedingt einen sehr hohen Einfluss weit in der Zukunft liegender Zeitpunkte. Daher wird hier, wie in Abschnitt 2.5 beschrieben, eine zeitvariable Gewichtung der Varianz des Gütefunktionalen erfolgen. Ausgehend von einem, wie in (2.101) angegeben diskretisierten Gütefunktional, wird zunächst $\gamma = 100 (t_i - t_k)^{-2}$ verwendet.

Um eine direkte Vergleichbarkeit der Simulationsstudien zu gewährleisten, hat die robuste Nichtlineare-Modellprädiktive-Folgeregelung im Rahmen der Monte-Carlo-Simulationsstudie exakt die gleichen Störungen zu bewältigen, wie die klassische Nichtlineare-Modellprädiktive-Folgeregelung.

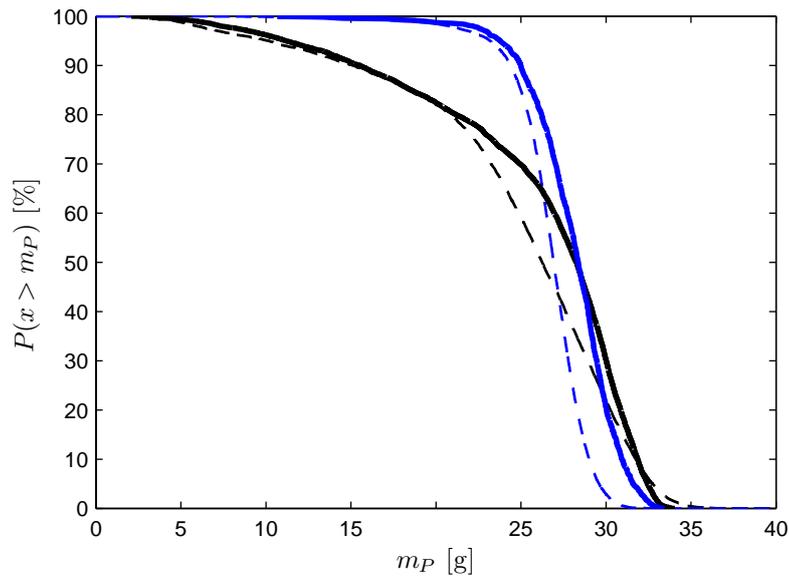


Abbildung 4.9.: Summenhistogramm der Produktionswahrscheinlichkeiten der robusten Trajektorienplanung (blau gestrichelt), der klassischen Trajektorienplanung (schwarz gestrichelt) und der klassischen Nichtlinearen-Modellprädiktiven-Folgeregelung (schwarz durchgezogen) im Vergleich zur robusten Nichtlinearen-Modellprädiktiven-Folgeregelung (blau durchgezogen).

Das Ergebnis einer Studie mit über 1000 geregelten Fermentationen ist als Summenverteilung der Produktivität in Abbildung 4.9 dargestellt. Als Vergleich sind als gestrichelte Linien die Ergebnisse aus der Umsetzung der klassischen Trajektorie (schwarz) und der robusten Trajektorie (blau) eingezeichnet. Die schwarze durchgezogene Linie markiert das Ergebnis der Simulationsstudie aus dem letzten Abschnitt bei Anwendung der klassischen Nichtlinearen-Modellprädiktiven-Folgeregelung.

Die robuste Nichtlineare-Modellprädiktive-Folgeregelung (blau durchgezogen) liefert ein weitaus besseres Ergebnis als die klassische Nichtlineare-Modellprädiktive-Folgeregelung. Der Mittelwert der Produktmenge ist zwar nur nahezu identisch zur Nichtlinearen-Modellprädiktiven-Folgeregelung, die Varianz jedoch weitaus geringer. Allerdings ist dieses Ergebnis auch nur marginal besser als die gesteuerte Umsetzung der robusten Trajektorie.

Die durch die Messungen während der Fermentation zugänglichen Informationen bringen hier nur ein geringes Maß an zusätzlichen Informationen. Beim Auftreten von Störungen, die nicht explizit in der Planung berücksichtigt werden können, ist dieser regelnde Eingriff jedoch essentiell.

Inbesondere der nur geringe Vorteil den die Regelung gegenüber der robusten Steue-

4. Anwendung robuster Prozessführungsverfahren

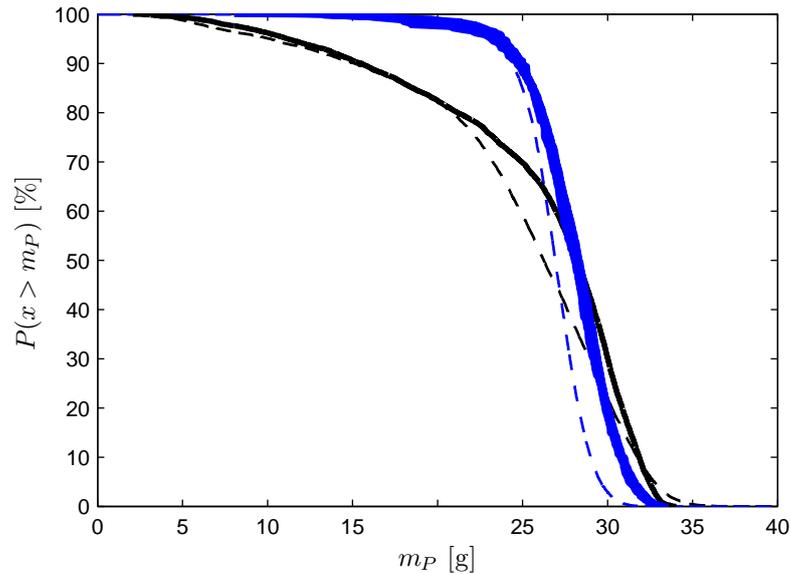


Abbildung 4.10.: Summenhistogramm der Produktionswahrscheinlichkeiten der robusten Trajektorienplanung (blau gestrichelt), der klassischen Trajektorienplanung (schwarz gestrichelt) und der klassischen Nichtlineare-Modellprädiktive-Folgerregelung (schwarz durchgezogen) im Vergleich zu verschiedenen eingestellten robusten Nichtlinearen-Modellprädiktiven-Folgerregelung (alle blau durchgezogen).

Die Abbildung zeigt, bedingt die Frage nach der optimalen Einstellung der Parameter des Gütefunktionals. Daher wurde die beschriebene Simulationsstudie mit $\gamma = 1 (t_i - t_k)^{-2}$, $\gamma = 1 (t_i - t_k)^{-1}$, $\gamma = 10 (t_i - t_k)^{-2}$, $\gamma = 10 (t_i - t_k)^{-1}$, $\gamma = 100 (t_i - t_k)^{-1}$, $\gamma = 1000 (t_i - t_k)^{-2}$, $\gamma = 1$, $\gamma = 0,1$ und $\gamma = 0$ wiederholt. Abbildung 4.10 zeigt die Ergebnisse dieser Simulationsstudien mit jeweils mindestens 300 Simulationen im Vergleich. Erstaunlicherweise ist der Einfluss des Parameters γ nur minimal.

Die Ergebnisse (alle in blau durchgezogen) lassen sich praktisch nicht voneinander unterscheiden. Selbst eine Wahl von $\gamma = 0$ liefert noch sehr gute Ergebnisse. Im vorliegenden Fall bedingt allein die Nutzung des approximierten statistischen Mittelwertes der Prädiktionen diese massive Verbesserung der Regelung. Damit zeigt sich insbesondere ein signifikanter Vorteil der Nutzung der Approximation zweiter Ordnung. Eine Approximation erster Ordnung wird immer die Simulation mit den nominellen Parametern als approximierten Mittelwert liefern.

Es lässt sich damit jedoch nicht verallgemeinernd schließen, dass der Value@Risk Ansatz unnötig ist. Vielmehr bedingt die besondere Modellstruktur des gewählten biotechnologischen Prozesses diese Auswirkungen. Innerhalb des Prädiktionshorizontes gibt es einfach keine Einflussmöglichkeiten, um die Varianz des Gütefunktionals weiter zu senken.

4.1.6. Robuste Online-Trajektorienplanung

In vorangegangenen Abschnitten ist die Online-Trajektorienplanung als Alternative zur Nichtlinearen-Modellprädiktiven-Folgeregelung bei der Regelung von Batchprozessen vorgestellt. Eine Erweiterung dieses Prozessführungsverfahrens über einen Value@Risk Ansatz ist, wie in Abschnitt 2.5 beschrieben, ebenfalls möglich. Wie die Online-Trajektorienplanung die Gütefunktion der Trajektorienplanung online löst, basiert die robuste Online-Trajektorienplanung auf dem Optimierungsproblem der robusten Trajektorienplanung. Die robuste Online-Trajektorienplanung löst somit in jedem Abtastschritt das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\bar{\Phi}_{\text{OT}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \underline{\theta}) + \gamma \sigma_{\Phi_{\text{OT}}}^2) \\
 \text{mit: } \Phi_{\text{OT}} &= -m_{\text{P}}(t_{\text{Ende}}) \\
 \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \mathbf{u}_{\text{S}}(t), \underline{\theta}), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u_{\text{S}}(t) \leq 0,051/\text{h} \\
 \bar{r}_i(\mathbf{x}(t)) + \lambda \sigma_{r_i} &\leq \underline{0}, \quad i = 1, 2, 3 \\
 r_1 &= c_{\text{X}}(t) - c_{\text{X,max}}, r_2 = c_{\text{S}}(t) - c_{\text{S,max}}, r_3 = V - V_{\text{max}}.
 \end{aligned} \tag{4.9}$$

Aus den gleichen Gründen, die in Abschnitt 4.1.2 angegeben sind, wird $\lambda = 0$ gewählt. Die Bewertung der Varianz des Gütefunktional ist zu $\gamma = 5$ eingestellt.

In Abbildung 4.11 sind die Resultate der Anwendung der robusten Online-Trajektorienplanung auf das System im Vergleich zu den anderen beschriebenen Verfahren dargestellt. Die Ergebnisse resultieren aus einer Monte-Carlo-Simulationsstudie mit den gleichen Einstellungen, die oben bereits angegeben sind. Aufgrund des sehr hohen Rechenaufwands zur Berechnung eines mit der robusten Online-Trajektorienplanung geregelten Versuchslaufes, stehen nur 167 Samples zur Auswertung zur Verfügung. Im vorliegenden Fall liefert die robuste Online-Trajektorienplanung, ebenso wie die robuste Nichtlineare-Modellprädiktive-Folgeregelung, etwas bessere Ergebnisse als die gesteuerte Umsetzung der robust geplanten Trajektorie. Obwohl die Online-Trajektorienplanung marginal bessere Ergebnisse liefert als die Nichtlineare-Modellprädiktive-Folgeregelung, ist bei der robusten Regelung die robuste Nichtlineare-Modellprädiktive-Folgeregelung leicht im Vorteil.

4.1.7. Zusammenfassung

In den letzten Abschnitten wurden verschiedene klassische und auf der Unscented-Transformation basierende robuste Prozessführungsverfahren bei der Anwendung auf ein einfaches Modell eines Sekundärmetabolitbildners untersucht. Bemerkenswert ist hierbei das Ergebnis der robusten Trajektorienplanung, welches nahezu genauso gute Produktivität liefert wie die robusten Regelungsverfahren. Allein durch die Beachtung zu erwartender

4. Anwendung robuster Prozessführungsverfahren

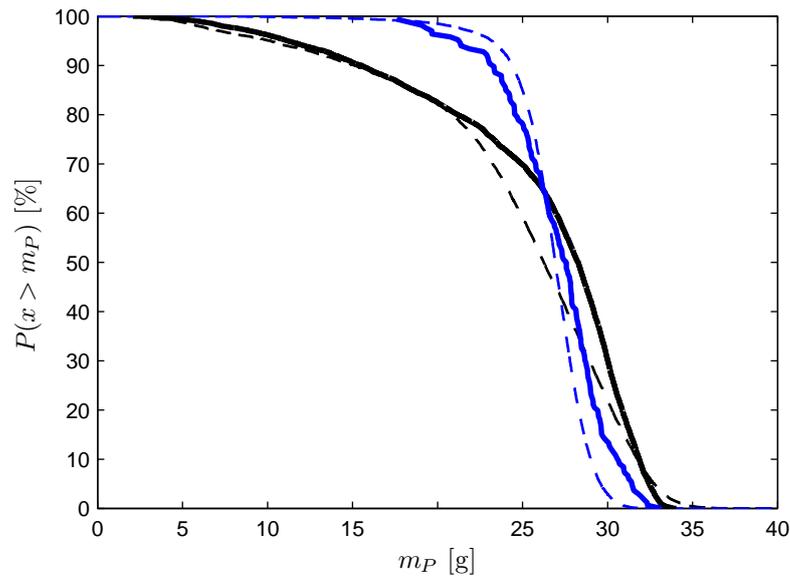


Abbildung 4.11.: Summenhistogramm der Produktionswahrscheinlichkeiten der robusten Trajektorienplanung (blau gestrichelt), der klassischen Trajektorienplanung (schwarz gestrichelt) und der klassischen Nichtlinearen Modellprädiktiven-Folgeregelung (schwarz durchgezogen) im Vergleich zur robusten Online-Trajektorienplanung (blau durchgezogen).

Unsicherheiten, liefern alle robusten Verfahren eine weniger schwankende Produktivität als die klassischen Ansätze.

Insbesondere das gute Ergebnis der robusten Trajektorienplanung wirft die Frage nach dem Sinn des Einsatzes von Regelungsverfahren auf, da die Implementierung eines modellprädiktiven Reglers oder sogar eines robusten modellprädiktiven Reglers sehr aufwändig ist. Die robuste Steuerung beachtet zwar Unsicherheiten, Einflüsse, die nicht im Modell beschrieben sind, werden jedoch auch bei robuster Planung Abweichungen und Produktivitätsschwankungen bedingen. Daher wird in realer Umsetzung der Einsatz eines Regelungsverfahrens verlässlichere Ergebnisse liefern, als eine reine Steuerung.

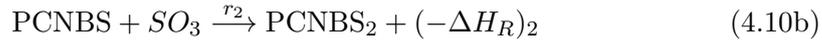
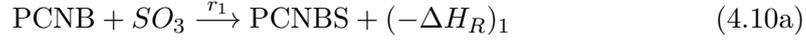
4.2. Sulfonierung von Parachlornitrobenzol

Neben der im letzten Abschnitt beschriebenen Anwendung der robusten Prozessführungsverfahren zur Verringerung von Produktionsschwankungen, ist es über die Verwendung des Parameters λ in der erweiterten Beschränkungsfunktion möglich, einen Sicherheitsabstand zu Systembeschränkungen einzuhalten. Im Folgenden soll die exotherme Umsetzung von Parachlornitrobenzol (PCNB) zur Parachlornitrobenzolsulfonsäure (PCNBS)

4.2. Sulfonierung von Parachlornitrobenzol

diese Anwendung robuster Prozessführungsverfahren am Beispiel näher bringen. Die Reaktion ist [Kin89] entnommen, wird hier jedoch der Anschaulichkeit wegen in einigen Punkten vereinfacht wiedergegeben.

Die Umsetzung von Parachlornitrobenzol zu Parachlornitrobenzolsulfonsäure erfolgt im Semibatch-Betrieb. Im Reaktor wird Schwefeltrioxid (SO_3) vorgehalten und das Parachlornitrobenzol sukzessive zudosiert. Das Reaktionsschema der Umsetzung ist



Neben der direkten Reaktion von Schwefeltrioxid und Parachlornitrobenzol zu Parachlornitrobenzolsulfonsäure kann das Produkt Parachlornitrobenzolsulfonsäure auch weiter zu Parachlornitrobenzoldisulfonsäure reagieren. Die Parachlornitrobenzoldisulfonsäure wiederum reagiert mit dem Edukt Parachlornitrobenzol in einer ebenfalls exothermen Reaktion wieder zum gewünschten Produkt Parachlornitrobenzolsulfonsäure.

Zusätzlich zu diesen drei Reaktionen kann sich das Produkt Parachlornitrobenzolsulfonsäure bei zu hohen Temperaturen in einer autokatalytischen Reaktion zersetzen, was die besondere Brisanz dieser Umsetzung ausmacht. Nähere Informationen zu dieser Zerfallsreaktion können [Kin89] entnommen werden. Hier wird dieser Zerfallsreaktion im Rahmen der Prozessführung durch Vorgabe einer maximalen Reaktionstemperatur Rechnung getragen. Die auf den Reaktionspfeilen angegebenen r_i bezeichnen die jeweiligen auf die Gesamtmasse bezogenen Reaktionsgeschwindigkeiten, die im Folgenden noch definiert werden. Werte der Reaktionsenthalpien $(-\Delta H_R)_i$ können der Tabelle 4.2 entnommen werden.

Bilanzierung eines ideal durchmischten zylindrischen Rührkesselreaktors mit ideal geregelter Manteltemperatur liefert das gewöhnliche Differentialgleichungssystem

$$\dot{n}_{PCNB} = -(r_1 + r_3) \cdot m + \frac{\dot{m}_{zu}}{M_{PCNB}} \quad (4.11a)$$

$$\dot{n}_{SO_3} = -(r_1 + r_2) \cdot m \quad (4.11b)$$

$$\dot{n}_{PCNBS} = (r_1 - r_2 + 2r_3) \cdot m \quad (4.11c)$$

$$\dot{n}_{PCNBS_2} = (r_2 - r_3) \cdot m \quad (4.11d)$$

$$\dot{m} = \dot{m}_{zu} \quad (4.11e)$$

$$\begin{aligned} \dot{U} = & m \cdot (r_1 \cdot (-\Delta H_R)_1 + r_2 \cdot (-\Delta H_R)_2 + r_3 \cdot (-\Delta H_R)_3) \\ & + \alpha A (T_M - T) + \dot{m}_{zu} \bar{c}_p T_{zu} \end{aligned} \quad (4.11f)$$

Die Variablen $n_{(\cdot)}$ bezeichnen die Molzahl der verschiedenen Reaktanden, $\dot{m}_{zu} = u(t)$ und M_{PCNB} den Massenstrom und die Molmasse des zudosierten Edukts Parachlornitrobenzol und m die Gesamtmasse der Reaktanden. Im Rahmen der Energiebilanz (4.11f) finden sich weiterhin die Reaktortemperatur T , die aktive Wärmeaustauschfläche

4. Anwendung robuster Prozessführungsverfahren

A , die sich aus dem Durchmesser D und der Füllhöhe des Reaktors zusammensetzt, der Wärmedurchgangskoeffizient α , die ideal geregelte Manteltemperatur T_M , \bar{c}_p die als konstant und unabhängig von den Konzentrationen der Reaktanden angenommene spezifische Wärmekapazität und T_{zu} die Temperatur des Zulaufes. Die Reaktionskinetiken k_i sind

$$r_1 = k_{10} e^{-\frac{E_1}{RT}} c_{PCNB}^{0.6} c_{SO_3}^3 \quad (4.11g)$$

$$r_2 = k_{20} e^{-\frac{E_2}{RT}} c_{PCNBS} c_{SO_3} \quad (4.11h)$$

$$r_3 = k_{30} e^{-\frac{E_3}{RT}} c_{PCNBS_2} c_{PCNB}, \quad (4.11i)$$

wobei k_{i0} den Stoßfaktor und E_i die Aktivierungsenergie der i -ten Reaktion und $c_{(\cdot)}$ die volumetrischen Molkonzentrationen der Reaktanden beschreiben. Die Reaktionskinetik r_1 ist abweichend von [Kin89] vereinfacht angenommen.

Mit dem Volumen $V = \frac{m}{\rho}$ der Reaktanden im Reaktor lassen sich die Molkonzentrationen mit

$$c_{PCNB} = \frac{n_{PCNB}}{V} \quad c_{SO_3} = \frac{n_{SO_3}}{V} \quad (4.11j)$$

$$c_{PCNBS} = \frac{n_{PCNBS}}{V} \quad c_{PCNBS_2} = \frac{n_{PCNBS_2}}{V} \quad (4.11k)$$

berechnen. Der Systemzustand des Modells setzt sich als

$$\underline{x} = (n_{PCNB}, n_{SO_3}, n_{PCNBS}, n_{PCNBS_2}, m, U)^T \quad (4.11l)$$

zusammen. Als Messgröße steht die Reaktortemperatur zur Verfügung. Es ergibt sich die Messgleichung

$$y = T = \frac{U}{m\bar{c}_p} \quad (4.11m)$$

Die Zahlenwerte sämtlicher auftretenden Modellparameter $\underline{\theta}$ können der Tabelle 4.2 entnommen werden.

4.2.1. Trajektorienplanung

Ausgehend von dem mathematischen Modell der Sulfonierungsreaktion (4.11) soll zunächst eine optimale Steuertrajektorie zur Durchführung dieser Umsetzung berechnet werden. Ziel ist es dabei bei einer Vorlage von 20 kmol Schwefeltrioxid in einer Konzentration von 6,7 mol/g und einer initialen Reaktortemperatur von 350K nach einer Batchlaufzeit von $t_{\text{Ende}} = 3$ h eine maximale Menge Parachlornitrobenzolsulfonsäure (n_{PCNBS}) zu produzieren.

4.2. Sulfonierung von Parachlornitrobenzol

Tabelle 4.2.: Parameter des Sulfonierungsmodells

ρ	1, 2 g/cm ³	k_{10}	$e^{39,7} (\text{cm}^3)^{3,6} / (\text{mol}^{2,6} \text{g s})$
\bar{c}_P	1, 6 J/(g K)	k_{20}	$e^{125,2} (\text{cm}^3)^2 / (\text{mol g s})$
M_{PCNB}	153, 5 g/mol	k_{30}	$e^{85,7} (\text{cm}^3)^2 / (\text{mol g s})$
α	100 J/(s m ² K)	$(-\Delta H_R)_1$	66170 J/mol
D	1, 5 m	$(-\Delta H_R)_2$	15100 J/mol
E_1	97251 J/mol	$(-\Delta H_R)_3$	51070 J/mol
E_2	439001 J/mol	R	8, 314 J/(g K)
E_3	266174 J/mol	T_M	350 K

Die autokatalytische Zersetzung des Produktes soll verhindert werden. Daher darf die Reaktortemperatur T eine kritische Temperatur $T_{\text{krit}} = 405\text{K}$ nicht überschreiten. Das Optimierungsproblem liest sich dann wie folgt.

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} (\Phi_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})) \\
 \Phi_{\text{TP}} &= - \int_0^{t_{\text{Ende}}} n_{\text{PCNBS}}^2(t) dt + 1000 n_{\text{PCNB}}^2(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u(t) \leq 500 \text{g/s} \\
 r(\underline{x}(t)) &\leq 0 \\
 r &= T(t) - T_{\text{krit}}
 \end{aligned} \tag{4.12}$$

Um eine schnelle Umsetzung zu erhalten, wird hier eine Integralform des Gütefunktional verwendet. Da das Edukt Parachlornitrobenzol am Ende des Batches möglichst nicht mehr akkumuliert im Reaktor vorliegen soll, ist das Gütefunktional um eine Bewertung des finalen Zustandes erweitert. Als Freiheitsgrad für die Optimierung dient die stückweise konstante Zudosierungsrate des Eduktes Parachlornitrobenzol, mit Umschaltzeitpunkten alle 300 Sekunden. Die Lösung des Optimierungsproblems (4.12) wurde mit Hilfe des Multiple-Shooting-Verfahrens und des nichtlinearen Optimierers SNOPT aus dem Softwarepaket Tomlab berechnet.

Abbildung 4.12 zeigt die berechnete Lösung des Optimierungsproblems. Der resultierende Verlauf der Umsetzung entspricht den Erwartungen. Zunächst wird in der ersten Phase Parachlornitrobenzol mit maximaler Rate zudosiert, bis die kritische Reaktortemperatur von 405 K ungefähr bei $t = 0,5 \text{h}$ erreicht wird. In der zweiten Phase wird Parachlornitrobenzol genau so zudosiert, dass die Reaktortemperatur bei 405 K gehalten wird. In der dritten und letzten Phase, die nur schwer zu erkennen ist, wird die

4. Anwendung robuster Prozessführungsverfahren

Zudosierung von Parachlornitrobenzol gestoppt und der Reaktor somit heruntergefahren.

Wie bereits beschrieben, stellt das mathematische Modell nur eine Näherung des realen Systems dar. Insbesondere sind die Modellparameter aus verrauschten Messdaten identifiziert und somit ebenfalls zufällige Größen. Im Folgenden sollen zunächst die Auswirkungen von Parameterfehlern auf den geplanten Prozessverlauf näher untersucht werden. Dazu wird hier angenommen, dass die Modellparameter α , D , E_1 , E_2 , E_3 , k_{10} , k_{20} , k_{30} , $(-\Delta H_R)_1$, $(-\Delta H_R)_2$ und $(-\Delta H_R)_3$ jeweils einen normalverteilten zu erwartenden Fehler mit 2% relativer Standardabweichung beinhalten. Die Auswirkungen auf den Verlauf der chemischen Umsetzung wurden mit einer Monte-Carlo-Simulation mit 10000 zufällig aus der Verteilung der Modellparameter gezogenen Samples berechnet.

Zur grafischen Veranschaulichung soll hier zunächst eine schattierte Darstellung in Anlehnung an die Quantenphysik dienen. Abbildung 4.13 zeigt qualitativ die Verteilung der resultierenden Prozessverläufe. Je dunkler ein Bereich in dieser Darstellung ist, desto mehr Realisierungen der Monte-Carlo-Simulation laufen durch diesen Punkt, desto höher ist also der Wert der Wahrscheinlichkeitsdichtefunktion an diesem Punkt. Insbesondere am Verlauf der Reaktortemperatur ist zu erkennen, dass viele Realisierungen der geplanten Trajektorie die maximale Reaktortemperatur überschreiten.

In Abbildung 4.14 sind die Verteilungsdichte als Histogramm und die kumulative Verteilung der sich in den simulierten Prozessläufen jeweils maximal ergebenden Reaktortemperatur dargestellt. Diese Abbildung zeigt noch einmal deutlich, dass in sehr vielen Fällen die in der Planung der Trajektorie verwendete Temperaturgrenze überschritten wird.

Sei die erste Alarmgrenze bei 410 K, dann wird ein Alarm mit einer Wahrscheinlichkeit von 31 % ausgelöst. Etwa 29 % der Realisierungen überschreiten die maximal zulässige Reaktortemperatur um mehr als 10 K. Dies soll in diesem Szenario den Eingriff sicherheitstechnischer Einrichtungen auslösen und somit zum Abbruch des Prozesses führen. Vergleicht man den so erzielten mittleren Umsatz mit dem in der Trajektorienplanung berechneten Wert von 15,3 kmol, so beträgt die Produktivität der gesteuerten Umsetzung der klassischen Trajektorie 71 %. Viel kritischer ist jedoch die hohe Wahrscheinlichkeit des Auslösens sicherheitsgerichteter Schaltungen oder gar eines Störfalls.

4.2.2. Robuste Trajektorienplanung

Um auch im gesteuerten Betrieb ein hohes Maß an Sicherheit zu gewährleisten, kann die Prozesstrajektorie so geplant werden, dass die Chance einer Grenzverletzung minimal ist. Bevor jedoch eine solche robuste Trajektorie für die Sulfonierungsreaktion geplant werden kann, ist zunächst zu überprüfen, wie gut die im Rahmen der Planung benötigten ersten zwei statistischen Momente des Prozessverlaufes mit den in Abschnitt 2.1 beschriebenen Approximationsverfahren wiedergegeben werden können.

4.2. Sulfonierung von Parachlornitrobenzol

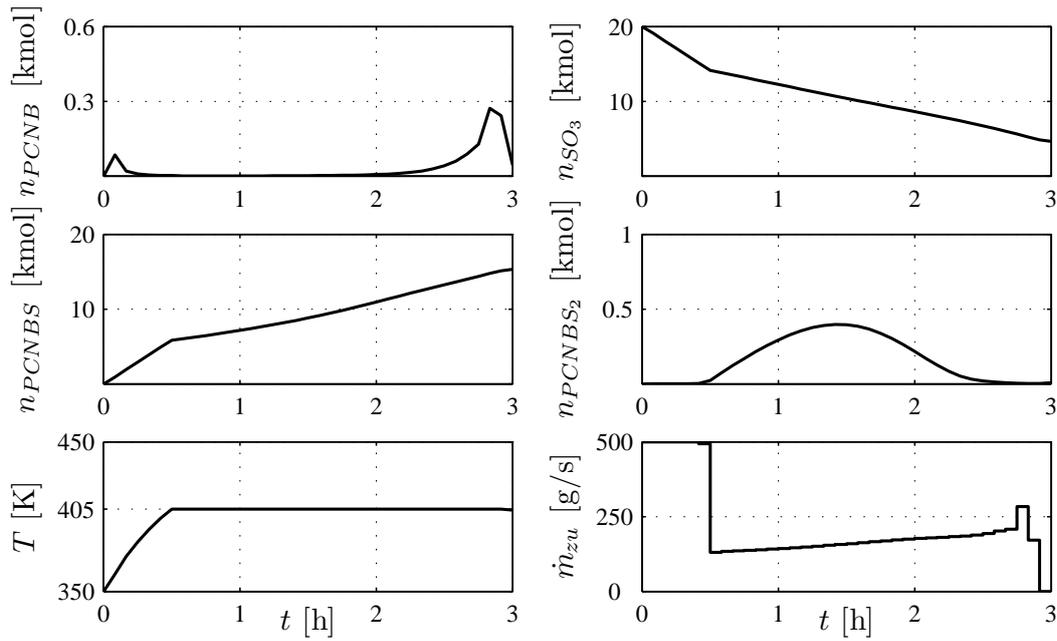


Abbildung 4.12.: Ergebnis der Trajektorienplanung für die Sulfonierung von Parachlornitrobenzol.

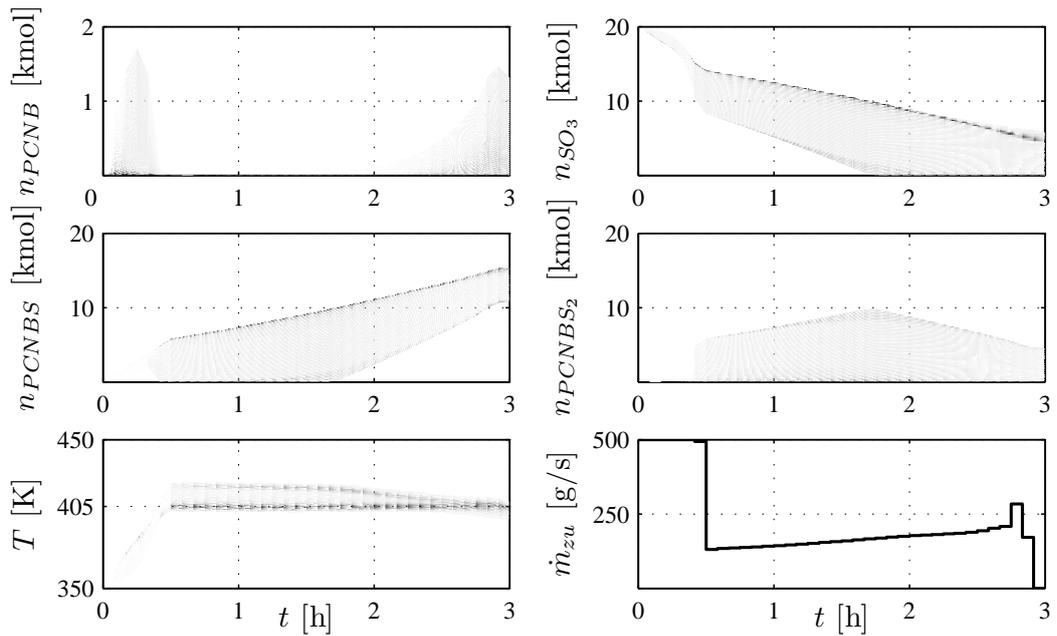


Abbildung 4.13.: Darstellung der statistischen Verteilung des Prozessverlaufes bei gesteuerter Umsetzung der klassischen Trajektorie und Annahme von Modellunsicherheiten.

4. Anwendung robuster Prozessführungsverfahren

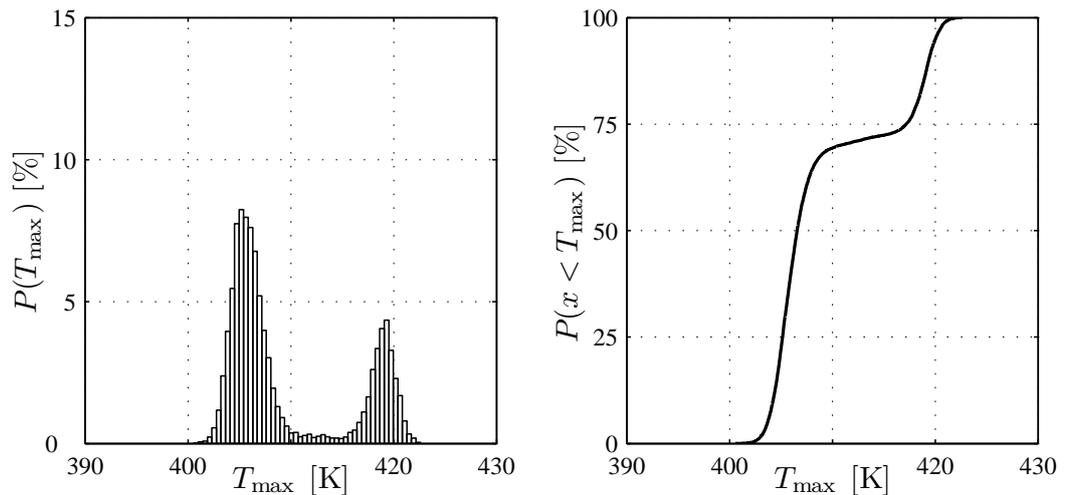


Abbildung 4.14.: Histogramm (links) und Summenverteilung (rechts) der maximalen auftretenden Reaktortemperatur bei gesteuerter Umsetzung der klassisch geplanten Trajektorie.

Grundlage des Vergleiches bilden die mit (2.9) aus der oben beschriebenen Monte-Carlo-Simulation mit 10000 Samples berechneten „wahren“ ersten zwei statistischen Momente. Abbildung 4.15 zeigt in schwarz als durchgezogene Linie die Simulation der Trajektorie mit den nominellen Parametern und als gestrichelte Linie, die sich bei Verwendung von nur zwei statistischen Momenten ergebene 3σ -Grenzen des Prozessverlaufes.

Da das Systemmodell nicht stetig differenzierbar ist, kann die Berechnung der für die Approximation erster Ordnung benötigten Sensitivitäten (2.14) nur mit Hilfe von Differenzenquotienten erfolgen. Die resultierenden 3σ -Grenzen sind in Abbildung 4.15 in rot wiedergegeben. Das Verfahren erster Ordnung liefert somit nur eine sehr schlechte Approximation der statistischen Momente, unterschätzt die Breite der Verteilung in den meisten Bereichen sogar sehr stark. Eine Anwendung dieser Approximationsmethode zur robusten Prozessplanung wird somit nur unzureichende Ergebnisse liefern.

Die Berechnung der statistischen Momente der Prozessgrößen mittels der Unscented-Transformation erfolgt mit der optimalen Schrittweite von $h = \sqrt{3}$. Abbildung 4.15 beinhaltet das Ergebnis der Approximation zweiter Ordnung in blau. Wie zu erwarten war, ermöglicht auch die Unscented-Transformation keine exakte Berechnung der statistischen Momente, jedoch stellen die gezeigten Ergebnisse eine gute Näherung der Ergebnisse der Monte-Carlo-Simulation dar.

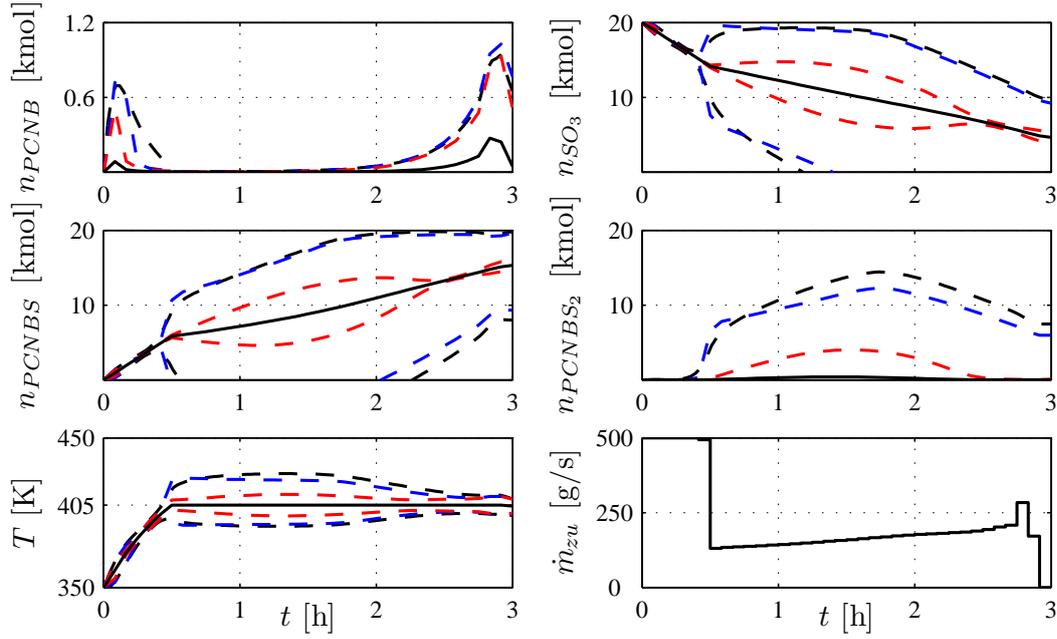


Abbildung 4.15.: Vergleich der Approximation der Varianz des Prozessverlaufes (dargestellt als 3σ -Grenze) bei Berechnung mittels der Monte-Carlo-Simulation (schwarze gestrichelte Linie), der Approximation erster Ordnung (rot) und der Unscented-Transformation (blau). Die durchgezogenen schwarzen Linien markieren die nominelle Simulation.

Um eine robuste Prozessplanung durchführen zu können, ist das Gütefunktional der Trajektorienplanung wie in Abschnitt 2.5 in eine Value@Risk Form zu erweitern. Für das beschriebene System ergibt sich das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} (\bar{\Phi}_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta}) + \gamma \sigma_{\Phi_{\text{TP}}}^2) \\
 \Phi_{\text{TP}} &= - \int_0^{t_{\text{Ende}}} n_{\text{PCNCBS}}^2(t) dt + 1000 n_{\text{PCNCNB}}^2(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq \underline{u}(t) \leq 500 \text{g/s} \\
 \bar{r}(\underline{x}(t)) + \lambda \sigma_r &\leq 0 \\
 r &= T(t) - T_{\text{krit}}.
 \end{aligned} \tag{4.13}$$

Da im vorliegenden Fall die Einhaltung der Temperaturgrenze höchste Priorität hat, wird $\lambda = 3$ und $\gamma = 0$ gewählt. Es soll somit eine Steuertrajektorie gefunden werden, die von der maximalen Reaktortemperatur einen 3σ -Sicherheitsabstand einhält. Die

4. Anwendung robuster Prozessführungsverfahren

Lösung des Optimierungsproblems erfolgte wiederum mittels nichtlinearer Optimierung mit dem Optimierer SNOPT. Hier kam ein Single-Shoot-Verfahren zum Einsatz, da die hohe Anzahl an Übergangsbedingungen pro Schnitt den Vorteil des Multiple-Shootings aufwiegt.

Das Modell hat sechs Zustände. Bei 11 unsicheren Parametern sind 23 Simulationen des Modells zur Berechnung der statistischen Momente notwendig. Das ergibt pro Schnitt $6 \cdot 23$ Gleichungsnebenbedingungen. Schneidet man bei jedem Stellgrößenumschaltzeitpunkt ergeben sich 73 Schnitte und somit in Summe 10074 zusätzliche Freiheitsgrade und Gleichungsnebenbedingungen. Aufgrund der Größe der dann im Rahmen der Optimierung zu verarbeitenden Matrizen ergeben sich somit massive numerische und speichertechnische Probleme.

Das Ergebnis der Optimierung ist in Abbildung 4.16 dargestellt. Die durchgezogene Linie markiert den Mittelwert, die gestrichelte Linie wieder die aus zwei statistischen Momenten resultierende 3σ -Grenze. Man erkennt leicht, dass der Prozesslauf jetzt so geplant ist, dass die obere 3σ -Grenze der Verteilung der Reaktortemperatur entlang der kritischen Reaktortemperatur von 405K verläuft. Dies wird zum einen dadurch erreicht, dass die Dosierung des Parachlornitrobenzol zu Beginn des Batches etwas zeitiger herabgesetzt wird. Zum anderen wird im Folgenden der Prozess so geführt, dass die Varianz der einzelnen Stoffmengen gering gehalten wird.

Im Bereich um die 2. Stunde findet sich hier wieder ein Phänomen, welches schon bei der robusten Trajektorienplanung des biotechnologischen Beispielprozesses zu finden war. Hier wird zweimal die Dosierung des Parachlornitrobenzol so stark herabgesetzt, dass dieses fast vollständig verbraucht wird. Das dargestellte Ergebnis zeigt dabei deutlich, dass der Mittelwert der erwarteten Reaktortemperatur keinen festen, sondern einen sich aus den zu erwartenden Unsicherheiten ergebenden Sicherheitsabstand zur kritischen Reaktortemperatur einhält. Dies stellt letztlich auch den größten Unterschied zum ingenieurmäßigen Ansatz dar, bei dem heuristisch eine feste Reserve eingeplant wird.

Die Auswirkungen der robust geplanten Steuertrajektorie auf den umgesetzten Prozessverlauf wurden ebenfalls im Rahmen einer Monte-Carlo-Simulation untersucht. Um dabei einen direkten Vergleich mit der klassischen Trajektorie zu ermöglichen, basierten die Simulationen auf exakt denselben 10000 Samples aus der vorhergehenden Studie.

Abbildung 4.17 zeigt die Verteilung der maximalen Reaktortemperatur in blau im Vergleich zu den schon in Abbildung 4.14 dargestellten Verteilung bei Umsetzung der Trajektorienplanung in schwarz. Während die Trajektorienplanung eine stark ausgeprägte bimodale Verteilung zeigt, liefert die robuste Planung beinahe eine unimodale Verteilung.

Ein Großteil der Realisierungen (ca. 95 %) verletzt die obere Grenze der Reaktortemperatur nicht. Geht man auch hier wieder davon aus, dass bei 410 K eine Alarmierung

4.2. Sulfonierung von Parachlornitrobenzol

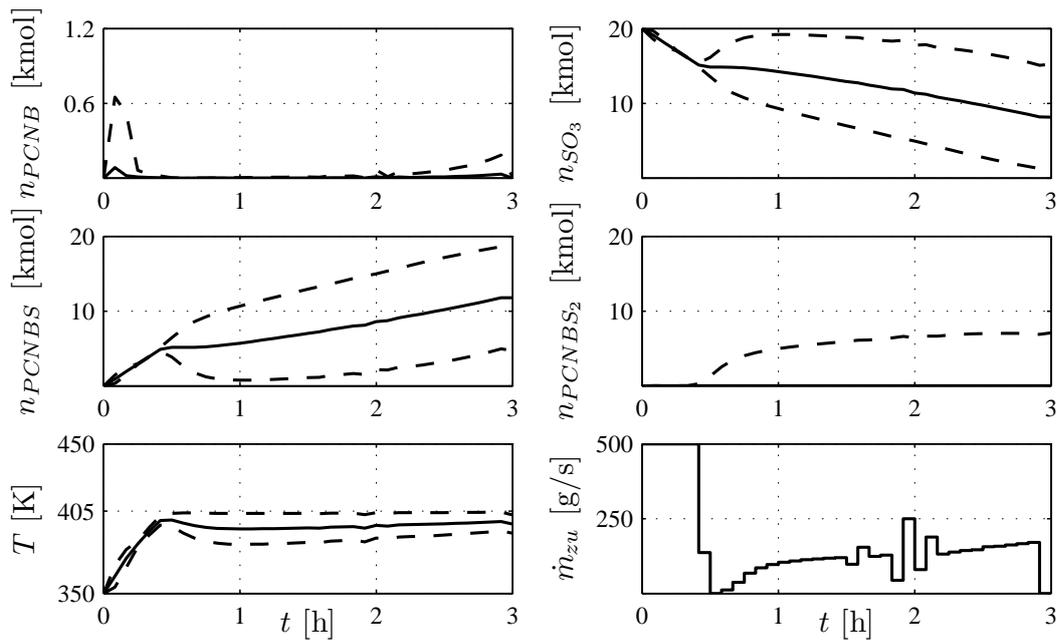


Abbildung 4.16.: Ergebnis der robusten Trajektorienplanung bei Approximation der Unsicherheiten mit der Unscented-Transformation. Die durchgezogenen Linien stellen den approximierten Mittelwert des Prozessverlaufes, die gestrichelten Linien die 3σ -Grenzen dar.

stattfindet, so wäre das nur in 2% der Realisierungen der Fall. Keine der 10000 Realisierungen verletzt die Grenze um mehr als 10 K, womit das primäre Ziel eines sicheren Prozesses gewährleistet ist. Die Produktivität liegt bei 73% bezogen auf den mit der klassischen Trajektorienplanung berechneten Wert und damit sogar geringfügig höher als die Umsetzung der klassischen Trajektorie erbracht hat.

4. Anwendung robuster Prozessführungsverfahren

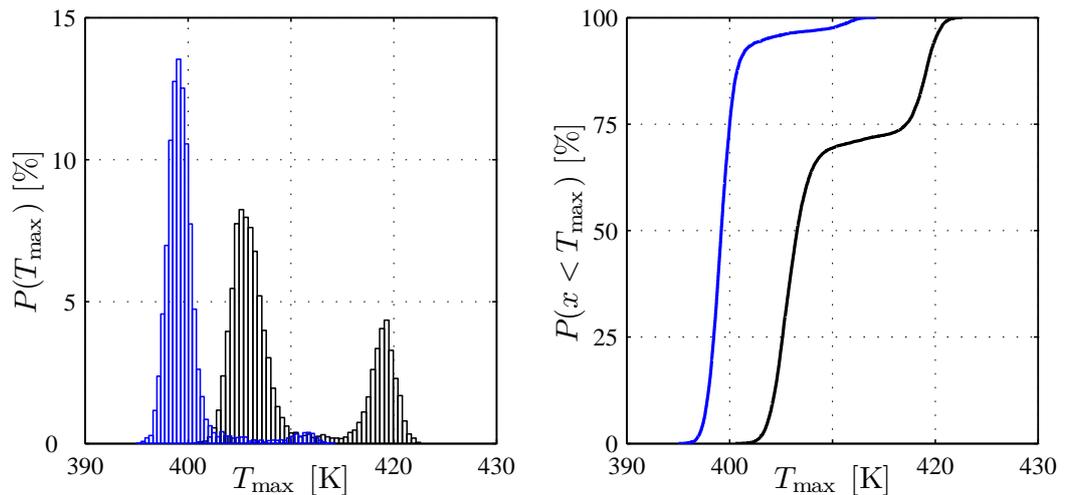


Abbildung 4.17.: Histogramm (links) und kumulative Wahrscheinlichkeitsverteilung der maximal im Laufe des Prozesses auftretenden Reaktortemperatur bei jeweils 10000 simulierten Realisierungen der klassischen Trajektorie (schwarz) und der robust geplanten Trajektorie (blau).

4.2.3. Nichtlineare-Modellprädiktive-Folgeregelung

Obwohl die robust geplante Steuerung eine hohe Sicherheit für den Prozessverlauf bietet, sorgt zum einen der hohe Abstand zu sicherheitsrelevanten Grenzen für eine relativ geringe Produktivität, zum anderen wird insbesondere bei solch kritischen Prozessen eine Regelung bevorzugt. Eines der mächtigsten Regelungsverfahren ist die Nichtlineare-Modellprädiktive-Folgeregelung. Diese soll hier in einer Simulationsstudie mit der reinen Steuerung des Prozesses verglichen werden.

Als Grundlage für den Vergleich dient folgendes Szenario. Ziel der Regelung ist es, der klassisch geplanten Trajektorie zu folgen, wobei der Fokus auf der Produkt- und Energietrajektorie liegt. Die Regelung greift alle 300s ohne Totzeit ($T_C = 0$) ein. Die Stellgrößentrajektorie ist wie bei der Planung als stückweise konstant mit einer Abtastzeit von ebenfalls 300s vorgesehen. Der Systemzustand wird von einem idealen Zustandsschätzer rekonstruiert. Diese Annahme der idealen Zustandsschätzung wird hier verwendet, um ausschließlich die Auswirkungen der Regelungsverfahren und nicht eventueller Schätzfehler zu überprüfen. Obwohl es eher unüblich ist, zeigten Screeningversuche, dass es in diesem Fall sinnvoll ist den Prädiktionshorizont T_P und den Stellhorizont T_S identisch zu wählen. Hier werden $T_P = T_S = 1200$ s verwendet.

Die Nichtlineare-Modellprädiktive-Folgeregelung löst zu jedem Abtastschritt das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} (\Phi_{\text{NMPC}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \underline{\theta})) \\
 \text{mit: } \Phi_{\text{NMPC}} &= \int_{t_k}^{t_k+T_P} \frac{1}{20000^2} \left(n_{\text{PCNB}}(t|\hat{\mathbf{x}}_k) - n_{\text{PCNB}}^{(\text{TP})}(t) \right)^2 \\
 &\quad + \frac{1}{1000000^2} \left(U(t|\hat{\mathbf{x}}_k) - U^{(\text{TP})}(t) \right)^2 dt \\
 \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \underline{\mathbf{u}}(t), \underline{\theta}), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_k + T_P \\
 0 &\leq \underline{\mathbf{u}}(t) \leq 500 \text{g/s} \\
 r(\mathbf{x}(t)) &\leq 0 \\
 r &= T(t) - 405 \text{ K}.
 \end{aligned} \tag{4.14}$$

Für diese Regelung wird ebenfalls eine Monte-Carlo-Simulation durchgeführt. Ausgehend von einem relativen Fehler von 2% in den Modellparametern $\alpha, D, E_1, E_2, E_3, k_{10}, k_{20}, k_{30}, (-\Delta H_R)_1, (-\Delta H_R)_2$ und $(-\Delta H_R)_3$ wird für jede Applikation des Reglers ein Parametersatz aus der statistischen Verteilung dieser Parameter gezogen. Das mathematische Modell mit diesem Parametersatz simuliert den wahren gestörten Prozess. Das Regelungsverfahren arbeitet immer mit dem nominellen Parametersatz.

Abbildung 4.18 zeigt eine Auswertung von 1000 geregelten Semibatch-Prozessen. Dargestellt sind die kumulativen Wahrscheinlichkeitsverteilungen der maximal im Batch auftretenden Temperatur. Als gestrichelte Linien eingezeichnet finden sich nochmals die Verteilungen bei der gesteuerten Umsetzung der klassischen Trajektorie (schwarz) und der robust geplanten Trajektorie (blau). Die schwarze, durchgezogene Linie markiert das Ergebnis dieser Simulationsstudie. Es zeigt sich, dass aufgrund der Regelung sehr viele Umsetzungen nahe an der kritischen Temperatur ($T_{\text{krit}} = 405 \text{ K}$) liegen. Nimmt man wieder an, dass die erste Alarmgrenze bei 410 K liegt, dann wird diese immer noch mit einer Wahrscheinlichkeit von 12% erreicht. Etwa 8% der Realisierungen überschreiten auch jetzt noch die maximal zulässige Reaktortemperatur um mehr als 10 K, was wie bereits beschrieben, in diesem Szenario den Eingriff sicherheitstechnischer Einrichtungen auslösen soll. Die Produktivität der klassischen geregelten Umsetzung beträgt 88%. Kritischer ist jedoch immernoch die hohe Wahrscheinlichkeit des Auslösens sicherheitsgerichteter Schaltungen oder gar eines Störfalls einzuschätzen.

Der einfachste Ansatz, um diese kritischen Temperaturüberschreitungen zu verhindern, ist üblicherweise, eine Trajektorie mit einer geringeren maximal zulässigen Temperatur zu planen und entlang dieser zu regeln. In einer zweiten Simulationsstudie wird daher der in (4.14) angegebene Regler benutzt, um entlang einer mit $T_{\text{krit}} = 400 \text{ K}$ geplanten Trajektorie zu regeln. Das Ergebnis von ebenfalls 1000 geregelten Umsetzungen ist als schwarze Strichpunktlinie in Abbildung 4.18 dargestellt. Man erkennt viele Realisierungen, die in der Nähe von einer maximalen Reaktortemperatur von 400 K liegen. Dennoch

4. Anwendung robuster Prozessführungsverfahren

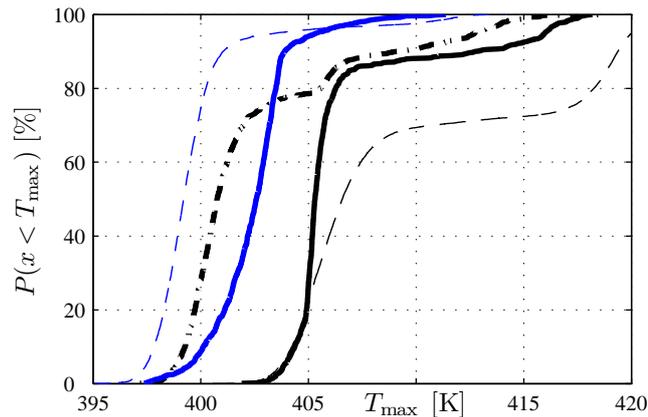


Abbildung 4.18.: Kumulative Wahrscheinlichkeitsverteilung der maximal im Laufe eines Prozesses auftretenden Reaktortemperatur.
gestrichelte Linien - klassische Trajektorienplanung (schwarz) und robuste Trajektorienplanung (blau),
durchgezogene Linien - klassische Nichtlineare-Modellprädiktive-Folgeregelung (schwarz) und robuste Nichtlineare-Modellprädiktive-Folgeregelung (blau),
strichgepunktete Linie - klassische Nichtlineare-Modellprädiktive-Folgeregelung mit verringerter maximaler Reaktortemperatur.

erreichen noch etwa 20% der Umsetzungen eine Temperatur von über 405 K. Hier findet sich ein Knick in der Wahrscheinlichkeitsverteilung, da aufgrund der Beschränkung $r = T(t) - 405$ K innerhalb der Regelung, ein massiveres Gegensteuern des Reglers gegen eine weitere Temperaturerhöhung stattfindet. In 9% der Fälle wird noch die Alarmgrenze erreicht, in 1% aller Umsetzungen löst eine sicherheitsgerichtete Schaltung aus. Die Produktivität sinkt auf 81% der nominellen Planung.

4.2.4. Robuste Nichtlineare-Modellprädiktive-Folgeregelung

Eine weitere Verringerung der Alarme ist durch Absenken der maximal zulässigen Temperatur innerhalb des klassischen Regelungsverfahrens zu erreichen. Die Produktivität würde damit aber noch weiter sinken. Eine Alternative bildet hier die robuste Nichtlineare-Modellprädiktive-Folgeregelung. Anstatt einen festen Abstand zu einer Prozessgrenze einzuhalten, wird hier ein sich aus Modell- und Messunsicherheiten ergebender Sicherheitsabstand gewählt. So kann die Prozesstrajektorie online so nachgeplant werden, dass die Chance einer Grenzverletzung minimal ist.

Die robuste Nichtlineare-Modellprädiktive-Folgeregelung löst zu jedem Abtastschritt das Optimierungsproblem

$$\begin{aligned}
 \mathbf{U}_k^* &= \arg \min_{\mathbf{U}_k} \left(\bar{\Phi}_{\text{NMPC}}(\hat{\mathbf{x}}_k, \mathbf{U}_k, \underline{\theta}) + \gamma \sigma_{\bar{\Phi}_{\text{NMPC}}}^2 \right) \\
 \text{mit: } \bar{\Phi}_{\text{NMPC}} &= \int_{t_k}^{t_k+T_P} \frac{1}{20000^2} \left(n_{\text{PCNB}}(t|\hat{\mathbf{x}}_k) - n_{\text{PCNB}}^{(\text{TP})}(t) \right)^2 \\
 &\quad + \frac{1}{1000000^2} \left(U(t|\hat{\mathbf{x}}_k) - U^{(\text{TP})}(t) \right)^2 dt \\
 \dot{\mathbf{x}} &= \underline{f}(t, \mathbf{x}(t), \underline{u}(t), \underline{\theta}), \quad \mathbf{x}(t = t_k) = \hat{\mathbf{x}}_k \\
 t_k &\leq t \leq t_k + T_P \\
 0 &\leq \underline{u}(t) \leq 500 \text{g/s} \\
 \bar{r}(\mathbf{x}(t)) + \lambda \sigma_r &\leq 0 \\
 r &= T(t) - 405 \text{ K}.
 \end{aligned} \tag{4.15}$$

Die Einstellungen der Regelung entsprechen denen der klassischen Nichtlinearen-Modellprädiktiven-Folgeregelung. Da wie bereits bei der Planung einer robusten Prozesstrajektorie erwähnt, der Fokus der robusten Regelung auf der Verhinderung von Grenzüberschreitungen liegt, wird hier ebenfalls $\gamma = 0$ gewählt. Da die Regelung regelmäßig in den Prozess eingreift, ist kein so großer Sicherheitsabstand wie bei der Planung einer reinen Steuerung notwendig. Daher wird in diesem Fall $\lambda = 1$ gewählt.

Das Ergebnis von 1000 simulativen Umsetzungen der robusten Nichtlinearen-Modellprädiktiven-Folgeregelung unter den gleichen Voraussetzungen, wie im letzten Abschnitt für die klassische Nichtlineare-Modellprädiktive-Folgeregelung beschrieben, ist in Abbildung 4.18 in blau wiedergegeben. Es zeigt sich, dass etwa 91% aller Umsetzungen eine maximale Reaktortemperatur von weniger als 405 K erreichen. In nur 0,4% der Fälle wird überhaupt eine Meldung ausgelöst. Keine Realisierung überschreitet die Temperaturgrenze um mehr als 10 K.

Trotz der in den meisten Fällen leicht unter 405 K liegenden Reaktortemperatur liefert die robuste Nichtlineare-Modellprädiktive-Folgeregelung eine Produktivität von 82%. In Verbindung mit der extrem verringerten Wahrscheinlichkeit einer Grenzverletzung zeigt sich das Potential der robusten Regelung gegenüber dem ingenieurmäßigen Ansatz, die Temperaturgrenze um einen festen Betrag zu verringern.

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

In diesem Abschnitt sollen die Verfahren zur robusten Planung und -führung von Batchprozessen auf die Produktion des Antibiotikums Nikkomycin durch den Stamm *Streptomyces tendae* angewendet werden. Ausgangspunkt der Betrachtungen ist ein dynamisches mittelstrukturiertes Kompartimentmodell, welches das Wachstums- und Produktionsverhalten des Stammes *Streptomyces tendae* beschreibt. Das Modell wurde von [Kin97] entwickelt, später von [Maj98] erweitert und sowohl von [Maj98] als auch von [Hei04] bereits zur modellgestützten Prozessführung verwendet.

Im Rahmen der Erweiterung des Modells ergänzte [Maj98] die vorhandenen Versuchsdaten um einen optimal geplanten Versuchslauf und führte Untersuchungen zu den zu erwartenden Parameterfehlern durch. Das Modell und die von [Maj98] errechneten verbleibenden Parameterfehler sind in Anhang C angegeben. Hier ist jedoch zu erwähnen, dass die von [Maj98] verwendete Methodik zur Berechnung der verbleibenden Parameterfehler einige Punkte enthält, die zu sehr optimistischen Resultaten führen. So teilt er die Parameter in verschiedene Gruppen ein und ignoriert bei der Abschätzung der verbleibenden Parameterfehler einer Gruppe die Parameterfehler der anderen Gruppen. Als Folge sind auch keine Kreuzkovarianzen zwischen den Parametern berechnet. Die Versuchsdaten der verwendeten Identifikationsversuche sind jedoch nicht mehr vollständig verfügbar, weswegen im Rahmen dieser Arbeit auf die Ergebnisse von [Maj98] zurückgegriffen werden muss.

4.3.1. Simulationsstudie

Ausgangspunkt der Untersuchungen ist zunächst eine klassisch geplante optimale Prozessstrajektorie, wie sie auch in den Arbeiten von [Hei04] errechnet wurde. Da es sich bei dem Produkt Nikkomycin um ein im Vergleich zu den benötigten Einsatzstoffen sehr teures Produkt handelt, ist das Ziel der Trajektorienplanung ausschließlich eine maximale Menge an Nikkomycin (m_{Ni}) bis zum Ende des Prozesslaufes bei $t = 100$ h zu produzieren.

Dabei dürfen die Konzentrationen der Nährmedien Ammonium (c_{NH_4}), Phosphat (c_{PO_4}) und Glukose (c_C) nicht zu hoch sein, um wachstums-/produktionshemmende oder sogar toxische Auswirkungen auf die Kultur zu vermeiden. Zusätzlich ist eine zu geringe Konzentration von Glukose (c_C) nicht zulässig, da dann das Modell seine Gültigkeit verliert. Desweiteren gilt für die Konzentration der Biotrockenmasse (c_X) ein maximaler Wert von 17 g/l, um eine ausreichende Versorgung der Organismen mit Sauerstoff sicher zu stellen.

Die Kultivierung erfolgt in 15l-Fermentern der Firma Sartorius. Als maximal zulässige Arbeitsvolumen wird 10l festgelegt. Zum Start der Fermentation ist der Reaktor mit 10l befüllt, aufgrund der Probenentnahmen sinkt das Volumen jedoch über die Zeit, so dass

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

ausreichend Volumen für eine Nährstoffdosierung vorhanden ist. Die initiale Biotrockenmasse $m_{X,0}$ beträgt 1,5 g, die sich, wie in Tabelle C.3 angegeben aus den verschiedenen Kompartimenten zusammensetzt. Als Freiheitsgrade der Optimierung stehen die initial im Reaktor vorgelegten Mengen der oben genannten Nährmedien, sowie die Zudosierprofile dieser Nährstoffe zur Verfügung.

Aus [Hei04] ist ersichtlich, dass die Menge der initial vorliegenden Nährmedien und die Parametrierung der Stellgrößen keinen wesentlichen Einfluss auf das Optimierungsergebnis haben. Daher werden diese im Rahmen der Optimierung nicht variiert, sondern fest auf $m_{\text{NH}_4} = 8$ g, $m_{\text{PO}_4} = 3$ g und $m_C = 399,9$ g gesetzt und stückweise konstante Zudosierprofile verwendet.

Die Nährmedien in den Fütterströmen liegen in den Konzentrationen $c_{\text{NH}_4, \text{Feed}} = 14$ g/l, $c_{\text{PO}_4, \text{Feed}} = 24$ g/l und $c_{C, \text{Feed}} = 450$ g/l vor. Das Optimierungsproblem der Trajektorienplanung ergibt sich somit zu

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} (\Phi_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})) \\
 \Phi_{\text{TP}} &= -m_{\text{Ni}}(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u_{\text{NH}_4}(t) \leq 0,11/\text{h} \\
 0 &\leq u_{\text{PO}_4}(t) \leq 0,11/\text{h} \\
 0 &\leq u_C(t) \leq 0,11/\text{h} \\
 r(\underline{x}(t)) &\leq \underline{0} \\
 r_1 &= c_X(t) - 17 \text{ g/l}, \quad r_2 = c_{\text{NH}_4}(t) - 1,4 \text{ g/l} \\
 r_3 &= c_{\text{PO}_4}(t) - 1 \text{ g/l}, \quad r_4 = c_C(t) - 40 \text{ g/l} \\
 r_5 &= -c_C(t) + 10 \text{ g/l}
 \end{aligned} \tag{4.16}$$

Die Lösung des Optimierungsproblems wurde mittels Nichtlinearer-Optimierung mit einem Multiple-Shooting-Verfahren berechnet. Das Ergebnis ist in Abbildung 4.19 gezeigt.

Die Fermentation ist, wie beim Simulationsbeispiel in Abschnitt 4.1, in zwei Phasen geteilt. Zunächst wird mit hohen Nährmedienkonzentrationen exponentielles Wachstum der Organismen angeregt. Im zweiten Teil der Fermentation resultiert aus einer gezielten Phosphatlimitation die Aktivierung des Sekundärstoffwechsels mit der Produktion von Nikkomycin.

Nimmt man nun an, dass die Modellparameter unsicher sind und die statistischen Eigenschaften denen von [Maj98] beschriebenen Werten entsprechen, kann wie in den gezeigten Beispielmotellen die zu erwartende Unsicherheit des Prozessverlaufes mit Hilfe einer Monte-Carlo-Simulation ermittelt werden. Abbildung 4.20 zeigt das Ergebnis einer solchen Studie mit 10000 Einzelsimulationen. Die Aufteilung der Abbildung ist identisch

4. Anwendung robuster Prozessführungsverfahren

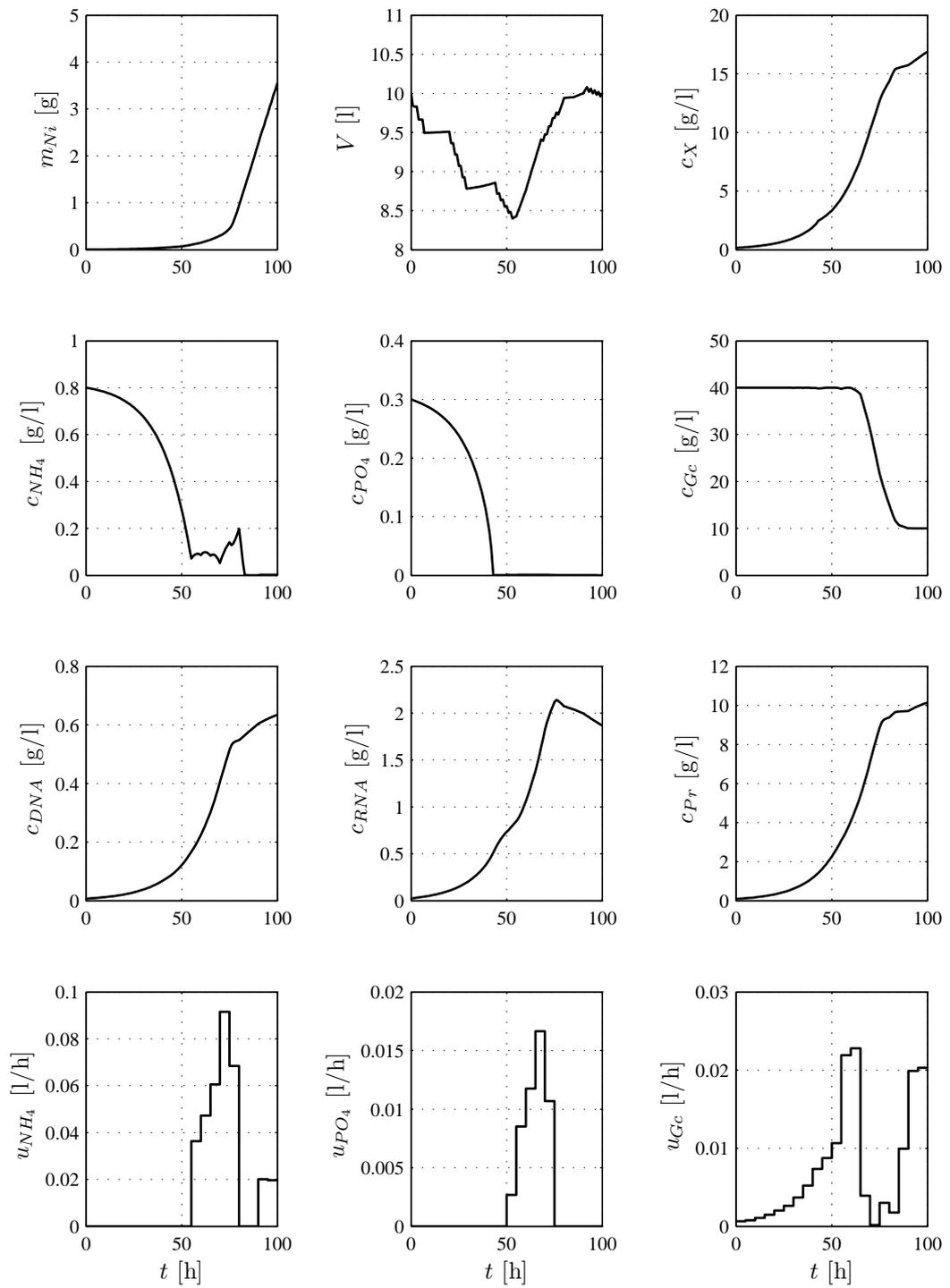


Abbildung 4.19.: Verlauf der mit (4.16) geplanten Trajektorie.

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

zu Abbildung 4.19. Die durchgezogene schwarze Linie markiert den aus der Monte-Carlo-Simulation ermittelten statistischen Mittelwert der Prädiktion. Die gestrichelten Linien markieren die 3σ -Grenze ermittelt aus der Auswertung der ersten zwei statistischen Momente.

Schon die geringen angenommenen Parameterfehler haben signifikante Auswirkungen. So ergeben sich insbesondere in der vorhergesagten Masse des Kompartiments DNA massive Unsicherheiten, die sich direkt auf den Nährstoffverbrauch und die Nikkomycinproduktion niederschlagen. Die -3σ -Grenze der erwarteten Nikkomycinmenge ist sogar so gering, dass sie unter 0 liegt und somit in der Abbildung nicht mehr sichtbar ist.

Eine genauere Darstellung der zu erwartenden Nikkomycionproduktion ist in Abbildung 4.21 ersichtlich. Abbildung 4.21 zeigt ein Histogramm der in den Monte-Carlo-Simulation erreichten Produktionsmengen. Diese nähere Betrachtung der Verteilung der in der Monte-Carlo-Simulation erreichten Produktivität liefert ein ähnliches Bild, wie bei der Simulationsstudie des unstrukturierten Modells in Abschnitt 4.1 auftrat. Der statistische Mittelwert der Produktmenge ist 3,21 g im Gegensatz zu 3,56 g als Voraussage der reinen Simulation der klassisch geplanten Trajektorie.

Eine robuste Trajektorienplanung soll zunächst zeigen, ob es möglich ist, durch Variation der Nährmedienzugabe eine höhere Sicherheit bezüglich der Produktmenge zu erreichen. Das Optimierungsproblem der robuste Trajektorienplanung liest sich wie folgt.

$$\begin{aligned}
 \mathbf{U}^* &= \arg \min_{\mathbf{U}} (\bar{\Phi}_{\text{TP}}(\underline{x}_0, \mathbf{U}, \underline{\theta})) + \gamma \sigma_{\Phi_{\text{TP}}}^2 \\
 \Phi_{\text{TP}} &= -m_{\text{Ni}}(t_{\text{Ende}}) \\
 \text{mit: } \dot{\underline{x}} &= \underline{f}(t, \underline{x}(t), \underline{u}(t), \underline{\theta}), \quad \underline{x}(t=0) = \underline{x}_0 \\
 0 &\leq t \leq t_{\text{Ende}} \\
 0 &\leq u_{\text{NH}_4}(t) \leq 0,11/\text{h} \\
 0 &\leq u_{\text{PO}_4}(t) \leq 0,11/\text{h} \\
 0 &\leq u_{\text{C}}(t) \leq 0,11/\text{h} \\
 \bar{r}_i(\underline{x}(t)) + \lambda \sigma_{r_i} &\leq \underline{0}, \quad \text{mit } i = 1, \dots, 5 \\
 r_1 &= c_{\text{X}}(t) - 17 \text{ g/l}, \quad r_2 = c_{\text{NH}_4}(t) - 1,4 \text{ g/l} \\
 r_3 &= c_{\text{PO}_4}(t) - 1 \text{ g/l}, \quad r_4 = c_{\text{C}}(t) - 40 \text{ g/l} \\
 r_5 &= -c_{\text{C}}(t) + 10 \text{ g/l}
 \end{aligned} \tag{4.17}$$

Das Ergebnis der robusten Trajektorienplanung mit $\gamma = 0,5$ und $\lambda = 0$ unter Beachtung der in Tabelle C.1 angegebenen Unsicherheiten ist in Abbildung 4.22 in blau gezeigt. Die durchgezogenen Linien markieren den mit Hilfe der Unscented-Transformation approximierten Mittelwert, die gestrichelten Linien die 3σ -Grenzen. In schwarz ist zum Vergleich das Ergebnis einer mit den oben beschriebenen Randbedingungen durchgeführten Monte-Carlo-Simulation für die robust geplante Trajektorie mit den aus zwei statistischen Momenten resultierenden 3σ -Grenzen (gestrichelt) dargestellt.

4. Anwendung robuster Prozessführungsverfahren

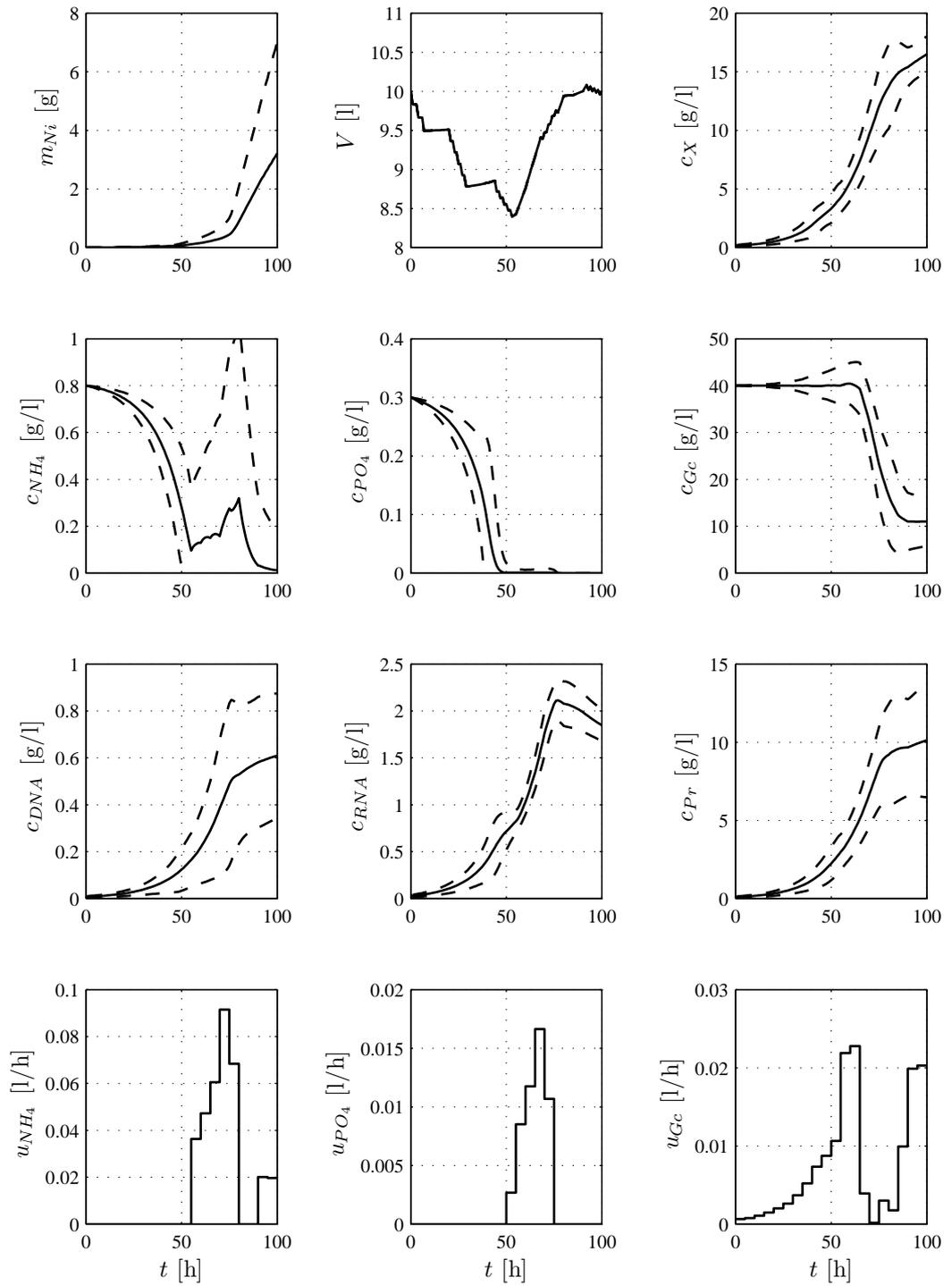


Abbildung 4.20.: Ergebnis der Monte-Carlo-Simulation der mit (4.16) geplanten Trajektorie, durchgezogene Linie - Mittelwert, gestrichelte Linien - 3σ Grenzen aus der Auswertung der ersten zwei statistischen Momente.

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

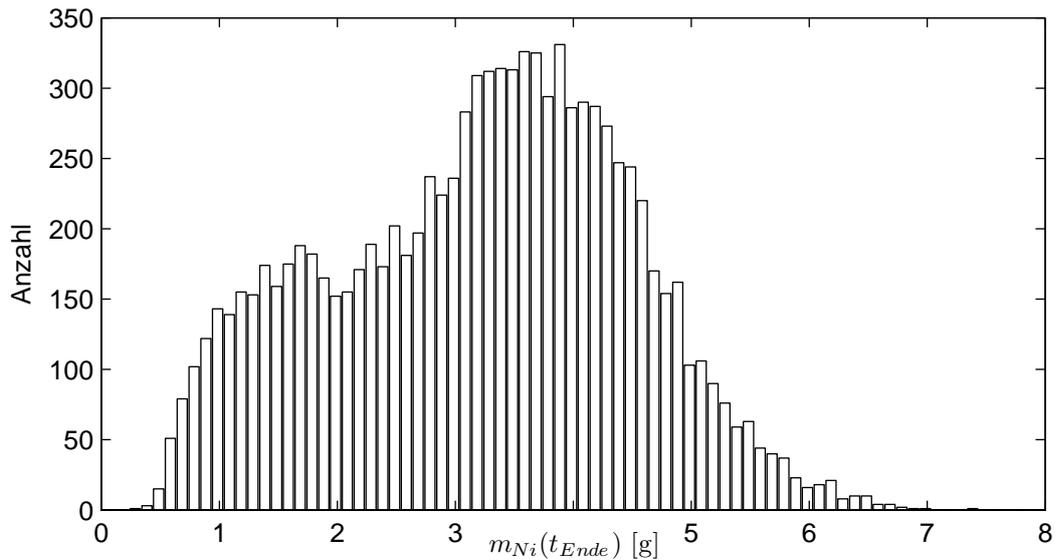


Abbildung 4.21.: Histogramm der in der Monte-Carlo-Simulation ermittelten Produktionsmenge für die klassisch geplante Trajektorie.

Zum einen zeigt sich die hohe Approximationsgenauigkeit der Unscented-Transformation. Diese liefert eine approximierte mittlere Produktionsmenge von 3,05 g was sehr nahe an dem mit der Monte-Carlo-Simulation ermittelten statistischen Mittelwert von 3,04 g liegt. Auch die 3σ -Grenzen sind mit Rücksicht auf die Komplexität des Modells von der Unscented-Transformation gut wiedergegeben.

Zum anderen ist in der Abbildung 4.22 die gegenüber der klassischen Trajektorienplanung (Abbildung 4.20) geringere Streuung insbesondere der Nikkomycinmenge am Ende der Fermentation zu erkennen. Die ist in Abbildung 4.23 nochmals als Histogramm genauer dargestellt. Die robuste Planung liefert somit eine im Mittel zwar etwas geringere Produktivität, aber eine signifikant höhere Sicherheit bezüglich des Resultates der durchzuführenden Fermentation.

Dieses Ergebnis ist äquivalent zu den Resultaten der Simulationsstudie am einfachen unstrukturierten Modell in Abschnitt 4.1. Dies zeigt, dass der vorgestellte robuste Prozessführungsansatz unter Verwendung der Unscented-Transformation auch auf komplexere Modelle anwendbar ist.

4. Anwendung robuster Prozessführungsverfahren

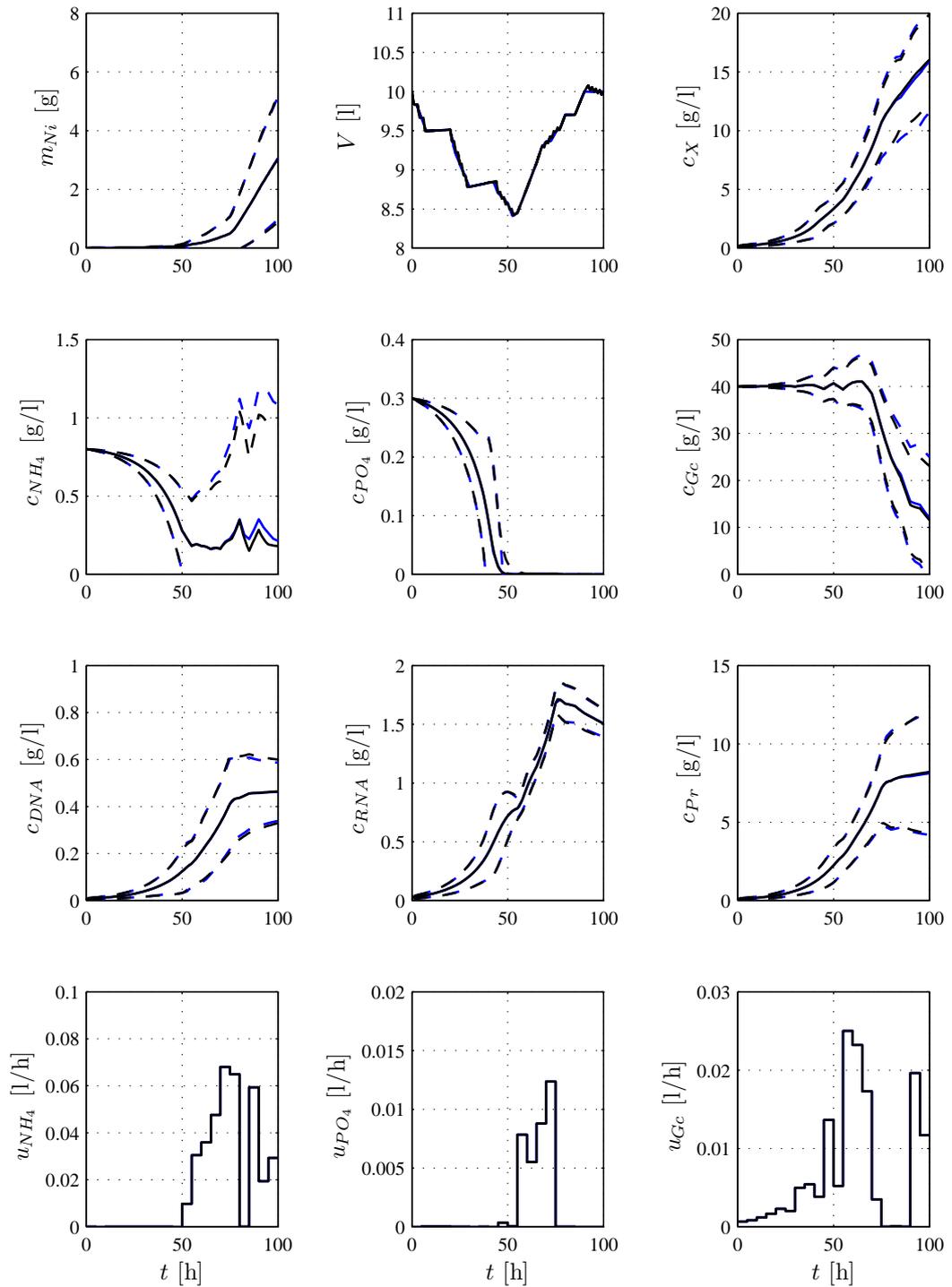


Abbildung 4.22.: Ergebnis der Monte-Carlo-Simulation der mit (4.17) geplanten Trajektorie, durchgezogene Linie - Mittelwert, gestrichelte Linien - 3σ Grenzen aus der Auswertung der ersten zwei statistischen Momente, blaue Ergebnis der Approximation mit der Unscented-Transformation, durchgezogene Linie - Mittelwert, gestrichelte Linien - 3σ Grenzen.

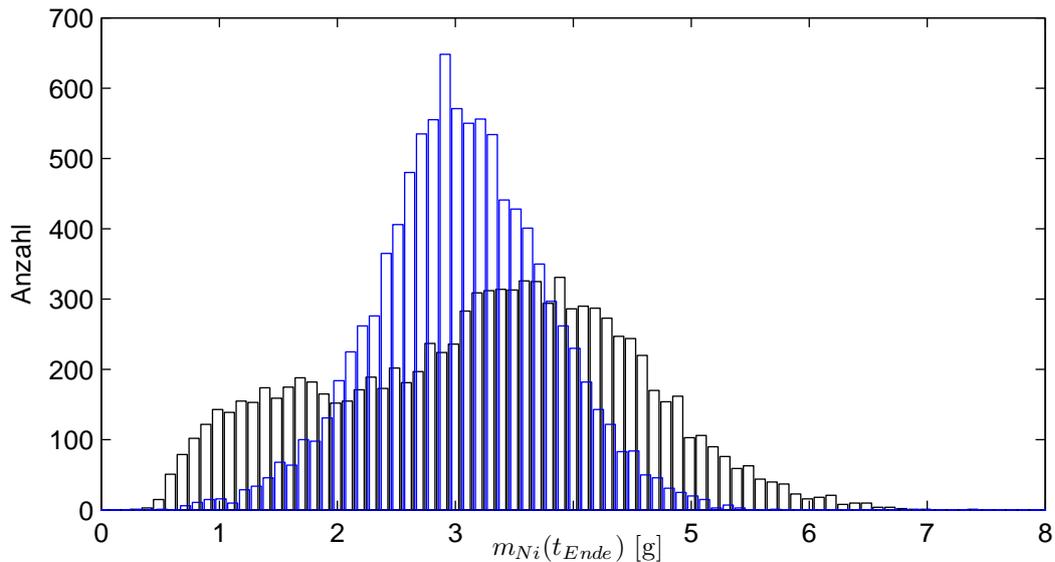


Abbildung 4.23.: Histogramm der in der Monte-Carlo-Simulation ermittelten Produktionsmenge für die klassisch geplante Trajektorie (schwarz) und die robust geplante Trajektorie (blau).

4.3.2. Umsetzung einer robusten Nichtlinearen-Modellprädiktiven-Folgeregelung und einer robusten Online-Trajektorienplanung

Neben der rein simulativen Überprüfung der Anwendbarkeit robuster Verfahren ist deren reale Applikation von entscheidender Bedeutung. Im Folgenden soll daher die Umsetzung robuster Regelungsverfahren zur Führung von Fermentationen des Stammes *Streptomyces tendae* gezeigt werden. In [Hei04, KHK07] ist die Prozessführung solcher Fermentationen mit der klassischen Nichtlinearen-Modellprädiktiven-Folgeregelung und der Online-Trajektorienplanung gezeigt.

Zustandsschätzung

Eine wichtige Grundlage für die Applikation modellgestützter Regelungsverfahren ist die möglichst genaue Rekonstruktion des Modellzustandes anhand der wenigen online verfügbaren Messgrößen. Hierfür werden üblicherweise Zustandsschätzverfahren, wie das Extended-Kalman-Filter [KB61] oder die Moving-Horizon-State-Estimation [MR94] verwendet. Die Anwendung und Einstellung dieser Verfahren für Fermentationen des Stammes *Streptomyces tendae* ist in [Hei04, KHK07] gezeigt.

Neben der möglichst guten Wiedergabe des Prozesszustandes ist bei der Verwendung robuster Prozessführungsverfahren auch eine möglichst genaue Approximation der Ko-

4. Anwendung robuster Prozessführungsverfahren

varianzmatrix \mathbf{C}_x des Prozesszustandes nötig. Die o.g. Verfahren basieren üblicherweise auf einer Approximation erster Ordnung. In [JUDW95, NPR00b] und [Mer04] werden Kalman Filter vorgestellt, die die Unscented-Transformation zur Approximation der benötigten statistischen Eigenschaften verwenden und somit eine höhere Approximationsgenauigkeit versprechen. Hier kommt daher ein Unscented-Kalman-Filter in Central-Difference-Form nach [Mer04] mit Verwendung des gleichen Approximationsansatzes wie in Abschnitt 2.1.3 beschrieben zur Anwendung.

Die Einstellungen des Unscented-Kalman-Filters sind in den Tabellen 4.3 und 4.4 angegeben. Für die Kovarianzmatrix des Parameterfehlers werden die Daten aus Tabelle C.1 verwendet. Die Unsicherheiten des Startzustandes basieren auf Annahmen über die

Tabelle 4.3.: Einstellungen UKF

Zustand	Standardabweichung	
	Startzustand (\underline{x}_0)	Systemrauschen ($\dot{\xi}$)
m_{DNA}	0,016875 g	0,0080791 g/h
m_{RNA}	0,05625 g	0,0010131 g/h
m_{Pr}	0,225 g	0,0015882 g/h
m_{Aa}	0,015 g	0,0015524 g/h
m_{Nu}	0,015 g	0,00043833 g/h
m_{SE}	0,046875 g	0,00088497 g/h
Φ	0,031623	0,0023674 1/h
m_{NH_4}	0,5 g	0,0016777 g/h
m_{PO_4}	0,25 g	0,0075 g/h
m_{Gc}	2 g	0,0037568 g/h
m_{Ni}	0,031623 g	0,0010474 g/h
V_{Base}	0,01 l	0,00031623 l/h

Genauigkeiten der Einwaage der verschiedenen Medien und der Unsicherheit der Masse des Inoculums. Die Elemente des Systemrauschens ergeben sich aus einer Optimierung der Schätzgüte, wie sie in [Hei04] beschrieben ist. Das Messrauschen ist [HKK03] entnommen.

Technische Umsetzung der robusten Regelungsverfahren

Die Kultivierung des Stammes *Streptomyces tendae* erfolgt in 15l-Fermentern. Um einen möglichst vollautomatischen Betrieb der bis zu 100 h laufenden Versuche zu ermöglichen, wurde ein modulares Prozessleitsystem aufgebaut, welches die Anwendung moderner Regelungsverfahren ermöglicht. Die Struktur des verwendeten modularen Prozessleitsystems kann [Hei04] entnommen werden.

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

Tabelle 4.4.: Einstellungen Messwerte UKF

Messgröße	Standardabweichung Messrauschen (η)
c_X	$\left(\frac{0,25}{12}c_X + 0,05\right)$ g/l
c_{NH_4}	$\left(\frac{0,015}{2}c_{NH_4} + 0,003\right)$ g/l
c_{PO_4}	$\left(\frac{0,011}{0,6}c_{PO_4} + 0,007\right)$ g/l
c_{Gc}	$\left(\frac{0,5}{40}c_{Gc} + 0,25\right)$ g/l
c_{Ni}	0,05 g/l
c_{DNA}	$\left(\frac{0,024}{12}c_X + 0,006\right)$ g/l
c_{RNA}	0,04 g/l
c_{Pr}	$\left(\frac{0,12}{14}c_X + 0,02\right)$ g/l
V_{Base}	$\max(1; 0,02 \cdot V_{Base})$ l

Für die Lösung des robusten Optimierungsproblems ist allerdings ein sehr hoher Rechenaufwand zu bewältigen. Bei Annahme von Unsicherheiten in 13 Modellzuständen - als Resultat der Zustandsschätzung - und in allen 46 Modellparametern sind zur Berechnung einer Prädiktion mit Unsicherheiten 119 einzelne Simulationen des Systemmodells nötig. Somit würde die Optimierung ohne Erweiterungen der Rechenleistung mindestens die 119-fache Rechenzeit der klassischen Regelungsverfahren in Anspruch nehmen und somit online nicht umsetzbar sein.

Aus dem genannten Grund ist es notwendig, die einzelnen Simulationen auf eine wählbare Anzahl von MatlabTM Instanzen zu verteilen. Da jede MatlabTM Instanz strikt „Single Threaded“ ist, lässt sich durch die Verwendung mehrerer Instanzen die Vielfach-Kern Technologie aktueller Prozessoren nutzen. Die Hauptinstanz von MatlabTM öffnet dazu über COM (Component Object Model) [Ste98] so viele zusätzliche Prozesse, wie Prozessorkerne zur Verfügung stehen und verteilt die durchzuführenden Simulationen auf die verschiedenen MatlabTM Prozesse.

Auch mit der beschriebenen Erweiterung sind bei der verfügbaren Hardware nur vier parallele Prozesse möglich. Die COM Technologie ermöglicht über DCOM (Distributed COM) allerdings auch Prozesse auf anderen Rechnern über das lokale Netzwerk zu starten und fernzusteuern. Im Rahmen der Implementierung eines verteilten Rechensystems ist daher auch dies in die Erweiterung eingeflossen.

Mit der beschriebenen Implementierung ist eine robuste Optimierung im online notwendigen Zeitrahmen möglich. Insgesamt waren an der Berechnung letztlich 5 Computer mit insgesamt 12 Prozessorkernen beteiligt.

Umsetzung einer robusten Nichtlinearen-Modellprädiktiven-Folgeregelung

Abbildung 4.24 zeigt das Ergebnis einer an einer realen Anlage umgesetzten robusten Nichtlinearen-Modellprädiktiven-Folgeregelung. Die Grundlage dieser robusten Nichtlinearen-Modellprädiktiven-Folgeregelung bildet die oben beschriebene klassisch geplante Trajektorie. Das Gütefunktional der Nichtlinearen-Modellprädiktiven-Folgeregelung für die Optimierung wurde direkt aus [Hei04] übernommen. Die Unsicherheiten der Prädiktion sind mit $\gamma = 0,01(t_i - t_k)$ bewertet.

Die Regelung ist gut in der Lage, der offline geplanten Trajektorie (schwarz) zu folgen. Wie in [Hei04] beschrieben, nutzt auch die robuste Nichtlineare-Modellprädiktive-Folgeregelung die verhältnismäßig geringe Gewichtung der Abweichungen der Ammoniumkonzentration, um mit Abweichungen bei der Ammoniummenge die anderen Regelungsziele zu erreichen.

Nach 70 Stunden Batchlaufzeit divergieren die als Softsensor genutzte zudosierte Basemenge und die atline bestimmte Biotrockenmasse aufgrund von Problemen der pH-Regelung des Reaktors. Dies führt zu Problemen bei der Zustandsschätzung die nur durch manuellen Eingriff in die Schätzung gelöst werden können. Die Auswirkungen sind in starken Sprüngen insbesondere bei den Schätzungen der internen Kompartimente DNA, RNA und Proteine ersichtlich. Als Folge der manuellen Korrekturen entspricht die geschätzte Basemenge nicht mehr der Messung, die weiteren Systemzustände sind jedoch noch plausibel, so dass die Regelung erfolgreich bis zu Ende laufen kann.

Umsetzung einer robusten Online-Trajektorienplanung

Die bei der Umsetzung der robusten Nichtlinearen-Modellprädiktiven-Folgeregelung auftretenden Probleme bei der Zustandsschätzung sind in ähnlicher Form auch in [Hei04] beim Einsatz des Extended-Kalman-Filter beschrieben. Dort und in [KHK07] wurde gezeigt, dass ein Constrained-Extended-Kalman-Filter, welches beim Measurement-Update ein beschränktes Maximum-a-Posteriori-Problem löst, die Genauigkeit der Zustandsschätzung erhöht.

Für die reale Umsetzung einer robusten Online-Trajektorienplanung kommt daher ein Constrained-Unscented-Kalman-Filter zum Einsatz. Das hier umgesetzte Verfahren nutzt abweichend von [Mer04] für das Update des Zustandes die gleiche Implementierung, die in [KHK07] für das Constrained-Extended-Kalman-Filter angegeben ist. Ohne weitere Herleitung sind die Gleichungen dieses Constrained-Unscented-Kalman-Filters im Anhang B angegeben. Die biologisch begründeten Beschränkungen für die Zustandsschätzung werden aus [Hei04] übernommen.

Die Grundlage der robusten Online-Trajektorienplanung bildet die oben beschriebene robust geplante Trajektorie. Für die online durchgeführte Optimierung kommt das gleiche Gütefunktional wie bei der robusten Trajektorienplanung zum Einsatz. Die Abbildung

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

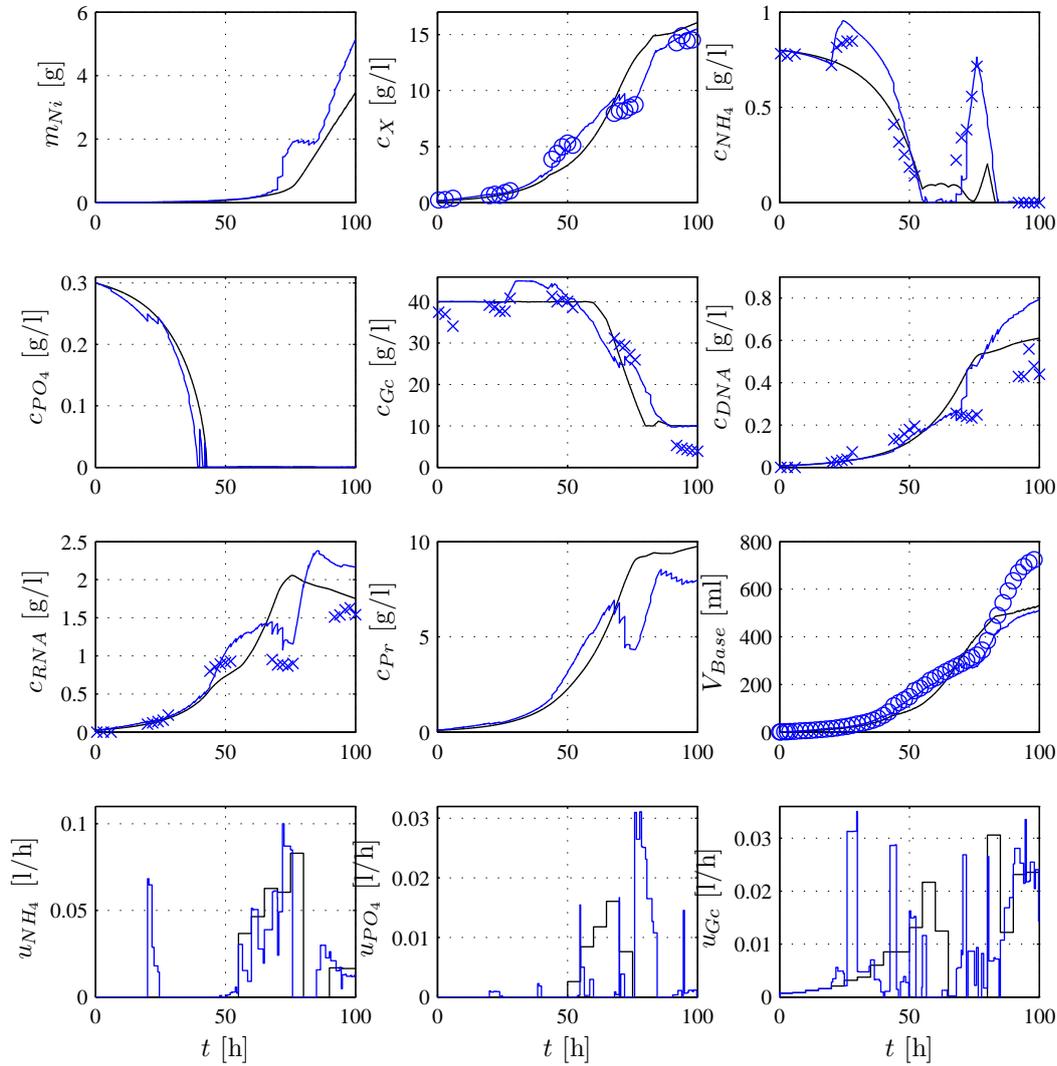


Abbildung 4.24.: Ergebnis der realen Umsetzung einer robusten Nichtlinearen-Modellprädiktiven-Folgeregelung (STdef12) bei Einsatz eines Unscented-Kalman-Filters zur Schätzung der Systemzustände. schwarz - Referenztrajektorie, blaue Linien - mit Hilfe der Zustandsschätzung errechneter Batchverlauf, blaue Kreise - Online/Atline verfügbare Messdaten, blaue Kreuze - Offline Analysen.

4. Anwendung robuster Prozessführungsverfahren

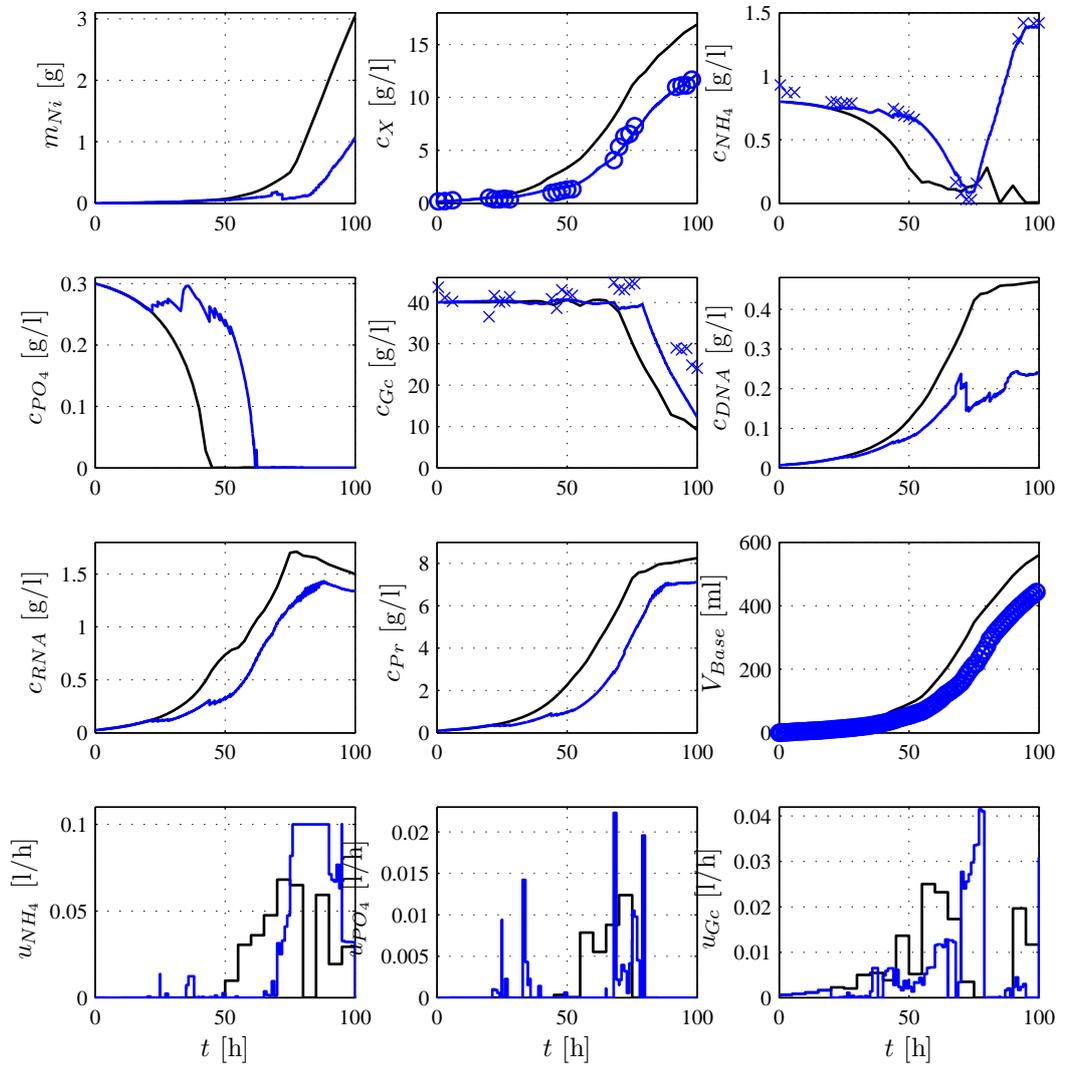


Abbildung 4.25.: Ergebnis der realen Umsetzung einer robusten Online-Trajektorienplanung (STdef14) bei Einsatz eines Constrained-Unscented-Kalman-Filters zur Schätzung der Systemzustände. schwarz - Referenztrajektorie, blaue Linien - mit Hilfe der Zustandsschätzung errechneter Batchverlauf, blaue Kreise - Online/Atline verfügbare Messdaten, blaue Kreuze - Offline Analysen.

4.3. Anwendung auf die Produktion von Nikkomycin durch *Streptomyces tendae*

4.25 zeigt eine von drei realen Umsetzungen der robusten Online-Trajektorienplanung. Erstaunlicherweise kommt es in allen Umsetzungen zu einem starken Wachstumsverzug der Organismen.

Trotz der starken Störung im gezeigten Fall arbeitet die beschränkte Zustandschätzung gut und ermöglicht es somit, den Batch erfolgreich zu Ende zu führen. Eine genaue Bewertung der Versuchsergebnisse ist dennoch nicht möglich, da aufgrund dieser extremen Veränderung des Wachstumsverhaltens der Organismen kein Vergleich mit der robusten Nichtlinearen-Modellprädiktiven-Folgeregelung bzw. den in [Hei04] gezeigten klassischen modellprädiktiven Verfahren möglich ist, weshalb hier letztendlich auch auf eine aufwändige Analyse der Zellkompartimente verzichtet wurde.

Nachbetrachtung der Versuche

Genauere Untersuchungen des bei der Umsetzung der Regelungsverfahren aufgetretenen Wachstumsverzuges legen die Vermutung nahe, dass der ausschließlich am Fachgebiet verfügbare Organismenstamm mutiert ist und somit vollständig neue Eigenschaften entwickelt hat. Mehrere nachträglich durchgeführte Experimente mit verschiedenen Umgebungsszenarien zeigen alle den beobachteten Wachstumsverzug und nur minimale Nikkomycinproduktion.

Auch wenn mit den wenigen Versuchen und der besonderen aufgetretenen Wachstumsstörung keine genaue Bewertung der Regelgüte möglich war, ist doch festzuhalten, dass alle Versuche technisch erfolgreich verlaufen sind. Aufgrund der langen Laufzeit der einzelnen Experimente war auch nicht zu erwarten, dass eine statistische Auswertung wie in den Simulationsstudien möglich ist. Dennoch ist die Umsetzbarkeit einer robusten Prozessführung an realen Anlagen erfolgreich praktiziert worden. Im Gegensatz zu klassischen modellbasierten Regelungsverfahren ist ein höherer Rechenbedarf zu verzeichnen, der im Rahmen steigender Leistungsfähigkeit der zur Verfügung stehenden Computer schnell zu kompensieren sein dürfte.

4. Anwendung robuster Prozessführungsverfahren

5. Zusammenfassung und Ausblick

Grundlage dieser Arbeit bildet die Tatsache, dass im Rahmen der Prozessführung verwendete mathematische Modelle immer nur eine Näherung der Realität darstellen. Auch bei bekannter Modellstruktur sind beispielsweise die Modellparameter immer aus verrauschten Messdaten zu identifizieren und somit zufällige Größen. Klassische Prozessführungsmethoden beachten die statistischen Eigenschaften des Systemmodells nicht.

Zunächst werden in Abschnitt 2.1 Methoden vorgestellt, die es ermöglichen, Auswirkungen zufälliger Störungen auf einen geplanten Prozessverlauf approximativ zu berechnen. Neben der Monte-Carlo-Simulation sind hier zusätzlich eine Approximation 1. Ordnung auf Basis von Sensitivitäten und die Unscented-Transformation, ein Verfahren 2. Ordnung zur Berechnung von Mittelwert und Varianz eines Prozessverlaufes, angegeben und in ein einheitliches Framework integriert, das Unsicherheiten in den Modellparametern, Anfangszustand, Messrauschen und den Einfluss eines Systemrauschens, wie es im Extended-Kalman-Filter verwendet wird, beinhaltet. Obwohl die Unscented-Transformation eine Approximation 2. Ordnung darstellt und, wie gezeigt, eine höhere Approximationsgenauigkeit der statistischen Momente liefert, skaliert der Rechenaufwand zur Berechnung, wie beim Ansatz erster Ordnung, nur linear mit der Anzahl der zufälligen Einflussgrößen.

Neben der Berechnung der Auswirkungen zufälliger Einflüsse ist vor allem auch die Beschreibung der statistischen Eigenschaften der Modellparameter von signifikanter Bedeutung. Hier werden die Bootstrap-Analyse und die Fisher'sche-Informationsmatrix verwendet. Die Grundlagen dazu sind in Abschnitt 2.2 zusammengefasst. Zusätzlich wäre es hier denkbar, die Unscented-Transformation auch zur Analyse der Parameterunsicherheiten zu verwenden, indem die Monte-Carlo-Simulation die im Rahmen der Bootstrap Analyse durchgeführt wird, durch die Unscented-Transformation ersetzt wird.

Neben der Analyse von Parameterfehlern, wird die Fisher'sche-Informationsmatrix auch zur Optimalen-Versuchsplanung verwendet. Dieses Verfahren basiert auf einer Parameterschätzung mit Hilfe eines Maximum-Likelihood-Ansatzes. Nutzt man zur Parameteridentifikation anstelle des Maximum-Likelihood-Ansatzes eine wiederholte Maximum-a-Posteriori-Schätzung ergibt sich das gleiche Ergebnis, wie bei Verwendung der Likelihood. Auf dieser Grundlage lässt sich eine Formulierung für eine neue Methode zur Optimalen-Versuchsplanung herleiten, die bei Verwendung einer Approximation erster Ordnung äquivalent zum Fisher'schen Ansatz ist. Zusätzlich bietet dieser neue Ansatz aber auch die Möglichkeit, Approximationsverfahren höherer Ordnung, wie z.B. die Unscented-Transformation zur Berechnung der benötigten statistischen Eigenschaften zu verwenden.

5. Zusammenfassung und Ausblick

den. Die Herleitung des Verfahrens ist in Abschnitt 2.3.2 dargestellt, die Anwendung in Kapitel 3 anhand eines kleinen Simulationsbeispiels gezeigt.

Im Rahmen der Anwendung des Verfahrens zeigen sich auch Schwächen, die u.a. auf die zur Berechnung der benötigten statistischen Momente verwendete Unscented-Transformation zurückzuführen sind. Das neue Verfahren basiert im Wesentlichen auf der Nutzung der Kovarianzmatrix $\mathbf{C}_{\theta y}$. Die Berechnung dieser Matrix mit dem Verfahren 2. Ordnung beinhaltet eine Formel (2.43), die letztendlich identisch zu einer Approximation erster Ordnung ist. Hier sind weitere Arbeiten nötig, um eine möglichst gute Berechnung dieser Kovarianzmatrix zu gewährleisten. Eine genaue Untersuchung des neuen Ansatzes zur Optimalen-Versuchsplanung würde den Rahmen dieser Arbeit sprengen und bleibt damit für folgende Arbeiten offen.

Der Kern der Arbeit befasst sich mit robusten Prozessführungsstrategien. Die mächtigsten, auch schon im industriellen Umfeld angewandten Verfahren nutzen physikalische dynamische Modelle der Prozesse, um optimale Steuerungen und Regelungen zu berechnen. Doch auch diese Verfahren können immer nur so gut sein, wie das unterliegende Modell. Dies bedingt, dass trotz der Anwendung solch komplexer Verfahren, als Folge von Modellunsicherheiten Produktionsschwankungen und Überschreitungen von Grenzwerten resultieren. Die vorgestellte Unscented-Transformation bietet die Möglichkeit, Auswirkungen von Modellunsicherheiten und Störungen auf den zukünftigen Prozessverlauf mit moderatem Rechenaufwand in guter Genauigkeit zu berechnen. Erweitert man die für die Planung verwendeten Gütefunktionale in einen aus der Finanzmathematik bekannten Value@Risk-Ansatz, bietet sich die Möglichkeit, die so vorausberechneten zu erwartenden Prozessschwankungen bei der Planung und Regelung zu berücksichtigen. Die Value@Risk Formulierung wird für die Trajektorienplanung, die Nichtlineare-Modellprädiktive-Folgeregelung und die Online-Trajektorienplanung exemplarisch eingeführt und in umfangreichen Simulationsstudien an zwei Beispielen getestet.

Schon bei der gesteuerten Umsetzung offline geplanter Trajektorien, lässt sich durch den robusten Ansatz die Schwankungsbreite der Produktivität stark verringern (siehe Abschnitt 4.1). Im vorgestellten Beispiel ist diese Steuerung sogar besser als die Nichtlineare-Modellprädiktive-Folgeregelung bzw. die Online-Trajektorienplanung. Interessanterweise ist der Vorteil eines robust geregelten Prozesses im vorliegenden Fall auch nur sehr gering verglichen mit der robusten Steuerung. Dennoch kann auch eine robuste Steuerung nur Auswirkungen von Störungen erfassen, deren Einfluss auf den Prozess sowie deren statistische Eigenschaften bekannt sind. Die umfangreichen Tests mit verschiedenen Einstellungen der robusten Nichtlinearen-Modellprädiktiven-Folgeregelung zeigen auch, dass der Vorteil der robusten Regelung in diesem Fall fast ausschließlich auf der genaueren Berechnung des statistischen Mittelwertes der Modellprädiktion beruht. Die Vorteile einer robusten Regelung entfalten sich jedoch hauptsächlich dann, wenn auch ein signifikanter Einfluss auf die Varianz des Gütefunktionals besteht.

Durch die im Value@Risk Ansatz enthaltene Erweiterung der Beschränkungsfunktion, ist es ebenfalls möglich, einen statistisch errechneten variablen Sicherheitsabstand zu

wichtigen Prozessgrenzen einzuhalten und damit die Gefahr eines Auslösens sicherheitsgerichteter Schaltungen bei hoher Anlagenproduktivität zu verringern. In Abschnitt 4.2 sind daher Simulationsstudien für einen sicherheitskritischen Prozess gezeigt. Bereits die robuste Steuerung verringert das Risiko des Auslösens einer Sicherungseinrichtung massiv. Die robuste Nichtlineare-Modellprädiktive-Folgeregelung kann bei hoher Produktivität dieses Risiko fast auf Null reduzieren.

Sämtliche in dieser Arbeit vorgestellten Verfahren sind in dem Matlab-Projekt „Advanced Batch Control“, welches am Fachgebiet Mess- und Regelungstechnik der Technischen Universität Berlin entwickelt wurde, implementiert. Eine direkte Anbindung dieser Implementierung an über OPC ansprechbare Prozessleitsysteme ist ebenfalls vorhanden. In Abschnitt 4.3 ist die Umsetzbarkeit der Verfahren an einem realen Prozess, der Produktion von Nikkomycin durch den Stamm *Streptomyces tendae* gezeigt. Leider musste im Rahmen dieser Umsetzung festgestellt werden, dass der Stamm seit den Arbeiten von [Hei04] mutiert und der originale Stamm nicht mehr erhältlich ist. Daher ist ein Vergleich der Ergebnisse mit [Hei04] unmöglich.

Die technische Anwendbarkeit robuster nichtlinearer modellbasierter Prozessführungsverfahren für reale Systeme wurde dennoch erfolgreich nachgewiesen. Auch wenn bei heutiger Rechenkapazität noch ein ganzes Rechencluster zur online Berechnung der Regelungen nötig ist, wird mit steigender Rechenleistung auch eine einfachere Umsetzung der Verfahren möglich sein. Während [Hei04] noch ca. eine Stunde für die bei der Nichtlinearen-Modellprädiktiven-Folgeregelung durchzuführende nichtlineare Optimierung benötigte, ist eine solche Optimierung durch bessere Implementierungen und moderne Rechner heute in wenigen Minuten möglich.

Im Allgemeinen zeigt sich, dass der Einsatz robuster Verfahren zur offline Planung von Prozessen einen wesentlichen Vorteil gegenüber der klassischen Planung bietet. Robuste modellbasierte Regelungsverfahren haben genau dann bedeutende Vorteile, wenn wie in der Biotechnologie üblich, nur wenige Messgrößen zur Verfügung stehen und als Basis nur Modelle mit hohen Unsicherheiten verfügbar sind. Die Performance klassischer Regelungsverfahren steigt im Gegensatz dazu mit höherer Abtastfrequenz und besseren Zustandschätzverfahren. Die weitere Entwicklung muss daher zeigen, ob es sich als sinnvoll erweist, den Aufwand der Implementierung robuster modellbasierter Regelungsverfahren zu betreiben oder ob die Unzulänglichkeiten klassischer modellbasierter Regelungsverfahren durch moderne Algorithmen und höhere Rechenkapazität verringert werden können. Diese Arbeit zeigt in Abschnitt 2.6.3 eine Möglichkeit auf, die in allen gezeigten Verfahren nötige nichtlineare Optimierung durch die Verwendung eines Multiple-Shooting-Ansatzes zu beschleunigen. Auch die Verwendung von Filteralgorithmen, die auf der Unscented-Transformation beruhen, kann hier einen wesentlichen Beitrag leisten.

Letztendlich bietet die vorgestellte Approximation 2. Ordnung nur die Möglichkeit zwei statistische Momente zu approximieren. Insbesondere bei nichtlinearen Prozessmodellen sind die resultierenden Verteilungsfunktionen mit zwei Momenten jedoch nicht vollständig beschreibbar. Die Unscented-Transformation bietet die Möglichkeit, durch Erwei-

5. Zusammenfassung und Ausblick

terung des Stirling-Polynoms auch eine Approximation höherer Ordnung zu erhalten und damit mehr als zwei Momente zu berechnen. Erste Ansätze dazu sind in [JU04, NPR00a] zu finden. Andererseits wäre es auch denkbar, die Verteilungen durch eine Summe von Gaussverteilungen zu approximieren und damit eine höhere Approximationsgenauigkeit zu erreichen. Insbesondere durch eine bessere Approximation der statistischen Momente des Prozessverlaufes sind nochmals signifikante Verbesserungen der robusten Prozessführungsverfahren zu erwarten.

A. Ableitungsregeln für Vektoren

$$\frac{da}{d\underline{x}} = \left(\frac{da}{dx_1} \quad \frac{da}{dx_2} \quad \cdots \quad \frac{da}{dx_n} \right) \quad (\text{A.1a})$$

$$\frac{d(\underline{a}^T \underline{x})}{d\underline{x}} = \underline{a}^T \quad (\text{A.1b})$$

$$\frac{d(\underline{x}^T \underline{a})}{d\underline{x}} = \underline{a}^T \quad (\text{A.1c})$$

$$\frac{d(\underline{x}^T \mathbf{A} \underline{a})}{d\underline{x}} = \underline{a}^T \mathbf{A}^T \quad (\text{A.1d})$$

$$\frac{d(\underline{a}^T \mathbf{A} \underline{x})}{d\underline{x}} = \underline{a}^T \mathbf{A} \quad (\text{A.1e})$$

$$\frac{d(\underline{x}^T \mathbf{A} \underline{x})}{d\underline{x}} = \underline{x}^T (\mathbf{A} + \mathbf{A}^T) \quad (\text{A.1f})$$

Für symmetrische Matrizen \mathbf{A} gilt:

$$\frac{d(\underline{x}^T \mathbf{A} \underline{x})}{d\underline{x}} = 2\underline{x}^T \mathbf{A} \quad (\text{A.1g})$$

A. Ableitungsregeln für Vektoren

B. Constrained-Unscented-Kalman-Filter

Die Grundlage des hier gezeigten Algorithmus bildet das Systemmodell (2.2) ein geschätzter Systemzustand $\hat{\underline{x}}_{k-1}$ zum Zeitpunkt t_{k-1} mit der Kovarianz $\mathbf{C}_{\underline{x}_{k-1}}$, ein Modellparametersatz $\hat{\underline{\theta}}$ mit der Kovarianz $\mathbf{C}_{\underline{\theta}}$ und die Kovarianzmatrizen $\mathbf{C}_{\underline{\xi}}$ des Systemrauschens und $\mathbf{C}_{\underline{\eta}}$ des Messrauschens.

B.1. Time-Update

Im Time-Update werden die Prädiktionen des Systemzustandes $\hat{\underline{x}}_{k|k-1}$ zum Zeitpunkt t_k und dessen zu erwartende Unsicherheit $\mathbf{C}_{\underline{x}_{k|k-1}}$ wie folgt berechnet.

1. Zunächst sind die Cholesky-Zerlegungen der Kovarianzmatrizen der zufälligen Einflussgrößen \underline{x}_{k-1} , $\underline{\theta}$ und $\underline{\xi}$

$$\mathbf{S}_{\underline{x}_{k-1}} = \text{chol}(\mathbf{C}_{\underline{x}_{k-1}}) \quad \mathbf{S}_{\underline{\theta}} = \text{chol}(\mathbf{C}_{\underline{\theta}}) \quad \text{und} \quad \mathbf{S}_{\underline{\xi}} = \text{chol}(\mathbf{C}_{\underline{\xi}}) \quad (\text{B.1})$$

zu berechnen. Die i -te Spalte in einer Matrix $\mathbf{S}_{(\cdot)}$ sei hier analog zu der oben eingeführten Nomenklatur mit $\underline{s}_{(\cdot)}^{(i)}$ bezeichnet.

2. Aus den Verteilungsfunktionen der unsicheren Einflussgrößen sind die Sigmapunkte

$$\begin{aligned} \underline{x}_{k-1}^{(SP_i+)} &= \hat{\underline{x}}_{k-1} + h \underline{s}_{\underline{x}_{k-1}}^{(i)} \\ \underline{x}_{k-1}^{(SP_i-)} &= \hat{\underline{x}}_{k-1} - h \underline{s}_{\underline{x}_{k-1}}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{x}} \quad (\text{B.2a})$$

$$\begin{aligned} \underline{\theta}^{(SP_i+)} &= \hat{\underline{\theta}} + h \underline{s}_{\underline{\theta}}^{(i)} \\ \underline{\theta}^{(SP_i-)} &= \hat{\underline{\theta}} - h \underline{s}_{\underline{\theta}}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{\theta}} \quad (\text{B.2b})$$

$$\begin{aligned} \underline{\xi}^{(SP_i+)} &= +h \underline{s}_{\underline{\xi}}^{(i)} \\ \underline{\xi}^{(SP_i-)} &= -h \underline{s}_{\underline{\xi}}^{(i)} \end{aligned} \quad \text{mit } i = 1, \dots, L_{\underline{\xi}} \quad (\text{B.2c})$$

zu bestimmen.

3. Durch Einsetzen der Werte $\hat{\underline{x}}_{k-1}$, $\hat{\underline{\theta}}$ und $\underline{\xi} = \underline{0}$ und Lösung der Modellgleichung wird zunächst $\underline{x}_{k|k-1}^{(\overline{SP})}$ berechnet.

B. Constrained-Unscented-Kalman-Filter

4. Durch jeweiliges Einsetzen der Werte $\underline{x}_{k-1}^{(SP_i\pm)}$, $\hat{\underline{\theta}}$ und $\underline{\xi} = \underline{0}$ und Lösung der Modellgleichung werden die Zustände $\underline{x}_{k|k-1;\underline{x}_{k-1}}^{(SP_i\pm)}$ berechnet. Mit den Eingangsgrößen $\hat{\underline{x}}_{k-1}$, $\underline{\theta}^{(SP_i\pm)}$ und $\underline{\xi} = \underline{0}$ ergeben sich die korrespondierenden Werte $(\cdot)_{\hat{\underline{\theta}}}^{(SP_i\pm)}$ und mit $\hat{\underline{x}}_{k-1}$, $\hat{\underline{\theta}}$ und mit $\underline{\xi}^{(SP_i\pm)}$ die Größen $(\cdot)_{\underline{\xi}}^{(SP_i\pm)}$.

5. Mit (2.36) oder (2.37) lässt sich der Mittelwert $\hat{\underline{x}}_{k|k-1}$ des prädierten Systemzustands berechnen. Hier sei das Ergebnis nach (2.36) angegeben.

$$\begin{aligned}
 \hat{\underline{x}}_{k|k-1} = & \underline{x}_{k|k-1}^{(\overline{SP})} + \frac{1}{2} \sum_{i=1}^{L_x} \frac{\underline{x}_{k|k-1;\underline{x}_{k-1}}^{(SP_i+)} + \underline{x}_{k|k-1;\underline{x}_{k-1}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})}}{h_i^2} \\
 & + \frac{1}{2} \sum_{i=1}^{L_\theta} \frac{\underline{x}_{k|k-1;\hat{\underline{\theta}}}^{(SP_i+)} + \underline{x}_{k|k-1;\hat{\underline{\theta}}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})}}{h_i^2} \\
 & + \frac{1}{2} \sum_{i=1}^{L_x} \frac{\underline{x}_{k|k-1;\underline{\xi}}^{(SP_i+)} + \underline{x}_{k|k-1;\underline{\xi}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})}}{h_i^2}
 \end{aligned} \tag{B.3}$$

Die Kovarianzmatrix $\mathbf{C}_{\underline{x}_{k|k-1}}$ lässt sich mit (2.40) oder (2.41) schätzen. Mit (2.40)

ergibt sich

$$\begin{aligned}
 \mathbf{C}_{\underline{x}_{k|k-1}} &= \sum_{i=1}^{L_{\underline{x}}} \frac{1}{4h_i^2} \left(\underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i-)} \right) \left(\underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i-)} \right)^T \\
 &+ \sum_{i=1}^{L_{\underline{x}}} \frac{1}{2h_i^4} \left(\underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right) \\
 &\quad \cdot \left(\underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{x}_{k-1}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right)^T \\
 &+ \sum_{i=1}^{L_{\underline{\theta}}} \frac{1}{4h_i^2} \left(\underline{x}_{k|k-1, \underline{\theta}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{\theta}}^{(SP_i-)} \right) \left(\underline{x}_{k|k-1, \underline{\theta}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{\theta}}^{(SP_i-)} \right)^T \\
 &+ \sum_{i=1}^{L_{\underline{\theta}}} \frac{1}{2h_i^4} \left(\underline{x}_{k|k-1, \underline{\theta}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{\theta}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right) \\
 &\quad \cdot \left(\underline{x}_{k|k-1, \underline{\theta}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{\theta}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right)^T \\
 &+ \sum_{i=1}^{L_{\underline{\xi}}} \frac{1}{4h_i^2} \left(\underline{x}_{k|k-1, \underline{\xi}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{\xi}}^{(SP_i-)} \right) \left(\underline{x}_{k|k-1, \underline{\xi}}^{(SP_i+)} - \underline{x}_{k|k-1, \underline{\xi}}^{(SP_i-)} \right)^T \\
 &+ \sum_{i=1}^{L_{\underline{\xi}}} \frac{1}{2h_i^4} \left(\underline{x}_{k|k-1, \underline{\xi}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{\xi}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right) \\
 &\quad \cdot \left(\underline{x}_{k|k-1, \underline{\xi}}^{(SP_i+)} + \underline{x}_{k|k-1, \underline{\xi}}^{(SP_i-)} - 2\underline{x}_{k|k-1}^{(\overline{SP})} \right)^T.
 \end{aligned} \tag{B.4}$$

B.2. Measurement-Update

Ausgehend vom Ergebnis des Time-Updates lassen sich der prädizierte Messwert $\hat{y}_{k|k-1}$, dessen Unsicherheit $\mathbf{C}_{\underline{y}_{k|k-1}}$ und die benötigte Kreuzkovarianz $\mathbf{C}_{\underline{x}_{k|k-1}\underline{y}_{k|k-1}}$ wie folgt berechnen.

1. Zunächst sind die Cholesky-Zerlegungen der Kovarianzmatrizen der zufälligen Einflussgrößen $\underline{x}_{k|k-1}$, und $\underline{\theta}$

$$\mathbf{S}_{\underline{x}_{k|k-1}} = \text{chol}(\mathbf{C}_{\underline{x}_{k|k-1}}) \quad \text{und} \quad \mathbf{S}_{\underline{\theta}} = \text{chol}(\mathbf{C}_{\underline{\theta}}) \tag{B.5}$$

zu berechnen. Die i -te Spalte in einer Matrix $\mathbf{S}_{(\cdot)}$ sei hier analog zu der oben eingeführten Nomenklatur mit $\underline{s}_{(\cdot)}^{(i)}$ bezeichnet.

B. *Constrained-Unscented-Kalman-Filter*

2. Aus den Verteilungsfunktionen der unsicheren Einflussgrößen sind die Sigmapunkte

$$\underline{x}_{k|k-1}^{(SP_i+)} = \hat{\underline{x}}_{k|k-1} + h_{\underline{s}_{\underline{x}_{k|k-1}}}^{(i)} \quad \text{mit } i = 1, \dots, L_{\underline{x}} \quad (\text{B.6a})$$

$$\underline{x}_{k|k-1}^{(SP_i-)} = \hat{\underline{x}}_{k|k-1} - h_{\underline{s}_{\underline{x}_{k|k-1}}}^{(i)}$$

$$\underline{\theta}^{(SP_i+)} = \hat{\underline{\theta}} + h_{\underline{s}_{\underline{\theta}}}^{(i)} \quad \text{mit } i = 1, \dots, L_{\underline{\theta}} \quad (\text{B.6b})$$

$$\underline{\theta}^{(SP_i-)} = \hat{\underline{\theta}} - h_{\underline{s}_{\underline{\theta}}}^{(i)}$$

$$(\text{B.6c})$$

zu bestimmen.

3. Durch Einsetzen der Werte $\hat{\underline{x}}_{k|k-1}$ und $\hat{\underline{\theta}}$ und Lösung der Messgleichung wird zunächst $\underline{y}_{k|k-1}^{(\overline{SP})}$ berechnet.

4. Durch jeweiliges Einsetzen der Werte $\underline{x}_{k|k-1}^{(SP_i\pm)}$ und $\hat{\underline{\theta}}$ und Lösung der Messgleichung werden die Messgrößen $\underline{y}_{k|k-1, \hat{\underline{x}}_{k|k-1}}^{(SP_i\pm)}$ berechnet. Mit den Eingangsgrößen $\hat{\underline{x}}_{k|k-1}$ und $\underline{\theta}^{(SP_i\pm)}$ ergeben sich die korrespondierenden Werte $(\cdot)_{\underline{\theta}}^{(SP_i\pm)}$

5. Die benötigten statistischen Größen sind

$$\begin{aligned} \hat{\underline{y}}_{k|k-1} &= \underline{y}_{k|k-1}^{(\overline{SP})} + \frac{1}{2} \sum_{i=1}^{L_x} \frac{\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} + \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} - 2\underline{y}_{k|k-1}^{(\overline{SP})}}{h_i^2} \\ &\quad + \frac{1}{2} \sum_{i=1}^{L_\theta} \frac{\underline{y}_{k|k-1, \underline{\theta}}^{(SP_i+)} + \underline{y}_{k|k-1, \underline{\theta}}^{(SP_i-)} - 2\underline{y}_{k|k-1}^{(\overline{SP})}}{h_i^2} \end{aligned} \quad (\text{B.7})$$

$$\begin{aligned} \mathbf{C}_{\underline{y}_k} &= \mathbf{C}_\eta + \sum_{i=1}^{L_x} \frac{1}{4h_i^2} \left(\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} - \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} \right) \\ &\quad \cdot \left(\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} - \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} \right)^T \\ &\quad + \sum_{i=1}^{L_x} \frac{1}{2h_i^4} \left(\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} + \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} - 2\underline{y}_{k|k-1}^{(\overline{SP})} \right) \\ &\quad \cdot \left(\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} + \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} - 2\underline{y}_{k|k-1}^{(\overline{SP})} \right)^T \\ &\quad + \sum_{i=1}^{L_\theta} \frac{1}{4h_i^2} \left(\underline{y}_{k|k-1, \underline{\theta}}^{(SP_i+)} - \underline{y}_{k|k-1, \underline{\theta}}^{(SP_i-)} \right) \left(\underline{y}_{k|k-1, \underline{\theta}}^{(SP_i+)} - \underline{y}_{k|k-1, \underline{\theta}}^{(SP_i-)} \right)^T \end{aligned} \quad (\text{B.8})$$

$$\begin{aligned} \mathbf{C}_{\underline{x}_{k|k-1} \underline{y}_{k|k-1}} &= \sum_{i=1}^{L_x} s_{x_0}^{(i)} \frac{\left(\underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i+)} - \underline{y}_{k|k-1, \underline{x}_{k|k-1}}^{(SP_i-)} \right)}{2h_i} \end{aligned} \quad (\text{B.9})$$

Für das unbeschränkte Unscented-Kalman-Filter lautet die Gleichung zum Update des Systemzustandes nach [Mer04] mit dem zur Verfügung stehenden Messvektor \underline{y}_k

$$K = \mathbf{C}_{\underline{x}_{k|k-1} \underline{y}_{k|k-1}} \cdot \left(\mathbf{C}_{\underline{y}_k} \right)^{-1} \quad (\text{B.10})$$

$$\hat{\underline{x}}_k = \hat{\underline{x}}_{k|k-1} + K \cdot \left(\underline{y}_k - \hat{\underline{y}}_{k|k-1} \right) \quad (\text{B.11})$$

Im Constrained-Unscented-Kalman-Filter gilt

$$\begin{aligned} \hat{\underline{x}}_k &= \arg \min_{\underline{x}} \left(\underline{x} - \underline{x}_{k|k-1} \right)^T \mathbf{C}_{\underline{x}_{k|k-1}}^{-1} \left(\underline{x} - \underline{x}_{k|k-1} \right) \\ &\quad + \left(\underline{y}_k - \underline{h}(t_k, \underline{x}_k, \hat{\underline{\theta}}) \right)^T \mathbf{C}_\eta^{-1} \left(\underline{y}_k - \underline{h}(t_k, \underline{x}_k, \hat{\underline{\theta}}) \right), \end{aligned} \quad (\text{B.12})$$

B. Constrained-Unscented-Kalman-Filter

wobei dieses Optimierungsproblems mit verschiedensten Nebenbedingungen erweitert werden kann. Die Kovarianz des geschätzten Zustandes $\hat{\underline{x}}_k$ ist dann mit

$$\mathbf{C}_{\underline{x}_k} = \mathbf{C}_{\underline{x}_{k|k-1}} - \mathbf{K} \cdot \mathbf{C}_{\underline{y}_k} \cdot \mathbf{K}^T \quad (\text{B.13})$$

zu berechnen.

C. Kompartimentmodell des Stammes *S. tendae*

Das Modell und die Parameter wurden von [Kin94] entwickelt und mit den Anpassungen von [Maj98] übernommen. Die Massenbilanzen aller am Wachstum beteiligten Kompartimente, der Substrate und des Produktes sind in den folgenden Gleichungen zusammengefaßt, wobei $\mu(\cdot)$ und $\mu_z(\cdot)$ die Auf- und Abbaureaktionen des Kompartiments (\cdot) beschreiben.

$$\begin{aligned}
 \frac{d}{dt} \begin{pmatrix} m_D(t) \\ m_R(t) \\ m_{Pr}(t) \\ m_{As}(t) \\ m_{Nu}(t) \\ m_U(t) \\ m_{Am}(t) \\ m_{Ph}(t) \\ m_C(t) \\ m_{Ni}(t) \end{pmatrix} &= \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \dot{V}_{Am} \cdot c_{Am,Feed}(t) \\ \dot{V}_{Ph} \cdot c_{Ph,Feed}(t) \\ \dot{V}_C \cdot c_{C,Feed}(t) \\ 0 \end{pmatrix}}_{\text{Zulauf}} - \underbrace{\frac{\dot{V}_{ab}(t)}{V(t)} \begin{pmatrix} m_D(t) \\ m_R(t) \\ m_{Pr}(t) \\ m_{As}(t) \\ m_{Nu}(t) \\ m_U(t) \\ m_{Am}(t) \\ m_{Ph}(t) \\ m_C(t) \\ m_{Ni}(t) \end{pmatrix}}_{\text{Ablauf}} - \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mu_E(t)V_X(t) \\ 0 \end{pmatrix}}_{\text{Erhaltung}} + \\
 V_X(t) \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 1 & -Y_{AsNu} & -Y_{AsU} & 0 & Y_{AsPr} & 0 \\ -1 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 - Y_{AsPr} & 0 \\ 0 & 0 & 0 & -Y_{AAs} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -Y_{PhNu} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -Y_{CAs} & -Y_{CNu} & Y_{AsU} - 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{Baustoffwechsel}} \begin{pmatrix} \mu_D(t) \\ \mu_R(t) \\ \mu_{Pr}(t) \\ \mu_{As}(t) \\ \mu_{Nu}(t) \\ \mu_U(t) \\ \mu_{zR}(t) \\ \mu_{zPr}(t) \\ \mu_{Ni}(t) \end{pmatrix} & \quad (C.1)
 \end{aligned}$$

Als Indizes treten in den Modellgleichungen für den Stamm *S. tendae* auf: D - DNA; R - RNA; Pr - Proteine; As - Aminosäuren; Nu - Nukleotide; U - Strukturelemente; A - Ammonium; Ph - Phosphat; C - Glucose; X - Biotrockenmasse.

Allgemeines

$$m_X = m_D + m_R + m_{Pr} + m_{As} + m_{Nu} + m_U \quad (C.2a)$$

$$g_{(\cdot)}(t) = \frac{m_{(\cdot)}(t)}{V_X(t)} \quad c_{(\cdot)}(t) = \frac{m_{(\cdot)}(t)}{V(t)} \quad (C.2b)$$

DNA

$$\mu_D(t) = \mu_{D,max} \cdot \frac{g_{As}(t)}{g_{As}(t) + K_{PrAs}} \cdot \frac{g_{Nu}(t)}{g_{Nu}(t) + K_{PrNu}} \cdot g_R(t) \cdot g_D(t) \cdot \phi \quad (C.3)$$

RNA

$$\mu_R(t) = \mu_{R,max} \cdot \frac{g_{Nu}(t)}{g_{Nu}(t) + K_{RNu} + K_{RG} \cdot \frac{K_G}{K_G + g_{As}(t)} \cdot g_R(t)} \cdot g_D(t) \quad (C.4a)$$

$$\mu_{zR}(t) = \mu_{zR,max} \cdot \frac{K_{zR}}{K_{zR} + c_A(t)} \cdot g_R(t) \quad (C.4b)$$

Proteine

$$\mu_{Pr}(t) = \mu_{Pr,max} \cdot \frac{g_{As}(t)}{g_{As}(t) + K_{PrAs}} \cdot \frac{g_{Nu}(t)}{g_{Nu}(t) + K_{PrNu}} \cdot g_R(t) \quad (C.5a)$$

$$\mu_{zPr}(t) = \mu_{zPr,max} \cdot \frac{K_{zPr}}{K_{zPr} + c_A(t)} \cdot g_{Pr}(t) \quad (C.5b)$$

Aminosäurenbildung

$$\mu_{As} = \left(\mu_{As1,max} \frac{c_A(t)}{c_A(t) + K_{As1}} + \mu_{As2,max} \frac{K_{As2}}{K_{As2} + c_A(t)} \right) \cdot \frac{c_C(t)}{c_C(t) + K_{AsC}} \cdot \rho(g_{Nu}) \cdot g_{Pr}(t) \cdot \frac{1}{1 + c_{Ph}^3(t) \cdot K_{AsPh}} \quad c_A > 0 \quad (C.6a)$$

$$\rho(g_{Nu}) = 1 + \mu_{AsNu} \cdot \frac{g_{Nu}(t)}{g_{Nu}(t) + K_{AsNu}} \quad (C.6b)$$

Nukleotidbildung

$$\mu_{\text{Nu}}(t) = \left(\mu_{\text{Nu1,max}} \frac{c_{\text{Ph}}(t)}{c_{\text{Ph}}(t) + K_{\text{Nu1}}} + \mu_{\text{Nu2,max}} \frac{K_{\text{Nu2}}}{K_{\text{Nu2}} + c_{\text{Ph}}(t)} \right) \cdot \frac{g_{\text{As}}(t)}{g_{\text{As}}(t) + K_{\text{NuAs}}} \cdot g_{\text{Pr}}(t) \quad c_{\text{Ph}} > 0 \quad (\text{C.7})$$

Strukturelemente, Speicherstoffe und andere Zellbestandteile

Für $c_{\text{C}}(t) > 0$ gilt

$$\mu_{\text{U}}(t) = \mu_{\text{U,max}} \cdot \frac{g_{\text{As}}(t)}{g_{\text{As}}(t) + K_{\text{UAs}} + g_{\text{As}}^2(t)/K_{\text{UI}} + K_{\text{UNu}} \cdot g_{\text{Nu}}(t)} \cdot g_{\text{Pr}}(t) \quad (\text{C.8})$$

Erhaltungsstoffwechsel

$$\mu_{\text{E}}(t) = \mu_{\text{E,max}} \cdot \frac{c_{\text{C}}(t)}{c_{\text{C}}(t) + K_{\text{EC}}} \cdot g_{\text{X}} \quad (\text{C.9})$$

Produktbildung

$$\mu_{\text{Ni}}(t) = \left(\mu_{\text{Ni1,max}} \cdot \frac{K_{\text{NiAs}}}{K_{\text{NiAs}} + g_{\text{As}}(t)} + \mu_{\text{Ni2,max}} \cdot \frac{K_{\text{NiNu}}}{K_{\text{NiNu}} + g_{\text{Nu}}(t)} \right) \cdot g_{\text{D}}(t) \quad (\text{C.10a})$$

$$\mu_{\text{zNi}}(t) = \mu_{\text{zNim}} \cdot e^{[K_{\text{zNi1}} \cdot pH(t) - K_{\text{zNi2}}/T(t)]} \cdot c_{\text{Ni}} \quad (\text{C.10b})$$

Reaktorvolumen

$$\frac{dV(t)}{dt} = \dot{V}_{\text{Am}} + \dot{V}_{\text{Ph}} + \dot{V}_{\text{C}} - \dot{V}_{\text{ab}}(t) \quad (\text{C.11})$$

C. Kompartimentmodell des Stammes *S. tendae*

Aktivität

$$\frac{d\phi(t)}{dt} = -\mu_{\phi, \max} \cdot \frac{K_{\phi As}}{K_{\phi As} + g_{As}(t)} \cdot \phi(t) \quad \phi(t_0) = 1 \quad (\text{C.12})$$

Für die Prozessführungen wurde das oben angegebene Modell durch die zusätzliche Schätzgleichung

$$\frac{dV_{\text{Base}}(t)}{dt} = Y_{\text{BpH}} \cdot \dot{m}_X(t) \quad (\text{C.13})$$

erweitert.

Tabelle C.1.: Parameter des Modells des Stammes *S. tendae* mit $g_X = 1$ g/l

	Parameter	Wert		Standardabweichung aus [Maj98]	Einheit
		aus[Kin94]	aus [Maj98]		
μ_D	$\mu_{D,max}$	0.776	0.705	4.5%	1/(gh)
	K_{DAs}	0.0985	0.0164	0.4%	g/l
μ_R	$\mu_{R,max}$	0.288	1.888	3.7%	1/h
	K_{RNu}	$2.82 \cdot 10^{-4}$	0.0183	2.8%	g/l
	K_{RG}	52.6	11263	0.6%	1
	K_G	$2.3 \cdot 10^{-4}$	$0.138 \cdot 10^{-5}$	0.5%	g/l
μ_{zR}	$\mu_{zR,max}$	0.0194	0.0154	1.3%	1/h
	K_{zR}	0.0427	0.607	6.7%	g/l
μ_{Pr}	$\mu_{Pr,max}$	0.275	0.43	5%	1/h
	K_{PrAs}	0.0278	0.013	0.5%	g/l
	K_{PrNu}	0.00142	$0.556 \cdot 10^{-2}$	2.3%	g/l
μ_{zPr}	$\mu_{zPr,max}$	0.00812	$0.9 \cdot 10^{-2}$	0.5%	1/h
	K_{zPr}	0.0341	$0.29 \cdot 10^{-3}$	5.6%	g/l
μ_U	$\mu_{U,max}$	0.0917	0.018	1.5%	1/h
	K_{UAs}	0.272	0.1	9%	g/l
	K_{UI}	0.336	0.241	9.7%	g/l
	K_{UNu}	31.6	1.85	15%	1
μ_{Ni}	$\mu_{Ni1,max}$	$4.85 \cdot 10^{-3}$	$0.377 \cdot 10^{-2}$	0.4%	1/h
	K_{NiAs}	$1.73 \cdot 10^{-2}$	0.368	12.1%	g/l
	$\mu_{Ni2,max}$	$3.3 \cdot 10^{-2}$	0.03	0.6%	1/h
	K_{NiNu}	$1.4 \cdot 10^{-3}$	0.165	1%	g/l
μ_{zNi}	$\mu_{zNi,max}$	$2.74 \cdot 10^8$	$2.74 \cdot 10^8$		1/h
	K_{zNi1}	$7.87 \cdot 10^{-1}$	$7.87 \cdot 10^{-1}$		1
	K_{zNi2}	$8.83 \cdot 10^3$	$8.83 \cdot 10^3$		K
μ_{As}	$\mu_{As1,max}$	0.04	0.02	0.6%	1/h
	K_{As1}	0.0321	$0.414 \cdot 10^{-2}$	3.3%	g/l
	$\mu_{As2,max}$	0.0258	$0.64 \cdot 10^{-2}$	1.5%	1/h
	K_{As2}	0.0399	0.606	3.1%	g/l
	K_{AsC}	11.6	1.1418	7.5%	g/l
	μ_{AsNu}	3.51	2.07	3.1%	1
	K_{AsNu}	$3.99 \cdot 10^{-4}$	$0.248 \cdot 10^{-2}$	2.1%	g/l
	K_{AsPh}	5.07	0.856	7.2%	l^3/g^3
μ_{Nu}	$\mu_{Nu1,max}$	0.0592	0.0334	1%	1/h
	K_{Nu1}	0.0723	0.0774	3.5%	g/l
	$\mu_{Nu2,max}$	0.0325	0.606	1.7%	1/h
	K_{Nu2}	0.00605	$0.931 \cdot 10^{-2}$	0.5%	g/l
	K_{NuAs}	0.152	$0.436 \cdot 10^{-2}$	10.6%	g/l
μ_E	$\mu_{E,max}$	0.0634	0.0353	0.7%	1/h
	K_{EC}	7.81	0.6	15%	g/l
μ_ϕ	$\mu_{\phi,max}$	0.0320	0.0237	1%	1/h
	$K_{\phi As}$	$9.53 \cdot 10^{-4}$	$0.123 \cdot 10^{-3}$	1.5%	g/l
	Y_{AsNu}	0.0972	0.0183	3%	g/g
	Y_{AsU}	0.737	0.175	7.4%	g/g
	Y_{AsPr}	0.188	0.157	9.8%	g/g
	Y_{AAs}	0.192	0.226	0.8%	g/g
	Y_{PhNu}	0.327	0.313	1%	g/g
	Y_{CAs}	0.709	2.143	3.2%	g/g
	Y_{CNu}	1.35	1.096	11.8%	g/g

Der Koeffizient der zusätzlichen Zustandsgleichung beträgt $Y_{BpH} = 0.0032353$ l/g.

Prozessführungen mit dem Modell des Stammes *S. tendae*

Tabelle C.2.: Fermentationsbedingungen für den Stamm *S.tendae*.

Belüftungsrate	5	l/min
Rührerdrehzahl	500	h^{-1}
Temperatur	27	$^{\circ}C$
pH	6	-
Arbeitsvolumen	8.5 – 10	l
Inoculum	0.5	l

Tabelle C.3.: Startzusammensetzung für den Stamm *S. tendae* nach [Wal96].

Biotischer Anteil der Gesamtbiomasse bei $t = 0$	Wert [%]
g_D	4.5
g_R	15
g_{Pr}	60
g_{As}	4
g_{Nu}	4
g_U	12.5

Tabelle C.4.: Parameter der Fütterströme und des Basezufflusses für den Stamm *S. tendae*.

Teil-system	Para-meter	Wert	Ein-heit	Teil-system	Para-meter	Wert	Ein-heit
Am	$c_{Am,Feed}$	14	g/l	C	$c_{C,Feed}$	450	g/l
	ρ_{Am}	1025	g/l		ρ_C	1159	g/l
Ph	$c_{Ph,Feed}$	24	g/l	Base (3n-NaOH)	c_{Base}	3normal	
	ρ_{Ph}	1031	g/l				

Literaturverzeichnis

- [AG06] ARRELANO-GARCIA, A.: *Chance Constrained Optimization of Process Systems under Uncertainty*, Technische Universität Berlin, Diss., 2006
- [AZ00] ALLGÖWER, F. ; ZHENG, A.: *Nonlinear Model Predictive Control*. Birkhäuser, 2000
- [Bai98] BAILEY, J. E.: Mathematical modelling and analysis in biochemical engineering: past accomplishment and future opportunities. In: *Biotechnology Progress* 14 (1998), S. 8–20
- [Bau90] BAUER, I.: *Numerische Verfahren zur Lösung von Anfangswertaufgaben und zur Generierung von ersten und zweiten Ableitungen mit Anwendungen bei Optimierungsaufgaben in Chemie und Verfahrenstechnik*, Universität Heidelberg, Diss., 1990
- [BBKS00] BAUER, I. ; BOCK, H. G. ; KÖRKEL, S. ; SCHLÖDER., J. P.: Numerical Methods for Optimum Experimental Design in DAE Systems. In: *Journal of Computational and Applied Mathematics* 120 (2000), S. 1–25
- [BDLS00] BOCK, H.G. ; DIEHL, M. ; LEINWEBER, D.B. ; SCHLÖDER, J.P.: A direct multiple shooting method for real-time optimization of nonlinear DAE processes. In: ALLGÖWER, F. (Hrsg.) ; ZHENG, A. (Hrsg.): *Nonlinear Predictive Control* Bd. 26. Basel Boston Berlin : Birkhäuser, 2000 (Progress in Systems Theory), S. 246–267
- [BSSR94] BALTES, M. ; SCHNEIDER, R. ; STURM, C. ; REUSS, M.: Optimal Experimental Design for Parameter Estimation in Unstructured Growth Models. In: *Biotechnology Progress* 10 (1994), S. 480–488
- [CA98] CHEN, H. ; ALLGÖWER, F.: A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. In: *Automatica* 34 (1998), Nr. 10, S. 1205–1218
- [CB99] CAMACHO, E. F. ; BORDONS, C.: *Model Predictive Control*. Berlin : Springer Verlag, 1999
- [CR80] CUTLER, C. R. ; RAMAKER, B. L.: Dynamic matrix control - a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference*, 1980

- [CSA97] CHEN, Hong ; SCHERER, C. W. ; ALLGÖWER, Frank: A Game Theoretic Approach to Nonlinear Robust Receding Horizon Control of Constrained Systems. In: *Proceedings of American Control Conference, Albuquerque, NM, 1997*, S. 3073–3077
- [CV95] CHALONER, K. ; VERDINELLI, I.: Bayesian Experimental Design: A Review. In: *Statistical Science* 6 (1995), Nr. 3, S. 273–304
- [DBK06] DIEHL, M. ; BOCK, H. G. ; KOSTINA, E.: An approximation technique for robust nonlinear optimization. In: *Mathematical Programming B* 107 (2006), S. 213–230
- [DFS+02] DIEHL, M. ; FINDEISEN, R. ; SCHWARZKOPF, S. ; USLU, I. ; ALLGÖWER, F. ; BOCK, H. G. ; GILLES, E. D. ; SCHLÖDER, J. P.: An Efficient Algorithm for Nonlinear Model Predictive Control of Large Scale Systems. Part I: Description of the Method. In: *Automatisierungstechnik* 50 (2002), 12, S. 557–567
- [DFS+03] DIEHL, M. ; FINDEISEN, R. ; SCHWARZKOPF, S. ; USLU, I. ; ALLGÖWER, F. ; BOCK, H. G. ; GILLES, E. D. ; SCHLÖDER, J. P.: An Efficient Algorithm for Nonlinear Model Predictive Control of Large Scale Systems. Part II: Experimental Evaluation for a Distillation Column. In: *Automatisierungstechnik* 51 (2003), 1, S. 22–29
- [DJH02] DEVILLE, Y. ; JANSSEN, M. ; HENTENRYCK, P. van: Consistency Techniques for Ordinary Differential Equations. In: *Constraint* 7 (2002), S. 289–315
- [DR88] DICICCIO, T. J. ; ROMANO, J. P.: A Review of Bootstrap Confidence Intervals. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 50 (1988), Nr. 3, S. 338–354
- [Efr79] EFRON, B.: Bootstrap Methods: Another Look at the Jackknife. In: *The Annals of Statistics* 7 (1979), S. 1–26
- [Eng02] ENGELL, S.: Online-Optimierung in der Automatisierung. In: *Automatisierungstechnik* 50 (2002), 12, S. 555–562
- [Est95] ESTLER, M.: *Neue Ansätze zur adaptiven nichtlinearen Regelung von Fed-Batch-Bioreaktoren*, Universität Stuttgart, Diss., 1995
- [ET93] EFRON, B. ; TIBSHIRANI, R. J.: *Monographs on Statistics and Applied Probability*. Bd. 57: *An Introduction to the Bootstrap*. Chapman & Hall, 1993
- [Far92] FARWIG, H. M.: *Ein Beratungssystem zur Systemidentifikation*, Universität Bochum, Diss., 1992
- [Fis71] FISHER, R. A.: Proc. Cambridge Philos. Soc. (1925). In: *Collected papers of R.A. Fisher*. Adelaide: University of Adelaide, 1971

- [GP77] GOODWIN, G. ; PAYNE, R.: *Dynamic System Identification: Experimental Design and Data Analysis*. New York : Academic Press, 1977
- [Hal88] HALL, P.: Theoretical Comparison of Bootstrap Confidence Intervals. In: *The Annals of Statistics* 16 (1988), Nr. 3, S. 927–953
- [Hal92] HALL, P.: *The Bootstrap and Edgeworth Expansion*. Springer, 1992
- [Haz02] HAZEWINKEL, M. (Hrsg.): *Encyclopedia of Mathematics*. Springer-Verlag Berlin Heidelberg New York, 2002 <http://eom.springer.de/default.htm>
- [Hei04] HEINE, T.: *Modellgestützte Überwachung und Führung von Fed-Batch-Prozessen zur Antibiotikaproduktion*, TU Berlin, Diss., 2004
- [HKK03] HEINE, T. ; KAWOHL, M. ; KING, R.: Parameterschätzung für nichtlineare dynamische Modelle auf Basis stark verrauschter, statistisch unvollständig bestimmter Messdaten. In: *Chemie Ingenieur Technik* 75 (2003), Nr. 5, S. 543–550
- [Ise88] ISERMANN, R.: *Identifikation dynamischer Systeme, Band I*. Berlin : Springer Verlag, 1988
- [JKSM06] JOSHI, M. ; KREMLING, A. ; SEIDEL-MORGENSTERN, A.: Model based statistical analysis of adsorption equilibrium data. In: *Chemical Engineering Science* 61 (2006), S. 7805–7818
- [JSMK06] JOSHI, M. ; SEIDEL-MORGENSTERN, A. ; KREMLING, A.: Exploiting the bootstrap method for quantifying parameter confidence intervals in dynamical systems. In: *Metabolic Engineering* (2006), S. 447–455
- [JU96] JULIER, S. ; UHLMANN, J.: *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. citeseer.ist.psu.edu/julier96general.html. Version:1996
- [JU04] JULIER, S. ; UHLMANN, J.: Unscented Filtering and Nonlinear Estimation. In: *Proceedings of the IEEE* Bd. 92, 2004, S. 401–422
- [JUDW95] JULIER, S. ; UHLMANN, J. ; DURRANT-WHYTE, H.: A new approach for filtering nonlinear systems. In: *Proceedings of the American Control Conference*, 1995, S. 1628–1632
- [KB61] KALMAN, R. E. ; BUCY, R.: New results in linear filtering and prediction. In: *Journal of Basic Engineering (ASME)* 83D (1961), S. 98–108
- [KDM⁺05] KÜHL, P. ; DIEHL, M. ; MILEWSKA, A. ; MOLGA, E. ; BOCK, H.: NMPC for Runaway-Safe Fed-batch Reactors. In: *Assessment and Future Directions of Nonlinear Model Predictive Control, Freudenstadt-Lauterbach*, 2005, S. 467–474

- [KHK07] KAWOHL, M. ; HEINE, T. ; KING, R.: Model Based Estimation and Optimal Control of Fed-Batch Fermentation Processes for the Production of Antibiotics. In: *Journal of Chemical Engineering and Processing* 46 (2007), Nr. 11, S. 1223–1241
- [Kin89] KING, R.: *Modellgestützte Überwachung kritischer Reaktionssysteme*. Düsseldorf : VDI-Verlag, 1989 (Fortschritt-Berichte VDI : Reihe 8. 185). – ISBN 318148508X. – Zugl.: Stuttgart, Univ., Diss., 1988
- [Kin94] KING, R.: *Mathematische Modelle myzelförmig wachsender Mikroorganismen*. Düsseldorf : VDI-Verlag, 1994 (Fortschritt-Berichte VDI : Reihe 17. 103). – ISBN 3183103176. – Zugl.: Stuttgart, Univ., Habil.-Schr., 1992
- [Kin97] KING, R.: A Structured Mathematical Model for a Class of Organisms: 1st Development of a Model for *Streptomyces tendae* and Application of Model-Based Control. In: *Journal of Biotechnology* 52 (1997), S. 219–234
- [Kör02] KÖRKEL, S.: *Numerische Methoden für Optimale Versuchsplanungsprobleme bei nichtlinearen DAE-Modellen*, Universität Heidelberg, Diss., 2002
- [KRAH05] KLETTING, M. ; RAUH, A. ; ASCHEMANN, H. ; HOFER, E.: Consistency Techniques for Simulation of Wastewater Treatment Processes with Uncertainties. In: *Proceedings of the IFAC World Congress 2005*. Praha, Czech Republic, 2005
- [KSK00] KÜEHNLE, J. ; SCHAICH, D. ; KING, R.: Accelerating the mathematical modelling of bioprocesses with an integrated tool using qualitative and quantitative techniques. In: *Book of Abstracts Biotechnology 2000* (2000), S. 518–520
- [KW03] KÖPPE, M. ; WEISMANTEL, R.: *An Algorithm for Mixed Integer Optimization*. citeseer.nj.nec.com/572699.html. Version: 2003
- [Kwa93] KWAKERNAAK, Huibert: Robust control and H_∞ -optimization–Tutorial paper. In: *Automatica* 29 (1993), Nr. 2, S. 255 – 273
- [Li07] LI, Pu: *Prozessoptimierung unter Unsicherheiten*. Oldenbourg Verlag, 2007
- [LK04] LEIFHEIT, J. ; KING, R.: (Semi-)automatic modeling of biological reaction systems with TAM-B. In: *9th Int. Conference Computer Application in Biotechnology CAB9*, 2004
- [LPR92] LEGER, C. ; POLITIS, D. N. ; ROMANO, J. P.: Bootstrap Technology and Applications. In: *Technometrics* 34 (1992), Nr. 4, S. 378–398
- [LSBS03] LEINEWEBER, D. B. ; SCHÄFER, A. ; BOCK, H. G. ; SCHLÖDER, J. P.: An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization - Part II: Software aspects and applications. In: *Computers & Chemical Engineering* 27 (2003), Februar, Nr. 2, S. 167–174

- [Lud99] LUDEWIG, D.: *Expertensystem zur Entwicklung von Prozeßmodellen für biotechnische Prozesse*. Als Ms. gedr. Düsseldorf : VDI-Verl., 1999 (Fortschritt-Berichte VDI : Reihe 20. 289). – ISBN 3183289202. – Zugl.: Hamburg-Harburg, Techn. Univ., Arbeitsbereich Bioprozess- und Bioverfahrenstechnik, Diss., 1999
- [Maj98] MAJER, P. C.: *Parameterschätzung, Versuchsplanung und Trajektorienoptimierung für verfahrenstechnische Prozesse*. Als Ms. gedr. Düsseldorf : VDI-Verl., 1998 (Fortschritt-Berichte VDI : Reihe 3. 538). – ISBN 3183538032. – Zugl.: Stuttgart, Univ., Diss., 1997
- [MD03] MAHADEVAN, R. ; DOYLE, F. J.: Efficient optimization approaches to nonlinear model predictive control. In: *International Journal of Robust and Nonlinear Control* 13 (2003), S. 309–329
- [Mer04] MERWE, R. van d.: *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. Oregon (USA), OGI School of Science and Engineering at Oregon Health and Science University, Diss., 4 2004
- [Moo79] MOORE, R.E.: *Methods and Applications of Interval Analysis*. Studies in Applied Mathematics, Philadelphia, 1979
- [MP89a] MUNACK, A. ; POSTEN, C.: Design of Optimal Dynamical Experiments for Parameter estimation. In: *Proceedings of the 1989 American Control Conference* 4 (1989), S. 2010–2016
- [MP89b] MUNACK, A. ; POSTEN, C.: Optimale Versuchsplanung zur Modellierung des Wachstums von Pflanzenzellen in Submerskultur. In: *Automatisierungstechnik* 37(2) (1989), S. 55–65
- [MR94] MUSKE, K. R. ; RAWLINGS, J. B.: Nonlinear Moving Horizon State Estimation. In: *NATO ASI Series* 293 (1994), S. 349–365
- [Mun91] MUNACK, A.: Optimization of Sampling. In: *Biotechnology* 4 (1991), S. 251–264
- [NA00] NATH, Ravi ; ALZEIN, Zak: On-line dynamic optimization of olefins plants. In: *Computers and Chemical Engineering* 24 (2000), Nr. 2-7, S. 533 – 538
- [NB03] NAGY, Z. ; BRAATZ, R.: Worst-Case and Distributional Robustness Analysis of Finite-Time Control Trajectories for Nonlinear Disturbed Parameter Systems. In: *IEEE Transactions on Control Systems Technology* 11 (2003), S. 694–704
- [NB04] NAGY, Z. ; BRAATZ, R.: Open-loop and closed-loop robust optimal control of batch processes using distributional and worst-case analysis. In: *Journal of Process Control* 14 (2004), S. 411–422
- [Neu90] NEUMAIER, A.: *Interval Methods for Systems of Equations*. Cambridge : Cambridge University Press, 1990

- [NM65] NELDER, J. A. ; MEAD, R.: A Simplex Method for Function Minimization. In: *Computer Journal* 7 (1965), S. 308–313
- [NPR00a] NORGAARD, M. ; POULSEN, N. ; RAVN, O.: Advances in Derivative-Free State Estimation for Nonlinear Systems / Dept. of Mathematical Modelling. Technical University of Denmark, April 2000. – IMM-REP-1998-15
- [NPR00b] NORGAARD, M. ; POULSEN, N. ; RAVN, O.: New Developments in State Estimation for Nonlinear Systems. In: *Automatica* 36 (2000), Nr. 11, S. 1627–1638
- [Pap96] PAPAGEORGIOU, M.: *Optimierung. statische, dynamische, stochastische Verfahren für die Anwendung*. München, Wien : Oldenburg Verlag, 1996
- [Poh99] POHLHEIM, H.: *Evolutionäre Algorithmen, Verfahren, Operatoren und Hinweise für die Praxis*. Berlin, Heidelberg, New York : Springer-Verlag, 1999
- [QB03] QIN, S. J. ; BADGEWELL, T. A.: A survey of industrial model predictive control technology. In: *Control Engineering Practice* 11 (2003), S. 733–764
- [RS05] RUSZCZYNSKI, A. ; SHAPIRO, A.: Conditional Risk Mappings / Economics Working Paper Archive EconWPA. 2005. – Risk and Insurance. – <http://ideas.repec.org/p/wpa/wuwpri/0404002.html>
- [RU00] ROCKAFELLAR, R. T. ; URYASEV, S.: Optimization of Conditional Value-at-Risk. In: *Journal of Risk* 2 (2000), S. 21–41
- [Sah04] SAHINIDIS, N.: Optimization under uncertainty: state-of-the-art and opportunities. In: *Computers and Chemical Engineering* 28 (2004), S. 971–983
- [SB92] STRYK, O. von ; BULIRSCH, R.: Direct and indirect methods for trajectory optimization. In: *Annals of Operations Research* 37 (1992), S. 357–373. – ISSN 0254–5330
- [SBVP03] SRINIVASAN, B. ; BONVIN, D. ; VISSER, E. ; PALANKI, S.: Dynamic Optimization of Batch Processes: II. Role of Measurements in Handling Uncertainty. In: *Computers and Chemical Engineering* 27 (2003), S. 27–44
- [Sch82] SCHITTKOWSKI, Klaus: The nonlinear programming method of Wilson, Han, and Powell with an augmented Lagrangian type line search function. In: *Numerische Mathematik* 38 (1982), S. 83–114
- [Sha00] SHAPIRO, A.: *Stochastic programming by Monte Carlo simulation methods*. Stochastic Programming E-Print Series. <http://dohost.rz.hu-berlin.de/speps/>. Version: 2000
- [SK00] SCHAICH, D. ; KING, R.: Automatic Modelling of Chemical and Biological Systems. In: *Proc. of the 3rd IMACS Symp. on Mathematical Modelling, Wien* (2000)

- [Ste98] STEARNS, D.: *The Basics of Programming Model Design*. <http://msdn.microsoft.com/de-de/library/ms809977%28en-us%29.aspx>.
Version: 1998
- [Sti30] STIRLING, J: *Methodus differentialis, sive tractatus de summation et interpolation serierum infinitarum*. London. English translation by J. Holliday "The Differential Method: A Treatise of the Summation and Interpolation of Infinite Series" (1749). 1730
- [Sto85] STOER, J.: Principles of sequential quadratic programming methods for solving nonlinear programs. In: *Computational Mathematical Programming* 15 (1985), S. 165–208
- [VG00] VALAPPIL, J. ; GEORGAKIS, C.: Systematic estimation of state noise statistics for Extended Kalman Filters. In: *AIChE J.* 46 (2000), S. 292–308
- [Wal96] WALDRAFF, W.: *Modellgestützte Überwachung und Führung von Fed-Batch-Bioreaktoren*, Universität Stuttgart, Diss., 1996
- [WKG97] WALDRAFF, W. ; KING, R. ; GILLES, E. D.: Optimal feeding strategies by adaptive mesh selection for fed-batch bioprocesses. In: *Bioprocess Engineering* 17 (4) (1997), S. 221–227