# Forschungsbericht Institut für Automatisierungstechnik und Softwaresysteme

Hrsg.: Prof. Dr.-Ing. Dr. h. c. Michael Weyrich

Benjamin Hendrik Lindemann

# Datengetriebene Kompensation anomaler Prozessdynamiken in Automatisierungssystemen auf Basis adaptiver LSTM-Netze

Band 1/2021

Universität Stuttgart

## Datengetriebene Kompensation anomaler Prozessdynamiken in Automatisierungssystemen auf Basis adaptiver LSTM-Netze

Von der Fakultät Informatik, Elektrotechnik und Informationstechnik der Universität Stuttgart zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.) genehmigte Abhandlung

> Vorgelegt von Benjamin Hendrik Lindemann aus Lohne (Oldenburg)

Hauptberichter: Mitberichter: Prof. Dr.-Ing. Dr. h. c. Michael Weyrich Prof. Dr.-Ing. Dr. h. c. Mathias Liewald MBA

Tag der mündlichen Prüfung: 09.06.2021

Institut für Automatisierungstechnik und Softwaresysteme der Universität Stuttgart

2021

IAS-Forschungsberichte

Band 1/2021

## Benjamin Hendrik Lindemann

# Datengetriebene Kompensation anomaler Prozessdynamiken in Automatisierungssystemen auf Basis adaptiver LSTM-Netze

D 93 (Diss. Universität Stuttgart)

Shaker Verlag Düren 2021

#### Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Zugl.: Stuttgart, Univ., Diss., 2021

Copyright Shaker Verlag 2021 Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8440-8186-2 ISSN 1610-4781

Shaker Verlag GmbH • Am Langen Graben 15a • 52353 Düren Telefon: 02421/99011-0 • Telefax: 02421/99011-9 Internet: www.shaker.de • E-Mail: info@shaker.de Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Automatisierungstechnik und Softwaresysteme (IAS) der Universität Stuttgart.

Mein besonderer Dank gilt meinem Doktorvater und Leiter des Instituts, Herrn Prof. Dr.-Ing. Dr. h. c. Michael Weyrich, für Betreuung dieser wissenschaftlichen Arbeit, die vielen konstruktiven Diskussionen und wertvollen Hinweise und Anregungen in den letzten Jahren sowie für die Übernahme des Hauptberichts.

Herrn Prof. Dr.-Ing. Dr. h. c. Mathias Liewald MBA danke ich für die lehrreiche und angenehme Zusammenarbeit im BMWi-Projekt EMuDig 4.0, für das entgegengebrachte Interesse an meiner Arbeit und die Übernahme des Mitberichts.

Meinem Projektpartner im BMWi-Projekt, Herrn Dr. sc. techn. Celalettin Karadogan, danke ich für die gute Zusammenarbeit beim Aufbau der Stuttgarter Modellfabrik.

Bei meinen Kolleginnen und Kollegen bedanke ich mich für die gegenseitige Unterstützung, die vielen guten und konstruktiven Diskussionen sowie die freundschaftliche Atmosphäre am und auch außerhalb des IAS. Ein besonderer Dank gilt Herrn Dr.-Ing. Nasser Jazdi für seine fortwährende Unterstützung während der Arbeit und die vielen wertvollen Gespräche.

Ebenso gilt mein Dank den zahlreichen Studierenden, die im Rahmen ihrer Abschlussarbeiten einen Beitrag zur technischen Umsetzung des Konzepts geleistet haben.

Ganz besonderer Dank gilt meiner Freundin Isabel, meiner Familie und meinen Freunden für die Unterstützung, den Rückhalt und die Motivation.

Stuttgart, im Juni 2021 Benjamin Lindemann

### Widmung

Diese Arbeit widme ich meinem Großvater Heinrich Schockemöhle.

# Inhaltsverzeichnis

Ab	bildu	ngsverz	zeichnis	iii
Ta	bellen	verzeic	chnis	v
Ab	kürzu	ingsver	rzeichnis	vi
Zu	samm	enfassi	ung	ix
Ab	stract	t <b></b>		x
1	Einl	eitung.		1
	1.1	Daten	getriebene Prozessmodellierung	1
	1.2	Proble	emstellung und Abgrenzung	
	1.3	Herau	isforderungen	
	1.4	Anfor	derungen an die Konzeption	5
	1.5	Zielse	tzung der Arbeit	6
	1.6	Aufba	u der Arbeit	7
2	Gru	ndlager	Ω	10
	2.1	Nichtl	lineare zeitvariante Prozessmodellierung	
		2.1.1	Anomalien	
		2.1.2	Rekurrente neuronale Netze	
		2.1.3	Long short-term memory	
		2.1.4	Autoencoder	17 18
	22	Komn	pensation anomaler Systemdynamiken	
	2.2	2.2.1	Beobachteransatz	
		2.2.1	Modellbasierter prädikativer Ansatz	
3	Stan	d der V	Wissenschaft und Technik	22
	3.1	Nichtl	lineare zeitvariante Prozessmodellierung	
		3.1.1	Detektion von Anomalien	
		3.1.2	RNN und LSTM zur Modellierung	
		3.1.3	Weitere Netzarchitekturen	
		3.1.4	Sequence-to-Sequence LSTM	
	2.2	5.1.5 V	Beweitung der Ansatze	
	3.2	Komp	Dislante heuteille ere eren Ausätze	
		3.2.1	Ansätze für mehrstufige Prozesse	
		3.2.3	Modellbasierte prädikative Ansätze	
		3.2.4	Adaptive Ansätze	
		3.2.5	Bewertung der Ansätze	45
	3.3	Aufze	igen des Forschungsbedarfs	

4	Kon	zept zu	r adaptiv	en Qualitätssicherung	49	
	4.1	1 Beschreibung der Konzeptidee				
	4.2	Ansat	zessmodellierung	53		
		4.2.1	Modelli	erung einzelner Fertigungsprozesse	54	
			4.2.1.1	Numerische Simulation diskreter Fertigungsprozesse	54	
		422	4.2.1.2 Madalli	Ablentung von diskreten E/A-Modenen	50	
		4.2.2	4.2.2.1	Graphentheoretischer Modellierungsansatz		
			4.2.2.2	Integration der diskreten E/A-Modelle	59	
	4.3	Ansat	z zur Qua	litätssicherung	61	
		4.3.1 Erweiterung des Prozessmodells			61	
		4.3.2	Detektio	on anomaler Systemdynamiken	67	
			4.3.2.1	Beobachterstruktur zur Detektion von Anomalien	67	
		422	4.3.2.2 M-1-11:	weiterverarbeitung der erlassten Anomalie-Daten	70	
		4.3.3	4 3 3 1	LSTM-basierte Prädiktionsmodelle	74 74	
			4.3.3.2	Kooperativer Prädiktionsansatz	82	
		4.3.4	Komper	sation anomaler Systemdynamiken	84	
			4.3.4.1	Rekursive Verknüpfung kooperativer Prädiktionen	85	
			4.3.4.2	Anpassung der Stellgrößen zur Kompensation von Anomalien.	88	
5	Real	isierun	g		93	
	5.1	Überb	lick über	das Gesamtsystem	94	
	5.2	2 Komponente zur Datenerfassung		r Datenerfassung	95	
		5.2.1	Subkom	ponente Bauteilrückverfolgung und Datenverarbeitung	95	
		5.2.2	Subkom	ponente Datenverwaltung und -sicherheit	97	
	5.3	.3 Komponente zur adaptiven Qualitätssicherung				
		5.3.1	Subkom	ponente Detektion	100	
		5.3.2	Subkom	ponente Modellierung	100	
	5 4	5.5.5 Vomm	Subkolli	r die Denutzersberfläche	102	
	5.4	Komponente für die Benutzeroberfläche				
6	Eval	luierun	g		106	
	6.1	Evalu	ierungsfal	ll 1: Anomalie-Detektion und Prädiktion	107	
		6.1.1	Beschre	ibung der Prozessketten: Modellfabrik und Otto Fuchs	108	
		6.1.2	Diskuss	ion und Bewertung der empirischen Ergebnisse	108	
	6.2	Evalu	ierungsfal	12: Kompensation von Anomalien und Nutzerassistenz	115	
		6.2.1	Beschre	ibung der Prozessketten: Modellfabrik und Otto Fuchs	115	
	()	0.2.2	Diskuss	non und Bewertung der empirischen Ergebnisse	122	
	0.3	Abgie	ien mit A	morderungen und Zielseizung der Arbeit	122	
7	Schl	hlussbetrachtung				
	7.1	Zusan	nmenfassu	ing der Ergebnisse	124	
	7.2	Ausbl	ick auf w	eiterführende Forschungsaktivitäten	126	
8	Lite	raturve	rzeichnis	S	127	

# Abbildungsverzeichnis

Abbildung 1.1: Aufbau der Arbeit	8
Abbildung 2.1: Illustration der drei Anomalie-Typen univariater Zeitreihen nach [40] 1	2
Abbildung 2.2: RNN mit direkten, indirekten und lateralen Rückkopplungen (links) und	
aufgefaltetes Neuron eines RNN (rechts)1	4
Abbildung 2.3: LSTM-Zelle (links) im Vergleich zur GRU-Zelle (rechts) 1	5
Abbildung 2.4: Seq2Seq Varianten mit flexibel einstellbaren Ein- und Ausgangssequenzlängen	
1	9
Abbildung 2.5: Aufbau eines Beobachters mit Filter und Störgrößenaufschaltung [7] 2	0
Abbildung 3.1: CNN und LSTM zur Detektion von Anomalien gemäß [40]2	6
Abbildung 3.2: Seriell-parallele Systemidentifikation mit konvexem LSTM-Netz nach [81] 2	8
Abbildung 3.3: Bidirektionales LSTM (links) und Filterschicht zur Verarbeitung lückenhafter	
Datensätze (rechts) nach [91]	0
Abbildung 3.4: Erweiterung der LSTM-Zellen im 2d Grid LSTM im Vergleich zu Stacked	
LSTM gemäß [100]	1
Abbildung 3.5: Cross modal LSTM mit Informationsaustausch zwischen zwei Streams [102]. 3	2
Abbildung 3.6: Seq2Seq-Netz mit LSTM-Zellen zur Vorhersage von Ausgabesequenzen	
variabler Zeithorizonte nach [105]	3
Abbildung 3.7: EWMA-Regelung bezogen auf einen Prozess über alle Bauteile (oben) sowie ein	n
Bauteil über alle Prozesse (unten) nach [119]	7
Abbildung 3.8: Überlagerung einer zeitkontinuierlichen Prozessgrößenregelung und einer	
diskreten bauteilbezogenen Regelung nach [4] 3	8
Abbildung 3.9: Kaskadiertes Regelungsschema zur Kompensation von Wechselwirkungen vor-	
und nachgelagerter Prozessschritte nach [8]	9
Abbildung 3.10: Komposition aus der Beobachtung auftretender Störungen und der darauf	
aufbauenden Prädiktion mit neuronalem Prozessmodell nach [13] 4	3
Abbildung 3.11: Datenspeicherung zur Umsetzung des Concurrent Learning [54]	5
Abbildung 4.1: Illustration der grundlegenden Konzeptidee zur adaptiven Regelung der Qualität	t
	9
Abbildung 4.2: Vereinfachte Darstellung des Schemas zur Kompensation von Anomalien 5	0
Abbildung 4.3: Modellbildungsprozess und Entwurf des Kompensationsschemas5	2
Abbildung 4.4: Beispiel für die Durchführung einer DoE-Studie zur Untersuchung von	
Eingangs-Ausgangsrelationen diskreter Fertigungsprozesse [24] 5	6
Abbildung 4.5: Modellierung von Prozessketten als gerichtete azyklische Graphen in Anlehnung	g
an [29]5	9
Abbildung 4.6: Verknüpfung zweier Prozesse innerhalb des Graphen-basierten	
Prozesskettenmodells	1
Abbildung 4.7: Informationsfluss des um die Modellierung interner Modellunsicherheiten	
erweiterten Prozessmodells	3
Abbildung 4.8: Informationsfluss des um die Modellierung unbekannter und auf die Aktoren	
wirkender Störeffekte erweiterten Prozessmodells	6
Abbildung 4.9: Schema zur Anomalie-Detektion basierend auf einem inversen Prozessmodell 6	8
Abbildung 4.10: LSTM-Autoencoder mit vorgeschalteter DWT zur Anomalie-Detektion 6	9
Abbildung 4.11: Erfassung und Verwaltung von Prozessdaten für das Vor- und Weitertraining	
von Modellen in Anlehnung an [54]7	1
Abbildung 4.12: Training des Anomalie-Detektors mit verschiedenen Datensätzen	2
Abbildung 4.13: Seq2Seq-Architektur basierend auf LSTM-Zellen7	5
Abbildung 4.14: Kombinationsmöglichkeiten ausgewählter Modellierungselemente für die	
Prädiktion von Prozess- und Anomalie-Verhalten7	6

Abbildung 4.15: Vereinfachte Darstellung des kooperativen Prädiktionsschemas zur Vorhersage
des Prozessausgangs (oben) unter Berücksichtigung anomaler
Prozessdynamiken (unten)
Abbildung 4.16: Verknüpfung von Anomalie-Detektor und kooperativem Prädiktionsschema zur
Modellierung des Prozess- und Anomalie-Verhaltens
Abbildung 4.17: Veranschaulichung des rekursiven Verfahrens zur Bestimmung der Stellgrößen für alle Prozesse der Prozesskette
Abbildung 4.18: Blockschaltbild zur Kompensation von Anomalien und Einhaltung der
Produktqualität auf Basis optimierter Stellgrößen
Abbildung 5.1: Systemarchitektur des entwickelten datengetriebenen Assistenzsystems zur
adaptiven Qualitätsregelung
Abbildung 5.2: Komponente zur Vereinheitlichung heterogener Daten (links) und semantische
Annotation zur eindeutigen Bauteilrückverfolgung (rechts)
Abbildung 5.3: Aufbau des realisierten Hadoop-Clusters (links) sowie der eingerichteten
Absicherung des Datenzugriffs durch Kerberos (rechts)
Abbildung 5.4: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Detektion 100
Abbildung 5.5: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Modellierung 101
Abbildung 5.6: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Kompensation
Abbildung 5.7: Startseite des Assistenzsystems (links) und Ansicht zur Prozesskontrolle (rechts)
Abbildung 5.8: Monitoring gefilterter Prozessdatenverläufe (links) und Überwachung definierter
Qualitätsmetriken (rechts)
Abbildung 5.9: Bauteilrückverfolgung entlang der Prozesskette (links) und Anzeige von
Metadaten zu den Produktionseinheiten (rechts)
Abbildung 6.1: Beispiel eines normalen Druckverlaufs im Vergleich zu den Auswirkungen von
auf das Pumpensystem wirkenden Anomalien
Abbildung 6.2: Detektion und Charakterisierung von Anomalien 109
Abbildung 6.3: Durchschnittliche Abweichung in der Rekonstruktion über die Zeitsequenzen
ausgewählter Ereignisse
Abbildung 6.4: Schleichende Abweichung über die Zeit (grün) und Normalverhalten
(dunkelblau) sowie deren Detektion mittels des Anomalie-Detektors (hellblau)
Abbildung 6.5: Kooperative Vorhersage des Prozessverhaltens unter Berücksichtigung von
Anomalien112
Abbildung 6.6: GRU vs. LSTM zur Prädiktion von Kurzzeitanomalien113
Abbildung 6.7: Kooperative Prädiktion des Prozessverhaltens unter Berücksichtigung
auftretender Langzeitanomalien 114
Abbildung 6.8: Prozesskette der Modellfabrik mit induktiver Erwärmung, hydraulischer Presse
sowie Transporteinheit (links) und Werkzeug mit zwei Kavitäten (rechts) 116
Abbildung 6.9: Durchschnittliches Konvergenzverhalten von vier neuronalen Netzen mit
unterschiedlich strukturierten Eingangsparameterräumen 118
Abbildung 6.10: Aggregierte Darstellung der Netzvorhersage unter Berücksichtigung von
Störeffekten119
Abbildung 6.11: Detailansicht zur Störcharakterisierung bei zweistufiger Umformung (Stufe 1
links, Stufe 2 rechts) 120
Abbildung 6.12: Datengetriebene Kompensation von Anomalien auf der Grundlage des
eingeführten Kontrollschemas 120
Abbildung 6.13: Assistenzsystem bei der Otto Fuchs KG und Untersuchungen zu
Berechnungszeiten verschiedener Recheneinheiten und Netzgrößen 121

# Tabellenverzeichnis

Tabelle 3.1: Bewertung der Netzarchitekturen hinsichtlich der Anforderungen	35
Tabelle 3.2: Bewertung der Qualitätsregelungen hinsichtlich der Anforderungen	46
Tabelle 4.1: Datensatz-Konstellationen und resultierende Störgrößendifferenz	73
Tabelle 5.1: Daten zur verwendeten Hardware	99
Tabelle 6.1: Übersicht über die verwendeten Datensätze und die durchgeführten Testläufe.	106
Tabelle 6.2: Vorhersagegenauigkeiten der untersuchten Architekturen für abrupte Anomali	en
	113
Tabelle 6.3: Übersicht über die einstellbaren Parameter der Hyperparameteroptimierungen	117

# Abkürzungsverzeichnis

AE	Autoencoder
API	Application Programming Interface
AR	Autoregressiv
ARMA	Autoregressive Moving Average
ARMAX	Autoregressive Moving Average with Exogenous Inputs
ARIMA	Autoregressive Integrated Moving Average
BD	Bidirektional
BPTT	Backpropagation Through Time
CAT	Control and Automation Technology
CentOS	Community Enterprise Operating System
CNN	Convolutional Neural Network
CPS	Cyber-physisches System
CPU	Central Processing Unit
CSS	Cascading Style Sheets
CSV	Comma Separated Values
CUDA	Compute Unified Device Architecture
DOB	Disturbance Observer
DoE	Design of Experiments
DWT	Diskrete-Wavelet-Transformation
DZ	Digitaler Zwilling
E/A-Modell	Eingangs-Ausgangs-Modell
EMuDig4.0	Effizienzschub in der Massivumformung durch Entwicklung und Integration digitaler Technologien im Engineering der gesamten Wertschöpfungskette
ETL	Extract Transform Load
EWMA	Exponentially Weighted Moving Average
FEM	Finite-Elemente-Methode
FPS	Flexibles Produktionssystemen
GAN	Generative Adversarial Network

GB	Gigabyte
GPU	Graphics Processing Unit
GRNN	General Regression Neural Network
GRU	Gated Recurrent Unit
GUI	Graphical User Interface
HDF	Hierarchical Data Format
HDFS	Hadoop Distributed File System
HTML	Hypertext Markup Language
НТТР	Hypertext Transfer Protocol
IAS	Institut für Automatisierungstechnik und Softwaresysteme
IFU	Institut für Umformtechnik
IP	Internet Protocol
JSON	JavaScript Object Notation
KG	Kommanditgesellschaft
LMA	Levenberg-Marquardt-Algorithmus
LSTM	Long Short-Term Memory
MA	Moving Average
MAPE	Mean Absolute Percentage Error
MB	Megabyte
MLP	Multilayer Perzeptron
MPC	Model Predictive Control
ms	Millisekunden
μs	Mikrosekunden
NARMA	Nonlinear Autoregressive Moving Average
NARX	Nonlinear Autoregressive Model with Exogenous Inputs
OLAP	Online Analytical Processing
RAM	Random Access Memory
RBFN	Radial Basis Function Network
REST	Representational State Transfer
RMSE	Root-Mean-Square Error

RNN	Rekurrentes Neuronales Netz
SELU	Scaled Exponential Linear Units
Seq2Seq	Sequence-to-Sequence
SPS	Speicherprogrammierbare Steuerung
SQL	Structured Query Language
ТВ	Terabyte
ТСР	Transmission Control Protocol
TGS	Ticket Granting Server
vCore	Virtual Core
WS	Werkstück
WZ	Werkzeug
YARN	Yet Another Resource Negotiator
XML	Extensible Markup Language

## Zusammenfassung

Trotz des umfangreichen Expertenwissens von Prozessingenieuren sowie physikalisch nahezu vollständig quantifizierbarer und automatisierter Fertigungsprozesse treten heutzutage entlang mehrstufiger, durch eine hohe Komplexität gekennzeichneter Prozessketten immer wieder unerwartete Qualitätsschwankungen auf [1]. Diese Dissonanzen gehen mit einer Reduktion in der Qualität von Ressource, Prozess und Produkt einher und können eine Fertigung von fehlerhaften, den Sollwert- sowie Toleranzvorgaben des Kunden nicht gerecht werdenden Produkten nach sich ziehen. Eine geringere Effizienz und erhöhte Kosten sind die Folge. Unbekannte anomale Systemdynamiken sind ursächlich für diese Entwicklung. Das Ziel der vorliegenden Arbeit ist daher die Entwicklung einer adaptiven Qualitätsregelung zur Kompensation dieser Effekte.

Da das Expertenwissen bei der Erfassung von Anomalien für verteilte Automatisierungsprozesse an seine Grenzen stößt, wird das Wissen in Form von Prozessmodellen nutzbar gemacht und um einen datengetriebenen Ansatz erweitert. Zur Reduktion von Unsicherheiten der Prozessmodelle werden diese Erweiterungen als adaptive und kontinuierlich lernende Long Short-Term Memory (LSTM) Netze realisiert.

Um Anomalien kompensieren zu können, müssen diese zunächst genau identifiziert und verstanden werden. Es wird daher ein Konzept zur Detektion unbekannter anomaler Systemdynamiken für automatisierte Produktionssysteme entwickelt, sodass deren Integration in eine ganzheitliche Modellierung des Prozessverhaltens möglich ist. Für die nichtlineare, zeitvariante Modellbildung der Anomalien werden die LSTM-Netze eingesetzt. Um auf dieser Basis eine verbesserte Abbildung zukünftigen Prozessverhaltens zu ermöglichen, wird ein kooperatives Präventionsschema vorgestellt, das eine stetige Interaktion von Prozess- und Anomalie-Modell vorsieht. Im letzten Schritt wird die ganzheitliche, kooperative Prädiktion der Prozessdynamiken für eine prädikative Kompensation von Anomalien eingesetzt. Dazu wird ein rekursives Optimierungsschema eingeführt. Das Schema ermittelt optimale Stellstrategien für jedes Produkt entlang der Prozesskette, die zu einer Anpassung der vom Produktionsmitarbeiter eingestellten Sollwerte und somit zur Kompensation von Anomalien führen.

Das Konzept wurde in Form eines Web-basierten Assistenzsystems realisiert. Dieses deckt die vollständige Kette der Datenverarbeitung von der Erfassung semantisch annotierter Daten bis hin zur automatisierten Anpassung der Stellstrategien in den Anlagensteuerungen ab. Eine Evaluierung konnte anhand von drei Prozessketten der Umformtechnik, eine am Institut für Umformtechnik der Universität Stuttgart und zwei im realen industriellen Umfeld der Otto Fuchs KG, durchgeführt werden. Die empirischen Ergebnisse zeigen, dass eine Prävention auftretender Anomalien ermöglicht wird.

## Abstract

Despite the extensive expert knowledge of process engineers as well as physically almost completely quantifiable and thus automated manufacturing process dynamics, unexpected fluctuations still repetitively occur along large-scale and highly complex manufacturing systems [1]. These dissonances go along with a reduction in the quality of resources, processes and products and can result in the production of defective parts that do not meet the requirements of customers. A lower efficiency and increased costs are the result. Unknown anomalous system dynamics are the cause of the problems. Hence, the aim of the present work is the development of an adaptive quality control scheme to realize a compensation of these effects.

Due to the fact that expert knowledge has its limits in the detection of anomalous process dynamics in distributed automation systems, the knowledge is utilized in terms of process models and extended by a data-driven approach. In order to reduce resultant uncertainties in the process models, these model extensions are realized as adaptive and continuously learning Long Short-Term Memory (LSTM) networks.

In order to compensate anomalies, they first have to be precisely identified and understood. Thus, a concept for the detection of unknown anomalous system dynamics for automated production systems is developed, so that the integration of anomalies in a holistic modeling of the process dynamics is possible. The LSTM networks are used for the nonlinear, time-variant modeling. In order to enable an improved prediction of future process outputs based on the identified anomaly model, a cooperative prediction scheme based on a continuous interaction of process and anomaly model is presented. In the last step, the holistic cooperative prediction of process dynamics is used for a predictive compensation of anomalies. For this purpose, a recursive optimization scheme is introduced. The scheme determines optimal control trajectories for each product in the process chain leading to an adjustment of the control variables that have been set by the production worker. Thus, the adaptive adjustments lead to their compensation.

The concept is realized in the form of a web-based assistance system. It covers the complete chain of data processing from the acquisition of semantically annotated data structures to the automated adjustment of the control strategies in the plant control units. Evaluation studies have been carried out using three process chains of metal forming, one at the Institute of Metal Forming Technology of the University of Stuttgart and two in the real industrial environment of Otto Fuchs KG. The empirical results show that a prevention of occurring anomalies can be realized.

## 1 Einleitung

Unerwartete Ausfälle von Anlagenkomponenten und unvorhersehbare Prozessereignisse sind Treiber erhöhter Ineffizienzen in Form von Stillstandzeiten und schwankender Produktqualität in der automatisierten Produktion. Produzierende Unternehmen stehen vor der Herausforderung, trotz entlang der gesamten Wertschöpfungskette auftretender Dissonanzen ein hochwertiges und reproduzierbares Qualitätsergebnis zu gewährleisten. Abweichungen von Sollwert- und Toleranzvorgaben des Kunden bis hin zur Produktion von Ausschuss gehen mit hohen monetären Verlusten einher. Daher werden Lösungen benötigt, die die zunehmende Komplexität entlang hochgradig vernetzter Prozessketten beherrschbar machen, um Effizienz und Qualität von Prozess und Produkt zu steigern und die Kosten zu minimieren. Intelligente Sensor-Netzwerke sowie immer leistungsstärkere Recheneinheiten erlauben den Aufbau umfangreicher Infrastrukturen zur Erfassung und Verwaltung großer Datenmengen und sind Wegbereiter für den Einzug datenanalytischer Konzepte im Kontext der Industrie 4.0 [2]. Im Rahmen dieser Arbeit wurde ein neuartiger, datengetriebener Ansatz zur intelligenten Qualitätsregelung der durch wechselseitige Abhängigkeiten gekennzeichneten Prozessketten entwickelt, der im Folgenden vorgestellt wird.

### 1.1 Datengetriebene Prozessmodellierung

Unabhängig von der Anwendungsdomäne verfügen Prozessingenieure heutzutage über detailliertes Prozesswissen, das in großen Teilen quantifizierbar und in Modellen abbildbar ist. Diese Prozessmodelle haben das Ziel, das reale Prozessverhalten abzubilden und sind Grundlage für eine Kontrolle des Prozessgeschehens hinsichtlich definierter Metriken. Nichtsdestotrotz kommt es immer wieder zu Schwankungen im Prozess, zum Verschleiß von Aktoren und zur Entstehung fehlerhafter Bauteile [3]. Anomale Systemdynamiken, die dem Ingenieur bislang unbekannt sind und daher nicht bei der Modellbildung berücksichtigt wurden, sind ursächlich für diese Effekte [4]. Sie sind durch eine hohe Komplexität gekennzeichnet, die aus einer hohen Dynamik, Nichtlinearität sowie Zeitvarianz resultiert, sodass eine datengetriebene Herangehensweise zur Identifikation notwendig ist, da physikalisch motivierte Modelle nicht existieren [5]. Für die Charakterisierung solch unbekannter, komplexer Systeme bzw. Physiken gibt es unterschiedliche Identifikationsmechanismen, bspw. serielle, parallele oder seriellparallele, sowie verschiedene neuronale Netzstrukturen, bspw. statische oder dynamische, die für ein Identifikationsproblem herangezogen werden können [6]. Vertreter von Netzen, die sich vor allem für die Abbildung statischer Nichtlinearitäten eignen sind bspw. General Regression Neural Networks (GRNN) oder Radial Basis Function Networks (RBFN). Zur Modellbildung dynamischer Nichtlinearitäten eignen sich verschiedene Architekturen von rekurrenten neuronalen Netzen (RNN). Mit den innerhalb der Identifikationsmechanismen genutzten Netzen lassen sich Approximationsaufgaben unterschiedlichster Art realisieren. Auch in Teilen bekannte oder bereits vorstrukturierte Modelle, wie Hammerstein- oder Wiener-Modelle, lassen sich dadurch bestimmen [7]. Um im Rahmen einer datengetriebenen Identifikation bereits bestehendes Expertenwissen nicht erneut lernen zu müssen, gibt es sogenannte Grey-Box-Ansätze, die eine Kombination aus beiden erlauben. Das bestehende, physikalisch motivierte Vorwissen wird dazu um zusätzliches Wissen erweitert, das mit den Netzen gelernt werden konnte.

## 1.2 Problemstellung und Abgrenzung

Im Rahmen der vorliegenden Arbeit stehen Prozesse der diskreten Fertigung im Vordergrund der Betrachtung. Dabei werden nicht einzelne Prozesse, sondern Kompositionen multipler Prozesse untersucht, die in ihrer Gänze mehrstufige Prozessketten darstellen und zur Fertigung definierter Produkte eingesetzt werden. Eine wesentliche Problematik besteht in der Repräsentanz verfügbarer Prozessmodelle der einzelnen Prozesse, da diese ausschließlich numerisch vorliegen. Analytische Modelle sind nicht oder nur mit deutlich reduzierter Genauigkeit vorhanden, sodass eine adäquate Abbildung der realen Systemdynamiken nicht oder nur stark eingeschränkt möglich ist [8]. Die Modelle fokussieren dabei lediglich einen speziellen Prozess ohne die Abhängigkeiten, die potentiell entlang mehrstufiger Prozessketten auftreten können und damit in die Modellbildung zu integrieren wären, zu berücksichtigen [5]. Neben der unzureichenden ganzheitlichen Betrachtung der Prozesskette in aktuellen Ansätzen der Modellbildung stellt die bereits im vorangegangenen Kapitel ausgeführte fehlende Identifikation unbekannter Anomalien wesentliches Problem dar [4]. Echtzeitfähige analytische Modelle ein diskreter Fertigungsprozesse, die im Kontext einer adaptiven Beeinflussung bzw. Regelung der Qualität eingesetzt werden könnten, stehen nicht zur Verfügung. In der industriellen Praxis werden die Prozesseinstellungen durch Erfahrungswissen der Produktionsmitarbeiter determiniert und trotz der vorhandenen Anomalie-Dynamiken nicht oder nur selten in manuellen Vorgängen geändert [9]. Es gibt bislang kaum Ansätze, die auf Grundlage adäquater Prozessmodelle einen Ausgleich von anomalen Systemdynamiken zur Optimierung der Produktqualität zum Ziel haben. Eine Prävention auftretender und durch unbekannte Anomalien verursachter Schwankungen der Oualität entlang mehrstufiger Prozessketten ist daher nicht möglich. Die Probleme können wie folgt zusammengefasst werden:

- Fehlende Kombinationsmöglichkeit des existierenden Vorwissens mit datengetriebenen Methoden zur Reduktion von Modellunsicherheiten.
- Fehlende Berücksichtigung von Abhängigkeiten entlang mehrstufiger Prozessketten für eine ganzheitliche Modellierung aller relevanten Einflüsse auf die Prozesse.
- Fehlende Berücksichtigung und Identifikation unbekannter anomaler Systemdynamiken.
- Fehlender Mechanismus zur Kompensation auftretender Anomalien auf Basis adäquater Prozessmodelle mit dem Ziel der Optimierung der Produktqualität.

Die Problemstellung schneidet regelungs-, informations- sowie fertigungstechnische Themen an und stellt somit eine interdisziplinäre Problematik dar. Nachfolgend wird die Abgrenzung zu eng mit dem vorliegenden Thema verwandten Forschungsbereichen erörtert. Aus dem Bereich der Regelungstheorie sind die Konzepte der Beobachter [10] [11] [12], der Störgrößenunterdrückung [13] [14] sowie der modellbasierten prädikativen Regelung [7] [15] von Bedeutung. Diese werden im Rahmen der Arbeit auf diskrete, mehrstufige Fertigungssysteme übertragen und für die prädikative Kontrolle derselben eingesetzt. Die dafür zu Grunde liegende Prozessmodellierung bedient sich datengetriebener Ansätze. Dabei steht die Untersuchung rekurrenter neuronaler Netzarchitekturen [16] [17] zur Identifikation anomaler Systemdynamiken im Vordergrund. Die intensive Analyse bestehender Ansätze zu Long short-term memory (LSTM) Netzen bildet einen Schwerpunkt [18]. Da es sich bei LSTM um eine spezielle Zellarchitektur handelt, wird diese innerhalb verschiedener Architekturvarianten wie Sequence-to-Sequence (Seq2Seq) Netzen [19] [20] untersucht. Die diskrete Fertigung stellt die Anwendungsdomäne dar. Das Expertenwissen wird für diese Domäne eruiert [4] und eine Übertragbarkeit des Konzeptes auf andere Bereiche innerhalb dieser anvisiert. Auf Basis dieser thematischen Abgrenzung sowie der Beschreibung der Problemstellung lassen sich die nachfolgenden Herausforderungen ableiten.

### 1.3 Herausforderungen

Adaptive, lernfähige Systeme erhalten im Kontext von Industrie 4.0 Einzug in Anwendungen wie der prädiktiven Wartung [21], dem Digitalen Zwilling [22] und weiteren Bereichen [23] [24]. Die Konzeption eines adaptiven Systems zur Qualitätssicherung bringt zahlreiche Herausforderungen mit sich, die bei der Konzeption berücksichtigt werden müssen. Hinsichtlich der in Kapitel 1.2 aufgezeigten Problemstellung ergeben sich die folgenden fünf Herausforderungen:

#### (H1) Kein Vorhandensein echtzeitfähiger analytischer Prozessmodelle.

Um eine adaptive Qualitätsregelung mehrstufiger, aus diskreten Fertigungsschritten bestehender Prozessketten realisieren zu können, müssen die Prozessdynamiken verstanden und echtzeitfähige Prozessmodelle abgeleitet werden. Diese liegen allerdings als numerische Modelle vor, die wegen der örtlichen und zeitlichen Verteilung in der Modellierung sowie hoher Berechnungszeiten nicht unmittelbar im Kontext einer Regelung verwendet werden können [5]. Darüber hinaus ist für die Qualitätsregelung einer mehrstufigen Prozesskette eine eindeutige Rückverfolgung aller Bauteile notwendig, da die Eingriffe in die Prozesse bauteilbezogen erfolgen sollen [25]. Daher besteht die erste Herausforderung darin, die vom Experten erstellten, numerischen Modelle für eine Regelung der Qualität nutzbar zu machen und eine bauteilbezogene Diskretisierung vorzunehmen.

#### (H2) Komplexe Abhängigkeitsstrukturen entlang mehrstufiger Prozessketten.

Eine weitere Problematik ergibt sich durch den Umstand, dass nicht nur ein einzelner Prozess betrachtet wird, sondern eine aus mehreren Prozessen zusammengesetzte Prozesskette kontrolliert werden soll. Zwischen den Prozessen bestehen komplexe Abhängigkeitsbeziehungen, die nur schwer fassbar sind und bei Nichtberücksichtigung zu Schwankungen in der Produktqualität führen können. Die in H1 erwähnten numerischen Modelle beziehen sich allerdings zumeist auf einzelne Prozesse [26]. Aus diesem Grund macht eine Auflösung der Problematik die Erweiterung um datengetriebene Ansätze notwendig. In der Konsequenz besteht die zweite Herausforderung darin, einen datengetriebenen Ansatz zu konzipieren, der Abhängigkeiten berücksichtigt.

#### (H3) Unbekannte anomale Systemdynamiken, die die Aktorsysteme beeinflussen.

Fertigungstechnische Systeme zeichnen sich durch komplexe Aktor-Dynamiken, wie sie bspw. bei hydraulischen Werkzeugsystemen auftreten können, aus [27]. Diese werden von anomalen Systemdynamiken überlagert, die dem Prozessingenieur unbekannt sind und bislang nicht in die numerischen Modelle integriert werden können [5]. Die unbekannten Effekte sind der wesentliche Treiber für Qualitätsschwankungen im System. Sie können z.B. auf Verschleiß, Vibrationseffekte oder Änderungen der Umgebungsparameter zurückgehen. Sie können als stationäre, kurzfristig auftretende Anomalien für dynamische Abweichungen in der Nähe des Sollverhaltens sorgen oder als nicht-stationäre Anomalien zu einer grundsätzlichen Abweichung mit Langzeitcharakter führen. Die dritte Herausforderung besteht darin, die unterschiedlichen Anomalien zuverlässig detektieren und in Modellen abbilden zu können.

#### (H4) Umgang mit Vorhersagefehlern des Prozessverhaltens.

Bedingt durch die fehlende Modellierung von Abhängigkeiten entlang von Prozessketten aus H2 und die fehlende Detektion von anomalen Systemdynamiken aus H3 kommt es zu Fehlern bei der Vorhersage des Prozessverhaltens. Die in infinitesimalem Umfang durch die Natur der Numerik in den numerischen Modellen des Experten bestehenden Unsicherheiten können vernachlässigt werden [28]. Zusätzlich entstehen jedoch aufgrund der Diskretisierung der in H1 geforderten Ableitung bauteilbezogener Modelle weitere Ungenauigkeiten. Die vierte Herausforderung besteht folglich darin, ein Prädiktionsschema zu konzipieren, das durch Berücksichtigung aller genannten Einflüsse zu einer Minimierung von Vorhersagefehlern des Prozessverhaltens führt.

#### (H5) Kompensation von Anomalien und Schwankungen in der Produktqualität.

Wie bereits im Rahmen der Herausforderungen H1 bis H4 erläutert, kann es zu Schwankungen der Produktqualität entlang von Prozessketten kommen. Dies liegt neben den beschriebenen Problematiken an dem Umstand, dass Fertigungssysteme basierend auf Erfahrungswissen und zumeist mit konstanten Einstellungen betrieben werden. Um Anomalien und daraus entstehende Schwankungen der Produktqualität kompensieren zu können, ist eine adaptive Einstellung von Stellgrößen für jedes zu produzierende Bauteil notwendig [29]. Die letzte Herausforderung besteht somit darin, ein Schema zur adaptiven Stellgrößenanpassung auf Basis eines, durch die Herausforderungen H1 bis H4 entwickelten, lernfähigen Modellierungsansatzes zu konzipieren.

#### 5

### 1.4 Anforderungen an die Konzeption

Aus den im vorherigen Kapitel aufgezeigten Herausforderungen lassen sich Anforderungen ableiten, die das im Rahmen dieser Arbeit entwickelte Konzept erfüllen muss, um den Herausforderungen gerecht zu werden und die beschriebene Problemstellung zu lösen. Analog zu den Herausforderungen werden im Folgenden die daraus abgeleiteten Anforderungen aufgeführt.

#### (A1) Nutzung von bestehendem Expertenwissen im Rahmen der Modellbildung.

Um die vom Experten erstellten, numerischen Simulationsmodelle für eine Qualitätsregelung nutzbar zu machen und eine bauteilbezogene Diskretisierung zu ermöglichen, muss das Konzept einen Ansatz für die Ableitung von Metamodellen enthalten, deren Ein- und Ausgangsdatenräume diskretisierte und bauteilbezogene Prozessparameter umfassen. Die aus den numerischen Modellen abgeleiteten Metamodelle sollen unter minimalem Informationsverlust generiert werden. Das Konzept muss zusätzlich eine Lösung für die eindeutige Zuordnung von Prozessparametern zu Bauteilen beinhalten.

#### (A2) Berücksichtigung von Abhängigkeiten zwischen Prozessschritten.

Um Abhängigkeiten entlang von Prozessketten zu berücksichtigen, muss das Konzept einen Ansatz zur Modellierung von Verknüpfungen zwischen Prozessen enthalten. Diese Verknüpfungen müssen die diskreten Prozessmodelle der sequentiell stattfindenden Prozesse einer Prozesskette miteinander verbinden, sodass ein ganzheitliches Prozesskettenmodell erstellt werden kann. Dabei müssen die logistischen Transportwege genau nachgebildet werden. Die Auswirkungen von Ausgängen vorheriger Prozessschritte auf die Ausgänge nachfolgender Prozesse soll aus Daten extrahiert werden.

#### (A3) Detektion und Charakterisierung anomaler Prozessdynamiken.

Um die unterschiedlichen Anomalien zuverlässig detektieren und charakterisieren zu können, muss das Konzept einen Ansatz zur Anomalie-Detektion beinhalten. Der Mechanismus zur Anomalie-Detektion muss stationäre, kurzfristig auftretende Anomalien genauso erfassen können wie nicht-stationäre Langzeitabhängigkeiten. Auf Basis des Detektionsergebnisses muss ein Rückschluss auf den aktuellen Zustand des Systems möglich sein und formalisiert werden können. Zusätzlich soll durch die Detektionsergebnisse sollen als Datengrundlage für die Erzeugung eines Modells zur Abbildung und Vorhersage des Anomalie-Verhaltens genutzt werden können.

#### (A4) Ganzheitliche und lernfähige Prädiktion des Prozess- und Anomalie-Verhaltens.

Um eine Prädiktion des Prozessverhaltens unter Berücksichtigung aller bekannten und unbekannten Einflüsse durchführen zu können, muss das Konzept einen Ansatz für ein ganzheitliches und lernfähiges Prädiktionsschema beinhalten. Es müssen Abhängigkeiten entlang von Prozessketten, Modellunsicherheiten und unbekannte anomale Systemdynamiken modellieroder schätzbar und in die Prädiktion integrierbar sein. Es sollen sowohl stationäre, kurzfristig auftretende Abhängigkeiten als auch nicht-stationäre Langzeitabhängigkeiten probabilistisch vorhergesagt werden können. Der Vorhersagehorizont soll flexibel einstellbar und das dem Prädiktionsschema zu Grunde liegende Prozessmodell lernfähig sein.

#### (A5) Nutzerassistenz bei der Kompensation von Anomalien zur adaptiven Qualitätsregelung.

Um eine adaptive Stellgrößenanpassung auf Basis des zu entwickelnden Modellierungsansatzes und Prädiktionsschemas zu realisieren, muss das Konzept über einen Ansatz für die Kompensation von Anomalien verfügen. Die Prädiktion des Prozess- und Anomalie-Verhaltens soll genutzt werden, um Maßnahmen für eine zukünftige Unterdrückung erwarteter Anomalien abzuleiten. Die Anpassung der Stellgrößen soll auf detektierte Effekte rückführbar sein, die ursprüngliche Regeleinheit selbst nicht verändern und dieser bzw. dem Produktionsmitarbeiter als Assistenzfunktion zur Verfügung stehen. Dabei soll die Kompensation bauteilspezifisch erfolgen, sodass die Stellgrößenanpassungen für jedes Bauteil optimiert werden. Das Konzept soll zu einer Steigerung der Produktqualität führen.

## 1.5 Zielsetzung der Arbeit

Basierend auf den zuvor beschriebenen Anforderungen kann die Zielsetzung dieser Arbeit abgeleitet werden. Das übergeordnete Ziel ist die Entwicklung eines Konzeptes zur adaptiven Kompensation von Anomalien auf Basis eines ganzheitlichen Modellierungsansatzes für mehrstufige Prozessketten der diskreten Fertigung. Dies soll im Kontext dieser Arbeit für umformtechnische Prozesse realisiert werden, da sich diese durch eine hohe physikalische Komplexität und durch Abhängigkeitsbeziehungen entlang von Prozessketten auszeichnen und daher für eine Evaluierung des Konzeptes geeignet sind. Die Prozesse sind durch eine zeitliche und räumliche Verteilung von thermischen Effekten und Materialeigenschaften gekennzeichnet. Anderseits bestehen zusätzlich Abhängigkeiten zwischen den Prozessen der mehrstufigen Prozessketten, die bspw. durch Chargenschwankungen zu Tage treten und bislang nur durch Erfahrungs- bzw. Expertenwissen beherrschbar sind. Die Analyse von Prozessdaten soll helfen, diese Phänomene in Modellen abzubilden und für eine Optimierung der Qualität von Prozess und Produkt nutzbar zu machen. Diese adaptive Qualitätsregelung soll einen Beitrag zur nachhaltigen Steigerung der Produktqualität in Form einer minimierten Ausschussproduktion leisten. Gängige industrielle Praxis ist heutzutage trotz hochdynamischer Prozesse die manuelle, erfahrungsbasierte und dadurch situationsunabhängige Einstellung der Prozessparameter, die Schwankungen im Prozessergebnis nach sich ziehen [3] [9]. Im Fokus steht daher die Konzeption eines Mechanismus, der den hohen Prozessdynamiken Rechnung trägt und die bestehenden Prozesssituationen adaptiv erfasst sowie automatisiert entsprechende Einstellungen vornimmt, die zu einer Kompensation von Anomalien und folglich zu einer Reduktion auftretender Schwankungen führen sollen. Im Hinblick auf die zu diesem Zweck zu konzipierende ganzheitliche Prozessmodellierung wird darauf abgezielt, einen Ansatz zu entwickeln, der bestehendes Expertenwissen einbezieht und Abhängigkeiten entlang von Prozessketten sowie unbekannte anomale Prozessdynamiken berücksichtigt. Dazu wird ein zuverlässiger Identifikationsmechanismus der Anomalien angestrebt, sodass eine prädikative Modellierung derselben und ein präventives Eingreifen in die Prozesskette zur Steigerung der Produktqualität ermöglicht wird. Die Güte der Detektion, der Prädiktion sowie der Kompensation von Anomalien soll im Rahmen der Evaluierung eruiert werden. Die einzelnen Teil-Methoden zur Detektion, Prädiktion und Kompensation sollen dabei separat sowie in Kooperation anhand definierter Anomalie-Szenarien untersucht werden. Die auf Basis von zwei unabhängigen Prozessketten gewonnenen empirischen Ergebnisse und die daraus abgeleiteten Erkenntnisse sollen die Funktionsfähigkeit des Konzepts sowie die Erfüllung der Anforderungen belegen.

### 1.6 Aufbau der Arbeit

Die vorliegende Arbeit ist gemäß der in Abbildung 1.1 visualisierten Struktur in sieben Kapitel gegliedert. Im zweiten Kapitel wird ein Überblick über die relevanten Grundlagen geben. Dieser beginnt mit den grundlegenden Konzepten der zeitvarianten nichtlinearen Prozessmodellierung in dessen Kontext verschiedene rekurrente Netz- und Zellarchitekturen, wie LSTM, erläutert werden. Im zweiten Teil des Kapitels wird auf die für eine Kompensation anomaler Systemdynamiken relevanten Grundlagen, wie bspw. Beobachter, eingegangen.

Im dritten Kapitel erfolgt eine detaillierte Untersuchung des Stands der Wissenschaft und Technik. Das Kapitel gliedert sich wie das Grundlagenkapitel in die Prozessmodellierung und die darauf aufbauende Kompensation anomaler Systemdynamiken. Der erste Abschnitt fokussiert aktuelle Arbeiten zur Detektion von Anomalien sowie deren Modellierung und Prädiktion mit neuronalen Netzen. Neben Arbeiten zu RNN und LSTM werden zahlreiche Varianten und Neuentwicklungen von Netz- und Zellarchitekturen untersucht. Im zweiten Abschnitt werden bauteilbezogene Qualitätsregelungen, Ansätze für die Kontrolle mehrstufiger Prozessketten sowie modellbasierte prädikative und adaptive Methoden untersucht. Das Kapitel schließt mit einer Bewertung der Ansätze und der Darstellung des aktuellen Forschungsbedarfs.

Im vierten Kapitel wird das Konzept zur Qualitätssicherung auf Basis der Kompensation anomaler Systemdynamiken vorgestellt, das die aufgezeigte Forschungslücke schließt. Zunächst wird die entwickelte Modellierung einzelner Prozesse und mehrstufiger Prozessketten vorgestellt, die bestehendes Expertenwissen um neuronale Netze erweitert. Darauf aufbauend wird die adaptive Qualitätssicherung erörtert. Diese berücksichtigt neben Abhängigkeiten entlang von Prozessketten zusätzlich unbekannte anomale Prozessdynamiken, deren Detektion mit dem entwickelten neuronalen Beobachtungsmechanismus im Anschluss vorgestellt wird. Danach erfolgt die Beschreibung des konzipierten Prädiktionsschemas zur kooperativen Vorhersage von Prozess- und Anomalie-Verhalten, bevor der darauf aufbauende, rekursive Ansatz zur prädikativen Stellgrößenoptimierung erläutert wird. Durch die Anpassung der Stellgrößen wird eine Kompensation anomaler Prozessdynamiken erreicht. Abschließend wird dazu die Umsetzung der Anpassung zum Ziel der adaptiven Qualitätssicherung thematisiert.



Abbildung 1.1: Aufbau der Arbeit

Das Konzept wurde in Form eines adaptiven Assistenzsystems zur Qualitätssicherung realisiert, dessen Aufbau und Funktionsweise im fünften Kapitel beschrieben wird. Dabei wird zunächst auf die Datenerfassung, -verarbeitung und -verwaltung der großen Datenmengen eingegangen. Die Realisierung als Assistenzsystem gliedert sich in die Komponenten zur Detektion von Anomalien, zur kooperativen Prädiktion von Prozess- und Anomalie-Verhalten sowie zur Kompensation von Anomalien auf. Abschließend wird die Web-basierte Benutzerschnittstelle, über die die Funktionalitäten zur Verfügung gestellt werden, vorgestellt.

Im sechsten Kapitel werden die erzielten empirischen Ergebnisse vorgestellt. In diesem Kontext wird zunächst auf die beiden Prozessketten zur Evaluierung eingegangen. Die in Kooperation mit dem Institut für Umformtechnik (IFU) der Universität Stuttgart aufgebaute Modellfabrik dient zur Nachbildung ungelöster Forschungsprobleme. Die zweite Prozesskette bei der Otto Fuchs KG wird für die Übertragung der Erkenntnisse aus der Modellfabrik herangezogen. Die Ergebnisse zeigen hohe Detektions- und Prädiktionsgüten für kurzfristig auftretende, stationäre Anomalien sowie für nicht-stationäre Langzeitabhängigkeiten. Auf dieser Basis konnten mit dem rekursiven Optimierungsschema Stellgrößenanpassungen abgeleitet und zur adaptiven Beeinflussung der Produktqualität eingesetzt werden. Kapitel sieben fasst die Ergebnisse der Arbeit zusammen und schließt diese mit einem Ausblick über potentielle weiterführende Forschungsaktivitäten ab.

## 2 Grundlagen

In diesem Kapitel werden die für diese Arbeit relevanten theoretischen Grundlagen vorgestellt. Dabei stehen die grundlegenden Methoden zur nichtlinearen Modellbildung technischer Systeme sowie eine darauf aufbauende Beherrschung derselben im Fokus der Betrachtungen. Im Hinblick auf die Modellbildung nichtlinearer Systeme gibt es im Gegensatz zu linearen Systemen noch keine einheitliche Theorie. Nichtlineare Systeme sind in den meisten Fällen zudem dynamisch, zeitvariant und durch multiple Ein- sowie Ausgänge gekennzeichnet. Linearisierungen um definierte Arbeitspunkte sind entweder aufgrund des hohen Grades der Nichtlinearität oder durch sich dynamisch ändernde Umgebungsbedingungen und damit einhergehende Verschiebungen von Arbeitspunkts zumeist ungenau. Es bestehen in diesem Kontext intensive Forschungsaktivitäten, die die Entwicklung von Ansätzen der Modellbildung hinsichtlich bestimmter Klassen nichtlinearer Prozesse anstreben [7] [30]. Bevor diese Ansätze jedoch erörtert werden, wird genauer auf die Begrifflichkeit des "Modells" eingegangen. Ein Modell ist ein möglichst genaues Abbild einer Realität. Häufig wird durch gewisse vereinfachende Grundannahmen eine Approximation definierter Granularität erzielt, wie es bspw. bei der Auswahl der Charakteristiken finiter Elemente bei numerischen Modellen der Fall ist. Ist die Granularität des Modells festgelegt, muss eine passende Modellstruktur bestimmt werden, die in einem letzten Schritt durch eine möglichst exakte Parametrierung feinjustiert wird [31]. In Abhängigkeit davon, wie viel Vorwissen besteht, können die folgenden drei Modelltypen unterschieden werden:

- White-Box-Modelle: Das Wissen über den zu Grunde liegenden Prozess ist vollständig vorhanden und quantifizierbar. Das Modell kann durch eine theoretische Analyse, bspw. durch das Aufstellen von nichtlinearen Differentialgleichungssystemen, abgeleitet werden. Sowohl der Modellstruktur als auch den Parametern kann eine physikalische Bedeutung zugeordnet werden. Die Modelle sind sehr genau, es erfordert jedoch einen erhöhten Zeitaufwand für deren Erstellung im Rahmen der theoretischen Analyse [7].
- Grey-Box-Modelle: Hier besteht Vorwissen hinsichtlich aller oder einzelner Bereiche der Modellstruktur. Häufig kann auf Basis von Expertenwissen eine Modellstruktur antizipiert werden, die im Rahmen einer experimentellen Analyse verifiziert und parametriert wird. Der Ansatz setzt sich aus parametrischer und nichtparametrischer Modellierung zusammen und kann durch den Einfluss des Expertenwissens physikalisch motiviert sein. Die identifizierten Parameter können jedoch nicht direkt physikalischen Größen zugeordnet werden. Der Modelltyp ist ein Hybrid aus White- und Black-Box-Modell [32].
- Black-Box-Modelle: Von diesem Modelltyp ist bei vollständig fehlenden Kenntnissen zu Modellstruktur und Parametrierung die Rede. Im Rahmen einer mit erfassten Prozessdaten durchgeführten experimentellen Analyse können Modelle abgeleitet werden, die das Eingangs/Ausgangsverhalten des Prozesses abbilden (E/A-Modelle). Eine physikalische Bedeutung des nichtparametrischen Modells im Hinblick auf Struktur und Parameter ist

nicht gegeben. Interne Approximationsmechanismen, wie bspw. Gewichte bei neuronalen Netzen, kapseln die physikalischen Parameter indirekt. [7].

Im Kontext der vorliegenden Arbeit spielt der Grey-Box-Ansatz die zentrale Rolle. Nachfolgend wird näher auf den nichtparametrischen Modellierungsanteil des Modelltyps und entsprechende datengetriebene Methoden eingegangen, die das vorhandene Expertenwissen erweitern.

### 2.1 Nichtlineare zeitvariante Prozessmodellierung

Zur nichtparametrischen Modellierung unbekannter nichtlinearer Prozessdynamiken wurden in Jahren allem neuronale Netze untersucht den letzten vor Durch die guten Approximationseigenschaften statischer und dynamischer Zusammenhänge, die Fähigkeit zur Erkennung von Korrelationen und Mustern sowie der effizienten Lernmethoden eignen sie sich zur Modellierung der hochgradig nichtlinearen Prozesse, die heutzutage in der diskreten Fertigung anzutreffen sind [33]. Sind die betrachteten Prozesse zusätzlich zeitvariant, kommen zumeist RNN zum Einsatz. Diese können Muster aus Zeitreihen extrahieren, in eine zeitvariante Modellstruktur überführen und diese bei Änderungen der Systemdynamiken kontinuierlich adaptieren [34]. Dadurch eignen sich RNN zur Modellierung nichtlinearer zeitvarianter Systeme, die stetigen Änderungen bspw. in Form von Anomalien ausgesetzt sind. Bevor die für diese Arbeit relevanten Typen von RNN erläutert werden, wird zunächst genauer auf Anomalien eingegangen.

#### 2.1.1 Anomalien

Der Begriff der Anomalie stammt aus dem Altgriechischen und bedeutet so viel wie ,von der Regel abweichend' oder ,Unregelmäßigkeit' [35]. Anomalien sind in verschiedenen Bereichen von Bedeutung, wie bspw. der Medizin, und kommen dort in diversen Kontexten vor [36]. Im Allgemeinen werden Wahrscheinlichkeitsdichten für die zu Grunde liegenden Kenngrößen ermittelt und definierte Perzentile als anomales Verhalten deklariert. Eine stochastische Ausreißer-Erkennung dieser Art entspricht der grundlegenden Vorgehensweise zur Detektion von Anomalien [37]. Bezogen auf technische Systeme können nach [38] neben den klassischen Ausreißern in Form von Punktanomalien auch kontextuelle sowie kollektive Anomalien auftreten. Abbildung 2.1 zeigt die drei Anomalie-Typen am einfachen Beispiel einer univariaten Zeitreihe.

Kontextuelle Anomalien beziehen sich auf einzelne Datenpunkte, die für sich betrachtet normal sein können, in dem vorhandenen Kontext jedoch auf anomales Verhalten hindeuten. Bei multivariaten Zeitreihen kann ein Kontext der Größe N als eine Vereinigung von Datenvektoren aufgefasst werden. Mit D wird die Menge aller Kontexte bezeichnet:

$$K_{t} = \bigcup_{i=t-N}^{t} \underline{x}_{i} \quad und \quad D = \{K_{1}, \dots, K_{t-1}\}$$
(1)

Zur Beschreibung einer kontextuellen Anomalie wird ein distanzbasierter Abgleich des aktuellen Datenvektors  $\underline{x}_t$  und seines Kontextes  $K_t$  mit allen historischen Kontexten der Menge D durchgeführt.

$$f(v(K_t)) = \begin{cases} v(K_t) > \tau_t: Anomalie\\ v(K_t) \le \tau_t: K_t \bigcup D \end{cases} \quad mit \, v(K_t) = \min_{K_t} (dist(K_t, K_t)) \, \forall \, K_t \, \epsilon \, D \qquad (2) \end{cases}$$

Beim Überschreiten eines dynamischen Grenzwertes kann von einer kontextuellen Anomalie ausgegangen werden. Andernfalls wird die Datenbasis erweitert [39].



Abbildung 2.1: Illustration der drei Anomalie-Typen univariater Zeitreihen nach [40]

Kollektive Anomalien beziehen sich im Gegensatz dazu auf eine Sequenz an mehreren Datenpunkten, bei denen jeder einzelne Datenpunkt für sich betrachtet normal sein kann, das Ensemble jedoch auf eine Anomalie schließen lässt. Dabei wird auf den restlichen Datensatz Bezug genommen. In [41] wird für multivariate Zeitreihen ein Tupel aus dem Stream  $S_i$ , der Zeitspanne zwischen dem aktuellen Zeitpunkt und dem potentiellen Anfangszeitpunkt der kollektiven Anomalie  $[t_a, t]$  und dem Grad der Anomalie  $N_i$  zur Charakterisierung kollektiver Anomalien vorgeschlagen:

$$\langle S_{i}, [t_{a}, t], N_{i}(t) \rangle \quad mit \ N_{i}(t) = \sum_{\tau=t}^{t_{a}} N_{i,\tau}(t) \ und \ N_{i,\tau}(t) = N_{i,\tau} e^{-\lambda(t-\tau)} \ und \ t > t_{a}$$
(3)

Der Grad  $N_i$  dient zur Anomalie-Bewertung der Daten der einzelnen Streams *i*. Die Bedeutung eines einzelnen Datenpunktes, der zum Zeitpunkt  $\tau$  erfasst wurde, wird dabei als mit der Zeit *t* exponentiell abklingend modelliert. Eine kollektive Anomalie ist vorhanden, wenn folgende Bedingung gilt:

$$N_i(t) > 2 * \left( N_{median}(t) - N_{min}(t) \right)$$
(4)

Durch die Verwendung des Medians zur Grenzwertbildung entsteht implizit eine dynamische Charakterisierung kollektiver Anomalien. Die Verwendung des Mittelwertes ist nicht geeignet. Alternativ könnten Perzentile zur Bestimmung herangezogen werden.

#### 2.1.2 Rekurrente neuronale Netze

Zur Modellierung nichtlinearer zeitvarianter Systemdynamiken, die sowohl normalem als auch anomalem Prozessverhalten entsprechen können, werden RNN herangezogen. Diese orientieren sich an seinem biologischen Vorbild. Biologische neuronale Netze bestehen aus einem komplexen Konstrukt an Neuronen, die aktiviert werden und zu feuern beginnen, wenn die Eingangssignale einen Schwellwert überschreiten. Das konventionelle mathematische Modell dieses biologisch hochkomplexen Vorgangs sieht einfache Aktivierungsfunktionen vor [42]. Ein Neuron verarbeitet dabei die gewichtete Summe aller Eingangssignale bzw. -vektoren und einen Bias-Wert. Mit Hilfe der Aktivierungsfunktion wird die Ausgabe eines Neurons, bspw. des Neurons j der Netzschicht l zum Zeitpunkt t, erzeugt:

$$h_{j}^{l}(t) = f_{j}^{l} \left( \sum_{i=1}^{n_{l-1}} w_{ij}^{l-1} x_{i}^{l-1}(t) + w_{bj}^{l-1} b_{j}^{l-1} \right)$$
(5)

Dabei stellt  $f_j^l(\cdot)$  die Aktivierungsfunktion des betrachteten Neurons dar und  $w_{ij}^{l-1}$  sowie  $w_{bj}^{l-1}$  die Netzgewichte zwischen Neuron *j* und allen  $n_{l-1}$  Neuronen der vorangegangenen Schicht l-1 sowie dem Bias-Wert. Die Eingänge des Neurons entsprechen den Ausgängen aller  $n_{l-1}$  Neuronen der vorangegangenen Schicht  $x_i^{l-1}(t)$  und sind genauso wie die Ausgabe  $h_j^l(t)$  abhängig von der Zeit. Der Bias wird mit  $b_j^{l-1}$  bezeichnet. Als Aktivierungsfunktion wird häufig der Tangens hyperbolicus verwendet. Die Beschreibung eines Neurons kann zur Darstellung eines aus multiplen Schichten bestehenden Netzes erweitert werden. Das Netz kann durch die Gewichte charaketrisiert werden, die sich als Matrizen darstellen lassen:

$$W^{l-1} = \begin{pmatrix} w_{11}^{l-1} & \cdots & w_{n_{l-1}1}^{l-1} & w_{b1}^{l-1} \\ w_{12}^{l-1} & \cdots & w_{n_{l-1}2}^{l-1} & w_{b2}^{l-1} \\ \vdots & \ddots & \vdots & \vdots \\ w_{1n_{l}}^{l-1} & \cdots & w_{n_{l-1}n_{l}}^{l-1} & w_{bn_{l}}^{l-1} \end{pmatrix} W^{l} = \begin{pmatrix} w_{11}^{l} & \cdots & w_{n_{l}1}^{l} & w_{b1}^{l} \\ w_{12}^{l} & \cdots & w_{n_{l}2}^{l} & w_{b2}^{l} \\ \vdots & \ddots & \vdots & \vdots \\ w_{1n_{l+1}}^{l} & \cdots & w_{n_{l}n_{l+1}}^{l} & w_{bn_{l+1}}^{l} \end{pmatrix}$$
(6)

Dabei ist in Schicht l - 1 eine Anzahl an  $n_{l-1}$  Neuronen, in Schicht l eine Anzahl an  $n_l$  Neuronen usw. vorhanden. Zur Charakterisierung des gesamten Netzes mit  $l_{max}$  Schichten können die Matrizen in folgender Gesamtmatrix zusammengefasst werden:

$$W = \begin{pmatrix} W^{l_{max}} & 0 & \cdots & 0 \\ 0 & W^{l_{max}-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & W^{0} \end{pmatrix}$$
(7)

Um die Berechnung der Ausgabe aus Gleichung 5 auf das gesamte Netz zu übertragen, kann folgende vektorielle Schreibweise gewählt werden:

$$\underline{h}^{l}(t) = \underline{f}^{l} \left( W^{l-1} \, \underline{q}^{l-1}(t) \right) \, mit \, \, \underline{q}^{0}(t) = \left[ \underline{x}^{0}(t) \, b \right] \tag{8}$$

Die vektorielle Aktivierungsfunktion erzeugt aus dem Eingangsvektor der gewichteten Summen einen Vektor für die Ausgänge aller Neuronen. Die rekursive Berechnungsvorschrift wird durch den Netzeingangsvektor  $\underline{x}^0(t)$  initialisiert und kann über eine variable Anzahl an Schichten propagiert werden. Die Matrizen müssen dabei auch die Gewichte der innerhalb des Netzes vorkommenden Rückkopplungen beinhalten. Hinsichtlich Gleichung 8 gilt bei RNN für l > 0somit:

$$q^{l}(t) = \left[\underline{x}^{l}(t) \ \underline{h}^{l}(t) \ b\right]$$
(9)

Rückkopplungen können direkt bei einem Neuron als Rückführung des Ausgangs zum Eingang, indirekt zwischen Neuronen unterschiedlicher Schichten oder lateral zwischen Neuronen derselben Schicht realisiert sein [33]. Die Rückkopplungen müssen nicht unmittelbar ausgeführt werden, sondern können mit Verzögerungsgliedern variablen Grades versehen sein. Dieser Umstand erlaubt die flexible Abbildung zeitvarianten Verhaltens. Die Varianten der Rückkopplung sind in Abbildung 2.2 visualisiert.



Abbildung 2.2: RNN mit direkten, indirekten und lateralen Rückkopplungen (links) und aufgefaltetes Neuron eines RNN (rechts)

Außerdem ist dort ein einzelnes Neuron mit direkter Rückkopplung entfaltet dargestellt. Durch die zeitlich versetzte Rückführung eines Ausgangs entsteht eine rekursive Abhängigkeit innerhalb des Netzes, die dazu führt, dass der aktuelle Ausgangswert des rekurrenten Neurons von der Menge aller historischen Eingänge bestimmt ist und der aktuelle Eingang die Änderung des Ausgangs bestimmt [7]. Die skalare Darstellung von Gleichung 8 mündet in einer erweiterten Darstellung von Gleichung 5:

$$h_j^l(t) = f_j^l \left( \sum_{i=1}^{n_{l-1}} w_{ij}^{l-1} x_i^{l-1}(t) + \sum_{k=l}^{l_{max}} \sum_{p=1}^{n_k} w_{pj}^k h_p^k(t - d_{kp}) + w_{bj}^{l-1} b_j^{l-1} \right)$$
(10)

Die Ausgänge werden mit einer Verzögerung von  $d_{kp}$  Zeitschritten rückgeführt. Die Verzögerung hängt damit genauso wie die Gewichte vom rückgeführten Neuron p aus der entsprechenden Schicht k ab. Wird die Rekursion für einen Zeitschritt aufgelöst ergibt sich:

$$h_{j}^{l}(t) = f_{j}^{l} \left( \sum_{l=1}^{n_{l-1}} w_{lj}^{l-1} x_{l}^{l-1}(t) + \sum_{k=l}^{l_{max}} \sum_{p=1}^{n_{k}} w_{pj}^{k} f_{j}^{l} \left( \sum_{l=1}^{n_{l-1}} w_{lj}^{l-1} x_{l}^{l-1} \left( t - d_{kp} \right) + \cdots \right) \right)$$
(11)

Dadurch wird ersichtlich, dass weiter in der Vergangenheit liegende Eingangswerte eine sich stetig verringernde Gewichtung erfahren und damit 'in Vergessenheit' geraten [43]. Dieser Vanishing-Gradient-Effekt von RNN ist Gegenstand zahlreicher aktueller Untersuchungen und hat zur Entwicklung von Gating-Mechanismen geführt [44].

#### 2.1.3 Long short-term memory

Gating-Mechanismen ersetzen die einfachen Aktivierungsfunktionen innerhalb der Neuronen durch ein komplexes Konstrukt an Berechnungsvorschriften. Ihr gemeinsames Ziel ist die Unterbindung des ,Vergessens' im Rahmen des Vanishing-Gradient-Effekts und der Aufbau eines flexiblen Gedächtnisverhaltens, das sowohl Kurzzeit- als auch Langzeitabhängigkeiten kapseln kann. Bei allen Gating-Mechanismen ist übereinstimmend, dass sie einen Zustandsvektor für das Neuron definieren, der über die ,Gates' geändert werden kann. Der Gating-Mechanismus, der sich in seiner originalen sowie in abgewandelten Formen primär durchgesetzt hat, ist der der LSTM-Zelle [18]. LSTM verfügt über drei Gates, die mit dem Zellzustand interagieren. Darüber wird das Hinzufügen und Löschen von Informationen sowie die Erzeugung des Ausgangs realisiert. Der Informationsfluss ist in Abbildung 2.3 visualisiert. Dieser steuert den Prozess des Vergessens und ist in der Lage, sowohl Kurzzeit- als auch Langzeitabhängigkeiten zu erfassen.



Abbildung 2.3: LSTM-Zelle (links) im Vergleich zur GRU-Zelle (rechts)

Der Zellzustandsvektor  $\underline{z}(t)$  verwaltet die für die Ausgabe relevanten Informationen. Er wird durch den Forget-Vektor f(t) und den Add-Vektor  $\underline{n}(t)$  iterativ angepasst:

$$\underline{f}(t) = \sigma \Big( W_1 \big\{ \underline{x}(t); \underline{h}(t-1); \underline{b}_1 \big\} \Big)$$
(12)

$$\underline{n}(t) = \underline{i}(t) * \sigma \left( W_1 \{ \underline{x}(t); \underline{h}(t-1); \underline{b}_1 \} \right)^{-1}$$
(13)

wobei  $W_1$  eine Gewichtungsmatrix ist,  $\underline{b}_1$  ein Bias-Vektor,  $\underline{h}(t-1)$  ein rückgeführter Ausgang und  $\underline{x}(t)$  der aktuelle Eingangsvektor. Die in der vorliegenden Arbeit verwendete Architektur sieht eine inverse Verknüpfung der beiden Gates vor. Der dem Add-Vektor zu Grunde liegende Vektor  $\underline{i}(t)$  basiert auf einer separaten Gewichtsmatrix:

$$\underline{i}(t) = \sigma \Big( W_2 \big\{ \underline{x}(t); \underline{h}(t-1); \underline{b}_2 \big\} \Big)$$
(14)

Die iterative Anpassung des Zellzustandsvektors durch die beiden Gates kann wie folgt beschrieben werden:

$$\underline{z}(t) = f(t) \odot \underline{z}(t-1) + \underline{n}(t)$$
(15)

Über das Output-Gate wird in Abhängigkeit des Zellzustands der Ausgang des Neurons erzeugt. Im Gegensatz zu klassischen RNN kapselt der Ausgang alle vergangenen Informationen nach Relevanz und nicht nach einem temporalen Kriterium.

$$\underline{h}(t) = tanh(\underline{z}(t)) * \sigma(W_3\{\underline{x}(t); \underline{h}(t-1); \underline{b}_3\}$$
(16)

Daher dominieren kürzlich erfasste Eingänge die Ausgabe nicht notwendigerweise in hohem Maße, da Informationen über wichtige, durchaus auch weit zurückliegende Ereignisse innerhalb des Zellzustands gespeichert und entsprechend gewichtet werden. Ein weiterer Gating-Mechanismus ist der der Gating Recurrent Unit (GRU), der in Abbildung 2.3 dargestellt ist [45]. Diese verfügt im Vergleich zu LSTM über einen vereinfachten Zellaufbau, der ebenfalls über ein Gating-System operiert, jedoch lediglich über ein Update-Gate  $\underline{u}(t)$  und Reset-Gate  $\underline{r}(t)$  verfügt.

$$\underline{u}(t) = \sigma \Big( W_1 \{ \underline{x}(t); \underline{z}(t-1); \underline{b}_1 \} \Big)$$
(17)

$$\underline{r}(t) = \sigma \Big( W_2 \{ \underline{x}(t); \underline{z}(t-1); \underline{b}_2 \} \Big)$$
(18)

Der wesentliche Unterschied zu LSTM besteht in der Tatsache, dass der Zellzustand über das Reset-Gate bei jeder Iteration vollständig überarbeitet und mit aktuellen Informationen beschrieben werden kann.

$$\underline{\tilde{z}}(t) = tanh\left(W_3\{\underline{x}(t); \underline{b}_3\} + W_4\{\underline{r}(t) \odot \underline{z}(t-1); \underline{b}_4\}\right)$$
(19)

Bei LSTM begrenzt hingegen ein mit dem Add-Gate verknüpfter Mechanismus bei jeder Iteration den Änderungsgradienten, der realisiert werden kann. Dadurch besteht bei LSTM keine Möglichkeit, vergangene Informationen vollständig zu verwerfen, bei GRU hingegen schon. Der neue Zustand wird dabei wie folgt berechnet:

$$\underline{\underline{z}}(t) = \underline{\underline{u}}(t) \odot \underline{\underline{\tilde{z}}}(t) + \left(1 - \underline{\underline{u}}(t)\right) \odot \underline{\underline{z}}(t-1)$$
(20)

Dadurch sollen neben Langzeitabhängigkeiten auch kurzfristige Abhängigkeiten präzise modelliert werden. Die Ausgabe kann direkt in Form des neuen Zellzustands erzeugt oder über eine nichtlineare Abbildung desselben generiert werden.

#### 2.1.4 Autoencoder

Die im vorherigen Kapitel beschriebenen Zellarchitekturen können in verschiedene Architekturen von Netzen integriert werden. So kann durch die Integration von LSTM-Zellen in eine Autoencoder-Netzarchitektur ein LSTM-Autoencoder aufgebaut werden, der nicht nur dimensionsreduzierende Eigenschaften aufweist, sondern zusätzlich zeitvariantes Verhalten modellieren kann. Autoencoder realisieren, wie viele klassische statistische Verfahren zur Dimensionsreduktion (genannt sei an dieser Stelle bspw. die Hauptkomponentenanalyse), eine Projektion des Eingangsdatenraumes in einen niederdimensionalen Unterraum unter Minimierung des Informationsverlustes bei gleichzeitiger Maximierung des Reduktionsgrades. Neben dem Encoder, der diese Kompression leistet, verfügen sie zusätzlich über einen Decoder, der eine Dekompression zurück auf die Eingangsdimension vornimmt [46]. Für den Encoder gilt:

$$\underline{k}(t) = \underline{f}_E\left(\underline{x}(t)\right) = \sigma\left(W_E \,\underline{x}(t) + \underline{b}_E\right) \quad mit \, \underline{f}_E \colon \mathbb{R}^n \to \mathbb{R}^m \, und \, n \gg m \tag{21}$$

Für den Decoder gilt entsprechend:

$$\underline{y}(t) = \underline{f}_D\left(\underline{k}(t)\right) = \sigma\left(W_D \,\underline{k}(t) + \underline{b}_D\right) \approx \underline{f}_E^{-1}\left(\underline{k}(t)\right) \text{ mit } \underline{f}_D \colon \mathbb{R}^m \to \mathbb{R}^n \text{ und } m \ll n \quad (22)$$

Die Netzgewichte des Encoders  $W_E$  sowie des Decoders  $W_D$  werden so trainiert, dass die sich ergebenden Abbildungen inverses Verhalten aufweisen. Die entstehenden niederdimensionalen Merkmale <u>k(t)</u> in der Zwischenschicht können extrahiert und für weitere Verarbeitungsschritte in anderen Netzen verwendet werden. Das Ziel eines Autoencoders ist somit das Erlernen einer exakten Rekonstruktion auf Basis effizient reduzierter Eingangsdaten. Dabei kann die Zielfunktion als Minimierung aller Rekonstruktionen formuliert werden:

$$\min_{W_E, W_D} \sum_{t=1}^{n} \left\| \underline{x}(t) - \sigma \left( W_D \ \sigma \left( W_E \ \underline{x}(t) + \underline{b}_E \right) + \underline{b}_D \right) \right\|^2$$
(23)

Oder als Minimierung der Rekonstruktion eines durchschnittlichen Eingangsvektors:
$$\min_{W_E, W_D} \left\| \underline{x}(t) - \sigma \left( W_D \ \sigma \left( W_E \ \underline{x}(t) + \underline{b}_E \right) + \underline{b}_D \right) \right\|^2 \quad mit \ \underline{x} = \sum_{t=1}^n \underline{x}(t)/n \tag{24}$$

Der trainierte Autoencoder liefert durch den Encoder-Teil ein Netz zur Dimensionsreduktion. Die Gesamtheit des Autoencoders kann unbekannte Daten nur unzureichend rekonstruieren [47]. Dieser Umstand kann für die Detektion unbekannter Anomalien genutzt werden, die einen erhöhten Rekonstruktionsfehler zur Folge haben.

### 2.1.5 Sequence-to-Sequence Netze

Neben der Detektion von Anomalien stellt in der vorliegenden Arbeit die Modellierung bzw. Prädiktion nichtlinearer zeitvarianter Prozesse und Anomalien ein übergeordnetes Ziel dar. Mit RNN und LSTM kann durch die Zuführung einer neuen Eingabe lediglich eine neue Ausgabe vorhergesagt werden. Für die Vorhersage weiterer Ausgaben werden weitere Eingaben benötigt. Es kommt durch die Rückführung vorhergesagter und mit Unsicherheit behafteter Ausgabewerte zudem zu einer Fehlerfortpflanzung. Seq2Seq-Netze lösen diese Probleme und können zu jedem beliebigen Zeitpunkt eine Vorhersage multipler Zeitschritte mit Horizont  $T_A$  auf Basis einer Eingabesequenz der Länge  $T_E$  erzeugen. Die Länge der Eingabesequenz  $(x(1), ..., x(T_E))$  sowie der Ausgabesequenz  $(y(1), ..., y(T_A))$  können flexibel gewählt werden. Das Netz erzeugt für jeden diskreten Vorhersagezeitpunkt des Horizonts  $T_A$  eine bedingte Wahrscheinlichkeitsdichte. Diese werden gemäß des Bayes Theorems verrechnet, sodass sich eine Dichte über alle potentiellen Ausgabesequenzen unter Bedingung der Eingabesequenz ergibt:

$$p(y(1), \dots, y(T_A) | x(1), \dots, x(T_E)) = \prod_{t=1}^{T_A} p(y(t) | v, y(1), \dots, y(t-1))$$
(25)

Der Vektor v stellt eine reduzierte Repräsentanz der Eingabesequenz mit fest definierter Größe dar [19]. Dieser Kopiermechanismus verknüpft Eingabe und Ausgabe des Netzes. Im Rahmen der Arbeit von [20] wurde dieser Mechanismus durch die Einführung eines über die Eingabesequenz gleitenden ,Transducers' flexibilisiert, sodass bereits erste Ausgabewerte erzeugt werden können, bevor die Eingabe vollständig verarbeitet wurde. Die Eingabesequenz wird dazu in *N* äquidistante Blöcke der Länge *W* aufgeteilt. Jeder Endpunkt *b* eines Eingabeblocks erzeugt einen vorrübergehenden Endzeitpunkt der Ausgabesequenz e(b). Es gilt:

$$e(b): \mathbb{N} \to \mathbb{N} \text{ mit } b \in 1, ..., N \text{ und } e(N) = T_A \text{ sowie } N = \frac{T_E}{W}$$
 (26)

Die für jeden Block durchgeführte Seq2Seq-Prädiktion kann damit wie folgt beschrieben werden:

$$p(y(e(b-1)+1), ..., y(e(b))|x(1) ..., x(bW), y(1), ..., y(e(b-1)))$$

$$= \prod_{i=e(b-1)+1}^{e(b)} p(y(i)|x(1) ..., x(bW), y(1), ..., y(i-1))$$
(27)

Die Ausgabe wird dabei iterativ durch die sequentielle Verarbeitung der Eingabeblöcke aufgebaut. Die Verrechnung der Wahrscheinlichkeitsdichten aller Teilbereiche der Ausgabe gemäß des Bayes Theorems erzeugt wie beim konventionellen Seq2Seq-Netz eine Dichte über alle potentiellen Ausgabesequenzen unter Bedingung der Eingabesequenz. Es ergibt sich:

$$p(y(1), ..., y(T_A)|x(1), ..., x(T_E)) = p(y(1), ..., y(e(1))|x(1) ..., x(W)) *$$

$$\prod_{b=2}^{N} p(y(e(b-1)+1), ..., y(e(b))|x(1) ..., x(bW), y(1), ..., y(e(b-1)))$$
(28)

Die Dichte für den ersten Teil der Ausgabe kommt ohne vorherige Ausgaben als Bedingung aus. Die Faktoren des Produktes aus Gl. 28 entsprechen dem Ergebnis aus Gl. 27. Durch die Flexibilisierung des Kopiermechanismus können Prädiktionsgüte und Echtzeitfähigkeit verbessert werden.



Abbildung 2.4: Seq2Seq Varianten mit flexibel einstellbaren Ein- und Ausgangssequenzlängen Im Allgemeinen können hinsichtlich des Verhältnisses der Sequenzlängen  $T_E$  und  $T_A$  die in

Abbildung 2.4 dargestellten Seq2Seq-Architekturvarianten ,many-to-one', ,one-to-many' und ,many-to-many' unterschieden werden [48].

# 2.2 Kompensation anomaler Systemdynamiken

Die im vorangegangenen Kapitel beschriebenen Ansätze können zur Modellierung nichtlinearer zeitvarianter Prozesse eingesetzt werden. Um Anomalien bei der Modellierung berücksichtigen und die Modelle in ein geeignetes Prädiktionsschema einsetzen zu können, wird im Folgenden auf diese beiden Aspekte eingegangen.

### 2.2.1 Beobachteransatz

Zur Beobachtung von Störungen liefert die Literatur eine große Auswahl verschiedener Schemata. In dieser Arbeit werden Störungen bzw. Anomalien untersucht, die das Aktorverhalten beeinflussen. Daher wird in diesem Abschnitt genauer auf das in Abbildung 2.5 dargestellte Schema eingegangen.



Abbildung 2.5: Aufbau eines Beobachters mit Filter und Störgrößenaufschaltung [7]

Der Stellgrößenvektor  $\underline{u}(z)$  wird durch die auf die Aktoren wirkenden Störungen  $\underline{d}(z)$  beeinflusst. Dadurch entsteht der Ausgang  $\underline{y}(z)$  auf Basis der Übertragungsfunktion des Prozesses G(z):

$$\underline{y}(z) = G(z)\left(\underline{u}(z) + \underline{d}(z)\right)$$
<sup>(29)</sup>

Im Folgenden wird eine vereinfachte Schreibweise ohne z-Transformierte verwendet. Das Modell des Prozesses wird mit  $G_m(z)$  bezeichnet. Es wird grundsätzlich angenommen, dass der Prozess nicht vollständig bekannt und das Prozessmodell daher mit Unsicherheiten behaftet ist. Der Filter Q(z) hat zum Ziel, die Schätzung der Störung um diese Unsicherheit zu bereinigen. Für den Fall, dass das inverse Prozessmodell  $G_m^{-1}(z)$  bspw. durch Adaption der realen inversen Dynamik entsprechen sollte, wird die Übertragungsfunktion des Filters zu eins. Für die Störungsschätzung gilt:

$$\underline{\hat{d}} = Q \left( G_m^{-1} G (\underline{u} + \underline{d}) - \underline{u} \right)$$
(30)

Die Schätzung wird zum Ausgleich der tatsächlichen Störung genutzt [13]. Durch Einsetzen von  $u(z) = \underline{\bar{u}}(z) - \underline{\hat{d}}(z)$  und Gl. 30 in Gl. 29 kann das Übertragungsverhalten zwischen ursprünglicher Stellgröße und Ausgang bestimmt werden:

$$\underline{y} = \frac{G(1-Q)}{1+Q(G_m^{-1}G-1)}\underline{d} + \frac{G}{1+Q(G_m^{-1}G-1)}\overline{\underline{u}}$$
(31)

Um einen optimalen Ausgleich der Störung erzielen zu können, muss eine Beziehung zwischen Filter Q(z) und Modellunsicherheit  $\Delta(z)$  von  $G_m(z)$  abgeleitet werden. Dazu wird für den Zusammenhang von Modellunsicherheit und realem Prozess folgende Beziehung angenommen  $G(z) = G_m(z)(1 - \Delta(z))$  und in Gl. 31 eingesetzt:

$$\underline{y} = \frac{G_m(1-\Delta)(1-Q)}{1+Q\Delta}\underline{d} + \frac{G_m(1-\Delta)}{1+Q\Delta}\underline{\bar{u}}$$
(32)

Der Filter muss demnach in Abhängigkeit der Modellunsicherheit so eingestellt werden, dass die Übertragungsfunktion der Störung gegen null geht [7].

## 2.2.2 Modellbasierter prädikativer Ansatz

Neben einer direkten Kompensation von Störverhalten ist auch eine indirekte Kompensation bspw. im Rahmen eines modellbasierten prädikativen Ansatzes (Model Predictive Control, MPC) möglich, der auch auf einen Beobachter zurückgreifen kann. Dabei wird dieser zur Erzeugung eines Modells des Störverhaltens genutzt. Das Störmodell wird in das eigentliche Prozessmodell integriert. Das Gesamtmodell wird dann zur Vorhersage über einen definierten Horizont genutzt. Basierend auf der Prädiktion wird das folgende nichtlineare Optimierungsproblem gelöst:

$$J(t) = min\left\{\sum_{i=1}^{T_1} q_i \left(\underline{y}_s(t+i) - \underline{\hat{y}}(t+i)\right)^2 + \sum_{j=1}^{T_2} w_i \,\Delta \underline{u}^2(t+j-1)\right\}$$
(33)

Dabei beschreibt der erste Summand die Abweichung der Vorhersage  $\hat{y}$  vom Sollwert  $\underline{y}_s$  über den Horizont  $T_1$  und der zweite Summand den Stellaufwand  $\Delta \underline{u}$  über den Horizont  $T_2$ . Die Lösung des Problems liefert die optimale Stellstrategie zur Minimierung zukünftiger Abweichungen, wobei lediglich die erste Stellgröße  $\underline{u}(t + 1)$  umgesetzt wird. Die Optimierung wiederholt sich iterativ mit jedem diskreten Zeitschritt. Da das Prozessmodell das Störverhalten mitmodelliert, erfolgt indirekt auch eine Störgrößenkompensation, da die Stellgrößen so optimiert werden, dass trotz auftretender Störungen eine minimale Abweichung vom Sollwert erzielt wird. Ein entsprechendes autoregressives Prozessmodell (AR-Modell), das die Anforderungen erfüllt, kann bspw. folgende Form haben:

$$\underline{\hat{y}}(t+1) = f\left(\underline{y}(t), \dots, \underline{y}(t-N_y), \underline{u}^*(t+1), \dots, \underline{u}(t-N_u), \underline{\hat{d}}(t), \dots, \underline{\hat{d}}(t-N_{\hat{d}})\right)$$
(34)

Dabei stellt  $\underline{u}^{*}(t + 1)$  die unter Berücksichtigung vergangener Ausgangs-, Stell- und Störgrößen zu optimierende zukünftige Stellgröße dar [13]. Das modellbasierte prädikative Optimierungsschema ist für das im Rahmen dieser Arbeit entstandene Konzept von Bedeutung und umfasst in den meisten Fällen zusätzlich noch Nebenbedingungen, die vor allem Restriktionen der Stellgrößen betreffen.

# 3 Stand der Wissenschaft und Technik

Trotz des hohen Wissens im Engineering der Prozesse treten immer wieder unbekannte und teilweise unkontrollierbare Effekte auf, die von kleineren Ausreißern und Schwankungen bis hin zu weitreichenden Dissonanzen, die mit einer Reduktion der Prozessstabilität einhergehen, reichen. Beim Entwurf einer Qualitätsregelung bzw. im Allgemeinen beim Entwurf von Reglern ermöglicht das Vorhandensein eines möglichst exakten Prozessmodells die Konzeption eines geschlossenen Regelkreises mit optimierten Eigenschaften im Hinblick auf Stabilität und Performanz. Ein in Teilen von der Realität abweichendes Prozessmodells, das aufgrund bislang unbekannter und nicht identifizierbarer Prozessdynamiken mit Ungenauigkeiten behaftet ist, kann zum Entwurf eines Reglers mit geringer Regelgüte oder gar zum Entwurf eines instabilen Reglers führen. Anomalien haben genau diesen apriori unkalkulierbaren Effekt auf die Auslegung einer Oualitätsregelung im Kontext der diskreten Fertigung. Daher ist die Beobachtung bzw. die Detektion von zur Entwurfszeit des Fertigungssystems unbekannten Anomalien von essentieller Bedeutung für die Güte einer Qualitätsregelung. Die klassische Regelungstheorie liefert dazu Ansätze für Störungsbeobachter, die in Form eines definierten Schemas in den Regelkreis integriert und zur Unterdrückung auftretender Störungen eingesetzt werden. Für die in dieser Arbeit betrachtete Problemstellung sind sie allerdings nicht anwendbar, da sie zumeist die Invertierbarkeit eines analytisch beschreibbaren Prozessmodells voraussetzen. Daher ist im Rahmen dieser Arbeit die Konzeption eines neuronalen Anomalie-Detektors von Nöten, der an die Stelle des klassischen Störungsbeobachters tritt und weder Anforderungen an das vorhandene Prozessmodell noch an die zu beobachtenden Effekte stellt. Durch die Integration der Anomalie-Detektion, und der damit einhergehenden, präventiven Erkennung und Charakterisierung von Anomalien, ist es möglich, die Prozessmodellierung stetig zu optimieren und eine zuverlässige Qualitätsregelung zu realisieren. Anomalien können nach verschiedenen Gesichtspunkten voneinander unterschieden werden. Dies können nach [28] z.B. die nachfolgend aufgeführten sein:

- Angriffspunkt
- Messbarkeit
- Beschaffenheit
- Zeitverhalten

Anomalien können auf die Aktoren einwirken, den Prozess direkt beeinflussen oder den Prozessausgang verfälschen. Sie können direkt messbar oder lediglich beobachtbar sein. Sie weisen in den wenigsten Fällen lineare Charakteristiken auf, sodass von einer nichtlinearen Beschaffenheit ausgegangen werden kann. Das Zeitverhalten wird im Stand der Technik in verschiedenen Dimensionen analysiert. So wird in [49] zwischen stationärem und nichtstationärem Verhalten unterschieden, wohingegen in [50] kurzfristige von langfristigen Effekten separiert werden. In [40] werden Anomalien in Punktanomalien, kontextuelle und kollektive Anomalien unterteilt. Die Informationen zu den Charakteristiken liegen zumeist nicht vor, sodass Annahmen getroffen oder Einschränkungen auf bestimmte Typen von Anomalien vorgenommen werden müssen. Nachfolgend wird der Stand der Wissenschaft und Technik zur Detektion, Modellierung und Kompensation von anomalen Systemdynamiken vorgestellt. Im ersten Teil wird auf die neuronalen Ansätze zur Prozessmodellierung und im zweiten Teil auf den Ausgleich der Anomalien eingegangen. Dabei werden Ansätze aus verschiedenen Anwendungsbereichen untersucht und hinsichtlich der in Kapitel 1.4 aufgezeigten Anforderungen eruiert. Abschließend wird daraus der aktuelle Forschungsbedarf abgeleitet.

## 3.1 Nichtlineare zeitvariante Prozessmodellierung

Fertigungssysteme zeichnen sich durch hochgradig nichtlineare und zeitvariante Prozesse aus, die häufig von unbekannten Effekten in Form von internen und externen Einflüssen gestört werden [4]. Auf der anderen Seite entstehen durch das Vorhandensein großer Parameterräume erhebliche Datenmengen, die zur Identifikation der Einflüsse im Rahmen von Analysen verwendet werden können. Neuronale Netze wurden bereits in diversen Anwendungen zur Modellierung und Vorhersage komplexer Systemdynamiken eingesetzt und eignen sich zur Approximation nichtlinearer Zusammenhänge. Es stehen zahlreiche Netzwerkarchitekturen zur Verfügung, deren Modellierungsgenauigkeit jedoch stark von der Eignung für das zu Grunde liegende Problem abhängt. So kann durch die Integration von Rekurrenzen zeitvariantes Verhalten abgebildet werden [51]. Um genaue Vorhersagen treffen zu können, die sowohl stationäres und nichtstationäres Verhalten als auch kurz- und langfristige Abhängigkeiten berücksichtigen, können bspw. LSTM-Netze herangezogen werden, die in der Lage sind komplexe Gedächtnisstrukturen abzubilden [18, 52]. In Kombination mit einer dimensionsreduzierenden Vorverarbeitung der Daten kann effizient mit großen Datenräumen umgegangen werden. Dabei werden diese unter minimalem Informationsverlust in einen niederdimensionalen Unterraum projiziert. Die auf diese Weise erzeugten Merkmale können dann in den Netzstrukturen für den Lernprozess genutzt werden [53]. Das gemeinsame Ziel aller untersuchten Ansätze ist die Modellbildung nichtlinearer zeitvarianter Prozesse zur Prädiktion des Systemverhaltens. Die bestehenden Ansätze werden hinsichtlich der folgenden Anforderungen untersucht:

- Prädiktion nichtlinearer und zeitvarianter Abhängigkeiten
- Kurzzeit- und Langzeitgedächtnisverhalten
- Mehrdimensionale Datenverarbeitung
- Prädiktion multimodaler bzw. multivariater Abhängigkeiten
- Prädiktion multipler Zeitschritte und Fehlerfortpflanzung

Durch den Abgleich bestehender Ansätze mit den Anforderungen soll aufgezeigt werden, welche Netzarchitekturen die beste Eignung zur Lösung der zu Grunde liegenden Problemstellung aufweisen und welcher weitere Forschungsbedarf hinsichtlich dieser Netzarchitekturen besteht. In diesem Abschnitt werden verschiedene neuronale Netze vorgestellt. Bevor dazu auf die neuronale Modellbildung zur Prädiktion von Anomalien oder Prozessdynamiken im Allgemeinen eingegangen wird, werden Mechanismen für deren Detektion vorgestellt.

### 3.1.1 Detektion von Anomalien

Auf dem Gebiet der Regelungstheorie wurden in den letzten Jahren und Jahrzehnten zahlreiche Detektionsansätze entwickelt, die für die Beobachtung externer Störungen oder unbekannter interner Prozesszustände konzipiert wurden [54]. Die Theorie zu solchen Störungsbeobachtern, die eine Schätzung von Prozessinterferenzen liefern, wurde dabei von [55] entwickelt und von [56] für die Anwendung auf nichtlineare Störungen erweitert. Durch die Fortschritte im Bereich des maschinellen Lernens wurden in den letzten Jahren auch zahlreiche, auf lernfähigen Algorithmen basierende Ansätze zur Detektion von unbekannten Anomalien entwickelt. Hier haben sich primär unüberwachte Lernansätze, wie der den Clustering-Algorithmen zuzuordnende k-Nearest-Neighbour-Algorithmus, durchgesetzt [38]. Damit wird dem in der diskreten Fertigung häufig anzutreffenden Umstand Rechnung getragen, dass die Daten über keine Label verfügen. Durch den Einsatz unüberwachter Lernverfahren kann eine Detektion von Anomalien erfolgen. Hinsichtlich der Detektionsgüte komplexer Anomalien konnte mit Ansätzen, die auf neuronalen Netzen basieren, in den letzten Jahren verbesserte Ergebnisse erzielt werden. Dies hängt mit den vielfältigen strukturellen Gestaltungsmöglichkeiten zusammen. Hier sind in besonderem Maße Encoder-Decoder bzw. Autoencoder-Netze zu erwähnen [57].

Im Folgenden wird eine Übersicht über die für diese Arbeit relevanten Ansätze gegeben. Dabei steht die Anomalie-Detektion in Zeitreihendaten, basierend auf unüberwachten Lernverfahren, im Fokus der Betrachtungen. In [58] wird eine probabilistische, auf dem Bayes Theorem beruhende Methodik zur Detektion von sowohl lokalen als auch globalen Anomalien vorgestellt. Lokale Anomalien bezeichnen Abweichungen einzelner oder multipler Datenpunkte, wohingegen globale Anomalien die Diskrepanz zwischen ganzen Dateninstanzen und den Charakteristiken der sie umgebenden Datenreihen beschreiben. Die Methodik zur Detektion von Anomalien auf Basis von Fahrzeugdaten vorgestellt. Dabei wird das reguläre Kommunikationsverhalten einzelner Komponenten mit LSTM-Netzen gelernt und zur Unterbindung externer Eingriffe durch Cyber-Attacken eingesetzt. Signifikante Abweichungen der Systemdynamiken von der Netzausgabe entsprechen anomalem Verhalten und lassen auf einen externen Eingriff schließen. Ansätze zur Anomalie-Detektion mit LSTM-Netzen haben sich als vielversprechend herausgestellt und sind Gegenstand aktueller Forschungsaktivitäten.

Durch das Erlernen niederdimensionaler temporaler Repräsentanzen sowie durch die Erfassung von Langzeitabhängigkeiten eignen sich LSTM-Netze in besonderem Maße zur Modellierung von Zeitreihen und zeitvarianten Systemen [18]. Dieser Umstand wird in [60] beschrieben und dort in Form einer geschichteten LSTM-Architektur umgesetzt. Dem Netz werden die Zeitreihendaten direkt und ohne vorgeschaltete Dimensionsreduktion zugeführt. Auf die Ausgabendifferenz werden multivariate Gaußverteilungen angewendet und definierte Perzentile als Anomalien klassifiziert. In zahlreichen darauf aufbauenden Arbeiten wurden Erweiterungen entwickelt. So präsentieren [61] einen Deep Autoencoder, der darauf trainiert wird, die ihm zugeführten Eingaben vollständig zu rekonstruieren. Die Eingaben werden in einem Encoder-Teil in einen niederdimensionalen Merkmalsraum projiziert und in einem Decoder-Teil wieder entschlüsselt. Der trainierte Autoencoder hat somit die Fähigkeit, bekannte Zusammenhänge reduzieren und rekonstruieren zu können. Anomalien werden durch einen erhöhten Fehler bei der Rekonstruktion erkennbar. In [62] werden Encoder und Decoder im Rahmen einer Seg2Seg-LSTM-Architektur verwendet. Die Detektion von Anomalien basiert auf den innerhalb des Netzes propagierten Zellzuständen. Anomalien erzeugen unbekannte Zellzustände, die durch einen nachgeschalteten Clustering-Algorithmus ausgewertet werden. Ein weiterer Seq2Seq-Ansatz wird in [63] präsentiert. Dieser erkennt verschiedene Anomalien durch die Modellierung multipler Attribute. In einer empirischen Vergleichsstudie auf Basis eines Referenzdatensatzes können höhere Genauigkeiten in der Detektion erzielt werden als in [60]. In dem Ansatz von [64] werden ,Generative Adversarial Networks' (GAN) nach [65] zur Detektion von Anomalien verwendet. GANs bestehen aus zwei Netzen, dem Generator, der die Struktur realer Daten nachbildet, und dem Diskriminator, dessen Ziel die Unterscheidung der vom Generator erzeugten, künstlichen Daten von den realen Daten ist. In [64] wird ein GAN auf einen Autoencoder abgebildet, wobei der Generator rekonstruierte Zeitreihendaten ausgibt und der Diskriminator zur Unterscheidung von normalen und anomalen Daten genutzt wird.

Neben Encoder-Decoder Netzen wurden in der Vergangenheit vor allem stochastische Verfahren zu Anomalie-Detektion eingesetzt. Sowohl in [38] als auch in [66] werden Übersichten zu Verfahren und verschiedenen Aspekten der Anomalie-Detektion gegeben. So wird im Rahmen der erstgenannten Arbeit neben einer Erörterung der Anwendungsbereiche eine Klassifikation nach stochastischen, überwachten und unüberwachten Detektionsverfahren vorgenommen. Die zweite Arbeit erweitert die Darstellung neuronaler Ansätze um Bayes'sche und hierarchische Netze. Temporale Abhängigkeiten stehen bei der Detektion von Anomalien nicht im Vordergrund und spiegeln sich dementsprechend auch nicht in der Gestaltung der Netze wider. Beide Arbeiten unterscheiden die drei eingangs erläuterten Anomalie-Typen, namentlich die Punktanomalien, die kontextuellen sowie die kollektiven Anomalien. In [40] wird ein faltendes Netz ("Convolutional Neural Network", kurz CNN) mit LSTM-Zellen zur holistischen Detektion aller Anomalie-Typen kombiniert. Es wird dazu befähigt, durch die Faltung sowohl räumliche Abhängigkeiten und durch LSTM-Zellen zusätzlich zeitliche Abhängigkeiten kapseln zu können. Niederdimensionale Repräsentanzen räumlicher und zeitlicher Relationen stellen die Grundlage für die Modellierung

des Normalverhaltens dar. Divergenzen von tatsächlichem Verhalten und Netzausgabe werden durch die Kreuzentropie erfasst. Der Ansatz ist in Abbildung 3.1 illustriert.



Abbildung 3.1: CNN und LSTM zur Detektion von Anomalien gemäß [40]

Detaillierte Untersuchungen zu kontextuellen Anomalien werden in [67] vorgenommen. Dazu werden temporale Strukturen mit ähnlichen Charakteristiken aus verschiedenen Zeitreihen zu einem Kontext zusammengefasst. Die dynamische Zeitnormierung wird dabei angewandt, um auf Basis von Zeitfenstern definierter Größe Distanzmaße zwischen den erfassten Zeitsequenzen zu bestimmen und diese einem Kontext zuzuweisen. Die Minimierung der Varianz der Distanzmaße innerhalb der Kontexte steht der Maximierung derselben zwischen den vorhandenen Kontexten gegenüber. Ein wesentlicher Vorteil des Ansatzes ist der Umstand, dass alle in einer neuen Zeitsequenz enthaltenen kontextuellen Anomalien im Rahmen eines Detektionsvorgangs erfasst werden können, sodass es zu deutlichen Effizienzvorteilen gegenüber anderen stochastischen Methoden kommt. Detaillierte Untersuchungen zu kollektiven Anomalien sind in [68] zu finden. Dazu werden LSTM-Netze eingesetzt, um die zeitlichen Abhängigkeiten besser erfassen und kollektive Anomalien besser detektieren zu können. Im Allgemeinen zeigen die in diesem Kapitel präsentierten Ansätze auf, wie sowohl stochastische als auch LSTM-basierte Ansätze eingesetzt werden können, um aus generierter und realer Ausgabe Rückschlüsse auf Anomalien ziehen zu können. Im nächsten Abschnitt wird nun genauer auf die Modellierung nichtlinearen zeitvarianten Prozessverhaltens eingegangen.

### 3.1.2 RNN und LSTM zur Modellierung

Neuronale Netze wurden bereits im Rahmen zahlreicher Forschungsarbeiten zur Modellierung und Vorhersage von Systemdynamiken eingesetzt. Die Architektur der Netze weist dabei eine starke Abhängigkeit von der zu Grunde liegenden Problemstellung auf. In [13] wurden bspw. RBFN zur Modellierung quasi-statischer Störeffekte mit geringer Zeitvarianz herangezogen, da diese primär durch Nichtlinearitäten hoher Ordnung gekennzeichnet sind und RBFN dafür gute Approximationseigenschaften aufweisen. Bei einer ähnlichen Problematik haben [14] ein Multilayer Perzeptron (MLP) zur Modellierung eingesetzt. Um der leichten zeitlichen Varianz des Störverhaltens Rechnung zu tragen, wurde über ein Verzögerungsglied eine Sequenz an vergangenen Aus- und Eingängen mit endlichem Horizont als Netzeingang genutzt. Treten jedoch starke zeitliche Abhängigkeiten auf, weisen RNN eine bessere Eignung zur Modellierung auf als rein vorwärts gerichtete Netze. Auf Grundlage der in Kapitel 2.1.2 eingeführten Eigenschaften ist dies in besonderem Maße für kurzfristige, hochdynamische Zeitvarianzen der Fall. So werden RNN in [69] für die Prädiktion stationärer, abrupt auftretender Anomalien verwendet und zeigen eine gegenüber stochastischen Ansätzen wie ,Autoregressive Moving Average' (ARIMA) Modellen oder Gauß-Prozessen erhöhte Vorhersagegenauigkeit.

Eine Übersicht über RNN ist in [17] zu finden. Dort wird zwischen vollständig verknüpften (,fully connected<sup>4</sup>) und partiell verknüpften (,partially connected<sup>4</sup>) RNN differenziert. Das erstgenannte wurde in einer grundlegenden Arbeit von Williams und Zipser bereits Ende der 80er Jahre entwickelt [70]. Zu den ,partially connceted' RNN zählen das Elman Netz [71], das Jordan Netz [72] sowie die rekurrente Ausprägung des MLP [73], die allesamt das Ziel verfolgen, Strukturen in Zeitreihen und Kontextinformationen zu finden und zu modellieren. Auf Basis dieses Fundaments sind in den letzten Jahren zahlreiche Erweiterungen von RNN für ein weites Feld an Anwendungen entstanden. So haben [74] selbst-organisierende RNN entwickelt, die während des Betriebs strukturelle Änderungen im Netz zur Adaption des aktuellen Zustands vornehmen. Zur Verarbeitung großer Datenräume wurde in [75] die Diskrete-Wavelet-Transformation (DWT) in RNN integriert und in einer Studie mit gewöhnlichen Aktivierungsfunktionen verglichen. Der Anwendungsbereich von RNN reicht von der Modellierung nicht-sequentieller, zyklischer Datenreihen sowie parallel stattfindender Prozesse im Business Monitoring [76] über die nichtlineare Systemidentifikation im Kontext der Regelung verfahrenstechnischer Prozesse [77] bis hin zur Trajektorienplanung autonomer Fahrzeuge [78]. Nichtsdestotrotz weisen RNN den Nachteil des Vanishing-Gradient-Effekts auf [43]. Über einen längeren Zeitraum vorhandene, nicht-stationäre Abhängigkeiten werden von RNN mit einfachen Aktivierungsfunktionen weniger gut erfasst. Ein Ansatz, der auf die Erzeugung eines Gedächtnisses innerhalb regulärer RNN zur Abbildung von Langzeitabhängigkeiten abzielt, wird in [79] beschrieben. RNN werden hier durch die Verwendung von variablen, rekurrenten "Sprung"-Verbindungen flexibilisiert, um durch das Auslassen von Neuronen direkt Informationen aus vergangenen Zeitschritten verarbeiten und so Gedächtnisstrukturen mit einer definierten Speicherkapazität schaffen zu können. Die Länge der Verbindungen hat einen Einfluss darauf, ob primär kurzfristige oder langfristige Abhängigkeiten einbezogen werden.

Der Ansatz, ein Gedächtnis über Verbindungskonstrukte zwischen Neuronen zu erzeugen, hat sich im Vergleich zur Flexibilisierung des Informationsflusses innerhalb der Neuronen als nachteilig erwiesen. Daher sind RNN gemäß [80] primär für hochgradig nichtlineare und dynamische Zeitvarianzen geeignet, die stationären oder nicht stationären Kurzzeitabhängigkeiten unterliegen. Langzeitabhängigkeiten sowie multiple Abhängigkeiten mit unterschiedlichen Charakteristiken werden nur unzureichend erfasst. LSTM-Netze lösen dieses Problem durch die Propagation eines internen Zellzustands innerhalb der Neuronen und erzeugen dadurch ein definiertes Gedächtnisverhalten [18]. Auch für LSTM sind zahlreiche Varianten entstanden, die zur Erfüllung spezieller Anforderungen entwickelt wurden.



Abbildung 3.2: Seriell-parallele Systemidentifikation mit konvexem LSTM-Netz nach [81]

So wird in [81] eine konvexe Struktur vorgestellt, die eine gewichtete Zusammenführung multipler LSTM-Prädiktionen vorsieht und in parallelen und seriell-parallelen Strukturen zur Systemidentifikation getestet wird. Dies ist in Abbildung 3.2 dargestellt. Zur Modellierung von Wiener-Hammerstein-Systemen wird in [82] eine Matrixstruktur verwendet, die zur Abbildung statischer Nichtlinearitäten durch RBFN erweitert werden kann. Da bei Prädiktionsaufgaben hohe Echtzeitanforderungen bestehen können, wurde von [53] ein Online-Lernverfahren entwickelt, das es ermöglicht, LSTM-Netze im Betrieb weiter zu trainieren. LSTM-Ansätze wurden für verschiedenste Anwendungsbereiche vom Business Monitoring [83] über energietechnische Anwendungen [84] bis hin zu Verkehrsbewegungen [85] zur Modellierung zeitvarianter Prozesse entwickelt. Auch in regelungstechnische Anwendungen erhalten LSTM-Ansätze zunehmend Einzug. In [86] wurde dazu eine MPC auf Basis von LSTM-Netzen entworfen und mit gängigen Verfahren zur Prozessmodellierung verglichen. LSTM zeigte dabei deutliche Vorteile sowohl hinsichtlich der Systemidentifikation als auch beim Einsatz als Prädiktor in der Regelung. In [87]

wurde der Ansatz auf einen verfahrenstechnischen Prozess übertragen. Untersuchungen zur Modellierungsgüte von LSTM im Vergleich zu GRU werden erstmalig in [88] im Rahmen der Regelung eines mehrstufigen Prozessablaufs beschrieben. Die Studie deutet darauf hin, dass der Einsatz von Kompositionen komplexer neuronaler Netzarchitekturen in Regelkreisen trotz der nur schwer nachweisbaren Stabilität zu deutlichen Verbesserungen des Prozessergebnisses beitragen kann. Aus diesem Grund werden nachfolgend weitere LSTM-Netzarchitekturen untersucht.

### 3.1.3 Weitere Netzarchitekturen

In diesem Abschnitt werden Netzarchitekturen beschrieben, die vor allem für den Bereich der Bild- und Sprachverarbeitung zur Modellierung zeitvarianten Verhaltens entwickelt wurden und deren Eigenschaften eine Übertragung auf fertigungstechnische Prozesse ermöglichen. Bevor jedoch weitere LSTM-Netzarchitekturen vorgestellt werden, wird zunächst auf die GRU-Zelle als Abwandlung der LSTM-Zelle eingegangen. GRU wurde von [45] mit dem Ziel eingeführt, einen Mechanismus zu schaffen, der die Fähigkeit zur Prädiktion von Langzeitabhängigkeiten, wie sie bspw. bei der LSTM-Zelle ausgeprägt vorhanden ist, um eine verbesserte Integration von aktuellen Informationen ergänzt. Bei GRU besteht die Möglichkeit, vergangene Informationen vollständig zu verwerfen. Dadurch soll eine adaptive Modellierung von Relationen über variable Zeithorizonte ermöglicht werden. In [89] wird dazu ein Vergleich der beiden Zellarchitekturen sowie konventioneller Aktivierungsfunktionen hinsichtlich der Modellierung von Zeitsequenzen vorgenommen. Die beiden Zellen mit Gating-Mechanismus erzielen deutlich bessere Resultate als die klassischen Ansätze, es kann jedoch keine allgemeingültige Aussage getroffen werden, welcher der Mechanismen für welche Problemstellung zu bevorzugen ist. In [90] wird hingegen im Rahmen einer groß angelegten Studie zu verschiedenen Netzarchitekturen die Überlegenheit von LSTM gegenüber GRU ermittelt. Weiterhin konnte gefunden werden, dass Gating-Systeme zur Filterung irrelevanter Eingangsinformationen beitragen, LSTM jedoch eine höhere Präzision in der Modellierung zeitvarianten Verhaltens erzielt. Aus diesem Grund wird in der weiteren Analyse des Stands der Technik von einer genaueren Betrachtung von GRU-Netzarchitekturen abgesehen.

Im folgenden Abschnitt wird auf den Stand der Wissenschaft und Technik zu LSTM-basierten Netzarchitekturen eingegangen und deren Erfüllung der bestehenden Anforderungen erörtert. Eine Unterteilung wird in dieser Arbeit nach LSTM-Netzen mit optimierter Repräsentation des Zellzustands auf der einen und LSTM-Netzen mit interagierenden Zellzuständen auf der anderen Seite vorgenommen. Zu den erstgenannten gehören bspw. bidirektionale oder hierarchische Ansätze. Bidirektionale LSTM-Netze propagieren den in Kapitel 2.1.3 eingeführten LSTM-Zustandsvektor nicht nur in Vorwärts- sondern auch in Rückwärtsrichtung. Das hat den Vorteil, dass Relationen in beide Richtungen in der Modellierung berücksichtigt werden. Durch die rückwärtsgerichtete Zustandspropagation erfolgt eine Antizipation zukünftigen Verhaltens, sodass zu erwartende Informationen in die aktuelle Ausgabe einbezogen werden können. In [91]

wird ein solcher Ansatz zur Abbildung räumlich und zeitlich verteilter Informationen genutzt und durch einen Filtermechanismus zur Festlegung der Zell-Verknüpfungen flexibilisiert. So kann mit unvollständigen Daten umgegangen werden. Die Architektur ist in Abbildung 3.3 dargestellt.



Abbildung 3.3: Bidirektionales LSTM (links) und Filterschicht zur Verarbeitung lückenhafter Datensätze (rechts) nach [91]

Auch in [92] zeigt sich die Eignung bidirektionaler LSTM für die Lösung multidimensionaler Probleme. Hier werden aus verschiedenen Dimensionen extrahierte Merkmale parallel verarbeitet und zusammengeführt. Bidirektionale LSTM könnten sich durch ihre Eigenschaften auch zur Abbildung der räumlich und zeitlich verteilten Physiken diskreter Fertigungsprozesse eignen. Hierzu gibt es jedoch noch keine Vorarbeiten. Hierarchische LSTM lösen mehrdimensionale Probleme, indem sie das Gesamtproblem in Teilprobleme zerlegen und diese in einer hierarchischen Struktur organisieren. Das hat den Vorteil, dass der Fokus auf ein bestimmtes Teilproblem gelegt und das Netz eine gewisse Aufmerksamkeit erzeugen kann. Anwendung findet diese Methodik bspw. in [93], die Bilder strukturieren und die Trajektorien verschiedener Objekte parallel vorhersagen können. Auch [94] beschäftigt sich mit dem Thema der Aufmerksamkeit und untersucht für ereignisdiskrete Systeme Mechanismen, die die Wichtigkeit vergangener Einzelevents für zukünftige Ereignisse beurteilen. In [95] wird innerhalb eines LSTM-Netzes eine separate ,Aufmerksamkeits'-Schicht eingeführt, die mit Hilfe von apriori vorhandenem Wissen angereichert wird. In [96] werden sie zur Prädiktion von Langzeitabhängigkeiten herangezogen. Um die über einen längeren Horizont erfassten Daten nach Relevanz zu filtern, können Faltungsoperationen in LSTM-Netze eingebracht oder diese mit faltenden Netzen kombiniert werden. In den nachfolgenden Arbeiten hat dieses Vorgehen das Ziel, die Prädiktionsgüte von Langzeitabhängigkeiten zu steigern. In [97] wird das konventionelle LSTM erweitert, indem Eingänge und Gewichte durch Faltung auf die relevanten Informationen reduziert und Korrelationen in den Daten in Form von abstrakten Merkmalen extrahiert werden. Die Faltungsoperationen wirken direkt in der Zelle. Anwendungen von CNN in Kombination mit LSTM finden sich bspw. in [98] und [99]. Im ersten Fall ermöglicht die Komposition aus CNN und LSTM eine zeitvariante Modellierung verteilter Zusammenhänge. Zwischenschichten werden mit, Scaled Exponential Linear Units' (SELU) realisiert, da diese über verbesserte Normierungsund Konvergenzeigenschaften verfügen. Auch in der zweiten Arbeit wird eine Komposition zur Vorhersage räumlich und zeitlich verteilter Dynamiken genutzt. Die CNN-Schichten extrahieren Merkmale und gebe den reduzierten Datenraum an die LSTM-Schichten weiter. CNN-LSTM-Netze könnten sich aufgrund ihrer charakteristischen Eigenschaften ebenfalls zur Modellierung der räumlich und zeitlich verteilten Physiken diskreter Fertigungsprozesse eignen.

Neben den drei beschriebenen LSTM-Varianten wurden in den letzten Jahren zahlreiche weitere Netzarchitekturen entwickelt. [100] erweitern LSTM um Verbindungen in beliebige Dimensionen innerhalb der Zellen (Grid LSTM). Zusätzlich zu der konventionellen Propagation des Zellzustands entlang der zeitlichen Dimensione können Verbindungen bspw. entlang räumlicher oder parameterspezifischer Dimensionen der Eingaben integriert werden. Diese erlauben neben einer Prädiktion der in den Dimensionen modellierten Größen zusätzlich eine Abbildung der Zusammenhänge zwischen diesen Größen. Das Netz eignet sich daher zur simultanen Prädiktion multipler Größen, die linear abhängig oder unabhängig sein können. In Abbildung 3.4 ist einem Grid LSTM mit zwei Dimensionen ein geschichtetes LSTM ("Stacked LSTM") gegenübergestellt.



Abbildung 3.4: Erweiterung der LSTM-Zellen im 2d Grid LSTM im Vergleich zu Stacked LSTM gemäß [100]

Dabei wird anhand des hellblau dargestellten Informationsflusses innerhalb der Zellen deutlich, dass das Grid LSTM Ausgänge von Dimensionen als zusätzliche Eingänge anderer Dimensionen verwendet. Durch diese Abbildung von Relationen kann die Prädiktionsgüte mehrdimensionaler Probleme verbessert werden. Geschichtete LSTM zeichnen sich durch eine "Stapelung" mehrerer funktionaler Netzschichten aus, die in [101] als Stacked LSTM-Autoencoder realisiert werden. Sie lösen das Problem der hohen Dimension in den Daten und der Prädiktion multimodaler Parameterräume durch vorgeschaltete Encoder zur Projektion auf niederdimensionale Merkmale oder durch eine direkte Integration von En- und Decoder in die LSTM-Zellstruktur. Für die gekoppelte Vorhersage multipler Größen liefern [102] einen Ansatz, der eine Verknüpfung von LSTM-Netzen vorsieht, die zur separaten Modellierung der Größen genutzt werden. Diese ,LSTM-Streams' interagieren über rekurrente Verbindungen, um den Abhängigkeiten der zu prädizierenden Größen Rechnung zu tragen. Ausgänge definierter Schichten fungieren als Eingänge nachfolgender Schichten anderer Streams. Dadurch kann eine multimodale Vorhersage realisiert werden ("Cross modal LSTM"). Das Konzept ist in Abbildung 3.5 visualisiert.



Abbildung 3.5: Cross modal LSTM mit Informationsaustausch zwischen zwei Streams [102]

Neben der Beherrschung der Mehrdimensionalität von Eingangsdaten und einer multimodalen Vorhersage von Zielgrößen wurden weitere relevante Ansätze zur Detektion und Extraktion von Korrelationen sowie zum Umgang mit unausgewogenen Eingangsdaten untersucht. So wurde in [103] ein duales holographisches LSTM entwickelt, das Zusammenhänge in Datenströmen mit Hilfe der zirkularen Kreuzkorrelation untersucht. In [104] wird eine Komposition von LSTM und GAN vorgestellt, die zum Umgang mit unvollständigen oder unausgewogenen Daten genutzt und in diesem Kontext zur Erzeugung künstlicher Anomalie-Daten herangezogen wird. Das LSTM wird zur Prädiktion multipler Zeitschritte in die Zukunft verwendet, indem die entsprechende Anzahl an Zeitschritte über das Netz iteriert wird. Vorhergesagte Ausgänge werden zur Vorhersage weiterer Ausgänge zurückgeführt. Eine Problematik bei dieser Vorgehensweise zur Prädiktion einer Ausgabesequenz ist der Umstand, dass es zu einer Fehlerfortpflanzung durch die bereits vorhergesagten und mit Vorhersagefehlern behafteten Ausgangsrückführungen kommt. Um mit LSTM-Netzen zu jedem Zeitpunkt multiple Zeitschritte vorhersagen zu können, ist eine Erweiterung der beschriebenen Netzarchitekturen notwendig.

### 3.1.4 Sequence-to-Sequence LSTM

In [19] wurde zur Prädiktion multipler Ausgänge die Seq2Seq-Architektur eingeführt, die es erlaubt, Eingabesequenzen in Ausgabesequenzen zu überführen und dadurch eine Vorhersage über multiple Zeitschritte zu realisieren. RNN oder Netzen mit Gate-Mechanismen wie LSTM ist es nur unter iterativer Propagation des Vorhersagefehlers möglich, Eingabesequenzen in Ausgabesequenzen zu überführen [45]. Es steht darüber hinaus keine Methodik zur Verfügung,

um Sequenzen unterschiedlicher Länge zu konvertieren, da für jede neue Ausgabe neben der rückgeführten Vorhersage eine neue Eingabe erforderlich ist. Seq2Seq löst diese Problematiken über eine Encoder-Decoder Struktur, die eine Transformation der Eingabesequenz in einen Vektor fest definierter Dimension vorsieht. Diese abstrakte Repräsentanz der Eingabe wird als initialer Zellzustand in den Decoder kopiert und dort von den LSTM-Zellen in eine Ausgabesequenz variabler Länge überführt. Die Vorhersage kann probabilistischer Natur sein und damit in einer Wahrscheinlichkeitsfunktion über einem diskreten Merkmalsraum resultieren. Seq2Seq-Netze auf Basis von LSTM-Zellen wurden in [105] detailliert untersucht und konnten sowohl Kurzzeitabhängigkeiten über einen Horizont von Sekunden als auch Langzeitabhängigkeiten über Tage robust vorhersagen, wohingegen klassische LSTM bei der Modellierung von kurzfristigen Zeitvarianzen höhere Ungenauigkeiten aufweisen. Das Netz ist in Abbildung 3.6 dargestellt.



Abbildung 3.6: Seq2Seq-Netz mit LSTM-Zellen zur Vorhersage von Ausgabesequenzen variabler Zeithorizonte nach [105]

Die blauen Pfeile visualisieren die Fehlerrückführung beim Training durch den Backpropagation through time (BPTT) Algorithmus. In [106] wird ein "many-to-one" Seq2Seq-Netz für das Komprimieren großer historischer Datenreihen genutzt, das diese in eine Sequenz der Länge eins überführt. Diese dient wiederum als Startwert eines "many-to-many" Seq2Seq-Netzes, das auf Basis der Komposition aktueller Eingänge und des Startwertes eine Ausgabesequenz erzeugt. In einer weiteren Anwendung wird in [107] die Vorhersage multipler Größen auf Seq2Seq-Netze übertragen, sodass eine multivariate Vorhersage über multiple Zeitschritte durchgeführt werden kann. Die Eingangssequenz besteht aus Merkmalen, die die Informationen der multivariaten Zeitreihe über einen definierten Zeithorizont kapseln. Es kann im Gegensatz zu den zuvor erörterten Methoden nicht nur eine simultane Vorhersage multipler Größen realisiert, sondern der Vorhersagehorizont unabhängig von der Dimension der Eingabesequenz beliebig variiert werden.

Auch zu Seq2Seq sind bereits einige Architekturvarianten entstanden, von denen die zur Erfüllung der bestehenden Anforderungen relevanten Varianten näher betrachtet werden. Bei Seq2Seq-Netzen wird die vollständige Eingabe über einen Vektor definierter Länge in den Decoder kopiert.

In [108] wird zur Flexibilisierung dieses Kopiervorgangs ein Mechanismus eingeführt, der vom aktuellen Kontext abhängige Verteilungsfunktionen auf die Eingabe anwendet, diese dadurch gezielt komprimiert und an definierte Stellen des Decoders kopiert. Da bei klassischen Seq2Seq als auch bei diesem Ansatz alle Eingabewerte im Encoder verarbeitet und alle Ausgaben im Decoder berechnet sein müssen, um eine weitere Ausgabe zu erzeugen, ist die Echtzeitfähigkeit begrenzt. In [20] wird die Problematik durch Zerlegung des Seq2Seq in kleinere Seq2Seq gelöst. Es wird bereits auf Basis des ersten Eingabewertes eine probabilistische Ausgabesequenz erzeugt, die mit jeder weiteren Eingabe sowie rückgekoppelter Werte verfeinert wird. So kann zu jeder Zeit eine zu erwartende Ausgabe ermittelt werden. Ein Ansatz zur Nutzung von Mechanismen der ,Aufmerksamkeit<sup>\*</sup> in Seq2Seq wird von [109] vorgestellt. Hier wird eine auf probabilistischen Gewichtungen basierende Abbildung der Eingabesequenz auf jede einzelne Ausgabe vorgestellt, wobei die Gewichte von Korrelationen zwischen den Eingangsgrößen bestimmt werden. Seq2Seq wurden ursprünglich für die Sprach- und Bildverarbeitung [110] entwickelt, haben sich jedoch auch als geeignet für die Zeitreihenvorhersage [109] und die Modellierung zeitvarianter Systeme [111] erwiesen. Ein empirischer Vergleich verschiedener Varianten ist in [112] zu finden.

## 3.1.5 Bewertung der Ansätze

Tabelle 3.1 zeigt eine Übersicht zur Bewertung der LSTM-Ansätze anhand der Anforderungen. Im Gegensatz zu den weiteren Architekturen kann mit Seq2Seq-LSTM-Ansätzen eine simultane Vorhersage multipler Größen über eine variabel einstellbare Sequenzlänge durchgeführt werden. Die in Abschnitt 3.1.3 eingeführten Ansätze können lediglich durch Rückführung vorhergesagter Ausgaben Prädiktionen weiterer Schritte realisieren. Die dadurch auftretende Fehlerfortpflanzung findet bei Seq2Seq-LSTM-Ansätzen nicht in der Form statt, da die mehrstufige Vorhersage in einem einzigen Vorgang ohne Rückführungsmechanismus realisiert werden kann. Zusätzlich sind sie prädestiniert für mehrstufige Prädiktionen mit variabel einstellbarem Horizont. Grid LSTM-Netze können kooperativ mehrere Größen mit hoher Präzision vorhersagen. Sowohl hierarchische als auch Autoencoder-basierte LSTM-Konzepte liefern Mechanismen für eine verbesserte Propagation des Zellzustands zur Verarbeitung mehrdimensionaler Daten. Alle Ansätze sind in der Lage, nichtlineares zeitvariantes Verhalten genau vorherzusagen, Seq2Seq-LSTM-Ansätze mit optimiertem Kopiermechanismus des Zustandsvektors zeigen hingegen die beste Eignung, um sowohl Kurz- als auch Langzeitabhängigkeiten zu modellieren. Die wichtigsten Ergebnisse können wie folgt zusammengefasst werden:

• LSTM-Architekturen, die auf eine optimierte Darstellung des Zellzustands abzielen, wie z.B. hierarchische LSTM-Netze und Ansätze zu Aufmerksamkeits-Mechanismen, zeigen eine verbesserte Fähigkeit zur Verarbeitung multidimensionaler Datenstrukturen.

- LSTM-Architekturen, die auf eine Interaktion von regulären Zellzuständen abzielen, wie z.B. Grid und Cross modal LSTM, sind in der Lage, eine kooperative Vorhersage multipler Größen mit hoher Präzision durchzuführen.
- Seq2Seq-LSTM-Architekturen können multiple Größen über einen variabel einstellbaren Zeithorizont mit minimierter Fehlerfortpflanzung vorhersagen.
- Darüber hinaus zeigen Seq2Seq-LSTM-Ansätze mit optimiertem Kopiermechanismus die beste Eignung, sowohl kurzfristige als auch langfristige Abhängigkeiten zu prädizieren.

	Bidirek- tionale LSTM	Hier. & Aufm. LSTM	CNN- LSTM	LSTM- Auto- encoder	Grid LSTM	Cross modal LSTM	Reguläre Seq2Seq	Seq2Seq mit Kop Mech.
Prädiktion nichtlinearer zeitvarianter Abhängigkeiten	+	+	+	+	++	++	++	++
Kurzzeit- und Langzeitgedächtnis- verhalten	0	+	0	+	+	++	+	++
Mehrdimensionale Datenverarbeitung	+	++	+	++	+	+	+	+
Prädiktion multimodaler bzw. multivariater Abhängigkeiten	-	0	+	+	++	++	+	+
Prädiktion multipler Zeitschritte und Fehlerfortpflanzung	-	-		-	0	-	++	++

Tabelle 3.1: Bewertung der Netzarchitekturen hinsichtlich der Anforderungen

Erfüllungsgrade der Anforderungen: -- sehr gering, - gering, 0 durchschnittlich, + hoch, ++ sehr hoch

Anwendungen und Weiterentwicklungen im Kontext der diskreten Fertigung hinsichtlich der Modellierung zeitvarianter Prozesse sowie Anomalien existieren bislang nicht. Es wurden bislang nur die konventionellen LSTM-Ansätze zur Modellierung zeitvarianter Prozesse getestet. Die in den Abschnitten 3.1.3 und 3.1.4 vorgestellten und aufgrund ihrer Eigenschaften vielversprechenden Ansätze wurden bislang nicht zur Modellierung von Prozessen eingesetzt.

# 3.2 Kompensation anomaler Systemdynamiken

In den vorangegangenen Kapiteln wurde ein Überblick über den Stand der Wissenschaft und Technik zur neuronalen Prozessmodellierung gegeben. Um das bessere Prozessverständnis zur Optimierung der Produktqualität nutzen zu können, werden im Folgenden Ansätze zur adaptiven Qualitätsregelung mit Anomalie- bzw. Störungs-Kompensation vorgestellt. Diese entspringen unter anderem regelungstechnischen Anwendungen, da in der Regelungstechnik Methoden zur Verfügung stehen, die eine Unterdrückung von Störverhalten ermöglichen. Es werden Ansätze für mehrstufige Prozessketten sowie modellbasierte prädikative Ansätze eruiert, da sie auf Basis einer Vorhersage das zukünftige Prozessverhalten in die Einstellung der Stellgrößen einbeziehen. Adaptive Ansätze im Sinne des Concurrent Learning verbinden die Auswertung großer historischer Datenmengen mit Modellanpassungen im Betrieb [113]. Das gemeinsame Ziel aller untersuchten Ansätze ist der Aufbau eines diskreten bauteilbezogenen Regelungsschemas zur Kompensation von Störungen und Anomalien auf der Basis eines nichtlinearen und zeitvarianten Prädiktionsmodells des Prozesses. In diesem Beitrag werden die bestehenden Ansätze hinsichtlich der folgenden Anforderungen untersucht:

- Diskrete und bauteilbezogene Eingriffsmöglichkeit in das Prozessgeschehen.
- Berücksichtigung der Mehrstufigkeit der Prozesskette.
- Integration von Beobachtung, Prädiktion und Kompensation von Anomalien.
- Adaptivität und Lernfähigkeit des Modellierungsansatzes.
- Verwendbarkeit historischer Daten zum Vortraining der Modelle.

Durch den Abgleich bestehender Ansätze mit den Anforderungen soll aufgezeigt werden, welche Ansätze die beste Eignung zur Lösung der zu Grunde liegenden Problemstellung aufweisen und welcher weitere Forschungsbedarf hinsichtlich der Ansätze zur Unterdrückung von anomalen Systemdynamiken besteht.

## 3.2.1 Diskrete bauteilbezogene Ansätze

Die klassische Regelungstheorie liefert mit den Konzepten zur diskreten Regelung, bspw. mit Moving-Average-basierten (MA-basierten) Regelungsschemata, die theoretische Grundlage für deren Anwendung auf diskrete technische Systeme. Diskretisierung im Kontext der Fertigung bedeutet eine bauteilbezogene Diskretisierung, die ein Eingreifen in den Prozess erst nach jeder Prozessdurchführung durch das Festlegen neuer Sollwerte für die Prozessparameter vorsieht. Der Einsatz übergeordneter, bauteilbezogener Regelungen zur gezielten Verbesserung von Prozesstabilität und Produktqualität hat in der diskreten Fertigung bislang noch Seltenheitswert. In einigen Segmenten der diskreten Fertigungstechnik, die durch hohe Qualitätsanforderungen gekennzeichnet sind, wie bspw. der Halbleiterfertigung, hat sich die diskrete Regelung der Qualität unter dem Schlagwort ,Run-to-Run Control' bereits als Stand der Technik etabliert.

In [114] wird eine auf einem erweiterten Zustandsbeobachter (engl. ,Extended state observer') basierende Regelung zur Optimierung eines Halbleiterfertigungsprozesses verwendet. Störgrößen werden in die Zustandsmodellierung integriert, sodass eine dezidierte Störgrößenunterdrückung erreicht werden kann. Auch in [115] wird ein erweiterter Zustandsbeobachter innerhalb eines ,Run-to-Run Control'-Schemas genutzt, das gezielt Informationen vorangegangener Bauteile zum Ausgleich abrupter sowie schleichender Schwankungen für nachfolgende Bauteile einsetzt. Eine weitere Methode ist die Exponentially Weighted Moving Average (EWMA) Regelung, die die Prozess- und Stellhistorie für eine dynamische Sollwertanpassung nutzt. Auf Basis einer E/A-

basierten Prozessmodellierung wird in [116] ein EWMA-Schema entworfen, das durch den Einsatz eines lernfähigen Softsensors fehlende Qualitätsmessungen ausgleichen kann, sodass eine Sollwertanpassung für die Fertigung des nächsten Bauteils ermöglicht wird. In [117] wird der Ansatz auf die Batch-Fertigung übertragen. Ein Batch wird mit Hilfe des k-Means Clustering-Algorithmus in Bauteilgruppen mit minimierter interner Varianz aufgeteilt, sodass einheitliche Stellgrößen zu einer Störgrößenunterdrückung für alle Bauteile der Gruppe führen. Für die Übertragung des Ansatzes auf mehrstufige Prozessketten kann nach [118] zwischen der werkzeugbezogenen Regelung eines Prozesses sowie der bauteilbezogenen Regelung entlang der Prozesskette unterschieden werden. Beim erstgenannten wird die Qualitätsabweichung bei einem Prozess über alle Bauteile minimiert. Beim zweiten Fall wird diese hingegen bei einem Bauteil über alle Prozesse minimiert. Eine Illustration der beiden Varianten ist in Abbildung 3.7 zu sehen.



Abbildung 3.7: EWMA-Regelung bezogen auf einen Prozess über alle Bauteile (oben) sowie ein Bauteil über alle Prozesse (unten) nach [119]

In [120] wird eine weitere Erweiterung des Ansatzes um einen Beobachter zur Schätzung und Kompensation von Störungen beschrieben, sodass bei vorangegangenen Prozessen entstehende Schwankungen bei noch ausstehenden ausgeglichen werden können. Ein Überblick über das Thema ,Run-to-Run Control' in der diskreten Halbleiterfertigung wird in [119] gegeben. Aber auch in einigen Anwendungen abseits der Halbleiterfertigung sind bereits bauteilbezogene Regelungen zum Einsatz gekommen. In [121] wird eine bauteilbezogene Sollwertanpassung auf Basis von selbstorganisierenden Karten für einen verfahrenstechnischen Prozess vorgestellt, die durch Lösung des auf dem Prozessmodell aufbauenden Optimierungsproblems zur Minimierung von Qualitätsschwankungen beiträgt. Zur Herstellung von Kohlenstofffasern wird eine exakte Einstellung der Prozesstemperatur benötigt, die in [122] durch eine iterative Anpassung der Regelungsparameter, basierend auf einer nichtlinearen Optimierung unter Nebenbedingungen mit inversem Prozessmodell, realisiert wird. Auch in der diskreten Fertigung gibt es Anwendungen bauteilbezogener Regelungen. In der inkrementellen Blechumformung muss bei jedem Bauteil eine Rückverformung bei der Einstellung der Zielgeometrie berücksichtigt werden, die in [123] durch eine Trajektorienregelung, basierend auf einem linearisierten Prozessmodell, geregelt wird.

Im Bereich der Umformtechnik sind aus numerischen Prozessmodellen extrahierte und durch Linearisierung vereinfachte Modelle nicht hinreichend genau, um die Komplexität der örtlich und zeitlich verteilten Physiken abzubilden. Daher wird entweder unter starken Restriktionen hinsichtlich der Echtzeitfähigkeit direkt mit den numerischen Modellen gerechnet oder es wird auf numerische Simulationsdaten zur Erzeugung echtzeitfähiger E/A-Modelle zurückgegriffen. In [124] wird gemäß dem ersten Fall ein numerisches Prozessmodell zur Auslegung einer iterativ lernenden Regelung herangezogen, die sich durch ein definiertes Gedächtnisverhalten hinsichtlich der zurückliegenden Produktionshistorie auszeichnet. Ein Neuronales Netz lernt und optimiert die Sollwertanpassung mit jedem Bauteil. Darauf aufbauend wird in [125] dieser übergeordnete Regelkreis durch einen Zustandsbeobachter für während des Prozesses entstehende geometrische Divergenzen erweitert. In [126] wird ein weiterer iterativ lernender Ansatz zur bauteilbezogenen Regelung vorgestellt, der zusätzlich den Vorteil einer effizienten Störgrößenunterdrückung aufweist. Störungen bestehen hier durch hohe Varianzen in der geometrischen Beschaffenheit der Bauteile, deren Einfluss auf das Qualitätsergebnis durch eine adaptive Anpassung der Stellgrößen reduziert werden kann. In [127] wird die Übertragbarkeit der Ansätze auf weitere geometrische Qualitätsmetriken untersucht. Die Arbeiten beschränken sich jedoch auf den Einsatz ineffizienter numerischer Modelle, die mit hohem Rechenaufwand verbunden sind und die durch die ausschließliche Betrachtung geometrischer Qualitätsmetriken in ihrer Aussagekraft Restriktionen unterliegen.

Der zweite zuvor eingeführte Fall der Reduktion numerischer Modelle zu echtzeitfähigen nichtlinearen (E/A-)Modellen auf Basis extrahierter Daten ist die Grundlage der Arbeiten von [3]. Hier wird ein auf verschiedene Prozesstypen übertragbarer Modellierungsansatz auf Basis von AR- und MA-Modellen vorgestellt, der die allgemeine Zustandsraumdarstellung nachbildet. Für die Zustandsdifferenzengleichung wird ein Beobachter eingeführt, der mit Hilfe der Historie gefertigter Bauteile sowie aktueller Stellgrößen den Prozesszustand schätzt. Die Regelung erfolgt modellbasiert und prädikativ und berücksichtigt Modellunsicherheiten und externe Störeinflüsse. Die Arbeit führt auf, dass es für das inkrementelle Umformen noch keine Untersuchungen zum Einsatz von diskreten Regelungen der Qualität gibt.



Abbildung 3.8: Überlagerung einer zeitkontinuierlichen Prozessgrößenregelung und einer diskreten bauteilbezogenen Regelung nach [4]

Ein darauf aufbauendes Konzept ist in [4] entwickelt worden. Das Regelungsschema wird hier für eine zeitkontinuierliche Einstellung definierter Parameterkurven während des Prozesses und eine übergeordnete diskrete Regelung von Bauteil zu Bauteil eingesetzt. Damit wird ,Run-to-Run Control' mit modellbasierten prädikativen Ansätzen verbunden. Abbildung 3.8 zeigt die Überlagerung der Regelkreise. Eine Schwäche des Ansatzes ist das linearisierte Zustandsraummodells, das bei größeren Änderungen des Arbeitspunktes ungenau wird. Mit Arbeitspunktänderungen ist beim diskreten Regelkreis aufgrund von Schwankungen der Prozessausgänge vorheriger Prozesschritte, wie bspw. sich sprunghaft ändernder Bauteilgeometrien, jedoch zu rechnen, sodass eine Betrachtung der gesamten Prozesskette unerlässlich ist, um diese Varianzen schätzen und in die Modellbildung integrieren zu können.

#### 3.2.2 Ansätze für mehrstufige Prozesse

Um der Anforderung der Mehrstufigkeit gerecht zu werden, wird in [128] eine Übersicht an Ansätzen zur Regelung für mehrstufige Systeme und Prozessketten beschrieben. Dabei wird von einer sequentiellen Abfolge von Einzelprozessen ausgegangen, die durch Zustandsraummodelle beschrieben und über den Prozesszustand verknüpft werden. Eine Optimierungsstrategie zum Ausgleich von Qualitätsschwankungen entlang der Prozesskette wird nicht konkretisiert. In [129] wird der Ansatz für mehrstufige Batch-Prozesse erweitert. Dabei werden nach jedem Prozess Produkte mit ähnlichen Eigenschaften durch ein Clustering neu gruppiert. So führt die minimierte Varianz der Produkteigenschaften innerhalb der Batches zu einer genaueren Modellierung der Abhängigkeiten zwischen den Prozessen. In [8] werden die Einzelprozesse von lokalen Reglern innerhalb des kaskadierten Regelungsschemas kontrolliert, die die Abhängigkeiten zu vor- und nachgelagerten Prozessen berücksichtigen. Äußere Regelkreise liefern die Referenzen für innere Regelkreise, sodass die Prozesskette auf das optimale Ergebnis des äußersten letzten Prozesses geregelt werden kann. Dieser Ansatz ist in Abbildung 3.9 visualisiert. Auch in [130] werden lokale Regler zur Kontrolle von Subsystemen in einem verteilten Regelungsschema verwendet. Die Verknüpfung der Regler wird durch eine übergeordnete Koordinationsinstanz vorgenommen, die regelbasiert agiert. Die Kontrolle von mehrstufigen Prozessketten der diskreten Fertigung stellt weiterhin eine große Herausforderung dar, wie in [131] hervorgehoben wird.



Abbildung 3.9: Kaskadiertes Regelungsschema zur Kompensation von Wechselwirkungen vorund nachgelagerter Prozessschritte nach [8]

In [131] wird ein weiterer Modellierungsansatz für serielle Produktionslinien vorgestellt, der durch die Integration von stochastischen Verlässlichkeits- und Verschleißmodellen einzelner Komponenten eine verteilte Regelung unter Berücksichtigung langfristig auftretender Störeffekte ermöglicht. In [132] wird ein Ansatz einer verteilten modellbasierten Regelung auf Basis eines ähnlichen Prozesskettenmodells vorgestellt, wohingegen hier auf Sprungprozesse nach Markow zurückgegriffen wird, die über die Produkteigenschaften verknüpft sind und diese als Störung modellieren. In [29] ist die entsprechende Anwendung des Konzeptes auf eine Prozesskette der Umformtechnik beschrieben. Das verteilte Regelungsschema erlaubt einen Austausch von Informationen entlang der Prozesskette, die dann in der modellbasierten Vorhersage der einzelnen Prozesse berücksichtigt und die Stellgrößen entsprechend optimiert werden. Das ermöglicht eine Regelung der Prozesskette und einen Ausgleich von Varianzen. Eine Prädiktion und Optimierung entlang der Prozesskette unter Berücksichtigung von Anomalie-Modellen findet jedoch nicht statt.

Die Optimierung mehrstufiger Prozessketten und die Kontrolle ganzer Produktionsanlagen ist Bestandteil der Forschung an flexiblen Produktionssystemen (FPS). Bei den im Folgenden untersuchten Arbeiten liegt der Fokus weniger auf der sauberen Ausarbeitung eines dezidierten Regelungsschemas als vielmehr auf der Einführung übergeordneter Konzepte zur Optimierung von Prozess- und Produktqualität, die auf Basis lokal an den einzelnen Maschinen wirkender Regelungsmechanismen umgesetzt werden können. In [52] wird dazu für jeden Prozessschritt ein Modell gelernt, das die relevanten Qualitätsmetriken vorhersagen und den Prozesszustand stetig adaptieren kann. Auf Basis des Gesamtmodells der Prozesskette wird dann eine Optimierung zur Generierung von Anpassungsvorschlägen der Prozessparameter durchgeführt. Um die Pareto-Front der optimalen Lösungen effizient zu erreichen, wird ein Mechanismus zur Reduktion der Dimension des Zustandsraumes eingesetzt. Ein ähnlicher Ansatz wird in [133] vorgestellt. Die hohe Dimension des Optimierungsraumes und der damit einhergehende Rechenaufwand wird hier durch parallel und unter variierenden Bedingungen ausgeführte Simulationen gehandhabt, die zu optimalen Anpassungen der Prozessparameter führen. Im Gegensatz zu [29] wird in [134] kein stochastisches sondern ein deterministisches Modell zur Beschreibung der Prozesskette bzw. des FPS herangezogen. Die Kombination aus übergeordneter Optimierung und überwachender, prädikativer Qualitätsregelung führt zur Vermeidung instabiler Zustände. Eine Übersicht über verschiedene Verfahren zur Regelung und Optimierung flexibler Produktionssysteme wird in [135] gegeben. Hier wird zwischen hierarchischen, semi-hierarchischen, zentralisierten und heterarchischen bzw. verteilten Architekturen unterschieden. Der Fokus der Kontrollarchitekturen liegt im Gegensatz zu den Anwendungen aus der klassischen Regelungstechnik in der Produktorientierten Optimierung von Stellgrößen. Allerdings werden nur in [52] adaptive und lernfähige Prozessmodelle verwendet, die sich dynamisch an Änderungen anpassen. Die Umsetzung eines definierten Gedächtnisverhaltens beim Lernen findet jedoch nicht statt.

#### 3.2.3 Modellbasierte prädikative Ansätze

Modellbasierte prädikative Ansätze zeichnen sich dadurch aus, dass sie unter Zuhilfenahme eines möglichst genauen Prozessmodells die zukünftigen Prozessausgänge über einen definierten Zeithorizont vorhersagen und diese durch eine Stellgrößenoptimierung an eine Sollwerttrajektorie angleichen. Dazu wird eine, die Sollwertabweichungen und Stellaufwände beinhaltende. Kostenfunktion unter Nebenbedingungen, wie Stellgrößenbeschränkungen, minimiert [136]. Die berechneten Stellgrößen berücksichtigen somit eine zu jedem Zeitpunkt bestmögliche Schätzung des zukünftigen Prozessgeschehens möglichst unter Einbeziehung potentieller Störeffekte. Da hinsichtlich fertigungstechnischer Prozesse keine analytischen Modelle verfügbar sind, die sich für diesen Ansatz eignen würden [137], muss auf aus numerischen Simulationsdaten extrahierte Metamodelle zurückgegriffen werden [5]. In [138] wird ein inkrementell lernendes MLP zur zeitinvarianten Prozessmodellierung statischer und dynamischer Zusammenhänge genutzt. Dabei werden Störeffekte bei der Prädiktion berücksichtigt, sodass es zu einer Störgrößenunterdrückung kommt. In [139] werden RBFN aufgrund ihrer guten Approximationseigenschaften zur adaptiven Modellierung vernetzter nichtlinearer Prozesse verwendet. Unterschiedliche Übertragungsraten in der Datenverarbeitung der Prozesse und dadurch entstehende Anomalien in Form von Totzeiten können durch den prädikativen Ansatz kompensiert werden. Durch den Einsatz von RNN kann das adaptive Verhalten auch für die Modellierung zeitvarianter Prozesse realisiert werden. So wird bspw. in [140] ein Gedächtnis-gestützter Regelungsansatz basierend auf RNN vorgestellt, der in der Lage ist, stationäres und nicht-stationäres Zeitverhalten zu modellieren. Ein weiterer Ansatz für komplexe Systeme wird in [141] beschrieben, bei dem mit RNN eine reduzierte Repräsentanz gelernt wird, in der nur die für das Regelungsergebnis relevanten Prozesszusammenhänge gekapselt sind. Im Vergleich zu RNN mit klassischen Aktivierungsfunktionen zeigen LSTM-Netze eine erhöhte Prädiktionsgüte kurzfristiger sowie längerfristiger Dynamiken. In [86] wird dazu ein modellbasiertes prädikatives Regelungsschema basierend auf LSTM-Netzen vorgestellt. Diese werden auf eine niederdimensionale, lineare Modellstruktur in Zustandsraumdarstellung reduziert, die vom aktuellen Arbeitspunkt abhängig ist und iterativ weiterlernt. Trotz der guten Approximation verschiedener zeitvarianter Prozesscharakteristiken werden externe Störeinflüsse bei den beschriebenen Ansätzen nur indirekt über den Prädiktionsfehler des Netzes erfasst und über den Lernvorgang in das neuronale Modell integriert. Es findet keine direkte Erfassung und Charakterisierung der Störungen statt.

Um eine Unterscheidung von Modellunsicherheiten und verschiedenen Störungstypen vornehmen und diese besser charakterisieren zu können, werden Beobachterstrukturen verwendet. Die erfassten Störungen können auf zwei Arten zur Kompensation herangezogen werden. Im ersten Fall wird ein direkter Ausgleich durch die Erzeugung und Aufschaltung eines inversen Kompensationssignals angestrebt, sodass der Prozess auf Basis des regulären störungsfreien Prozessmodells geregelt werden kann. Im zweiten Fall werden die beobachteten Störungen zum Erlernen eines Störmodells verwendet, das die Genauigkeit der Prädiktion erhöht und damit zu einer Störgrößenunterdrückung beiträgt. Eine detaillierte Charakterisierung von internen und externen Störeffekten wird in [142] vorgenommen. Gemäß dem ersten Fall wird durch das Aufschalten der inversen Dynamik beobachteter Störungen eine direkte Unterdrückung erreicht. Dadurch wird der Prozess frei von störenden Interferenzen, sodass eine prädikative Regelung auf Basis des normalen Prozessmodells ohne Störungsmodellierung erfolgen kann. In [143] wird dieser Ansatz durch die Einführung eines diskontinuierlichen Übertragungsglieds zwischen Beobachter und Regler flexibilisiert, das die Störungsschätzung als Eingabe erhält und zwischen verschiedenen Betriebsmodi des Reglers umschaltet. Dadurch kann je nach beobachteter Störung eine bestmögliche Kompensation erfolgen, da der Regler sich an das dynamisch ändernde System anpasst und der Prozessausgang stetig entlang einer Referenztrajektorie gleiten kann. Um für nicht invertierbare Systeme eine Kompensation zu erreichen, liefern [144] einen alternativen Ansatz, der eine Reduktion des Systems in invertierbare Subsysteme vorsieht. Dazu werden inverse Prozessmodelle mit RBFN gelernt, die zur Störungsbeobachtung dienen. Die geschätzten Störungen aller Subsysteme werden in ein ganzheitliches Kompensationssignal umgesetzt. In [145] wird das Konzept um die separate Modellierung und Kompensation des Approximationsfehlers des Netzes erweitert, was einem zusätzlichen Ausgleich der internen Modellunsicherheit entspricht.

Ein Beobachter ist auch die Grundlage im Regelungsschema von [13], bei dem im Gegensatz zu den zuvor beschriebenen Ansätzen das neuronale Prozessmodell gemäß dem zweiten Fall um eine Modellierung der Störungen erweitert wird. Dabei werden die vergangenen und vorhergesagten Beobachtungen als direkte Netzeingänge in einem rekursiven Prädiktionsschema verwendet. Die Komposition aus Störungs- und Prozessmodell führt zu einer genaueren Vorhersage und folglich zu einer optimierten Kompensation der Störeffekte. Abbildung 3.10 zeigt das zugehörige Schema. Auch in [146] wird ein ähnliches Konzept beschrieben mit dem Unterschied, dass die Schätzung und Vorhersage des Störverhaltens durch einen erweiterten Zustandsbeobachter in einem vom Prozessmodell unabhängigen Modell erfolgt. Im Rahmen der Arbeiten von [147] wird der Ansatz auf multiple Prozesse erweitert, wobei Modellunsicherheiten in Form von Unsicherheiten in den Parametern im Fokus des Lernproblems stehen. Ein Ansatz ohne den Einsatz neuronaler Modelle ist in [148] beschrieben. Hier werden auf den Prozessausgang wirkende Störungen mit stochastischen Modellen abgebildet. Dabei werden stationäre rauschartige Störungen mit Zufalls-Prozessen modelliert und mit einem affinen Prozessmodell kombiniert. Weiterhin wird zwischen Anlaufschwankungen, Schwankungen zwischen Aufträgen und Produktionsschwankungen innerhalb von Aufträgen unterschieden, die durch nicht-stationäre sprunghafte Störungen gekennzeichnet sind. In [14] wird hingegen ein Beobachter zur Erfassung von auf die Stellgrößen wirkenden Störungen eingeführt, die somit einen anderen Angriffspunkt aufweisen als in [148]. Der Störeffekt wird analog zu Kapitel 2.2.1 aus der gefilterten Differenz von Stellgröße und rückpropagiertem Prozessausgang ermittelt. Im Rahmen der MPC wird neben dem Prozessmodell ein separates autoregressives Störmodell für eine ganzheitliche Vorhersage verwendet. Dazu werden zwei verschiedene Netze für eine verbesserte Kompensation des Störverhaltens genutzt.

Ein neuer Ansatz zur prädikativen Beobachtung unbekannter, zeitvarianter Systeme wird in [149] eingeführt. Dieser schlägt eine Komposition aus Beobachter und Prädiktor zur unmittelbaren Schätzung zukünftigen Störverhaltens vor. Der Prädiktor wird in ein Schema zur erweiterten Zustandsbeobachtung integriert, sodass stationäre und nicht-stationäre Dynamiken kompensiert werden können.



Abbildung 3.10: Komposition aus der Beobachtung auftretender Störungen und der darauf aufbauenden Prädiktion mit neuronalem Prozessmodell nach [13]

Die in diesem Abschnitt beschriebenen Ansätze sind lernfähig und zeichnen sich durch gute Prädiktionseigenschaften aus. Im Gegensatz zu den in Kapitel 3.2.1 vorgestellten Arbeiten integrieren sie die Modellierung von Störverhalten für eine effiziente Unterdrückung desselben, berücksichtigen jedoch keine Abhängigkeiten entlang mehrstufiger Prozessketten. Diese Lücke wird hingegen von den Ansätzen aus Kapitel 3.2.2 abgedeckt, sodass eine Komposition aus den vorgestellten Ansätzen zu einer Lösung der dieser Arbeit zu Grunde liegenden Problemstellung beitragen könnte.

### 3.2.4 Adaptive Ansätze

Die zuvor vorgestellten Konzepte zur Kompensation von Störungen zeichnen sich dadurch aus, dass durch neuronale Beobachter- und Prädiktionsschemata eine Unterdrückung erzielt wird und ein vorhandener Regler beibehalten werden kann. Adaptive Regelungen sehen hingegen einen flexiblen Regler oder eine Zusammensetzung von Reglern vor, die ohne zusätzliche Strukturen auskommen und daher selbst als neuronale Netze realisiert sein können. In [150] wird ein hybrider Ansatz mit einem Ensemble an Netzen, die auf verschiedene Prozesssituationen trainiert wurden, getestet. Das Gesamtverhalten des Reglers verschiebt sich immer in die Richtung, die dem aktuellen Zustand am besten entspricht. Die diskrete Auswahl eines der Netze aus dem Ensemble entspricht der Integration eines diskontinuierlichen Übertragungsgliedes und einem dadurch erzeugten gleitenden Zustand des Gesamtsystems. Mit einer solchen variabel strukturierten Regelung wird auch in [151] gearbeitet. Die Dynamiken des variablen Gesamtsystems werden

kontinuierlich mit einem Referenzmodell abgeglichen und steuern die Umschaltung zwischen den Reglern. Zusätzlich verhalten sich die als RBFN umgesetzten Regler durch einen inkrementellen Lernansatz selbst adaptiv. Auch in [152] werden RBFN für eine stetige Adaption des dynamischen Prozess- und Störverhaltens verwendet, die hier ausschließlich über einen Online-Lernprozess ohne Umschaltung erfolgt. Die Konvergenz des Lernvorgangs in endlicher Zeit kann für eine definierte, endliche Umgebung des Zielwertes nachgewiesen werden. Eine Kombination aus adaptivem Regelungsschema und neuronalem Beobachter ist in [153] gegeben, das ebenfalls ohne Schaltmuster auskommt. Die nichtlineare dynamische Inversion wird zur Schätzung der auf die Aktoren wirkenden Störungen genutzt, die direkt zur Adaption des Reglers eingebracht werden. In [154] wird eine variabel strukturierte Regelung mit RNN realisiert, um zeitliche Abhängigkeiten in den Datenstrukturen abzubilden. Die Umschaltung zwischen den Reglern wird durch eine flexible Umschaltung freier Ordnung realisiert. Bei einer verteilten Umschaltung wird von einer Zerlegung des Gesamtprozesses in Teilprozesse ausgegangen. Mit dem Ansatz ist es möglich, verschiedene Prozesse entlang einer Prozesskette zu kontrollieren.

Adaptive Systeme können sich nur dann ausreichend genau der Realität anpassen, wenn sie beständig durch neue Daten angeregt werden. Das gilt besonders für Online-Lernvorgänge. Sind bestimmte Bereiche des Eingangsdatenraumes nicht in repräsentativem Maße in den Daten vorhanden, kann die angestrebte Konvergenz des Lernvorgangs nicht erreicht werden. Um zum Anfangszeitpunkt bereits eine ausreichende Störgrößenkompensation sowie hohe Regelgüte erzielen und durch den Lernprozess verursachte Oszillationen vermeiden zu können, führen [113] das Konzept des 'Concurrent Learning' ein. Dieses sieht den Einsatz vortrainierter neuronaler Regler vor, die zu Beginn der Regelung durch die bereits vorhandenen Wissensstrukturen zu einer verbesserten Stabilität sowie Extrapolationsfähigkeit beitragen, die bei sicherheitskritischen Anwendungen von hoher Relevanz sind. Dadurch kann es trotz nicht beständiger Anregung zu einer Konvergenz kommen. Das Konzept wird in [155] mit experimentellen Daten auf Basis einer adaptiven Regelung mit Referenzmodell evaluiert. In [156] wird ein Vorgehen zur Erzielung einer möglichst hohen Generalisierbarkeit der vortrainierten Netze präsentiert. Dabei wird beim Vortraining nicht nur ein Netz, sondern eine größere Anzahl an unterschiedlich strukturierten Netzen einbezogen, die parallel trainiert und im Rahmen einer Kreuzvalidierung getestet werden. Von der Netzstruktur mit dem besten Extrapolationsverhalten werden weitere strukturelle Varianten erzeugt, die in der nächsten Iteration wieder parallel trainiert und gegeneinander getestet werden. Werden die vorgeschriebenen Gütekriterien erreicht, bricht das Verfahren ab und das vortrainierte Netz kann im Betrieb eingesetzt werden und dort weiterlernen. Da eine gleichzeitige Nutzung historischer und aktueller Daten stattfindet, ist eine Speicherung der historischen Daten notwendig, da diese nicht umgehend in die Modelle integriert werden. In [54] werden dazu "History Stacks' gebildet, die die zu diskreten Zeitpunkten erfassten Stell- und Prozessgrößen verwalten. Dies ist in Abbildung 3.11 für zwei Prozesszustände visualisiert. Informationen, die bei einer direkten Verarbeitung der erfassten Daten nicht vorhanden sind, können nachträglich in den Adaptionsprozess integriert werden.



Abbildung 3.11: Datenspeicherung zur Umsetzung des Concurrent Learning [54]

In [157] wird ein Ansatz geliefert, der eine Integralform der ursprünglichen Concurrent Learning Problematik liefert und dadurch die Notwendigkeit der Integration von Zustandsableitungen in die Datenspeicherung überflüssig macht. Der Ansatz des Concurrent Learning ergänzt die in den vorangegangenen Kapiteln eingeführten Ansätze um eine wesentliche Komponente, die zur Robustheit adaptiver Regelungen und zu einer verbesserten Kompensation von Störungen beiträgt. Das Vortraining neuronaler Modelle bietet einen wesentlichen Vorteil hinsichtlich fertigungstechnischer Anwendungen, da große Mengen an historischen Daten nutzbar gemacht werden können.

## 3.2.5 Bewertung der Ansätze

Auf Basis der in den vorherigen Kapiteln beschriebenen Ansätze ist in Tabelle 3.2 eine Übersicht über die Bewertung der Ansätze hinsichtlich der aufgestellten Anforderungen zu sehen. Die wesentliche Erkenntnis ist der Umstand, dass sich die beste Erfüllung der jeweiligen Anforderungen in etwa auf der Diagonalen der Tabelle ergibt. Eine diskrete bauteilbezogene Regelung ist mit allen Ansätzen adäquat realisierbar. Verteilte kaskadierte Regelungsschemata sowie die im Kontext flexibler Produktionssysteme existierenden Optimierungsansätze eignen sich zur Einhaltung und Verbesserung der Produktqualität entlang mehrstufiger Prozessketten. Um die in Kapitel 3.1 untersuchten Methoden der Detektion und Modellierung von Anomalien für eine verbesserte Kompensation derselben nutzbar zu machen, wird ein Schema benötigt, das diese Methoden integrieren kann. Modellbasierte prädikative Ansätze mit Beobachterstrukturen eignen sich dazu in besonderem Maße. Die Verfügbarkeit diverser Prädiktionsschemata erlaubt die Integration der in Kapitel 3.1 vorgestellten Netzarchitekturen. Darüber hinaus zeichnen sich viele der Schemata durch die Möglichkeit einer stetigen Adaption der verwendeten Modelle aus. Die Modelle selbst können vortrainiert sein und während des Betriebs weiterlernen. Im regelungstechnischen Kontext hat sich für das Vor- und Weitertraining von Prozessmodellen das Concurrent Learning als geeigneter Ansatz etabliert. Für die vorliegende Arbeit bedeutet dieses Bewertungsergebnis, dass eine Komposition der Eigenschaften bestehender Ansätze zu einer optimalen Erfüllung der aufgestellten Anforderungen führen würde. Die wichtigsten in der Tabelle dargestellten Ergebnisse können wie folgt zusammengefasst werden:

- Verteilte und kaskadierte Regelungs- sowie Optimierungsschemata eignen sich in besonderem Maße für die Beherrschung der Komplexität mehrstufiger Prozessketten.
- Modellbasierte Ansätze können flexibel um Beobachterstrukturen erweitert und mit diversen Prädiktionsschemata realisiert werden, die eine Integration der beobachteten Störungen in die Prädiktion erlauben.
- Modellbasierte Ansätze auf Basis neuronaler Prozessmodelle zeigen gegenüber den weiteren Ansätzen deutlich verbesserte Adaptions- und Modellierungseigenschaften zukünftiger Prozessdynamiken.
- Das Concurrent Learning erlaubt die Integration vortrainierter Prozessmodelle und deren Weitertraining während des Betriebs. Dadurch können große historische Datenmengen für eine Steigerung der Modellierungsgenauigkeit nutzbar gemacht werden.

	Run-to- Run Control	Iterative Learning Control	Kaska. & verteilte Ansätze	Opti- mierung von FPS	Neuronale MPC	DOB & neuronale MPC	Variabel strukt. Regelung	Con- current Learning
Diskrete und bauteilbezogene Eingriffsmöglichkeit in den Prozess	++	+	+	++	+	+	+	+
Berücksichtigung der Mehrstufigkeit der Prozesskette	+	+	++	++	0	+	0	+
Integration von Beobachtung, Prädiktion und Kompensation von Anomalien	0	0	0		+	++	0	+
Adaptivität und Lernfähigkeit der Modelle	-	+	0	-	++	++	+	++
Verwendbarkeit historischer Daten zum Vortraining der Modelle	-	+	-	0	0	0	+	++

Tabelle 3.2: Bewertung der Qualitätsregelungen hinsichtlich der Anforderungen

Erfüllungsgrade der Anforderungen: -- sehr gering, - gering, 0 durchschnittlich, + hoch, ++ sehr hoch

Weiterentwicklungen für mehrstufige Fertigungssysteme hinsichtlich der Kompensation von zeitvarianten anomalen Systemdynamiken, die durch eine Erweiterung und Kombination der Ansätze alle Anforderungen erfüllen, existieren bislang nicht.

# 3.3 Aufzeigen des Forschungsbedarfs

Auf Basis des Stands der Wissenschaft und Technik soll die Forschungslücke bzw. der daraus entstehende Forschungsbedarf aufgezeigt werden. Die Automatisierung und die digitale Transformation der Produktion im Sinne von Industrie 4.0 wird durch die Entwicklung neuer Konzepte auch in der anwendungsorientierten Forschung immer weiter vorangetrieben. Datengetriebene Ansätze, die Daten nutzen um aus diesen bisher unbekanntes Wissen zu generieren und selbiges zu einer Produktionsoptimierung sowie folglich zu einer Steigerung der Wertschöpfung einzusetzen, spielen dabei eine übergeordnete Rolle. Der Stand der Forschung und Technik zu datengetriebenen Ansätzen für die Optimierung der Produktqualität zeigt hier noch hohes Potential für weitere Entwicklungen, da es bislang nur eine geringe Anzahl an Ansätzen und Konzepten gibt, die datengetriebene Qualitätsregelungen oder eine datengetriebene Automatisierung der Produktoptimierung in der diskreten Fertigung thematisieren. Für die Integration dieser Art von automatisierungstechnischen Konzepten in die Produktion besteht daher auch auf übergeordneter Ebene noch enormer Forschungsbedarf. Die Untersuchungen dieses Kapitels haben darüber hinaus im Hinblick auf die in Abschnitt 1.2 erläuterte Problemstellung die nachfolgend beschriebene konkrete Forschungslücke ergeben. Die Kapitel 3.1.5 sowie 3.2.5 liefern dazu eine Bewertung der bestehenden Ansätze zur nichtlinearen zeitvarianten Prozessmodellierung und zur Kompensation von Störverhalten in der diskreten Fertigung. Die zweitgenannte Bewertung offenbart, dass es keinen Ansatz gibt, der alle aufgestellten Anforderungen in vollem Umfang erfüllt. Optimierungsbasierte und kaskadierte Ansätze sind besonders geeignet für die Kontrolle diskreter Systeme und erlauben eine bauteilspezifische Optimierung der Einstellparameter unter Berücksichtigung Abhängigkeiten, die entlang mehrstufiger Prozessketten auftreten. Sie zeigen allerdings Schwächen bei der Integration geeigneter Beobachterstrukturen zur Identifikation anomaler Prozessdynamiken. Diesbezüglich gibt es jedoch Ansätze aus dem Bereich des Beobachterentwurfs, die in Kombination mit modellbasierten prädikativen Regelungsschemata eine Identifikation von Anomalien sowie eine Prädiktion des zukünftigen Verhaltens dieser Anomalien erlauben. Da bei diesen Ansätzen die bislang erprobten neuronalen Prozessmodelle primär auf vorwärts gerichteten Netzen oder einfachen RNN beruhen, werden die in Kapitel 3.1.5 erforderlichen Anforderungen an die nichtlineare zeitvariante Prozessmodellierung nicht erfüllt. Daher muss ergänzend die erstgenannte Bewertung herangezogen werden, um das Schema zur Kompensation von Störverhalten auf einer den Anforderungen gerecht werdenden Prozessmodellierung aufzubauen. Es gibt bislang kein Schema, das Prozessmodelle dieser Art integriert und für eine Regelung der Qualität nutzbar macht. Darüber hinaus gibt es kein Schema, das diese Modelle im Sinne des Concurrent Learning adaptierbar macht.

Daher kann die identifizierte Forschungslücke wie folgt formuliert werden:

"Es existiert kein Konzept zur adaptiven Qualitätsregelung mehrstufiger Prozessketten der diskreten Fertigung, die einen Mechanismus zur Detektion und Kompensation unbekannter anomaler Prozessdynamiken integriert, um eine Optimierung der Produktqualität zu erreichen."

Auf Grundlage der identifizierten Forschungslücke wird im nachfolgenden Kapitel das zur Füllung dieser Lücke erarbeitete Konzept vorgestellt.

# 4 Konzept zur adaptiven Qualitätssicherung

Basierend auf der identifizierten Forschungslücke wird in dieser Arbeit ein Konzept für eine adaptive Qualitätssicherung vorgestellt. Dabei werden zusätzlich die in Abschnitt 1.2 erörterten grundlegenden Probleme sowie die in Abschnitt 1.4 aufgezeigten Anforderungen berücksichtigt. Dieses Kapitel beginnt mit einer kurzen Vorstellung der zentralen Konzeptidee, die eine adaptive Qualitätsregelung zur Erfüllung der Anforderungen ermöglichen soll.

# 4.1 Beschreibung der Konzeptidee

Im Folgenden wird auf die Konzeptidee und die grundlegenden Konzeptentscheidungen eingegangen. So werden Fertigungsprozesse im Allgemeinen während des Prozessgeschehens über interne zeitkontinuierliche Regelungen einzelner Prozessparameter kontrolliert. Dies kann bei einem Umformprozess bspw. die Regelung der Kraft auf einen definierten Kurvenverlauf, der während der Prozessdurchführung einzuhalten ist, sein. Eine diskrete bauteilbezogene Regelung würde mit jedem neuen Prozessdurchlauf und damit mit jedem, neu zu fertigendem Bauteil den Sollwert für diesen Kurvenverlauf der Kraft neu bestimmen. Ein diskretes Eingreifen findet also vor und nicht während der Prozessdurchführung statt. Im Rahmen dieser Arbeit sind zeitdiskrete Regelkreise mit zeitkontinuierlichen Regelkreisen über eine solche Sollwertanpassung verknüpft. Das übergeordnete Ziel des Konzeptes ist die Prävention von Anomalien zur Reduktion von entlang vernetzter Prozessketten auftretender Qualitätsschwankungen durch eine bauteilbezogene Sollwertanpassung. Dazu soll eine Modifikation der Stellgrößen durch eine übergeordnete bauteilbezogene Regelung inkl. Kompensationsschema vorgenommen werden, die adaptiv für eine Unterdrückung zur Entwurfszeit unbekannter Störeinflüsse sorgt und so zu einer Steigerung der Produktqualität beiträgt. In Abbildung 4.1 ist diese Idee in vereinfachter Form illustriert.



Abbildung 4.1: Illustration der grundlegenden Konzeptidee zur adaptiven Regelung der Qualität Es wird nach einem Grey-Box-Ansatz verfahren, der auf einem durch den Experten entworfenen, initialen Prozessmodell beruht. Dieses steht zum Entwurf eines diskreten Reglers zur Verfügung. Es wird vorausgesetzt, dass dieser initiale Regler bereits entworfen wurde und der geschlossene Regelkreis aus initialem Prozessmodell und diskreter Reglung stabil ist. Sie konvergieren in ihrem Übertragungsverhalten:

$$\lim_{k \to \infty} e(k) = 0 \quad mit \ e(k) = \left(\frac{1}{R(z)_{th}G(z)_{th}} - 1\right) r(k)$$
(35)

In der industriellen Praxis wird die Stabilität häufig durch den Produktionsmitarbeiter gewährleistet, der als ,diskreter Regler' fungiert. Die stabile Einstellung des Prozessmodells durch den Regler oder Produktionsmitarbeiter in einem festen Arbeitspunkt wird in der Illustration durch die jeweils schwarzen Koordinatenpunkte auf den Geraden repräsentiert. Die Geraden entsprechen dabei einer stark vereinfachten Darstellung des Übertragungsverhaltens von initialem Prozessmodell und Regler und dienen lediglich zu Erklärungszwecken.

Ein wesentliches Problem hinsichtlich diskreter Fertigungsprozesse ist das Auftreten unbekannter Anomalien, die zu unvorhergesehenen sowie bislang nicht kontrollierbaren Schwankungen führen und eine Reduktion der Produktqualität nach sich ziehen. Das initiale, vom Prozessingenieur entworfene Prozessmodell kann aufgrund dieser zur Entwurfszeit unbekannten Effekte nicht alle Zusammenhänge adäquat kapseln und weist in der Folge Abweichungen gegenüber dem realen Prozessverhalten auf. Diese durch unbekannte, anomale Systemdynamiken entstehende Diskrepanz aus gemäß Prozessmodell  $G(z)_{th}$  erwartetem Ausgang  $y_1$  und tatsächlichem Ausgang  $y_2$  wird zur Anpassung der Stellgröße  $u_1$  genutzt. Die Stellgröße wird zu  $u_2$  angepasst, sodass auch der reale Prozess  $G(z)_{tat}$  den erwarteten Ausgang liefert. Um Anpassungen dieser Art für jedes Bauteil realisieren zu können, wird die erfasste Abweichung e(k), wie in Abbildung 4.2 Lernsignals vereinfacht dargestellt. zur Generierung des für ein neuronales Kompensationsschema genutzt, das die beschriebene Idee umsetzt. Die initial entworfene Regelung wird durch die in der Abbildung blau dargestellten Übertragungsglieder erweitert.



Abbildung 4.2: Vereinfachte Darstellung des Schemas zur Kompensation von Anomalien Die Netzstruktur kapselt das Wissen über die Anomalien und wird dazu verwendet, der Stellgröße ein Kompensationssignal  $\underline{q}(k)$  aufzuschalten, das der Differenz der Kurvenverläufe  $R(z)_{th}$  und  $R(z)_{komp}$  in Abbildung 4.1 entspricht. Durch das Aufschalten wird indirekt auch das Übertragungsverhalten des Gesamtsystems geändert. Bezogen auf die Dynamik der Stellgrößen kann der Regelung als Ganzes ein durch die Kompensation angepasstes Übertragungsverhalten  $R(z)_{komp}$  zugeschrieben werden, das vorhanden sein muss, um die Auswirkung der Anomalie auf den Ausgang des Prozesses zu unterdrücken. Das angepasste Verhalten kann in Abbildung 4.1 in der Darstellung des Reglers nachvollzogen werden. Der geschlossene Regelkreis inklusive  $G(z)_{tat}$  und  $R(z)_{komp}$  erzeugt, wie in Abbildung 4.1 angedeutet, die avisierten Sollausgänge. Der ursprünglich entworfene Regler  $R(z)_{th}$  kann dadurch unverändert bleiben. Das Regelkreisschema in Abbildung 4.2 skizziert die Idee für einen einzelnen Fertigungsprozess. Im Rahmen dieser Arbeit wird die Idee auf mehrstufige Prozessketten übertragen. Zu verdeutlichen ist an dieser Stelle, dass alle Stellgrößen bauteilbezogen, d.h. mit jedem am Prozess ankommenden Bauteil und damit in diskreter Zeit *k* neu eingestellt werden. Eine Online-Regelung von Parametern während des Prozesses ist nicht vorgesehen. Durch dieses bauteilbezogene Eingreifen sollen bisher unbekannte, die Produktqualität beeinflussende Effekte erlernt und kompensiert werden.

Die Erläuterung macht deutlich, dass das Konzept in zwei wesentliche Bestandteile unterteilt ist. Dies ist zum einen die datengetriebene Prozessmodellierung, die orientiert an der linken Grafik in Abbildung 4.1 zum Ziel hat, das tatsächliche Prozessverhalten inkl. aller relevanten Einflüsse in einem ganzheitlichen Modell  $G(z)_{tat}$  abzubilden. Und zum anderen ist dies die darauf aufbauende Kompensation von unbekannten Anomalien, die, gemäß der rechten Grafik, durch eine adaptive Anpassung der Stellgrößen zur Realisierung des Eingriffsverhaltens  $R(z)_{komp}$  erreicht werden soll. Es kann in Analogie zu dem, in den Grafiken als Geraden dargestellten Vorwissen auch im Rahmen des Konzeptes auf Vorwissen zu Regelstrecke bzw. Prozessmodell auf der einen und dem Regler auf der anderen Seite aufgebaut werden. Die blau angedeuteten polynomialen Kurven des tatsächlichen Verhaltens sollen durch die Erweiterung um datengetriebene Methoden erlernt werden. Das Ziel ist somit, das Wissen über den Prozess unter Nutzung von Daten von  $G(z)_{th}$  auf  $G(z)_{tat}$  zu erweitern und mit diesem besseren Prozessverständnis eine robustere Einstellung der Stellgrößen von  $R(z)_{th}$  nach  $R(z)_{komp}$  zu erzielen.

Um die zu diesem Zweck umgesetzte Vorgehenssystematik zu verdeutlichen, wird nachfolgend eine Übersicht über die wesentlichen Schritte der Modellbildung, ausgehend vom vorhandenen Vorwissen  $G(z)_{th}$  bis zum finalen Prozessmodell  $G(z)_{tat}$ , gegeben. Abbildung 4.3 zeigt die zeitliche Abfolge in der Modellbildung anhand des bei jedem Schritt bestehenden Wissens und der entsprechenden Regelungsgüte, die dadurch erzielt werden kann, von  $R(z)_{th}$  bis  $R(z)_{komp}$ . Numerische Modelle stellen den Ausgangspunkt der Modellbildung dar und kapseln die Gesamtheit des vorhandenen Expertenwissens. Ein Problem hinsichtlich der Modellierung diskreter Fertigungsprozesse ist der Umstand, dass die Prozessdynamiken örtlich und zeitlich verteilt sowie hochgradig nichtlinear sind. Deshalb erstellt der Prozessingenieur numerische Simulationsmodelle, die die komplexen Physiken abbilden können. Im Hinblick auf deren Erweiterung um datengetriebene Ansätze treten die folgenden Probleme auf:

- Die numerischen Simulationsmodelle beschreiben die während des Prozesses auftretenden Dynamiken. Sie bieten jedoch keine unmittelbaren Informationen für die Auswirkung von Stellgrößenanpassungen auf den Prozessausgang, die von Bauteil zu Bauteil und damit von Simulation zu Simulation auftreten.
- Die numerischen Modelle beziehen sich auf einzelne Prozesse, sodass Abhängigkeiten zwischen Prozessschritten nicht berücksichtigt werden.
- Dem Prozessingenieur zur Entwurfszeit unbekannte Anomalien sind in den numerischen Modellen nicht abgebildet.

Die numerischen Modelle des Prozessingenieurs sind sehr genau, allerdings nicht direkt für die Auslegung einer Qualitätsregelung geeignet. Daher müssen diese in E/A-Modelle überführt und dadurch für eine diskrete, bauteilbezogene Regelung nutzbar gemacht werden. Die Reduktion von örtlich und zeitlich hochaufgelösten numerischen Modellen in eine die Eingangs-Ausgangs-Beziehungen beschreibende Modellform geht mit einem Informationsverlust einher. Das entspricht dem zweiten Schritt in Abbildung 4.3. Die Diskrepanz wird im weiteren Verlauf als interne Modellunsicherheit bezeichnet, die durch das Training auf realen Produktionsdaten minimiert werden kann. Das Prozessmodell hat dann eine für den Entwurf einer prädikativen Regelung nutzbare, echtzeitfähige Form und kapselt gleichzeitig wieder das bereits vorhandene Prozesswissen. Das entspricht dem dritten Schritt in der Abbildung.



Abbildung 4.3: Modellbildungsprozess und Entwurf des Kompensationsschemas

Die Abbildung zeigt im weiteren Verlauf das durch die angedeuteten Probleme bedingte, fehlende Wissen über Abhängigkeiten und Anomalien beim Modellbildungsprozess auf. Daher wird im Rahmen des Konzeptes eine Lösung für die Integration von Abhängigkeiten entlang mehrstufiger Prozessketten vorgestellt. Dabei kommt der formalen Beschreibung von Prozessketten als Komposition verknüpfter Einzelprozesse eine besondere Bedeutung zu, da sie das Systemmodell darstellt, auf dessen Basis Regelungen sowie prozessübergreifende Optimierungsprobleme spezifiziert werden können. Um unbekannte anomale Systemdynamiken in die Modellierung zu integrieren, wird ein Detektionsmechanismus entwickelt, der die Grundlage für eine Prädiktion des tatsächlichen Prozessverhaltens darstellt. Das sich ergebende Prozessmodell  $G(z)_{tat}$  kann für die Auslegung einer Qualitätsregelung zur Kompensation von Anomalien  $R(z)_{komp}$  genutzt werden. Das entspricht dem letzten Schritt in der Abbildung.

Die sich anschließenden Abschnitte des Kapitels sind wie folgt organisiert: Zunächst wird erläutert, wie bestehendes Expertenwissen für eine nichtlineare zeitvariante Prozessmodellierung im Kontext der angestrebten Qualitätsregelung nutzbar gemacht werden kann. Dabei wird auf die Ableitung von neuronalen Metamodellen auf Basis einer numerischen Modellierung der einzelnen Prozesse eingegangen (Kap. 4.2.1). Im nächsten Abschnitt wird ein graphentheoretischer Ansatz zur Modellierung von mehrstufigen Prozessketten und dessen Verknüpfung mit den Metamodellen vorgestellt (Kap. 4.2.2). Die Nutzung des vorhandenen Expertenwissens und die sich daraus ergebende Prozessmodellierung stellt die Grundlage für eine Erweiterung um datengetriebene Methoden dar. Dazu wird im Weiteren auf die Modellierung von Abhängigkeiten entlang von Prozessketten (Kap. 4.3.1) sowie die Detektion von anomalen Prozessdynamiken eingegangen. In diesem Kontext wird ein Mechanismus zur Anomalie-Detektion eingeführt (Kap. 4.3.2), der als Grundlage für ein kooperatives Prädiktionsschema des Prozess- und Anomalie-Verhaltens dient (Kap. 4.3.3). Für das Schema werden LSTM-Netze vorgestellt, die aufgrund ihrer speziellen Architekturen Detektionsaufgaben realisieren und hochgradig nichtlineare, zeitvariante Prozessdynamiken abbilden können. Diese neuronalen Prozessmodelle werden im Rahmen eines rekursiven Optimierungsschemas genutzt, das zur Erzeugung einer Kompensation von Anomalien und damit zur Bestimmung optimaler bauteilbezogener Stellgrößen dient (Kap.4.3.4). Dadurch wird eine adaptive Qualitätsregelung mehrstufiger Prozessketten erreicht.

## 4.2 Ansatz zur Prozessmodellierung

In dieser Arbeit wird ein Grey-Box-Ansatz zur Modellierung diskreter Fertigungsprozesse verfolgt, indem das verfügbare Expertenwissen in Form von Simulationen mit neuronalen Netzen kombiniert wird. Dieses Kapitel beschäftigt sich mit dem Teil der Modellierung, der in Form von Expertenwissen realisiert werden kann. Dabei wird zunächst auf die durch den Experten initiierte Modellierung einzelner Fertigungsprozesse eingegangen, bevor diese dann in den Kontext der Prozesskette gebracht und in das übergeordnete Modell derselben integriert werden. Die Modellierung einzelner Prozesse basiert auf numerischen Simulationen, aus denen Eingangs-
Ausgangsrelationen der relevanten Prozessparameter abgeleitet werden. Die extrahierten Relationen werden in Form von nichtlinearen autoregressiven Modellen abgebildet und in einen die gesamte Prozesskette beschreibenden, azyklischen Graphen integriert.

# 4.2.1 Modellierung einzelner Fertigungsprozesse

Im Folgenden wird genauer auf die durch den Experten erstellten numerischen Simulationen im Hinblick auf Aufbau und Durchführung eingegangen. Der Aufbau der Simulationen verlangt tiefes Verständnis der zu Grunde liegenden Physiken und ist exklusiv dem Experten des Prozesses vorbehalten. Bei der Durchführung der Simulationen entstehen Daten, sodass bei der Gestaltung von Simulationsstudien hingegen auch Wissen aus dem Bereich der Datenanalyse und Modellbildung verlangt wird. Der Automatisierungsingenieur partizipiert hier mit dem Ziel, einen möglichst gut aufgelösten Datenraum an Eingangs-Ausgangsrelationen zu erreichen.

#### 4.2.1.1 Numerische Simulation diskreter Fertigungsprozesse

Diskrete Fertigungsprozesse, wie bspw. umformtechnische Prozesse, können aufgrund der komplexen zu Grunde liegenden Physiken nur numerisch beschrieben werden. Genaue analytische Modelle sind in den meisten Fällen nicht vorhanden. Die Komplexität resultiert aus der räumlichen und zeitlichen Verteilung von Materialeigenschaften und thermischen Dynamiken, die in der Regel mit Finite-Elemente-Methoden (FEM) erfasst und modelliert werden. So werden für jedes räumliche Element des Modellierungsgegenstandes zeitvariante Differentialgleichungen und physikalische Gleichgewichte definiert, die das Material- sowie Wärmeübertragungsverhalten des Elements zu jedem diskreten Zeitpunkt beschreiben. Zur Berechnung des gesamten, sich auf Basis der Dynamik aller interagierenden Elemente ergebenden Prozessverhaltens werden von den Experten prozessspezifische Simulationswerkzeuge eingesetzt. Die physikalische Komplexität diskreter Fertigungsprozesse kann somit durch den Einsatz entsprechender Simulationen numerisch approximiert werden. Der Schwerpunkt der im Rahmen der vorliegenden Arbeit durchgeführten FEM-Simulationen liegt auf einer detaillierten Modellierung der Interaktion zwischen Produkt und Werkzeug [5]. Dies sei anhand eines Beispiels aus der Umformtechnik illustriert. Hier wird bspw. auf das allgemeine Johnson-Cook-Modell zurückgegriffen, um die von der effektiven inkrementellen Dehnungsrate und den thermischen Fließparametern abhängigen Spannungstensoren im Material zu modellieren. Die Ableitung der erzeugten Fließkurve dient als Grundlage der Kraftberechnung pro Raumelement auf der Oberfläche des Produkts sowie für dessen innere Strukturen. Hinsichtlich der auf der Oberfläche geltenden Spannungsgleichgewichte müssen zusätzlich Reibungseffekte von Produkt und Werkzeug berücksichtigt werden. Um die Gesamtkomplexität der Interaktion zwischen Produkt und Werkzeug zu erfassen, sind neben diesen mechanischen Betrachtungen noch weitere notwendig. Dies ist bspw. bezüglich des Wärmeübergangs zwischen Produkt und Werkzeug sowie der Wärmeerzeugung innerhalb des Produkts aufgrund der durch das Werkzeug verursachten plastischen Verformung erforderlich.

Nach dem Aufbau der Simulation durch den Experten unter Berücksichtigung der beschriebenen Charakteristiken wird diese nun im Rahmen einer experimentellen Studie zur Variation der Eingangsparameter genutzt (Design of Experiments, kurz: DoE), die die Simulation in den Kontext einer bauteilbezogenen Qualitätsregelung setzt. Dies ist erforderlich, da numerische Simulationsmodelle zwar eine detaillierte mathematische Beschreibung der Prozessphysik liefern, diese jedoch nicht in Echtzeit berechnet werden können. D.h. die Prozessausgänge, die durch bestimmte Eingangs-Anregungen erzeugt werden, können nicht Ad-hoc kalkuliert werden, da die Durchführung jeder Simulation eine gewisse Zeit in Anspruch nimmt. Um nun ein echtzeitfähiges Modell zu erhalten, werden die rechenintensiven Simulationen für eine Vielzahl an verschiedenen Eingangsparameter ausgeführt. Eine solche DoE zielt darauf ab, die für das Simulationsziel relevanten Anregungen des Systems und die dabei entstehenden Systemantworten systematisch zu eruieren. Jeder Eingangsparameter des Prozesses kann dabei in einem fest definierten Intervall variiert werden. Die Grenzwerte dieser Intervalle entsprechen entweder den physikalischen Grenzen des Aktors oder den zur Wahrung der Betriebssicherheit vorgeschriebenen Grenzen seitens des Experten. Die gewählten Werte der Eingangsparameter und die daraus entstandenen Prozessausgänge bilden die Datengrundlage für die Erzeugung eines neuen, echtzeitfähigen Prozessmodells. Da im Rahmen dieser Arbeit eine Regelung der Qualität von Bauteil zu Bauteil durchgeführt werden soll, kommt es dabei zu einer iterativen Sollwertanpassung der Prozessgrößen. Dieser Umstand hat unmittelbare Auswirkungen auf die Gestaltung der DoE und die Durchführung der Simulation. Die diskrete bauteilbezogene Regelung bedingt, dass übergeordnete Sollwerte den Eingangsparametern der Simulation entsprechen, die vor jeder Durchführung des diskreten Fertigungsprozesses zur Bearbeitung des nächsten Bauteils adaptiv und damit von Simulation zu Simulation neu eingestellt werden. Während der Durchführung der Fertigungsprozesse findet hingegen keine Regelung statt, während sodass der Simulationsdurchläufe keine Einstellungen geändert werden. Multiple aufeinanderfolgende Simulationen müssen durchgeführt werden, um die Auswirkungen von Sollwertänderungen der Prozessgrößen auf die Produktqualität abbilden zu können. Im Rahmen der vorliegenden Arbeit beinhalten diese Simulationen nicht nur einen bestimmten Fertigungsprozess, sondern sie sind auf die Abbildung einer ganzen Prozesskette ausgelegt. Berechnungen sequentiell stattfindender Prozesse werden in einer Simulation kombiniert, um das Verhalten über die gesamte Prozesskette zu modellieren. Die DoE erzeugt dabei charakteristische Kennfelder für jeden Fertigungsprozess, die Informationen über die Beziehungen zwischen verschiedenen Eingangs- und Ausgangsvariablen erfassen. Ein Beispiel für diese Vorgehensweise ist in Abbildung 4.4 für die Transferzeit zwischen induktiver Erwärmung und Umformprozess, die Bauteilgeometrie und die beim Umformen aufgebrachte Kraft dargestellt [25]. Die dargestellten Simulationen stammen dabei vom Institut für Umformtechnik der Universität Stuttgart.



Abbildung 4.4: Beispiel für die Durchführung einer DoE-Studie zur Untersuchung von Eingangs-Ausgangsrelationen diskreter Fertigungsprozesse [24]

Um die funktionalen Zusammenhänge von Ein- und Ausgangsvariablen genau erfassen zu können, werden zur Verfeinerung der DoE bestimmte Teilbereiche der Eingangsintervalle ausgewählt und genauer untersucht. Dadurch soll die Abbildung von Ein- auf Ausgangsvariablen auch für kritische Bereiche des Datenraumes hinreichend genau werden. Dies gilt insbesondere auch für Simulationen der kritischsten Teilbereiche des Produkts, wie bspw. Kanten oder Hohlräume in Abbildung 4.4, hinsichtlich derer der Einfluss von Variationen der Eingänge auf die Produktqualität besser bewertet werden kann. Die im Rahmen der vorliegenden Arbeit verwendeten Simulationen wurden von Experten des Instituts für Umformtechnik der Universität Stuttgart erstellt. Ein Beispiel für Simulationsmodelle der Experten ist in Abbildung 4.4 gegeben.

## 4.2.1.2 Ableitung von diskreten E/A-Modellen

Die Simulationen kapseln das gesamte Expertenwissen des Prozessingenieurs, können jedoch aufgrund der hohen Rechenzeiten nicht direkt als Vorhersagemodelle in einer Regelung eingesetzt werden. Das Rechnen und Prädizieren mit numerischen Modellen ist sehr rechenintensiv und dadurch mit hohen Zeitkosten verbunden. Linearisierungen um definierte Arbeitspunkte können die Problematik in den meisten Fällen nicht ausreichend lösen. Aus diesem Grund wurde in [141] ein Ansatz basierend auf RNN zur Prozessmodellierung entwickelt, der die Umschaltung zwischen dem numerischen Modell und einem stark reduzierten Ersatzmodell vorsieht, das lediglich die für das Qualitätsergebnis des Prozesses wesentlichen Stellgrößen berücksichtigt. Das Ersatzmodell kommt dann zum Einsatz, wenn das numerische Modell aufgrund bestehender Zeitrestriktionen keine Vorhersage liefern kann. Dadurch können prädikative Regelungsaufgaben performanter und ohne größere Einbußen im Hinblick auf die Regelgüte realisiert werden. Nichtsdestotrotz ist das reduzierte Ersatzmodell weder präzise genug noch um reale Prozessdaten erweiterbar. Daher wird im Rahmen dieser Arbeit der Ansatz verfolgt, die bei der Durchführung der Simulationen entstehenden Daten nicht nur in Form von Kennfeldern zu visualisieren, sondern aus dem Simulations-Werkzeug zu extrahieren. FEM-Tools bieten in den meisten Fällen die Möglichkeit, Simulationsdaten zu cachen und über standardisierte Schnittstellen für die weitere Verarbeitung in externen Anwendungen nutzbar zu machen. Numerische Simulationen erzeugen durch das Lösen von Differentialgleichungen unter der Randbedingung definierter physikalischer Gleichgewichte bei jeder Simulationsdurchführung zu jedem Zeitpunkt für jedes Raumelement Prozessdaten. Um auf der Grundlage dieser Simulationsdaten ein zeitdiskretes Prozessmodell zu erzeugen, wird die Dimension der ausgewählten und zwischengespeicherten Daten reduziert und diskretisiert. Die Diskretisierung wird bauteilbezogen durchgeführt. Auf Basis der reduzierten Daten kann ein die Eingangs-Ausgangsrelationen beschreibendes Metamodell (E/A-Modell), wie z.B. ein autoregressives Modell, erstellt werden, das die Auswirkung von Stellgrößen auf die Qualität jedes produzierten Bauteils beschreibt.

In diesem Kontext wird von einem Metamodell gesprochen, da dieses auf der Basis eines bereits vorhandenen numerischen Modells erzeugt wird. Ein Problem bei der Extraktion von E/Abasierten Metamodellen auf der Grundlage numerischer Modelle ist der entstehende Informationsverlust. Aufgrund der hohen Rechenzeiten von numerischen Simulationen kann nur eine begrenzte Anzahl von Input-Output-Kombinationen innerhalb der für die DoE definierten Intervalle berechnet werden. Die zwischengespeicherten Daten und in der Folge auch das erzeugte Metamodell erfassen somit lediglich eine reduzierte Darstellung des numerischen Modells. Die E/A-Modelle sind daher mit Unsicherheiten behaftet. Darüber hinaus bestehen auch bei dem numerischen Modell selbst infinitesimale Modellunsicherheiten, die auf dessen räumliche und zeitliche Diskretisierung, wie bspw. der Größe der finiten Elemente, zurückgehen. Ein weiteres Problem sind darüber hinaus dem Experten unbekannte Effekte, die das Prozessverhalten zusätzlich beeinflussen. Diese Anomalien zeichnen sich durch hohe Nichtlinearität, geringe Messbarkeit, variierende Arbeitspunkte und eine temporale Überlagerung von Kurz- und Langzeitzeitabhängigkeiten aus. Die vorliegende Arbeit konzentriert sich daher auf die oben genannten Modellunsicherheiten sowie auf die Erkennung und Vorhersage der unbekannten Anomalien, die vom Experten nicht modelliert oder simuliert wurden und daher nicht in den E/A-Modellen enthalten sind. Dabei wird ein datengetriebener Ansatz angestrebt, der auf dem vorhandenen Expertenwissen aufbaut und dieses um die Modellunsicherheiten sowie die apriori unbekannten Zusammenhänge erweitert. Im Rahmen dieses Grey-Box-Ansatzes werden die das Expertenwissen kapselnden E/A-Metmodelle daher in Form von neuronalen Modellen realisiert, sodass eine Erweiterbarkeit durch die Integration realer Prozessdaten gegeben ist.

# 4.2.2 Modellierung mehrstufiger Prozessketten

Die im vorherigen Abschnitt behandelten numerischen Modelle und die korrespondierenden E/A-Metamodelle beziehen sich auf einzelne Fertigungsprozesse, die unter Berücksichtigung bestehender Verknüpfungen entlang der Prozesskette in einer gemeinsamen Simulation berechnet und extrahiert werden. Um die prozesspezifischen E/A-Metamodelle im Rahmen eines Gesamtmodells der mehrstufigen Prozesskette nutzbar zu machen, wird ein graphentheoretischer Ansatz verfolgt, auf den im Folgenden genauer eingegangen wird.

## 4.2.2.1 Graphentheoretischer Modellierungsansatz

Eine Prozesskette wird als sequentieller Verbund von diskreten Fertigungsprozessen aufgefasst. Sie kann als gerichteter azyklischer Graph  $(\mathcal{P}, \mathcal{T})$  beschrieben werden. Dabei stellt  $\mathcal{P}$  die Menge aller Knoten und  $\mathcal{T}$  die Menge aller Kanten dar. Die Knoten repräsentieren die Prozesse und die Kanten den Transport zwischen den Prozessen. Ein Prozess  $P_{ij} \in \mathcal{P}$  und ein entsprechender Transportvorgang  $T_{ij} \in \mathcal{T}$  befinden sich dabei an einer definierten Stelle *i* innerhalb der Prozesskette. Alle Prozesse  $P_{ij}$  mit demselben Indexwert *i* entsprechen einem eindeutigen Prozesstypen wie bspw. dem Erwärmen. Der Index *j* repräsentiert die jeweilige Fertigungseinheit für diesen Prozesstypen, z.B. eine bestimmte Erwärmungsanlage, bzw. die jeweilige Transporteinheit für den zugehörigen Transportvorgang. Von jedem Prozesstypen können also eine oder mehrere funktionsgleiche Fertigungseinheiten vorhanden sein. Die zu modellierende Prozesskette muss in Anlehnung an [29] folgende Anforderungen erfüllen:

- Pro diskretem Fertigungsprozess ist nur eine Prozessdurchführung pro Bauteil möglich. In der diskreten Fertigung kommen des Öfteren mehrstufige Fertigungsprozesse wie bspw. zwei- oder dreistufige Umformprozesse vor. Jede Stufe wird dabei als separater Prozess modelliert, sodass der azyklische Charakter des Prozesskettenmodells gewährleistet ist und Rückführungen nur zur selben Fertigungseinheit aber nicht zum selben Prozess möglich sind.
- Es muss eine einheitliche Zykluszeit vorhanden sein, sodass es zu keiner Anhäufung von Bauteilen in Form von Zwischenlagerungen vor bestimmten Fertigungseinheiten kommt. Entweder müssen höhere oder niedrigere Prozesszeiten durch den Einsatz von mehr bzw. weniger Fertigungseinheiten desselben Typs kompensiert werden, oder aber die maximale Prozesszeit wird als Zykluszeit der Prozesskette realisiert.

Der verfolgte Modellierungsansatz sieht vor, dass eine bauteilbezogene Diskretisierung der Zeitachse gemäß einer Zykluszeit erfolgt, die der Summe aus Prozess- und Transportzeit entspricht. Mit jedem diskreten Zeitschritt trifft daher an jedem Prozess ein neues Bauteil ein. Weiterhin ist vorgesehen, dass die einzelnen Fertigungsprozesse in ihrer Dynamik über die Eigenschaften der durch die Prozesskette laufenden Bauteile verknüpft sind [29]. Die Änderungen der Bauteileigenschaften werden durch die Prozesse bestimmt, sodass das Prozessergebnis des jeweils vorangegangenen Prozesses die gesamte Prozesshistorie dieses Bauteils kapselt. Jeder Prozess der Prozesskette wird in einem separaten Prozessmodell abgebildet. Die Bauteileigenschaften können als externer Störeinfluss in die Prozessmodellierung integriert werden [131], sodass darüber eine Verknüpfung aller Prozessmodelle zu einem übergeordneten Prozesskettenmodell entsteht.



Abbildung 4.5: Modellierung von Prozessketten als gerichtete azyklische Graphen in Anlehnung an [29]

In Abbildung 4.5 ist eine Prozesskette mit vier aufeinanderfolgenden Prozessen dargestellt. Prozess drei besitzt die doppelte Prozesszeit im Vergleich zu den anderen Prozessen. Dies wird in der illustrierten Darstellung durch die Verwendung von zwei funktionsgleichen Fertigungseinheiten ausgeglichen, um die Anforderung einer einheitlichen Zykluszeit zu erfüllen. Die Bauteile treten in die Prozesskette ein und werden bei der ersten Fertigungseinheit bearbeitet. Nach der Bearbeitung werden sie durch die sich anschließende Transporteinheit zur darauffolgenden Fertigungseinheit transportiert. Dies kann bspw. eine Materialeingangskontrolle und eine sich daran anschließende Erwärmung des Rohmaterials sein. Das Modell impliziert weiterhin, dass zu jedem Zeitpunkt genau ein Tupel  $(P_{ii}, T_{ii})$  für jedes feste *i*, bestehend aus Prozesstyp i und sich anschließendem Transportvorgang zu i, mit einem Bauteil belegt ist. Es sind demnach zu jedem Zeitpunkt multiple Bauteile zur Fertigung in der Prozesskette, deren Anzahl der Summe aller vorhandenen Fertigungseinheiten entspricht. Bei jedem Prozesstypen befindet sich zu jedem Zeitpunkt daher mindestens ein Bauteil. Projiziert auf Abbildung 4.5 würde dies zwei Bauteile beim dritten Prozesstypen und jeweils ein Bauteil bei allen anderen Prozessen bedeuten. Für den Transport würde das ein Alternieren zwischen Pfad  $[(P_{21}, T_{21}); (P_{31}, T_{31})]$ sowie Pfad [(P<sub>21</sub>, T<sub>22</sub>); (P<sub>32</sub>, T<sub>32</sub>)] mit jedem Zyklus bedeuten.

# 4.2.2.2 Integration der diskreten E/A-Modelle

In diesem Kapitel wird die Struktur der diskreten E/A-basierten Prozessmodelle erläutert, auf die bereits in Abschnitt 4.2.1.2 im Kontext der Extraktion aus Simulationsdaten eingegangen wurde. Diese aus den numerischen Simulationen extrahierten Metamodelle werden in das Gesamtmodell des gerichteten azyklischen Graphen an die Stellen der Knoten integriert. Sie bestehen aus mehrdimensionalen Input-Output-Relationen zur Beschreibung des Prozesses und kapseln die bauteilbezogene Auswirkung von Stellgrößen auf die Prozessausgänge. Es lassen sich verschiedene Formen von E/A-Modellen unterscheiden. Grundsätzlich können diese auf Methoden des gleitenden Mittelwertes (MA-Modelle) oder einer Autoregression (AR-Modelle) aufbauen. Eine Kombination aus beiden Techniken mit entsprechender Gewichtung mündet in einem ARMA-Modell und ermöglicht die Abbildung nicht-stationären Verhaltens. ARMA-Modelle werden zu ARIMA-Modellen, wenn diese die Änderungen des ursprünglichen

zeitvarianten Prozesses modellieren. Sie lassen sich durch Integration wieder auf die ursprüngliche Form zurückführen. Darüber hinaus können der Einfluss externer Variablen auf das Zeitverhalten (ARMAX-Modelle) sowie Nichtlinearitäten (NARMA-Modelle) modelliert werden [158]. Im Rahmen der vorliegenden Arbeit werden nichtlineare autoregressive Modelle unter Berücksichtigung exogener Variablen (NARX-Modell) zur Modellierung der einzelnen Prozesse verwendet. Die E/A-Modelltypen wurden bereits im Kontext zahlreicher kontrolltheoretischer Anwendungen zur Abbildung nichtlinearer zeitvarianter Prozesse eingesetzt [14]. Für das verwendete Modell gilt:

$$\underline{\hat{y}}_{n}(k) = f_{n}\left(\underline{y}_{n}(k-1), \dots, \underline{y}_{n}(k-n_{y}), \underline{u}_{n}(k), \dots, \underline{u}_{n}(k-n_{u}), k\right)$$
(36)

wobei  $f_n(\cdot)$  das nichtlineare zeitvariante Prozessmodell des Prozesses n zum Zeitpunkt k darstellt. Der Vektor  $\underline{\hat{y}}_n(k)$  entspricht der Vorhersage des Prozessausgangs, der Vektor  $\underline{u}_n(k)$  den aktuellen Stellgrößen und  $n_y$  sowie  $n_u$  den Horizonten vergangener Prozessausgänge bzw. vergangener Stellgrößen. Diese ursprüngliche Form des NARX-Modells kann jedoch nicht in den azyklischen Graphen integriert werden. Das Modell muss aufgrund der Tatsache, dass die Prozesse über die Bauteileigenschaften miteinander verbunden sind, um eine Modellierung der Abhängigkeiten entlang der Prozesskette erweitert werden. Die Verknüpfung zwischen den Prozessen wird hergestellt, indem der Ausgang des vorangegangenen Prozesses n - 1 als zusätzlicher Eingang des nachfolgenden Prozesses n modelliert wird. Durch die Prozessdauer und den Transport des Bauteils zwischen den örtlich getrennten Fertigungseinheiten wird mit einer Verzögerung von einem diskreten Zeitschritt gerechnet. Dies entspricht der Zykluszeit zwischen zwei Prozessdurchführungen. Die folgende Bedingung hat daher Gültigkeit:

$$\underline{w}_{n}(k) = y_{n-1}(k-1) \tag{37}$$

wobei der Vektor  $\underline{w}_n(k)$  die als zusätzlichen Eingang des nachfolgenden Prozesses modellierten Bauteileigenschaften kapselt. Der Vektor kann als eine, im Zuge jedes neu zu fertigenden Bauteils wirkende Störung interpretiert werden, deren dynamisches Verhalten längerfristig entlang der Prozesskette auftretende Prozess- und Chargenschwankungen widerspiegelt. Die Vorhersage eines Prozessausgangs ist daher nicht nur von der eigenen Prozesshistorie abhängig, sondern durch die entstehende Rekursion auch von den, um entsprechende Verzögerungszeiten versetzten Ausgängen aller vorangegangenen Prozesse. Eine entsprechende Erweiterung des autoregressiven Modells lautet:

$$\underline{\hat{y}}_{n}(k) = f_{n}\left(\underline{y}_{n}(k-1), \dots, \underline{y}_{n}(k-n_{y}), \underline{u}_{n}(k), \dots, \underline{u}_{n}(k-n_{u}), \underline{w}_{n}(k), k\right)$$
(38)

Das erweiterte Prozessmodell berücksichtigt somit die Auswirkungen und Schwankungen aller vorangegangenen Prozesse. Die Interdependenz zweier aufeinanderfolgender Prozessschritte und die entsprechenden Modellvariablen sind beispielhaft in Abbildung 4.6 dargestellt.

$$\underbrace{\frac{\underline{u}_{n-1}(k)}{\underline{u}_{n-1}(k-n_u)}}_{\underbrace{\underline{y}_{n-1}(k-1)}} f_{n-1}(\cdot) \underbrace{\underbrace{\underline{y}_{n-1}(k)}_{\underline{y}_{n-1}(k-n_y)}}_{\underbrace{\underline{y}_{n-1}(k-n_y)}} f_{n}(\cdot) \underbrace{\underbrace{\underline{y}_{n}(k+1)}_{\underline{y}_{n-1}(k-n_y)}}_{\underbrace{\underline{y}_{n}(k+1-n_y)}} f_{n}(\cdot)$$

Abbildung 4.6: Verknüpfung zweier Prozesse innerhalb des Graphen-basierten Prozesskettenmodells

Die Abbildung veranschaulicht, dass nicht die Vorhersage, sondern gemäß Gl. 37 der tatsächliche Ausgang als zusätzlicher Eingang verwendet wird. Der tatsächliche Ausgang des vorherigen Prozessschritts zur Zeit k ist durch den Zeitversatz zum Zeitpunkt k + 1 bereits messbar und liefert einen genaueren Wert für  $\underline{w}_n(k + 1)$  als die Vorhersage. Neben vorwärts gerichteten Abhängigkeiten werden in [8] auch rückwärts gerichtete Abhängigkeiten berücksichtigt. Im Rahmen des in dieser Arbeit verfolgten Ansatzes sind diese Abhängigkeiten bei der Formulierung des Optimierungsproblems von Bedeutung, das in Kapitel 4.3.4 behandelt wird.

# 4.3 Ansatz zur Qualitätssicherung

Nachdem im Rahmen des verfolgten Grey-Box-Ansatzes die Prozessmodellierung erörtert wurde, beschäftigt sich der nachfolgende Teil mit der Erfassung und Integration unbekannter anomaler Systemdynamiken in die Modelle. Trotz des vorhandenen Expertenwissens des Prozessingenieurs sowie des weiteren Faktors "Maschinenbediener", der durch die Anwendung seines über Jahre gesammelten Erfahrungswissens zu einer zusätzlichen Prozessstabilität beiträgt, kommt es in der diskreten Fertigung weiterhin zu Qualitätsschwankungen und kostenintensivem Ausschuss [9]. Dieser Umstand verdeutlicht, dass datengetriebene Ansätze zur Erfassung bisher unbekannter Zusammenhänge das Potenzial haben, die bestehenden Probleme zu lösen und zu einer verbesserten Einhaltung der Produktqualität beizutragen. Der im Rahmen dieser Arbeit vorgestellte Ansatz setzt sich aus einem Detektionsmechanismus unbekannter anomaler Systemdynamiken, einer Modellierung und kooperativen Prädiktion derselben sowie einem darauf aufbauenden rekursiven Kompensationsschema zur Optimierung der Produktqualität zusammen. Bevor die drei wesentlichen Teilaspekte des Konzeptes genauer erläutert werden, kommt es zunächst zu einer Erweiterung des in Abschnitt 4.2.2.2 eingeführten Prozessmodell um unbekannte anomale Prozessdynamiken.

# 4.3.1 Erweiterung des Prozessmodells

Um die Produktqualität entgegen entlang von Prozessketten auftretender Interdependenzen und unbekannter Anomalien kontrollieren zu können, muss das im vorherigen Abschnitt vorgestellte Prozessmodell, das bereits die Modellierung von Abhängigkeiten zwischen Prozessen berücksichtigt, um eine Modellierung dieser Anomalien erweitert werden. Anomales Prozessverhalten ist der wesentliche Treiber für Varianzen in der Produktgualität, die bis zur Entstehung von Ausschuss führen können. Es kann bei Anomalien zwischen internen Modellunisicherheiten und externen Störeffekten unterschieden werden. Als intern werden dabei bekannte Prozessstrukturen bezeichnet, deren Transformation hereits in ein fiir regelungstechnische Zwecke nutzbares Prozessmodell mit unvermeidbaren Informationsverlusten verknüpft ist. Als extern werden unbekannte Prozessstrukturen in Form von Störungen bezeichnet, die hinsichtlich Struktur und Parametrierung vollständig nicht bekannt sind. Interne Modellunsicherheiten entstehen im vorliegenden Fall aufgrund des durch die Reduktion numerischer Simulationsmodelle auftretenden Informationsverlustes. Die E/A-basierten Metamodelle kapseln die numerisch beschriebenen Zusammenhänge nicht in ihrer vollständigen Komplexität, da die extrahierten Simulationsdaten approximiert werden. Aber auch die vom Experten erstellten numerischen Modelle enthalten selbst infinitesimale Modellunsicherheiten, die durch die zeitliche und örtliche Diskretisierung bedingt sind. Die Unsicherheiten hängen zum Beispiel von der gewählten Größe der finiten Elemente ab. Die Numerik selbst erzwingt an dieser Stelle einen Kompromiss zwischen höherer Modellierungsgenauigkeit, bspw. durch eine feinere Strukturierung mit finiten Elementen, sowie einer damit einhergehenden Erhöhung der Rechenzeit. Eine unendlich feine und damit exakte Replikation der Realität ist aufgrund des infiniten Rechenaufwands physikalisch nicht umsetzbar [159]. Der in dieser Arbeit verfolgte Grey-Box-Ansatz zielt darauf ab, diese internen Unsicherheiten durch Einbeziehung realer Daten zu minimieren. Zur Modellierung interner Modellunsicherheiten  $\Delta \hat{y}_n(k)$  wird ein autoregressiver Ansatz verfolgt:

$$\Delta \underline{\hat{y}}_n(k) = g_n \left( \Delta \underline{y}_n(k-1), \Delta \underline{y}_n(k-2), \dots, \Delta \underline{y}_n(k-n_{\Delta y}), k \right)$$
(39)

Die Prädiktionsfunktion  $g_n(\cdot)$  wird dabei als zeitvariante Funktion auf Basis historischer Fehlerterme realisiert. Die Vorhersage der zu erwartenden internen Modellunsicherheit basiert auf vergangenen, über den Horizont  $n_{\Delta y}$  gemessenen Abweichungen von vorhergesagtem und realem Prozessausgang.

$$\Delta \underline{y}_n(k-m) = \underline{y}_n(k-m) - \hat{\underline{y}}_n(k-m) \quad \forall \ 1 \le m \le n_{\Delta y}$$

$$\tag{40}$$

Durch eine stetige Integration der geschätzten internen Modellunsicherheit in die Prädiktion des Prozessausgangs wird ein integrierendes Prädiktionsverhalten erzeugt, das eine iterative Reduktion des Vorhersagefehlers zur Folge hat, da der Fehlerterm immer genauer approximiert werden kann. Die Schätzung des Prozessausgangs umfasst dann das normale Prozessverhalten mit verbesserter Schätzung der internen Modellunsicherheit:

$$\underline{\hat{y}}_{n}(k) = f_{n}\left(\underline{y}_{n}(k-1), \dots, \underline{y}_{n}(k-n_{y}), \underline{u}_{n}(k), \dots, \underline{u}_{n}(k-n_{u}), \underline{w}_{n}(k), k\right) + \Delta \underline{\hat{y}}_{n}(k)$$
(41)

Die Funktion  $f_n(\cdot)$  entspricht dem auf Basis des Expertenwissens generierten E/A-Metamodell, das bereits die Abhängigkeiten entlang der Prozesskette berücksichtigt. Die Korrektur der Vorhersage des Prozessausgangs um die Schätzung der internen Modellunsicherheit führt zu einer genaueren Abbildung der Systemdynamiken und damit zu einer genaueren Vorhersage  $\hat{y}_n(k)$ .



Abbildung 4.7: Informationsfluss des um die Modellierung interner Modellunsicherheiten erweiterten Prozessmodells

Abbildung 4.7 visualisiert die Prozessmodellierung inklusive der Informationsflüsse anhand eines Blockdiagramms. Es zeigt die Modellierung der realen Prozessdynamiken durch  $f_n(\cdot)$  und  $g_n(\cdot)$ . Die sich durch das integrierende Verhalten des Gesamtsystems iterativ verbessernde Schätzung des internen Modellfehlers  $\Delta \hat{y}_n(k)$  macht den vorhergesagten Prozessausgang  $\hat{y}_n(k)$  genauer, das Modell  $f_n(\cdot)$  hingegen nicht. Um jedoch die Modellgenauigkeit zu erhöhen und eine Modellierung der internen Modellunsicherheit obsolet zu machen, wird das E/A-Modell  $\underline{f}_n(\cdot)$  parallel auf realen Daten weitertrainiert. Dies ist in der Abbildung durch die Rückführung des Prädiktionsfehlers  $\underline{e}_{y,n}(k)$  dargestellt, der als Lernsignal fungiert. Im Rahmen dieser Arbeit wird angenommen, dass reale historische Datensätze nach normalen und anomalen Daten aufgeteilt werden können. Das weitere Training des Prozessmodells basiert daher auf historischen Realdaten, die das normale Prozessverhalten darstellen. Dadurch wird die Modellgenauigkeit stetig verbessert und der interne Fehler reduziert, sodass die Notwendigkeit einer Modellierung desselben mit der Trainingszeit abnimmt:

$$\underline{\hat{y}}_{n}(k) \approx f_{n}\left(\underline{y}_{n}(k-1), \dots, \underline{y}_{n}(k-n_{y}), \underline{u}_{n}(k), \dots, \underline{u}_{n}(k-n_{u}), \underline{w}_{n}(k), k\right) \Delta \underline{\hat{y}}_{n}(k) \approx 0 \quad (42)$$

Daher trägt die Modellierung interner Modellunsicherheiten vor allem zu Beginn des Betriebs zu einer erhöhten Prädiktionsgüte bei, bis das Prozessmodell ausreichend genau an die realen Daten angepasst werden konnte [28]. Um zusätzlich anomales Verhalten in Form von externen Störeffekten bei dem Modellierungsansatz zu berücksichtigen, muss eine letzte Modellerweiterung erfolgen. Externe Störeffekte sind bislang unbekannte physikalische Erscheinungen, die das Prozessverhalten zusätzlich beeinflussen. Es wird davon ausgegangen, dass alle bekannten Physiken bereits in den vom Experten erstellten Modellen enthalten sind. Daher liegt der Fokus der vorliegenden Arbeit auf allen vom Normalverhalten abweichenden Effekten, die nicht vom Experten modelliert oder simuliert wurden und daher als unbekannt betrachtet werden. Zur Klassifizierung von Störungen können verschiedene Kriterien herangezogen werden, wie z.B. Angriffspunkt, Zeitverhalten, Linearität oder Messbarkeit [160]. Im Hinblick auf den Angriffspunkt kann sowohl eine direkte Beeinflussung der Prozessdynamiken als auch eine indirekte Beeinflussung derselben über an Aktoren oder Sensoren wirkende Störungen erfolgen. So kann der Angriffspunkt auch am Prozesseingang oder -ausgang liegen. In dieser Arbeit werden externe Störeffekte untersucht, die auf die Aktoren wirken und deren Dynamik verändern. Für die untersuchten Störungen werden hinsichtlich der oben genannten Kriterien allgemein die folgenden Charakteristiken angenommen:

- · Angriffspunkt an den Aktoren
- Zeitvariantes Verhalten
- · Hochgradig nichtlineare Dynamiken
- · Geringe oder keine Messbarkeit

Aufgrund der Tatsache, dass davon ausgegangen wird, dass zu den Störeffekten keine separaten Prozessparameter oder messbare Signale vorhanden sind, die diese direkt charakterisieren, müssen diese indirekt über ein Beobachterschema abgeleitet werden. Der Umstand, dass externe Störeffekte bislang nicht im Prozessmodell berücksichtigt wurden, führt zu Fehlern bei der Prädiktion des Prozessausgangs. Im Falle des Vorhandenseins von Störungen gilt Gleichung 42 daher nicht mehr. Da auf die Aktoren wirkende Störungen betrachtet werden, wird  $\Delta \underline{u}_n(k)$  als Bezeichnung eingeführt. Um diese einzubeziehen, wird das Prozessmodell um einen Korrekturterm für die Stellgrößen erweitert, der zu einer Schätzung der tatsächlich am Prozess wirkenden Stellgrößen führt:

$$\underline{\hat{u}}_{n}^{p}(k) = \underline{u}_{n}(k) + \Delta \underline{\hat{u}}_{n}(k) \tag{43}$$

wobei  $\Delta \underline{\hat{u}}_n(k)$  einer Schätzung der realen Störung  $\Delta \underline{u}_n(k)$  entspricht. Der Vektor  $\underline{u}_n(k)$  markiert die vom initialen Regler ausgegebenen Stellgrößen und  $\underline{\hat{u}}_n^p(k)$  entspricht einer Schätzung des tatsächlich am Prozess wirkenden Stellvektors  $\underline{u}_n^p(k)$ . Die Schätzung der realen Störung selbst wird durch ein separates Zeitreihenmodell abgebildet. Dazu wird ein zeitvariantes autoregressives Modell  $h_n(\cdot)$  verwendet, das als Eingänge die vergangenen, über den Horizont  $n_u$  beobachteten bzw. detektierten Störvektoren  $\Delta \underline{\tilde{u}}_n(k-n)$  erhält:

$$\Delta \underline{\hat{u}}_n(k) = h_n \left( \Delta \underline{\tilde{u}}_n(k-1), \Delta \underline{\tilde{u}}_n(k-2), \dots, \Delta \underline{\tilde{u}}_n(k-n_u), k \right)$$
(44)

Um einen Abgleich von Vorhersage und Realwert der Störungen vornehmen zu können, ist aufgrund der eingeschränkten Messbarkeit ein Beobachtungs- bzw. Detektionsschema zur Ermittlung der ,tatsächlich' vorhandenen Störungen notwendig. Dieser Anomalie-Detektor konstruiert die Eingänge des AR-Modells und erlaubt ein entsprechendes Training desselben. Auf Aufbau und Funktionsweise des Anomalie-Detektors wird in Kapitel 4.3.2 näher eingegangen. Entsprechend der Gleichung 45 können durch beobachtete Störungen am Prozess beobachtete Stellgrößen abgeleitet werden:

$$\underline{\tilde{u}}_{n}^{p}(k-m) = \underline{u}_{n}(k-m) + \Delta \underline{\tilde{u}}_{n}(k-m) \quad \forall \ 1 \le m \le n_{u}$$

$$\tag{45}$$

Unter Berücksichtigung der Schätzung und Beobachtung der Zusammenhänge 43 bis 45 kann das erweiterte Prozessmodell wie folgt formuliert werden:

$$\underline{\hat{y}_n}(k) = f_n\left(\underline{y_n}(k-1), \dots, \underline{y_n}(k-n_y), \underline{\hat{u}_n}^p(k), \underline{\tilde{u}_n}^p(k-1), \dots, \underline{\tilde{u}_n}^p(k-n_u), \underline{w_n}(k), k\right) 
+ \Delta\hat{y}_n(k)$$
(46)

Die Vorhersage  $\hat{y}_n(k)$  basiert auf Beobachtungen vergangener Anomalien  $\Delta \underline{\tilde{u}}_n(k - \underline{\tilde{u}}$ 1), ...,  $\Delta \underline{\tilde{u}}_n(k - n_u)$ , aus denen sich die am Prozess wirkenden Stellgrößen  $\underline{\tilde{u}}_n^p(k - 1)$ , ...,  $\underline{\tilde{u}}_n^p(k - 1)$  $n_{\mu}$ ) ableiten lassen. Darüber hinaus hängt die Vorhersage des Prozessausgangs von der Schätzung der zukünftigen Anomalie  $\Delta \hat{u}_n(k)$  ab, die die Schätzung der zukünftig am Prozess wirkenden Stellgrößen  $\hat{u}_n^p(k)$  bestimmt. In Abbildung 4.8 ist das um die externen Störeffekte erweiterte Prozessmodell dargestellt. Die in Abbildung 4,7 berücksichtigten internen Modellunsicherheiten sowie die Abhängigkeiten entlang der Prozesskette sind in dieser Darstellung der Übersicht halber nicht visualisiert. Die Abbildung veranschaulicht den durch die Gleichungen 43 bis 46 beschriebenen Informationsfluss. Zusätzlich sind für die Modelle  $f_n(\cdot)$  und  $h_n(\cdot)$  die Lernprozesse visualisiert. Das Lernsignal  $e_{\Delta u,n}(k)$  dient zum Training des autoregressiven Modells  $h_n(\cdot)$  und wird durch einen Abgleich von vorhergesagter und im Nachgang tatsächlich beobachteter Störung erzeugt. Das Lernsignal  $e_{y,n}(k)$  wird zum Training des E/A-basierten Metamodells des normalen Prozessverhaltens  $f_n(\cdot)$  verwendet und auf Basis der Abweichung von vorhergesagtem und im Nachgang tatsächlich gemessenem Prozessausgang erzeugt. Das Lernsignal entspricht dabei der Summe aus der durch Gleichung 39 beschriebenen internen Modellunsicherheit sowie der durch Gleichung 40 erfassten, für denselben diskreten Zeitpunkt tatsächlich gemessenen Abweichung von vorhergesagtem und realem Prozessausgang. Neben den Teilmodellen ist der Anomalie-Detektor als weiterer Block in das Schema integriert. Dieser wird zur Generierung der Beobachtungen und damit der Eingänge für beide Teilmodelle benötigt [28] [159]. Das Gesamtmodell ist eine Komposition aus den Teilmodellen  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$ , die das normale Prozessverhalten, die externen Störeffekte und die interne Modellunsicherheit abbilden.



Abbildung 4.8: Informationsfluss des um die Modellierung unbekannter und auf die Aktoren wirkender Störeffekte erweiterten Prozessmodells

Die Verknüpfung des Prozessmodells mit der Schätzung der internen Modellunsicherheit sowie der der unbekannten Störeffekte hat zum Ziel, die Vorhersagegüte durch eine ganzheitliche Abbildung aller Einflussfaktoren zu verbessern. Eine optimierte Prädiktionsfähigkeit ermöglicht im Kontext modellbasierter prädikativer Ansätze eine Steigerung der Regelgüte. Das soll durch die Erzeugung einer Kompensation der auftretenden Anomalien zur Anpassung der Stellgrößen erreicht werden, die zu einer verbesserten Qualität von Prozess und Produkt beiträgt.

Um das zu erreichen, behandeln die folgenden drei Kapitel die noch offenen und dafür notwendigen Aspekte, die die Detektion, die Modellierung sowie die Kompensation von Anomalien betreffen. Sie beantworten die bei der vorangegangenen Beschreibung des erweiterten Prozessmodells unbeantworteten Fragen:

- Wie können die auf die Aktoren wirkenden, unbekannten Störeffekte zuverlässig detektiert und die entsprechenden Beobachtungen Δ<u>ũ<sub>n</sub>(k)</u> generiert werden?
- Welcher (neuronale) Ansatz der Modellbildung kann zur Quantifizierung der Teilmodelle  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$  genutzt werden, um der Komplexität diskreter Fertigungsprozesse und der vorhandenen Wirkzusammenhänge Rechnung zu tragen?

Nach der Beantwortung der beiden Fragen wären an dieser Stelle alle Komponenten spezifiziert und das erweiterte Prozessmodell vervollständigt, sodass dieses für die Modellierung jedes diskreten Fertigungsprozesses der Prozesskette eingesetzt werden kann. Die Prozessmodelle können in ein übergeordnetes Schema zur Optimierung der Qualität entlang der gesamten Prozesskette eingebettet werden, das Schwankungen und Ausschuss reduziert. Das letzte Kapitel des Konzeptes beantwortet in diesem Zusammenhang die folgende Frage:

 Wie kann ein alle Fertigungsprozesse verknüpfendes Schema zur Optimierung der Qualität gestaltet werden, das unter Verwendung des erstellten Prozessmodells eine Kompensation von Anomalien durch eine Anpassung der Stellgrößen realisiert?

# 4.3.2 Detektion anomaler Systemdynamiken

Nachdem das Prozessmodell in seiner Struktur im vorangegangenen Kapitel vollständig hergeleitet wurde, kann nun genauer auf die Ausgestaltung einzelner Modellkomponenten eingegangen werden. Das Prozessmodell setzt zur Generierung der Beobachtungen  $\Delta \underline{\tilde{u}}_n(k)$  einen Mechanismus zur Anomalie-Detektion voraus. Im vorherigen Kapitel wurde dieser als Black Box betrachtet und dessen generierte Beobachtungen in die Struktur des Prozessmodells eingebettet. Im Rahmen dieser Arbeit wird somit ein Beobachter für dynamische Nichtlinearitäten mit zumeist nicht messbarem Eingangsraum, die als externe Störungen auf das System einwirken, benötigt. Um diese unbekannten und nicht direkt messbaren Störeinflüsse auf das Prozessverhalten erfassen zu können, muss der Mechanismus aus Abweichungen des Prozessausgangs vom Sollwert Rückschlüsse auf die im Prozess vorhandenen Anomalien ziehen können. Darüber hinaus sollte auf Basis der detektierten Abweichungen eine Modellbildung möglich sein, sodass Vorhersagen über zu erwartende Anomalien und dadurch induzierte Prozessschwankungen getroffen werden können. An dieser Stelle wird nun auf den Aufbau und die Funktionsweise des konzipierten Anomalie-Detektors sowie die Weiterverarbeitung der dadurch erfassten Daten eingegangen.

#### 4.3.2.1 Beobachterstruktur zur Detektion von Anomalien

Um auf Basis von Zusammenhang 46 eine angemessene Schätzung  $\Delta \underline{\hat{u}}_n(k)$  der unbekannten und nicht direkt messbaren realen Störung  $\Delta \underline{u}_n(k)$  zu erhalten, wird eine möglichst exakte Beobachtung  $\Delta \underline{\hat{u}}_n(k-m)$  vergangener Störungen  $\Delta \underline{u}_n(k-m)$  angestrebt. Dafür wird ein Störungsbeobachter (engl. Disturbance Observer, DOB) entworfen, der bereits aufgetretene Störeffekte aus messbaren Kenngrößen extrahieren und zu  $\Delta \underline{\hat{u}}_n(k-m)$  rekonstruieren kann. Aufgrund des in Abschnitt 4.3.1 beschriebenen Umstands, dass interne Modellunsicherheiten durch zusätzliches Training auf realen historischen Daten reduziert werden können, wird ein neues Modell auf denselben Daten trainiert, um das inverse Prozessverhalten abzubilden. Die Datengrundlage für das inverse Modell besteht neben den realen Daten des normalen Prozessverhaltens zusätzlich aus den bereits vorhandenen Simulationsdaten. Das Modell kapselt die inversen Dynamiken des normalen Prozessverhalten unter Berücksichtigung vergangener externer Störungen. Es kann, wie in Gleichung 47 zu sehen, beschrieben werden:

$$\underline{\tilde{u}}_{n}^{p}(k) = f_{n}^{-1}\left(\underline{y}_{n}(k) - \Delta \underline{\hat{y}}_{n}(k), \underline{y}_{n}(k-1), \dots, \underline{y}_{n}(k-n_{y}), \underline{\tilde{u}}_{n}^{p}(k-1), \dots, \underline{\tilde{u}}_{n}^{p}(k-n_{u}), \underline{w}_{n}(k), k\right)$$
(47)

wobei  $\tilde{u}_n^p(k)$  die Beobachtung der tatsächlich am Prozess wirkenden Stellgrößen darstellt. Die ohne das Vorhandensein externer Störungen am Prozess wirkenden Stellgrößen entsprechen  $\underline{u}_n(k-m)$ . Das inverse Modell  $f_n^{-1}(\cdot)$  benötigt als Eingang den um die Schätzung der internen Modellunsicherheit bereinigten realen Prozessausgang, da das Modell  $f_n(\cdot)$  das durch die Simulation beschriebene normale Prozessverhalten mit interner Modellunsicherheit darstellt. Die Beobachtung entspricht folglich der Stellgröße, die aufgebracht werden muss, um die Ausgabe  $y_n(k)$  zu erzielen. Daher kann das inverse Modell zur Erzeugung der tatsächlich am Prozess wirkenden Stellgrößen verwendet werden. Die Beobachtung kapselt allerdings neben beobachteten Störeffekten auch die bei der Rücktransformation durch Gleichung 47 entstehenden Fehler, die auf dem Schätzfehler der internen Modellunsicherheit  $\Delta \hat{y}_n(k)$  sowie den Beobachtungsfehlern der vergangenen externen Störungen  $\underline{\tilde{u}}_n^p(k-1), ..., \underline{\tilde{u}}_n^p(k-n_u)$  beruhen. Diese Fehlerterme sind ursächlich für Fehler bei der Beobachtung und werden bei der Rücktransformation auf die Stellgröße  $\underline{\tilde{u}}_n^p(k)$  projiziert. Durch einen Abgleich mit dem ursprünglich aufgebrachten Stellvektor  $\underline{u}_n(k)$  kann die um die Fehlerterme verrauschte Beobachtung der externen Störung  $\Delta \tilde{\underline{u}}_n(k) + \Delta$  extrahiert werden. Durch die Integration eines Filters in das Detektionsschema kann eine Fehlerfortpflanzung unterdrückt werden, sodass die bereinigten Beobachtungen der externen Störeffekte  $\Delta \underline{\tilde{u}}_n(k)$  extrahiert werden können. Ein die Informationsflüsse dieser Vorgehensweise beschreibendes Blockschaltbild ist in Abbildung 4.9 zu finden. Dieses orientiert sich hinsichtlich seiner grundlegenden Struktur an gewöhnlichen Störungsbeobachtern, wie man sie aus regelungstechnischen Anwendungen kennt [161].



Abbildung 4.9: Schema zur Anomalie-Detektion basierend auf einem inversen Prozessmodell An dieser Stelle bleibt nun zu klären, wie die Modellbildung für die inverse Abbildung  $f_n^{-1}(\cdot)$ realisiert werden kann. Dazu wird auf die in Kapitel 2.1.3 vorgestellte LSTM-Zelle

zurückgegriffen, die in diverse Netzwerkarchitekturen zur Erfüllung verschiedenster Aufgaben eingebettet werden kann. Zur Realisierung eines inversen Modells des normalen Prozessverhaltens können die Vorteile der in Kapitel 2.1.4 eingeführten Autoencoder Netzstruktur in Kombination mit der LSTM-Zelle genutzt werden. Der aus einem Encoder und einem Decoder zusammengesetzte LSTM-Autoencoder kann durch das Training mit normalen Prozessdaten die gewünschte Inversion dieser Dynamiken in Form einer netzinternen Repräsentation erzeugen. Dazu wird im Folgenden der in Abbildung 4.10 illustrierte Aufbau des LSTM-Autoencoders sowie dessen Integration in das zuvor vorgestellte Detektionsschema beschrieben. Dem LSTM-Autoencoder ist zur Vorverarbeitung und Bereinigung der Eingangsdaten eine DWT vorgeschaltet. Dazu werden sowohl aus den numerischen Simulationen extrahierte Daten als auch reale historische Daten des normalen Prozessverhaltens als Eingangsdaten für Training, Validierung und Test verwendet. Die durch die DWT erzeugten Merkmale werden im nächsten Schritt an den Encoder weitergeleitet. Dieser hat die Aufgabe, eine Projektion dieses Merkmalsraums in einen niedrigdimensionalen Unterraum durchzuführen und kapselt dabei die Eingangs-Ausgangs-Beziehungen des Prozesses. Der erzeugte Unterraum entspricht abgebildet auf das Detektionsschema den Prozessausgängen bei normalem Prozessverhalten. Der trainierte Encoder kann auch separat zur Dimensionsreduktion im Rahmen weiterführender Anwendungen herangezogen werden. Der Decoder erhält die Prozessausgänge als Eingaben und erzeugt daraus mit Hilfe einer die Rohsignale rekonstruierenden DWT die Eingänge, die hätten aufgebracht werden müssen, um die erzielten Ausgänge zu erzeugen. Der Decoder wird darauf trainiert, die Prozessdaten aus dem reduzierten Merkmalraum der verdeckten Schicht zu rekonstruieren und erlernt damit die inverse Dynamik des normalen Prozessverhaltens. Somit dient der Decoder-Teil als inverses Modell  $f_n^{-1}(\cdot)$  des ursprünglichen Metamodells  $f_n(\cdot)$ . Dieses neuronale inverse Modell wird zur Erkennung von Abweichungen und zur Erzeugung von Schätzungen auf der Grundlage der Ausgabe des realen Prozesses verwendet. Der Rekonstruktionsfehler des gesamten Autoencoders zeigt dann die Fähigkeit des Decoders zur Rekonstruktion der Eingangsdaten an und wird als Indikator für anomales Prozessverhalten verwendet [27].



Abbildung 4.10: LSTM-Autoencoder mit vorgeschalteter DWT zur Anomalie-Detektion

Zur Realisierung des Modells  $f_n^{-1}(\cdot)$  wird der Decoder-Teil aus dem trainierten Gesamtnetz entnommen und gemäß Abbildung 4.9 in das Detektionsschema eingesetzt.

#### 4.3.2.2 Weiterverarbeitung der erfassten Anomalie-Daten

Der im vorherigen Kapitel vorgestellte Anomalie-Detektor kann nun für die Charakterisierung der Störeffekte  $\Delta u_n(k)$  und zur entsprechenden Erzeugung der Beobachtungsdaten  $\Delta \tilde{u}_n(k)$ eingesetzt werden. Dieser Abschnitt widmet sich basierend darauf der Erstellung eines Konzeptes zur Verwaltung und Nutzung der zur Verfügung stehenden historischen Prozessdaten sowie der durch den Anomalie-Detektor generierbaren Beobachtungs- bzw. Anomalie-Daten. Die im Kontext der Anomalie-Detektion untersuchte Literatur zeigt im Allgemeinen auf, dass die entwickelten Ansätze primär zum Erlernen des normalen Prozessverhaltens eingesetzt werden, um durch einen Abgleich der das normale Verhalten repräsentierenden Netzausgabe und der realen Ausgabe Anomalien zu detektieren. Wie bereits zuvor angedeutet, wird im Rahmen dieser Arbeit zusätzlich zu der Detektionsfunktionalität eine weiterführende Nutzung der detektieren Anomalie-Daten angestrebt. Die Nutzung eines Mechanismus zur Anomalie-Detektion als Grundlage für das Labeling von Daten und des Aufbaus von Prädiktionsmodellen ist bislang in der Literatur nicht thematisiert worden. Der konzipierte Anomalie-Detektor soll hier ein indirektes Labeling der erfassten Daten realisieren, sodass der Aufbau von Prädiktionsmodellen auf der Grundlage einer Anomalie-Detektion möglich wird. Die durch den Anomalie-Detektor neu erzeugbare Datengrundlage wird für das Training eines weiteren LSTM-Netzes herangezogen, dessen Ziel die Modellierung und Prädiktion anomaler Prozessdynamiken ist. Das Modell wurde zuvor auf abstrakter Ebene mit  $h_n(\cdot)$  bezeichnet. Die zu diesem Zweck entwickelten Modellierungsansätze für nichtlineares zeitvariantes Prozessverhalten werden in den nächsten Abschnitten thematisiert. In diesem Abschnitt wird hingegen das Konzept zur Erzeugung und Verwaltung der Daten zum Schaffen einer geeigneten Datengrundlage behandelt. Dazu wird in dieser Arbeit zwischen den folgenden, bereits in vorherigen Kapiteln in Teilen angedeuteten Prozessdatenquellen bzw. -typen unterschieden:

- Aus numerischen Simulationen extrahierte Prozessdaten, die das normale Prozessverhalten widerspiegeln und bereits vor dem Betrieb zur Verfügung stehen und daher zum Vortraining genutzt werden können.
- Reale Prozessdaten, die das normale Prozessverhalten widerspiegeln und bereits vor dem Betrieb zur Verfügung stehen und daher zum Vortraining genutzt werden können.
- Reale Prozessdaten, die das anomale Prozessverhalten widerspiegeln und bereits vor dem Betrieb zur Verfügung stehen und daher zum Vortraining genutzt werden können.

Gemäß des im Stand der Technik vorgestellten Ansatzes des Concurrent Learning wird im Rahmen dieser Arbeit eine Abwandlung desselben angewandt, um einen flexiblen Lernvorgang auf historischen sowie auf während des Betriebs erfassten Prozessdaten durch die Nutzung einer effizienten Datenverwaltung zu ermöglichen [113]. Der Leitgedanke ist dabei, ein flexibles Vortraining auf historischen Daten jedweder Art und ein darauf aufbauendes Weitertraining während des Betriebs zu gewährleisten. Dadurch können mit neuronalen Netzen vortrainierte Modelle im laufenden Betrieb erweitert und für ein robustes, situationsabhängiges Eingreifen in den Prozess eingesetzt werden. Dazu werden die zu den vier genannten Prozessdatenquellen erfassten Daten für jedes Bauteil n und zu jedem diskreten Zeitpunkt k als 4-Tupel der folgenden Form gespeichert:

$$\langle \underline{u}_n(k), y_n(k), \Delta \underline{\tilde{u}}_n(k), \Delta y_n(k) \rangle$$
(48)

Das Vortraining wird hinsichtlich der Datenquellen chronologisch, der Aufzählungsreihenfolge entsprechend durchgeführt. Es beginnt mit den Simulationsdaten und wird mit den realen historischen Prozessdaten fortgeführt. Da diese ersten beiden Datenquellen keine anomalen Prozessdynamiken enthalten, reduziert sich das 4-Tupel hier zu einem 3-Tupel [54]. Die nach dem Training auf den ersten beiden Datenquellen entstehenden Modelle kapseln das normale Prozessverhalten. Sie werden während des Trainings stetig verfeinert und dienen als Grundlage für den Anomalie-Detektor, der für die dritte Datenquelle zur Generierung der Anomalie-Daten benötigt wird. Dies entspricht einer Anwendung des Anomalie-Detektor auf historischen Daten. Die Erzeugung der dabei entstehenden Datenvektoren  $\Delta \tilde{u}_n(k)$  und  $\Delta y_n(k)$  setzt folglich einerseits das Vorhandensein der Prozessmodelle und seiner Inversen sowie den dadurch konstruierten Anomalie-Detektor voraus. Andererseits werden die Modelle stetig durch die neu generierten 4-Tupel verfeinert. sodass das iterative Vorgehen zu einer Verbesserung der Modellierungsgenauigkeit sowie der Detektionsgüte und in der Folge auch der Güte der weiteren Datengenerierung führt.



Abbildung 4.11: Erfassung und Verwaltung von Prozessdaten für das Vor- und Weitertraining von Modellen in Anlehnung an [54]

Eine entsprechende Systematik gilt für die Erzeugung der Daten-Tupel beim Weitertraining während des Betriebs. Der verfolgte Ansatz ist in Abbildung 4.11 visualisiert. Die Daten werden in einem echtzeitfähigen Datenbanksystem abgelegt, das das iterative Training der Modelle erlaubt. Vereinfacht ausgedrückt beinhalten die Datensätze eins und zwei die Daten zum Training des Modells  $f_n(\cdot)$  auf dessen Basis die Daten für das Training von Modell  $g_n(\cdot)$  erzeugt werden. Datensatz drei ermöglicht dann die Anwendung des auf  $f_n(\cdot)$  basierenden Anomalie-Detektors zum Generieren der Daten für das Training von Modell  $h_n(\cdot)$ . Die anderen beiden Modelle werden parallel weitertrainiert. Das Prozessmodell kann anschließend im Betrieb nach demselben Schema für die Erzeugung der zum Weitertraining der Modelle erforderlichen Daten eingesetzt werden.

Mit dem verbesserten Modellverständnis sollen an dieser Stelle die Funktionalitäten des Anomalie-Detektors rekapituliert und diesbezüglich einige Gedankenexperimente angestellt werden. Unter der Annahme, dass das inverse Prozessmodell lediglich auf dem ersten Datensatz basiert und der erzeugte Anomalie-Detektor dann auf den zweiten Datensatz angewendet wird, ergäbe sich aufgrund der fehlenden anomalen, die Stellgrößen verzerrenden Dynamiken als Beobachtung eine Repräsentanz der internen Modellunsicherheit. Dieser Umstand ist in Abbildung 4.12 illustriert. Der zweite Datensatz unterscheidet sich vom ersten Datensatz der Metamodelle durch die Integration realer Prozessdaten. Im links dargestellten Anomalie-Detektor aus Abbildung 4.9 ist dies durch eine blau markierte Grenzziehung, die das Wissen über die Modellunsicherheit  $\Delta f_n(k)$  einschließt, angedeutet. Datensatz eins fehlt die Information  $\Delta f_n(k)$ , sodass dieser ebenjene Modellunsicherheit aufweist. Datensatz zwei hingegen beinhaltet reale Prozessdaten, die die fehlenden Informationen aus Datensatz eins über diese Unsicherheit  $\Delta f_n(k)$ kapseln. Der rechte Teil der Abbildung repliziert Abbildung 4.3 und ordnet jeder Stufe des Prozesswissens einen Datensatz zu, in dem die entsprechenden Informationen enthalten sind, um das Wissen zu extrahieren bzw. zu lernen. Im Vergleich zu Datensatz zwei enthält Datensatz drei zusätzlich Informationen über anomale Prozessdynamiken, was durch eine weitere Markierung einschließlich  $\Delta \underline{u}_n(k)$  angedeutet und auf das Diagramm des Prozesswissens abbildbar ist.



Abbildung 4.12: Training des Anomalie-Detektors mit verschiedenen Datensätzen

Ist nun das inverse Modell im unteren Teil des Blockschaltbildes, wie oben erwähnt, auf Datensatz eins trainiert und der reale Prozess im oberen Teil zeigt normales Verhalten gemäß der Daten aus Datensatz zwei, so lässt sich die Modellunsicherheit als Differenz detektieren. Diese wird allerdings nicht direkt erfasst, sondern durch die Rückführung von  $\underline{y}_n(k)$  in das inverse Modell auf die Differenz der Stellgrößen  $\Delta \underline{\tilde{u}}_n(k)$  abgebildet. Die Differenz von Decoder-Ausgang  $\underline{\tilde{u}}_n^p(k)$ und aufgebrachter Stellgröße  $\underline{u}_n(k)$  entspricht dem Wert, um den die Stellgröße  $\underline{u}_n(k)$  geändert werden müsste, um den angestrebten Prozessausgang trotz der Modellunsicherheiten  $\Delta f_n(k)$  zu erreichen. Die in Kapitel 4.1.1 beschriebenen, von den simulationsbasierten Metamodellen ausgehenden Modellunsicherheiten werden demnach indirekt in Form eines Korrekturwerts für die aufzubringenden Stellgrößen geschätzt werden.

Würde der auf dem ersten Datensatz basierende Anomalie-Detektor nun auf den dritten Datensatz angewendet, so würde das Detektionsergebnis  $\Delta \underline{\tilde{u}}_n(k)$  eine Komposition aus indirekt erfasster Modellunsicherheit  $\Delta f_n(k)$  und Störeffekt  $\Delta \underline{u}_n(k)$  darstellen, die über einen Filter separiert werden müssten. Ein auf dem zweiten und dritten Datensatz beruhender Decoder bzw. Anomalie-Detektor liefert hingegen direkt den auf die Stellgrößen wirkenden Störeffekt  $\Delta \underline{u}_n(k)$ , da die den Metamodellen inhärenten Modellunsicherheiten durch die realen Daten reduziert werden. Der Decoder erzeugt eine Beobachtung der tatsächlich am Prozess wirkenden Stellgröße  $\underline{\tilde{u}}_n^p(k)$ , die theoretisch für den von Anomalien freien Prozess aufgebracht werden muss, um die gewünschte Ausgabe zu erzielen. Die Differenz aus Beobachtung und aufgebrachter Stellgröße führt dann zur Schätzung der Störung  $\Delta \underline{u}_n(k)$ . Tabelle 4.1 gibt einen Überblick über das Detektionsergebnis des Anomalie-Detektors in Abhängigkeit davon, wie er trainiert wurde.

Training des Anomalie-Detektors	Abbildung der Differenz des Prozesswissens auf die Stellgrößen
Datensatz 1 und 2	$\Delta f_n(k)$ wird auf die Stellgrößendifferenz $\Delta \underline{\tilde{u}}_n(k)$ abgebildet
Datensatz 1 und 3	$\Delta f_n(k) + \Delta \underline{u}_n(k)$ wird auf $\Delta \underline{\tilde{u}}_n(k) = \underline{u}_n(k) - \underline{\tilde{u}}_n^p(k)$ abgebildet
Datensatz 2 und 3	$\Delta \underline{u}_n(k)$ wird auf $\Delta \underline{\tilde{u}}_n(k) = \underline{u}_n(k) - \underline{\tilde{u}}_n^p(k)$ abgebildet

|--|

Es ist an dieser Stelle zu berücksichtigen, dass Schätzung  $\Delta \underline{\tilde{u}}_n(k)$  der Störung bei unmittelbarer Anwendung zur Kompensation der Anomalien einen fehlerbehafteten und damit unvollständigen Ausgleich vornimmt, da davon auszugehen ist, das sich die realen inversen Systemdynamiken und das inverse Modell stets durch einen kleinen Fehlerterm unterscheiden. Das im vorherigen Kapitel 4.3.2.1 erläuterte inverse Modell berücksichtigt auch vergangene externe Störungen und reduziert Beobachtungsfehler dadurch auf ein Minimum. Diese Konstellation wird beim Training des Anomalie-Detektors auf dem dritten historischen Datensatz und während des sich anschließenden Betriebs genutzt und führt zu der zuvor erörterten iterativen Verfeinerung von inversem Modell und Beobachtung [28].

## 4.3.3 Modellierung anomaler Systemdynamiken

Nachdem im vorherigen Kapitel auf den Mechanismus zur Detektion von auf die Aktoren wirkenden Anomalien eingegangen wurde, beschäftigt sich dieses Kapitel nun mit der Modellierung derselben. Das im Rahmen von Abschnitt 4.3.2.2 beschriebene Vorgehen dient als Grundlage der Modellbildung für die Modelle  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$ . Zu diesem Zweck erfolgt eine detaillierte Erläuterung der neuronalen Netze, die die drei Teilmodelle abbilden. Darüber hinaus wird ein neuartiges Schema zur Prädiktion vorgestellt, das die Vorhersagen multipler neuronaler Netze auf kooperative Art und Weise verknüpft und dadurch zu einer gesteigerten Prädiktionsgüte beiträgt.

## 4.3.3.1 LSTM-basierte Prädiktionsmodelle

Das Prozessmodell  $f_n(\cdot)$  und die Anomalie-Modelle der externen Störeffekte  $h_n(\cdot)$  sowie der internen Modellunsicherheit  $g_n(\cdot)$  werden, wie in Kapitel 4.2.2.2 erörtert, als nichtlineare autoregressive Modelle mit exogener Variablen aufgefasst. Dadurch ist die Struktur der Ein- und Ausgänge klar definiert. Für die abbildende Funktion selbst ist hingegen nur der Umstand der Nichtlinearität berücksichtigt. Aus den in Kapitel 3.1 vorgestellten Anforderungen geht hervor, dass der Ansatz neben der Fähigkeit zur Abbildung nichtlinearer Zusammenhänge zusätzlich die dort aufgeführten Eigenschaften in sich vereinen sollte. Um in besonderem Maße dem Gesichtspunkt der Prädiktion multipler Zeitschritte unter reduzierter Fehlerfortpflanzung und optimierter Kombination aus Kurzzeit- und Langzeitgedächtnisverhalten Rechnung zu tragen, eignen sich primär Seq2Seq-LSTM-Netze. Diese werden zur Realisierung der Modelle  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$  herangezogen und mit weiteren Ansätzen, die den aufgezeigten Anforderungen ebenfalls genügen, verglichen. Ein Seq2Seq-Netzwerk ist, wie bereits in Abschnitt 2.1.5 dargestellt, in einen die Eingabe verarbeitenden Encoder und einen die Ausgabe erzeugenden Decoder unterteilt. Die Gewichte von En- und Decoder werden separat trainiert. Der Encoder verarbeitet eine Sequenz von Eingabedaten und extrahiert anomales Prozessverhalten. Um sich das anomale Verhalten zu merken, wird über das Netzwerk iteriert, bis die Eingangssequenz vollständig vom Encoder verarbeitet ist. Der flexible Kopiermechanismus erzeugt eine reduzierte Repräsentanz der Informationen, die dann in den Decoder kopiert werden. Dieser liefert eine Vorhersage der Anomalien auf Grundlage des durch die verwendeten Neuronen erzeugten Speicherverhaltens. Diese multivariate Ausgabe entspricht ebenfalls einer Sequenz, sodass zeitgleich multiple Zeitschritte vorhergesagt werden können. Um kurz- und langfristige Anomalien modellieren und vorhersagen zu können, werden Neuronen mit Gating-Mechanismus präferiert, sodass in diesem Fall LSTM-Zellen in die Seq2Seq-Architektur integriert werden. Kurzfristig auftretende Anomalien können bspw. durch Vibrationseffekte und Langzeitanomalien durch Verschleißvorgänge verursacht werden. Die Parameter werden an die Größe der Eingangssequenz angepasst, um das bestmögliche Speicherverhalten zu realisieren. Das verwendete Seq2Seq-LSTM-Netzwerk ist in Abbildung 4.13 dargestellt.



Abbildung 4.13: Seq2Seq-Architektur basierend auf LSTM-Zellen

Der verwendete Kopiermechanismus verfügt über keinen Transducer, wie er in [20] eingeführt wurde, da aus der bauteilbezogenen Diskretisierung keine harten Echtzeitanforderungen im unteren Millisekunden-Bereich resultieren. Stattdessen wird, wie in [19] beschrieben, eine reduzierte Repräsentanz zur Verknüpfung von En- und Decoder genutzt. Diese hat im Gegensatz zum originalen Kopiermechanismus hingegen keine feste Dimension, sondern kann von Prädiktion zu Prädiktion flexibel eingestellt werden. Der verwendete Ansatz lässt sich in Anlehnung an [19] wie folgt beschreiben:

$$p(y(1), ..., y(T_A)|x(1), ..., x(T_E))$$

$$= \prod_{t=1}^{T_A} p(y(t)|v(x(1), ..., x(T_E)), y(1), ..., y(t-1))$$
(49)

Dabei wird ersichtlich, dass die reduzierte Repräsentanz v von der Eingabesequenz abhängig ist. Mit jeder Verschiebung eines gleitenden Fensters über den Eingabedaten, und damit mit jedem neu zur Eingabesequenz hinzugefügten Eingabewert, passt sich der Kopiervektor v entsprechend des Informationsgehaltes in den Daten an. Die Flexibilisierung trägt dem Umstand Rechnung, dass nicht jede neue Eingabesequenz von jedem produzierten Bauteil über eine äquivalente Informationsdichte für die Charakterisierung und Vorhersage anomaler Systemdynamiken verfügt. Durch die flexibilisierte Reduktion können redundante oder weniger relevante Informationen ausgespart und Informationen hoher Signifikanz stärker berücksichtigt werden. Darüber hinaus kommt eine leicht modifizierte Variante der in [18] vorgestellten, originalen LSTM-Zelle zum Einsatz. Die in dieser Arbeit verwendete Modifikation ist durch eine inverse Verbindung zwischen Add- und Forget-Gate gekennzeichnet. Dieser Gestaltungsvariante liegt die Idee zugrunde, dass kein Gedächtnis unendlich ist, sondern jedes Gedächtnis begrenzte Kapazitäten besitzt. Daher kommt es immer, wenn dem Zustandsvektor Informationen hinzugefügt werden, zu einer parallelen Elimination der entsprechenden Informationsmenge.



Abbildung 4.14: Kombinationsmöglichkeiten ausgewählter Modellierungselemente für die Prädiktion von Prozess- und Anomalie-Verhalten

LSTM-Zellen sind in der Lage, nichtlineare Kurz- und Langzeitzeitverhalten zu erfassen und den gesamten Informationsfluss sowie den Prozess des Vergessens über die modifizierten Gates zu steuern. Neben dem Seq2Seq-LSTM mit flexiblem Kopiermechanismus wird eine definierte Auswahl weiterer zusammengesetzter Architekturen betrachtet, die die bestehenden Anforderungen erfüllen. Auf dieser Basis werden die in Kapitel 6.1.2 dargelegten Vergleichsstudien angestellt. Die untersuchten Kombinationen aus Datenvorverarbeitung, Dimensionsreduktion, Netzwerkarchitektur, Zellarchitektur sowie Rekonstruktionsverfahren sind in Abbildung 4.14 dargestellt und an das Vorgehen in [162] angelehnt. GRU-Zellen werden aufgrund ihrer auf Kurzeitabhängigkeiten optimierten Prädiktionsfähigkeiten als alternative Zellarchitektur verwendet. GRU entspricht einer reduzierten Form von LSTM mit weniger Gates, reduziertem Parameterraum und verbesserter Berechnungszeit. Diese Architekturen werden angewandt, um die Dynamiken des Prozesses sowie die der Anomalien in getrennten Modellen, nämlich  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$ , zu erfassen. Um die Architekturen zu trainieren, muss vor allem ein auf die komplexe Struktur der verwendeten Gating-Mechanismen abgestimmtes Trainingsverfahren genutzt werden. Der nachfolgende Abschnitt fokussiert in erster Linie auf die architektonische Besonderheit der Zelle und beschreibt die Einbettung der partiellen Ableitungen entlang der Gates in die zu minimierende Kostenfunktion des Netzes. Das Vorgehen orientiert sich dabei an dem in [7] beschriebenen Vorgehen zur Ableitung mehrschichtiger Netze und kombiniert dieses mit der in [164] beschriebenen BPTT für RNN. Der Vorhersagefehler e(W, k)

des Netzes, der der Differenz aus realem Ausgang und Netzausgang entspricht, wird im Rahmen des Trainings reduziert:

$$E(W,k) = \frac{1}{2}e(W,k)^{2} = \frac{1}{2}\left(\underline{y}_{n}(k) - \underline{\hat{y}}_{n}(W,k)\right)^{2}$$
(50)

wobei W die aktuelle Gewichtsmatrix des Netzes darstellt und dieses zum Zeitpunkt k charakterisiert. Das Vorhersageergebnis ist direkt abhängig von der Gewichtsmatrix. Die Kostenfunktion wird mit E(W, k) bezeichnet und zur Erzeugung des Trainingssignals verwendet. Die partielle Ableitung der Kostenfunktion ergibt sich zu:

$$\frac{\partial E}{\partial W}(W,k) = -e(W,k) \frac{\partial \hat{y}_n}{\partial W}(W,k)$$
(51)

Die Ableitung der Kostenfunktion bestimmt die Anpassung der Gewichte bei jeder Iteration. Dieser Umstand geht auf die grundlegenden Erkenntnisse von [163] zurück und wird durch den Backpropagation-Algorithmus beschrieben:

$$W(k+1) = W(k) + \eta \, e(W,k) \, \frac{\partial \hat{y}_n}{\partial W}(W,k) \tag{52}$$

Dieses Lerngesetz beschreibt die iterative Anpassung der Gewichte mit jedem diskreten Zeitschritt bzw. Bauteil. Die Anpassung der Gewichte wird von der partiellen Ableitung des Netzausgangs bestimmt. Die Ableitung muss rekursiv von Schicht zu Schicht über das gesamte Netzwerk berechnet werden [54]. Dazu werden die Ableitungen entlang der Struktur des Netzes und entlang der Gates der LSTM-Zellen bestimmt. Die Aggregation der einzelnen Ableitungen hängt von der Gesamtarchitektur des Netzes ab. Daher unterscheidet sich dieser Vorgang von Netzarchitektur zu Netzarchitektur. Um die Ableitungen entlang der Netzarchitektur mit den Ableitungen innerhalb der LSTM-Zellarchitektur zu verknüpfen, wird die abstrakte Funktion  $\Gamma$ als Repräsentanz der von der LSTM-Zelle realisierten Transformation definiert. Diese kapselt die Berechnungsstruktur der LSTM-Zelle inklusive der mit dem Gating-Mechanismus verbundenen Transformationen des Zellzustands. Für die Verknüpfung zweier LSTM-Schichten gilt folgende Beziehung:

$$\underline{h}^{l}(k) = \underline{\Gamma}\left(\underline{s}^{l}(k)\right) = \underline{\Gamma}\left(W^{l}[\underline{x}^{l}(k) \ \underline{h}^{l+n}(k-1)] + \underline{b}^{l}\right) \quad \text{mit } 0 \le n \le l_{max} - l \tag{53}$$

wobei l die betrachtete Schicht kennzeichnet und  $\underline{\Gamma}$  der vektoriellen Ausprägungsform der abstrakten Funktion  $\Gamma$  entspricht, die die Transformation einer ganzen LSTM-Schicht repräsentiert. Die Matrix  $W^l$  entspricht den Gewichten und  $\underline{b}^l$  dem Bias-Vektor. Die Ausgänge der vorangegangenen Schicht zum vorherigen Zeitschritt fungieren als Eingänge  $\underline{x}^l(k)$  der betrachteten Schicht l. Es gilt daher folgender Zusammenhang zur Verknüpfung zweier aufeinanderfolgender Schichten:

$$\underline{x}^{l}(k) = \underline{h}^{l-1}(k-1)$$
(54)

Darüber hinaus werden die Ausgänge beliebiger Neuronen nachfolgender Schichten l + n wieder zurückgeführt und mit  $\underline{h}^{l+n}(k-1)$  bezeichnet. Ein Wert von n = 0 würde einer direkten oder lateralen und ein Wert von  $n \neq 0$  einer indirekten Rückführung entsprechen. Zell- bzw. Neuronenausgänge innerhalb des Netzes werden dabei im Allgemeinen mit h und Ausgänge des Gesamtnetzes mit y abgekürzt. Der Parameter  $l_{max}$  gibt die Gesamtzahl an Netzschichten an. Die Gesamtheit aller Eingänge einer Schicht l zum Zeitpunkt k wird mit  $\underline{s}^{l}(k)$  zusammengefasst. Auf Basis dieser allgemeinen Beschreibung rekurrenter Netze können durch die Matrix Wverschiedene Netzarchitekturen und durch die Funktion  $\underline{\Gamma}$  verschiedene Zellarchitekturen repräsentiert werden. So kann  $\underline{\Gamma}$  bspw. auch klassische Aktivierungsfunktionen oder die GRU-Architektur kapseln. Um die für das in Gleichung 52 beschriebene Lerngesetz relevante partielle Ableitung des Netzausgangs herzuleiten, wird der Netzausgang mit jedem einzelnen im Netz enthaltenen Gewicht in Beziehung gesetzt. Die Ableitung des Netzausgangs nach dem Gewicht  $W_{ij}^{l}$ , das Neuron i in Schicht l und Neuron j einer beliebigen anderen Schicht verbindet, lässt sich wie folgt beschreiben:

$$\frac{\partial \hat{\underline{y}}_n}{\partial W} = \begin{bmatrix} \frac{\partial \hat{\underline{y}}_n}{\partial W^l} \end{bmatrix} \quad mit \frac{\partial \hat{\underline{y}}_n}{\partial W^l} = \frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^l} \frac{\partial \underline{s}^l}{\partial W^l} = \begin{bmatrix} \frac{\partial \hat{\underline{y}}_n}{\partial W^l_{ij}} \end{bmatrix} \quad mit \ \frac{\partial \hat{\underline{y}}_n}{\partial W^l_{ij}} = \frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^l_i} \frac{\partial \underline{s}^l_i}{\partial W^l_{ij}} \tag{55}$$

Auf die Angabe der diskreten Zeitschritte k wird der Übersicht halber im Folgenden verzichtet. Der erste Zusammenhang beschreibt die für die vollständige Darstellung des Lerngesetzes in Gl. 52 gesuchte Ableitung als Zusammensetzung partieller Ableitungen nach den einzelnen Schichten des Netzes l. Diese können wiederum durch die Verknüpfung des Netzausgangs mit den Zelleingängen der einzelnen Schichten ermittelt werden, was durch den zweiten Zusammenhang offensichtlich wird. Die Ableitungen nach den einzelnen Schichten können schließlich als Zusammensetzung partieller Ableitungen nach den einzelnen Gewichten in der jeweiligen Schicht aufgefasst werden. Dies wird durch den dritten Zusammenhang beschrieben. Im Folgenden werden zur Bestimmung der für das Lerngesetz benötigten Ableitungsmatrix die beiden Multiplikatoren des zweiten Zusammenhangs genauer aufgelöst. Der zweite Faktor lässt sich auf Basis von Gleichung 53 leicht zu:

$$\frac{\partial \underline{s}^{l}}{\partial W^{l}} = \begin{bmatrix} \frac{\partial \underline{s}^{l}_{i}}{\partial W^{l}_{ij}} \end{bmatrix} = \begin{bmatrix} X^{l} H^{l+n} \end{bmatrix} \quad mit \frac{\partial \underline{s}^{l}_{i}}{\partial W^{l}_{ij}} = \begin{bmatrix} \underline{x}^{l}_{j} \underline{h}^{l+n}_{j} \end{bmatrix}$$
(56)

ermitteln. Die Änderung von  $\underline{s}^{l}$  mit allen Gewichten der Schicht lässt sich in Matrixform beschreiben. Sie beinhaltet eine Teilmatrix mit allen Ausgängen der vorangegangenen Schicht, die aus der Anzahl an Neuronen der Schicht *l* entsprechenden Anzahl an Vektoren besteht. Die zweite Teilmatrix enthält dieselbe Anzahl an Vektoren und umfasst alle rückgeführten Ausgänge aus der Schicht *l* sowie aller nachfolgenden Schichten. Die Gesamtmatrix hat die Dimension  $n \times m$ , wobei *n* der Anzahl der Neuronen der Schicht *l* und *m* der Summe der Neuronen in den Schichten l - 1 bis  $l_{max}$  entspricht. Diese Vektoren unterscheiden sich hingegen aufgrund der für jedes Neuron der Schicht *l* unterschiedlich strukturierten Rückführungen. Für die partielle Ableitung des Netzausgangs mit den Zelleingängen beliebiger Schichten aus Gleichung 55 lässt sich eine rekursive Berechnungsvorschrift herleiten:

$$\frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^l} = \frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^{l+1}} \frac{\partial \underline{s}^{l+1}}{\partial \underline{s}^l} \quad mit \ \frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^{l+1}} = \frac{\partial \hat{\underline{y}}_n}{\partial \underline{s}^{l+2}} \frac{\partial \underline{s}^{l+2}}{\partial \underline{s}^{l+1}} \ etc.$$
(57)

Der erste Faktor lässt sich formal durch die Ableitung entlang der abstrakten, die Zellstruktur kapselnden Funktion  $\underline{\Gamma}$  bestimmen und entspricht nach der Auflösung der Rekursion immer der partiellen Ableitung der letzten Netzschicht. Dementsprechend gilt:

$$\frac{\partial \hat{y}_n}{\partial \underline{s}^{l_{max}}} = \underline{\dot{\Gamma}}(\underline{s}^{l_{max}}) \quad mit \ \underline{\hat{y}}_n = \left[\underline{\hat{y}}_{i,n}\right] und \ \underline{\hat{y}}_{i,n} = \underline{h}_i^{l_{max}} = \Gamma(\underline{s}_i^{l_{max}})$$
(58)

Der zweite Faktor entspricht der partiellen Ableitung zwischen den einzelnen Schichten des Netzes. Die Rückführungsstrukturen des rekurrenten Netzes bedingen dabei allerdings eine Abhängigkeit von nachfolgenden Schichten. Eine Ableitung von Schicht l + 1 nach der Schicht l kann aufgrund der Rückführungen auch partielle Ableitungen von l + 1 nach l + 1 + n enthalten. Der zweite Faktor kann demnach wie folgt aufgelöst werden:

$$\frac{\partial \underline{s}^{l+1}}{\partial \underline{s}^{l}} = \frac{\partial \underline{s}^{l+1}}{\partial \underline{x}^{l+1} \partial \underline{h}^{l+1+n}} \frac{\partial \underline{x}^{l+1} \partial \underline{h}^{l+1+n}}{\partial \underline{s}^{l}} = W^{l+1} [\underline{\dot{x}}^{l+1} \underline{\dot{h}}^{l+1+n}] = W^{l+1} \underline{\dot{\Gamma}}(\underline{s}^{l}) \underline{\dot{\Gamma}}(\underline{s}^{l+n})$$
(59)

Die bereits in Gleichung 53 eingeführte Gewichtsmatrix  $W^l$  verknüpft die vorwärts gerichteten und rekurrenten Verbindungen  $\underline{x}^{l}$  und  $\underline{h}^{l+n}$  und führt diese zu den Ausgängen der betrachteten Schicht  $\underline{h}^{l}$  zusammen. Sie umfasst daher alle Gewichte der vorwärts gerichteten und rekurrenten Verbindungen, die zu allen Neuronen i von Schicht l führen und die Eingänge dieser Neuronen festlegen. Die Matrix hat ebenfalls die Dimension  $n \times m$ , wobei n der Anzahl der Neuronen der Schicht l und m der Summe der Neuronen in den Schichten l - 1 bis  $l_{max}$  entspricht. Die ersten p Spalten der Matrix entsprechen den Verbindungen von Schicht l zu Schicht l-1. Die erste Spalte enthält bspw. die Gewichte der Verbindungen von allen Neuronen der Schicht l zum ersten Neuron der Schicht l - 1. Die folgenden Spalten beinhalten die Gewichte der Verbindungen von Schicht l zu l, l zu l + 1 usw. bis zur Verknüpfung von l und  $l_{max}$ . Daher entspricht die Gewichtsmatrix unter Berücksichtigung der Indexverschiebung der Ableitungsmatrix des Neuroneneingangs  $\underline{s}^{l+1}$  nach allen eingehenden Verbindungen. Der zweite Faktor entspricht den partiellen Ableitungen entlang der Zellstrukturen der mit den Neuronen der Schicht l+1verbundenen Neuronen der Schichten l-1 bis  $l_{max}$ . Um die gesuchte Ableitungsmatrix aus Gleichung 54 und damit das Lerngesetz aus Gleichung 51 vollständig bestimmen zu können, muss an dieser Stelle somit die Struktur der genutzten Neuronen bzw. Zellen genauer betrachtet werden, um die noch unbestimmten Ableitungsfunktionen  $\underline{\dot{\Gamma}}(\underline{s}^{l})$  und  $\underline{\dot{\Gamma}}(\underline{s}^{l+n})$  quantifizieren zu können. Die Zellarchitektur bestimmt das Übertragungsverhalten der Funktion  $\Gamma$  und folglich auch die Ableitung  $\dot{\Gamma}$ . Sie unterscheidet zwischen GRU- sowie LSTM-Zellen oder klassischen Aktivierungsfunktionen. Im Folgenden wird die Ableitung  $\dot{\Gamma}$  für die zuvor beschriebene LSTM-Architektur ermittelt [164]. Der Zellzustand  $\underline{z}(t)$  erfasst bei der LSTM-Zelle die zu speichernde Information. Dieser wird durch das Forget und Add Gate, die das Vergessen und Hinzufügen von Informationen und damit das Verhalten des Gedächtnisses steuern, angepasst. Dabei wird das Vorgehen beispielhaft anhand des Forget Gates erörtert. Es ergibt sich auf Basis der in Kapitel 2.1.3 eingeführten Zellarchitektur folgende Beziehung für die Ableitung der Zellausgabe  $\underline{h}_i(k)$ nach dem Zellzustand  $z_i(k)$  der Zelle *i* zum Zeitpunkt *k*:

$$\frac{\partial \underline{h}_{i}(k)}{\partial \underline{z}_{i}(k)} = \left(1 - tanh^{2}\left(\underline{z}_{i}(k)\right)\right) \odot \underline{o}_{i}(k) \tag{60}$$

Sie wird durch den Aufbau des Output Gates festgelegt. Der Vektor  $\underline{o}_i(k)$  kapselt den Einfluss der Ausgänge vorangegangener Schichten sowie den der rückgekoppelten und zeitverzögerten Ausgänge beliebiger Zellen nachfolgender Schichten auf die aktuelle Ausgabe der Zelle *i*:

$$\underline{o}_{i}(k) = \sigma_{\underline{o}}\left(\underline{s}_{i}^{l}(k)\right) = \sigma_{\underline{o}}\left(W_{i}^{l}\left[\underline{x}_{i}^{l}(k)\,\underline{h}_{j}^{l+n}(k-1)\right] + \underline{b}_{i}^{l}\right)$$
(61)

Die Gewichtsmatrix  $W_i^l$  entspricht dabei einer Teilmatrix der Gewichtsmatrizen  $W^l$  bzw. W, die die Gewichte der Schicht l bzw. die des ganzen Netzes enthalten. Die Teilmatrix hat die Dimension  $1 \times m$  und stellt damit die die Gewichte für die Zelle i kapselnde Zeile dar. Die Sigmoid-Funktion des Output Gates wird mit  $\sigma_{\underline{o}}$  bezeichnet. Darüber hinaus ergibt sich für die Ableitung der Zellausgabe  $\underline{h}_i(k)$  nach dem Forget-Vektor  $f_i(k)$  folgende Beziehung:

$$\frac{\partial \underline{h}_{i}(k)}{\partial \underline{f}_{i}(k)} = \frac{\partial \underline{h}_{i}(k)}{\partial \underline{z}_{i}(k)} \frac{\partial \underline{z}_{i}(k)}{\partial \underline{f}_{i}(k)} = \frac{\partial \underline{h}_{i}(k)}{\partial \underline{z}_{i}(k)} \Big( \underline{z}_{i}(k-1) - \underline{i}_{i}(k) \Big) \quad mit \, \underline{i}_{i}(k) = \sigma_{\underline{i}} \left( \underline{s}_{i}^{l}(k) \right) \tag{62}$$

Diese rekursive, ,Gate-bezogene' Vorgehensweise ähnelt der auf die Netzarchitektur angewendete Vorgehensweise, die eine schichtweise Auflösung der partiellen Ableitung des Netzausgangs nach den Gewichten vorsieht. Die nachfolgend dargestellte Ableitung des Zellausgangs beinhaltet die Ableitung 62, die wiederum Ableitung 60 beinhaltet:

$$\frac{\partial \underline{h}_i(k)}{\partial \underline{s}_i^l(k)} = \frac{\partial \underline{h}_i(k)}{\partial f_i(k)} \frac{\partial \underline{f}_i(k)}{\partial \underline{s}_i^l(k)}$$
(63)

$$mit\frac{\partial \underline{f}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)} = \dot{\sigma}_{\underline{f}}\left(\underline{s}_{i}^{l}(k)\right) = \left(1 + \tanh\left(\underline{s}_{i}^{l}(k)\right)\right) \left(1 - \tanh\left(\underline{s}_{i}^{l}(k)\right)\right)$$
(64)

Das Forget Gate kann einfach nach den Zelleingängen abgeleitet werden. Das Resultat entspricht der Ableitung der für das Forget Gate genutzten Sigmoid-Funktion. Die Ableitung des

Zellzustands nach dem Forget Gate geht auf den um einen Zeitschritt versetzten Zellzustand zurück und die des Zellausgangs basiert, wie eingangs beschrieben, auf der Struktur des Output Gates. Wird nicht nur das Forget Gate sondern der gesamte Gating-Mechanismus betrachtet, wird Gleichung 63 zu Ausdruck 65 erweitert, der die Ableitungen nach allen Gates inkludiert und damit der gesuchten Ableitungsfunktion  $\dot{\Gamma}$  entspricht:

$$\dot{\Gamma}\left(\underline{s}_{i}^{l}(k)\right) = \frac{\partial \underline{h}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)} = \left[\frac{\partial \underline{h}_{i}(k)}{\partial \underline{f}_{i}(k)} \frac{\partial \underline{h}_{i}(k)}{\partial \underline{i}_{i}(k)} \frac{\partial \underline{h}_{i}(k)}{\partial \underline{o}_{i}(k)}\right]^{T} \left[\frac{\partial \underline{f}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)} \frac{\partial \underline{i}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)} \frac{\partial \underline{o}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)} \frac{\partial \underline{o}_{i}(k)}{\partial \underline{s}_{i}^{l}(k)}\right]$$
(65)

Die Ableitungsfunktion  $\dot{\Gamma}(\underline{s}_{i}^{l})$  entspricht derjenigen einer Zelle *i* der Schicht *l*. Um die zur Vervollständigung von Gl. 55 und damit zur Vervollständigung des Lerngesetzes in Gl. 52 benötigten Ableitungsfunktionen  $\underline{\Gamma}(\underline{s}^{l})$  aller Schichten *l* bzw. die Ableitungsfunktion  $\underline{\Gamma}(\underline{s})$  für das gesamte Netz zu erhalten, wird folgende Matrix aus den Teilmatrizen in Gl. 65 erzeugt:

$$\underline{\dot{\Gamma}}(\underline{s}(k)) = \left[\dot{\Gamma}\left(\underline{s}^{l}(k)\right)\right] \quad mit \ \underline{\dot{\Gamma}}(\underline{s}^{l}(k)) = \left[\dot{\Gamma}\left(\underline{s}^{l}(k)\right)\right] \tag{66}$$

Dadurch kann die partielle Ableitung des kompletten Netzausgangs nach allen Gewichten des Netzes zu jedem diskreten Zeitschritt bestimmt werden, sodass eine Anpassung der Gewichte nach Gl. 52 mit jedem Zeitschritt möglich ist. Daraus folgt gemäß [54] und Gleichung 52 für die Ableitung des Parameterfehlers der Gewichte  $\delta$ :

$$\frac{\partial}{\partial t}\delta = \frac{\partial}{\partial t}W = -\eta \ e(W,k) \ \underline{\dot{\Gamma}}\left(\underline{s}(W,k)\right) \quad mit \ \delta = W - W^*$$
(67)

Die Matrix  $W^*$  beschreibt die durch das Training zu erreichenden Zielgewichte des Netzes, die der exakten Abbildung des realen Prozessverhaltens entsprechen würden. Um die Stabilität des Lernvorgangs belegen zu können, muss nachgewiesen werden, dass es sich bei der Ableitung des Parameterfehlers um eine negativ semidefinite Funktion handelt und der Parameterfehler selbst somit konvergiert. Dies ist in Anlehnung an den Beweis in [54] für die in dieser Arbeit genutzten LSTM-Netze genau dann der Fall, wenn für ihre Struktur  $\Gamma$  folgende Beziehung Gültigkeit hat:

$$e(W,k) = \underline{\Gamma}\left(\underline{s}(W^*,k)\right) - \underline{\Gamma}\left(\underline{s}(W,k)\right) = \underline{\dot{\Gamma}}\left(\underline{s}(W,k)\right) * \left(W^{*T} - W^T\right)$$
(68)

Das bedeutet, dass die Netzstruktur  $\Gamma$  eine Transformation des Ausgabefehlers des Netzes in eine Repräsentation der Gewichtsmatrizen und der hergeleiteten Ableitungsfunktion erlaubt. Damit ist der stabile Lernvorgang in seiner Gesamtheit beschrieben, sodass ein Training der vorgestellten Architekturen und eine entsprechende Nutzung der trainierten Netze zur Prädiktion von Prozessund Anomalie-Dynamiken möglich ist. Die gelernten neuronalen Modelle sind einerseits von Natur aus zeitvariant mit *k* und ändern andererseits ihre Gewichte *W* iterativ auf Basis des vorgestellten Trainingsverfahrens. Die Anpassung der Gewichte erfolgt dabei ebenfalls mit jedem diskreten Zeitschritt *k*. Das Verfahren wird sowohl zum Offline-Vortraining auf den historischen Datensätzen eins bis drei sowie zum Online-Weitertraining während des Betriebs eingesetzt.

## 4.3.3.2 Kooperativer Prädiktionsansatz

Die im vorherigen Kapitel vorgestellten Netzarchitekturen und das zugehörige Trainingsverfahren werden in diesem Abschnitt angewandt, um die Prozessdynamiken sowie die externen Störeffekte und die internen Modellunsicherheiten in den separaten Modellen  $f_n(\cdot)$ ,  $h_n(\cdot)$  und  $g_n(\cdot)$ abzubilden und für eine gemeinsame und kooperative Prädiktion miteinander zu verknüpfen. Erste Ansätze für die Verknüpfung von prädizierten Störinformationen oder Störmodellen mit regulären Prozessmodellen finden sich in [13] und [14]. Das in dieser Arbeit entwickelte Verfahren orientiert sich an den genannten Vorarbeiten und erweitert diese. Das Verfahren zur kooperativen Prädiktion des Prozesses durch zusätzliche Berücksichtigung der LSTM-basierten Vorhersage der anomalen Prozessdynamiken ist in Abbildung 4.15 dargestellt. Die reduzierte Darstellungsweise ist derjenigen in [14] entlehnt. Die vereinfachten, oben dargestellten Netzdarstellungen veranschaulichen die Vorhersagen mit dem Prozessmodell  $f_n(\cdot)$  inklusive derjenigen des integrierenden Modells  $g_n(\cdot)$  zur Reduktion interner Modellunsicherheiten. Die unten dargestellten Netze entsprechen den Vorhersagen mit dem Modell der Störeffekte  $h_n(\cdot)$ . Die abgebildeten Netze werden als Seq2Seq-LSTM-Netze mit flexiblem Kopiermechanismus realisiert. Die Vorhersage des Anomalie-Modells wird gemäß Gleichung 44 generiert und fungiert als Eingang des Prozessmodells. Die kooperative Prädiktion einer Ausgabesequenz kann, aufbauend auf dem erweiterten Prozessmodell aus Kapitel 4.3.1, wie folgt beschrieben werden:

$$\left[\underline{\hat{y}_n}(k+n_{\hat{y}}), \dots, \underline{\hat{y}_n}(k)\right] = f_n\left(\underline{y_n}(k-1), \dots, \underline{y_n}(k-n_y), \underline{\hat{u}_n}^p(k+n_{\hat{y}}), \dots, \underline{\hat{u}_n}^p(k), \\ \underline{\tilde{u}_n}^p(k-1), \dots, \underline{\tilde{u}_n}^p(k-n_u), \underline{w_n}(k), k\right) + \left[\Delta \underline{\hat{y}_n}(k+n_{\hat{y}}), \dots, \Delta \underline{\hat{y}_n}(k)\right]$$
(69)

Das Modell  $f_n(\cdot)$  basiert auf den vergangenen Messungen der Prozessausgänge, den vergangenen Beobachtungen der am Prozess wirkenden Stellgrößen und den beim vorangegangenen Prozessschritt erzielten Produkteigenschaften. Im Sinne des Seq2Seq-Ansatzes werden Eingabesequenzen zur Prädiktion von Ausgabesequenzen verwendet [19]. Daher erzeugen die durch  $h_n(\cdot)$  bestimmte Sequenz an Stellgrößen mit Störungsschätzung  $\hat{\underline{u}}_n^p(k + n_{\hat{y}}), ..., \hat{\underline{u}}_n^p(k)$ sowie die durch  $g_n(\cdot)$  vorhergesagte Sequenz an Modellunsicherheiten  $\Delta \underline{\hat{y}}_n(k + n_{\hat{y}}), ..., \Delta \underline{\hat{y}}_n(k)$ die Vorhersage der Ausgabesequenz. Für die Vorhersage einer Sequenz externer Störeffekte gilt:

$$\left[\Delta\underline{\hat{u}}_{n}(k+n_{\hat{y}}), \dots, \Delta\underline{\hat{u}}_{n}(k)\right] = h_{n}\left(\Delta\underline{\tilde{u}}_{n}(k-1), \dots, \Delta\underline{\tilde{u}}_{n}(k-n_{u}), k\right)$$
(70)

Für die Ermittlung der in Gleichung 69 benötigten Sequenzen  $\underline{\hat{u}}_n^p$  und  $\underline{\tilde{u}}_n^p$  aus den hier kalkulierten Sequenzen  $\Delta \underline{\hat{u}}_n$  und  $\Delta \underline{\tilde{u}}_n$  kann auf die in Gleichung 43 und 45 dargelegten Berechnungsvorschriften zurückgegriffen werden. Für die Vorhersage einer Sequenz an internen Modellunsicherheiten gilt:

$$[\Delta \underline{\hat{y}}_n(k+n_y), \dots, \Delta \underline{\hat{y}}_n(k)] = g_n\left(\Delta \underline{y}_n(k-1), \dots, \Delta \underline{y}_n(k-n_y), k\right)$$
(71)

Diese Vorhersagesequenz antizipiert die trotz der Integration der Anomalie-Vorhersage aus Gleichung 70 noch bestehende Ungenauigkeit der Vorhersage des Prozessausgangs. Sie beruht auf vergangenen Vorhersagefehlern. Durch das integrierende Verhalten dieses Modells kann der Fehler stetig reduziert werden. Alle Modelle verarbeiten dabei Eingabesequenzen unterschiedlicher Länge und erzeugen Ausgabesequenzen identischer Länge  $n_{\hat{v}}$ .



Abbildung 4.15: Vereinfachte Darstellung des kooperativen Prädiktionsschemas zur Vorhersage des Prozessausgangs (oben) unter Berücksichtigung anomaler Prozessdynamiken (unten)

Diese Verbindung zwischen den Modellen führt aufgrund der ganzheitlichen Integration aller relevanten Effekte zu einer verbesserten Vorhersagegenauigkeit mit reduziertem Fehler e(W, k)zwischen realem und vorhergesagtem Ausgang. Ohne die Berücksichtigung der Anomalien und deren Integration in die Modellierung würde es im Falle die Aktoren manipulierender, externer Störungen zu einem erhöhten Vorhersagefehler kommen. Durch die Nutzung von Seq2Seq-LSTM-Netzen kann zusätzlich die Fehlerfortpflanzung für größere Vorhersagehorizonte reduziert werden. Sollen hingegen weitere Vorhersagen auf Grundlage bereits durchgeführter Vorhersagen realisiert werden, kommt es auch hier zu einer Fehlerfortpflanzung. Die Vorhersagen für weiter in der Zukunft liegende Zeitsequenzen basieren zunehmend auf den Anomalie-Vorhersagen  $\Delta \hat{u}_n(k+n_n), ..., \Delta \underline{\hat{u}}_n(k)$  und weniger auf den in der Vergangenheit beobachteten Anomalien  $\Delta \underline{\tilde{u}}_n(k-1), \dots, \Delta \underline{\tilde{u}}_n(k-n_o)$ . Die Summe der Anzahl an vorhergesagten Anomalien  $n_p$  und der in der Vergangenheit beobachteten Anomalien  $n_o$  entspricht dem Horizont  $n_u$  [159]. Das in Abbildung 4.16 abgebildete Blockschaltbild zeigt das mit dem Anomalie-Detektor verknüpfte kooperative Prädiktionsschema. Dieses erhält, wie in Gleichung 70 dargelegt, die Beobachtungen vergangener Anomalien als Eingang für die Prädiktion zukünftig auftretender Anomalien. Diese werden wiederum mit der Prädiktion der Prozessdynamiken verknüpft und deren Vorhersagegenauigkeit durch stetiges Training weiter optimiert.



Abbildung 4.16: Verknüpfung von Anomalie-Detektor und kooperativem Prädiktionsschema zur Modellierung des Prozess- und Anomalie-Verhaltens

Das Prozessmodell berücksichtigt durch die Verknüpfung mit dem Anomalie-Modell die zu erwartenden Anomalien bei der Prädiktion der Prozessausgänge durch eine Verschiebung des Arbeitspunktes der Stellgrößen. Die dem Prädiktionsschema zu Grunde liegenden Modelle passen sich darüber hinaus adaptiv an die aktuelle Prozesssituation an [28]. Dieser iterative Lernprozess ist durch die Rückführung des Prädiktionsfehlers visualisiert. Insgesamt kann dadurch mit Hilfe des Schemas eine adaptive und lernfähige Prozessmodellierung unter Berücksichtigung anomaler Systemdynamiken erfolgen. Das Schema ist in der dargestellten Form auf alle Einzelprozesse einer Prozesskette übertragbar. Wie in Kapitel 4.3.1 eingeführt, sind die innerhalb der einzelnen Schemata vorhandenen Prozessmodelle zur Modellierung der entlang von Prozessketten auftretenden Abhängigkeiten miteinander verknüpft. Dadurch ergibt sich für die Modellierung ganzer Prozessketten ein aus multiplen Einzelschemata zusammengesetztes übergeordnetes Schema.

# 4.3.4 Kompensation anomaler Systemdynamiken

In den vorangegangenen Kapiteln wurde die Detektion von Anomalien sowie ein darauf aufbauender kooperativer Prädiktionsansatz zur Modellierung von Prozess- und Anomalie-Verhalten vorgestellt. Um die dadurch verbesserte Vorhersagegenauigkeit für eine Qualitätsregelung nutzbar zu machen, soll eine Kompensation von Anomalien erfolgen, die sowohl die externen Störungen als auch die Interdependenzen zwischen einzelnen Prozessen entlang der Prozesskette berücksichtigt. Es sollen Maßnahmen zur Kompensation des anomalen Verhaltens abgeleitet werden, die zur Folge haben, dass die Produktqualität entgegen der auftretenden Effekte im Referenzbereich gehalten werden kann. Dabei werden die Stellgrößen von Bauteil zu Bauteil angepasst, um die Störungen zu kompensieren und die Produktqualität zu optimieren. Das zur Erfüllung dieses Zwecks entwickelte Konzept wird in diesem Kapitel im Detail erläutert.

## 4.3.4.1 Rekursive Verknüpfung kooperativer Prädiktionen

Um das kooperative Prädiktionsschema für eine Qualitätsregelung nutzbar zu machen, wird die Grundidee der modellbasierten prädiktiven Regelung auf das Problem der Optimierung von Produktqualität entlang mehrstufiger Prozessketten projiziert. Ein MPC-Schema prognostiziert auf der Grundlage eines Prozessmodells zukünftige Prozessausgänge über einen definierten Horizont und optimiert diese Prozessausgänge im Hinblick auf eine Sollwerttrajektorie, um die optimalen zukünftigen Stellgrößen zu extrahieren. Das Problem wird unter Berücksichtigung von Nebenbedingungen, wie z.B. Stellgrößenbeschränkungen, gelöst und der Lösungsvektor an zukünftigen Stellgrößen, die zu einer minimalen Abweichung von der Sollwerttrajektorie führen, generiert. Es wird nur die erste Stellgröße auf den Prozess angewendet. Die Prozessantwort auf diese Stellgröße stellt die Grundlage für die nächste Iteration zur Vorhersage und Optimierung dar, die dann von Neuem begonnen wird. Somit resultiert der wesentliche Anreiz des MPC-Ansatzes aus der Optimierung zukünftigen Stellverhaltens. Wenn das Prozessmodell eine Modellierung von Anomalien beinhaltet, werden diese automatisch durch die optimalen Stellgrößen kompensiert [13] [159].

Die Projektion des MPC-Ansatzes auf den in Kapitel 4.2.2.1 beschriebenen gerichteten Graphen lässt sich als bauteilbezogenes Optimierungsproblem formulieren. Die zentrale Idee dabei ist, die prozessbezogene Vorhersage der klassischen MPC als eine prozessübergreifende Vorhersage neu zu formulieren. Der Vorhersagehorizont bezieht sich nicht mehr auf die zu diskreten Zeitschritten erzeugten Prozessausgänge eines Prozesses, sondern auf die zu diskreten Zeitschritten erzeugten Prozessausgänge von in der Prozesskette aufeinanderfolgender Prozesse. Die Vorhersagetrajektorie beinhaltet somit eine Zuordnung von diskretem Zeitschritt zum zu diesem Zeitschritt betrachteten Prozess. Einfach formuliert, bezieht sich die erste Vorhersage auf den Ausgang des Prozesses, an dem sich das betrachtete Bauteil aktuell befindet, die zweite Vorhersage auf den nachfolgenden Prozess, an dem sich das Bauteil zum nächsten Zeitschritt befinden wird, die dritte Vorhersage auf den wiederum darauffolgenden Prozess usw. Gemäß der Projektion des MPC-Ansatzes werden zukünftige Prozessausgänge noch ausstehender Prozessschritte auf der Grundlage der zugehörigen Prozessmodelle für ein bestimmtes Bauteil vorhergesagt und in die Optimierung der Qualität für dieses Bauteil einbezogen. Das bedeutet, dass für jeden noch ausstehenden Prozess der Prozesskette in einem sequentiellen Ablauf das Eintreffen und Prozessieren des betrachteten Bauteils modelliert und der entsprechende Prozessausgang prädiziert wird. Aufgrund der Tatsache, dass die Prozesse durch die Eigenschaften der Bauteile miteinander verknüpft sind, dient der zu einem bestimmten Zeitschritt vorhergesagte Ausgang eines Prozesses als Eingang für den nachfolgenden Prozess zum nächsten Zeitschritt. Die Vorhersagen der Prozessausgänge werden mit dem kooperativen Prädiktionsschema realisiert. Die kooperative Prädiktion ist nicht auf einen einzelnen Prozess beschränkt, sondern wird für alle Prozesse in jeder Iteration genutzt. Es werden für alle Prozesse entlang der Prozesskette bei jeder Iteration Vorhersagen über definierte Horizonte auf der Grundlage der jeweiligen Prozessmodelle durchgeführt, sodass die erörterte bauteilspezifische Betrachtung der prädizierten Prozessausgänge möglich ist. Die entstehende Trajektorie an Vorhersagen gilt für das betrachtete Bauteil und dient als Grundlage zur Qualitätsoptimierung desselben. Ein wesentlicher Unterschied zu einer regulären MPC ist der Umstand, dass die Vorhersage eines bestimmten Prozesses von der Vorhersage vorangegangener Prozesse abhängig ist. Das Verhalten eines bestimmten Prozesses kann nicht vorhergesagt werden, ohne die vorangegangenen Prozesse vorhergesagt zu haben. Die erste Vorhersage basiert auf vergangenen Werten und kann durchgeführt werden. Die Vorhersage für einen weiteren Schritt in die Zukunft basiert hingegen auf der ersten Vorhersage des vorangegangenen Prozessschrittes. Die Vorhersage für einen dritten Schritt in die Zukunft basiert auf der zweiten Vorhersage des vorangegangenen Prozessschrittes, die wiederum auf der ersten Vorhersage des Schrittes davor basiert usw. Für jeden weiteren Prädiktionsschritt eines Prozesses werden aufgrund der Verknüpfung der Prozesse über die Bauteileigenschaften weitere Vorhersagen vorangegangener Prozessschritte benötigt. Einfach formuliert, kann keine prozessspezifische Vorhersage über die Verarbeitung von Bauteilen durchgeführt werden, wenn für die vorherigen Prozesse nicht bestimmt wurde, wie sich die Bauteile dort verhalten haben. Auf der anderen Seite kann genauso keine bauteilspezifische Vorhersage bezüglich aller für dieses Teil noch ausstehenden Prozesse durchgeführt werden, ohne alle nachfolgenden Prozesse auf der Grundlage von Bauteilen vorherzusagen, die dort verarbeitet wurden bzw. werden, bevor das betrachtete Bauteil eintrifft. Zusammenfassend lässt sich schlussfolgern, dass die kooperativen Prädiktionen aller Prozessschritte voneinander abhängig sind und das Produktionsgeschehen innerhalb der Prozesskette nachbilden. Daher ermöglicht das korporative Prädiktionsschema eine bauteilbezogene Optimierung zukünftiger Prozessausgänge. Das Optimierungsproblem kann aufgrund der kaskadierten Abhängigkeitsstruktur der Stellgrößen als rekursives Problem formuliert werden. Die Zielfunktion lässt sich dabei wie folgt darstellen:

$$\bigwedge_{k} \bigwedge_{p=N}^{1} \left[ \underline{u}_{p}^{*}(k), \dots, \underline{u}_{N}^{*}(k+N-p) \right] \\
= \min_{\underline{u}_{n}(\cdot)} \sum_{n=p}^{N} \varphi_{k}(n,p) * q_{k}(p) * \left( \underline{\bar{y}}_{n}(k+n-p) - \underline{\hat{y}}_{n}(k+n-p) \right)^{2} + r_{k}(p) * \left( \underline{u}_{n}(k+n-p) - \underline{u}_{n}^{*}(k+n-p-1) \right)^{2}$$
(72)

wobei durch  $\varphi_k(n, p)$  eine Gewichtung der auftretenden Abweichung von der Solltrajektorie vorgenommen wird, die bei einem zukünftigen Prozessschritt *n*, ausgehend vom aktuellen, zum Zeitpunkt *k* durchgeführten Prozess *p*, auftritt. Die Vorhersage entlang der Prozesskette zum Zeitpunkt *k* ausgehend von Prozess *p* entspricht derjenigen für das zu diesem Zeitpunkt dort verarbeitete Bauteil. Die Gewichte der Funktion erhöhen sich mit größeren Werten von *n*, da die Qualität des Bauteils am Ende der Prozesskette optimal sein soll. Entlang der für dieses Bauteil prädizierten Trajektorie auftretende Abweichungen von der Sollwerttrajektorie sollen im Rahmen

der Optimierung der Stellgrößen über die Prozesskette hinweg ausgeglichen werden, um eine optimale Qualität des Endprodukts zu gewährleisten. Die Ausdrücke  $q_k(p)$  und  $r_k(p)$  nehmen eine Gewichtung zwischen einer Minimierung des Ausgabefehlers gegenüber dem dafür notwendigen Stellaufwand vor. Die universellen Quantifikatoren in Gleichung 72 charakterisieren den Umstand, dass die Formulierung der Minimierungsfunktion rekursiv, mit demjenigen am letzten Prozessschritt befindlichen Bauteil beginnt und mit dem zuletzt in die Prozesskette eingetretenen Bauteil endet. Das liegt an der bereits erläuterten Abhängigkeit einer bauteilspezifischen Vorhersage von den Vorhersagen der noch ausstehenden Prozessschritte hinsichtlich der Bauteile, die hier noch vor dem betrachteten Bauteil verarbeitet werden. Der zweite Summand berücksichtigt folglich die optimale Lösung  $\underline{u}_n^*(\cdot)$  der dem betrachteten Prozess in der Prozesskette nachfolgenden Prozesse. Die Optimierung der Stellgrößen hinsichtlich der Zielfunktion erfolgt unter Nebenbedingungen. Um den Stellaufwand beim jeweils ersten Zeitschritt für jeden Prozess bestimmen zu können, muss die reale, zuletzt aufgebrachte Stellgröße verwendet werden, die wiederum der optimalen Lösung  $\underline{u}_n^*(\cdot)$  bei der letzten Iteration entspricht [159]. Daher gilt:

$$\bigwedge_{n=1}^{N} \underline{u}_{n}^{*}(k-1) = \underline{u}_{n,real}(k-1)$$
(73)

Darüber hinaus gelten für alle Stellgrößen Beschränkungen, die bei der Optimierung berücksichtigt werden müssen. Diese beziehen sich einerseits auf die absoluten Werte der Stellgrößen für jeden noch ausstehenden Prozess von p bis N, wobei N die Gesamtzahl an Prozessen der Prozesskette markiert:

$$\bigwedge_{p=1}^{N}\bigwedge_{n=p}^{N}\underline{u}_{n,min} = \underline{u}_{n}(k+n-p) = \underline{u}_{n,max}$$
(74)

Für die inkrementellen Änderungen der Stellgrößen gelten ebenfalls Beschränkungen. Die Änderungen werden durch die optimale Lösung des am nachfolgenden Prozessschritt befindlichen Bauteils bestimmt:

$$\bigwedge_{p=1}^{N} \bigwedge_{n=p}^{N} \Delta \underline{u}_{n,min} = \underline{u}_n(k+n-p) - \underline{u}_n^*(k+n-p-1) = \Delta \underline{u}_{n,max}$$
(75)

Die Nebenbedingung impliziert, dass die Lösung des Optimierungsproblems durch eine aufeinander aufbauende Lösung von Teilproblemen für die einzelnen Bauteile erfolgen muss. Dies muss den Quantoren der Zielfunktion entsprechend erfolgen, um für alle Stellgrößen die zuvor erzielten optimalen Lösungen bereitstellen zu können. Eine detaillierte Lösung des Optimierungsproblems wird im nächsten Kapitel beleuchtet. Abschließend wird das Prozessmodell aufgezeigt, auf dessen Grundlage die Vorhersagen der Prozessausgänge  $\hat{y}_n(k + n - p)$  für die Zielfunktion erzeugt werden und das Optimierungsproblem gelöst wird. Dieses entspricht dem für die kooperative Prädiktion hergeleiteten Prozessmodell, bei dem die Verschiebung des Zeitstempels k + n - p berücksichtigt wird. Es folgt:

$$\begin{split} [\hat{\underline{y}}_{n}(k+n-p+n_{\hat{y}}), \dots, \hat{\underline{y}}_{n}(k+n-p)] \\ &= f_{n}\left(\underline{y}_{n}(k+n-p-1), \dots, \underline{y}_{n}(k+n-p-n_{y}), \\ \dots \hat{\underline{u}}_{n}^{p}(k+n-p+n_{\hat{y}}), \dots, \hat{\underline{u}}_{n}^{p}(k+n-p), \\ \dots \underline{\tilde{u}}_{n}^{p}(k+n-p-1), \dots, \underline{\tilde{u}}_{n}^{p}(k+n-p-n_{u}), \underline{w}_{n}(k+n-p), k \right) \\ &+ [\Delta \hat{\underline{y}}_{n}(k+n-p+n_{\hat{y}}), \dots, \Delta \hat{\underline{y}}_{n}(k+n-p)] \end{split}$$
(76)

Die rekursive Formulierung des in diesem Abschnitt beschriebenen Optimierungsproblems ermöglicht dessen Zerlegung in Teilprobleme. Das zu deren Lösung entwickelte Vorgehen erzeugt die zur Kompensation von Anomalien benötigten Stellgrößen. Im folgenden Abschnitt werden diese Thematiken detailliert beleuchtet.

#### 4.3.4.2 Anpassung der Stellgrößen zur Kompensation von Anomalien

Das Optimierungsproblem, bestehend aus der zu minimierenden Zielfunktion, den für die Stellgrößen zu berücksichtigenden Nebenbedingungen und dem die Ausgänge erzeugenden Prozessmodell, kann durch eine separate Betrachtung der einzelnen bauteilspezifischen Teilprobleme rekursiv gelöst werden. Die Zielfunktion in Gl. 72 wird dazu zum Zeitpunkt *k* für jeden Prozess der Prozesskette *p* bzw. das zu diesem Zeitpunkt dort vorhandene Bauteil aufgestellt. Die dadurch gemäß der Quantoren zum Zeitpunkt *k* erzeugten *N* Zielfunktionen werden beginnend mit dem letzten Prozesschritt *N* und endend mit dem ersten Prozessschritt 1 unter jeweiliger Berücksichtigung von Nebenbedingungen und Prozessmodell minimiert. Diese rekursive Lösung der aufeinander aufbauenden Teilprobleme ermöglicht die Lösung des übergeordneten Gesamtproblems zu jedem Zeitpunkt *k*. Das Gesamtproblem wird zu jedem diskreten Zeitschritt *k* gelöst. Die Zerlegung der Zielfunktion wird nachfolgend anhand der ersten beiden zu minimierenden Zielfunktionen für Prozess *N* bzw. das dort vorhandene Bauteil sowie N - 1 erläutert. Zunächst wird die Zielfunktion für das am letzten Prozessschritt befindliche Bauteil bestimmt zu:

$$\begin{bmatrix} \underline{\boldsymbol{u}}_{N}^{*}(\boldsymbol{k}) \end{bmatrix} = \min_{\underline{\boldsymbol{u}}_{n}(\cdot)} \left( \varphi_{k}(N,N) * q_{k}(N) * \left( \underline{\bar{y}}_{N}(k) - \underline{\hat{y}}_{N}(k) \right)^{2} + r_{k}(N) \\ * \left( \underline{\boldsymbol{u}}_{N}(k) - \underline{\boldsymbol{u}}_{N}^{*}(k-1) \right)^{2} \right)$$
(77)

Wenn der aktuelle Prozess p, an dem sich das betrachtete Bauteil befindet, der letzte Prozess N ist, wird lediglich eine Vorhersage für diesen Prozess für den Zeitschritt k durchgeführt. Für das nächste, am vorletzten Prozess N - 1 befindliche Bauteil wird hingegen eine Vorhersage für

diesen und den letzten Prozess für die Zeitschritte k und k + 1 durchgeführt. Dies lässt sich bis zum Beginn der Prozesskette fortführen, wobei für jeden Prozess p bzw. das dort zum Zeitpunkt k vorhandene Bauteil Vorhersagen über die noch ausstehenden Prozesse p bis N für die Zeitschritte k bis k + N - p, an denen das Bauteil dort ist, durchgeführt werden. Die Lösung von p = N führt zur optimalen Stellgröße  $\underline{u}_N^*(k)$  für Prozess N zum Zeitpunkt k und erzeugt dort den Ausgang  $\underline{\hat{y}}_N^*(k)$ . Die generierte Stellgröße wird für die Zielfunktion und der Ausgang für das Prozessmodell von p = N - 1 benötigt und damit zur Lösung des Teilproblems des nächsten, im Prozess befindlichen Bauteils verwendet. Diese rekursive Verknüpfung über die Stellgröße zwischen p = N und p = N - 1 ist in den Gleichungen 77 und 78 hervorgehoben. Die entsprechende Lösung von p = N - 1 führt zu den optimalen Stellgrößen  $[\underline{u}_{N-1}^*(k), \underline{u}_N^*(k+1)]$ für die Prozesse N - 1 und N zu den Zeitpunkten k sowie k + 1 und erzeugt dort die Ausgänge  $[\underline{\hat{y}}_{N-1}^*(k), \underline{\hat{y}}_N^*(k+1)]$ . Diese werden wiederum zur Lösung des sich anschließenden Teilproblems benötigt. Für p = N - 1 gilt entsprechend die folgende Zielfunktion:

$$\begin{aligned} \left[\underline{u}_{N-1}^{*}(k), \underline{u}_{N}^{*}(k+1)\right] \\ &= \min_{\underline{u}_{N}(\cdot)} \left(\varphi_{k}(N-1, N-1) * q_{k}(N-1) * \left(\underline{\bar{y}}_{N-1}(k) - \underline{\hat{y}}_{N-1}(k)\right)^{2} \\ &+ r_{k}(N-1) * \left(\underline{u}_{N-1}(k) - \underline{u}_{N-1}^{*}(k-1)\right)^{2} + \varphi_{k+1}(N, N-1) \\ &\quad * q_{k+1}(N-1) * \left(\underline{\bar{y}}_{N}(k+1) - \underline{\hat{y}}_{N}(k+1)\right)^{2} + r_{k+1}(N-1) \\ &\quad * \left(\underline{u}_{N}(k+1) - \underline{u}_{N}^{*}(k)\right)^{2} \end{aligned} \end{aligned}$$
(78)

Für das zum Zeitpunkt k am Prozess N - 1 befindliche Bauteil ist dies die zu realisierende Stellstrategie hinsichtlich der beiden noch ausstehenden Prozesse. Der finale Ausgang  $\hat{y}_N^*(k + 1)$ sollte dabei der optimalen Produktqualität entsprechen. Bei der nächsten zum Zeitpunkt k + 1durchgeführten Iteration erfolgt eine Neuberechnung der optimalen Stellstrategie für dieses Bauteil, die dann lediglich noch den letzten Prozesschritt betrifft [159].

Abbildung 4.17 zeigt eine Visualisierung des Vorgehens zur Zerlegung der übergeordneten Zielfunktion in einzelne Zielfunktionen bzw. des Optimierungsproblems in einzelne Teilprobleme. Der rekursive Lösungsansatz und die bereits erwähnten Abhängigkeiten von Stellgrößen und Prozessausgängen werden im Folgenden näher beleuchtet. Entlang der Prozesskette sind die Prozesse 1,2, ..., *N* dargestellt. Das Verfahren beginnt wie zuvor beschrieben mit dem Bauteil bei Prozess *N*. Für diese Optimierung wird die zum vorherigen Zeitschritt tatsächlich aufgebrachte Stellgröße benötigt, um den Stellaufwand für die Zielfunktion bestimmen zu können. Darüber hinaus werden alle vergangenen Prozessausgänge und Stellgrößen benötigt, um basierend auf dem Prozessmodell den nächsten Prozessausgang prädizieren zu können. Das Ergebnis der Stellgrößenoptimierung sowie die sich durch Einsetzen in das Prozessmodell
ergebende Vorhersage des Prozessausgangs werden für die Lösung des nächsten Teilproblems benötigt. Dieses bezieht sich auf das Bauteil, das sich gerade bei Prozess 3 befindet. Der Prozess N kann für den Zeitschritt k + 1 nicht vorhergesagt und in die Optimierung einbezogen werden, wenn die vorherigen optimalen Stellgrößen und die zugehörigen Vorhersagen der Prozessausgänge fehlen, da diese wieder im Prozessmodell und zur Bestimmung des Stellaufwands benötigt werden. Daher wird das Ergebnis des ersten Teilproblems für die Lösung des zweiten Teilproblems als Eingabe des Prozessmodells von Prozess N zum Zeitpunkt k + 1verwendet. Dann kann die zweite Teiloptimierung entlang der beiden letzten Prozesse stattfinden. Im Allgemeinen wartet das betrachtete Bauteile auf die optimale Lösung, die für das beim nachfolgenden Prozess befindliche Bauteil generiert wird und berechnet auf dieser Grundlage seine eigene optimale Stellstrategie entlang der noch ausstehenden Prozesse. Die Optimierung zur Erzeugung der Stellstrategie für dieses Bauteil findet in der Visualisierung entlang der Diagonalen von Prozess 3 zum Zeitpunkt k und Prozess N zum Zeitpunkt k + 1 statt. Einfach formuliert werden die Ergebnisse an die nächste "Diagonale" für die dritte Optimierung weitergeleitet, die das sich zum Zeitpunkt k bei Prozess 2 befindliche Bauteil betreffen. Die Vorhersage und Optimierung für dieses Bauteil über die noch anstehenden Prozesse 2, 3 und N führt zu dessen Stellstrategie  $\left[\underline{u}_{2}^{*}(k), \underline{u}_{3}^{*}(k+1), \underline{u}_{N}^{*}(k+2)\right]$ .



Abbildung 4.17: Veranschaulichung des rekursiven Verfahrens zur Bestimmung der Stellgrößen für alle Prozesse der Prozesskette

Diese sowie die Vorhersagen der Prozessausgänge, die durch die Stellstrategie entstehen würden, bilden die Basis für die letzte in der Abbildung dargestellte Teiloptimierung. Das übergeordnete Optimierungsproblem wird somit rekursiv gelöst. Die sich beim letzten Schritt ergebende Stellstrategie gilt dem Bauteil, das neu in die Prozesskette eingetreten ist. Sie wird unter Berücksichtigung aller bereits zuvor gelösten Teilprobleme und damit mit der größten Unsicherheit generiert. Die Stellstrategien der am Ende der Prozesskette befindlichen Bauteile sind hingegen mit der geringsten Unsicherheit behaftet, da sie weniger auf Vorhersagen als vielmehr auf vergangenen tatsächlichen Prozessausgängen und Stellgrößen basieren. So werden für jedes Bauteil in Abhängigkeit von dem Prozessschritt p, bei dem es sich aktuell befindet, Vorhersagen für die Prozesse von p bis N durchgeführt. Die Stellgrößenbeschränkungen müssen entlang aller ,Diagonalen' eingehalten werden. In Anlehnung an die klassische MPC werden die aus dem Optimierungsproblem hervorgehenden, bauteilspezifischen Stellstrategien für alle zukünftigen Prozesschritt lediglich für den aktuellen Prozesschritt umgesetzt und zum nächsten diskreten Zeitschritt neu berechnet. Es werden daher lediglich die ersten Stellgrößen der jeweiligen Stellstrategien umgesetzt. Dies resultiert in der folgenden, zur Kompensation von Anomalien benötigten Stellgrößenmatrix für den Zeitpunkt k:

$$U^{*}(k) = \left[\underline{u}_{1}^{*}(k), \underline{u}_{2}^{*}(k), \dots, \underline{u}_{N-1}^{*}(k), \underline{u}_{N}^{*}(k)\right]$$
(79)

Bei der nächsten Iteration zum Zeitpunkt k + 1 werden die Stellstrategien auf Basis der auf die *N* Prozesse angewandten, optimalen Stellgrößen sowie der dabei tatsächlich erzeugten Prozessausgänge von Neuem berechnet.

Die Innovation liegt hier weniger im konkreten Optimierungsverfahren als vielmehr in der neuartigen Beschreibung des Optimierungsproblems. Das konkrete Lösungsverfahren, das in die beschriebene Vorgehensweise eingebettet ist und zur Lösung der einzelnen Teilprobleme eingesetzt wird, steht nicht im Fokus dieser Arbeit. Da die Teilprobleme zu jedem diskreten Zeitpunkt gelöst werden und zu diesem Zeitpunkt alle Eingänge des Prozessmodells fest definiert sind, reduziert sich dieses zu einer zeitinvarianten sowie statischen Abbildung der zu optimierenden Stellgrößen auf die aktuellen Prozessausgänge. Da es sich dadurch bei den einzelnen Zielfunktionen um affine Funktionen handelt und die einzelnen Teilprobleme konvex sind, eignen sich nichtlineare Optimierungsverfahren zur Lösung derselben [165]. Es können dazu verschiedene Verfahren verwendet werden, deren Lösungsgenauigkeit und Performance sich hinsichtlich der dieser Arbeit zu Grunde liegenden Anforderungen nicht bedeutend unterscheiden. Beispiele dafür wären der Levenberg-Marquardt-Algorithmus oder das Nelder-Mead-Simplex-Verfahren. In dieser Arbeit wird aufgrund der guten Konvergenzeigenschaften und der Robustheit in der Lösungsfindung der Levenberg-Marquardt-Algorithmus genutzt [166]. Die beschriebene Methodik zur Erzeugung der optimalen Stellgrößen wird in der in Abbildung 4.18 dargestellten Konstellation eingesetzt. Das Konstrukt ähnelt dem in [13] vorgestellten Schema hinsichtlich der Verknüpfung von Beobachter und Prädiktor, unterscheidet sich jedoch wesentlich von diesem durch die Ausgestaltung dieser Komponenten. Das vereinfachte Blockdiagramm stellt eine Erweiterung der in den Abbildungen 4.7, 4.8 bzw. 4.16 eingeführten Diagramme dar und verknüpft den Anomalie-Detektor, die kooperative Prädiktion sowie die darauf aufbauende Kompensation von Anomalien.



Abbildung 4.18: Blockschaltbild zur Kompensation von Anomalien und Einhaltung der Produktqualität auf Basis optimierter Stellgrößen

Der Adaptionsblock beinhaltet die erläuterte bauteilbezogene Optimierung zur Erzeugung der Kompensationssignale in Form von optimierten Stellgrößen. Die vom Werker an der Anlage vorgenommenen Einstellungen werden durch Aufschaltung der berechneten optimalen Stellgrößen korrigiert. Die Referenz wird anhand der Qualitätsanforderungen der Kunden festgelegt. Da die Abhängigkeiten zwischen den Prozessschritten, interne Modellunsicherheiten sowie externe Störeffekte in die Modellierung einbezogen werden, kompensieren die erzeugten Stellgrößen diese Einflüsse. Der Umstand beruht darauf, dass das um diese Einflüsse erweiterte Prozessmodell die Grundlage der Optimierung ist und die Stellgrößen entsprechend so gewählt werden, dass die Sollwerttrajektorie trotz des Auftretens der Einflüsse eingehalten werden kann. Die optimierten Stellgrößen kompensieren daher automatisch die im Rahmen der Modellierung berücksichtigten und vom Anomalie-Detektor erfassbaren Anomalien stationärer und nichtstationärer Natur. Wie bereits in Abschnitt 4.1 erwähnt, werden in der industriellen Praxis häufig sowohl Prozessstabilität als auch Produktgualität durch den Produktionsmitarbeiter und sein Erfahrungswissen gewährleistet, der in diesem Fall als bestehender ,diskreter Regler' direkt an der Anlage fungiert. Die Kompensationssignale werden diesem Regler als Korrektur aufgeschaltet ohne dessen ursprüngliches Verhalten anpassen zu müssen. Dies geschieht für jeden Prozess der Prozesskette und der zugehörigen, von den bestehenden Reglern ausgegebenen Stellgrößen. Das dargestellte Schema zur Kompensation von Anomalien unterstützt daher auch den übergeordneten Planer bei der Optimierung der Produktqualität entlang der gesamten Prozesskette.

## 5 Realisierung

Nachdem das entwickelte Konzept im vorangegangenen Abschnitt im Detail erläutert wurde, wird im Rahmen dieses Kapitels dessen Realisierung in Form eines datengetriebenen Assistenzsystems vorgestellt. Um das Konzept nutzbar zu machen und gewinnbringend in der Produktion einsetzen zu können, wird zunächst eine verlässliche Datenerfassung benötigt, die eine eindeutige Zuordnung von Prozess- und Anlagendaten zu den jeweils produzierten Bauteilen vornimmt. Ohne diese semantische Annotation von Rohdaten ist das bauteilbezogene Konzept nicht realisierbar. Weiterhin muss eine Verdichtung der während des Prozessgeschehens erfassten Datenkurven erfolgen, um im Kontext der Stellgrößenadaption die Sollwerte dieser Kurven für jede weitere Durchführung neu anpassen zu können. Zur Verwaltung der großen, entlang mehrstufiger Prozessketten entstehenden Datenmengen wird ein Datenbank- oder Dateisystem benötigt. Die verdichteten Daten müssen darüber flexibel für weitere Verarbeitungsschritte bereitgestellt werden können. Die Implementierung der Kernelemente des Konzeptes, die die Detektion von Anomalien, die kooperative Prädiktion auf Basis des Prozessmodells sowie die Kompensation von Anomalien zur Qualitätsoptimierung umfassen, bauen auf der Datengrundlage auf. Um ein Vortraining mit historischen Daten und ein kontinuierliches Weitertraining während des Betriebes zu ermöglichen, werden Schnittstellen zur Datenverwaltung sowie entsprechende Trainingsmechanismen implementiert. Das System soll dem Nutzer über eine graphische Benutzeroberfläche (GUI) zugänglich gemacht werden. Die wesentliche Zielgruppe sind in erster Instanz die direkt an der Anlage operierenden Produktionsmitarbeiter. Darüber hinaus sollen die Resultate den Produktionsplanern sowie den Prozessingenieuren zur Verfügung gestellt werden. Die GUI muss daher den Anforderungen dieser unterschiedlichen Nutzergruppen gerecht werden und innerhalb eines Betriebes flexibel abrufbar sein

Eine weitere Herausforderung besteht in der Systemintegration, d.h. der Umsetzung des Konzeptes in Form einer funktionsfähigen und performanten Private-Cloud-Architektur. Sie geht einerseits auf das komplexe Muster an Informationsflüssen entlang mehrdimensionaler Produktionsketten und die damit einhergehenden zahlreichen Datenströme zurück. Dazu ist eine Umsetzung von entsprechenden Kommunikationsschnittstellen sowie eines effizienten Mechanismus zur Orchestrierung und Verarbeitung der transferierten Daten notwendig. Andererseits besteht die Herausforderung in der Integration einer Plattform bzw. eines Frameworks zur Speicherung und Verwaltung der großen Mengen an erfassten Prozess- und Anlagendaten. Plattformen wie bspw. Hadoop zeichnen sich durch eine hohe Komplexität aus, sodass die bedarfsgerechte Einrichtung solcher Systeme mit hohem Initialaufwand verbunden ist. Eine weitere Problematik ist die Datensicherheit, die bei sensiblen Prozessdaten gewährleistet sein muss. Authentifizierungsdienste oder Ticketing-Systeme müssen passgenau in das Gesamtsystem integriert und mit verschiedenen weiteren Diensten, wie der Benutzerverwaltung, verknüpft werden, um einen sicheren Schutz vor unbefugtem Zugriff zu bieten. Auch hier ist für die Einrichtung ein hoher Initialaufwand vorhanden. Das weitere Kapitel ist wie folgt aufgebaut:

zunächst wird zur Einführung ein Überblick über die Systemarchitektur des Gesamtsystems gegeben und die Interaktion zwischen den einzelnen Komponenten des Systems beschrieben. Darauf aufbauend wird die entwickelte Komponente zur Datenerfassung erläutert. Daran schließt sich eine Beschreibung der Implementierung der drei Kernelemente des Konzeptes an, bevor auf die entwickelte GUI und ihre verschiedenen Ansichten eingegangen wird.

## 5.1 Überblick über das Gesamtsystem

Das datengetriebene Assistenzsystem zur adaptiven Qualitätsregelung setzt sich aus den nach [167] drei wesentlichen Komponenten von KI-Anwendungen zusammen: dem Wahrnehmen, dem Lernen bzw. Verstehen sowie dem Handeln. Projiziert auf den Kontext dieser Arbeit entspricht das Wahrnehmen dem Erfassen und Annotieren von Daten, das Verstehen der Charakterisierung von anomalen Prozessdynamiken sowie deren kooperative Prädiktion und das Handeln der Kompensation dieser Anomalien zur Optimierung der Produktqualität. Abbildung 5.1 zeigt einen Überblick über die realisierte Systemarchitektur und die einzelnen Komponenten, die in mehreren studentischen Arbeiten realisiert und getestet werden konnten [162, 169, 174-177, 180-182]. Die dem Wahrnehmen, dem Verstehen und dem Handeln zugeordneten Komponenten werden durch die verschiedenen Blautöne markiert.



Abbildung 5.1: Systemarchitektur des entwickelten datengetriebenen Assistenzsystems zur adaptiven Qualitätsregelung

Das Wahrnehmen beginnt mit der Erfassung der Prozess- und Anlagendaten über einen Konnektor, der die Struktur der heterogenen Datenquellen auf Sensor- und Steuerungsebene kapselt und die akquirierten Daten in einem einheitlichen Format weitergibt. Die homogenisierten Daten werden auf Basis der Software-Komponente zur eindeutigen Rückverfolgung aller Bauteile annotiert. Diese beinhaltet das Modell der Transportlogik für die Prozesskette in Form des in Kapitel 4.2.2.1 vorgestellten gerichteten Graphen. Im Anschluss werden die annotierten Datenkurven mit Hilfe eines ETL-Stacks (Extract-Transform-Load) verdichtet und in einer multidimensionalen Echtzeitdatenbank, die zum Online Analytical Processing (OLAP) geeignet ist, bereitgestellt. Für Langzeitanalysen auf großen Datenmengen wurde ein Hadoop Distributed File System (HDFS) mit zwei Knoten aufgesetzt, das eine verteilte, redundante Speicherung und durch die Anwendung des MapReduce-Algorithmus einen effizienten Zugriff auf die Daten erlaubt. Nach dem Wahrnehmen folgt das Verstehen der Daten. Das geschieht durch das Erzeugen eines Verständnisses für die anomalen Systemdynamiken. Zwecks der Implementierung des Anomalie-Detektors sowie der Prozessmodelle wird auf Zeppelin als Bestandteil des Hadoop-Ökosystems zurückgegriffen. Dadurch wird eine Integration weiterer Services deutlich erleichtert. Nachdem die Anomalien genauer charakterisiert wurden, erfolgt auf Basis des neu gewonnenen Wissens eine Handlung zur Verbesserung der Produktqualität. Diese Handlung wird mit Hilfe des Schemas zur Kompensation von Anomalien, das auf einer rekursiven Optimierung der Stellgrößen basiert, durchgeführt. Die Komponente ist eng an die GUI gekoppelt, da dem Nutzer die mit jedem diskreten Zeitschritt neu berechneten Informationen in Form von Handlungs- bzw. Anpassungsempfehlungen zur Verfügung gestellt werden. Darüber hinaus bestehen Schnittstellen zu den Anlagensteuerungen der einzelnen Prozessschritte, um die zur Kompensation berechneten Stellstrategien automatisiert oder nach Bestätigung durch den Werker umsetzen zu können.

## 5.2 Komponente zur Datenerfassung

Im Folgenden wird genauer auf die Software-Komponente zur Erfassung der Daten eingegangen. In diesem Zuge wird zunächst die Subkomponente zur eindeutigen Bauteilrückverfolgung erörtert, bevor auf die Implementierung der Datenverwaltung eingegangen wird.

## 5.2.1 Subkomponente Bauteilrückverfolgung und Datenverarbeitung

In [124] wird die Fähigkeit, hochpräzise, einheitliche und semantisch annotierte Prozessdaten aus heterogenen Datenquellen zu erfassen, als zentrale Herausforderung bei der industriellen Umsetzung datengetriebener Assistenzsysteme beschrieben. Die im Rahmen dieser Arbeit entwickelte Konnektor-Komponente ermöglicht es, Daten aus verschiedenen Datenquellen zu extrahieren, zu homogenisieren und eine eindeutige Zuordnung der Daten zum produzierten Bauteil herzustellen. Die Komponente besteht aus einer individuellen Schnittstelle für jede Datenquelle und einer standardisierten Schnittstelle zur einheitlichen Weiterverarbeitung der Daten. Die Umsetzung orientiert sich an den Ergebnissen von [168]. Die individuelle Schnittstelle

läuft auf den speicherprogrammierbaren Steuerungen (SPS) der Anlagen und ist als Mealy-Automat im strukturierten Text implementiert, der zyklisch aufgerufen wird, die Protokollstruktur scannt und die Prozessdaten von nicht benötigten Metadaten trennt. Dabei wird mit verschiedenen Protokollen wie EtherCAT oder Profibus sowie Steuerungen wie TwinCAT 3 gearbeitet.



Abbildung 5.2: Komponente zur Vereinheitlichung heterogener Daten (links) und semantische Annotation zur eindeutigen Bauteilrückverfolgung (rechts)

Die Prozessdaten werden in einer einheitlichen Frequenz abgetastet, mit einem Zeitstempel versehen und über einen REST Post (Representational State Transfer) an einen Server übermittelt. Der Konnektor zur Homogenisierung von Daten ist in Abbildung 5.2 visualisiert. Die zum Server transferierten Daten werden der logistischen Ablauflogik entsprechend den Bauteilen zugeordnet, bei deren Produktion sie angefallen sind. Die Ablauflogik beinhaltet die Informationen des in Kapitel 4.2.2.1 eingeführten graphentheoretischen Modellierungsansatzes, der die gesamte Prozesskette abbildet. Sie wird in einem speziellen, auf die Modellierung von Graphen ausgerichteten XML-Format kodiert. Ein Python-Skript nimmt die Annotation der per REST Post eingehenden Daten auf Basis der XML-Datei vor. Spezielle Vorkommnisse wie Bedienungsfehler des Produktionsmitarbeiters oder die Entstehung von Ausschussteilen werden bei der Zuordnung berücksichtigt. Die annotierten Daten werden im Anschluss in einen niederdimensionalen Unterraum projiziert. Diese Verdichtung der Daten dient der Erzeugung prozessablaufspezifischer Merkmale, die die individuellen Charakteristiken der jeweils stattgefundenen Prozessdurchführung kapseln. Dazu wird ein ETL-Stack implementiert, dessen Kernelement entweder einer DWT oder einer Hauptkomponentenanalyse entspricht, die zur Berechnung der Merkmale eingesetzt werden. Diese dienen wiederum als Analysegrundlage zur Ableitung neuer Sollwerte für die Einstellung der Kurvenverläufe der als nächstes zu produzierenden Bauteile. Die homogenisierten, annotierten und verdichteten Daten werden in einer multidimensionalen Echtzeitdatenbank abgelegt, sodass der Zugriff schnellstmöglich und während des Betriebs erfolgen kann. Sie werden in Dimensionen mit einer internen hierarchischen Struktur organisiert, die eine aggregierte Betrachtung auf verschiedenen Abstraktionsebenen erlaubt. So kann von einzelnen Bauteilen zu übergeordneten Chargen bis hin zu ganzen Aufträgen navigiert und die zugehörige Repräsentanz der Prozess- und Anlagendaten abgefragt werden. In [24] werden dazu weitere Ausführungen betrachtet. Für den Fall der in Abbildung 5.2 dargestellten drei Dimensionen Ort, Zeit und Produktion lässt sich ein multidimensionaler Datenwürfel visualisieren. Als Software wird die Jedox Suite verwendet, die für die Berechnung der Merkmale einen Datenaustausch zwischen Python-Code und ETL-Stack erlaubt.

## 5.2.2 Subkomponente Datenverwaltung und -sicherheit

Neben dem unmittelbaren Zugriff auf die Daten für ein echtzeitfähiges Weitertraining der Modelle mit jedem neu verarbeiteten Bauteil werden die Daten zusätzlich für weiterführende Langzeitanalysen in einem HDFS abgelegt. HDFS ist ein verteiltes Dateisystem, das für die effiziente Verwaltung großer Datenmengen genutzt werden kann. Es setzt sich aus einem Namenode und mindestens einem Datanode zusammen und zeigt aufgrund der verteilten Datenhaltung deutliche Effizienzsteigerungen bei Erweiterungen auf multiple Datanodes. Das HDFS erlaubt mehreren Clients parallelen Zugriff auf alle Ressourcen und die simultane Bearbeitung von Dateien. Der Namenode administriert die über mehrere Recheneinheiten verteilten Datanodes, die wiederum die Daten beherbergen. Diese werden mit Metadaten wie dem Namensraum, den Zugriffsrechten sowie einer Zuordnung semantisch annotiert. Letztgenannte dient zur Zusammenfassung von Dateien zu Blöcken. Die Blockgröße wird auf 64 MB festgesetzt, um der Größe der verarbeiteten Dateien Rechnung zu tragen. Dadurch kann eine effizientere Kommunikation realisiert werden. Beim Namenode und den Datanodes handelt es sich um Prozesse, die auf Benutzerebene betrieben werden [169]. Datenodes treten gegenüber dem Namenode als Clients auf, stellen eine Verbindung zu diesem her und erhalten nach einer Prüfung der Rechte Zugriff auf die Dateien. Dabei wird die Lokation der Dateien mit Hilfe der Blockinformation kodiert [170]. Das Hadoop-Ökosystem bietet neben dem HDFS selbst noch zahlreiche weitere Services, von denen die für diese Arbeit relevantesten kurz vorgestellt werden. Spark ist ein System zur Verarbeitung verteilter Datenstrukturen, das zur Homogenisierung und Erzeugung nutzbarer Zusammenführungen der verteilt gespeicherten Daten dient. Es beruht im Wesentlichen auf "Resilient Distributed Datasets" [171]. Innerhalb des Prototyps wird es zur Konvertierung und Komprimierung von Hive-Tabellen verwendet. Die Daten werden dabei gezippt abgespeichert. Hive stellt eine Hadoop-interne Abfragesprache dar, die aus Spark heraus genutzt werden kann und daher keine separate Nutzung des Dienstes notwendig macht. Die beiden Services zur logischen Zusammenführung abgefragter Daten bauen auf Yarn als Ressourcenmanager auf. Zeppelin stellt die verwendete Entwicklungsumgebung dar, die gleichzeitig auch den Betrieb der neuronalen Netze erlaubt. Die Netze selbst werden in Python mit Keras und Tensorflow-GPU implementiert, das sich mit Zeppelin nutzen lässt. Tensorflow-GPU stellt eine auf das Training mit Grafikkarten optimierte Variante von Tensorflow dar. Es

wird in der Version 1.11.0 in Kombination mit CUDA Version 9.0 verwendet. Ambari liefert eine Web-basierte Ansicht auf das Gesamtsystem und erlaubt eine nutzerfreundliche Kontrolle aller stattfindenden Trainings- und Abfrageprozesse. Die Systemarchitektur ist in Abbildung 5.3 visualisiert [169] [172].



Abbildung 5.3: Aufbau des realisierten Hadoop-Clusters (links) sowie der eingerichteten Absicherung des Datenzugriffs durch Kerberos (rechts)

Hadoop integriert im Normalfall keine Benutzerauthentifizierung. Dies ermöglicht jedem externen Nutzer vollen Zugriff auf potentiell sensible Daten. Um diese Sicherheitslücke zu schließen, wurde zusätzlich Kerberos aufgesetzt, um eine Erhöhung der Datensicherheit zu erreichen. Kerberos wird im Rahmen des realisierten Prototyps mit einem Server zur Authentifizierung, einem Server zur Erstellung von Tickets, dem Ticket Granting Server (TGS), sowie einem Kontrollmechanismus beim jeweiligen Dienst verwendet. Im ersten Schritt sendet der Nutzer eine Anforderung an den für die Authentifizierung zuständigen Server. Als den Zugriff auf einen Dienst anfragender Nutzer kann sowohl ein realer Nutzer als auch ein anderer Dienst fungieren. Der Server prüft daraufhin die Nutzerdaten und sendet dem Nutzer zwei verschlüsselte Tickets mit Session Key, Zeitstempel und der Kennung des Dienstes. Ein Ticket kann durch den Nutzer mit Hilfe seines Passworts geöffnet werden, das zweite ist nur für den TGS-Server einsehbar. Der registrierte Nutzer verwendet den entschlüsselten Session Key zur Kommunikation mit dem TGS-Server, indem er seine Service-Anfrage verschlüsselt an den TGS-Server übermittelt. Dieser kann das zweite ungeöffnete Ticket öffnen und kontrollieren, ob der Nutzer den korrekten TGS Session Key verwendet hat. Ist dies der Fall, kann für den Nutzer geprüft werden, ob dieser die Befugnis für die Durchführung der Service-Anfrage, wie bspw. das Auslesen bestimmter Hive-Tabellen, hat. Ist auch dies der Fall, beginnt dieselbe Prozedur bezüglich des Dienstes, zu dem die Anfrage gestellt wurde. Der Nutzer erhält zwei verschlüsselte Tickets vom TGS-Server, von dem das eine mit dem bereits erhaltenen TGS Session Key entschlüsselt werden kann und das zweite nur für den Dienst einsehbar ist. Der neu entschlüsselte HTTP-Key kann zur Kommunikation mit dem Dienst verwendet werden. Die mit dem HTTP-Key verschlüsselte Anfrage wird an den Dienst übermittelt, der nach einem erfolgreichen Abgleich des

Keys mit dem des zweiten ungeöffneten Tickets zur Ausführung der Anfrage für die Dauer der Session berechtigt wird [173]. Der implementierte Ablauf ist in Abbildung 5.3 visualisiert. Kerberos erlaubt die direkte Verknüpfung der Realms mit den registrierten Benutzern. Da die Benutzer-Accounts unter Linux in verschiedenen Gruppen mit variierenden Berechtigungen organisiert sind, können die Zugriffsrechte über die Gruppen auf die Hadoop Services umgesetzt werden. Neben personenbezogenen Zugriffsrechten können auch Services Rechte für den Zugriff auf definierte Daten sowie weitere Services zugewiesen werden. Kerberos zeichnet sich daher durch eine hohe Eignung für die Organisation der komplexen Zugriffskonstellationen innerhalb des Hadoop-Systems und die Sicherung sensibler Daten aus.

## 5.3 Komponente zur adaptiven Qualitätssicherung

Im Folgenden wird auf die Software-Komponenten zur adaptiven Qualitätsregelung eingegangen. Dabei werden die zusammenhängenden Subkomponenten zur Detektion, Modellierung und Kompensation von anomalen Systemdynamiken näher betrachtet. Für das Training der zu Grunde liegenden neuronalen Netzarchitekturen wird eine performancestarke Hardware benötigt, die ein zeiteffizientes Training und damit eine schnelle Adaption des Prozessgeschehens ermöglicht. Eine Übersicht über die verwendete Hardware ist in Tabelle 5.1 gegeben [169].

Eigenschaften	IAS Namenode & Datanode 1	IAS Datanode 2	
IP-Adresse	129.69.102.200	129.69.102.201	
CPU Cores	72	32	
RAM	512 GB	64 GB	
GPU	2 x Tesla K80	-	
Speicherplatz	1.8 TB	7.2 TB	
Betriebssystem	CentOs 7.5	CentOs 7.5	
Hostname	Ias1223.ias.uni-stuttgart.de	Ias1224.ias.uni-stuttgart.de	
Hadoop-Version	3.0.1.0-187	3.0.1.0-187	

Tabelle 5.1: Daten zur verwendeten Hardware

Die verwendeten Ressourcen bestehen dabei aus zwei Servern, die über eine Speicherkapazität von 16 TB bei einer Rechenunterstützung von 4 GPU's (2x Tesla K80) verfügen. Die vorhandenen Server können auf 8 GPU's und 128 TB erweitert werden, sodass eine Übertragung des Ansatzes auf weitere Prozessketten möglich ist.

### 5.3.1 Subkomponente Detektion

Die Subkomponente zur Detektion von Anomalien beginnt mit einer Abfrage aktueller Daten in der Echtzeitdatenbank und bezieht zusätzlich Daten aus dem Hadoop-System mit ein. Die Abfragen werden als in Python eingebettete SQL-Abfragen implementiert, die in anwendungsspezifische Abfragen für die Jedox Suite bzw. das HDFS umgewandelt werden. Die Abfrage der Daten der Jedox Suite wird quasi-kontinuierlich ausgeführt, sodass der Eingang neuer Daten stetig registriert werden kann. Sind keine neuen Daten vorhandenen, geht die Komponente für eine definierte Zeitspanne in einen Idle-Zustand über und führt die Abfrage danach erneut aus. Da die Standard- bzw. Null-Werte der an der Anlage angebrachten Sensoren automatisch gefiltert und verworfen werden, kommt es genau dann zu einer erfolgreichen Abfrage, wenn ein neues Bauteil produziert wird. Die auf diese Weise während des Betriebs erfassten Daten sind bereits annotiert und verdichtet. Die historischen Daten müssen noch in einem weiteren Schritt im Rahmen der Vorverarbeitung durch eine DWT komprimiert und Merkmale mit hohem Informationsgehalt extrahiert werden. Zusätzlich wird eine Unterdrückung von Rauscheffekten erzielt. Die Implementierung ist in Python 3.6 realisiert.



Abbildung 5.4: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Detektion

Im Anschluss werden die Daten dem LSTM-Autoencoder zugeführt, der daraufhin eine Rekonstruktion der Daten vornimmt. Auf Basis des Rekonstruktionsfehlers kann, in Abhängigkeit vom Datensatz, einerseits ein Weitertraining erfolgen und andererseits eine Detektion umgesetzt werden. Dazu wird der trainierte Decoder als inverses Modell in den in Abschnitt 4.3.2.1 beschriebenen Mechanismus zur Detektion eingebettet, der über dynamische Grenzwertbildungen eine probabilistische Aussage über den Grad der detektierten Abweichung generiert. Die bestimmten Kennzahlen, wie Varianzen und Konfidenzintervalle, werden einerseits als Label für die verarbeiteten Daten verwendet, die dann via SQL-Statement zurückgeschrieben und für die Modellierung genutzt werden. Die gelbe Markierung deutet die Verknüpfung zur Subkomponente der Modellierung an. Andererseits werden die Kennzahlen via REST Post an die in Abschnitt 5.4 erläuterte Web-basierte Benutzerschnittstelle übermittelt und dem Endnutzer zur Einschätzung der aktuellen anomalen Prozessdynamiken visualisiert. Der beschriebene Ablauf ist in Abbildung 5.4 dargestellt.

## 5.3.2 Subkomponente Modellierung

Aufbauend auf den vom Anomalie-Detektor gelabelten Daten werden Prädiktionsmodelle trainiert. Die Daten werden in einem separaten Schema des Hadoop-Clusters gehalten und können

über SQL-basierte Abfragen gelesen werden. Das Prozessmodell  $f_n(\cdot)$  ist gemäß eines NARX-Modells aufgebaut und wird als Seq2Seq-LSTM-Netz in Python 3.6 mit Keras und Tensorflow-GPU implementiert. Die Modelle der internen Modellunsicherheit  $g_n(\cdot)$  sowie der externen Störeffekte  $h_n(\cdot)$  entsprechen nichtlinearen autoregressiven Modellen, die ebenfalls als Seq2Seq-LSTM-Netze implementiert sind. Die Eingänge verfügen dementsprechend über keine externen Variablen. Alle Modelle werden mit dem Levenberg-Marquardt-Algorithmus im Rahmen einer Hyperparameteroptimierung trainiert. Die trainierten Modelle werden dann zur Vorhersage auf Basis der während des Betriebes neu erfassten Prozess- und Anlagendaten genutzt und kontinuierlich weitertrainiert.



Abbildung 5.5: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Modellierung

Die drei Netze werden für jeden Prozessschritt implementiert. Dazu werden dieselben Netze, mit angepassten Strukturen der Ein- und Ausgangsparameter, für alle Prozessschritte genutzt und jeweils neu instanziiert. Das kooperative Prädiktionsschema entspricht einer Python-Klasse, die für den Datenaustausch zwischen den Netzen konzipiert wurde. Auch sie wird für jeden Prozessschritt neu instanziiert und auf den Datenraum angepasst. Die der Anzahl an Prozessschritten entsprechende Anzahl an Prädiktionsschemata werden über eine definierte Schnittstelle in den nachfolgend erläuterten Prototypen des rekursiven Optimierungsschemas eingebunden. Die Schnittstelle regelt den Austausch von Daten und kapselt die Funktionalität der Prädiktionsschemata. Sie übergibt diesen die Prozessnummer, den Vorhersagehorizont sowie die Stellgrößen und erhält als Ausgabe die erzeugte Trajektorie an Prozessausgängen. Die rote Markierung zeigt die entsprechende Verbindung zur Subkomponente der Kompensation an. Zur Bereitstellung von Informationen für den Nutzer ist auch an dieser Stelle eine REST-Schnittstelle implementiert, die die reinen Vorhersagedaten zur Visualisierung an die GUI weitergibt. Der beschriebene Ablauf ist in Abbildung 5.5 dargestellt.

## 5.3.3 Subkomponente Kompensation

Basierend auf der Implementierung des kooperativen Prädiktionsschemas und der Seq2Seq-LSTM-Netze wird die rekursive Optimierung implementiert. Diese beginnt mit einer Zerlegung des Problems in Teilprobleme. Jedes Teilproblem entspricht einem nichtlinearen Optimierungsproblem mit Nebenbedingungen, das auf Basis der kooperativen Prädiktion mit dem Levenberg-Marquardt-Algorithmus gelöst wird. Zur Implementierung des rekursiven Optimierungsschemas und der übergeordneten Regelungslogik wurde Matlab und dessen Optimization Toolbox genutzt. Um Daten zwischen Matlab und Python auszutauschen und gegenseitige Funktionsaufrufe realisieren zu können, wurde die Matlab Engine API verwendet. In Matlab vorhandene Datentypen werden automatisch auf die äquivalenten Datentypen in Python abgebildet und umgekehrt. Dadurch ist eine parallele Nutzung beider Anwendungen unter Ausnutzung der jeweiligen Vorteile möglich.



Abbildung 5.6: Vereinfachte Darstellung des Aufbaus der Subkomponente zur Kompensation

Für die Entwicklung von Python-Code wurde der Paketmanager Anaconda in Kombination mit der Entwicklungsumgebung PyCharm eingesetzt. Die Implementierungen in Python beruhen vor allem auf den Bibliotheken NumPy, SciPy, scikit-learn, Pandas und Tensorflow. Der REST Post zur Übergabe der berechneten Stellstrategien wird hingegen am Ende der rekursiven Optimierungsprozedur in Matlab ausgeführt. Die mit jedem neuen Teilproblem berechneten optimalen Stellgrößen und die zugehörigen Prozessausgänge werden für die Lösung des nächsten Teilproblems zurückgeführt. Dabei werden die entsprechenden Matlab-Vektoren in Python-Arrays umgewandelt. Größere Datenmengen werden über das für beide Seiten native HDF5 Format ausgetauscht, wobei Python-seitig dazu mit Pandas gearbeitet wird. Die Subkomponente wird mit dem Eintreffen neuer Bauteile an den Produktionseinheiten gestartet und geht nach abgeschlossener Optimierung und dem Post der Stellstrategien in einen Wartezustand über. Dieser hält an, bis erneut neue Bauteile an den Produktionseinheiten registriert werden. Die an die GUI per REST Post übermittelten Stellgrößen können automatisiert an die Anlagensteuerungen weitergeleitet und dort bei der nächsten Iteration zur Kompensation anomaler Systemdynamiken umgesetzt werden. Der beschriebene Ablauf ist in Abbildung 5.6 dargestellt.

## 5.4 Komponente für die Benutzeroberfläche

Damit der Benutzer mit den Funktionalitäten der in den vorherigen Abschnitten vorgestellten Implementierungen interagieren kann, wird eine Web-basierte Software-Applikation entwickelt. Dabei handelt es sich um einen REST-Server, der in Python mit Django entwickelt wurde und mit dem über REST Calls sowie WebSockets kommuniziert werden kann. Es wurde eine Nutzerverwaltung inklusive Registrierungsmechanismus für neue Nutzer über einen E-Mail-Versand implementiert. Die Nutzerdaten werden in einer PostgreSQL-Datenbank verwaltet, über die der Server auch weitere anwendungsbezogene Daten administrieren kann. Der Server kann innerhalb von Unternehmen als Private Cloud aufgesetzt werden, um verschiedenen registrierten Nutzer aus unterschiedlichen Abteilungen und Unternehmensbereichen einen Zugriff auf das Assistenzsystem zu ermöglichen. Die auf dem REST-Server ausgestaltete Software-Applikation kapselt alle Komponenten der vorherigen Abschnitte und kommuniziert mit diesen, um definierte Metriken und Parameter zu extrahieren, die dann in verschiedenen Ansichten visualisiert werden. Die drei Komponenten zur adaptiven Qualitätsregelung interagieren mit dem Server über REST Calls. Informationen über detektierte Anomalien und deren Charakteristiken werden im JSON-Format an den Server übermittelt und visualisiert. Hinsichtlich der Prozessmodelle können Lernvorgang und Prädiktionen abgebildet werden. Für die darauf aufbauende Kompensation der Anomalien wird dem Nutzer eine Ansicht zur Anpassung der Stellgrößen bereitgestellt. Diese ermöglicht die individuelle Anpassung der durch das System vorgeschlagenen Stellstrategien, die einmalige Übernahme aller Vorschläge sowie den durchgehenden vollautomatischen Betrieb auf Basis der generierten Stellstrategien. Die Zugriffsrechte sind auf ausgewählte Mitarbeiter beschränkt, die für die Prozesssteuerung zuständig sind. Abbildung 5.7 zeigt die Startseite des Web-basierten Assistenzsystems und eine Eingabemaske für Stellgrößen. Das Hadoop-System ist vollständig in die Software-Applikation eingebunden.



Abbildung 5.7: Startseite des Assistenzsystems (links) und Ansicht zur Prozesskontrolle (rechts)

Es stehen diverse Ansichten zum Monitoring und zur Überwachung kritischer Parameter sowie der durch die Umsetzung der Stellstrategien erzielten Qualitätsmetriken zur Verfügung. Das Monitoring erlaubt das Verfolgen von Echtzeitverläufen und integriert zusätzlich Ansichten auf die historischen Daten. In Abbildung 5.8 ist dazu ein Beispiel sowie die Überwachung der Qualitätsmerkmale dargestellt. Da es, wie zuvor in Abschnitt 5.2.2 erläutert, ein separates Authentifizierungssystem zur Datensicherung gibt, können nur autorisierte Benutzer neben den Funktionalitäten des Assistenzsystems zusätzlich auch die sensiblen Rohdaten nutzen.



Abbildung 5.8: Monitoring gefilterter Prozessdatenverläufe (links) und Überwachung definierter Qualitätsmetriken (rechts)

Die Komponente zur Datenerfassung und eindeutigen Rückverfolgung aller Bauteile inklusive der zugehörigen multidimensionalen Echtzeitdatenbank ermöglicht die Visualisierung des Trackings ausgewählter Bauteile innerhalb der Prozesskette. So können zu jedem Zeitpunkt die an jedem Prozess befindlichen Bauteile dargestellt und verdichtete Prozessparameter angezeigt werden. Dies ermöglicht den Produktionsmitarbeitern eine verbesserte Kontrolle über die logistischen Abläufe sowie die Qualität der Bauteile. Neben aktuellen, das Prozessgeschehen beschreibenden Kennzahlen können zugehörige Metadaten in einer sich stetig aktualisierenden Ansicht eingesehen werden. Dies umfasst z.B. die Topologie der jeweiligen Produktionseinheit, die Informationen über aktuell eingebaute Werkzeuge oder neu angebrachte Sensoren sowie verschiedene statistische Kenngrößen. In Abbildung 5.9 ist die Rückverfolgung von Bauteilen anhand einer festen Identifikationsnummer sowie die Anzeige von Metadaten illustriert.



Abbildung 5.9: Bauteilrückverfolgung entlang der Prozesskette (links) und Anzeige von Metadaten zu den Produktionseinheiten (rechts)

104

Die auf dem vorgestellten REST-Server basierende Software-Applikation zur Realisierung der benutzerfreundlichen GUI kann über verschiedene mobile Endgeräte, wie Smart Phones und Tablets, sowie über handelsübliche Rechner aufgerufen werden und zeichnet sich durch ein responsive' Design aus. Die mit HTML 5.2 und CSS entwickelten Ansichten passen sich dadurch an die Größe des verwendeten Endgeräts an. Für die Realisierung der den Webseiten zu Grunde liegenden Logik wurde JavaScript genutzt. Zusätzlich wurden Smart Glasses, eine RealWear HMT-1 und eine Vuzix M400, an das Assistenzsystem gekoppelt, um den Produktionsmitarbeitern Handlungsempfehlungen unmittelbar und nutzerfreundlich anzeigen zu können. Das Assistenzsystem läuft auf der in Abschnitt 5.3 tabellarisch aufgeführten Hardware. Diese ist örtlich von den Prozessketten getrennt. Für die Umsetzung des bidirektionalen Datentransfers zwischen Prozesskette und Assistenzsystem werden TCP/IP-Sockets, die in den verteilten Steuerungen der Anlagen initiiert werden, und darauf aufbauende REST Calls verwendet. Für die Rückführung von Stellgrößen wird eine Verbindung zwischen der Web-Oberfläche und den Steuerungen der Produktionseinheiten benötigt, die über WebSockets und TCP/IP-Sockets realisiert wurde. Die an den Anlagen eingehenden Stellstrategien werden dann mit dem nächsten eintreffenden Bauteil von den Steuerungen umgesetzt.

# 6 Evaluierung

Nachdem das realisierte Assistenzsystem und seine Komponenten im vorangegangenen Abschnitt ausführlich beschrieben wurden, wird in diesem Abschnitt auf die erzielten empirischen Ergebnisse eingegangen. Das Assistenzsystem wurde anhand von drei Prozessketten der Massivumformung evaluiert. Diese Prozessketten unterscheiden sich dabei in vielerlei Hinsicht. Die erste Prozesskette wurde am Institut für Umformtechnik der Universität Stuttgart im Rahmen des vom Bundesministerium für Wirtschaft und Energie geförderten Projektes EMuDig4.0 als Modellprozesskette unter Laborbedingungen aufgebaut. Sie dient primär zur Nachbildung zahlreicher in der Industrie ungelöster Probleme und unbekannter Phänomene. Die Prozesskette wird zur Fertigung von Stahlerzeugnissen genutzt und beinhaltet als zentralen Prozessschritt einen zweistufigen Warmumformprozess. Darüber hinaus wird hier kein in einem realen System nutzbares Produkt gefertigt, sondern ein Musterbauteil hergestellt. Das Musterbauteil dient, wie die Prozessschritte und die Anlagenkomponenten selbst, als Referenz für in der Industrie ungelöste Probleme. Es ist daher so ausgelegt, dass die empirische Untersuchung verschiedener unbekannter Effekte möglich ist. Die Datengrundlage setzt sich aus apriori generierten Simulationsdaten sowie im Rahmen mehrerer Testdurchläufe während des Betriebs erfasster Daten zusammen. Tabelle 6.1 zeigt eine Übersicht zu den wichtigsten Daten der Prozessketten.

	Modellfabrik am IFU	Pilot-Anlagen bei Otto Fuchs		
Historische Daten	Aus numerischen Simulationen	Aus Anlagensteuerungen und Datenbanksystemen		
Historische Daten	6 Prozessschritte	8 Prozessschritte		
Historische Daten	28 Parameter/Signale	32 Parameter/Signale		
Historische Daten	4.000 Durchläufe/gefertigte Produkte	Ca. 30.000 Durchläufe/gefertigte Produkte		
Historische Daten	Datenmenge: ca. 90 GB	Datenmenge: ca. 1 TB		
,Live'-Tests	172 Durchläufe/gefertigte Produkte	1157 Durchläufe/gefertigte Produkte		
Prozesskette	Aluminium	Aluminium		
Prozesskette	2-stufiger Prozess – Warm/Druck	3-stufiger Prozess – Warm/Druck		
Prozesskette	,Labor-Bauteil'	Felgen/Fenster		

Tabelle 6.1: Übersicht über die verwendeten Datensätze und die durchgeführten Testläufe

Die zweite und dritte Prozesskette befinden sich bei der Otto Fuchs KG, einem Zulieferer der metallverarbeitenden Branche für die Automobil- sowie Luft- und Raumfahrtindustrie. Die Prozessketten dienen zur Übertragung und weiteren Evaluierung der im Rahmen der Modellfabrik gewonnenen Erkenntnisse. Die Prozessketten dienen im Gegensatz zur Modellfabrik zur Fertigung von realen Bauteilen aus Aluminium. Dabei handelt es sich zum einen um Felgen für Automobile und zum anderen um Fensterrahmen für Flugzeuge. Die erstgenannte Prozesskette eignet sich vor allem für die Evaluierung des Detektions- sowie des Prädiktionsmechanismus und die zweitgenannte für die Evaluierung der adaptiven Kompensation sowie der Gesamtheit aller Funktionalitäten des Assistenzsystems. Die Funktionsfähigkeit des Konzeptes bei beiden Prozessketten soll darüber hinaus die Übertragbarkeit auf weitere Produktionssysteme belegen. Die Prozessketten beinhalten jeweils mehrstufige Schmiedeprozesse. Die Datengrundlage setzt sich aus historischen Daten der Prozessketten im Terabyte-Bereich zusammen. Darüber hinaus können stetig während der Produktion weitere Daten erfasst und Untersuchungen durchgeführt werden.

Die verfolgte Strategie zur Evaluierung des Konzepts sieht einen Bottom-Up-Ansatz vor, indem zunächst die einzelnen integralen Bestandteile des Gesamtsystems auf ihre Funktionsfähigkeit und Güte hin bewertet werden und darauf aufbauend das Zusammenspiel der in wechselseitigen Abhängigkeiten stehenden Bestandteile eruiert wird. Letzteres geschieht durch die sukzessive Erweiterung der betrachteten Funktionalitäten bis hin zur Gesamtheit aller Funktionalitäten bei der Evaluierung des Gesamtsystems. Gemäß dieser Evaluierungsstrategie wird bspw. bei der Evaluierung des Prädiktionsmechanismus die Funktion des Anomalie-Detektors vorausgesetzt und deren Zusammenspiel bewertet. Die nachfolgende Darstellung der empirischen Ergebnisse gliedert sich dementsprechend nach den drei zentralen Bestandteilen des Konzeptes, der Anomalie-Detektion, der kooperativen Prädiktion sowie der Kompensation von Anomalien. Dadurch kann ein gezielter Abgleich mit den bestehenden Anforderungen erfolgen.

## 6.1 Evaluierungsfall 1: Anomalie-Detektion und Prädiktion

Der erste Evaluierungsfall dient zur Evaluierung des konzipierten Anomalie-Detektors und des kooperativen Prädiktionsschemas. Dazu wird zunächst auf die dafür relevanten Bestandteile der beiden Prozessketten und die zugehörigen Daten eingegangen. Darauf aufbauend werden die erzielten empirischen Ergebnisse hinsichtlich der Detektion von Anomalien sowie der Charakterisierung derselben anhand definierter Metriken vorgestellt. Darüber hinaus werden die Ergebnisse der kooperativen Prädiktion präsentiert, die die Resultate für die Vorhersage kurzfristiger, stationärer Dynamiken sowie die Vorhersage von nicht-stationären Langzeitabhängigkeiten umfassen. Eine Bewertung der ganzheitlichen Vorhersage aller auftretender Effekte für alle Aktorsysteme sowie eine Performanceanalyse der zu Grunde liegenden neuronalen Netzarchitekturen schließt das Kapitel ab.

### 6.1.1 Beschreibung der Prozessketten: Modellfabrik und Otto Fuchs

Zur Evaluierung und Bewertung des Ansatzes steht eine industrielle Produktionsanlage aus dem Bereich der Umformtechnik zur Verfügung. Die Anlage ist Teil einer Produktionslinie bei der Otto Fuchs KG, einem deutschen Zulieferer der Automobil- sowie Luft- und Raumfahrtindustrie für Metallkomponenten. Die untersuchte Anlage produziert Felgen für die Automobilindustrie und ermöglicht die Bewertung des Konzepts zur Anomalie-Detektion und Prädiktion in einem realen industriellen Umfeld. Zur Herstellung der Felgen findet ein mehrstufiger Warmschmiedeprozess auf einer hydraulischen Presse statt. Der dabei generierte und untersuchte Datensatz wurde über die Anlagensteuerung der hydraulischen Presse akquiriert und hat eine nominelle Größe von 1 TB im kommaseparierten CSV-Format. Die analysierten historischen Daten wurden über einen Zeitraum von 4 Monaten aufgezeichnet. Der Datenraum umfasst bis zu 86 Parameter an Prozess- und Anlagendaten, die zum einen aus der SPS und zum anderen direkt aus Sensoren extrahiert wurden. Die Abtastrate variiert dabei im Bereich von 50 us bis 10 ms. Die Presse verfügt über ein komplexes Pumpensystem, das aus acht Pumpen besteht, die den Gesamtdruck erzeugen. Wenn eine Pumpe anomale Dynamiken aufweist, können andere Pumpen die Schwankungen ausgleichen, sodass der Gesamtausgangsdruck ungefähr auf dem gewünschten Niveau verbleibt und die Anomalie für den Produktionsmitarbeiter nicht unmittelbar erkennbar ist. Infolgedessen liegt die umgesetzte Prozess- und Produktqualität innerhalb der Toleranz, die stark belasteten Pumpen verschleißen hingegen schneller. Weitere Informationen zum Anwendungsfall sind in [27] und [28] zu finden. Eine genaue Detektion von kurzfristig und langfristig auftretenden Anomalien sowie eine Vorhersage dieser unterschiedlichen Dynamiken auf der Grundlage von historischen Messdaten könnte zu einer Erhöhung der Prozessstabilität beitragen. Bevor der im Hadoop-System verwaltete Datensatz für Training und Auswertung eingesetzt wird, werden die Daten normiert, bereinigt und verdichtet. Auf diese Weise werden Merkmale mit erhöhtem Informationsgehalt generiert, die ein effizientes Training und ein verbessertes Lernergebnis ermöglichen. Das Training der Seq2Seq-LSTM-Netze wird mit GPU-Unterstützung auf der in Tabelle 5.1 vorgestellten Hardware durchgeführt, um die Rechenzeit zu optimieren.

## 6.1.2 Diskussion und Bewertung der empirischen Ergebnisse

## Detektion von Anomalien

Ein Beispiel dafür, wie sich die Druckerzeugung während des ersten Hubes und des zweistufigen Schmiedevorgangs verhält, ist in Abbildung 6.1 dargestellt. Das Pumpensystem entspricht einem Hauptaktuator der hydraulischen Presse und wird im Rahmen der Evaluierung des vorgestellten Konzepts zur Anomalie-Detektion und kooperativen Prädiktion intensiv untersucht. Ein Auszug an aufgetretenen und direkt auf den Druckverlauf abbildbaren Anomalien sowie das normale Verhalten dieser physikalischen Größe sind im untenstehenden Diagramm überlagert.



Abbildung 6.1: Beispiel eines normalen Druckverlaufs im Vergleich zu den Auswirkungen von auf das Pumpensystem wirkenden Anomalien

Basierend auf den Daten der Felgenlinie wird der Ansatz für die Detektion und kooperative Prädiktion von Anomalien auf den hochdimensionalen Prozessdatenraum angewendet. Die Einstellparameter der Seq2Seq-LSTM-Netzarchitekturen werden durch eine Hyperparameter-Optimierung bestimmt, um einen möglichst stabilen Lernprozess zu realisieren. Dabei wird sowohl die Anzahl an Neuronen pro Schicht als auch die Anzahl an Schichten in fest definierten Bereichen variiert. Die Lernrate wird in den Grenzen von 0,0005 bis 0,004 eingestellt. Insgesamt wird in 500 Wiederholungen über die dem Netz zugeführten Batches iteriert, die jeweils über eine Länge von 128 und 2048 Datenpunkten haben.



Abbildung 6.2: Detektion und Charakterisierung von Anomalien

#### 109

Die Dropout-Rate wird im Bereich von 0 bis 0,4 variiert. Der Lernprozess zeigt Konvergenz für alle getesteten Parametrierungen und endet innerhalb einer Trainingszeit von bis zu 8 Stunden.

Ergebnisse der Netzarchitekturen für die Erkennung von Anomalien sind in Abbildung 6.2 dargestellt. Dabei ist die Häufigkeit des Auftretens eines bestimmten Prozesszustands gegenüber der für Autoencoder-Strukturen spezifischen Metrik des Rekonstruktionsfehlers aufgetragen [178][179]. Es ist erkennbar, dass eine klare Klassifikation von normalem Prozessverhalten und anomalen Systemdynamiken erreicht und eine Charakterisierung der Abweichungen vorgenommen werden kann Mit Hilfe einer zeitvarianten Ausprägung des Rekonstruktionsfehlers konnten sowohl schleichende bzw. kontinuierliche als auch abrupt auftretende Änderungen des Betriebspunktes von Aktorsystemen wie Verschleißprozesse (dunkelblau) bis hin zu Komponentenausfällen (hellblau) charakterisiert werden. Weitere empirische Ergebnisse sind in den Abbildungen 6.3 und 6.4 dargestellt. Abbildung 6.3 zeigt den vom Anomalie-Detektor ausgegebenen Rekonstruktionsfehler für eine Sequenz von Testdaten des anomalen Prozessverhaltens einschließlich der durch Abbildung 6.1 aufgezeigten Anomalien. Die Ergebnisse sind für das Pumpensystem der hydraulischen Presse visualisiert. Pumpen, die bereits über einen längeren Zeitraum verschleißen, befinden sich im Zeitintervall der Detektion nahe einem Defekt. Eine Pumpe zeigte mehrere Plateau-Peaks aufgrund kurzfristig auftretender Druckausgleichseffekte. Die einzelnen Pumpen werden in einer separaten Pumpe zusammengeführt, sodass sie als Knotenpunkt aller anderen Pumpen fungiert und dabei bspw. Vibrationseffekten ausgesetzt ist. Diese im Vergleich zu anderen Teilen des Aktorsystems geänderte Funktion macht sich auch in der Detektion bemerkbar. Alle auftretenden Anomalien können erkannt und klassifiziert werden.



Abbildung 6.3: Durchschnittliche Abweichung in der Rekonstruktion über die Zeitsequenzen ausgewählter Ereignisse

Ein über einen längeren Zeitraum auftretender Drift des Druckplateaus, der auf Verschleißeffekte zurückgeht, kann innerhalb dieses Datenauszugs explizit festgemacht werden. Die Verläufe der anderen Pumpen, die keinen externen Störeffekten ausgesetzt sind und ein normales Verhalten zeigen, liegen aufgrund des rauschminimierenden Charakters der DWT dicht beieinander nahe der Referenz.

Abbildung 6.4 veranschaulicht die Möglichkeiten des Anomalie-Detektors, den durch Abnutzung von einzelnen Aktoren verursachten Langzeitprozess einer Arbeitspunktverschiebung des gesamten Aktorsystems zu charakterisieren. Dies ist anhand der Korrelation des sich ändernden Arbeitspunktes gegenüber einer reduzierten Fähigkeit des Druckaufbaus einer einzelnen Pumpe und des entsprechend absinkenden Druckniveaus dargestellt. Der an der Pumpe detektierte Störeffekt nahm ungefähr linear über einen Zeitraum von 30 Tagen zu, bis die Fähigkeit des Druckaufbaus auf das in der Abbildung dargestellte Druckniveau eingeschränkt war (grün). Die entsprechende Ausgabe des Detektors zur daraus resultierenden Arbeitspunktverschiebung des gesamten Pumpensystems ist anhand des Rekonstruktionsfehlers abgebildet und entspricht einem exponentiellen Wachstumsverhalten der Arbeitspunktdifferenz über der Zeit (hellblau). Die flache graue Linie zeigt im Vergleich dazu die theoretische Ausgabe des Detektors für den Fall fehlender Störeffekte. Das in Kapitel 4.3.2.1 eingeführte Konstrukt zur Detektion von Anomalien zeigt somit gute Eigenschaften zur Erkennung und Charakterisierung sowohl für schleichende bzw. kontinuierliche als auch abrupt auftretende Anomalien. Die anomalen Effekte können auf Basis von aus dem Rekonstruktionsfehler abgeleiteten Metriken mit hoher Genauigkeit spezifiziert werden. Eine Ableitung aktuell vorhandener Anomalie-Charakteristiken ist daher möglich.



Abbildung 6.4: Schleichende Abweichung über die Zeit (grün) und Normalverhalten (dunkelblau) sowie deren Detektion mittels des Anomalie-Detektors (hellblau)

Darüber hinaus können die physikalischen Effekte mit den ausgewerteten Metriken quantifiziert werden. Im abgebildeten Fall kann bspw. die stetig weiter eingeschränkte Fähigkeit zum Druckaufbau formal in Form einer Korrelation zu der exponentiell wachsenden Rekonstruktionsmetrik beschrieben werden. Anhand der Exponentialfunktion kann eine Schätzung des aktuellen Arbeitspunktes und des Verschleißzustandes der Einzelkomponente berechnet und bei der weiteren Einstellung von Stellgrößen berücksichtigt werden.

## Kooperative Prädiktion

In diesem Abschnitt wird genauer auf die empirischen Ergebnisse zur kooperativen Prädiktion von Prozess- und Anomalie-Verhalten eingegangen. Abbildung 6.5 zeigt ein Beispiel für eine einstufige kooperative Prädiktion des Prozessverhaltens unter Berücksichtigung der vorhergesagten Anomalien, in diesem Fall eines über einen langen Zeitraum evolvierenden Druckabfalls durch Verschleißprozesse, der im Pumpensystem auftritt.



Abbildung 6.5: Kooperative Vorhersage des Prozessverhaltens unter Berücksichtigung von Anomalien

Abbildung 6.6 zeigt eine aggregierte Darstellung von Vorhersagen kurzfristiger, von einer Prozessdurchführung zur nächsten auftretender Störeffekte. Bei der dritten Prozessdurchführung kommt es in diesem Beispiel zu einer abrupten Anomalie. Zusätzlich zu der beschriebenen LSTM-Zellarchitektur wird die GRU-Zellarchitektur auf die Modellierung kurzfristig auftretender Störungen getestet. Das Diagramm impliziert, dass die GRU-Zelle stationäres Anomalie-Verhalten besser modellieren kann als die LSTM-Zelle. Der Verlauf für die LSTM-Zelle weist eine geringe Anpassung an das durch den Störeffekt ausgelöste Absenken auf. Die GRU-Zelle kann den Einfluss von erfassten und im Zellzustandsvektor abgebildeten Langzeitabhängigkeiten auf die Vorhersage bei Vorhandensein kurzfristig auftretender Dissonanzen hingegen stark reduzieren. Dadurch ist eine adaptive Abbildung stationärer Kurzzeiteffekte möglich. Der Gating-LSTM-Zelle Mechanismus der lässt eine derartig starke Unterdrückung der

Langzeitabhängigkeiten nicht zu. Diese Gegebenheiten können durch die erzielten Ergebnisse für den in dieser Arbeit betrachteten Anwendungsfall bestätigt werden. Daher sind Netzarchitekturen, die auf der GRU-Zellarchitektur beruhen, prädestiniert für die Modellierung kurzfristiger und stationärer Anomalien.



Abbildung 6.6: GRU vs. LSTM zur Prädiktion von Kurzzeitanomalien

Unter Einbeziehung nicht-stationärer langfristiger Anomalien zeigt die Seq2Seq-Architektur mit LSTM-Zellen bessere Ergebnisse als für stationäre kurzfristige Anomalien. Die durchschnittlich erzielten Werte für die verwendeten Fehlermetriken sind in Tabelle 6.2 dargestellt.

	RMSE (Root-Mean-Square Error)	MAPE (Mean Absolute Percentage Error)
Seasonal ARIMA	0,19441	28,117
Stacked-GRU-DWT	0,09398	21,815
Seq2Seq-LSTM-DWT	0,24326	48,219

Tabelle 6.2:	Vorhersagegenauigke	iten der untersucl	nten Architekturen	für abrupte A	Anomalien

Der ARIMA-Ansatz wird als klassischer Ansatz der Statistik, der in der Regelungstechnik eine breite Anwendung findet, als Referenz herangezogen. Dieser zeigt geringere Vorhersagefehler als der GRU-Ansatz im Hinblick auf Langzeitabhängigkeiten. Daher ist die geschichtete Netzarchitektur mit GRU-Zellen aufgrund ihrer Fähigkeit zur Dimensionsreduktion und der flexiblen Adaption aktuell gemessener Prozessdaten am besten geeignet, um stationäres Kurzzeitverhalten zu modellieren. Im Gegensatz dazu ist die Seq2Seq-LSTM-Architektur aufgrund der inversen Gedächtnisstruktur und des flexible einstellbaren Vorhersagehorizonts prädestiniert für die Abbildung nicht-stationärer Langzeitabhängigkeiten. In der Folge würde eine

Kombination beider Ansätze zu einer ganzheitlichen Darstellung von Kurzzeit- und Langzeitanomalien führen.

Abbildung 6.7 veranschaulicht eine aggregierte kooperative Prädiktion über einen Zeitraum von zwei Wochen. Die Abbildung zeigt eine normalisierte Kurve, die den Messungen des tatsächlichen Prozessverhaltens im betrachteten Zeitraum entspricht sowie eine Prädiktionskurve, die auf Basis des vorgestellten Schemas vorhergesagt wurde. Die Prädiktion berücksichtigt neben den normalen Prozessdynamiken die Modellierung interner Modellunsicherheiten sowie anomaler Dynamiken, die durch externe Störeffekte ausgelöst werden. Diese beziehen sich auf das gesamte Aktorsystem. Zusätzlich ist die Unsicherheit der Vorhersage in Form des 95%-Ouantils gegeben, die mit dem Vorhersagehorizont zunimmt. Es ist zu erkennen, dass der vorhergesagte Verlauf dem tatsächlichen sehr gut folgt und der reale Verlauf durchgehend im erwarteten 95%-Bereich liegt. Die Vorhersage ist ausreichend genau, um den Prozesszustand hinsichtlich eines definierten Zeitintervalls im Voraus bewerten zu können. In Abhängigkeit von den gewählten Dimensionen des Ein- und Ausgangsdatenraumes der Seq2Seq-Netzarchitektur können Vorhersagen über Horizonte von Sekunden bis hin zu Stunden und Tagen umgesetzt werden. Entscheidend dafür sind Parameter wie die Sequenzlänge, die Batchgröße und der auf den angestrebten Vorhersagehorizont abgestimmte Verdichtungsgrad der Eingangsdaten. Der zu wählende Horizont hängt stark damit zusammen, in welchen zeitlichen Abständen in den Prozess eingegriffen werden soll. Ist eine hochfrequente Adaption der Stellgrößen notwendig wird der Horizont entsprechend kleiner gewählt. Träge, sich nur langsam ändernde Prozessdynamiken erfordern hingegen ein Eingreifen innerhalb von Stunden bis Tagen. Unabhängig vom eingestellten Vorhersagehorizont ist die prädizierte Sequenz stets von probabilistischer Natur.



Abbildung 6.7: Kooperative Prädiktion des Prozessverhaltens unter Berücksichtigung auftretender Langzeitanomalien

Für jeden Zeitpunkt der Vorhersage kann somit eine Dichtefunktion angegeben werden, die die potentiellen zukünftigen Prozesstrajektorien mit einer Wahrscheinlichkeit bemisst. Hinsichtlich der Abbildung ist noch eine Besonderheit zu erkennen. Theoretisch sollte der Rekonstruktionsfehler bei kontinuierlichem Verschleiß stetig zunehmen. In der Abbildung ist jedoch an den Tagen 2 und 10 eine starke Abnahme bzw. Zunahme zu erkennen. Dies wird durch einen Produktwechsel während der Produktion verursacht. Das zweite Produkt zeigt einen kleineren zeitvarianten Rekonstruktionsfehler zwischen den Tagen 2 und 10. Nichtsdestotrotz gibt es im Laufe der Zeit einen zunehmenden Rekonstruktionsfehler innerhalb dieses Abschnitts, der vom Netz korrekt vorhergesagt wird. Dies führt nach dem erneuten Produktwechsel zu einem erhöhtem Fehlerwert, sodass die isolierte Betrachtung der beiden, den verschiedenen Produkten zugeordneten Abschnitten eine passende Vorhersage zeigt. Eine Detektion von Anomalien bei dem zweiten Produkt wird durch das geringe Niveau des Rekonstruktionsfehlers erschwert. Die Vorhersagegenauigkeit wird durch den Produktwechsel hingegen nicht beeinträchtigt, was suggeriert, dass das kooperative Prädiktionsmodell in der Lage ist, stationäre und nicht-stationäre Anomalien unabhängig vom Produkt zu modellieren.

# 6.2 Evaluierungsfall 2: Kompensation von Anomalien und Nutzerassistenz

Der zweite Evaluierungsfall fokussiert auf die Anwendung des Schemas zur Kompensation von anomalen, auf die Stellgrößen wirkenden Systemdynamiken. Dazu werden analog zum Aufbau des vorherigen Kapitels zunächst die relevanten Bestandteile der beiden Prozessketten beschrieben sowie die zugehörigen Datensätze und die Parametrierung der neuronalen Netze erläutert. Darauf aufbauend werden die erzielten empirischen Ergebnisse hinsichtlich der Modellierung von Abhängigkeiten entlang von Prozessketten sowie von internen Modellunsicherheiten und externen Störungen erörtert. Auf die Auswirkung von deren Integration auf die Modellierungsgüte von Prozessausgängen wird detailliert eingegangen. Abschließend wird die dadurch erreichte Kompensation von Anomalien beschrieben und anhand von ausgewählten Stellgrößenanpassungen visualisiert.

## 6.2.1 Beschreibung der Prozessketten: Modellfabrik und Otto Fuchs

Zur Evaluierung und Bewertung des Ansatzes zur Kompensation von Anomalien steht am Institut für Umformtechnik der Universität Stuttgart eine Prozesskette der Massivumformung zur Verfügung. Diese unter Laborbedingungen entwickelte Modellfabrik dient als Testumgebung zur Erprobung verschiedener Szenarien und hat die Nachbildung in der Industrie häufig auftretender und bisher ungelöster Problemstellungen zum Ziel. Sie ist auf die Herstellung von Stahlbauteilen ausgelegt und umfasst einen zweistufigen Prozess der Warmumformung. Nach einer Prüfung der Eigenschaften des Rohmaterials und der Geometrie am Eingang der Prozesskette erfolgt eine

induktive Erwärmung der Bauteile. Anschließend wird das Ober- und Untergesenk des Werkzeugs mit Sprühmittel versehen, um einen definierten Reibungskoeffizienten einzustellen. Das Sprühsystem arbeitet mit diskreten Sprühmittelmengen und Sprühausführungen, sodass darüber Reibungskoeffizienten angenähert werden können. Nach der Erwärmung legt ein Transportroboter das Bauteil in das Werkzeug der hydraulischen Presse ein. Das Bauteil wird daraufhin verformt und mit einem Laserprofilometer auf charakteristische Qualitätsmerkmale geometrischer Natur untersucht. Weiterhin können Temperaturprofile erstellt werden, die einen Rückschluss auf Materialeigenschaften geben könnten. Nach der Qualitätsmessung transportiert der Roboter das Bauteil zurück zur Presse und legt es in die zweite Kavität des Werkzeugs zur Durchführung des zweiten Umformschritts ein. Nach der zweiten Iteration wird das Bauteil aus der Presse entnommen und anschließend erneut auf die relevanten Qualitätsmerkmale geprüft. Am Ende der Prozesskette wird das Bauteil ausgelagert und kühlt ab. Eine ausführliche Beschreibung der realisierten Bauteilrückverfolgung entlang der Prozesskette ist in [25] zu finden. Abbildung 6.8 zeigt den Aufbau der Fertigungszelle mit dem Transportroboter mit angebautem Sprühsystem, der hydraulischen Presse sowie dem Produktionsmitarbeiter, der die Prozessabläufe durch die Umsetzung der automatisiert erzeugten Stellgrößen regelt. Ein Beispiel für ein komplexes Aktorsystem, das potentiellen externen Störungen ausgesetzt sein kann, ist durch das in derselben Abbildung dargestellte Werkzeug gegeben.



Abbildung 6.8: Prozesskette der Modellfabrik mit induktiver Erwärmung, hydraulischer Presse sowie Transporteinheit (links) und Werkzeug mit zwei Kavitäten (rechts)

Das Produktionsgeschehen entlang der Prozesskette wurde in numerischen Prozesssimulationen nachgebildet. Die numerischen Simulationen lieferten apriori Daten für eine Sequenz von 4000 nacheinander hergestellten Bauteilen. Somit kann auf eine historische Datenmenge der Größe von 90 GB zurückgegriffen werden, um die Modelle im Rahmen eines Vortrainings einzulernen. Bevor der Datensatz zum Training und zur Auswertung verwendet wird, werden die Daten normiert, bereinigt und verdichtet. Auf diese Weise werden, wie beim Datensatz des ersten Evaluierungsfalls, Merkmale mit erhöhtem Informationsgehalt generiert, die die Grundlage für

ein effizientes Training und ein verbessertes Lernergebnis ebnen. Darüber hinaus werden weitere Daten während des Betriebs erfasst und entsprechende Tests durchgeführt. Dazu werden über 150 weitere Bauteile produziert, die eine Bewertung des adaptiven Eingreifens in die Prozesskette zulassen. Es werden Störungen auf der Grundlage der erfassten Daten der hydraulischen Presse und der umgebenden Prozessschritte beobachtet und zum Ausgleich von Qualitätsschwankungen eingesetzt. Der hier erfasste Datenraum besteht aus 28 Parameter an Prozess- und Anlagendaten, die ebenso wie beim ersten Evaluierungsfall auf Basis des in Abschnitt 5.2.1 beschriebenen Konnektors zur Vereinheitlichung heterogener Datenquellen erfasst wurden.

Einstellbare	Anzahl	Neuronen/	Lernrate	Dropout-	Batch-	Lern-
Parameter	Schichten	Schicht		Rate	Größe	Algorith.
Wertebereich bzw. Methode	1 – 5	1 - 2000	0,0001 - 0,004	0-0,4	0-200	BPTT, LMA

Tabelle 6.3: Übersicht über die einstellbaren Parameter der Hyperparameteroptimierungen

Die Einstellparameter der auf den Daten trainierten Seq2Seq-LSTM-Architekturen werden im Rahmen von Hyperparameteroptimierungen variiert, um möglichst stabile Lernprozesse mit guten Konvergenzeigenschaften sowie hohen Modellgenauigkeiten zu erzielen. Dabei werden die in Tabelle 6.3 dargestellten Parameter in den angegebenen Bereichen variiert. Die Anzahl von Neuronen wird bspw. genauso angepasst wie die Anzahl der Netzschichten selbst. Die Lernrate wird im Bereich von 0,0001 bis 0,004 eingestellt. Die Dropout-Rate variiert zwischen 0 und 0,4.

## 6.2.2 Diskussion und Bewertung der empirischen Ergebnisse

Abbildung 6.9 zeigt die beim Training von vier unterschiedlich strukturierten Netzarchitekturen entstandenen Konvergenzkurven. Die erste Kurve veranschaulicht das Training von Netzen auf der Grundlage des NARX-Modells aus Gleichung 36. Die Netzeingänge sind daher so strukturiert, dass weder Abhängigkeiten entlang der Prozesskette noch interne Modellunsicherheiten oder externe Störungen berücksichtigt werden. Die Eingangsparameterräume der zu der zweiten und dritten Konvergenzkurve gehörenden Netze basieren auf den NARX-Modellen der Gleichungen 38 bzw. 46. Bei Kurve zwei wurden zunächst Interdependenzen einbezogen, die zu Beginn des Trainings zu einem sichtbar stabileren Lernprozess beitragen und insgesamt auch zu einer besseren Konvergenz führen. Die durch das erweiterte Modell erreichten, verbesserten Generalisierungseigenschaften der Netze führen zu einer optimierten Extrapolationsfähigkeit, die insbesondere im Hinblick auf die ersten produzierten Bauteile beim Training während des Betriebs zu einem robusteren Eingreifen führen. Die Berücksichtigung von Anomalien in Form von internen Modellunsicherheiten und externen Störeffekten führt zu einer schnelleren Konvergenz, da alle prozessrelevanten Faktoren in den Lernprozess einbezogen und bestehende Informationsdefizite minimiert werden. Die Berücksichtigung von Abhängigkeiten und Anomalien führt sowohl zu Beginn des Lernprozesses als auch insgesamt zu dem geringsten, über alle Trainingsdurchführungen gemittelten Verlustwert.



Abbildung 6.9: Durchschnittliches Konvergenzverhalten von vier neuronalen Netzen mit unterschiedlich strukturierten Eingangsparameterräumen

Das Weitertraining der vortrainierten Netze während des Betriebs führt zu einer iterativen Anpassung und Verbesserung der zeitvarianten Modelle und im Laufe der Zeit zu einer Steigerung der Vorhersagegenauigkeit. Abbildung 6.10 zeigt eine aggregierte Darstellung der Netzvorhersagen bei unterschiedlichen Abhängigkeiten entlang der Prozesskette und verschiedenen externen Störeffekten. Die Auswirkung von gemessenen, die Interdependenzen charakterisierenden Bauteileigenschaften sowie beobachteten Störungen ist in Form von verschiedenen Kurven über der diskreten Zeit bzw. über die in dieser Zeit verarbeiteten Bauteile aufgetragen. Die Abbildung ist so zu verstehen, dass es mit jedem neuen Bauteil zu einer Variation des in den Diagrammen für den Bereich angegebenen Parameters kommt. Einflüsse von Abhängigkeitsbeziehungen entlang der Prozesskette sowie häufig auftretender Störeffekte werden durch die Variationen einzelner Parameter oder einer Kombination von Parametern repräsentiert, die diesen im Wesentlichen ausgesetzt sind. Die trotz der Einflüsse erzielten Ausgaben der Netze und die tatsächlichen Ausgaben werden in der Abbildung in Form des Vorhersagefehlers verglichen, sodass eine Aussage über die Adaption dieser Effekte getroffen werden kann. Die ersten beiden Bereiche zu Variationen, bei denen die Parameter der Bauteiltemperatur und -größe im Vordergrund stehen, entsprechen denjenigen, die maßgeblich durch den vorangegangenen Prozess verursacht werden. Es kann folglich die Fähigkeit des Netzes zur Modellierung von Interdependenzen beurteilt werden. Der Vorhersagefehler ist für diese Bereiche beschränkt und der hohe Einfluss geometrischer Bauteileigenschaften auf nachfolgende Prozessschritte kann bestimmt werden. Externe Störeffekte im Hinblick auf Werkzeugtemperaturund Geschwindigkeitskurven, die bspw. auf Ausschussentstehung, Umgebungswärme oder Vibrationseinflüsse zurückgehen, können durch die Netze modelliert werden. Die Vorhersage ist

bis auf wenige Ausreißer nahezu identisch mit der Referenz, sodass eine hohe Vorhersagegenauigkeit vorhanden ist. Reibungsphänomene zeigen langfristige, nicht-stationäre Charakteristiken, die jedoch adäquat abgebildet werden können. Gegen Ende zeigt sich eine etwas größere Diskrepanz von Netzausgabe und Sollwertverlauf, die auf unbekannte Effekte bezüglich des Einbauraums zurückgeht. Diese konnte in Simulationen nicht adäquat abgebildet werden und wird daher während des Betriebes gelernt.



Abbildung 6.10: Aggregierte Darstellung der Netzvorhersage unter Berücksichtigung von Störeffekten

Zusätzlich zu diesen Ergebnissen zeigt Abbildung 6.11 eine detaillierte Sicht auf diesen ausgewählten Störeffekt. Die Diagramme zeigen die Qualität, repräsentiert durch die Gratbildung, in Abhängigkeit von der am Prozesseingang festgestellten Bauteilgröße sowie der Einbauraum-Verstellung, die der unbekannten externen Störung ausgesetzt ist. Links ist die erste und rechts die zweite Umformstufe dargestellt. Der Bereich im Graphen um eine Bauteilhöhe von 48,4 mm zeigt erhöhte Werte der Qualitätsmetrik im Vergleich zur optimalen horizontalen Ebene für die erste Umformstufe. Die Erhöhung findet unter bestimmten Umständen bei einer Abnahme des Einbauraums statt, die dann unbekannte Effekte verursacht. Dies ist bezogen auf die Bauteilgröße bei mittelgroßen Bauteilen der Fall. Dieser kurzfristige Effekt wird auf Basis im Betrieb erfasster Daten neu erlernt und für 150 Teile vorhergesagt. Für die zweite Stufe zeigt sich eine Gratbildung für eine geringe Verstellung bei großen Bauteilen. Die neu erlernten Modellstrukturen sind hinreichend genau, um diese Erkenntnisse auf den Prozessausgang abzubilden.



Abbildung 6.11: Detailansicht zur Störcharakterisierung bei zweistufiger Umformung (Stufe 1 links, Stufe 2 rechts)

Basierend auf dem kooperativen Prädiktionsschema, das alle Modellbestandteile berücksichtigt, wird das in Kapitel 4.3.4 eingeführte Optimierungsschema zur Kompensation von Anomalien für alle Bauteile zu jedem diskreten Zeitschritt gelöst. Eine Visualisierung der Ergebnisse für die Stellgrößenanpassung ist in Abbildung 6.12 dargestellt. Der blaue Verlauf zeigt die anfänglichen Stellgrößen für die analysierten 150 Bauteile ohne Kompensation von Anomalien. Diese wurden vom Produktionsmitarbeiter eingestellt. Die Stellgrößen, die primär durch Störungen beeinflusst werden, sind einerseits Reibungsparameter und andererseits Werkzeugeinstellungen. Die Verknüpfungen von aufeinanderfolgenden Prozessen variieren in besonderem Maße durch schwankende geometrische Merkmale der Bauteile. Diese sind exemplarisch in der Ansicht abgebildet.



Abbildung 6.12: Datengetriebene Kompensation von Anomalien auf der Grundlage des eingeführten Kontrollschemas

120

Der grüne Verlauf visualisiert die Anpassung der Stellgrößen durch den vorgestellten Ansatz zur Kompensation von Anomalien. Die Differenz zwischen den beiden Kurven entspricht der durch die Netzarchitekturen und das Optimierungsschema erzeugten Anpassungen für die dargestellten Stellgrößen. Jedes Paar von im Graphen einander zugeordneten Punkten entspricht einem diskreten und bauteilspezifischen Eingreifen in das Prozessgeschehen durch eine Anpassung der Stellgrößen gemäß der Vektordifferenz dieser beiden Punkte. Die korrigierten Stellgrößen entsprechen folglich dem grünen Verlauf. Die Regelung zeigt, dass der Einbauraum als Werkzeugparameter auf größere Werte eingestellt wird, was zu einer Reduktion der Gratbildung und damit zu einer Steigerung der Produktqualität führt. Hinsichtlich des Einbauraums erfolgt langfristig aufgrund von nicht-stationärem Verschleiß und kurzfristig aufgrund der entstehenden Reibung eine entsprechende Anpassung. Die Ausschussrate, die den Anteil der Bauteile beschreibt, die die Qualitätsanforderungen nicht erfüllen, konnte durch den Einsatz des Assistenzsystems versuchsreihenübergreifend reduziert werden. Zusätzlich konnte durch die Automatisierung des Vorgangs zur Stellgrößeneinstellung eine erhebliche Zeitersparnis erzielt werden.



Abbildung 6.13: Assistenzsystem bei der Otto Fuchs KG und Untersuchungen zu Berechnungszeiten verschiedener Recheneinheiten und Netzgrößen

Des Weiteren wurde eine Performanceanalyse durchgeführt, die zu interessanten Ergebnissen für kleine Netze führt, deren Training auf einer CPU schneller berechnet werden kann als auf einer leistungsstarken GPU. Der Vorteil der GPU kommt scheinbar erst dann zum Tragen, wenn die Netzwerkgröße deutlich zunimmt. Dieses Ergebnis ist in Abbildung 6.13 dargestellt. Erst eine ausreichende Netzgröße erlaubt eine effiziente Parallelisierung der Berechnungen. Die beschriebenen Funktionalitäten wurden, wie in Abschnitt 5.4 vorgestellt, in Form eines Assistenzsystems realisiert, das dem Produktionsmitarbeiter die wesentlichen Informationen zur Verfügung stellt, um den Prozess zu optimieren. Abbildung 6.13 zeigt neben der Performanceanalyse mit logarithmisch aufgetragener Rechenzeit die innerhalb der Produktionsanlage der Otto Fuchs KG abrufbare Benutzerschnittstelle.

## 6.3 Abgleich mit Anforderungen und Zielsetzung der Arbeit

Die eingangs beschriebene Evaluierungsstrategie sieht abschließend eine Bewertung der erzielten Ergebnisse vor. Auf Grundlage der empirischen Ergebnisse soll in diesem Abschnitt daher ein Abgleich mit den in Kapitel 1.4 aufgestellten Anforderungen sowie der in Kapitel 1.5 aufgezeigten Zielsetzung der Arbeit erfolgen. Dadurch soll mit Hilfe der Realisierung des Konzeptes und der damit erzeugten Ergebnisse der Nachweis erfolgen, dass das Konzept die zur Schließung der Forschungslücke nötigen Eigenschaften aufweist.

## (A1) Nutzung von bestehendem Expertenwissen im Rahmen der Modellbildung.

Der Modellierungsansatz baut erfolgreich auf Expertenwissen auf, indem die Ableitung neuronaler Metamodelle sowie deren Integration in ein Prozesskettenmodell eine Lösung für die aufgestellte Anforderung liefern. Die Erfüllung dieser Anforderung ist indirekt, durch die korrekte Abbildung der grundlegenden und bekannten physikalischen Zusammenhänge, aus den Ergebnissen ablesbar. Ein Beispiel dafür zeigt Abbildung 6.11. Hier zeigt das System die Notwendigkeit einer grundsätzlichen Erhöhung des Einbauraums zur Einhaltung einer konstanten Produktqualität bei größer werdenden Bauteilen an. Dies entspricht dem vom Experten erwarteten Verhalten.

## (A2) Berücksichtigung von Abhängigkeiten zwischen Prozessschritten.

Das Konzept liefert eine Lösung für die Berücksichtigung von Abhängigkeiten zwischen Prozessen sowie für die Integration derselben in die Prozessmodellierung in Form exogener Variablen. Die Erfüllung dieser Anforderung tritt direkt bei den Metriken zur Güte der Prozessmodelle zum Vorschein, die im Vergleich zu Prozessmodellen ohne Berücksichtigung von Abhängigkeiten erreicht werden können. Dazu sei Abbildung 6.9 als belegendes Ergebnis herangezogen. Hier wird sichtbar, dass zwischen Prozessmodellen mit und ohne Abhängigkeitsberücksichtigung unterschieden werden kann, wobei die erstgenannten verbesserte Konvergenzeigenschaften und eine deutlich erhöhte Stabilität und Robustheit zu Beginn des Lernvorgangs aufweisen.

### (A3) Detektion und Charakterisierung anomaler Prozessdynamiken.

Das entwickelte Konzept beinhaltet einen Mechanismus zur Detektion von Anomalien, der auf LSTM-basierten Autoencoder beruht und durch die einem Anwendung von Rekonstruktionsmetriken eine Charakterisierung der anomalen Prozessdynamiken erlaubt. Dies lässt sich anhand der zahlreichen Ergebnisse der Abbildungen 6.2 bis 6.4 nachvollziehen. Die Ergebnisse zeigen, dass eine zuverlässige Detektion stationärer und nicht-stationärer Anomalien sowie von Kurz- und Langzeitabhängigkeiten möglich ist. Die Anomalien können einem einzelnen oder einer Kombination aus Teilsystemen zugeordnet werden. Anhand der Metriken ist eine Quantifizierung des Anomalie-Verhaltens möglich.

### (A4) Ganzheitliche und lernfähige Prädiktion des Prozess- und Anomalie-Verhaltens.

Die vorliegende Arbeit liefert zu dieser Anforderung den Lösungsansatz eines kooperativen Prädiktionsschemas, das auf einer stetigen Interaktion von Prozess- und Anomalie-Modell in Form eines Austausches prädizierter Ausgänge beruht. Die Abbildungen 6.5 bis 6.7 sowie die Tabelle 6.2 zeigen Beispiele für eine Erfüllung dieser Anforderung. Sowohl stationäre, kurzfristig auftretende Anomalien als auch nicht-stationäre Langzeitanomalien konnten in einem Modell abgebildet und in eine kooperative Prädiktion des Gesamtprozessverhaltens mit variabel einstellbarem Horizont integriert werden. Es wurde dabei gefunden, das GRU Seq2Seq-Netze kurzfristig auftretende und LSTM Seq2Seq-Netze Langzeitanomalien besonders gut modellieren.

### (A5) Nutzerassistenz bei der Kompensation von Anomalien zur adaptiven Qualitätsregelung.

Das Konzept beinhaltet ein Kompensationsschema, das eine rekursive und bauteilspezifische Optimierung der Stellgrößen zur Unterdrückung von Anomalien und der Einhaltung der Produktqualität vorsieht. In Kapitel 5 wird dessen Realisierung in Form eines Web-basierten Assistenzsystems vorgestellt. Die empirischen Ergebnisse der Abbildungen 6.11 und 6.13 belegen die Nutzerunterstützung und die Erfüllung der Anforderung. Das System zeigt, wie in 6.10 aggregiert dargestellt, eine Reaktion auf verschiedene Störeffekte und passt daraufhin die Einstellungen der Produktionsmitarbeiter, wie in 6.12 zu sehen, adaptiv an, sodass Schwankungen unterbunden werden können und die Produktqualität eingehalten werden kann.

Zusammenfassend kann konstatiert werden, dass sowohl in der Stuttgarter Modellfabrik am Institut für Umformtechnik als auch an den Pilotanlagen der Otto Fuchs KG signifikante Ergebnisse zum Nachweis der Funktionsfähigkeit des Konzeptes sowie der Erfüllung der aufgestellten Anforderungen erzielt werden konnten. Der Mechanismus zur Detektion von anomalen Systemdynamiken konnte wirksam auf Basis von Daten der Otto Fuchs KG eingesetzt werden und hat Anomalien verschiedener Typen in multivariaten Zeitreihen mit hoher Präzision erfasst. Dadurch konnten Modelle trainiert und Vorhersagen über das Auftreten dieser Anomalien getätigt werden, die einerseits für kurzfristige Effekte und anderseits für Langzeitabhängigkeiten mit geringen Abweichungen vom realen Systemverhalten durchgeführt werden konnten. Dadurch ist die Grundlage zum erfolgreichen Einsatz des Kompensationsschemas geschaffen, das im Einsatz innerhalb der Prozessketten eine Steigerung hinsichtlich der Produktqualität sowie eine verbesserte Kontrolle und Stabilität des Prozesses sowie des Produktionsablaufs herbeigeführt hat. Dieser Umstand konnte empirisch auf die intelligente Anpassung verschiedener Stellgrößenkonstellationen sowie die durch das Assistenzsystem optimierte Stellstrategie zurückgeführt werden.

# 7 Schlussbetrachtung

Die Schlussbetrachtung dieser Arbeit umfasst neben einer Zusammenfassung der Ergebnisse einen Ausblick auf weiterführende Forschungsaktivitäten.

## 7.1 Zusammenfassung der Ergebnisse

Stetig steigende Anforderungen an die Produktqualität im Zusammenhang mit immer komplexer werdenden technischen Produkten stellt Unternehmen vor große Herausforderungen hinsichtlich der Realisierung einer effizienten und verlässlichen Qualitätssicherung. Die hohe Komplexität diskreter Fertigungsprozesse und der vernetzten Prozessketten, in denen sie eingesetzt werden, erschwert die Erklärung entstandener Fehler, die in vielen Fällen nicht auf die bekannten physikalischen Gesetzmäßigkeiten zurückgeführt werden können. Dem Prozessingenieur unbekannte anomale Systemdynamiken sind ursächlich für entlang mehrstufiger Prozessketten auftretender Dissonanzen und Schwankungen der Produktqualität. Ein tieferes Verständnis bislang unbekannter Zusammenhänge und Anomalien ist unerlässlich für eine übergreifende Kontrolle und Optimierung hochgradig vernetzter Prozessketten. Im Rahmen dieser Arbeit wurde dazu ein datengetriebener Ansatz entwickelt, der eine adaptive Qualitätsregelung auf Basis von LSTM-Netzen ermöglicht.

In der Industrie herrscht auch im Zeitalter von Industrie 4.0 noch die gängige Praxis einer, auf der Erfahrung der Produktionsmitarbeiter basierenden und damit über einen längeren Zeitraum zumeist konstanten Einstellung der Prozessstellgrößen vor. Um eine adaptive Einstellung zur Kompensation auftretender Anomalien zu ermöglichen, wurde der Stand der Wissenschaft und Technik hinsichtlich bestehender Verfahren zur Qualitätsregelung und Störgrößenunterdrückung untersucht. Neben adaptiven und modellbasierten prädikativen Ansätzen wurden zusätzlich diskrete bauteilbezogene Regelungsansätze sowie Optimierungsverfahren für mehrstufige und flexible Prozessketten bzw. Fertigungssysteme in die Betrachtung einbezogen. Dabei konnte festgestellt werden, dass kein Ansatz verfügbar ist, der die bestehenden Anforderungen gänzlich erfüllt. Weiterhin haben komplexere neuronale Netzarchitekturen wie LSTM oder Seq2Seq noch keinen bedeutenden Einzug in automatisierungstechnische Kontroll- und Assistenzsysteme erhalten, sodass Defizite hinsichtlich der Modellierung nichtlinearer zeitvarianter Prozesse bestehen.

In der vorliegenden Arbeit wurde daher ein Ansatz für eine adaptive Qualitätsregelung mehrstufiger Prozessketten konzipiert, der einen Mechanismus zur Detektion und Kompensation unbekannter anomaler Systemdynamiken integriert, um eine Optimierung der Produktqualität zu erreichen. Das Konzept sieht dazu zunächst eine Nutzung von in Form numerischer Prozessmodelle vorliegendem Expertenwissen durch eine Ableitung von neuronalen Metamodellen vor. Diese werden in einen gerichteten Graphen integriert, der eine Modellierung

mehrstufiger Prozessketten ermöglicht. Um Abhängigkeiten zu integrieren, werden die Modelle um exogene Variablen erweitert. Um darüber hinaus die unbekannten anomalen Prozessdynamiken zu berücksichtigen, werden Ansätze für deren Detektion und Modellierung entwickelt. Der konzipierte Mechanismus zur Detektion unbekannter Anomalien basiert auf einem unüberwachten Lernansatz mit LSTM-Autoencodern und ermöglicht eine Charakterisierung von Anomalie-Verhalten mit Hilfe der Rekonstruktionsmetriken. Die gewonnenen Anomalie-Daten werden für das Training von Anomalie-Modellen verwendet und anschließend in einem kooperativen Prädiktionsschema eingesetzt. Die neuronalen Anomalie-Modelle basieren auf Seq2Seq-LSTM-Netzen, die über einen effizienten Kopiermechanismus zwischen dem En- und Decoder verfügen. Das Prädiktionsschema sieht eine direkte Interaktion von Prozess- und Anomalie-Modell durch den iterativen Austausch prädizierter Ausgänge vor und erzielt dadurch eine verbesserte Vorhersagegenauigkeit. Schema wird Das prozesskettenübergreifend in einer rekursiven, bauteilspezifischen und prädikativen Optimierung zur Bestimmung der optimalen Stellgrößen verwendet. Durch die Berücksichtigung des identifizierten Anomalie-Verhaltens bei der Optimierung werden die Stellgrößen so angepasst, dass eine Kompensation der Anomalien realisiert werden kann.

Das Konzept wurde in Form eines Web-basierten Assistenzsystems realisiert. Es deckt den gesamten Regelkreis von der Datenerfassung aus Sensoren und Steuerungen bis hin zur automatisierten Einstellung der Stellgrößen an den Anlagen ab. Neben verschiedenen Möglichkeiten des Monitorings und der Rückverfolgung einzelner Bauteile entlang der Prozessketten wurden Möglichkeiten zur Beurteilung von detektierten Anomalien sowie zur automatisierten Kontrolle der Prozesse geschaffen. Das System ist als Private Cloud konzipiert und steht den Produktionsmitarbeitern an der Anlage genauso zur Verfügung wie dem Produktionsplaner oder dem Prozessingenieur. Die durch die Umsetzung des Gesamtsystems erbrachte Leistung umfasst neben den genannten Aspekten die Beherrschung der Komplexität der Informationsflüsse sowie der Prozessdynamiken der mehrdimensionalen Prozessketten. Die Verwaltung großer Datenmengen sowie die Gewährleistung der Datensicherheit sind als weitere integrale Bestandteile zu nennen. Eine Evaluierung konnte anhand von drei Prozessketten der Massivumformung, an der Universität Stuttgart sowie bei der Otto Fuchs KG, durchgeführt werden. Am Institut für Umformtechnik der Universität Stuttgart wurde eine Modellfabrik in Kooperation beider Institute aufgebaut. Dies umfasste die Konzeption und Realisierung mehrerer Systeme und Komponenten, von der Sensor- und Aktorebene bis zur übergeordneten Cloudebene. Dadurch war eine Durchführung von zahlreichen Versuchen im Feld sowie eine Evaluierung aller Funktionalitäten des Systems möglich. Hinsichtlich aller Evaluierungsszenarien konnten empirische Ergebnisse zur Detektion von Anomalien, zur Prädiktion sowie zur Kompensation derselben erzielt werden. Auf Basis realer Datenmengen im Terabyte-Bereich sowie ausgewählter Testläufe konnten die zentralen Beiträge dieser Arbeit in Form einer erhöhten Genauigkeit in der Vorhersage des Prozessverhaltens, einer adaptiven Anpassung von Stellgrößen sowie einer Optimierung der Produktqualität erzielt werden.
## 7.2 Ausblick auf weiterführende Forschungsaktivitäten

Im Rahmen der Arbeit wurden zahlreiche weitere Ansatzpunkte und Stoßrichtungen in der Forschung ausgemacht, die für weiterführende Untersuchungen zu interessanten Ergebnissen führen könnten. So wurde sich in dieser Arbeit auf die Modellierung vorwärts gerichteter Abhängigkeiten entlang von Prozessketten fokussiert, die direkt mit dem Materialfluss und den Bauteileigenschaften verknüpft sind. Eine interessante Möglichkeit wäre hingegen die Integration potentiell rückwärts gerichteter Abhängigkeiten, die bspw. über Korrelationsuntersuchungen identifiziert werden könnten. Dadurch könnte dem Ziel der möglichst vollständigen Abbildung aller, für die resultierende Produktqualität relevanter Einflüsse näher gerückt werden. Der verfolgte Grey-Box-Ansatz aus der Kombination von Expertenwissen und datengetriebenen Ansätzen in Form von neuronalen Netzen hätte den Einsatz von Regularisierungsmethoden erlaubt. Diese gehen mit einer Vorstrukturierung von Netzen einher, um vorhandenes Vorwissen zu implementieren, das im Rahmen eines Trainings nicht abgeändert werden darf. Regularisierungsmethoden könnten zu einer verbesserten Symbiose aus Expertenwissen und Daten beitragen und stellen daher eine interessante Stoßrichtung für weitere Untersuchungen dar. Da es sich insgesamt um eine empirisch geprägte Arbeit handelt, könnten weitere Kopiermechanismen hinsichtlich der Seq2Seq-Netze sowie weitere Modifikationen von LSTModer GRU-Zellen getestet werden.

## 8 Literaturverzeichnis

- Lu X., und Huang M., "Modeling, Analysis and Control of Hydraulic Actuator for Forging", Springer, Singapore, 2018.
- [2] Lee J., Davari H., Singh J., und Pandhare V., "Industrial Artificial Intelligence for industry 4.0-based manufacturing systems", Manufacturing Letters, Vol. 18, S. 20–23, 2018.
- [3] Polyblank J.A., Allwood J.M., und Duncan S.R., "Closed-loop control of product properties in metal forming: A review and prospectus", Journal of Materials Processing Technology, Vol. 214, Issue 11, S. 2333–2348, 2014.
- [4] Allwood J.M., Duncan S.R., Cao J., Groche P., Hirt G., Kinsey B., Kuboki T., Liewald M., Sterzing A., und Tekkaya A.E., "Closed-loop control of product properties in metal forming", CIRP Annals Vol. 65, Issue 2, S. 573–596, 2016.
- [5] Bonte M.H.A., van den Boogaard A.H., und Huétink J., "A Metamodel Based Optimisation Algorithm for Metal Forming Processes", Advanced methods in material forming, Springer, Berlin, S. 55–72, 2007.
- [6] Plett G.L., "Adaptive inverse control of linear and nonlinear systems using dynamic neural networks", IEEE Transactions on Neural Networks, Vol. 14, Issue 2, S. 360–376, 2003.
- [7] Schroeder D., und Buss M., "Intelligente Verfahren. Identifikation und Regelung Nichtlinearer Systeme", Springer Vieweg, 2. Auflage, Berlin, Heidelberg, 2018.
- [8] Lu X., und Huang M., "Modeling, Analysis and Control of Hydraulic Actuator for Forging", Springer, Singapore, 2018.
- [9] Liewald M., Karadogan C., Felde A., und Lodwig R., "Entwicklung und Integration digitaler Technologien in Prozessfolgen der Massivumformung", Neuere Entwicklungen in der Massivumformung, S. 257, 2017.
- [10] Chen W.H., "Disturbance Observer Based Control for Nonlinear Systems", IEEE/ASME Transactions on Mechatronics, Vol. 9, Issue 4, S. 706–710, 2004.
- [11] Shim H., Park G., Joo Y., Back J., und Jo N.H., "Yet another tutorial of disturbance observer: robust stabilization and recovery of nominal performance", Control Theory and Technology, Vol. 14, Issue 3, S. 237–249, 2016.
- [12] Ding S., Chen W.H., Mei K., und Murray-Smith D.J., "Disturbance Observer Design for Nonlinear Systems Represented by Input–Output Models", IEEE Transactions on Industrial Electronics, Vol. 67, Issue 2, S. 1222–1232, 2020.
- [13] Li J., Wu C., Li S., Yang J., und Li S., "Optimal disturbance rejection control approach based on a compound neural network prediction method", Journal of Process Control, Vol. 24, Issue 10, S. 1516–1526, 2014.
- [14] Vatankhah B., und Farrokhi M., "Nonlinear model-predictive control with disturbance rejection property using adaptive neural networks", Journal of the Franklin Institute, Vol. 354, Issue 13, S. 5201–5220, 2017.

- [15] Chen S.W., Wang T., Atanasov N., Kumar V., und Morari M., "Large Scale Model Predictive Control with Neural Networks and Primal Active Sets", arXiv:1910.10835, 2019.
- [16] Kruse R., Borgelt C., Braune C., Klawonn F., Moewes C., und Steinbrecher M., "Computational Intelligence. Eine methodische Einführung in künstliche neuronale Netze, evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze", Springer Vieweg, 2. Auflage Computational Intelligence, Wiesbaden, 2015.
- [17] Solovyeva E.B., "Types of recurrent neural networks for non-linear dynamic system modelling", Proceedings of 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), 24.-26. Mai, St. Petersburg, Russland, S. 252–255, 2017.
- [18] Hochreiter S., und Schmidhuber J., "Long short-term memory", Neural Computation, Vol. 9, Issue 8, S. 1735–1780, 1997.
- [19] Sutskever I., Vinyals O., und Le Q.V., "Sequence to Sequence Learning with Neural Networks", Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS), Vol. 2, S. 3104 - 3112, arXiv:1409.3215, 2014.
- [20] Jaitly N., Le Q.V., Vinyals O., Sutskever I., Sussillo D., und Bengio S., "An Online Sequence-to-Sequence Model Using Partial Conditioning", Advances in Neural Information Processing Systems 29 (NIPS), S. 5067–5075, 2016.
- [21] Maschler B., Jazdi N., und Weyrich M., "Maschinelles Lernen für intelligente Automatisierungssysteme mit dezentraler Datenhaltung am Anwendungsfall Predictive Maintenance", VDI-Berichte 2351 (2019), S. 739-751, 2019.
- [22] Ashtari Talkhestani B., Jung T., Lindemann B., Sahlab N., Jazdi N., Schloegl W., und Weyrich M., "An architecture of an Intelligent Digital Twin in a Cyber-Physical Production System", at - Automatisierungstechnik, Band 67, Heft 9, S. 762–782, 2019.
- [23] Müller T., Jazdi N., Schmidt J.P., und Weyrich M., "Cyber-Physical Production Systems: enhancement with a self-organized reconfiguration management", Procedia CIRP, Vol. 99, Issue 3, S. 549-554, 2021.
- [24] Lindemann B., Karadogan C., Jazdi N., Liewald M., und Weyrich M., "Cloud-based Control Approach in Discrete Manufacturing Using a Self-Learning Architecture", IFAC-PapersOnLine, Vol. 51, Issue 10, S. 163–168, 2018.
- [25] Liewald M., Karadogan C., Lindemann B., Jazdi N., und Weyrich M., "On the tracking of individual workpieces in hot forging plants", CIRP Journal of Manufacturing Science and Technology, Vol. 22, S. 116–120, 2018.
- [26] Lu X., und Huang M., "System-Decomposition Based Multi-level Control Approach", Modeling, Analysis and Control of Hydraulic Actuator for Forging, Springer, Singapore, S. 191–209, 2018.
- [27] Lindemann B., Fesenmayr F., Jazdi N., und Weyrich M., "Anomaly detection in discrete manufacturing using self-learning approaches", Procedia CIRP, Vol. 79, S. 313–318, 2019.

- [28] Lindemann B., Jazdi N., und Weyrich M., "Anomaly detection and prediction in discrete manufacturing based on cooperative LSTM networks", 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), S. 1003-1010, 2020.
- [29] Tonne J., "Robust model predictive control for large-scale manufacturing systems subject to uncertainties", Dissertation, Kassel University Press GmbH, Kassel, 2018.
- [30] Rau M., "Nichtlineare modellbasierte pr\u00e4diktive Regelung auf Basis lernf\u00e4higer Zustandsraummodelle", Dissertation, Universit\u00e4tsbibliothek der TU M\u00fcnchen, M\u00fcnchen, 2003.
- [31] Sturm M., "Neuronale Netze zur Modellbildung in der Regelungstechnik", Dissertation, Universitätsbibliothek der TU München, München, 1999.
- [32] Hauth J., "Grey-Box Modelling for Nonlinear Systems", Dissertation, Technische Universität Kaiserslautern, Kaiserslautern, 2008.
- [33] Krüger T., "Zur Anwendung neuronaler Netzwerke in adaptiven Flugregelungssystemen", Dissertation, Logos Verlag Berlin, Technische Universität Braunschweig, 2012.
- [34] Kruse R., Borgelt C., Braune C., Klawonn F., Moewes C., und Steinbrecher M., "Computational Intelligence. Eine methodische Einführung in künstliche neuronale Netze, evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze", Springer Vieweg, 2. Auflage Computational Intelligence, Wiesbaden, 2015.
- [35] Kluge F., "Etymologisches Wörterbuch der deutschen Sprache", de Gruyter, 23. Auflage, Berlin, 1999.
- [36] Zilles K., Tillmann B.N., "Anatomie", Springer-Lehrbuch, Springer, Berlin, 2010.
- [37] Zimek A., und Schubert E., "Outlier Detection", Encyclopedia of Database Systems, Springer, New York, NY, S. 1–5, 2019.
- [38] Chandola V., Banerjee A., und Kumar V., "Anomaly detection: A survey", ACM Computing Surveys, Vol. 41, Issue 3, S. 1–58, 2009.
- [39] Munir M., Erkel S., Dengel A., und Ahmed S., "Pattern-Based Contextual Anomaly Detection in HVAC Systems", Proceedings of the 17th IEEE International Conference on Data Mining Workshops, IEEE Computer Society, S. 1066–1073, 2017.
- [40] Kim T.Y., und Cho S.B., "Web traffic anomaly detection using C-LSTM neural networks", Expert Systems with Applications, Vol. 106, S. 66–76, 2018.
- [41] Jiang Y., Zeng C., Xu J., und Li T., "Real time contextual collective anomaly detection over multiple data streams", Semantic Scholar, Corpus ID: 18868065, 2014.
- [42] Kufieta K., "Nichtlineare adaptive Regelung mit neuronalen Netzen im unbemannten Flugversuch", Dissertation, Technische Universität Braunschweig, Braunschweig, 2015.
- [43] Hochreiter S., "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions", International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol. 6, No. 2, S. 107–116, 1998.

- [44] Pascanu R., Mikolov T., und Bengio Y., "On the difficulty of training Recurrent Neural Networks", Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, JMLR: W&CP, Vol. 28., S. 1310-1318, 2013.
- [45] Cho K., van Merrienboer B., Gulcehre C., Bahdanau D., Bougares F., Schwenk H., und Bengio Y., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation", Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, S. 1724-1734, 2014.
- [46] Hinton G.E., und Zemel R.S., "Autoencoders, Minimum Description Length and Helmholtz Free Energy", Proceedings of the 6<sup>th</sup> International Conference on Neural Information Processing Systems (NIPS), S. 3–10, 1993.
- [47] Zhou Y., Arpit D., Nwogu I., und Govindaraju V., "Is Joint Training Better for Deep Auto-Encoders?", arXiv:1405.1380, 2015.
- [48] Luong M.T., Le Q.V., Sutskever I., Vinyals O., und Kaiser L., "Multi-task Sequence to Sequence Learning", arXiv:1511.06114, 2015.
- [49] Kuznetsov V., und Mohri M., "Time series prediction and online learning", 29th Annual Conference on Learning Theory, PMLR, Columbia University, New York, NY, Band 49, S. 1190–1213, 2016.
- [50] Lai G., Chang W.C., Yang Y., und Liu H., "Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks", arXiv:1703.07015, 2017.
- [51] Han M., Fan M., und Xi J., "Study of Nonlinear Multivariate Time Series Prediction Based on Neural Networks", Advances in Neural Networks, 2<sup>nd</sup> International Symposium on Neural Networks (ISNN), Springer Berlin Heidelberg, Band 3497, S. 618–623, 2005.
- [52] Lughofer E., Zavoianu A.C., Pollak R., Pratama M., Meyer-Heye P., Zörrer H., Eitzinger C., und Radauer T., "Autonomous supervision and optimization of product quality in a multi-stage manufacturing process based on self-adaptive prediction models", Journal of Process Control, Vol. 76, S. 27–45, 2019.
- [53] Guo T., Xu Z., Yao X., Chen H., Aberer K., und Funaya K., "Robust Online Time Series Prediction with Recurrent Neural Networks", 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), S. 816–825, 2016.
- [54] Schroeder D., und Buss M., "Intelligente Verfahren. Identifikation und Regelung Nichtlinearer Systeme", Springer Vieweg, 2. Auflage, Berlin, Heidelberg, 2018.
- [55] Nakao M., Ohnishi K., und Miyachi K., "A Robust decentralized joint control based on interference estimation", Proceedings of the IEEE International Conference on Robotics and Automation, S. 326-331, 1987.
- [56] Chen W.H., Ballance D.J., Gawthrop P.J., und O'Reilly J., "A nonlinear disturbance observer for robotic manipulators", IEEE Transactions on Industrial Electronics, Vol. 47, Issue 4, S. 932–938, 2000.
- [57] Schmidhuber J., "Deep learning in neural networks: an overview", Neural Networks, Vol. 61, S. 85–117, 2015.

- [58] Günnemann-Gholizadeh N., "Machine Learning Methods for Detecting Rare Events in Temporal Data", Dissertation, Technische Universität München, München, 2018.
- [59] Taylor A., Leblanc S., Japkowicz N., "Anomaly Detection in Automobile Control Network Data with Long Short-Term Memory Networks", 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), S. 130–139, 2016.
- [60] Malhotra P., Vig L., Shrof G., und Agarwal P., "Long Short Term Memory Networks for Anomaly Detection in Time Series", Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), S.89 -94, 2015.
- [61] Zhou C., und Paffenroth R.C., "Anomaly Detection with Robust Deep Autoencoders", Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, S. 665–674, 2017.
- [62] Fernando T., Denman S., Sridharan S., und Fookes C., "Soft + Hardwired Attention: An LSTM Framework for Human Trajectory Prediction and Abnormal Event Detection", Neural Networks, Vol. 108, S. 466-478, 2018.
- [63] Loganathan G., Samarabandu J., und Wang X., "Sequence to Sequence Pattern Learning Algorithm for Real-Time Anomaly Detection in Network Traffic", IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), S. 1–4, 2018.
- [64] Zenati H., Foo C.S., Lecouat B., Manek G., und Chandrasekhar V.R., "Efficient GAN-Based Anomaly Detection", arXiv:1802.06222, 2018.
- [65] Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., und Bengio Y., "Generative Adversarial Nets", Advances in Neural Information Processing Systems 27, S. 2672–2680, 2014.
- [66] Ahmed M., Naser Mahmood A., und Hu J., "A survey of network anomaly detection techniques", Journal of Network and Computer Applications, Vol. 60, S. 19–31, 2016.
- [67] Dimopoulos G., Barlet-Ros P., Dovrolis C., und Leontiadis I., "Detecting network performance anomalies with contextual anomaly detection", 2017 IEEE International Workshop on Measurement and Networking (M&N), S. 1–6, 2017.
- [68] Bontemps L., van Cao L., McDermott J., und Le-Khac N.A., "Collective Anomaly Detection based on Long Short Term Memory Recurrent Neural Network", arXiv: 1703.09752, 2017.
- [69] Shivani, Sandhu K.S., und Nair A.R., "A Comparative Study of ARIMA and RNN for Short Term Wind Speed Forecasting", 10<sup>th</sup> International Conference on Computing, Communication and Networking Technologies (ICCCNT), S. 1–7, 2019.
- [70] Williams R.J., und Zipser D., "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks", Neural Computation, Vol. 1, No. 2, S. 270–280, 1989.
- [71] Elman J., "Finding structure in time", Cognitive Science, Vol. 14, Issue 2, S. 179–211, 1990.

- [72] Jordan M.I., "Attractor dynamics and parallelism in a connectionist sequential machine", Artificial Neural Networks: Concept learning, IEEE Computer Society, neural networks technology series, IEEE Computer Society Press, S. 112-127, 1990.
- [73] Parlos A., Atiya A., Chong K., Tsai W., und Fernandez B., "Recurrent multilayer perceptron for nonlinear system identification", IJCNN International Joint Conference on Neural Networks, Vol. 2, S. 537–540, 1991.
- [74] Han H.G., Zhang L., Hou Y., und Qiao J.F., "Nonlinear Model Predictive Control Based on a Self-Organizing Recurrent Neural Network", IEEE Transactions on Neural Networks and Learning Systems, Vol. 27, No. 2, S. 402–415, 2016.
- [75] Motazedian Z., und Safavi A.A., "Nonlinear and Time Varying System Identification Using a Novel Adaptive Fully Connected Recurrent Wavelet Network", 27<sup>th</sup> Iranian Conference on Electrical Engineering (ICEE), IEEE, S. 1181–1187, 2019.
- [76] Metzger A., und Neubauer A., "Considering Non-sequential Control Flows for Process Prediction with Recurrent Neural Networks", 44<sup>th</sup> Euromicro Conference on Software Engineering and Advanced Applications (SEAA), IEEE, S. 268–272, 2018.
- [77] Wong W., Chee E., Li J., und Wang X., "Recurrent Neural Network-Based Model Predictive Control for Continuous Pharmaceutical Manufacturing", Mathematics, Vol. 6, Issue 11, 242, 2018.
- [78] Quan Y.S., und Chung C.C., "Approximate Model Predictive Control with Recurrent Neural Network for Autonomous Driving Vehicles", 58th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), IEEE, S. 1076–1081, 2019.
- [79] Chang S., Zhang Y., Han W., Yu M., Guo X., Tan W., Cui X., Witbrock M., Hasegawa-Johnson M., und Huang T.S., "Dilated Recurrent Neural Networks", 31st Conference on Neural Information Processing Systems (NIPS), arXiv:1710.02224, 2017.
- [80] Qin Y., Song D., Cheng H., Cheng W., Jiang G., und Cottrell G., "A Dual-Stage Attention-Based Recurrent Neural Network for Time Series Prediction", Proceedings of the 26<sup>th</sup> International Joint Conference on Artificial Intelligence (IJCAI), arXiv:1704.02971, 2017.
- [81] Wang Y., "A new concept using LSTM Neural Networks for dynamic system identification", American Control Conference (ACC), IEEE, S. 5324–5329, 2017.
- [82] Gonzalez J., und Yu W., "Non-linear system modeling using LSTM neural networks", IFAC-PapersOnLine, Vol. 51, Issue 13, S. 485–489, 2018.
- [83] Tax N., Verenich I., La Rosa M., und Dumas M., "Predictive Business Process Monitoring with LSTM Neural Networks", International Conference on Advanced Information Systems Engineering (CAiSE), 10253(2), S. 477–492, 2017.
- [84] Yan K., Li W., Ji Z., Qi M., und Du Y., "A Hybrid LSTM Neural Network for Energy Consumption Forecasting of Individual Households", IEEE Access, Vol. 7, S. 157633– 157642, 2019.
- [85] Li Y., und Cao H., "Prediction for Tourism Flow based on LSTM Neural Network", Procedia Computer Science, Vol. 129, S. 277–283, 2018.

- [86] Schwedersky B.B., Flesch R.C.C., und Dangui H.A.S., "Practical Nonlinear Model Predictive Control Algorithm for Long Short-Term Memory Networks", IFAC-PapersOnLine, Vol. 52, Issue 1, S. 468–473, 2019.
- [87] Terzi E., Farina M., und Scattolini R., "Model predictive control design for dynamical systems learned by Long Short-Term Memory Networks", arXiv:1910.04024, 2019.
- [88] Lanzetti N., Lian Y.Z., Cortinovis A., Dominguez L., Mercangoz M., und Jones C., "Recurrent Neural Network based MPC for Process Industries", 18th European Control Conference (ECC), IEEE, S. 1005–1010, 2019.
- [89] Chung J., Gulcehre C., Cho K., und Bengio Y., "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling", arXiv:1412.3555, 2014.
- [90] Britz D., Goldie A., Luong M.T., und Le Q., "Massive Exploration of Neural Machine Translation Architectures", Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, S. 1442-1451, 2017.
- [91] Cui Z., Ke R., Pu Z., und Wang Y., "Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-wide Traffic Speed Prediction", arXiv:1801.02143, 2018.
- [92] Ma X., Zhang J., Du B., Ding C., und Sun L., "Parallel Architecture of Convolutional Bi-Directional LSTM Neural Networks for Network-Wide Metro Ridership Prediction", IEEE Transactions on Intelligent Transportation Systems, Vol. 20, No. 6, S. 2278–2288, 2019.
- [93] Xue H., Huynh D.Q., und Reynolds M., "SS-LSTM: A Hierarchical LSTM Model for Pedestrian Trajectory Prediction", IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, S. 1186–1194, 2018.
- [94] Hu L., "Integrating Hierarchical Attentions for Future Subevent Prediction", IEEE Access, Vol. 8, S. 3106–3114, 2020.
- [95] Su Z., und Jiang J., "Hierarchical Gated Recurrent Unit with Semantic Attention for Event Prediction", Future Internet, Vol. 12, Issue 2, 39, 2020.
- [96] Villegas R., Yang J., Zou Y., Sohn S., Lin X., und Lee H., "Learning to Generate Longterm Future via Hierarchical Prediction", Proceedings of the 34<sup>th</sup> International Conference on Machine Learning, PMLR 70, 2017.
- [97] Chu K.F., Lam A.Y.S., und Li V.O.K., "Deep Multi-Scale Convolutional LSTM Network for Travel Demand and Origin-Destination Predictions", IEEE Transactions on Intelligent Transportation Systems, Vol. 21, No. 8, S. 3219-3232, 2020.
- [98] Huang C.J., und Kuo P.H., "A Deep CNN-LSTM Model for Particulate Matter (PM<sub>2.5</sub>) Forecasting in Smart Cities", Sensors, Vol. 18, Issue 7, 2220, 2018.
- [99] Kim T.Y., Cho S.B., "Predicting residential energy consumption using CNN-LSTM neural networks", Energy, Vol. 182, S. 72–81, 2019.

- [100] Kalchbrenner N., Danihelka I., und Graves A., "Grid Long Short-Term Memory", arXiv:1507.01526, 2015.
- [101] Gensler A., Henze J., Sick B., und Raabe N., "Deep Learning for solar power forecasting — An approach using AutoEncoder and LSTM Neural Networks", IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, S. 2858–2865. 2016.
- [102] Veličković P., Karazija L., Lane N.D., Bhattacharya S., Liberis E., Liò P., Chieh A., Bellahsen O., und Vegreville M., "Cross-modal Recurrent Models for Weight Objective Prediction from Multimodal Time-series Data", arXiv:1709.08073, 2017.
- [103] Tay Y., Phan M.C., Tuan L.A., und Hui S.C., "Learning to Rank Question Answer Pairs with Holographic Dual LSTM Architecture", Proceedings of the 40<sup>th</sup> International ACM SIGIR Conference on Research and Development in Information Retrieval, S. 695–704, 2017.
- [104] Xu P., Du R., und Zhang Z., "Predicting pipeline leakage in petrochemical system through GAN and LSTM", Knowledge-Based Systems, Vol. 175, S. 50–61, 2019.
- [105] Marino D.L., Amarasinghe K., und Manic M., "Building energy load forecasting using Deep Neural Networks", Proceedings of the IECON 2016 - 42<sup>nd</sup> Annual Conference of the IEEE Industrial Electronics Society, IEEE, S. 7046–7051, 2016.
- [106] Cherrier N., Castaings T., und Boulch A., "Deep Sequence-to-Sequence Neural Networks for Ionospheric Activity Map Prediction", Proceedings of the 24<sup>th</sup> International Conference on Neural Information Processing (ICONIP), Part V, Band 10638, S. 545– 555, 2017.
- [107] Du S., Li T., und Horng S.J., "Time Series Forecasting Using Sequence-to-Sequence Deep Learning Framework", 9<sup>th</sup> International Conference on Parallel Architectures, Algorithms and Programming (PAAP), Conference Publishing Services, IEEE Computer Society, S. 171–176, 2018.
- [108] Gu J., Lu Z., Li H., und Li V.O.K., "Incorporating Copying Mechanism in Sequence-to-Sequence Learning", Proceedings of the 54<sup>th</sup> Annual Meeting of the Association for Computational Linguistics, Vol. 1, S. 1631-1640, 2016.
- [109] Cinar Y.G., Mirisaee H., Goswami P., Gaussier E., Ait-Bachir A., und Strijov V., "Position-based Content Attention for Time Series Forecasting with Sequence-tosequence RNNs", arXiv:1703.10089, 2017.
- [110] Gehring J., Auli M., Grangier D., Yarats D., und Dauphin Y.N., "Convolutional Sequence to Sequence Learning", Proceedings of the 34<sup>th</sup> International Conference on Machine Learning, PMLR 70, 2017.
- [111] Park S.H., Kim B., Kang C.M., Chung C.C., und Choi J.W., "Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture", IEEE Intelligent Vehicles Symposium (IV), S. 1672-1678, 2018.
- [112] Krause B., Lu L., Murray I., und Renals S., "Multiplicative LSTM for sequence modelling", arXiv:1609.07959, 2016.

- [113] Chowdhary G., und Johnson E., "Concurrent learning for convergence in adaptive control without persistency of excitation", 49<sup>th</sup> IEEE Conference on Decision and Control (CDC), S. 3674–3679, 2010.
- [114] Wang H., Tan F., Sheng B., Bian J., und Pan T., "Run-to-run control for semiconductor manufacturing processes using extended state observer", Proceedings of the 28<sup>th</sup> Chinese Control and Decision Conference (CCDC), IEEE Industrial Electronics, S. 854–857, 2016.
- [115] Wang H.Y., Pan T.H., Wong D.S.H., und Tan F., "An Extended State Observer-Based Run to Run Control for Semiconductor Manufacturing Processes", IEEE Transactions on Semiconductor Manufacturing, Vol. 32, No. 2, S. 154–162, 2019.
- [116] Moyne J., "Run-to-Run Control in Semiconductor Manufacturing", Encyclopedia of Systems and Control, Band 20, Springer London, London, S. 1–7, 2014.
- [117] Wang K., und Han K., "A batch-based run-to-run process control scheme for semiconductor manufacturing", IIE Transactions, Vol. 45, No. 6, S. 658–669, 2013.
- [118] Zheng Y., Lin Q.H., Wang D.S.H., Jang S.S., und Hui K., "Stability and performance analysis of mixed product run-to-run control", Journal of Process Control, Vol. 16, Issue 5, S. 431–443, 2006.
- [119] Tan F., Pan T., Li Z., und Chen S., "Survey on Run-to-Run Control Algorithms in High-Mix Semiconductor Manufacturing Processes", IEEE Transactions on Industrial Informatics, Vol. 11, No. 6, S. 1435–1444, 2015.
- [120] Lee A.C., Pan Y.R., und Hsieh M.T., "Output Disturbance Observer Structure Applied to Run-to-Run Control for Semiconductor Manufacturing", IEEE Transactions on Semiconductor Manufacturing, Vol. 24, No. 1, S. 27–43, 2011.
- [121] Cano Marchal P., Gamez Garcia J., und Gomez Ortega J., "Application of Fuzzy Cognitive Maps and Run-to-Run Control to a Decision Support System for Global Set-Point Determination", IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 47, No. 8, S. 2256–2267, 2017.
- [122] Yeoh I.L., Reinhall P.G., Berg M.C., Chizeck H.J., und Seibel E.J., "Run-to-Run Optimization Control Within Exact Inverse Framework for Scan Tracking", Journal of Dynamic Systems and Measurement Control, Vol. 139, No. 9, 910111–9101112, 2017.
- [123] Hao W., und Duncan S., "Optimization of tool trajectory for Incremental Sheet Forming using closed loop control", IEEE International Conference on Automation Science and Engineering, IEEE, S. 779-784, 2011.
- [124] Endelt B., Tommerup S., und Danckert J., "A novel feedback control system Controlling the material flow in deep drawing using distributed blank-holder force", Journal of Materials Processing Technology, Vol. 213, Issue 1, S. 36–50, 2013.
- [125] Endelt B., "Design strategy for optimal iterative learning control applied on a deep drawing process", The International Journal of Advanced Manufacturing Technology, Vol. 88, Ausgabe 1-4, S. 3–18, 2017.

- [126] Endelt B., "Proposing a new iterative learning control algorithm based on a non-linear least square formulation - Minimising draw-in errors2", Journal of Physics: Conference Series, Vol. 896, Artikel.-Nr. 12036, 2017.
- [127] Thomsen A.N., Endelt B., und Kristiansen M., "Feedback control of laser forming using flattening simulations for error determination", IOP Conference Series: Materials Science and Engineering, 651, 12093, 2019.
- [128] Shi J., und Zhou S., "Quality control and improvement for multistage systems: A survey", IIE Transactions, Vol. 41, No. 9, S. 744–753, 2009.
- [129] Han K., und Wang K., "Coordination and control of batch-based multistage processes", Journal of Manufacturing Systems, Vol. 32, Issue 2, S. 372–381, 2013.
- [130] Qamsane Y., Tajer A., und Philippot A., "Distributed Supervisory Control Synthesis For Discrete Manufacturing Systems", IFAC-PapersOnLine, Vol. 49, Issue 12, S. 396–401, 2016.
- [131] Zou J., Chang Q., Arinez J., und Xiao G., "Data-driven modeling and real-time distributed control for energy efficient manufacturing systems", Energy, Vol. 127, S. 247– 257, 2017.
- [132] Tonne J., und Stursberg O., "Fast Robust Model Predictive Control for Nonlinear Jump Markov Systems", IFAC-PapersOnLine, Vol. 50, Issue 1, S. 9342–9348, 2017.
- [133] Frazzon E.M., Kück M., und Freitag M., "Data-driven production control for complex and dynamic manufacturing systems", CIRP Annals, Vol. 67, Issue 1, S. 515–518, 2018.
- [134] Pena P.N., Costa T.A., Silva R.S., und Takahashi R.H.C., "Control of Flexible Manufacturing Systems under model uncertainty using Supervisory Control Theory and evolutionary computation schedule synthesis", Information Sciences, Vol. 329, S. 491– 502, 2016.
- [135] González S.R., Mondragón I., Zambrano G., Hernandez W., und Montaña H., "Manufacturing Control Architecture for FMS with AGV: A State-of-the-Art", Proceedings of Advances in Automation and Robotics Research in Latin America, Springer, Vol. 13, S. 157–172, 2017.
- [136] Camacho E.F., und Bordons C., "Model Predictive control", Advanced Textbooks in Control and Signal Processing, Springer, London, 2007.
- [137] de Souza Neto E.A., Perić D., und Owen D.R.J., "Computational methods for plasticity: Theory and applications", John Wiley & Sons, Chichester, 2008.
- [138] Spyridon P., und Boutalis Y.S., "Adaptive One-Step Model Predictive Control Using Lyapunov Theory-Based Deep Neural Networks", The 26<sup>th</sup> Mediterranean Conference on Control and Automation, IEEE, S. 326–331, 2018.
- [139] Wang T., Gao H., und Qiu J., "A Combined Adaptive Neural Network and Nonlinear Model Predictive Control for Multirate Networked Industrial Process Control", IEEE Transactions on Neural Networks and Learning Systems, Vol. 27, No. 2, S. 416–425, 2016.

- [140] Heess N., Hunt J.J., Lillicrap T.P., und Silver D., "Memory-based control with recurrent neural networks", arXiv:1512.04455, 2015.
- [141] Bieker K., Peitz S., Brunton S.L., Kutz J.N., und Dellnitz M., "Deep Model Predictive Control with Online Learning for Complex Physical Systems", arXiv:1905.10094, 2019.
- [142] Yang J., Li S., Chen X., und Li Q., "Disturbance rejection of dead-time processes using disturbance observer and model predictive control", Chemical Engineering Research and Design, Vol. 89, Issue 2, S. 125–135, 2011.
- [143] Yang J., Li S., und Yu X., "Sliding-Mode Control for Systems With Mismatched Uncertainties via a Disturbance Observer", IEEE Transactions on Industrial Electronics, Vol. 60, No. 1, S. 160–169, 2013.
- [144] Li J., Li S., Chen X., und Yang J., "RBFNDOB-based neural network inverse control for non-minimum phase MIMO system with disturbances", ISA Transactions, Vol. 53, Issue 4, S. 983–993, 2014.
- [145] Chen M., Shi P., und Lim C.C., "Adaptive Neural Fault-Tolerant Control of a 3-DOF Model Helicopter System", IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 46, No. 2, S. 260–270, 2016.
- [146] Zheng Q., und Gao Z., "Predictive active disturbance rejection control for processes with time delay", ISA Transactions, Vol. 53, Issue 4, S. 873–881, 2014.
- [147] Yao J., und Deng W., "Active disturbance rejection adaptive control of uncertain nonlinear systems: theory and application", Nonlinear Dynamics, Vol. 89, No. 3, S. 1611–1624, 2017.
- [148] Lenz E., "Eine Methodik zur stochastischen adaptiven Qualitätsregelung", Dissertation, Technische Universität Darmstadt, Darmstadt, 2013.
- [149] González Sorribes A., und García Gil P., "A novel observer-predictor control for uncertain systems with unknown time-varying input and output delays", International Journal of Control, Vol. 94, Issue 6, S. 1630-1640, 2019.
- [150] Steege F.F., Groß H.M., Krabbes M., und Riedmiller M., "Einsatz adaptiver Lernverfahren zur Regelung industrieller Verbrennungsprozesse", Dissertation, Technische Universität Ilmenau, Univ.-Bibliothek, Univ.-Verl. Ilmenau, Ilmenau, 2015.
- [151] Fang Y., Fei J., und Ma K., "Neural network-based model reference adaptive control of active power filter based on sliding mode approach", 41<sup>st</sup> Annual Conference of the IEEE Industrial Electronics Society (IECON), IEEE, S. 31–36, 2015.
- [152] Wang F., Chen B., Lin C., Zhang J., und Meng X., "Adaptive Neural Network Finite-Time Output Feedback Control of Quantized Nonlinear Systems", IEEE Transactions on Cybernetics, Vol. 48, No. 6, S. 1839–1848, 2018.
- [153] Schnetter P., "Sliding-Mode-Lernverfahren für neuronale Netzwerke in adaptiven Regelungssystemen", Dissertation, Niedersächsisches Forschungszentrum für Luftfahrt, Technische Universität Braunschweig, Braunschweig, 2017.

- [154] Fei J., und Lu C., "Adaptive Sliding Mode Control of Dynamic Systems Using Double Loop Recurrent Neural Network Structure", IEEE Transactions on Neural Networks and Learning Systems, Vol. 29, No. 4, S. 1275–1286, 2018.
- [155] Chowdhary G., Wu T., Cutler M., Ure N.K., und How J., "Experimental Results of Concurrent Learning Adaptive Controllers", AIAA Guidance, Navigation, and Control Conference, S. 1130, 2012.
- [156] Krüger T., "Zur Anwendung neuronaler Netzwerke in adaptiven Flugregelungssystemen", Dissertation, Technische Universität Braunschweig, Braunschweig, 2012.
- [157] Parikh A., Kamalapurkar R., und Dixon W.E., "Integral concurrent learning: Adaptive control with parameter convergence using finite excitation", International Journal of Adaptive Control and Signal Processing, Vol. 33, No. 12, S. 1775–1787, 2019.
- [158] Stier W., "Methoden der Zeitreihenanalyse", Springer-Lehrbuch, Springer, Berlin, Heidelberg, 2001.
- [159] Lindemann B., Jazdi N., und Weyrich M., "Adaptive Quality Control for discrete largescale Manufacturing Systems subjected to Disturbances", 25<sup>th</sup> International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, S. 379-386, 2020.
- [160] Korn U., und Wilfert H.H., "Mehrgrößenregelungen. Moderne Entwurfsprinzipien im Zeit- und Frequenzbereich", 1. Auflage Verlag Technik, Springer, Wien, 1982.
- [161] Chen W.H., Yang J., Guo L., und Li S., "Disturbance-Observer-Based Control and Related Methods—An Overview", IEEE Transactions on Industrial Electronics, Vol. 63, No. 2, S. 1083–1095, 2016.
- [162] Klein P., "Zeitreihenvorhersage auf Basis von rekurrenten neuronalen Netzen am Beispiel einer diskreten Fertigung", Masterarbeit, Nr. 3101, IAS, Universität Stuttgart, 2020.
- [163] Hebb D., "The Organization of Behavior: A Neuropsychological Theory", John Wiley & Sons, Inc., 1949.
- [164] Chen G., "A Gentle Tutorial of Recurrent Neural Network with Error Backpropagation", arXiv:1610.02583, 2016.
- [165] Luenberger D.G., und Ye Y., "Linear and Nonlinear Programming", International Series in Operations Research & Management Science, Springer, Band 228, 2015.
- [166] Wilson P., und Mantooth H.A., "Model-Based Engineering for Complex Electronic Systems - Model-Based Optimization Techniques", Elsevier, S. 347–367, 2013.
- [167] Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. und Deutsches Forschungszentrum f
  ür K
  ünstliche Intelligenz GmbH, "K
  ünstliche Intelligenz -Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung", 2017.
- [168] Faul A., Jazdi N., und Weyrich M., "Approach to interconnect existing industrial automation systems with the Industrial Internet", 21<sup>st</sup> International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, S. 1–4, 2016.

- [169] Schoder P., "Entwicklung einer adaptiven LSTM Autoencoder Netzstruktur zur Analyse hochdimensionaler, industrieller Prozessdaten", Masterarbeit, Nr. 3038, IAS, Universität Stuttgart, 2019.
- [170] Turkington G., "Hadoop beginner's guide. Learn how to crunch big data to extract meaning from the data avalanche", Packt Publishing, Birmingham, UK, 2013.
- [171] Zaharia M., Chowdhury M., Das T., Dave A., Ma J., McCauly M., Franklin M.J., Shenker S., und Stoica I., "Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing", 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI), S. 15–28, 2012.
- [172] Miner D., "Hadoop. What You Need to Know", 1. Auflage, O'Reilly Media, Inc., 2016.
- [173] Spivey B. und Echeverria J., "Hadoop Security: Protecting your Big Data Platform", O'Reilly Media, Inc., 2015.
- [174] Spies D., "Entwicklung eines unüberwachten maschinellen Lernverfahrens zur Erkennung von Anomalien in Prozessdaten", Masterarbeit, Nr. 3018, IAS, Universität Stuttgart, 2019.
- [175] Körber J., "Entwicklung einer Cloud-basierten Softwareapplikation zur Realisierung des Human Machine Interfaces eines Assistenzsystems", Masterarbeit, Nr. 3019, IAS, Universität Stuttgart, 2019.
- [176] Simmel M., "Entwicklung einer Software-Anwendung zur Identifikation und logistischen Steuerung smarter Produkte", Masterarbeit, Nr. 2920, IAS, Universität Stuttgart, 2017.
- [177] Fesenmayr F., "Entwicklung eines adaptiven maschinellen Lernverfahrens zur Analyse von hochdimensionalen, industriellen Prozessdaten", Masterarbeit, Nr. 2962, IAS, Universität Stuttgart, 2018.
- [178] Lindemann B., Jazdi N., und Weyrich M., "Detektion von Anomalien zur Qualitätssicherung basierend auf Sequence-to-Sequence LSTM Netzen", at -Automatisierungstechnik, Band 67, Heft 12, S. 1058–1068, 2019.
- [179] An J., und Cho S., "Variational Autoencoder based Anomaly Detection using Reconstruction Probability", Semantic Scholar, Corpus-ID: 36663713, 2015.
- [180] Beck M., "Analyse von Prozessdaten zur Qualitätsoptimierung eines Produktionsprozesses basierend auf künstlichen neuronalen Netzen", Masterarbeit, Nr. 3077, IAS, Universität Stuttgart, 2020.
- [181] Zapp M., "Entwicklung eines lernfähigen Assistenzsystems zur Optimierung der industriellen Qualitätssicherung", Masterarbeit, Nr. 2997, IAS, Universität Stuttgart, 2018.
- [182] Engist R., "Reinforcement Learning für Cyber-Physische Produktionssysteme", Masterarbeit, Nr. 2982, IAS, Universität Stuttgart, 2018.