

Masterthesis
im Studiengang
Informatik Master

Digital Twin as a Service

Ressourcenmanagement mit Energiedaten aus
cyber-physischen Systemen

Referent : Prof. Dr. Christoph Reich

Koreferent : Philipp Ruf M. Sc.

Vorgelegt am : 25.09.2019

Vorgelegt von : Daniel Schönle
Matrikelnummer: 259382
Vogt-Dufner-Straße 3, 78120 Furtwangen
daniel.schoenle@hs-furtwangen.de

Abstract

Die Energiewende führt zu einer Paradigmenänderung. Der Zeitpunkt der Energieabnahme wird sich zunehmend an dem der Energieerzeugung orientieren. Die Steuerung des Energiebedarfs kann durch energieorientierte Produktionsplanung gesteigert werden. Dies erfordert eine Vorhersage des Energiebedarfs. Hierfür wird ein System entwickelt, das eine Modellierung mittels maschinellen Lernens nutzt. Die Datenbasis wird durch eine Vorgehensweise zur Abstrahierung von Fertigungsmaschinen erzeugt. Das System besteht aus gruppierten Microservices, es berücksichtigt die unterschiedlichen Anforderungen der Modelle an die Infrastruktur. Die Modelle sind in digitalen Zwillingen integriert, die als Dienst genutzt werden. Hierdurch ist eine effiziente Adaption von Änderungen an Fertigungsmaschine oder Modell-Methodik möglich. Eine exemplarische Anwendung der Abstraktionsmethode und der Modellierung mittels neuronalen Netzes demonstrieren die Umsetzbarkeit.

The energy transition in Germany leads to a shift of paradigm. Time of energy consumption will increasingly be oriented to that of energy production. Control of the energy demand can be increased by energy-oriented production planning. This requires a prediction of the energy demand. For this purpose, a system is developed which uses modelling by machine learning. A procedure for abstraction of production machines generates the data basis. The models are integrated in digital twins as services, following the microservice architecture. An exemplary application of the abstraction method and modelling by means of neural networks demonstrate the feasibility.

Inhaltsverzeichnis

Abstract	I
Inhaltsverzeichnis	III
Abbildungsverzeichnis	IX
Tabellenverzeichnis	XI
Listingsverzeichnis	XIII
Gleichungsverzeichnis	XV
Abkürzungsverzeichnis	XVII
1 Einleitung	1
1.1 Hintergrund	1
1.2 Zielsetzung und Vorgehensweise	3
1.3 Übersicht	4
2 Stand der Forschung	5
2.1 Cyber-Physische Systeme (CPS)	5
2.2 Digitale Zwillinge (DT)	9
2.3 Energievorhersage	11
2.3.1 Energieverbrauch in der Fertigung (EVR)	11
2.3.2 Energieeffizienz	12
2.3.3 Simulation des Energieverbrauchs	13
2.3.4 Modellierung des Energieverbrauchs (EV)	15
2.3.5 Neuronale Netze	19

2.4	Planungssysteme (PS)	21
2.4.1	Entscheidungsverfahren	21
2.4.2	Energieverbrauch als Entscheidungskriterium (PE)	22
3	Rahmenbedingungen und Anforderungen	29
3.1	Vorgehensweise	29
3.2	Festlegung der Domäne	31
3.3	Festlegung der Domäne für den Proof of Concept	31
3.4	Planungsverfahren	33
3.4.1	Statische Planung (StaPla)	33
3.4.2	Dynamisches Ressourcenmanagement (DynaRes)	34
3.4.3	Simulation eines dynamischen Ressourcenmanagements (Sim-DynaRes)	35
3.4.4	Variationen der Use Cases	35
3.5	Anforderungen an die Vorhersage	36
3.6	Anforderungen an die Architektur	37
3.7	Abstraktion	37
3.7.1	Produktionsauftrag	38
3.7.2	Produktionsmaschine	39
3.7.3	Energieaufwand	39
4	Modellierung des Energieaufwands	41
4.1	Aufbau der Datenbasis	41
4.1.1	Abstraktion des Produktionsauftrags	41
4.1.2	Abstraktion von Produktionsmaschinen	43
4.1.3	Abstraktion des Energieaufwands	43
4.1.4	Klassifizieren der Datenbasis	43
4.2	Vorgehensmodell: Datenschemablone	46

4.2.1	Entwicklung des Vorgehensmodells	47
4.3	Vorhersage des Energieaufwands	49
4.3.1	Modellierung mittels maschinellen Lernens	49
4.3.2	Architektur des neuronalen Netzes	50
4.3.3	Umsetzung der Architektur und Feature Extraktion	52
4.3.4	Anpassung des Modells	52
5	Entwicklung der Architektur	57
5.1	Abgeleitete Use Cases	57
5.2	Architekturentscheidungen	58
5.2.1	Microservices	60
5.2.2	Digitale Zwillinge	61
5.3	Ein Wald aus digitalen Zwillingen	61
5.4	Aufbau und Übersicht des Systems	62
5.4.1	Peripherie und Interaktion	62
5.4.2	Teilsystem Energie-Vorhersage	62
5.4.3	Teilsystem Modellierung	63
5.4.4	Teilsystem Berechnung	63
5.4.5	Teilsystem Container Verwaltung	64
5.5	Use Case Multiprediction	64
5.6	Evaluation	66
6	Anwendung des Vorgehensmodells	67
6.1	Domäne Fräsmaschinen	67
6.1.1	Simulation von Fräsmaschinen	67
6.1.2	Anwendung des Vorgehensmodells auf Fräsmaschinen	69
6.2	Domäne 3D-Druck	71
6.2.1	Zugrundeliegende Umgebung	71

6.2.2	Anwendung des Vorgehensmodells auf 3D-Druck	72
6.3	Evaluation und weitere Schritte	77
7	Exemplarische Energievorhersage	79
7.1	Vorgaben und Ziel	79
7.2	Maschinenbeschreibung	81
7.3	Vom Objekt zum Modell	81
7.4	Informationserfassung und Verarbeitung	83
7.5	Der Weg zur Prognose	84
7.6	Implementierung des Systems	85
7.7	Experimentelle Entwicklung der Architektur	85
7.8	Entwicklungsstufen der Vorhersage	86
7.8.1	Phase 0: Datenoptimierung	87
7.8.2	Phase 1: Evolution 1 (Basis)	87
7.8.3	Phase 1: Evolution 2 (Datenanalyse)	87
7.8.4	Phase 1: Evolution 3 (Reduktion)	89
7.8.5	Phase 1: Evolution 4 (maximale Abweichung)	89
7.8.6	Phase 2: Evo 5 (Tiefer)	90
7.8.7	Phase 2: Evo 6 (Breiter)	90
7.8.8	Phase 2: Evo 7 (Tiefer und Breiter)	90
7.8.9	Phase 2: Evo 8 (Datenbasis erweitern)	91
7.9	Evaluation	92
8	Fazit und Ausblick	93
8.1	Fazit	93
8.2	Ausblick	94
	Literaturverzeichnis	97
	Eidesstattliche Erklärung	109

A	Energieverbrauch	111
B	Ausgewählte 3D-Modelle	113
C	Cura-Druckprofil	115
D	Implementierung Datenaufbereitung	117
E	Implementierung Vorhersage	123
F	Anfrage Vorhersage	125

Abbildungsverzeichnis

Abbildung 1:	Kapitelübersicht	4
Abbildung 2:	Mechatronisches System [Ral17]	6
Abbildung 3:	Cyber-Physisches System [Ral17]	7
Abbildung 4:	Entwicklung der Einflussnahme auf den Energieverbrauch	12
Abbildung 5:	Einteilung der Simulationsmethoden, angelehnt an [Wen10]	14
Abbildung 6:	Netzwerkmodell zur Analyse des Einflusses von Prozessparametern, dabei steht w für weight und b für bias [QCR11]	20
Abbildung 7:	Strukturierte Übersicht der Rahmenbedingungen und Anforderungen	30
Abbildung 8:	Schematische Darstellung des FDM-Prozesses [PH18]	32
Abbildung 9:	Maschinenabstraktion	44
Abbildung 10:	Von der Abstraktion zum Modell	45
Abbildung 11:	Abtrennen von Modifikationen bei häufigem Nachjustieren	54
Abbildung 12:	Übersicht über die Teilsysteme und deren Microservices	59
Abbildung 13:	Kombinationen von Parametern	62
Abbildung 14:	Energy Prediction Use Case Multiprediction	65
Abbildung 15:	Simulation des Werkzeugs	68
Abbildung 16:	Simulation der Bearbeitung	68
Abbildung 17:	Proof of Concept	80
Abbildung 18:	Modell Deckel	81
Abbildung 19:	Modell Apfel [Diza]	82

Abbildung 20: Modell Vase. Neben dem primären Filament ist auch die Stütz- Struktur zu sehen.	82
Abbildung 21: Proof of Concept: Cyber-Schicht	85
Abbildung 22: Zeit für Druckaufträge in Sekunden	88
Abbildung 23: Verhältnis Zeit zu Energie von Druckaufträgen	89
Abbildung A.1: Anteil erneuerbarer Energie 2017[Sta17]	111
Abbildung A.2: Energieverbrauch nach Produktionsbereiche 2017 [AGE17]	111
Abbildung B.1: Modell Deckel, Ursprung [Dizb]	113
Abbildung B.2: Modelle von Schrauben und Muttern, Ursprung [Dizb]	113
Abbildung B.3: Modell Träger, Ursprung [Dizb]	113
Abbildung B.4: Modell Pfeiler	114
Abbildung B.5: Modell Apfel [Diza]	114
Abbildung B.6: Modell Vase	114

Tabellenverzeichnis

Tabelle 1:	Wissenschaftliche Arbeiten mit dem Schwerpunkt CPS	8
Tabelle 2:	Wissenschaftliche Arbeiten mit dem Schwerpunkt 'Digitaler Zwilling'	10
Tabelle 3:	Möglichkeiten der Einflussnahme auf den Energieverbrauch, angelehnt an Blesl et al. [BK13]	13
Tabelle 4:	Zentrale Arbeiten der modellbasierten Energievorhersage	18
Tabelle 5:	Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 1	24
Tabelle 6:	Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 2	25
Tabelle 7:	Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 3	26
Tabelle 8:	Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 4	27
Tabelle 9:	Anforderungen je Use Case	36
Tabelle 10:	Anforderungsvariationen der Use Cases	36
Tabelle 11:	Modellformate	42
Tabelle 12:	Schablone Fräsmaschine: Maschine	70
Tabelle 13:	Schablone Fräsmaschine: Materialien	70
Tabelle 14:	Schablone Fräsmaschine: Task	71
Tabelle 15:	Schablone Fräsmaschine: Umgebungsparameter *Wird nicht simuliert	71
Tabelle 16:	Schablone Fräsmaschine: Prognoseparameter *Wird nicht simuliert	71
Tabelle 17:	CAD-Software: 3D-Druck+Consumer Bereich	72

Tabelle 18: 3D-Druck-Software	72
Tabelle 19: Schablone Additiver 3D-Druck Consumer Bereich: Maschine	74
Tabelle 20: Schablone Additiver 3D-Druck Consumer Bereich: Werkzeug	74
Tabelle 21: Schablone Additiver 3D-Druck Consumer Bereich: Material	75
Tabelle 22: Schablone Additiver 3D-Druck Consumer Bereich: Umgebung	75
Tabelle 23: Schablone Additiver 3D-Druck Consumer Bereich: Task	76
Tabelle 24: Ausgewählte 3D-Druck-Modelle	83
Tabelle 25: Exemplarische Messwerte. Stromstärke in Ampere, Wirkleistung und Scheinleistung in Watt, sowie die Differenz von Wirkleistung und Scheinleistung in Prozent.	83
Tabelle 26: Verwendete Infrastruktur und Dauer der Prognose-Erstellung in Stunden und Minuten	86
Tabelle 27: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE) und Standardabweichung (in Klammern)	87
Tabelle 28: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)	88
Tabelle 29: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)	89
Tabelle 30: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)	90
Tabelle 31: Entwicklung der Vorhersagequalität von Evo 4 zu Evo 5	90
Tabelle 32: Entwicklung der Vorhersagequalität von Evo 5 zu Evo 6	90
Tabelle 33: Entwicklung der Vorhersagequalität von Evo 6 zu Evo 7	91
Tabelle 34: Entwicklung der Vorhersagequalität von Evo7 zu Evo8	91

Listings

C.1	Auszug aus einem Cura Druckprofil	115
D.1	Datenaufbereitung:Suche und Verarbeitung der Druckjobs	117
D.2	Datenaufbereitung:Erweiterung der Datenbasis	119
D.3	Datenaufbereitung:Druckjobklasse	120
E.1	Vohersage Evo4	123
F.1	Auszug aus einem Energy Prediction Request:Singlepredict	125

Gleichungsverzeichnis

2.1 Energieaufwand für Maschinenwerkzeug: Start-Phase Motor [LXL15]	16
2.2 Modell des Energieverbrauchs: Start-Phase Motor [LXL15]	16
2.3 Gleichung zu Modell des Energieverbrauchs: Start-Phase Motor [LXL15] . . .	16
2.4 Berechnung des Energieverbrauchs [SWRS19]	17
5.1 Anzahl digitaler Zwillinge bei freier Kombination	61
5.2 Beispielhafte Berechnung der Anzahl von digitaler Zwillinge	61

Abkürzungsverzeichnis

AM laser-assisted additive manufacturing

CAD computer-aided design, rechnerunterstütztes Konstruieren

CNC Computerized Numerical Control, computerisierte numerische Steuerung

CPPS Cyber-Physical Production System

CPS Cyber-Physical Space

DT Digital Twin

DyaRes Dynamische Ressourcenplanung

DZ Digitaler Zwilling

EVR Energieverbrauchs

FDM Fused Deposition Modeling /Fused filament fabrication

G-Code DIN/ISO-Programmierung für CNC-Maschinen

MADM Multi-Attribute-Decision-Making

M-Code Miscellaneous-Operations, standardisierte/herstellerspezifische Maschinenfunktionen

MP Maschinen-Profil

MSE mean squared error

MTConnect Shop-Floor-Protokoll des MTConnect Institute

NC-Code Numerical Control-Code

PLM Product Lifecycle Management

PLS Production Lot Size

PPS-System Produktionsplanungs- und Steuerungssystem

SCA Simscape

SEC Specific Energy Consumption

SimDynaRes Simulation einer Dynamischen Ressourcenplanung

StaPla Statische Produktionsplanung

UAV Unmanned Aerial Vehicle

UPN-Modell Umwandlungs-, Produktions-, Nebenanlagen-Modell

VMT Virtual Machine Tool

Wh Watt-Stunde

WP Werk-Profil

1. Einleitung

Fortschritte in der Energiewende führen neben zunehmender ökologischer Energieproduktion auch zu steigendem Aufwand in der Energieproduktion und Bereitstellung. Der Anteil nachhaltig erzeugter Energie nimmt zu, zugleich nimmt der Anteil bedarfsorientierter Erzeugung ab (siehe Abbildung A.1) [Sta17].

Immer weniger Energie wird zum Zeitpunkt des Bedarfs erzeugt. Eine Lösung ist die Energiespeicherung, die technologisch als auch faktisch vorangetrieben wird. Sie ist jedoch immer mit Energieverlusten behaftet und verbraucht selbst erhebliche Ressourcen [KBB⁺16]. Eine Lösung dafür stellt die Steuerung des Energieverbrauchs seitens des Verbrauchers dar. Somit reagiert der Verbrauch auf das aktuelle und voraussichtliche Energieangebot [MT16]. Durch den Einsatz neuer Technologien in der Informationstechnologie können Prognosen erstellt werden, deren Genauigkeit und Flexibilität Grundlage für eine energieorientierte Ressourcenplanung sind. Da der Energieverbrauch des verarbeitenden Gewerbes 39% des gesamten Energieverbrauchs aller wirtschaftlichen Aktivitäten ausmacht, bietet sich dieser Bereich zur Optimierung an (siehe Abbildung A.2) [AGE17].

1.1. Hintergrund

Planungs- und Ressourcenmanagement. In dieser Arbeit wird darunter die Verwaltung von Fertigungsmaschinen verstanden. Eingehende Aufträge werden in Tasks aufgeteilt und diese in einen Maschinenbelegungsplan überführt [Eve13]. Dabei kann die Maschine an sich eine Ressource sein, die durch den Task belegt wird. Die Dauer dieser Belegung wird daher zu Planungsbeginn festgelegt.

Forschungen im Ressourcenmanagement brachten Lösungen für Standardprobleme in der Planung, in dynamischen Änderungen und für Priorisierungen hervor. Entscheidend ist dabei eine genaue Abschätzung des zu erwarteten Ressourcenbedarfs und der Umgang mit deren Abweichungen im Zeitverlauf [KLL18].

Details und eine Übersicht über die aktuelle Forschung hierzu sind in Kapitel

2.4 zu finden.

Produktionsprozess. Ziel dieser Arbeit ist ein Vorhersage-System das unabhängig von Fertigungsverfahren eingesetzt werden kann. Zur Adaption ist jedoch Domänenwissen notwendig.

Im Produktionsprozess durchläuft ein Werkstoff eine Produktionskette. Sie besteht aus Fertigungsmaschinen die ein oder mehrere Werkzeuge nutzen, um den Werkstoff zu bearbeiten. Konkret durchläuft ein Stahl-Rohling eine Bearbeitung in einer Fräsmaschine, in der er mittels eines Fräsbohrers bearbeitet wird.

Es gibt eine Reihe an Fertigungsverfahren, wie das Biegen (Molding), Schweißen (Welding), Schleifen, Fräsen, sowie den 3D-Druck. Die eingesetzten Materialien variieren, so können Kunststoffe oder Metalle genutzt werden. Neben der Herstellungsdauer sind vor allem Qualität-Parameter von Interesse, wie die mechanische Belastbarkeit oder die Genauigkeit (Toleranz) [Löd16].

Details zum 3D-Druck finden sich in Kapitel 3.3.

Energie im Produktionsprozess. Der in dieser Arbeit betrachtete Energieaufwand umfasst die Energieaufnahme der Maschine während der Bearbeitung eines Werkstücks. Dies entspricht dem Energieaufwand zu Herstellung des Werkstücks. Verfahren und Details hierzu werden in Kapitel 2.3 vorgestellt.

Digitale Zwillinge. Um physische Systeme abzubilden, können Fertigungsmaschinen als virtuelle Systeme erzeugt werden. Das in dieser Arbeit entwickelte System ergänzt diese virtuellen Systeme um Autonomie und Self-Awareness. Diese interaktiven digitalen Repräsentanzen werden als digitale Zwillinge betrachtet. In Kapitel 2.2 wird auf Details eingegangen.

Cyber-physische Systeme. Sie sind eine Erweiterung von mechatronischen Systemen und enthalten eingebettete Geräte, Sensoren, Recheneinheiten und Konnektoren. Die zusätzlichen Subsysteme ermöglichen tiefer gehende Analysen und den Zugriff auf entfernte Systeme. Digitale Zwillinge können in diese Subsysteme integriert sein [CTT19]. Details befinden sich in Kapitel 2.1.

Energieorientierung. Ziel ist eine frühe Berücksichtigung der Energieaufwands des Produktionsprozesses. Bereits bei der Planung der Produktionsmaschinen bestehen dabei Betrachtungspunkte wie die Produktionsmaschine an sich, deren Bearbeitungswerkzeuge bis hin zur gesamten Produktionskette. Somit sind in der Planungsphase bereits Informationen über das Energieverhaltensverhalten notwendig, um den Produktionsprozess energieeffizient entwerfen zu können

[Scha].

Bereiche wie Standby-Zeiten, Heiz und Kühlkreisläufe, Technologiewahl und Kosten/Nutzen-Verhältnis können im Verlauf der Ressourcenplanung optimiert werden. Zum Zeitpunkt der Auftragsplanung wird der Aufwand aller Aufträge seriell betrachtet. Um auf kurzfristige Auftragsänderungen reagieren zu können, muss die Möglichkeit nachträglicher Planungsänderungen bestehen [Wei10].

Ein weiterer Aspekt der Energieorientierung ist die angebotsorientierte Planung. Während der Produktion werden Maßnahmen ergriffen, um den Energieverbrauch auf bevorstehende Änderungen der Energiekosten (Energieangebot) anzupassen [RD08].

1.2. Zielsetzung und Vorgehensweise

In dieser Arbeit wird ein System entwickelt, das Energiedaten zur Ressourcenplanung abschätzt, unter Verwendung von digitalen Zwillingen der Produktionsmaschinen. Auf Basis dieser Schätzung kann der zu erwartende Energieaufwand des Herstellungsprozesses ermittelt werden. Da die digitalen Zwillinge nicht auf Simulationen oder Modellen basieren, sondern durch maschinelles Lernen die Vorhersagen treffen, können sie sich an Veränderungen im Ablauf anpassen.

Teil der Ressourcenplanung ist die Entscheidung, welchen der möglichen Wege der Werkstoff durch die Prozesskette nimmt. Die Energie, die eine Produktionsmaschine für die Bearbeitung des Werkstücks aufwenden muss, variiert. Somit kann jeder Maschine ein spezifischer Energieaufwand und damit ein Grad der Energie-Effektivität zugeordnet werden. Bezogen auf Prozessketten, die sich über mehrere Maschinen erstrecken, entstehen komplexere Lösungen aufgrund kombinatorischer Verkettungen. Kapitel 2.4 beschreibt den aktuellen Forschungsstand dieser Problemstellung. Zu Fragestellungen der Entscheidungstheorie sei auf [Scha] verwiesen.

Eine Methode, die den zu erwartenden Energieaufwand anhand digitaler Zwillinge nutzt, um von den oben beschriebenen Vorteilen profitieren zu können, existiert bis dato nicht.

Zur Abstraktion der zugrundeliegenden Systeme wird eine Vorgehensweise entwickelt. Sie hat das Ziel die Bindung zwischen Domänen-Wissen und Data Science herzustellen.

Es wird eine Architektur vorgestellt, die ein Vorhersagesystem in Dienste gliedert.

Eine Reihe exponierter Dienste lassen sich zur Erzeugung von Vorhersagen in die Ressourcenplanung einhängen. Basis hierfür ist ein Produktionsauftrag, der in seine Tasks aufgeteilt wird. Die Vorhersagen über den Energieaufwand werden durch digitale Zwillinge erstellt, die den genutzten physischen Maschinen zugeordnet sind.

In der Planung der Tasks werden diese Daten den Tasks zugeordnet. Die Auswahl der Produktionsmaschine wird durch diese Energiedaten beeinflusst. Sie werden dabei nicht das einzige Kriterium sein, so werden Fertigungskosten und Fertigungstermine weiterhin hohe Relevanz haben.

1.3. Übersicht

Abbildung 1 gibt einen strukturierten Überblick über die Kapitel.

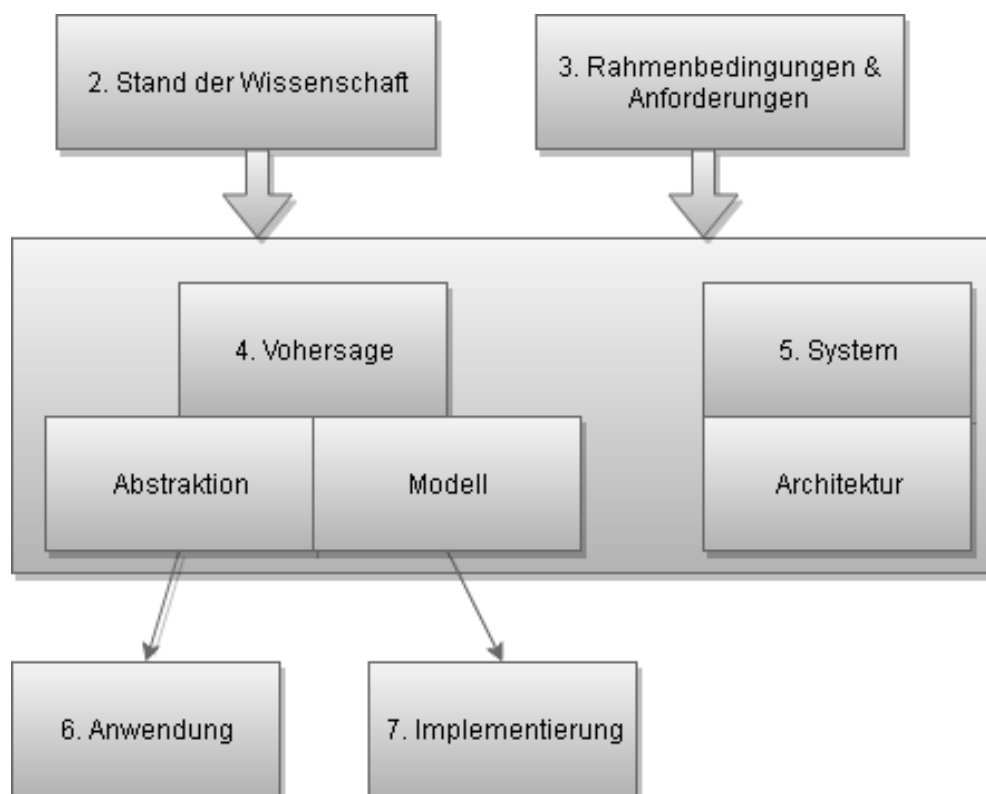


Abbildung 1.: Kapitelübersicht

2. Stand der Forschung

Dieses Kapitel beginnt mit der Betrachtung der Forschung in den Architekturbereichen des cyber-physischen Systems und den digitalen Zwillingen (DZ). Auch wenn sich nur wenige Arbeiten mit Energievorhersagen mittels DZ beschäftigen, dienen sie als Basis für Architekturentscheidungen. Die grundlegende Unterscheidung von Simulation und Modell ist Teil davon.

In den darauffolgenden Abschnitten des Kapitels stehen die Energievorhersagen im Mittelpunkt. Zum einen das Erstellen von Energievorhersagen anhand von Modellen, zum anderen das Planen der Produktion unter Berücksichtigung des Energieaufwands. Die wissenschaftlichen Arbeiten werden auf folgende Teilaspekte hin untersucht:

- Wie wird der Fertigungsprozess analysiert?
- Wie wird der Energieaufwand im Fertigungsprozess erfasst und verortet?
- Wie wird eine Vorhersage erstellt?
- In wie fern fließt die Vorhersage in die Produktionsplanung ein?

Weitgehend ausgeklammert bleibt der Bereich der Energieeffizienz.

2.1. Cyber-Physische Systeme (CPS)

Technologien die Kommunikation und Interaktion zwischen Maschinen, Menschen und peripheren Komponenten mittels Hardware und Software ermöglichen, werden unter dem Konzept der Cyber-Physischen-Systeme (CPS) zusammengefasst. [CLL⁺18]. Die Forschung hat sich bereits mit vielen Aspekten der CPS beschäftigt, Tabelle 1 gibt hierfür einen Überblick. Die Architekturen der CPS variieren, es existiert nur ein allgemeingültiges Konzept. Einfache Architekturen eines CPS bestehen aus den drei Ebenen physische Objekte, Modelle und Dienste [RA14], komplexere aus einem 5-Schicht Modell [LBK15] bis hin zu Layer-Modellen, bestehend aus Sensing, Networking, Service und Interface [XD19].

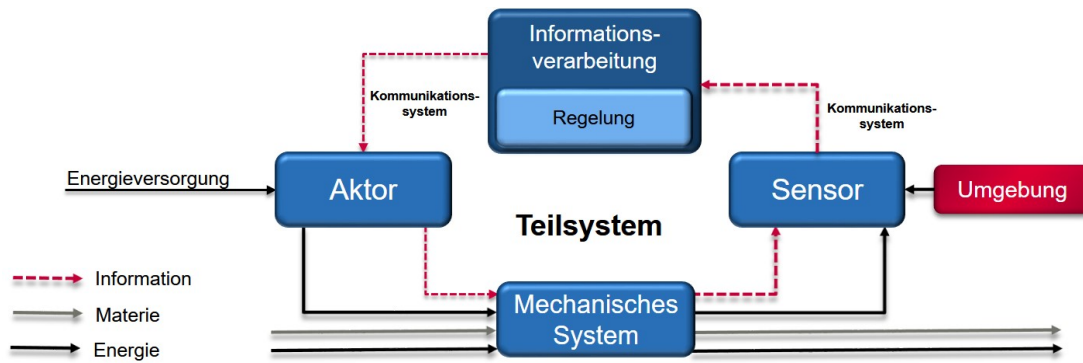


Abbildung 2.: Mechatronisches System [Ral17]

Basis eines CPS ist ein mechatronisches System, wie es in Abbildung 2 dargestellt ist. In Geräten, Maschinen, Gebäuden, Transportsystemen, Produktionssystemen oder medizinischen Systemen sind IT-Hardware und Software eingebettet, die als System physikalische Daten mittels Sensoren erfassen. Teil des Systems sind Aktoren, die physikalische Prozesse beeinflussen. Die Komponenten des lokalen Systems kommunizieren untereinander und mittels Mensch-Maschine-Schnittstelle auch mit Menschen [Ral17].

CPS erweitern das mechatronische System, wie in Abbildung 3 ersichtlich ist. Aufgezeichnete Daten werden ausgewertet und dauerhaft gespeichert. Die Interaktion der Systeme erweitert sich von reiner Reaktivität, auf aktiv und pro-aktives Vorgehen. Die Vernetzung ist nicht auf die lokale Umgebung beschränkt, die lokalen Systeme können mit entfernten Systemen kommunizieren. Durch die Anbindung an globale Netze können auch weltweit verfügbare Daten und Dienste genutzt werden [Ral17].

Die Architekturerweiterungen durch ein CPS spiegeln die gestiegenen Fähigkeiten wider. Diese führen zu einem Anstieg der Anforderungen in Bereichen wie Daten-, System- und Netzwerksicherheit [HLLL17]. Der Forschungsbereich ist umfangreich, ein Beispiel für Systemsicherheit ist die Arbeit von Javed et al. [JFS⁺19]. Darin wird eine Architektur erstellt die das CPS in Partitionen teilt, um die Sicherheit zu erhöhen.

Aufgrund der hohen Komplexität beschäftigen sich Arbeiten wie [CLL⁺18] mit Optimierungsmöglichkeiten. Es wird ein Manufacturing Execution System entwickelt, um Sensorwerte zu erfassen und damit das MTConnect-Protokoll zu ergänzen. Die Weiterentwicklung des bewährten Konzepts eines System Message Bus hin zu einem Production Message Bus ist Gegenstand von [MRR⁺18]. Den noch ausstehenden Schritt hin zu skalierbaren Lösungen untersuchen Caldarola et al. in Synchronizing physical and digital factory: Benefits and technical challenges [MCST19].

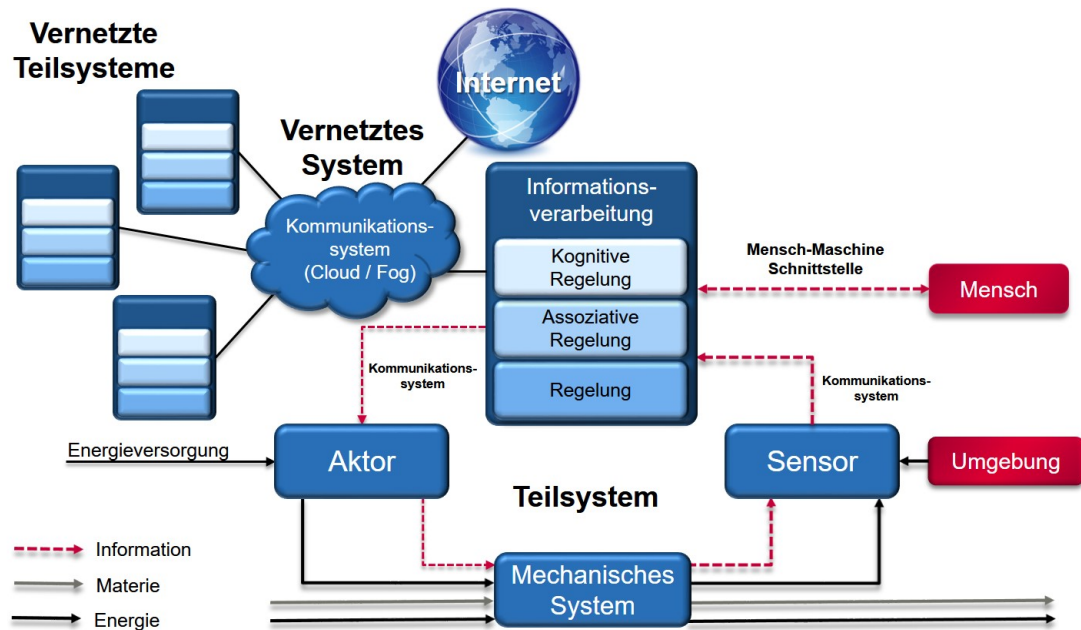


Abbildung 3.: Cyber-Physisches System [Ral17]

Diese Arbeit beschäftigt sich mit den Anforderungen an die höhere Eigenintelligenz. Einfache Regelungen wie in mechatronischen Systemen reichen nicht aus um assoziative und kognitive Algorithmen abzubilden.

CPS gliedern sich in das Konzept der Industrie 4.0 ein. Lee [LBK15] entwirft eine Architektur die dies ermöglicht. Eine Reihe von Survey-Arbeiten liefern einen Überblick [XD19].

Das von Liu et al. [LX17] entworfene Machine-Tool wird in einigen Arbeiten wieder aufgegriffen. So verwendet Armendia et al. [AAP⁺19] das Konzept um Simulationen in digitalen Zwillingen zu verwirklichen.

Zentrale Fragen der Nachhaltigkeit des Begriffs der Industrie 4.0 haben Drath et al. in Industrie 4.0: hit or hype? [RA14] bereits 2014 gestellt, sowie den Begriff umfassend erklärt und dessen Entwicklung seit 2011 dokumentiert. Wobei sich Kritik ob des Erfolgs in den zurückliegenden Jahren zerschlagen hat. Abstrakte Grundlagen werden von Platzer et al. [Pla18] in seinem Buch Logical foundations of cyber-physical systems entwickelt. Darin beschreibt er unter anderem CPS unter dem Aspekt der Spieltheorie und der logischen Beweisbarkeit des Systems. Er führt Optimierungen anhand arithmetischer Modelle vor.

Der Übergang hin zur Integration von digitalen Zwillingen ist fließend. Leng et al. [LZY⁺19] untersuchen dies in Bezug auf Smart Workshops. Das Konferenz-Sammelwerk Product Lifecycle Management (PLM) to Support Industry 4.0 [CBNR18] bein-

Autor	Basis/Vorgehen/Art	Aim/Scope	Begrifflichkeit	Übersicht	Formal Beweisbarkeit	Weiterentwicklung	Security	Digitale Zwillinge
MRR + 18	Entwurf eines Production Message Bus	Cyber-physical production system architecture	◐	○	●	○		
LBK15	Erstellung einer Architektur	CPS	◐			◐		
JFS + 19	Erstellung einer Architektur	CPS Security 2017					●	
XD19	Survey	Big Data		●				
CTT19	Sammelwerk	Design, Simulation und Formale Methoden	◐		●			
LX17	Review	Machine Tool 4.0				◐		○
HLLL17	Survey	CPS Security 2017	◐				●	
LZY + 19	Review	cyber-physical system architecture	◐	◐		◐		
Thi19	Sammelwerk	Energy Efficiency/Demand + Cyber Physical Systems	◐	●	○	◐		◐
RA14	Übersicht über Industrie 4.0	Anforderungen für Industrie 4.0	◐	●				
Pla18	Monographie: Abstrakte Gesetzmäßigkeiten erkennen	Mathematische Beweisbarkeit und Optimierbarkeit von CPS	◐		●	◐		
AAP + 19	Machine Tool	CPS+DZ	◐	○				●
CLL + 18	Entwickeln eines neuen Manufacturing Execution System	Erfassung von Sensorwerten mit MTConnect+MES kombinieren	◐			●		
CBNR18	Sammelwerk	u.a. CPS aus Sicht des PLM	●	○		◐		
MCST19	Review: volle Synchronisation Physische-Digitale Welt	Skalierbare Lösung	◐	○		●		

Tabelle 1.: Wissenschaftliche Arbeiten mit dem Schwerpunkt CPS

haltet einige Arbeiten die sich aus Sicht des PLM damit beschäftigen. Unter anderem arbeiten Durao et al. [DHA⁺18] die Anforderungen aus der Industrie an digitale Zwillinge, sowie deren Stärken auf Basis von Literatur heraus. Es ergibt sich eine hohe Eignung für Real-Time-Anwendungen.

Die Entwicklung der Eco-Factories basiert auf CPS. Thiede et al. [Thi19] verbinden in ihrem Sammelwerk die Entwicklung von energieeffizienten Fabriken und CPS. Einige Werke des Sammelbandes werden im weiteren Verlauf behandelt, da sie sich in andere Themenfelder einordnen.

2.2. Digitale Zwillinge (DT)

Digitale Zwillinge bilden physische Elemente, in dieser Arbeit ausschließlich Produktionsmaschinen, in der digitalen Welt ab. Digitale Zwillinge beschränken sich auf Teilspekte der physischen Welt. Sie sind in der Literatur ausführlich diskutiert [LLWZ19].

Durch Spezialisierung kann die Komplexität des einzelnen Zwillings verringert werden ohne dabei die Leistungsfähigkeit des gesamten Systems zu reduzieren. Das Umsetzen der objektorientierten Prämissen, hohe Kohäsion und lose Bindung, bietet sich dabei an. Durch kleinere Instanzen können die Reaktionsfähigkeit und Redundanz erhöht werden [DHA⁺18].

Mechatronische Systeme können, wie bereits erwähnt, keine komplexen Analysen leisten und werden zu cyber-physischen Systemen weiterentwickelt. Teil der Entwicklung kann die Einbettung von digitalen Zwillingen sein. Diese können somit auf globale Informationssysteme zugreifen, als auch mit entfernten Teilsystemen Informationen austauschen [HNT⁺18].

Die Abbildung der physischen Welt lässt sich als Simulation oder Modell umsetzen. Der Großteil der Forschung nutzt dazu Modelle. Simulationen haben sich als aufwendig herausgestellt, da der initiale Aufwand zum Erstellen verglichen mit dem modellbasierten Vorgehen hoch ist [ZLC16]. Nach [LGG18] ist die Vorhersagequalität nicht stabil. Änderungen des physischen Systems, wie kleine Veränderungen der Funktionsweise der Maschine durch Verschleiß, führen zu aufwendigen Anpassungen der Simulation [ABS15].

Bei digitalen Zwillingen für Produktionsmaschinen kann die Maschine selbst oder das Werkstück im Fokus stehen. Letzteres ist für die Vorhersage von Eigenschaften des Werkstücks geeignet, ersteres für die Funktionsweise der Maschine. Anhand der Funktionsweise kann eine Vorhersage des Energieaufwands für die Bearbeitung des

Autor	Basis/Vorgehen/Art	Aim/Scope	Relevanz	Digital Twin	Simulation	Modell	Additive Fertigung	Energievorhersage
AAP + 19	Machine Tool	CPS+DZ	●		●			●
LLWZ19	Survey	DT Model Features	●	●		●		
HNT + 18	Knowledge Ressource Center, Agenten, FDM+DLP Druck	DT in CPCM-System mit Info-Modell + MTConnect Protokoll	●	●		●	●	
MD19	Druckprozess-Analyse, Survey	DT Metall-Legierungs-Druck	●	●	●	●	●	
PJIZ18	Software Circuit Breakers	Feature Toggles/Safeguards für DTs	●	●				
KMZ + 17	Transient, 3D Modelle	Additive Fertigung: Oberflächen Vorhersagen	○			●	●	
DZTB17	Survey	Vorhersage von Mikrostruktur und Mechanischen Parametern	○			●	●	
CFF17	Digital Twin basierend auf Emissionen + Gradient boosted Trees	Vorhersage der Druck-Ergebnisse	●			●	●	
DHA + 18	Survey	Industry 4.0	●				●	
ZLX18	Review	Digital Twin	●				●	
MAH18		Model-Basierte Definition von DZ	●	●		●		
AGOP19	Simulation von Maschinen-Werkzeugen	Optimierung von Maschinenwerkzeug-Nutzung	●	●	●			
NGP + 18	Nutzen von Simulationstools ((NX CAD, Simcenter 3D, LMS System Synthesis, LMS Amesim, ANSYS, STAR-CCM))	Entwicklungsumgebungen für verschieden Use Cases	●	●	●			
GNU + 18	Anwenden von Tools für die Simulation von UAV	Auslagern der Simulation für PLM und Model-Based-Systems Engineering	●	●	●			

Tabelle 2.: Wissenschaftliche Arbeiten mit dem Schwerpunkt 'Digitaler Zwilling'

Werkstücks erstellt werden. Die Betrachtung des Werkstücks hat in der Forschung hohe Relevanz. Vorhersagen über dessen Beschaffenheit und eine Optimierung der Werkstück-Bearbeitung sind von großem Interesse.

Einige Arbeiten in diesem Bereich sind wegweisend und werden daher im Rahmen dieser Arbeit betrachtet, so die Arbeit von Knapp et al. [KMZ+17]. Sie entwickeln wissenschaftliche Methoden, zur Entwicklung mathematischer Modelle, die das bearbeitete Werkstück physikalisch beschreiben. In der Domäne des additiven 3D-Drucks, hier laser-assisted additive manufacturing (AM), bilden sie das Verfahren physikalisch nach und können mathematische Modelle der Maschinen erstellen. Ziel ist eine Vorhersage der Oberflächen-Beschaffenheit des Werkstücks zu erstellen. Sie führen als erste sogenannte Building Blocks ein. Diese Arbeit dient vielen weiteren als Basis, DebRoy et al. haben diese in einem Survey-Paper untersucht [DZTB17]. Im Projekt Twin-Control [AGOP19], wurde ein Simulations-Framework entwickelt, mit dem

Fokus auf das Maschinen-Werkzeug (Machine-Tool). Eine Datenbank von virtuellen Maschinen-Werkzeugen wurde angelegt, um diese als Bausteine für Simulationen von Produktionsmaschinen zu verwenden. Der Energieaufwand wird auf Basis empirischer Daten durch ein spezielles Modul berechnet, das mit der Simulation verbunden ist [AGOP19] [Arm19].

Duro et al. widmen sich in ihrem Survey-Paper der Definition des Begriffs digitaler Zwilling und in welchen Bereichen er Anforderungen an das CPS stellt [DHA⁺18]. Auf Basis einer Literatur-Recherche und Interviews ergaben sich Anforderungen in den Bereichen Real-Time Data, Integration, und Fidelity. Letzteres ist die Wiedergabetreue der physischen Parameter und Basis für die Adaption an Veränderungen der physischen Umgebung (Self-Awareness).

2.3. Energievorhersage

Arbeiten die sich mit dem Energieverbrauch beschäftigen werden in den folgenden Abschnitten betrachtet.

2.3.1. Energieverbrauch in der Fertigung (EVR)

Die Arbeiten dieser Kategorie dienen dem Verständnis des Energieaufwands und zeigen wo und weshalb Energie aufgewendet wird. Zudem dienen sie der Entwicklung eines Grundverständnisses über die Funktionsweise von Produktionsmaschinen.

Der komplette Energieaufwand im Lebenszyklus eines Produkts kumuliert sich aus dem jeweiligen Aufwand der einzelnen Phasen. Laut Fallböhrer [Fal00] und Spur et al. [SK97] beginnt der Lebenszyklus mit dem Produktentstehungsprozess, der wiederum in Produktplanung, Produktgestaltung, Produktionsplanung und Produktion unterteilt wird. Nach Ehrelnspiel et al. [Ehr07] eignen sich die ersten Planungsphasen für eine Einflussnahme auf den Energieaufwand, was im Graphen 4 dargestellt ist.

In der Produktionsplanung wird eine zeit-, kosten- und qualitätsoptimierte Fertigung angestrebt [Eve13]. Für die einzelnen Teilbereiche gewichten Blesl et al. [BK13] die Einflussmöglichkeiten auf die Energiekosten wie in Tabelle 3 dargestellt.

Daraus leitet sich der Ansatz dieser Arbeit ab, den Energieaufwands in den Planungsprozess einzubeziehen. Da dieser von Maschine und Fertigungsmittel abhängig ist, wird er bei den folgenden Produktionsplanungsschritten laut Eversheim et al. [Eve13] mit einbezogen.

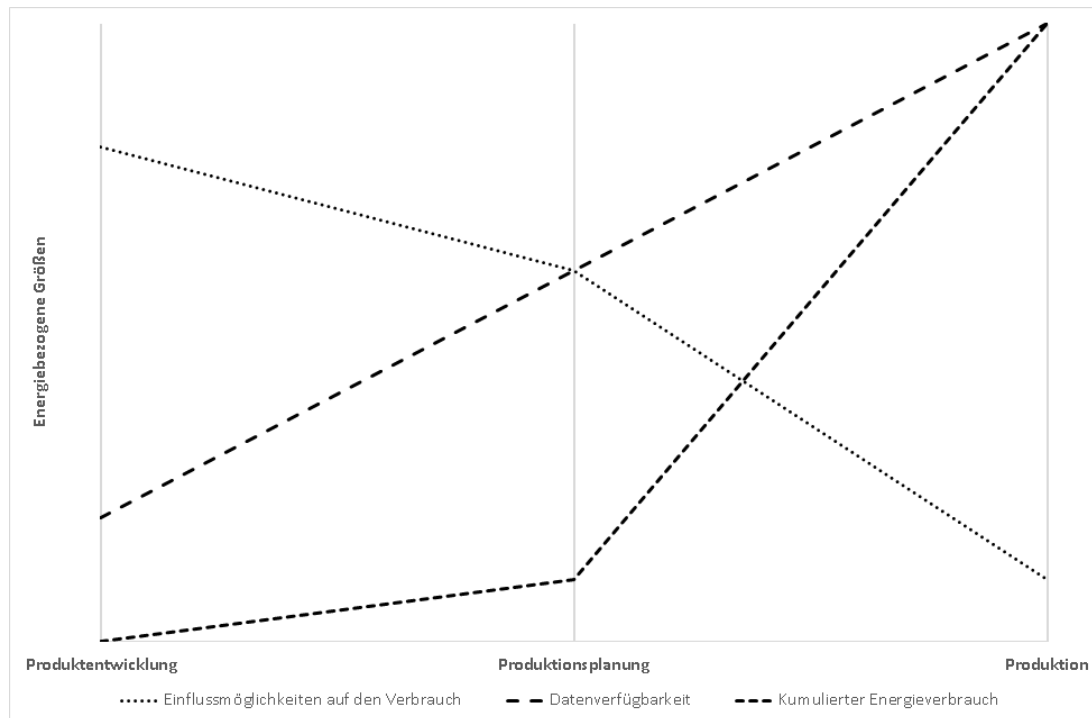


Abbildung 4.: Entwicklung der Einflussnahme auf den Energieverbrauch

Eine Analyse des Energieverbrauchs von Maschinen auf technischer Ebene wird schon seit geraumer Zeit vorgenommen. Bonnesky et al. [Bon02] haben dazu Energiekennzahlen entwickelt, die nach dem UPN-Modell PPS-Systeme abbilden. Ein UPN-Modell beschreibt dabei, in welchem Umfang die Energie in Umwandlungs-, Produktions-, Nebenanlagen verwendet wird.

Junge [Jun07] entwickelt mit Hilfe einer Materialflusssimulation ein Modell um Heizenergie zu simulieren. Für die Domäne des 3D-Drucks haben Peng et al. [Pen16] ein Survey erstellt, indem Arbeiten untersucht werden, die sich dem Energieverbrauch widmen. Die Arbeiten in diesem Abschnitt beschäftigen sich neben der Primär-Energie, die der eigentliche Herstellungsprozess erfordert, auch mit Sekundär-Energie, wie der Kühlung von einzelnen Aggregaten der Produktionsmaschine.

In dieser Arbeit wird der gesamte Energieverbrauch der Fertigungsmaschine erfasst und vorhergesagt.

2.3.2. Energieeffizienz

Die Energieeffizienz zu steigern ist bei einigen Arbeiten zur Energievorhersage die Zielsetzung. Um evaluieren zu können wie sich der Energieaufwand verändert, werden zu jeder Veränderung der Parameter Energievorhersagen getroffen. Diese dienen dazu















Aufgaben der Produktionsplanung	Einflussmöglichkeiten	Ansatzpunkte
Planungsvorbereitung		Wenige Bauteil-Informationen verfügbar
Stücklistenverarbeitung		Erste Informationen über Bauteil
Prozessplanung		Auswahl von Fertigungsmitteln
Operationsplanung		Wahl der Prozessparameter
Montageplanung		Kaum Einflussmöglichkeiten
Prüfplanung		Kaum Einflussmöglichkeiten
Fertigungs- und Prüfmittelplanung		Planung der Fertigungsmittel und Maschinenauswahl
NC - RC Programmierung		Energieoptimierte Bahnplanung
Kostenplanung		Alternative Fertigungsverfahren und Produktionsmittel
Fertigungsmittelplanung		Auswahl der Bearbeitungsmaschinen
Lager- und Transportplanung		Geringer Anteil am Gesamtenergieverbrauch
Personalplanung		Mitarbeiterschulungen
Flächen- und Gebäudeplanung		Kaum Einflussmöglichkeiten
Investitionsplanung		Kaum Einflussmöglichkeiten

Tabelle 3.: Möglichkeiten der Einflussnahme auf den Energieverbrauch, angelehnt an Blesl et al. [BK13]

Entscheidungen zu treffen, so können zur Entwurfszeit Parameter des Maschinen-Werkzeugs betrachtet werden oder in der Produktionsplanungsphase die Ressourcenplanung der Maschinen optimiert werden. Ein Forschungszweig, der sich mit der Energieeffizienz-Forschung überlappt, ist das Eco Design. Dabei wird die Produktion auf umfängliche Umweltverträglichkeit optimiert [Haa] [HPT13].

Die Arbeiten zu diesem Thema sind den Tabellen 5, 6, 7, 8 am Ende des Kapitels zu entnehmen.

2.3.3. Simulation des Energieverbrauchs

Auch wenn die Simulation von physischen Maschinen nicht dem in dieser Arbeit verwendeten Verfahren entsprechen, bringen einige Arbeiten Ansätze hervor, die adaptiert

werden könnten.

Die zumeist holistisch angelegten Simulationen sind aus einzelnen Komponenten aufgebaut, die entweder aus einem Katalog wie Simscape/Matlab stammen oder selbst entwickelt sind. Die Simulation wird in einem Framework wie Simlab/Matlab ausgeführt, um mit den Wechselwirkungen der einzelnen Komponenten das System zu simulieren. Die unterschiedlichen Simulationsmethoden können wie in Abbildung 5 in Kategorien aufgeteilt werden.

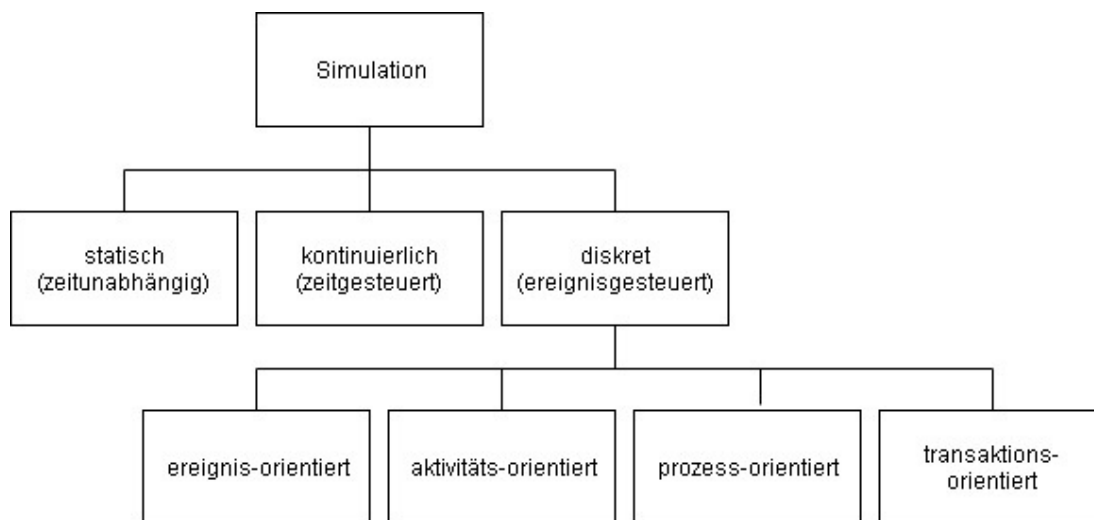


Abbildung 5.: Einteilung der Simulationsmethoden, angelehnt an [Wen10]

Die Forschungsarbeiten im Projekt Twin-Control [Arm19] nutzen Simulationen umfangreich. Basis ist dabei ein Katalog von simulierten Subsystemen von Fertigungsmaschinen in der Automotive- und Aerospace-Domäne. Darauf bauen weitere Entwicklungen auf, so auch ein Ansatz den Energieaufwand in der Simulation zu messen. Dazu wurden empirische Daten an mehreren Messpunkten der Maschine erfasst und als Datengrundlage für die Simulation in Matlab und Simscape verwendet. Simscape ist eine Erweiterung zu Matlab, die bereits einen Katalog an Subsystemen mit sich bringt. Zudem ist ein Zugriff auf einen Marktplatz von Simulationen möglich. Im weiteren Verlauf dieser Arbeit wird eine Verwendung von Simscape untersucht. Twin-Control kooperierte mit einem französischen Automobilhersteller. Mittels des zu Verfügung stehenden Maschinenparks konnten so umfangreichen Sensordaten erfasst werden. Die Ergebnisse von Twin-Control sind vielversprechend, der Einsatz in einer produktiven Umgebung steht noch aus.

In Holistic Simulation Environment for Energy Consumption Prediction of Machine Tools von Abele et al. [ABS15] wird eine holistische Simulation entwickelt, die sowohl mathematische als auch empirische Daten verwendet. Die Grundidee ist, einen virtuellen g-Code-Interpreter (ECOMATION) zu entwickeln, der die Kommandos in

den g-Codes in einen Energieverbrauch übersetzt. Es wurden Interpreter für Maschine aus den Domänen Bohren und Fräsen entwickelt und der Vorhersagequalität positiv evaluiert. Die Simulation wurde in Matlab und Simulink realisiert. Diesem Ansatz folgen weitere Arbeiten, wie Towards Energy-Efficient Machine Tools through the Development of the Twin-Control Energy Efficiency Module von Flum et al. [Eis14] [FSSA19].

Die Arbeiten zu diesem Thema sind den Tabellen 5, 6, 7, 8 am Ende des Kapitels zu entnehmen.

2.3.4. Modellierung des Energieverbrauchs (EV)

Die Modellierung der Funktionsweise der Maschine hat das Ziel, das Verhalten der Maschine für einen definierten Zielbereich vorherzusagen. Es besteht aus einer mathematischen Formel, die die abhängigen Parameter und Ziel-Parameter enthält. Abhängige Parameter können als Datenbasis bzw. Input, Ziel-Parameter als Vorhersage bzw. Output gesehen werden. So kann der Energieaufwand abhängig vom Gewicht des Werkstoffs oder der Bearbeitungsgeschwindigkeit des Werkzeugs sein.

Die Modelle variieren abhängig von Domäne, Verwendungszweck, Umfang, Ziel, Entwicklungsgrundlage und Methodik. Ausgewählte Arbeiten sind in Tabelle 4 zusammengefasst.

Li und Kara [LK11] entwickeln ein Modell für Schneidwerkzeuge in Drehmaschinen, um den Energieverbrauch für die Bearbeitung eines Werkstücks zu prognostizieren. Sie reduzieren die abhängigen Faktoren auf die Werkzeug-Verfassung, das Werkstück-Material, die Schneide-Parameter und auf die Notwendigkeit einer Werkzeug-Kühlung. Sie führen experimentell Reihemessungen durch und erhalten nach Ausgleichsrechnung (curve-estimation) ein mathematisches Modell.

Liu et al. [LXL15] [Shu12] erfassen drei Typen von Zeitabschnitten: Start-Phase, Idle-Phase und Schneidphase/Arbeitsphase. Sie entwickeln ein Modell zum Energieverbrauch für jede der drei Phasen, basierend auf empirischen Daten und physikalischen Überlegungen. Daraus wird ein Vorhersagemodell abgeleitet, das als operative mathematische Funktion formuliert wird. Die Komplexität der Modelle variiert, so ergibt sich für die Start-Phase das Modell für den Energieaufwand 2.1. Dabei ist P_i die gesamte Energie des Motors, P_{ie} der Energieverlust, P_{im} der Energieverlust des Getriebes, E_m das magnetische Feld des Motors, E_k die kinetische Energie des

Systems.

$$P_i(t) = P_{le}(t) + P_{lm}(t) + dE_m/dt + dE_k/dt \quad (2.1)$$

Die Vorhersagefunktion für den Energieverbrauch E_s ist davon das Integral über die Zeit t . Um eine operative Gleichung zu erhalten, ist der Energieaufwand für das Bewegen der Spindel zu Berechnen. Im Proof of Concept [LXL15] steht dieser in einfachem Verhältnis zur Geschwindigkeit.

$$E_s = \int_0^t P_i(t)dt = \int_0^t (P_{le}(t) + P_{lm}(t) + dE_m/dt + dE_k/dt)dt \quad (2.2)$$

Aus dem Modell leitet sich die Gleichung 2.3 ab, die von der Geschwindigkeit der Spindel (n) und der Dauer der Phasen (x) abhängt. In der Arbeitsphase hängt das Modell von der Vorwärtsbewegung sowie der Schnitttiefe und Geschwindigkeit ab und wird entsprechend entwickelt. Der Idle-Phase wird ein konstanter Energieverbrauch zugeordnet. Der gesamte Energieaufwand ergibt sich aus der Zeit der einzelnen Phasen und der entsprechend gewichteten Summe der jeweiligen Energieverbrauchswerte.

$$E_s = x_1 * n^2 + x_2 * n + x_3 \quad (2.3)$$

Hu et al. [HLHH12] unterteilen den Energieaufwand in konstante und variable Verbrauchswerte. Sie identifizieren den Bearbeitungsprozess nach dem Energieverbrauch des Werkzeugs. Aus Domänen-Wissen leiten sie eine Energieverbrauchsgleichung ab.

Liu et al. [LLQ17] nutzt die Ergebnisse von Hu et al. [HLHH12] und Huang et al. [SCY15] und führt den Energieaufwand für eine Bearbeitung mittels bestimmter Maschinen-Werkzeuge hin zu einer Energieeffizienz der Maschine zusammen. Die Effizienz dient als Entscheidungskriterium für die Ressourcenplanung.

Lee et al. [LKPM17] entwickelt eine Methode, um die Bearbeitung zu optimieren. Dazu wird das CNC-Programm mittels eines Modells (VMT) interpretiert, das den Bearbeitungsprozess virtuell durchführt und den Energieverbrauch vorhersagt. Die Simulationsergebnisse werden mittels genetischer Algorithmen für eine Umprogrammierung verwendet. Das Modell wurde durch Überlegungen zur Funktionsweise und

auf Basis experimenteller Daten erstellt.

Abele et al. [ASB⁺19] verwenden ebenfalls ein virtuelles Werkzeug, um den Energieverbrauch vorherzusagen, bauen dieses jedoch als Simulation auf.

Zhao et al. [ZLH⁺17] untersuchen den Energieverbrauch auf der Betrachtungsebene von Maschinen-Werkzeug, Maschine, und Maschinen-Park. Zudem erstellen sie einen Überblick über Modellierungsverfahren.

Shin et al. [SWRS19] verwenden zum Aufbau ihrer Modelle physikalische Grundlagen und empirische Daten. Sie ziehen den NC-Code heran um die Bearbeitung zu simulieren, die Verbrauchs-Daten entnehmen sie den MTConnect-Dokumenten, die dem Maschinen-Werkzeug zugeordnet sind. Auf Basis dieses üblichen Verfahrens betrachten sie die Daten aus datenanalytischer Sicht. Die Sensorattribute entsprechen so Daten-Elementen, die mit Sensorwerten instanziiert werden. Die Sensorwerte werden in einem Intervall kumuliert. Dabei ergibt sich Gleichung 2.4, mit der Delta-Energie e , der Power P , dem Sampling Intervall t_c und dem Timestamp t , um den Energieverbrauch zu berechnen. Sie betrachten insbesondere die Datensynchronisierung der Daten, was auch in der vorliegenden Arbeit beachtet wird, da die Sensordaten vom Drucker unabhängig erfasst worden sind.

$$e_i = P_i * t_c \quad (2.4)$$

Peng et al. [PS17] betrachten den Energieaufwand für den gesamten Lebenszyklus eines Produkts in der Domäne des 3D-Drucks. Für die Produktion erstellen sie ein rudimentäres Modell, basierend auf Domänen-Wissen und einer Test-Studie. Die verbrauchte elektrische Energie ist dabei im Wesentlichen abhängig vom Materialverbrauch und der Druckdauer. Das Modell der vorliegenden Arbeit verwendet je nach Evolutionsschritt unterschiedlich komplexe Modelle. Ebenso variiert die Anzahl der abhängigen Attribute, die bei [PS17] auf wenige begrenzt bleibt.

Als Exkurs in weitere Modellierungsverfahren dient die Arbeit von Swan et al. [SU09]. Sie evaluieren zwei Verfahren, Top-Down und Bottom-Up. Ausgehend von makroökonomischen Größen schließt das Top-Down-Verfahren auf die Veränderung des Energieverbrauchs, so beeinflusst beispielsweise das Wetter den Heizbedarf. Beim Bottom-Up-Verfahren werden einzelne Haushalte betrachtet und deren Verhalten auf die Allgemeinheit deduziert. Hierbei werden zur Vorhersage des Energieverbrauchs auch neuronale Netzwerke verwendet.

Arbeit	Domäne	Verwendungszweck	Umfang	Ziel	Entwicklungsgrundlage	Methodik
LK11	Drehmaschine	Werkzeug-Optimierung	Schneidvorgang	Vorhersage elektrische Energie	Empirisch	Ausgleichsrechnung
RK12	Zerspanung	Prozessplanung/ Werkzeugherstellung	Bohrvorgang	Vorhersage Energieaufwand	Empirisch	verm. Ausgleichsrechnung
Sch14	Fertigung	Ressourcenplanung	Maschinenpark	Vorhersage Energieaufwand	Physikalisch	Ableiten von Gesetzmäßigkeiten
LX15	Drehmaschine	Energie Effizienz	Schneidvorgang	Vorhersage kinetische Energie	Empirisch + Physikalisch	Ableiten von Gesetzmäßigkeiten
PS17	FDM	Energieaufwand	Produkt-Lebenszyklus	Vorhersage elektrische Energie	Empirisch + Physikalisch	Ableiten von Gesetzmäßigkeiten
ZLH + 17	Maschinen, Maschinen- Werkzeuge, -Parks	-Survey-	Maschine	Vorhersage Energieaufwand	Empirisch + Physikalisch	Ausgleichsrechnung + Ableiten von Gesetzmäßigkeiten
LKPM17	Maschinen/ Maschinen Werkzeuge	CNC-Programm- Optimierung	Bearbeitungsvorgang	Vorhersage elektrische Energie	Empirisch + Physikalisch	Ableiten von Gesetzmäßigkeiten + Genetischer Algorithmus
LLQ17	Maschinen Werkzeuge	Ressourcenplanung	Maschine/ Maschinenpark	Vorhersage kinetische Energie	Weitere Paper	Zusammenführen
SWRS19	Maschinen/ Maschinen Werkzeuge	Prozessplanung	Maschine/ Maschinenpark	Vorhersage Energieaufwand	Physikalisch + Daten- Analytisch	Ableiten von Gesetzmäßigkeiten
HLHH12	Drehmaschine	Energie Effizienz	Schneidvorgang	Vorhersage kinetische Energie	Physikalisch	Ableiten von Gesetzmäßigkeiten
SU09	Wohnhaus	Methodenvergleich	Haus	Vorhersage Energieverbrauch	Top-down: Statistisch, Bottom-up: Empirisch	Ableiten von Gesetzmäßigkeiten

Tabelle 4.: Zentrale Arbeiten der modellbasierten Energievorhersage

Die Arbeiten zu diesem Thema sind den Tabellen 5, 6, 7, 8 am Ende des Kapitels zu entnehmen.

2.3.5. Neuronale Netze

Aufgrund der Komplexität des Energieverbrauchs von Werkzeugmaschinen wurde die Verwendung neuronaler Netzwerke zur Vorhersage deren Energieverbrauchs bereits in einigen Arbeiten untersucht.

Neuronale Netze sind vereinfachte mathematische Modelle, die auf der dicht vernetzten Parallelstruktur biologischer neuronaler Netzwerke basieren. Sie bestehen aus Neuronen, die in verschiedenen Schichten (Layers) angeordnet und durch variable Gewichte verbunden sind. In der Eingangsschicht (Input-Layer) werden die Eingangsparameter übernommen, im Output-Layer wird das Endergebnis berechnet. Die dazwischen liegenden Schichten (Hidden-Layer) verarbeiten die Daten wie folgt. Die zu Beginn zufällig festgelegten Gewichte werden während des Trainingsprozesses durch ein iteratives Verfahren berechnet. Die Neuronen gewichten die Eingangs-Signale mit ihrem Gewicht und geben das Ergebnis an die Neuronen der folgenden Schicht weiter. Der Lernprozess findet durch die Anpassung der Gewichte statt, mit dem Ziel die Abweichung des Endergebnisses vom Label (Zielwert) der Trainingsdaten zu minimieren [CQGR08] [CAO09].

Das BP-Netzwerk ist ein neuronales Netz, das auf dem Algorithmus der Fehlerrückvermehrung (Error Backpropagation) basiert. Xie et al. etablieren ein Drehmaschinen-Energieverbrauchsvorhersagemodell auf Basis des neuronalen BP-Netzwerks [XCS⁺12]. Der Input besteht aus Schnittgeschwindigkeit, Vorschubgeschwindigkeit und Schnitttiefe; der berechnete Output aus dem Energieverbrauch der Werkzeugmaschine beim Schneiden. Experimentelle Ergebnisse zeigen, dass die Vorhersagegenauigkeit des Modells über 92% liegt.

Al-Hazza et al. entwickeln ein neuronales Netzwerkmodell zur Vorhersage des Energieverbrauchs von Werkzeugmaschinen für das Hochgeschwindigkeits-Hartdrehen von AISI 4340 Stahl [FTM⁺11]. In der Eingangsschicht befinden sich 4 Neuronen, die Schnittgeschwindigkeit, Vorschubgeschwindigkeit, Schnitttiefe und negativen Spanwinkel (Winkel zwischen Spanfläche und Werkzeug-Bezugsebene) ausdrücken. Es gibt 2 Neuronen in der Ausgabeschicht, die den Energieverbrauch der Werkzeugmaschine und die Leistung beim Schneiden repräsentieren. Die experimentellen Ergebnisse zeigen die beste Vorhersagequalität bei Verwendung von 20 Hidden-Layern. Damit wird eine Abweichung von ca. 2,6% der tatsächlichen Werten erreicht. Für den Energie-

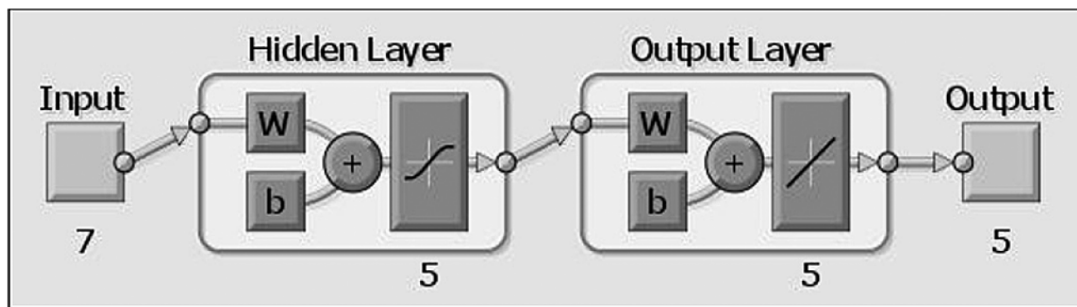


Abbildung 6.: Netzwerkmodell zur Analyse des Einflusses von Prozessparametern, dabei steht w für weight und b für bias [QCR11]

verbrauch der Werkzeugmaschine beim Schneiden ergibt sich als relevantester Faktor die Schnittgeschwindigkeit.

Um den Einfluss von Prozessparametern auf den Stromverbrauch bei Hochgeschwindigkeits-Kugelpfhräsarbeiten am AISI H13-Stahl zu analysieren, entwickeln Quintana et al. ein neuronales Netzwerkmodell zur Vorhersage des Stromverbrauchs beim Schneiden [QCR11]. Die neuronale Netzwerkstruktur ist in Abbildung 6 dargestellt. 7 abhängige Attribute werden als Input des Netzwerks verwendet: Spindeldrehzahl, Vorschubgeschwindigkeit, Vorschub pro Fräs-Zahn, axiale Schnitttiefe, radiale Schnitttiefe, Werkzeugradius und die Kühlmittelbedingungen. Ausgegeben werden 5 Elemente: die Leistungsaufnahme der Spindel, X-Achse, Y-Achse und Z-Achse sowie die gesamte Leistungsaufnahme der Werkzeugmaschine beim Schneiden. Der Regressionswert R misst die Korrelation zwischen den Ausgängen und den Messdaten. Die experimentellen Ergebnisse zeigen, dass R sich 100% nähert und die Vorhersagegenauigkeit des Modells nahe 98% liegt. Das Modell ermöglicht die Reduzierung des Stromverbrauchs durch die Optimierung der Parameter des Schneideprozesses.

Neuronale Netze sind in der Lage, den Energieverbrauch von Werkzeugmaschinen vorherzusagen. Das neuronale Netz ist jedoch ein lokaler Optimierungsalgorithmus, der leicht in den lokalen Grenzwert fällt. Es kann sich daher bei der Lösung um ein lokales Optimum handeln und nicht um das gesuchte globale [Alt13]. Entscheidend für die Anwendbarkeit von neuronalen Netzen ist die Qualität der Trainings-Daten.

2.4. Planungssysteme (PS)

2.4.1. Entscheidungsverfahren

Die Produktionsplanung umfasst eine Reihe von Aufgaben. So hat sie das Ziel die optimale Produktions-Stück-Größe (Production Lot Size) zu finden. Diese hängt von den vorhandenen Kapazitäten und den Kundenaufträgen ab [GKM17].

Die Auswahl geeigneter Fertigungsmittel, die den Anforderungen des Produkts, des Kunden und des Auftragnehmers berücksichtigen, ist ein weiterer Aspekt der Produktionsplanung [Sch14].

Das Fraunhofer-Project-Center beim Institute of Computer Science and Control beschäftigt sich in einigen Projekten mit Problemen des Scheduling und hat einige Lösungen mit Hilfe von Hierarchischen Planungssystemen, die unter anderem genetische Algorithmen verwenden, entworfen [GKM17].

Ein wiederkehrendes Problem in der Planung ist die Entscheidungsfindung, sowie die Auswahl und Gewichtung der zugrundeliegenden Kriterien. Diese Entscheidungsprobleme wurden umfassend in der Entscheidungstheorie erforscht. Neben der deskriptiven Entscheidungstheorie, die unter anderem versucht Hypothesen über das Verhalten von realen Menschen zu treffen, gibt es den Zweig der präskriptiven Entscheidungstheorie. Sie entwickelt Verfahren, die Empfehlungen zur Lösung eines Problems erstellen [LGSM95].

Diese Arbeit nutzt die präskriptive Entscheidungstheorie, um die Energiedaten mit in den Entscheidungsprozess einzubeziehen.

Nach Bamberg et al. [BCK19] besteht ein Entscheidungsprozess aus den vier Phasen Problemstellung, Lösungsfindung, Optimierung und Implementierung. Diese Prozesse werden in ein Modell überführt, das in der präskriptiven Entscheidungstheorie aus folgenden Elementen besteht: Handlungsalternative, Umweltzustände, Ergebnisse und Ziele [Din91].

Die Menge der Handlungsalternativen sind in der Produktionsplanung begrenzt, es existieren definierte Alternativen, wie die Wahl der Ressourcenbelegung. Die Entscheidungen werden nach [Sch14] aufgrund der Attribute und Ziele der Entscheidung getroffen, typischerweise handelt es sich um Multi-Attribute-Entscheidungen (MADM) [ZG13]. Mittels Verfahren wie Analytic Hierarchy Process und Analytic Network Process können MADM bearbeitet und eine Empfehlung bzw. Lösung erstellt werden.

Die Arbeiten zu diesem Thema sind den Tabellen 5, 6, 7 und 8 am Ende des Kapitels zu entnehmen.

2.4.2. Energieverbrauch als Entscheidungskriterium (PE)

Technologische oder ökonomische Kriterien bildeten in der Vergangenheit die Ressourcen, mit denen die Produktionsplanung durchgeführt wurde. Um den Energieaufwand zu berücksichtigen, wird die Entscheidungsgrundlage erweitert. Sogenannte energieorientierte Systeme berücksichtigen zudem auch das variierende Angebot an Energie. Die Arbeiten in diesem Kapitel beschäftigen sich damit, welche Änderungen und Ergänzungen an den bisherigen Methoden möglich sind. In der Betrachtung liegt der Fokus auf der Produktionsplanung, nicht auf der Art und Weise der Vorhersage.

Schrems [Sch14] entwickelt eine Methode, um den elektrischen Energiebedarf von Produktionsmaschinen abzuschätzen. Diese Schätzung wird in der Produktionsplanung verwendet und beeinflusst die Entscheidung über die Fertigungsmittelauswahl. Für den Entwurf der Architektur des Planungssystems wurde Schrems Methode für diese Arbeit adaptiert.

Liu et al. [LZZ08] nutzen genetische Algorithmen, um die Produktionsplanung unter Berücksichtigung des Energieverbrauchs zu erstellen. Sie entwickeln eine energieorientierte Fertigung für einen Hybrid-Flow-Shop. Dabei handelt es sich um ein Planungsproblem, bei dem eine Reihe von Bearbeitungszentren die Aufträge bearbeitet. Jedes Bearbeitungszentrum besteht aus einem Satz identischer Parallelmaschinen, wobei die Bearbeitung eines Auftrags auf genau einer der Maschinen jedes einzelnen Zentrums erfolgen muss. Auch Luo et al. [LDH⁺13] beschäftigen sich mit diesem Problem unter Einbeziehung des Energieaufwands.

Weinert [Wei10] erstellt eine mathematische Formulierung von zustandsabhängigen Energiebedarfsmengen (Energy-Blocks) für Produktionssysteme. Die Energy-Blocks werden für verschiedene Entscheidungen in der Ablauforganisation verwendet. Sie sind in einer Bibliothek hinterlegt, um sie so in die Fertigungsplanung einzubinden. Diese Vorgehensweise ähnelt der Erstellung von digitalen Zwillingen, auch wenn sie bei Weinert nur als passive Blaupause in einer Datenbank vorliegen.

Keller [Kel] nutzt produktionsabhängige Energievorhersagen, um diese in der Planung zu berücksichtigen. Ziel dabei ist es, den Verbrauch unter dem Aspekt zu planen die vertraglichen Regelungen mit dem Energieversorger zu erfüllen und somit ein variables Lastmanagement zu erhalten. Keller erstellt zudem ein Verfahren, um den erzielten Gewinn zu erfassen der durch den Einsatz des Systems erzielt wurde.

Schultz [Scha] sieht die elektrische Energie als begrenzte Ressource an, die aus zu unterscheidenden Energiequellen stammt. Das von Schultz entwickelte Verfahren berücksichtigt Störungen im Produktionsablauf und sieht Reaktionen vor um diese zu kompensieren. Zudem werden auch produktionsferne Verbraucher berücksichtigt, die in ein energieorientiertes Informationssystem des gesamten Unternehmens integriert werden. Die Modellierung der elektrischen Energievohersage ist dabei in das energieorientierte Informationssystem integriert. Die Abweichungen von der Prognose werden erfasst und zur Korrektur der Vorhersage verwendet.

Einige der obigen Verfahren ermöglichen die Planung dynamisch zu ändern, auch nach Beginn der Produktion. So kann auf ein verändertes Energieangebot, seitens des Energieversorgers reagiert werden. Eine Adaption der Planung kann dabei energieintensive Tasks verschieben, Maschinen mit anderen Fertigungsmethoden verwenden oder die Fertigungsgeschwindigkeit drosseln. Ein weitergehender Schritt umfasst die Änderung der Produktionsparameter in Echtzeit. Diese Verfahren, insbesondere Anforderungen an Hardware und Software, untersuchen Monizza et al. [MRR⁺18].

Die Arbeiten zu diesem Thema sind der Tabellen 5, 6, 7, 8 zu entnehmen.

Verfasser	Basis/Vorgehen/Art	Aim/Scope	Evaluation Criteria											Abschnitt				
			Relevanz	Planung	Energiebedarf	Energieeffizienz	Digital Twin	Simulation	Modell	Additive	CNC	UPN-Modell	Virtueller Interpretier		Simulations-Anwendung			
Abele, Braun et al. 2015	ABS15	Holistische Simulation	●	●	●	○	●	●	●	●	●	●	●	●	●	Matlab	EV	
Abele, Schraml et al. 2019	ASB + 19	Simulation	●	●	●	○	●	●	●	●	●	●	●	●	●	Matlab /SCA	VR	
Agha 2009	Agh19	Mixed Integer Linear Programming MILP	○	●	●	○	○	○	○	○	○	○	○	○	○	○	PE	
Beier 2017	Bei17	Regelkreis-Fertigungssteuerung: Speichermöglichkeiten	○	●	●	○	○	○	○	○	○	○	○	○	○	○	EN	
Bonnensky 2002	Bon02	UPN-Modell	○	●	●	○	○	○	○	○	○	○	○	○	○	○	VR	
Bruzzone et al. 2012	TBAP12	Mathematisches Lösungsverfahren	○	●	●	○	○	○	○	○	○	○	○	○	○	○	EN	
Eisele 2014	Eis14	Empirische Daten + mathematische Modelle	○	●	●	○	○	○	○	○	○	○	○	○	○	○	Matlab /SCA	EV
Flum, Sossenheimer et al. 2019	FSSA19	Simulation basierend auf Datenblätter	○	●	●	○	○	○	○	○	○	○	○	○	○	○	Matlab /SCA	EE
Giret et al. 2015	GTP15	Vergleich	●	●	●	○	○	○	○	○	○	○	○	○	○	○	EN	
Gutowski, Dahmus et al. 2006	GD106	Empirischer Ansatz	●	●	●	○	○	○	○	○	○	○	○	○	○	○	VR	
Gyulai, Kádár et al. 2017	GKM17	Short-Term-Scheduler mittels genetischer Algorithmen	○	●	●	○	○	○	○	○	○	○	○	○	○	○	EN	
Haag 2013	Haa	Materialflusssimulation	○	●	●	○	○	○	○	○	○	○	○	○	○	○	PE	
Hawkins 2018	Haw19	Veröffentlichung steht aus	●	●	●	○	○	○	○	○	○	○	○	○	○	○	VR	

Tabelle 5.: Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 1

Verfasser	Basis/Vorgehen/Art	Aim/Scope	Relevanz	Planung	Energiebedarf	Energieeffizienz	Digital Twin	Simulation	Modell	Additive	CNC	UPN-Modell	Virtueller Interpreter	Simulations-Anwendung	Abschnitt
Junge 2007	Materialflussimulation	Heizenergie													VR
Keller	mathematisches Modell + Analyse Planungsverfahren	Integration des Energieverbrauchs in Produktionsplanung												Matlab	EV- PE
Kemmoé et al. 2015	linearen Programmierung	Spitzenlastmanagement													EN
Lee, Kim et al. 2017	Simulation: VMT + Genetischer Algorithmus	Optimierung Maschinenparameter. Virtueller CNC-controller												VMT	EV
Li, Kara 2011	Empirischer Ansatz	Energieverbrauchsmodell für Werkzeugmaschine													EV
Li, Yan et al. 2013	Force Model basierend auf empirischen Daten	Energie-Verbrauchsmodelle													EV
Liao, Garg et al. 2018	Algorithmen & Modelle	Energie-Verbrauchsmodell													EV
Liu et al. 2008	Energiemenge und Durchlaufzeit	Energieorientierte Fertigung + Hybrid- Flow-Shop													PE
Liu et al. 2014	LDL + 13 Vorgehen	Reihenfolgeplanung: Maschinen + produktabhängiger Energiebedarf													PE
Liu, Guo et al. 2019	Mixed Integer Programming MIP / Empirisch	Energieverbrauchsstrategie												Matlab	PE
Liu, Xie et al. 2015	Vorhersage des Energieverbrauchs des Hauptantriebssystems	Vorhersage Energieverbrauch für Werkzeuge, Werkstück, Planung													EV
Liu, Zhang et al. 2015	Erweitertes Schnittkraftmodell	Energieverbrauchsmodell als Funktion der Schneidekraft													VR
Lödding et al. 2016	Abstrakte Betrachtung des Fertigungsprozesses	Regelkreis der Fertigungssteuerung													EN

Tabelle 6.: Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 2

Verfasser	Basis/Vorgehen/Art	Aim/Scope	Relevanz	Planung	Energiebedarf	Energieeffizienz	Digital Twin	Simulation	Modell	Additive	CNC	UPN-Modell	Virtueller Interpretier	Simulations-Anwendung	Abschnitt
Luo et al. 2013	LDH + 13 Hybrid-Flow-Shop	Planung der Elektrische Energie													PE
Madan, Mani et al. 2015	MMML15 Komplexes Modell	Wissenschaftlich fundierter Leitfaden zur Energievorhersage													EV
Mouzon 2008	C: 08 Lineare Programmierung + heuristische Ansätze	Abschaltung von Maschinen in Nebenzeiten													PE
Peiji 2017	LLQ17 Spindelzahl	Modell der Echtzeit-Energieeffizienz													EV
Peng 2016	Pen16 Survey	3D-Printer Energy- Model													VR
Peng, Sun 2017	PS17 Modelle pro Produktlebenszyklus	Lifecycle-Energieverbrauch													EV
Popp, Liebl et al. 2017	PLZ17 Evaluation	Energy-Flexibility-Model													EV
Rager 008, 2014	RD08 UPN-Modell, Wechselwirkungen zwischen den Produktionsanlagen	Spitzenlastmanagement													PE
Rief, Karpuschewski 2012	RK12 Virtuelles Maschinenwerkzeug	Energieoptimierte Prozessplanung													EV
Schrems 2014	Sch14 Mathematisches Modell + Analyse Planungsverfahren	Modell zur Produktionsplanung													EV-PE
Schultz	Schch Energiemodell (Zustandsorientiert) + Bedarfsüberwachung + Steuerung	Produktionssteuerung													PE
Sharma et al. 2015	SZS15 Heuristisch	Maschinenbelegung mit Energiebeschäftigungsoptionen													PE
Shin, Woo et al. 2019	SWRS19 Generative Process Planning (GPP) + Variant Process Planning (VPP)	Energiemodelle und Eco-Prozess-Planung													EE-EV

Tabelle 7.: Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 3

Verfasser	Basis/Vorgehen/Art	Aim/Scope	Relevanz	Planung	Energiebedarf	Energieeffizienz	Digital Twin	Simulation	Modell	Additive	CNC	UPN-Modell	Virtueller Interpretier	Simulations-Anwendung	Abschnitt
Simon, Roltsch et al. 2017	SRGR17 Energy Flexibility Measures (EFM) + Energy States	Lastanpassung von Produktionssystemen	○	○	○	○	○	○	○	○	○	○	○		PE
Sproedt, Plehn et al. 2015	SPSH15 Diskrete Event-Simulation	Material-Strömungssimulationsmodell	●	○	○	○	○	○	○	○	○	○	○		EV
Suwa, Samukawa 2015	HT16	Auftragsabhängige Werkzeugauswahl	○	○	○	○	○	○	○	○	○	○	○		PE
Swan, Ugursal 2009	SU09 Neuronales Netz, Top-Down + Bottom-Up, Statistisch + Technisch	Energieverbrauchsmoellierung	○	○	○	○	○	○	○	○	○	○	○		EV
Thiede 2019	Thi19 Sammelwerk	Energy Efficiency/Demand + Cyber Physical Systems	○	○	○	○	○	○	○	○	○	○	○		EE-CP
Verma, Rai 2013	VR13 Model + Framework	Optimierung: Materialverlust/Laserenergie	○	○	○	○	○	○	○	○	○	○	○		EE
Weinert 2019	Wei10 Energy-Blocks	Energiebedarfsmengen	○	○	○	○	○	○	○	○	○	○	○		PE
Zhao, Liu et al. 2017	ZLH + 17 Review	Energieverbrauch bei der Fertigung	○	○	○	○	○	○	○	○	○	○	○		EV
Zhou, Liu et al. 2016	ZLC16 Energy-Consumption-Steps + Energieverbrauchsmoell	Energieeinsparungsgesetz + Energy-Consumption Allowance	○	○	○	○	○	○	○	○	○	○	○		EV

Tabelle 8.: Wissenschaftliche Arbeiten aus dem Themenbereich Energievorhersage Abschnitt 4

3. Rahmenbedingungen und Anforderungen

In diesem Kapitel werden die Rahmenbedingen und Anforderungen an die Architektur definiert. Zu Beginn wird die Domäne beschrieben. Vom Planungssystem ausgehend werden die Anforderungen an die Vorhersage des Energieaufwands entwickelt. Sie manifestieren sich in konkreten Anforderungen an die Vorhersage, an das Vorhersage-Verfahren und an die Architektur. Entscheidungen über die Architektur und das Modellierungsverfahren führen zu weiteren Anforderungen. Abschließend werden die Rahmenbedingungen betrachtet. Sie bestehen einerseits aus Eigenschaften der Daten die deren Verwendung einschränken. Andererseits aus der Festlegung auf die Domäne der Produktionsplanung von Fertigungsmaschinen. Wobei sich der Proof of Concept auf den den 3D-Druck beschränkt.

3.1. Vorgehensweise

Die Vorhersage beruht auf ihrer Datenbasis, so bestehen Rahmenbedingungen seitens der verfügbaren Daten. Der Rahmen spannt sich zwischen den verfügbaren Eigenschaften der Produktionsmaschinen und den technischen und administrativen Informationen die sich aus den Auftragsdaten ableiten lassen. Abgesehen von den geschäftlichen Anforderungen die aus dem Kundenauftrag und den Eigeninteressen des Produktionsunternehmens abgeleitet werden, stellt das Planungssystem funktionelle Anforderungen.

Zu Beginn steht die Funktionsweise und der Energieverbrauch von Produktionsmaschinen im Mittelpunkt. Dabei wird in der Domäne des additiven 3D-Drucks auf energierelevante Faktoren des zu druckenden Modells eingegangen. Es wird auf Verfahren eingegangen, um Energiedaten zu erfassen. Abschließend werden die Anforderungen an die Prognose, die sich aus dem Planungssystem ergeben, abgeleitet.

Abbildung 7 stellt die Rahmenbedingungen und Anforderungen strukturiert dar. Das Planungssystem stellt Anforderungen an die Vorhersage. Dieses wiederum an das verwendete Vorhersage-Verfahren und die Infrastruktur. Weitere Anforderungen leiten sich aus Architekturentscheidungen ab.

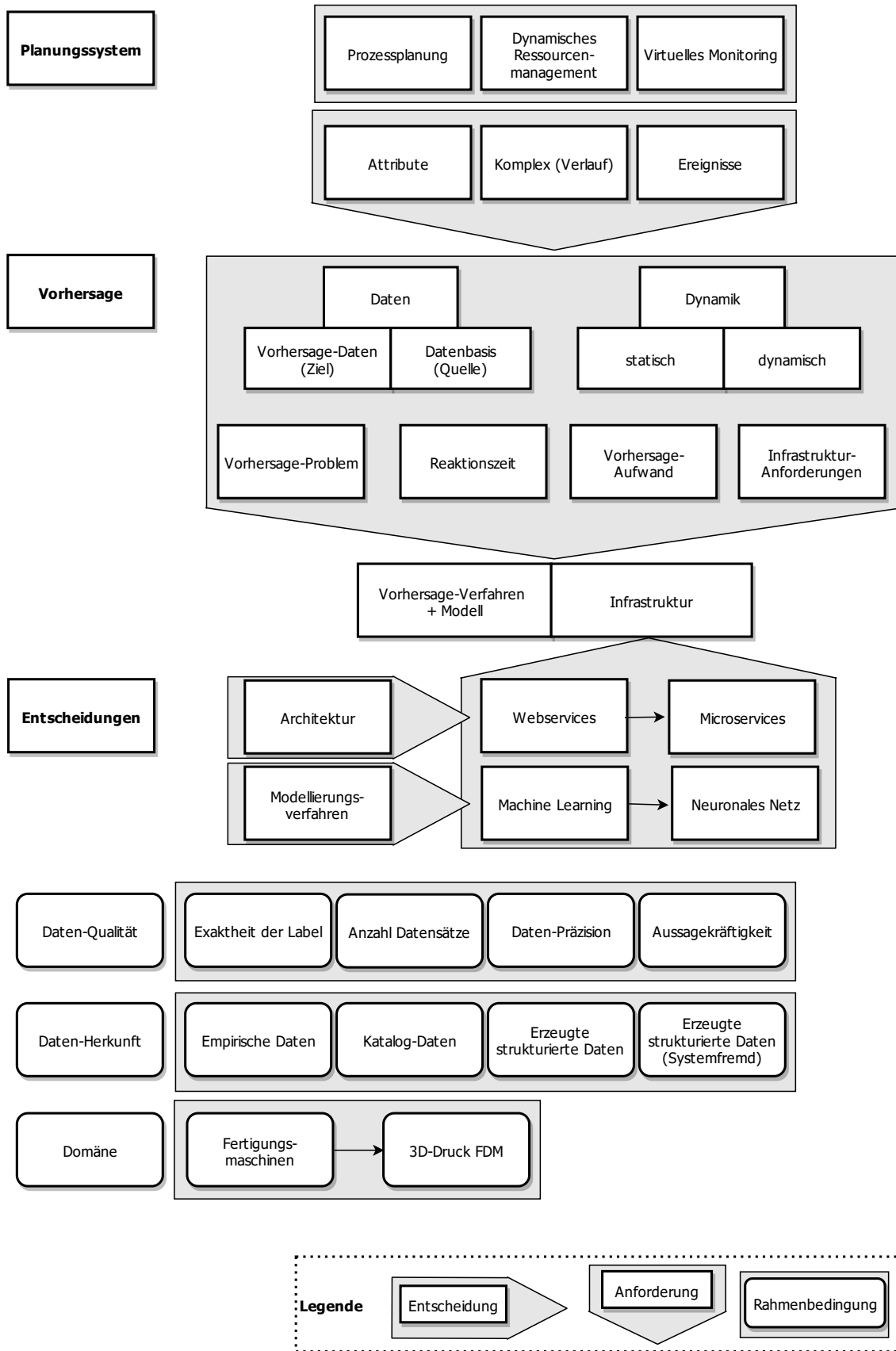


Abbildung 7.: Strukturierte Übersicht der Rahmenbedingungen und Anforderungen

3.2. Festlegung der Domäne

Die grundlegende Motivation, den Energieverbrauch in höheren Maß zu steuern und dies durch eine energieoptimierte Produktionsplanung zu erreichen, wurde bereits in der Einleitung erläutert.

Domäne: Produktionsplanungssysteme für Fertigungsmaschinen

Definition: Produktionsplanungssysteme, die Tasks für ein oder mehrere Fertigungsmaschinen planen. Dabei können ein oder mehrere Maschinen für den Task geeignet sein. Fertigungsmaschinen bearbeiten die Werkstücke einem CNC-Programm folgend.

3.3. Festlegung der Domäne für den Proof of Concept

Der Proof of Concept wird in der Domäne des 3D-Drucks durchgeführt.

Domäne: Produktionsplanungssystem für 3D-Drucker (FDM-Verfahren)

Definition: Produktionsplanungssysteme, die Tasks für ein oder mehrere 3D-Drucker planen. Dabei können ein oder mehrere 3D-Drucker für den Task geeignet sein. Die 3D-Drucker nutzen dabei verschiedene Modell-Formate des Werkstücks.

Beim Prozess der additiven Fertigung werden drei-dimensionale Objekte aus digitalen Modellen hergestellt. Dabei wird Material in Schichten aufeinander aufgetragen und bei jedem Schritt mit der vorigen Schicht verbunden. Abbildung 8 stellt dies dar. Damit unterscheidet es sich von herkömmlichen Verfahren in der Zerspanungs-Domäne wie dem Bohren, Schleifen, Schneiden, die alle auf dem Entfernen von Material basieren [Verma.2013].

Folgende Verfahren werden im 3D-Druck unterscheiden [DZTB17]:

- Finite element method (FEM)
- Finite difference method (FDM)
- Level set method (LSM)
- Volume of fluid (VOF) method with FDM
- Lattice Boltzmann method (LBM)

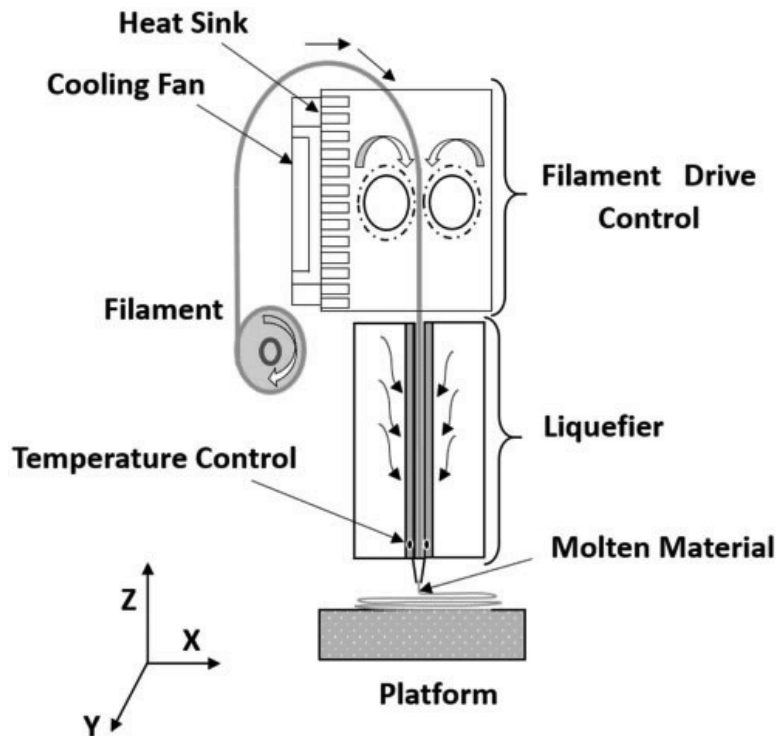


Abbildung 8.: Schematische Darstellung des FDM-Prozesses [PH18]

- Arbitrary Lagrangian-Eulerian (ALE)

Eine detaillierte Prozessbeschreibung sowie Informationen zu 3D-Druck-Technologien und verwendeten Materialien finden sich in der Literatur [Hag15]. Die Vielseitigkeit des 3D-Drucks dokumentieren die Anwendungsfelder, in denen er Verwendung findet.

- Werkzeugbau im produzierenden Gewerbe
- Vorrichtungs- und verwandter Fertigungsbau
- Stromerzeugung
- Robotik
- Wärmeaustauscher und Klimaregler
- Luft- und Raumfahrt
- Kraftfahrzeugbau und Schiffsbau
- Architektur und Gebäudebau

In dieser Arbeit wird von der Verwendung im produzierenden Gewerbe ausgegangen.

3.4. Planungsverfahren

Im Planungssystem wird eine Produktionsplanung und -steuerung (PPS) durchgeführt. Die Planung hängt zusammengefasst von folgenden Faktoren ab [KLL18]:

- Kundennachfrage
- Freie Kapazitäten
- Unternehmensstrategie
- Mengenmäßiger und zeitlicher Produktionsablauf
- Produktionswirtschaftliche Formalziele

In dieser Arbeit wird die Produktionsablaufplanung als Methode festgelegt. Diese besteht wiederum aus folgenden Zwischenzielen [KLL18]:

- Hohe Kapazitätsauslastung
- Geringe Kapitalbindung
- Geringe Beschaffungskosten
- Hohe Termintreue

Davon ist in dieser Arbeit die Kapazitätsauslastung von Relevanz. Das Ziel hierbei können hohe Stückzahlen pro Maschinenstunde oder geringe Leerzeiten sein [KLL18]. Die zu planenden Aufträge werden in Tasks aufgeteilt. Jeder Task entspricht einem Bearbeitungsschritt in einer Fertigungsmaschine. Den Tasks sind spezifische Eigenschaften zugeordnet, wie die erforderlichen Begebenheiten bei der Bearbeitung (Druckparameter), Priorität des Auftrags oder voraussichtliche Dauer der Tasks.

Zur Konkretisierung werden Use-Cases bzw. Funktionsarten des Planungssystems erstellt. Die Tabellen 9 am Ende des Kapitels enthalten eine Zusammenfassung der Anforderungen.

3.4.1. Statische Planung (StaPla)

Die Belegung der Kapazitäten (Maschinenbelegungsplan) wird durch das Planungssystem erstellt. Die zu planenden Aufträge und deren Tasks stehen zu Beginn der Planung fest, eine nachträgliche Änderung ist nicht möglich. Prioritäten der Task wer-

den dabei berücksichtigt, eine optimale Kapazitätsauslastung wird angestrebt. Dabei werden zum Zeitpunkt der Planung die voraussichtlich belegten Ressourcen benötigt, so die Dauer und der Energieaufwand des Tasks. Die Vorhersage des Energieaufwands wird für alle Belegungen einmalig ermittelt. Bei der Entscheidung über die Belegung wird der Energieaufwand berücksichtigt.

Anforderung an die Vorhersage: Die Vorhersage basiert auf den Informationen der Tasks und den Eigenschaften der Fertigungsmaschine. Dabei handelt es sich um einzelne Parameter, bis hin zu einer Menge von hunderten von Parametern. Vom Planungssystem wird der Energieaufwand als simples Attribut eines jeden Tasks erwartet. Aus Sicht der Vorhersage handelt es sich um ein Regressionsproblem. Die Zeit zur Erstellung der Vorhersage ist nicht kritisch, sie wirkt sich jedoch auf die benötigte Zeit zur Planungserstellung aus. Die Modelle für die Fertigungsmaschinen liegen zum Zeitpunkt der Vorhersage bereits vor. Die Anzahl der abhängigen Attribute ist, wie in Kapitel 4.2 erläutert, überschaubar. Der Aufwand um die Vorhersage zu berechnen wird daher moderat sein. Aus der moderaten Reaktionszeit und dem moderaten Rechenaufwand ergeben sich moderate Anforderungen an die Infrastruktur.

3.4.2. Dynamisches Ressourcenmanagement (DynaRes)

Anstatt einer statischen Planung wird ein dynamisches Ressourcenmanagement betrieben, das die Belegung aufgrund veränderter Bedingungen anpassen kann. Änderungen sind unter anderem erforderlich, wenn weitere Tasks zeitnah einzuplanen sind, aktive Tasks ihren Ressourcenverbrauch ändern oder sich Parameter der Bearbeitung durch die Fertigungsmaschinen ändern. Die Belegung der Kapazitäten wird durch das Planungssystem erstellt. Die zu planenden Aufträge und deren Tasks werden dynamisch in die Planung aufgenommen. Prioritäten der Tasks werden dabei berücksichtigt, eine optimale Kapazitätsauslastung wird angestrebt. Dabei werden beim Eintreffen der Aufträge die zu belegenden Ressourcen benötigt, somit auch der Energieaufwand. Eine Prognose des Energieaufwands wird für geplante Belegungen einmalig ermittelt. Bei der Entscheidung über die Belegung wird der Energieaufwand berücksichtigt. Der tatsächliche Ressourcenverbrauch wird erfasst. Abweichungen von der geplanten Belegung werden während der Fertigung überwacht und behandelt.

Anforderung an die Vorhersage: Anders als bei StaPla ist die Zeit für die Erstellung der Vorhersage kritisch. Die Trägheit der Ressourcenplanung verringert maßgeblich die Effektivität des Ressourcenmanagers. Daher ist eine geringe Reaktionszeit

notwendig, was höhere Anforderungen an die Infrastruktur stellt. Um die Infrastruktur zu entlasten, kann die Anzahl an untersuchten Belegungsszenarien begrenzt werden.

3.4.3. Simulation eines dynamischen Ressourcenmanagements (SimDynaRes)

Zusätzlich zum dynamischen Ressourcenmanagement wird die Belegung im Zeitverlauf simuliert. So kann zu jedem Zeitpunkt der prognostizierte Ressourcenverbrauch beziffert werden. Die Belegung der Kapazitäten wird durch das Planungssystem erstellt und simuliert. Die zu planenden Aufträge und deren Tasks werden dynamisch in die Planung aufgenommen. Die Prioritäten der Tasks werden berücksichtigt mit dem Ziel einer optimalen Kapazitätsauslastung. Dabei werden beim Eintreffen der Aufträge die zu belegenden Ressourcen benötigt, dies betrifft auch den Energieaufwand. Eine Prognose des Energieaufwands wird für mögliche Belegungen einmalig ermittelt. Bei der Entscheidung über die Belegung wird der Energieaufwand berücksichtigt.

Anforderung an die Vorhersage: Anders als bei StaPla und DynaRes beschäftigt sich die Vorhersage nicht mit einzelnen Attributen. Um die Belegung zu simulieren wird ein Zeitverlauf prognostiziert, dieser beschreibt den Energieaufwand je Zeitabschnitt (Takt). Dies setzt eine erweiterte Datenbasis voraus, die Zeitverläufe der abhängigen Attribute enthält. Dies erfordert eine rechenintensive Vorhersage von Zeitreihen. Um die Anforderungen an die Infrastruktur zu verringern können Einschränkungen der Dynamik sinnvoll sein.

3.4.4. Variationen der Use Cases

Remod: Eine Variation der Use Cases ist die Berücksichtigung von Änderungen an den Maschinenparametern, so können sich durch Verschleißerscheinungen Parameter der Maschinen-Werkzeuge verändern.

Ereignisse: Als Basis für Vorhersagen kann Event-Detection verwendet werden. Insbesondere für SimDynaRes kann sie vorteilhaft sein um die Datenmengen zu reduzieren.

Timeseries als Basis: Als Basis für StaPla und DynaRes können Zeitreihen verwendet werden, so könnte der Energieverbrauch als Messreihe von Energieaufnahmen anstatt als Energieaufwand (einzelnes Attribut) zur Verfügung stehen.

Bereich	StaPla	DynaRes	SimDynaRes
Basis-Daten	Attribute	Attribute	Komplex
Ziel-Daten	Attribute	Attribute	Komplex
Vorhersage-Problem	Regression	Regression	Zeitreihen-Regression
Reaktionszeit	Moderat	Hoch	Moderat
Vorhersage-Aufwand	Moderat	Moderat	Sehr Hoch
Dynamik	Statisch	Statisch	Statisch
Infrastruktur-Anforderung	Moderat	Hoch	Hoch

Tabelle 9.: Anforderungen je Use Case

Bereich	mögliche Anforderungen	
Basis-Daten + Ziel-Daten	Attribute	Komplex
Vorhersage-Problem	Regression	Zeitreihen-Regression
Reaktionszeit	Moderat	Hoch/Nahe Echtzeit
Dynamik	Statisch	Dynamisch

Tabelle 10.: Anforderungsvariationen der Use Cases

3.5. Anforderungen an die Vorhersage

Neben den funktionalen Anforderungen des Planungssystems, wird das System folgende nichtfunktionale Anforderungen erfüllen:

Flexible Anwendbarkeit. Das System wird nach der Maßgabe entworfen, eine hohe Adaptionfähigkeit zu besitzen. Es wird generische Komponenten vorsehen, die die Vorhersage des Energieaufwands unabhängig von der genutzten Fertigungsmaschine machen. Konkret kann es in der Domäne 'Fertigung' auf Fräsmaschine, Schleifmaschinen, Schweißmaschinen oder 3D-Drucker angewendet werden.

Domänenwissen. Die Adaption soll nur ein geringes Maß an Domänenwissen erfordern. Im Vergleich zu parametrisch-mathematischen Modellen und holistischen Simulationen, die auf umfangreichem physikalischen Wissen beruhen.

Empirische Daten. Es soll davon ausgegangen werden, dass gelabelte empirische Daten vorhanden sind. Diese Messwerte des Energieaufwands liegen als getakteter Datenstrom bzw. einer Zeitreihe vor. Vom Produktionsplanungssystem können strukturierte Daten bezogen werden. Ebenfalls liegen die Datenblätter der Fertigungsmaschine sowie Umgebungsmessungen vor.

Die funktionalen Anforderungen des Planungssystems an die Vorhersage fasst Tabelle

10 zusammen.

3.6. Anforderungen an die Architektur

Die Produktionsplanung erstellt eine Menge an Maschinenbelegungspläne und entscheidet, welche der Pläne umgesetzt wird. Dabei entstehen je nach Anzahl und Gewichtung von Ressourcen (Maschinen, Maschinenkonfigurationen, Aufträgen) und Zielen (Auslastung, Gewinn) eine Menge an Alternativen. Jede dieser Alternativen beinhaltet wiederum eine Menge an Tasks, für die eine Energieaufwands-Prognose erstellt wird. Aus den Anforderungen des Planungssystems ergibt sich folgendes Profil für die System-Architektur:

Skalierbarkeit. Die Anzahl der zu erstellenden Prognosen kann stark variieren.

Kurze Bereitstellungszeit. Die Komponenten zur Erstellung der Prognosen (Digitale Zwillinge), müssen je nach Use Case moderate Reaktionszeiten bis hin zu Echtzeit-Reaktionszeiten erreichen. Die Reaktionszeit umfasst dabei Bereitstellung und Berechnung der Prognose. Folgerung: Die Zeit zur Bereitstellung der Komponente ist zu minimieren, da sie die Erstellung der Prognose verlängert. Eine gesonderte Lösung für die Echtzeit-Anforderung ist zu entwickeln.

Flexibilität bei Änderungen der modellierten Systeme. Fertigungsmaschinen können im Zeitverlauf Verschleiß aufweisen oder Umbauten erfahren. Dies führt zu Änderungen der Funktionsweise und somit veränderten Produktionsparametern. Auf diese und weitere Änderungen der modellierten Systeme soll das System vorbereitet sein.

Deployment. Ein effektives Deployment der Vorhersage-Komponenten soll in der Architektur berücksichtigt werden.

3.7. Abstraktion

Die Verknüpfung zwischen Physischer- und Cyber-Welt stellt unter anderem Messwerte, Modelldaten, physikalische Gesetze und Simulationen dar. Die Abstraktion der realen Welt bildet eine grundlegende Rahmenbedingung bei der Erstellung von Modellen, sie bildet die Datenbasis. Es ist anzunehmen, dass die Qualität des Modells proportional zur Qualität der Daten verläuft. Die Annahmen zur Qualität gliedern sich in folgende Aspekte:

Exaktheit der Label. Das System dieser Arbeit setzt gelabelte Daten voraus. Die Korrektheit der Label soll als hoch angenommen werden.

Anzahl Datensätze. Aufgrund des hohen Aufwand zu Erfassung von empirischen Produktionsdaten, ist die Anzahl der gelabelten Datensätze je Maschinenkonfiguration als gering, Größenordnung 50, anzunehmen.

Präzision der Daten. Die Präzision der Daten ist als exakt anzunehmen.

Aussagekräftigkeit. Die Untersuchung, welche Daten relevant sind, ist Teil dieser Arbeit.

Die Herkunft der Daten bestimmt darüber inwieweit sie vor Verwendung interpretiert werden müssen (Pre-Processing). Dies kann Parsen, rechnerisches Aufbereiten oder Fehleradaption umfassen. Annahmen für die jeweiligen Daten-Quellen sind wie folgt:

Empirische Daten. Liegen als rohe Messdaten vor, in der Regel als Datenstrom. Ein hohes Maß an Pre-Processing ist anzunehmen.

Katalog Daten. Informationen die strukturiert abgelegt sind, in Dokumenten oder Datenbanken. Zum Beispiel können dies Daten aus Datenblättern des Herstellers sein. Sie liegen als Fremddaten vor, in lokalen oder entfernten Systemen. Sie sind heterogen strukturiert und müssen teils interpretiert (geparst) werden.

Erzeugte strukturierte Daten. Daten die von eigenen Systemen erzeugt werden. Dabei kann auf deren Struktur Einfluss genommen werden. Dies sind i.d.R. Daten des Auftrags, die vom Planungssystem bereitgestellt werden.

Systemfremde erzeugte stukturtrierte Daten. Strukturierte Daten die aus Fremdsystemen stammen. Eine homogene Datenstruktur kann vorliegen, sie ist zur direkten Weiterverarbeitung ungeeignet. Pre-Processing Maßnahmen zur Adaption werden benötigt.

3.7.1. Produktionsauftrag

Die Tasks der einzelnen Aufträge enthalten administrative und produktionsbezogene Informationen und sollen Teil der Datenbasis zur Vorhersage des Energieaufwands sein. Sie bilden den Auftrag auf einzelne Planungselemente ab. Einfluss auf die Planung nimmt dabei die Priorität des Auftrags, in dem dessen Tasks entsprechend der Priorität eingeplant werden. Weitere Parameter können Termin-Erreichung oder Tageszeit-Vorgaben sein. Produktionsbezogene Informationen beschreiben wie das Werkstück

in der Fertigungsmaschine durch das Maschinen-Werkzeug bearbeitet werden. Teile davon sind die Parameter des Werkstück-Materials, Anforderungen an die Umgebung sowie das CNC-Programm.

3.7.2. Produktionsmaschine

Es soll davon ausgegangen werden, dass für die Produktionsmaschinen Datenblätter, die alle notwendigen Parameter beinhalten, vorliegen. Die Produktionsparameter lassen sich durch Konfiguration der Maschinen anpassen. Die Produktion kann durch die Umgebung beeinflusst werden.

3.7.3. Energieaufwand

Produktionsmaschinen bestehen aus einer Vielzahl an energiekonsumierenden Komponenten, wie den Maschinen-Werkzeugen, dem zentralen Antrieb oder den Kühlsystemen. Die Energieaufnahme der Komponenten variiert mit deren Verwendung. Sie tragen indirekt oder direkt zur Bearbeitung bei. Des Weiteren gibt es durch ungewollte Energieumwandlungen Energieverluste, dies können Reibungsverluste in Getrieben oder Wärmeverluste in Heiz-Einheiten sein [Sch14]. Um die Energie zur Herstellung oder Bearbeitung eines Werkstücks zu erhalten, kann, wie in Kapitel 2 beschrieben, auf folgende Weisen vorgegangen werden:

Physikalisches Modell. Beschreibt wie die aufgenommene elektrische Energie in Wärme oder kinetische Energie zur Bearbeitung des Werkstücks umgewandelt wird.

Modelldaten des Herstellers. Legt den Energieaufwand für die Bearbeitung des Werkstücks in Abhängigkeit zu den Bearbeitungsparametern fest.

Empirische Messung. Messung der elektrischen Energieaufnahme während der Bearbeitung des Werkstücks. Dabei wird die aufgenommene elektrische Leistung (Watt) getaktet (i.d.R. Sekundentakt) gemessen. Die Messwerte der Bearbeitungszeit ergeben kumuliert die elektrische Arbeit bzw. Energie (Wattstunde).

In Kapitel 2 ist ersichtlich, welche Herangehensweisen sich zur empirischen Erfassung des Energieverbrauchs etabliert haben. Bezogen auf die Fertigung eines Werkstücks sind dies folgende:

Messung an den einzelnen elektrischen Bauteilen. Relevante elektrische Bau-

teile der Maschine werden an festgelegten Messpunkten vermessen. Wird für die Erstellung von holistischen Simulationen angewendet.

Messung der Komponenten. Relevante Baugruppen der Maschine werden an deren Stromaufnahme-Verbindungen vermessen. Wird für gekapselte Simulationen verwendet.

Messung der Maschinen-Werkzeuge. Nur die Baugruppen die das Werkstück bearbeiten werden vermessen. Dies sind die Maschinen-Werkzeuge.

Messung der Maschine. Die Stromaufnahme der gesamten Fertigungsmaschine wird an ihrem Stromanschluss zum Hausnetz vermessen.

Der Aufwand, um die Messung an der Maschine durchzuführen, ist von allen Verfahren die geringste. Zum Aufbau des Messstandes wird kein elektrisches Fachwissen benötigt. Zudem ist kein Eingriff in die Maschine notwendig, dies bringt Vorteile bei Sicherheitsaspekten und rechtlichen Vorgaben mit sich. Das System soll dafür ausgelegt sein, die Prognosen auf Basis der Messdaten der gesamten Maschine zu erstellen.

4. Modellierung des Energieaufwands

In diesem Kapitel werden zunächst Verfahren entwickelt, um die Abstraktion der physischen Daten strukturiert durchzuführen. Es wird analysiert, welche Daten in welcher Form zu Verfügung stehen. Diese werden strukturiert und relevante Daten der Nutzung im Modell zugeordnet. Das Verfahren Datenschablone zur systematischen Erfassung wird entwickelt. Es folgt eine Analyse zur Erstellung der Vorhersage mittels maschinellem Lernens. Schritte zur Erstellung und Weiterentwicklung von neuronalen Netzen werden besprochen.

4.1. Aufbau der Datenbasis

Im Folgenden wird die Datenbasis diskutiert. Eine Analyse zur Abstraktion der Elemente Produktionsauftrag, Produktionsmaschine und Energieaufwand wird durchgeführt. Dabei wird die Akquirierung von relevanten Daten untersucht, sowie die erhaltenen Daten beschrieben. Die Quellen werden nach den in Kapitel 3 aufgestellten Kriterien des Pre-Processing klassifiziert. Der zu erwartende Prozess der Aufbereitung wird umrissen.

4.1.1. Abstraktion des Produktionsauftrags

Als Datenquelle werden Information des Produktions-Planungs-Systems genutzt. Diese bestehen aus dem Auftrag, respektive den daraus abgeleiteten Tasks. Sie beinhalten welche Vorgänge auf das Werkstück angewendet werden. Teil des Auftrags sind auch Qualitätsvorgaben, die sich in nachträglichen Änderungen von Bearbeitungsparametern (Geschwindigkeit) oder zusätzlichen Parametern (Nachbearbeitung) manifestieren. Weitere Parameter werden generiert um die Bearbeitung technisch umzusetzen (Hilfsmaterial). Die Ablaufplanung erzeugt oder ändert ebenfalls Parameter (Priorität, zeitliche Abbruchkriterien).

Bei Task-Daten kann zwischen Werkstück-Daten und Bearbeitungsparametern unterschieden werden. Die Werkstück-Daten beinhalten eine Beschreibung der geome-

Modell-Art	Beispiel Format	Bearbeitungsdaten
2D Model	DXF	Nein
3D Modell(Prop)	CATProduct	Optional
3D Modell(ISO)	STP	Ja
Mesh	STL, OBJ	Nein
CNC-Programm	g-code	Ja

Tabelle 11.: Modellformate

trischen Form, sowie Beschreibung des Materials, aus dem das Werkstück bestehen soll. Die geometrische Form, ausgedrückt als Modell, durchläuft in seinem Modellierungsprozess Stufen der Entwicklung, jede erzeugt dabei ein Modell unterschiedlichen Formats. Zumeist lassen sich die Modelle nicht mehr in die vorherige Form überführen. So ist der Übergang der Beschreibungsmethode von Algorithmen zu Dreiecks-Netzen (Mesh) irreversibel. Ziel aller Schritte ist die Erstellung eines CNC-Programms. Tabelle 11 gibt eine Übersicht über einige Formate bzw. Schnittstellen zur Objektbeschreibung [Bob] [Schb].

Wie ein Modell gefertigt werden soll, beschreiben die Bearbeitungsparameter. Sie beinhalten unter anderem Informationen über Toleranzen, Materialeigenschaften, Texturen, Materialarten und Topologien des Objekts. Diese Information können implizit in der Objektbeschreibung enthalten sein, so stellt ein g-Code den gesamten Bearbeitungsprozess dar. In Mesh-Dateien hingegen fehlen diese Informationen und müssen explizit in peripheren Dateien definiert werden.

Typ: Strukturierte Fremd-Daten.

Die Tasks können auf Fremdquellen (Stücklisten-Datenbanken) verweisen, in interpretierbare Formate übertragen werden oder aufwandsvoll interpretiert werden. Sie müssen daher teilweise aufbereitet werden. Dies sollte der gleiche Prozess sein, wie zur Aufbereitung der Daten für die spätere Produktion. So kann es erforderlich sein, das Gewicht des Werkstücks anhand von Materialart und Volumen zu berechnen. Dazu kann das relative Gewicht des Materials durch Verweis auf die Materialart aus einer Fremddatenbank geladen werden. Des Weiteren kann eine Umrechnung von Maßeinheiten stattfinden, zum Beispiel eine Konvertierung der Bemaßung aus dem angloamerikanischen (Inch etc.) in das metrische System.

In dieser Arbeit wird eine eindeutige 1 zu 1 Zuordnung von Task zu Task-Beschreibung angenommen. Die Tasks verweisen nicht auf andere, somit ändern sich sämtliche Informationen von Task zu Task.

4.1.2. Abstraktion von Produktionsmaschinen

Zur Beschreibung von Produktionsmaschinen können Datenblätter der Hersteller herangezogen werden. Nach Umbauten an der Maschine können diese aktualisiert werden.

Diese Daten stellen den Zustand der Maschine nach Auslieferung dar. Zu erwartende Änderungen aufgrund von Verschleiß und ähnlichem können ebenfalls in den Datenblättern enthalten sein. Da tatsächliches Verhalten der Maschine von vielen Faktoren abhängig ist, können die Datenblätter nur als Referenz dienen. Die Produktionsstunden und weitere Parameter die den aktuellen Zustand beschreiben sind notwendig, um die Maschine exakt zu erfassen.

Eine Reihe von Parametern ist von der Umgebung abhängig. So ist die Fertigungs-Präzision von Erschütterungen oder Vibrationsdämpfung, der energetische Aufwand um ein Werkzeug aufzuheizen von der Umgebungstemperatur abhängig. Diese Parameter stehen als Umgebungsbeschreibung zu Verfügung, teilweise als Zeitverlaufsdaten bzw. Datenstrom aus Messdaten.

Typ: Strukturierte Fremd-Daten & unstrukturierte Fremd-Daten (Datenstrom).

4.1.3. Abstraktion des Energieaufwands

Der Aufwand zur Bearbeitung bzw. Herstellung des Werkstücks wird empirisch erfasst. Nach Aufbereitung der Daten liegt eine Zeitreihe von Messpunkten vor, die den Stromverbrauch während der Bearbeitung beschreiben. Dies kann direkt verwendet werden, wie im Use Case der Simulation (SimDynaRes) beschrieben wurde.

Typ: Unstrukturierte Fremd-Daten (Datenstrom).

Um den gesamten Energieaufwand zur Herstellung eines Werkstücks zu erhalten, werden die einzelnen Messungen zu einem kumulierten Wert zusammengefasst.

Typ: Strukturierte Daten.

4.1.4. Klassifizieren der Datenbasis

Die Datenbasis zur Modellierung des Energieaufwands umfasst alle dafür notwendigen Informationen, bestehend aus der Abstraktion der Fertigungsmaschine, des Fertigungstasks und des resultierenden Energieaufwands. Inwiefern das Modell von In-

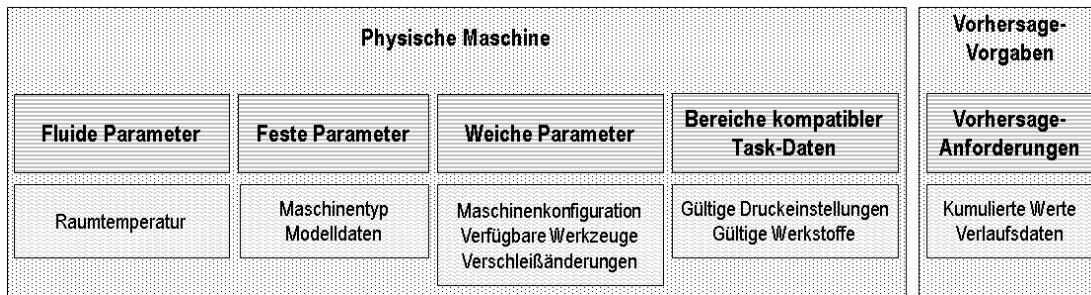


Abbildung 9.: Maschinenabstraktion

formationen abhängt, ist von Parameter zu Parameter unterschiedlich, ebenso die Variabilität der Daten.

Aus der Maschinenabstraktion ließe sich ein vollständiges Modell entwickeln. Die Task-Daten könnten als abhängige Attribute bzw. Features als Input für die Prognose genutzt werden.

Jedoch ändern sich manche Parameter der Maschine. Erfordert die Bearbeitung eine neue Konfiguration mit geringer Rüstzeit des Maschinenwerkzeugs, ändert sich der Zustand der Maschine von Task zu Task. Zudem müssen die Umgebungsparameter in die Prognose einfließen.

Eine differenzierte Untersuchung der Daten, die zur Modellierung herangezogen werden und die als abhängige Parameter bzw. Features dienen, ist notwendig. Schaubild 9 fasst diese Abstraktion zusammen.

Die Information über die Fertigungsmaschine und deren Umgebung werden nach deren Variabilität und Auswirkung klassifiziert. Dafür werden die Klassen Fluid, Weich und Fest eingeführt.

Fluide Parameter ändern sich häufig. Sie liegen als Messreihe vor. Beispiele hierfür ist die Umgebungstemperatur.

Weiche Parameter variieren oft, beispielsweise mit jedem Task der bearbeitet wird. So kann ein Wechsel eines Teils des Werkzeugs, wie der Fräskopf oder einer Druckdüse einige Parameter ändern. Die Rüstzeit ist gering. Diese Änderungen bleiben jedoch auf Subsysteme beschränkt.

Feste Parameter werden einmalig festgelegt. Die Rüstzeit ist hoch. Eine Änderung führt zu einer Reihe von Änderungen in weiteren Parametern.

Die weiteren Klassen sind:

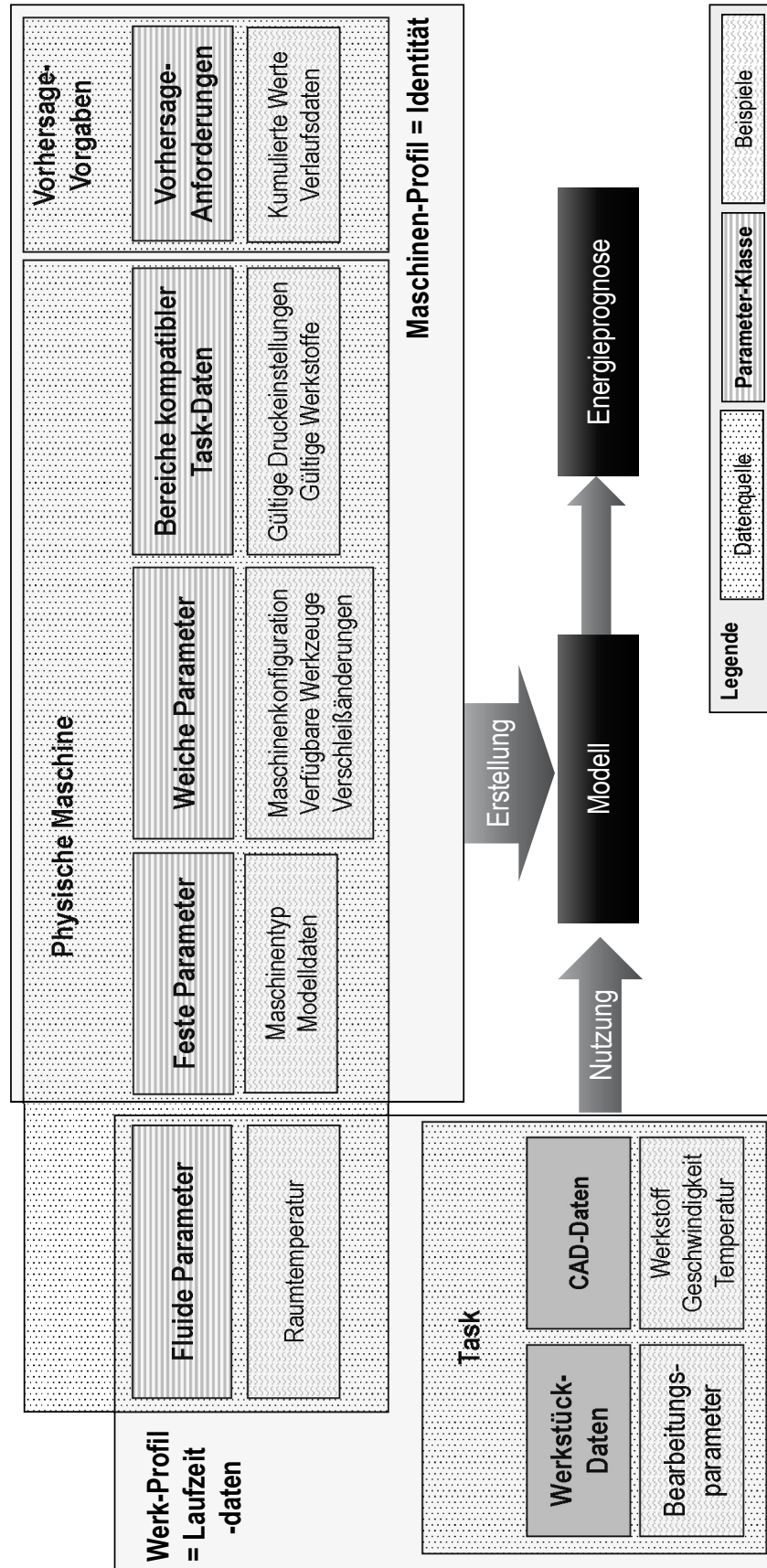


Abbildung 10.: Von der Abstraktion zum Modell

Die Task-Daten des Produktionssystems werden in zwei Klassen unterteilt. Analog zur Analyse aus Abschnitt 4.1.1 sind dies die Werkstück-Daten, die das Werkstück als Objekt beschreiben (z.B. CAD-Modell). Zuzüglich dazu beschreiben die Bearbeitungsparameter, wie das Werkstück-Objekt zu einem physischen Werkstück wird (z.B. Bearbeitungsgeschwindigkeit) .

Energiedaten. Für die Energiedaten leitet sich eine eigene Klasse ab. Der Inhalt ändert sich für jede Prognose und ist von den Task-Daten abhängig. Die Eigenschaften der Klasse ändern sich abhängig vom Use Case. So kann es sich um ein einfaches Daten-Tupel handeln, das den Energieaufwand der Produktion darstellt oder um eine Reihe von Tupeln, die den Zeitverlauf der aufgewendeten Leistung darstellen.

Gültige Produktionsbereiche. Eine weitere Klasse stellen die gültigen Produktionsbereiche dar. Sie ergeben sich aus den Maschinen-Parametern und den Bearbeitungsparametern. Da deren Erstellung überwacht werden muss und einen erheblichen Aufwand darstellt, werden sie als redundante Daten abgelegt. Da sie auch dazu dienen kompatible Modelle für Tasks zu finden, wird somit ein schneller Zugriff ermöglicht.

Die weichen und festen Parameter der Maschine, sowie die gültigen Produktionsbereiche bilden zusammen mit den Energiedaten das Maschinen-Profil. Es bildet die Identität einer Maschine ab. Eine Änderung darin führt zu einer neuen Identität und daher zu einem neuen Modell.

Die fluiden Parameter der Maschine, sowie die beiden Klassen (Werkstück + Bearbeitung) der Task-Daten, Werkstück-Daten und Bearbeitungsparameter bilden das Werk-Profil. Das Modell kann sich nicht auf die Daten stützen. Sie bilden die abhängigen Attribute bzw. Features, von denen die Prognose abhängt. Sie hängen mit dem Werkstück direkt zusammen, sowie dessen Bearbeitung-Umstände.

Schaubild 10 gibt einen Überblick über die Klassifizierung, wie die Klassen zusammengefasst und für die Nutzung der Prognose verwendet werden.

4.2. Vorgehensmodell: Datenschlablone

Zur Erstellung eines digitalen Zwillings wird ein Maschinen-Profil benötigt, respektive zur Nutzung ein Werkprofil. Domänenwissen ist notwendig um die Parameter zu identifizieren, die den Energieverbrauch beeinflussen.

Dieser Schritt bildet die Basis des digitalen Zwillings und muss daher Anforderungen wie Nachvollziehbarkeit und Vollständigkeit erfüllen. Die wiederholte Analyse derselben Maschine sollte gleiche Ergebnisse erzielen. Weder überflüssige Parameter sollten einbezogen werden, noch relevante Parameter ausgeschlossen werden, um die Vorhersagequalität zu gewährleisten.

4.2.1. Entwicklung des Vorgehensmodells

Im Folgenden wird ein Vorgehen entwickelt und exemplarisch durchgeführt, mit dem Ziel es auf Maschinen anwenden zu können, unabhängig ihrer Domäne. Es besteht aus zwei Abschnitten.

Im ersten Schritt wird eine Schablone erstellt, die alle relevanten Daten der Domäne abbildet. Dabei werden Daten von physischen Maschinen erfasst und klassifiziert. Das Werkstück wird auf relevante Kriterien hin untersucht. Der Task wird anhand der Werkstück-Daten und der Bearbeitungs-Parameter untersucht. Die Anforderungen an die Vorhersage werden erfasst.

Aus den entstandenen Klassen werden im zweiten Schritt das Maschinen-Profil und das Werkstück-Profil abgeleitet.

1. Erstellung einer Schablone

1.1. Festlegung der exakten Domäne

1.2. Analyse des Produktionsprozesses

1.3. Erfassen der Parameter und Einteilen in Klassen

1.3.1. Maschinen-Sicht

1.3.1.1. Welche Parameter definieren die Maschine?

1.3.1.2. Welche Parameter der Maschine korrelieren mit dem Energieaufwand?

1.3.1.3. Welche Werkzeuge sind installiert?

1.3.2. Werkzeug-Sicht:

1.3.2.1. Welche Parameter definieren das Werkzeug?

1.3.2.2. Welche Parameter des Werkzeugs korrelieren mit dem Energieaufwand?

1.3.3. Werkstoff-Sicht

1.3.3.1. Welche Parameter definieren den Werkstoff?

1.3.3.2. Welche Parameter des Werkstücks korrelieren mit dem Energieaufwand? Hierbei ist das Anlegen einer Werkstoff-Datenbank sinnvoll.

1.3.4. Task-Sicht

1.3.4.1. Welche Anforderungen gibt es an den Produktionsprozess, jeweils für Maschine, Werkzeug, Werkstoff und Umgebung?

1.3.4.2. Zentral ist das CAD-Programm, das aus dem 3D-Modell erzeugt wird. Darin enthalten ist der Tool-Path, eine programmatische Beschreibung des Wegs, den die jeweiligen Werkzeuge nehmen. Auf diesem Weg aktivieren sie verschiedene Fähigkeiten, dies kann die Erhöhung der Fräser-Drehzahl oder die Temperaturerhöhung des Druckkopfs sein. Beim Fräsen führt der Weg durch das Werkstück hindurch, beim 3D-Druck wird durch Abfahren des Wegs das Werkstück erzeugt.

1.3.5. Prognoseparameter

Die Anforderungen an die Vorhersage finden sich in den Prognoseparametern. Grundsätzlich handelt es sich dabei um Regression-Vorhersagen, jedoch mit Unterscheidungen an Komplexität, Aktualität und Form. So kann entweder ein kumulierter Wert gefordert sein, der die Stromvorhersage über den Zeitraum der Werkstoff-Bearbeitung liefert oder der Verlauf des Energieverbrauchs während der Werkstoff-Bearbeitung. Grundlage für letzteres wären einige tausend Datensätze, der Energiebedarf im Sekundentakt. Denkbar hierbei sind auch Vorhersagen darüber, wann welches Werkzeug verwendet wird oder wann das Werkzeug welche Handlung vornimmt.

2. Ableiten von Maschinen-Profil und Werkstück-Profil

Die Parameter werden in Klassen unterteilt. Die Daten der Maschine werden dabei in fluide Parameter, weiche Parameter und harte Parameter klassifiziert. Fluide Parameter haben die Eigenschaft sich anhaltend zu ändern. Umgebungs-

parameter fallen in der Regel in diese Klasse. Harte Parameter stehen charakteristisch seit Herstellung der Maschine fest oder lassen sich nur durch Umbau der Maschine ändern.

Die zwei Klassen der Task-Daten bilden die Werkstück-Daten und die CAD-Daten. Das Werkstück wird aus den CAD-Daten gebildet, wobei die Bearbeitungsparameter definieren wie die CAD-Daten physisch umgesetzt werden.

2.1. Maschine

2.1.1. FP - Fluide Parameter

2.1.2. WP - Weiche Parameter

2.1.3. HP - Harte Parameter

2.2. Taskdaten

2.2.1. WD - Werkstück-Daten

2.2.2. BP - Bearbeitungsparameter

2.3. PP - Prognoseparameter

Das Maschinen-Profil wird aus den Klassen weiche Parameter, harte Parameter und Prognoseparameter; das Werkstückprofil aus den Klassen fluide Parameter, Werkstück-Daten und Bearbeitungsparameter gebildet.

4.3. Vorhersage des Energieaufwands

Vorhersagen über die Reaktion von Systemen basieren auf einer Modellierung. Diese kann, wie in Kapitel 2 beschrieben, als holistische Simulation oder mathematisches Modell erfolgen.

4.3.1. Modellierung mittels maschinellen Lernens

Wird die Modellierung als Regressionsproblem betrachtet, können Machine-Learning-Verfahren angewendet werden. Ein von Cavalieri et al. [CMP04] durchgeführter Vergleich ergibt eine höhere Effizienz von neuronalen Netzen gegenüber parametrisierten Modellen.

Zur Erstellung von Modellen mittels Machine-Learning ist weniger Domänen-Wissen notwendig. Eine einfache Analyse der Fertigungsprozesse ist ausreichend, um zwei grundlegende Schritte im Machine-Learning-Verfahren durchführen zu können:

Grundlagen auf Basis von Domänen-Wissen schaffen.

Dies ist als Basis für Modellierungsüberlegungen notwendig, so kann der zu modellierende Prozesse in Unterprozesse aufgeteilt, passende Machine-Learning-Verfahren eruiert oder relevante Messpunkte für empirische Daten festgelegt werden.

Erstellen einer Datengrundlage.

Die Gesamtheit der Daten wird geschaffen. Dies können empirische Daten sein, die gezielt erfasst werden müssen oder Fremd-Daten, die extrahiert werden müssen.

Bei neuronalen Netzen ist eine spätere Anpassung des Modells ohne Domänenwissen möglich. Sollten sich Parameter des Systems ändern, kann das Modell nach trainiert werden. Die Basis der Daten kann im produktiven Ablauf erweitert werden und das Modell wiederum trainiert werden.

Ein wesentlicher Nachteil von neuronalen Netzen ist die fehlende Nachvollziehbarkeit der Ergebnisse. Weitere Vor- und Nachteile sind bereits in der Literatur ausführlich diskutiert worden [BHM], [Pat97], [Nic]. In dieser Arbeit werden neuronale Netze zur Vorhersage des Energieaufwands genutzt.

4.3.2. Architektur des neuronalen Netzes

Im Folgenden werden vier Architekturen von neuronalen Netzen auf deren Eignung hin betrachtet.

Mehrschichtige Feedforward-Netze (MLP)

Sie bestehen aus einer oder mehreren Schichten von Neuronen. Die Daten werden dem Input-Layer zugeführt, ein oder mehrere Hidden-Layer abstrahieren die Daten. Die Vorhersagen werden durch den Output-Layer getroffen. MLPs sind für Klassifikationsprobleme passend, insbesondere wenn die Daten gelabelt sind oder Klassen zugeordnet sind. Sie eignen sich auch für Regressionsprobleme, bei denen eine reale Größe aufgrund einer Reihe von Eingaben vorhergesagt wird. Die Daten werden oft in einem tabellarischen Format bereitgestellt [Jas19]. Generell sind sie flexibel und eignen sich für eine Verknüpfung der Ein- und Ausgaben. Sie eignen sich für Bilddaten,

Textdaten, Zeitreihendaten und andere Arten von Daten [Fro18].

Convolutional Neural Networks (CNN)

Convolutional Neural Networks wurden entwickelt, um Bilddaten auf eine Ausgabevariable abzubilden. Sie haben sich als so effektiv erwiesen, dass sie die ideale Methode für jede Art von Vorhersageproblem mit Bilddaten als Input sind. Der Vorteil der Verwendung von CNNs ist ihre Fähigkeit, eine interne Darstellung eines zweidimensionalen Bildes zu entwickeln. Dadurch kann das Modell die Position und Skalierung in Variantenstrukturen in den Daten erlernen, was bei der Arbeit mit Bildern wichtig ist. Generell funktionieren CNNs gut mit Daten, die einen räumlichen Bezug haben. Obwohl sie nicht speziell für Nicht-Bilddaten entwickelt wurden, werden CNNs erfolgreich bei Problemen wie der Dokumentenklassifizierung und der Sentiment-Analyse verwendet. Sie eignen sich für Textdaten, Zeitreihendaten und Sequenzeingangsdaten. [Jas19]

Recurrent Neural Networks (RNN)

Rekurrente Netzwerke oder rückgekoppelte Netzwerke besitzen zusätzliche Verbindungen, durch die Informationen bestimmte Bereiche des Netzwerkes auch rückwärts oder wiederholt durchlaufen können. Sie eignen sich zur Informationsverarbeitung von Sequenzen (Sequence-to-Sequence Netze) [Jul19].

Recurrent Neural Networks wurden für die Sequenzvorhersage entwickelt. Sequenzvorhersageprobleme lassen sich nach Art der unterstützten Ein- und Ausgänge beschreiben. Einige Beispiele für Probleme bei der Sequenzvorhersage sind [Jas19]:

1 zu n Problem: Eine Beobachtung als Input, die einer Sequenz mit mehreren Schritten als Output zugeordnet ist.

n zu 1 Problem: Eine Folge von mehreren Schritten als Eingabe, die auf Klassen- oder Mengenvorhersage abgebildet wird.

n zu n Problem: Eine Folge von mehreren Schritten als Eingabe, die auf eine Folge mit mehreren Schritten als Ausgabe abgebildet wird.

RNNs sind schwer zu trainieren. Diesen Nachteil weisen die speziellen RNNs, die Long Short-Term Memorys (LSTM), nicht auf. RNNs und LSTMs werden unter anderem für die Vorhersage von Zeitreihen entwickelt, wobei die Ergebnisse noch nicht ausreichend sind. [Jas19]

Für die Vorhersage des Energieverbrauchs für die jeweiligen Use Cases ergeben sich

folgende erste Orientierungen.

Statische Planung (StaPla) Die Verwendung eines mehrschichtigen Feedforward-Netzes ist angezeigt.

Dynamisches Ressourcenmanagement (DynaRes) Die Verwendung eines mehrschichtigen Feedforward-Netzes ist angezeigt.

Simulation (SimDynaRes) Die Verwendung eines Convolutional Neural Networks oder Recurrent Neural Networks ist angezeigt.

4.3.3. Umsetzung der Architektur und Feature Extraktion

Die konkrete Architektur des Modells beschreibt den Aufbau des neuronalen Netzwerks. Dies umfasst die Anzahl der Neuronen in den einzelnen Layern, die Anzahl an Hidden-Layer, sowie die Aktivierungsfunktion der Neuronen in den Hidden-Layers. Dabei beschreibt die Aktivierungsfunktion aus Sicht des Neurons an welche Ausgänge bzw. wie stark das gewichtete Eingangssignal weitergegeben wird. Zudem wird die Funktion festgelegt, die zur Berechnung der Fehlerabweichung verwendet wird [Jas19].

Neben einem singulären Netzwerk sind auch Konstruktionen möglich, bei denen mehrere Netzwerke unterschiedlicher Art verwendet werden. Denkbar wäre eine Baumstruktur, wobei die Zweige jeweils durch ein Netzwerk repräsentiert werden, die eine Teilmenge der Features bearbeiten. Die Ausgaben der verzweigten Netzwerke sind mit den Inputs der übergeordneten Netzwerke gekoppelt. Den Wurzelknoten des Baums stellt ebenfalls ein Netzwerk dar, das die finale Ausgabe bzw. die Vorhersage liefert.

Die Entwicklung der konkreten Architektur ist von Faktoren wie den Input-Daten abhängig. Diese werden durch Feature-Extraktion dem Input-Layer zugeführt [Fro18].

4.3.4. Anpassung des Modells

Manche Parameter, aus Sicht des Modells sind dies die Features, des Maschinen-Profiles können sich ändern, welche dies sind hängt vom Anwendungsfall ab. Damit ändert sich die Identität des Modells bzw. des digitalen Zwillings. Die Änderungen sind in der Daten-Klasse Weiche-Parameter zu erwarten. Dies bringt Probleme mit sich, da von nun an das Modell nicht mehr zu seinem physischen System passt. Dies hat eine sinkende Vorhersagequalität zur Folge.

Die Verwendung von neuronalen Netzen ermöglicht es, eine komplette Neumodellierung zu vermeiden. Sie können mit weiteren Daten trainiert werden, womit sich das Modell in Form des neuronalen Netzes die neuen Informationen adaptiert.

Nachtrainieren. Bei zeitlichen Änderungen, wie Verschleiß, können die dadurch entstehenden Änderungen nachtrainiert werden. Dazu wird das Netz nochmals mit Daten trainiert die dem aktuellen verschlissenen Zustand der Maschine entsprechen.

Abtrennung von Modifikationen. Ein Basis-Modell wird für die Grundkonfiguration einer fabrikneuen Maschine erstellt (Basis-Identität). Es enthält sämtliche Parameter als Features in den Input-Layern. Jedoch wird deren Gewichtung manuell verändert. Das Modell gewichtet die festen Parameter und die stabilen Anteile der weichen Parameter deutlich höher. Der Entwicklung dieses Modells stehen aufgrund der Grundkonfiguration umfangreiche Daten zur Verfügung. Es ist eine sehr gute Vorhersagequalität zu erwarten.

Nach Modifikation bzw. Konfiguration wird das vorhandene Modell weiterentwickelt (Aktive-Identität). Nun liegt das Training auf den veränderlichen Features. Dies sind die von der Konfiguration abhängigen Parameter sowie die nun feststehenden Definition von gültigen Task-Daten. Dafür ist eine geringe Anzahl an Datensätzen und weniger Trainingsaufwand notwendig. Abbildung 11 veranschaulicht diese Lösung.

Unterteilung nach Änderungsverhalten. Die Vorhersage wird aufgetrennt, in einen statischen Teil der von gleichbleibenden Daten-Klassen abhängt und einen variablen. Beide Teile werden durch ein neuronales Netz repräsentiert. Änderungen betreffen somit nur das variable Modell. Ein finales neuronales Netz erstellt die Prognose aufgrund der Ergebnisse der Ausgaben der beiden Teil-Modelle.

Feature-Modularisierung. Die Modellierung kann auch gänzlich nach anderen Kriterien unterteilt werden. Die Features können nach beliebigen Kriterien klassifiziert werden. Zur Modellierung sind sie in Gruppen aufzuteilen und je Gruppe ist ein Teil-Modell bzw. neuronales Netz anzusetzen. Auch hier ist ein finales Netz nötig, das als Wurzel-Knoten die finale Vorhersage trifft. Als Kriterien zur Unterteilung können datenorientierte Kriterien dienen, wie Korrelation und Varianz zum Vorhersage-Attribut, die sich durch eine Informationsanalyse ergeben.

Modell/Feature-Baum. Zu jedem Feature wird ein neuronales Netz erstellt und mit den Daten trainiert. Diese Teil-Modelle sind als Baum angeordnet und werden äquivalent zu den Features angeordnet. Features die von anderen abhängen,

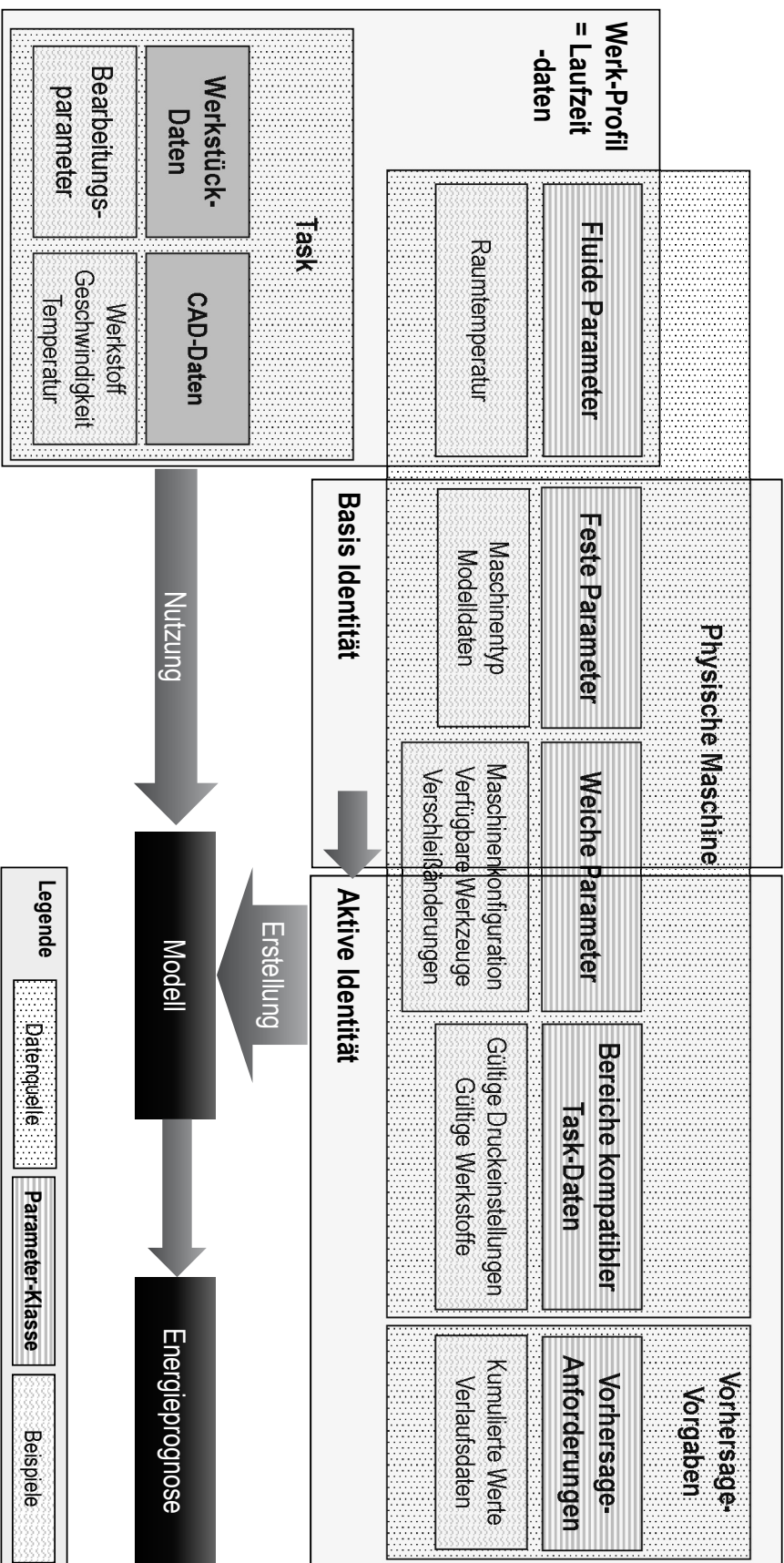


Abbildung 11.: Abtrennen von Modifikationen bei häufigem Nachjustieren

wie die Gesamtlaufzeit von der Initialisierungsphase bzw. Aufwärmphase, erhalten ihren Input aus deren Ausgabe. So würde das Netz der Gesamtlaufzeit, Input aus denen der Aufwärmzeit (3D-Druck) bzw. Initialisierungszeit(Fräsen), der Bearbeitungszeit und der Dauer der Abschlussphase erhalten. Ein neuronales Netz im Wurzel-Knoten erstellt die Vorhersage aufgrund der Ausgaben aller Teil-Modelle. Änderungen betreffen somit nur einzelne Teil-Modelle.

Die Lösungen zur Adaption von Änderungen sind mit Mehraufwand verbunden. Der Nutzen einer erhöhten Komplexität steht in Relation zur Änderungshäufigkeit. Zudem ist die Häufigkeit von Umrüstungen der Fertigungsmaschinen domänenabhängig.

5. Entwicklung der Architektur

In diesem Kapitel wird die Architektur des Systems vorgestellt. Zu Beginn werden für die drei Arten von Planungssystemen (StaPla, DyaRes, SimDynaRes) Use Cases des Vorhersagesystems beschrieben. Es folgen Architekturentscheidungen, die aus den Rahmenbedingungen abgeleitet sind. Die Architektur und deren Komponenten werden vorgestellt. Der Ablauf eines Use Case wird beschrieben. Abschließend wird eine Evaluation durchgeführt.

5.1. Abgeleitete Use Cases

Die Use Cases bzw. Arten des Planungssystems, wie sie in Kapitel 3 erläutert sind, können in Use Cases des Vorhersage-Systems überführt werden. Im Folgenden wird eine Auswahl dieser Use Cases vorgestellt.

1. Art: Statische Planung (StaPla)

1.1. Single-Prediction/Einfach Vorhersage

Input: Eine Anfrage zu einem Task und einer Maschine.

Output: Der spezifische Energieverbrauch für die Fertigung des Werkstücks.

1.2. Multi-Prediction/Mehrfache Vorhersage

Input: Task-Daten und Vorgaben für die Maschinenparameter sowie der zu erwarteten Ergebnisse.

Output: Die n besten (effektivsten) Maschinen, jeweils mit Energie-Vorhersage.

1.3. Search-Config/Konfig-Suche

Input: Task-Daten und Anzahl n an besten Konfigurationen

Output: Die besten denkbaren Konfigurationen mit dem geringstem Energieaufwand

2. Art: Dynamisches Ressourcenmanagement (DynaRes)

2.1. Begleitung des produktiven Betriebs mit dynamischer Planung und Reaktion auf Planabweichungen.

Das System verhält sich wie ein Ressourcenmanager. Es kommen im Zeitverlauf Tasks hinzu und werden in die aktuelle Planung übernommen. Tasks die mehr Ressourcen, explizit Energie, verbrauchen werden gegebenenfalls zur Fehlersuche unterbrochen.

3. Art: Simulation der aktuellen Planung (SimDynaRes)

3.1. Debugging/Fehlersuche.

Bevor die Planung in die Produktion übernommen wird, kann sie schrittweise analysiert werden.

3.2. Energieauslastung zu bestimmten Zeiten.

Nach beginn der Simulation kann sie jederzeit angehalten werden und die aktuellen Energieverbrauchswerte ausgeben. Dies sind die Energieaufnahmen der Maschinen, die zu diesem Zeitpunkt Tasks bearbeiten, Tasks initialisieren oder sich im Leerlauf befinden.

3.3. Analyse peripherer Systeme.

Der Energieverbrauch lässt Rückschlüsse auf weitere Systeme zu. So kann die Abwärme mit ihm korrelieren. Durch Simulation des Energieverbrauchs kann auch auf die Belastung der Kühlsysteme Rückschlüsse gezogen werden.

3.4. Analyse Lastspitzen.

Sind mehrere Maschinen im Einsatz kann es zu Verbrauchsspitzen kommen. Dies geschieht wenn diese zur gleichen Zeit einen überdurchschnittlichen Energieverbrauch aufweisen. Lastspitzen können vom Energieversorger mit höheren Kosten der Stromversorgung belegt sein. Eine Verbrauchssimulation kann dazu beitragen, diese vorherzusagen. Die Planung der Tasks kann entsprechend angepasst werden [Sch14] [Scha].

5.2. Architekturentscheidungen

Aus den Anforderungen und Rahmenbedingungen wurden zwei grundlegende Entscheidungen abgeleitet. Das System basiert auf einer Microservice-Architektur und setzt das Konzept von digitalen Zwillingen um.

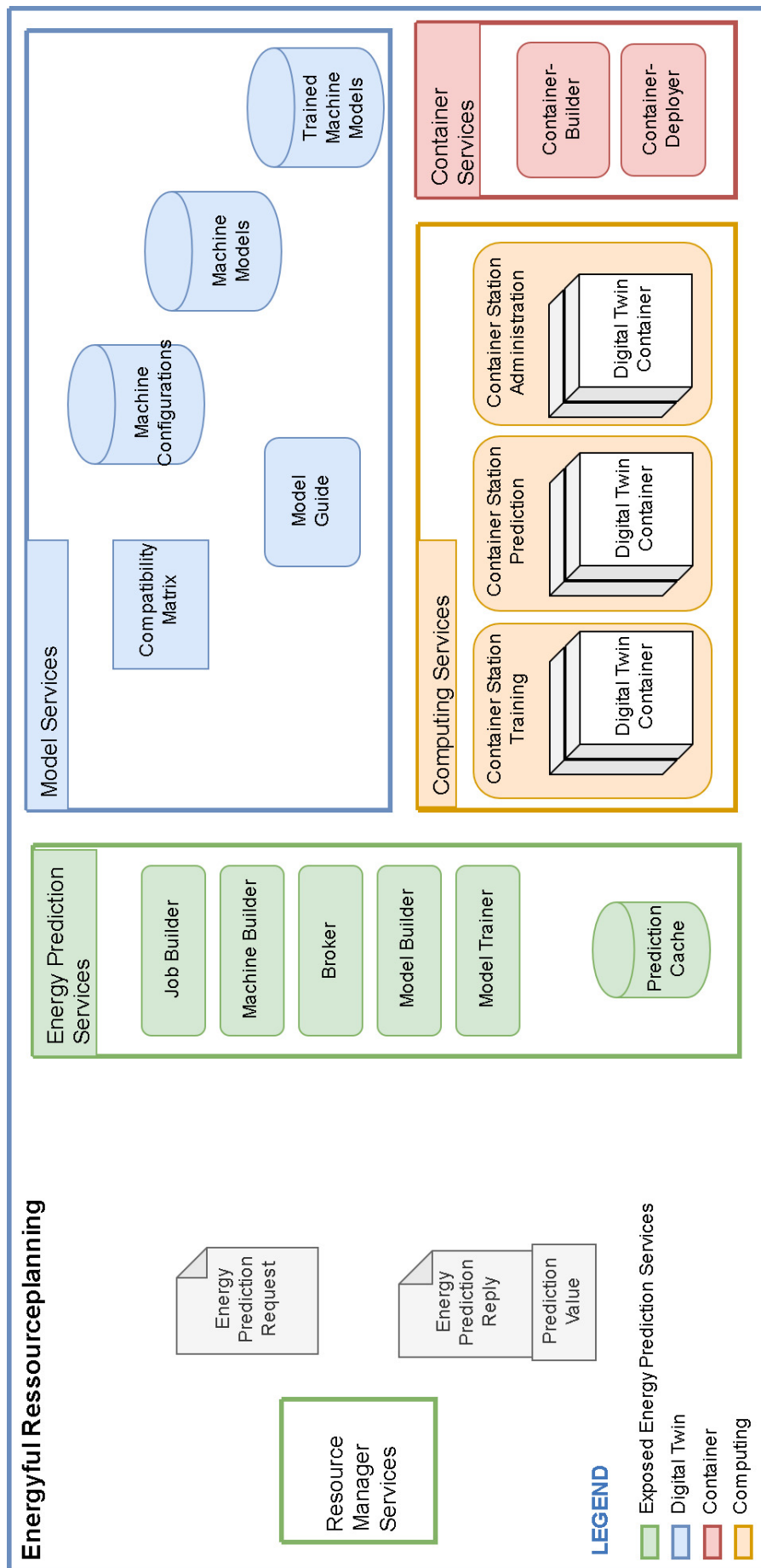


Abbildung 12.: Übersicht über die Teilsysteme und deren Microservices

5.2.1. Microservices

Die Microservice-Architektur stellt eine Vielzahl an Architektur-Pattern dar, die auf einer Gliederung in kleine Dienste (Mikroservice) basieren. Ein Mikroservice ist eine unabhängig einsetzbare Komponente, die einen kleinen Funktionsumfang beinhaltet. Die Interoperabilität wird durch nachrichtenbasierte Kommunikation hergestellt. Ziel ist eine lose Kopplung der Dienste untereinander, jeder Dienst verfügt über eigene Strukturen, Konzepte und Datenhaltungen [Nad16].

Die Steuerung der Verarbeitungsschritte durch die Microservices folgt dem Prinzip der Choreographie. Es findet keine zentrale Steuerung statt, die Verarbeitungsschritte sind durch die Reaktionen der einzelnen Microservices implementiert. Die Umsetzung des Use Case 5.5 demonstriert dies.

Die Menge an Anfragen an das Vorhersagesystem kann stark variieren, wie in Kapitel 5.3 erläutert wird. Daher ist eine Ausrichtung der Architektur hin zu leichter Skalierbarkeit angezeigt. Microservices bieten gute Skalierung, da sie leicht replizierbar und leichtgewichtig sind. Aufgrund ihrer Kapselung und Autonomie sind sie nahtlos in Container-Orientierte Infrastrukturen implementierbar. So lassen sie sich als funktionsfähiges Abbild (Container-Images) ablegen und entsprechend leicht deployen. Um die Microservice-Instanz zu starten wird sie in den Speicher geladen und gestartet. Aufgrund der Kompaktheit der Microservices können sie in größerer Zahl gestartet bleiben, im Vergleich zu kompletten Server-Instanzen. Die Bereitstellungszeit des Dienstes sinkt durch all diese Eigenschaften erheblich.

Ein weiterer Vorteil von Microservices ist die angestrebte Autonomie. Organisatorisch werden die einzelnen Dienste von unterschiedlichen Teams verantwortet, was den Einsatz von DevOps-Teams ermöglicht. Diese sind dabei frei in der Wahl der Architektur und Implementierung des Dienstes. [New15]. So können die Modelle innerhalb der Dienste von spezialisierten Teams entwickelt werden, unter freier Wahl der Vorgehensweise und Implementierung.

Etwaige Weiterentwicklungen des Modells lassen sich aufgrund der Kapselung leicht integrieren. Microservices propagieren robuste Schnittstellen, wodurch eine Abwärtskompatibilität gegeben ist [New15].

5.2.2. Digitale Zwillinge

Digitale Zwillinge wurden bereits in Kapitel 1 vorgestellt.

Die Modelle der Produktionsmaschinen stellen an sich bereits abstrahierte Maschinen dar. Daher bietet sich die Verwendung dieser als digitale Zwillinge an. Hierdurch können objektorientierte Methodiken angewendet werden. Die Kapselung der Modelle bringt Vorteile wie Information-Hiding und eine gut durchsetzbare enge Kopplung innerhalb und lose Bindung untereinander.

Zudem können sie administrative Informationen über ihr Modell bereitstellen. So kann der digitale Zwilling bezüglich seiner Eignung für Vorhersagen oder seiner Anforderungen an die IT-Infrastruktur angefragt werden.

5.3. Ein Wald aus digitalen Zwillingen

Ein Problem bei der Suche nach Maschinen-Konfigurationen ist deren große Anzahl. Prinzipiell kann jeder Parameter mit anderen kombiniert werden. Unter Berücksichtigung der Parameter-Klassen, wie in Abbildung 13 ergibt sich Formel 5.1 zur Berechnung der Menge an Modellen und damit auch an digitalen Zwillingen. Damit entspricht d den digitalen Zwillingen, f den festen Parametern, w den weichen Parametern, b den Bereichen kompatibler Task-Daten und v den Vorhersage-Anforderungen. In 5.2 wird die Berechnung exemplarisch durchgeführt.

$$n_d = n_f * n_w * n_b * n_v \quad (5.1)$$

$$n_d = 10 * 10 * 4 * 2 = 800 \quad (5.2)$$

Es entsteht ein Wald aus digitalen Zwillingen. Jeder dieser digitalen Zwillinge erstellt Energieprognosen, was zu einem entsprechend hohen Rechenaufwand führt. Dazu können neben anderen auch Maßnahmen zur Reduzierung der Antwort-Latenz zählen, wie sie in The Tail at Scale [DB13] erläutert werden. So kann der Broker-Dienst digitale Zwillinge mit schlechter Vorhersage-Historie ignorieren. Eine weitere Methode ist das Setzen einer Deadline, innerhalb dieser die digitalen Zwillinge ihre Prognosen erstellt haben müssen.

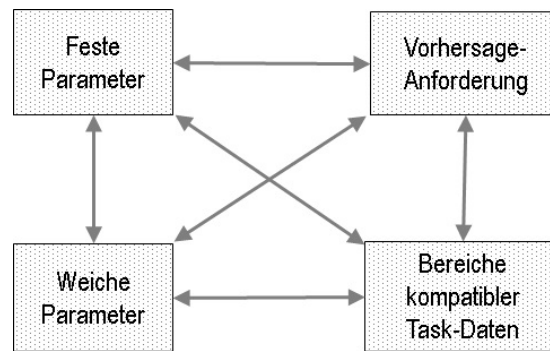


Abbildung 13.: Kombinationen von Parametern

Des Weiteren lässt sich die Bildung großer Mengen digitaler Zwillinge vermeiden, indem Einschränkungen (Constraints) der Kombinationsmöglichkeiten erstellt werden.

5.4. Aufbau und Übersicht des Systems

Das System gliedert sich in vier Teilbereiche, wie Abbildung 12 dargestellt. Aufgrund der Microsystem-Architektur bilden diese lose Subsysteme. Sie sind nach Funktion gruppiert, interagieren jedoch mit Diensten aus allen Subsystemen. Dabei nutzen sie nachrichten-orientierte Kommunikation, die als RESTful API implementiert sind.

5.4.1. Peripherie und Interaktion

Das System zur Produktionsplanung und die Interaktion mit diesem bilden die Peripherie wie folgt:

Resource Manager Services: Dienste des Resource Managers. Aus Sicht des Vorhersage-Systems sind dies externe Dienste.

Energy Prediction Request: Anfrage an das Vorhersage-System. Eine JSON-Datei wie sie in Listing F.1 im Anhang beschrieben ist.

Energy Prediction Reply: Antwort des Vorhersage-Systems. Beinhaltet je nach Anforderung ein oder mehrere Prognosen.

5.4.2. Teilsystem Energie-Vorhersage

Die Energy Prediction Services bilden die Schnittstelle nach außen, nur sie können mit externen Systemen wie dem Planungssystem kommunizieren.

Job Builder: Erstellt aus den Anfragen (Energy Prediction Request) von außen den Prediction Job. Dazu kann er auch auf Daten in entfernten Systemen zugreifen, wie etwa Stücklisten-Datenbanken und Daten des Maschinen- oder Werk-Profiles zu vervollständigen.

Machine Builder: Erstellt Maschinen-Profile bzw. Identitäten. Diesen können Modelle zugeordnet werden.

Broker: Entscheidet welche Prognosen zurückgesendet werden. Wählt die digitalen Zwillinge aus, deren Prognosen den Vorgaben entsprechen.

Model Builder: Erstellt eine Architektur eines neuronalen Netzes anhand eines Maschinen-Profiles und ordnet sie ihm zu. Es können mehrere Modelle einem Maschinen-Profil zugeordnet werden.

Model Trainer: Nutzt Trainings-Daten dazu Modelle von Maschinen-Profilen zu erstellen.

Prediction Cache: Ist an den Broker und Job Builder angegliedert. Darin befinden sich eine Anzahl an Energievorhersagen.

5.4.3. Teilsystem Modellierung

Die Model Services verwalten die Modelle.

Compatibility Matrix: Beinhaltet welche Modelle mit welchen Anforderungen und Vorhersageverfahren kompatibel sind.

Model Guide: Verwaltet die Compatibility Matrix.

Machine Configurations: Datenbank aller bekannten Maschinen-Profile bzw. Identitäten.

Machine Models: Datenbank aller entworfenen Modelle. Dies umfasst die Art des neuronalen Netzes, sowie Details über dessen Architektur.

Trained Machine Models: Datenbank bereits erstellter Modelle.

5.4.4. Teilsystem Berechnung

Die Computing Services beinhalten die sich im Betrieb befindlichen digitalen Zwillinge.

Container Station Training: Eine Verwaltungseinheit, um die Instanzen der digitalen Zwillinge zu betreiben, deren Modell erstellt wird. Hohe Ressourcen-Last.

Container Station Prediction: Eine Verwaltungseinheit, um die Instanzen der digitalen Zwillinge zu betreiben, deren Modelle Vorhersagen erstellen. Mittlere Ressourcen-Last.

Container Station Administration: Eine Verwaltungseinheit, um die Instanzen der digitalen Zwillinge zu betreiben, die momentan administriert werden. Dabei kann es sich um Vorgänge wie Re-Konfiguration oder Standby-Haltung handeln. Geringe Ressourcen-Last.

5.4.5. Teilsystem Container Verwaltung

Die Container Services erstellen und deployen die Container, in denen sich die digitalen Zwillinge befinden.

Container-Builder: Erstellt aus den Modellen digitale Zwillinge. Diese liegen daraufhin als Container-Images vor.

Container-Deployer: Verwaltet das Ausliefern, Starten oder Beenden der Container-Images. Nutzt dazu die Container Stations.

5.5. Use Case Multiprediction

Abbildung 14 stellt den Multiprediction-Use Case dar. Die einzelnen Schritte sind wie folgt:

1. Der Ressource Manager schickt eine Anforderung (Energy Prediction Request) an den Job Builder. Die Anforderung beinhaltet 20 Maschinen-Profile, 1 Werk-Profil, die Vorgabe kumulierte Energieprognosen zu verwenden und die 3 besten Maschinen-Profile zu ermitteln.
2. Der Job Builder erstellt daraus einen Prediction Job und schickt ihn an die Prediction Station.
3. Die Prediction Station startet alle kompatiblen DZ mit den Daten aus dem Prediction Job.
4. Da nicht alle DZ bereits deployt sind, fordert sie zusätzliche DZ vom Container

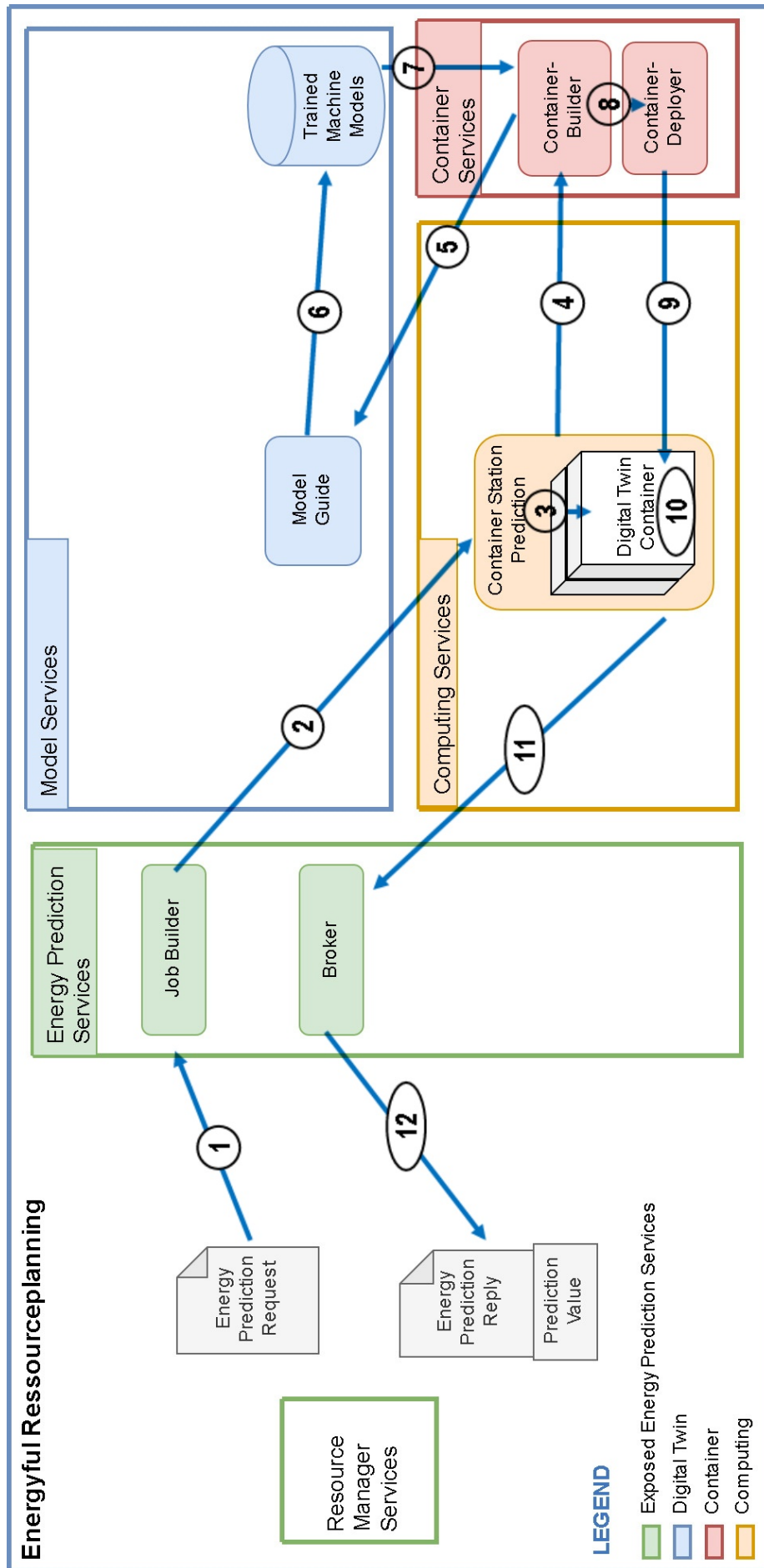


Abbildung 14.: Energy Prediction Use Case Multiprediction

Builder an.

5. Dieser gibt den einen Such-Auftrag an den Model Guide. Dieser sucht nach kompatiblen Modellen.
6. Der Model Guide findet bereits trainierte Modelle.
7. Diese werden an den Conatiner Builder übermittelt.
8. Der Conatiner Builder erstellt daraus einen neuen Container.
9. Den der Conatiner Deployer in die Conationer Station hochlädt und zum Start frei gibt.
10. Die DZ haben ihre Prognosen berechnet.
11. Die Ergebnisse der DZ werden an den Broker geschickt
12. Dieser wählt aus den Antworten der DZ die 3 effektivsten aus und schickt die Vorhersage-Werte mit der ID des Maschinen-Profils als Antwort (Energy Prediction Reply) auf die Anforderung an den Ressource Manager.

5.6. Evaluation

Eine Architektur auf Basis von Microservices wurde entworfen. Bezüglich des maschinellen Lernens kann die Zustandslosigkeit der Microservices eine Herausforderung darstellen. Die Modelle bei neuronalen Netzen sind durch Training entstanden und sind aus Sicht des untrainierten Netzes zustandsvoll.

Durch das Nachladen von Modellparametern, wie von Pahl und Loipfinger in [PL18] vorgeschlagen, wird die Zustandslosigkeit wiederhergestellt. Jedoch ist dieses Verfahren mit einer Latenz verbunden. In den betrachteten Use-Cases ist die Zustandshaftigkeit nicht relevant. Die Performance steht im Vordergrund. Zudem ist der Umfang des trainierten Modells gering. Daher wird dieses Design-Pattern bezüglich der Modelle nicht umgesetzt.

6. Anwendung des Vorgehensmodells

Das in Kapitel 4.2 entwickelte Verfahren wird verwendet um in zwei konkreten Domänen eine Abstraktion mittels einer Schablone zu erstellen. Dabei werden ausgewählte Schritte zur Adaption der Domäne durchgeführt und Schablonen der Domänen erstellt.

6.1. Domäne Fräsmaschinen

Im Folgenden wird das Vorgehensmodell exemplarisch auf die Domäne 'Fräsmaschinen' angewandt. Der Fokus liegt dabei auf der universellen Anwendbarkeit des Verfahrens.

6.1.1. Simulation von Fräsmaschinen

Da für diese Arbeit keine Fräsmaschine zur Verfügung stand, wurde stattdessen auf eine Simulation zurückgegriffen. Das Programm 'CNC Simulator Pro' ist in der Lage eine Reihe von CNC-Maschinen simulieren, neben 3D-Druckern, Schleifmaschinen und Robotern auch Fräsmaschinen.

Die Simulation ist umfangreich konfigurierbar. So wird das eingesetzte Werkzeug wie in Abbildung 15 detailliert beschrieben. Die dabei genutzten Parameter bilden alle relevanten Eigenschaften ab und sind daher als Datenbasis gut geeignet.

Beim Bearbeitungsprozess, siehe Abbildung 16, wird die Ausführung des CNC-Programms, in Form einer g-code-Datei, in Echtzeit simuliert.

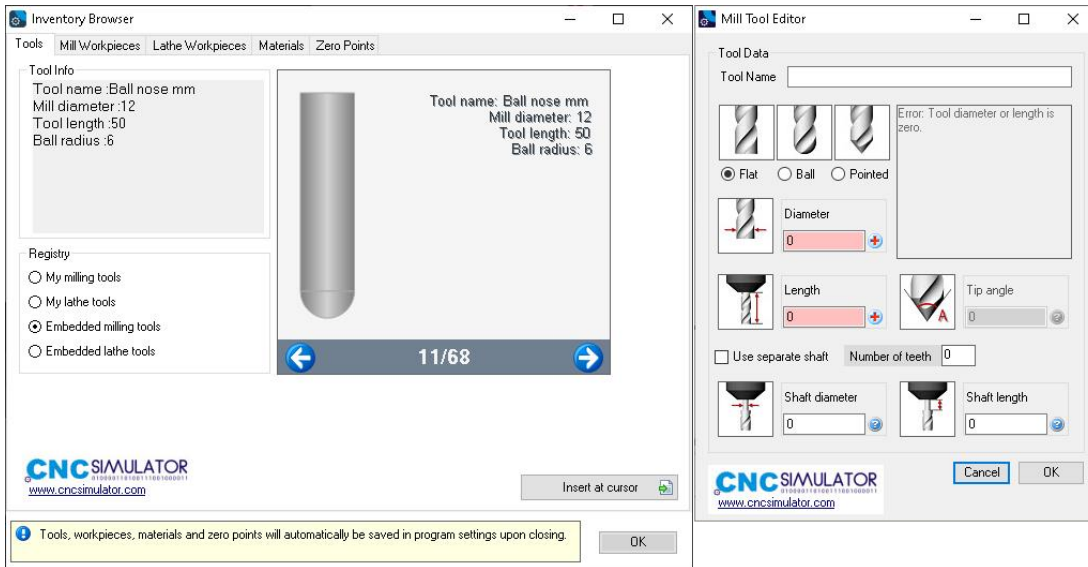


Abbildung 15.: Simulation des Werkzeugs

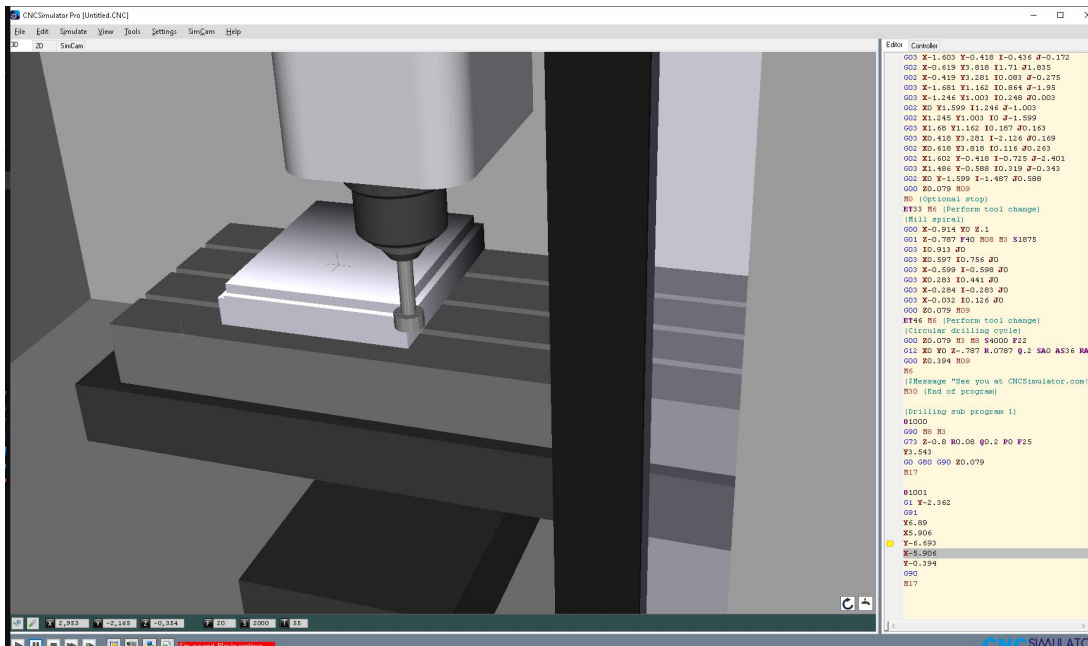


Abbildung 16.: Simulation der Bearbeitung

6.1.2. Anwendung des Vorgehensmodells auf Fräsmaschinen

Die Schritte des Vorgehensmodells werden angewendet.

1. Erstellung der Schablone

1.1. Festlegung der exakten Domäne.

Die Schablone soll im Bereich industrieller Fräsmaschinen verwendet werden.

1.2. Analyse des Produktionsprozesses.

Die Maschine verwendet ein Werkzeug, um ein Werkstück zu bearbeiten. Das Werkzeug ist ein Fräsbohrer, der dazu verwendet wird, um Material vom Werkstück abzutragen.

Beim Werkzeug ist die Form des Fräsbohrers und die Anzahl der Windung relevant. Daraus ergeben sich die maximale Geschwindigkeit mit der das Material bearbeitet werden kann und Vorgaben über spezifische Eigenschaften des Materials, wie des Härtegrades.

1.3. Erfassen der Parameter und Einteilen in Klassen:

Aus der Reihe der simulierten Fräsmaschinen wurde das Milling-Center ausgewählt, da es das Standardgerät von 'CNC Simulator Pro' für Fräsmaschinen ist und daher ein repräsentatives Verhalten angenommen werden kann. Um die Zuordnung zu den Klassen zu erleichtern und das Vorgehensmodell abzukürzen wurden die Parameter erfasst und diese mit Daten aus der Simulation gefüllt.

Parametername	Wert	Klasse
Name	Milling-Center	HP
Hersteller	CNC Simulator Pro	HP
Max Feedrate XYZ	500	HP
Rapid Feedrate XYZ	30000	HP
Max Spindle RPM	5000	HP
Mittlere Zeit zum Werkzeugwechsel (s)	5	HP
Maschinenkosten pro h	400	HP
Compatible G-Codes	marvin	HP
Compatible M-Codes	marvin	HP
Maximale physische Abmessung Werkstück X (mm)	100	HP
Maximale physische Abmessung Werkstück Y (mm)	100	HP
Maximale physische Abmessung Werkstück Z (mm)	100	HP
Maximales physisches Gewicht Werkstück (kg)	7	HP
Unterstützte Werkzeugarten	Fräser	HP
Werkzeuge		
Name	Drill1	WP
Fräs-Durchmesser	10	WP
Werkzeug Länge	100	WP
Ball-Radius	5	WP
Werkstück		
X	15	WP
Y	15	WP
Z	15	WP
Material	Aluminium	WP

Tabelle 12.: Schablone Fräsmaschine: Maschine

Parametername	Wert	Klasse
Bezeichnung	AHT23	WP
Typ	Metal	WP
Ident	3243243438	WP
Name	Aluminium	WP
Preis/kilo	8,50	WP
Gewicht/cm ³ (g/cm ³)	2,70	WP
Oberflächen Geschwindigkeit SMM	80	WP
Werkzeug Feed (mm/Zahn)	0,18	WP

Tabelle 13.: Schablone Fräsmaschine: Materialien

2. Ableiten von Maschinen-Profil und Werkstück-Profil.

Aus den Klassen werden nun das Maschinen- und Werk-Profil gebildet. Diese werden zur Erstellung und Nutzung des digitalen Zwillings verwendet. Eine konkrete Umsetzung wird für diese Domäne übersprungen.

Parametername	Wert	Klasse
Bezeichnung	testdruck3	WD
Ident	0005646	WD
Datentyp	ID	WD
PreProc	0	-
PreProcExec	-	-
2D/3D-Zeichnung	-	WD
Mesh-Datei	-	WD
3D Werkzeugpfad	testdruck3.gcode	WD

Tabelle 14.: Schablone Fräsmaschine: Task

Parametername	Wert	Klasse
Umgebungsparameter	_*	FP
Simulation	CPU-Last Hostsystem	FP

Tabelle 15.: Schablone Fräsmaschine: Umgebungsparameter
*Wird nicht simuliert

Parametername	Datentyp	Klasse
Energieverbrauch	_*	PP
Simulation: Zeit	195m	PP
Simulation: Zeit auf Prozessor (Last)	137m	PP

Tabelle 16.: Schablone Fräsmaschine: Prognoseparameter
*Wird nicht simuliert

6.2. Domäne 3D-Druck

Im Folgenden wird das Vorgehensmodell exemplarisch auf die Domäne 3D-Druck angewandt.

6.2.1. Zugrundeliegende Umgebung

Zur Entwicklung der Schablone werden verschiedene Quellen genutzt. Dienstanwendungen zum Drucken bieten eine vollständige Quelle an Informationen. Sie bilden sowohl die Maschine, in diesem Fall den 3D-Drucker, als auch den Task parametrisiert ab.

Zur Analyse der Maschinenparameter werden die folgenden Anwendungen aus dem CAD (Tabelle 17) und spezielle Drucker-Anwendungen des 3D-Drucks (Tabelle 18) herangezogen. Als Quelle dienen Parameter der Anwendungen sowie Konfigurations-

Anwendung	ReMake	Meshmixer	Fusion 360
Version	17.23.0.38	3.5.474	2.05827
Hersteller	Autodesk, Inc.	Autodesk, Inc.	Autodesk, Inc.
Lizenz	kostenpflichtig	Frei	Education License

Tabelle 17.: CAD-Software: 3D-Druck+Consumer Bereich

Anwendung	cura	voxelizer
Version	4.10	2.10
Hersteller	Ultimaker B.V.	ZMorph S.A , Inc.
Lizenz	Frei	kostenpflichtig

Tabelle 18.: 3D-Druck-Software

dateien. Im Anhang ist ein gekürztes Drucker-Profil der Anwendung Cura zu finden (Listing C.1).

6.2.2. Anwendung des Vorgehensmodells auf 3D-Druck

Die Schritte des Vorgehensmodells werden nun durchgeführt.

1. Erstellung der Schablone

1.1. Festlegung der exakten Domäne.

Die Schablone soll für Maschinen im Bereich des additiven 3D-Drucks verwendet werden. Hierbei liegt der Fokus auf dem Consumer-Bereich.

1.2. Analyse des Produktionsprozesses.

Ein Maschinen-Werkzeug wird verwendet, um das Werkstück in Schichten aufzutragen. Das Werkzeug beim 3D-Druck ist ein Druckkopf, durch den das Filament geführt wird. Im Druckkopf wird es durch Erwärmung verflüssigt und durch den Vorschub durch die Druckdüse gepresst. Es entsteht ein feiner Faden aus Filament.

Zu Beginn des Druckvorgangs bewegen sich Druckbett und Druckkopf in die Startposition. Das Druckbett justiert sich vertikal, die Druckdüse horizontal. Das Druckbett wärmt sich auf, der Druckkopf wird erhitzt und der Vorschub beginnt. Der erhitzte Filamentfaden extrudiert aus dem Druckkopf und bleibt auf dem Druckbett kleben. Im Druckkopf kühlt der Ventilator das Filament auf dem Druckbett ab, um es zu verfestigen. Der

Druckkopf trägt, sich zweidimensional bewegend, die erste Schicht Filament auf. Die zweite Schicht wird begonnen. Das Druckbett wird um Druckschichthöhe nach unten justiert und der Druckkopf wird an den Startpunkt bewegt. Eine weitere Schicht wird aufgetragen. Dieser Vorgang wiederholt sich, bis sämtliche Schichten aufgetragen sind.

Spezielle Modi ermöglichen eine erweiterte Kontrolle des Drucks über die notwendigen Prozesse hinaus. So kann eine Nachbearbeitung der Außenhaut des Druckobjekts, nachdem die betreffende Schicht aufgetragen ist, zum Druckprozess hinzugefügt werden. Des Weiteren kann die Zeit beeinflusst werden, die zum Positionieren des Druckkopfs aufgewendet wird. Wasserdichtes Drucken kann aktiviert werden, wodurch der Druckkopf einen ununterbrochenen vollständigen Umlauf je Schicht druckt.

1.3. Erfassen der Parameter und Einteilen in Klasse.

Die Maschinenprofile von Cura stellen die umfangreichsten Informationen über eine Maschine und deren Werkzeuge dar. Sie unterteilen sich in maschinenspezifische und druckkopfspezifische Daten. Bei Voxelizer lassen sich nur rudimentäre Einstellungen vornehmen, weshalb diese als essenziell zu betrachten sind. Die restlichen Anwendungen verbergen die Druckparameter oder sind auf herstellerspezifische Dienstprogramme angewiesen, um die g-Code-Dateien für den 3D-Drucker zu erstellen. Zur Analyse weiterer Parameter der Werkzeuge und der Daten eines Druckauftrags wurden die Druckprofile von Cura herangezogen.

Weitere Parameter des Druckauftrags sind in den g-Code-Dateien zu finden. Diese bestehen aus einem Header, der grundlegende Informationen des Druckauftrags beinhaltet, sowie dem eigentlichen g-Code.

In der g-Code-Datei sind alle Druck-Parameter indirekt enthalten, da er auf Basis der Daten erstellt wird. Der Informationsgehalt ist jedoch geringer, da diese relevanten Daten darin nur einen geringen Anteil haben. Die g-Code-Dateien im Proof of Concept sind bis zu 100 Megabyte groß.

Um die Parameter des Materials zu erhalten wird das Material-Profil von Cura als Orientierung verwendet.

Parametername	Datentyp	Klasse
Bezeichnung	String	HP
Typ	String	HP
Hersteller	String	HP
Ident	String	HP
Firmware	String	HP
Maximale Druckabmessung X	Integer	HP
Maximale Druckabmessung Y	Integer	HP
Maximale Druckabmessung Z	Integer	HP
Druckbett Form Rechteck	Boolean	HP
Druckbett Beheizbar	Boolean	HP
g-Code Sprache	String	HP
Anzahl Druckköpfe	Integer	HP
Maximale Bewegung des Druckkopfs X	Integer	HP
Maximale Bewegung des Druckkopfs Y	Integer	HP
Maximale Bewegung des Druckkopfs Z	Integer	HP

Tabelle 19.: Schablone Additiver 3D-Druck Consumer Bereich: Maschine

Parametername	Datentyp	Klasse
Bezeichnung	String	WP
Typ	String	WP
Ident	String	WP
Druckkopf-Größe	Float	WP
Kompatible Material-Durchmesser	Float	HP
Anzahl Lüfter	Integer	HP
Anzahl Druckdüsen	Integer	HP
Auto Filament Wechsel	Boolean	HP
Dual-Druck	Boolean	HP

Tabelle 20.: Schablone Additiver 3D-Druck Consumer Bereich: Werkzeug

Parametername	Datentyp	Klasse
Bezeichnung	String	BP
Typ	String	BP
Ident	String	BP
Hersteller	String	BP
Dichte	Float	BP
Durchmesser	Float	BP
Druckparameter Material		
Druck Temperatur	Integer	BP
Druckbett Temperatur	Integer	BP
Druckgeschwindigkeit	Integer	BP
Standby Temperatur	Integer	BP
Lüfter Geschwindigkeit	Integer	BP

Tabelle 21.: Schablone Additiver 3D-Druck Consumer Bereich: Material

Parametername	Datentyp	Klasse
Verortung Standort	String	WP
Verortung Maschine	String	WP
Ident	String	WP
Umgebungstemperatur: Celsius	Float	FP
Gehäusetemperatur: Celsius	Float	FP
Belüftung: Luftbewegung	Float	FP
Erschütterung: Schwinggeschwindigkeit X	Float	FP
Erschütterung: Schwinggeschwindigkeit Y	Float	FP

Tabelle 22.: Schablone Additiver 3D-Druck Consumer Bereich: Umgebung

Parametername	Datentyp	Klasse
Bezeichnung	String	BP
Typ	String	BP
Ident	String	BP
Maschinen-Parameter		
Layer-Höhe	Float	BP
Support	Boolean	BP
Support Extruder	Integer	BP
Support Typ	String	BP
Druckkopf-Parameter		
Druckkopf-Nummer	Integer	BP
Zusatz-Priming	Boolean	BP
Pfadlänge-Optimieren	Boolean	BP
Wasserdichter-Druck	Boolean	BP
Bottom-layers	Integer	BP
Bottom-thickness	Integer	BP
Top-bottom-thickness	Integer	BP
Top-layers	Integer	BP
Top-thickness	Integer	BP
Wall-line-count	Integer	BP
Wall-thickness	Integer	BP
Lüftergeschwindigkeit	Integer	BP
Füll-Muster	String	BP
Füll-Muster-Dichte	Integer	BP
Material-Fluss	Integer	BP
Material-Druck-Temperatur	Integer	BP
Druckbett Temperatur	Integer	BP
Druckgeschwindigkeit	Integer	BP
Support-Menge	Integer	BP
Standby Temperatur	Integer	BP
Modell-Parameter		
g-Code-Header	String:CSV	WD
g-Code-Body	String:g-Code	WD
g-Code-Footer	String:Comments	WD

Tabelle 23.: Schablone Additiver 3D-Druck Consumer Bereich: Task

2. Ableiten von Maschinen-Profil und Werkstück-Profil

Als die Basis des Modells kann nun das Maschinen-Profil und das Werk-Profil aus den zugehörigen Daten-Klassen, wie in Kapitel 4.1.4 dargestellt, erstellt werden.

6.3. Evaluation und weitere Schritte

Um die Schablone effektiv nutzen zu können, werden die Datenquellen für die Parameter-Felder definiert. Dies dient der Implementierung einer automatisierten Übertragung der Daten.

Die im **Maschinen-Profil** verwendeten Parameter werden im Verlauf wie folgt genutzt:

- Identifikation der Maschine
- Wahl des Vorhersage-Verfahrens
- Entwicklung des Vorhersage-Verfahrens

Das **Werk-Profil** ist im produktiven Betrieb relevant. Die Daten darin ändern sich je Task und sind daher automatisiert zu erfassen. Da für diese Arbeit die Verwendung von neuronalen Netzen angenommen wird, dient es als Quelle für abhängige Attribute. Dies sind Parameter die das neuronale Netz als Input nutzt.

Das Verfahrensmodell konnte ohne tiefgreifendes Domänenwissen durchgeführt werden. Die Daten wurden erfolgreich erfasst und strukturiert.

7. Exemplarische Energievorhersage

In diesem Kapitel wird die Energievorhersage exemplarisch durchgeführt. Es gliedert sich in den Teil zur Entwicklung der Grundlagen und in den zur konkreten Implementierung eines neuronalen Netzes.

Zunächst werden Vorgaben und konkrete Ziele entwickelt. Daraufhin wird die verwendete Maschine kurz beschrieben. Es folgt die Erfassung und Verarbeitung der Daten, die als Basis zu Prognose zur Verfügung gestellt werden. Im zweiten Teil wird ein neuronales Netz entwickelt und implementiert. Die Entwicklungsstufen der Implementierung werden beschreiben. Mit der Diskussion der Ergebnisse schließt das Kapitel.

7.1. Vorgaben und Ziel

Das grundlegende Verfahren der Prognose des Energieverbrauchs ist in den vorangegangenen Kapiteln beschrieben worden. Im Zentrum steht dabei der digitale Zwilling und dessen Prognosefähigkeit. Diese basiert auf historischen empirischen Daten.

Um die Performance von Vorhersage-Systemen zu vergleichen, existieren Datenbanken mit gelabelten empirischen Daten. Dabei ist deren Authentizität nur insofern relevant, als sie die generelle Tauglichkeit des Vorhersage-Systems für die entsprechende Domäne belegen. Häufig sind die Daten unvollständig und werden mittels arithmetischer Methoden nachbearbeitet. Grundsätzlich legen so die Daten die Domäne fest in der das Vorhersage-System arbeitet.

Für die Domäne '3D-Druck' lagen zum Zeitpunkt dieser Arbeit keine gelabelten empirischen Daten vor die den Qualitätsansprüchen genügen. Ein Ziel war daher empirische Daten zu sammeln. Qualitativ soll die Varianz der Druckmodelle hoch sein, um die Datenbasis per Imputation erhöhen zu können. Es sollen nur Daten von erfolgreichen, vollständigen Druckprozessen verwendet werden. Der Druckprozess soll möglichst vollständig auf dessen Energieverbrauch erfasst werden. Es soll ein Zusammenhang zwischen den Auftragsdaten - im Proof of Concept sind dies das zu

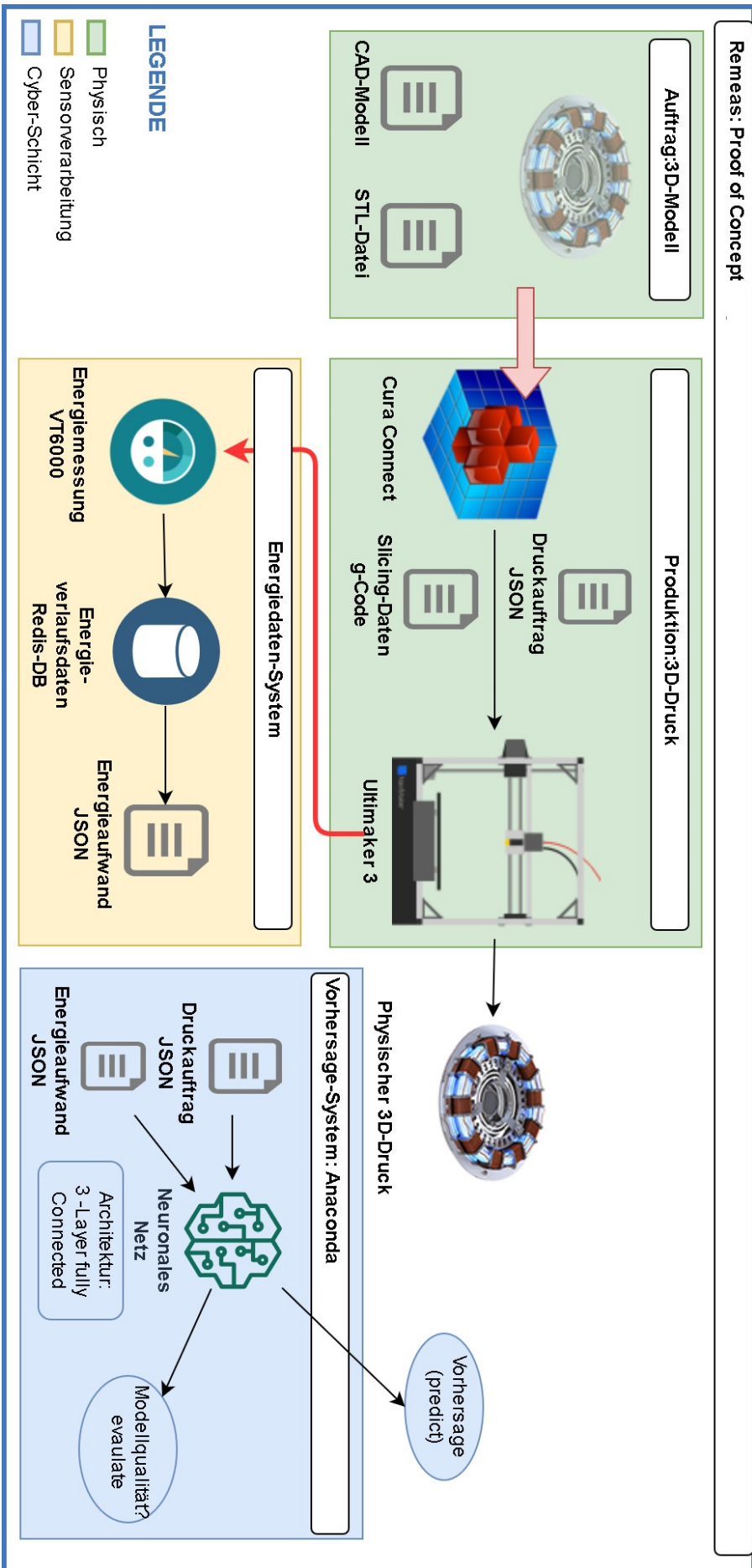


Abbildung 17.: Proof of Concept

druckende 3D-Modell und der Druckauftrag - und dem erfasstem Energieverbrauch hergestellt werden.

Auf Basis der gelabelten Daten soll ein Verfahren entwickelt werden das mit Hilfe von neuronalen Netzen eine Prognose über den Energieverbrauch des Druckprozesses erstellt. Dieses Verfahren soll evolutionär in Schritten weiterentwickelt werden.

7.2. Maschinenbeschreibung

Der Proof of Concept wird in der Domäne 3D-Druck durchgeführt. Als physische Maschine wird daher ein 3D-Drucker verwendet. Dieser wird durch einen Ultimaker 3 repräsentiert. Es handelt sich hierbei um einen 3D-Drucker aus dem gehobenen Consumer-Bereich [Rat16].

Im vorliegenden Fall verfügt der 3D-Drucker über zwei Werkzeuge, in Form von zwei Druckdüsen. Nutzt der Druckauftrag beide Druckdüsen zum Druck des Modells, wird abwechselnd der Anteil des aktuellen Layers von der jeweiligen Druckdüse gedruckt. Es handelt sich um eine serielle Nutzung innerhalb des Druckprozesses.

Details zum 3D-Druck sind bereits in Kapitel 3.3 beschrieben worden.

7.3. Vom Objekt zum Modell

Die Erstellung der Datenbasis ist Teil des Proof of Concept. Es werden 102 3D-Drucke durchgeführt. Der Großteil der Modelle wird aus frei verfügbaren Datenbanken (z.B. Thingiverse [Dizb]) entnommen. Bei der Auswahl wurde auf eine große Varianz geachtet. Folgende drei Modelle stellen die Spannweite der Varianz dar:

Deckel. Besteht aus einer Scheibe mit Rand. Beim Drucken wird hohe Effektivität erreicht, da zirkulär gedruckt wird, wodurch keine Druckkopfbewegung zum Positionieren benötigt wird. Das kleinste Modell mit der kürzesten Druckzeit.

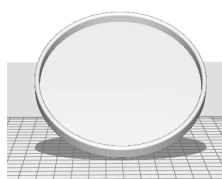


Abbildung 18.: Modell Deckel

Apfel. Ein Apfel, der aus einem 5-schichtigen Wabenmuster besteht. Trotz der geringen Größe des Modells wird durch das Slicing eine große g-Code-Datei erzeugt. Die Druckzeit ist mit 17 Stunden hoch.

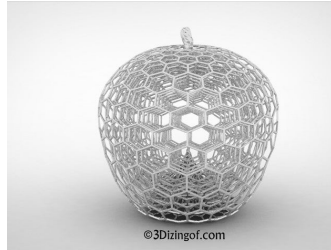


Abbildung 19.: Modell Apfel [Diza]

Vase. Da die Aussagekraft des 3D-Drucks mit dessen Druckdauer steigt, wurde ein entsprechendes Modell selbst entworfen. Die Vase gruppiert sich in zwei Abschnitte. Einer geöffneten Halbkugel, aus der zwei symmetrische Röhren gezogen sind die nach oben Ringe formen. Die Druckzeit beträgt 53 Stunden und 28 Minuten.

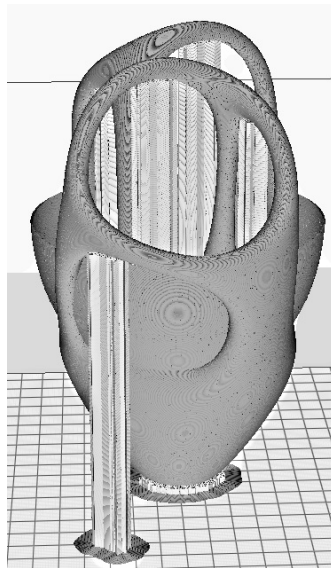


Abbildung 20.: Modell Vase. Neben dem primären Filament ist auch die Stützstruktur zu sehen.

In Tabelle 24 sind die genauen Daten der ausgewählten Druckmodelle zu finden. Im Anhang B werden diese und weitere ausgewählte Modelle vorgestellt.

Die verwendete Infrastruktur konnte diese Auslastung nicht bewältigen. Häufige Instandhaltungen des 3D Druckers und 29 abgebrochene Drucke waren die Folge. Einige Modelle wurden mit verschiedenen Druckeinstellungen (z.B. Geschwindigkeit) gedruckt. Bei 96 Drucken wurde als primäres Filament (Druckmaterial) PLA und als

Modell	STL-Datei (KB)	g-Code (KB)	Druckzeit	Materialverbrauch (cm)
Deckel	63	479	0h47m	35
Apfel	11450	162971	17h15m	62
Vase	35451	124892	53h28m	212

Tabelle 24.: Ausgewählte 3D-Druck-Modelle

Aufnahmezeit	Stromstärke	Wirkleistung	Scheinleistung	Differenz
08.05.2019 12:39	0,063	4,773384	14,4648	67
08.05.2019 12:40	0,424	56,610784	97,6048	42
08.05.2019 12:41	0,831	179,505972	190,9638	6
08.05.2019 12:42	0,881	188,118168	202,2776	7
08.05.2019 12:43	0,503	99,579916	115,7906	14
09.05.2019 19:15	0,091	0	20,8845	100
09.05.2019 19:16	0,091	0	20,8663	100

Tabelle 25.: Exemplarische Messwerte. Stromstärke in Ampere, Wirkleistung und Scheinleistung in Watt, sowie die Differenz von Wirkleistung und Scheinleistung in Prozent.

Stütz-Filament PVA verwendet. Versuchsweise wurde Nylon, PET und ABS verwendet. Details zu den Filamenten sind [Hag15] zu entnehmen.

7.4. Informationserfassung und Verarbeitung

Im Energiedatensystem werden die Sensordaten erfasst und zur weiteren Verwendung aufbereitet. Da der Proof of Concept in evolutionären Schritten entwickelt wird, variiert das System über die Evolution.

Zur Erfassung der Energiedaten wird der Volcraft Energy-Logger 4000 verwendet. Dieser erfasst im Minutentakt die Scheinleistung, Wirkleistung und Stromstärke des 3D-Druckers. Ein Datensatz entspricht daher dem kumulierten Wert der vergangenen 60 Sekunden des jeweiligen Sensors. Tabelle 25 stellt die Datensätze exemplarisch dar.

Im Consumer-Bereich wird für den Verbraucher die Wirkleistung, im industriellen Umfeld für den Hersteller die Scheinleistung in Rechnung gestellt. Die Blindleistung variiert stark, im Standby des Druckers macht sie die gesamte Leistung aus. In diesem Proof of Concept wird die Scheinleistung des 3D-Druckers verwendet, da somit der gesamte Energieverbrauch erfasst wird. Die Messdaten werden durch den Volcraft Energy-Logger 4000 lokal gespeichert und müssen manuell heruntergeladen werden.

Sie liegen in einem proprietären Format vor, das per Dienstprogramm in ein CSV-Format überführt wird. Als CSV-Datei wird die Messreihe in eine Redis-Datenbank übertragen. Redis ist ein Key-Value-Store, der aufgrund der hohen Schnelligkeit und der Eignung für einfache Datenstrukturen ausgewählt wurde.

Die Druckjobdaten werden über eine Swagger-Schnittstelle des Druckers abgerufen und als JSON-Dokument in die Redis-Datenbank übertragen. Die Timestamps der Messreihen sind nicht synchron, weswegen in wöchentlichem Takt Offsets ermittelt wurden um sie zu synchronisieren.

Eine Jedis-Anwendung labelt nun die Druckjobs mit den Energiedaten. Für jeden Druckjob, der erfolgreich beendet worden ist, werden die Energiedaten in der Datenbank gesucht, die zwischen Start und Ende des Druckjobs erfasst wurden. Siehe Listing D.2 im Anhang. Die gelabelten Daten stehen nun dem Vorhersagesystem zur Verfügung und können dort zum Erstellen eines Modells verwendet werden.

7.5. Der Weg zur Prognose

Das Vorhersagesystem nutzt zur Erstellung der Energievorhersage ein neuronales Netz. Dessen Architektur und Implementation wird in den Evolutionsstufen definiert. Abbildung 21 beschreibt den Ablauf. Die Architektur ist abhängig von der Abstraktion des Druckauftrags und wird experimentell entwickelt.

Die Erstellung des Modells findet durch das Training des Netzes statt. Dabei werden einzelne Parameter des Druckauftrags als Eingangs-Attribute (abhängige Attribute) des Datensatzes verwendet. Diesem wird der gemessene Energieverbrauch zugeordnet, er ist damit das Label des Datensatzes. Das neuronale Netz kann so die abhängigen Attribute aus dem Druckauftrag mit dem Label (Energieverbrauch) in Verbindung bringen.

Die Vorhersagequalität des neuronalen Netzes wird durch den Mean Square Error (MSE) definiert. Dieser wird als Gesamtwert für die Ergebnisse aller Datensätze berechnet. Je höher er ist, desto stärker weicht die Prognose des Energieverbrauchs von den Labels (gemessener Verbrauch) ab.

Zur Erstellung der Prognose werden die selben Parameter des Druckauftrags wie beim Training verwendet. Das neuronale Netz liefert für jeden Druckauftrag eine Energieprognose. Dieser Schritt würde im produktiven Betrieb genutzt. Im Proof of Concept ist die Vorhersagequalität relevant, daher entfällt die Erstellung einer Prognose.

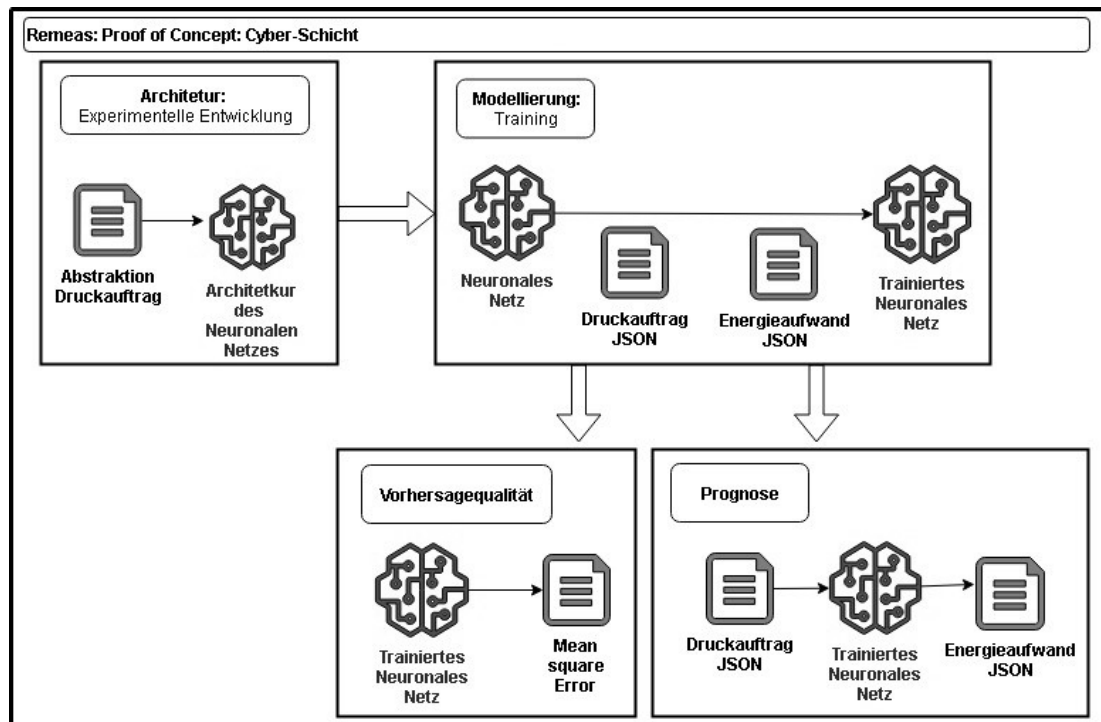


Abbildung 21.: Proof of Concept: Cyber-Schicht

7.6. Implementierung des Systems

Da die nachhaltige Implementierung von maschinellem Lernen nicht im Mittelpunkt des Proof of Concept steht, wird auf vereinfachte Vorgehensweisen zurückgegriffen. Als Basis dient bei der Arbeit mit neuronalen Netzen die Open-Source-Bibliothek Keras. Als Backend wird Tensorflow verwendet, wobei die Kapselung von Keras auch auf andere Backends wie GPU-Tensorflow oder Theano gewechselt werden kann. Die Implementierung wird in Python umgesetzt. Um die Systemumgebung möglichst unabhängig vom Betriebssystem zu halten, wird das Paketverwaltungssystem 'Conda' verwendet.

Für die Berechnungen wurden die in Tabelle 26 gelisteten Rechner verwendet, unter anderen auch ein GPU-Cluster der Fakultät Informatik. Für den Betrieb auf dem GPU-Cluster wurde Tensorflow auf die Nutzung von GPUs umgestellt.

7.7. Experimentelle Entwicklung der Architektur

Die Umsetzung wird experimentell entwickelt. Ausgehend von einer einfachen Struktur wird die Architektur weiter entwickelt, mit dem Ziel die Vorhersagequalität zu maximieren.

System	GPU-Cluster	Server	Notebook
CPU-Cores	8*Intel Xeon	16*AMD Opteron	4*Intel Core
CPU	Xeon 1620	Opteron 6128	Intel Core i5-6200
Leistung (MIPS)	7000	4000	4800
RAM (MB)	1133604	96673	8120
GPU	2*GeForce GTX1080	-	-
Dauer Evo1	14:47	49:10	32:24
Dauer Evo2	5:45	-	-
Dauer Evo3	5:37	-	-
Dauer Evo4	6:30	-	-
Dauer Evo5	3:39	-	-
Dauer Evo6	7:46	-	-
Dauer Evo7	8:53	-	-
Dauer Evo8	4:31	-	-

Tabelle 26.: Verwendete Infrastruktur und Dauer der Prognose-Erstellung in Stunden und Minuten

Die erste Schicht (Input-Layer) orientiert sich an den Eingangs-Attributen (Abhängige Attribute), so wird für jedes Attribut ein Neuron im Input-Layer erzeugt. Die Weiterentwicklung findet in den nachfolgenden Schichten statt. Beginnend mit einem einzelnen schmalen Hidden-Layer (3 Neuronen) werden die Hidden-Layer in den Evolutionsstufen weiterentwickelt. Der Output-Layer besteht immer aus einem Neuron, da immer der Energieverbrauch als einzelner Vorhersagewert gesetzt ist.

7.8. Entwicklungsstufen der Vorhersage

Die Phasen der Evolution:

Phase 0: Prüfen und Optimieren der Daten

Diese Phase dient zur Etablierung der Verarbeitungssysteme und der Datenbasis. Die Daten werden vom Messgerät geladen und in das CSV-Format gebracht und gelabelt. Die Daten werden auf Erfassungsfehler überprüft. Eine Imputation wird durchgeführt, wodurch sich die Anzahl der Druckjobs vervielfacht.

Phase 1: Weiterentwicklung der Daten

In den Evolutionsstufen wird eine einfache Architektur des neuronalen Netzes verwendet. Das neuronale Netz wird mit den Daten trainiert und eine Fehlerabschätzung zur Qualitätsmessung durchgeführt. Die Ergebnisse werden analysiert und die Datenbasis angepasst.

Phase 2: Weiterentwicklung des neuronalen Netzes

In jeder Evolution wird die Architektur weiterentwickelt. Je nach Entwicklung der Vorhersagequalität wird sie angepasst.

7.8.1. Phase 0: Datenoptimierung

Die Datensätze werden vervielfacht, indem der originale Datensatz mit einem Zufallswert, der auf der Gaußschen Normalverteilung liegt, multipliziert wird. Die maximale Abweichung wird auf 5% Abweichung begrenzt, um unrealistische Werte auszuschließen. Siehe Listing D.2 im Anhang. So entstehen aus 73 erfassten Druckaufträgen 210.085 imputierte Datensätze.

7.8.2. Phase 1: Evolution 1 (Basis)

Eine einfache Architektur des Netzes wird verwendet: 1-3-1

Phase-Evolution-Lauf	Datensätze	Ergebnisse
1-1-1	210085	-4039246732.54 MSE (11865742134.72)
1-1-2	210085	-4039246744.62 MSE (11865742114.99)

Tabelle 27.: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE) und Standardabweichung (in Klammern)

Das neuronale Netz wurde zweimal trainiert und eine Fehlerwertung durchgeführt. Die Ergebnisse sind überraschend. Wie in Tabelle 27 zu sehen ist, sind die Fehlerwert beider Läufe sehr hoch. Die zwei Durchläufe ergeben ähnliche Werte, weswegen von nun an nur noch ein Durchlauf vorgenommen wird.

7.8.3. Phase 1: Evolution 2 (Datenanalyse)

Da die Ergebnisse der Evolution 1 nicht zufriedenstellend sind, wurde für die in dieser Evolutionsstufe eine Datenanalyse und Programmanalyse durchgeführt.

Bei den Daten wird grundsätzlich zwischen den gemessenen Daten und den imputierten unterschieden. Datensätze der gemessenen Daten werden genauer untersucht, die imputierten algorithmisch auf zu große Abweichung untersucht und gegebenenfalls gelöscht.

In den gemessenen Daten wurden manuell Datensätze identifiziert, die unrealistische Werte enthalten. Zur Prüfung wurde die Druckzeit betrachtet, Abbildung 22 bildet das Ergebnis ab. Zudem wurden das Verhältnis von Zeit zu Energieverbrauch der einzelnen Drucke verglichen, siehe Abbildung 23. Weicht ein Druckauftrag zu weit ab, wird er als fehlerhaft entfernt.

In der Implementierung wurde die Erkennung von fehlerhaften imputierten Daten verbessert, so werden Messfehler früher erkannt und die fehlerhaften Datensätze entfernt. Die Anzahl der erzeugten Datensätze wurde reduziert.

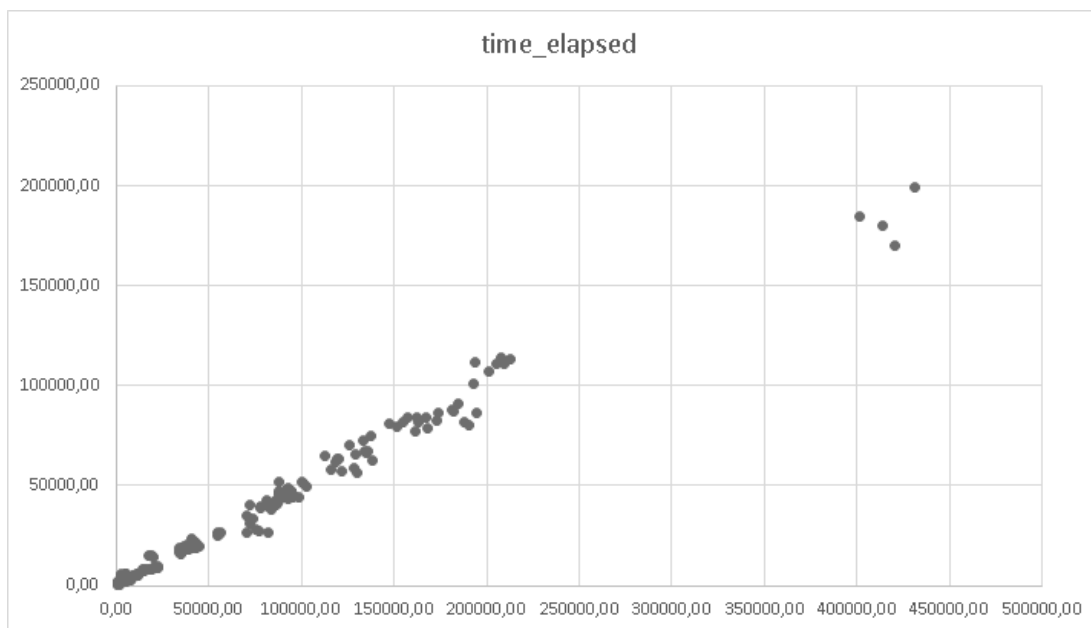


Abbildung 22.: Zeit für Druckaufträge in Sekunden

Die Anzahl der originären Datensätze ist auf 41 gesunken, durch Imputation werden 82.042 Datensätze erzeugt. Das neuronale Netz wird trainiert und eine Fehlerbewertung berechnet. Das Ergebnis hat sich deutlich verbessert, der Fehlerwert hat sich fast halbiert (Tabelle 28).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
1-1-1 (alt)	210085	-4039246732.54 MSE (11865742134.72)
1-2-1 (neu)	82042	-2816580175.71 MSE (5426442172.76)

Tabelle 28.: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)

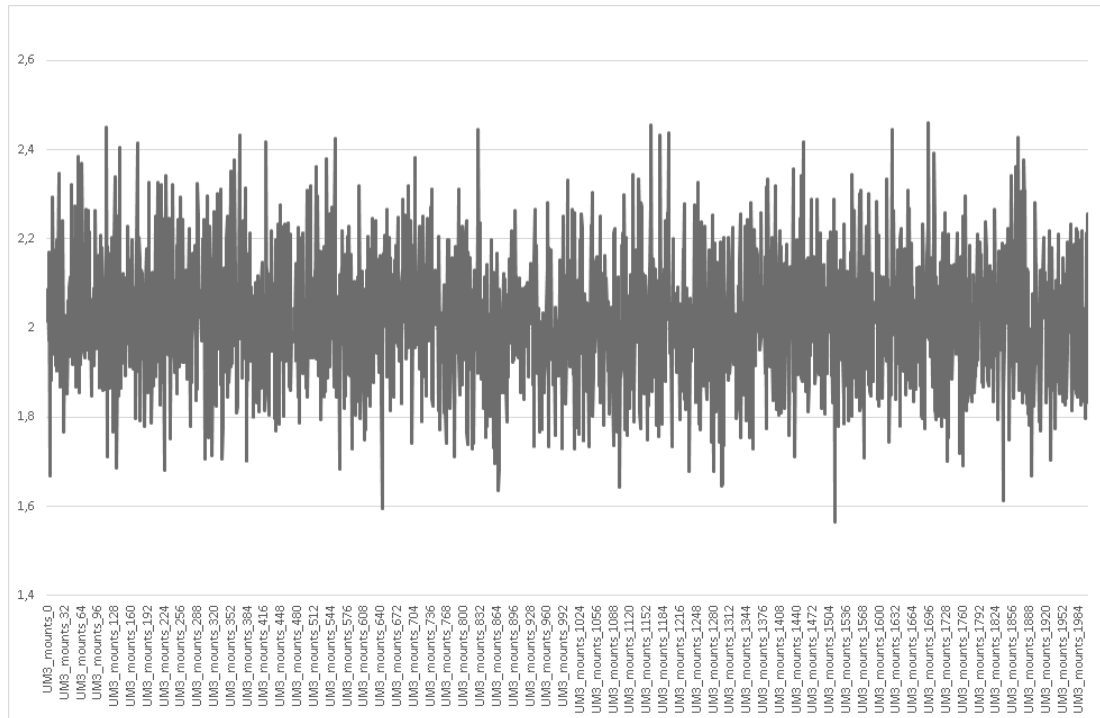


Abbildung 23.: Verhältnis Zeit zu Energie von Druckaufträgen

7.8.4. Phase 1: Evolution 3 (Reduktion)

Um die Qualität der Daten zu testen, wird eine Reduktion der Eingabe-Attribute durchgeführt. Lediglich die Dauer des Druckjobs wird als Attribut verwendet. Aufgrund der hohen Korrelation des Attributs zum Vorhersagewert werden eventuelle Störfaktoren eliminiert.

Phase-Evolution-Lauf	Datensätze	Ergebnisse
1-2-1	82042	-2816580175.71 MSE (5426442172.76)
1-3-1	82042	-6399130558.42 MSE (13543085721.47)

Tabelle 29.: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)

Das Ergebnis hat sich verschlechtert, der Fehlerwert (MSE) hat sich mehr als verdoppelt (Tabelle 29).

7.8.5. Phase 1: Evolution 4 (maximale Abweichung)

Um Fehler bei der Vermehrung der Daten zu erkennen, wird die maximale Abweichung bei der Imputation auf 2% begrenzt.

Das Ergebnis hat sich minimal verbessert (Tabelle 30).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
1-3-1	82042	-6399130558.42 MSE (13543085721.47)
1-4-1	82042	-6354580850.99 MSE (13497787571.80)
Differenz	-	4454970743

Tabelle 30.: Bewertung der Vorhersagen durch mittlere quadratische Abweichung (MSE)

7.8.6. Phase 2: Evo 5 (Tiefer)

Eine tiefere Architektur wird erstellt. Eine zusätzliche Schicht wird eingesetzt. Damit wird die Anzahl der Neuronen erhöht. Die neue Architektur sieht wie folgt aus: 1-3-6-1

Der Abweichung ist gesunken und damit die Vorhersagequalität gestiegen (Tabelle 31).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
1-4-1	82042	-6354580850.99 MSE (13497787571)
2-5-1	82042	-497592483.04 MSE (1262303301)
Differenz	-	5856988368

Tabelle 31.: Entwicklung der Vorhersagequalität von Evo 4 zu Evo 5

7.8.7. Phase 2: Evo 6 (Breiter)

Eine breitere Architektur wird erstellt und entspricht damit: 1-15-1 Die Anzahl von Epochen zum Training werden auf 50 reduziert.

Die Vorhersagequalität ist stark gestiegen (Tabelle 32).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
2-5-1	82042	-497592483.04 MSE (1262303301.67)
2-6-1	82042	-78152660.48 MSE (81377847.64)
Differenz	-	419439822,6

Tabelle 32.: Entwicklung der Vorhersagequalität von Evo 5 zu Evo 6

7.8.8. Phase 2: Evo 7 (Tiefer und Breiter)

Die Architektur wird breiter und tiefer. Sie entspricht: 1-15-9-5-1 Die Anzahl von Epochen zum Training werden auf 100 erhöht.

Die Vorhersagequalität ist stark gestiegen (Tabelle 33).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
2-6-1	82042	-78152660 MSE (81377847.64)
2-7-1	82042	-29693752 MSE (65680644)
Differenz	-	48458908

Tabelle 33.: Entwicklung der Vorhersagequalität von Evo 6 zu Evo 7

7.8.9. Phase 2: Evo 8 (Datenbasis erweitern)

Die Datenbasis wird erweitert, es werden nun die folgenden drei Attribute verwendet: Anzahl Mess-Datensätze, Druckzeit, geschätzte Druckzeit.

Die Vorhersagequalität ist gestiegen (Tabelle 34).

Phase-Evolution-Lauf	Datensätze	Ergebnisse
2-7-1	82042	-29693752 MSE (65680644)
2-8-1	82042	-1225841 MSE (2288374)
Differenz	-	28467911

Tabelle 34.: Entwicklung der Vorhersagequalität von Evo7 zu Evo8

7.9. Evaluation

Die Qualität der Vorhersage ist in hohem Maß von der Datenqualität abhängig, dies hat auch Evolution 2 gezeigt. Daher wurden Maßnahmen ergriffen, um die Datenqualität zu steigern.

Die Reduktion der Input-Attribute hat die Qualität verschlechtert. Irrelevant war die Reduzierung der Abweichung der Daten von 5% auf 2% in der Imputation. Großen Nutzen hatte eine Vergrößerung der Tiefe und der Breite des neuronalen Netzes. Die Evolutionsstufen 7 und 8 nutzen diese Erkenntnisse und erreichen die beste Vorhersagequalität.

Im Verlauf des Proof of Concept konnte die Eignung von neuronalen Netzen gezeigt werden um den Energieverbrauch vorherzusagen. Die Fehlerwerte konnten in den Evolutionsschritten gesenkt werden. Weitere Entwicklung ist notwendig um die Vorhersagequalität zu steigern.

Der Aufwand zur Erstellung der Datenbasis im Proof of Concept war sehr hoch. Zum einen der Zeitaufwand um die Drucke vorzubereiten, zu drucken und nach zu bereiten. Insgesamt hat dies insgesamt etwa 300 Arbeitsstunden umfasst. Von den 82 durchgeführten Drucken waren 73 erfolgreich, 43 wurden korrekt erfasst.

Die korrekte Zuordnung der Druckaufträge zu den Energie-Messreihen war anspruchsvoll. Vermutlich hat die Datenqualität unter der automatisierten Zuordnung gelitten. Eine manuelle Zuordnung war aus Zeitgründen nicht möglich.

8. Fazit und Ausblick

8.1. Fazit

Basis dieser Arbeit war eine tiefgehende Analyse aller Forschungsbereiche die im Zusammenhang zur energieorientierten Produktionsplanung stehen. Sie ist als Survey (Kapitel 2) festgehalten. Ausgeklammert werden dabei drei Bereiche, die im Verhältnis zur Relevanz zu komplex sind: Die genaue Betrachtung von Fertigungsmaschinen, Systeme zur Energieoptimierung durch Effizienzsteigerung sowie die Entscheidungstheorie.

Der umfassende Blick in die Forschung dokumentiert die Fortschritte in der Berücksichtigung des Energieaufwands in der Produktionsplanung. Durch die Integration neuer Technologien, wie dem Konzept von digitalen Zwillingen, wurden in der Vergangenheit zunehmend leistungsfähigere Systeme entworfen. Die nun vorliegende Verwendung von maschinellem Lernen in Kombination modernen Architekturkonzepten führt diese Entwicklung weiter.

Basis für die Modellierung ist die Abstraktion. Die systematische Erfassung der Daten ist in der Forschung kaum beachtet worden, daher wurde in dieser Arbeit ein Fokus darauf gelegt und das Verfahren 'Datenschablone' erstellt.

Die Modellierung im Proof of Concept hat die Schwächen der Methode aufgezeigt. Die Effektivität von Machine Learning verglichen mit etablierten mathematisch-physikalischen Verfahren ist ernüchternd. Zum einen kann nicht sichergestellt werden ob nicht bessere Ergebnisse erzielt werden könnten, zum anderen wird das Ergebnis im besten Fall das der mathematischen Methoden erreichen [Nic].

Versuche die Modellierung per Matlab und Simscape durchzuführen wurden frühzeitig abgebrochen. Es zeigte sich, dass der Aufwand eine holistische Simulation zu erzeugen zu hoch ist. Selbst wenn auf einen Katalog (Simscape) zurückgegriffen wird, ist für die Erstellung einer funktionsfähigen Simulation umfangreiches Domänenwissen erforderlich.

Der Aufwand zur Erstellung der Datenbasis im Proof of Concept war mit 300

Arbeitsstunden sehr hoch. Zudem hat die Datenqualität unter der automatisierten Zuordnung gelitten.

Die Nutzung des Konzepts der digitalen Zwillinge brachte viele Vorteile bei der Entwicklung der Systemarchitektur. Sie können als autonome Subsysteme verwendet werden, die ihre Funktion, ihre Daten und die eigene Leistungsbeschreibung selbständig verwalten.

Durch die Verwendung von Microservice-Pattern konnte ein dynamisches, feingliedriges System entworfen werden, das dennoch keinen Verwaltungs-Overhead hat. Dies ist auch auf die zugrundeliegende Steuerung durch Choreographie zurückzuführen.

Microservices bieten auch organisatorische Vorteile. Aufgrund der Autonomie der Microservices, können diese von autarken Teams entwickelt und betrieben werden (DevOps-Teams). So können Machine-Learning-Experten kompakte Teams bilden und sich auf die Modellierung konzentrieren und innerhalb des Microservices unabhängig agieren.

Die entwickelte Architektur ist nicht aus definiert. Nach Definition der einzelnen Verarbeitungsschritte und Schnittstellen steht einer Implementierung nichts mehr im Wege.

8.2. Ausblick

Die Modellierung kann weiterentwickelt werden. So könnten Verfahren aus verschiedenen Modellierungsdisziplinen verwendet werden. Die bekannten Methoden um physikalisch-mathematische Modelle zu erzeugen, sind zwar leistungsfähig, jedoch statisch und mit Domänenwissen zu erstellen. Methoden mittels Maschine Learning tendieren zu lokalen Ergebnissen und reagieren empfindlich auf die Qualität der Trainingsdaten. Durch Kombination der Methoden lassen sich die jeweiligen Nachteile kompensieren.

Das vorgestellte Verfahren zur Datenerfassung kann durch Methoden aus der Data Science ergänzt werden. So können Daten, die mit der Vorhersage korrelieren, besser identifiziert werden.

Die Anwendung neuronaler Netze wurde nur exemplarisch anhand einfacher Architekturen demonstriert. Wie bereits diskutiert, lassen sich jedoch komplexere Architekturen erstellen.

Die Vorverarbeitung der Daten kann weiterentwickelt werden. Eine Analyse des g-

Codes (CNC-Programm) bietet sich an. So können die Anzahl der Befehle für die Verwendung von energieintensiven Programmschritten kumuliert werden. Beispielsweise die Verwendung von Heizung und Motoren.

Die Verwendung von Zeitverläufen (Time-Series) stellt neben den Daten auch die zeitliche Entwicklung zu Verfügung. Um diese für neuronale Netze aufzubereiten können sie in ein Markov-Transition Field überführt werden. Die Zeitreihen werden dadurch zu einer mehrdimensionalen Matrix umgewandelt. Auf diese können die weit entwickelten Methoden zur Bildverarbeitung, mittels neuronaler Netze, angewendet werden.

Literaturverzeichnis

- [.2016] *Interdisziplinäre Aspekte der Energiewirtschaft*. Springer, 2016
- [AAP⁺19] ARMENDIA, Mikel ; ALZAGA, Aitor ; PEYSSON, Flavien ; FUERTJES, Tobias ; CUGNON, Frédéric ; OZTURK, Erdem ; FLUM, Dominik: Machine Tool: From the Digital Twin to the Cyber-Physical Systems. In: ARMENDIA, Mikel (Hrsg.): *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle*. Springer, 2019, S. 3–21
- [ABS15] ABELE, Eberhard ; BRAUN, Steffen ; SCHRAML, Philipp: Holistic simulation environment for energy consumption prediction of machine tools. In: *Procedia CIRP* 29 (2015), S. 251–256
- [AGE17] *Primärenergieverbrauch 2017*. 2017
- [Agh19] AGHA, Mujtaba: Integrated management of energy and production: Scheduling of batch process and Combined Heat & Power (CHP) plant. (2019)
- [AGOP19] ARMENDIA, Mikel (Hrsg.) ; GHASSEMPOURI, Mani (Hrsg.) ; OZTURK, Erdem (Hrsg.) ; PEYSSON, Flavien (Hrsg.): *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle*. Cham : Springer International Publishing, 2019. <http://dx.doi.org/10.1007/978-3-030-02203-7>. <http://dx.doi.org/10.1007/978-3-030-02203-7>. – ISBN 9783030022044
- [AIP19] AIP (Hrsg.): *AIP Conference Proceedings*. 2019
- [Alt13] ALT, Walter: *Nichtlineare Optimierung: Eine Einführung in Theorie, Verfahren und Anwendungen*. Springer-Verlag, 2013
- [Arm19] ARMENDIA, Mikel (Hrsg.): *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle*. Springer, 2019
- [ASB⁺19] ABELE, Eberhard ; SCHRAML, Philipp ; BECK, Martin ; FLUM, Dominik ; EISELE, Christian: Two Practical Approaches to Assess the Energy Demand of Production Machines. In: THIEDE, Sebastian (Hrsg.): *Eco-Factories of the Future*. Springer, 2019, S. 127–146
- [BCK19] BAMBERG, Günter ; COENENBERG, Adolf G. ; KRAPP, Michael: *Betriebswirtschaftliche entscheidungslehre*. Vahlen, 2019
- [Bei17] BEIER, Jan: Simulation Approach Towards Energy Flexible Manufacturing Systems. (2017). <http://dx.doi.org/10.1007/978-3-319->

- 46639-2. – DOI 10.1007/978-3-319-46639-2
- [BHM] BARTHEL, Kai U. ; HEZEL, Nico ; MACKOWIAK, Radek: Graph-based browsing for large video collections, S. 237–242
- [BK13] BLES, Markus ; KESSLER, Alois: *Energieeffizienz in der Industrie*. Springer, 2013
- [Bob] BOB WARFIELD: *Secrets of Going From CAD, Image, DXF, or STL to GCode for CNC and 3D Printing*. <https://www.cnccookbook.com/secrets-going-cad-image-dxf-stl-gcode-cnc-3d-printing/>
- [Bon02] BONNESCHKY, Alexis: *Energiekennzahlen in PPS-Systemen: Zugl.: Cottbus, Techn. Univ., Diss., 2002 u.d.T.: Bonneschky, Alexis: Integration energiewirtschaftlicher Aspekte in Systeme der Produktionsplanung und -steuerung*. Als Ms. gedr. Berlin : dissertation.de, 2002 (BTU-Forschungshefte Energie). – ISBN 3-89825-454-2
- [C. 08] C. MOUZON, Gilles: Operational methods and models for minimization of energy consumption in a manufacturing environment. (2008)
- [CA09] CIURANA, J. ; ARIAS, G. ; OZEL, Tugrul: Neural network modeling and particle swarm optimization (PSO) of process parameters in pulsed laser micromachining of hardened AISI H13 steel. In: *Materials and Manufacturing Processes* 24 (2009), Nr. 3, S. 358–368
- [CBNR18] CHIABERT, Paolo ; BOURAS, Abdelaziz ; NOËL, Frédéric ; R\ \IOS, José: *Product Lifecycle Management to Support Industry 4.0: 15th IFIP WG 5.1 International Conference, PLM 2018, Turin, Italy, July 2-4, 2018, Proceedings*. Bd. 540. Springer, 2018
- [CFF17] CHHETRI, S. R. ; FAEZI, Sina ; FARUQUE, M. A.: Digital twin of manufacturing systems. In: *Center for Embedded & Cyber-Physical Systems* (2017)
- [CIR06] *13th CIRP international conference on life cycle engineering*. 2006
- [CLL+18] CORONADO, Pedro Daniel U. ; LYNN, Roby ; LOUHICHI, Wafa ; PARTO, Mahmoud ; WESCOAT, Ethan ; KURFESS, Thomas: Part data integration in the Shop Floor Digital Twin: Mobile and cloud technologies to enable a manufacturing execution system. In: *Journal of manufacturing systems* 48 (2018), S. 25–33
- [CMP04] CAVALIERI, Sergio ; MACCARRONE, Paolo ; PINTO, Roberto: Parametric vs. neural network models for the estimation of production costs: A case study in the automotive industry. In: *International Journal of Production Economics* 91 (2004), Nr. 2, 165–177. <http://dx.doi.org/10.1016/j.ijpe.2003.08.005>. – DOI 10.1016/j.ijpe.2003.08.005
- [CQGR08] CIURANA, J. ; QUINTANA, G. ; GARCIA-ROMEU, M. L.: Estimating

- the cost of vertical high-speed machining centres, a comparison between multiple regression analysis and the neural networks approach. In: *International Journal of Production Economics* 115 (2008), Nr. 1, S. 171–178
- [CTT19] CHAMBERLAIN, Roger ; TAHA, Walid ; TÖRNGREN, Martin: *Cyber Physical Systems. Design, Modeling, and Evaluation: 7th International Workshop, CyPhy 2017, Seoul, South Korea, October 15-20, 2017, Revised Selected Papers*. 2019 (Information Systems and Applications, incl. Internet/Web, and HCI). <https://doi.org/10.1007/978-3-030-17910-6>. – ISBN 978-3-030-17910-6
- [DB13] DEAN, Jeffrey ; BARROSO, Luiz A.: The Tail at Scale. In: *Commun. ACM* 56 (2013), Nr. 2, 74–80. <http://dx.doi.org/10.1145/2408776.2408794>. – DOI 10.1145/2408776.2408794. – ISSN 0001-0782
- [DHA+18] DURÃO, Luiz Fernando C. ; HAAG, Sebastian ; ANDERL, Reiner ; SCHÜTZER, Klaus ; ZANCUL, Eduardo: Digital Twin Requirements in the Context of Industry 4.0. In: *IFIP International Conference on Product Lifecycle Management*, 2018, S. 204–214
- [Din91] DINKELBACH, Werner: *Entscheidungstheorie*. Univ., Fachbereich Wirtschaftswiss, 1991
- [Diza] DIZINGOF: *Concentric Apple - Math Art*. thingiverse.com/thing:1537142/files
- [Dizb] DIZINGOF: *Thingiverse*. thingiverse.com
- [DWZ+18] DEBROY, Tarasankar ; WEI, H. L. ; ZUBACK, J. S. ; MUKHERJEE, T. ; ELMER, J. W. ; MILEWSKI, J. O. ; BEESE, Allison M. ; WILSON-HEID, A. ; DE, A. ; ZHANG, W.: Additive manufacturing of metallic components—process, structure and properties. In: *Progress in Materials Science* 92 (2018), S. 112–224
- [DZTB17] DEBROY, Tarasankar ; ZHANG, Wei ; TURNER, J. ; BABU, Sudarshanam S.: Building digital twins of 3D printing machines. In: *Scripta Materialia* 135 (2017), S. 119–124
- [EDO18] *2018 IEEE 22nd International Enterprise Distributed Object Computing Workshop (EDOCW)*. 2018
- [Ehr07] EHRENSPIEL, Klaus: *Kostengünstig Entwickeln und Konstruieren*. Springer, 2007
- [Eis14] EISELE, Christian: *Simulationsgestützte Optimierung des elektrischen Energiebedarfs spanender Werkzeugmaschinen*. Shaker, 2014
- [Eve13] EVERSHEIM, Walter: *Organisation in der Produktionstechnik Band 4: Fertigung und Montage*. Springer-Verlag, 2013

- [Fal00] FALLBÖHMER, Markus: *Generieren alternativer Technologieketten in frühen Phasen der Produktentwicklung*. Shaker, 2000
- [Fro18] FROCHTE, Jörg: *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. München : Hanser, 2018. – ISBN 978-3-446-45291-6
- [FSSA19] FLUM, Dominik ; SOSENHEIMER, Johannes ; STÜCK, Christian ; ABEL, Eberhard: Towards Energy-Efficient Machine Tools Through the Development of the Twin-Control Energy Efficiency Module. In: ARMENDIA, Mikel (Hrsg.): *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle*. Springer, 2019, S. 95–110
- [FTM+11] FAIZI AL-H, Muataz H. ; T ADESTA, Erry Y. ; MOHD ALI, Afifah ; AGUSMAN, Delvis ; YUHAN SUPR, Mohammad: Energy cost modeling for high speed hard turning. In: *Journal of Applied Sciences* 11 (2011), S. 2578–2584
- [GDT06] GUTOWSKI, Timothy ; DAHMUS, Jeffrey ; THIRIEZ, Alex: Electrical energy requirements for manufacturing processes. In: *13th CIRP international conference on life cycle engineering* Bd. 31, 2006, S. 623–638
- [GKM17] GYULAI, Dávid ; KÁDÁR, Botond ; MONOSTORI, László: Scheduling and operator control in reconfigurable assembly systems. In: *Procedia CIRP* 63 (2017), S. 459–464
- [GTP15] GIRET, A. ; TRENTESAUX, Damien ; PRABHU, Vittal: Sustainability in manufacturing operations scheduling: A state of the art review. In: *Journal of manufacturing systems* 37 (2015). <http://dx.doi.org/10.1016/j.jmsy.2015.08.002>. – DOI 10.1016/j.jmsy.2015.08.002
- [Haa] HAAG, Holger: *Eine Methodik zur modellbasierten Planung und Bewertung der Energieeffizienz in der Produktion*, Diss.
- [Hag15] HAGL, Richard: *Das 3D-Druck-Kompendium: Leitfaden für Unternehmer, Berater und Innovationstreiber*. 2. Aufl. Wiesbaden : Springer Gabler, 2015. – ISBN 978-3-658-07046-5
- [Haw19] HAWKINS, Calvin: An Energy Profile Model for Fused Deposition Modeling 3D Printing Process. (2019)
- [HLHH12] HU, Shaohua ; LIU, Fei ; HE, Yan ; HU, Tong: An on-line approach for energy efficiency monitoring of machine tools. In: *Journal of Cleaner Production* 27 (2012), 133–140. <http://dx.doi.org/10.1016/j.jclepro.2012.01.013>. – DOI 10.1016/j.jclepro.2012.01.013
- [HLLL17] HUMAYED, Abdulmalik ; LIN, Jingqiang ; LI, Fengjun ; LUO, Bo: Cyber-physical systems security—A survey. In: *IEEE Internet of Things Journal* 4 (2017), Nr. 6, S. 1802–1831
- [HNT+18] HU, Liwen ; NGUYEN, Ngoc-Tu ; TAO, Wenjin ; LEU, Ming C. ; LIU,

- Xiaoqing F. ; SHAHRIAR, Md R. ; AL SUNNY, S. N.: Modeling of Cloud-Based Digital Twins for Smart Manufacturing with MT Connect. In: *Procedia Manufacturing* 26 (2018), S. 1193–1203
- [HPT13] HERRMANN, Christoph ; POSSELT, Gerrit ; THIEDE, Sebastian: *Energie- und hilfsstoffoptimierte Produktion*. Springer-Verlag, 2013
- [HT16] HARUHIKO SUWA ; TETSUO SAMUKAWA: A New Framework of Energy-efficient Manufacturing Systems Based on Energy Load Profiles. In: *Procedia CIRP* 41 (2016), 313–317. <http://dx.doi.org/10.1016/j.procir.2015.10.011>. – DOI 10.1016/j.procir.2015.10.011
- [IEE08] *2008 Chinese Control and Decision Conference*. 2008
- [Jas19] JASON BROWNLEE: *Deep Learning With Python*. 2019 <https://machinelearningmastery.com/deep-learning-with-python/>
- [JFS⁺19] JAVED, Yahya ; FELEMBAN, Muhamad ; SHAWLY, Tawfeeq ; KOBES, Jason ; GHAFOR, Arif: A Partition-Driven Integrated Security Architecture for Cyber-Physical Systems. In: *arXiv preprint arXiv:1901.03018* (2019)
- [Jul19] JULIAN MOESER: *Künstliche neuronale Netze – Aufbau & Funktionsweise*. <https://jaai.de/kuenstliche-neuronale-netze-aufbau-funktion-291/>. Version: 2019
- [JUM⁺16] JASMIN GUTH ; UWE BREITENBUCHER ; MICHAEL FALKENTHAL ; FRANK LEYMANN ; LUKAS REINFURT: *Comparison of IoT Platform Architectures: A Field Study based on a Reference Architecture: Paris, France, November 23-25, 2016*. Piscataway, NJ : IEEE, 2016 <http://ieeexplore.ieee.org/servlet/opac?punumber=7867028>. – ISBN 9781509049592
- [Jun07] JUNGE, M.: *Simulationsgestützte Entwicklung und Optimierung einer energieeffizienten Produktionssteuerung*. 2007
- [KBB⁺16] KONDZIELLA, Hendrik ; BROD, Kristina ; BRUCKNER, Thomas ; OLBERT, Sebastian ; MES, Florian: Stromspeicher für die „Energiewende“ – eine akteursbasierte Analyse der zusätzlichen Speicherkosten. In: *Interdisziplinäre Aspekte der Energiewirtschaft*. Springer, 2016, S. 179–190
- [Kel] KELLER, Fabian K.: *Methodik zur energiebezugsorientierten Auftragsplanung*, Technische Universität München, Diss.
- [KLL18] KELLNER, Florian ; LIENLAND, Bernhard ; LUKESCH, Maximilian: *Produktionswirtschaft: Planung, Steuerung und Industrie 4.0*. Berlin : Springer Gabler, 2018. <http://dx.doi.org/10.1007/978-3-662-54341-2>. <http://dx.doi.org/10.1007/978-3-662-54341-2>. – ISBN 978-3-662-54341-2

- [KLT] KEMMOE, Sylvain ; LAMY, Damien ; TCHERNEV, Nikolay: A Job-shop with an Energy Threshold Issue Considering Operations with Consumption Peaks. <http://dx.doi.org/10.1016/j.ifacol.2015.06.179>. – DOI 10.1016/j.ifacol.2015.06.179
- [KMZ⁺17] KNAPP, G. L. ; MUKHERJEE, T. ; ZUBACK, J. S. ; WEI, H. L. ; PALMER, T. A. ; DE, A. ; DEBROY, Tarasankar: Building blocks for a digital twin of additive manufacturing. In: *Acta Materialia* 135 (2017), S. 390–399
- [LBK15] LEE, Jay ; BAGHERI, Behrad ; KAO, Hung-An: A cyber-physical systems architecture for industry 4.0-based manufacturing systems. In: *Manufacturing letters* 3 (2015), S. 18–23
- [LDH⁺13] LUO, Hao ; DU, Bing ; HUANG, George ; CHEN, Huaping ; LI, Xiaolin: Hybrid flow shop scheduling considering machine electricity consumption cost. In: *International Journal of Production Economics* 146 (2013), S. 423–439. <http://dx.doi.org/10.1016/j.ijpe.2013.01.028>. – DOI 10.1016/j.ijpe.2013.01.028
- [LDL⁺13] LIU, Ying ; DONG, Haibo ; LOHSE, Niels ; PETROVIC, Sanja ; GINDY, Nabil: An Investigation into Minimising Total Energy Consumption and Total Weighted Tardiness in Job Shops. In: *Journal of Cleaner Production* 65 (2013). <http://dx.doi.org/10.1016/j.jclepro.2013.07.060>. – DOI 10.1016/j.jclepro.2013.07.060
- [LGG18] LIAO, Wei ; GARG, Akhil ; GAO, Liang: Design of robust energy consumption model for manufacturing process considering uncertainties. In: *Journal of Cleaner Production* 172 (2018), S. 119–132
- [LGSM95] LAUX, Helmut ; GILLENKIRCH, Robert M. ; SCHENK-MATHES, Heike Y.: *Entscheidungstheorie*. Bd. 4. Springer, 1995
- [LGW19] LIU, Zhengchao ; GUO, Shunsheng ; WANG, Lei: Integrated green scheduling optimization of flexible job shop and crane transportation considering comprehensive energy consumption. In: *Journal of Cleaner Production* 211 (2019), S. 765–786
- [LK11] LI, W. ; KARA, S.: An empirical model for predicting energy consumption of manufacturing processes: a case of turning process. In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 225 (2011), Nr. 9, S. 1636–1646
- [LKPM17] LEE, Wonkyun ; KIM, Seong H. ; PARK, Jaesang ; MIN, Byung-Kwon: Simulation-based machining condition optimization for machine tool energy consumption reduction. In: *Journal of Cleaner Production* 150 (2017), S. 352–360
- [LLQ17] LIU, Peiji ; LIU, Fei ; QIU, Hang: A novel approach for acquiring the real-time energy efficiency of machine tools. In: *Energy*

- 121 (2017). <http://dx.doi.org/10.1016/j.energy.2017.01.047>.
– DOI 10.1016/j.energy.2017.01.047
- [LLWZ19] LIU, Qing ; LIU, Bin ; WANG, Guan ; ZHANG, Chen: A comparative study on digital twin models. In: AIP (Hrsg.): *AIP Conference Proceedings* Bd. 2073, 2019, S. 020091
- [LöD16] LÖDDING, Hermann: *Verfahren der Fertigungssteuerung*. 2016. <http://dx.doi.org/10.1007/978-3-662-48459-3>. <http://dx.doi.org/10.1007/978-3-662-48459-3>
- [LX17] LIU, Chao ; XU, Xun: Cyber-physical machine tool—the era of machine tool 4.0. In: *Procedia CIRP* 63 (2017), S. 70–75
- [LXL15] LIU, Fei ; XIE, Jun ; LIU, Shuang: A method for predicting the energy consumption of the main driving system of a machine tool in a machining process. In: *Journal of Cleaner Production* 105 (2015), S. 171–177
- [LYX13] LI, Lin ; YAN, Jihong ; XING, Zhongwen: Energy requirements evaluation of milling machines based on thermal equilibrium and empirical modelling. In: *Journal of Cleaner Production* 52 (2013), S. 113–121
- [LZL15] LIU, N. ; ZHANG, Y. F. ; LU, W. F.: A hybrid approach to energy consumption modelling based on cutting power: a milling case. In: *Journal of Cleaner Production* 104 (2015), S. 264–272
- [LZY+19] LENG, Jiewu ; ZHANG, Hao ; YAN, Douxi ; LIU, Qiang ; CHEN, Xin ; ZHANG, Ding: Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop. In: *Journal of Ambient Intelligence and Humanized Computing* 10 (2019), Nr. 3, S. 1155–1166
- [LZZ08] LIU, Xiang ; ZOU, Fengxing ; ZHANG, Xiangping: Mathematical model and genetic optimization for hybrid flow shop scheduling problem based on energy consumption. In: *2008 Chinese Control and Decision Conference*, 2008, S. 1002–1007
- [MAH18] MILLER, Alexander M. ; ALVAREZ, Ramon ; HARTMAN, Nathan: Towards an extended model-based definition for the digital twin. In: *Computer-Aided Design and Applications* 15 (2018), Nr. 6, S. 880–891
- [MCST19] MODONI, Gianfranco E. ; CALDAROLA, Enrico G. ; SACCO, Marco ; TERKAJ, Walter: Synchronizing physical and digital factory: benefits and technical challenges. In: *Procedia CIRP* 79 (2019), S. 472–477
- [MD19] MUKHERJEE, T. ; DEBROY, Tarasankar: A digital twin for rapid qualification of 3D printed metallic components. In: *Applied Materials Today* 14 (2019), S. 59–65
- [MMLL15] MADAN, Jatinder ; MANI, Mahesh ; LEE, Jae H. ; LYONS, Kevin W.: Energy performance evaluation and improvement of unit-manufacturing

- processes: injection molding case study. In: *Journal of Cleaner Production* 105 (2015), S. 157–170
- [MMR97] MEHROTRA, Kishan ; MOHAN, Chilukuri K. ; RANKA, Sanjay: *Elements of artificial neural networks*. MIT press, 1997
- [MRR⁺18] MONIZZA, G. P. ; ROJAS, Rafael A. ; RAUCH, Erwin ; GARCIA, M. R. ; MATT, Dominik T.: A Case Study in Learning Factories for Real-Time Reconfiguration of Assembly Systems Through Computational Design and Cyber-Physical Systems. In: *IFIP International Conference on Product Lifecycle Management*, 2018, S. 227–237
- [MT16] MATZEN, Frank J. ; TESCH, Ralf: *Industrielle Energiestrategie: Praxishandbuch für Entscheider des produzierenden Gewerbes*. Springer, 2016
- [Nad16] NADAREISHVILI, Irakli: *Microservice architecture: Aligning principles, practices, and culture*. Sebastopol, CA : O'Reilly Media, 2016 <http://proquest.tech.safaribooksonline.de/9781491956328>. – ISBN 978-1-4919-5625-0
- [New15] NEWMAN, Sam: *Microservices: Konzeption und Design*. Frechen : mitp, 2015 <http://gbv.ebib.com/patron/FullRecord.aspx?p=2089872>. – ISBN 978-3-95845-081-3
- [NGP⁺] NIKOLAEV, S. ; GUSEV, M. ; PADALITSA, D. ; MOZHENKOV, E. ; MISHIN, S. ; UZHINSKY, I.: Implementation of “Digital Twin” Concept for Modern Project-Based Engineering Education, S. 193–203
- [Nic] NICO HEZEL: *KNN: Vorteile und Nachteile*. <https://data-science-blog.com/blog/2015/09/12/knn-vorteile-und-nachteile/>
- [NKI⁺18] NGO, Tuan D. ; KASHANI, Alireza ; IMBALZANO, Gabriele ; NGUYEN, Kate T. Q. ; HUI, David: Additive manufacturing (3D printing): A review of materials, methods, applications and challenges. In: *Composites Part B: Engineering* 143 (2018), S. 172–196
- [NOM18] NOMS (Hrsg.): *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*. 2018
- [Pat97] PATTERSON, Dan W.: *Künstliche neuronale Netze: das Lehrbuch*. Prentice Hall, 1997
- [Pen16] PENG, Tao: Analysis of energy utilization in 3d printing processes. In: *Procedia CIRP* 40 (2016), S. 62–67
- [PH18] PAPON, Easir A. ; HAQUE, Anwarul: Tensile Properties, Void Contents, Dispersion and Fracture Behaviour of 3D printed Carbon Nanofiber Reinforced Composites. In: *Journal of Reinforced Plastics and Composites* 37 (2018), S. 381–395. <http://dx.doi.org/10.1177/0731684417750477>. – DOI 10.1177/0731684417750477

- [PJIZ18] PREUVENEERS, Davy ; JOOSEN, Wouter ; ILIE-ZUDOR, Elisabeth: Robust digital twin compositions for Industry 4.0 smart manufacturing systems. In: *2018 IEEE 22nd International Enterprise Distributed Object Computing Workshop (EDOCW)*, 2018, S. 69–78
- [PL18] PAHL, M. ; LOIPFINGER, M.: Machine learning as a reusable micro-service. In: NOMS (Hrsg.): *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018, S. 1–7
- [Pla18] PLATZER, André: *Logical foundations of cyber-physical systems*. Cham : Springer, 2018. – ISBN 978–3–319–63588–0
- [PLZ17] POPP, Richard S-H ; LIEBL, Corinna ; ZAEH, Michael F.: Evaluation of the energy flexible operation of machine tool components. In: *Procedia CIRP* 63 (2017), S. 76–81
- [PS17] PENG, Tao ; SUN, Wenjun: Energy modelling for FDM 3D printing from a life cycle perspective. In: *International Journal of Manufacturing Research* 12 (2017), Nr. 1, S. 83–98
- [QCR11] QUINTANA, G. ; CIURANA, J. ; RIBATALLADA, J.: Modelling power consumption in ball-end milling operations. In: *Materials and Manufacturing Processes* 26 (2011), Nr. 5, S. 746–756
- [RA14] R. DRATH ; A. HORCH: Industrie 4.0: Hit or Hype? [Industry Forum]. In: *IEEE Industrial Electronics Magazine* 8 (2014), Nr. 2, S. 56–58. <http://dx.doi.org/10.1109/MIE.2014.2312079>. – DOI 10.1109/MIE.2014.2312079. – ISSN 1932–4529
- [Ral17] RALF MOCK: *Von der Mechatronik zu Cyber-physikalischen Systemen*. <https://blog.zhaw.ch/industrie4null/2017/02/06/von-der-mechatronik-zu-cyber-physikalischen-systemen/>. Version:2017
- [Rat16] RATTAT, Christian: *3D-Druck für Anspruchsvolle: Mit dem Ultimaker perfekte Werkstücke erstellen*. 1. Aufl. s.l. : dpunkt, 2016 <http://gbv.ebib.com/patron/FullRecord.aspx?p=4405482>. – ISBN 3864918804
- [RD08] RAGER, Markus ; D A TUMA, P.: *Energieorientierte Produktionsplanung: Analyse, Konzeption und Umsetzung*. 2008. <http://dx.doi.org/10.1007/978-3-8350-5569-8>. <http://dx.doi.org/10.1007/978-3-8350-5569-8>
- [RK12] RIEF, Markus ; KARPUSCHEWSKI, Bernhard: *Berichte aus dem Institut für Fertigungstechnik und Qualitätssicherung Magdeburg*. Bd. 29: *Vorhersagemodell für den Energiebedarf bei der spanenden Bearbeitung für eine energieeffiziente Prozessgestaltung: Zugl.: Magdeburg, Univ., Fak. für Maschinenbau, Diss., 2012*. Aachen : Shaker, 2012. – ISBN 978–3844015409

- [Scha] SCHULTZ, Cedric M.: *System zur energieorientierten Produktionssteuerung in der auftragsbezogenen Fertigung*, Universitätsbibliothek der TU München, Diss.
- [Schb] SCHWEIHER WERKZEUGBAU GMBH & CO. KG: *CAD / CAM*. <http://www.schweiher-werkzeugbau.de/leistungen/cad>
- [Sch14] SCHREMS, Sebastian: *Methode zur modellbasierten Integration des maschinenbezogenen Energiebedarfs in die Produktionsplanung*. Shaker, 2014
- [SCY15] S. HUANG ; C. LU ; Y. LO: Evaluation of AMI and SCADA Data Synergy for Distribution Feeder Modeling. In: *IEEE Transactions on Smart Grid* 6 (2015), Nr. 4, S. 1639–1647. <http://dx.doi.org/10.1109/TSG.2015.2408111>. – DOI 10.1109/TSG.2015.2408111. – ISSN 1949–3053
- [SFF13] SFFS (Hrsg.): *solid freeform fabrication symposium, Austin, TX*. 2013
- [Shu12] SHUANG, L. Fei L.: Multi-period Energy Model of Electro-mechanical Main Driving System during the Service Process of Machine Tools [J]. In: *Journal of Mechanical Engineering* 21 (2012), S. 020
- [SK97] SPUR, Günter ; KRAUSE, Frank-Lothar: *Das virtuelle Produkt: Management der CAD-Technik*. Hanser München, 1997
- [Spr18a] *IFIP International Conference on Product Lifecycle Management*. 2018
- [Spr18b] *IFIP International Conference on Product Lifecycle Management*. 2018
- [Spr18c] *IFIP International Conference on Product Lifecycle Management*. 2018
- [SPSH15] SPROEDT, A. ; PLEHN, J. ; SCHÖNSLEBEN, P. ; HERRMANN, C.: A simulation-based decision support for eco-efficiency improvements in production systems. In: *Journal of Cleaner Production* 105 (2015), S. 389–405
- [SRGR17] SIMON, Peter ; ROLTSCH, Floris ; GLASSCHRÖDER, Johannes ; REINHART, Gunther: Approach for a Potential Analysis of Energy Flexible Production Systems. In: *Procedia CIRP* 63 (2017), S. 580–585
- [Sta17] *Energieverwendung der Betriebe im Verarbeitenden Gewerbe 2017*. 2017
- [SU09] SWAN, Lukas G. ; UGURSAL, V. I.: Modeling of end-use energy consumption in the residential sector: A review of modeling techniques. In: *Renewable and sustainable energy reviews* 13 (2009), Nr. 8, S. 1819–1835
- [SWRS19] SHIN, Seung-Jun ; WOO, Jungyub ; RACHURI, Sudarsan ; SEO, Wonchul: An energy-efficient process planning system using machine-monitoring data: A data analytics approach. In: *Computer-Aided Design* 110 (2019), S. 92–109

- [SZS15] SHARMA, Abhay ; ZHAO, Fu ; SUTHERLAND, John: Econological Scheduling of a Manufacturing Enterprise Operating under a Time-of-Use Electricity Tariff. In: *Journal of Cleaner Production* 108 (2015). <http://dx.doi.org/10.1016/j.jclepro.2015.06.002>. – DOI 10.1016/j.jclepro.2015.06.002
- [TBAP12] TONELLI, Flavio ; BRUZZONE, Alessandro ; ANGHINOLFI, Davide ; PAOLUCCI, Massimo: Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops. In: *CIRP Annals - Manufacturing Technology* 61 (2012), S. 459–462. <http://dx.doi.org/10.1016/j.cirp.2012.03.084>. – DOI 10.1016/j.cirp.2012.03.084
- [Thi19] THIEDE, Sebastian (Hrsg.): *Eco-Factories of the Future*. Springer, 2019
- [VDCC08] VERLINDEN, B. ; DUFLOU, J. R. ; COLLIN, P. ; CATTRYSSE, D.: Cost estimation for sheet metal parts using multiple regression and artificial neural networks: A case study. In: *International Journal of Production Economics* 111 (2008), Nr. 2, 484–492. <http://dx.doi.org/10.1016/j.ijpe.2007.02.004>. – DOI 10.1016/j.ijpe.2007.02.004
- [VR13] VERMA, Anoop ; RAI, Rahul: Energy efficient modeling and optimization of additive manufacturing processes. In: SFFS (Hrsg.): *solid freeform fabrication symposium, Austin, TX, 2013*, S. 231–241
- [Wei10] WEINERT, Nils: Vorgehensweise für Planung und Betrieb energieeffizienter Produktionssysteme. In: *Fraunhofer IPK* (2010)
- [Wen10] WENZEL, Sigrid: VDI-Richtlinien zur Modellbildung und Simulation. In: *Integrationsaspekte der Simulation: Technik, Organisation, Personal. Tagungsband der 14* (2010), S. 543–550
- [XCS+12] XIE, Dong ; CHEN, Guorong ; SHI, Jinliang ; XU, Dijian ; WANG, Feng: Research on CNC machine tool cutting energy consumption based on BP neural network. In: *Machine Tool & Hydraulics* 1 (2012)
- [XD19] XU, Li D. ; DUAN, Lian: Big data for cyber physical systems in industry 4.0: a survey. In: *Enterprise Information Systems* 13 (2019), Nr. 2, S. 148–169
- [ZG13] ZIMMERMANN, Hans-Jürgen ; GUTSCHE, Lothar: *Multi-Criteria Analyse: Einführung in die Theorie der Entscheidungen bei Mehrfachzielsetzungen*. Springer-Verlag, 2013
- [ZLC16] ZHOU, Xiaona ; LIU, Fei ; CAI, Wei: An energy-consumption model for establishing energy-consumption allowance of a workpiece in a machining system. In: *Journal of Cleaner Production* 135 (2016), S. 1580–1590
- [ZLH+17] ZHAO, G. Y. ; LIU, Z. Y. ; HE, Y. ; CAO, H. J. ; GUO, Y. B.: Energy consumption in machining: Classification, prediction, and reduction strategy.

In: *Energy* 133 (2017), S. 142–157

- [ZLX18] ZHUANG, Cunbo ; LIU, Jianhua ; XIONG, Hui: Digital twin-based smart production management and control framework for the complex product assembly shop-floor. In: *The International Journal of Advanced Manufacturing Technology* 96 (2018), Nr. 1-4, S. 1149–1163
- [ZZZZ14] ZHAO, Guoyong ; ZHAO, Qingzhi ; ZHENG, Guangming ; ZHAI, Jingtao: Specific Energy Consumption Prediction Method Based on Machine Tool Power Measurement. In: *Sensors and Transducers* 174 (2014), S. 115–122

Eidesstattliche Erklärung

Ich versichere, dass ich die vorstehende Arbeit selbständig verfasst und hierzu keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Stellen der Arbeit die wörtlich oder sinngemäß aus fremden Quellen entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt oder an anderer Stelle veröffentlicht.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

Furtwangen, den 25.09.2019 Daniel Schönle

A. Energieverbrauch

25. Beitrag erneuerbarer Energien am Primärenergieverbrauch in Deutschland

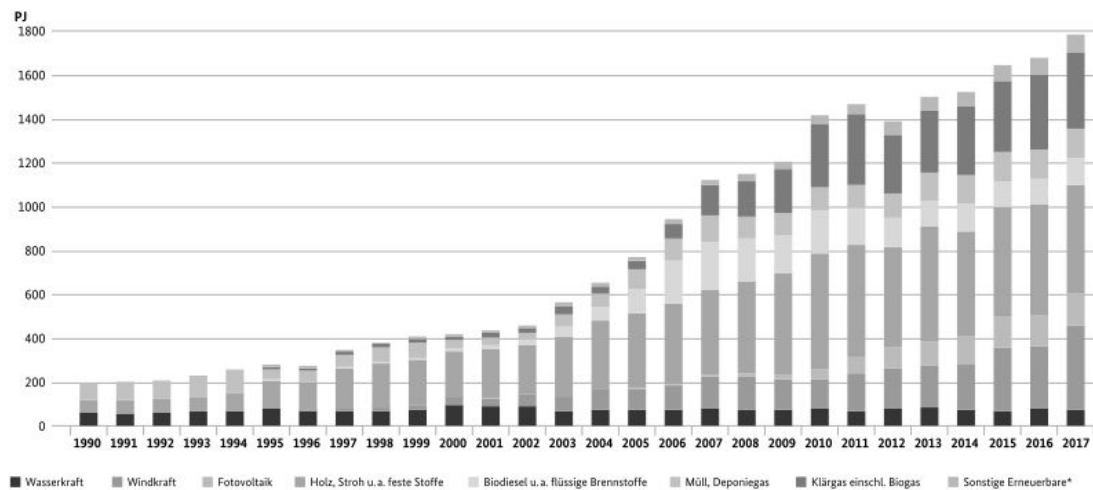


Abbildung A.1.: Anteil erneuerbarer Energie 2017[Sta17]

Anteil wirtschaftlicher Aktivitäten am Primärenergieverbrauch aller Produktionsbereiche 2015

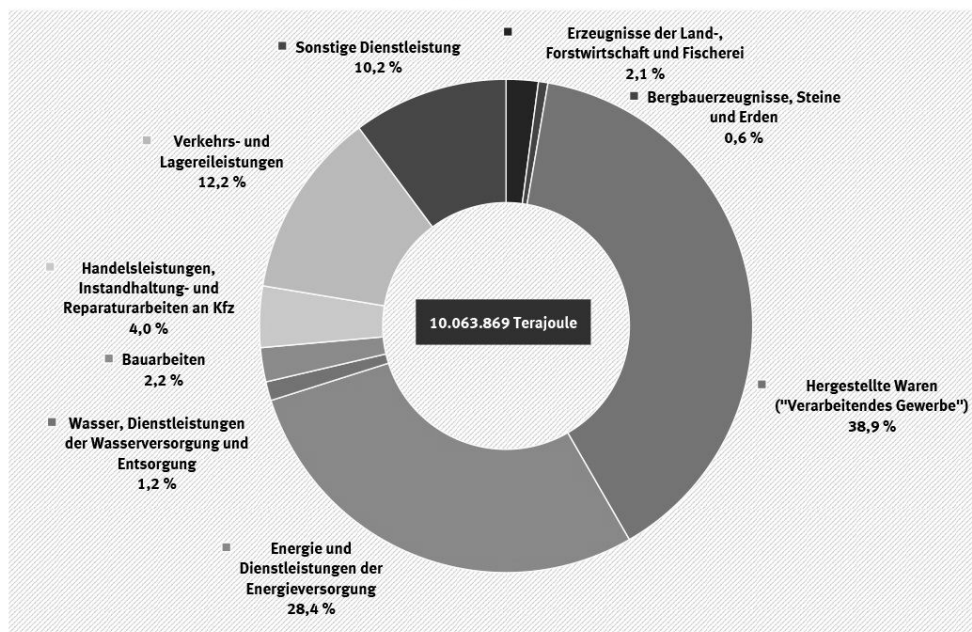


Abbildung A.2.: Energieverbrauch nach Produktionsbereiche 2017 [AGE17]

B. Ausgewählte 3D-Modelle

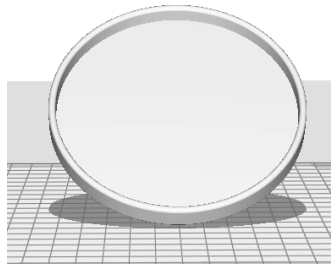


Abbildung B.1.: Modell Deckel, Ursprung [Dizb]

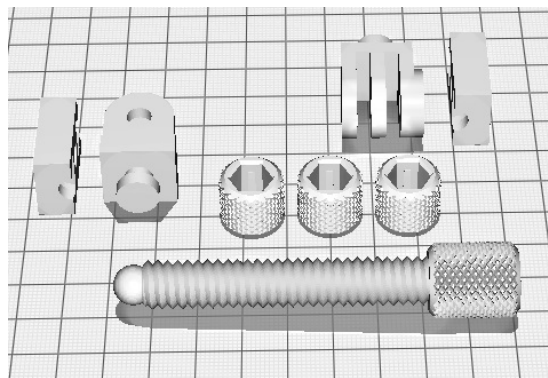


Abbildung B.2.: Modelle von Schrauben und Muttern, Ursprung [Dizb]

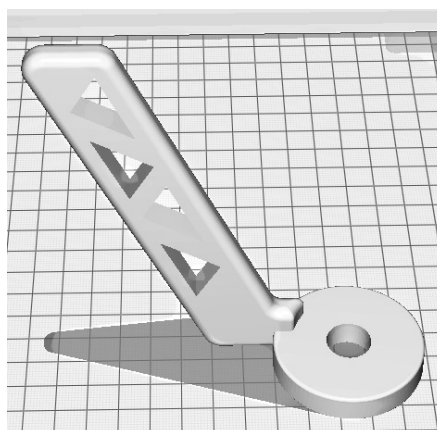


Abbildung B.3.: Modell Träger, Ursprung [Dizb]

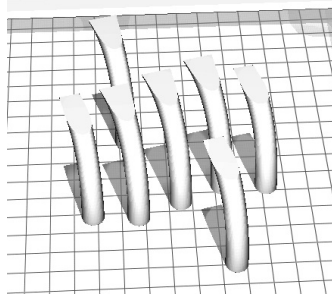


Abbildung B.4.: Modell Pfeiler

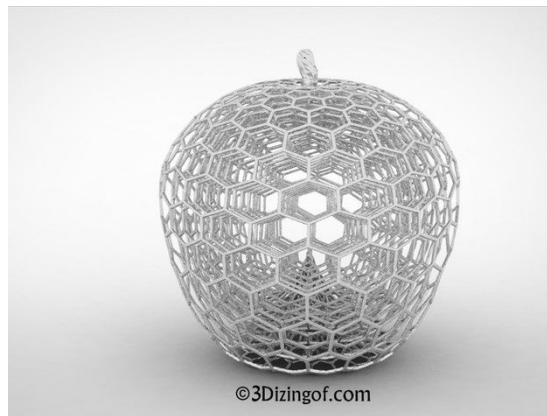


Abbildung B.5.: Modell Apfel [Diza]

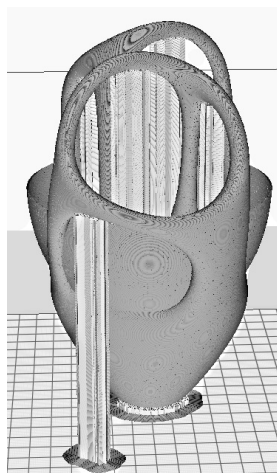


Abbildung B.6.: Modell Vase

C. Cura-Druckprofil

Listing C.1: Auszug aus einem Cura Druckprofil

```
1 ultimaker3_printing_profile_2
  [general]
3 version = 4
  name = printing_profile_2
5 definition = ultimaker3
  [metadata]
7 setting_version = 8
  type = quality_changes
9 quality_type = draft
  [values]
11 adhesion_extruder_nr = 1
  adhesion_type = skirt
13 layer_height = 0.3
  material_bed_temperature = 67
15 prime_tower_enable = False
  ultimaker3_extruder_right_#3_printing_profile_2
17 [general]
  version = 4
19 name = printing_profile_2
  definition = ultimaker3
21 [metadata]
  position = 1
23 setting_version = 8
  type = quality_changes
25 quality_type = draft
  [values]
27 infill_sparse_density = 30
  ultimaker3_extruder_left_#3_printing_profile_2
29 [general]
  version = 4
31 name = printing_profile_2
  definition = ultimaker3
33 [metadata]
  position = 0
35 setting_version = 8
  type = quality_changes
37 quality_type = draft
  [values]
39 cool_fan_speed = 41
  infill_pattern = lines
```

```
41 infill_sparse_density = 50
   material_print_temperature = 256
43 prime_blob_enable = False
   speed_print = 71
45 wall_thickness = 1.4
```


D. Implementierung Datenaufbereitung

Listing D.1: Datenaufbereitung:Suche und Verarbeitung der Druckjobs

```

1 public static int getDateRange(Date startDate, Date >
    endDate) {
2     List<Integer> rangeList = new ArrayList<Integer>();
3     int measuresInRange = 0;
4     Date actualDate = null;
5     System.out.println("--Searching for Datasets between " >
        + startDate + " and " + endDate);
6     Jedis jedis = new Jedis(redisHost, redisPort, >
        redisTimeout);
7     jedis.select(0);
8     try {
9         long dateCount = jedis.sort("Measure_ids",
            new SortingParams().by("Measure|*->Aufnahmezeit >
                ").get("Measure|*->Aufnahmezeit", "#").alpha >
                ),
10            "sortedDates");
11        jedis.del("rangeList");
12        for (int j = 0; j < dateCount; j = j + 2) {
13            actualDate = buildDateEnergy(jedis.lindex(" >
                sortedDates", j));
14            if (isDateBetween(actualDate, startDate, endDate) >
                ) {
15                measuresInRange++;
16                try {
17                    jedis.rpush("rangeList", jedis.lindex(" >
                sortedDates", j + 1));
18                } catch (NumberFormatException e) {
19                    System.out.println("Unparseable Index");
20                }
21            } else {
22            }
23        }
24    }
25    System.out.println("--Datasets within Data Range: " >
        + measuresInRange);
26    resultlist = jedis.lrange("rangeList", 0, -1);
27    System.out.println("--ID RangeList Redis(resultlist >
        ): " + resultlist);
28    jedis.del("sortedDates");
29 } catch (JedisException e) {
    if (null != jedis) {

```

```

31     System.out.println("——getDateRange:Redis␣>
        Exception");
        jedis.close();
33     jedis = null;
    }
35 } finally {
    if (null != jedis)
37     jedis.close();
    }
39     return measuresInRange;
}
41 public static PrintjobLabeld getSumSensor(Date ␣
    startDate, Date endDate, String sensor, Printjob ␣
    printjob) {
    List<Integer> rangeList = new ArrayList<Integer>();
43     int measuresInRange = 0;
    Float SensorSum = 0f;
45     String SensorValue = "";
    Date actualDate = null;
47     System.out.println("—Searching␣for␣Datasets␣between␣"␣
        + startDate + "␣and␣" + endDate);
    Jedis jedis = new Jedis(redisHost, redisPort, ␣
        redisTimeout);
49     jedis.select(0);
    try {
51         long dateCount = jedis.sort("Measure_ids", new ␣
            SortingParams().by("Measure|*→Aufnahmezeit")
                .get("Measure|*→Aufnahmezeit").get("Measure␣
                    |*→Scheinleistung␣(W)", "#").alpha(), "␣
                    sortedDates");
53         jedis.del("rangeList");
        for (int j = 0; j < dateCount; j = j + 3) {
55             actualDate = buildDateEnergy(jedis.lindex("␣
                sortedDates", j));
            if (isDateBetween(actualDate, startDate, endDate)␣
                ) {
57                 measuresInRange++;
                SensorValue = jedis.lindex("sortedDates", j + ␣
                    1);
59                 SensorValue = SensorValue.replace(",", ".");
                SensorSum = SensorSum + Float.parseFloat(␣
                    SensorValue);
61                 try {
                    jedis.rpush("rangeList", jedis.lindex("␣
                        sortedDates", j + 1));
63                 } catch (NumberFormatException e) {
                    System.out.println("Unparseable␣Index");
65                 }
            }
        }
    }
}

```

```

        } else {
67     }
    }
69     System.out.println("—Datasets within Data Range: " +
        + measuresInRange);
    System.out.println("—SensorSum: " + SensorSum);
71     jedis.del("sortedDates");
} catch (JedisException e) {
73     if (null != jedis) {
        jedis.close();
75         jedis = null;
    }
77 } finally {
    if (null != jedis)
79     jedis.close();
}
81 PrintjobLabeld pjob = new PrintjobLabeld(printjob);
pjob.energy = Float.toString(SensorSum);
83 pjob.datasets = Integer.toString(measuresInRange);
return pjob;
85 }

```

Listing D.2: Datenaufbereitung:Erweiterung der Datenbasis

```

1 public List<String[]> expandData(PrintjobLabeld pjob,
    printjob, int amount) {
    List<String[]> dataLines = new ArrayList<>();
3     dataLines.add(
        new String[] { printjob.id, printjob.name + "_0",
            printjob.datetime_started, printjob.datetime_
                finished,
5            printjob.energy, printjob.datasets, printjob.time_
                elapsed, printjob.time_estimated });
    for (int i = 1; i <= amount; i++) {
7        float test = Float.parseFloat(printjob.energy);
        String nenergy = Float
9            .toString(gaussFloat(test, (int) Math.round(
                Float.parseFloat(printjob.energy) * 0.02)));
        String ndatasets = Float.toString(gaussFloat(Float.
            parseFloat(printjob.datasets),
11            (int) Math.round(Integer.parseInt(printjob.
                datasets) * 0.02)));
        String ntime_elapsed = Float.toString(gaussFloat(
            Float.parseFloat(printjob.time_elapsed),
13            (int) Math.round(Integer.parseInt(printjob.
                time_elapsed) * 0.02)));
        String ntime_estimated = Float.toString(gaussFloat(
            Float.parseFloat(printjob.time_estimated),
15            (int) Math.round(Integer.parseInt(printjob.

```

```

        time_estimated) * 0.02)));
    dataLines.add(new String[] { printjob.id, printjob.name + "_" + i, printjob.datetime_started,
17     printjob.datetime_finished, nenergy, ndatasets,
        ntime_elapsed, ntime_estimated });
    }
19     return dataLines;
    }
21     public float gaussFloat(float base, int range, int
        mrange) {
        Random r = new Random();
23     boolean valok = false;
        float gauss;
25     do {
        double val = r.nextGaussian() * range + base;
27     gauss = Math.round(val);
        if (gauss < gauss + mrange || gauss < gauss -
            mrange) {
29     valok = true;
        }
31     } while (!valok);
        return gauss;
33     }
    public float gaussFloat(float base, int range) {
35     Random r = new Random();
        float gauss;
37     double val = r.nextGaussian() * range + base;
        gauss = Math.round(val);
39     return gauss;

```

Listing D.3: Datenaufbereitung: Druckjobklasse

```

1 package jedis1;
  public class PrintjobLabeld {
3 /*     {
        "datetime_cleaned": "1970-01-26T11:52:43",
5         "datetime_finished": "1970-01-26T11:49:35",
        "datetime_started": "1970-01-26T10:13:45",
7         "name": "UM3_W126",
        "reprint_original_uuid": null,
9         "result": "Finished",
        "source": "WEB_API/Unknown/Cura connect",
11        "time_elapsed": 5573,
        "time_estimated": 5582,
13        "time_total": 0,
        "uuid": "4a1919c0-83bb-4c8d-95b7-62ecb625dd30"
15    }*/
    String datetime_cleaned;
17    String datetime_finished;

```

```
String datetime_started;
19 String name;
String reprint_original_uuid;
21 String result;
String source;
23 String time_elapsed;
String time_estimated;
25 String time_total;
String uuid;
27 String id;

29 String energy;
String datasets;
31 public PrintjobLabeld(Printjob parent) {
    datetime_cleaned=parent.datetime_cleaned;
33    datetime_finished=parent.datetime_finished;
    datetime_started=parent.datetime_started;
35    name=parent.name;
    reprint_original_uuid=parent.reprint_original_uuid;
37    result=parent.result;
    source=parent.source;
39    time_elapsed=parent.time_elapsed;
    time_estimated=parent.time_estimated;
41    time_total=parent.time_total;
    uuid=parent.uuid;
43    id=parent.id;
    }
45 }
```


E. Implementierung Vorhersage

Listing E.1: Vohersage Evo4

```
1 import pandas
  from keras.models import Sequential
3 from keras.layers import Dense
  from keras.wrappers.scikit_learn import KerasRegressor
5 from keras import backend as K
  from sklearn.model_selection import cross_val_score
7 from sklearn.model_selection import KFold
  from sklearn.preprocessing import StandardScaler
9 from sklearn.pipeline import Pipeline
  import numpy
11 import time
  start_time = time.time()
13 dataframe = pandas.read_csv("labelddata_77.csv", header=
    =1)
  dataset = dataframe.values
15 X = dataset[:,6]
  Y = dataset[:,4]
17 def baseline_model():
  # create model
19   model = Sequential()
  model.add(Dense(3, input_dim=1,
    kernel_initializer='normal', activation='relu'))
21   model.add(Dense(1, kernel_initializer='normal'))
  # Compile model
23   model.compile(loss='mean_squared_error',
    optimizer='adam')
  return model
25 seed = 7
  numpy.random.seed(seed)
27 estimator = KerasRegressor(build_fn=baseline_model,
    epochs=100, batch_size=5, verbose=1)
  kfold = KFold(n_splits=10, random_state=seed)
29 results = cross_val_score(estimator, X, Y, cv=kfold)
  print("Results: %0.2f %0.2f MSE" % (results.mean(),
    results.std()))
31 end_time = time.time()
  print("%0.10f seconds" % (end_time - start_time))
```


F. Anfrage Vorhersage

Listing F.1: Auszug aus einem Energy Prediction Request:Singlepredict

```
{
2  "reqtype": "singlepredict",
   "version": 1.1,
4  "reqDate": "2019-08-25T00:00:00.000Z",
   "debug": false,
6  "prediction": {
     "type": "kummulative",
8     "": {}
   },
10 "machine": {
     "id": 009857,
12     "name": "milling_machine_12",
     "registered": true,
14     "machinetool": [
         {
16         "id": 004654,
            "name": "fast_mill",
18         "diameter": 1.2
         },
20         {
            "id": 005345,
22         "name": "wide_mill",
            "diameter": 3.1
24         }
     ]
26 },
   "task": {
28     "material": {
        "id": 006734,
30         "name": "steel"
     },
32     "model": {
        "type": "g-code",
34         "name": "spout_12",
        "data": "file",
36         "filename": "spout_12.g-code"
     }
38 }
}
```