

Reihe 1

Konstruktions-
technik/
Maschinen-
elemente

Nr. 449

Dipl.-Ing. Thilo Breitsprecher,
Zirndorf

Entwicklung eines selbstlernenden Assistenzsystems zur automatischen Akquisition von konstruktions- relevantem Fertigungs- wissen

Lehrstuhl für

Konstruktionstechnik

Friedrich-Alexander-Universität Erlangen-Nürnberg
Prof. Dr.-Ing. Sandro Wartzack



Entwicklung eines selbstlernenden Assistenzsystems zur automatischen Akquisition von konstruktionsrelevantem Fertigungswissen

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg
zur
Erlangung des Doktorgrades Dr.-Ing.
vorgelegt von
Thilo Breitsprecher
aus Frankfurt (Oder)

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander Universität Erlangen-Nürnberg
Tag der mündlichen Prüfung: 29.10.2018

Vorsitzender des Promotionsorgans: Prof. Dr.-Ing. Reinhard Lerch

Gutachter: Prof. Dr.-Ing. Sandro Wartzack
Prof. Dr.-Ing. Hansgeorg Binz

Fortschritt-Berichte VDI

Reihe 1

Konstruktionstechnik/
Maschinenelemente

Dipl.-Ing. Thilo Breitsprecher,
Zirndorf

Nr. 449

Entwicklung eines
selbstlernenden
Assistenzsystems zur
automatischen Akquisition
von konstruktions-
relevantem Fertigungs-
wissen

Lehrstuhl für

Konstruktionstechnik

Friedrich-Alexander-Universität Erlangen-Nürnberg
Prof. Dr.-Ing. Sandro Wartack



Breitsprecher, Thilo

Entwicklung eines selbstlernenden Assistenzsystems zur automatischen Akquisition von konstruktionsrelevantem Fertigungswissen

Fortschr.-Ber. VDI Reihe 1 Nr. 449 . Düsseldorf: VDI Verlag 2019.

192 Seiten, 93 Bilder, 14 Tabellen.

ISBN 978-3-18344901-9 ISSN 0178-949X,

€ 67,00/VDI-Mitgliederpreis € 60,30.

Für die Dokumentation: Design for X – Fertigungsgerechtes Konstruieren – Blechmassivumformung – Data Mining – Selbstlernen – Automatische Wissensakquisition – Assistenzsysteme

Die vorliegende Arbeit wendet sich an Wissenschaftler und Ingenieure, die an der Schnittstelle zwischen der klassischen Konstruktionstechnik und der Wissensentdeckung in Datenbanken tätig sind. Sie befasst sich mit dem Einsatz von Methoden des Data Minings zur automatischen Akquisition von konstruktionsrelevantem Fertigungswissen. Dieses benötigt der Produktentwickler zur Beurteilung der Fertigungsgerechtigkeit seiner Bauteilentwürfe. Im Fokus stehen Fertigungsverfahren, die sich in einem frühen Entwicklungsstadium befinden. Als exemplarischer Prozess wird in der Arbeit die Blechmassivumformung betrachtet. Grundlage für die automatische Wissensakquisition ist der in der Arbeit entwickelte Selbstlernprozesses. Dieser nutzt Data Mining Methoden, um aus den Daten der Fertigungsprozessentwicklung das relevante Wissen ohne Benutzereingriff zu akquirieren. Der Selbstlernprozess ist in ein prototypisches Konstruktionsassistenzsystem integriert.

Bibliographische Information der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Daten sind im Internet unter www.dnb.de abrufbar.

Bibliographic information published by the Deutsche Bibliothek

(German National Library)

The Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliographie (German National Bibliography); detailed bibliographic data is available via Internet at www.dnb.de.

Vorwort und Danksagung

Die folgenden Zeilen möchte ich nutzen, um mich bei den Menschen zu bedanken, die zum erfolgreichen Abschluss dieser Arbeit beigetragen haben. Sie ist das Ergebnis meiner Tätigkeit als wissenschaftlicher Assistent am Lehrstuhl für Konstruktionstechnik (KTmfk) der Friedrich-Alexander Universität Erlangen-Nürnberg.

Mein besonderer Dank gilt Prof. Dr.-Ing. Sandro Wartzack. Durch die gleichermaßen fordernde wie fördernde Betreuung meines Promotionsvorhabens hat er maßgeblich zum erfolgreichen Abschluss dieser Arbeit beigetragen.

Herrn Prof. Dr.-Ing. Hansgeorg Binz, Leiter des Instituts für Konstruktionstechnik und Technisches Design (IKTD) an der TU Stuttgart, danke ich herzlich für sein Interesse an dieser Arbeit und die Anfertigung des Zweitgutachtens.

Des Weiteren danke ich Herrn Prof. Dr. Lutz Schröder für seine Bereitschaft die Funktion des fachfremden Prüfers zu übernehmen. Der fachliche Austausch mit ihm und mit Herrn Prof. Dr. Michael Kohlhase zum Thema „Wissensbasiertes Konstruieren“ haben mir neue Blickwinkel auf diese vielschichtige Thematik ermöglicht.

Auch Herrn Prof. Dr.-Ing. Tino Hausotte sei an dieser Stelle für die gute Zusammenarbeit im Sonderforschungsbereich Transregio 73 und für die Leitung der Disputation als Vorsitzender der Prüfungskommission gedankt.

Den ehemaligen Kolleginnen und Kollegen des KTmfk danke ich für die gemeinsame Zeit, die fachlichen Kritiken und Anregungen, die Unterstützung bei anderweitigen Aufgaben und vor allem das motivierende Miteinander. *Alle* haben auf verschiedene Weise zum Abschluss dieser Arbeit beigetragen. Besonders erwähnen möchte ich hier dennoch Sebastian Röhner („So jung komm’mer nix mehr zamml!“), Christof Küstner („Tunnel, Tunnel!“) und Michael Walter („Bilo!“). Den Mitarbeitern des IKTD Martin Kratzer, Daniel Roth und Alexander Laukemann sei zudem für den freundschaftlichen und fachlichen Austausch über Universitätsgrenzen hinweg gedankt.

Einen wichtigen Anteil an dieser Arbeit haben auch zahlreiche Studierende, denn im Rahmen von Bachelor-, Studien- und Masterarbeiten wurde wertvolle Unterstützung geleistet. Insbesondere Jürgen Mitsch, Sina Wipfel, Nico Meintker, Christopher Sauer, Sebastian Lukas, Florian Kowohl, Christian Dingfelder, Niklas Ewingmann und Christian Sperber sei an dieser Stelle herzlich gedankt.

Meinen Eltern Beate und Uwe sowie meinem Bruder David danke ich für das beigebrachte Rüstzeug um ein solches Vorhaben zu meistern und die familiäre Unterstützung und Motivation. Danken möchte ich auch meinen Schwiegereltern und deren Kindern, meinen

Freunden aus der Abi-Clique, den Kletterkameradinnen und -kameraden des DAV Schwabach und der Bergwacht Fürth sowie dem Kirchenchor Unterreichenbach. Ihr habt mich regelmäßig auf andere Gedanken gebracht und den Promotionsstress vergessen lassen.

Letztendlich gilt der größte Dank jedoch meiner Frau Bianca, die mir während der gesamten Entstehung der Arbeit mehr als nur viel Verständnis entgegengebracht und mich in jeder Hinsicht motiviert und unterstützt hat. Unsere Kinder Martha und Jonathan können stolz sein, eine solche Mutter zu haben :*

Zirndorf im November 2018

Thilo Breitsprecher

Inhaltsverzeichnis

Nomenklatur	VIII
Abstract	XI
1 Einleitung	1
1.1 Problemstellung	3
1.2 Zielsetzungen und Aufbau der Arbeit	4
2 Wissenschaftliche Grundlagen und Stand der Forschung	6
2.1 Konstruieren als wissensintensive Tätigkeit	6
2.1.1 Der Konstruktionsprozess	7
2.1.2 Wissen im Konstruktionsprozess	9
2.1.3 Wissensstrukturierung	11
2.1.4 Wissensakquisition	14
2.2 Wissensbasierte Systeme in der Produktentwicklung	18
2.2.1 Wissensbasiertes Konstruieren	19
2.2.2 Die Technologie der wissensbasierten Systeme	23
2.2.3 Beispiele wissensbasierter Konstruktionssysteme	26
2.3 Analyse der Fertigungsgerechtigkeit	34
2.3.1 Fertigungsgerechtigkeit im Kontext des Design for X	34
2.3.2 Geometriebasierte Design for Manufacture Analysen	36
2.3.3 Rechnerunterstützung durch Prozesssimulation	39
2.4 Wissensentdeckung in Datenbanken - KDD	41
2.4.1 Der KDD Prozess nach Fayyad	42
2.4.2 Darstellung der KDD-Prozessschritte	43
2.5 Einsatz von KDD im Produktentstehungsprozess	49
2.5.1 KDD in der Anforderungsanalyse	49
2.5.2 Wissensbasierte Schichtentwicklung	51
2.5.3 Stochastische Analysen in der Feinschneid-Methodenplanung	53

3 Die Blechmassivumformung als Anwendungskontext	55
3.1 Die Blechmassivumformung aus Sicht der Fertigungstechnologie	55
3.2 Bedeutung der Blechmassivumformung für die Produktentwicklung	58
3.3 Klassifikation von Nebenformelementen der Blechmassivumformung	61
4 Ableiten des Handlungsbedarfs	66
4.1 Fazit zum Stand der Forschung	66
4.2 Anforderungen an ein verbessertes Konzept	68
5 Der Selbstlernprozess	69
5.1 Vorüberlegung zum Begriff Selbstlernen	69
5.1.1 Menschliches Lernen	70
5.1.2 Übertragung auf die Problemstellung	72
5.2 Statistisch robuste Performanzbestimmung	76
5.3 Überwachtes Lernen und Optimieren der Metamodelle	79
5.4 Auswahl des bestgeeigneten Metamodells	82
5.4.1 Analyse der Varianzen	86
5.4.2 Durchführung des post-hoc Tests	88
6 Das selbstlernende Assistenzsystem Slassy	89
6.1 Integration in das Arbeitsumfeld der Benutzer	89
6.2 Die Komponenten und ihr Zusammenwirken	92
6.3 Die Selbstlernkomponente	94
6.3.1 Auswahl der eingesetzten Metamodelle	95
6.3.2 Vorverarbeitung der Daten	101
6.3.3 Automatische Attributselektion	103
6.3.4 Umsetzung des ROPE-Prozesses	104
6.3.5 Selektion des bestgeeigneten Metamodells	109
6.3.6 Weitere Umsetzung des Selbstlernprozesses	110
6.4 Das Synthesewerkzeug	112
6.4.1 Bereitstellung der Hauptformelemente	112
6.4.2 Bereitstellung der Nebenformelemente	114
6.4.3 Erzeugung eines Bauteilentwurfs	116
6.5 Das Analysewerkzeug	117
6.5.1 Verarbeitung des Konstruktionswissens	118
6.5.2 Ergebnisdarstellung und Erklärungskomponente	120

6.6	Das Produkt- und Prozessdatenmodell	121
6.6.1	Die multidimensionale Daten- und Wissensbasis	121
6.6.2	Strukturierung der Produkt- und Prozessdaten	123
6.6.3	Operative Integration der Produkt- und Prozessdaten	128
7	Anwendungsbeispiel: Tiefziehen-Querfließpressen	131
7.1	Integration des Bauteils in Slassy	131
7.1.1	Aufbau des Hauptformelements <i>Napf</i>	132
7.1.2	Aufbau des Nebenformelements <i>Sperrverzahnung</i>	133
7.1.3	Synthese des Bauteilmodells	134
7.2	Erhebung der Fertigungsdaten	135
7.2.1	Auswahl der relevanten Parameter und Entwicklung des Versuchsplans	136
7.2.2	Durchführung der Parameterstudie und Auswertung	139
7.3	Verarbeitung der Daten in der Selbstlernkomponente	141
7.3.1	Import und Vorbehandlung der Daten	141
7.3.2	Automatische Attributselektion	142
7.3.3	Durchlaufen des ROPE-Prozesses	143
7.3.4	Modellauswahl und -evaluierung	145
7.4	Analyse des Bauteilentwurfs	151
8	Zusammenfassung und Ausblick	154
Anhang		157
Literaturverzeichnis		164

Nomenklatur

Abkürzungen

AI	Artificial Intelligence
ANOVA	Analysis of Variances, Varianzanalyse
BMBF	Bundesministerium für Bildung und Forschung
BMU	Blechmassivumformung
CAD	Computer-aided Design
CAE	Computer-aided Engineering
CAM	Computer-aided Manufacturing
CAPP	Computer Aided Process Planning
CNC	Computer Numeric Control
COD	Coefficient of Determination, lineares Bestimmtheitsmaß
CoP	Coefficient of Prognosis
CSG	Construktive Solid Geometry
DfX	Design for X
DfM	Design for Manufacture, Fertigungsgerechte Gestaltung
DGL	Differentialgleichung
DIN	Deutsche Institut für Normung e. V.
DoE	Design of Experiments
ECO	Engineering Change Order, Änderungsanträge
EDV	Elektronische Datenverarbeitung
EDM	Engineering Data Management
FEM	Finite Elemente Methode
FGK	Fertigungsgerechtes Konstruieren
HKA	Hauptkomponentenanalyse
HSD	Honest Significance Difference Test (nach TUKEY)
IT	Informationstechnologie
JT	Jupiter Tesselation
KBE	Knowledge-based Engineering
KDD	Knowledge Discovery in Databases
KE	Knowledge Engineering
KI	Künstliche Intelligenz
KNN	Künstliches Neuronales Netz
KS	Konstruktionssystem
KSmfk	Konstruktionssystem methodisches und fertigungsgerechtes Konstruieren
LHS	Latin Hypercube Sampling

OLAP	Online Analytical Processing
ProKon	Proaktive Unterstützung von Konstruktionsprozessen durch Softwareagentensysteme
PDM	Produktdatenmanagement
PEP	Produktentstehungsprozess
PLM	Produktlebenszyklusmanagement
PPDM	Produkt- und Prozessdatenmodell
ROPE	Robust Optimization and Performance Estimation
RSM	Responce Surface Methode
RMSE	Root mean squared error, Wurzel des mittleren Fehlerquadrats
SFB/TR 73	Sonderforschungsbereich Transregio 73
SIT	Short Intervall Technology
Slassy	Selbstlernendes Assistenzsystem
SQL	Structured Query Language, relationale Datenbankbeschreibung
STEP	Standard for the exchange of product model data
SVM	Support Vector Machine, mathematisches Verfahren zur Mustererkennung
TRIP	Transformation Induced Plasticity
UDF	User defined feature, benutzerdefiniertes Konstruktionselement
VDI	Verein Deutscher Ingenieure
WBS	Wissensbasiertes System
WEKA	Waikato Environment for Knowledge Analysis
WOP	Werkstattorientierte NC-Programmierung
XPS	Expertensystem

Formelzeichen und Indizes

A	-	Anzahl der Attribute in einem Datensatz
D	-	Bezeichnung für einen Datensatz
D_k	-	k -ter Subdatensatz von D in einer Kreuzvalidierung
\bar{X}	-	Mittelwert der Einträge in einem Performanzvektor
F	-	F-Wert einer ANOVA
$F_{u_{max}}$	kN	maximale Umformkraft
\bar{G}	-	Gesamtmittelwert aller Performanzvektoren
H_0	-	Nullhypothese, die beobachteten Mittelwerte sind identisch
H_1	-	Alternativhypothese
M	-	Anzahl der Metamodelle die im Selbstlernprozess verglichen werden
M_i	-	i-te Metamodell das der ROPE-Prozess ausgibt
N	-	Anzahl der Datentupel in einem Datensatz
a_i	-	Tatsächlicher (engl. <i>actual</i>) Wert eines Labels
\bar{a}	-	Mittelwert der Label im Trainingsdatensatz
c	-	Kontaktverhältnis zwischen Werkstoff und Umformwerkzeug
d_f	-	Freiheitsgrad (engl. <i>degree of freedom</i>) in der Statistik
e_i	-	Fehler (engl. <i>error</i>) zwischen a_i und p_i

i	-	allgemeiner Laufindex
k	-	Anzahl der Teildatensätze in einer Kreuzvalidierung
k_H	-	Heuversfaktor
l	-	Anzahl der Linearkombinationen einer HKA
m	-	Reibfaktor, Maß für die Reibung zwischen zwei Körpern
n	-	Anzahl der Wiederholungen einer k -fachen Kreuzvalidierung
p -Wert	-	Wahrscheinlichkeit der beobachteten oder einer extremeren Prüfgröße
p_i	-	Vorhergesagter (engl. predicted) Wert eines Labels
q	-	Prüfgröße im post-hoc Test nach TUKEY
t	-	Prüfgröße für statistischen t-Test
t_{krit}	-	kritische Grenze für die Prüfgröße t , bei $t < t_{krit}$ gilt i. A. H_0
$t_{\Delta,max}$	mm	maximale Blechausdünning
\bar{x}_{M_i}	-	Mittelwert der geschätzten Performanz eines Metamodells
z	-	Anzahl der Einzelwerte in einem Performanzvektor
α	-	Signifikanzniveau eines statistischen Hypothesentests
β_{max}	-	Grenzziehverhältnis zur Beurteilung der Tiefzieheignung
φ_{max}	-	Umformgrad
φ_{vG}	-	Grenzumformgrad

Abstract

Companies within different industries have to face several key challenges, among which the demand for shortening the time-to-market whilst decreasing costs and the strong individuality and dynamic of globalized markets are highlighted. In order to overcome those challenges companies not only have to optimize their products but also the underlying product *development* and *manufacturing* processes. The objective from engineering design point of view is the exchange of design-relevant manufacturing knowledge between both disciplines as early as possible. Especially for newly emerging manufacturing technologies common knowledge acquisition methods to assist such knowledge exchange processes fail.

This thesis proposes an approach for a self-learning engineering **assistance system** (referred to as *Slassy*) for the automatic acquisition of design-relevant manufacturing knowledge. Such knowledge is represented in terms of so-called metamodels. In the context of this thesis a metamodel is capable of predicting manufacturing process related parameter such as forming force, equivalent plastic strain, sheet thinning (and thickening) or forming tool cavity filling. The prediction depends on design related parameters such as lengths, widths, heights or angles of geometric design features. Design features that are considered in this work are for example toothings, straps or tapets. The self-learning process incorporated in Slassy's knowledge acquisition component is based on two core items: the robust optimization and performance estimation of different metamodels and a two-step inference mechanism. The former is referred to as ROPE process and ensures that the prediction quality of all metamodels is robust and reliably estimated. It facilitates methods from the field of knowledge discovery in databases (KDD). The latter enables Slassy to chose the most suitable metamodel, that is, the prediction quality of the chosen metamodel is significantly better than the remaining. This metamodel is incorporated in Slassy's knowledge base. Hence, Slassy can assist design engineers during the analysis of design concepts in terms of design-for-manufacture, an important aspect in the field of design for X (DfX).

The emerging manufacturing technology *sheet-bulk metal forming* (SMBF) creates the background for the concluding use case. Data from an exemplary SMBF process development is analysed by means of KDD methods and a set of metamodels is derived with the developed ROPE process. The selection of the most suitable metamodel by the prototypically implemented Slassy is comprehended. Afterwards the prediction of relevant manufacturing process parameters with Slassy is shown. By enabling the prediction of these process parameters the number of time-consuming iterations between product and manufacturing process development can be reduced.

1 Einleitung

Produzierende Unternehmen unterschiedlicher Industriebranchen (z. B. Maschinen- und Anlagenbau, Automotive, Luftfahrt) werden von drei wesentlichen Trends geleitet. Dies sind die fortschreitende Globalisierung der Märkte, der steigende Innovations- und Kostendruck sowie eine stärkere Individualität und Dynamik der weltweiten Absatzregionen. SCHUH ET AL. gehen davon aus, dass dieser Trend bestehen bleiben und sich weiter verstärken wird [SGS+07]. Die Unternehmen reagieren auf diese Trends und haben erkannt, dass nicht nur die Produkte, sondern auch die Produktentstehungsprozesse optimiert werden müssen. Bild 1 zeigt das Zusammenspiel der Bereiche „Entwicklung und Konstruktion“ sowie „Fertigung und Montage“, die hier exemplarisch aus dem Produktentstehungsprozess herausgegriffen wurden.

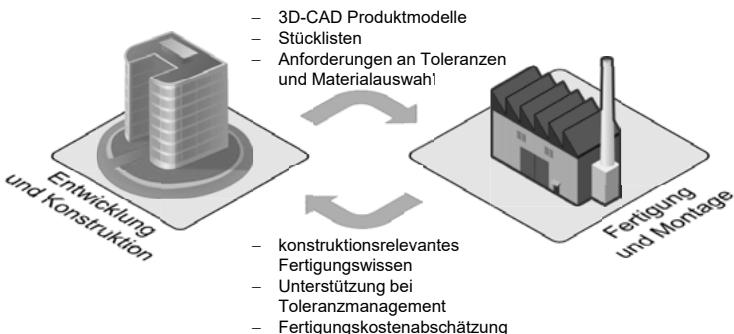


Bild 1: Zusammenspiel der Bereiche „Entwicklung und Konstruktion“ sowie „Fertigung und Montage“ im Produktentstehungsprozess.

Von den skizzierten Trends sind die Entwicklungs- und Konstruktionsabteilungen, in denen die Kernprozesse aller ingenieurmäßigen Planungstätigkeiten ablaufen und die in den frühen Phasen des Produktentstehungsprozesses aktiv sind, in besonders hohem Maße betroffen. Hintergrund ist, dass diese Abteilungen auf der einen Seite 60 bis 80 % der variablen Produktselbstkosten festlegen, auf der anderen Seite jedoch nur ca. 5 % dieser Kosten verursachen [EM13; VDM01]. Die Ingenieure und Konstrukteure bewegen sich dabei stets im sogenannten „Magischen Dreieck“ aus Entwicklungskosten, Produktqualität und Entwicklungszeit [HKL+15]. Unterstützung erfahren sie im Allgemeinen durch Methoden und Werkzeuge der rechnerunterstützten Produktentwicklung [Vaj09; Lin09].

Mit Blick auf die späteren Phasen des Produktentstehungsprozesses tritt die Fertigung in den Vordergrund. In dieser Phase werden zwar rund ein Drittel der entstehenden Kosten verursacht, jedoch weniger als zehn Prozent festgelegt [EM13; VDM01]. Optimierungen im Bereich der Fertigung ergeben sich zum einen durch neue organisatorische Konzepte (z. B. Kanban [Eve02], SIT [KS11] oder Lean Production [SW03]), zum anderen durch die Neu- oder Weiterentwicklung von Fertigungstechnologien, sowohl in den Unternehmen, als auch in Forschungseinrichtungen¹. Durch die Entwicklung neuartiger Fertigungsprozesse sorgt das produzierende Gewerbe für eine stetige Reduzierung der zur Produktion von Gütern notwendigen Ressourcen [BJS⁺11; ES05].

Neue Fertigungsverfahren müssen jedoch auch klar erkennbare Vorteile für die Entwicklung und Konstruktion in Form von erweiterten oder neuen Gestaltungsmöglichkeiten, engeren Toleranzfeldern bei mindestens gleichen Kosten oder höheren Oberflächengüten bieten. Den Herausforderungen die sich durch die hohe Kostenverantwortung auf der einen Seite (Konstruktion) und der hohen Kostenverursachung (Fertigung) auf der anderen Seite ergeben kann u. a. mit Methoden des fertigungsgerechten Konstruierens (FGK, engl. Design for Manufacture (DfM)) begegnet werden. Ziel ist eine möglichst enge Verzahnung der beiden Bereiche und ein frühzeitiger Austausch sowie eine Bereitstellung von *konstruktionsrelevantem Fertigungswissen*, das die Konstrukteure zu Design for Manufacture Analysen ihrer Entwürfe befähigt. *Frühzeitig* meint hier sowohl früh im Konstruktionsprozess als auch früh in der Fertigungsprozessentwicklung (s. Bild 1).

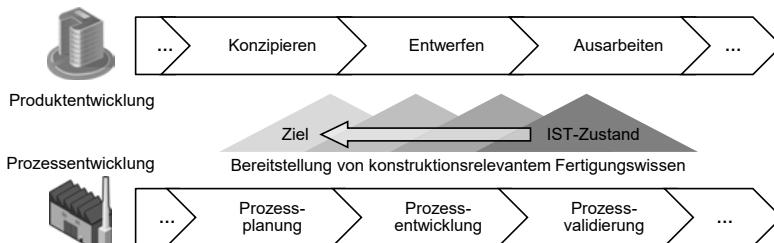


Bild 2: Je früher Wissensaustausch zwischen Produkt- und Prozessentwicklung erfolgt, desto besser für den gesamten Produktentstehungsprozess.

Austausch und Bereitstellung von Wissen im Unternehmen sind klassische Aufgabengebiete des Wissensmanagements [Non07]. Dass diese Domäne immer mehr an Bedeutung gewinnt, wird u. a. durch die Forderung nach aktivem Wissensmanagement in der neuen Version der Norm ISO 9001:2015 deutlich [Hin15; NBS16]. Aktivitäten des Wissensmanagements im Umfeld der Produktentwicklung werden unter der Bezeichnung „Wissensbasiertes Konstruieren“ (KBE, engl. Knowledge Based Engineering) zusammengefasst. In diesen Kontext ordnet sich auch die Problemstellung der vorliegenden Arbeit ein.

¹ Als Beispiel sei hier der Sonderforschungsbereich Transregio 73 (SFB/TR 73) genannt. Kapitel 3 enthält weitere Ausführungen.

1.1 Problemstellung

Wissensbasiertes Konstruieren (KBE, engl. Knowledge Based Engineering) bezeichnet im Allgemeinen die gedankliche Durchdringung eines Konstruktionsprozesses, so dass dieser, zumindest teilweise, rechnerunterstützt umgesetzt werden kann [Rud98]. KBE-Anwendungen weisen im industriellen Umfeld ein breites Spektrum auf und reichen von einfachen, CAD-internen Makros über angepasste Lösungen auf Basis von KBE-Modulen bis hin zu kundenspezifischen Produktkonfiguratoren und Design Automation Anwendungen [VDI15]. Im akademischen Umfeld der Produktentwicklung liegt der Fokus der Forschungsarbeiten auf wissensbasierten Konstruktions- bzw. Assistenzsystemen. Beispiele hierzu werden in Abschnitt 2.2 dargestellt.

Die Entwicklung, der Betrieb und die Benutzung wissensbasierter (Assistenz)systeme werden unter der Bezeichnung „Knowledge Engineering“ (KE) zusammengefasst und ist als Prozess zu verstehen, mit dem Ziel, Problemlösungswissen von einer Wissensquelle in eine Anwendungssoftware zu überführen [SBF98; Sto01]. Die bisher entwickelten KE-Prozessmodelle (z. B. Semma, MOKA, CommonKADS, MIKE) unterscheiden sich in den grundlegenden Konzepten teilweise sehr, zeigen jedoch in einem Aspekt hohe Übereinstimmung: Die *Wissensakquise* wird als der wichtigste und zugleich kritischste Schritt angesehen. FEIGENBAUM spricht hier von einem „Flaschenhals“ [Fei81]. Die Wissensakquisition, deren Details im Abschnitt 2.1.4 behandelt werden, kann direkt, indirekt oder automatisch erfolgen. Direkte und indirekte Verfahren sind an menschliche Wissensquellen (z. B. Experten, Fachkräfte) gebunden, während automatische Verfahren bisher aus verfügbaren Texten (z. B. Fachbücher, Normen, Tabellenwerke) Wissen extrahieren. Ein „textverständendes“ Programm erfasst ohne Einwirken des Nutzers den Text, analysiert diesen linguistisch und formalisiert ihn in Form von Fakten und Regeln [SK97].

Vor dem Hintergrund der einleitend erwähnten, notwendigen frühen und engen Verzahnung zwischen der Produktentwicklung bzw. Konstruktion und der Fertigung muss ein wissensbasiertes Assistenzsystem den Konstrukteur bei DfM-Analysen seiner Gestaltentwürfe unterstützen. Bisher entwickelte Konstruktions- oder Assistenzsysteme nutzen erfolgreich direkte oder indirekte Wissensakquisitionsmethoden und decken in Form von Insellösungen etablierte Fertigungsverfahren wie beispielsweise Drehen, Blechbiegen, Tiefziehen, Druck- und Spritzguss sowie Punktschweißen ab (s. Abschnitt 2.2.3). Eine problematische Situation ergibt sich jedoch, wenn für neue Fertigungstechnologien oder neuartige Kombinationen von bestehenden Verfahren wissensbasierte DfM-Analysewerkzeuge entwickelt werden sollen. Während bei etablierten Verfahren menschliche und textuelle Wissensquellen vorhanden sind, fehlt es bei neuen Prozessen besonders in den frühen Entwicklungsphasen sowohl an ausreichendem, formalisierbarem und sicherem Expertenwissen über den Prozess als auch an textbasierten oder tabellarischen Wissensquellen. Fällt es Experten i. d. R. schwer, bereits für Sachverhalte, in denen sie über langjährige Erfahrung verfügen, ihr Erfahrungswissen zu formalisieren, so wird dieser Umstand bei neuen Wissensfeldern noch verschärft. Direkte bzw. indirekte Methoden können für die Akquisition von konstruktionsrelevantem

Fertigungswissen vor diesem Hintergrund in frühen Phasen des Produktentstehungsprozesses nicht eingesetzt werden und automatische Methoden rücken in den Fokus.

Die Arbeiten im Bereich der automatischen Wissensakquisition werden im Allgemeinen unter dem Begriff „Knowledge Discovery in Databases“ (KDD, Wissensentdeckung in Datenbanken) zusammengefasst [SK97; Her97; ES00]. Im Rahmen von KDD-Prozessen (s. Abschnitt 2.4) werden Methoden und Werkzeuge aus unterschiedlichen Wissensdomänen wie Data-Mining, maschinelles Lernen und Inferenzstatistik genutzt, um aus Datenbeständen Informationen bzw. Wissen zu extrahieren [CPSK07]. Eine effektive Nutzung von KDD im Umfeld der Konstruktionstechnik setzt voraus, dass dem Nutzer, sprich dem Konstrukteur, diese zugrundeliegenden Theorien und Methoden bekannt sind. Ist dies nicht der Fall und setzt er entsprechende Software ohne Hintergrundwissen ein, kann es zu falschen Ergebnissen, Fehlinterpretationen oder Abwehrhaltungen gegenüber der Benutzung von so genannten „Black-Box“ Programmen kommen. Durch die Nutzung von KDD-Technologien rücken neben den etablierten Wissensquellen auch Datenquellen in den Vordergrund, von denen es entlang des Produktentstehungsprozesses unzählige gibt: Tabellen mit Anforderungen, Sammlungen an Konzeptlösungen, Protokolle, Stücklisten sowie unterschiedliche CAD-, Simulations- und Werkstoffdaten.

Die Lösung dieser Problemstellungen ist von enormer Bedeutung, denn Technologien wie KDD können im Produktentstehungsprozess enorm viel Optimierungspotential erschließen (s. [Sch14b; Har14]) und sind zudem eine Voraussetzung für die Realisierung der Industrie 4.0 Strategie des BMBF [Bau14; TAN15]. Die skizzierten Problemstellungen und Hintergründe sind bei der im Folgenden entwickelten Zielsetzung zu berücksichtigen.

1.2 Zielsetzungen und Aufbau der Arbeit

Für die Lösung der beschriebenen Problemstellung setzt sich diese Arbeit zum Hauptziel, ein wissensbasiertes Assistenzsystem zu entwickeln, das in der Lage ist, mit Methoden der automatischen Wissensakquisition konstruktionsrelevantes Fertigungswissen zu erheben und für Design for Manufacture Analysen zur Verfügung zu stellen.

Wesentliche Randbedingungen ergeben sich aus der Problemstellung, dass sich das jeweilige Fertigungsverfahren in einer frühen Entwicklungsphase befindet. Außerdem sollen die Personen der Zielgruppe, zum Beispiel Betriebsmittelkonstrukteure und Fertigungingenieure, mit den Details zu KDD-Prozessen, -Methoden und -Werkzeugen nicht zusätzlich belastet werden. Das Assistenzsystem muss in der Lage sein, unabhängig vom Nutzer verlässliches Wissen selbstständig zu akquirieren und zu bewerten. Diese zentrale Eigenschaft, die es von bisherigen Assistenzsystemen abgrenzt, wird im weiteren Verlauf der Arbeit als *Selbstlernfähigkeit* bezeichnet. Damit wird ein Beitrag zur Weiterentwicklung wissensbasierter Systeme im Bereich der Konstruktionstechnik geleistet und die Problematik des Flaschenhalses „Wissensakquisition“, der in bisherigen Systemen zur „Alterung“ der Wissensbasis führte, entschärft.

Zu Beginn erfolgt eine Analyse des Einsatzkontextes „Konstruktion“ des Assistenzsystems mit einem Fokus auf den Aspekt *Wissen* und dessen Bedeutung für die Produktentwicklung. Insbesondere die Wissensstrukturierung und die Wissensakquisition werden dabei genauer betrachtet. Das in der vorliegenden Arbeit zu entwickelnde selbstlernende Assistenzsystem basiert auf der Technologie der wissensbasierten Systeme, deren Grundlagen darzustellen und wichtige Begriffe voneinander abzugrenzen sind. Weiterhin werden Beispiele für wissensbasierte Assistenzsysteme aus der jüngeren Forschungsgeschichte vorgestellt, die entweder einen starken Design for Manufacture Bezug aufweisen oder sich insbesondere dem Problem der Wissensakquisition widmen. Ob sich diese Ansätze für die Lösung der vorliegenden Problemstellung eignen, wird anschließend im Kapitel 4 beurteilt. Der Aspekt der Fertigungsgerechtigkeit im Kontext des Design for X wird ebenfalls in einem eigenen Abschnitt behandelt. Das Ziel ist zum einen die Abgrenzung des konstruktionsrelevanten Fertigungswissens zu anderen Wissensarten und die Erarbeitung bisheriger Ansätze zur Beurteilung der Fertigungsgerechtigkeit. Hier stehen sowohl die geometriebasierten Design for Manufacture Analysen als auch die Rechnerunterstützung durch Simulation im Fokus. Abschließend werden die Grundlagen der Wissensentdeckung in Datenbanken zusammengefasst, soweit sie als notwendiges Hintergrundwissen relevant sind. Ausgangspunkt ist das weit verbreitete KDD-Prozessmodell nach FAYYAD [FPSS96]. Das Kapitel schließt mit der Darstellung bisheriger KDD-Anwendungen im Kontext der Produktentstehung.

Kernpunkt der Zielsetzung ist die Akquisition von Design for Manufacture relevantem Wissen über Fertigungsprozesse in frühen Entwicklungsstadien. Als exemplarischer Prozess wird die Blechmassivumformung genutzt, die im Kapitel 3 sowohl aus der Sicht der Fertigung als auch der Konstruktionstechnik dargestellt wird. Da mit der Erforschung der Blechmassivumformung u. a. das Ziel verfolgt wird, bestehende Fertigungsprozesse für komplexe Nebenformelemente zu ersetzen, wurde ein Katalog an Nebenformelementen entwickelt, mit dessen Hilfe das Substitutionspotential der Blechmassivumformung qualitativ beurteilt werden soll. Ausgehend von den erarbeiteten Grundlagen (Kapitel 2) und der Darstellung der Blechmassivumformung (Kapitel 3) erfolgt eine Ableitung des Handlungsbedarfs und die Formulierung von Anforderungen an ein selbstlernendes Assistenzsystem.

Der Entwicklung des Selbstlernprozesses wird ein eigenes Kapitel gewidmet. Zunächst wird der Frage nachgegangen, was unter *Selbstlernen* zu verstehen ist und welche Theorien aus anderen Wissenschaftsdomänen zur Beantwortung dieser Frage beitragen können. Daraus wird der Selbstlernprozess abgeleitet und mit den geeigneten KDD-Methoden zunächst als allgemeingültiges Konzept entwickelt. Die konkrete Umsetzung des Selbstlernprozesses erfolgt im Rahmen der Implementierung des Assistenzsystems, das mit Synthese- bzw. Analysetool und einem Produkt- und Prozessdatenmodell an das Konzept des Konstruktionssystems *mfk* (s. Abschnitt 2.2.3) anknüpft. Ein Anwendungsbeispiel auf Basis einer typischen Prozesskombination der Blechmassivumformung (Tiefziehen und Fließpressen von Blechronden) beschreibt die nicht sichtbaren Abläufe während der automatischen Wissensakquisition. Zudem wird auch die generelle Benutzung des Assistenzsystems zur Synthese und Design for Manufacture Analyse blechmassivumgeformter Bauteile dargestellt.

2 Wissenschaftliche Grundlagen und Stand der Forschung

Im folgenden Kapitel erfolgt eine Analyse des Umfelds der Konstruktion, die den Einsatzkontext des zu entwickelnden Assistenzsystems darstellt. Besonders der Aspekt des *Wissens* wird dabei genauer untersucht und dargestellt. Das Feld der wissensbasierten Konstruktion, mit den wissensbasierten Systemen als bekannteste Technologie, wird am Anschluss daran vorgestellt. Bekannte und aktuelle Beispiele für wissensbasierte Assistenzsysteme werden präsentiert und v. a. hinsichtlich der eingesetzten Wissensakquisitionsmethoden analysiert. Eine Prüfung inwiefern im Rahmen der vorliegenden Arbeit auf den Entwicklungen dieser Systeme aufgebaut werden kann, erfolgt im Kapitel 4. Das zu entwickelnde Assistenzsystem soll Bauteile hinsichtlich eines fertigungsgerechten Gestaltentwurfs analysieren. Daher wird der Begriff „Fertigungsgerechtigkeit“ in einem eigenen Abschnitt behandelt und seine Bedeutung im Kontext des Design for X (DfX) hervorgehoben. Die Darstellung ausgewählter, bestehender Design for Manufacture Lösungen rundet den Abschnitt ab. Zur Lösung der skizzierten Aufgabenstellung wird der Technologie der Wissensentdeckung in Datenbanken (engl. knowledge discovery in databases (KDD)) hohes Potential beigemessen. Zu KDD existieren in der Literatur verschiedene Vorgehensmodelle, von denen das nach FAYYAD im Detail vorgestellt wird. Im Vergleich zur Konstruktionstechnik ist KDD noch ein relativ junges Forschungsfeld, dennoch lassen sich in der Literatur bereits erste Anwendungen von KDD im Produktentstehungsprozess finden. Diese werden umrissen, um das generelle Potential von KDD zu betonen und ein objektives Bild vom Stand der Forschung zu präsentieren. In jedem der folgenden Abschnitt erfolgt zudem an geeigneter Stelle eine Klärung oder Einführung wichtiger Fachbegriffe, sofern diese für den späteren Verlauf der Arbeit relevant sind.

2.1 Konstruieren als wissensintensive Tätigkeit

Die Erkenntnis, dass fundiertes Ingenieurwissen notwendig für das Durchlaufen des methodischen Konstruktionsprozess ist, lässt sich in der Literatur bereits in den frühen 1940er Jahren finden. WÖGERBAUER sieht diese Notwendigkeit insbesondere dadurch verstärkt, dass sich „Anforderungen, Baustoffe und Herstellverfahren weiter entwickeln und [immer] mehr Menschen sich über konstruktive Fragen verständigen müssen“ [Wög42]. Dieser Sichtweise schließt sich VOLK an und betont dabei, dass Konstruieren neben Wissen auch Können, Phantasie und Einschätzungsvermögen erfordert [Vol41]. Über ein halbes Jahrhundert

später betont RICHTER, dass Wissen im Kampf um Wettbewerbsfähigkeit weitaus wichtiger geworden ist als bspw. hohe Produktionskapazitäten und schnelle Fertigungslinien [RV95]. Aufgrund dieser Bedeutung und der Tatsache, dass *Konstruktionswissen* im Rahmen der vorliegenden Arbeit eine zentrale Rolle einnimmt, wird seine Bedeutung für den Konstruktionsprozess in den folgenden Abschnitten näher betrachtet.

2.1.1 Der Konstruktionsprozess und X-gerechtes Konstruieren

Nach KOLLER [Kol13] beinhaltet die Tätigkeit des Konstruierens alle Synthese- und Analyseschritte, die notwendig sind, um auf der Basis einer definierten Aufgabenstellung die zu einem bestimmten Zeitpunkt bestmögliche Lösung zu beschreiben. Die *bestmögliche* Lösung ist dabei eine genügend zuverlässige, wirtschaftlich realisierbare und sonstigen Bedingungen genügende Lösung. WEBER [Web05] beschreibt die Syntheseschritte zum einen als qualitative Definition relevanter Merkmale (Entwicklung und Auswahl von Prinziplösungen, qualitative Gestaltung) und zum anderen als quantitative Festlegung der Merkmalausprägungen (Dimensionierung). Während der Analyse wird das Produktverhalten, das sich durch die Merkmalausprägungen ergibt, untersucht. Dies geschieht anhand des virtuellen Produkts durch Simulationen und Berechnungen oder anhand physikalischer Prototypen durch Versuche [Lut11].

Das Entwickeln wird im Allgemeinen als ein Unternehmensprozess angesehen, der unmittelbar zum Wert des Produktes beiträgt. Damit eine ausreichende Wertschöpfung sichergestellt ist, sollte dieser Prozess möglichst strukturiert durchlaufen werden. Hierzu können Firmen auf allgemeingültige Vorgehensmodelle zurückgreifen. PAHL ET AL. [PBF07], LINDEMANN [Lin09], KREHMER [Kre12], die VDI 2221 [VDI93] oder die VDI 2206 [VDI04b] präsentieren entsprechende Ansätze. Auch wenn unterschiedliche Bezeichnungen genutzt werden, so gliedern alle Ansätze den Produktentstehungsprozess (PEP) in Teilprozesse, in denen die Ausführung einzelner Arbeitsschritte verlangt wird. Nach jedem erfolgreich durchlaufenem Arbeitsschritt steht den Produktentwicklern ein Arbeitsergebnis zur Verfügung. Je nach Teilprozess und Aufgabenstellung werden die Arbeitsschritte vollständig, teilweise und sogar mehrmals durchlaufen, falls das jeweilige Arbeitsergebnis nicht den Anforderungen genügt.

Die Vorgehensweise nach VDI 2221 ist insbesondere im Maschinenbau anerkannt und wird detaillierter betrachtet (s. Bild 3). Sie zeigt in sieben Phasen einen branchenunabhängigen Ansatz zur methodischen Entwicklung von Produkten auf. Durch eine gesonderte Auseinandersetzung mit den Möglichkeiten der modernen, IT-unterstützten Konstruktion ist eine Abgrenzung gegenüber anderen Produktentwicklungsprozessbeschreibungen festzustellen. Ergänzend sind für eine detaillierte Darstellung der einzelnen Arbeitsschritte die Richtlinien VDI 2222 [VDI97] für die Schritte 1 bis 3 sowie die VDI 2223 [VDI04a] für die Schritte 4 bis 7 heranzuziehen.

Im Zusammenhang mit der Problemstellung der vorliegenden Arbeit ist der Vorgang des Gestaltens von zentraler Bedeutung. Der Gestaltungsprozess ist Bestandteil des übergeord-

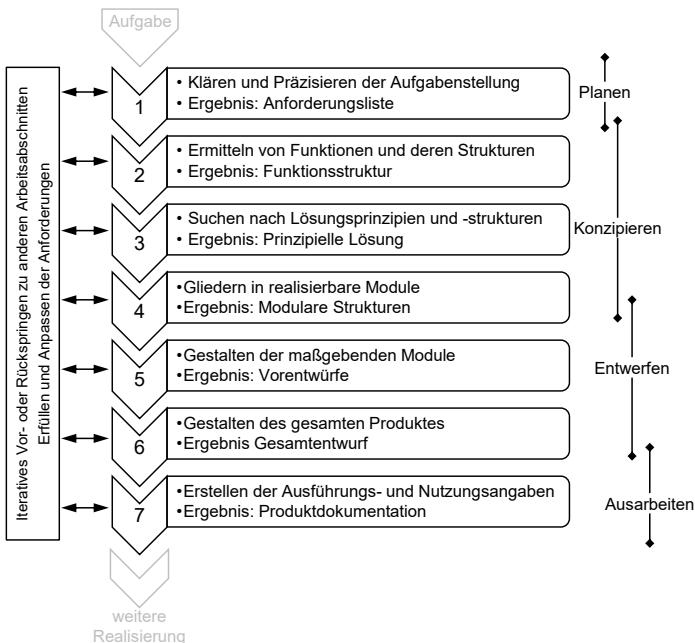


Bild 3: Die sieben Arbeitsschritte beim Entwickeln und Konstruieren nach [VDI93]

neten Entwurfsprozesses und umfasst alle Tätigkeiten, bei denen der Konstrukteur Gestalt- und Werkstoffeigenschaften von Gestaltungselementen festlegt. Gestaltungselemente i. S. d. VDI 2223 sind Elemente von technischen Produkten wie Einzelteileflächen, Formelemente, Einzelteile und Teileverbände [VDI04a]. Diese Elemente werden durch den Konstrukteur in der technischen Zeichnung und der Stückliste, bzw. im virtuellen Produktmodell dokumentiert. Das Gestalten erfolgt immer hinsichtlich zuvor definierter Funktionen und Anforderungen. Die Anforderungen wiederum resultieren aus dem gesamten Lebenszyklus des Produkts und sind zahlreich, vielfältig, teilweise widersprüchlich oder auch voneinander abhängig [Wög42; PBFG07].

Zur Lösung dieser Problematik und zur Unterstützung der Produktentwickler wurden in der Vergangenheit zahlreiche Methoden, Strategien und Werkzeuge entwickelt, die unter dem Begriff *Design for X* (Design for X (DFX)) zusammenzufassen sind [Hub95; Hua96]. Das *X* ist als Platzhalter anzusehen und repräsentiert Einflussfaktoren durch die Anforderungen bestimmt werden [MW00]. Diese sog. Dispositionen stammen entweder aus den Phasen des Produktlebenszyklus (z. B. Fertigung², Montage, Nutzung, Recycling) oder können Produkteigenschaften wie Kosten entsprechen [SAKR05]. HUBKA [Hub95] ver-

² Das Design for Manufacture wird im Abschnitt 2.3 ausführlicher beschrieben.

steht unter DfX ein System, in dem das Wissen darüber, wie gewünschte Eigenschaften eines Produktes im Gestaltungsprozess zu erreichen sind, in geeigneter Form bereitgestellt wird. Der Anwender dieses Wissenssystems ist der Konstrukteur. Auch für TiCHEM [Tic97] verbergen sich hinter DfX die Abhängigkeiten zwischen den Produktparametern, die der Konstrukteur gestalten kann und den erwünschten (wie unerwünschten) Produkteigenschaften im Produktlebenszyklus. Das Wissen, das diese Abhängigkeiten beschreibt, nennt er DfX-Wissen.

Es wird deutlich, dass der Konstruktionsprozess durch eine ständige Wissensverarbeitung und -erzeugung gekennzeichnet ist. Aufgrund dieser Bedeutung wird der Begriff *Wissen* und seine Rolle im Konstruktionsprozess im folgenden Abschnitt genauer betrachtet.

2.1.2 Wissen und seine Rolle im Konstruktionsprozess

In der alltäglichen Sprache sind Begriffe wie *Wissen*, *Daten* und *Informationen* nicht klar abgegrenzt, sondern werden vermischt oder sogar synonym verwendet. Auch zwischen verschiedenen wissenschaftlichen Disziplinen gibt es teilweise keine übereinstimmende Auffassung zu diesem Thema, was laut SCHMALTZ et al. [SH03] u. a. an den unterschiedlichen Sichtweisen auf den Aspekt *Wissen* liegt. Eine Betrachtung aus dem Blickwinkel der Konstruktion verlangt demnach einige Begriffsklärungen.

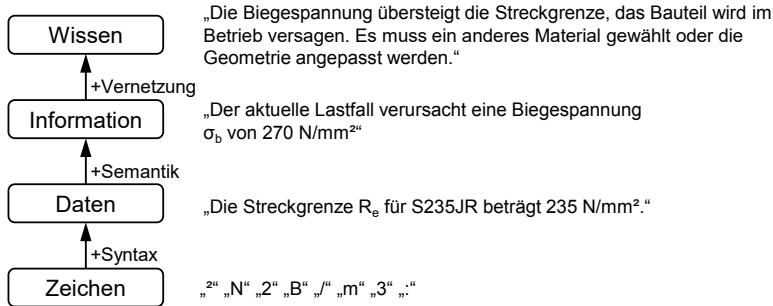


Bild 4: Ein Beispiel zur Erklärung der Taxonomie des Wissens, nach PROBST [PRR03]

PROBST ET AL. [PRR03] liefern die bislang bekannteste Taxonomie zur Abgrenzung der Begriffe Daten, Information und Wissen (s. Abbildung 4). Auf der untersten Stufe stehen eine Reihe von Zeichen (bspw. Zahlen, Buchstaben), diese werden durch Ordnungsregeln in Daten umgewandelt und bestehen aus Zeichen oder einer Zeichenfolge. Durch Angabe des Kontextes oder einer Bedeutung werden die Daten zu Informationen. Die nächste Ebene Wissen entsteht durch die Vernetzung der Informationen und Eingliederung von individuellen Erfahrungen und Erwartungen. Allen Begriffshierarchien gemein ist die Feststellung, dass Wissen immer in Bezug bzw. Verbindung zu Personen steht und äußerst vielfältige Ausprägungen zeigt (s. Bild 5).

SANDER ET AL. [SKH00] stellen fest, dass ein Konstrukteur, der das Ziel der erfolgreichen Bewältigung einer Konstruktionsaufgabe verfolgt, in der Lage sein muss, Handlungen (Festigkeitsberechnung, Getriebeauslegung, usw.) auszuführen. Dazu benötigt er Werkzeuge (Papier, Bleistift, CAD-System, CAE-System, usw.) und Wissen. Dieses Konstruktionswissen versetzt den Konstrukteur in die Lage, seine Handlungen auszuführen (Was?) und seine Werkzeuge effektiv und effizient zu nutzen (Wie?). Auch SPUR ET AL. [SK97] stimmen darin überein, dass konstruktionsrelevantes Wissen an die Tätigkeiten gebunden ist, die der Konstrukteur ausführen muss bzw. diese ermöglicht (z. B. Berechnen und Gestalten). Für FELDHUSEN ET AL. [FG13] ist Konstruktionswissen eine notwendige Voraussetzung um Produktfeatures, sogenannte “vorgedachte Produktelelemente”, richtig anwenden zu können. Es umfasst alle Daten und Informationen über die Elemente selbst, die Regeln ihrer Kombinierbarkeit sowie deren Anwendungskontext. Dabei werden die jeweiligen Rahmenbedingungen der einzelnen Lebensabschnitte des Produkts berücksichtigt. Hier ist deutlich der Bezug zum DfX zu erkennen (siehe Abschnitt 2.1.1). Für PAHL und BEITZ [PBFG07] muss ein Konstrukteur über ein breites Konstruktionswissen und unterschiedlichste Fähigkeiten verfügen, damit er der Vielfalt seiner Aufgaben und den Zielen im Konstruktionsprozess gerecht werden kann. Für besondere Probleme werden Spezialisten mit Fachwissen herangezogen. KOLLER ET AL. [KB90] bezeichnen sowohl das Wissen als Konstruktionswissen, das für eine erfolgreiche Durchführung des Konstruktionsprozesses vorausgesetzt wird, als auch jenes, das während des Konstruierens erworben wird.



Bild 5: Bedeutung des Konstruktionswissens im Produktentwicklungsprozess

Wie die Ausführungen zeigen, sind die Interpretationsmöglichkeiten des Begriffs *Konstruktionswissen* äußerst vielfältig. Damit es jedoch mit Hilfe von Wissensverarbeitungsmethoden im Rahmen der wissensbasierten Konstruktion (siehe Abschnitt 2.2.1) zur rechnerunterstützten Abbildung des Konstruktionsprozesses genutzt werden kann, ist es zunächst zweckmäßig zu strukturieren. Der folgende Abschnitt diskutiert daher Ansätze zur Wissensstrukturierung und betrachtet insbesondere den Aspekt des fertigungsrelevanten Konstruktionswissens.

2.1.3 Strukturierung des Konstruktionswissens

KOLLER ET AL. [KB90] stellten bereits in den frühen neunziger Jahren ein Gliederungskonzept für Konstruktionswissen vor. Sie teilen es in verfahrensbedingtes und aufgabenbedingtes Wissen auf. Ersteres lässt der Bezeichnung nach den Schluss zu, dass es sich um Fertigungswissen handelt. Das *verfahrensbedingte* Konstruktionswissen beschreibt jedoch vielmehr, wie beim Konstruieren vorzugehen ist, also wann welches Wissen benötigt wird. Ebenso zählt Wissen dazu, wie eine Aufgabe strukturiert, d.h. in Teilaufgaben zerlegt werden kann. Da KOLLER ET AL. an dieser Stelle keinen Bezug zur Fertigung und dem damit verbundenem Wissen herstellen, scheidet dieser Teil ihres Gliederungskonzeptes als Referenz für die vorliegende Arbeit aus. Etwas ausführlicher ist das *aufgabenbedingte* Konstruktionswissen gegliedert, das sowohl aufgabenspezifisches Wissen beinhaltet, als auch alle Grundlagen, die ein Ingenieur beherrschen oder nachschlagen kann. Wissenobjekte dieser Art werden durch je drei inhaltliche und formale, voneinander unabhängige Parameter klassifiziert (s. Tabelle 1). Die inhaltlichen Parameter sollen bei der Beantwortung von für die Konstruktion wichtigen Fragen helfen:

- Wo ist das Wissen in der Produktgeschichte zu verorten?
- Welchen Bezug hat das Wissen zur Produkthierarchie?
- Wie ist das Wissen in den Entwicklungsprozess einzurordnen?

Tabelle 1: Wissensstrukturierung nach KOLLER [KB90]

	Parameter	Attributgruppe	Attribut
inhaltliche Parameter	Produktgeschichte	Konstruktionsphase	Funktion, physikalisches Prinzip, Struktur, Gestalt, quantitative Größen
		Lebensphasen	Herstellung, Verkauf, Gebrauch, Beseitigung
Produkthierarchie	allgemein		Ingenieurbereich, Unternehmen, Produktgruppe
		produktspezifisch	Produkt, Teilsystem, Teilkörper, Fläche, Linie, Punkt
Entwicklungsprozess			Basiswissen
			Vorgaben aus der Aufgabe
		Resultat	Ergebnis, Zwischenergebnis
formale oder äußere Parameter	Darstellungsart	Text	ungeordnet, Fakt, Regel, semantisches Netz usw.
		Bild	Zeichnung, Foto, gut-schlecht Darstellung
		Diagramm	Balkendiagramm, Darst. mathemat. Funktion
Wissensquelle	niedergeschriebenes Wissen		Aufgabenstellung, Lehrbuch, Zeitschrift
	nicht niedergeschriebenes Wissen		Erfahrung, fremdes Wissen
Wahrheitswert			auf jeden Fall richtig
			wahrscheinlich richtig
			möglicherweise unrichtig
			wahrscheinlich unrichtig

In diesem Gliederungssystem kann jedem Wissen eine Kombination von Attributen dieser sechs Parameter zugeordnet werden, jedoch ist nicht jede Parameterkombination gleichzeitig auch Wissen. Die Attributgruppen dürfen nicht beliebig kombiniert werden, wodurch das Konzept an Flexibilität einbüßt. Wissen in Form einer Regel zur gussgerechten Gestaltung von Kanten zählt nach KOLLER ET AL. zu Basiswissen und ist nicht veränderbar. Um bestehendes Fertigungswissen über "ausgereifte" Verfahren in Wissensbasen zu hinterlegen, mag dies hilfreich sein. Für Fertigungsverfahren, die sich in einem frühen Entwicklungsstadium befinden und bei denen in kurzer Zeit neue grundlegende Erkenntnisse erlangt und alte verworfen werden, ist dieses Gliederungskonzept nicht anwendbar.

Aus der Problemstellung heraus, dass bis dato entwickelte Strukturierungsansätze keine Struktur bieten, die gleichermaßen passend für jedes Entwicklungsspezifische Problem ist, führt STORATH eine Referenzhierarchie ein, die dieser Problematik begegnet [Sto96]. Im Mittelpunkt stehen verschiedene Sichtweisen auf eine Wissensbasis, die in einer Sammlung, wie sie in Bild 6 dargestellt ist, zusammengefasst sind. Der Nutzer wählt zur Wissensstrukturierung für ein vorliegendes Problemfeld die Merkmalgruppe aus, die die Referenzhierarchie bestimmt. Zum Beispiel sind Dfx-Gerechtheiten essenziell für jeden Konstruktionsprozess. Zur Abstimmung der Wissensstruktur auf ein konstruktionspezifisches Problem werden weitere Merkmalgruppen ausgewählt, um beispielsweise den Ursprung einer Dfx-Gerechtheit (z. B. Produktmontage) zu definieren.

STORATH stellt außerdem fest, dass bei der Klassifizierung von Wissen zwei getrennte Aspekte zu betrachten sind. Zum einen *Wissen für Analyseaufgaben*, wie beispielsweise die Bestimmung einer Lagerlebensdauer, sowie Vorschriften über die korrekte *Anwendung des Wissens*. Er entspricht damit der anerkannten Unterteilung zwischen Fach- und Steuerungswissen [GB88], aufgaben- und verfahrensbedingtem [KB90] oder problemlösungsunabhängigem und -abhängigem Wissen [MKRS93].

Weiterhin stellen ROTH ET AL. fest, dass bestehende Modelle zur Wissensstrukturierung nicht aufzeigen können, welches Wissen in welchem Schritt der Produktentwicklung (s. Abschnitt 2.1.1) vorherrscht [RBW10b]. Auch die Beziehungen und Abhängigkeiten zwischen den einzelnen Wissenstypen sind nicht hinreichend genau beschreibbar. Aus diesem Grund führen ROTH ET AL. eine Terminologie aus fünf Strukturparametern ein, mit der Wissenobjekte im Produktentwicklungsprozess konsistent beschrieben werden können. Der *Wissenstyp* beschreibt den thematischen Bereich innerhalb des PEP, den ein Wissenobjekt abbildet, bspw. Fachwissen, Produktwissen oder Methodenwissen. Insgesamt kann auf 14 verschiedene Wissenstypen zurückgegriffen werden. Hervorzuheben ist der Versuch, die Bedeutung von Wissenstypen für die unterschiedlichen Phasen des PEP zu definieren. Der Strukturparameter *Wissensart* befasst sich mit der Vielzahl an charakteristischen Besonderheiten eines Wissenstyps, ob es sich beispielsweise um implizites, explizites, kollektives oder externes Wissen handelt. Mit der *Wissenform* wird zum Ausdruck gebracht, wie das Wissen vorliegt, zum Beispiel als Text, Formel, Regel oder Bild. Unter *Wissensorient* wird der Ursprung des Wissens wie Personen, Datenbanken oder Abteilungen erfasst. Zu guter Letzt wird mit der *Wissenqualität* eine Bewertung des Wissens hinsichtlich dessen Richtigkeit für einen bestimmten Geschäftsprozess ermöglicht.

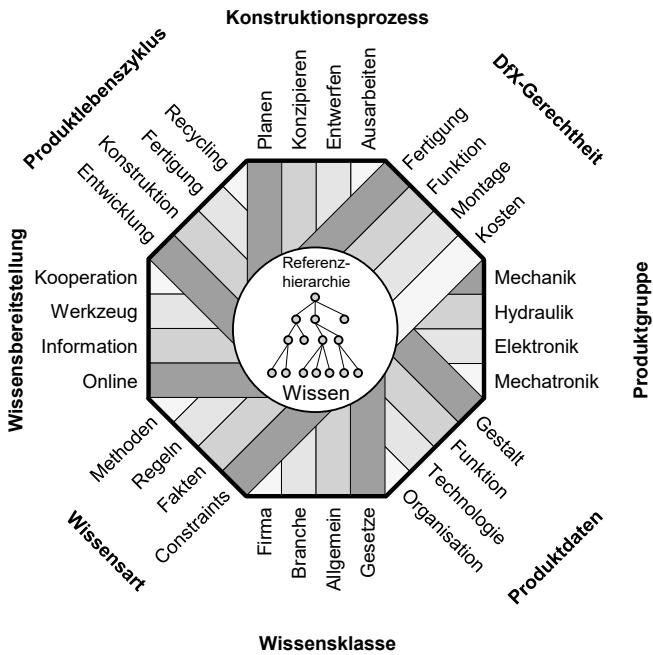


Bild 6: Wissensstrukturierung nach STORATH [Sto96]

Für die vorliegende Arbeit ist der Begriff des fertigungsrelevanten Konstruktionswissens von enormer Bedeutung. Es ist im Umfeld der Konstruktion verortet und dient dort der Beurteilung von Fertigbarkeit bzw. Fertigungsgerechtigkeit durch den Konstrukteur. Nach HUBKA sind die Quellen dieses Wissens die mechanische Technologie, Werkstofflehre und Fertigungstechnik [Hub90]. Wird konstruktionsrelevantes Fertigungswissen dem Konstrukteur zum richtigen Zeitpunkt und in geeigneter Form zur Verfügung gestellt, kann er, wie in Bild 7 dargestellt, feststellen, ob sein derzeitiger Gestaltentwurf durch ein bestimmtes Verfahren realisiert werden kann (Analyse). Bei nicht gegebener Fertigbarkeit versetzt es den Konstrukteur außerdem in die Lage, die Gestalt derart anzupassen (Synthese), dass das Bauteil realisiert werden kann. Weiterhin kann er alternative Produktkonzepte, die einen wesentlichen Einfluss auf die Fertigungsstrategie haben, analysieren, bewerten und je nach Anforderungserfüllung ein Konzept wählen, das im „magischen Dreieck“ Zeit-Kosten-Qualität [EM13] ein Optimum darstellt [War01].

Vom *konstruktionsrelevanten* Fertigungswissen hebt sich aus Sicht des Autors das „reine“ Fertigungswissen ab, das in den Bereichen Arbeitsplanung und Fertigung zu verorten ist. Dort repräsentiert es z. B. in expliziter Form Regeln zur Schnittwertbestimmung bei Fräsprozessen oder dient der Bestimmung von Arbeitsgangfolgen im Werkzeug- und Formenbau [Jur04; Sch10]. In impliziter Form steht es dem CNC-Programmierer u. a. für die

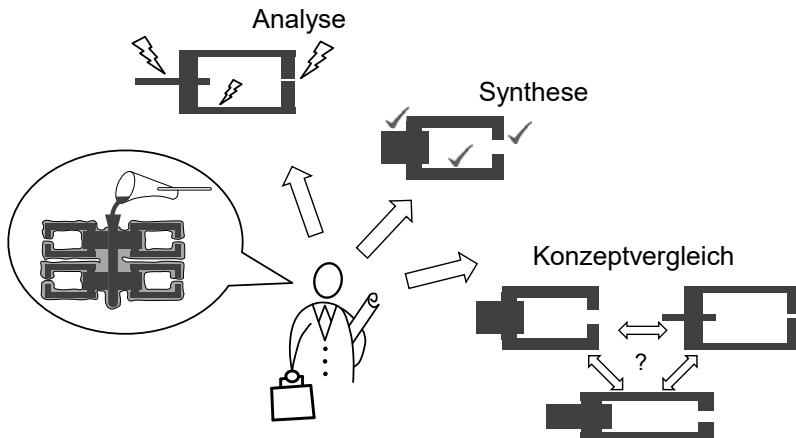


Bild 7: Fertigungsrelevantes Konstruktionswissen kann gleichermaßen zur Analyse, Synthese und zum Konzeptvergleich genutzt werden (im Bild für Feinguss)

adäquate Abstufung des Fräswerkzeugdurchmessers und der Eckradien sowie zur Festlegung der Bearbeitungsstrategie zur Verfügung [DSW08]. Diese Zusammenhänge liegen losgelöst von der Konstruktion vor und sind primär nur für die Fertigung selbst relevant. Weitere Aspekte sind die Bedienung von Fertigungsmaschinen, Sicherheitsrichtlinien, notwendige Kühlenschmierstoffe oder Einsatzrandbedingungen der Werkzeuge. In der Fertigung erhobenes Wissen, das Aspekte beinhaltet die direkt durch die Konstruktion beeinflusst werden, muss jedoch unverzüglich dorthin zurück gespielt werden und wird somit in fertigungsrelevantes Konstruktionswissen transformiert.

2.1.4 Akquisition von Konstruktionswissen

Der Prozess der Wissensakquisition ist als Teil eines übergeordneten Wissensmanagements zu verstehen, dessen Grundlagen durch NONAKA und TAKEUCHI gelegt wurden [NT95]. Ihr SECI-Modell beschreibt die ständige Umwandlung zwischen explizitem und implizitem Wissen in einer Organisation. In den sequenziellen Teilprozessen **Sozialisation** (implizit zu implizit), **Externalisierung** (implizit zu explizit), **Kombination** (explizit zu explizit, engl. **Combination**) und **Internalisierung** (explizit zu implizit) wird Wissen in einer Firma spiralförmig von personengebundenem Wissen auf höhere Stufen wie Teams oder Gruppen gehoben und im gesamten Unternehmen verteilt [NT12].

Ziel der Wissensakquisition ist die Formalisierung von Wissen, also dessen Überführung in eine rechnerverarbeitbare Repräsentationsform. Auch die Aufgabe der Verbesserung und Wartung einer bestehenden Wissensbasis kommt der Wissensakquisition zu [SK97]. Als Wissensquelle dienen u. a. Experten, Fachbücher, Normen oder elektronischen Dokumente.

Die Darstellung der Phasen, die während der Wissensakquisition zu durchlaufen sind, ist in der Literatur nicht einheitlich. Tabelle 2 fasst eine Vielzahl von Ansätzen zusammen, die entweder einen zwei- oder einen dreistufigen Prozess postulieren.

Tabelle 2: Verschiedenen Ansätze zur Beschreibung des Wissensakquisitionsprozesses

AutorIn	Phase I	Phase II	Phase III
BORNDORF [BE97]	Wissenserhebung	Wissensinterpretation	Wissensoperationalisierung
BUCHANAN [BBB ⁺ 83]	Wissenstransfer	Wissenstransformation	—
BUTTENBRUCH [BF89]	Wissenserhebung	Wissensanalyse	—
DIAPER [Dia89]	Wissenserhebung	Wissensrepräsentation	Wissensimplementierung
FRICK [Fri98]	Wissenserhebung	Wissensanalyse	Wissensmodellierung
GÖBLER [Göb92]	Wissenserhebung	Wissensinterpretation	Wissensoperationalisierung
HOPPE [Hop92]	Wissenserhebung	Wissensanalyse	Wissensoperationalisierung
KARBACH [Kar88]	Wissenserhebung	Wissensabbildung	—
KRATZER [Kra14]	Wissenserhebung	Wissensanalyse	Wissensrepräsentation
LENZ [Len91]	Wissenserhebung	Wissensanalyse	Wissensmodellierung
MINOR [Mir06]	Wissenserschließung	Wissensintegration	—
PUPPE [Pup91]	Wissensidentifikation	Wissensformalisierung	Wissenswartung
SCHMIDT [Sch95]	Wissenserhebung	Wissensinterpretation	Wissensformalisierung
SCHNEIDER [Sch94]	Wissenserhebung	Wissenstrukturierung	Wissensimplementierung
SPUR ET AL. [SK97]	Wissenserhebung	Wissensformalisierung	—
SUHUM [Suh93]	Wissensentnahme	Wissenstransformation	Wissensübertragung
WARTZACK [War01]	Wissenserhebung	Wissensinterpretation	—

In jedem Fall stellt die Wissensakquisition den zeit- und arbeitsaufwendigsten Teilprozess des Knowledge Engineerings (KE) dar. Bereits FEIGENBAUM [Fei81] bezeichnete sie als „the most critical bottleneck problem“. An dieser Darstellung hat sich über die Jahrzehnte nichts geändert, wie den Arbeiten von CREMERS [Cre91], KURBEL [Kur92] und BEIERLE [BKI08] zu entnehmen ist. Die Kategorisierung von Akquisitionsmethoden orientiert sich vor allem am Zusammenwirken von Wissensquelle, Wissensbasis und ggf. dem sogenannten Wissensingenieur. Grundsätzlich lassen sich drei Methodengruppen identifizieren, die in Bild 8 dargestellt sind und im Folgenden beschrieben werden.

Direkte Wissensakquisition

Diese Form der Wissensakquisition liegt vor, wenn der Experte sein Wissen über ein Akquisitionstool *direkt* an das Assistenzsystem übergibt. Häufig wird einem solchen Akquisitionstool die Eigenschaft *intelligent* beigemessen, da es in der Lage sein muss die Sprache des Experten zu nutzen [Pup91]. Die Akquisitionskomponente tritt dabei mit dem Spezialisten in einen Dialog und ermöglicht ihm die Eingabe von Fakten und Lösungsstrategien, wobei eine graphische Fakteneingabe anzustreben ist [BG89]. Das System muss zusätzlich die Aufgabe der Übersetzung des eingegebenen Wissens und der internen Darstellung übernehmen [SK97]. Es darf ihn nicht bei der Formalisierung seines Wissens durch die Verwendung von Ausdrucksmitteln aus anderen Domänen behindern, wie etwa der künstlichen Intelligenz (KI). Alternativ verfügt der Experte selbst über ein ausreichendes Verständnis für die Wissensstruktur des Assistenzsystems. Von der Verwendung direkter Methoden

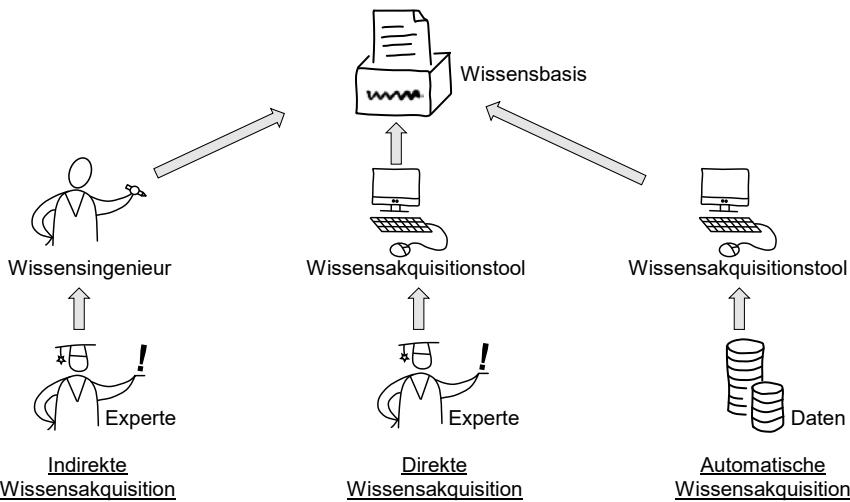


Bild 8: Methoden der Wissensakquisition lassen sich nach [SK97; Kur92] in indirekte, direkte und automatische Ansätze einteilen

wird bei der erstmaligen Entwicklung eines wissensbasierten Systems abgeraten, im Zuge der laufenden Pflege und Erweiterung der Wissensbasis beim späteren Einsatz des Systems können sie aber unter Umständen sinnvoll sein [Kur92].

Indirekte Wissensakquisition

Das wesentliche Merkmal der indirekten Wissensakquisition ist der Einsatz eines Spezialisten, der die Aufgabe hat, das personengebundene Wissen der Experten zu erheben, zu formalisieren und in die Wissensbasis zu integrieren. Aufgrund dieser Tatsache hat sich die Bezeichnung *Wissensingenieur* (engl. knowledge engineer) durchgesetzt [Sto01]. Wissen gelangt somit *indirekt* vom Experten über den Wissensingenieur zum Assistenzsystem. Weitere Aufgaben eines Wissensingenieurs sind nach [KL90] der gesamte Planungs-, Entwicklungs- und Betriebsprozess eines Assistenzsystems. Hierzu gehören Vorstudien (z. B. Machbarkeitsstudien), die Auswahl von geeigneten Werkzeugen sowie die Integration in die IT-Landschaft des Unternehmens und die Weiterentwicklung des Wissensmanagementsystems einschließlich sämtlicher KBE-Projekte. Nach BIMAZUBUTE müssen Wissensingenieure eine möglichst breite Ausbildung und eine umfassende Allgemeinbildung aufweisen [Bim05]. Nützlich sind weiterhin Kenntnisse in den Informations- und Kommunikationswissenschaften sowie in den Humanwissenschaften. Zudem müssen Wissensingenieure auch über grundlegendes Wissen der Psychologie, Logik, Systemanalyse und Linguistik verfügen sowie geeignete Methoden aus der empirischen Sozialforschung beherrschen.

Automatische Wissensakquisition

Ansätze zur automatischen Wissensakquisition verzichten gänzlich auf Experten oder Wissensingenieure. Das wissensbasierte System „lernt“, indem es Problemstellungen löst und dadurch seine Wissensbasis *automatisch* erweitert. Andere (induktive) Ansätze gehen davon aus, dass dem System Problemstellungen und die dazugehörigen Lösungen zugeführt werden und es daraufhin seine Wissensbasis aufbaut oder modifiziert [BB99; Kon90]. Eine weitere wichtige Form ist das Extrahieren von Wissen aus textuellen Quellen wie z. B. aus Texten und Büchern. Ein textverstehendes Programm erfasst dabei den Text ohne Eingriff des Menschen, analysiert diesen linguistisch und formalisiert Wissen in Form von Fakten und Regeln. Die automatische Wissensakquisition galt zu Beginn der Expertensystemära noch „weitgehend [als] Zukunftsmusik“ [Kur92]. SPUR und KRAUSE [SK97] verlagern den Schwerpunkt der automatischen Wissensakquisition inhaltlich in die Nähe zum maschinellen Lernen. Methoden des Data-Mining und der Wissensentdeckung in Datenbanken (Knowledge Discovery in Databases, KDD, siehe Abschnitt 2.4) können, sofern sie zielführend eingesetzt werden, den Prozess der Wissensakquisition verbessern [BKI08].

Diskussion der drei Verfahren

Die Identifikation und die Verfügbarkeit der Wissensquellen sind methodenübergreifende Problemfelder bei der Akquisition von Konstruktionswissen. Der Erfolg bei der Erhebung von personengebundenem Wissen (direkt und indirekt) hängt in besonderem Maße von der Bereitschaft der Experten ab, sich als Wissensquelle zu offenbaren. Sind sich Mitarbeiter der Qualität ihres Wissens nicht bewusst oder ängstigen sie sich vor Ideenraub und davor austauschbar zu werden, so kann es bei der Erhebung des Wissens zu Schwierigkeiten kommen [VDI09b]. Wissensingenieure sehen sich mit der Herausforderung konfrontiert, dass nur ein begrenzter Teil des menschlichen Wissens als aktives Wissen jederzeit abrufbar ist. Der größte Teil ist unbewusst und kann nur situationsbedingt abgerufen werden. Dies betrifft insbesondere Wissensteile, die nicht durch einen bewussten Lernprozess, sondern durch Erfahrung erworben wurden [Dia89]. DENGEL [Den94] stellt fest, dass es gerade Experten schwer fällt ihr Wissen und ihren Umgang damit auszudrücken. Die hieraus hervorgehende Problematik der Verbalisierbarkeit und Formalisierbarkeit von Wissen führt zu einer unvollständigen und inkonsistenten Wissensakquisition. Dies kann vor allem bei der indirekten Wissensakquisition Verständnisprobleme zwischen Wissensingenieur und Experte hervorrufen, ein typisches Problem beim Informationsaustausch wie WARTZACK [War01] zeigt. Die Rolle der Wissensingenieure als Vermittler zwischen Experte und wissensbasiertem System ist nicht nur aus Zeit- und Kostengründen kritisch zu sehen. Durch den „Umweg“ über den Wissensingenieur kann es zu Verlust von Expertenwissen kommen. Diese Gefahr ist insbesondere dann gegeben, wenn der Wissensingenieur nicht über ausreichend Kenntnisse der Expertendomäne verfügt [SG87]. Bei Betrachtung der automatischen Wissensakquisition ist festzustellen, dass neben den notwendigen IT-Werkzeugen und -Spezialisten auch geeignete Daten zur Verfügung stehen müssen. Gerade bei praxisnahen Anwendungen sind diese mitunter unvollständig, fehlerhaft und über diverse Speicheror-

te verteilt [Ott04; HLW06]. Ein ganzheitliches EDM/PDM-Konzept kann diesem Problem begegnen, indem Daten über Produkte und Produktentstehungsprozesse konsistent gespeichert, verwaltet und für alle relevanten Organisationseinheiten bereitgestellt werden (siehe [Vaj09]). Neben den notwendigen Wissens- bzw. Datenquellen sind eine klare Zielsetzung, ein methodisches Vorgehen sowie die Integration von Hintergrundwissen aus den Bereichen Mathematik und maschinellem Lernen erforderlich, um verlässliches Wissen zu akquirieren. Tabelle 3 fasst die Beurteilung der Akquisitionsmethoden hinsichtlich der wichtigsten Kriterien zusammen.

Tabelle 3: Vergleich von Wissensakquisitionsmethoden nach [RBW10a]

Kriterium	Direkt	Indirekt	Automatisch
Identifikation und Verfügbarkeit der Wissensquelle	◐	◐	●
Fehlerfreie und interpretierbare Formalisierung des Wissens	◐	◐	●
Reduzierung von Störungen durch Menschen im Akquisitionsprozess	◐	◐	●
Reduzierung von Zeitbedarf und Kosten	◐	◐	●

Symbolbedeutung:

◐ ungeeignet

◑ kaum geeignet

● teilweise geeignet

● geeignet

2.2 Wissensbasierte Systeme in der Produktentwicklung

Damit das im Akquisitionsprozess gewonnene Wissen genutzt werden kann, muss es dem Anwender „zur rechten Zeit am rechten Ort in der rechten Menge und in der rechten Form“ vorliegen [Sto95]. Im Abschnitt 2.1.2 wurde bereits die Bedeutung des Wissens für den Produktentwicklungsprozess erläutert. Wird die Forderung von STORATH aus Sicht des Produktentwicklers interpretiert, so handelt es sich beispielsweise bei der *Zeit* um einen bestimmten Schritt im Produktentwicklungsprozess und der *Ort* entspricht der jeweiligen Arbeitsumgebung des Entwicklers (z. B. CAD-, CAE- oder PDM-System). Lösungen zur Bereitstellung von Wissen in diesem Sinne sind vielfältig [Her10; Sto96; Man06; Göb92]. STORATH nennt Vorgehensweisen ohne Rechnerunterstützung (kooperativ, informal, formal, s. Bild 9), hebt jedoch hervor, dass diese bezüglich Art der Bereitstellung, Informationssuche und -dichte sowie Interpretationsaufwand den rechnerunterstützten Ansätzen unterlegen sind. Diese modernen Ansätze der Wissensbereitstellung werden unter der Bezeichnung *wissensbasierte Systeme* (WBS) zusammengefasst.

Die folgenden Abschnitte analysieren die Einsatzmöglichkeiten wissensbasierter Systeme im Bereich der Produktentwicklung. Sie sind nach RUDE [Rud98] in den Themenkomplex *wissensbasiertes Konstruieren* (Knowledge-based Engineering (KBE)) einzuordnen. Nach

einer Einführung in die Thematik KBE im Abschnitt 2.2.1 wird die Technologie der WBS im Abschnitt 2.2.2 näher erläutert und es werden Beispiele aus dem Stand der Forschung vorgestellt. Ein Hauptaugenmerk bei der Darstellung liegt auf der Durchführung der Wissensakquisition in den einzelnen Systemen.

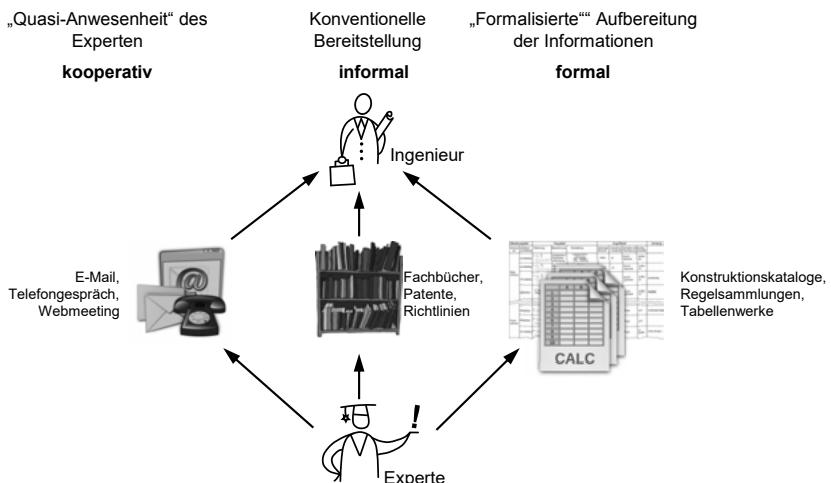


Bild 9: Methoden zur Wissensbereitstellung nach [Sto96]

2.2.1 Wissensbasiertes Konstruieren

Die Bezeichnung *wissensbasiertes* Konstruieren kann von Fachfremden als irreführende Tautologie missverstanden werden, erscheint doch ein Konstruieren ohne Wissen grundsätzlich als unmöglich. Nach RUDE umfasst KBE jedoch die „gedankliche Durchdringung“ des Produktentstehungsprozesses und dessen anschließende (teilweise) Abbildung im Rechner [Rud98]. Im Fokus von KBE-Anwendungen stehen Routinetätigkeiten, die den Konstrukteur eher selten vor unerwartete Probleme stellen, jedoch als langwierig und somit als zeit- und kostenintensiv gelten [Ska07; SVLBG15]. Von diesen Tätigkeiten soll der Konstrukteur entlastet werden, denn sie haben nach STOKES [Sto01] einen Zeitanteil von etwa 80 % am gesamten Konstruktionsprozess. Eng damit verbunden ist die langwierige Suche nach Informationen, die der Konstrukteur beispielsweise zur Beurteilung von Gestaltentwürfen im Sinn des DFX benötigt [Sto96].

Die Ursprünge von KBE in seiner heutigen Form gehen bis in die frühen 1980er Jahre zurück, als erstmalig AI- und KE-Methoden im CAD-Bereich angewandt wurden [CL07; DLK08]. Je nach Autor werden in der KBE-Definition unterschiedliche Schwerpunkte gesetzt. In der Beschreibung von LA ROCCA [La 12] werden die unterschiedlichen

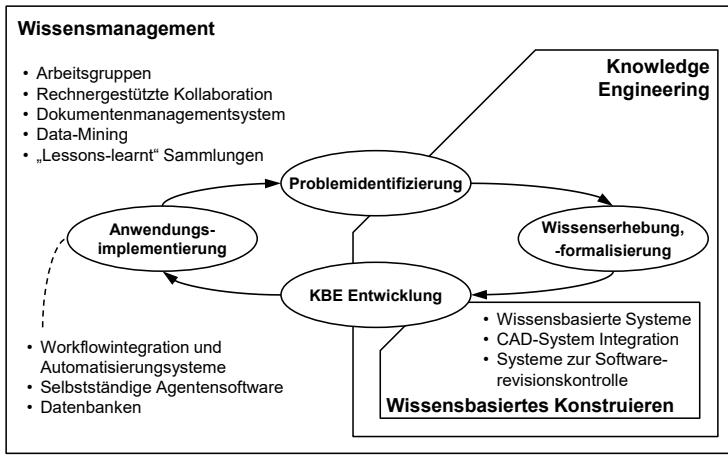


Bild 10: Gegenüberstellung der Domänen KBE, KE und Wissensmanagement [La 12]

Stakeholder einer KBE-Anwendung sehr differenziert betrachtet. Dazu zählen die Unternehmensleitung bzw. Personen des gehobenen Managements (z. B. Entwicklungsleiter), Wissensingenieure, Konstrukteure, Produktentwickler sowie Wissensmanager. LA ROCCA zeigt ein pragmatisches Verständnis für KBE, da der Einsatz von Spezialsoftware zur Wissenserfassung und -wiederverwendung im Mittelpunkt steht. Das Hauptziel von KBE ist seiner Lesart nach die Automatisierung sich wiederholender, monotoner Tätigkeiten beim Konstruieren [La 11]. Wie in Bild 10 zu erkennen, ordnet LA ROCCA das wissensbasierte Konstruieren in ein Gesamtkonzept mit Knowledge Management und Knowledge Engineering auf den übergeordneten Ebenen ein. Dabei ist festzuhalten, dass eine scharfe Abgrenzung zwischen den Disziplinen mitunter schwierig ist, denn ihre Ziele und die angewandten Wissenstechnologien überschneiden und bedingen sich teilweise. Beim Wissensmanagement liegt das Hauptaugenmerk auf dem *strategischen* Ziel das Unternehmenswissen effizient und effektiv zu nutzen, ohne dass einzelne Unternehmensbereiche ausgeklammert oder stärker betont werden. Das Knowledge Engineering wirkt gezielt an *operativen* Unternehmensprozessen mit, indem es die Akquise von Wissen sowie dessen rechnerbasierte Abbildung, Bereitstellung und Nutzung zum Ziel hat. Das wissensbasierte Konstruieren hat letztlich die Entwicklung von Anwendersoftware im Umfeld der Produktentwicklung zum Ziel.

VAJNA ET AL. sehen im KBE die natürliche und logische Fortführung der rechnerunterstützten Produktentwicklung (CAx) [Vaj09]. Wissensbasiertes CAD ist somit die anspruchsvollste, aber auch mächtigste Form der 3D-Produktmodellierung (s. Bild 11). Ein spezielles wissensbasiertes System wird seinem Verständnis nach nicht eingesetzt, stattdessen erfolgt die Wissensbereitstellung direkt durch das CAx-System. Ausprägungen von KBE sind nach Vajna die wissensbasierte Parametrik, erweiterte Features, „intelligente“ Komponentenkataloge sowie wissensbasierte Produktkonfiguratorer. Bei der

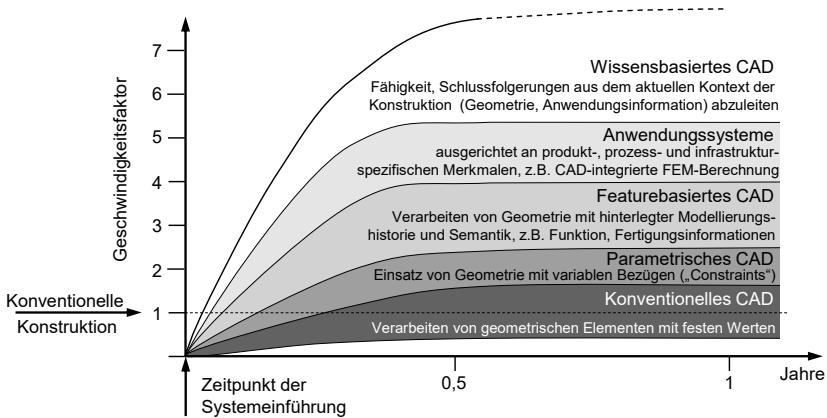


Bild 11: Übersicht der 3D-Modellierungsgrundlagen und deren Einflüsse auf die „Geschwindigkeit“ beim Konstruieren nach [Vaj09; VDI09a; SK97]

wissensbasierten Parametrik wird Wissen in Form von Konstruktions- und Konfigurationsregeln zur Geometriebestimmung direkt im Produktmodell (adaptiver Ansatz) oder in einem externen Speicherort (generativer Ansatz) eingesetzt. Erweiterte Features enthalten nicht nur die Geometrie eines Produktes, sondern auch Wissen mit dem die Funktion und weitere Eigenschaften modelliert werden können. „Intelligente“ Komponentenkataloge unterstützen die Angebots- und Auftragsbearbeitung umfassend, inklusive Produktauswahl, Angebotserstellung und Bereitstellung von CAD-Daten für die Firmenkunden. Wissensbasierte Produktkonfigurerer grenzen sich hierzu vor allem dadurch ab, dass die zur Kombination notwendigen Regeln direkt enthalten sind und ein Produkt widerspruchsfrei konfiguriert werden kann [Bre99a].

Die präziseste Beschreibung für KBE lässt aus Sicht des Autors bei RUDE finden, der in seiner Definition von der Prämisse ausgeht, dass der komplette Konstruktionsprozess für ein Produkt bekannt und verstanden („durchdrungen“) ist [Rud98]. Die rechnergestützte Abbildung des Produktentwicklungsprozesses ist dann nur als Verifizierung dieses Verständnisses zu sehen. Damit hebt sich RUDE von anderen Definitionen für KBE ab. Die Begrenztheit des menschlichen Vermögens, Informationen aufzunehmen und zu verarbeiten, die zukunftssichere Ausbildung junger Ingenieure, die Erhaltung von Entwicklungswissen sowie die Beherrschung komplexer Produktentwicklungsprozesse multidisziplinärer Produkte sind für RUDE die Hauptargumente für eine wissensbasierte Konstruktion. Für ihn grenzt sich KBE klar von erfahrungsbasiertem Vorgehen ab, da es eine formalisierbare Konstruktionstheorie benötigt, wohingegen letzteres nur von einer hinreichend hohen Anzahl an praktischen Beispielen abhängt. Für die Repräsentation von Wissen im Rechner stehen verschiedene Methoden zur Verfügung, von denen die vier häufigsten im Folgenden kurz erläutert werden. Bild 12 zeigt zudem anhand von Beispielen den Aufbau dieser Repräsentationsarten.

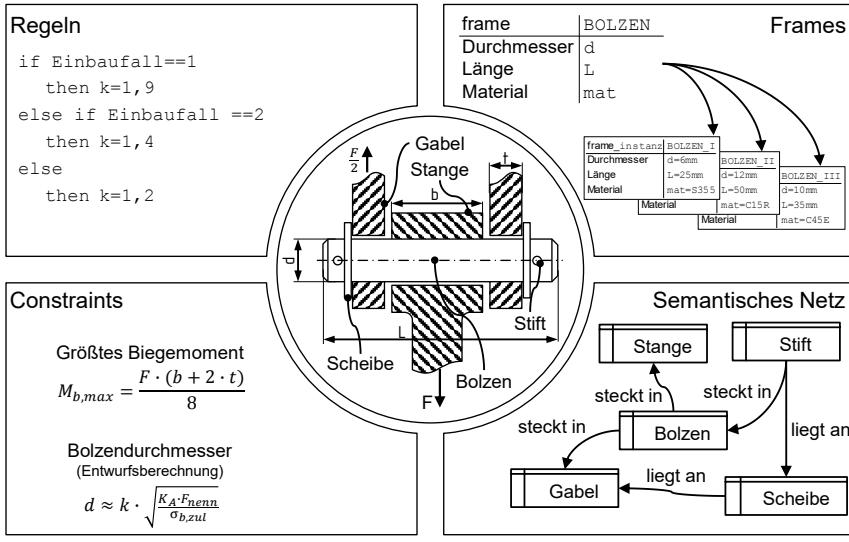


Bild 12: Darstellung häufig verwendeter Wissensrepräsentationsarten am Beispiel einer Bolzenverbindung.

Regeln bestehen aus einem Bedingungsteil (*Wenn...*) und einem Aktionsteil (*Dann...*). Nach [War01] geht der Problemkontext in den Bedingungsteil der Regel mit ein. Regeln entsprechen im Grunde der Denk- und Problemlösungsstrategie von Menschen. Erfahrungswissen in Form von Regeln auszudrücken fällt Experten daher meist sehr leicht. Regelnetzwerke können eine komplexe Struktur annehmen, wodurch die Aktualisierung des Wissens erschwert wird. [BKI14]

Frames (engl. Klasse) entsprechen einem „Karteikartensystem“ zur Repräsentation ähnlicher und sich wiederholender Objekte. Deskriptive Attribute (z. B. Durchmesser, Länge, Material) sind als leere Platzhalter vorhanden und je nach Instanz werden ihnen die entsprechenden Werte zur eindeutigen Identifikation zugewiesen. Klassen eignen sich vor allem aufgrund der Möglichkeit prozedurale Berechnungen an Frames koppeln zu können sehr gut zur Integration von Software in technische Anwendungen. [Pup90; Haa95]

Constraints beschreiben mathematische Zusammenhänge zwischen Parametern oder Kenngrößen von Objekten (z. B. $F = m \cdot a$). Geometrische Constraints werden eingesetzt, um Elemente wie Punkte, Geraden oder Kreise eindeutig zueinander zu positionieren. Ingenieursconstraints beschreiben Konstruktionswissen, beispielsweise für die Berechnung von Maschinenelementen. Constraints können ungerichtete Zusammenhänge beschreiben, so dass in vielen technischen Anwendungsfällen nach einer Variable aufgelöst werden kann. [Rud98; SK97]

Semantische Netze beschreiben die Relationen zwischen Objekten auf Basis markierter, gerichteter Graphen. In einem solchen Graph entspricht ein Knoten dem repräsentierten Objekt (z. B. *Bolzen*) und eine Kante der Relation zu einem anderen Objekt (z. B. *steckt in*). Neben physischen Gegenständen können auch Ereignisse oder Situationen abgebildet und deklarative oder prozedurale Informationen gespeichert werden. [Rud98; SK97]

2.2.2 Die Technologie der wissensbasierten Systeme

Die Anfänge der Wissensbereitstellung durch Maschinen können laut STOKES bis in die Mitte des 19. Jahrhunderts zurückverfolgt werden [Sto01]. Der Ingenieur Charles Babbage konstruierte eine Differenzmaschine³, mit der es möglich war für eine Vielzahl von Artilleriegeschützen die notwendigen Feuerleitbefehle (Abschussrichtung und -winkel) schnell zu berechnen. Das Wissen erfahrener Kanoniere konnte somit erstmals außerhalb ihrer Köpfe „maschinenverarbeitbar“ abgelegt werden.

Von dieser eher populärwissenschaftlichen Auffassung unterscheiden sich moderne wissensbasierte Systeme grundlegend. Sie sind als Teilgebiet der „künstlichen Intelligenz“ (KI) (siehe [MMR06; Ert13]) in der Informatik verortet [BKI14]. Ziel der künstlichen Intelligenz ist die Nachbildung der kognitiven Leistung des Menschen im Rechner.

Architektur wissensbasierter Systeme

Ein wissensbasiertes System besteht prinzipiell, wie in Bild 13 dargestellt, aus mehreren Komponenten, deren Kern die Wissensbasis und die Wissensverarbeitungskomponente bilden. Diese Aufteilung spiegelt in ihrer Struktur auch die funktionale Trennung von Fachwissen und den Problemlösungsstrategien (Wissensverarbeitung) wider, wie sie beispielsweise von KURBEL [Kur92] gefordert wird. Die Dialogkomponente kann in das wissensbasierte System integriert sein oder durch ein externes System (z. B. CAD) angekoppelt werden. Erklärungs- und Wissensakquisitionskomponenten sind nach [SK97] oft nicht direkter Bestandteil des wissensbasierten Systems.

In der *Wissensbasis* ist das gesamte, dem wissensbasierten System zur Verfügung stehenden Wissen gespeichert. Ein modularer Aufbau sorgt für die Trennung des Wissens in begrenzte Anwendungsfelder [War01]. Je nach Herkunft des Wissens wird in bereichsbezogenes Expertenwissen, fallspezifisches Benutzerwissen sowie Zwischen- und Endergebnisse der Wissensverarbeitung unterschieden [Pup90]. Neben dem Wissen selbst sind auch Regeln und/oder Prozeduren gespeichert, welche die korrekte Anwendung des Wissens sicherstellen [SK97]. Das Wissen kann weiterhin in exakte, unvollständige, vage und unsichere Wissensarten untergliedert werden. Andere Ansätze zur Strukturierung der Wissensbasis wurden im Abschnitt 2.1.3 erläutert.

³ Mechanisches Rechenwerk zur Auswertung polynomialer Funktionen und Interpolation von Tabelleneinträgen

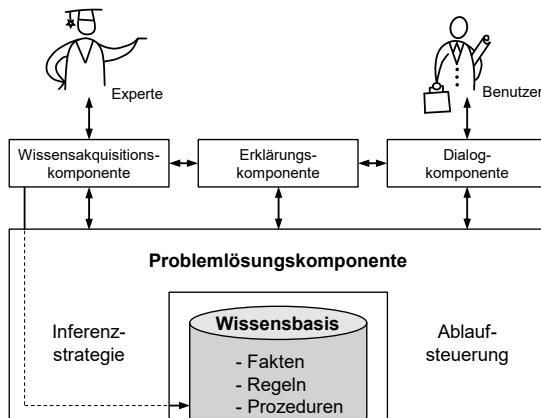


Bild 13: Prinzipieller Aufbau eines wissensbasierten Systems nach [SK97]

Die *Wissensverarbeitungskomponente*, oder auch Problemlösungs- bzw. Inferenzkomponente, bildet die Schnittstelle zwischen der Wissensbasis und den Modulen, die direkt mit dem Anwender kommunizieren. In der Darstellung nach [SK97] sind in der Wissensverarbeitungskomponente Inferenzstrategien und die Ablaufsteuerung aktiv. Erstere verwendet systematische Ableitungsverfahren zur Problemlösung, um von gegebenen Prämissen zu Schlussfolgerungen zu gelangen. Letztere bestimmt die Reihenfolge des Schlussfolgerungsprozesses.

Die *Dialogkomponente* steuert die Kommunikation zwischen wissensbasiertem System und Benutzer. Sie ermöglicht die Eingabe von Daten durch den Anwender und steuert die Ausgabe von Ergebnissen, Ratschlägen oder Auskünften. Daten können hierbei im Dialog mit dem Benutzer gewonnen oder aus einer Datei eingelesen werden.

Die *Erklärungskomponente* stellt die Vorgehensweise des Systems sowohl für Benutzer als auch für den Experten nachvollziehbar dar. Mit ihrer Hilfe kann der Experte Fehler in der Wissensbasis im Rahmen von Testszenarien erkennen. Für den Benutzer bietet sich die Möglichkeit, das Systemverhalten sowie den Lösungsweg zu verstehen. Die vom System gelieferten Erklärungen und Begründungen müssen dabei für den Benutzer verständlich formuliert werden. Dessen Vorwissen ist bei der Erklärung zu berücksichtigen.

Über die *Wissensakquisitionskomponente* erfolgt der Aufbau der Wissensbasis. Je nach Akquisitionsmethode (siehe Abschnitt 2.1.4) ermöglicht sie dem Experten (direkt) sein Wissen in das wissensbasierte System einzugeben oder zu aktualisieren. Die Wissensakquisitionskomponente übernimmt dabei die Aufgabe der Strukturierung, Konsistenzprüfung und Protokollierung. Wird die Wissensakquisition durch einen Wissensingenieur durchgeführt, übernimmt er die Wissensstrukturierung und -eingabe.

Abgrenzung wichtiger Begriffe

In der Literatur finden sich neben dem Begriff des wissensbasierten Systems häufig die Bezeichnungen *Expertensystem* und *Assistenzsystem*. Da die vorliegende Arbeit in den Bereich der Konstruktionstechnik einzuordnen ist, sind auch die in den 1990er Jahren entwickelten *Konstruktionssysteme* von Bedeutung, die einen enormen Technologievorsprung gegenüber den damaligen kommerziellen CAD-Systemen bedeuteten. Die Taxonomie der genannten Begriffe, die im Folgenden kurz diskutiert wird, ist in Bild 14 dargestellt.

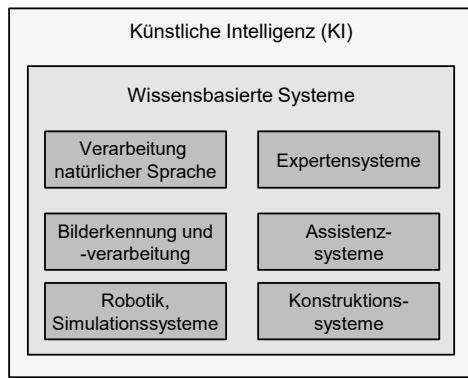


Bild 14: Taxonomie wichtiger Begriffe nach [Wat86]

Eine Abgrenzung zwischen wissensbasiertem System und Expertensystem lässt sich in der einschlägigen Literatur nur bei [Pup90] finden. Nach PUPPE ist ein Expertensystem eine spezielle Ausprägung eines wissensbasierten Systems, bei dem das Wissen letztlich vom Experten stammt. Im weiteren Verlauf der Arbeit spielt dieser Unterschied nur eine untergeordnete Rolle, daher werden beide Begriffe synonym verwendet. BEIERLE ET AL. merken an, dass nach dem Puppe'schen Kriterium viele der derzeit existierenden wissensbasierten Systeme Expertensysteme sind [BKI08].

Auch die Familie der *Assistenzsysteme* wird von Puppe beschrieben. Hierbei handelt es sich um eine Gruppe von interaktiven Expertensystemen, die eine beratende, kritisierende oder vorschlagende Funktion übernehmen. Das System soll keine Ergebnisse generieren, sondern problembezogen Wissen vermitteln. Es wendet sich an sachkundige Benutzer, die ein Problem prinzipiell eigenständig lösen können. Sie tragen als Nebeneffekt aber zur Weiterqualifizierung des Benutzers bei und können daher auch zur Aus- und Weiterbildung eingesetzt werden. Nach STORATH sind Assistenzsysteme die Verwirklichung der Ziele von wissensbasierten Systemen der zweiten Generation [Sto96]. Er zieht Parallelen zu menschlichen Assistenten die zunächst den Wortschatz ihrer Chefs beherrschen müssen bevor sie (pro)aktiv den Problemlösungsprozess mitgestalten und den Vorgesetzten *assistieren* können. Für rechnerunterstützte Assistenzsysteme in der Produktentwicklung leitet

sich daraus ab, dass das System zunächst die Elemente der Konstrukteurssprache (Name und Bedeutung konstruktiver Elemente) beherrschen muss.

Solche *Konstruktionssysteme* (engl. KBE system) zeichnen sich nach [Ver07] dadurch aus, dass die Komponenten eines wissensbasierten Systems zumindest mit einer Benutzerschnittstelle zur geometrischen Produktmodellierung ergänzt werden. [Rud98] liefert an dieser Stelle eine sehr präzise Beschreibung. Sobald aufgrund des hinterlegten Konstruktionswissens mehr Information erzeugt als durch den Konstrukteur eingegeben wurde, im äußersten Fall das wissensbasierte System alle Tätigkeiten des Konstrukteurs unterstützt, spricht RUDE von einem wissensbasiertem Konstruktionssystem. Konstruktionssysteme unterscheiden sich nach [Bac97] von „klassischen“ CAD-Systemen v. a. dadurch, dass der Benutzer in Konstruktionssystemen auf semantisch-hochwertige Objekte aus seinem Konstruktionsumfeld wie Passfedernuten, Gusswände oder komplexe Lagersitze zurückgreifen kann. Außerdem verfügt jedes Konstruktionssystem über ein eigenes Produktmodell, in dem neben geometrischen auch funktionale, technologische und organisatorische Informationen gespeichert werden. Wird der Grundgedanke eines Konstruktionssystems konsequent wie bei FINKENWIRTH umgesetzt und dient das CAD-System lediglich zur Realisierung des Konstruktionsdialogs und Darstellung der Ergebnisse, so ist das Produktmodell des CAD-Systems für das Konstruktionssystem „bedeutungslos“ [Fin90].

Eine sehr umfassende Definition findet sich bei KRATZER [Kra14], wonach es sich bei einem Konstruktionssystem in jedem Fall um ein Softwaresystem handelt, dessen Ausführung zunächst unabhängig von der Anbindung anderer IT-Bausteine (z. B. CAD-System, wissensbasiertes System, Datenbanken) ist. Dieses System untersucht einen CAD-Entwurf unter Einbeziehung der Aspekte Gestaltung, Auslegung und Nachrechnung und ist in der Lage, bei der Produktsynthese Lösungsvorschläge umzusetzen. Hierzu nutzt es Konstruktions- und Konfigurationsregeln sowie Berechnungsformeln, die in einer Wissensbasis hinterlegt sind. Der Konstrukteur wird im Idealfall ständig mit aktuellem Wissen aus anderen Abschnitten des Produktentstehungsprozesses versorgt. Im Hinblick auf die verschiedenen Phasen des Produktentstehungsprozesses ist die Definition von KRATZER dahingehend zu erweitern, dass ein Konstruktionssystem seine Arbeitsergebnisse (z. B. CAD-Modelle) so zur Verfügung stellt, dass sie in allen nachfolgenden Phasen optimal genutzt werden können (z. B. Simulation, Arbeitsvorbereitung).

Um einen Überblick zum gegenwärtigen Stand der Forschung zum Einsatz wissensbasierter Systeme in der Produktentwicklung zu vermitteln, werden im Folgenden einige Forschungssysteme vorgestellt.

2.2.3 Beispiele wissensbasierter Konstruktionssysteme

Die Erforschung von Konstruktionssystemen lässt sich im deutschsprachigen Raum bis in die späten 1980er Jahre zurückverfolgen [Nei90]. Über die Entwicklungshistorie hinweg werden Systeme der ersten bzw. zweiten Generation unterschieden, wobei eindeutige und konsistente Kriterien schwer auszumachen sind. [Kra14] bezeichnet diese als *klassische Konstruktionssysteme*.

sche bzw. *agentenbasierte* Konstruktionssysteme. Seine Unterscheidung basiert auf der Betrachtung der Aspekte Wissensakquise (direkte Kodierung vs. methodischer Akquisitionsprozess), Wissensbasis (monolithisch, unverteilt vs. modular, multirepräsentativ) und Rollenverständnis (keine Rollen vs. dedizierte Rollen) bei der Systementwicklung. FUNKAT ET AL. sehen in der direkten Kodierung des Wissens über Regeleditoren ein markantes Merkmal wissensbasierter Systeme der ersten Generation, gleichzeitig jedoch auch eine ihrer größten Schwächen [FF03]. Zur Überwindung dieses Nachteils verfolgen wissensbasierte Systeme der zweiten Generation u. a. einen methodischen Modellierungsansatz während der Wissensakquise. Außerdem wird bei derartigen Systemen strikt zwischen Wissensrepräsentation und -verarbeitung getrennt.

Im Rahmen dieser Arbeit wurden verschiedene wissensbasierte Konstruktionssysteme aus dem Stand der Forschung analysiert. Ein Schwerpunkt der Analyse lag auf der Durchführung der Wissensakquisition in den Phasen Systementwicklung und -wartung. Alle untersuchten Systeme können an dieser Stelle nicht im Detail diskutiert werden. Ausgewählte Systeme werden aufgrund ihrer bedeutsamen Forschungshistorie (KSmfk) und wegen ihres alternativen Grundkonzeptes (ProKon) im Folgenden vorgestellt. Weiterhin wird ein dritter, beachtenswerter Ansatz beschrieben, der das Ziel einer automatisierten Akquise von Fertigungswissen im Werkzeug- und Formenbau verfolgt.

Das Konstruktionssystem mfk

Die ersten Überlegungen zum KSmfk wurden 1988 von MEERKAMM ET AL. am Lehrstuhl für Konstruktionstechnik der Universität Erlangen-Nürnberg präsentiert [MF88]. FINKENWIRTH formulierte als Ziele die verarbeitungsgerechte Dokumentation von Konstruktionsergebnissen sowie die Unterstützung beim Suchen und Beurteilen konstruktiver Lösungen [Fin90]. Er entwickelt das Grundkonzept des KSmfk, das durch die rechnerunterstützte Abbildung einer Konstruktionsmethodik den gesamten Konstruktionsprozess unterstützt. Die Grobstruktur des KSmfk beinhaltet in den ersten Entwicklungsstufen einen *Syntheseteil* mit Konstruktionsmodulen, die für bestimmte konstruktive Aufgaben Funktionen in Form konstruktiver Elemente bereitstellen. Die daraus erzeugten *Produktmodelle* können im *Analyseteil* unter Berücksichtigung von Konstruktionswissen aus der *Wissensbasis* beurteilt werden. Ein kommerzielles CAD-System wird dann nur noch zur 2D/3D-Visualisierung bzw. zur Kommunikation mit dem Anwender benötigt. Um das Konstruktionssystem mit produktspezifischen konstruktiven Elementen aus allen Konstruktionsphasen zu erweitern, hat BACHSCHUSTER den von [Web92] geforderten Merkmaleditor entwickelt. Dieser ermöglicht dem Anwender neue Konstruktionselemente sowie die zugehörige Semantik datenbasiert und ohne Eingriff in den Programmcode in das KSmfk zu integrieren [Bac97]. Die ursprüngliche Grobstruktur von FINKENWIRTH wurde im Laufe der Jahre somit sukzessive um verschiedene Module bis zur in Abbildung 15 gezeigten KSmfk-Architektur erweitert.

Der Aspekt der Wissensakquisition zum Aufbau und/oder zur Pflege der Wissensbasis wird in den einzelnen Arbeiten zum KSmfk unterschiedlich dargestellt. FINKENWIRTH [Fin90] unterscheidet bei der *Wissensdokumentation*, die an dieser Stelle mit Wissensakquisition

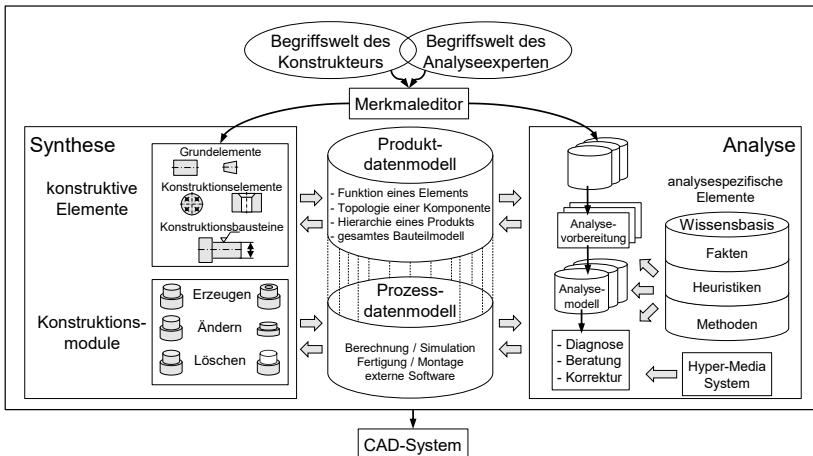


Bild 15: Konzeptionelle Architektur des Konstruktionssystems *mfk* unter Berücksichtigung der Arbeiten von [Fin90; Bac97] und [War01]

gleichgesetzt wird, Faktenwissen und Konstruktionsregeln. Ersteres kann entweder explizit im Code des KSmfk als Formel programmiert sein oder in Form von tabellarischen Dateien bereitgestellt werden. Da keine genaueren Angaben in der Literatur zu finden sind, muss angenommen werden, dass bei FINKENWIRTH die explizite Programmierung des Faktenwissens durch den Softwareentwickler erfolgt, nachdem dieser sich in das Fachgebiet eingearbeitet hat. Er nimmt somit die Rolle eines Wissensingenieurs ein (indirekte Akquise). Der Aufbau von Dateien, bspw. mit Werkstoffinformationen, kann durch den Experten selbst erfolgen (direkte Akquise), sofern der Dateiaufbau einer nachvollziehbaren Struktur folgt. Konstruktionsregeln müssen auch durch den im Programmieren ungewöhnlichen Konstrukteur definierbar sein. Daher schlägt FINKENWIRTH vor, dass sowohl Bedingungs- als auch Aktionsteil einer Regel nur unter Verwendung von UND-Verknüpfungen und ohne Bezug zu anderen Regeln formuliert werden dürfen. Durch diese Unabhängigkeit der Regeln untereinander erreicht FINKENWIRTH, dass jede Regel im Bedarfsfall ohne Konsequenzen für andere Regeln durch den Konstruktionsexperten (direkte Akquise) geändert werden kann.

KRAUSE [Kra92] fokussiert in seiner Arbeit auf die Akquise von quasi-statischem und dynamischem Wissen. Statisches Wissen bedarf keiner Wissensakquisition im eigentlichen Sinn, da es nicht verändert werden muss. Als Beispiel dienen Werkstoffkennwerte, die keinen Änderungen unterliegen. Dem quasi-statischem Wissen sind unternehmensabhängige Fakten über Maschinen, Werkzeuge, Normteile oder Kataloge für Prinziplösungen zugeordnet. Die Aufgabe des Aufbaus und der Aktualisierung der Wissensbasis fällt nach KRAUSE der jeweiligen Fachabteilung zu. Die Ausführungen von KRAUSE lassen auf ein (manuelles) direktes und/oder indirektes Vorgehen bei der Wissensakquise schließen. Er hebt hervor,

dass die Konstrukteure nicht selbst neue Prinziplösungen oder Wissen zur Maschinenelementberechnung in das KSmfk eintragen dürfen, da es sonst zu „Wildwuchs“ kommt. Dynamisches Wissen wird nach KRAUSE für Dimensionierungs- und Auswahlberechnung sowie zur Nachrechnung benötigt. Die bei der Dimensionierungs- und Auswahlberechnung getroffenen Entscheidungen des Konstrukteurs in Bezug auf die Prinziplösungen werden in der Wissensbasis abgespeichert.

Das KSmfk erfasst hierzu, wie oft eine Prinziplösung bei einer gegebenen Belastung zur Auswahl stand und wie häufig diese ausgewählt wurde. Der Konstrukteur kann zusätzlich Gewichtungen für Montage- oder Fertigungsaufwand vergeben. Nach einem zu definierenden Zeitraum werden Eintragungen automatisch in der Wissensbasis deaktiviert, so dass eine Anpassung an die sich verändernden Bedingungen im Betrieb erfolgt. Mit den Informationen über Häufigkeit einer gewählten Lösung, den Gewichtungen und den Anmerkungen des Experten kann ein unerfahrener Konstrukteur bei der Durchführung der Dimensionierungs- und Auswahlberechnung nachvollziehen, welche Lösung aller Wahrscheinlichkeit nach geeignet ist. Da das System zwar „automatisch“ die gewählten Prinziplösungen zählt, der Experte aber dennoch die ausschlaggebenden Informationen manuell eingeben muss, liegt auch im Fall des dynamischen Wissens bei KRAUSE eine direkte Akquise vor.

BACHSCHUSTER [Bac97] integriert in den Analyseteil des KSmfk ein Expertensystemtool⁴ mit eigener graphischer Benutzeroberfläche und verschiedenen Regeditoren zur Eingabe des Wissens. Neue Analysen, die mit Hilfe der Akquisitionskomponente des Tools erstellt werden, werden automatisch erkannt und in die Benutzeroberfläche des Konstruktionsystems integriert. Die Durchführung der Wissensakquisition erfolgt durch Konstrukteure (direkte Akquise), die im Umgang mit dem Expertensystem geschult werden müssen.

Die Akquisition von Konstruktionswissen für DfX-Analysen erfolgt bei WARTZACK [War01] zum einen für den Aufbau eines initialen Wissensbestandes und zum anderen zur Pflege der vorliegenden Wissensbasen. Der erstmalige Aufbau dieser Wissensbasen stützt sich auf Interviews mit Fachabteilungen sowie Literaturrecherchen in verschiedenen Bereichen des DfX. An die 200 Gestaltungsregeln wurden analysiert, formalisiert und durch den Systementwickler in das KSmfk eingebracht (indirekte Wissensakquise). Als Beispiel für die dynamische Pflege der Wissensbasen führt WARTZACK die Rückführung von Wissen über Prozessstörungen aus der Montage in die Konstruktion an. Die Ursache der Störung eines Montageprozesses wird an die Konstruktion zurück geleitet, wo sie durch den Konstrukteur in die Wissensbasis eingetragen wird (direkte Akquise), um bei gleichartigen Aufgabenstellungen frühzeitig ähnliche Fehler vermeiden zu können. Um diese Ursache definieren zu können, ist allerdings Erfahrungswissen erforderlich, das idealerweise von den Fertigungsexperten mit der Störungsmeldung angegeben wird. Da Erfahrungswissen zumeist in impliziter Form (siehe Abschnitt 2.1.2) vorliegt, greift WARTZACK für die Wissensinterpretation auf die Methode der *Heuristic Classification* (siehe [Cla85; SBF98]) zurück. Diese Problemlösungsmethode dokumentiert den gesamten Problemlösevorgang und macht

⁴ Bachschuster und später auch WARTZACK verwenden *NEXPERT OBJECT*, eine neutrale Expertensystem-Shell die an nutzerspezifische Anforderungen angepasst werden kann.

ihn transparent, verlangt aber eine gewisse Erfahrung vom Anwender. Dieser Anwender könnte ein Wissensingenieur sein (indirekte Akquise), der die Heuristic Classification regelmäßig anwendet. Der Wissensingenieur übernimmt bei WARTZACK zudem die Aufgaben eine Verbindung zwischen dem Expertensystemtool und den Wissensbasen des KSmfk herzustellen und den Inferenzmechanismus zur Wissensverarbeitung während der Analysen zu definieren.

Abschließend ist festzustellen, dass bei den (Weiter-)Entwicklungen des Konstruktionssystems mfk über die Jahre hinweg nur direkte und indirekte Wissensakquisitionsmethoden zur Anwendung kamen.

Das agentenbasierte Konstruktionssystem ProKon

ProKon (Proaktive Unterstützung von Konstruktionsprozessen durch Softwareagentensysteme) ist eine Gemeinschaftsentwicklung des Instituts für Konstruktionstechnik und Technisches Design (IKTD) und dem Institut für Automatisierungs- und Softwaretechnik (IAS) der Universität Stuttgart. Das System stellt für den Benutzer einerseits ein aktives Nachschlagewerk für Konstruktionswissen dar, auf der anderen Seite überprüft es selbstständig die CAD-Produktmodelle während des Konstruktionsvorgangs auf Inkonsistenzen bezüglich nicht erfüllter Anforderungen [KBR10]. ProKon unterscheidet sich als agentenbasiertes System in Struktur und Wissensverarbeitung von klassischen wissensbasierten Systemen (siehe Abschnitt 2.2.2). Im Bereich der Informatik bezeichnet ein Agent eine Softwareeinheit, die autonom ein bestimmtes Ziel verfolgt und dabei sowohl mit seiner Umgebung als auch mit anderen Agenten interagiert [GUW04]. Agenten treten jedoch nie einzeln, sondern nur im Verbund als Agentensystem auf. Im Konstruktionssystem Prokon sind neun unterschiedliche Agententypen im Basissystem (s. Bild 16) verankert [Kra14]. Wie viele einzelne Agenten letztlich im System aktiv sind, hängt von der Anwendung ab, also ob bspw. die Konstruktion einer Getriebewelle oder eines kompletten Zahnradgetriebes durch ProKon überwacht wird. Neben dem Basissystem zählt ein Wissensintegrationsystem zum ProKon-Kernsystem. Beide haben Zugriff auf eine zentrale Wissensbasis, die Wissen enthält, das für mehrere Agenten relevant ist (z. B. Materialkennwerte). Daneben verfügt jeder Agent über eine eigene gekapselte Wissensbasis zur Zielverfolgung. Somit besitzt das ProKon-System eine verteilte Wissensbasis. Die Interaktion mit dem Konstrukteur wird über je eine graphische Benutzeroberfläche (GUI) für das Basissystem und für das Wissensintegrationssystem sichergestellt. Wie auch andere Konstruktionssysteme nutzt ProKon ein kommerzielles CAD-System zur Visualisierung des Produktmodells über die CAD-GUI. Anders als beim KSmfk wird das CAD-Datenmodell als digitales Produktmodell genutzt. Die Kommunikation zwischen CAD und dem ProKon-Kernsystem erfolgt über eine JT-Schnittstelle⁵. Das Basissystem greift auf die Berechnungslogik von KISSsoft zu, um standardisierte und normgerechte Berechnungen auszuführen [KRBG11a]. Dieser Ansatz eines externen, kommerziellen „Berechnungspakets“ zur Dimensionierungsberech-

⁵ Proprietäres Format zum Austausch von 3D-CAD-Daten, das als besonders kompakt gilt und auch Objekt- und Metadaten (z. B. Toleranzangaben) unterstützt [ISO12].

nung findet sich auch bei KRAUSE, der die Anforderung stellt, dass dieses Programm ohne Dialog im Batch-Betrieb lauffähig sein muss [Kra92].

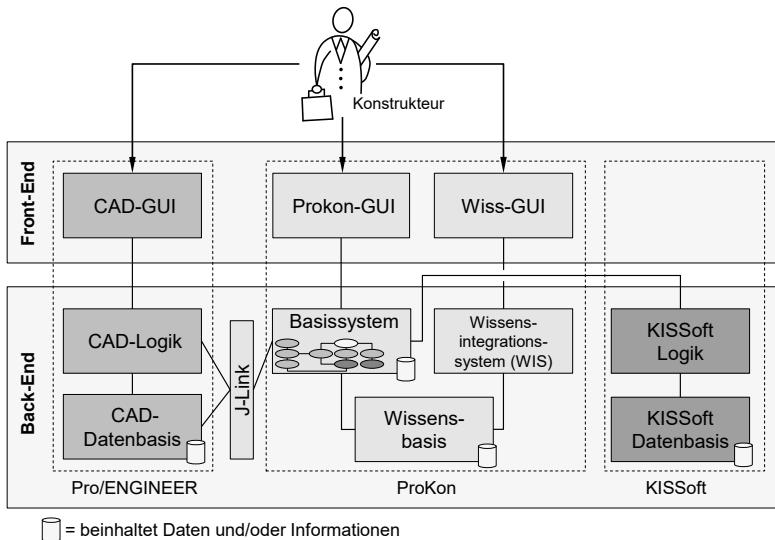


Bild 16: Architektur des agentenbasierten Systems ProKon nach [Kra14]

Die Wissensakquisition bei ProKon teilt sich in die Phasen Wissenserhebung, -analyse und -repräsentation (siehe Tabelle 2). Zur Wissenserhebung greift KRATZER auf die Methode des Exzerpierens für explizites Wissen zurück [Kra14]. Die Erhebung impliziten Expertenwissens spielte im ProKon Projekt mangels Experten nur eine untergeordnete Rolle, jedoch hält sich KRATZER an den Vorschlag von HUA [Hua08], halbstrukturierte Interviews durchzuführen. Da die Durchführung solcher Interviews in den Aufgabenbereich des Wissensingenieurs fällt (siehe Abschnitt 2.1.4), liegt in diesem Fall indirekte Wissensakquise vor. Ist das anwendungsspezifische ProKon-System implementiert, kann beispielsweise das Inkrafttreten oder die Änderung einer Norm die Anpassung der zentralen oder einer agentenspezifischen Wissensbasis erfordern. Diese Aufgabe sieht KRATZER beim Konstrukteur. Hierfür wurde das ProKon-Wissensintegrationsystem entwickelt, das sich auf zwei Prinzipien stützt, nämlich dem Prinzip des fallbasierten Kreislaufs sowie dem Prinzip der Initialisierung [KRBG11b]. Ersteres ist eine adaptierte Form des fallbasierten Schließens (s. [Rud98]) und kontrolliert den Wissensbasiszugriff durch den Konstrukteur, die Verarbeitung semi-formaler Wissensobjekte (z. B. Texte, Skizzen) und Konsistenzprüfungen. Das Prinzip der Initialisierung regelt nach der Modifikation der ProKon-Wissensbasis die Neuordnung des gesamten Systems. Abschließend ist festzustellen, dass beim ProKon-System direkte und indirekte Wissensakquisitionsmethoden zum Einsatz kommen.

Ein automatisiert lernendes CAD/CAM-System

Die Arbeiten von SCHNEIDER am Lehrstuhl für Fertigungslehre der Technischen Universität Chemnitz sind im Bereich der Arbeitsvorbereitung des Werkzeug- und Formenbaus zu verorten. Dennoch findet sich in [Sch10] ein beachtenswertes Konzept zur automatisierten Wissensakquisition von erfahrungsbasiertem Fertigungswissen innerhalb des Produktentstehungsprozesses, dem die Konstruktion bzw. die Produktentwicklung nach [FG13] zugeordnet werden können. Fertigungswissen umfasst im Kontext der Arbeit von SCHNEIDER Erfahrungswissen der NC-Werkstattprogrammierer und ist die Grundlage für eine effiziente CAD/CAM-gestützte Fertigungsplanung (Computer-aided Manufacturing - CAM). Bei der NC-Programmierung im direkten Maschinenumfeld (werkstattorientierte Programmierung – WOP) verwendet der Facharbeiter ein grafisch-interaktives CAM-System und erzeugt bzw. optimiert die NC-Programme selbstständig [Eve02]. Somit wird enorm wertvolles, individuelles Erfahrungswissen zur Sicherstellung der Prozesssicherheit und Effizienz bei der NC-Bearbeitung erzeugt [DSW08]. SCHNEIDER geht davon aus, dass dieses Wissen in den textbasierten NC-Codedatensätzen der Maschinensteuerungen enthalten ist. Um es zu erheben und zur Standardisierung der Prozessplanung zu nutzen, entwickelt er das in Bild 17 gezeigte Vorgehensmodell mit den vier Hauptschritten Datenbeschaffung, Analyse der Wissensdomäne, automatisierte Wissensakquisition und Standardisierung.

Für die *Datenbeschaffung* stehen unstrukturierte, textbasierte Dateien in Form von Werkzeuglisten und Bearbeitungsparametern zur Verfügung. SCHNEIDER entwickelt eine auf regulären Ausdrücken basierende Methodik, die eine vom Format der Quelldatei unabhängige, automatische Konvertierung unstrukturierter Textinformationen ermöglicht. Nach einer notwendigen Datenaufbereitung zur Beseitigung inhaltlicher Fehlstellen stehen ungefähr 24.000 Datenobjekte (NC-Programme) als Trainingsdaten zur Verfügung.

Die *Analyse der Wissensdomäne* hat das Ziel die zu den Trainingsdaten assoziierbaren Wissensbedarfe, unter Berücksichtigung des besonderen Umfelds der werkstattorientierten NC-Programmierung, zu identifizieren. Das Ergebnis der Analyse ist die formale Beschreibung der Wissensziele, also die Definition der bekannten Eingangs- und der unbekannten Ausgangsinformationen.

Zwischen beiden Objekten, deckt der KDD-Algorithmus während der *automatisierten Wissensakquisition* unbekannte Zusammenhänge auf. Als Lernverfahren nutzt SCHNEIDER Markov-Ketten und multivariate Entscheidungsbäume. Diese erfüllen seine Prämisse, dass die explizite Darstellung der gefundenen Zusammenhänge in Form von Regeln durch den Menschen leicht interpretiert und gleichzeitig durch rechnergestützte Systeme effizient verarbeitet werden können. Markov-Ketten (siehe [Bré99b; Mür14]) werden bei SCHNEIDER eingesetzt, um anhand der Attribute Durchmesser und Eckradius eines aktuellen Werkzeugs (z. B. Schruppfräser) diese Attribute für ein unbekanntes, nachfolgendes Werkzeug (z. B. Schlichtfräser) vorherzusagen. Multivariate Entscheidungsbäume, wie der EM-Algorithmus (Expectation-Maximization-Algorithmus, siehe [DLR77]), werden zur expliziten Abbildung der Zusammenhänge zwischen den Attributen Bearbeitungssituation, Werkzeug und Schnittwerten genutzt.

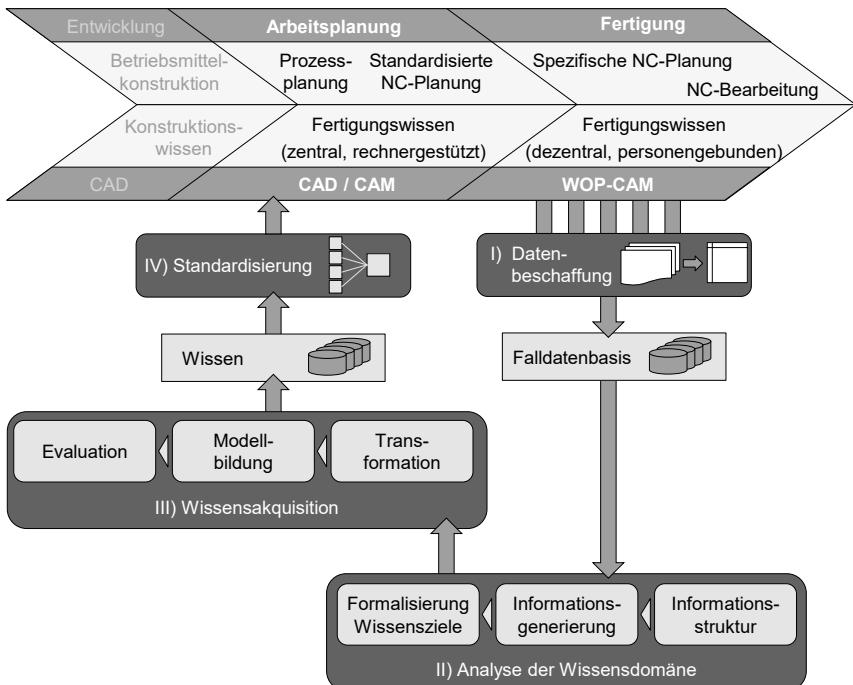


Bild 17: Das Vorgehensmodell nach SCHNEIDER zur automatisierten Akquisition von NC-Fertigungswissen [Sch10].

Abschließend ist eine *Standardisierung* notwendig, da die Trainingsdaten auf den manuell programmierten NC-Datensätzen vieler Facharbeiter basieren und SCHNEIDER eine enorm hohe Varianz und Vielfältigkeit im erhobenen Wissen erwartet. Zur Selektion standardisierbarer Planungsentscheidungen in Bezug auf Fräspräzessattribute nutzt SCHNEIDER für Attribute mit diskretem Wertebereich (z. B. Frästrategie⁶ oder Werkzeugaufnahme) die Lorenzkurve bzw. den Gini-Index [FKPT07]. Somit identifiziert er Prozesskonfigurationen die von den Experten je nach Kontext besonders häufig gewählt und als „Standard“ betrachtet werden. Zur Identifikation häufig verwendeter kontinuierlicher Attribute (z. B. Ausspannlänge, Vorschub, Drehzahl) werden die Quartilsabstände innerhalb der jeweiligen Attributnormalverteilung berechnet.

⁶ Die Frästrategien können den typischen Phasen der Bearbeitung von Freiformgeometrien Schrullen und Schlitten zugeordnet werden.

2.3 Analyse von Bauteilentwürfen hinsichtlich Fertigungsgerechtheit

Aufgabe des im Rahmen dieser Arbeit zu entwickelnden selbstlernenden Assistenzsystems ist die Unterstützung des Konstrukteurs bei der Analyse von Bauteilentwürfen hinsichtlich deren Fertigungsgerechtheit bezogen auf die Blechmassivumformung. Daher sind die Ziele der folgenden Abschnitte die Auseinandersetzung mit dem Begriff *Fertigungsgerechtheit* im Kontext des Design for X und die Darstellung existierender Methoden und Werkzeuge die der Analyse der Fertigungsgerechtheit von Gestaltentwürfen dienen. Eine Betrachtung des Aspekts Fertigungsgerechtheit im Kontext der Blechmassivumformung erfolgt in Kapitel 3.

2.3.1 Fertigungsgerechtheit im Kontext des Design for X

Jeder Fertigungsprozess stellt spezifische Anforderungen an die Konstruktionsabteilung, damit ein Bauteil den anwendungstechnischen Erfordernissen entspricht und gleichzeitig wirtschaftlich hergestellt werden kann. Gerade der Aspekt der Wirtschaftlichkeit grenzt die unterschiedlichen Bedeutungen der Begriffe *Fertigbarkeit* und *Fertigungsgerechtheit* voneinander ab. Für WARTZACK ist die Fertigungsgerechtheit bzw. das *fertigungsgerechte Konstruieren* (engl. Design for Manufacture) einer der komplexesten Themenbereiche des DfX [War01] (s. Bild 18).

Jeder Fertigungsprozess besteht aus einer bestimmten Anzahl an Teilprozessen und Akteuren, die, sowohl zusammen als auch im einzelnen, enorme Auswirkungen auf die Produktkosten und -qualität sowie auf die gesamte Produktivität haben. Erschwerend kommen Abhängigkeiten zwischen den einzelnen Teilaспектen hinzu, was dazu führt, dass eine Entscheidung, die nur einen Gesichtspunkt betrifft, dennoch Auswirkungen auf andere Aspekte ausübt. Ziel des Design for Manufacture ist, ein Verständnis für die komplexen Wechselwirkungen zu entwickeln und dieses Wissen zur Minimierung von Fertigungskosten und -zeit bei gleichzeitiger Berücksichtigung fertigungsabhängiger Qualitätsmerkmale zu nutzen [Sto91]. Wie unterschiedlich das Verständnis für Design for Manufacture in Unternehmen ausgeprägt sein kann, haben Mottonen et al. untersucht. Während im klassischen Maschinen- und Anlagenbau vor allem die Fertigungsvorbereitung und -durchführung im Fokus stehen [AHL92], umfasst Design for Manufacture im I&K-Sektor⁷ das *design for board assembly, design for final assembly, design for testing* und das *design for packaging* [MHB⁺09].

Der Literatur können zur Umsetzung von Design for Manufacture unterschiedliche Maßnahmen entnommen werden. TICHEM schlägt zunächst die Auswahl einer bestimmten Fertigungsprozesskette vor, um in einem zweiten Schritt die Bauteilgestalt in Bezug auf das gewählte Verfahren zu optimieren [Tic97]. Er hebt außerdem hervor, dass beim Design for Manufacture Ansatz sowohl die Produktstruktur als auch jedes Einzelteil mit seinen Materialeigenschaften, Abmaßen, Toleranzen und Oberflächengüten Berücksichtigung findet.

⁷ Informations- und Kommunikationstechnik

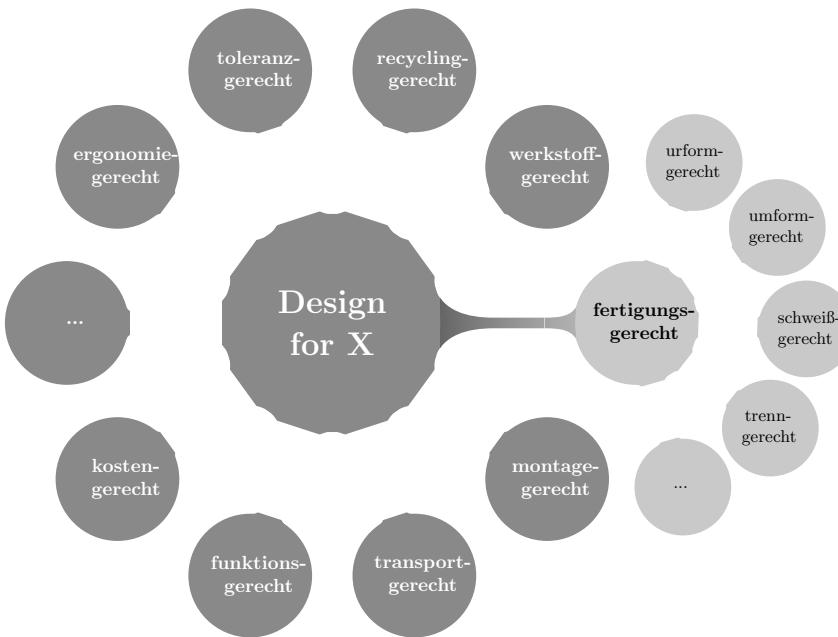


Bild 18: Fertigungsgerechtes Konstruieren im Kontext des Design for X.

Eine ähnliche Beschreibung, allerdings mit einem anderen Blick auf die Optimierung, findet sich bei VAN VLIET [van01]. Er schlägt eine Definition für Design for Manufacture vor, die das Durchlaufen von drei Schritten beinhaltet:

1. Auswahl der besten Kombination aus Material, Geometrie und Fertigungsverfahren für jedes Einzelteil unter Berücksichtigung der an das Produkt gestellten funktionellen Anforderungen.
 2. Fortlaufende Analyse der Fertigbarkeit während des gesamten Konstruktionsprozesses durch Verifikation und Quantifizierung.
 3. „Optimierung“ der Fertigungsgerechtigkeit aller Einzelteile, um die Fertigung zu gewährleisten, zu verbessern oder zu vereinfachen.

VAN VLIET sieht insbesondere bei der Quantifizierung und der „Optimierung“ enormen Forschungsbedarf, da aus seiner Sicht keine Methode zur objektiven „Berechnung“ der Fertigbarkeit existiert. Dies ist wiederum eine Voraussetzung für die „Optimierung“, die er deshalb in Anführungszeichen setzt, weil aus mathematischer Sicht keine optimale Fertigbarkeit existiert [van01].

Für die Design for Manufacture Unterstützung des Produktentwicklers wurden, je nach Fertigungsverfahren, bereits zu Beginn des 20. Jahrhunderts erste Methoden entwickelt. Nach

TICHEM lassen sich diese in die Bereiche Konstruktionsrichtlinien, stand-alone Werkzeuge, CAD-integrierte Systeme sowie rechnerunterstützte Prozessplanung (Computer Aided Process Planning, CAPP) einordnen [Tic97]. In den folgenden Abschnitten werden charakteristische Beispiele dieser Gruppen vorgestellt. Der Übersicht halber erfolgt eine Aufteilung in die Abschnitte *geometriebasierte Analysen* und *rechnerunterstützte Prozessplanung*. Abschließend wird der Bereich *wissensbasierte Ansätze* in einem eigenen Abschnitt behandelt. Die Gruppe der stand-alone Werkzeuge wird nicht im Detail betrachtet, da hier die Herstellkostenberechnung im Fokus steht. Dieser Aspekt ist bei industriellen Anwendungen von enormer Bedeutung, wird jedoch in dieser Arbeit nicht berücksichtigt.

2.3.2 Geometriebasierte Design for Manufacture Analysen

Zu den geometriebasierten Design for Manufacture Analysemöglichkeiten zählen im Folgenden Konstruktionsrichtlinien sowie die CAD-integrierten Systeme. Konstruktionsrichtlinien oder -regeln zur fertigungsgerechten Gestaltung lassen sich entsprechend der Fertigungsverfahren nach DIN 8580 klassifizieren [DIN03] (s. Bild 19). Sie sind ebenso vielfältig wie Herstellungsverfahren existieren, daher werden an dieser Stelle nur einzelne Beispiele herangegriffen.

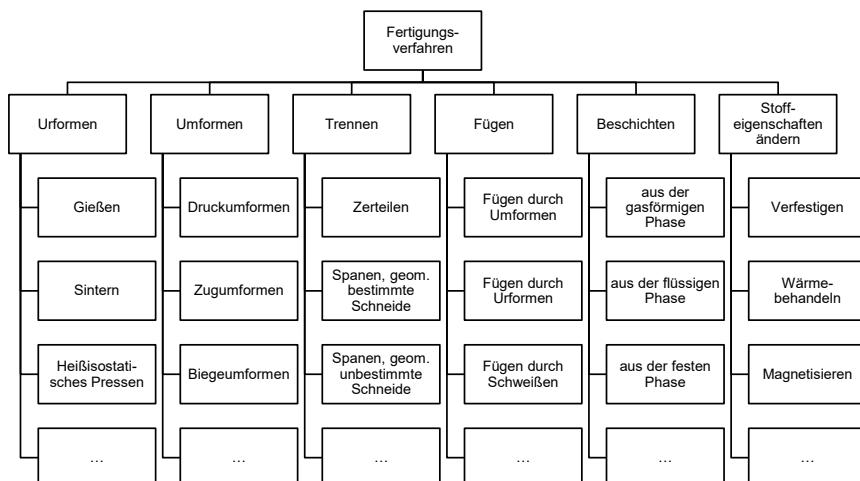


Bild 19: Einteilung der Fertigungsverfahren nach DIN 8580 [DIN03].

Mit Blick auf das Urformen stammen die bekanntesten Konstruktionsrichtlinien von Aloys Heuver, der sich intensiv mit dem Gießen von Stahl beschäftigte. Die nach ihm benannte *Heuvers'sche Kreismethode* zur Bestimmung und Korrektur von Gussquerschnitten hat seit ihrer Einführung in den 1920er Jahren [Heu29] nicht an Bedeutung verloren und findet sich noch heute im Lehrplan technischer Studiengänge [Mee07; War15]. Ausgangspunkt

der Heuvers'schen Kreismethode ist, dass der Speiser für das Nachspeisen des Gussstückes notwendig ist und daher richtig bemessen und angeordnet werden muss. Das Material im Speiser darf erst erstarren, wenn das Gussstück bereits fest ist. Eine einfache Wanddicken- und Speiserbemessung ist mit der Kreismethode nach Heuvers möglich. Das sogenannte Einkreisen der Gussstückquerschnitte beginnt im Bereich der minimalen Wanddicke und wird in Richtung des Speisers fortgeführt. Die Konstruktionsregel lautet, dass die Kreisflächen sich um den materialabhängigen Heuversfaktor k_H bis zum Speiser vergrößern sollen.

Neben der Heuvers'schen Kreismethode sind auch auf gut/schlecht bzw. günstig/ungünstig Beispielen basierende Richtlinien im umformgerechten Konstruieren verbreitet [Rot94; FG13]. Diese bieten den Vorteil der Verständlichkeit und Verallgemeinerbarkeit, denn die Anwendung von Konstruktionsregeln wie etwa das Vorsehen von Entformungsschrägen oder das Vermeiden von Hinterschneidungen ist nicht nur auf Stahl- oder Grauguss begrenzt. Auch für Umformprozesse existieren Konstruktionsrichtlinien und -regeln die einschlägigen Fachbüchern entnommen werden können, wie z. B. [Kug09; Tsc05]. Der Zugriff auf die Wissensinhalte erfolgt beispielsweise wie bei ROTH über Tabellen, die neben den ausformulierten Regeln auch den technologischen Hintergrund sowie ungünstig/günstig Beispiele angeben [Rot94] (s. Bild 20).

Umformabschnitt	Konstruktionsregel zur Gestaltung beim Umformen	Grund	Beispiele für nicht-fertigungsgerechte Gestaltung		Beispiele für fertigungsgerechte Gestaltung	
Fließpressen	Beim Fließpressen Konizität vermeiden, ebenso im Verhältnis zum Durchmesser zu lange Bohrungen	Zu hohe Querschnittsänderungen von kaltem Material				
Tiefziehen	Die Bodenprofile sollten eben sein, die Mantelflächen ggf. bauchig (konvex) und keine oder kleine Kragen haben	Einfacher zu ziehen, weniger Ziehvorgang				

Bild 20: Beispiel einer Konstruktionsrichtlinie zum umformgerechten Konstruieren nach [Rot94].

Andere Konstruktionsrichtlinien basieren auf der Berechnung materialabhängiger Kennwerte wie dem Grenzziehverhältnis β_{max} bei dem die Grenze der Ziehfähigkeit eines Bleches durch den gerade noch nicht eintretenden Bodenreißer erreicht ist [Kug09]. Da das Grenzziehverhältnis nur durch Versuchsreihen ausreichend genau und nur werkstoffspezifisch ermittelt werden kann, ist die Entwicklung umfangreicher, auf β_{max} basierender Richtliniensammlungen sehr zeit- und kostenintensiv. In jedem Fall ist die Erarbeitung von Konstruktionsrichtlinien mit enormen Aufwand verbunden. Sie repräsentieren Exper-

tenwissen, dessen Aneignung (und Bestätigung) viel Zeit und Geld in Anspruch nimmt, gleichzeitig aber eine enorm hohe Sicherheit (Verlässlichkeit) aufweist. Nicht selten verbleibt das relevante Wissen aus Wettbewerbsgründen lange in den Firmen bis es über DIN-Normen oder VDI-Richtlinien Dritten zur Verfügung gestellt wird.

CAD-integrierte Systeme und Ansätze können als Fortführung von Konstruktionsrichtlinien verstanden werden. Dies wurde erst durch die formalen Methoden zur Geometriepräsentation⁸ möglich. Sowohl bei der Geometriesynthese als auch der -analyse gewährleisten CAD-integrierte Ansätze eine Design for Manufacture Unterstützung des Konstrukteurs. An dieser Stelle sei insbesondere die Features-Technologie erwähnt, die auf Forschungs- und Entwicklungsarbeiten aus den 70er Jahren des 20. Jahrhunderts zurück geht [Gra76; CD88; Sha91]. Features als CAD-Gestaltkomplex dienen nicht nur der beschleunigten Erzeugung häufig auftretender Standardgeometrien. Ihr Nutzen zeigt sich vor allem durch die Zuweisung von Semantik und Design for X Wissen, das in einem wissensbasiertem System verarbeitet werden kann. Beim Ansatz nach WARTZACK ET AL. wird beim Start einer Analyse hinsichtlich eines DfX-Aspektes (z. B. spritzgussgerechte Bauteilgestaltung) im Produktmodell nach relevanten Featureinformationen (z. B. Informationen über Gussaugen, Versteifungsrippen, Entformungsschrägen) gesucht und diese im wissensbasierten System verarbeitet [WM00]. Generell kann eine Vielzahl an CAx-Werkzeugen die geometrischen und semantischen Informationen direkt weiterverarbeiten, wie bspw. in den Arbeiten von [UA92; Haa95; War99; KW⁺15] gezeigt wird. Beim featurebasierten Modellieren wählt der Konstrukteur aus einer sogenannten Featurebibliothek ein gewünschtes Feature aus und platziert es auf der Basis parametrischer Bezüge im CAD-Modell. Das zur Dimensionierung notwendige Wissen (bspw. Durchmesser einer Bolzenbohrung, Größe einer Wellennut) kann im Feature selbst hinterlegt sein. Die Verwendung von Features unterstützt eine fertigungsgerechte Produktsynthese dergestalt, dass der Konstrukteur in seiner Auswahl auf Features beschränkt ist, für die sicher ein Standardwerkzeug existiert, bspw. für Senkbohrungen oder Bohrungen eines Passungssystems (siehe [Fis11]). Auch für das Urformen (z. B. Gussauge, Rippe mit Entformungsschrägen) oder das Umformen (z. B. Biegelasche, Falz) können Features mit definierter fertigungsgerechter Gestalt und entsprechender Semantik bereitgestellt werden.

Verschiedene Ansätze zur featureunabhängigen, geometriebasierten Design for Manufacture Analyse stellt WARTZACK in seiner Arbeit vor [War01]. Für die Verfahren Tiefziehen, Druckguss, Spritzguss und Punktschweißen stellt er in seinem Assistenzsystem Analysewerkzeuge zur Untersuchung der fertigungsgerechten Gestaltung beliebig komplexer Bauteilgeometrien zur Verfügung. Bei einem tiefgezogenen Bauteil werden Höhe und Innen durchmesser des Napfes über die B-Rep Datenstruktur und den darin gespeicherten Flächen, Konturelementen, Kanten und Punkten ermittelt. Der Anwender muss lediglich die Tiefziehrichtung vorgeben und das System berechnet die Tiefziehverhältnisse aller Bauteilflächen und vergleicht sie anschließend mit den im Produktmodell gespeicherten, materialabhängigen Grenzziehverhältnis. Druck- und spritzgussgerechte Analysen basieren auf der Ermittlung von Materialanhäufungen, Wandstärkensprüngen und Hinterschneidungen.

⁸ Details zu Repräsentationsformen wie B-Rep oder CSG können u. a. [Vaj09] entnommen werden.

Im Gegensatz zu RÄSE [Räs91], der in seiner Arbeit einen sehr rechenintensiven volumenabhängigen Ansatz verfolgt, berechnet WARTZACK zunächst für jede Teilfläche einen Analysepunkt aus allen die Teilfläche umgebenden Kanten. Aus der Werkzeugschliefrichtungssachse und den durch die Analysepunkte führenden Normalenvektoren der Teilflächen, werden Schnittpunkte mit anderen Teilflächen berechnet. Die Erkennung von Hinterschneidungen beispielsweise erfolgt durch Prüfung jedes Flächenelements auf seine Sichtbarkeit in Richtung der Hauptentformungsrichtung. Die punktschweißgerechte Gestaltung einer Baugruppe erfordert eine ausreichende Zugänglichkeit der Schweißzange. Dies wird überprüft, indem von den im CAD-Assembly definierten Schweißpunkten ein Bündel an geometrischen Strahlen in verschiedene Raumrichtungen erzeugt wird. Trifft ein Strahl auf ein anderes Bauteil so wird diese Entfernung ermittelt und das Expertensystem entscheidet über die Zugänglichkeit eines definierten Werkzeugs.

2.3.3 Rechnerunterstützung durch Prozesssimulation

Der rechnerunterstützten Prozessplanung (engl. Computer-aided Process Planning, CAPP) kommt eine Schlüsselfunktion im Produktentstehungsprozess zu, da sie als Bindeglied zwischen Konstruktion und Fertigung anzusehen ist [SK97]. Ziel ist die Erzeugung formaler, auftragsneutraler, sequentieller und detaillierter Pläne, in denen der gesamte Fertigungsprozess abgebildet ist und anhand derer er optimiert werden kann. Als ein Teilbereich wird im Folgenden die Simulation einzelner Fertigungsprozesse herausgegriffen, um die Möglichkeiten des Rechnereinsatzes darzustellen. Hierbei steht die Umformsimulation im Fokus, da in dieser Arbeit die Blechmassivumformung den Einsatzkontext des zu entwickelnden Assistenzsystems bildet. Ziel dieses Abschnitts ist weniger eine Einführung in die Theorie der FE-basierten Umformsimulation, hierzu wird u. a. auf [Lan93] [DB10] und [Tsc05] verwiesen. Vielmehr sollen das generelle Vorgehen und die Möglichkeiten der Design for Manufacture Unterstützung skizziert werden.

Die Umformsimulation hat sich sowohl im Bereich der Blech- als auch der Massivumformung etabliert. Sie ermöglicht es, Fehler und Probleme bei der Umformung, wie zum Beispiel Falten, Bodenreißer oder Mikrorisse im Gefüge, bereits frühzeitig am Computer zu erkennen. Dies hilft die Zahl der notwendigen physischen Prototypenwerkzeuge zu reduzieren und Kosten zu sparen. Die gängige Vorgehensweise bei FE-basierten Umformprozessanalysen entspricht, wie in Bild 21 gezeigt, im Wesentlichen den bekannten Prinzipien des Preprocessing, Solving und Postprocessing (siehe [GMS07; Vaj09; DB10]).

Beim *Preprocessing* wird ein FEM-Modell des Fertigungsprozesses erzeugt. Bei Umformsimulationen müssen dazu Geometrien, Materialeigenschaften, Rand- und Anfangsbedingungen sowie die Modelldiskretisierung festgelegt werden. Die Geometrien werden üblicherweise aus einem CAD-Modell importiert. Zwar verfügen die meisten FEM-Simulationsprogramme über Funktionen zur Erstellung von Geometrien, jedoch sind in der Regel bereits CAD-Modelle von Betriebsmittelkonsteuren vorhanden. Die Randbedingungen spezifizieren den Prozess genauer. Typische festzulegende Rand- und Anfangsbedingungen bei Umformsimulationen sind z. B. die Eigenschaften des Werkzeugs und der

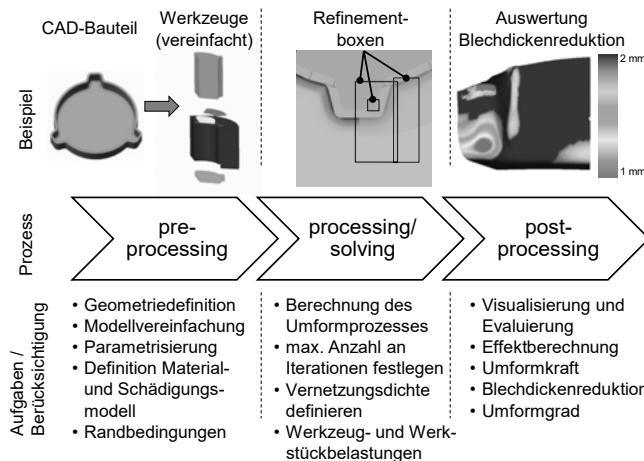


Bild 21: FEM-Simulationsprozess in der Umformtechnik entsprechend [DB10].

Umformmaschine (z. B. Vorschub, Kinetik, Temperatur). Weiterhin müssen die Reibung zwischen den Kontaktpartnern, der Wärmeübergang in der Wirkfuge zwischen Werkzeug und Werkstück sowie die Wärmeabstrahlung an die Umgebung berücksichtigt werden. Die Diskretisierung betrifft die Geometrien. Bei der Diskretisierung werden Starrkörper, die in erster Näherung undeformierbare Geometrien (z. B. Umformwerkzeuge) repräsentieren, mit einem Oberflächennetz und deformierbare Geometrien mit Volumennetzen versehen. Das gesamte FEM-Modell wird in ein Gleichungssystem für die unbekannten Zustandsvariablen überführt.

Beim *Processing* wird das zuvor erstellte Gleichungssystem gelöst (engl. *solving*) und die benötigten Größen wie z. B. Verschiebungen, Verzerrungen, Spannungen, Energien usw. ermittelt. Bei Umformsimulationen müssen typischerweise große Verzerrungen abgebildet werden, weshalb derartige Problemstellungen als nichtlinear anzusehen sind [Vaj09]. Die Lösung derartiger Gleichungssysteme kann durch direkte Iterationsverfahren oder Verfahren nach Newton-Raphson geschehen (siehe [BB09; JL13]). Das Lösen der Gleichungen läuft automatisch ab und erfordert in der Regel keinen Eingriff des Anwenders.

Der letzte Schritt bei FEM-Simulationen ist das *Postprocessing*. In diesem Schritt werden die Ergebnisse der Simulationen ausgewertet. Dabei müssen die Ergebnisse auf Plausibilität geprüft werden, weil die Festlegung der Randbedingungen und die Diskretisierung ein hohes Fehlerpotential mit sich bringen. Idealerweise kann eine Simulation mit einem Versuch verifiziert werden. Sind große Unterschiede zwischen Versuchs- und Simulationsergebnisse vorhanden, muss das Modell im Preprocessing angepasst werden. Hinsichtlich der Design for Manufacture Unterstützung durch Umformsimulationen lässt sich das Vorgehen nach dem derzeitigem Stand der Technik eher mit dem „try-and-error“ Prinzip beschreiben. Sofern sich aus dem Ergebnis der Simulation ableiten lässt, dass bestimmte Kriterien

nicht erfüllt sind (z. B. zu hohe Werkzeugbelastungen, zu hohe Umformkräfte, unzureichende Kavitätsfüllung) muss eine Bauteilform als nicht-fertigungsgerecht eingestuft werden. Prozessingenieur und Konstrukteur diskutieren anschließend über mögliche Änderungsmaßnahmen, um zu einem stabilen, die Qualitätsanforderungen erfüllenden Prozess zu gelangen. Verfügt der Umformtechniker über ausreichendes Erfahrungswissen, so wird er sowohl dem Konstrukteur Hinweise für eine Änderung der Bauteilgestalt geben als auch Prozessrandbedingungen (z. B. Reibung, Umformstufen) anpassen können.

Der gesamte Simulationsprozess kann zudem durch die Methodik der statistischen Versuchsplanung (engl. Design of Experiment, DoE) unterstützt werden. Ziel ist die systematische Identifikation der relevanten, unabhängigen Systemgrößen (z. B. Werkzeuggeometrie, tribologische Verhältnisse in der Wirkfuge Werkzeug-Werkstück) sowie die quantitative Beschreibung ihres Einflusses auf abhängige Größen (z. B. Werkzeugbeanspruchung, Lebensdauer, Bauteilqualität). Grundlagen zur statistischen Versuchsplanung und zur Erstellung sowie Auswertung von Versuchsplänen sind der einschlägigen Literatur zu entnehmen (z. B. [Fis35; Mat05; SvH10]).

2.4 Wissensentdeckung in Datenbanken - KDD

Der Einsatz von Datenbanken ermöglicht seit den 1960er Jahren die Verwaltung und Bereitstellung umfangreicher Datenmengen. Gleichzeitig können mit Methoden der Statistik und der Rechnerunterstützung beim maschinellen Lernen diesen großen Datenmengen effizient ausgewertet und Informationen sowie Wissen extrahiert werden. An der Schnittstelle zwischen Datenbereitstellung auf der einen und Datenauswertung durch Statistik und maschinelles Lernen auf der anderen Seite hat sich zu Beginn der 1990er Jahre das Forschungsfeld der *Wissensentdeckung in Datenbanken* (engl. Knowledge Discovery in Databases, KDD) entwickelt (s. Bild 22) [FPSS96].

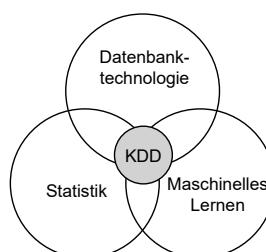


Bild 22: Knowledge Discovery in Databases als Schnittmenge der Datenbanktechnologie, der Statistik und des maschinellen Lernens.

In der anwendungsorientierten Literatur wird KDD teilweise mit dem Begriff des *Data-Mining* gleichgesetzt [AN00; Ott04]. Die vorliegende Arbeit folgt jedoch der Lesart von

[PS91] und FAYYAD ET AL. [FPSS96], wonach KDD den gesamten Prozess von der Rohdatensammlung bis hin zur Extraktion und Verteilung des relevanten Wissens bezeichnet. Data-Mining hat hier einen eher technologischen Charakter und umfasst in diesem Prozess die Anwendung von Algorithmen und Methoden aus den Bereichen Statistik, Mustererkennung und maschinelles Lernen. Der KDD Prozess nach FAYYAD ET AL. mit dem integrierten Schritt des Data-Minings ist Gegenstand der folgenden Abschnitte. Beispiele für die erfolgreiche Anwendung von KDD im Kontext des Produktentstehungsprozesses werden im Abschnitt 2.5 behandelt.

2.4.1 Der KDD Prozess nach Fayyad

KDD beschreibt einen *nichttrivialen* mehrstufigen Prozess (s. Bild 23), in dem ein Datenanalytiker das Ziel verfolgt, *neues, nützliches und interessantes* Wissen in großen Datenmengen zu finden und in *nachvollziehbarer* Form dem Anwender bereitzustellen. Die kursiv hervorgehobenen Adjektive sind mitunter sehr kontextabhängig und müssen daher kurz erläutert werden.

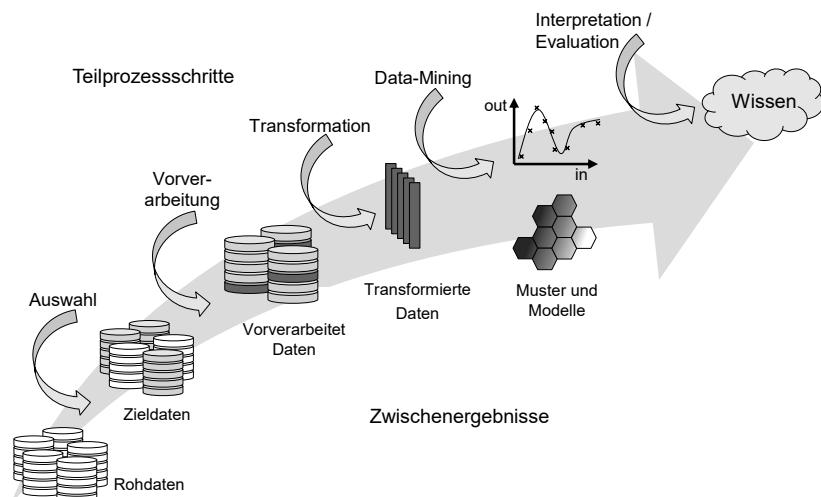


Bild 23: Der KDD-Prozess mit den einzelnen Prozessschritten und Zwischenergebnissen nach [FPSS96]

Nichttrivial ist dieser Prozess weil beispielsweise passende Such- und Schlussfolgerungsalgorithmen angewandt werden müssen. Hierin liegt ein wichtiger Unterschied zu einfachen Datenbankabfragen oder reinen statistischen Auswertungen, bei denen z. B. für einen bekannten Systemparameter lediglich der Mittelwert aus einer Versuchsreihe berechnet wird [AN00]. FAYYAD ET AL. beschreiben den KDD Prozess zudem als in hohem Maße interdisziplinär und nur bewältigbar, wenn neben Experten aus Informatik, Mathematik oder

maschinellem Lernen auch die Domänenexperten miteinbezogen werden, die letztendlich vom Wissen profitieren sollen. In einem industriellen Unternehmen können dies die Mitarbeitenden aus dem Vertrieb, dem Controlling, der Fertigung oder der Produktentwicklung sein. Ein KDD Prozessmanager kann in diesem Sinne der in Abschnitt 2.1.4 beschriebenen Rolle des Wissensingenieur gleichgesetzt werden.

Das durch KDD erhobene Wissen umfasst meist implizite und bis dato unbekannte Zusammenhänge. Die Ergebnisse des Data-Mining Schritts beinhalten in der Regel viele Informationen und Modelle, die miteinander verglichen werden müssen, um das plausibelste zu finden. Außerdem ist ein Vergleich mit bereits bekanntem oder gespeichertem Wissen notwendig. Was nützliches oder interessantes Wissen ausmacht, hängt von der jeweiligen Anwendungsdomäne ab. Im betriebswirtschaftlichen Kontext wird Wissen über den Gewinn, die Kosten oder Kundenkennzahlen von Interesse sein. In der Produktionstechnik werden die Zusammenhänge zwischen Einfluss- und Ergebnisgrößen eines Fertigungsprozesses von Interesse sein. Ein Beispiel ist die Auswirkung der Menge an Heiz- und Kühlmitteln sowie Sauerstoff auf den Herstellungsprozess von Rohstahl in einem Konverter [Sch14b]. Bei KDD Prozessen mit wissenschaftlichem Hintergrund kommen zumeist qualitative Kriterien wie Genauigkeit auf der einen und Allgemeingültigkeit auf der anderen Seite zum Einsatz.

Die Sicherstellung der Nachvollziehbarkeit der Ergebnisse beeinflusst in hohem Maße deren nachhaltige Berücksichtigung in der Anwendungsdomäne. Für den Benutzer müssen Wert und Nutzen des KDD Projektes erkennbar sein. Die gefundenen Information sind anschaulich sowie mit möglichst geringem Aufwand zugänglich zu machen.

2.4.2 Darstellung der KDD-Prozessschritte

Einarbeitung und Datenbeschaffung

Zu Beginn eines KDD Prozesses wird relevantes Hintergrundwissen über die Anwendungsdomäne und ein Verständnis für die Problem- und Aufgabenstellung aufgebaut. Selten gehören alle Teilnehmenden eines KDD Projektes dem gleichen Fachbereich an und Missverständnissen oder Fehlinterpretationen muss vorgebeugt werden. Der Aufgabenstellung entsprechend muss den insgesamt zur Verfügung stehenden Datenquellen ein Grunddatensatz entnommen werden. Hierzu werden die Daten abhängig von den durchzuführenden Analysen ausgewählt. Dabei können Methoden des Online Analytical Processing (OLAP) eingesetzt werden, um durch Variablen- oder Fallselektion einen problemadäquaten Datensatz zu erzeugen [Pyl99; TSK10]. Zur Gewinnung analysierbarer Daten können unterschiedliche Quellen in den einzelnen Unternehmensbereichen in Betracht gezogen werden [HKMW01]:

- Produktion (z. B. Maschinenparameter, Einsatzplanungsdaten, Auslastungen)
- Qualitätsmanagement (z. B. Protokolle, Kundenbeschwerden, Änderungsanträge)
- Vertrieb (z. B. Absatzzahlen, Kundenstammdaten, Marktentwicklungen)

- Produktentwicklung (z. B. Experimente, PDM-Datensätze, Prüfstandsdaten, Echtzeitmessungen)

Bei vielen Daten-Mining Verfahren ist die Grundvoraussetzung, dass die zu analysierenden Daten in einer vollständig ausgefüllten Datentabelle, der sogenannten „Flat Table“ vorliegen. Die Spalten beschreiben üblicherweise die Eingangsgrößen (Attribute) und Ausgangsgrößen (Label) des Systems und die N Zeilen entsprechen den Ausprägungen zu bestimmten Abtastzeiten oder vorher definierten Versuchsdurchführungen (Instanzen, Tupel). Häufig liegen die Daten verteilt vor (z. B. Prozessdatenbank, PDM-System, lokale Verzeichnisse) und müssen auf einer Plattform physisch zusammengeführt und in einer kohärenten Flat Table zusammengefasst werden.

Datenvorbereitung

Die Daten müssen vorverarbeitet und bereinigt werden, um einen möglichst auswertungsfähigen Datensatz zu generieren. Zu den häufigsten Schritten gehören nach [TSK10] z. B. die Eliminierung fehlender Werte und die Datenreduktion. Ein Großteil der Data-Mining Algorithmen kann keine Fehlstellen verarbeiten, daher müssen diese Lücken gefüllt oder die entsprechenden Tupel entfernt werden. Die Verwendung des Mittelwerts oder des Medians aller bekannten Einträge des entsprechenden Attributs ist die schnellste Möglichkeit eine Fehlstelle zu füllen, allerdings verzerrt dies die Daten. Empfohlen wird daher, den *wahrscheinlichsten* Wert anzunehmen. Dieser lässt sich zum Beispiel mit einem Bayes-Schätzer (siehe [Koc00]) bewerkstelligen.

Die Datenreduktion hat das Ziel große Datensätze zu verkleinern, ohne dabei den Informationsgehalt zu reduzieren. Zu den effektivsten Methoden zählen die *Dimensionsreduzierung* (z. B. Attributselektion, Hauptkomponentenanalyse) und die *Datenkompression* (z. B. String-Kompression) [HKP12]. Die Datenkompression verfolgt das Ziel einen verkleinerten Datensatz auf Basis der Originaldaten zu erzeugen. Sie wird im weiteren Verlauf nicht näher betrachtet.

Bei der Dimensionsreduzierung wird die Anzahl A der „zufälligen“ Eingangsgrößen (Attribute) im Datensatz reduziert und nur die relevanten Attribute berücksichtigt. Zufällig bedeutet hier, dass trotz variierender Werte kein signifikanter Einfluss auf die Zielgröße (Label) zu erkennen ist. Die Attributselektion kann in Form einer schrittweisen Vorwärts- oder Rückwärtsselektion erfolgen. Die Vorwärtsselektion beginnt mit einer leeren Attributmenge und erhöht sie iterativ unter Beobachtung der Modellperformanz (s. Tabelle 4) auf die ursprüngliche Menge A . Dann wird unter den restlichen Attributen wieder eines ausgewählt und dem vorhandenen Set hinzugefügt. Dieser Vorgang wird so lange wiederholt, bis sich keine Verbesserung der Performanz mehr einstellt. Die Rückwärtsselektion arbeitet nach dem gleichen Schema, beginnt jedoch mit der maximalen Attributmenge A und reduziert diese dann iterativ. [KNW96; LM98]

Die Hauptkomponentenanalyse nähert eine bestimmte Anzahl A von variierenden Attributen durch eine geringere Anzahl l möglichst aussagekräftiger Linearkombinationen dieser

Attribute an. Anders als die Attributselektion, bei der mindestens eine Teilmenge der Attribute aus A im Datensatz D verbleibt, kombiniert die Hauptkomponentenanalyse die „Essenz“ der Attribute durch die Bildung eines alternativen, kleineren Sets an Variablen. Die Hauptkomponentenanalyse kann für geordnete und ungeordnete Attribute gleichermaßen angewandt werden. Ebenso ist auch die Verarbeitung kleiner oder verzerrter Datensätze möglich. [FHT96; HKP12]

Data-Mining

Der Schritt des Data-Minings umfasst bei FAYYAD ET AL. die Festlegung der Data-Mining Aufgabenstellung, die Auswahl eines entsprechenden Data-Mining Algorithmus und dessen Anwendung auf den vorbereiteten Datensatz [FPSS96]. Die Aufgabenstellung wird im Wesentlichen von den Zielen des KDD Projektes beeinflusst. Es muss geklärt werden, welchem Zweck die gefundenen Zusammenhänge dienen und in welcher Form sie repräsentiert werden sollen. Abhängig davon ist eine Data-Mining Methode entsprechend Bild 24 auszuwählen.

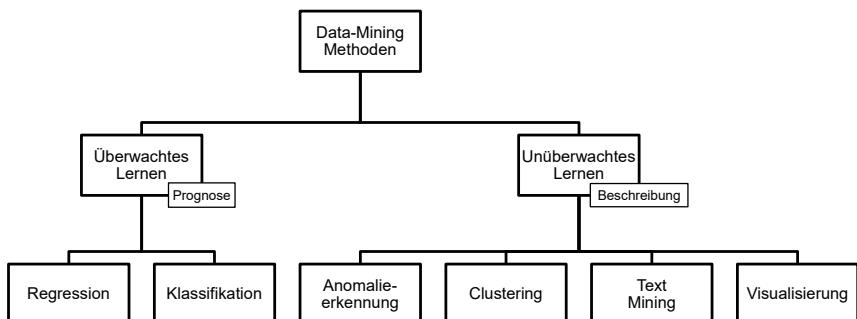


Bild 24: Eingruppierung der Data-Mining Methoden nach [TSK10; FPSS96].

Ziel von **Prognosemethoden** ist die Vorhersage eines Ergebnisattributs basierend auf den zu einem Eingabevektor zusammengefassten Werten von Eingangsattributen. Die dazu erzeugten Modelle werden gelernt bzw. trainiert, wobei der Trainingsprozess häufig „überwacht“ wird. Dies verhindert eine Unter- bzw. Überanpassung des Modells. Bei den **beschreibenden Methoden** ist die Identifikation von Mustern in den Daten das Ziel. So können unbekannte Zusammenhänge wie Trends, Gruppierungen diskreter Werte (Cluster), Korrelationen oder Anomalien aufgedeckt werden. Eine Überwachung des Prozesses wie bei den Prognosemethoden findet nicht statt.

Die **Regression** hat das Ziel, ein Label mit numerischem, realem Wert vorherzusagen. Die Ergebnisse mehrerer Vorhersagen können entsprechend einer Ordinalskala in Relation zueinander gebracht werden. Die Regression findet vielfach Anwendung in der Messdatenauswertung zur Interpolation zwischen Messpunkten oder zur Vorhersage einer Prozessgröße.

ße in der chemischen Industrie. Eine gängige Unterscheidung betrifft zudem die Annahme bezüglich eines linearen oder nicht-linearen Systemverhaltens. [CPSK07]

Bei einer **Klassifikation** ist das zu prognostizierende Label dadurch gekennzeichnet, dass es in mindestens zwei diskreten Ausprägungen vorliegt. Die Ordnung der Klassen hat dabei keine Bedeutung. Charakteristische Beispiele sind die Klassen *sicher* und *risikoreich* bei Finanzgeschäften, der Einsatz eines der Medikamente *A*, *B* und *C* oder eben *fertigungsrecht* oder *nicht fertigbar* in der Produktionstechnik.

Bei den Algorithmen der **Anomalieerkennung** (auch Ausreißererkennung) ist das Ziel, Datentupel zu erkennen, deren Charakteristiken signifikant vom Rest der Daten abweichen. Die besondere Herausforderung besteht darin zu verhindern, dass normale Objekte fälschlicherweise als „anormal“ gekennzeichnet werden. Eine hohe Aufdeckungsrate und eine möglichst geringe Fehlalarmrate kennzeichnen gute Algorithmen. Typische Anwendung sind das Aufdecken von Kreditkartenbetrug, Netzwerkinfiltrierungen oder andere Fehlfunktionen in einem komplexen, schwer zu überwachenden System. [TSK10]

Beim **Clustering** wird im Gegensatz zu den bisher beschriebenen Methoden kein Klassenlabel zur Analyse herangezogen. Daher eignen sich Cluster-Algorithmen besonders dann, wenn noch gar keine möglichen Klassen bekannt sind und der Anwender erste Hinweise benötigt. Die Gruppierung erfolgt nach dem Prinzip der maximalen Gleichheit innerhalb einer Gruppe und der minimalen Ähnlichkeit zwischen den Gruppen. Cluster können beispielsweise im Marketing unterschiedliche Kundengruppen beschreiben, denen sich ein Unternehmen zuvor noch nicht bewusst war. [HKP12]

Das **Text-Mining**, die Suche nach Mustern in Texten, wird der Vollständigkeit halber genannt, gilt streng genommen jedoch als eigenes, interdisziplinäres Forschungsfeld [FD95]. Die wichtigsten Schnittmengen besitzt es mit den Bereichen des Information Retrieval, dem maschinellen Lernen, der Statistik und insbesondere der linguistischen Informatik. Ein zentraler Unterschied zum Data-Mining besteht in der wesentlich geringeren Strukturierung von Textdaten. Beim Text Mining werden daher die Primärdaten zunächst stärker strukturiert, um sie mit Verfahren des Data-Mining zu erschließen [HNP05]. Text Mining gilt als relativ junges Forschungsgebiet, sein aktueller Stand ist vergleichbar mit dem des maschinellen Lernens in der Mitte der 1980er Jahre [WEH11].

Die verschiedenen Methoden der **Visualisierung** sollen dem Anwender helfen auch komplexe Zusammenhänge in den Daten bildhaft darstellbar und für Berichte, Bewertungen und Managementaufgaben nutzbar zu machen. Für die Abbildung multidimensionaler Daten, wie sie in relationalen Datenbanken vorkommen, können pixelorientierte Techniken, geometrische Projektionen, hierarchische oder graphenbasierte Methoden genutzt werden. [TSK10; FGW02]

Ist die Data-Mining Aufgabenstellung geklärt und sind die entsprechenden Algorithmen ausgewählt, werden diese auf die vorbereiteten Datensätze angewandt. Hierzu stehen dem Anwender diverse kommerzielle und frei verfügbare Programme zur Verfügung, die teilweise gängige Prozessmodelle bei der Modellierung berücksichtigen.

Evaluation, Interpretation und Nutzung

Nach der Analyse des Datensatzes sind die gefundenen Zusammenhänge und Muster zu evaluieren. Im Folgenden wird lediglich die Evaluation von Prognosemethoden für Regressionsprobleme beschrieben, da unüberwachte Verfahren (s. Abb. 24) und Klassifikation in dieser Arbeit keine Anwendung finden. Der Anwender benötigt mindestens ein Gütemaß, um eine Aussage über die Qualität der gefundenen Prognosemodelle treffen zu können. Hier muss unterschieden werden zwischen den Gütekriterien selbst und dem Vorgehen, wie diese zu bestimmen sind.

Ein einfaches Vorgehen ist das sogenannte *Holdout* Verfahren [WEH11]. Hierbei wird der gesamte zur Verfügung stehende Datensatz zufällig in einen Trainings- und einen Testdatensatz unterteilt (Sampling). Der Data-Mining Algorithmus analysiert nach dem Sampling nur den Trainingsdatensatz und erzeugt auf diesem reduzierten Teildatensatz ein Prognosemodell. Für die Datentupel im Testdatensatz werden mit Hilfe dieses Prognosemodells die jeweiligen Testlabel berechnet und mit den realen Labelwerten im Testdatensatz verglichen. Auf dieser Basis kommt ein pessimistischer Schätzwert der Modellperformanz (s. u.) zustande. Das Verhältnis von Trainings- und Testdatensatz beträgt üblicherweise zwischen 2:1 und 3:1 [HKP12]. Zur Reduzierung der Varianz des Performanzkennwerts wird eine k -fache Wiederholung des Verfahrens mit anschließender Mittelwertbildung aus den einzelnen Performanzen empfohlen (wiederholtes, zufälliges Sampling) [DGB07; Smy96]. Dieses *Monte-Carlo Sampling* eignet sich jedoch nur für hochdimensionale Datensätze mit $A \geq 750$ Attributen bzw. $N \geq 300$ Datensätzen [MSP05].

Für kleinere Datensätze hat sich die *Kreuzvalidierung* als geeigneter erwiesen [Sto74]. Hierbei wird der Datensatz D zufällig in k annähernd gleich große Teildatensätze D_1, D_2, \dots, D_k aufgeteilt. Das Trainieren und Testen des Prognosemodells wird nun k mal wiederholt, wobei in jeder Iteration i der Teildatensatz D_i als Testdatensatz und die verbleibenden $k - 1$ Datensätze als Trainingsdatensatz genutzt werden. Aus den ermittelten k einzelnen Performanzkennwerten wird anschließend ein Mittelwert berechnet. Anders als beim Holdout oder beim Monte-Carlo Sampling ist jedes Tupel gleich oft ($k - 1$)-mal im Trainingsdatensatz und einmal im Testdatensatz vertreten. Bild 25 beschreibt das Vorgehen bei der Kreuzvalidierung für $k = 5$. Ein Wert der sich als Kompromiss zwischen verlässlicher Performanzberechnung und Rechendauer erwiesen hat ist $k = 10$ [BS92].

Spezielle Formen der Kreuzvalidierung sind die Leave-One-Out Kreuzvalidierung und das 0.632-Bootstrap-Verfahren. Bei der *Leave-One-Out Kreuzvalidierung* wird mit $k = N$ jedes Datentupel einmal als Testdatentupel verwendet, während der verbleibende Datensatz zum Trainieren genutzt wird. WITTEN empfiehlt sie für kleine Datensätze [WEH11], denn bei sehr großen Datensätzen verlängert sich die Rechenzeit enorm und der Performanzkennwert unterliegt nach [Koh95] einer hohen Streuung. Das *0.632-Bootstrap-Verfahren* eignet sich ebenfalls für kleinere Datensätze und basiert auf dem Prinzip „Ziehen mit Zurücklegen“. Für ein Datentupel, das dem Trainingsdatensatz bereits einmal zugeordnet wurde, besteht anschließend die Chance, dass es ein weiteres Mal in den Trainingsdatensatz aufgenommen wird. Die verbleibenden Datentupel werden dem Testdatensatz zugeordnet [HKP12]. Die

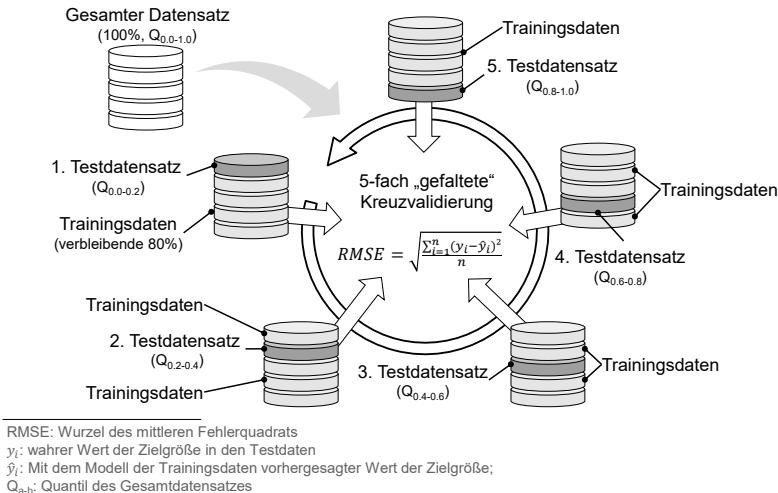


Bild 25: Das Verfahren der Kreuzvalidierung für $k = 5$.

Bezeichnung 0.632 leitet sich aus der Wahrscheinlichkeit $\frac{1}{N}$ ab, dass ein Tupel ausgelost bzw. $1 - \frac{1}{N}$, dass es nicht gelost wird. Da N -mal aus dem N -großen Datensatz gezogen wird, ergibt sich mit der Euler'schen Zahl e

$$P = \left(1 - \frac{1}{N}\right)^N \approx e^{-1} = 0,368 \quad (2.1)$$

die Wahrscheinlichkeit P , dass ein Tupel nicht dem Trainingsdatensatz zugeordnet wird, bzw. $1 - 0,368 = 0,632$, dass dieser Fall eintritt [B⁺03; Joh01].

Die Berechnung der Gütekriterien erfolgt bei jedem Vorgehen entsprechend den Formeln aus Tabelle 4. Da es sich, bis auf den CoP, dabei um berechnete Abweichungen handelt, liegt eine hohe Güte bei niedrigen Fehlerwerten vor. p_i entspricht den auf Basis des jeweiligen Trainingsdatensatzes vorhergesagten (engl. **predicted**) Werten des Labels, a_i den tatsächlichen (engl. **actual**) Labelwerten und \bar{a} dem Mittelwert des Labels im Trainingsdatensatz. Es muss an dieser Stelle erwähnt werden, dass jedes Gütekriterium nur eine Schätzung der Modellperformanz darstellt, da das finale Modell nach der Validierung mit allen Datentupel trainiert wird.

Eine pauschale Empfehlung welches Kriterium bei der Interpretation verwendet werden soll, ist in der Literatur nicht zu finden. Vielmehr wird auf die unterschiedlichen Eigenschaften und den notwendigen Vergleich mit dem Data-Mining Ziel hingewiesen [WEH11]. Dennoch lässt sich feststellen, dass quadratische Kriterien, sowie deren Wurzeln, große Abweichungen stärker gewichten als absolute Kennzahlen. Einige Werte besitzt zudem die gleiche Dimension wie das Label, was deren Interpretation erleichtert. Relative Kennzahlen

Tabelle 4: Gütekriterien zur Beurteilung von Regressionsmethoden nach [WEH11].

Kriterium	Berechnung (engl. Abkürzung)
Mittleres Fehlerquadrat	$MSE = \frac{1}{N} \sum_{i=1}^N (p_i - a_i)^2$
Wurzel des mittleren Fehlerquadrats	$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (p_i - a_i)^2}$
Relativer quadratischer Fehler	$RelSE = \sum_{i=1}^N \frac{(p_i - a_i)^2}{(p_i - \bar{a})^2}$
Wurzel des rel. quadratischen Fehlers	$RRelSE = \sqrt{\sum_{i=1}^N \frac{(p_i - a_i)^2}{(p_i - \bar{a})^2}}$
Relativer absoluter Fehler	$RelAE = \sum_{i=1}^N \frac{ p_i - a_i }{ p_i - \bar{a} }$
Mittlerer absoluter Fehler	$MAE = \frac{1}{N} \sum_{i=1}^N (p_i - a_i)$
Coefficient of Prognosis ([MW08; MW11])	$COP = 1 - \frac{SS_{E_{Pred}}}{SS_T} = 1 - \frac{\sum_{i=1}^N (p_i - a_i)^2}{\sum_{i=1}^N (a_i - \bar{a})^2}$

sind in der Lage die subjektive, „gefühlte“ oder erwartete Vorhersagbarkeit eines Lables zu objektivieren. Angenommen das Label im Trainingsdatensatz zeigt eine geringe Streuung, dann wird der Datenanalytiker eine gute Vorhersagbarkeit erwarten. Die tatsächliche Performanz hängt jedoch auch sehr stark vom eingesetzten Algorithmus und der Datenqualität ab.

Zum Abschluss eines KDD Prozesses muss das erhobene Wissen in seiner jeweiligen Form gesichert und zur Verfügung gestellt werden. Je nach Anforderung kann dies durch die Erstellung und Verteilung eines einfachen Reports geschehen [She00]. Denkbar ist aber auch die Skalierung eines prototypischen Data-Mining Prozesses und dessen Implementierung im gesamten Unternehmen zur ständigen Überwachung der internen Datenbanken (z. B. PDM- oder ERP-System) [Ott04].

2.5 Einsatz von KDD im Produktentstehungsprozess

Die KDD Technologie hat seit dem Beginn ihrer Entwicklung schrittweise unterschiedlichste Anwendungsfelder erschlossen und zu signifikanten Verbesserungen beigetragen [CFPS99; LK01; SBM⁺12]. Die folgenden Abschnitte stellen KDD-basierte Ansätze aus dem Stand der Forschung entlang des Produktentstehungsprozesses vor.

2.5.1 KDD in der Anforderungsanalyse

Das Erarbeiten der Produktanforderungen stellt einen der ersten Schritte im Produktentwicklungsprozess dar, wobei Anforderungen bei neuen Produkten durch Markt- und Zielgruppenanalysen oder Interviews mit Kunden erhoben werden [FG13] können. Für abgeleitete Versionen oder Varianten eines bestehenden Produktes kann es zielführend

sein, Änderungsanträge (engl. Engineering Change Order, ECO) zu analysieren und die Änderungsgründe in Form geänderter oder neuer Anforderungen zu erfassen [Lin09]. Die manuelle Analyse von ECOs ist jedoch mit sehr hohem Zeit- und Kostenaufwand verbunden.

Dieser Problemstellung widmen sich die Arbeiten von SHARAFI [Sha13] sowie ELEZI ET AL. [ESM⁺11]. In einer umfangreichen empirischen Studie werden 53.000 ECO analysiert und hinsichtlich der unterschiedlichen Änderungsgründe kategorisiert. Die Analyse der ECO Daten erfolgt mit Methoden des Text-Minings. Nach statistischen Untersuchungen hinsichtlich Auswirkung, Anzahl und Länge der notwendigen Iterationen zur Erfüllung eines ECO wird der gesamte Datensatz, $N \approx 1.244.010$ Einträge für 53.000 ECOs, in das Data-Mining Tool RapidminerTM geladen. Bild 26 zeigt den gesamten KDD-Prozess von [ESM⁺11] inklusive des geschachtelten Text-Mining Schritts. Zu Beginn erfolgt eine Vorfilterung hinsichtlich der relevanten ECO die iterativ mehr als einmal durchgeführt werden mussten. Zur Verarbeitung durch Text-Mining Algorithmen werden Dateneinträge, denen der Typ *nominal* oder *numerisch* zugeordnet ist, in den Datentyp *String* umgewandelt. In diesen Strings werden Umlaute bereinigt, Groß- in Kleinbuchstaben umgewandelt und der Text für das eigentliche Text-Mining gebildet.

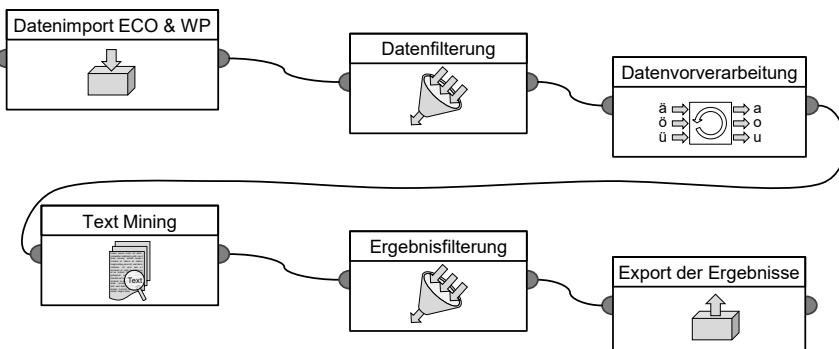


Bild 26: Text-Mining Prozess zur Analyse von ECOs nach [ESM⁺11].

Eine Tokenisierung teilt den gesamten Text in sogenannte Tokens auf. Dabei handelt es sich um Einheiten auf der Wortebene, in selten Fällen auch um Sätze oder Satzfragmente [Nor12; MM04]. Auch wenn die Aufteilung in einzelne Wörter zunächst simpel erscheint, so lassen sich aus der Häufigkeitsverteilung von Tokens bereits wertvolle Rückschlüsse auf die Art des Textes ziehen [HNP05], bspw. wenn viele Fachbegriffe einer bestimmten Domäne darin vorkommen. Von den gefundenen Tokens werden alle mit weniger als drei Buchstaben entfernt, da sie als nicht ausschlaggebend für die Kategorisierung der ECOs angesehen werden. Zudem wurde in einem vorherigen Schritt eine Liste an Wörtern erstellt, die für die Analyse von ELEZI ET AL. [ESM⁺11] nicht relevant sind. Ein Filter entfernt auch diese Tokens. Ein weiterer wichtiger Schritt ist das sogenannte Stemming. Hierbei werden

verschiedene morphologische Ausprägungen eines Tokens (Wort) auf ihren gemeinsamen Wortstamm zurückgeführt [HK14]. Zum Beispiel wird aus den Token *Häuser* und *Hauses* der Stamm *Haus*. Anschließend werden aus den Stems N-Gramme mit $N = 2$ gebildet, wobei ein N-Gramm N aufeinanderfolgende Tokens zusammenfasst. Zu diesen 2-Grammen existiert eine Liste an speziellen Kombinationen die auf die Zurückweisung einer erarbeiteten Lösung und somit zu einer erneuten Iteration für einen ECO schließen lässt. Diese Kombinationen werden jedoch nicht weiter behandelt, da sie vermutlich wichtiges Firmenwissen darstellen.

Wie bereits im Abschnitt 2.4.2 zu Text-Mining dargestellt, folgt dieser Textvorverarbeitung ein Data-Mining Schritt, um die relevanten Muster in den Daten zu finden. ELEZI ET AL. prüfen hierzu verschiedene Ansätze (z.B. Clusteranalysen, Ontologieerzeugung, Schlüsselwortsuche), kommen jedoch zu dem Schluss, dass in ihrem Fall nur eine manuelle Klassifizierung zu akzeptablen Ergebnissen führt. Abschließend werden sechs Kategorien von Gründen identifiziert, die in der Vergangenheit zu iterativen Änderungen geführt haben. In einer weiteren Kategorie werden nicht näher begründete ECOs zusammengefasst (s. Bild 27).

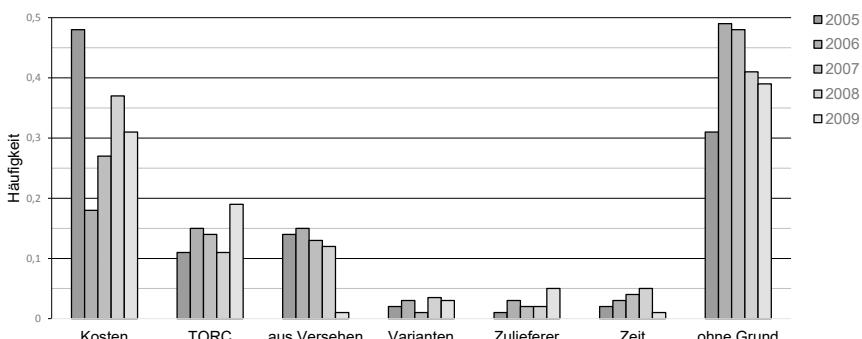


Bild 27: Von Elezi et al. identifizierte Kategorien für die Zurückweisung von geänderten Konzepten nach einem Änderungsantrag (ECO) [ESM⁺11].

2.5.2 Wissensbasierte Schichtentwicklung

Die Zuordnung von tribologischen Schichtsystemen in den Produktentstehungsprozess erscheint zunächst irreführend, galt die Tribologie lange als eigenständiges Forschungsgebiet [Dow98]. Im Kontext einer modernen, interdisziplinären Ingenieurwissenschaft wird ihre Bedeutung heute jedoch weiter gefasst [CGH⁺10]. Beschichtungen zur gezielten Einstellung von Oberflächeneigenschaften und Produktverhalten werden heutzutage als „Maschinenelemente“ ähnlich zu Schrauben, Bolzen oder Kupplungen angesehen [Het14; Tre09]. Aufgrund der komplexen Mechanismen und Wechselwirkungen im tribologischen System ist die frühzeitige Analyse des Reibungs- und Verschleißverhaltens durch Experimente mit

hohem Aufwand verbunden. SCHULZ verfolgt in seiner Arbeit einen wissensbasierten Ansatz zur Vorhersage des Reibverhaltens im tribologischen System Nocke/Tassenstöbel und erhebt das relevante Wissen durch Data-Mining aus tribologischen, physikalischen, mechanischen und chemischen Analysedaten [Sch13].

In den Analysedaten werden die Attribute der Beschichtung, verwendeter Schmieröle sowie der Versuchsbedingungen am Tribometer erfasst. Insgesamt 52 Attribute, verteilt auf 497 Einzelmessungen werden in einer Datenmatrix kombiniert. In diesem Datensatz sind in einem ersten Vorprozess fehlende Attributwerte durch den jeweiligen Attributmittelwert zu ersetzen, nominale Attribute in numerische umzuwandeln und alle Daten durch z-Transformation zu normieren. Anschließend erfolgt eine Hauptkomponentenanalyse zur Reduzierung der Attributanzahl.

In den Arbeiten von SCHULZ liegt eine Prognoseproblemstellung zur Vorhersage eines numerischen Labels (Reibzahl μ) vor. Daher entscheidet er sich für den Einsatz sogenannter künstlicher neuronaler Netze⁹ (KNN) [SBM⁺12]. Zunächst wird die Netztopologie, die Anzahl der versteckten Schichten und der enthaltenen Neuronen, hinsichtlich eines niedrigen und wenig streuenden Vorhersagefehlers optimiert. In dieser Topologie wird der Informationsgehalt der Daten durch eine integrierte Gitteroptimierung gleichmäßig über das KNN verteilt. Die Gitteroptimierung variiert die für das Training des KNN relevanten Parameter wie Lernrate, Trainingszyklen, Momentum und Decay in vorher festgelegten Bereichen und wählt die Parameterkombination aus, die zum geringsten Vorhersagefehler führt. Die von SCHULZ erzielten Ergebnisse sind zufriedenstellend hinsichtlich der erreichten Vorhersagefehler (s. Tabelle 5).

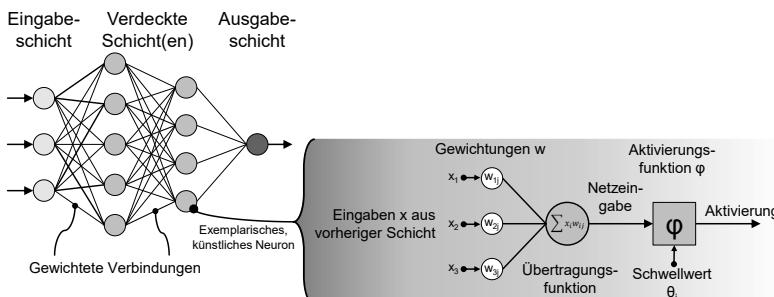


Bild 28: Darstellung eines künstlichen neuronalen Netzes wie es in [Sch13] zum Einsatz kommt.

Ein kurzer Vergleich von KNN mit anderen Data-Mining Algorithmen unterstützt die Argumentation von SCHULZ, dass ein KNN für seine Problemstellung die passendste Modellierungsmethode darstellt (s. Bild 29). Dies ist für das von ihm skizzierte praxisnahe Umfeld eines Automobilzulieferers eine pragmatische Annahme. Eine detaillierte, statistische Un-

⁹ Nähere Informationen zu KNN und deren Funktionsweise können bspw. [KBK⁺13] entnommen werden.

Tabelle 5: Ergebnisse von Schulz [Sch13].

Validierungsmethode	RMSE	abs. Fehler	rel. Fehler	Korr.-koeff.
Kreuzvalidierung mit Originaldatensatz ($k = 10$)	$0,017 \pm 0,003$	$0,014 \pm 0,003$	$16,33\% \pm 3,41\%$	$0,734 \pm 0,133$
Hold-Out Verfahren mit Originaldatensatz	$0,015 \pm 0,000$	$0,012 \pm 0,010$	$17,11\% \pm 25,63\%$	$0,784 \pm 0,000$
Hold-Out Verfahren mit neuen Daten	$0,012 \pm 0,000$	$0,01 \pm 0,006$	$11,96\% \pm 6,75\%$	$0,9 \pm 0,00$

tersuchung der Vorhersagegüten verschiedener Modelle würde hier Klarheit schaffen, wird jedoch nicht durchgeführt.

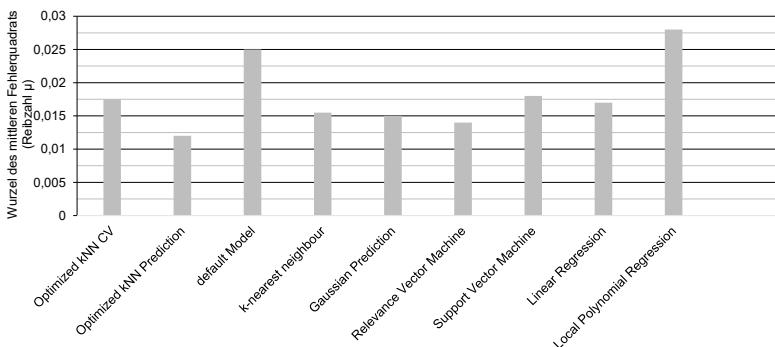


Bild 29: Per Kreuzvalidierung ermittelte Fehlerkennwerte für verschiedene Regressionsmethoden in [Sch13]

2.5.3 Stochastische Analysen in der Feinschneid-Methodenplanung

HITZ identifiziert in seiner Arbeit Herausforderungen, die sich insbesondere durch die Prozessintegration von Feinschneid- und Umformoperationen auf der einen und der mangelnden Rechnerunterstützung beim Prozessbeginn der Methodenplanung auf der anderen Seite ergeben [HMH10]. Der Methodenplaner (Prozessingenieur) muss sich zum großen Teil auf heuristisches Erfahrungswissen, ergänzt um formales Wissen aus Fachbüchern oder Überschlagsrechnungen, verlassen. Das Ziel, eine fehlerfreie und iterationsarme Auslegung von Werkzeugen zur Realisierung metallischer Werkstücke zu erreichen, wird somit aufgrund der komplexer werdenden Teilegeometrie zunehmend schwieriger.

Das von HITZ entwickelte Planungssystem unterstützt den Methodenplaner bei der Auslegung seiner Fertigungsprozesse. Dazu nutzt es Kriging Modelle, die in der Lage sind virtuelles Fertigungswissen zu konservieren [Hit12] und sich in einem Vergleich mit anderen Regressionsmethoden (Lineare Regression, Response Surface Methode, künstliche neuronale

Netzwerke) als vorteilhafter erwiesen haben. Kriging stammt ursprünglich aus der Geostatistik und ermöglicht für Bereiche ohne Stichprobe eine Vorhersage der Zielgröße durch Interpolation umliegender Messwerte [OW15]. HITZ wählt den Ansatz eines Expertensystems und erweitert das enthaltene, formalisierte Expertenwissen durch virtuelles Wissen, wodurch ein *hybrides* Planungssystem zur Auslegung von Feinschneid-Umformprozessen entsteht (s. Bild 30).

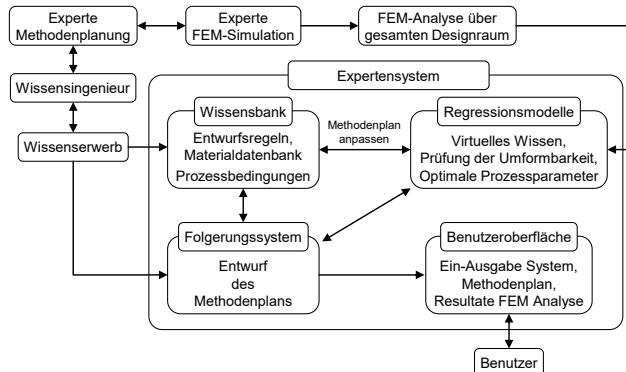


Bild 30: Konzept des hybriden Planungssystems METAPlaner nach [Hit12].

Der Aufbau des Planungssystem METAPlaner entspricht einem dreistufigen Prozess aus den Teilschritten wissensbasierte Auslegung, virtueller Tryout und Regressionsmodellierung. Während der wissensbasierten Auslegung erfasst und strukturiert ein Wissensingenieur das Expertenwissen des Unternehmens über Formfeatures, die den verschiedenen Feinschneid-Umformprozessen zugeordnet sind. Ziel ist die Konservierung von empirischem und analytischem Wissen, um nachfolgende Methodenplanungen zu unterstützen. Die Methodenpläne können als Zeichnungen oder mittels gekoppelter CAD-Modelle dem Benutzer bereitgestellt werden. Im virtuellen Tryout erfolgt die Untersuchung des Systemverhaltens mittels rechnerbasierter Werkzeuge. Der FEM-Experte erstellt, unter Verwendung der Informationen aus der wissensbasierten Auslegung, ein FEM-Modell des Prozesses und analysiert das System innerhalb des vom Methodenplaner definierten Wertebereichs. In der Regressionsmodellierung werden die Ergebnisse des virtuellen Tryouts in den Krigingmodellen abgelegt. Sie bilden einen Wissensspeicher, der es erlaubt, Voraussagen über neue Designs zu treffen, die nicht im virtuellen Tryout oder der wissensbasierten Auslegung untersucht wurden. [Hit12]

3 Die Blechmassivumformung als Anwendungskontext für das selbstlernende Assistenzsystem

Das folgende Kapitel hat das Ziel, dem Leser die Blechmassivumformung als Anwendungskontext des selbstlernenden Assistenzsystems näher zu bringen. Dazu erfolgt eine Betrachtung sowohl aus Sicht der Fertigungstechnik als auch der Konstruktion. Die Synthese aus beiden Blickwinkeln bildet die Katalogisierung von Nebenformelementen für den Prozess der Blechmassivumformung. Dieser Katalog trägt zum einen zur Systematisierung der Blechmassivumformung bei und unterstützt zum anderen bei der Abschätzung ihres Potentials zur Substituierung oder Erweiterung bestehender Verfahren.

3.1 Die Blechmassivumformung aus Sicht der Fertigungstechnologie

Bis vor wenigen Jahren waren Leichtbaubemühungen durch den Einsatz sogenannter Leichtbauwerkstoffe (z. B. Legierungen auf Aluminium- oder Magnesiumbasis, faserverstärkte Kunststoffe) gekennzeichnet [HKS02; Kle99]. Die aus diesen Materialien bestehenden Baugruppen mussten häufig in Verbundbauweise entwickelt, aufwändig gefertigt und montiert werden [Pow00]. Mit Blick auf die Automobilindustrie standen vornehmlich die Karosserie, Komponenten des Antriebstrangs oder der Lenkung im Fokus [Mal10; CL01]. Heutige Leichtbaubemühungen verfolgen nun mitunter das Ziel, durch den Einsatz höherfester und ultrahochfester Stähle die Größe und damit das Gewicht eines einzelnen Werkstückes bzw. einer Komponentengruppe zu reduzieren [Sie14; Fri13]. Neu in den Fokus gerückt sind außerdem Bereiche wie Sitzgestelle, Türmodule oder auch Schließeinheiten sowie darin verbaute Mitnehmer, Rastelemente oder Anschlüsse. Zudem geht der Trend dahin, die Bauteilanzahl bei gleichbleibender Systemfunktion zu verringern, was die Entwicklung und Fertigung funktionsintegrierter Teile erfordert [DL09]. Es kann daher vorkommen, dass Einzelteile im Zuge der Funktionsintegration eine Gewichtszunahme erfahren, während das Gesamtprodukt bei diesem Systemleichtbau an Gewicht verliert.

Die hohen Investitionskosten für Umformmaschinen und -werkzeuge rentieren sich im Allgemeinen nur für sehr große Stückzahlen [Kug09]. Wenn diese Forderung erfüllt ist, bietet die Umformtechnik diverse Vorteile [Lan84; Kug09]:

- effiziente Werkstoffausnutzung im Vergleich zu spanenden Verfahren
- hohe Mengenleistungen, hoher Automatisierungsgrad bei kurzen Stückzeiten
- hohe Maß- und Formgenauigkeit innerhalb bestimmter Toleranzen
- beanspruchungsgerechte Einstellung der Bauteileigenschaften

Allerdings stoßen sowohl die Massiv- als auch die Blechumformung an ihre jeweiligen Verfahrensgrenzen, wenn Präzisionsteile wie in Bild 31 gefertigt werden sollen. Massivumformverfahren zeichnen sich bei präzisen und nacharbeitsfreien Fließpressteilen durch geschlossene Gesenke (Gratfreiheit) aus. Dadurch wird jedoch der Werkstofffluss aus den Blechbereichen, die die Formelemente umgeben, drastisch reduziert und eine ausreichende Formfüllung ist nicht mehr gewährleistet. Zudem wirken in der Umformzone, die relativ zum Werkzeug einen kleinen Bereich einnimmt, für die Blechumformung untypisch hohe Beanspruchungen auf Werkstück und Werkzeug. In der Werkzeugauslegung eingeplante Beanspruchungsreserven werden durch Schwankungen des Halbzeugs (z. B. Anisotropie, Chargenschwankungen der Blechdicke) sehr schnell erschöpft und das Werkzeug versagt bzw. das Werkstück erreicht nicht die geforderte Qualität.

Die Erforschung der Blechmassivumformung, die durch den Sonderforschungsbereich/Transregio 73 (SFB/TR 73) vorangetrieben wird, ist somit die Konsequenz aus modernen Leichtbaubestrebungen und den Grenzen der etablierten Umformverfahren.

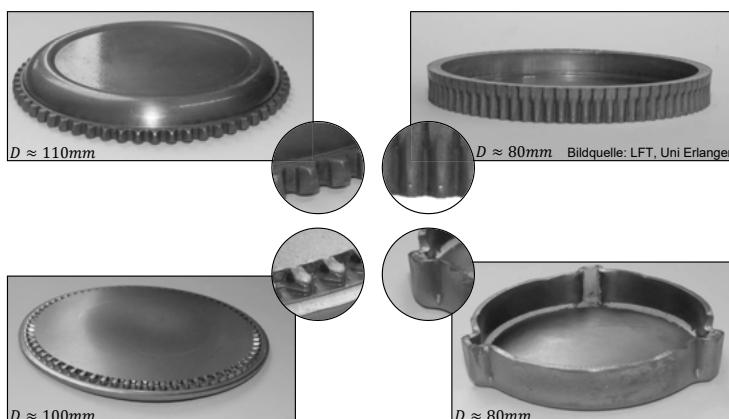


Bild 31: Blechmassivumgeformte Bauteile mit Nebenformelementen.

Unter Blechmassivumformung wird die Anwendung von Verfahren der Massivumformung an Blechhalbzeugen mit einer Dicke zwischen 1 bis 5 mm verstanden [GSH⁺16]. Ziel ist die Ausformung geometrisch komplexer Nebenformelemente (siehe Abschnitt 3.3). Während der Umformung treten charakteristische, komplexe Wechselwirkungen zwischen den lokal eng begrenzten Umformzonen mit hohen und niedrigen 2- und 3-achsigen Spannungs- und Formänderungszuständen auf [MAB⁺12]. Zur Erhöhung der Prozessqualität werden häufig prozessangepasste Halbzeuge verwendet, deren Herstellung ebenfalls durch Prozesse der

Blechmassivumformung erfolgen kann [PO13]. Bild 32 fasst die Prozessgruppen innerhalb der Blechmassivumformung zusammen und stellt wichtige Prozesseigenschaften dar.

Lineare Werkzeugbewegung				Rotierendes Werkzeug		
Stauchen	Abstreck- gleiten	Schmieden	Prägen	Fließdrücken	Taumeln	Nabendrücken
						
Mögliche Dickenänderung						
Known Combinations with Sheet Metal Forming	ja	ja	ja	ja	ja	nein
Forming Forces	hoch	mittel	hoch	hoch	niedrig	mittel

Bild 32: Prozessgruppen innerhalb der Blechmassivumformung nach [MAB⁺12]

Der Einsatz der Blechmassivumformung geht mit der Erreichung unterschiedlicher Ziele einher. Aus Sicht Fertigungstechnologen sind hier zunächst *verkürzte*, aber gleichzeitig *robuste* und zuverlässige Prozessketten zu sehen. Mit Fertigungsprozessen des Stands der Technik wären typische blechmassivumgeformte Bauteile (s. Bild 31) nur in einer aufwändigen Reihe von Fertigungs- und Fügeoperationen (z. B. Laserstralschweißen oder -löten) herstellbar. Diese bringen jedoch Nachteile wie thermischen Verzug und nicht-deterministische Störgrößen (z. B. Temperaturschwankungen, Anfälligkeit für Bodenerschütterungen) mit sich [Nee15]. Zudem werden Werkstoffe verarbeitet, die die Anforderungen des Systemleichtbaus nur bedingt erfüllen, beispielsweise Bronze- oder Messingsonderlegierungen [Sch02; Kir07]. Blechmassivumgeformte Bauteile weisen hingegen keine Schweißnähte und Wärmeeinflusszonen auf, was deren Zuverlässigkeit im Betrieb deutlich erhöht. Da sich die Blechmassivumformung gegen etablierte Verfahren behaupten muss, ist die Verbesserung der Wirtschaftlichkeit ein weiteres Ziel. Dieses wird zum einen dadurch erreicht, dass Bauteile einbaufertig hergestellt werden. *Einbaufertig* bedeutet, dass lediglich Maßnahmen zur Entfernung von Fertigungshilfsmitteln (z. B. Umformöl) dem Umformprozess folgen dürfen [Raa06]. Zum anderen müssen die Auslegung und Fertigung der Blechmassivumformwerkzeuge selbst erforscht werden. Hierzu zählen auch Oberflächenbehandlungen wie das Aufbringen von bionischen Mikrostrukturen [Her13] oder tribologischer Dünnschichten [Het14] sowie das Behandeln durch Schleifen [BHV⁺15] oder Plasmanitrieren [LBH⁺12].

Diese und weitere Ziele werden durch das Lösen von Problemstellungen hinsichtlich unterschiedlicher, teils in Wechselwirkung stehender Aspekte erreicht. In den letzten Jahren gehörten hierzu der Materialfluss und seine Steuerung während der Umformung, die Einstellung der Reibung zwischen Werkzeug und Werkstück, die Minimierung von Werkzeugbelastungen und die numerische Modellierung des gesamten Prozesses auf unterschiedlichen Skalen. [GSH⁺16]

Jede gelöste Problemstellung trägt zur Qualifizierung der Blechmassivumformung bei, erhöht das Spektrum an realisierbaren Nebenformelementen und führt vor allem zu neuen

Erkenntnissen. Dieser Erkenntnis- und Wissensgewinn ist durch eine hohe Dynamik gekennzeichnet, zu der auch ein umfangreicher Rechnereinsatz beiträgt. Dadurch wird neues Wissen innerhalb kurzer Zeit – in Relation zur Entwicklungshistorie der Umformtechnik – generiert. Im Folgenden beschreibt ein Beispiel aus dem SFB/TR 73, wie unterschiedliche Ansätze zur Optimierung des Materialflusses und der Formfüllung beitragen.

In [MKS⁺10] wird festgestellt, dass die Reibverhältnisse bei einem Blechmassivumformprozess einen hohen Einfluss auf die Formfüllung haben. Durch FEM-basierte Umformsimulationen können MERKLEIN ET AL. zeigen, dass die Formfüllung, beim Quer-Fließpressen der Verzahnung aus Bild 31 (links unten), verbessert werden kann, wenn der Reibfaktor von $m = 0,12$ auf $m = 0,3$ angehoben wird. Die mögliche experimentelle Umsetzung dieses Ansatzes zeigen wenig später VIERZIGMANN ET AL. [VKME12], indem sie die Oberflächenrauheit und damit den Reibfaktor von Blechhalbzeugen lokal durch Sandstrahlen erhöhen. Anhand eines Zapfenpressversuchs können sie die prinzipielle Wirksamkeit dieser Maßnahme zur Steuerung des Materialflusses nachweisen. Dem steht allerdings die notwendige Dauer des Sandstrahlens gegenüber, die mit 25 Sekunden pro Halbzeug sehr lang gewählt werden muss und dem Ziel der Wirtschaftlichkeit nicht dienlich ist. Einen anderen Ansatz verfolgt HENSE indem er die tribologischen Eigenschaften der Werkzeugoberfläche durch Fräsen mit induzierten Werkzeugschwingungen lokal einstellt [Hen13]. Die Schwingungen des Fräswerkzeugs erzeugen Strukturen auf der Werkzeugoberfläche, die während der Umformung aufgrund hoher KontaktNormalspannungen¹⁰ teilweise plastisch geglättet werden. Durch die Glättung können sich Vertiefungen bilden, in denen sich Schmierstoff zur Reibungsreduzierung ansammelt. Über den Anteil dieser geschlossenen Schmiertaschen lassen sich die Reibverhältnisse in der Wirkfuge *Werkzeug-Werkstück* beeinflussen [BHH⁺11]. Neben der Erhöhung des Reibfaktors ist auch die Verwendung prozessangepaster Halbzeuge zielführend wie GRÖBEL ET AL. zeigen. Durch solche *Tailored Blanks*, die im Rondennärbereich bspw. durch Stauchen, Taumeln oder flexibles Walzen [Ope13] aufgedickt werden können, lässt sich die Formfüllung und damit die Produktqualität ebenfalls steigern [GHEM15].

3.2 Bedeutung für die Produktentwicklung

Die Produktentwicklung kann von der Flexibilität und Wirtschaftlichkeit der Blechmassivumformung profitieren, um damit den Forderungen nach kürzeren Produkteinführungszeiten und hoher Variantenvielfalt bei geringer interner Varianz entgegenzutreten. Die neuartigen Gestaltungsmöglichkeiten der Blechmassivumformung können zudem die Umsetzung der aktuellen Leichtbaubestrebungen (s. Abschnitt 3.1) in der Produktentwicklung unterstützen. Bild 33 zeigt den Demonstrator des SFB/TR 73 mit Detailansichten typischer Nebenformelemente.

Damit die *time-to-market* verkürzt werden kann, muss u.a. auch die Zahl der Iterationen im gesamten Produktentwicklungsprozess auf ein Minimum reduziert werden

¹⁰ Nach [LGE⁺15] werden bis zu 3.000 MPa erreicht.

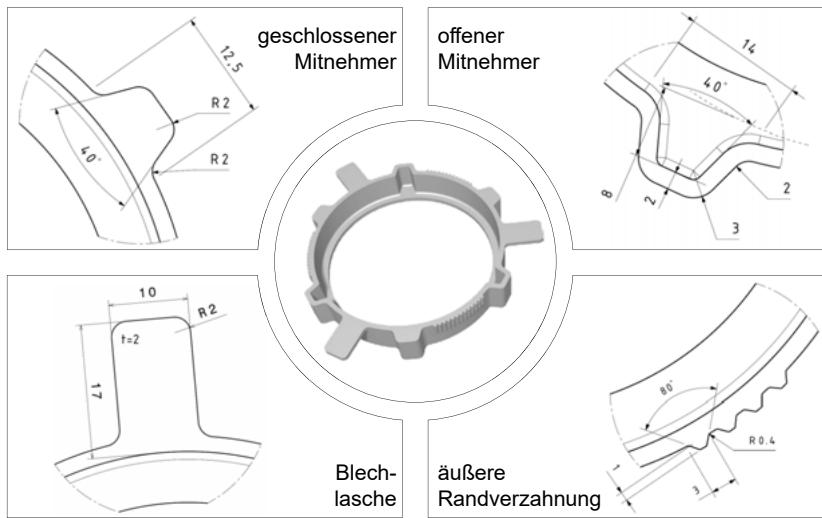


Bild 33: Demonstrator des SFB/TR 73 mit vergrößert dargestellten Nebenformelementen.

[Her10; JEWC03]. Mit Blick auf den Aspekt des Design for Manufacture werden viele Iterationen durch nicht-fertigungsgerechte Bauteilentwürfe verursacht, die von der Fertigungsprozessentwicklung aufgrund zu teurer oder nicht fertigbarer Formfeatures zurückgewiesen werden [FG13]. Gerade die Prozesse der Blechmassivumformung zeigen jedoch eine hohe Sensitivität gegenüber kleinen Geometrieveränderungen (z. B. Blechdicke, Nebenformelementvolumen, Höhe der Verzahnung oben rechts in Abb. 31) [GSH⁺16; SM11]. Die zeitnahe Ableitung unterschiedlicher Bauteilentwürfe zur Erfüllung der vom Markt geforderten Variantenvielfalt wird somit erschwert. Zwar kann die Prozessauslegung und die damit verbundene Design for Manufacture Analyse durch den Einsatz von Umformsimulationswerkzeugen (z. B. Simufact Forming[®]) unterstützt werden¹¹, doch dieses mehrmalige Durchlaufen des CAE-Prozesses, bestehend aus Preprocessing, Solving und Postprocessing ([MG15]), wirkt dem Ziel der Iterationsminimierung entgegen.

Die Ingenieure der Konstruktionsabteilung sind darauf angewiesen, frühzeitig konstruktionsrelevantes Fertigungswissen zur Verfügung zu haben (s. Bild 2, S. 2), das eine schnelle Design for Manufacture Analyse von Bauteilentwürfen ermöglicht, „up-to-date“ in Bezug auf die bei der Erforschung der Blechmassivumformung gefundenen Erkenntnisse ist und die komplexen Wechselwirkungen (s. Abschnitt 3.1) der Blechmassivumformprozesse berücksichtigt.

Aus Gesprächen mit Fertigungsexperten innerhalb des SFB/TR 73 konnte der Schluss gezogen werden, dass allein auf Basis der Geometrie eines blechmassivumgeformten Bau-

¹¹ Zur FE-Modellierung von Blechmassivumformprozessen siehe u. a. [MKS⁺10; MAB⁺12; SLV⁺11].

teils keine allgemeingültigen Aussagen über dessen Fertigbarkeit getroffen werden können. Hierin liegt ein wesentlicher Unterschied zu geometriebasierten Analysen bei etablierten Fertigungsverfahren (s. Abschnitt 2.3.2). Bei der Design for Manufacture Analyse eines blechmassivumgeformten Bauteils muss die Ausprägung unterschiedlicher, teilweise wechselwirkender Parameter geprüft werden, bevor eine fundierte Aussage zur Fertigungsge rechtigkeit getroffen werden kann. Das Ergebnis eines Blechmassivumformprozesses gilt als akzeptabel, wenn alle für diesen Prozess relevanten Parameter innerhalb zuvor festgelegter Ober- und/oder Untergrenzen liegen. Welche Parameter als relevant eingestuft werden, muss in interdisziplinären Gesprächen geklärt und protokolliert werden. Die folgende Liste beschreibt typische Produkt- und Prozessparameter die in Gesprächen mit Fertigungsex perten erhoben wurden:

- vertikale Stößelkraft F_u [kN]: Eine ausreichende (vertikale) Umformkraft ist notwendig, um im Material die erwünschten Spannungs- und Formänderungszustände zu erzeugen. In Abhängigkeit von der eingesetzten Umformmaschine existiert jedoch eine technologische Obergrenze $F_{u,max}$.
- horizontale Stößelkraft F_h [kN]: Unsymmetrische Bauteile induzieren seitliche Kräfte auf den Stößel. Diese verursachen eine Verlagerung des Werkzeugs und damit unzulässige Form- und Maßabweichungen. Eine maximale horizontale Stößelkraft $F_{h,max}$ darf nicht überschritten werden [SKH¹³].
- Werkzeugbeanspruchung σ_v [MPa]: Die Blechmassivumformung zeichnet sich u. a. durch lokal eng begrenzte aber sehr hohe Werkzeugbeanspruchungen aus. Für manche Blech massivumformprozesse (bspw. inkrementelle Umformung) ist dies der limitierende Pro zessparameter [GSH¹⁶].
- Umformgrad φ [-]: Die dimensionslose, auch als logarithmische Formänderung bezeich nerte, Kenngröße beschreibt die bleibende geometrische Formveränderung eines Werk stücks. Ist der Umformgrad an einer Stelle des Bauteils größer als der Grenzumformgrad φ_{vG} des Werkstoffs, kommt es an dieser Stelle zu Werkstoffabrisse, Auffaltungen oder zum Bruch des Werkstücks.
- Blechausdünnung $t_{\Delta,max}$ [mm, %]: Unter der Annahme der Volumenkonstanz führt das Fließen des Werkstoffs während der Umformung zu lokalen Blechausdünnungen wie in [SM11; SLV¹¹] für das Tiefziehen gezeigt. Eine maximale Blechausdünnung $t_{\Delta,max}$ darf nicht überschritten werden, sonst kann das Bauteil im Betrieb versagen.
- Blechaufdickung $t_{\Delta,min}$ [mm, %]: Im Falle der prozessangepassten Halbzeuge ist wieder um eine Blechaufdickung wünschenswert, um für weitere Umformschritte ausreichend Werkstoff innerhalb der Umformzone bereitstellen zu können. Ein Mindestmaß $t_{\Delta,min}$ an Blechdickenerhöhung muss überschritten werden.
- Kontaktverhältnis c [%]: Das Kontaktverhältnis zwischen Werkzeugkavität und Werk stück nach der Umformung gibt Auskunft über die erreichte Ausformung des Formele ments. Das Nebenformelement wurde vollständig ausgefüllt, wenn das Kontaktverhältnis 100 % beträgt.

Zu beachten ist, dass diese Liste zum einen keinen Anspruch auf Vollständigkeit erhebt, denn neue blechmassivumgeformte Bauteile erfordern ggf. die Berücksichtigung anderer Parameter. Zum anderen können an dieser Stelle keine allgemeingültigen Grenzwerte definiert werden. Manche Parameter sind maschinen- (z. B. maximale Umformkraft $F_{u_{max}}$), andere material- (z. B. maximaler Umformgrad φ_{max}) oder werkzeugabhängig (z. B. maximale Spannung im Werkzeug σ_v) [GSH¹⁶; MTB¹¹]. Die Fertigbarkeit des tiefgezogenen Napfs mit offenen Mitnehmern (Abb. 31 rechts unten) wird beispielsweise durch Betrachtung von $F_{u_{max}}$, φ_{max} und $t_{\Delta_{max}}$ beurteilt. Hingegen werden bei der Ronde mit der fließgepressten Außenverzahnung (Abb. 31 links unten) die Ausprägungen von $F_{u_{max}}$, φ_{max} und c_{max} betrachtet. [BSSW15]

3.3 Klassifikation von Nebenformelementen

Die Blechmassivumformung zielt auf die Herstellung von komplexen Funktionsbauteilen mit Nebenformelementen mittels (kalt-)umformender Verfahren ab [MH16]. Prinzipiell lassen sich derartige Bauteile mit existierenden Fertigungsverfahren herstellen, jedoch nur in aufwendigen, teils unwirtschaftlichen Folgen von umformenden und spanenden Verfahren sowie Fügeoperationen. Damit eine Etablierung der Blechmassivumformung unterstützt werden kann, muss es möglich sein, ihr Potential zur Substituierung oder sinnvollen Ergänzung bestehender Verfahren qualitativ abzuschätzen. Zu diesem Zweck wurde ein Ordnungssystem (Katalog) mit Klassen entwickelt, mit dem Nebenformelemente von Bauteilen hinsichtlich ihrer geometrischen Relation zum Hauptformelement klassifiziert werden können. Von dieser Klassifizierung hängen die einzusetzenden und ggf. zu kombinierenden Blechmassivumformverfahren (s. Bild 32) ab.

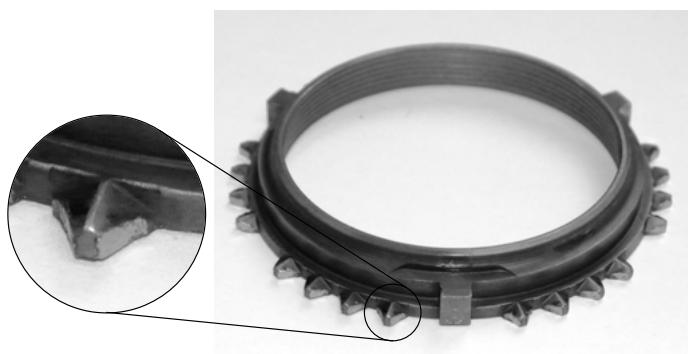


Bild 34: Ein Synchronring eines Fahrzeuggetriebes mit der Sperrverzahnung als exemplarisches Nebenformelement.

Die Unterscheidung von Hauptformelement und Nebenformelement hat v. a. in der Fertigungstechnik praktische Bedeutung, da die Fertigung von Nebenformelementen oft zu-

sätzliche Fertigungsschritte, teils auf weiteren Maschinen mit entsprechenden Umspannmaßnahmen erfordert. In der Literatur sind hingegen wenig eindeutigen Unterscheidungen von Haupt- und Nebenformelementen verfügbar. Eine Möglichkeit ist das Maß, in welchem ein Geometrieelement das Aussehen des Werkstücks beeinflusst. Ist zu erkennen, dass sich das Element mit seinen Abmaßen stark von allen anderen abhebt und charakterisiert es die Gestalt des Bauteils, ist es als Hauptformelement einzufordern. Nebenformelemente hingegen sind von geringer Bedeutung für das Aussehen und bewirken lediglich kleinere Veränderungen an der Grobgestalt [HKR70; Gre95]. Das wichtigste Merkmal für ein Nebenformelement ist, dass es als Hauptfunktionsträger angesehen werden kann. Das bedeutet, dass mindestens eine der Berandungsflächen eines Nebenformelements als die Wirkfläche bezeichnet werden kann, an der die Erfüllung einer Hauptfunktion eines Bauteils unter Anwendung eines physikalischen Effekts erzwungen wird [FG13]. Insofern können die Bezeichnungen komplexes Nebenformelement und komplexes Funktionsbauteil im Rahmen dieser Arbeit synonym verwendet werden. Im Falle des in Bild 34 gezeigten Synchronrings kann als Beispiel die Funktion „Verhindern axialen Durchschaltens“ genannt werden. Bei der Gangsynchronisation darf die Schiebemuffe erst in den Kupplungskörper des Gangrads durchschalten, wenn die Drehzahlen beider Körper synchronisiert sind [Kir07]. Vorher verhindern die in axiale Richtung orientierten Dachschrägen der Sperrverzahnung durch einen Reib- und Formschluss vorzeitiges, axiales Durchschalten der Schiebemuffe.

Bei der Entwicklung des Nebenformelementekatalogs wurden unterschiedliche Anforderungen berücksichtigt. Hinsichtlich der Repräsentation ist die Genauigkeit und der Umfang des Ordnungssystems am Einsatzzweck zu orientieren. Die Klasseneinteilungen verfügbarer Ordnungssysteme, wie bspw. von OPITZ [Opi66] oder ZIMMERMANN [Zim67], sind teilweise über mehrere Tabellen verteilt, was den Zugriff auf die jeweiligen Systeme erschwert. Das Ordnungssystem ist derart zu gestalten, dass Bauteile unabhängig vom aktuellen Herstellungsprozess und ihrer Funktion klassiert werden können. Zur Beschreibung der Formelementegeometrien müssen Begriffe verwendet werden, die von Fertigungingenieuren und Produktentwicklern gleich interpretiert werden. Dabei und bei der Umsetzung des Katalogs ist auf eine ausreichende Praxistauglichkeit zu achten.

Die Klassifikation von Nebenformelementen der Blechmassivumformung besteht aus den Schritten (I) Entwicklung der Klassifizierung und (II) Klassierung der Formelemente. Zu Beginn werden zur Unterscheidung der Nebenformelemente passende Klassen mit den beschreibenden sog. *Klassemen*¹² entworfen. Die Einteilung nach [Zim67] in Positiv- und Negativformen sowie Öffnungen wird hierfür angepasst und erweitert. Tabelle 6 fasst die definierten Formelemente mit Beispielen zusammen.

Jede Klasse wird durch weitere Merkmale in Untergruppen unterteilt, mit denen festgelegt wird, auf welcher Seite des Hauptformelements das Nebenformelement positioniert wird, zu welcher Grundfläche es referenziert wird und in welcher Größenordnung der Querschnitt des Nebenformelements im Vergleich zum Blechhalbzeug liegt. Innerhalb der oben genannten Klassen wird bei der Positionierung entsprechend zwischen Nebenformelementen an der

¹² Ein Klassem ist ein semantisches Merkmal, mit dem eine Gruppe von Objekten erfasst werden kann.

Tabelle 6: Übersicht der Klassen für Nebenformelemente der Blechmassivumformung.

Formenklasse	Formelement	Beispiel
Formenklasse 1	Erhebung	 Anschlag am Synchronring
Formenklasse 2	Vertiefung	 Offener Mitnehmer an SFB/TR73 Bauteil
Formenklasse 3	Verzahnung	 Sperrverzahnung am Synchronring
Formenklasse 4	Öffnung	 Öffnung/ Langlöcher an Lamellenträger
Formenklasse 5	Rillierung/Rändel	 Umfangsriillierung an SFB/TR 73 Bauteil

Innen- und an der Außenseite des Hauptformelements unterschieden. Als Positionierungsreferenz kommt die Stirnfläche (axiale Ausrichtung) oder die Ringfläche (radiale Ausrichtung) des Hauptformelements in Frage. Dieses Merkmal ist von hoher Relevanz, da es die Auswahl einsetzbarer Blechmassivumformprozesse wesentlich beeinflusst. Die Festlegung des Merkmals „Querschnitt“ ist v. a. für die Formelemente der Klasse *Erhebung* relevant. Dadurch kann definiert werden, ob ein Nebenformelement in seiner Größe der Dicke des Blechhalzeugs entspricht oder nicht. Für Funktionselemente, deren Querschnitt von der ursprünglichen Dicke abweicht, ist ein vollständiges Fließen des Materials unter Einbeziehung des umgebenden Bereichs erforderlich. Ist das Nebenformelement bspw. größer, kann der Fertigungsexperte daraus ableiten, dass der Einsatz von prozessangepassten Blechhalzeugen mit lokal erhöhter Blechdicke notwendig sein wird. Die Vergabe der Merkmale ist im Einzelfall zu prüfen. Für das Formelement *Öffnung* ist die Unterscheidung zwischen Innen- oder Außenseite aus konstruktionstechnischer Sicht zum Beispiel nicht zwingend notwendig. Dennoch übt dieses Formelement einen Einfluss auf den Fertigungsprozess aus. Denn je nach zu integrierendem Verfahren (z. B. Scherschneiden, Laserstrahlschneiden), sind unterschiedliche Anforderungen zu berücksichtigen.

Hinsichtlich der Hauptformelemente werden die Geometrien Scheibe, Ring, Hülse und Napf festgelegt. Diese Benennung wird aus Gründen der Praxistauglichkeit gewählt, auch wenn rein geometrisch betrachtet ein Ring als Napf mit sehr niedriger Wandhöhe ausgedrückt werden kann. Bereits im Abschnitt 3.2 wurde hervorgehoben, dass zur Erfüllung der Hauptfunktion eines Bauteils die Nebenformelemente ausschlaggebend sind. Das Hauptforme-

ment übernimmt im Wesentlichen die Aufgabe, die Nebenformelemente zu tragen, bzw. von ihnen aufgenommene Kräfte und Momente weiterzuleiten. Hauptformelemente werden daher nicht detaillierter beschrieben.

Als Repräsentationsform des Ordnungsschemas wird eine Matrixstruktur vorgeschlagen. Die Eingruppierung in eine hierarchische Baumstruktur ist prinzipiell möglich, diese kann jedoch weniger kompakt und übersichtlich ausgeführt werden. Der höhere Aufwand bei der Erstellung einer matrixbasierten Repräsentation wird nach [Lin09] von der Möglichkeit aufgewogen, Lücken („weiße Felder“) für neue Entwicklungen zu identifizieren. Mit der verfügbaren Klassifizierung wurden aus unterschiedlichen Komponenten mehrere Bauteile ausgewählt, analysiert und gefundene Formelemente klassiert (Schritt 2). Die Komponenten stammen aus Produkten des Automobilbereichs wie Doppelkupplungsautomatikgetriebe und Sitzsysteme. Zu Anschauungszwecken wurde auch der Bauteildemonstrator des SFB/TR 73 in den Katalog aufgenommen. Im Anhang A1 (Seite 157) ist das gesamte Ordnungsschema mit den definierten Klassemen und Merkmalen sowie die klassifizierten Formelemente abgebildet.

Es ist zu erkennen, dass eine homogene Befüllung der Klassifizierungsmatrix nicht gelingt. Der Grund hierfür liegt jedoch weniger an den gewählten Gruppen und weiteren Unterscheidungsmerkmalen selbst, sondern vielmehr an der Häufigkeit, mit welcher die verschiedenen Funktionselemente auftreten. Besonders oft sind Funktionselemente der Formenklasse 1 (Erhebung) an der Außenseite einer Ringfläche klassifiziert worden. Diese Klasse beinhaltet u. a. Mitnehmer und verzahnte Elemente, die zur Kraft- oder Momentübertragung dienen. Sie sind somit in der Bauteilauswahl besonders häufig vertreten. Nebenformelemente der Formenklasse 1 wurden im Vergleich dazu weitaus weniger identifiziert. Erhebungen mit Orientierung nach innen stellen die existierenden Umformverfahren vor Probleme, da mit solchen Geometrien oft Hinterschneidung verbunden sind. Ebenso häufig wie Erhebungen sind auch Verzahnungen an einer Umfangsfläche klassifiziert worden. Der Unterschied zwischen Innen- und Außenseite fällt dabei nur marginal aus. Verzahnungen an der Stirnseite eines Bauteils wurden im Rahmen dieser Klassifikation nicht identifiziert, da keine entsprechenden Bauteile in den analysierten Produkten vorkommen. Als Beispiel wäre an dieser Stelle eine Hirth-Verzahnung zu nennen, die mit konventionellen, spanenden Verfahren einen hohen Fertigungs- und Kostenaufwand verursacht. Die Benutzung von Begriffen, die häufig sprachlich verwendet werden, deren Abgrenzung jedoch mitunter unscharf ausfällt, kann sich negativ auf die Objektivität auswirken. So lässt sich ein Bereich, der durch zwei umlaufende Erhebungen begrenzt wird, auch als Vertiefung interpretieren. Der Benutzer kann sich zur Lösung dieser Problematik an bereits eingeordneten Elementen orientieren oder Rücksprache mit Fertigungsexperten halten, um festzustellen ob bei bestimmten Formelementen eine Materialaufdickung oder -ausdünnung im Fertigungsprozess überwiegt. Auffällig selten treten Funktionselemente an Bauteilstirnflächen auf. Eine Begründung hierfür ist die getroffene Auswahl an Bauteilen, bei denen eine radiale Krafteinleitung (vorrangig Drehbewegungen mit Momentübertragung) überwiegt.

Eine ausführliche Analyse des Potentials der Blechmassivumformung mithilfe des Formelementekatalogs findet sich in [BW13]. Im Folgenden wird anhand des Stahlblech-

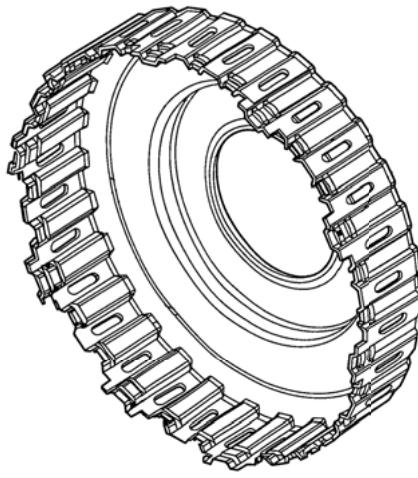


Bild 35: Exemplarischer Stahlblech-Lamellenträger aus [BGSS03].

Lamellenträgers ein Beispiel zur Klassifikation und Potentialabschätzung gezeigt. Wie in Bild 35 zu erkennen, verfügt das Bauteil über axiale, außenliegende Stege. Diese werden in einem mehrstufigen Kaltwalzprozess aus einer Stahlronde gefertigt. Weitere Prozessschritte sind das Prägen der Lamellenanschläge auf den umlaufenden Stegen sowie das Stanzen der Langlöcher auf den Außenseiten der Stege [Kög10; Klo07]. Alle Formelemente des Lamellenträgers lassen sich in der Klassifikationsmatrix für Formelemente der Blechmassivumformung wiederfinden. Die axialen Stege ähneln den offenen Mitnehmern, die im Rahmen eines experimentellen Blechmassivumformprozesses durch einen einstufigen Tiefziehprozess gefertigt werden konnten [MAB⁺12]. Die Lamellenanschläge erfordern einen lokal begrenzten Werkstofffluss aus der Blechebene heraus, was durch den Blechmassivumformprozess der inkrementellen Umformung bei ähnlichen Formelementen bereits erreicht wurde [SWG⁺16]. Das Stanzen von Öffnungen ist zwar kein spezifischer Blechmassivumformprozess, wurde jedoch bereits erfolgreich in eine Kette von Blechmassivumformprozessen integriert [BBVe15], so dass auch bei diesem Formelement Rationalisierungspotential besteht. Die genaue Auslegung eines Blechmassivumformprozesses zur Fertigung eines derartigen Bauteils mit den gezeigten Formelementen hat im Anschluss an die Klassifikation durch den Fertigungsexperten zu erfolgen.

4 Ableiten des Handlungsbedarfs

4.1 Fazit zum Stand der Forschung

Bezüglich der zentralen Thematiken dieser Arbeit wurden in den vorangegangenen Abschnitten die relevanten Grundlagen dargestellt und die bestehenden Forschungsarbeiten analysiert. Die Bedeutung des Faktors Wissen für die Produktentwicklung konnte gezeigt werden (Abschnitt 2.1). Die Erhebung dieses Wissens erfolgt entweder durch direkte, indirekte oder automatische Methoden, die in Abschnitt 2.1.4 bzw. Tabelle 3 (S. 17) gegenübergestellt wurden. Die automatischen Methoden zeigen dabei hinsichtlich unterschiedlicher Aspekte ein höheres Potential um effiziente und effektive Wissensakquise zu gewährleisten. Dennoch wurde in Abschnitt 2.2.3 festgestellt, dass im Bereich der Produktentwicklung wissensbasierte Systeme (Abschnitt 2.2) mit Wissensakquisitionskomponenten vorherrschen, die direkte oder indirekte Methoden nutzen (Abschnitt 2.2.3, S. 27, S. 30 und S. 32).

Im vorangegangenen Kapitel 3 erfolgte eine Untersuchung der Blechmassivumformung hinsichtlich der Thematik des Design for Manufacture. Es wurde festgestellt, dass eine rein geometrische Betrachtung der Fertigbarkeit, bspw. durch die Identifikation zu starker Wanddicken sprünge oder Hinterschneidungen, den komplexen Prozesszusammenhängen bei der Blechmassivumformung nicht gerecht werden kann. Auch allgemeingültige Konstruktionsregeln für Umformprozesse, wie sie z. B. bei ROTH zu finden sind (s. Bild 20, S. 37), können keine ausreichende Unterstützung bei Design for Manufacture Analysen von blechmassivumgeformten Bauteilen bieten. Frühzeitige Prozessauslegungen und Aussagen über die Fertigungsgerechtigkeit sind aktuell nur durch zeit- und kostenintensiven Rechner-einsatz im Rahmen FEM-basierter Prozesssimulationen möglich [SLV⁺11; MKS⁺10]. Die dafür eingesetzten Werkzeuge (z. B. Simufact Forming[®]) können jedoch nur von speziell geschultem Personal bedient werden, binden Rechenkapazitäten und erfordern ein zeitaufwendiges Durchlaufen der Simulationsschritte Preprocessing, Solving und Postprocessing. In der Produktentwicklung besteht demnach der Bedarf nach rechnerunterstützten Methoden und -werkzeugen, die die frühzeitige Nutzung des Potentials von Verfahren der Blechmassivumformung im Produktentwicklungsprozess erlauben. Dass ein solches Potential existiert, hat die Klassifikation von Nebenformelementen an bestehenden, konventionell gefertigten Funktionsbauteilen (s. Abschnitt 3.3) gezeigt.

Der Gedanke, dass die Technologie der Wissensentdeckung in Datenbanken einen wichtigen Beitrag zum Lösungskonzept der vorliegenden Arbeit leisten kann, ist bereits in der Einleitung der Arbeit skizziert. In den Abschnitten 2.4 sowie 2.5 erfolgt daher die Darstellung der relevanten KDD-Grundlagen sowie bisheriger KDD-basierter Ansätze entlang

des Produktentstehungsprozesses. Dabei fällt auf, dass jeder Ansatz beim KDD-Schritt „Data-Mining“ (s. Abschnitt 2.4.2, S. 43) einen bestimmten Algorithmus favorisiert und den KDD-Prozess hinsichtlich dieses gewählten Algorithmus optimiert.

SCHULZ zum Beispiel entscheidet sich für den Einsatz künstlicher neuronaler Netze (KNN) und ermittelt für die beim überwachten Training von KNN relevanten Parameter (z. B. Trainingszyklen, Momentum, Decay, Lernrate) die optimalen Einstellungen [Sch13]. Diese halb-automatische Vorgehensweise muss für jeden neuen Datensatz wiederholt werden und erfordert für den dauerhaften Betrieb eines derartigen wissensbasierten Systems einen Experten, der sowohl das Data-Mining Tool beherrscht als auch mit der Theorie der künstlichen neuronalen Netze vertraut ist. Zudem repräsentieren KNN das erhobene Wissen auf subsymbolischer Ebene (s. Bild 28), was die Transparenz reduziert und die Interpretation erheblich erschwert [Läm03; KBK⁺13].

Einen ähnlichen Weg verfolgt auch [Hit12] bei der metamodellbasierten Methodenplanung für Feinschneidprozesse. Seine Ziele sind die virtuelle Generierung von Prozesswissen zur Auslegung von Feinschneidprozessen und die Speicherung dieses Wissens in Form von Metamodellen. Dazu vergleicht er unterschiedliche Metamodellierungstechniken (lineare Regression, polynomiale Regression, Kriging und KNN) sowohl quantitativ als auch qualitativ miteinander. Der quantitative Vergleich basiert auf Trainingsdaten mit einer steigenden Anzahl N an Datentupeln ($N_1 = 9$, $N_2 = 25$, $N_3 = 64$), die mittels einer Testfunktion (Branin-Funktion, siehe [Bra72]) generiert wurden. Lineare und polynomiale Regressionsfunktionen (quadratischer Polynomansatz) schneiden im anschließenden Vergleich der Wurzel des mittleren Fehlerquadrats (RMSE, siehe Tabelle 4) sehr schlecht ab, wobei hierin eine methodische Schwachstelle liegt. Die Branin-Funktion beinhaltet immer einen Polynomterm vierten Grades, weshalb die Approximation durch Polynome niedrigeren Grades zwangsläufig zu hohen Vorhersagefehlern führt. Aus den verbleibenden Modellierungstechniken wird schließlich das Kriging auf Basis eines Vergleichs der RMSE-Varianzen ausgewählt. Die RMSE-Varianz des KNN fällt bei HITZ sehr groß aus, was auf die nur einmal durchgeführte k -fache Kreuzvalidierung zurückzuführen ist. Neue Datensätze können über ein selbstentwickeltes GUI mit der Bezeichnung „METAGenerator“ verarbeitet werden. Auch hier erfordert die Pflege und Erweiterung der Wissensbasis einen Kriging und Data-Mining Experten. Zudem kann das Wissen, repräsentiert durch die Kriging-Modelle, nicht ohne weitere Visualisierungshilfen anschaulich dargestellt werden.

Der Text-Mining Ansatz in [ESM⁺11] und [Sha13] eignet sich durchaus für die Verarbeitung der im Rahmen des ECO-Prozesse genutzten Dokumente. Allein die Vielzahl (53.000 Unterlagen) und die Art (strukturierte Formblätter) unterscheiden sich jedoch zu den Dokumenten die in den frühen Phasen der Entwicklung von Blechmassivumformprozessen üblicherweise anfallen. Für FEM-Simulationen werden nur teilweise Berechnungsberichte angefertigt und Protokolle zu Umformexperimenten folgen keiner einheitlichen Struktur. Eine Verarbeitung durch Text-Mining zur Erhebung von fertigungsrelevantem Konstruktionswissen ist somit nicht zielführend.

4.2 Anforderungen an ein verbessertes Konzept

In der Entwurfsphase blechmassivumgeformter Bauteile müssen Produktentwickler fortwährend Design for Manufacture Analysen an Gestaltentwürfen durchführen, bspw. um die Auswirkungen kleiner geometrischer Änderungen an komplexen Nebenformelementen abschätzen zu können. Das hierfür notwendige konstruktionsrelevante Fertigungswissen muss aus den Daten der Fertigungsprozessentwicklung akquiriert und in den Produktentwicklungsprozess transferiert werden. Eine geeignete Bereitstellung kann über ein wissensbasiertes System erfolgen. Daraus und aus der vorangegangenen Bilanz werden im Folgenden Anforderungen an ein solches System formuliert:

- Das wissensbasierte System muss den Benutzer bei der Design for Manufacture Analyse blechmassivumgeformter Bauteile unterstützen (assistieren). Das hierfür notwendige konstruktionsrelevante Fertigungswissen soll das System selbstständig aus Fallbeispielen der Prozessentwicklung lernen. Insofern wird im weiteren Verlauf der Arbeit die Bezeichnung *selbstlernendes Assistenzsystem* (Slassy) genutzt. Im Abschnitt 5.1 (S. 69) wird der Begriff „Selbstlernen“ eingehend diskutiert.
- Der Benutzer muss durch Slassy in allen Phasen des Wissenserwerbs unterstützt werden. Dies beinhaltet die Lokalisierung der Wissensquellen, die Erhebung, Strukturierung und Formalisierung des Wissens und die Implementierung in der Wissensbasis. Eine Unterstützung der Wissensbasispflege soll ebenso durch Slassy möglich sein. Dies betrifft die Erweiterung, Modifikation sowie Aktualisierung der Wissensbasis. So kann einer Wissensalterung effektiv entgegengewirkt werden.
- Damit der Benutzer das Ergebnis des Selbstlernprozesses beurteilen kann, müssen ihm quantitative Informationen zur Güte bzw. Verlässlichkeit des akquirierten Wissens bereitgestellt werden. Da das Wissen durch sogenannte Vorhersage- oder Metamodelle repräsentiert werden soll, sind Methoden zur Berechnung geeigneter Kennwerte zu entwickeln.
- Die Wissensakquisitionskomponente muss Robustheit gegenüber Ausreißern und Fehlern in den Daten aufweisen. Die Selbstlernkomponente muss verlässlich und wiederholbar das beste Modell zur Repräsentation des konstruktionsrelevanten Fertigungswissens identifizieren können.
- Slassy muss sich in die gewohnte Arbeitsumgebung des Benutzers (Konstrukteur) integrieren lassen, um eine möglichst hohe Akzeptanz zu gewährleisten. Ein in sich abgeschlossenes Werkzeug („Insellösung“), das der Benutzer zusätzlich beherrschen muss, ist nicht zielführend.
- Es muss gewährleistet werden, dass der analysierte Bauteilentwurf als Ergebnis des Assistenzsystems im weiteren Verlauf des Produktentstehungsprozesses genutzt werden kann. Die digitale Prozesskette zwischen Produktion und Fertigung darf nicht unterbrochen werden.

5 Der Selbstlernprozess

Ziel des folgenden Kapitels ist die Entwicklung eines automatischen Wissensakquisitionsprozesses für das selbstlernende **Assistenzsystem Slassy**. Nach einer kurzen Diskussion des Begriffs Selbstlernen, werden aktuelle Theorien zu menschlichen Lernstrategien analysiert, um sie anschließend auf die vorliegende Problemstellung zu übertragen und damit die Grundlage für einen Selbstlernprozess zu bilden. Der Selbstlernprozess wird später in der Wissensakquisitions- bzw. Selbstlernkomponente von Slassy eingesetzt.

5.1 Vorüberlegung zum Begriff Selbstlernen

Häufig fallen im Zusammenhang mit wissensbasierten Systemen die Begriffe „intelligent“, „selbstoptimierend“ oder „smart“ [KK12b]. Dadurch soll ausgedrückt werden, dass das wissensbasierte System die kognitiven Fähigkeiten und Lernmechanismen des Menschen nachahmt. Wissensbasierte Systeme, bei denen sogenannte künstliche neuronale Netze zum Einsatz kommen, sind ein Beispiel für diesen Trend der Mechanismennachahmung. Trotz weltweiter Forschungsbemühungen ist ein Durchbruch zu einer „echten“ künstlichen Intelligenz jedoch noch nicht in Reichweite. Teilweise wird auch das Konzept der KI oder das intelligenter Systeme grundsätzlich angezweifelt [BD11].

Das im Rahmen dieser Arbeit zu entwickelnde wissensbasierte System soll nicht den Anspruch erheben, „intelligent“ handeln zu können. Es wird aber im Folgenden der Versuch unternommen, die verschiedenen Theorien zu Lernstrategien des Menschen zu analysieren. Auf der Basis dieser Analyse werden Aspekte ausgewählt, die sich zur Lösung der vorliegenden Problemstellung nutzen, anpassen oder kombinieren lassen. Eine Recherche zum Begriff „Selbstlernen“ führt in der deutschsprachigen Literatur hauptsächlich in die Bereiche der Didaktik und des Wissensmanagements. Unter der englischen Entsprechung „self-learning“ des Begriffs lassen sich Arbeiten zusammenfassen, die die Nutzung künstlicher neuronaler Netze oder die Verbesserung sogenannter Fuzzy-Logiken (siehe [XL87]) zum Ziel haben. Auch hier adressieren die Autoren die Lernmechanismen und nicht die eigentlichen Lernstrategien.

Unter Selbstlernen wird im Rahmen dieser Arbeit die Eigenschaft verstanden, den Wissensakquisitionsprozess (das *Lernen*) selbstständig zu gestalten und die Steuerung des Lernprozesses zu übernehmen. Ein solcher Selbstlernprozess kann den Lernerfolg eigenständig evaluieren und auf dieser Basis Lernziele, -wege sowie -inhalte festlegen. Ein selbstlernendes, wissensbasiertes System kann somit unabhängig vom Nutzer aus Daten, Wissen in Form

sogenannter Metamodelle erheben, diese Modelle optimal an die Fallbeispiele anpassen und anschließend das bestgeeignete Modell identifizieren. Trotz eines hohen Automatisierungsgrades muss die Entscheidung des Systems anschließend für den Nutzer nachvollziehbar sein. Manuelle Eingriffe sind nur zur Änderung des Selbstlernprozesses zulässig, z. B. bei Update der Data-Mining Software oder bei Änderungen der zugrundeliegenden Datenbank.

5.1.1 Menschliches Lernen

Die Mathematik bildet als Wissenschaft des *Lernens* die Ergänzung zur Didaktik, die sich mit dem Vorgang des *Lehrens* beschäftigt [KK12b]. Lernprozesse treten bei Menschen immer in Folge von Umwelteinräumen, Wahrnehmungen und Erfahrungen auf, die aufgenommen und verarbeitet werden. Erst dieses Prinzip ermöglicht ein Existieren und Zurechtfinden in einer Umwelt, mit der interagiert werden muss. Zur Beschreibung des Ablaufs von Lernprozessen existieren unterschiedliche Theorien. Diese sind in Bild 36 zusammengefasst.

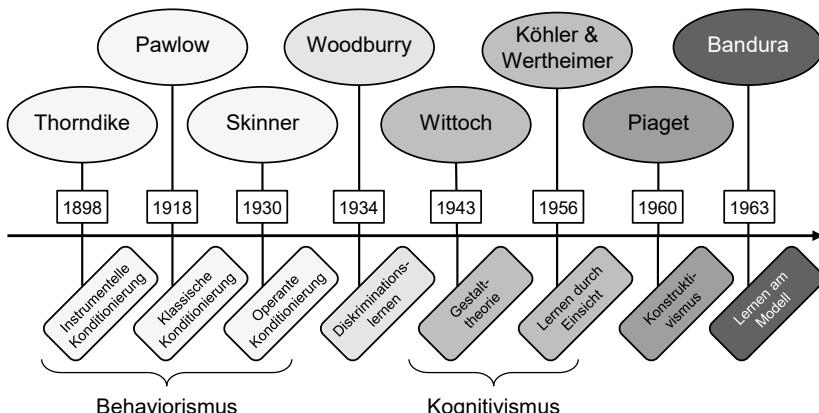


Bild 36: Entwicklungsübersicht der menschlichen Lerntheorien nach [FF11; KK12b]

Alle Theorien enthalten implizit die Fähigkeit zur Generalisierung und zur Diskrimination als eines der wichtigsten Elemente des Lernens. Von *Generalisierung* wird gesprochen, wenn unbekannte Situationen durch Anwendungen von altem Wissen bewältigt werden. Dabei wird eine Transferleistung erbracht, in der bekannte Inhalte einer Situation registriert und bereits erlernte Lösungsverfahren angewandt werden [Tho31; Tho32]. Die *Diskrimination* hingegen ermöglicht es, Unterschiede zwischen Reizen und Reaktionen oder einzelnen Situationen zu registrieren und das menschliche Verhalten entsprechend anzupassen [Wei89; Ski38].

Die Idee des *Behaviorismus* wurde ca. ab Mitte des 19. Jahrhunderts entwickelt und fasst Verhaltens- und Lerntheorien zusammen, die auf dem Reiz-Reaktions-Modell basieren und allgemein als Konditionieren bezeichnet werden. Die Erforschung des Lernens wird

im Behaviorismus auf Experimente zur Entstehung und Veränderung von Reiz-Reaktion-Beziehungen reduziert. Bei Erklärungsversuchen werden keine inneren physiologischen Vorgänge (z. B. im Nervensystem) der Individuen berücksichtigt, sondern die Interaktionen von Individuum und Umwelt als „Black-Box“ beschrieben. Hinzu kommt die Überzeugung, dass die Gesetzmäßigkeiten einer Reiz-Reaktion-Beziehung und ihrer Herausbildung allgemeingültig sind. Erste Behavioristen gingen davon aus, dass Untersuchungen an Tieren den Schlüssel zu den Gesetzmäßigkeiten der Reiz-Reaktion-Beziehungen beim Menschen liefern können. Die bekanntesten behavioristischen Ansätze sind die klassische Konditionierung nach Pavlov und die instrumentelle bzw. operante Konditionierung nach Thorndike bzw. Skinner. [Ski78]

Eine direkte Weiterentwicklung aus dem Behaviorismus ist das *Diskriminationslernen*. Es untersucht die Fähigkeit der Unterscheidung von guten und schlechten beziehungsweise relevanten und irrelevanten Reizen. Doch nicht nur die Differenzierung verschiedener Reize und deren Konsequenzen, sondern auch die Integration von Reizkombination als Reizmuster ist ein wichtiger Faktor dieser Theorie. Die ersten Versuche für die Untersuchung der kombinierten Reizmuster gehen auf Charles Woodbury zurück. Die mittlerweile vorherrschende Meinung der Mathematik ist, dass die menschliche Physiologie und Psychologie wesentlich komplexer arbeitet. Daher werden Ansätze aus dem Behaviorismus heutzutage hauptsächlich bei Tierversuchen angewandt. [FH20; KK12b]

Ab den 1960er und 1970er Jahren löste der *Kognitivismus* den Behaviorismus zunehmend als vorherrschendes Forschungsparadigma in der Psychologie ab. Der Kognitivismus geht davon aus, dass der Erwerb komplexer intellektueller Fähigkeiten durch behavioristische Prinzipien allein nicht zu erklären ist. Kognitive Prozesse wie Denken, Erinnern, Wahrnehmen und die Verwendung der Sprache müssen besondere Berücksichtigung finden. Lernen ist demnach kein reaktiver, sondern ein bewusster Prozess, der auf Denken und Verstehen basiert. Innerhalb des Kognitivismus werden u. a. die Lernstrategien „Lernen am Modell“, „Lernen durch Einsicht“ und „implizites Lernen“ unterschieden. [Ban77; KK12a]

Lernen am Modell findet statt, wenn Personen (oder auch Tiere) durch die Beobachtung des Verhaltens anderer Individuen („Vorbilder“) und der darauffolgenden Konsequenzen sich bewusst oder unbewusst neue Verhaltensweisen aneignen. Auch die Änderung bestehender Verhaltensmuster kann dadurch bewirkt werden. Dabei entscheiden die Motivation der lernenden Person und die Wertigkeit der eintretenden Konsequenzen darüber, ob sich das beobachtete Verhalten tatsächlich festigt. Nach BANDURA beeinflussen weiterhin Faktoren wie Aufmerksamkeit, Gedächtnisleistung, motorische Reproduktionsfähigkeit und Motivation die Aneignung und Nachahmung von beobachtetem Verhalten [Ban77]. Das Lernen am Modell setzt keine eigenen Erfahrungen voraus, es wird lediglich durch Beobachten gelernt. Die von extern eingehenden Informationen werden mit einer vorgegebenen aber veränderbaren kognitiven-mental Struktur des Beobachtenden verarbeitet. Denken wird demnach als Prozess der Informationsverarbeitung verstanden. [KK12a; KK12b]

Beim *Lernen durch Einsicht* (s. Bild 37) strukturiert das Individuum ein vorliegendes Problem denkend um und reorganisiert es „im Kopf“, um Handlungsalternativen für die Lösung

des Problems zu finden. Diese Aneignung und Umstrukturierung von Wissen beruht auf der Nutzung von kognitiven Fähigkeiten wie Wahrnehmungs- und Vorstellungsvermögen sowie der Möglichkeit, Vergleiche durchzuführen und daraus zu schlussfolgern. Lernt eine Person durch Einsicht, dann bedeutet dies, dass sowohl die Ursache-Wirkung Zusammenhänge eines Sachverhalt als auch der Sinn und die Bedeutung der daraus entstehenden (neuen) Situation erkannt und verstanden wurden. Für externe Beobachter ist die Verhaltensänderung mitunter sehr überraschend, da sie innerhalb kurzer Zeit auftreten kann. Diese Form des Lernens erfordert besondere Fähigkeiten zur Entwicklung von parallelen Konzepten, deren Analyse und Bewertung sowie zum Vergleich der verschiedenen Alternativen. [FF11]



Bild 37: Die sechs Phasen beim Lernen durch Einsicht.

Durch *implizites Lernen* werden die Lerninhalte unbewusst bzw. intuitiv aufgenommen und gespeichert, ohne dass dem lernenden Individuum die zugrunde liegenden Zusammenhänge einer Situation bewusst sind. Es läuft unbewusst und ohne den Einsatz analytischer Strategien wie Hypothesengenerierung und Hypothesentest ab, die beim expliziten Lernen erforderlich sind. Wie die Bezeichnung bereits vermuten lässt, mündet implizites Lernen immer in implizitem Wissen (s. Abschnitt 2.1.2, S. 9). Personen nehmen also ohne eine spezielle Absicht in komplexen Situationen Wissen auf, das sich später schwer verbalisieren lassen wird. [HE13; KK12a]

Als bisher jüngste Disziplin versteht der *Konstruktivismus* unter Lernen eine individuelle Entwicklung („Konstruktion“) von Wissen, die bei jedem Menschen anders abläuft. Für die Hauptaussage des Konstruktivismus bedeutet dies, dass es keine objektive Realität gibt, die der Lehrende dem Lernenden vermitteln kann, sondern dass Wissen immer und in jeder Form einen subjektiven Charakter besitzt. Die Wissensaneignung beruht dadurch immer auf den Randbedingungen des Lernens und wird durch unbewusste Faktoren gesteuert, die deshalb nur schwer beeinflussbar sind. Dadurch sind die Mechanismen des Lernens aus konstruktivistischer Sicht auch nicht ohne weiteres verallgemeinerbar. [Mie01]

5.1.2 Übertragung auf die Problemstellung

Nach der Darstellung der verschiedenen Lernstrategien wird im Folgenden diejenige identifiziert, nach deren Vorbild sich ein Selbstlernprozess für Slassy entwickeln ließe.

Das Reiz-Reaktions-Modell des Behaviorismus zeichnet sich dadurch aus, dass Lerninhalte durch Vermittlung von richtig/ falsch - Meldungen im Anschluss an ein bestimmtes Verhalten, das durch einen äußeren Reiz ausgelöst wurde, konditioniert werden. Während sowohl Reiz als auch Verhalten beobachtbar sind, ist der Lernprozess selbst durch eine „Black-Box“ ausgeblendet. Eine Nutzung dieses Konzepts für einen Selbstlernprozess erscheint schwierig. Durch die Fallbeispiele in den Daten der Fertigungsprozessentwicklung ist Feedback

an das wissensbasierte System, in Form von richtig/falsch-Meldungen, prinzipiell denkbar. Der Forderung nach einer Berücksichtigung der speziellen Design for Manufacture Problematik der Blechmassivumformung (s. Abschnitt 3.2, S. 58) kann somit jedoch nicht nachgekommen werden. Weiterhin ist anzumerken, dass das Lernen durch Konditionieren langsam abläuft und bereits gelernte Inhalte aufwändig in iterativer Form abtrainiert werden müssen. Der „Black-Box“ Charakter der behavioristischen Theorie blendet den eigentlichen Lernprozess aus, was zu Intransparenz und mangelnder Nachvollziehbarkeit der Ergebnisse führt.

Die Theorie des Konstruktivismus besagt, dass es keine objektive Realität gibt, die vom Lehrenden an den Lerner vermittelt werden kann. Lernprozesse sind in höchstem Maße subjektiv, denn was eine Person unter bestimmten Bedingungen lernt, hängt stark von seinen Motiven und Randbedingungen ab. Darüber hinaus spielt die sozio-kulturelle Lernumgebung eine wichtige Rolle. Insbesondere dieser Faktor macht den Ausgang eines konstruktivistischen Lernprozesses relativ unberechenbar. Die Wissensaneignung wird demnach durch Faktoren gesteuert, die unbewusst ablaufen und somit nur schwer beeinflussbar sind. Bei gleichen Lernrandbedingungen, können gleiche Lernstrategien zu unterschiedlicher Qualität des erworbenen Wissens führen. Die mangelnde Verallgemeinerbarkeit und Wiederholgenauigkeit von konstruktivistischen Lernprozessen lässt ihre Nutzung für den Selbstlernprozess als schwierig erscheinen.

In den Ansätzen des Kognitivismus wird Lernen nicht als reaktiver Prozess aufgefasst, der nur auf äußere Reize hin angestoßen, sondern bewusst vom lernenden Individuum initiiert wird. Das Lernen selbst kann dabei intuitiv erfolgen und der dabei stattfindende Ablauf schwer oder gar nicht verbalisiert werden (Implizites Lernen). Das Erlernen der Muttersprache durch Kleinkinder ist hierfür ein Beispiel. Das Lernen am Modell kann ebenfalls intuitiv/implizit ablaufen, wenn die Verhaltensweisen des Modells ohne explizite Erläuterung nachgeahmt werden. Dennoch existiert mit dem Lernen durch Einsicht eine Modellbeschreibung, die explizit die Vorgänge beim Wissenserwerb beschreibt und dadurch objektiv und transparent macht.

Die Phasen des Lernens durch Einsicht wurden aus lernpychologischer Sicht beschrieben (s. Bild 37, S. 72). Wird diese Abfolge verallgemeinert zur Beschreibung eines Wissensakquisitionsprozesses dargestellt, lassen sich folgende Schritte identifizieren:

1. Aufstellen von Konzepten zur Erklärung eines unbekannten Sachverhalts
2. Anwendung der Konzepte zur Problembeschreibung
3. Optimierung/Anpassung der Konzepte und Vergleich der neuen Varianten
4. Objektive Entscheidung für das passendste Konzept
5. Abspeicherung im Gedächtnis für zukünftige Anwendung
6. Wiederholung des Prozesses bei neuer Sach- oder Problemlage

An dieser Stelle ist anzumerken, dass „Konzept“ nicht im technischen Sinn zu verstehen ist, wie beispielsweise das Lösungskonzept „Zahnradgetriebe“ für die Problemstellung „Über-

tragung einer Drehbewegung“. Ein Konzept soll hier abstrakt als eine Möglichkeit oder ein Modell verstanden werden, das einen bestimmten Sachverhalt erklären kann. Bild 38 beschreibt die genannte Abfolge als Prozesschaubild.

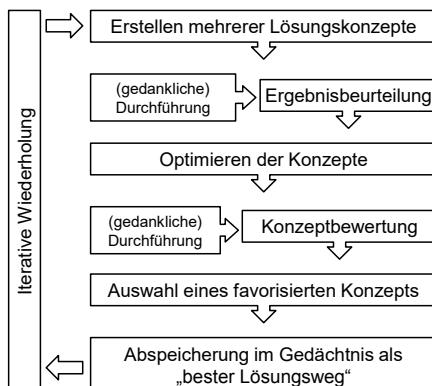


Bild 38: Menschlicher Lernprozess angelehnt an das Lernen durch Einsicht.

Der Vorteil, den ein solches Vorgehen mit sich bringt, ist eine enorme Flexibilität während der Wissensakquisition. Es wird nicht an einem einzigen Modell festgehalten, welches mit viel Aufwand so lange umstrukturiert wird, bis es den Sachverhalt hinreichend genau wiedergibt, dafür jedoch sehr komplex und schlecht nachvollziehbar ist. Vielmehr wird zunächst eine Vielzahl an Konzepten in Betracht gezogen, mit geringem Aufwand angepasst (optimiert) und nach objektiven Maßstäben ein möglichst verlässliches Modell ausgewählt. Ein weiterer Vorteil bei dieser Vorgehensweise ist, dass das Prinzip der Parsimonie, auch als Ockham's Razzor bekannt, berücksichtigt werden kann. Dieses besagt, dass von mehreren gültigen Erklärungen für ein und denselben Sachverhalt immer die einfachste Theorie allen anderen vorzuziehen ist [Sob15]. Somit bleiben Nachvollziehbarkeit und Transparenz gewahrt.

Das aus dem Lernen durch Einsicht abgeleitete Vorgehen kann auf die Problemstellung der vorliegenden Arbeit übertragen werden. Der „unerklärte“ Sachverhalt entspricht dabei den Zusammenhängen zwischen der Geometrie eines blechmassivumgeformten Bauteils und den für die Design for Manufacture Analyse dieses Bauteils relevanten Zielgrößen (s. Abschnitt 3.2, S. 58). Er ist implizit in den Daten der Fertigungsprozessentwicklung enthalten. Passende Konzepte, die diesen Zusammenhang abbilden können werden im weiteren Verlauf *Metamodelle* bezeichnet. Als „Modell eines Modells“ sind sie in der Lage auf Basis eines Vektors an Eingangsgrößen die Ausprägung einer Zielgröße vorherzusagen und somit den Sachverhalt zu erklären. Die Eingangsgrößen beschreiben dann beispielsweise die Geometrie blechmassivumgeformter Bauteile (z. B. Längen, Breiten, Winkel) des CAD-Modells und die Zielgröße entspricht dem Umformgrad, der sich bei dieser zu fertigenden Geometrie ergeben wird. Die Optimierung eines Metamodells kann auf verschiedenen Strategien be-

ruhen, richtet sich jedoch in jedem Fall nach den eingesetzten Metamodellen (s. Abschnitt 6.3.1, S. 95). Eine detaillierte Beschreibung der Umsetzung des Optimierungsansatzes kann Abschnitt 5.3 entnommen werden. Die Beurteilung eines Konzepts geschieht beim Menschen durch intuitives Abwagen, bspw. zwischen dem Risiko das eingegangen werden muss, um ein potentielles Ergebnis zu erzielen (Aufwand zu Nutzen). Ein Metamodell muss objektiv daran beurteilt werden, wie gut es eine Zielgröße vorhersagen oder einen Sachverhalt abbilden kann. Im Abschnitt 2.4.2 (S. 47) wurden die gängigen Performanzkenngrößen für Metamodelle bereits beschrieben. Wichtig bei der Beurteilung ist eine hohe Robustheit bzw. Verlässlichkeit des ermittelten Werts. Liegt für jedes entwickelte und optimierte Metamodell eine Beurteilung vor, so wird im nächsten Schritt das passendste Konzept bzw. Metamodell gewählt. Diese Auswahl muss nach objektiven Kriterien und Vorgaben erfolgen, um dem späteren Nutzer von Slassy das verlässlichste Metamodell (Design for Manufacture Wissen) bereitstellen zu können. Der prinzipielle Ablauf des Selbstlernprozesses ist in Bild 39 dargestellt

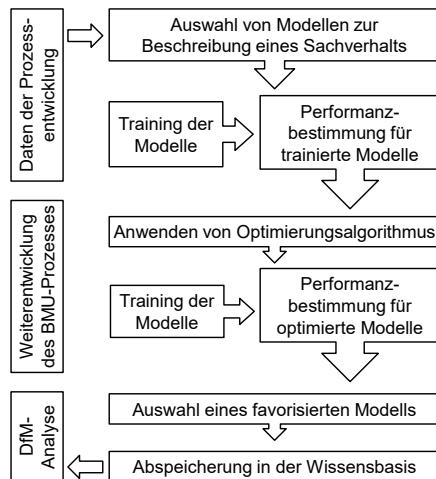


Bild 39: Entwurf des abgeleiteten Selbstlernprozesses.

Den Ausführungen ist zu entnehmen, dass den Aspekten

- Performanzbestimmung zur Beurteilung eines Metamodells,
- Optimierung der Modelle sowie
- Auswahl des bestgeeigneten Modells

eine zentrale Bedeutung beigemessen wird. Daher werden sie in den folgenden Abschnitten 5.2, 5.3 und 5.4 ausführlich behandelt.

5.2 Statistisch robuste Performanzbestimmung

Im vorangegangenen Abschnitt wurde hervorgehoben, dass die Schätzung des Kennwerts zur Beurteilung der Prognosequalität eines Metamodells robust und verlässlich erfolgen muss. Diese Performanz ist ein wichtiges Qualitätsmerkmal für einen Data-Mining Prozess und kann im Kontext dieser Arbeit anschaulich mit einer „Wissensgüt“ gleichgesetzt werden. Sie bildet außerdem die Basis für Vergleiche verschiedener Metamodelle und muss aus diesem Grund ein unabhängiger und robuster Kennwert sein. Im Abschnitt 2.4.2 wurde bereits die k -fache Kreuzvalidierung als gängiges Verfahren für die Schätzung der Prognosegüt bzw. Performanz von Regressionsmodellen vorgestellt (s. Bild 25, S. 48). Die zufällige Einteilung der Datentupel im Datensatz D in k Teildatensätze D_k und die Reihenfolge der Datentupel in D dürfen dabei keinen Einfluss auf das Ergebnis der Performanzschätzung haben, da dies der Forderung nach einem unabhängigen und robustem Bewertungskriterium entgegen steht, wie im Folgenden gezeigt wird.

Für eine zufällige Einteilung dieser Tupel nutzen Data-Mining Programme *Zufallszahlen*. Während manuelle Methoden wie Urnenziehen oder Zufallszahltabellen in gewissem Maße echte Zufallszahlen erzeugen (s. [Hel74] oder [ZS01]), sind rechnerunterstützte Methoden bei der Zufallszahlenberechnung auf einen Startwert, den sogenannten *random seed* angewiesen. Da gleiche Startwerte bei deterministischen Zufallsalgorithmen auch gleiche Zufallszahlen erzeugen, wird im Allgemeinen von Pseudozufallszahlen gesprochen. Wird die gleiche Pseudozufallszahl (konstanter random seed) für die Einteilung der k Teildatensätze D_k verwendet, so befinden sich, bei gleichem Aufbau von D , immer die gleichen Datentupel in den Teildatensätzen D_k . Dies resultiert bei jeder Wiederholung des Kreuzvalidierungsprozesses in unveränderten Vorhersagefehlern. Bild 40 verdeutlicht dies. Gezeigt ist ein Data-Mining Prozess, der einen frei verfügbaren Datensatz¹³ zum Training eines Regressionsmodells nutzt und die Prognosequalität des Modells durch eine 10-fache Kreuzvalidierung schätzt. Als Schätzer wird der mittlere relative Fehler zwischen wahrer und vorhergesagter Zielgröße ausgegeben. Wird dem Operator der Kreuzvalidierung ein konstanter random seed zugewiesen, so ändert sich der relative Fehler auch bei mehrmaliger Wiederholung nicht. Dies vermittelt den Eindruck, dass der berechnete Prognosefehler wiederholbar und somit robust ist. Wird jedoch der random seed geändert, so ergeben nacheinander ausgeführte Kreuzvalidierungen einen anderen, aber nach wie vor konstanten Performanzwert. Der Grund hierfür liegt in der veränderten Zuordnung der Datentupel aus D in die Teildatensätze D_k . Eine solche Änderung der Datentupel in D kann auftreten, wenn weitere Daten hinzugefügt oder ausgetauscht werden. Der Schätzwert der Metamodellperformanz ist somit, nach Standardvorgehensweise, kein robuster und unabhängiger Kennwert.

Zur Begegnung dieser Problematik werden die Datentupel in D gezielt durchmischt, die random seed Werte bei jedem Durchlauf variiert und eine eine mehrfache Wiederholung der Kreuzvalidierung mit anschließender Mittelwertbildung durchgeführt. Bild 41 zeigt die Umsetzung dieses Konzepts als prinzipiellen Data-Mining Prozess. Zur robusten Schätzung

¹³ Details zum Datensatz können dem Anhang Seite 159 entnommen werden.

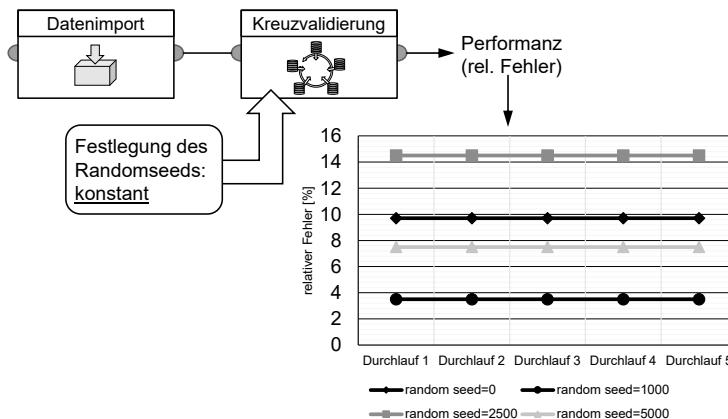


Bild 40: 10-fache Kreuzvalidierung bei verschiedenen (konstanten) Werten der Pseudozufallszahl

der Performanz werden die Datentupel in D zunächst zufällig gemischt. Dies erfolgt über den „Shuffle“ Operator, dem der globale, variable random seed zugewiesen wird. Die mehrmalige k -fache Kreuzvalidierung und Mittelwertbildung wird durch den Operator „Loop & Average“ erreicht, in dem der Operator für die Kreuzvalidierung verschachtelt integriert wird. Diesem wird für die Aufteilung der Datentupel in die k Datensätze D_k ebenfalls der globale, variable random seed zugewiesen.

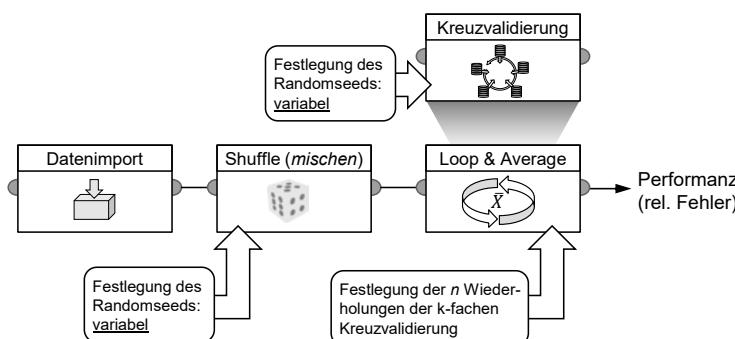


Bild 41: Data-Mining Prozess zur robusten Abschätzung der Metamodellperformanz.

Abschließend muss für den „Loop & Average“ Operator die Anzahl der n Wiederholungen festgelegt werden, aus denen der Mittelwert der Kreuzvalidierungen berechnet wird. Hierbei ist zu beachten, dass mit einer geringen Anzahl an Wiederholungen eine hohe Unsicherheit des Prognoseschätzwerts einhergeht. Wird die Anzahl der Wiederholungen erhöht, so

steigt zwar die Robustheit des Performanzkennwerts, allerdings nimmt auch die benötigte Rechenzeit zu. Es muss ein Kompromiss aus Robustheit und Rechenzeit gefunden werden. Die Boxplots in Bild 42 zeigen diesen Zusammenhang. Für die Erstellung eines einzelnen Boxplots wird der in Bild 41 dargestellte Data-Mining Prozess 50 mal durchgeführt (Testdatensatz aus Anhang 8, S. 159). Jede Durchführung beinhaltet eine zehnfache ($k = 10$) Kreuzvalidierung die n -mal wiederholt und gemittelt wird. Dabei erfolgt eine sukzessive Erhöhung von n ($n \in [1, 2, 3, 4, 5, 10, 25, 50, 100, 500, 1000]$).

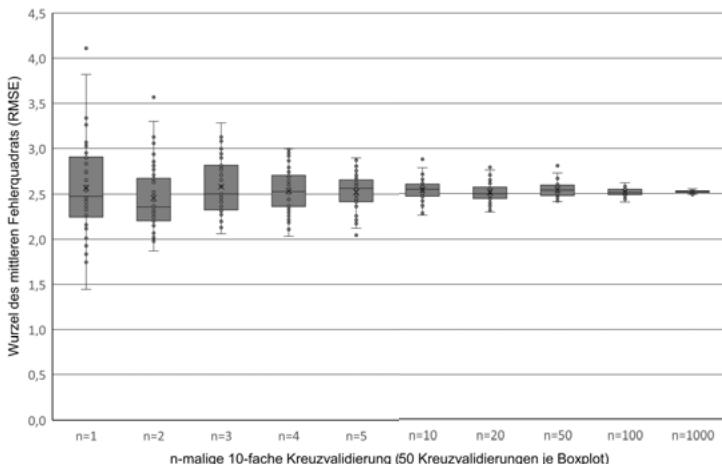


Bild 42: Boxplot der Performanzen mit steigender Anzahl von „Loop & Average“ Durchläufen. (X-Achse: n-malige 10-fache Kreuzvalidierung)

Jede Box umfasst 50% aller berechneten Mittelwerte, wobei jeder Einzelwert durch ein \times gekennzeichnet ist. Der Median, das geometrische Mittel einer Stichprobe, ist durch eine schmale Volllinie innerhalb einer Box gekennzeichnet. Fallen die Mitte einer Box und der Median zusammen, ist von einer annähernd symmetrischen Verteilung auszugehen. Es muss an dieser Stelle betont werden, dass bei KDD-Prozessen die Evaluation eines Modells üblicherweise nur auf einer einmalig durchgeführten 10-fachen Kreuzvalidierung beruht (z. B. [Hit12; Sch13]), was einem Punkt im linken Boxplot ($n=1$) entspricht. Hierin liegt eine sehr hohe Unsicherheit des Prognoseschätzwerts begründet, denn der Anwender hat in diesem Fall keine Möglichkeit festzustellen, ob das Modell über- oder unterschätzt wird. Ein Wert von $n > 1$ muss festgelegt werden, um eine ausreichende Robustheit bei vertretbarer Berechnungsdauer zu gewährleisten. In Vorversuchen wurde ermittelt, dass für $n= 20$ verlässliche Performanzwerte in einer vertretbaren Zeit berechnet werden können. Ergebnis dieser n -fach wiederholten k -fachen Kreuzvalidierung ist ein sogenannter *Performanzvektor*. Hierunter wird im weiteren Verlauf ein Vektor der Länge $n \cdot k$ verstanden, dessen Zeilenwerte mit den Vorhersagefehlern besetzt sind, die in den einzelnen Iterationen der Kreuzvalidierung ermittelt werden.

5.3 Überwachtes Lernen und Optimieren der Metamodelle

In Abschnitt 5.1.2 wurde festgestellt, dass die Optimierung der alternativen Konzepte bzw. deren Anpassung an die gegebene Problemstellung ein essentieller Bestandteil eines Selbstlernprozesses sein muss. Die Optimierung eines Metamodells hat das Ziel, dieses Modell sowohl optimal an den zur Verfügung stehenden Datensatz anzupassen als auch eine Überanpassung (engl. *overfitting*) zu vermeiden.

Grundsätzlich existieren beim Trainieren eines Metamodells verschiedene Möglichkeiten für dessen Optimierung. Ziel ist ein Set an optimalen Trainingsparametern zu finden, um das Metamodell an den jeweiligen Trainingsdatensatz anzupassen und gleichzeitig ein Overfitting zu vermeiden. Unter Parameterset wird hier eine Menge an Werten verstanden, durch die das Training eines Metamodells beeinflusst werden kann. Bild 43 zeigt für eine Auswahl an Trainingsoperatoren Beispiele für derartige Parameter. An dieser Stelle wird darauf hingewiesen, dass die Darstellungen zwar der Data-Mining Software Rapidminer® entnommen sind, jedoch lassen sich entsprechende Einstellungen in jeder Data-Mining Software vornehmen.

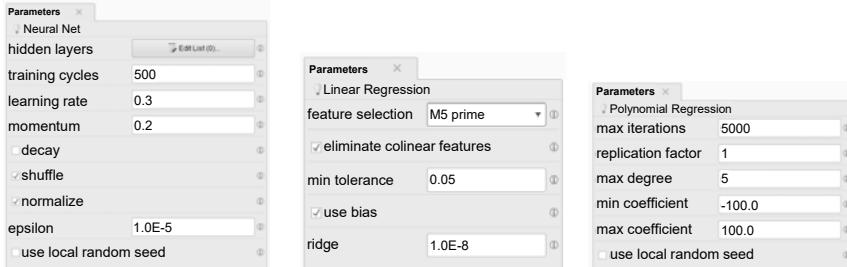


Bild 43: Beispiele von konfigurierbaren Parametern für das Training von künstlichen neuronalen Netzen, linearen und polynomialen Regressionsmodellen in Rapidminer®

Die folgenden Ausführungen beziehen sich auf dem Einsatz der Data-Mining Software Rapidminer®, sind jedoch in den zugrundeliegenden Theorien verallgemeinerbar und auf andere Data-Mining Werkzeuge übertragbar. Sehr weitreichende Möglichkeiten zur Optimierung sind in Rapidminer® durch den „Optimize Grid“ Operator sowie den „Optimize Evolutionary“ Operator gegeben. Der „Optimize Quadratic“ Operator steht ebenfalls zur Verfügung und rechnet als sog. Greedy-Algorithmus zwar schnell, löst Probleme jedoch nicht immer optimal. Zudem lässt er, wie auch der „Optimize Evolutionary“ Operator, keine Berücksichtigung diskreter Parameter bei der Optimierung zu, was nur durch den „Optimize Grid“ Operator ermöglicht wird. Die Optimierungsstrategie des „Optimize Grid“ Operators sieht vor, nacheinander (*schrittweise*) alle möglichen Parameterkombinationen für einen Trainingsoperator zu durchlaufen und die Konfiguration auszugeben, mit

denen die größtmögliche Performanz (kleinster Vorhersagefehler) erreicht wird. Die Anzahl der Optimierungsdurchläufe ist dabei durch die Menge an Parametern sowie deren Stufen bestimmt. Werden z. B. drei Parameter mit je zwei diskreten Stufen und ein kontinuierlicher Parameter, dessen Wert in festgelegten Grenzen über 20 Stufen durchlaufen wird, kombiniert, so ergeben sich 168 einzelne Iterationen.

Der „Optimize Evolutionary“ Operator verfolgt die Suche nach der optimalen Parameterkonfiguration auf Basis eines evolutionären Algorithmus (s. Bild 44). Damit unterscheidet er sich grundlegend von der Strategie des „Optimize Grid“ Operators, der quasi blind nach einem festen Versuchsplan angewandt wird. Streng genommen handelt es sich bei dem implementierten Operator um einen genetischen Algorithmus, der zu den bekanntesten Vertretern evolutionärer Optimierungsverfahren zählt. Der Algorithmus beginnt mit der probabilistischen Generierung einer Startpopulation an Individuen. Jedes Individuum ist durch sein „Genom“ eindeutig beschrieben. Dieses Genom entspricht einer bestimmten Ausprägung des Parametersets eines zu optimierenden Modells. In der Startpopulation werden zufällig Elternindividuen selektiert und untereinander rekombiniert, um Kindindividuen zu erzeugen. Das Genom dieser Individuen der zweiten Generation wird einer zufälligen Mutation unterzogen. Diese Mutation kommt jedoch nur mit einer sehr geringen Wahrscheinlichkeit zur Anwendung. Sie sichert die Erreichbarkeit aller Punkte im Suchraum und garantiert eine Grunddiversität in der Gesamtpopulation. Die Performanzen der mutierten Kindindividuen werden ermittelt und mit denen der Elterngeneration verglichen. Im Falle einer Verbesserung werden einzelne Elternindividuen ausgetauscht. Danach wiederholt sich das Vorgehen iterativ bis der Algorithmus eine Konvergenz feststellt oder ein anderes Abbruchkriterium erreicht ist, z. B. eine maximale Anzahl an Generationen.

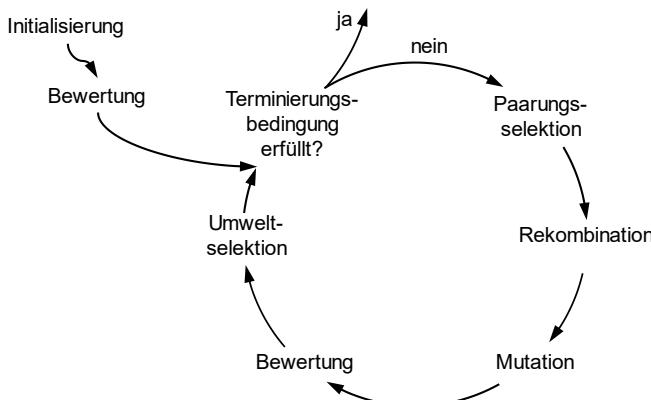


Bild 44: Der Ablauf eines evolutionären Algorithmus nach [Wei15].

Für die Operatoren zur Optimierung der Parametersets muss nun eine geeignete Umsetzung im Data-Mining Prozess festgelegt werden. Hierbei ist wiederum der Grundsatz der Robustheit zu berücksichtigen. Das betrifft zunächst die Performanzschätzung, die in je-

der Iteration eines Optimierungsdurchlaufs erfolgt. Um auch hier eine möglichst geringe Varianz der geschätzten Prognosequalität zu erreichen, wird das in Abschnitt 5.2 entwickelte Vorgehen für die iterativen Performanzschätzungen innerhalb einer Optimierung genutzt. Dadurch ergibt sich der in Bild 45 prinzipiell dargestellte Aufbau einer Optimierungseinheit. Die Implementierung dieser Struktur in Rapidminer® ist unabhängig vom eingesetzten Optimierungsoperator und der gewählten Trainingsfunktion.

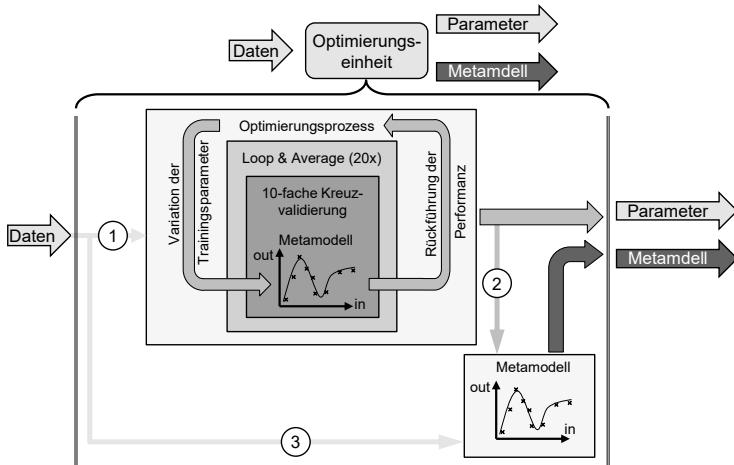


Bild 45: Der prinzipielle Aufbau der Optimierungseinheit.

Der Ablauf eines Optimierungslaufs beginnt mit der Übergabe der Daten an die Optimierungseinheit. Hier wird zunächst der Optimierungsprozess zur Identifikation des bestgeeigneten Sets an Trainingsparametern gestartet (①). Innerhalb dieses Prozesses werden die Einstellungen für die Trainingsparameter des entsprechenden Metamodells variiert und durch eine robuste Performanzschätzung (mehrfach wiederholte 10-fache Kreuzvalidierung) das sich mit diesen Parametern ergebende Metamodell validiert. Durch eine Rückführung der Performanzwerte können diese anschließend verglichen werden. Das Parameterset, das zum Modell mit dem geringsten Vorhersagefehler geführt hat, wird anschließend an einen Trainingsoperator übergeben (②) der mit den gleichen Trainingsdaten des Optimierungsprozesses (③) das optimierte Metamodell erzeugt und dieses anschließend zum Export an die Optimierungseinheit übergibt. Der Vollständigkeit halber wird neben diesem optimierten Modell das zugehörige Parameterset exportiert.

Beide Teilespekte, die robuste Performanzschätzung (Abschnitt 5.2) sowie die auf diese Methode zurückgreifende Optimierung der Metamodelle (Abschnitt 5.3), werden im Folgenden als ROPE-Prozess (engl. **R**obust **O**ptimization and **P**erformance **E**stimation) bezeichnet.

5.4 Auswahl des bestgeeigneten Metamodells

Der letzte Schritt des Selbstlernprozesses beinhaltet, wie in Abschnitt 5.1.2 gezeigt, die Auswahl des bestgeeigneten Metamodells aus einer Grundmenge an verfügbaren Modellen. Die Formulierung „bestgeeignetes“ Modell wird an dieser Stelle bewusst gewählt, denn bereits in [Box79] wird festgestellt, dass „alle Modelle falsch, manche aber brauchbar sind“. Für die Identifikation dieses Modells wird im Folgenden eine Methodik erarbeitet. Um eine möglichst hohe Flexibilität der Wissensakquisitionskomponente zu gewährleisten, soll die Methodik unabhängig von der Anzahl der verfügbaren Metamodelle anwendbar sein.

Der ROPE-Prozess trainiert für einen gegebenen Datensatz D eine bestimmte Anzahl M an Metamodellen und schätzt deren Prognosequalität als Mittelwert \bar{x}_{M_i} für eines der in Tabelle 4 angegebenen Gütekriterien. Ein zunächst naheliegender und trivialer Schritt wäre die Wahl des Modells mit dem geringsten Fehler bzw. der höchsten Prognosegüte. Es muss jedoch berücksichtigt werden, dass auch das Fehlermaß selbst einem Fehler bzw. einer Varianz unterliegt, die zudem nicht deterministisch, sondern rein probabilistisch zustande kommt. Zwei wesentliche Ursachen hierfür, der Einfluss des Aufbaus der Eingangsdaten in D sowie der des random seeds in der Kreuzvalidierung wurden bereits im Abschnitt 5.2 erläutert. Werden also zwei Metamodelle A und B derart miteinander verglichen, kann aufgrund der Varianz der Prognosegüte (s. Bild 42) keine verlässliche Aussage getroffen werden.

Ziel ist es das Modell zu identifizieren, welches sich *signifikant* von den anderen Modellen unterscheidet. Ausgangspunkt ist die Annahme, dass zu jedem Metamodell eine Grundgesamtheit existiert, die die zugrundeliegende Prognosegüte in Form eines Mittelwerts μ_{M_i} beschreibt. Durch die n -malige k -fache Kreuzvalidierung werden Stichproben $\{e_{A1}, e_{A2}, \dots, e_{Ak}\}$ sowie $\{e_{B1}, e_{B2}, \dots, e_{Bk}\}$ aus beiden Grundgesamtheit entnommen und daraus die Mittelwerte \bar{x}_A und \bar{x}_B gebildet (s. Bild 46).

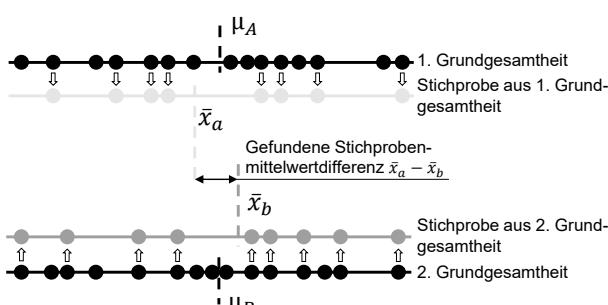


Bild 46: Schematische Darstellung für das Ziehen von Stichproben auf Basis von Kreuzvalidierungen.

Ein signifikanter Unterschied bedeutet nun, dass zwischen den Mittelwerten $\bar{\mu}_A$ und $\bar{\mu}_B$ der Grundgesamtheiten eine Differenz vorliegt, die mit hoher Wahrscheinlichkeit nicht zufällig

entstanden ist und auf deren Existenz mithilfe der Mittelwertdifferenz zwischen den Stichproben \bar{x}_A und \bar{x}_B geschlossen werden kann. Die Untersuchung eines solchen Sachverhalts wird mit dem Begriff *Hypothesentest* zusammengefasst. Zwischen zwei konkurrierenden Hypothesen wird auf Basis von Stichproben eine Entscheidung zugunsten einer der beiden Hypothesen gefällt. Die Literatur unterscheidet zwischen Nullhypothese H_0 und Alternativhypothese H_1 [RFHN14]. Entsprechend ihrer Bezeichnung besagt die Nullhypothese, dass zwischen zwei Merkmalen kein Unterschied vorhanden ist. Wird H_0 auf den Vergleich der beiden Mittelwerte \bar{x}_A und \bar{x}_B bezogen, können mehrere Hypothesenpaare gebildet werden. Gleichung 5.1 a) und 5.1 b) beschreiben jeweils eine *gerichtete* Alternativhypothese, während diese in Gleichung 5.1 c) *ungerichtet* ist.

$$H_0 : \mu_A - \mu_B = 0 \text{ bzw. } \mu_A = \mu_B \text{ versus} \begin{cases} a) & H_1 : \mu_A > \mu_B \\ b) & H_1 : \mu_A < \mu_B \\ c) & H_1 : \mu_A \neq \mu_B \end{cases} \quad (5.1)$$

Nachdem ein Hypothesenpaar formuliert wurde, erfolgt ein statistischer Test, aufgrund dessen die jeweilige Tragfähigkeit von H_0 bzw. H_1 bestimmt wird. Hierzu wird eine Prüfgröße aus den erhobenen (empirischen) Daten der Kreuzvalidierung berechnet. Wie diese Berechnung erfolgt, hängt vom jeweiligen Testverfahren ab, wobei jedes Verfahren unterschiedliche Anforderungen an die zu analysierenden Daten stellt. Die berechnete Prüfgröße wird mit ihrer zugrundeliegenden Verteilung verglichen, um festzustellen ob ihr Wert auffällig hoch oder niedrig ist, denn in diesen Fällen wird das als Beweis gegen die Nullhypothese gewertet. Die Verteilung der Prüfgröße wird unter der Annahme der Gültigkeit von H_0 als bekannt vorausgesetzt. Je nach Testverfahren wird bspw. von einer Standardnormalverteilung (z-Test) oder einer t-Verteilung (t-Test) der Prüfgröße ausgegangen. Die Grenze, ab der ein Wert als „auffällig“ gilt, wird als Signifikanzniveau bezeichnet. Der Vergleich der Prüfgröße mit dem zuvor festgelegten Signifikanzniveau α führt entweder zur Anerkennung von H_0 oder von H_1 und zur Ablehnung der jeweils unterlegenen Hypothese. Dabei kann es passieren, dass bspw. gegen die Nullhypothese entschieden wird, obwohl sie, bezogen auf die (unbekannten) Grundgesamtheiten der Prognosegüten von Modell A und Modell B, gültig wäre. Dies wird als Fehler 1. Art bezeichnet, wie in Tabelle 7 dargestellt.

Tabelle 7: Mögliche Fehlerarten bei statistischen Entscheidungen.

		Entscheidung	
		für H_0	gegen H_0
In Wahrheit gilt	H_0	korrekte Entscheidung	Fehler 1. Art (α -Fehler)
	H_0 nicht	Fehler 2. Art (β -Fehler)	korrekte Entscheidung

Ein Fehler 1. Art (α -Fehler) liegt vor, wenn die Nullhypothese, dass zwischen den Mittelwerten μ_{M_A} und μ_{M_B} kein Unterschied besteht, verworfen wird. Stattdessen wird fälschlicherweise angenommen, dass sich die Mittelwerte signifikant unterscheiden, obwohl dies in Wirklichkeit nicht zutrifft. Es wird irrtümlich eines der beiden Metamodelle für das bessere gehalten. Hier kommt das Signifikanzniveau α ins Spiel, denn es bezeichnet die Wahr-

scheinlichkeit, mit der die Ablehnung der Nullhypothese zu einem Fehler 1. Art führt. Der Versuchsleiter kann durch die Festlegung von α das Risiko, einen Fehler 1. Art zu begehen kontrollieren. Üblich sind Werte von $\alpha = 0,05$ oder $\alpha = 0,01$, wobei die Auswirkungen einer Fehlentscheidung vor jedem Hypothesentest abgewogen werden müssen [CD82].

Eine verbreitete Entscheidungsregel für die Annahme oder Ablehnung von H_0 ist der t-Test. Er wird zudem für kleine Datenmengen empfohlen, wie sie in der vorliegenden Arbeit zu erwarten sind. Dieser Test verwendet eine Prüfgröße, die unter der Annahme, dass die Nullhypothese gilt, t-verteilt¹⁴ ist. Die Prüfgröße t berechnet sich entsprechend Gleichung 5.2, wobei der Ausdruck $(\mu_A - \mu_B)$ aufgrund der Nullhypothese zu null gesetzt wird.

$$t = \frac{(\bar{x}_A - \bar{x}_B) - (\mu_A - \mu_B)}{s_{\bar{x}_1 - \bar{x}_2}} = \frac{(\bar{x}_A - \bar{x}_B)}{s_{\bar{x}_1 - \bar{x}_2}} \quad (5.2)$$

In dieser Gleichung bezeichnet $(\bar{x}_A - \bar{x}_B)$ die empirische Mittelwertdifferenz der Stichproben, $s_{\bar{x}_1 - \bar{x}_2}$ deren geschätzten Standardfehler und $(\mu_A - \mu_B)$ die theoretische Mittelwertdifferenz der Grundgesamtheiten. Sofern $t < t_{krit}$ eingehalten wird, gilt das Ergebnis als nicht signifikant und die Nullhypothese wird akzeptiert. Dennoch ist zu beachten, dass dies kein Beweis für die Korrektheit der Nullhypothese ist. Im Fall $t \geq t_{krit}$ wird von einem signifikanten Ergebnis gesprochen und H_0 zugunsten von H_1 abgelehnt. t_{krit} wird unter Berücksichtigung des Stichprobenumfangs (Freiheitsgrad) und des Signifikanzniveaus aus Tabellen ermittelt. Trotz seiner Verbreitung und Eignung für kleine Datenmengen bzw. Stichproben kann der t-Test für den Selbstlernprozess nicht eingesetzt werden. Der Grund hierfür liegt in der sogenannten α -Fehlerkumulierung, die grundsätzlich alle Arten statistischer Tests betrifft. Das Risiko, einen Fehler 1. Art (s. Tabelle 7) zu begehen, kumuliert sich und steigt mit der Anzahl M der zu vergleichenden Metamodelle:

$$\alpha_{global} = 1 - (1 - \alpha_{einzel})^{\frac{1}{2} \cdot M \cdot (M - 1)} \quad (5.3)$$

Werden vier Metamodelle miteinander verglichen ($M = 4$) so liegt das Gesamtrisiko, eines der Modelle fälschlicherweise als das bestgeeignete zu bewerten (Fehler 1. Art), bereits bei $\alpha_{global} \approx 26,5\%$. Gängige Hilfsmittel sind in diesem Fall die Bonferroni bzw. die Bonfferoni-Holm Korrektur, bei der eine Aufteilung von α_{global} auf die α Werte der einzelnen t-Tests erfolgt [BA95]. Für eine hohe Anzahl an Metamodellen bedeutet dies, dass die einzelnen Werte für α so niedrig werden, dass die entsprechenden Nullhypthesen immer seltener verworfen werden, oder nur bei sehr starken Unterschieden in den Stichproben. Wie bereits erwähnt, soll der Selbstlernprozess eine möglichst hohe Flexibilität aufweisen. Dies bedeutet, dass zum einen eine Vielzahl an Metamodellen für die Analyse der Fertigungsdaten zur Verfügung stehen muss, da es, wie in Kapitel 2 gezeigt, kein „universelles“ Metamodell gibt. Zum anderen soll die Wissensakquisitionskomponente in der Lage sein, in Zukunft weitere Metamodelle aufzunehmen und im Selbstlernprozess berücksichtigen zu können. Ein Vergleich der Metamodellperformanzen auf Basis eines t-Tests ist daher nicht zielführend.

¹⁴ t-Verteilungen sind symmetrische, eingipflige Verteilungen mit einem Erwartungswert $\mu = 0$ und nähern sich mit zunehmendem Freiheitsgrad einer Standardnormalverteilung an [Roo14].

Anstatt eines t-Tests wird für die Selbstlernkomponente von Slassy ein zweistufiges Vorgehen verfolgt. Zunächst werden mit einer *Varianzanalyse* (ANOVA, engl. Analysis of Variances) die Mittelwerte der Performanzschätzer durch einen einzigen Hypothesentest analysiert. Dabei wird festgestellt, ob sich die Metamodelle überhaupt signifikant von einander unterscheiden. Anschließend wird mithilfe eines *post-hoc Tests* das Metamodell identifiziert, dessen Performanz sich signifikant von den anderen unterscheidet. Diese Aussage wäre mit einer Varianzanalyse allein nicht möglich (ungerichteter Hypothesentest).

Die Varianzanalyse bringt den Vorteil mit sich, dass nur einmal auf einem Datensatz getestet wird und somit der α -Fehler nicht kumuliert. Zudem gehen in die ANOVA die Werte aller Datentupel gleichzeitig ein, wodurch die Teststärke¹⁵ der ANOVA sehr viel höher ist als die eines einzelnen t-Tests. Die ANOVA ist ein parametrisches Verfahren zur statistischen Untersuchung von Mittelwertsdifferenzen. Insbesondere bei mehr als zwei zu untersuchendem Stichprobengruppen ist sie dem t-Test überlegen, der vom Problem der α -Fehlerkumulierung betroffen ist. Die untersuchten Gruppen müssen bestimmte Voraussetzungen erfüllen, da es sonst zu großen Abweichungen des realen α -Fehlers gegenüber dem festgelegten Signifikanzniveau kommen kann. So müssen die Stichproben voneinander unabhängig sein und normalverteilt vorliegen. Eine weitere Voraussetzung ist, dass zwischen den Gruppen annähernd Varianzhomogenität herrscht. Im Kontext der vorliegenden Arbeit erweist sich die Eigenschaft der ANOVA, nur ungerichtete Hypothesentests zu erlauben, allerdings als Nachteil. Wird ein signifikantes Ergebnis nach einer Varianzanalyse festgestellt und die Nullhypothese verworfen, ist zunächst nur die Aussage gesichert, dass sich mindestens zwei Mittelwerte statistisch signifikant voneinander unterscheiden. Eine genauere Aussage ist nicht möglich. Darüber hinaus umfasst eine global formulierte Alternativhypothese („Die Mittelwerte unterscheiden sich.“) alle möglichen Kombinationen von Unterschieden (größer oder kleiner) zwischen den betrachteten Metamodellperformanzen. [Ste07; RFHN14]

In einem zweiten Schritt wird daher ein post-hoc Test durchgeführt, der einen paarweisen Vergleich der Performanzkennwerte erlaubt, ohne dass der α -Fehler kumuliert oder die Teststärke abnimmt. Als besonders geeignet bei vielen paarweisen Vergleichen hat sich der *Tukey's Honest Significance Difference Test* (Tukey's HSD) erwiesen. Er ermittelt für jeden paarweisen Vergleich die Mindestdifferenz, die zwischen den jeweiligen Mittelwerten auftreten muss, damit diese Differenz auf dem zuvor definierten Signifikanzniveau signifikant ist. Errechnet wird daher eine kritische Mittelwertdifferenz, deren Überschreitung auf eine signifikante Differenz zwischen den verglichenen Gruppen schließen lässt.

Im Rahmen des Selbstlernprozesses ist zunächst auszuschließen, dass die beobachteten Mittelwertsdifferenzen zwischen den Metamodellperformanzen rein zufällig auftreten. Dies wird durch die ANOVA sichergestellt. Ein post-hoc Test könnte beim paarweisen Vergleich signifikante Unterschiede feststellen, die aber, global bezogen auf alle Mittelwerte, gar nicht signifikant sind. Daher muss dem post-hoc immer eine ANOVA vorausgehen. Zudem benötigt jede Statistik-Software zur Durchführung eines post-hoc Tests Kennwerte aus der

¹⁵ Die Sensitivität des Experimentes, einen vorhandenen Effekt identifizieren zu können. [BS10]

ANOVA, was ein weiterer Grund für die hier festgelegte Reihenfolge ist. Im Folgenden werden beide Schritte, ANOVA und Tukey's HSD, detaillierter beschrieben.

5.4.1 Analyse der Varianzen

Ansatzpunkt für die Durchführung einer Varianzanalyse ist die Zerlegung der Gesamtvarianz in den Prognosefehlern, die im ROPE-Prozess ermittelt wurden, in systematische und unsystematische Anteile. Allgemein kann die Varianz als die mittlere Abweichung jedes einzelnen Werts x_i vom Mittelwert \bar{x} einer Stichprobe mit n Werten verstanden werden:

$$\sigma_x^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1} \quad (5.4)$$

Systematische Einflüsse sind auf real vorhandene Unterschiede in den Prognosefehlern zurückzuführen. Somit wird durch die systematische Varianz σ_{sys}^2 der Unterschied zwischen den Gruppen ausgedrückt. Unsystematische Einflüsse sind meist zufälliger Natur und bei experimentell ermittelten Daten auf Messungenauigkeiten zurückzuführen. Die Varianz, die durch unsystematische Einflüsse verursacht wird, heißt Residualvarianz σ_{res}^2 und entspricht der mittleren Abweichung der Einzelfehler e_i eines Performanzvektors von dessen Mittelwert \bar{E} . Sie wird daher auch als „innere“ Varianz bezeichnet. Unter idealen Bedingungen sollte Varianzhomogenität herrschen, die Residualvarianz der Abweichungen in den einzelnen Vektoren quasi gleich sein. Jedoch ist dies in der Realität oftmals nicht gegeben. Daher werden die einzelnen geschätzten Residualvarianzen der Gruppen addiert und durch die Gruppenanzahl geteilt, um einen Mittelwert zu bilden. Da durch den ROPE-Prozess in jeder Gruppe bzw. jedem Performanzvektor gleich viele Einzelwerte enthalten sind, gilt für die Residualvarianz:

$$\sigma_{res}^2 = \frac{\sum_{i=1}^M \sum_{j=1}^z (e_{ji} - \bar{E}_i)^2}{M \cdot (z - 1)} \quad (5.5)$$

Zur Berechnung der systematischen Varianz wird die Annahme getroffen, dass die Differenzen der Performanzen auf real existierende Unterschiede der Prognosequalitäten zurück gehen. Unter dieser Annahme würde jeder Einzelfehler der Performanzvektoren dem jeweiligen Mittelwert entsprechen. Da allerdings auf die Einzelwerte bereits die unsystematischen Einflüsse wirken, sind auch die Gruppenmittelwerte entsprechend verfälscht. Die systematische Varianz kann daher nicht isoliert berechnet werden und entspricht auch nicht dem Erwartungswert der Varianz zwischen den Gruppen. Der Erwartungswert der Varianz zwischen gleich großen Gruppen setzt sich zusammen aus der z -gewichteten, systematischen Varianz und der Residualvarianz:

$$E(\sigma_{zwischen}^2) = E(\sigma_{res}^2) = z \cdot \sigma_{sys}^2 + \sigma_{res}^2 \quad (5.6)$$

Die Berechnung der Varianz zwischen den Gruppen, oder auch Treatmentvarianz, erfolgt durch Einsetzen der einzelnen Mittelwerte \bar{E}_i und des Gesamtmittelwerts \bar{G} aller Perfor-

manzvektoren in die Varianzformel 5.4. Die Freiheitsgrade im Nenner ergeben sich aus der Anzahl der verglichenen Performanzvektoren. Die Quadratsumme der Mittelwertabweichungen vom Gesamtmittelwert muss zusätzlich mit der Menge der Einzelwerte in einer Gruppe multipliziert werden, damit die Genauigkeit der Mittelwerte als Populationsschätzer berücksichtigt wird:

$$\sigma_{treat}^2 = \frac{z \cdot \sum_{i=1}^M (\bar{E}_i - \bar{G})^2}{M - 1} \quad (5.7)$$

Die Varianzen zwischen den Abweichungen in den Performanzvektoren werden nun ins Verhältnis zu den jeweiligen inneren Varianzen gesetzt. Beide Werte schätzen auf unterschiedlichem Weg die gleiche Residualvarianz. Im Fall der einfaktoriellen ANOVA ist im Erwartungswert der Varianz zwischen den Abweichungen der Performanzvektoren zusätzlich die systematische Varianz enthalten (s. Gleichung 5.6) vorausgesetzt, es existiert ein systematischer (signifikanter) Einfluss auf die geschätzten Performanzen. Der Kennwert, der sich durch den Vergleich ergibt, wird als *F*-Wert bezeichnet:

$$F_{df_{treat}; df_{res}} = \frac{\sigma_{treat}^2}{\sigma_{res}^2} \quad \text{mit } df_{treat} = M - 1; df_{res} = M \cdot (z - 1) \quad (5.8)$$

Für den *F*-Wert können sich zwei mögliche Werte ergeben. Sofern kein systematischer Einfluss in den Daten und damit auch kein signifikant besseres Metamodell existiert, gehen sowohl in den Zähler als auch in den Nenner nur die „inneren“ Residualvarianzen ein und es ergibt sich $F = 1$. Sind die Mittelwerte der Abweichungen in den Performanzvektoren nicht nur wegen unsystematischer, statistischer Einflüsse verschieden, sondern aufgrund eines echten Unterschieds zwischen mindestens zwei Modellen, so ist der Zähler stets größer als der Nenner und damit $F > 1$. Wie auch beim t-Test muss bei der ANOVA eine Signifikanzprüfung des *F*-Werts erfolgen, um die Nullhypothese der ANOVA ($H_0 : \mu_1 = \mu_2 = \dots = \mu_M$) zugunsten der Alternativhypothese ($H_1 : \exists i, j : \mu_i \neq \mu_j$) verworfen zu können. Dies kann erfolgen wenn der berechnete *F*-Wert größer als der kritische Wert F_{krit} ist. In die tabellarische Ermittlung von F_{krit} gehen neben dem Signifikanzniveau (i. A. $\alpha = 0,05$) auch die Freiheitsgrade für $\sigma_{zwischen}^2$ und σ_{Res}^2 ein. Beim Einsatz von Statistiksoftware gibt es die Möglichkeit den als „beobachtetes Signifikanzniveau“ bezeichneten *p*-Wert zu berechnen. Dieser entspricht dem kleinsten Wert von α , für den das Testergebnis gerade noch Signifikanz erreicht bzw. der Wahrscheinlichkeit des errechneten *F*-Wertes unter Annahme der Nullhypothese. Gilt $p < \alpha$ wird H_0 abgelehnt.

Im Fall, dass zwischen den verglichenen Performanzvektoren kein signifikanter Unterschied existiert und H_0 nicht verworfen werden kann, wird eines der Modelle anhand bestimmter Kriterien ausgewählt, die in Abschnitt 6.3.5 (S. 109) erläutert werden. Ein signifikantes Ergebnis führt direkt zum nächsten Schritt, dem post-hoc Test mittels Tukey's HSD.

5.4.2 Durchführung des post-hoc Tests

Beim post-hoc Test nach Tukey erfolgt ein paarweiser Vergleich der Vorhersagefehler aller Metamodelle. Die dabei ermittelte Differenz der Performanzkennwerte muss einen zuvor berechneten Mindestwert (Tukey's Honest Significant Difference, HSD) überschreiten, damit sie als statistisch signifikant gilt. Diese Differenz ist dann auf einem zuvor definierten α -Niveau signifikant und das Risiko die Nullhypothese irrtümlicherweise zu verwerfen (Fehler 1. Art) stets geringer als dieses Niveau. Der post-hoc Test nach Tukey errechnet also eine kritische Mittelwertdifferenz, deren Überschreitung einer signifikanten Mittelwertdifferenz zwischen den zwei betrachteten Metamodellen gleichkommt.

Tukey's HSD wird über den Kennwert q ermittelt, der dem t -Wert beim t-Test entspricht und ähnlich definiert ist:

$$q_{(M; df_{innerhalb})} = \frac{\bar{x}_{M_i} - \bar{x}_{M_j}}{\sqrt{\frac{\sigma_{res}^2}{z}}}; \quad \exists i, j \text{ mit } i \neq j \quad (5.9)$$

Dem q -Wert liegt eine eigene Verteilung zugrunde, die sog. „Studentized-Range“-Verteilung, da er sich auf mehrfache Mittelwertsvergleiche bezieht. Im Gegensatz zur t -Verteilung ist es hier möglich, einen kritischen Wert q_{krit} in Abhängigkeit von der Anzahl der betrachteten Performanzvektoren zu bestimmen. Dies ist der eigentliche Grund, weshalb es beim Einsatz von Tukey's HSD zu keiner Kummulierung des α -Fehlers kommt. Die Residualvarianz wird aus der zuvor durchgeführten ANOVA übernommen. Nach der Bestimmung des kritischen Werts q_{krit} und Umordnung der Gleichung 5.9, kann für jeden paarweisen Vergleich direkt Tukey's HSD berechnet werden:

$$\bar{x}_{M_i} - \bar{x}_{M_j} = HSD = q_{krit(\alpha; M; df_{innerhalb})} \cdot \sqrt{\frac{\sigma_{res}^2}{z}} \quad (5.10)$$

Diese kritische Differenz wird mit der tatsächlichen Differenz zwischen den Performanzvektoren verglichen. Wird festgestellt, dass die tatsächliche Differenz zwischen zwei Metamodellperformanzen größer als Tukey's HSD ist, so besteht ein statistisch signifikanter Unterschied zwischen diesen beiden Modellen und der kleinere Vorhersagefehler des „besseren“ Modells beruht nicht auf rein-zufälligen Effekten.

6 Das selbstlernende Assistenzsystem Slassy

Im folgenden Kapitel wird das selbstlernende **Assistenzsystem**, kurz Slassy, mit allen Komponenten und deren Zusammenwirken dargestellt. Neben der Gesamtstruktur wird zu Beginn die Integration von Slassy in das CAD-Umfeld als Arbeitsumgebung des Konstrukteurs vorgestellt. Anschließend liegt das Hauptaugenmerk auf der Umsetzung des in Kapitel 5 entwickelten Selbstlernprozesses in Form der Wissensakquisitions- bzw. Selbstlernkomponente. Die Beschreibungen der Synthese- und Analysewerkzeuge sowie des Produktdatenmodells und dessen Anbindung an Slassy bilden den Abschluss des Kapitels.

6.1 Integration in das Arbeitsumfeld der Benutzer

Wie in Abschnitt 1.2 dargestellt, gehören zur Zielgruppe des selbstlernenden Assistenzsystems Betriebsmittelkonstrukteure bzw. Konstrukteure von Bauteilen mit komplexen Nebenformelementen (s. Bild 31, S. 56), die mittels Blechmassivumformung gefertigt werden sollen. Während der Entwicklung dieser Bauteile sind unterschiedliche Design for X Aspekte zu berücksichtigen. Unter anderem ist die fertigungsgerechte Gestaltung (s. Abschnitt 2.3.1) nach einer Geometrieveränderungen zu überprüfen, was unter den herrschenden Randbedingungen in der Entwicklung (Zeit- und Kostendruck) zügig, jedoch gleichzeitig auch verlässlich erfolgen muss.

Wie gehen die Benutzer bei der Design for Manufacture Überprüfung vor und wie muss Slassy aufgebaut und integriert werden, damit eine wissensbasierte Unterstützung sichergestellt ist? Bei der Beantwortung dieser Frage hilft ein Blick auf ein Detail des in Abschnitt 2.1.1 dargestellten Konstruktionsprozesses. Kern des Konstruierens ist ein iteratives Vorgehen mit ständigem Wechsel zwischen Produktsynthese und -analyse [FG13]. Nach [Web12] beinhaltet die Synthese, ausgehend von vorgegebenen geforderten Eigenschaften, die Bestimmung der Merkmale des Bauteils. Hierbei umfasst das „Bestimmen“ zum einen die Definition relevanter Merkmale (qualitatives Gestalten der Baustruktur, Geometrie- und Materialfestlegung) und zum anderen die Zuweisung konkreter Werte für bereits definierte Merkmale (Dimensionierung). In der sich anschließenden Analyse werden für das „synthetisierte“ Bauteil die Eigenschaften und das sich daraus ergebende Verhalten ermittelt bzw. vorhergesagt, sollte das Bauteil physisch (z. B. als Prototyp), noch nicht existieren [Web05]. Wird das geforderte Verhalten nicht erfüllt, so erfolgt in einem erneuten Syntheseschritt

eine Anpassung der Merkmale, bis schließlich die an das Bauteil gestellten Anforderungen erfüllt sind.

Übertragen auf den Kontext dieser Arbeit umfasst die Synthese blechmassivumgeformter Bauteile zunächst die Auswahl eines Hauptformelements und eines oder mehr Nebenformelemente (s. Abschnitt 3.3). Das Hauptformelement und insbesondere das Nebenformelement als Träger der Hauptfunktion werden anschließend feingestaltet. Dabei wählt der Konstrukteur die Merkmale des Bauteils so, dass zum einen die gewünschten Funktionen erfüllt und zum anderen die Grenzen der relevanten Prozessparameter Blechmassivumformung eingehalten werden (s. Abschnitt 3.2). Dies ist in Bild 47 gezeigt. Die Einbindung von neuen Haupt- oder Nebenformelementen nutzt Funktionen des CAD-Systems, ist dadurch besonders benutzerfreundlich und unterstützt die Standardisierung von Konstruktionsprozessen.

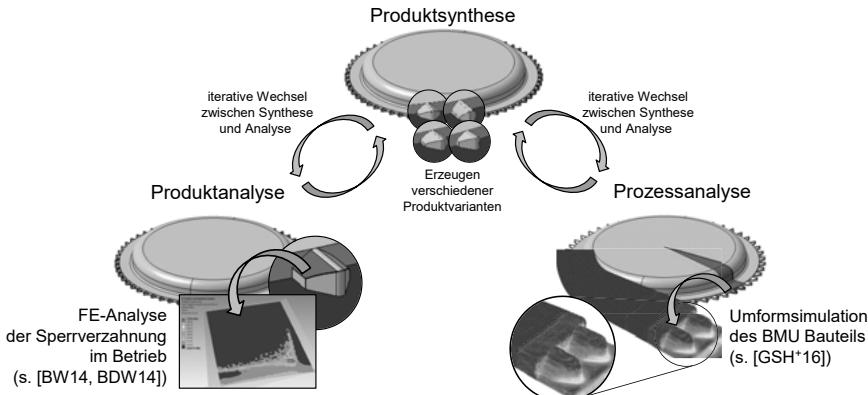


Bild 47: Wechselspiel zwischen Produktsynthese und -analyse sowie Umformprozessanalyse.

Die Analyse der Funktionserfüllung wird in den meisten Fällen vom Konstrukteur selbst durchgeführt, beispielsweise ein Tragfähigkeitsnachweis einer Verzahnung nach DIN 3990 oder eine FEM-basierte Simulation von verschiedenen Betriebszuständen, wie bspw. in [BDW14; BW14]. Die DfM-Analyse von blechmassivumgeformten Bauteilen hingegen kann, wie in Kapitel 4 dargestellt, auf herkömmliche Weise (s. Abschnitt 2.3, S. 34) nicht mehr vom Konstrukteur geleistet werden, denn Spezialsoftware zur simulativen Umformprozessabsicherung beherrscht im Regelfall nur der Fertigungsexperte und Konstruktionsrichtlinien stehen für die Blechmassivumformung (noch) nicht zur Verfügung. Sowohl die Synthese als auch die wissensbasierte Analyse von blechmassivumgeformten Bauteilen werden von Slassy direkt im CAD-System umgesetzt.

Damit Slassy den Konstrukteur bei der Design for Manufacture Analyse wissensbasiert unterstützen kann, muss das fertigungsrelevante Konstruktionswissen, bevor es ihm zur Verfügung gestellt wird, erhoben werden. Das geschieht durch den in Kapitel 5 entwickel-

ten Selbstlernprozess, der in der sog. Selbstlernkomponente umgesetzt wird. Der Selbstlernprozess basiert auf Methoden der Wissensentdeckung in Datenbanken (KDD, engl. Knowledge Discovery in Databases). Für die Benutzerakzeptanz wird der KDD-basierte Selbstlernprozess als besonders kritisch erachtet, denn dadurch kommt ein Aspekt hinzu, der untypisch ist für Konstruktions- oder Produktentwicklungsabteilungen. Muss der Konstrukteur zur Umsetzung des Selbstlernprozesses ein spezielles KDD-Tool zusätzlich zu den bereits eingesetzten Konstruktionswerkzeugen (CAD, PLM, Tabellenkalkulation etc.) beherrschen, wird dies nicht zu einem nachhaltigen Einsatz des Assistenzsystems beitragen. Ein Wissensaustausch zwischen Konstruktion und Fertigung würde nicht stattfinden, eine Alterung von Wissen wäre die Folge.

Um dies zu vermeiden, übernimmt Slassy, wie in Bild 48 dargestellt, für den Anwender den Großteil der automatischen Wissensakquisition, die an den KDD-Prozess nach [FPSS96] angelehnt ist. Dieser besteht aus neun Phasen, die im Abschnitt 2.4.1 (S. 42) detailliert beschrieben sind.

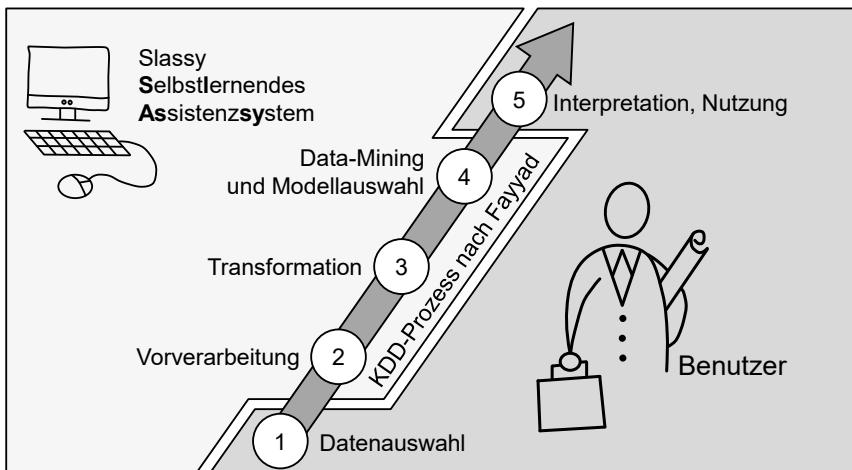


Bild 48: Aufteilung der KDD-Prozessschritte nach [FPSS96] zwischen Benutzer und Slassy.

Der erste Schritt erfordert Hintergrundwissen sowohl über den Fertigungsprozess als auch über das Produkt und dessen Funktionen. Konstrukteur und Fertigungsingenieur müssen gemeinsam Attribute und Label (s. Abschnitt 2.4.2, S. 43) definieren und eine Strategie für die Datenerhebung entwickeln. Eine rechnerunterstützte Umsetzung dieses Schritts im Rahmen dieser Arbeit wird nicht verfolgt, da er kognitive Fähigkeiten erfordert, die nicht durch Maschinen geleistet werden können. Zum anderen existiert für die Planung von Versuchen bereits eine Vielzahl probater Werkzeuge (z. B. Minitab, Cornerstone, R, siehe [Mat05; SvH10]). Das Zusammenführen der Daten und deren Vorverarbeitung ist ein Schritt, der teilweise außerhalb von Slassy durchgeführt werden kann. Dazu gehört

bspw. die Erstellung einer Tabelle mit den Versuchs- oder Simulationsdaten in Excel®. Die Zuordnung der Daten zu einer bestimmten Produktinstanz (z. B. „tiefegzogener Napf mit offenem Mitnehmer“) wird durch das in Slassy implementierte Produktmodell (s. Abschnitt 6.6) sichergestellt, ebenso wie die Übergabe an das KDD-Tool für den folgenden Schritt Datenreduktion und -abbildung. Die erste, im Rahmen dieser Arbeit entwickelte Version von Slassy ist mit einer Grundmenge an Data-Mining Algorithmen ausgestattet (s. Abschnitt 6.3.1), mit der der Anwender bereits effektiv arbeiten kann. Eine Erweiterung um zusätzliche Algorithmen kann jederzeit und mit geringem Aufwand durchgeführt werden. Die Kernelemente des Selbstlernprozesses, der ROPE-Prozess (s. Abschnitte 5.2 und 5.3) und die 2-stufige Modellauswahl (s. Abschnitt 5.4), sind den Schritten sieben und acht zugeordnet. Für den letzten Schritt lässt sich erneut eine Aufgabenteilung zwischen Assistenzsystem und Nutzer feststellen. Während Slassy das akquirierte Wissen in der Wissensbasis abspeichert und damit zugänglich macht, ist der Konstrukteur nun in der Lage dieses Wissen für seine DfM-Analysen zu nutzen.

6.2 Die Komponenten und ihr Zusammenwirken

Die dargestellte Integration von Slassy in das Arbeitsumfeld des Konstrukteurs wird durch die einzelnen Komponenten des Assistenzsystems und deren Interaktion während der Bedienung sichergestellt. Die Struktur der Komponenten folgt dem Grundgedanken des Konstruktionssystems *mfk* (s. Abschnitt 2.2.3, S. 27), mit einem Synthese- und einem Analyseteil, die über ein Produktmodell miteinander gekoppelt sind. Der wesentliche Unterschied zum KSmfk besteht in der Ausführung des Wissensakquisitionstools in Form der Selbstlernkomponente sowie in der Nutzung eines kommerziellen CAD-Systems (CATIA V5) zur Geometrieerzeugung und -verwaltung. Letzteres hat sich bereits in den Arbeiten von [War01] bewährt. Bild 49 fasst die einzelnen Komponenten von Slassy zusammen.

Im Syntheseteil des Konstruktionssystems ermöglicht der Formelemente- und Merkmaleditor das Arbeiten mit den Haupt- und Nebenformelementen. Dem Konstrukteur steht ein Vorrat an Konstruktionselementen (Verzahnungen, Mitnehmer etc.) und Anordnungsmöglichkeiten (Rotationssymmetrie, axiale oder radiale Positionierung, freie Anordnung etc.) zur Verfügung, der an die Erfordernisse bestimmter Bauteilgruppen angepasst ist. Da ein kommerzielles CAD-System zur Visualisierung der Bauteile zum Einsatz kommt, wird dessen Funktion zur Festlegung von benutzerdefinierten Features (UDF, engl. user defined feature) gleichzeitig zur Definition neuer Haupt- oder Nebenformelemente genutzt. Bei jedem Start des Assistenzsystems wird die Existenz neu definierter Formelemente geprüft und entsprechende Funde in die Benutzeroberfläche des Synthesewerkzeugs integriert. Die Verwendung eines kommerziellen CAD-Systems zur Produktvisualisierung und Modellspeicherung bietet außerdem den Vorteil, dass die digitale Prozesskette der Produktentstehung unterbrechungsfrei bleibt. Das CAD-Modell eines feingestalteten Bauteils kann somit direkt an nachfolgende Stakeholder (z. B. Arbeitsvorbereitung, Qualitätsmanagement, Prototypenbau) übergeben werden.

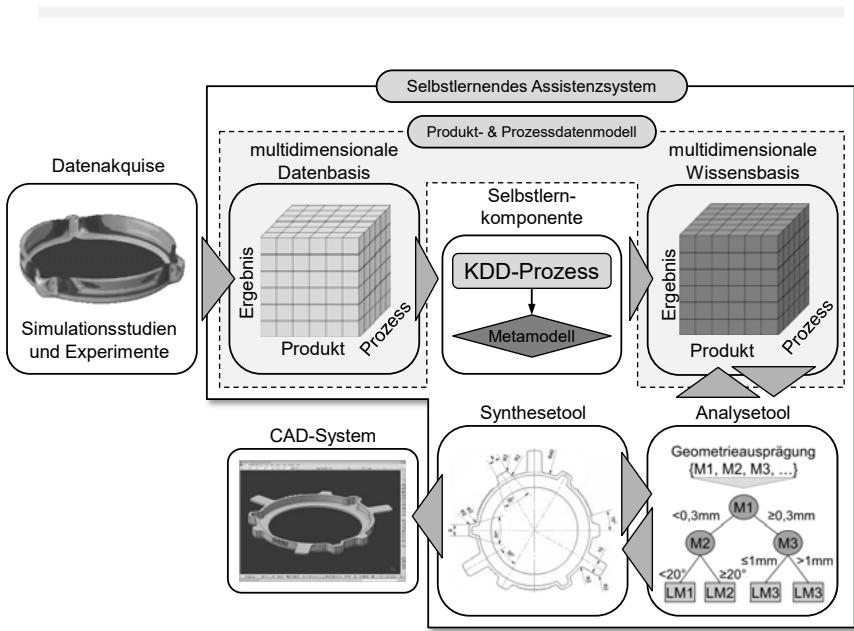


Bild 49: Übersicht der Komponenten von Slassy und ihrer Interaktionen.

Für die wissensbasierte Design for Manufacture Analyse der Bauteilentwürfe wechselt der Benutzer in den Analyseteil des Assistenzsystems. Sofern für eine Bauteilvariante bereits relevante Zielgrößen definiert wurden, Daten vorhanden sind und der Selbstlernprozess abgeschlossen ist, greift das Analysetool auf das CAD-Modell und die Wissensbasis im PPDM zu. Das CAD-Modell enthält die aktuelle Bauteilkonfiguration (Merkmalsausprägungen), die an die Metamodelle in der Wissensbasis zur Berechnung der Zielgrößen übergeben wird. Der Inferenzmechanismus von Slassy wurde im Rahmen dieser Arbeit entwickelt und nutzt die Methode der regulären Ausdrücke (s. [GL10; LR14]), um die von der Wissensbasis an das Analysetool übergebenen Metamodelle korrekt auszuwerten. In das Analysetool ist ferner eine Erklärungskomponente integriert, die dem Benutzer für jede Zielgröße das jeweils gültige Metamodell darstellt. So kann die Berechnung des Analysetools ggf. händisch überprüft werden.

Der Selbstlernkomponente nimmt eine zentrale Rolle in der Gesamtstruktur ein. Bei bisherigen Entwicklungen zum KSmfk wurde das Konstruktionswissen über graphische Editoren und Benutzeroberflächen ([Bac97]) oder direkt bei der Implementierung des Systems im Anschluss an Experteninterviews und Literaturrecherchen eingepflegt ([Kra92; War01]). Slassy hingegen akquiriert das konstruktionsrelevante Wissen selbstständig aus Daten der Fertigungsprozessentwicklung, die zuvor vom Anwender über eine flexible Benutzerschnitt-

stelle importiert wurden. Bereits vorhandene Daten können aufgerufen, angepasst und für einen weiteren Selbstlernprozess genutzt werden. Damit die Daten den späteren Metamodellen korrekt zugeordnet werden, führt die Selbstlernkomponente den Benutzer durch einen Workflow für die Definition der Attribute und Label. Zur Sicherstellung der Datenkonsistenz werden die vom Benutzer definierten Attribute mit den im UDF hinterlegten Parametern abgeglichen.

Daten, die im Selbstlernprozess verarbeitet werden sind im Produkt- und Prozessdatenmodell (PPDM) des Assistenzsystems gespeichert. Das Produkt- und Prozessdatenmodell erfüllt dabei die Funktion der hierarchischen Gliederung aller Fertigungsdaten über mehrere Ebenen vom Allgemeinen ins Detail. So kann beispielsweise ein blechmassivumgeformtes Bauteil aus mehreren Formelementen bestehen und jedes Formelement ist seinerseits durch verschiedene Parameter (Merkmale) beschreibbar. Es werden nur die geometriebeschreibenden Parameter verwaltet und nicht etwa die CAD-Geometrie mittels bestehender Modellierungstechniken (z. B. CSG, B-Rep oder Hybridmodelle, s. [Vaj09]) repräsentiert. Hierzu wird auf das CAD-System zurückgegriffen. Neben den Geometrieparametern werden auch nichtgeometrische Informationen wie die Ergebnisse und Randbedingungen von Umformsimulationen (z. B. Reibverhältnisse, Umformgeschwindigkeit, Werkzeugkonzept) oder Umformexperimenten (z. B. Schmierstoffe, Temperatur) im Produktmodell gespeichert. Nach erfolgreich durchlaufenem Selbstlernprozess wird der analysierten Bauteilvariante (z. B. „Ronde mit Außenverzahnung“) im Produkt- und Prozessdatenmodell das akquirierte Wissen zugewiesen. Die multidimensionale Wissensbasis (s. Abschnitt 6.6.1) ist somit in das Produkt- und Prozessdatenmodell integriert. Dies erleichtert den Zugriff auf die Daten- und Informationsobjekte, da nur die Methode der SQL-Datenbankanfrage genutzt wird.

6.3 Die Selbstlernkomponente

Die Selbstlernkomponente ist in der Lage, aus den Daten der Fertigungsprozessentwicklung Wissen, repräsentiert in Form sogenannter Metamodelle, automatisch zu akquirieren. Die Bezeichnung „selbstlernend“ bezieht sich auf die Fähigkeit, durch geeignete statistische Verfahren in einer Grundmenge an Metamodellen das jeweils bestgeeignete Modell robust und ohne Eingriff des Benutzers zu identifizieren. Der schematische Ablauf des in Kapitel 5 entwickelten Selbstlernprozesses ist in Bild 50 dargestellt.

Die detaillierte Erläuterung der genannten Schritte wird auf die Kapitel 6 und 7 aufgeteilt, da einzelne Aspekte im Rahmen des Anwendungsbeispiel anschaulicher dargestellt werden können. Der Fokus der folgenden Abschnitte liegt auf den eingesetzten Metamodellen sowie der Umsetzung des Selbstlernprozesses im KDD-Werkzeug Rapidminer® sowie in Matlab®.

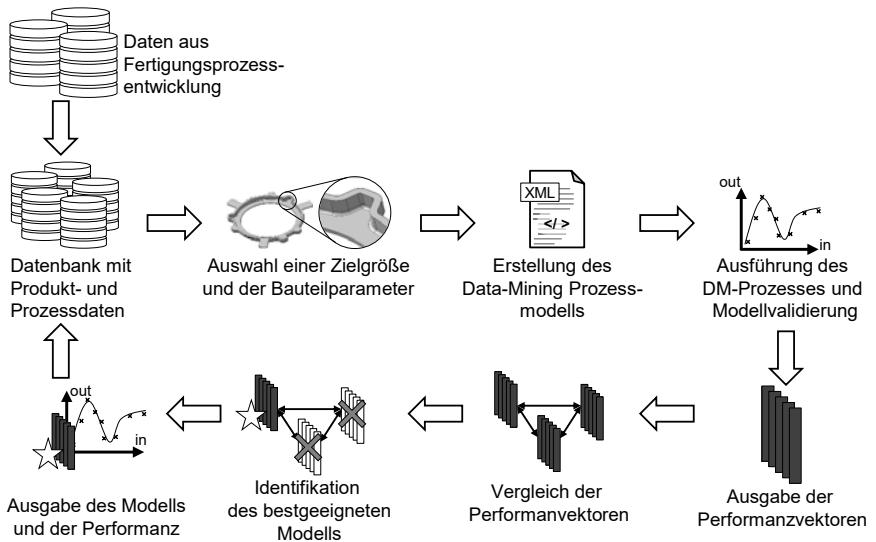


Bild 50: Ablauf des Selbstlernprozesses im Assistenzsystem.

6.3.1 Auswahl der eingesetzten Metamodelle

In der Literatur findet sich eine breites Spektrum an Algorithmen, die für die Erstellung eines Metamodells genutzt werden können. Es gilt jedoch zu berücksichtigen, dass der Benutzer in der Lage sein muss, das akquirierte Wissen, repräsentiert durch die Metamodelle, über die Erklärungskomponente interpretieren zu können (s. Abschnitt 4.2). Hinsichtlich dieser Interpretation von automatisch erhobenem Wissen werden zwei Arten der Modellrepräsentation unterschieden. Neben einfach abbildungsbaren Zusammenhängen (z. B. Constraints, Regeln), existieren Modelle, deren „innerer“ Aufbau zu komplex für eine nachvollziehbare Darstellung ist. Diese werden im Allgemeinen als „Black-Box“ Modelle bezeichnet [Ert13; HKP12]. Hintergrund ist, dass der Benutzer die Attribute vorgibt und das Black-Box Metamodell die entsprechende Vorhersage für das Label trifft. Dabei kann bspw. nicht festgestellt werden, ob und mit welcher Gewichtung einzelne Attribute in die Berechnung eingehen. Jede Vorhersage des Labels bleibt intransparent. Beispiele für derartige Modelle sind künstliche neuronale Netze (s. [Ert13]), Support Vector Machine (SVM, s. [SC08]) oder Kriging (s. [OW15]). Bild 51 stellt beide Modellarten gegenüber.

Rude [Rud98] stellt fest, dass von den verfügbaren Wissensrepräsentationsarten Constraints (ugs. Formeln) und Regeln für Menschen besonders leicht zu formulieren und dementsprechend gut zu interpretieren sind. In die Selbstlernkomponente werden daher Metamodellalgorithmen integriert, die diesen Repräsentationsarten zugeordnet werden können.

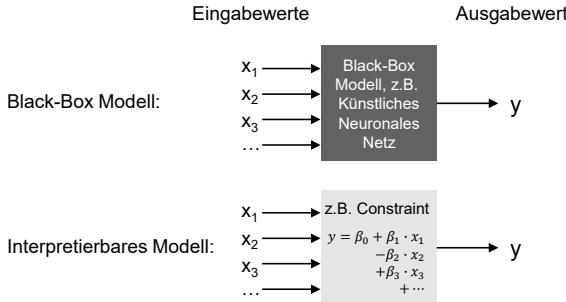


Bild 51: Gegenüberstellung eines Black-Box und eines durch den Menschen interpretierbaren Modells.

Lineare Regression

Lineare Regressionsmodelle werden durch die Bestimmung von $(A + 1)$ unbekannten Regressionskoeffizienten α_i für die lineare Funktion in Gleichung 6.1 trainiert. Jedem Attribut ist ein Regressionskoeffizient zugeordnet. Ziel ist die Minimierung des Fehlers e (Residuum) zwischen der wahren Größe y des Labels in den Trainingsdaten (mit $N > A$ Datentupeln) und der Vorhersage \hat{y} [JS11].

$$\hat{y} = \alpha_0 + \sum_{i=1}^A \alpha_i x_i \quad (6.1)$$

Die Berechnung der Koeffizienten α_i erfolgt nach der Methode der kleinsten Quadrate in Gleichung 6.2. Aus der Forderung, dass die partiellen Ableitungen der Gleichung 6.2 Null ergeben müssen $\partial g / \partial \alpha_j = 0$, kann das lineare Gleichungssystem 6.3 mit N Gleichungen für $(k+1)$ unbekannte Koeffizienten erzeugt werden.

$$\min \sum_{i=1}^N (y_i - \hat{y}_i)^2 = \min \sum_1^N \left(y_i - \alpha_0 - \sum_{j=1}^A \alpha_j x_{ij} \right)^2 = \min g(\alpha_0, \dots, \alpha_k) \quad (6.2)$$

Dieses Gleichungssystem kann wie in (6.4) gezeigt in Matrixschreibweise anschaulich dargestellt werden.

$$\mathbf{y} = \mathbf{X}\boldsymbol{\alpha} + \mathbf{e} \quad (6.3)$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1A} \\ 1 & x_{21} & x_{22} & \dots & x_{2A} \\ 1 & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{N1} & x_{N2} & \dots & x_{NA} \end{bmatrix}; \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}; \quad \mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \quad (6.4)$$

Zur Lösung des Problems wird der Vektor α mit den Schätzern für die kleinsten Quadrate gesucht, der die Minimalbedingung (6.2) erfüllt. Mit der Annahme $E = \sum(y_i - \hat{y}_i)^2$ gilt dann (6.5), mit der zusätzlichen Bedingung aus (6.6).

$$E = \mathbf{e}^T \mathbf{e} = (\mathbf{y} - \mathbf{X}\alpha)^T (\mathbf{y} - \mathbf{X}\alpha) = \mathbf{y}^T \mathbf{y} - 2\alpha^T \mathbf{X}^T \mathbf{y} + \alpha^T \mathbf{X}^T \mathbf{X} \alpha \quad (6.5)$$

$$\frac{\partial E}{\partial \alpha} = -2\mathbf{X}^T \mathbf{y} + 2\mathbf{X}^T \mathbf{X} \hat{\beta} \stackrel{!}{=} 0 \quad (6.6)$$

Unter der Bedingung, dass \mathbf{X} eine reguläre Matrix ist, kann (6.6) nach dem unbekannten Vektor mit den Regressionskoeffizienten α aufgelöst werden:

$$\hat{\alpha} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (6.7)$$

Dies führt zur linearen Regressionsgleichung für die Vorhersage des Labels:

$$\hat{y} = \hat{f}(x) = \mathbf{x}^T \hat{\alpha} = \hat{\alpha}_0 + \sum_{i=1}^A \hat{\alpha}_i x_i \quad (6.8)$$

Polynomiale Regression

Häufig beschreiben erhobene Daten nichtlineare Zusammenhänge zwischen den Attributen und dem Label. Da der Anwender des selbstlernenden Assistenzsystems nicht mit der aufwendigen Datenanalyse betraut werden soll, muss gewährleistet sein, dass nichtlineares Prozessverhalten durch Slassy korrekt identifiziert und bei der automatischen Wissensakquisition berücksichtigt wird. Aus diesem Grund wird neben der linearen eine polynomiale Regression in die Selbstlernkomponente integriert. Die Regressionskoeffizienten α_i entsprechen denen des linearen Regressionsmodells in Abschnitt 6.3.1. Hinzu kommen die Koeffizienten α_{ii} für die Quadrate aller Attribute sowie die Koeffizienten α_{ij} für die sogenannten Interaktionsterme. Mit diesen können verstärkende Wechselwirkungen zwischen Attributen beschrieben werden.

$$\hat{y} = \alpha_0 + \underbrace{\sum_{i=1}^A \alpha_i x_i}_{\text{linearer Term}} + \underbrace{\sum_{i=1}^A \alpha_{ii} x_i^2}_{\text{quadr. Term}} + \underbrace{\sum_{i=1}^A \sum_{j=i+1}^A \alpha_{ij} x_i x_j}_{\text{Interaktionsterm}} + e \quad (6.9)$$

Die polynomiale Regression aus (6.9) lässt sich ebenfalls in Matrixschreibweise ausdrücken (6.10). Die Einträge der Koeffizientenmatrix $\hat{\mathbf{B}}$ ergeben sich entsprechend des Vorgehens bei der linearen Regression aus der Methode der kleinsten Quadrate in (6.2) und entsprechend der Bedingung in (6.6).

$$\hat{y} = \alpha_0 + \mathbf{x}' \alpha + \mathbf{x}' \mathbf{B} \mathbf{x} \quad \text{mit}$$

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{pmatrix}; \quad \mathbf{B} = \begin{bmatrix} \alpha_{11} & \frac{1}{2}\alpha_{12} & \cdots & \frac{1}{2}\alpha_{1A} \\ \alpha_{21} & \alpha_{22} & \cdots & \frac{1}{2}\alpha_{2A} \\ \ddots & \ddots & \ddots & \vdots \\ & & & \frac{1}{2}\alpha_{NA} \end{bmatrix} \text{ sym.}; \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} \quad (6.10)$$

Ein höherer Polynomgrad als zwei (quadratisches Polynom) ist im Allgemeinen nicht zweckmäßiger, da solche extrem nichtlinearen Zusammenhänge in technischen Systemen äußerst selten auftreten. Zudem würden hochgradige Polynome nicht die Forderung erfüllen, ohne Hilfsmittel (z. B. Visualisierungswerkzeug) durch den Benutzer interpretierbar zu sein. Die polynomiale Regression zeichnet sich dadurch aus, dass die Trainingsdaten an den Stützstellen nicht exakt abgebildet, sondern bestmöglich angenähert (approximiert) werden. Dies verhindert zum einen die Überanpassung des Metamodells (s. Abschnitt 2.4.2) an die Trainingsdaten. Zum anderen wird durch die Approximation der Einfluss von Ungenauigkeiten in den Daten abgeschwächt. Diese sind bei experimentellen Daten unvermeidbar (stochastische Abweichungen) und auch Simulationsdaten sind aufgrund von Modellvereinfachungen mit numerischen Fehlern behaftet.

WEKA M5P Regressionsbaum

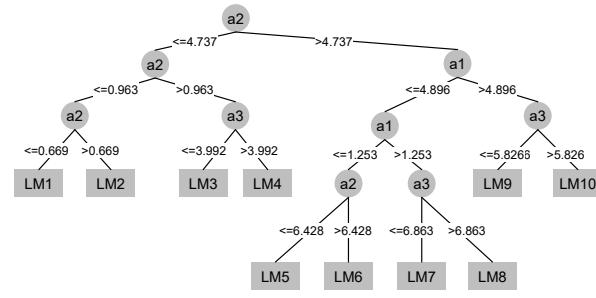
Der Nachteil von polynomialen Regressionsmodellen, mit steigendem Polynomgrad schlechter interpretierbar zu sein, kann durch sogenannte lokale lineare Modelle ausgeglichen werden. Für das Training solcher Modelle steht im Rapidminer® die integrierte Bibliothek WEKA (Waikato Environment for Knowledge Analysis) zur Verfügung. Die bekanntesten Vertreter von lokalen Metamodellen sind Regressionsbäume und Modellbäume. Beide basieren auf Entscheidungsbaumalgorithmen, die ursprünglich für Klassifikationsaufgaben entwickelt wurden (z. B. C4.5, CART).

Das Training eines Regressionsbaums gleicht zu Beginn dem eines Entscheidungsbaums. Nach dem „teile-und-herrsche“ Prinzip wird der Trainingsdatensatz anhand von Werten der Attribute in Teildatensätze unterteilt (engl. *split*) (s. Bild 52). Welches Attribut dabei zur Definition eines Splits gewählt und welcher Attributwert zur Unterteilung des Datensatzes festgelegt wird, hängt vom verwendeten Algorithmus ab, ebenso wie das abschließende *pruning*, das „Stutzen“ des Baums, das Überanpassung verhindert. Auf die Algorithmen soll hier nicht eingegangen werden, weiterführende Informationen sind [WEH11; Qui92; Qui93] und [BFSO84] zu entnehmen. Den Blättern eines erstellten Baums können anschließend Teilmengen an Datentupeln zugeordnet werden, die zusammengefasst den Originaldatensatz ergeben. Für einen Modellbaum werden die einzelnen Datentupel in den Blättern genutzt, um ein lineares Modell zu trainieren, das nur für den jeweiligen Zweig und die entsprechenden Werte der Attribute Gültigkeit besitzt. Diese linearen Modelle sind in Bild 52 mit „LM num: 1“, „LM num: 2“ usw. gekennzeichnet. Zur Erstellung eines Regressionsbaums wird im jeweiligen Blatt der Mittelwert des Labels als einfacher Schätzer hinterlegt. Um Unstetigkeiten zwischen benachbarten Metamodellen zu vermeiden, wer-

den Vorhersagen geglättet (engl. *smoothing*). Diese Glättung ist besonders bei Modellen wichtig, die auf kleinen Trainingsdatensets beruhen.

M5 pruned model tree:
(using smoothed linear models)

```
a2 <= 4.737 :  
| a2 <= 0.963 :  
| | a2 <= 0.669 : LM1 (15/3.042%)  
| | a2 > 0.669 : LM2 (12/2.574%)  
| a2 > 0.963 :  
| | a3 <= 3.992 : LM3 (26/3.313%)  
| | a3 > 3.992 : LM4 (51/13.677%)  
a2 > 4.737 :  
| a1 <= 4.896 :  
| | a1 <= 1.253 :  
| | | a2 <= 6.428 : LM5 (5/2.882%)  
| | | a2 > 6.428 : LM6 (7/4.948%)  
| | a1 > 1.253 :  
| | | a3 <= 6.863 : LM7 (27/6.364%)  
| | | a3 > 6.863 : LM8 (13/12.692%)  
a1 > 4.896 :  
| | a3 <= 5.826 : LM9 (20/11.667%)  
| | a3 > 5.826 : LM10 (24/11.292%)
```



```
LM num: 1 label = 9.051 * a1 + 23.6323 * a2 + 7.0861 * a3 + 0.4671 * a4 - 0.5273 * a5 - 83.2258  
LM num: 2 label = 10.1671 * a1 + 24.4886 * a2 + 8.2389 * a3 + 0.4671 * a4 - 0.5273 * a5 - 91.3816  
LM num: 3 label = 15.0693 * a1 + 30.5883 * a2 + 14.1799 * a3 + 0.4671 * a4 - 0.5273 * a5 - 143.9646  
LM num: 4 label = 22.0777 * a1 + 44.4208 * a2 + 14.9803 * a3 + 0.4671 * a4 - 0.5273 * a5 - 222.9581  
LM num: 5 label = 32.4628 * a1 + 28.3314 * a2 + 16.9035 * a3 + 1.3889 * a4 - 2.1412 * a5 - 220.9217  
LM num: 6 label = 33.7846 * a1 + 28.0643 * a2 + 18.9035 * a3 + 1.3889 * a4 - 2.1412 * a5 - 216.1791  
LM num: 7 label = 37.6627 * a1 + 34.1818 * a2 + 25.2041 * a3 + 1.3889 * a4 - 1.3389 * a5 - 298.1481  
LM num: 8 label = 46.686 * a1 + 39.577 * a2 + 20.0424 * a3 + 1.3889 * a4 - 1.3389 * a5 - 324.9269  
LM num: 9 label = 41.5359 * a1 + 52.1241 * a2 + 51.6383 * a3 + 0.1006 * a4 - 0.5653 * a5 - 564.0883  
LM num: 10 label = 55.6149 * a1 + 62.2628 * a2 + 53.0491 * a3 + 0.2451 * a4 - 0.5653 * a5 - 759.9052
```

Number of Rules : 10

Bild 52: Weka M5P Regressionsbaum in textueller und graphischer Darstellung.

Ein besonderer Vorteil eines M5P-basierten Metamodells besteht in der gleichzeitigen Verwendung diskreter und kontinuierlicher Attributwerte. Dies gilt ebenso für die M5R-basierten Modelle. Während Trainingsdaten für lineare und polynomiale Regressionsfunktionen nur kontinuierliche Variablen enthalten dürfen, können Regressions- und Modellbäume auch diskrete Werte verarbeiten. Beispiele für kontinuierliche Variablen sind Maße für Längen, Winkel oder Radien. Durch den Fertigungsexperten wählbare Konzepte für Werkzeugarmierungen, verschiedene einsetzbare Schmierstoffe oder Ziehringprofile sind Beispiele für diskrete Attribute.

WEKA M5R Regelgenerator

Da M5P Regressionsbäume auf Entscheidungsbäumen beruhen, können sie trotz Pruning eine sehr komplexe Struktur mit vielfachen Verzweigungen annehmen. Prinzipiell lässt sich der Pfad von der Wurzel des Regressionsbaums bis zu einem Modell in einem Blatt auch als Regel formulieren. Bei komplexen Strukturen mindert dies jedoch die Interpretierbarkeit durch den Benutzer. Zudem besteht bei Entscheidungsbäumen immer die Gefahr sich wiederholender Teilabschnitte des Baums, falls beispielsweise der Startknoten des Baums (Wurzel) mit einem ungünstigen Attribut besetzt wurde. Daher wird als Alternative zum

M5P Regressionsbaum der M5R Regelgenerator der WEKA Bibliothek in die Selbstlernkomponente integriert.

Das Training eines M5R Regelsets beginnt mit dem Training eines M5P Regressionsbaums auf der Basis des gesamten Trainingsdatensets D . Unter allen Modellen in den Blättern des Baums wird das lokale lineare Modell mit der geringsten Standardabweichung identifiziert. Alle Datentupel, die diesem lokalen Modell und dem entsprechenden Blatt zugeordnet sind, werden aus D entfernt. Für diesen Teildatensatz wird ein eigener Regelsatz sowie ein lokales, lineares Modell erstellt. Anschließend wird das Schema mit dem reduzierten Datensatz D' iterativ durchlaufen, bis alle Trainingstupel abgearbeitet sind. Dadurch ist sichergestellt, dass jedes Tupel durch eine Regel abgedeckt ist. [HHF99]

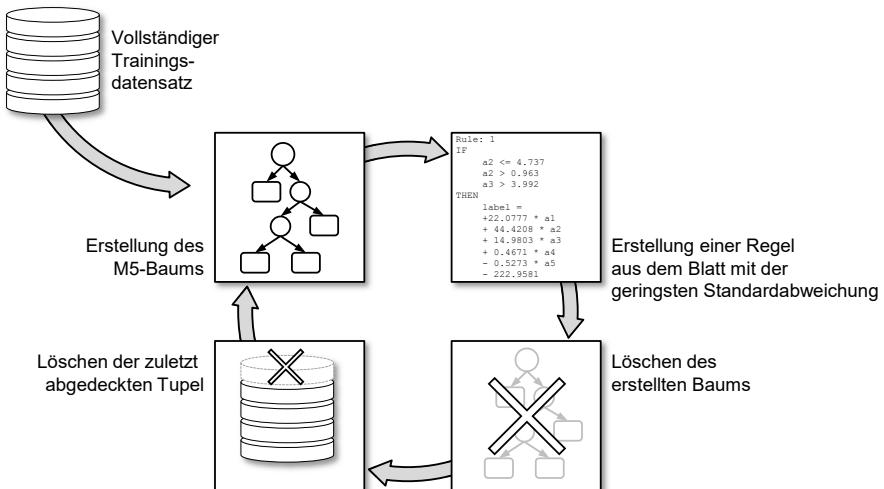


Bild 53: Erstellung eines Metamodells mit dem WEKA M5R Algorithmus [HHF99].

Sobald der Trainingsdatensatz erweitert werden kann, beispielsweise aufgrund neuer Experimente oder Simulationen, wird ein neues Metamodell trainiert. Dies gilt grundlegend auch für alle anderen eingesetzten Metamodelle.

Integration der Metamodelle in Rapidminer

Die Einbindung der genannten Metamodelllerner erfolgt über das Data-Mining Tool Rapidminer. Vorteilhaft an dieser Software ist die flexible und benutzerfreundliche Abbildung der zentralen KDD-Prozessschritte (s. Abschnitt 2.4.2). Jeder neue Data-Mining Prozess wird vom Entwickler durch das Kombinieren sogenannter *Operatoren* erstellt. Die Operatoren werden durch Links, die auf die Weitergabe von Daten, Metamodellen oder Performanzvektoren angepasst sind, miteinander verbunden. Zudem erfordern einige Operatoren den

Aufbau eines verschachtelten Subprozesses. Die Abspeicherung der Data-Mining Prozesse in Form textbasierter Dateien (XML¹⁶) ermöglicht später die batchbasierte und benutzerunabhängige Ausführung des Selbstlernprozesses. Bild 54 zeigt einen exemplarischen Data-Mining Prozess mit der entsprechenden textbasierten Repräsentation.



Bild 54: Exemplarischer Prozess in der Rapidminer® GUI mit XML-basierter Kodierung.

Da der Selbstlernprozess als Template entwickelt und für jeden neuen Durchlauf an die Bezeichnungen der Attribute und Label angepasst wird, werden Platzhalter vergeben, die später mit den entsprechenden Werten ersetzt werden. Für den Operator „Write as Text“ ③ in Bild 54 kann somit beispielsweise der Speicherort der exportierten Textdatei (`value="C:\model.txt"`) vor jeder Ausführung des Prozesses angepasst werden.

6.3.2 Vorverarbeitung der Daten

Dem in Abschnitt 5.2 entwickelten ROPE-Prozess gehen verschiedene Schritte der Datenvorverarbeitung voraus. Dies entspricht den KDD-Prozessschritten drei und vier in Bild

¹⁶ Die Extensible Markup Language ist eine Sprache zur Repräsentation hierarchisch strukturierter Informationen [RB01].

23 (S. 42). Ziel ist zunächst die Daten entsprechend der Vorgaben des Benutzers aus dem Produktdatenmodell abzurufen und für den eigentlichen Trainingsprozess im Rapidminer® bereitzustellen. Der Benutzer kann im Softwaretool zur Steuerung des Selbstlernprozesses aus einem zuvor importierten Versuchsdatensatz den Trainingsdatensatz für den Selbstlernprozess zusammenstellen (s. Bild 76). Wurden Attribute während einer Simulation oder eines Experiments nicht variiert, zum Beispiel aufgrund konstanter Reibverhältnisse, so wird die Selbstlernkomponente diese Attribute in keinem Metamodell berücksichtigen. Sie müssen daher nicht im Trainingsdatensatz vorhanden sein. Ein weiterer Punkt, den es zu berücksichtigen gilt, ist die Möglichkeit, dass einzelne Experimente oder Simulationen keine Ergebnisse liefern. Dieser Fall kann eintreten, wenn beispielsweise die Umformsimulation nicht konvergiert oder einzelne Experimente aufgrund eines Werkzeugschadens abgebrochen werden müssen. Im Trainingsdatensatz bedeutet dies, dass für das Label im entsprechenden Datentupel kein Eintrag existiert. Für große Datenmengen werden einzelne Fehlstellen meist durch einen Ersatzwert, beispielsweise das arithmetische Mittel aller anderen Werte der entsprechenden Größe, kompensiert [TSK10]. Für kleine Datenmengen, wie sie im Rahmen der vorliegenden Arbeit erwartet werden, kann dies jedoch zu einer starken Verzerrung führen. Daher werden Datentupel mit Fehlstellen aus dem Trainingsdatensatz entfernt. Zur Vorbereitung eines Data-Mining Prozesses gehört nach [WEH11; KD15] auch die Analyse der Eingangsgrößen hinsichtlich ihres Einflusses auf das Label. Sofern Attribute existieren, die nicht oder vernachlässigbar gering auf das Label wirken, können diese aus dem Trainingsdatensatz entfernt werden. Dies wirkt sich zudem positiv auf die Rechenzeit eines Data-Mining Prozesses aus. Üblicherweise werden sogenannte Zusammenhangsmaße im Rahmen von Korrelationsanalysen berechnet [FKPT07]. Auch dieser Schritt wird in den Selbstlernprozess integriert und der Benutzer somit entlastet. Bild 55 zeigt die Implementierung der Datenvorverarbeitung.

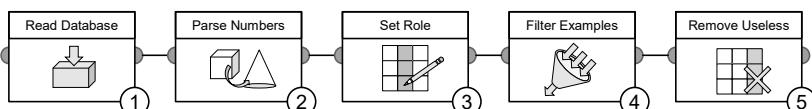


Bild 55: Darstellung der Operatorenkette zur Datenselektion und Datenvorverarbeitung mit Parametereinstellungen.

Der „Read Database“ Operator ① liest die Daten aus der multidimensionalen Datenbasis des Produkt- und Prozessdatenmodells. Da zunächst alle Variablen (Attribute und Label) in der Datenbank als *string* Datentyp deklariert sind, wandelt der „Parse Numbers“ Operator ② entsprechende Attribute in den Datentyp *real* um. Im „Set Role“ Operator ③ wird für den importierten Datensatz das Label definiert für das im späteren Verlauf die Metamodelle trainiert werden sollen. Im gleichen Schritt wird die Versuchsnr. als Datentupel-ID definiert. Geschieht dies nicht, würde Rapidminer® diesen Wert als zusätzliches Attribut interpretieren. Der folgende „Filter Examples“ Operator ④ sorgt über die Einstellung `no_missing_labels` dafür, dass alle Datentupel ohne Wert für das Label entfernt werden, da diese nicht für das überwachte Lernen geeignet sind. Die Möglichkeit,

fehlende Werte durch den Mittelwert der verfügbaren Einträge zu ersetzen, wurde bereits ausgeschlossen. Abschließend entfernt der „remove useless attributes“ Operator ⑤ alle Attribute, die keiner Variation unterliegen und damit keinen Einfluss auf das Label ausüben können. In vorbereitenden Versuchen hat sich gezeigt, dass eine Parameter-Einstellung für diesen Operator von exakt Null für die Einstellung „numerical min deviation“ keine Entfernung von „unwirksamen“ Attributen bewirkt. Erst ein Wert etwas größer als Null zeigt den gewünschten Erfolg, weshalb hier 0,01 gewählt wird.

6.3.3 Automatische Attributselektion

Die vorhergehende Filterung der Attribute (Schritt ⑤ in Bild 55) entspricht einer Vorgehensweise, die unabhängig vom später angewandten Data-Mining Algorithmus ausgeführt wird. Verfahren, die Variablen mit einer „Rückkopplung“ (s. Bild 56) zum Trainingsalgorithmus auswählen, werden unter der Bezeichnung *Attribute Subset Selection* zusammengefasst. Dabei wird aus dem ursprünglichen Set an Attributen mit der Attributanzahl A im Trainingsdatensatz D ein Subset D' mit A' Attributen derart ausgewählt, dass die Vorhersageperformanz des Metamodells maximiert wird. Da für A Attribute 2^A mögliche Kombinationen existieren, werden üblicherweise heuristische Methoden genutzt, die einen reduzierten Suchraum nach optimalen Kombinationen durchsuchen. Die übliche Bezeichnung lautet *greedy* (engl. *gefräbig, gierig*), da sie schrittweise den nächsten Zustand auswählen, der zum Zeitpunkt der Wahl den größten Gewinn bzw. das beste Ergebnis verspricht. Greedy-Algorithmen sind oft schnell, lösen viele Probleme jedoch nicht optimal [Cor01].

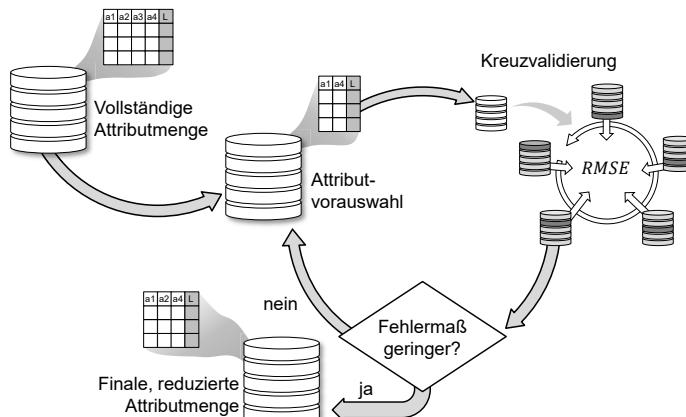


Bild 56: Ablauf der automatischen Attributselektion in Rapidminer.

Die im Rahmen dieser Arbeit genutzten greedy-Algorithmen sind die schrittweise Vorwärtsselektion und die schrittweise Rückwärtseliminierung. Die *schrittweise Vorwärtsselektion*

beginnt mit einer Nullmenge an Attributten und fügt dieser das beste der originalen Attribute hinzu. Bei jeder weiteren Iteration wird das jeweils beste der verbleibenden Attribute in die bereits ausgewählte Menge aufgenommen. Im Gegensatz dazu verfolgt die *schrittweise Rückwärtseliminierung* das Ziel aus der gesamten Menge an Attributten iterativ das jeweils schlechteste zu entfernen. Beide Verfahren können kombiniert werden [HKP12]. Für diskrete Zielgrößen können zudem Entscheidungsbäume (z. B. ID3, C4.5, CART) zum Einsatz kommen. Aus allen Daten und Attributten wird ein Entscheidungsbaum trainiert und sofern ein Attribut nicht in diesem Baum enthalten ist, wird es als irrelevant eingestuft und für das weitere Data-Mining ausgeschlossen.

Die Implementierung der Attributselektion in Rapidminer® ist äußerst benutzerfreundlich gestaltet (s. Bild 57). Für beide Varianten (schrittweise Vorwärtsselektion und Rückwärtseliminierung) werden die entsprechenden Operatoren gewählt und mit dem letzten Operator der Datenvorverarbeitung in Bild 55 verbunden. Da jeder Datenstrom nur zwei Operatoren verbinden kann, erfolgt eine Vervielfachung desselben („Multiply“).

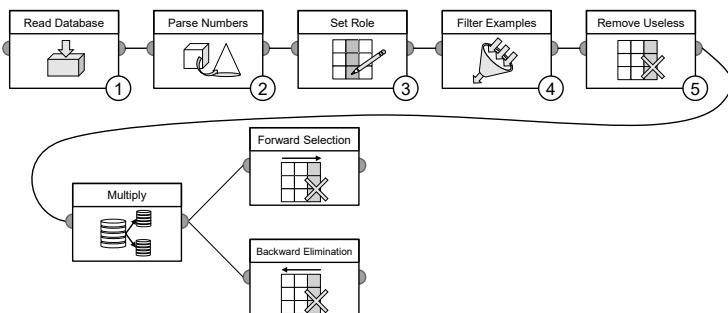


Bild 57: Integration der automatischen Attributselektion in den Data-Mining Prozess.

6.3.4 Umsetzung des ROPE-Prozesses

In den Abschnitten 5.2 und 5.3 wurde der ROPE-Prozess entwickelt, um Metamodelle robust zu optimieren und ihre Vorhersageperformanz abzuschätzen. Anschließend werden diese Performanzen verglichen und das bestgeeignete Metamodell an Slassy übergeben. Die folgenden Abschnitte behandeln die Details der Umsetzung des ROPE-Prozesses im Data-Mining Werkzeug Rapidminer®.

Statistisch robuste Performanzschätzung

Die statistisch robuste Performanzschätzung wird im gesamten Selbstlernprozess mehrfach benötigt, nämlich bei jeder Iteration eines Optimierungsprozesses zur Beurteilung der jeweils erzeugten Modellinstanz, bei der im Abschnitt 6.3.3 vorgestellten Attributselektion

mit „Rückkopplung“ zum Metamodelltraining und für die Ermittlung der Performanzvektoren, die für die Auswahl des bestgeeigneten Modells benötigt werden.

Jede Performanzschätzung beruht auf einer n -fach wiederholten k -fachen Kreuzvalidierung, die direkt in Rapidminer® umgesetzt werden kann. Eine Kreuzvalidierung wird durch den gleichnamigen Operator repräsentiert (s. Bild 58), für den die Teilprozesse Training und Test definiert werden müssen.

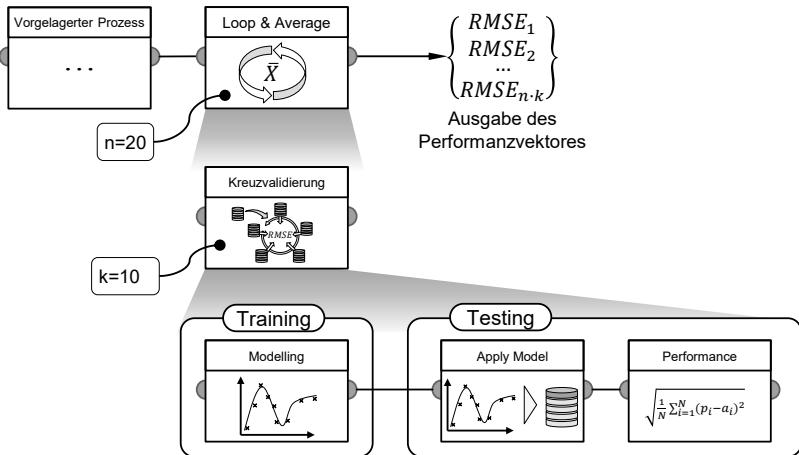


Bild 58: Prinzipieller Aufbau der robusten Performanzschätzung.

Im Training wird ein Metamodell mit einem reduzierten Datensatz generiert, um beim Testen mit diesem Modell Labelwerte vorherzusagen und mit den bekannten Werten zu vergleichen. Das Training ist in Bild 58 durch einen Operator mit der Bezeichnung „Modelling“ repräsentiert, der für die finale Implementierung jeweils mit einem Operatoren der ausgewählten Trainingsalgorithmen (s. Abschnitt 6.3.1) ersetzt werden muss. Für das Testen wird der beim Training ausgeschlossene Teil der Daten an den „Apply“ Operator übergeben. Dieser fügt dem Datensatz auf Basis des zuvor erzeugten Modells die Spalte „prediction“ hinzu. Aus den Differenzen zwischen den Werten des Labels und den Vorhersagen können die in Tabelle 4 (S. 49) gezeigten Kennwerte für Prognosefehler berechnet werden. Wurde der Operator für die Kreuzvalidierung einmal durchlaufen, d. h. mit k internen Wiederholungen, so können von ihm das (auf allen Daten) trainierte Metamodell sowie die für dieses Modell geschätzte Performanz an nachfolgende bzw. übergeordnete Operatoren übergeben werden. Die vollständige, robuste Performanzschätzung ist dadurch sichergestellt, dass die k -fache Kreuzvalidierung n -mal wiederholt wird und ihre n Ergebnisse gemittelt werden. Hierfür kann der „Loop and Average“ Operator genutzt werden, in den die Kreuzvalidierung integriert wird (s. Bild 58).

Die Ausgabe des „Loop and Average“ Operators beinhaltet den Gesamtmittelwert der n -mal wiederholten k -fachen Kreuzvalidierungen und die aus den n Werten berechnete

Standardabweichung. Diese Einzelwerte eignen sich zur „manuellen“ Beurteilung der Prognosequalität eines Data-Mining Prozesses, jedoch nicht für die maschinelle Verarbeitung in den späteren statistischen Tests (ANOVA und post-hoc). Daher müssen an geeigneter Stelle im Data-Mining Prozess die Performanzkennwerte protokolliert und zum Performanzvektor (s. Abschnitt 5.2, S. 5.2) zusammengefasst werden. Hierfür wird der Prozess aus Bild 58 um einen Export Operator erweitert, der alle berechneten Performanzkennwerte pro Kreuzvalidierungsiteration in eine Auslagerungsdatei schreibt. Da aufgrund der Erkenntnisse aus Abschnitt 5.2 $n = 20$ und $k = 10$ für die Performanzschätzung festgelegt werden, ergibt sich pro Metamodell ein Performanzvektor mit 200 Einträgen.

Optimierung des Trainings der Metamodelle

Im Abschnitt 5.3 wurden mit der schrittweisen Parametervariation und der evolutionären Parameteroptimierung zwei Strategien eingeführt, mit denen das Training eines Metamodells optimiert werden kann. Ziel ist es, für die Parameter, mit denen das Training eines Metamodells direkt beeinflusst werden kann, eine optimale Einstellung zu finden.

Beide Optimierungen sind in Rapidminer® durch entsprechende Operatoren realisiert. Ähnlich zur Kreuzvalidierung ist eine verschachtelte bzw. hierarchische Struktur erforderlich, bei der in den Optimierungsprozess ein Validierungsprozess eingebettet wird. Dieser ist für die Schätzung der Performanz der erzeugten Modellinstanzen in den Optimierungsschritten notwendig. Da auch diese Performanzschätzung ein robustes Ergebnis liefern muss, wird auf die Strategie der n -mal wiederholten k -fachen Kreuzvalidierung zurückgegriffen. Anders als bei der Performanzschätzung des finalen Metamodells, findet in der Optimierung jedoch kein Export der Fehlerkennwerte und keine Erzeugung eines Performanzvektors statt.

Bei der Umsetzung der Optimierung und der Festlegung der Werte für n und k müssen die Eigenschaften der Optimierungsstrategien berücksichtigt werden. Die schrittweise Parametervariation findet das optimale Parameterset für den jeweiligen Lernalgorithmus durch einen Versuchsplan, bei dem die Wertebereiche aller Lernparameter segmentiert und alle entstehenden Abschnitte frei miteinander kombiniert werden. Dadurch wird der gesamte Wertebereich abgetastet und sichergestellt, dass eine Parameterkonfiguration nahe am globalen Optimum gefunden wird. Wird der Operator der polynomialen Regression als Beispiel bemüht, stehen fünf Parameter zur Auswahl. Bei einer Einteilung der Wertebereiche dieser fünf Parameter in zehn Segmente ergeben sich $11^5 = 161.051$ Kombinationen, deren Berechnung je nach Computerleistung und zu verarbeitender Datenmenge viel Zeit in Anspruch nimmt, denn für jede Kombination wird eine robuste Performanzschätzung durchgeführt. Mit höheren Werten von n und k steigt der Zeitbedarf noch weiter. Für die Performanzschätzung innerhalb der schrittweisen Parameteroptimierung wird daher eine 20-malige 10-fache Kreuzvalidierung durchgeführt.

Die evolutionäre Optimierung der Trainingsparameter verläuft hingegen wesentlich schneller, da sie feststellt, welche Parameterwerte eher zu einer Reduzierung des Fehlermaßes

führen und dies bei den folgenden Iterationen berücksichtigt. So kann die Optimierungsfunktion schneller einen optimalen Wertebereichen für die Parameter finden. Für die Performanzschätzung innerhalb der evolutionären Optimierung wird daher eine 100-malige 10-fache Kreuzvalidierung durchgeführt.

Bild 59 zeigt für die evolutionäre Optimierung einer polynomialen Regression die schematische Implementierung des in dieser Arbeit entwickelten ROPE-Prozesses in Rapidminer. Nach der Datenvorverarbeitung und der Attributselektion ① wird der ROPE-Prozess der Übersicht halber als eigener Subprozess ② gestaltet. Der oberste „Loop and Average“ Operator ③ und die darin ausgeführte Kreuzvalidierung ④ sind für die robuste Schätzung der Performanz des optimierten Modells zuständig. Der Kreuzvalidierungsoperator beinhaltet im Trainingsabschnitt nicht das zu trainierende Metamodell, wie dies bspw. in Bild 58 dargestellt ist. An dieser Stelle wird ein Subprozess ④ eingefügt, in dem die eigentliche Optimierung ⑤ integriert ist. Bei jeder Iteration der 10-fachen Kreuzvalidierung ④ nutzt die Optimierung einen 90%-igen Trainingsdatensatz. Die verbleibenden 10% werden für die Schätzung der Performanz im „Testing“ Subprozess genutzt ⑥, welche zur späteren Weiterverarbeitung als Performanzvektor exportiert wird ⑦. Das Training des optimalen Metamodells mit 100% der Daten erfolgt nach der letzten Iteration der Kreuzvalidierung, im vorliegenden Fall nach zehn Iterationen. Jeder Aufruf des Subprozesses ④ leitet einen Datensatz an den Optimierungsoperator ⑤ weiter, der als Ergebnis die optimalen Trainingsparameter für diesen Datensatz ausgibt. Diese werden dem Trainingsoperator durch eine „Parameterfestlegung“ ⑪ zugewiesen, anschließend das entsprechende Metamodell erstellt ⑫ und „nach oben“ an die Kreuzvalidierung übergeben. Die Evaluierung der Modellinstanzen in den einzelnen Optimierungsschritten erfolgt durch eine robuste 50-malige 10-fache Kreuzvalidierung ⑨, deren Trainingssalgorithmus ⑩ dem auf der Ebene des Optimierungsoperators ⑧ entsprechen muss. Für die Einstellungen der evolutionären Optimierung haben sich die im Kasten ⑫ gezeigten Werte bewährt. Diese Grundeinstellungen werden für alle evolutionären Optimierungen im Selbstlernprozess beibehalten. Speziell für die Optimierung der polynomialen Regression, werden die im rechten Teil des Kastens gelisteten Trainingsparameter berücksichtigt. Für jeden Parameter werden eigene Grenzen festgelegt, in denen der Optimierungsalgorithmus arbeiten darf. Für den maximalen Polynomgrad wird bspw. ein Wert zwischen eins und fünf zugelassen. Das Ergebnis jedes ROPE-Prozesses ist ein mit dem Datensatz D optimal und robust trainiertes Metamodell sowie dessen robust geschätzte Performanz in Form eines Vorhersagefehlers (Wurzel d. mittleren Fehlerquadrats). Die Anzahl der ROPE-Prozesse je Selbstlernprozess wird in Abschnitt 6.3.6 festgelegt.

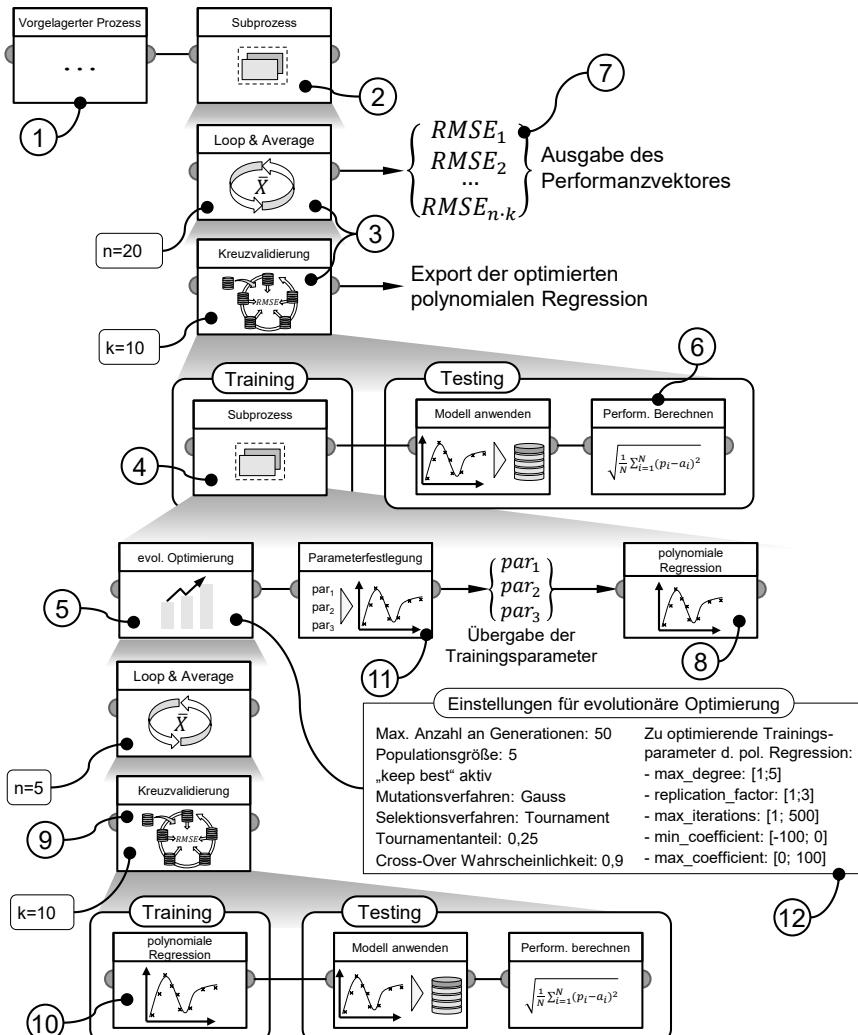


Bild 59: Umsetzung des in dieser Arbeit entwickelten ROPE-Prozesses für die polynomiale Regression mit evolutionärer Optimierung der Trainingsparameter.

6.3.5 Selektion des bestgeeigneten Metamodells

Wie in Abschnitt 5.4 dargestellt, erfolgt die Auswahl des bestgeeigneten Metamodells auf Basis der aufeinander folgenden statistischen Tests ANOVA und Tukey's HSD. Ebenso wurde gezeigt, dass die Voraussetzungen zur Durchführung der ANOVA erfüllt sind bzw. nur geringfügig verletzt werden und die ANOVA darauf sehr robust mit einer vernachlässigbar geringen Alphafehlerkummulierung reagiert.

Ausgangspunkt für die Implementierung der automatischen Modellselektion in Matlab® sind die Performanzvektoren des vorangegangenen Data-Mining Schritts (s. Bild 60). Sie wurden durch 20-mal wiederholte 10-fache Kreuzvalidierungen ermittelt und enthalten in den Zeilen die ermittelten Vorhersagefehler. Als Kennwert und Vergleichskriterium wird die Wurzel des mittleren Fehlerquadrats gewählt (s. Tabelle 4, S. 49), da dieses Maß die Einheit der Zielgröße (z. B. Kilonewton, Prozent, Millimeter) wiedergibt.

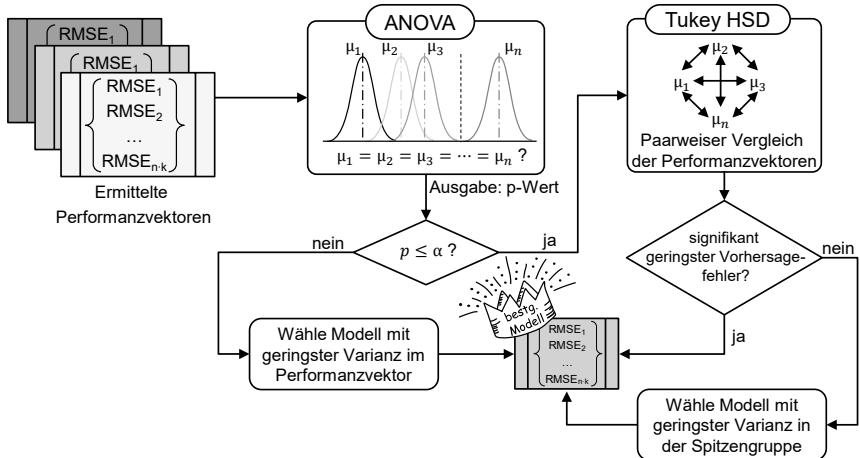


Bild 60: Ablauf der automatischen Metamodellselektion in Slassy.

Bild 60 zeigt die Durchführung der ANOVA als ersten Schritt zur Modellauswahl. Das Ergebnis jeder ANOVA wird in Form eines p-Werts ausgedrückt, der mit einem zuvor festgelegten Signifikanzniveau α zu vergleichen ist. Liegt der p-Wert unter dem Signifikanzniveau, es wird $\alpha = 0,05$ gesetzt, so besteht zwischen mindestens zwei Performanzvektoren ein signifikanter Unterschied, d. h. die Nullhypothese (s. Abschnitt 5.4, S. 82) wird verworfen. Im Falle eines nicht-signifikanten Ergebnisses, d. h. die Nullhypothese wird angenommen, kann davon ausgegangen werden, dass die trainierten und verglichenen Metamodelle den Wert der jeweiligen Zielgröße mit ähnlicher Güte vorhersagen. Einzig eine Unterscheidung hinsichtlich der Vorhersagevarianz bietet ggf. eine ausreichende Differenzierung der Metamodelle. Die Varianz kann an dieser Stelle mit einem Vertrauensintervall gleichgesetzt werden, in dem der vorhergesagte Wert der Zielgröße tatsächlich liegt. Je geringer die Varianz

anz, desto enger fällt das Vertrauensintervall aus und desto verlässlicher ist die Vorhersage. Der Selbstlernprozess endet vorzeitig und gibt als bestgeeignetes Metamodell dasjenige mit der geringsten Varianz und dessen Vorhersagegenauigkeit aus.

Sofern die ANOVA einen signifikanten Unterschied zwischen den Mittelwerten von mindestens zwei Performanzvektoren festgestellt hat, erfolgt im nächsten Schritt die Ausführung des Tukey HSD-Tests. Insbesondere in der Matlab® Umgebung benötigt der Tukey HSD-Test Berechnungen aus der ANOVA, weshalb eine vorherige Ausführung derselben notwendig ist. Als Ergebnis wird eine „ $\frac{M \cdot (M-1)}{2}$ “-Kreuz 6-Matrix ausgegeben, in der die Ergebnisse der paarweisen Vergleiche der Mittelwerte aller Performanzvektoren der M Metamodelle dargestellt sind, wie Tabelle 8 exemplarisch zeigt.

Tabelle 8: Beispielhafte Tabelle eines Tukey HSD-Tests für drei Mittelwertdifferenzen.

Modell μ_a	Modell μ_b	$Z_{5\%}$	$ \mu_a - \mu_b $	$Z_{95\%}$	p-value
1	2	0,926	1,500	2,074	0,000
1	3	1,676	2,250	2,824	0,000
2	3	0,176	0,750	1,324	0,016

Spalte eins und zwei markieren die Indizes der verglichenen Modelle. Die Spalten drei und fünf geben die untere ($Z_{5\%}$) bzw. die obere ($Z_{95\%}$) Grenze des Konfidenzintervalls für die wahre Mittelwertdifferenz in Spalte vier an. Da es sich bei jedem paarweisen Vergleich um einen Hypothesentest handelt, zeigt die letzte Spalte den p-Wert für diesen Test. Liegt der Wert unter der festgelegten Grenze von $\alpha = 0,05$ so ist der Unterschied zwischen den verglichenen Werten signifikant. Für Tabelle 8 zeigt sich, dass alle Mittelwertdifferenzen signifikant sind. Das Ergebnis des Tukey-HSD lässt eine Fallunterscheidung zu. Eine Möglichkeit wäre, dass sich die Performance eines Metamodells von den Performanzen der anderen Modelle signifikant unterscheidet. Daneben kann auch eine „Spitzenreitergruppe“ existieren bzw. kann zwischen keinem der Modelle ein signifikanter Unterschied festgestellt werden. In diesen beiden Fällen wird wie bei der ANOVA das Modell mit der geringsten Varianz als finales Metamodell gewählt.

Abschließend erfolgt die Übertragung des bestgeeigneten Modells in die Wissensbasis. Da das Data-Mining Tool Rapidminer® alle im Rahmen dieser Arbeit betrachteten Metamodelle in textbasierter Form (maschinenlesbarer String, s. Bild 52) ausgeben kann, werden diese Repräsentationen direkt in das Produkt- und Prozessdatenmodell übertragen. Eine Auswertung des Strings erfolgt zur Laufzeit der Design for Manufacture Analyse durch den Einsatz sogenannter regulärer Ausdrücke (s. Abschnitt 6.5.1).

6.3.6 Weitere Umsetzung des Selbstlernprozesses

Zum Abschluss der Umsetzung des Selbstlernprozesses wird die Anzahl der zu vergleichenden Metamodelle in der Grundversion von Slassy festgelegt. Ebenso werden der Import

der Trainingsdaten in den Selbstlernprozess und der Export des bestgeeigneten Modells und dessen Performanzkennwerte kurz erläutert.

Das Konzept des Selbstlernprozesses ermöglicht den Vergleich einer beliebigen Menge an Metamodellen. Für die erstmalige, konzeptionelle Implementierung von Slassy erfolgt jedoch, durch gezielte Kombination unterschiedlicher Data-Mining Methoden und Algorithmen, eine Begrenzung dieser Menge. In Abschnitt 6.3.1 werden insgesamt vier Methoden zum Training eines Metamodells unter Berücksichtigung der Anforderungen aus Abschnitt 4.2 ausgewählt. Des Weiteren werden mit der schrittweisen Vorwärtsselektierung und Rückwärtseliminierung (s. Abschnitt 6.3.3) zwei Attributselektionsverfahren vorgestellt mit denen, alternativ zum vollständigen Datensatz, reduzierte Datensätze erzeugt werden können, die unter Umständen zu Metamodellen mit einer besseren Vorhersageperformanz führen. Die robuste Optimierung der trainingsbestimmenden Parameter bietet zwei weitere Möglichkeiten, um den Data-Mining Prozess zu gestalten. Die Kombination der genannten Data-Mining Methoden (s. Bild 61) führt zu einer Basismenge an 24 Metamodellen, von denen eines durch die beschriebenen statistischen Tests als bestgeeignetes Modell identifiziert und für Design for Manufacture Analysen verwendet werden kann.

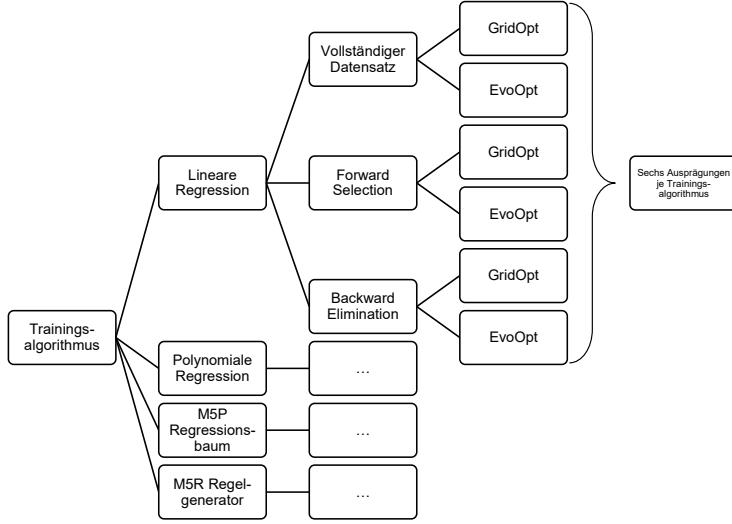


Bild 61: Die gewählten Kombinationen von Data-Mining Algorithmen und Methoden führen zu 24 Metamodellen.

Für den Import der Trainingsdaten stehen im Data-Mining Tool Rapidminer® Operatoren zur Verfügung, die in der Lage sind unterschiedliche Datenquellen (z. B. Spreadsheets, CSV-Dateien, Web-Datenbanken) zu nutzen. Die Datenschnittstelle zum Produkt- und Prozessdatenmodell ist mithilfe der Datenmodellierungssprache SQL umgesetzt, daher muss auch die Anbindung an Rapidminer® entsprechend gestaltet werden. Hierzu wird dem Import

Operator ein Platzhalter für den SQL-Befehl zugewiesen, der die korrekten Daten aus dem Produkt- und Prozessdatenmodell lädt. Die Definition des SQL-Befehls erfolgt zur Laufzeit am Beginn des Selbstlernprozesses und ist im Anwendungsbeispiel dargestellt. Ebenso wird beim Export des final ausgewählten Metamodells sowie des Performanzkennwerts verfahren.

6.4 Das Synthesewerkzeug

Der Konstrukteur wird durch das Synthesewerkzeug in die Lage versetzt, eine stofflich-geometrische Beschreibung des blechmassivumgeformten Bauteils zu erzeugen. Insbesondere die Definition der Haupt- und der Nebenformelemente (s. Abschnitt 3.3) ist eine Grundvoraussetzung für die spätere Design for Manufacture Analyse der Gestaltentwürfe. Die frühen Versionen des Konstruktionssystems *mfk* (KS_{mfk}) realisierten die geometrische Modellierung durch eine eigene Modellierungslogik. Hier war die Gestalt im Produktmodell enthalten (s. Bild 62). Mit konstruktionsorientierten Eingaben des Benutzers und durch elementspezifische Modellierungsfunktionen wurde jedes Gestaltelement als Volumenkörper mit Parametern beschrieben. Die Klassifikation der Nebenformelemente in Abschnitt 3.3 hat gezeigt, dass die Vielfalt blechmassivumgeformter Bauteilen weiter steigen wird. Wie bereits bei [Bac97; War01], wird daher ein kommerzielles CAD-System zur geometrischen Produktmodellierung genutzt. Durch Kopplung des Produkt- und Prozessdatenmodells im Assistenzsystem und dem 3D-Modell im CAD-System wird eine vollständige Produktbeschreibung erreicht (s. Bild 62).

6.4.1 Bereitstellung der Hauptformelemente

Wie in Abschnitt 3.3 dargestellt, erfüllen Hauptformelemente v. a. die Funktion, Nebenformelemente zu „tragen“ und die dort aufgenommenen Lasten weiterzuleiten. Als Beispiel hierfür wurde der Synchronring in Bild 34 (S. 61) genannt. Eine Recherche bezüglich weiterer blechmassivumgeformter Bauteile ergab zudem, dass rotationssymmetrische Formen vorherrschen, was auch dem häufigen Einsatz in Kfz-Getrieben geschuldet ist. Hinzu kommt die Eigenschaft derartiger Bauteile, dass während der Umformung keine seitlichen Kräfte auf den Stöbel bzw. das Umformwerkzeug wirken, ein Zustand der von den Fertigungsexperten aufgrund der höheren Genauigkeit und des geringeren Werkzeugverschleißes erwünscht ist. Daher werden im Synthesetool rotationssymmetrische und flächige bzw. dünnwandige Hauptformelemente implementiert.

Damit der Konstrukteur während der Synthese schnell auf Formelemente der Blechmassivumformung zugreifen kann, werden sie ihm in Form von Features (s. Abschnitt 2.2.1) angeboten. Für die Umsetzung der Features-Technologie stehen im CAD-System CATIA V5 verschiedene Methoden zur Verfügung (s. Bild 63).

Für das dynamische Erzeugen neuer Formelemente sowie deren schnelles Ändern bietet sich dabei das *UserFeature* an [Bra08]. Als sogenanntes „Black Box“ Element sind nach

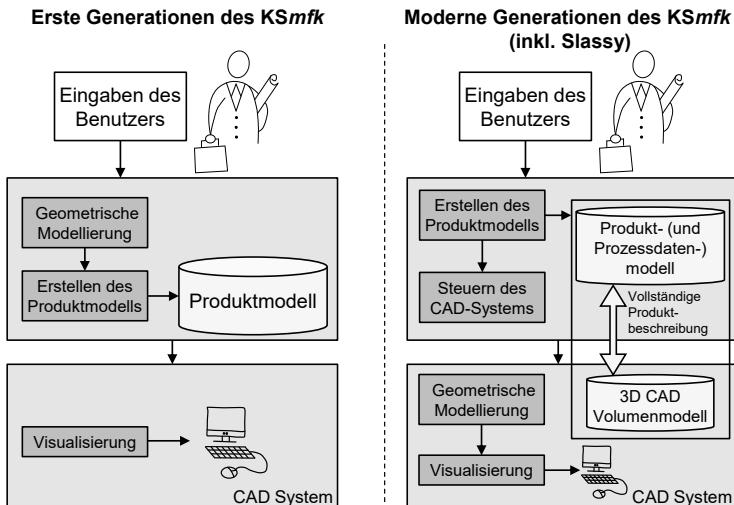


Bild 62: Wie bei den modernen Versionen des KSmfk nutzt auch Slassy das CAD-System zur Modellierung.

der Instanziierung eines UserFeatures nur die explizit veröffentlichten Geometrieparameter verfügbar. Auf geometriebestimmendes Wissen in Form von Regeln und Constraints des Features kann hingegen nicht zugegriffen werden. Dies dient nicht nur dem Wissenschutz, sondern verhindert gleichzeitig, dass veraltetes DfM-Wissen aus bereits instantiierten Bauteilen verbreitet wird. Das aktuellste Wissen darf nur über Slassy zugreifbar sein.

Jedes blechmassivumgeformte Bauteil besteht aus einem Hauptformelement-Typ und mindestens einem Nebenformelement-Typ, der gemustert auftreten kann. Zum Zeitpunkt des Verfassens dieser Arbeit lag der Fokus der Forschungsarbeiten auf Bauteilen mit gleichen Nebenformelementen (z. B. Lasche, Mitnehmer, Verzahnung). Da aber in Zukunft eine noch stärkere Funktionsintegration zu erwarten ist, unterstützt Slassy bei der Synthese auch die Kombination unterschiedlicher Nebenformelemente. Sobald entsprechende Prozessdaten vorliegen, kann mithilfe des Selbstlernprozesses das relevante Konstruktionswissen erhoben werden. Als Hauptformelement-Grundkörper werden die Grundformen „Scheibe“, „Napf mit umlaufenden Bord“ sowie „Napf ohne Bord“ implementiert. Jedem Grundkörper sind Standardwerte für dessen Geometrieparameter zugewiesen, die in Zusammenarbeit mit den Fertigungsexperten bestimmt werden. Beispielsweise wird dabei die Stärke und der Zuschnitt der Blechrunde berücksichtigt, aus der das blechmassivumgeformte Bauteil später geformt wird. Bei der Parametrisierung der Grundkörper von Hauptformelementen werden diejenigen Parameter identifiziert, die für die externe Steuerung der Geometrie zulässig sind. Unter externer Steuerung wird hierbei der Zugriff auf das CAD-Modell durch Slassy verstanden (s. Bild 64). Tabelle 9 fasst die implementierten

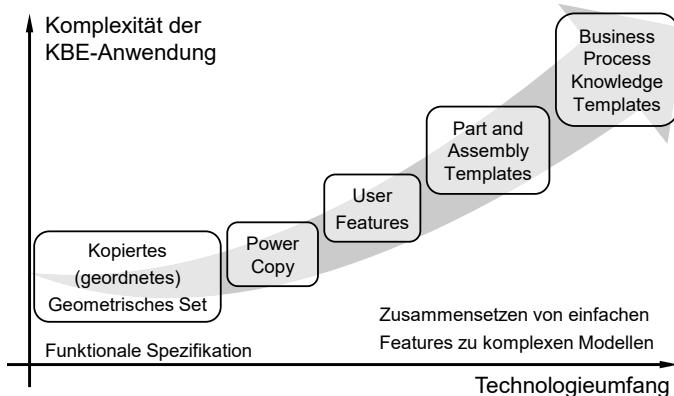


Bild 63: CAD-Methoden zur Wiederverwendung von Wissen nach [Bra08]

Hauptformelement-Grundkörper mit den entsprechenden Parametern zusammen. Die Präfixe der Parameterbezeichnungen (`W_C0_`, `W_C1_`, `W_P_`) bezeichnen die Abkürzungen der einzelnen Hauptformelemente (`W`), wobei C0 und C1 für je einen Typ eines tiefgezogenen Napfes (engl. *cup*) und P für das Hauptformelement „Scheibe“ (engl. *plate*) stehen.

Um eine größtmögliche Flexibilität bei der Gestaltung eines blechmassivumgeformten Bauteils zu gewährleisten, darf der Konstrukteur während der Kombination von Haupt- und Nebenformelementen keinen wesentlichen Einschränkungen unterliegen. Dies wird durch generische Referenzobjekte in den Features erreicht, die zur Positionierung eines Nebenformelements am Hauptformelement notwendig sind. Beim Zusammenbau des Bauteils durch das Synthesetool in Slassy werden die Referenzobjekte der Haupt- und Nebenformelemente miteinander synchronisiert (s. Abschnitt 6.4.3).

6.4.2 Bereitstellung der Nebenformelemente

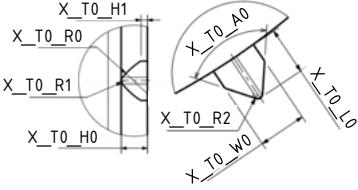
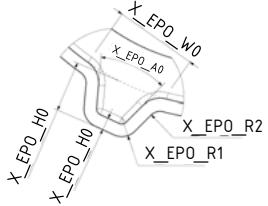
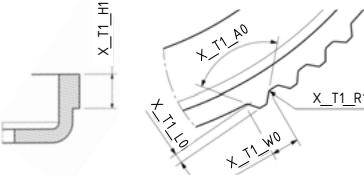
Die Nebenformelemente sind für die Funktionserfüllung eines blechmassivumgeformten Bauteils von zentraler Bedeutung (s. Abschnitt 3.3). Ihre Umsetzung im CAD-System CATIA V5 erfolgt wie bereits bei den Hauptformelementen durch den Einsatz von UserFeatures. Die bis zum Abschluss dieser Arbeit im Forschungsverbund SFB/TR73 berücksichtigten Nebenformelemente „Sperrverzahnung“, „offener Mitnehmer“, „geschlossener Mitnehmer“ sowie „Verzahnung“ werden hinsichtlich der notwendigen Parameter zur externen Steuerung analysiert. Diese sind, zusammen mit technischen Skizziken der Nebenformelemente, in Tabelle 10 dargestellt. Die Präfixe der Parameterbezeichnungen (`X_T0_`, `X_EPO_`, `X_T1_`) bezeichnen die Abkürzungen der einzelnen Nebenformelemente (`X`), wobei T0 und T1 für je einen Verzahnungstyp und EPO für das Element „offener Mitnehmer“ (engl. *engaging piece open*) stehen.

Tabelle 9: Implementierte Grundkörper von Hauptformelementen in Slassy

Bezeichnung	Abbildung / Skizze mit Parametern	Umformverfahren
Napf mit Bord		Tiefziehen (mit Niederhalter)
Napf ohne Bord		Tiefziehen
Scheibe		Walzen, Stauchen

Die Positionierung eines Nebenformelementes erfolgt über die externalisierten Referenzobjekte des Haupformelements „xy.plane“, „Point.Base“ und „Line.SymAxis“ (s. Bild 64). Diese werden wiederum innerhalb des Nebenformelements genutzt, um Konstruktionselemente (z. B. Skizzen, Solids, Ausbrüche) so zu definieren, dass nach dem Assemblieren des Bauteils mittels Bool'scher Operation das Haupt- und das Nebenformelement einen geschlossenen Körper bilden. Im Allgemeinen sind die Nebenformelemente als Positivkörper gestaltet, d. h. sie werden bei der Synthese dem Volumen des Hauptformelements hinzugefügt. Jedoch lässt sich nicht jedes Nebenformelement auf diese Weise realisieren. Der offene Mitnehmer in Tabelle 10 beispielsweise verlangt die Anwendung eines Negativkörpers, der vom Volumen des Hauptformelements abgezogen wird. Diese Abzugsgeometrien müssen bei der Konstruktion eines Nebenformelements berücksichtigt und ihre Geometrieparameter an das Formelement angepasst werden. Abschließend wird das hinzugefügte Nebenformelement als rotationssymmetrisches Muster vervielfältigt. Der Benutzer gibt dabei die Anzahl der gewünschten Instanzen an, wobei Slassy diesen Wert nach oben begrenzt. Dazu wird der Abstand zwischen den Instanzen im Bogenmaß berechnet und mit dem spezifischen Minimalwert verglichen, der als Parameter im UserFeature des jeweiligen Nebenformelements hinterlegt ist.

Tabelle 10: Implementierte Nebenformelemente in Slassy

Bezeichnung	Abbildung / Skizze mit Parametern	Umformverfahren
Sperrverzahnung		Fließpressen
Offener, bzw. geschlossener Mitnehmer		Tiefziehen, Stauchen
Randverzahnung		Inkrementelles Umformen

6.4.3 Erzeugung eines Bauteilentwurfs

Die Synthese eines blechmassivumgeformten Bauteils erfolgt über Bool'sche Operationen zwischen den UserFeatures des Haupt- und des Nebenformelementes. Die notwendige Synchronisierung der Referenzobjekte übernimmt das Assistenzsystem wie in Bild 64 dargestellt. Der Konstrukteur wählt in der Oberfläche des Assistenzsystems zunächst ein Hauptformelement aus und ändert ggf. dessen Standardwerte. Anschließend fügt er das Nebenformelement hinzu und legt die Anzahl der Instanzen am Umfang des Hauptformelements sowie die Geometrie des Nebenformelementes fest. Diese Informationen werden direkt im CAD-Modell des Bauteils hinterlegt. Durch den Benutzer können zudem gezielte Synchronisationen von Geometrieparametern zwischen dem Haupt- und dem Nebenformelement eingebracht werden, beispielsweise zwischen der Blechdicke des Hauptformelements und der Wandstärke des offenen Mitnehmers.

Wie in Bild 64 dargestellt, werden die Modelle der UserFeatures in die Windows-Ordnerstruktur des Assistenzsystems integriert. Jedes Feature wird, zusammen mit einer technischen Skizze und einer Icon-Darstellung, in einem eigenen Ordner abgelegt.

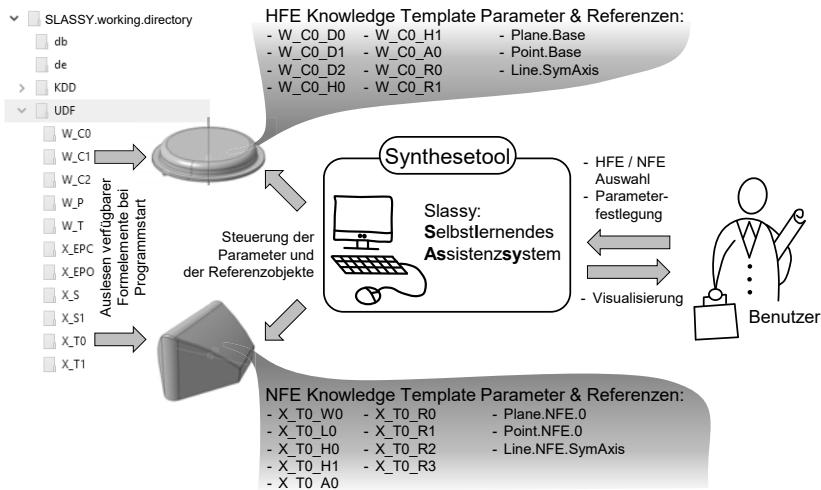


Bild 64: Das Synthesetool verwaltet die Geometrieparameter, Referenzen zwischen den Haupt- und Nebenformelementen (HFE, NFE) und dient als Schnittstelle zum Benutzer.

Die Benennung des Ordners identifiziert jedes UserFeature eindeutig, wobei das Präfix „W“ ein Hauptformelement und „X“ ein Nebenformelement bezeichnet. Die nachfolgende Buchstaben-Zahlen Kombination benennt den Typ des Formelements, also „C“ für *Cup* (engl. *Napf*) oder „EPC“ für *engaging piece open* (engl. *offener Mitnehmer*). Zur Integration neuer Formelemente muss der Benutzer lediglich einen weiteren Ordner erstellen und darin das vorbereitete UserFeatures sowie eine Skizze und eine Icon-Darstellung ablegen.

6.5 Das Analysewerkzeug

Das Analysetool des selbstlernenden Assistenzsystems unterstützt den Konstrukteur bei der Design for Manufacture Analyse blechmassivumgeformter Bauteile. Dabei berücksichtigt es die speziellen Anforderungen der Blechmassivumformung die in Abschnitt 3.2 dargestellt wurden. Für eine Design for Manufacture Analyse wird zunächst die Bauteilkonfiguration ausgelesen, die der Konstrukteur im Synthesetool definiert hat. Sofern für das zu analysierende Bauteil fertigungsrelevantes Konstruktionswissen vorliegt, wird dieses im Inferenzmechanismus verarbeitet und anschließend das Ergebnis dem Benutzer präsentiert. Falls für ein Bauteil noch kein Wissen durch die Selbstlernkomponente akquiriert wurde, beispielsweise aufgrund fehlender Daten aus der Prozessentwicklung, so bricht die DfM-Analyse vorzeitig ab und fordert den Nutzer zur Bereitstellung entsprechender Da-

ten auf. Im Folgenden werden die Wissensverarbeitung und die Ergebnisdarstellung näher beschrieben.

6.5.1 Verarbeitung des Konstruktionswissens

In der Wissensbasis wird das konstruktionsrelevante Fertigungswissen in Form der ermittelten Metamodelle hinterlegt. Als Metamodelltypen wurden die lineare und die polynomiale Regression sowie der Weka M5P Modellbaum und der Weka M5R Regelgenerator gewählt (s. Abschnitt 6.3.1). Die rechnerinterne Repräsentation der Modelle erfolgt textbasiert wie in Bild 52 (S. 99) gezeigt.

Die textbasierte Repräsentation, in der Informatik als *String* (engl. *Zeichenkette*) bezeichnet, ist Ausgangspunkt für die Wissensverarbeitung des Assistenzsystems. Bisherige Arbeiten greifen auf den Inferenzmechanismus eines kommerziellen Expertensystemtools zurück (z. B. [War01; Bac97]) oder nutzen externe Simulationssoftware (z. B. [Sch00]). Für das selbstlernende Assistenzsystem wird ein Interpreter genutzt, der die Zeichenfolgen der textbasierten Modellrepräsentation einliest, analysiert und auswertet. Die Information, welche Zielgrößen (z. B. Umformkraft, Kontaktverhältnis, s. Abschnitt 3.2) einer Bauteilvariante zugeordnet sind und ausgewertet werden müssen, entnimmt das Analysetool dem Produkt- und Prozessdatenmodell. Das Einlesen des Strings erfolgt direkt aus dem Produkt- und Prozessdatenmodell durch Ausführung entsprechender SQL-Befehle. Der bereitgestellte String muss zur Analyse einen sog. Parser durchlaufen, der durch die Methode der regulären Ausdrücke den String in auswertbare, mathematische Formeln überführt. Hier muss eine Fallunterscheidung durchgeführt werden, je nachdem welcher Metamodelltyp zu verarbeiten ist. Während die Strings der linearen und polynomiaen Regressionsgleichungen direkt ausgewertet werden können, erfordert die Struktur der Weka Modelle spezifische reguläre Ausdrücke.

Im Falle des Weka M5P Modellbaums wird der Metamodell-String zunächst nach den lokalen linearen Modellen für die Zielgröße durchsucht, denn diese geben die Attribute (z. B. Geometrieparameter) vor, deren Wert aus der aktuellen Konfiguration des CAD-Modells ausgelesen werden muss. Die hierarchische Struktur eines M5P Modellbaums ist in den sog. „Pipes“ (|-Zeichen) erkennbar. Diese Struktur wird durch entsprechende reguläre Ausdrücke in ein System aus *if-else-if* Schleifen überführt. Wie diese Schleifen durchlaufen werden, richtet sich nach den aktuellen Attributwerten im Synthesetool und gibt das gültige lokale Regressionsmodell vor. Dessen Auswertung liefert letztendlich den Wert für die Zielgröße (z. B. Umformkraft, Kontaktverhältnis). Bild 65 zeigt links das textbasierte Modell eines M5P Modellbaums sowie die regulären Ausdrücke.

Beim Weka M5R Regelgenerator wird ähnlich verfahren, nur dass keine Baumstruktur zu berücksichtigen ist. Die textuellen Regelausdrücke aus Rapidminer® werden in interpretierbare **if-then**-Regeln umgewandelt und die sich ergebenden Metamodelle ausgewertet bzw. die hinterlegten Werte der Zielgröße ausgegeben (s. Bild 66).

Textbasiertes Modell:

```
WMSPruned model tree:
(using smoothed linear models)

a2 <= 4.737 :
| a2 <= 0.963 :
| | a2 <= 0.669 : LM1 (15/3.042%)
| | a2 > 0.669 : LM2 (12/2.574%)
| a2 > 0.963 :
| | a3 <= 3.992 : LM3 (26/3.313%)
| | a3 > 3.992 : LM4 (51/13.677%)
a2 > 4.737 :
| a1 <= 4.896 :
| | a1 <= 1.253 :
| | | a2 <= 6.428 : LM5 (5/2.882%)
| | | a2 > 6.428 : LM6 (7/4.948%)
| | a1 > 1.253 :
| | | a3 <= 6.863 : LM7 (27/6.364%)
| | | a3 > 6.863 : LM8 (13/12.692%)
| a1 > 4.896 :
| | a3 <= 5.826 : LM9 (20/11.667%)
| | a3 > 5.826 : LM10 (24/11.292%)

LM num: 1
label =
9.051 * a1
+ 23.6323 * a2
+ 7.0861 * a3
+ 0.4671 * a4
- 0.5273 * a5
- 83.2258
```

[...]

Bild 65: Verarbeitung des Weka M5P Modellsbaums durch reguläre Ausdrücke.

Textbasiertes Modell:

```
Rule: [1]
IF
| X_T0_L0 > 2.75
| X_T0_R0 <= 0.45
THEN
| Zielfunktion = + 2.617 [50/66.193%]

Rule: [2]
IF
| X_T0_L0 > 2.75
| X_T0_R0 <= 0.45
THEN
| Zielfunktion = + 2.4476 [27/74.978%]

Rule: [3]
IF
| X_T0_L0 < 2.75
THEN
| Zielfunktion = + 2.377 [24/88.231%]

[...]
Rule: [9]
Zielfunktion = + 4.377 [24/88.231%]
```

Bild 66: Verarbeitung eines Weka M5R Regellerners durch reguläre Ausdrücke.

Reguläre Ausdrücke:

```
LM num: ([0-9]*?)\n.+? = (.*)\n\n|
```

Identifikation der lokalen Regressionsgleichungen am Ende des Baums

```
\|(.*)\|
```

Identifikation des Entscheidungsbaums ohne Header und Regressionsgleichungen

```
^([^\|].*?) : (.*)\n|([^\|].*?) : (.*)\|
```

Iteratives Durchlaufen der Äste (if-else Schleifen), die durch das „Pipe“-Symbol | gekennzeichnet sind

```
\|[\s]+(.*)$|
```

„Trimmen“ der Äste bzw. aufteilen in einzelne Zeilen

```
^LM([0-9]+).*?$
```

Identifikation der Modellbezeichnung in den Blättern des Baums

Reguläre Ausdrücke:

```
^Rule: (.*)
```

```
(?:IF(.*)THEN.*?|[:whitespace:]*)
```

```
Zielfunktion = (.*)\|(.*)$
```

Erzeugte Text-Strings:

```
(1, 'X_T0_L0 > 2.75\n\tX_T0_R0 <= 0.45',
+ 2.617');
```

```
(2, 'X_T0_L0 > 2.75\n\tX_T0_R0 <= 0.45',
+ 2.4476');
```

```
(3, 'X_T0_L0 < 2.75\n', '+ 2.377');
```

```
[...]
```

```
(9, ' ', '+ 4.377')
```

\n wird mit and ersetzt und if-Abfrage erzeugt:

```
if eval('X_T0_L0 > 2.75 and X_T0_R0 <= 0.45'):
    then return +2.617
```

6.5.2 Ergebnisdarstellung und Erklärungskomponente

Die Werte der Zielgrößen (z. B. Umformkraft, Kontaktverhältnis), die durch den Inferenzmechanismus vorhergesagt werden, sind dem Benutzer übersichtlich und nachvollziehbar zu präsentieren. Hierzu erzeugt Slassy eine Tabelle (s. Bild 67). Sie enthält die für alle Zielgrößen berechneten Vorhersagen, den im ROPE-Prozess ermittelten Performanzkennwert sowie das Datum der Ausführung des Selbstlernprozesses. Auf Basis der vorhergesagten Werte der Zielgrößen kann anschließend abgeschätzt werden, ob der Bauteilentwurf fertigungsgerecht ausgeführt ist. Von einem nicht-fertigungsgerechten Entwurf muss ausgegangen werden, wenn mindestens eine der Zielgrößen außerhalb ihrer zulässigen Grenzen liegt. Diese können in die Wissensbasis aufgenommen und die vorhergesagten Werte damit verglichen werden. In sehr frühen Stadien der Prozessentwicklung ist eine exakte Angabe von Grenzwerten jedoch nicht immer möglich oder nicht zielführend. Vielmehr sollten die Ausprägungen der Zielgrößen mit dem Fertigungsexperten diskutiert und verschiedene Entwurfsalternativen analysiert werden. Der Wissens- und Informationsaustausch zwischen Konstruktion und Fertigung wird somit unterstützt.

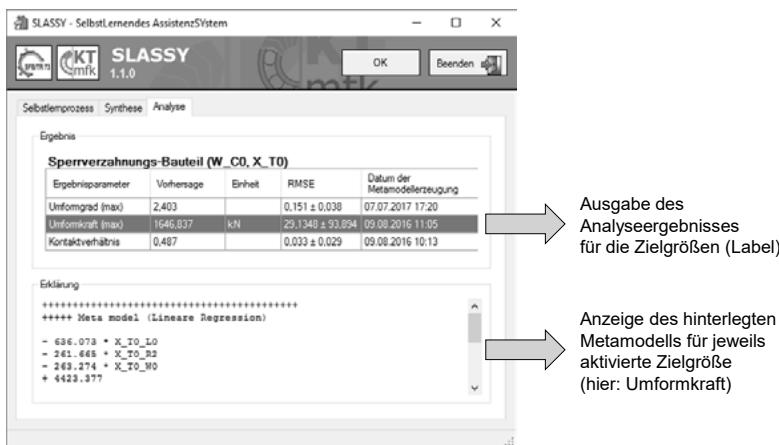


Bild 67: In der grafischen Benutzeroberfläche des Analysetools ist die Erklärungskomponente integriert.

Ein Vorteil der ausgewählten Metamodelle ist, dass sie auch ohne Hilfsmittel gut durch den Benutzer interpretiert werden können. Expertenwissen ist nach RUDE [Rud98] u. a. in Form von Constraints und Strukturen aus *Wenn-Dann*-Regeln repräsentiert. In diesen Formen lässt sich Wissen nach NORTH [Nor11] sowohl gut externalisieren (Überführung von implizitem zu explizitem Wissen) als auch internalisieren (Überführung von explizitem zu implizitem Wissen) (s. Abschnitt 2.1.2, S. 9). Diesen Vorteil nutzt das Assistenzsystem. Slassy bildet für jede Zielgröße das entsprechende Metamodell in textbasierter Form ab (s. Bild 67 unten).

6.6 Das Produkt- und Prozessdatenmodell

Als zentrale Komponente einer informations- und wissensverarbeitender Software muss dem Datenmodell während der Entwicklung hohe Aufmerksamkeit beigemessen werden. Das Datenmodell von Konstruktionsassistsystemen wird i. A. als *Produktmodell* bezeichnet. Es bildet alle relevanten Informationen eines Produkts ab. Diese Informationen beschreiben zum einen das Produkt als hierarchisches, technisches System mit allen Komponenten, Unterbaugruppen und Einzelteilen. Neben dieser Produktstruktur werden auch alle Charakteristiken berücksichtigt, die das Produkt aufgrund seines Lebenszyklus auszeichnen. Zum Produktlebenszyklus werden alle Phasen von der Entwicklung und Produktion über den Vertrieb bis hin zur Wartung und zum Recycling gezählt [SK97]. In dieser Hinsicht grenzen sich Produktmodelle von reinen geometrischen Modellen ab. Zum Stand der Technik gehören, neben den proprietären Formaten der CAD-Systeme, unterschiedliche neutrale und genormte Datenmodelle im Umfeld der Produktentwicklung. STEP (Standard for the exchange of product model data, s. [GAPW93; And00]) und JT (Jupiter Tesselation, s. [ISO12]) sind die aktuell am häufigsten genutzten Modellkonzepte. Erfolgreiche Arbeiten zur Entwicklung integrativer Produkt- und Prozessdatenmodelle finden sich ebenfalls in der Literatur, wobei die Modellkonzepte auf den Problemstellungen der jeweiligen Forschungsverbünde basieren. Bei [ES05] wird das PPDM in Module (Schemata) strukturiert, die aus logisch gruppierten Informationsentitäten aufgebaut sind. In den Arbeiten von [War01] stehen DfX-Analysen auf der Basis von heuristischem Erfahrungswissen im Vordergrund, die erst durch das Produkt- und Prozessdatenmodell realisierbar sind. Ein Übertragung verfügbaren Ansätze auf die Problemstellung der vorliegenden Arbeit ist nur in begrenztem Maße möglich. Da es sich bei der Blechmassivumformung um eine neue Gruppe von teils kombinierten Fertigungsprozessen handelt, steht kein Standard zur Gliederung eines Produktdatenmodells für Blechbauteile mit komplexen Nebenformelementen zur Verfügung. Zudem stellen Verfahrenskombinationen innerhalb der Blechmassivumformung und vor allem der KDD-basierte Selbstlernprozess des Assistenzsystems spezifische Anforderungen an die Datenmodellierung. Das entwickelte Produkt- und Prozessdatenmodell berücksichtigt die für den KDD-Prozess notwendigen Daten und Informationen zum einen in Form von einzelnen Entitäten, die in Klassen zusammengefasst sind. Neben dieser Grundstruktur wurde aber auch der multidimensionale Charakter der Produkt- und Prozessdaten der Blechmassivumformung erkannt und entsprechend modelliert. Aus dieser multidimensionalen Datenbasis geht, durch Anwendung des Selbstlernprozesses, die multidimensionale Wissensbasis hervor. Beides wird im folgenden Abschnitt beschrieben.

6.6.1 Die multidimensionale Daten- und Wissensbasis

Der multidimensionale Charakter der Daten, die während der Entwicklung der Blechmassivumformung anfallen, wird durch die drei Aspekte Produkt, Prozess und Zielgröße beschrieben (s. Bild 68).

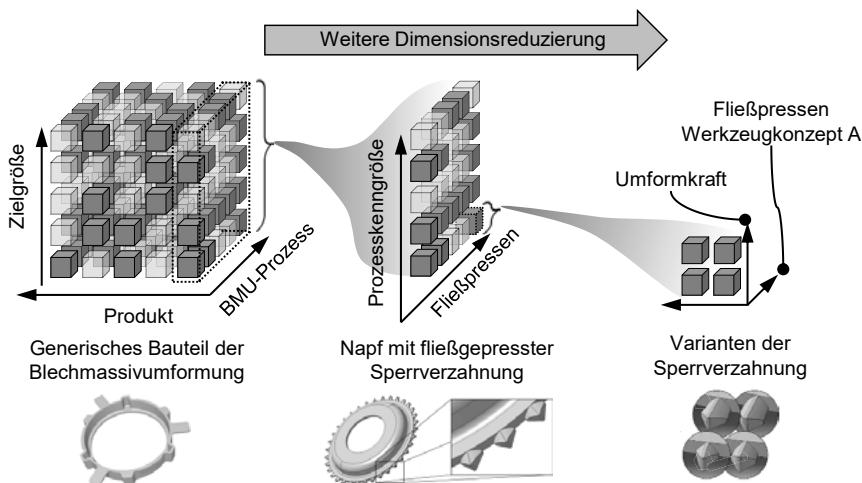


Bild 68: Prinzip der multidimensionalen Datenbasis für Slassy

Produkt bezeichnet die Ausprägung eines blechmassivumgeformten Bauteils. Allgemein betrachtet (Bild 68, links) ist hierunter ein generisches Bauteil zu verstehen, das mittels Blechmassivumformung zu fertigen ist. Dies wird durch den Demonstrator des SFB/TR 73 symbolisiert. Mit fortschreitender Dimensionsreduzierung wird diese generische Darstellung konkreter. Ein Napf mit fließgepresster Sperrverzahnung ist hierfür das Beispiel (Bild 68, mitte). Für die Sperrverzahnung dieses Napfes existieren wiederum verschiedene geometrische Varianten (Bild 68, rechts). *Prozess* beschreibt die Ausprägung der zur Verfügung stehenden Blechmassivumformprozesse. Bei Betrachtung eines konkreten blechmassivumgeformten Bauteils (z. B. Napf mit Sperrverzahnung) lässt sich bereits ein bestimmtes Blechmassivumformverfahren ableiten, z. B. das Fließpressen. Für dieses Verfahren stehen wiederum verschiedene Werkzeugkonzepte zur Verfügung. Eine Unterscheidung der Werkzeugkonzepte ergibt sich beispielsweise durch verschiedene Behandlungen der Werkzeugoberfläche (z. B. Beschichten, Nitrieren, Strukturieren). Die mit einem Werkzeugkonzept gefertigten Varianten einer Sperrverzahnung sind im Bild 68 rechts dargestellt. Die *Zielgröße* bezeichnet die Bauteil- oder Prozesskenngröße, zu welcher der Benutzer von Slassy konstruktionsrelevantes Wissen benötigt. Erst für ein konkretes Produkt (z. B. Napf mit Sperrverzahnung) kann festgelegt werden, welche Prozess- oder Bauteilkenngrößen zur Beurteilung der Fertigungsgerechtigkeit relevant sind. Wie im Bild 68 rechts zu erkennen, wird die Fertigungsgerechtigkeit der Sperrverzahnungsvarianten u. a. anhand der auftretenden Umformkräfte beurteilt.

Weiterhin muss auch die Heterogenität der Fertigungsdaten eines Blechmassivumformprozesses als wichtiges Kriterium bei der Modellierung berücksichtigt werden [FKKS08]. Dies lässt sich innerhalb der Forschungsaktivitäten des SFB/TR 73 sehr gut beobachten. Die

verschiedenen, sowohl systemabhängigen als auch -neutralen Daten, die in den einzelnen Teilprojekten erzeugt werden, müssen in einem KDD-Prozess verarbeitbar sein. Der multidimensionalen Charakter der in Bild 68 dargestellten Datenbasis wird im Rahmen dieser Arbeit anschaulich durch den sogenannten OLAP-Würfel (engl. Online Analytical Processing) repräsentiert (s. Bild 69).

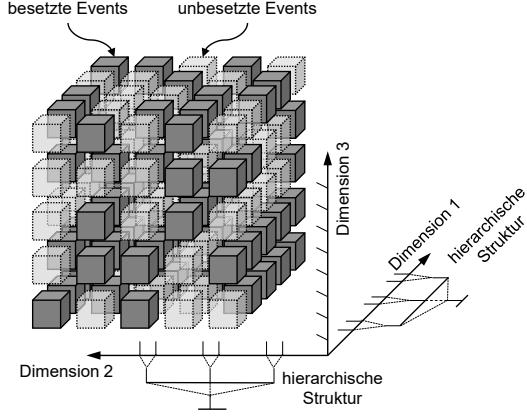


Bild 69: Der OLAP Würfel als Beispiel für multidimensionale Datenbasen, nach [CD97].

Das OLAP-Konzept fasst im ursprünglichen Sinne computerunterstützte Methoden und Werkzeuge zur analytischen Auswertung von multidimensionalen Daten für die Unternehmensführung zusammen (z. B. Umsatzerlöse, Verkaufszahlen, Inventurergebnisse) [Tot00]. Die Bezeichnung *Online* hebt die schnelle Bearbeitung der Anfragen des Benutzers hervor. Wie in Bild 69 zu sehen, kann ein OLAP-Würfel in jeder Dimensionen hierarchisch gegliedert werden, wodurch sogenannte *Events* entstehen [WK07]. Ein Event kann dabei der Verkauf eines bestimmten Artikels im Quartal eines Geschäftsjahres in der Vertriebsregion Süd darstellen. Die beschreibenden Dimensionen (Attribute) sind in diesem Fall das Produkt, die Zeit und der Vertriebsweg. Im allgemeinen ist ein OLAP-Würfel „dünn besetzt“, d.h. es sind nicht alle Events im Würfel mit Einträgen belegt. Der Umgang einer Software mit den besetzten Events des Würfels trägt entscheidend zum Speicherbedarf und zur Performance des OLAP-Systems bei [CD97].

6.6.2 Strukturierung der Produkt- und Prozessdaten

Nachfolgend werden der Aufbau und die Struktur der Datenbank für die Produkt- und Prozessdaten beschrieben. Als grundlegendes Konzept für die Umsetzung wird ein relationales Datenbankschema verwendet, das Flexibilität bei Strukturergänzungen und -änderungen bietet. Ein weiterer Vorteil ist die Möglichkeit, beliebig viele Sichtweisen auf die Produkt- und Prozessdaten mit unterschiedlichen logischen Strukturen einzunehmen, indem ver-

schiedene Entitätstabellen oder Teile davon kombiniert werden. Dies ermöglicht in der Zukunft die Entwicklung einer Version von Slassy, die auf die Bedürfnisse des Prozessentwicklers angepasst werden kann, ohne eine physische Änderung der bereits vorhandenen Produkt- und Prozessdatenbank vornehmen zu müssen.

Bei der Erstellung des relationalen Datenbankschemas werden insbesondere die ersten drei Normalformen (s. [KR05; Kle11]) beachtet, mit dem Ziel, Redundanzen in der Datenspeicherung zu beseitigen, funktionelle und transitive Abhängigkeiten zu vermeiden und ein klar strukturiertes Datenbankmodell zu erhalten. Im Fokus der Beschreibungen stehen die Erfassung der Umformversuche und der Blechmassivumformverfahren mit den jeweiligen Werk- und Halbzeugkonzepten sowie die Strukturierung der Ergebnisgrößen des Selbstlernprozesses, der Metamodelle und der Bauteile mit den verschiedenen Formelementen. Bild 70 zeigt die schematische Darstellung der Produkt- und Prozessdaten mit den jeweiligen Relationen. Zu den Elementen *Beschreibende Parameter* und *Werte der beschreibenden Parameter je Versuch* ist anzumerken, dass diese der Übersicht halber zusammengefasst wurden. In der physischen Ausführung der Datenbank werden je Aspekt (Zusatzeigenschaft, Formelement, Fertigungsverfahren, Werkzeugkonzept und Halbzeugkonzept) eigene entsprechende Tabellen angelegt.

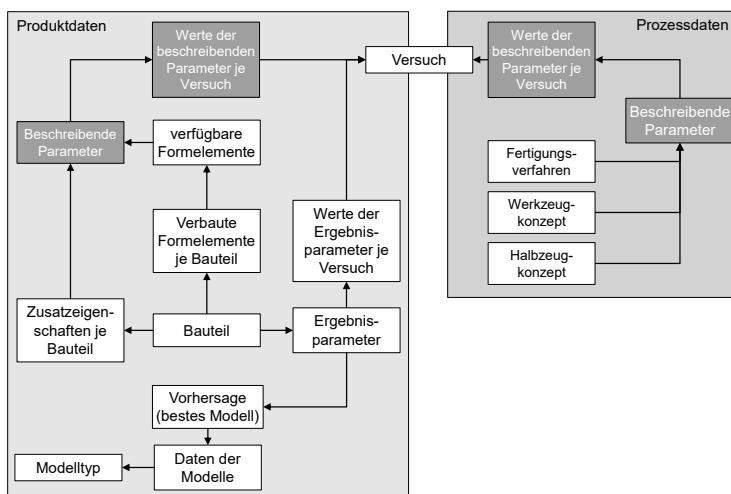


Bild 70: Schematische Darstellung der Verbindungen zwischen Produkt- und Prozessdaten.

Unter der Bezeichnung „Versuch“ werden sowohl Simulationen als auch Experimente mit Verfahren der Blechmassivumformung zusammengefasst. Um den Umformversuch bestmöglich zu beschreiben, werden alle Parameterwerte eines Versuchs (z. B. Geometrieparameter, Prozessparameter, Ergebnisgrößen) einer eindeutigen (fortlaufenden) Versuchsnr. (Versuchs-ID) zugeordnet. Dazu wird für jeden Versuchsdurchgang in der Entität

„Versuch“ ein Eintrag erstellt. Hier kann eine Beschreibung zum jeweiligen Versuch hinzugefügt werden, beispielsweise in welchem Kontext die Daten erzeugt oder welche Methoden zur Versuchsplanung genutzt wurden. So kann von jedem Wertetupel per Fremdschlüssel auf die Versuchs-ID verwiesen werden.

Die Datenstrukturen von Werk- und Halbzeugkonzepten sowie die der Blechmassivumformverfahren sind ähnlich aufgebaut und werden exemplarisch an der Struktur für die Fertigungsverfahren erläutert, wie sie in Bild 71 gezeigt ist. Zur Integration eines neuen Blechmassivumformverfahrens wird die Tabelle der Entität „Fertigungsverfahren“ um einen Eintrag mit der entsprechenden Bezeichnung, bspw. „Tiefziehen“, erweitert. Das Datenbanktool fügt der Bezeichnung automatisch einen eindeutigen Primärschlüssel zur Identifikation zu. Attribute, die den jeweiligen Fertigungsprozess auszeichnen, jedoch keine Ergebnisgrößen sind (z. B. geschätzte Reibzahl zwischen Werkzeug und Werkstück, Umformgeschwindigkeit, Maschinenummer), werden in der Tabelle „FertigungPara“ hinterlegt und das entsprechende Fertigungsverfahren per Fremdschlüssel „Fertigungsverfahren_ID“ referenziert. Die Integration der Daten bzw. des Datentupels eines Versuchs führt zur Belegung der Entität „FertParaWerte“ mit dem jeweiligen Wert. Die Relation zum Versuch wird wiederum durch einen Fremdschlüssel („Versuch_ID“) hergestellt. In ähnlicher Weise wird für die Struktur der Halb- und Werkzeugkonzepte verfahren.

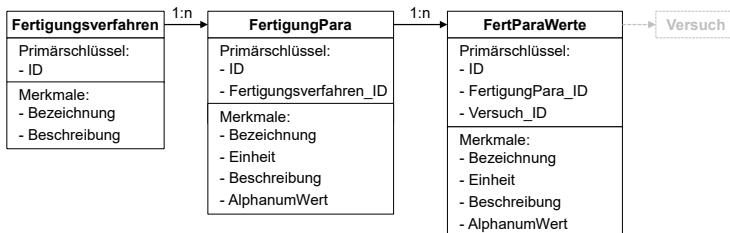


Bild 71: Darstellung der Datenbankstruktur zur Vorhaltung von Daten über Blechmassivumformverfahren

Die unterschiedlichen Varianten an blechmassivumgeformten Bauteilen (s. Bild 31, S. 56) werden in der Tabelle „Bauteil“ geführt. Neben der Bauteilbezeichnung enthält diese zur Verwaltung von Änderungen am Bauteil bei Bedarf auch zu vergebene Versionsnummern sowie einen kurzen Beschreibungstext. Bei der Abbildung der Bauteilstruktur wird das Konzept der Haupt- und Nebenformelemente (s. Abschnitt 3.3, S. 61) berücksichtigt. Entsprechend besteht ein Bauteil aus maximal einem Haupt- und mindestens einem Nebenformelement. Die Definition eines Formelementes erfolgt über die Tabelle „Formelemente“ (s. Bild 72). Sie enthält die Bezeichnung (z. B. „tiefgezogener Napf“, „Scheibe“) sowie ein Formelementkürzel, das bspw. als Präfix für die Geometrieparameter genutzt werden kann (z. B. „C“ für Napf, engl. Cup). Das Attribut Typ_HFE_NFE enthält die Angabe, ob es sich bei dem Formelement um ein Haupt- (Typ_HFE_NFE = HFE) oder ein Nebenformelement (Typ_HFE_NFE = NFE) handelt. Jedes Formelement wird im CAD-Modell

durch unterschiedliche Geometrieparameter (z. B. Längen, Breiten, Winkel) beschrieben, die im Produkt- und Prozessdatenmodell in der Tabelle „FormelementePara“ hinterlegt sind. Neben der Angabe einer Parameter-Bezeichnung sowie der Einheit kann zusätzlich ein Beschreibungstext angegeben werden. Um ein Formelement auch über alphanumerische Parameter beschreiben zu können, wird ein binäres Attribut „AlphanumWert“ definiert. Im Falle eines entsprechenden Parameters (z. B. Blechmaterial DC04 oder DP600) muss „AlphanumWert“ an dieser Stelle zu „true“ bzw. „1“ gesetzt werden. Da ein Formelement verschiedenen Bauteilvarianten zugeordnet sein kann (z. B. „Napf mit Sperrverzahnung“ oder „Scheibe mit Sperrverzahnung“) wird eine sogenannte Hilfstabelle benötigt. Diese Tabelle „Bauteil_has_Formelemente“ beinhaltet während des Betriebs von Slassy die Formelementekombinationen für ein bestimmtes Bauteil. Werden nun Versuchsstudien mit verschiedenen Werten der Geometrieparameter durchgeführt, erfolgt eine Speicherung dieser Versuchsdaten in der Tabelle „FormParaWerte“. Per Fremdschlüssel werden die zugehörigen FormelementePara-IDs und die Versuchs-IDs referenziert. Sofern ein numerischer Ergebniswert vorliegt, wird dieser in „wert_numerisch“ eingetragen, andernfalls in „wert_alphanumerisch“. Die Tabelle „FormParaWerte“ enthält nach der Integration eines Fertigungsdatensatzes die Ausprägungen der einzelnen Parameter (Eingangsgrößen, z. B. Geometrie) und spiegelt somit den zugrundeliegenden Versuchsplänen (s. Abschnitt 2.3.3, S. 39) wider. Ähnlich zu den Formelementen eines Bauteils wird mit dessen möglichen Zusatzeigenschaften verfahren. Zusatzeigenschaften sind bspw. funktionsrelevante Attribute wie die auftretende Flächenpressung an der Funktionsfläche der Sperrverzahnung (s. Abschnitt 3.3, Bild 21 sowie [BW14]). Somit kann eine Verknüpfung zwischen Produkt- und Prozessanforderungen hergestellt werden.

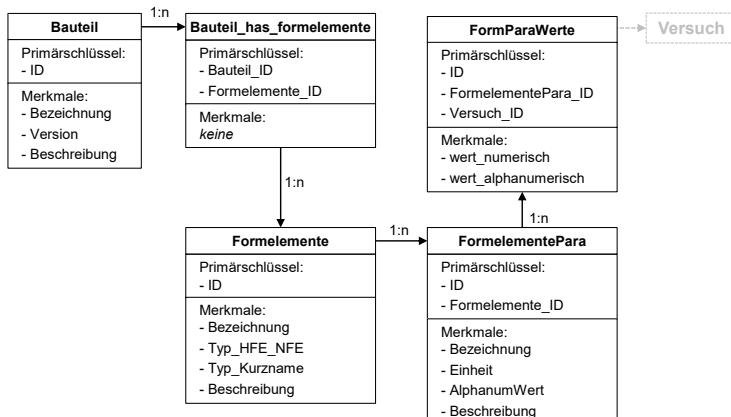


Bild 72: Datenbankstruktur zur Datenverwaltung der Formelemente.

Jedem blechmassivumgeformten Bauteil sind verschiedene Ergebnisparameter zugewiesen (s. Abschnitt 3.2). Dies in der Datenbank des PPDMs abzubilden ist Zweck der Tabel-

le „ErgebPara“ (s. Bild 73). Per Fremdschlüssel wird von dieser Entität auf die jeweilige Bauteil-ID referenziert. Bei der Definition einer Ergebnisgröße (s. Abschnitt 6.6.3) muss der Benutzer die Parameterbezeichnung sowie eine Einheit angeben (z. B. Umformkraft in kN). Die Ergebniswerte aus den Umformversuchen einer Parameterstudie werden per Fremdschlüssel auf die zugehörige Versuchs-, Bauteil- und Ergebnisparameter-ID referenziert.

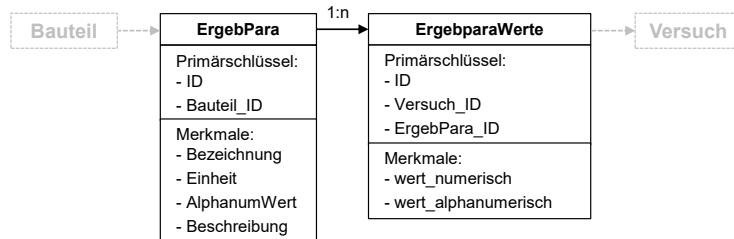


Bild 73: Darstellung der Datenbankstruktur für die Ergebniswerte und deren Verbindung zu Bauteil und Versuch.

In der Tabelle „ModellTyp“ (s. Bild 74) werden die im Abschnitt 6.3.1 ausgewählten Modelle „Lineare Regression“ (ID=1), „Polynomiale Regression“ (ID=2), „Weka M5P Modellbaum“ (ID=3) sowie „Weka M5R Regellerner“ (ID=4) hinterlegt. Die Beibehaltung der IDs ist für die Ausführung des Assistenzsystems erforderlich und bei neu hinzuzufügenden Modelltypen muss die Liste entsprechend fortlaufend nummeriert sein. Hat die Selbstlernkomponente das bestgeeignete Metamodell identifiziert, so wird in der Entität „Vorhersage“ die Referenz zur Ergebnisgröße („ErgebPara_ID“) eines bestimmten Bauteils festgelegt. Der Modelltyp gibt die spätere Auswertung des Modells durch das Analysetool vor und muss daher ebenfalls abgelegt werden („ModellTyp_ID“). Das KDD-Tool Rapidminer® ermöglicht den Export textbasierter Modellrepräsentationen in Form eines Strings. Das Attribut „RM_Bestes_Modell“ beinhaltet diesen String, dessen Verarbeitung im Abschnitt 6.5 beschrieben wurde. Neben dem Modellstring enthält die Entität mit dem Attribut „RM_Perf_Bestes_Modell“ auch die durch den ROPE-Prozess (s. Abschnitt 5.2) geschätzte Performanz.

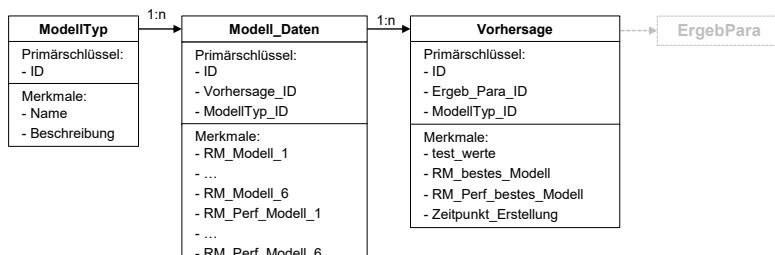


Bild 74: Datenbankstruktur für Vorhersagemodelle.

6.6.3 Operative Integration der Produkt- und Prozessdaten

Die Durchführung der Datenerhebung und die Vorbereitung der Wissensakquisition durch den Benutzer erfordert eine komfortable und einfach zu bedienende Benutzeroberfläche. Diese ist als Teil der Akquisitionskomponente des selbstlernenden Assistenzsystems zu betrachten. Das Vorgehen zur Integration der Produkt- und Prozessdaten ist in Bild 75 beschrieben.

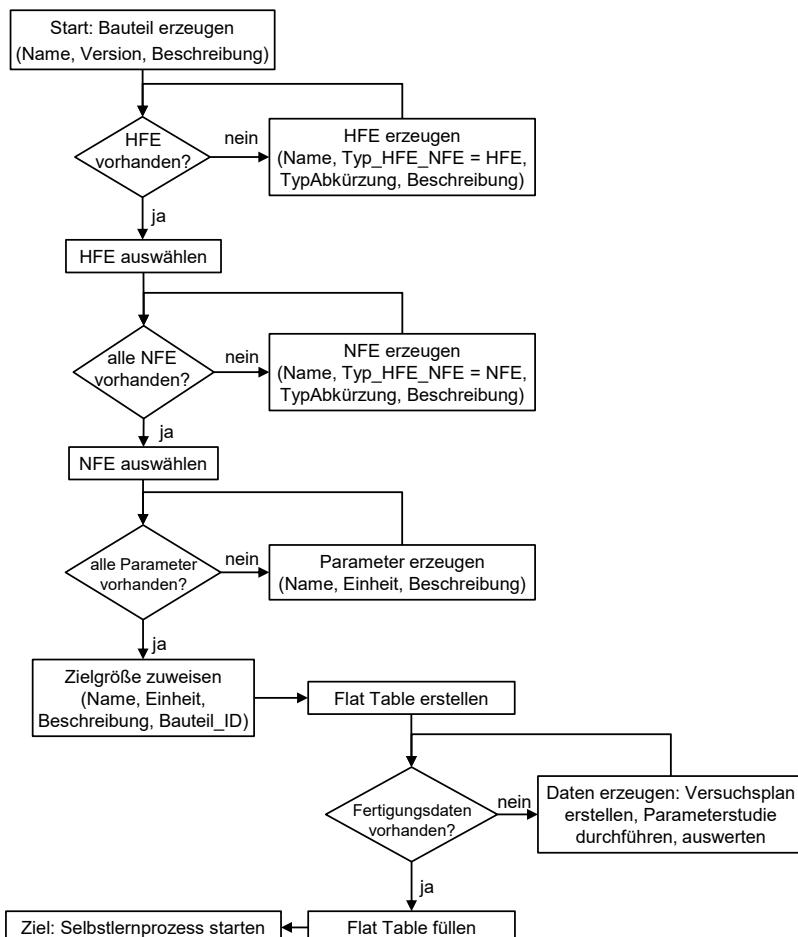


Bild 75: Ablaufschema zur Integration von Produkt- und Prozessdaten.

Zu Beginn erfolgt die Definition eines blechmassivumgeformten Bauteils für das eine Versionsnummer vergeben und eine kurze Beschreibung verfasst wird. Basis eines Bauteilmodells ist das Hauptformelement. Hier kann der Benutzer aus bestehenden Modellen (z. B. Näpfe, Scheiben etc.) wählen oder ein Hauptformelement erzeugen, das noch nicht in der Datenbank vorhanden ist. Diesem Hauptformelement werden anschließend die Nebenformelemente zugeordnet. Auch hier besteht die Möglichkeit, existierende Nebenformelemente zu wählen oder neue zu erzeugen. In den nächsten beiden Schritten müssen die veränderbaren Parameter (z. B. Längen und Winkel eines Formelements) sowie die relevanten DfM-Ergebnisgrößen (z. B. Umformkraft, Kontaktverhältnis) definiert werden. Hierbei sind die Benennungen im späteren synthetisierten CAD-Modell zu berücksichtigen. Zur Erhöhung des Benutzerkomforts wurde eine Supportfunktion implementiert, die das aktuell im Syntheseteil erzeugte Bauteilmodell analysiert und die vorhandenen Geometrieparameter aus den UserFeatures des Haupt- und der Nebenformelemente automatisch in die Importoberfläche übernimmt. Bild 76 zeigt die Umsetzung dieses Schemas in der graphischen Benutzeroberfläche von Slassy.

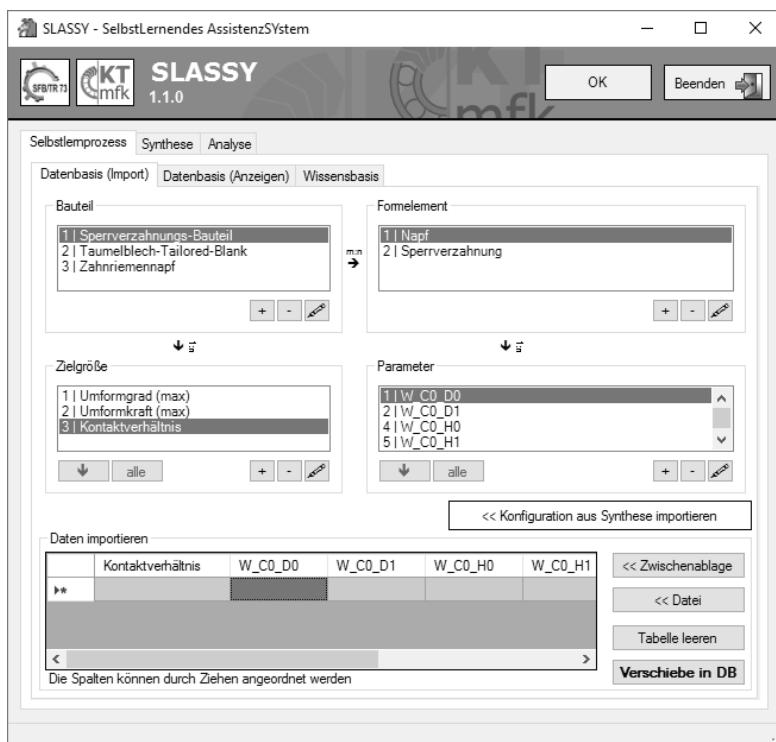


Bild 76: Benutzeroberfläche zur Integration von Produkt- und Prozessdaten.

Aus den Listen für Zielgröße und Parameter wird im unteren Bereich der Benutzeroberfläche eine zu Beginn noch leere Datentabelle erzeugt. Das Befüllen dieser Tabelle mit den Versuchsdaten kann manuell oder durch Import einer vorliegenden Excel® Tabelle erfolgen. Die Benennung der Spalten einer solchen Versuchsdatentabelle muss mit der im Datenakquisitionstool übereinstimmen. Die Übernahme der Eingaben durch das PPDM wird von einer kurzen Versuchsbeschreibung begleitet, die der Benutzer tätigen muss, um den Kontext der Versuchsdaten zu erfassen (z. B. „Versuchsreihe Bauteil A, Juni 2016). Die somit gewährleistete kontextabhängige Datenüberprüfung kann vom Benutzer in einer eigenen Ansicht des Assistenzsystems durchgeführt werden. Zum Starten des Selbstlernprozesses wechselt der Benutzer in die entsprechende Ansicht und wählt den Datensatz, aus dem das konstruktionsrelevante Fertigungswissen automatisch erhoben werden soll. Da die Laufzeit eines KDD-Prozesses im Allgemeinen mit dem Umfang der Daten steigt, kann der Benutzer hier bereits eine Vorauswahl der zu verarbeitenden Attribute und Label treffen. Nach dem Start und dem erfolgreichen Durchlaufen des Selbstlernprozesses, speichert das Assistenzsystem das ermittelte bestgeeignete Metamodell in der multidimensionalen Wissensbasis. Der Benutzer erhält einen kurzen Ergebnisbericht mit Informationen zum ermittelten Metamodell und dessen Performanz.

7 Anwendungsbeispiel: Tiefziehen-Querfließpressen

Am Beispiel eines blechmassivumgeformten Bauteils werden im Folgenden die Funktionsweise des Assistenzsystems und der Umgang mit der grafischen Benutzeroberfläche dargestellt. Im Fokus stehen die Aufnahme neuer Formelemente in Slassy, die Erhebung und Integration der Fertigungsdaten und das Durchlaufen des Selbstlernprozesses. Obwohl der Selbstlernprozess während der Laufzeit im Hintergrund aktiv ist und nur die wichtigsten Informationen an den Benutzer ausgibt, werden die einzelnen Prozessschritte in diesem Kapitel nachvollziehbar beschrieben.

7.1 Integration des Bauteils in Slassy

Als blechmassivumgeformtes Bauteil wird ein Demonstrator aus einem der Teilprojekte des Sonderforschungsbereichs Transregio 73 gewählt (s. Bild 77). Die Form des Bauteils orientiert sich an Synchronringen in Fahrzeuggetrieben, die u. a. die Funktion der Drehzahlanpassung von Antriebs- und Abtriebswelle erfüllen. Das gezeigte Bauteil wird in einem zweistufigen Fertigungsprozess zunächst tiefgezogen und anschließend werden durch Querfließpressen Verzahnungen am Umfang des Flanschs ausgeformt.

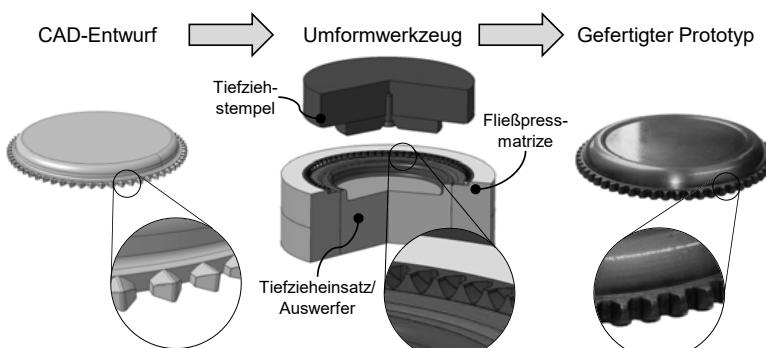


Bild 77: Bauteil des Anwendungsbeispiels aus dem SFB/TR 73, z. B. in [VKME12; MKOS11]).

7.1.1 Aufbau des Hauptformelementes *Napf*

Als Hauptformelement kann bei diesem Bauteil ein tiefgezogener Napf mit Flansch identifiziert werden. Bild 78 zeigt das Bauteil bzw. das Hauptformelement mit Bemaßungen, wobei die Geometrieparameter des Hauptformelementes in der Bezeichnung das Präfix *W_C0* erhalten (*W*: Hauptformelement, *C0*: Napf Form 0, von engl. *cup*), wie in Tabelle 9 (S. 115) bereits gezeigt. Die angegebenen Maße sind zu Anschauungszwecken gewählt, entsprechen in der Größenordnung jedoch den im SFB/TR 73 üblichen Werten. Der Flansch ist notwendig für den Tiefziehvorgang, da hier die Niederhalterkraft zur Vermeidung von Faltenbildung eingeleitet wird. Der Napfinnenradius *W_C0_R2* ergibt sich während der Umformung aus der Form des Tiefziehstempels, während der Napfaußenumfang *W_C0_R1* durch die Tiefziehmatrize geformt wird. Die Blechdicke *W_C0_H0* im Bereich der später ausgeformten Nebenformelemente wird wiederum durch das eingesetzte Blechhalzeug bestimmt. Wie in Abschnitt 6.4.1 beschrieben, verfügt jedes Hauptformelement über Referenzelemente (s. Bild 78), die zur Synthese mit den Nebenformelementen notwendig sind.

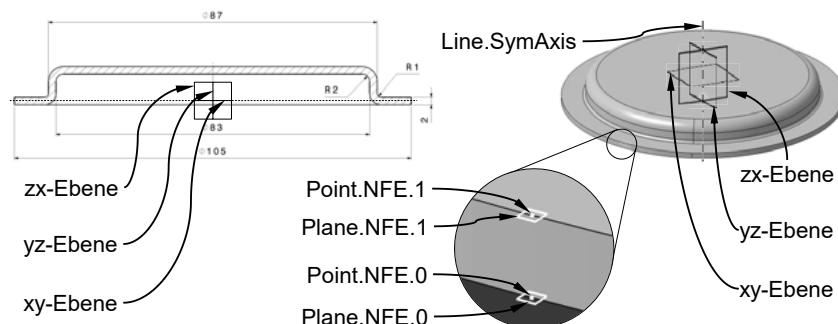


Bild 78: Geometrie und Referenzelemente des Hauptformelementes *tiefgezogenem Napf*.

Während der Laufzeit muss Slassy in der Lage sein, auf das Hauptformelement zuzugreifen, um Geometrieparameter durch den Benutzer ändern zu lassen und die definierten Referenzelemente für das Nebenformelement bereitzustellen. Hierzu erfolgt die Definition eines UserFeatures im CAD-System¹⁷ mit Ein- und Ausgabeelementen sowie veröffentlichten und von außen zugreifbaren Geometrieparametern. Das Hauptformelement benötigt als Eingabeelement die *xy*-Ebene zur korrekten Positionierung im Raum. Die Ausgabeelemente entsprechen den zuvor definierten Referenzelementen für das Nebenformelement. Neben den Geometrieparametern kann Slassy auf den Parameter „sync“ zugreifen. Dieser enthält die Namen der Geometrieparameter eines Nebenformelements, die mit Parametern des Hauptformelements synchronisiert, d. h. in ihren Werten gleich gesetzt werden können. Die Bereitstellung der CAD-Datei, einer Voransicht sowie einer technischen Skizze des Hauptformelements erfolgt über die Windows-Ordnerstruktur.

¹⁷ Details zu UserFeatures und deren Erstellung können [Bra08] entnommen werden.

7.1.2 Aufbau des Nebenformelementes *Sperrverzahnung*

Die Verzahnung als Nebenformelement wird nach dem Tiefziehen durch einen Querfließprozess radial am Umfang des Flanschs geformt. Tabelle 10 (S. 116) zeigt die Bezeichnungen der Geometrieparameter der Sperrverzahnung mit dem Präfix X_T0 (X: Nebenformelement, C0: Napf Form 0, von engl. *cup*)

Bild 79 zeigt die Geometrie des Nebenformelementes sowie die Bemaßungen, die mit dem Präfix X_T0 gekennzeichnet sind. Das UserFeature eines Nebenformelements muss intern drei Objekte aufweisen, die die drei Ausgabeelemente des UserFeatures des Hauptformelements als Referenz nutzen. An diesen Objekten richten sich alle weiteren Geometrien des Nebenformelementes aus. Im Falle der Sperrverzahnung handelt es sich dabei um die xy-Ebene des CAD-Modells, den Punkt „Point.Base“ sowie das Element „Line.SymAxis“ wie in Bild 79 gezeigt. Das dort gezeigte Nebenformelement ist als eine Art Startgeometrie zu verstehen. Diese erscheint beim ersten Aufruf des Formelements im Synthesetool. Der Benutzer passt anschließend die Werte der gezeigten Maße an.

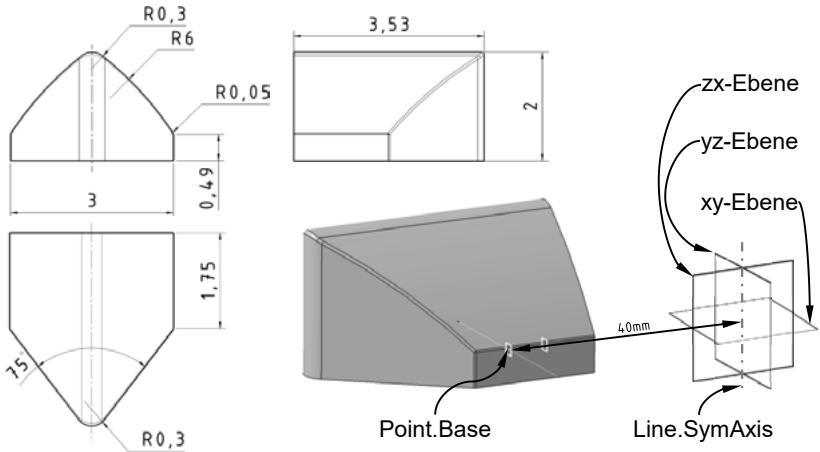


Bild 79: Geometrie und Referenzelemente des Nebenformelementes *Sperrverzahnung*.

Bei der Definition des UserFeatures eines Nebenformelements werden die oben genannten drei internen Objekte als Eingabeelemente festgelegt, die bei der Instanziierung des UserFeatures (s. Abschnitt 7.1.3) mit den Ausgabeelementen des UserFeatures des Hauptformelements verlinkt werden. Wie bereits beim tiefgezogenen Napf werden die Geometrieparameter der Sperrverzahnung publiziert, um sie für das Assistenzsystem und damit für den Benutzer zugreifbar zu machen. Für das Kreismuster eines Nebenformelements am Umfang des Hauptformelements kann der Benutzer die Anzahl der zu erzeugenden Kopien selbst bestimmen. Sie ist jedoch nach oben begrenzt, da zu eng beieinander liegende Features weder funktions- noch fertigungsgerecht sind. Daher enthält jedes UserFeatures den

publizierten Parameter „NFE_UsedAngle“, der dem Kreisbogenwinkel entspricht, den das Nebenformelement am Umfang des Hauptformelementes einnimmt. Während der Synthese prüft Slassy diesen Winkel und reduziert die Anzahl der Kreismusterelemente, falls es zu einer Überschreitung der Nebenformelemente kommt.

7.1.3 Synthese des Bauteilmodells

Bild 80 zeigt die graphische Benutzeroberfläche des Synthesewerkzeugs in Slassy. Im oberen Bereich wählt der Benutzer ein verfügbares Hauptformelement aus und passt über ein Kontextmenü dessen Geometrie an. Anschließend erfolgt die Auswahl und Geometrieanpassung eines oder mehrerer Nebenformelemente.

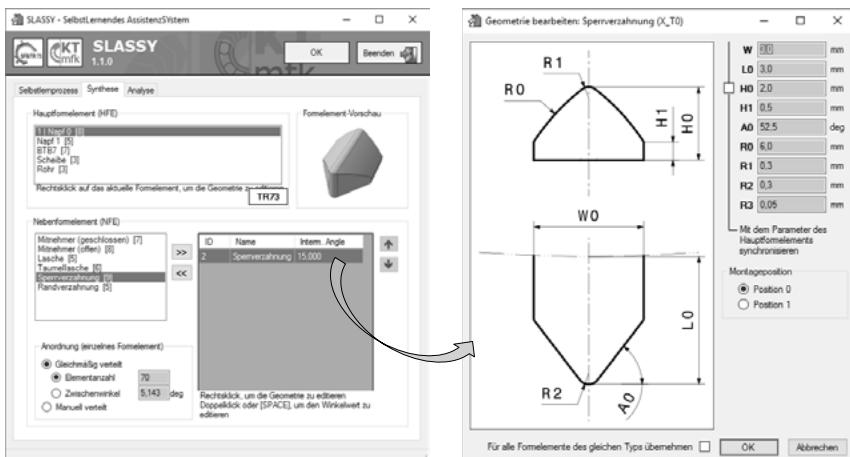


Bild 80: Graphische Benutzeroberfläche des Synthesewerkzeugs (links) mit Kontextmenü zur Geometrieanpassung eines Nebenformelements (rechts)

Die systeminterne Bauteilsynthese beginnt mit der Instanziierung eines UserFeatures für ein Hauptformelement in einem zunächst leeren CAD-Einzelteil mit den globalen Referenzebenen in XY-, YZ- und ZX-Richtung und einem leeren Hauptkörper. Hierzu erzeugt Slassy einen temporären Körper im CAD-Part, fügt das UserFeature ein und weist dem lokalen Eingabeelement „xy:plane“ des Features die globale XY-Referenzebene des CAD-Einzelteils zu. Anschließend wird der temporäre Körper durch eine Bool'sche Operation dem Hauptkörper hinzugefügt. Damit ist das Hauptformelement für den Nutzer im CAD-System sichtbar und kann von ihm geändert werden. Für das Nebenformelement wird ähnlich verfahren, indem ein weiterer temporärer Körper dem CAD-Einzelteil (mit einem Hauptformelement) hinzugefügt wird. In diesem wird das UserFeatures des Nebenformelements instantiiert und die Ausgabeelemente des Hauptformelements (s. oben) den Eingabeelementen des Nebenformelements zugewiesen. Durch eine weitere Bool'sche Operation

fügt Slassy auch den temporären Körper inklusive Nebenformelement dem Hauptkörper hinzu und erzeugt ein Kreismuster mit „Line.SymAxis“ (s. oben) als Drehachse. Nun kann der Benutzer die Gestalt des Nebenformelements anpassen. Bild 81 fasst den Ablauf zusammen.

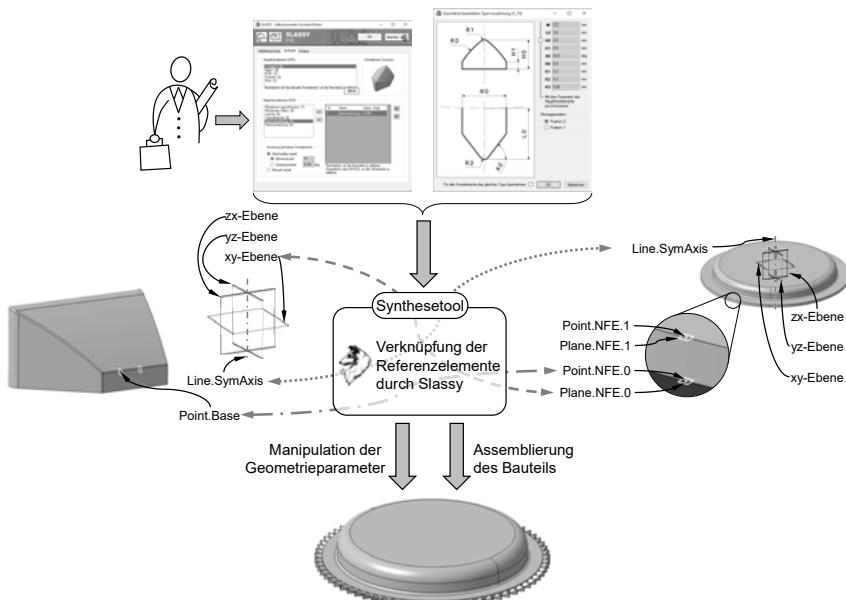


Bild 81: Zuweisung der Referenzelemente von Haupt- und Nebenformelement durch das Synthesetool.

Die Oberfläche des Synthesetools bietet dem Benutzer ein hohes Maß an Flexibilität. Neben Kreismustern mit variierender Nebenformelementanzahl, können verschiedene Nebenformelemente kombiniert sowie ihre Positionen am Umfang des Hauptformelements einzeln festgelegt werden.

7.2 Erhebung der Fertigungsdaten

Die in diesem Abschnitt dargestellte Datenerhebung ist fester Bestandteil eines KDD-Prozesses (s. Abschnitt 2.4.1, S. 42) und erfordert eine sorgfältige Vorbereitung, denn sie beeinflusst die Qualität des Gesamtergebnisses entscheidend. Da die Datenerhebung im Rahmen des SFB/TR 73 teilweise in der Fertigungsprozessentwicklung verortet war, wird im Folgenden nur die grobe Vorgehensweise dargestellt. Für Details zur Fertigungstechnologie wird auf bereits zitierte Veröffentlichungen im Abschnitt 3 verwiesen. Das Vorgehen

beginnt mit der Definition der Produktgrößen, deren Einfluss auf das Fertigungsergebnis untersucht werden soll (Attribute). Diese werden in einer simulativen Parameterstudie entsprechend eines statistischen Versuchsplans variiert. Die Auswertung der Simulationen zur Ermittlung der Zielgrößen (Label) schließt die Datenerhebung ab.

7.2.1 Auswahl der relevanten Parameter und Entwicklung des Versuchsplans

Das Synthesetool bietet dem Konstrukteur die Möglichkeit, sowohl am Haupt- (Napf mit Flansch) als auch am Nebenformelement (Sperrverzahnung) insgesamt 17 Geometrieparameter zu variieren. Der qualitative Einfluss dieser Parameter (Faktoren) auf Umformprozessgrößen kann durch Methoden der statistischen Versuchsplanung (s. Abschnitt 2.3.3, S. 39) ermittelt werden. Dabei werden die verschiedenen Ausprägungen (Einstellungen) der Faktoren nach festen Schemata kombiniert. Zur Erfassung nicht-linearer Zusammenhänge und Wechselwirkungen haben sich vor allem vollfaktorielle Versuchspläne mit drei Stufen je Faktor etabliert [Sch14a; DB10].

Ein vollfaktorieller, dreistufiger Versuchsplan mit allen Parametern des blechmassivumgeformten Bauteils würde jedoch $3^{17} = 129.140.163$ Simulationen erfordern, weshalb eine Reduzierung der untersuchten Parameter sowie eine Festlegung ihrer Grenzen notwendig ist. Durch Gespräche mit Fertigungsexperten können die Geometrieparameter identifiziert werden, die sowohl aus Sicht der Konstruktion als auch der Prozessentwicklung von hoher Bedeutung sind. Eine detaillierte Betrachtung dieser Parameter erfolgt in Abschnitt 7.2.2 (s. S. 139). Ziel der fachlichen Diskussionen ist letztlich ein Kompromiss, bei dem die Interessen beider Seiten berücksichtigt werden. Der Konstrukteur möchte beispielsweise während der Bauteilauslegung die Zahnbreite, die Zahnlänge und den Flankenwinkel besonders häufig variieren, um den Einfluss der sich dadurch ändernden Kontaktfläche (s. Bild 47, S. 90) während des Synchronisationsvorgangs zu untersuchen. Der Fertigungstechnologe ist wiederum an der Realisierung kleiner, komplexer Formfeatures wie dem Kopf- und dem Dachradius interessiert, da dies ein erklärtes Ziel der Forschungsarbeiten des SFB/TR 73 ist. Die Blechmassivumformung zeichnet sich des Weiteren durch sehr hohe und lokale Beanspruchungen mit starken Gradienten in der Umformzone aus [MAB¹²]. Eine systematische Variation kleiner Geometrien lässt hinsichtlich dieses Aspekts gezielte Untersuchungen zu.

Neben den zu variiierenden Parametern müssen vom Konstrukteur und Fertigungsexperten die Zielgrößen festgelegt werden, auf deren Basis eine Design for Manufacture Analyse durchgeführt werden kann. Für das Sperrverzahnungsbauteil sind dies der Umformgrad φ , die Umformkraft F_U sowie das Kontaktverhältnis c (engl. contact ratio) zwischen Werkstoff und Werkzeugkavität.

Der *Umformgrad* ist der natürliche Logarithmus des Formänderungsverhältnisses und eine wichtige Größe zur Beurteilung eines Umformprozesses, z. B. bei der Ermittlung des Kraft- und Arbeitsbedarfs [Kug09; DB10]. In besonders kritischen Bereichen des Bauteils

darf der Gesamtumformgrad einen höchstzulässigen Wert φ_{vG} nicht übersteigen. Dieser Grenzumformgrad ist nicht nur abhängig vom Werkstoff, sondern u. a. auch vom angewandten Verfahren, der Umformgeschwindigkeit und der Temperatur [Klo07]. Aber auch für den Konstrukteur ist der Umformgrad von Bedeutung, denn hohe Umformgrade gehen mit einer hohen Zunahme der Versetzungsichte im metallischen Gefüge einher und führen zu (Kalt-)Verfestigungen des Werkstoffs [IS10]. Grenzumformgrade lassen sich aufgrund einer Vielzahl an Einflussfaktoren (Werkstoff, Verfahren, Umformvorgeschichte, Temperatur, etc.) nur sehr schwer definieren. Der maximal erreichte Umformgrad nach der Umformung variiert im vorliegenden Beispiel je nach simulierter Bauteilvariante in Betrag und Ort (s. Bild 82). Für die Auswertung gilt es zudem zu beachten, dass der maximale Umformgrad auch in Bereichen liegen kann, die für die DfM-Analyse mitunter irrelevant sein können. In Bild 82 ist dies z. B. Bereich 1. Hier bildet sich während der Umformung ein Grat, der durch eine Nachbearbeitung (z. B. Laserschneiden) entfernt wird. Für die Auswertung ist der Bereich 2 relevant.

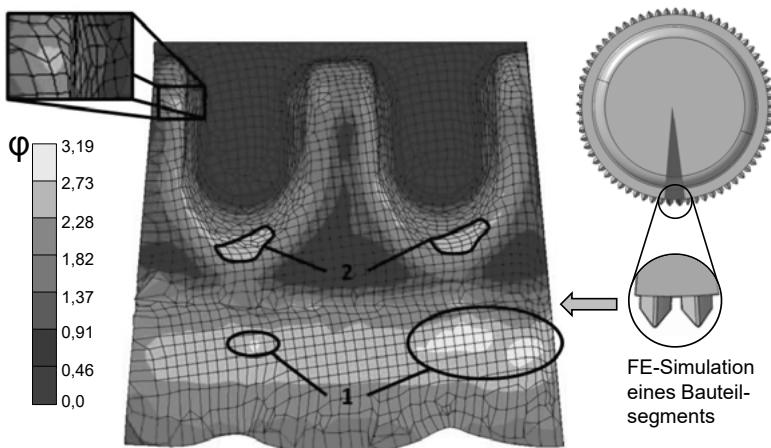


Bild 82: Auswertung der FE-Simulation des Sperrverzahnungsbauteil anhand des Umformgrads φ .

Die maximale *Umformkraft* $F_{u_{max}}$ muss bei einem Umformprozess durch die Umformmaschine aufgebracht werden und ist daher für die Wahl der Maschine von zentraler Bedeutung [DB10]. Ihre Berechnung ist eine komplexe Aufgabe, da unterschiedliche Einflussfaktoren berücksichtigt werden müssen. Neben der Bauteilgeometrie und dem -werkstoff hat vor allem die gewählte Umformtemperatur Einfluss auf die Umformkraft. Die insgesamt aufzubringende Umformkraft kann nach Siebel durch Addition der ideellen Umformkraft F_{id} mit dem Reib- (F_R), Schiebungs- (F_{Sch}) und Biegeanteil (F_B) ermittelt werden [Sie32]:

$$F_{ges} = F_{id} + F_R + F_{Sch} + F_B \quad (7.1)$$

Die Berechnung der einzelnen Anteile gestaltet sich je nach Prozess unterschiedlich, sodass keine allgemeingültigen Formeln angegeben werden können. Für etablierte Verfahren können empirisch ermittelte, spezifische Gleichungen bspw. aus [Tsc05] entnommen werden. Für die Prozesse der Blechmassivumformung existieren derartige Erfahrungswerte nicht, weshalb für den Fertigungstechnologen eine Vorhersage der Umformkraft eine Unterstützung in der Prozessentwicklung darstellt. Zudem kennt er die zur Verfügung stehenden Umformmaschinen und kann durch Kenntnis der maximalen Umformkraft entscheiden, ob eine experimentelle Validierung der Simulationsergebnisse möglich ist.

Als dritte Zielgröße beschreibt das *Kontaktverhältnis* die relative Größe der Kontaktfläche zwischen dem umgeformten Werkstoff und der Werkzeugkavität, die die Sperrverzahnung abbildet. Ein Wert von $c = 0,0$ entspricht dabei keinem Kontakt, während ein Wert von $c = 1,0$ einer komplett ausgefüllten Kavität und damit einer vollständigen Ausformung der Sperrverzahnung entspricht. Bild 83 zeigt beispielhaft die Ausformung einer Sperrverzahnung bei $c = 0,87$. Das Kontaktverhältnis ergibt sich aus dem Anteil der gelben (kein Kontakt) und blauen (Kontakt) Bereiche.

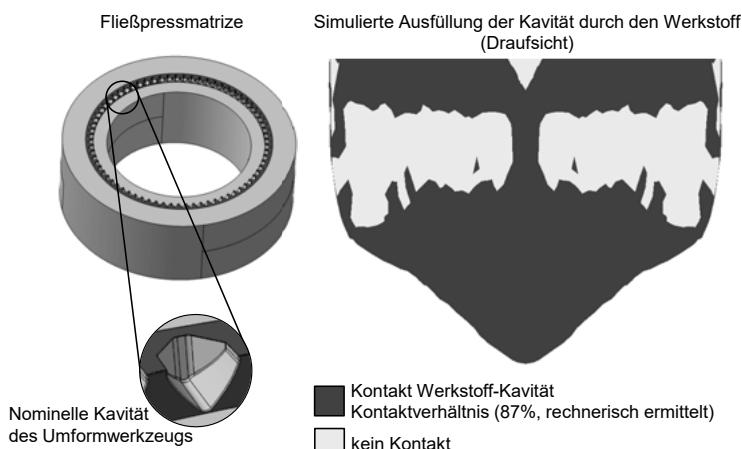


Bild 83: Der Formfüllungsgrad (entspr. Kontaktverhältnis) einer Simulation im Anwendungsbeispiel.

Als Versuchsplan kommt ein dreistufiger, vollfaktorieller Aufbau zum Einsatz, mit der Anpassung, dass der Parameter „Zahnlänge“ (X_T0_L0) nur in einem schmalen Wertebereich auf zwei Stufen variiert wird. Dies begründet sich v. a. durch die frühe Entwicklungsphase des Blechmassivumformprozesses, in welcher der lateral-radiale Stofffluss noch nicht ausreichend kontrolliert werden kann. Somit ergeben sich $N = 3^4 \cdot 2^1 = 162$ Simulationen. Der gesamte Versuchsplan mit Faktoreinstellungen und sich ergebenden Zielgrößen kann im Anhang aus der Tabelle 8 (S. 160) entnommen werden.

7.2.2 Durchführung der Parameterstudie und Auswertung

Die Parameterstudie wird simulativ mithilfe der Software *Simufact Forming* umgesetzt. Generell werden bei einer FEM-Simulation die Aufgaben a) Preprocessing (Erstellung eines FEM-Modells), b) Solving (Lösen der Differentialgleichungen durch Software) und c) Postprocessing (Ergebnisaufbereitung und -auswertung) unterschieden [MG15; Vaj09]. Abschnitt 2.3.3 enthält eine genaue Beschreibung dieser Schritte. Das FEM-Modell zur Simulation des Blechmassivumformprozesses basiert auf dem CAD-Modell des Umformwerkzeugs, wie in Bild 84 (links) gezeigt. Durch ausnutzen der Symmetrie kann aus dem Gesamtwerkzeug ein Kreissegment ($\beta = 10^\circ$) ausgeschnitten werden, das zwei Sperrverzahnungen umfasst (s. Bild 84). Dadurch kann bspw. der Stofffluss zwischen zwei Nebenformelementen untersucht werden, was bei Kreissegmenten mit nur einem Zahn nicht möglich wäre. Die Simulation rechnet inkrementweise, wobei nach Bedarf zu Beginn eines neuen Inkrementen eine Neuvernetzung des Bauteils erfolgt, falls im vorhergehenden Inkrement eine zu starke Netzverzerrung aufgrund der plastischen Verformung aufgetreten ist.

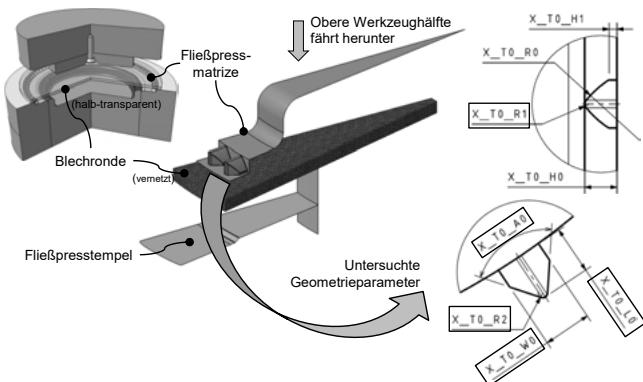


Bild 84: Darstellung des FEM-Preprocessings im Anwendungsbeispiel.

Von insgesamt 162 Simulationen konnten zehn nicht vollständig durchgeführt werden. Grund hierfür waren u.a. Fehler bei der automatischen Neuvernetzung zu Beginn einzelner Simulationsinkremente. Dem Datensatz werden zudem zehn zufällig ausgewählte Tupel entnommen, um die Performanz der durch die Selbstlernkomponente trainierten Modelle im Rahmen dieses Anwendungsbeispiels getrennt zu evaluieren (Hold-out Verfahren, s. Abschnitt 2.4.2, S. 47). Tabelle 11 fasst die Ergebnisse der simulativen Parameterstudie zusammen.

Die erhobenen Fertigungsdaten werden anschließend in die multidimensionale Datenbasis von Slassy übernommen. Hierzu nutzt der Anwender direkt die Oberfläche des Assistenzsystems und legt fest, für welches Bauteil und für welche Parameter er Daten importieren möchte (s. Bild 76). Nach dem Import können die Daten aller bisherigen Versuche aufgerufen, angepasst und ggf. mit neuen Daten ergänzt werden. Der Start des KDD-basierten

Tabelle 11: Zusammenfassung der Ergebnisse aus der Simulationsstudie.

Name	Einheit	Mittelwert	Std.-abw.	Stufen bzw. Range
X_T0_W0	mm	2.50662	0.40409	[2,00; 2,50; 3,00]
X_T0_L0	mm	2.75497	0.25078	[2,50; 3,00]
X_T0_R1	mm	0.36623	0.17005	[0,20; 0,30; 0,60]
X_T0_R2	mm	0.59801	0.24616	[0,30; 0,60; 0,9]
X_T0_A0	deg	55.6788	3.09480	[52,50; 55,00; 60,00]
Umformgrad (max)	-	2.44591	0.25966	[1,89125; 3,07478]
Umformkraft (max)	kN	1905,00	289,000	[1234,00; 2567,00]
Kontaktverhältnis	-	0.51476	0.05064	[0,2597; 0,60739]

Selbstlernprozesse erfolgt ebenfalls über die Oberfläche des Assistenzsystems. Der folgende Abschnitt beschreibt die Datenverarbeitung in der Selbstlernkomponente.

7.3 Verarbeitung der Daten in der Selbstlernkomponente

Die erhobenen und in der multidimensionalen Datenbasis gespeicherten Fertigungsdaten werden im hier beschriebenen Anwendungsfall für die drei Ergebnisgrößen getrennt verarbeitet, denn jeder instantiierte Selbstlernprozess trainiert pro Durchlauf nur Metamodelle für ein Label (Ergebnisgröße). Da die Funktionsweise der Selbstlernkomponente in jedem Fall gleich bleibt, wird sie im Folgenden nur für den maximalen Umformgrad beschrieben.

7.3.1 Import und Vorbehandlung der Daten

Da die multidimensionale Datenbasis im Assistenzsystem Slassy mithilfe der SQL-Datenbanksprache aufgebaut wird, muss ein Zugriff auf diese Daten durch entsprechende Befehle erfolgen. Um dem Benutzer dies zu erleichtern, wird eine temporäre SQL-Abfrage im Assistenzsystem hinterlegt (s. Bild 85). Diese wird entsprechend der Benutzerauswahl in der grafischen Benutzeroberfläche (s. Bild 86) angepasst.

```

SELECT
    FormParaWert.Versuch_ID,
    GROUP_CONCAT(IF(FormParaWert.FormelementePara_ID = 9, FormParaWert.wert_numerisch, NULL)) AS 'X_TO_W0',
    GROUP_CONCAT(IF(FormParaWert.FormelementePara_ID = 10, FormParaWert.wert_numerisch, NULL)) AS 'X_TO_L0',
    GROUP_CONCAT(IF(FormParaWert.FormelementePara_ID = 13, FormParaWert.wert_numerisch, NULL)) AS 'X_TO_A0',
    GROUP_CONCAT(IF(FormParaWert.FormelementePara_ID = 15, FormParaWert.wert_numerisch, NULL)) AS 'X_TO_R1',
    GROUP_CONCAT(IF(FormParaWert.FormelementePara_ID = 18, FormParaWert.wert_numerisch, NULL)) AS 'X_TO_R2',
    ErgebParaWerte.wert_numerisch AS 'Label'
FROM
    FormParaWerte, ErgebParaWerte
WHERE
    FormParaWerte.Versuch_ID = ErgebParaWerte.Versuch_ID
    AND FormParaWerte.ErgebPara_Bauteil_ID = 1
    AND FormParaWerte.ErgebPara_ID = 1
GROUP BY
    Versuch_ID;

```

Bild 85: SQL-Abfrage der Trainingsdaten für das Anwendungsbeispiel.

Wie im Abschnitt 6.3.2 (S. 101) dargestellt, werden nach dem Import der Daten zunächst alle im String-Format vorliegenden Tupel in Daten-Typen umgewandelt, die durch das Data-Mining Werkzeug Rapidminer® verarbeitet werden können („Parse Numbers“ Operator). Da es sich bei jedem Attribut um einen Geometrieparameter handelt, der grundsätzlich beliebige (positive) Werte annehmen kann, werden alle Attribute als „real“ (reelle Zahl) klassifiziert. Anschließend wird durch den „Set Role“ Operator die Ergebnisgröße (Umformgrad) als Label definiert. Die prozessinterne generische Benennung „Label“ wird dem spezifischen Namen vorgezogen, da so weniger Anpassungen im Prozesstemplate notwendig sind. Der Operator „Filter Examples“ entfernt die zehn Datentupel bei denen die Simulationen vorzeitig von der Software abgebrochen wurden. Ein Auffüllen der Lücken durch das arithmetische Mittel der vorhandenen Werte ist aufgrund der geringen Größe des Datensatzes ($N = 142$) nicht empfehlenswert [WEH11]. Abschließend analysiert der Operator „Remove Useless Attributes“ ob Eingangsgrößen aus dem Datensatz entfernt werden können. Da der Benutzer bereits über die Oberfläche des Assistenzsystems die Geometrieparameter ausgewählt hat, die im Rahmen der Simulationsstudie variiert wurden, entfernt dieser Operator keine weiteren Attribute. Der Selbstlernprozess wird mit fünf Attributen, einem Label und 142 Datentupel durchgeführt.

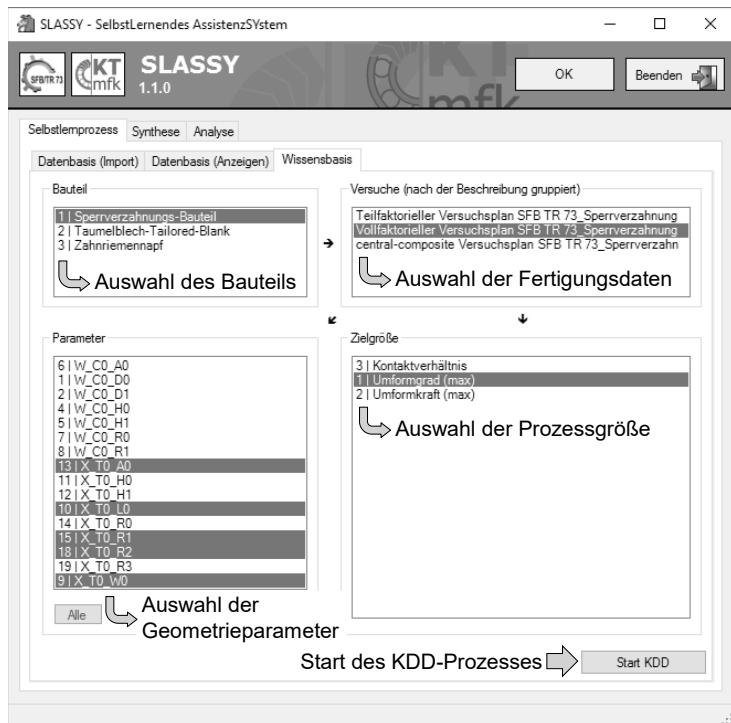


Bild 86: Vorbereitung des Selbstlernprozesses durch Parameter- und Datenauswahl.

7.3.2 Automatische Attributselektion

Wie im Abschnitt 6.3.3 dargestellt verfügt das Assistenzsystem über eine automatische Attributselektion, die durch Rückkopplung mit den jeweiligen Lernalgorithmen eine optimale Attributauswahl durchführt. Die beiden Selektionsstrategien „schrittweise Vorwärtsselektion“ und „schrittweise Rückwärtseliminierung“ werden für jeden der vier Lernalgorithmen (s. Abschnitt 6.3.1) je einmal angewandt, so dass an dieser Stelle im Gesamtprozess acht Teilprozesse durchlaufen werden. Die Teilergebnisse der Attributselektionen müssen im Standardbetrieb vom Benutzer weder ausgewertet noch berücksichtigt werden. Im Rahmen dieses Anwendungsbeispiels erfolgt jedoch eine kurze Erläuterung der Attributselektionen anhand des Bildes 87. Deutlich ist der Einfluss unterschiedlicher Selektionsstrategien auf die verbleibende Menge an Attributen zu erkennen. Mit Ausnahme der linearen Regression unterscheiden sich bei allen Metamodelltypen die identifizierten Attributgruppen. Ebenso sind auch bei den geschätzten Performanzen teilweise deutliche Unterschiede zu erkennen. Dies rechtfertigt das grundlegende Prinzip des selbstlernenden Assistenzsystems - ein

flexibles Training verschiedener Metamodelle mit anschließender statistisch abgesicherter Modellauswahl - umso mehr.

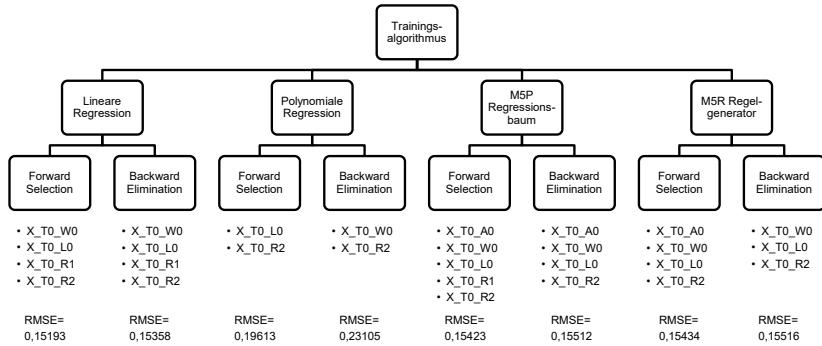


Bild 87: Die erzeugten Attributmengen mit geschätzten Fehlerkennwerten.

7.3.3 Durchlaufen des ROPE-Prozesses

Aufgrund der Kombination unterschiedlicher Metamodelltypen, Selektions- und Optimierungsverfahren werden durch den Selbstlernprozess 24 Metamodelle trainiert (s. Bild 61, S. 111) und die jeweiligen Vorhersageperformanzen geschätzt. In Abschnitt 5.1 wurde dargestellt, dass die Berechnung des Kennwerts zur Beurteilung der Prognosequalität eines Metamodells robust und verlässlich erfolgen muss. Außerdem kann durch die Optimierung der Trainingsparameter die Performance eines Metamodells weiter verbessert werden. Zu diesem Zweck wurde der sogenannte ROPE-Prozess (engl. Robust Optimization and Performance Estimation) entwickelt (s. Abschnitt 5.2 ff.) und in Rapidminer® implementiert (s. Abschnitt 6.3.4).

Neben den im vorherigen Abschnitt durch Attributselektion erzeugten und teilweise reduzierten Datensätzen ($N = 142$, $A = \text{variabel}$) wird je Metamodelltyp auch der vollständige Initialdatensatz ($N = 142$, $A = 5$) zum Training verwendet. Auf alle Datensätze werden die beiden Strategien der schrittweisen und der evolutionären Optimierung angewandt und somit die Basismenge an 24 Metamodellen mit den robust geschätzten Vorhersageperformanzen erzeugt. Exemplarisch wird im Folgenden der ROPE-Prozess mit evolutionärer Optimierung eines M5P-Modellbaums und dem Initialdatensatz herausgegriffen, um die Abläufe des Selbstlernprozesses zu erläutern.

Kennzeichnend für alle ROPE-Prozesse ist die hierarchische Struktur, die in generischer Form im Abschnitt 6.3.4 (S. 106) erläutert ist (s. auch Bild 59). Der Initialdatensatz wird an einen „Loop and Average“ Operator übergeben, in den eine Kreuzvalidierung (s. Abschnitt 2.4.2, S. 47) integriert ist. Diese Integration ist für die 20-malige Wiederholung der 10-fachen Kreuzvalidierung notwendig, da nur so eine robuste Schätzung des Vorhersage-

fehlers sichergestellt ist. In der Kreuzvalidierung erfolgt eine Aufteilung des Initialdatensatzes in einen Trainingsdatensatz ($N_{90\%} \approx 128$) und einen Testdatensatz ($N_{10\%} \approx 14$). In das Training der Kreuzvalidierung ist die Optimierung der Trainingsparameter des M5P-Modellbaums integriert.

Zu den verfügbaren Trainingsparametern eines M5P-Modellbaums gehören:

- N: Angabe ob der Baum vereinfacht (engl. *pruning*) werden soll (boolsche Variable)
- U: Vorhersagen in Übergangsbereichen des Baums werden geglättet (boolsche Var.)
- R: Training eines Regressionsbaums anstatt eines Modellbaums (boolsche Variable)
- M: minimale Anzahl an Instanzen pro Blatt ($M \in \mathbb{Z}$)
- L: Abspeicherung der Instanzen an Baumverzweigungen (boolsche Variable)

Von diesen verfügbaren Parametern werden N, U, R und M in der Optimierung berücksichtigt. L wird lediglich zu Visualisierungszwecken benötigt und hat keinen Einfluss auf den Vorhersagefehler des M5P-Metamodells. Für alle anderen Metamodelltypen (lineare und polynomiale Regression, M5R-Regellerner) werden die jeweiligen Trainingsparameter für die schrittweise und evolutionäre Optimierung ebenso festgelegt.

Die Optimierung im vorliegenden Beispiel verfolgt das Ziel einzelne „Individuen“ von M5P-Metamodellen mit konkreten Ausprägungen von N, U, R und M entsprechend einer evolutionären Strategie zu trainieren und zu validieren. Hierzu ist eine 10-fache Kreuzvalidierung in die Optimierung zu integrieren (untere Ebene) und zur Gewährleistung der Robustheit 5-mal zu wiederholen. Die Reduzierung der Anzahl an 10-fachen Kreuzvalidierungen auf $n = 5$ (im Vergleich zu $n = 20$ auf der oberen Ebene) begründet sich in dem enorm steigenden Rechenbedarf. Aufgrund der hierarchischen Struktur des ROPE-Prozesses erhöht sich die Rechenzeit, abhängig vom eingesetzten Computer, um das bis zu fünffache. In der Optimierung werden, im Unterschied zur Kreuzvalidierung auf der oberen Ebene, nur noch $N_{81\%} \approx 115$ Datentupel für das Training und $N_{9\%} \approx 13$ Datentupel für das Testen der einzelnen Optimierungsindividuen genutzt. Bild 88 zeigt den Verlauf der Trainingsparameter während der Optimierung. Die Konvergenz des Parameters sowie die tendenzielle Verkleinerung der kreisförmigen Piktogramme (entspricht reduziertem Vorhersagefehler) mit steigender Anzahl an Optimierungsiterationen sind deutlich zu erkennen.

Nachdem die Optimierung abgeschlossen ist, wird mit den ermittelten Werten der Trainingsparameter ein M5P-Modellbaum trainiert und im Rahmen der Kreuzvalidierung auf der oberen Ebene validiert. Der Trainingsteil der 10-fachen Kreuzvalidierung wird nach deren Abschluss ein elftes Mal mit dem kompletten Initialdatensatz durchlaufen und der dabei angestoßene Optimierungsprozess berechnet ein neues, optimales Set an Trainingsparametern auf Basis aller Trainingsdaten. Das mit diesem Set trainierte M5P-Metamodell wird nach Beendigung des „Loop and Average“ Operators abgerufen.

Grundlage für den späteren Vergleich der Metamodelle sind die durch die Kreuzvalidierung auf der oberen Ebene geschätzten Vorhersagefehler, die in den sogenannten Performanzvektoren zusammengefasst sind. Die Erzeugung der Performanzvektoren geschieht durch Zwischenspeichern der Werte des RMSE (Wurzel des mittleren Fehlerquadrats, s. Tabelle

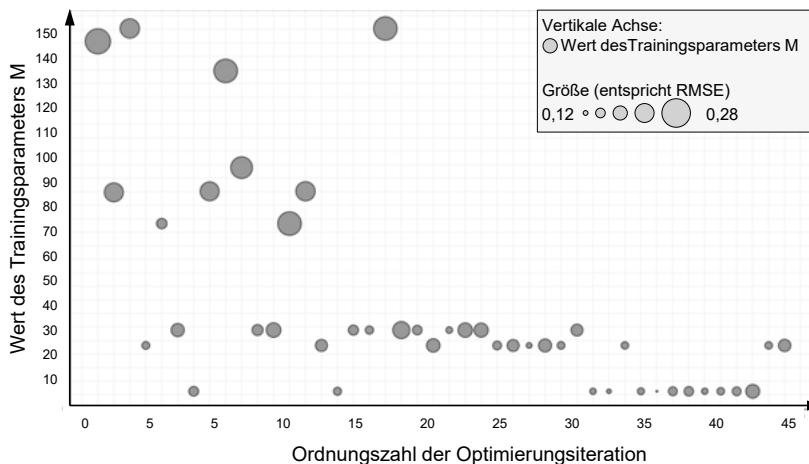


Bild 88: Gesamtansicht des Prozesses.

4) und Ausgabe als Spaltenvektor. Dessen Länge richtet sich nach den Werten für n sowie k und beträgt $n = 20 \cdot 10 = 200$.

7.3.4 Modellauswahl und -evaluierung

Alle 24 Metamodelle, die nach dem Durchlaufen der einzelnen ROPE-Prozesse vorliegen, können prinzipiell dazu genutzt werden, auf der Basis von Geometrieparametern der Sperrverzahnung den zu erwartenden maximalen Umformgrad vorherzusagen. Das Assistenzsystem ist in der Lage, in dieser Menge an Modellen das bestgeeignete zu identifizieren und für den Benutzer bereitzustellen. Hierzu wird ein zweistufiger statistischer Auswahlprozess durchlaufen, dessen Entwicklung in Kapitel 5 beschrieben ist. Zunächst wird auf Basis einer Varianzanalyse (ANOVA, engl. Analysis of Variances) die Aussage getroffen, ob bei mindestens zwei Performanzvektoren ein signifikanter Unterschied zwischen den Vorhersagefehlern vorliegt. Signifikant bedeutet dabei, dass der Unterschied nicht zufällig auftritt, sondern als deterministisch und wiederholbar angesehen werden kann. Das Modell mit dem kleineren Vorhersagefehler liefert in diesem Fall die verlässlichere Vorhersage des Umformgrads.

Die Eigenschaften der erzeugten Performanzvektoren sind in Tabelle 12 zusammengefasst. Die Tabelle zeigt in ihren Quadranten die vier Metamodelltypen lineare Regression ($1 \leq M_i \leq 6$), polynomiale Regression ($7 \leq M_i \leq 12$), M5P-Modellbaum ($13 \leq M_i \leq 18$) sowie M5R-Regelgenerator ($19 \leq M_i \leq 24$). Für eine genaue Aufschlüsselung der Modellbezeichnungen wird auf den Anhang A4 (S. 162) verwiesen. Die dargestellten Werte

entsprechen jeweils der Wurzel des mittleren Fehlerquadrats (RMSE), die in den 20-maligen 10-fachen Kreuzvalidierungen geschätzt wird.

Tabelle 12: Übersicht der Eigenschaften der erzeugten 24 Performanzvektoren.

M_i	\bar{x}_{M_i}	\tilde{x}_{M_i}	$Var[X_{M_i}]$	Min	Max		M_i	\bar{x}_{M_i}	\tilde{x}_{M_i}	$Var[X]$	Min	Max
1	0,515	0,513	0,00117	0,422	0,627		13	0,277	0,280	0,00154	0,168	0,409
2	0,489	0,490	0,00730	0,408	0,558		14	0,291	0,288	0,00159	0,177	0,390
3	0,702	0,704	0,00082	0,628	0,788		15	0,231	0,231	0,00141	0,107	0,335
4	0,588	0,591	0,00171	0,457	0,706		16	0,092	0,092	0,00037	0,040	0,154
5	0,485	0,480	0,00173	0,360	0,633		17	0,345	0,346	0,00104	0,257	0,439
6	0,523	0,524	0,00291	0,386	0,674		18	0,308	0,307	0,00110	0,228	0,398
7	0,828	0,827	0,00086	0,748	0,908		19	0,481	0,481	0,00034	0,434	0,531
8	1,049	1,046	0,00099	0,977	1,161		20	0,417	0,418	0,00251	0,289	0,518
9	1,139	1,139	0,00380	1,002	1,308		21	0,299	0,297	0,00158	0,176	0,398
10	1,022	1,027	0,00481	0,801	1,245		22	0,306	0,305	0,00214	0,178	0,450
11	1,199	1,200	0,00392	1,036	1,368		23	0,556	0,555	0,00175	0,446	0,655
12	0,914	0,916	0,00139	0,819	1,020		24	0,519	0,520	0,00141	0,427	0,618

Varianzanalyse (ANOVA) der Performanzvektoren

Grundgedanke der Varianzanalyse ist die Zerlegung der Gesamtvarianzen der Performanzvektoren in systematische und unsystematische Anteile. Überwiegen die systematischen Anteile, so kann davon ausgegangen werden, dass die unterschiedlichen Modelltypen, Attributselektionsstrategien und Optimierungsalgorithmen einen signifikanten Einfluss auf die Prognosequalität der Metamodelle haben.

Die Durchführung der ANOVA verlangt, dass die genutzten Daten bestimmten Anforderungen genügen die im Abschnitt 5.4 (S. 82) behandelt wurden. Die Unabhängigkeit der Einträge in den Performanzvektoren ist aufgrund der Ermittlung in getrennten ROPE-Prozessen gegeben. Die Varianzhomogenität wird üblicherweise durch gesonderte Tests (z. B. Levene-Test) genauer analysiert werden, kann allerdings aufgrund der gleichen Stichprobengrößen als erfüllt angesehen werden. Des weiteren lässt sich aus der von Bock formulierten Faustregel für $z \geq 50$ ableiten, dass die Performanzen normalverteilt vorliegen [Boc75].

Zunächst wird die totale Varianz aller Vorhersagefehler aus deren arithmetischem Gesamtmittel und den Einzelwerten berechnet:

$$\sigma_{tot}^2 = \frac{\sum_M \sum_z (x_{Mz} - \bar{X})^2}{n - 1} = \frac{\sum_1^{24} \sum_{1}^{200} (x_{Mz} - 0,5661)^2}{(24 \cdot 200) - 1} = 0,4021 \quad (7.2)$$

Zur Ermittlung der unsystematischen Einflüsse auf Unterschiede zwischen den Performanzvektoren muss die Residualvarianz, auch als Fehler- oder innere Varianz bezeichnet, berechnet werden. Unter idealen Bedingungen sollte Varianzhomogenität herrschen, die Residualvarianzen der einzelnen Performanzvektoren quasi gleich sein. Jedoch ist dies in der Realität oftmals nicht gegeben. Daher werden die einzelnen geschätzten Residualvarianzen

der Vektoren addiert und durch die Gesamtanzahl geteilt, um einen Mittelwert zu bilden. Die Residualvarianz ergibt sich im vorliegenden Beispiel zu:

$$\sigma_{res}^2 = \frac{\sum_M \sum_z (x_{Mz} - \bar{x}_{M_i})^2}{M \cdot (z - 1)} = \frac{\sum_1^{24} \sum_1^{200} (x_{Mz} - \bar{x}_{M_i})^2}{24 \cdot (200 - 1)} = 0,00174 \quad (7.3)$$

Die Berechnung der Treatmentvarianz erfolgt mithilfe der einzelnen Mittelwerte \bar{x}_{M_i} und des Gesamtmittelwerts \bar{X} aller Performanzvektoren. Die Freiheitsgrade im Nenner ergeben sich aus der Anzahl der verglichenen Performanzvektoren. Die Quadratsumme der Mittelwertabweichungen vom Gesamtmittelwert muss zusätzlich mit der Menge der Einzelwerte in einer Gruppe multipliziert werden, damit die Genauigkeit der Mittelwerte als Populationsschätzer berücksichtigt wird:

$$\sigma_{treat}^2 = \frac{z \cdot \sum_{i=1}^M (\bar{x}_{M_i} - \bar{X})^2}{M - 1} = \frac{200 \cdot \sum_1^{24} (\bar{x}_{M_i} - 0,5661)^2}{24 - 1} = 18,9456 \quad (7.4)$$

Am Ende der ANOVA steht die Überprüfung der Nullhypothese der Varianzanalyse die besagt, dass zwischen den Mittelwerten aller Performanzvektoren (s. Tabelle 12) kein signifikanter Unterschied besteht:

$$H_0 : \bar{x}_1 = \bar{x}_2 = \dots = \bar{x}_{24} \quad (7.5)$$

Eine Ablehnung der Nullhypothese zugunsten der Alternativhypothese, dass zwischen mindestens zwei Performanzvektoren ein Unterschied existiert, basiert auf der Berechnung der Prüfgröße F , die die Treatmentvarianz (Varianz zwischen den Gruppen) mit der Residualvarianz ins Verhältnis setzt (s. Gleichung 7.6). Ein Vergleich des ermittelten F -Werts mit dem kritischen F -Wert erlaubt im vorliegenden Beispiel die Ablehnung der Nullhypothese. Es kann davon ausgegangen werden, dass zwischen mindestens zwei Performanzvektoren ein statistisch signifikanter Unterschied besteht.

$$F_{wert} = \frac{\sigma_{treat}^2}{\sigma_{res}^2} = \frac{18,9456}{0,00174} = 10,872 \geq F_{krit(df_{treat}, df_{res}, \alpha_{krit})} = 1,5315 \quad (7.6)$$

Das Ergebnis einer ANOVA wird als sogenannte ANOVA-Tabelle bezeichnet (s. Tabelle 13) und enthält die zuvor „manuell“ berechneten Werte der Varianzen. Ein Vergleich zeigt, dass beide Wege zum gleichen Ergebnis führen, da auch die automatisierte ANOVA eine Ablehnung der Nullhypothese zulässt.

Tabelle 13: Ergebnistabelle der in Matlab® durchgeführten ANOVA.

Variation	SS	DOf	Mean SS	F _{value}	p-value	F _{crit}
Treatment	435,749	23	18,94561	10872,21	2,34e-132	1,5315
Residual	8,323	4776	0,00174			
Total	444,072	4799				

post-hoc Test der Performanzvektoren nach Tukey

Mithilfe der Varianzanalyse kann festgestellt werden, ob zwischen den Performanzvektoren ein genereller Unterschied besteht. Eine Aussage darüber, welches Metamodell den signifikant kleinsten Fehler aufweist, ist jedoch nicht möglich. Hierzu beinhaltet der Selbstlernprozess einen post-hoc Test nach Tukey, der die Performanzvektoren paarweise vergleicht, ohne dass es dabei zur α -Fehlerkumulierung (s. Abschnitt 5.4) kommt.

Bei diesem Test wird eine kritische Mittelwertdifferenz zwischen den Performanzvektoren berechnet, deren Überschreitung einem signifikanten Unterschied zwischen zwei betrachteten Performanzvektoren gleichkommt. Diese „Honest Significance Distance“ (HSD) wird über den Kennwert q ermittelt, der dem t -Wert beim gleichnamigen Test entspricht und ähnlich definiert ist:

$$q_{(M; df_{innerhalb})} = \frac{\bar{x}_{M_i} - \bar{x}_{M_j}}{\sqrt{\frac{\sigma_{res}^2}{z}}}; \quad \exists i, j \text{ mit } i \neq j \quad (7.7)$$

Dabei entsprechen \bar{x}_{M_i} und \bar{x}_{M_j} den Mittelwerten der paarweise zu vergleichenden Performanzvektoren und σ_{res}^2 der Residualvarianz, die bereits in der ANOVA berechnet wurde. Da dem q -Wert die „Studentized-Range“ Verteilung zugrunde liegt, lässt sich ein *kritischer* q -Wert ermitteln, ab dem eine beobachtete Mittelwertdifferenz statistisch signifikant auf einem bestimmten α -Niveau ist. Durch Einsetzen des kritischen q -Werts ist es möglich, eine kritische Differenz zu bestimmen, mit der die tatsächlichen Differenzen zwischen den Gruppenmittelwerten verglichen werden:

$$HSD = q_{krit(\alpha, M, df_{res})} \cdot \sqrt{\frac{\sigma_{res}^2}{z}} = \underbrace{q_{krit(0,05, 24, 4776)}}_{5,144} \cdot \sqrt{\frac{\sigma_{res}^2}{200}} = 0,01518 \quad (7.8)$$

Anhand der berechneten *Honest Significant Difference* können die Mittelwerte der 24 Metamodellperformanzen paarweise miteinander verglichen werden. Tabelle 14 (Anhang, S. 163) zeigt die Beträge der Mittelwertdifferenzen aller Performanzvektoren. Ein Großteil der Mittelwertdifferenzen übersteigt den berechneten Wert der *Honest Significant Difference*. Diese Differenzen können als statistisch signifikant eingestuft werden. Die Performanz des Modells Nummer 16 unterscheidet sich ebenfalls signifikant von allen anderen. Der geringe Vorhersagefehler dieses Metamodells kann somit auf systematische Ursachen (Attributselektion, Optimierungsstrategie) zurückgeführt und eine reine Zufälligkeit ausgeschlossen werden. Daher wird dieses Modell als bestgeeignetes Metamodell ausgewählt.

Eine derartige manuelle Vorgehensweise zur Identifikation des bestgeeigneten Modells ist für den normalen Gebrauch des Assistenzsystems unpraktikabel. Der Benutzer muss noch in weiterem Maße unterstützt werden, um seine Akzeptanz gegenüber Slassy zu erhöhen. Daher wird im Anschluss an die ANOVA auch der post-hoc Test nach Tukey automatisiert durchgeführt. Ergebnis dieses Tests ist Tabelle c mit den Ergebnissen der paarweisen

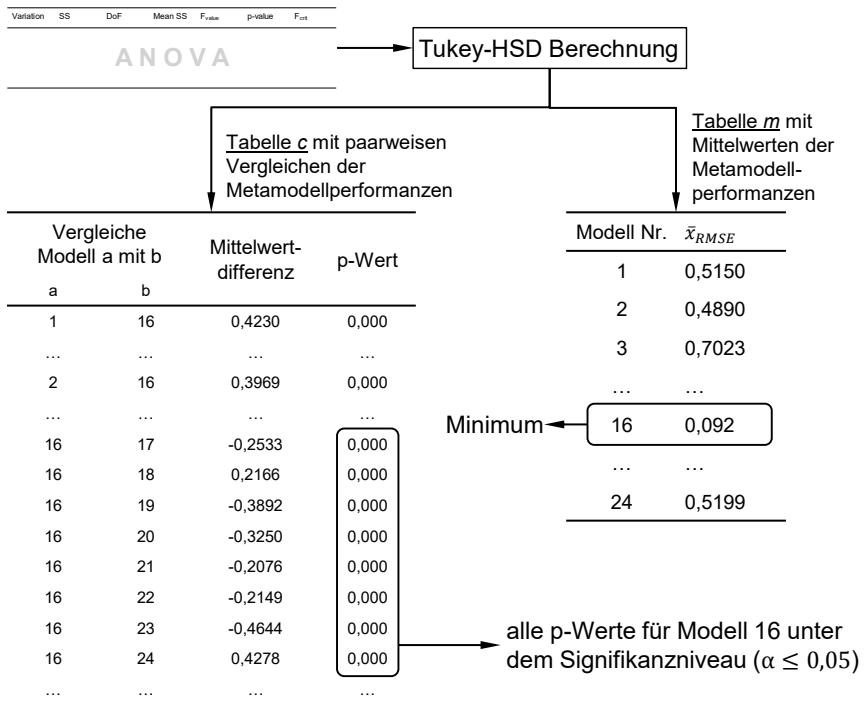


Bild 89: Auswahl des Metamodells Nr. 16 mit dem signifikant geringstem Fehler durch einen post-hoc Test nach Tukey.

Mittelwertvergleiche und Tabelle m mit den Mittelwerten und Varianzen der Performanzvektoren. Beide Tabellen sind in Ausschnitten in Bild 89 dargestellt.

Die automatische Auswahl des bestgeeigneten Modells erfolgt auf Basis der im post-hoc Test nach Tukey erzeugten Tabellen c und m . Aus m wird der niedrigste i -te Eintrag gesucht, im vorliegenden Beispiel $i = 16$. In c muss nun in jeder Zeile, in der der i -te Eintrag mit anderen Werten verglichen wird, der Eintrag für den p-Wert in der letzten Spalte unter dem festgelegten Signifikanzniveau ($\alpha = 0,05$) liegen. Nur wenn beide Bedingungen erfüllt sind, kann das i -te Modell mit dem signifikant niedrigstem Vorhersagefehler ausgewählt werden. Bild 89 zeigt dieses Vorgehen für das Modell 16 des vorliegenden Anwendungsbereichs.

Evaluierung der akquirierten Metamodelle

Aufgrund des ROPE-Prozesses ist davon auszugehen, dass die Schätzungen der Metamodellperformanzen robust sind und als verlässlich angesehen werden können. Eine weiterge-

hende Evaluierung der Modelle ist im Standardbetrieb nicht vorgesehen. Dennoch soll an dieser Stelle das von Slassy als bestgeeignetes Metamodell identifizierte M5P-Modellbaum mit den übrigen 23 Modellen verglichen werden. Dies geschieht nach dem Hold-Out Verfahren (s. Abschnitt 2.4.2) anhand der zehn Datentupel die im Abschnitt 7.2.2 aus dem ursprünglichen Datensatz der Simulationsstudie herausgenommen und somit nicht für das Training der Modelle genutzt wurden.

Die Tupel sowie die einzelnen Metamodelle werden in das Data-Mining Werkzeug Rapidminer® importiert. Da bei den Testdaten die realen Ergebnisse bekannt sind, können daraus die Vorhersagefehler direkt berechnet werden. Bild 90 zeigt die Gegenüberstellung der Modelle anhand der jeweils erreichten RMSE-Werte.

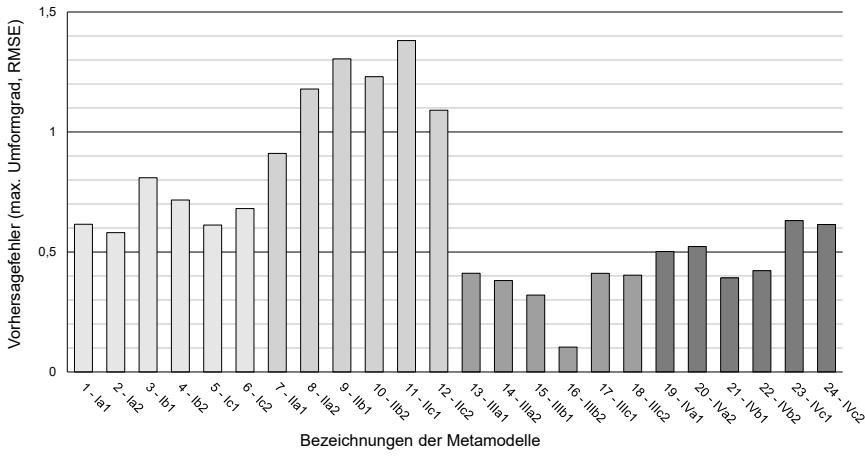


Bild 90: Vorhersagefehler der trainierten Metamodelle des Anwendungsbeispiels, ermittelt im Holdout Verfahren (s. Abschnitt 2.4.2).

Zu erkennen ist, dass das ausgewählte Modell 16 die durch den ROPE-Prozess geschätzte niedrige Vorhersagegenauigkeit beibehält und sich von den anderen Modellen abhebt. Es kann zum einen davon ausgegangen werden, dass der ROPE-Prozess die Forderung nach einer verlässlichen Schätzung der Performanz erfüllt und zum anderen, dass die verfolgte Selbstlernstrategie sich bewährt hat. Zu erkennen sind weiterhin Cluster von Metamodellen mit ähnlichen Vorhersagefehlern, wie die polynomialen Regressionsmodelle. Diese neigen bei höheren Polynomgraden zwar zur besseren Vorhersage der Trainingsdaten, gleichzeitig sinkt jedoch die Vorhersagegüte für neue Daten.

7.4 Analyse des Bauteilentwurfs

Nach Abschluss des Selbstlernprozesses wird das akquirierte Wissen in Form des M5P-Modellbaums in die Wissensbasis des Assistenzsystems eingefügt. Des Weiteren erfolgt eine Wiederholung des Selbstlernprozesses für die beiden verbleibenden Zielgrößen Umformkraft und Kontaktverhältnis. Da die Daten zum Training der Vorhersagemodelle in der multidimensionalen Datenbasis gespeichert sind (s. Abschnitt 6.6.1), kann die Charakteristik der Multidimensionalität auch für die Wissensbasis übernommen werden. Das akquirierte Wissen zur Vorhersage des maximalen Umformgrads und anderer Zielgrößen ist eindeutig dem Bauteil „Napf mit Sperrverzahnung“ und den abgeleiteten Instanzen zugeordnet.

Der Benutzer wechselt zur Vorhersage der Zielgrößen innerhalb der grafischen Oberfläche in den Bereich „Analyse“ (s. Bild 91). Das Assistenzsystem überprüft im Hintergrund die Wissensbasis, ob für das in der Synthese aktive Bauteil bereits Wissens akquiriert wurde. Ist dies nicht der Fall, wird der Benutzer auf den notwendigen Schritt der Datenerhebung hingewiesen. Für das vorliegende Anwendungsbeispiel wird auf Basis der aktuellen Ausprägung der Geometrieparameter eine Ergebnistabelle erzeugt, die neben den Zielgrößen auch die vorhergesagten Werte und den RMSE angibt. Im unteren Bereich des Analysetools wird je nach aktivierter Zielgröße das entsprechende Metamodell angezeigt. Dies lässt bspw. eine Überprüfung der angezeigten Ergebnisse zu. Anzumerken ist, dass im Falle der M5-Modelle die textbasierte Repräsentation und nicht die im Rapidminer® erzeugte grafische Struktur wiedergegeben wird.

Auf Basis der sich ergebenden Werte ($F_{u_{max}} = 12345 \text{ kN}$, $c = 0,644$, $\varphi_{max} = 0,888$) kann eine DfM orientierte Beurteilung des Bauteilentwurfs erfolgen. Eine Möglichkeit wäre diese Beurteilung vom Assistenzsystem übernehmen zu lassen und dem Benutzer durch eine Art Ampelsystem das Ergebnis zu präsentieren. Hierzu müssen die Grenzwerte der Ergebnisgrößen bekannt sein. Für sehr frühe Phasen der Prozessentwicklung ist dies jedoch nicht zielführend oder teilweise nicht möglich. Wie bereits im Kapitel 3 dargestellt, zeichnet sich die Entwicklung der Blechmassivumformung durch eine hohe Dynamik aus. Innerhalb kurzer Zeit können neue Entwicklungen, z. B. in den Bereichen Tribologie, Werkzeugauslegung oder Prozessführung zu einer deutlichen Verschiebung der Prozessgrenzen führen (s. z. B. [MKS+10]). Feste Grenzwerte sind zudem je nach Kennwert nicht bestimmbar. Bereits [Klo07] zeigt, dass für das Verfahren Kaltfließpressen für unterschiedliches Material die maximalen Umformgrade stark variieren und Randbedingungen wie geometriebedingte Fließbehinderungen, Reibung in der Wirkfuge zwischen Werkstück und Werkzeug sowie Grenzen der Belastbarkeit der Werkzeuge berücksichtigt werden müssen. Dies Aspekte müssen bei der DfM-Analyse berücksichtigt werden, liegen jedoch zu großen Teilen im Arbeitsbereich der Prozessentwicklung.

Bei der wissensbasierten Design for Manufacture Analyse soll Slassy daher *assistieren* und die Produkt- und die Prozessentwicklung gleichermaßen in eine sinnvolle Richtung lenken. Konstrukteur und Fertigungsingenieur beurteilen die vorhergesagten Kennwerte gemeinsam und passen ggf. die weitere Entwicklungsstrategien an. Für das vorliegende Beispiel

liegen sowohl die Umformkraft als auch der Umformgrad in realistischen und akzeptablen Größenordnungen (s. [MKOS11]). Das Kontaktverhältnis lässt hingegen auf eine zu geringe Formfüllung schließen, die deutlich unter bereits erreichten Werten liegt (z. B. in [MKS⁺10]). Dadurch kann bspw. die Funktionserfüllung des Nebenformelements „Sperrverzahnung“ nicht gewährleistet werden. Das Bauteil ist insgesamt als nicht fertigungsgerecht zu beurteilen. Für die weitere Prozessentwicklung sollte somit die Erhöhung der Formfüllung im Fokus stehen.

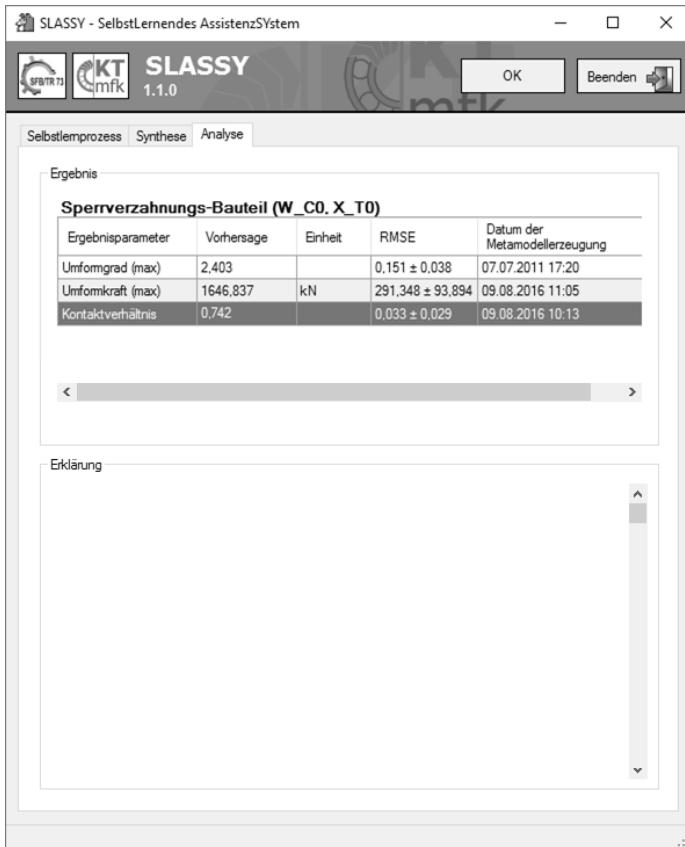


Bild 91: Oberfläche des Assistenzsystems zur Vorhersage der DfM-relevanten Zielgrößen.

Aufgrund der metamodellbasierten Vorhersage der Prozesskennwerte, ermöglicht das Assistenzsystem außerdem einen zügigen Wechsel zwischen Synthese- und Analysetool, um innerhalb kurzer Zeit verschiedene Varianten zu erzeugen und zu analysieren. Hierin besteht ein Vorteil gegenüber der reinen simulationsbasierten Vorgehensweise, bei der ein-

zelne Varianten mehrere Stunden bis Tage zur Analyse benötigen. Das Wissen in Form der Metamodelle kann andererseits auch zur direkten Unterstützung der Simulation genutzt werden. Ergibt die metamodellbasierte Vorhersage relevanter Zielgrößen, dass diese zu weit in unzulässigen Bereichen liegen, kann von einer Simulation abgesehen und eine andere Kombination von Eingangsgrößen untersucht werden.

8 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wird ein Ansatz für ein selbstlernendes Assistenzsystem (Slassy) zur automatischen Akquisition von konstruktionsrelevantem Fertigungswissen und dessen prototypische Umsetzung vorgestellt. Die Selbstlernfähigkeit des Assistenzsystems ermöglicht die Bereitstellung von *Design for Manufacture* (DfM) Wissen in frühen Phasen einer Fertigungsprozess- und Produktentwicklung. Der Selbstlernprozess Technologie der Wissensentdeckung in Datenbanken (engl. Knowledge Discovery in Databases, KDD) zur Analyse von Fertigungsdaten, die im Rahmen von simulativen und/oder experimentellen Studien bei der Prozessentwicklung erzeugt werden. Ausgangspunkt ist die Feststellung, dass einerseits indirekte und direkte Verfahren der Wissensakquisition in frühen Phasen der Fertigungsprozessentwicklung nur bedingt angewandt werden können. Andererseits erfüllen existerende Ansätze für eine automatische oder automatisierte Wissensakquisition wichtige Anforderungen nicht. Daraus wird die Notwendigkeit für ein Assistenzsystem abgeleitet, das den Nutzer vom kritischen Knowledge Engineering Schritt der Wissensakquisition entlastet (Stichwort „Flaschenhals“ nach [Fei81]). Entwicklungskontext von Slassy ist der SFB/TR 73, der sich mit der Erforschung der Blechmassivumformung beschäftigt. Aus diese neuartigen Verfahrensgruppe stammt das Anwendungsbeispiel der vorliegenden Arbeit.

Die Auseinandersetzung mit dem Begriff „Selbstlernen“ ist erforderlich, um eine ausreichende Abgrenzung zu Konzepten mit der Bezeichnung „intelligent“ zu erreichen. Derartige Entwicklungen münden bisher im Einsatz sog. künstlicher neuronaler Netze, die jedoch nicht die in dieser Arbeit gestellten Anforderungen erfüllen (Stichwort „Interpretierbarkeit“). Deshalb werden Strategien zum menschlichen Lernen aus unterschiedlichen Disziplinen (z. B. Konstruktivismus, Behaviorismus) zusammengetragen, analysiert und auf Basis formulierter Anforderungen verglichen. Das Konzept des „Lernens durch Einsicht“ dient als Basis für die Entwicklung des Selbstlernprozesses.

Der Selbstlernprozess beinhaltet zwei wesentliche Kernelemente: die robuste Optimierung und Performanzschätzung (ROPE-Prozess) sowie eine zweistufige Inferenzstatistik zur verlässlichen Identifikation des benötigten DfM-Wissens. Dieses Wissen ist in Form sogenannter Metamodelle repräsentiert. Ein Metamodell ist in der Lage, auf Basis von Eingangsgrößen (z. B. Geometrieparameter) eine Systemantwort (z. B. Umformgrad als Prozessergebnis) vorherzusagen. Wie genau diese Vorhersage ist, wird durch die Performanz ausgedrückt. Deren Wert wird im ROPE-Prozess stochastisch robust geschätzt. Die Optimierung betrifft die Erzeugung bzw. das Training eines Metamodells und ermöglicht eine *optimale* Anpassung des Modells an die zur Verfügung gestellten Daten. Aufgrund der Forderung

nach einer möglichst flexiblen Wissensakquisitionskomponente wird die Strategie verfolgt, eine Vielzahl an Metamodellen zu trainieren. Die Selbstlernkomponente wählt anschließend das bestgeeignete Metamodell aus. Diese Auswahl erfolgt durch ein zweistufiges Verfahren, das Methoden der Inferenzstatistik (statistische Tests) nutzt. Dadurch kann ausgeschlossen werden, dass die Unterschiede zwischen den Metamodellperformanzen stochastischen Ursprungs sind und ein Modell gerade nur „zufällig“ das bestgeeignete ist.

Die Verarbeitung von Fertigungsdaten im KDD-basierten Selbstlernprozess bedingt, dass bei der Systemanwendung Daten vorliegen, deren genaue Struktur zum Zeitpunkt der Systemimplementierung noch unbekannt ist. Eine in dieser Hinsicht größtmögliche Flexibilität wird durch die entwickelte multidimensionale Datenbasis erreicht, die sich am OLAP-Würfel Konzept orientiert. Sie ist in das Produkt- und Prozessdatenmodell (PPDM) integriert, das die Funktion der hierarchischen Gliederung aller Fertigungsdaten über mehrere Ebenen vom Allgemeinen ins Detail erfüllt. Inhalte des Produkt- und Prozessdatenmodells sind Informationen über den Fertigungsprozess die aus Sicht des Produktentwicklers über alle Phasen des Produktlebenszyklus hinweg konstruktionsrelevant sind.

Das Synthesetool des Assistenzsystems ermöglicht die Modellierung des blechmassivumgeformten Bauteils. Der Nutzer greift hierzu auf Haupt- und Nebenformelemente zu und erzeugt einen Gestaltentwurf. Die Bereitstellung der Formelemente erfolgt durch ein kommerzielles CAD-System und die damit verfügbare Feature-Technologie. Zu jeder erzeugten Bauteilvariante kann der Benutzer die relevanten Zielgrößen (z. B. Umformkraft, Umformgrad) und die möglichen Eingangsgrößen (z. B. Geometrieparameter) selbst definieren.

Im Analysewerkzeug wird das automatisch akquirierte Wissen zur Unterstützung des Anwenders bei DfM-Analysen genutzt. Das Assistenzsystem prognostiziert dazu die vom Konstruktions- und vom Fertigungingenieur festgelegten und bauteilspezifischen Zielgrößen auf Basis der aktuellen Bauteilausprägung. Neben dem vorhergesagten Wert einer Zielgröße wird dem Benutzer die Wurzel des mittleren Fehlerquadrats als Schätzwert für die Vorhersagegüte angezeigt. Die Erklärungskomponente ist in das Analysetool integriert und stellt dem Benutzer das jeweils gültige Metamodell dar. So kann die Berechnung des Analysetools bei Bedarf händisch überprüft werden.

Mithilfe der prototypischen Umsetzung und anhand eines Beispiels aus der Entwicklung eines Blechmassivumformprozesses wird das Konzept des selbstlernenden Assistenzsystems verifiziert und das Potential, das sich durch den Einsatz von Methoden der Wissensentdeckung in Datenbanken im Umfeld der Konstruktionstechnik ergibt, offengelegt.

Als sehr vielversprechend wird eine Weiterentwicklung des Synthesewerkzeugs angesehen. In der aktuellen Implementierung definiert der Benutzer ein blechmassivumgeformtes Bauteil und lässt diesen Entwurf von Slassy wissensbasiert analysieren. Ergibt die DfM-Beurteilung, dass keine fertigungsgerechte Gestalt vorliegt, so wechselt der Benutzer in das Synthesetool, ändert die Gestalt des Bauteils anhand eines Geometrieparameters und startet die wissensbasierte *Analyse* ein weiteres Mal. Dieses iterative Vorgehen könnte durch die Entwicklung einer wissensbasierten *Synthese* enorm beschleunigt werden. Ausgangspunkt wären die in der multidimensionalen Wissensbasis hinterlegten Metamodelle zur Vorher-

sage der Ergebnisgrößen eines Blechmassivumformprozesses. Ein nicht-fertigungsgerechter Gestaltentwurf liegt i. d. R. vor, wenn der Wert mindestens einer bauteilspezifischen Zielgrößen außerhalb der zulässigen Grenzen liegt, bspw. bei einer zu hohen Umformkraft. Um einen fertigungsgerechten Entwurf zu ermitteln, ist ein Optimierungsproblem zu lösen: Für einen gegebenen Lösungsraum Ω und eine Bewertungsfunktion $f : \Omega \rightarrow \mathbb{R}$ ist eine Lösung $x \in \Omega$ zu finden, die f minimiert. Da zu erwarten ist, dass pro Bauteilvariante nicht nur eine sondern mehrere Ergebnisgrößen zu berücksichtigen sind, müssen geeignete Methoden der Mehrzieloptimierung identifiziert werden, um diese Problemstellung zu lösen. Erschwerend kommt hinzu, dass bei einer Optimierung diskrete (z. B. Werkzeugkonzept, Armierungskonzept, eingesetzte Ziehringprofil) und kontinuierliche (z. B. geometrische Größen wie Länge, Breite, Winkel) Prozess-, Werkzeug- und Bauteilparameter miteinbezogen werden müssen. Ein erster Lösungsansatz hierfür wurde in [BSSW15] vorgestellt. Dadurch erscheint in naher Zukunft die „Berechnung“ fertigungsgerechter und pareto-optimaler Gestaltentwürfe realistisch. Diese Einschätzung wird von Autoren im Umfeld des Design for Manufacture bzw. des Design for X (z. B. [Sto91; Tic97; van01; Bau09; BDK10]) bisher nicht geteilt.

Aus den Forschungstätigkeiten zur vorliegenden Arbeit im Rahmen des SFB/TR 73 lässt sich außerdem Potential für den Einsatz weiterer KDD-Methoden ableiten. Während die Mehrzieloptimierung sich an den KDD-Prozess des Assistenzsystems anschließt, ließe sich bereits zu Beginn in den Prozess der Datenerhebung eingreifen und „aktiv“ gestalten. Das bisherige „passive“ Vorgehen bei der Datenerhebung besteht aus der Erstellung eines Versuchsplans (z. B. voll-faktoriell, zentral-zusammengesetzt, nach Box-Behnken; s. [SvH10]) der die Parameterstellungen für alle durchzuführenden Versuche vorgibt. Dies hat den Vorteil, dass mit einfachen Methoden (z. B. Mittelwertbildung) u. a. Effekte von einzelnen Faktoren berechnet werden können. Als nachteilig erweist sich jedoch der starre Versuchsplan und der exponentiell steigende Versuchsaufwand bei mehreren Eingangsgrößen. Mit Methoden des maschinellen Lernens ist es möglich, den Versuchsplan flexibel zu gestalten und die Parameterstellungen für jeden weiteren Versuch so zu wählen, dass der Informationsgewinn maximiert wird. Sogenannte Gaußprozesse können hier einen wichtigen Beitrag leisten, da sie, anders als bspw. M5-basierte Metamodelle, die Vorhersagegenauigkeiten lokal berechnen können. Jede Parameterstellung wird so gewählt, dass ein weiterer Versuch an die Stelle mit der höchsten Ungenauigkeit im Wertebereich des Metamodells gesetzt und ein neues Datentupel für das maschinelle Lernen mit Gaußprozessen erzeugt wird. In Vorversuchen konnte der Aufwand für FEM-basierte Strukturanalysen bei gleichbleibender Aussagefähigkeit der Ergebnisse um ca. 20% gesenkt werden [BDW14].

Die präsentierten Methoden und Konzepte haben einen hohen Grad der Realisierung erreicht. Am Lehrstuhl für Konstruktionstechnik (KTmfk) der Universität Erlangen-Nürnberg (FAU) wurden bereits in verschiedenen Forschungsprojekten KDD-Technologien erfolgreich eingesetzt. Zudem verfolgt die Arbeitsgruppe „Assistenzsysteme“ am KTmfk die Vision KDD-Methoden und -Werkzeuge entlang des gesamten Produktentstehungsprozesses zu etablieren [BKK⁺15].

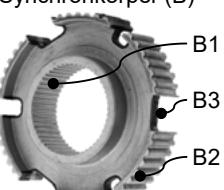
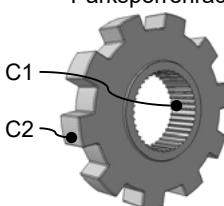
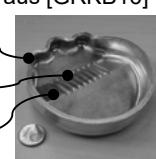
Anhang

Katalog für BMU-Nebenformelemente

Bild 92: Matrixbasierte Umsetzung eines Ordnungsschemas für Nebenformelemente.

		Innenseite		Außenseite			
Formenklasse 1 Erhebung	NFE-Quer schnitt NFE- Grundfläche	Querschnitt ≈ Blechdicke	Querschnitt ≠ Blechdicke	Querschnitt ≈ Blechdicke	Querschnitt ≠ Blechdicke		
	Stirnfläche		A1		E2, E3		
	Ringfläche			D2, C2, F1	A4, D3, F2, F4		
Formenklasse 2 Vertiefung	Stirnfläche						
	Ringfläche	A5		B3			
Formenklasse 3 Verzahnung	Stirnfläche						
	Ringfläche	B1, C1, D4, E1		A3, B2, D4, F3			
Formenklasse 4 Öffnung	Stirnfläche	D1					
	Ringfläche	D1					
Formenklasse 5 Rillierung/ Rändel	Stirnfläche	D5		D5			
	Ringfläche	A2		A2			

Bild 93: Exemplarische Klassierung von Nebenformelementen.

A1	Konusförmige Reibflächen	Kupplungskörper (A)		Synchronring (A)	
A2	Rillierung				
A3	Sperrverzahnung				
A4	Anschlag				
A5	Axialnuten				
B1	Innenverzahnung	Synchronkörper (B)		Parksperrrenrad (C)	
B2	Außenverzahnung				
B3	Aussparungen				
C1	Innenverzahnung			Lamellenträger (D)	
C2	Mitnehmer				
D1	Bohrungen, Langlöcher				
D2	Radienauslauf				
D3	Anschlag				
D4	Radialnuten				
D5	Verzahnung	Stahllamellen (D)			
E1	Mitnehmer				
E2	Verzahnung				
E3	Bodenschräge				
F1	Offener Mitnehmer	Bauteil aus [GRKB16] (E)		TR 73 Demonstrator (F)	
F2	Geschlossener Mitnehmer				
F3	Verzahnung				
F4	Lasche				

Testdatensatz „Concrete Slump Test“

Im Abschnitt 5.2 wird ein frei verfügbarer Testdatensatz zur Erläuterung der Robustheitsproblematik bei Kreuzvalidierungen genutzt. Die Daten wurden von [Yeh07] erhoben und sind über das Machine Learning Repository der University of California verfügbar. Sie umfassen 103 Datentupel mit sieben Attributen und drei Zielgrößen (Label). Die folgende Tabelle fasst den Datensatz zusammen. Die Mittelwerte der Attribute beziehen sich auf den Masseanteil des Stoffes in kg pro Kubikmeter Beton.

Attribut	Mittelwert [$\frac{kg}{m^3}$]	Std.-abw.	Wertebereich
Zement	229,89	78,787	[137,0; 374,0]
Schlacke	77,973	60,461	[0; 193,0]
Flugasche	149,01	85,418	[0; 260,0]
Wasser	197,17	20,208	[160,0; 240,0]
Betonverflüssiger	8,5400	2,8075	[4,4; 19,0]
grobe Bestandteile	883,97	88,391	[708,0; 1049,0]
feine Bestandteile	739,60	63,342	[640,6; 902,0]

Zielgröße	Mittelwert	Std.-abw.	Wertebereich
Setzungsvermögen [cm]	18,048	8,751	[0; 29,0]
Fliessvermögen [cm]	49,611	17,56	[20,0; 78,0]
Druckfestigkeit [MPa]	36,040	7,838	[17,19; 58,53]

Versuchsplan zum Anwendungsbeispiel „Sperrverzahnung“

Die Tabelle enthält die Fertigungsdaten des Anwendungsbeispiels aus Kapitel 7. Abgebrochene Simulationen sind mit **X** im Eintrag des Labels gekennzeichnet, die zufällig gezogenen Tupel der Hold-out Validierung (Abschnitt 7.3.4, S. 149) sind farbig hinterlegt.

ID	X_TO_A0	X_TO_W0	X_TO_L0	X_TO_R2	X_TO_R1	Umformgrad	Kontaktverhältnis	Umformkraft
1	52,50	3,00	3,00	0,30	0,30	2,350	0,492	1630,985
2	55,00	3,00	3,00	0,30	0,30	2,347	0,469	1916,886
3	60,00	3,00	3,00	0,30	0,30	2,635	0,514	1956,531
4	52,50	2,50	3,00	0,30	0,30	2,421	0,492	2033,704
5	55,00	2,50	3,00	0,30	0,30	2,745	0,525	2145,726
6	60,00	2,50	3,00	0,30	0,30	2,635	0,512	1878,541
7	52,50	2,00	3,00	0,30	0,30	2,620	0,540	2039,176
8	55,00	2,00	3,00	0,30	0,30	2,554	0,560	2115,011
9	60,00	2,00	3,00	0,30	0,30	2,511	0,540	2081,601
10	52,50	3,00	2,50	0,30	0,30	1,903	0,533	1708,621
11	55,00	3,00	2,50	0,30	0,30	2,082	0,528	1991,181
12	60,00	3,00	2,50	0,30	0,30	X	X	X
13	52,50	2,50	2,50	0,30	0,30	2,106	0,550	2239,970
14	55,00	2,50	2,50	0,30	0,30	2,096	0,561	2333,052
15	60,00	2,50	2,50	0,30	0,30	2,033	0,568	2293,047
16	52,50	2,00	2,50	0,30	0,30	2,460	0,592	2470,634
17	55,00	2,00	2,50	0,30	0,30	2,328	0,577	2128,243
18	60,00	2,00	2,50	0,30	0,30	2,371	0,601	2567,048
19	52,50	3,00	3,00	0,60	0,30	2,529	0,484	1802,428
20	55,00	3,00	3,00	0,60	0,30	2,500	0,469	1648,750
21	60,00	3,00	3,00	0,60	0,30	2,511	0,454	1701,550
22	52,50	2,50	3,00	0,60	0,30	2,451	0,455	1402,335
23	55,00	2,50	3,00	0,60	0,30	2,738	0,507	1503,678
24	60,00	2,50	3,00	0,60	0,30	2,744	0,517	2122,277
25	52,50	2,00	3,00	0,60	0,30	2,468	0,522	1852,679
26	55,00	2,00	3,00	0,60	0,30	2,387	0,551	1852,582
27	60,00	2,00	3,00	0,60	0,30	2,863	0,539	1961,542
28	52,50	3,00	2,50	0,60	0,30	2,057	0,493	1820,044
29	55,00	3,00	2,50	0,60	0,30	2,020	0,502	1972,949
30	60,00	3,00	2,50	0,60	0,30	2,132	0,512	1668,948
31	52,50	2,50	2,50	0,60	0,30	2,260	0,550	2123,321
32	55,00	2,50	2,50	0,60	0,30	2,541	0,523	1984,317
33	60,00	2,50	2,50	0,60	0,30	2,212	0,554	2211,937
34	52,50	2,00	2,50	0,60	0,30	2,738	0,585	2129,105
35	55,00	2,00	2,50	0,60	0,30	2,413	0,579	2392,372
36	60,00	2,00	2,50	0,60	0,30	X	X	X
37	52,50	3,00	3,00	0,90	0,30	2,986	0,454	1508,354
38	55,00	3,00	3,00	0,90	0,30	2,699	0,479	1568,319
39	60,00	3,00	3,00	0,90	0,30	2,978	0,453	1522,467
40	52,50	2,50	3,00	0,90	0,30	2,620	0,489	1685,638
41	55,00	2,50	3,00	0,90	0,30	2,611	0,468	1414,141
42	60,00	2,50	3,00	0,90	0,30	2,560	0,503	1763,109
43	52,50	2,00	3,00	0,90	0,30	2,956	0,522	1762,261
44	55,00	2,00	3,00	0,90	0,30	3,075	0,551	2119,213
45	60,00	2,00	3,00	0,90	0,30	X	X	X
46	52,50	3,00	2,50	0,90	0,30	2,476	0,510	2101,835
47	55,00	3,00	2,50	0,90	0,30	2,467	0,499	2195,810
48	60,00	3,00	2,50	0,90	0,30	2,055	0,470	1466,784
49	52,50	2,50	2,50	0,90	0,30	2,482	0,548	2074,842
50	55,00	2,50	2,50	0,90	0,30	2,243	0,549	2053,947
51	60,00	2,50	2,50	0,90	0,30	2,498	0,542	2019,635
52	52,50	2,00	2,50	0,90	0,30	2,600	0,590	2192,778
53	55,00	2,00	2,50	0,90	0,30	2,579	0,572	2103,596
54	60,00	2,00	2,50	0,90	0,30	2,552	0,597	2259,143
55	52,50	3,00	3,00	0,30	0,20	2,431	0,482	1823,336
56	55,00	3,00	3,00	0,30	0,20	2,210	0,457	1696,395
57	60,00	3,00	3,00	0,30	0,20	2,335	0,483	1908,838
58	52,50	2,50	3,00	0,30	0,20	2,338	0,459	1676,340
59	55,00	2,50	3,00	0,30	0,20	2,433	0,474	1664,069
60	60,00	2,50	3,00	0,30	0,20	2,579	0,528	1908,443
61	52,50	2,00	3,00	0,30	0,20	2,648	0,548	2063,638
62	55,00	2,00	3,00	0,30	0,20	2,102	0,521	1241,394
63	60,00	2,00	3,00	0,30	0,20	2,824	0,545	2046,446
64	52,50	3,00	2,50	0,30	0,20	2,032	0,478	1700,883
65	55,00	3,00	2,50	0,30	0,20	2,013	0,528	2053,659
66	60,00	3,00	2,50	0,30	0,20	X	X	X
67	52,50	2,50	2,50	0,30	0,20	2,197	0,565	2371,460
68	55,00	2,50	2,50	0,30	0,20	1,984	0,532	1808,147
69	60,00	2,50	2,50	0,30	0,20	2,261	0,543	2133,508
70	52,50	2,00	2,50	0,30	0,20	2,451	0,580	2147,163
71	55,00	2,00	2,50	0,30	0,20	2,315	0,577	2392,178
72	60,00	2,00	2,50	0,30	0,20	2,378	0,607	1911,316
73	52,50	3,00	3,00	0,60	0,20	2,668	0,475	1781,224
74	55,00	3,00	3,00	0,60	0,20	2,646	0,485	1857,025
75	60,00	3,00	3,00	0,60	0,20	2,740	0,465	1669,504

76	52,50	2,50	3,00	0,60	0,20	x	x	x
77	55,00	2,50	3,00	0,60	0,20	2,532	0,511	1598,672
78	60,00	2,50	3,00	0,60	0,20	2,697	0,500	1728,631
79	52,50	2,00	3,00	0,60	0,20	2,686	0,504	2135,320
80	55,00	2,00	3,00	0,60	0,20	2,741	0,557	2040,886
81	60,00	2,00	3,00	0,60	0,20	x	x	x
82	52,50	3,00	2,50	0,60	0,20	2,074	0,510	1987,448
83	55,00	3,00	2,50	0,60	0,20	1,987	0,444	1581,564
84	60,00	3,00	2,50	0,60	0,20	2,065	0,521	2108,311
85	52,50	2,50	2,50	0,60	0,20	2,350	0,552	2316,590
86	55,00	2,50	2,50	0,60	0,20	2,123	0,551	1992,834
87	60,00	2,50	2,50	0,60	0,20	2,200	0,547	2210,723
88	52,50	2,00	2,50	0,60	0,20	2,609	0,562	2414,781
89	55,00	2,00	2,50	0,60	0,20	2,192	0,605	1672,260
90	60,00	2,00	2,50	0,60	0,20	2,335	0,588	2537,416
91	52,50	3,00	3,00	0,90	0,20	2,684	0,476	1768,023
92	55,00	3,00	3,00	0,90	0,20	2,413	0,403	1546,605
93	60,00	3,00	3,00	0,90	0,20	2,499	0,470	1612,715
94	52,50	2,50	3,00	0,90	0,20	2,040	0,501	1609,864
95	55,00	2,50	3,00	0,90	0,20	2,853	0,456	1394,005
96	60,00	2,50	3,00	0,90	0,20	2,558	0,522	2011,617
97	52,50	2,00	3,00	0,90	0,20	2,763	0,556	1948,309
98	55,00	2,00	3,00	0,90	0,20	2,758	0,528	1839,235
99	60,00	2,00	3,00	0,90	0,20	2,737	0,525	1717,717
100	52,50	3,00	2,50	0,90	0,20	2,116	0,495	1850,326
101	55,00	3,00	2,50	0,90	0,20	2,285	0,524	1755,600
102	60,00	3,00	2,50	0,90	0,20	x	x	x
103	52,50	2,50	2,50	0,90	0,20	2,223	0,545	2238,563
104	55,00	2,50	2,50	0,90	0,20	2,402	0,539	1590,917
105	60,00	2,50	2,50	0,90	0,20	2,507	0,563	2125,776
106	52,50	2,00	2,50	0,90	0,20	2,277	0,260	1515,260
107	55,00	2,00	2,50	0,90	0,20	2,673	0,594	2213,169
108	60,00	2,00	2,50	0,90	0,20	2,556	0,587	2451,273
109	52,50	3,00	3,00	0,30	0,60	2,615	0,435	1366,779
110	55,00	3,00	3,00	0,30	0,60	2,721	0,476	1796,780
111	60,00	3,00	3,00	0,30	0,60	2,565	0,439	1679,264
112	52,50	2,50	3,00	0,30	0,60	2,398	0,506	1469,736
113	55,00	2,50	3,00	0,30	0,60	2,491	0,495	1937,021
114	60,00	2,50	3,00	0,30	0,60	2,606	0,470	1684,569
115	52,50	2,00	3,00	0,30	0,60	2,763	0,551	2208,149
116	55,00	2,00	3,00	0,30	0,60	2,435	0,493	1667,874
117	60,00	2,00	3,00	0,30	0,60	2,402	0,500	1987,066
118	52,50	3,00	2,50	0,30	0,60	1,891	0,513	1905,212
119	55,00	3,00	2,50	0,30	0,60	1,992	0,527	1755,479
120	60,00	3,00	2,50	0,30	0,60	2,135	0,561	2087,485
121	52,50	2,50	2,50	0,30	0,60	2,104	0,557	2206,208
122	55,00	2,50	2,50	0,30	0,60	2,294	0,567	2297,481
123	60,00	2,50	2,50	0,30	0,60	1,979	0,570	2299,951
124	52,50	2,00	2,50	0,30	0,60	2,197	0,576	2412,454
125	55,00	2,00	2,50	0,30	0,60	x	x	x
126	60,00	2,00	2,50	0,30	0,60	2,122	0,588	2137,473
127	52,50	3,00	3,00	0,60	0,60	2,573	0,434	1272,754
128	55,00	3,00	3,00	0,60	0,60	2,461	0,425	1233,869
129	60,00	3,00	3,00	0,60	0,60	2,298	0,435	1749,193
130	52,50	2,50	3,00	0,60	0,60	2,501	0,475	1455,003
131	55,00	2,50	3,00	0,60	0,60	2,833	0,448	1832,149
132	60,00	2,50	3,00	0,60	0,60	2,543	0,448	1471,910
133	52,50	2,00	3,00	0,60	0,60	2,539	0,532	1689,889
134	55,00	2,00	3,00	0,60	0,60	2,879	0,522	1848,334
135	60,00	2,00	3,00	0,60	0,60	2,801	0,509	1766,942
136	52,50	3,00	2,50	0,60	0,60	2,142	0,513	2146,990
137	55,00	3,00	2,50	0,60	0,60	2,146	0,483	1764,899
138	60,00	3,00	2,50	0,60	0,60	x	x	x
139	52,50	2,50	2,50	0,60	0,60	2,337	0,513	1921,867
140	55,00	2,50	2,50	0,60	0,60	2,147	0,513	1789,507
141	60,00	2,50	2,50	0,60	0,60	2,328	0,522	1909,704
142	52,50	2,00	2,50	0,60	0,60	2,191	0,581	2210,564
143	55,00	2,00	2,50	0,60	0,60	2,731	0,544	1912,033
144	60,00	2,00	2,50	0,60	0,60	2,199	0,566	2364,252
145	52,50	3,00	3,00	0,90	0,60	2,563	0,453	1609,679
146	55,00	3,00	3,00	0,90	0,60	2,644	0,442	1670,449
147	60,00	3,00	3,00	0,90	0,60	2,621	0,445	1551,660
148	52,50	2,50	3,00	0,90	0,60	2,662	0,487	1878,359
149	55,00	2,50	3,00	0,90	0,60	2,538	0,417	1571,946
150	60,00	2,50	3,00	0,90	0,60	2,475	0,436	1502,510
151	52,50	2,00	3,00	0,90	0,60	2,762	0,519	1923,694
152	55,00	2,00	3,00	0,90	0,60	x	x	x
153	60,00	2,00	3,00	0,90	0,60	2,860	0,519	1791,282
154	52,50	3,00	2,50	0,90	0,60	2,075	0,458	1876,830
155	55,00	3,00	2,50	0,90	0,60	2,364	0,476	1824,410
156	60,00	3,00	2,50	0,90	0,60	2,292	0,448	1632,680
157	52,50	2,50	2,50	0,90	0,60	2,485	0,545	2192,594
158	55,00	2,50	2,50	0,90	0,60	2,215	0,534	2082,904
159	60,00	2,50	2,50	0,90	0,60	2,505	0,536	1959,984
160	52,50	2,00	2,50	0,90	0,60	2,602	0,573	2255,853
161	55,00	2,00	2,50	0,90	0,60	x	x	x
162	60,00	2,00	2,50	0,90	0,60	2,688	0,577	2329,200

Übersicht der implementierten Metamodelle

ID	Kurzname	Parameteroptimierung	Attributauswahl	Metamodelltyp
1	Ia1	(1) Schrittweise	(a) Alle Attribute	
2	Ia2	(2) Evolutionär		
3	Ib1	(1) Schrittweise	(b) Forward Selection	(I) Lineare Regression
4	Ib2	(2) Evolutionär		
5	Ic1	(1) Schrittweise	(a) Backward Elimination	
6	Ic2	(2) Evolutionär		
7	Ia1	(1) Schrittweise	(a) Alle Attribute	
8	Ia2	(2) Evolutionär		
9	Ib1	(1) Schrittweise	(b) Forward Selection	(II) Polynomiale Regression
10	Ib2	(2) Evolutionär		
11	Ic1	(1) Schrittweise	(a) Backward Elimination	
12	Ic2	(2) Evolutionär		
13	Ia1	(1) Schrittweise	(a) Alle Attribute	
14	Ia2	(2) Evolutionär		
15	Ib1	(1) Schrittweise	(b) Forward Selection	(III) Weka M5P Modellbaum
16	Ib2	(2) Evolutionär		
17	Ic1	(1) Schrittweise	(a) Backward Elimination	
18	Ic2	(2) Evolutionär		
19	Ia1	(1) Schrittweise	(a) Alle Attribute	
20	Ia2	(2) Evolutionär		
21	Ib1	(1) Schrittweise	(b) Forward Selection	(IV) Weka M5R Regellerner
22	Ib2	(2) Evolutionär		
23	Ic1	(1) Schrittweise	(a) Backward Elimination	
24	Ic2	(2) Evolutionär		

Tabelle 14: Mittelwertdifferenzen der Performanzvektoren

Modell	\rightarrow	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
	\downarrow	perf.	0.515	0.489	0.702	0.588	0.485	0.523	0.828	1.049	1.139	1.022	1.199	0.914	0.277	0.291	0.231	0.092	0.345	0.308	0.481	0.417	0.299	0.306	0.556	0.519
1	0.515	0	0.026	0.187	0.073	0.03	0.008	0.313	0.534	0.624	0.507	0.684	0.39	0.238	0.224	0.284	0.423	0.17	0.207	0.034	0.098	0.216	0.209	0.041	0.004	
2	0.489	0.026	0	0.213	0.079	0.004	0.034	0.339	0.56	0.65	0.533	0.71	0.425	0.212	0.198	0.258	0.397	0.144	0.357	0.08	0.072	0.19	0.183	0.067	0.03	
3	0.702	0.187	0.213	0	0.114	0.217	0.179	0.126	0.347	0.437	0.32	0.497	0.212	0.425	0.411	0.61	0.357	0.394	0.221	0.285	0.403	0.396	0.146	0.183		
4	0.588	0.073	0.099	0.114	0	0.103	0.065	0.24	0.461	0.551	0.434	0.611	0.326	0.311	0.297	0.357	0.496	0.243	0.28	0.197	0.171	0.289	0.282	0.032	0.069	
5	0.485	0.03	0.004	0.217	0.103	0	0.038	0.343	0.564	0.654	0.537	0.714	0.429	0.208	0.194	0.254	0.393	0.14	0.177	0.004	0.068	0.186	0.179	0.071	0.034	
6	0.523	0.008	0.034	0.179	0.065	0.038	0	0.305	0.526	0.616	0.499	0.676	0.391	0.246	0.232	0.292	0.431	0.178	0.215	0.042	0.106	0.224	0.217	0.033	0.004	
7	0.828	0.313	0.339	0.126	0.24	0.343	0.305	0	0.221	0.311	0.194	0.371	0.086	0.551	0.537	0.597	0.736	0.483	0.52	0.347	0.411	0.529	0.522	0.272	0.309	
8	1.049	0.534	0.56	0.347	0.461	0.564	0.526	0.221	0	0.09	0.027	0.15	0.135	0.772	0.758	0.818	0.957	0.704	0.741	0.568	0.632	0.75	0.743	0.493	0.53	
9	1.139	0.624	0.65	0.437	0.551	0.654	0.616	0.311	0.09	0	0.117	0.06	0.225	0.862	0.848	0.908	1.047	0.794	0.831	0.658	0.722	0.84	0.833	0.583	0.62	
10	1.022	0.507	0.533	0.32	0.434	0.537	0.499	0.194	0.027	0.117	0	0.177	0.108	0.745	0.731	0.791	0.93	0.677	0.714	0.541	0.605	0.723	0.716	0.466	0.503	
11	1.199	0.684	0.71	0.497	0.611	0.714	0.676	0.371	0.15	0.06	0.177	0	0.285	0.922	0.908	0.968	1.107	0.854	0.891	0.718	0.782	0.9	0.893	0.643	0.68	
12	0.914	0.399	0.425	0.212	0.326	0.429	0.391	0.086	0.135	0.225	0.108	0.285	0	0.637	0.623	0.683	0.822	0.569	0.606	0.433	0.497	0.615	0.608	0.358	0.395	
13	0.277	0.238	0.212	0.425	0.311	0.208	0.246	0.551	0.772	0.862	0.745	0.922	0.637	0	0.014	0.046	0.185	0.668	0.031	0.204	0.14	0.022	0.029	0.279	0.242	
14	0.291	0.224	0.198	0.411	0.297	0.194	0.232	0.557	0.758	0.848	0.731	0.908	0.623	0.014	0	0.006	0.199	0.654	0.017	0.19	0.126	0.008	0.015	0.265	0.228	
15	0.231	0.284	0.258	0.471	0.357	0.254	0.292	0.597	0.818	0.908	0.683	0.046	0.006	0	0.139	0.114	0.077	0.25	0.186	0.068	0.075	0.325	0.288			
16	0.092	0.423	0.387	0.61	0.496	0.393	0.431	0.736	0.957	1.047	0.93	1.107	0.822	0.185	0.199	0.139	0	0.253	0.216	0.389	0.325	0.207	0.214	0.464	0.427	
17	0.345	0.17	0.144	0.357	0.243	0.14	0.178	0.483	0.704	0.794	0.677	0.854	0.569	0.068	0.054	0.114	0.253	0	0.637	0.136	0.072	0.046	0.339	0.211	0.174	
18	0.308	0.207	0.181	0.394	0.28	0.177	0.215	0.52	0.741	0.831	0.714	0.891	0.606	0.031	0.017	0.977	0.216	0.037	0	0.173	0.109	0.009	0.002	0.248	0.211	
19	0.481	0.034	0.008	0.221	0.107	0.004	0.042	0.347	0.568	0.658	0.541	0.718	0.433	0.204	0.19	0.25	0.389	0.136	0.173	0	0.064	0.182	0.175	0.075	0.038	
20	0.417	0.098	0.072	0.285	0.171	0.068	0.106	0.411	0.632	0.722	0.605	0.782	0.497	0.14	0.126	0.186	0.325	0.072	0.109	0.064	0	0.118	0.111	0.139	0.102	
21	0.299	0.216	0.19	0.403	0.289	0.186	0.224	0.529	0.75	0.84	0.723	0.9	0.615	0.022	0.008	0.068	0.207	0.046	0.009	0.182	0.118	0	0.007	0.257	0.22	
22	0.306	0.209	0.183	0.396	0.282	0.179	0.217	0.522	0.743	0.833	0.716	0.893	0.608	0.029	0.015	0.075	0.214	0.039	0.002	0.175	0.111	0.007	0	0.25	0.213	
23	0.566	0.041	0.067	0.146	0.032	0.071	0.033	0.272	0.493	0.583	0.466	0.643	0.358	0.279	0.265	0.325	0.464	0.211	0.248	0.075	0.139	0.257	0.25	0	0.037	
24	0.519	0.004	0.03	0.183	0.069	0.034	0.004	0.309	0.53	0.62	0.503	0.68	0.395	0.242	0.228	0.288	0.427	0.174	0.211	0.038	0.102	0.22	0.213	0.037	0	

Literaturverzeichnis

- [AHL92] AMBOS, E.; HARTMANN, R. ; LICHTENBERG, H.: *Fertigungsgerechtes Gestalten von Gussstücken*. Darmstadt: Hoppenstedt Technik Tabellen Verlag, 1992.
- [AN00] ALPAR, P.; NIEDEREICHOLZ, J.: *Data mining im praktischen Einsatz: Verfahren und Anwendungsfälle für Marketing, Vertrieb, Controlling und Kundenunterstützung*. Braunschweig: Vieweg, 2000
- [And00] ANDERL, R.: *STEP - Standard for the exchange of product model data. Eine Einführung*. Stuttgart: Teubner, 2000
- [B+03] BOOS, D. D. u.a.: Introduction to the bootstrap world. In: *Statistical science* 18 (2003), Nr. 2, S. 168–174
- [BA95] BLAND, J. M.; ALTMAN, D. G.: Multiple significance tests: the Bonferroni method. In: *British Medical Journal* 310 (1995), Nr. 6973, S. 170
- [Bac97] BACHSCHUSTER, S.: *Architektur und Konzept zu Realisierung eines produktspezifisch erweiterbaren Konstruktionssystems*. Erlangen, Universität Erlangen-Nürnberg, Dissertation, 1997
- [Ban77] BANDURA, A.: *Social learning theory*. Englewood Cliffs: Prentice Hall, 1977
- [Bau09] BAUER, S.: *Entwicklung eines Werkzeugs zur Unterstützung multikriterieller Entscheidungen im Kontext des Design for X*. Düsseldorf: VDI, 2009. – ISBN 9783183404018
- [Bau14] BAUERNHANSL, T.: *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien und Migration*. Wiesbaden: Springer Vieweg, 2014
- [BB99] BOOSE, J. H.; BRADSHAW, J. M.: Expertise transfer and complex problems: using AQUINAS as a knowledge-acquisition workbench for knowledge-based systems. In: *International journal of human-computer studies* 51 (1999), Nr. 2, S. 453–478
- [BB09] BUCALEM, M. L.; BATHE, K.-J.: *The mechanics of solids and structures - Hierarchical Modeling and the Finite Element Solution*. New York: Springer, 2009. – ISBN 9783540263319
- [BBB+83] BUCHANAN, B. G.; BARSTOW, D.; BECHTAL, R.; BENNETT, J.; CLANCEY, W.; KULIKOWSKI, C.; MITCHELL, T. ; WATERMAN, D. A.: Constructing an expert system. In: *Building expert systems* 50 (1983), S. 127–167
- [BBVe15] BEHRENS, B.-A.; BOUGUECHA, A.; VUCETIC, M. ; ET AL.: Numerical and Experimental Investigations of Multistage Sheet-Bulk Metal Forming Process with Compound Press Tools. In: *Key Engineering Materials* 651-653 (2015), S. 1153–1158
- [BD11] BETHGE, P.; DWORSCHAK, M.: Da ist niemand zu Hause. Interview mit dem Computerpionier David Gelernter über Künstliche Intelligenz. In: *Der Spiegel* (2011), Nr. 8. <https://magazin.spiegel.de/SP/2011/8/77108594/index.html>
- [BDK10] BOOTHROYD, G.; DEWHURST, P. ; KNIGHT, W. A.: *Product Design for Manufacture and Assembly, Third Edition*. 3. Auflage. Hoboken: CRC Press, 2010

- [BDW14] BREITSPRECHER, T.; DINGFELDER, C. ; WARTZACK, S.: Ein Ansatz zur adaptiven Simulationsdatenerhebung. In: BRÖKEL, K. (Hrsg.); FELDHUSEN, J. (Hrsg.); GROTE, K.-H. (Hrsg.); RIEG, F. (Hrsg.) ; STELZER, R. (Hrsg.): *Tagungsband 12. Gemeinsames Kolloquium Konstruktionstechnik 2014*. Bayreuth: Universität Bayreuth, 2014, S. 461–470
- [BEC97] BORNDORFF-ECCARIUS, S.: *Rechnergestützte Wissensakquisition für wissensbasierte Diagnosesysteme im Bereich dynamischer technischer Systeme*. Berlin: Infix, 1997
- [BF89] BUTTENBRUCH, P.; FRICK, D.: *Kognitionspychologische Aspekte der Wissensakquisition*. Univ, 1989 (Diskussionsbeiträge des Fachbereichs Wirtschaftswissenschaft der Universität Duisburg - Gesamthochschule). <https://books.google.de/books?id=KwW1ywAACAAJ>
- [BFSO84] BREIMAN, L.; FRIEDMAN, J.; STONE, C. ; OLSHEN, R.: *Classification and regression trees*. Boca Raton: CRC Press, 1984
- [BG89] BOOSE, J. H.; GAINES, B. R.: Knowledge acquisition for knowledge-based systems: Notes on the state-of-the-art. In: *Machine Learning* 4 (1989), Nr. 3-4, S. 377–394
- [BGSS03] BECKER, V.; GERATHEWOHL, R.; SCHMIDT, A. ; STEINBERG, I. E.: *Kfz-Getriebe mit Lamellenkupplung*. Pat.-Nr.: EP 1422430 A1. 2003
- [BHH⁺11] BREITSPRECHER, T.; HENSE, R.; HAUER, F.; WARTZACK, S.; BIERMANN, D. ; WILLNER, K.: Akquise von heuristischem Wissen für die Prädiktion von Oberflächenstrukturen beim Fräsen mit regenerativen Werkzeugschwingungen. In: MERKLEIN, M. (Hrsg.): *Tagungsband zum 1. Workshop Blechmassivumformung*. Bamberg: Meisenbach, 2011, S. 1–20
- [BHV⁺15] BEHRENS, B.-A.; HÜBNER, S.; VUCETIC, M.; KOCH, S. ; ET AL.: Prozessauslegung und Untersuchung von geschliffenen und CrAlN / CrAlCN-beschichteten Werkzeugoberflächen mittels schwingungsüberlagertem Flachstauchversuch für die Blechmassivumformung. In: TEKKAYA, A. E. (Hrsg.); LIEWALD, M. (Hrsg.); MERKLEIN, M. (Hrsg.) ; BEHRENS, B.-A. (Hrsg.): *Tagungsband zum 3. Workshop Blechmassivumformung*. Aachen: Shaker, 2015, S. 179–192
- [Bim05] BIMAZUBUTE, R.: *Die Nachbereitung von Experteninterviews im expertenzentrierten Wissensmanagement*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 2005
- [BJS⁺11] BRECHER, C.; JESCHKE, S.; SCHUH, G.; AGHASSI, S. ; ET AL.: Integrative Produktionstechnik für Hochlohnländer. In: BRECHER, C. (Hrsg.): *Integrative Produktionstechnik für Hochlohnländer* Bd. 1. Berlin: Springer, 2011, S. 17–81
- [BKI08] BEIERLE, C.; KERN-ISBERNER, G.: *Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen*. 4. Auflage. Wiesbaden: Vieweg & Teubner, 2008
- [BKI14] BEIERLE, C.; KERN-ISBERNER, G.: *Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen*. 5. Auflage. Wiesbaden: Springer Fachmedien, 2014
- [BKK⁺15] BREITSPRECHER, T.; KESTEL, P.; KÜSTNER, C.; SPRÜGEL, T. ; WARTZACK, S.: Einsatz von Data-Mining in modernen Produktentstehungsprozessen. In: *ZWF* 11 (2015), S. 744–750
- [Boc75] BOCK, R. D.: *Multivariate statistical methods in behavioral research*. New York: McGraw-Hill, 1975 (McGraw-Hill series in psychology)

- [Box79] BOX, G. E. P.: Robustness in the strategy of scientific model building. In: *Robustness in statistics* 1 (1979), S. 201–236
- [Bra72] BRANIN JR, F. H.: Widely convergent method for finding multiple solutions of simultaneous nonlinear equations. In: *IBM Journal of Research and Development* 16 (1972), Nr. 5, S. 504–522
- [Bra08] BRASS, E.: *Konstruieren mit CATIA V5: Methodik der parametrisch-assoziativen Flächenmodellierung*. 4. Auflage. München: Carl Hanser, 2008
- [Bre99a] BREITLING, F.: Wissensbasierte Konstruktion - Das Berufsbild des Ingenieurs im Wandel. In: *Kolloquium Integrierte Produktentwicklung Universität Magdeburg*, 1999, S. 1–10
- [Bré99b] BRÉMAUD, P.: *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*. New York: Springer, 1999
- [BS92] BREIMAN, L.; SPECTOR, P.: Submodel selection and evaluation in regression. The X-random case. In: *International statistical review/revue internationale de Statistique* (1992), S. 291–319
- [BS10] BORTZ, J.; SCHUSTER, C.: *Statistik für Human- und Sozialwissenschaftler*. 7. Auflage. Berlin: Springer, 2010
- [BSSW15] BREITSPRECHER, T.; SAUER, C.; SPERBER, C. ; WARTZACK, S.: Design-for-manufacture of sheet-bulk metal formed parts. In: WEBER, C. (Hrsg.); HUSUNG, S. (Hrsg.); CASCINI, G. (Hrsg.); CANTAMESSA, M. (Hrsg.) ; MARJANOVIC, D. (Hrsg.): *20th International Conference on Engineering Design (ICED15)*. Glasgow, Scotland: Design Society, 2015 (Proceedings of the 20th International Conference on Engineering Design), S. 183–192
- [BW13] BREITSPRECHER, T.; WARTZACK, S.: A classification system for secondary design features for the use within sheet-bulk metal forming. In: LINDEMANN, U. (Hrsg.); VENKATARAMAN, S. (Hrsg.); KIM, Y. S. (Hrsg.) ; LEE, S. W. (Hrsg.): *ICED 13, the 19th International Conference on Engineering Design* Bd. 75-5. Design Society, 2013, S. 39–48
- [BW14] BREITSPRECHER, T.; WARTZACK, S.: Der Einsatz von Gaußprozessen zur Beschleunigung der automatischen Wissensakquisition. In: KRAUSE, D. (Hrsg.): *Design for X*. Hamburg: TuTech-Verl., 2014, S. 227–236
- [CD82] COWLES, M.; DAVIS, C.: On the origins of the 0.05 level of statistical significance. In: *American Psychologist* 37 (1982), Nr. 5, S. 553
- [CD88] CUNNINGHAM, J. J.; DIXON: Designing with features: the origin of features. In: *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition* Bd. 1, 1988, S. 237–243
- [CD97] CHAUDHURI, S.; DAYAL, U.: An overview of data warehousing and OLAP technology. In: *ACM Sigmod record* 26 (1997), Nr. 1, S. 65–74
- [CFPS99] CHAN, P. K.; FAN, W.; PRODROMIDIS, A. L. ; STOLFO, S. J.: Distributed data mining in credit card fraud detection. In: *Intelligent Systems and their Applications, IEEE* 14 (1999), Nr. 6, S. 67–74
- [CGH⁺10] CZICHOS, H.; GERSCHWILER, K.; HABIG, K. H.; KLEINLEIN, K. ; ET AL.: *Tribologie-Handbuch: Tribometrie, Tribomaterialien, Tribotechnik*. 3. Auflage. Wiesbaden: Vieweg & Teubner, 2010

- [CL01] CHUDZICK, H.; LAUKAMP, S.: Neue Stähle in Querlenker und Verbundlenkerachse. In: *ATZ-Automobiltechnische Zeitschrift* 103 (2001), Nr. 10, S. 964–968
- [CL07] COOPER, D.; LA ROCCA, G.: Knowledge-based techniques for developing engineering applications in the 21st century. In: AMERICAN INSTITUTE OF AERONAUTICS AND ASTRONAUTICS (Hrsg.): *7th AIAA ATIO Conference, Belfast, Northern Ireland*, 2007, S. 1–22
- [Cla85] CLANCEY, W. J.: Heuristic classification. In: *Artificial intelligence* 27 (1985), Nr. 3, S. 289–350
- [Cor01] CORMEN, T. H.: *Introduction to algorithms*. 2. Auflage. Cambridge: MIT Press, 2001
- [CPSK07] CIOS, K. J.; PEDRYCZ, W.; SWINIARSKI, R. W. ; KURGAN, L. A.: *Data mining: A knowledge discovery approach*. New York: Springer, 2007
- [Cre91] CREMERS, A. B.: *Expertensysteme für die Planung der Produktion*. Verlag TÜV Rheinland, 1991
- [DB10] DOEGE, E.; BEHRENS, B.-A.: *Handbuch Umformtechnik: Grundlagen, Technologien, Maschinen*. Berlin: Springer, 2010
- [Den94] DENGEL, A.: *Meyers Forum*. Bd. 13: *Künstliche Intelligenz - Allgemeine Prinzipien und Modelle*. Mannheim: BI-Taschenbuchverl, 1994. – ISBN 3411102713
- [DGB07] DUBITZKY, W.; GRANZOW, M. ; BERRAR, D. P.: *Fundamentals of data mining in genomics and proteomics*. Springer Science & Business Media, 2007
- [Dia89] DIAPER, D.: *Knowledge elicitation - Principles, techniques, and applications*. Chichester: Ellis Horwood, 1989
- [DIN03] DIN 8580 - DEUTSCHES INSTITUT FÜR NORMUNG: *Fertigungsverfahren - Begriffe und Einteilung*. Berlin and Köln, 2003
- [DL09] DEGISCHER, H.-P.; LÜFTL, S.: *Leichtbau: Prinzipien, Werkstoffauswahl und Fertigungsvarianten*. Weinheim: Wiley, 2009
- [DLK08] DANJOU, S.; LUPA, N. ; KOEHLER, P.: Approach for automated product modeling using knowledge-based design features. In: *Computer-Aided Design and Applications* 5 (2008), Nr. 5, S. 622–629
- [DLR77] DEMPSTER, A. P.; LAIRD, N. M. ; RUBIN, D. B.: Maximum likelihood from incomplete data via the EM algorithm. In: *Journal of the royal statistical society. Series B (methodological)* (1977), S. 1–38
- [Dow98] DOWSON, D.: *History of tribology*. 2. Auflage. London: Professional Engineering, 1998
- [DSW08] DÜRR, H.; SCHNEIDER, T. ; WENSERSKI, J.: Automatisierte Akquisition von Fertigungswissen im Werkzeug- und Formenbau. In: *ZWF: Zeitschrift für Wirtschaftlichen Fabrikbetrieb* 103 (2008), Nr. 4, S. 209
- [EM13] EHRLENSPIEL, K.; MEERKAMM, H.: *Integrierte Produktentwicklung: Denkabläufe, Methodeneinsatz, Zusammenarbeit*. 5. München: Hanser, 2013
- [Ert13] ERTEL, W.: *Grundkurs Künstliche Intelligenz. Eine praxisorientierte Einführung*. 3. Auflage. Wiesbaden: Springer Vieweg, 2013
- [ES00] ESTER, M.; SANDER, J.: *Knowledge discovery in databases: Techniken und Anwendungen*. Berlin: Springer, 2000

- [ES05] EVERSHIM, W.; SCHUH, G.: *Integrierte Produkt- und Prozessgestaltung*. Berlin: Springer, 2005
- [ESM⁺11] ELEZI, F.; SHARAFI, A.; MIRSON, A.; PETRA, W.; KRCMAR, H. ; LINDEMANN, U.: A Knowledge Discovery in Databases (KDD) Approach for Extracting Causes of Iterations in Engineering Change Orders. In: *International Conference on Design Engineering, Computers and Information in Engineering*, ASME, 2011, S. 1–10
- [Eve02] EVERSHIM, W.: *Organisation in der Produktionstechnik 3: Arbeitsvorbereitung*. Berlin: Springer, 2002
- [FD95] FELDMAN, R.; DAGAN, I.: Knowledge Discovery in Textual Databases (KDT). In: FAYYAD, U. M. (Hrsg.); UTHURUSAMY, R. (Hrsg.): *First International Conference on Knowledge Discovery & Data Mining* Bd. 95. Menlo Park, Calif.: AAAI Press, 1995, S. 112–117
- [Fei81] FEIGENBAUM, E. A.: Expert systems in the 1980s. In: *State of the art report on machine intelligence*. Maidenhead: Pergamon-Infotech (1981)
- [FF03] FUNKAT, A.-K.; FUNKAT, G.: *Prozessbasiertes Knowledge engineering in medizinischen Problemdomänen*, Technische Universität Ilmenau, Dissertation, 2003
- [FF11] FRANKEN, R.; FRANKEN, S.: *Integriertes Wissens- und Innovationsmanagement*. Wiesbaden: Gabler, 2011
- [FG13] FELDHUSEN, J.; GROTE, K.-H.: *Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung*. 8. Auflage. Berlin, Heidelberg: Springer Vieweg, 2013
- [FGW02] FAYYAD, U.; GRINSTEIN, G. ; WIERSE, A.: *Information visualization in data mining and knowledge discovery*. San Francisco: Morgan Kaufmann Publishers, 2002
- [FH20] FISCHER, S. C.; HULL, C. L.: Quantitative aspects of the evolution of concepts. In: *Psychological monographs* (1920), Nr. 123
- [FHT96] FAHRMEIR, L.; HAMERLE, A. ; TUTZ, G.: *Multivariate statistische Verfahren*. Walter de Gruyter GmbH & Co KG, 1996
- [Fin90] FINKENWIRTH, K.-W.: *Fertigungsgerecht Konstruieren mit CAD: Konzept eines Konstruktionssystems zur Informationsverarbeitung mit CAD- Systemen*. Erlangen, Universität Erlangen-Nürnberg, Dissertation, 1990
- [Fis35] FISHER, R. A.: *The Design of experiments*. Edinburgh: Oliver & Boyd, 1935
- [Fis11] FISCHER, U.: *Tabellenbuch Metall*. 45. Auflage. Haan-Gruiten: Europa-Lehrmittel Nourney, 2011
- [FKKS08] FELLNER, D. W.; KAMPS, T.; KOHLHAMMER, J. ; STRICKER, A.: Vorsprung durch Wissen. In: *ZWF* 4 (2008), S. 205–208
- [FKPT07] FAHRMEIR, L.; KÜNSTLER, R.; PIGEOT, I. ; TUTZ, G.: *Statistik. Der Weg zur Datenanalyse*. 6. Auflage. Berlin: Springer, 2007. <http://dx.doi.org/10.1007/978-3-540-69739-8>. <http://dx.doi.org/10.1007/978-3-540-69739-8>
- [FPSS96] FAYYAD, U.; PIATETSKY-SHAPIRO, G. ; SMYTH, P.: From Data Mining to Knowledge Discovery in Databases. In: *AI Magazine* 17 (1996), Nr. 3, S. 37–54
- [Fri98] FRICK, D.: *Die Akquisition betriebswirtschaftlichen Wissens zum Aufbau von wissensbasierten Entscheidungsunterstützungssystemen*. Lang, 1998

- [Fri13] FRIEDRICH, H. E.: *Leichtbau in der Fahrzeugtechnik*. Wiesbaden: Springer Vieweg, 2013
- [GAPW93] GRABOWSKI, H.; ANDERL, R.; POLLY, A. ; WARNECKE, H.: *Integriertes Produktmodell*. Berlin: Beuth, 1993
- [GB88] GERBER, H.; BURGERMEISTER, W.: Wissensverarbeitung im Konstruktionsprozess. In: *VDI-Berichte* (1988), Nr. 700.2, S. 277–297
- [GHEM15] GRÖBEL, D.; HILDENBRAND, P.; ENGEL, U. ; MERKLEIN, M.: Influence of tailored blanks on forming of cold forged functional elements in a sheet bulk metal forming process. In: LÜNGEN, H. B. (Hrsg.): *METEC and 2nd ESTAT*, 2015, S. 1–8
- [GL10] GOYVAERTS, J.; LEVITHAN, S.: *Reguläre Ausdrücke Kochbuch*. Köln: O'Reilly, 2010
- [GMS07] GROTH, C.; MÜLLER, G. ; STELZMANN, U.: *FEM für Praktiker*. 8. Auflage. Renningen: Expert-Verlag, 2007
- [Göb92] GÖBLER, T.: *Modellbasierte Wissensakquisition zur rechnerunterstützten Wissensbereitstellung für den Anwendungsbereich Entwicklung und Konstruktion*. München: Hanser, 1992
- [Gra76] GRAYER, A. R.: *A Computer Link Between Design and Manufacture*. Camebridge (UK), University of Cambridge, Dissertation, 1976. <https://books.google.de/books?id=idAgwPAACAAJ>
- [Gre95] GRESKA, W.: *Wissensbasierte Analyse und Klassifizierung von Blechteilen*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 1995
- [GSH⁺16] GRÖBEL, D.; SCHULTE, R.; HILDENBRAND, P.; LECHNER, M.; ENGEL, U.; SIECZKAREK, P.; WERNICKE, S.; GIES, S.; TEKKAYA, A. E.; BEHRENS, B.-A.; HÜBNER, S.; VUCETIC, M.; KOCH, S. ; MERKLEIN, M.: Manufacturing of functional elements by sheet-bulk metal forming processes. In: *Production Engineering* 10 (2016), Nr. 1, S. 63–80. <http://dx.doi.org/10.1007/s11740-016-0662-y>. – DOI 10.1007/s11740-016-0662-y
- [GUW04] GÖHNER, P.; URBANO, P. ; WAGNER, T.: Softwareagenten–Einführung und Überblick über eine alternative Art der Softwareentwicklung. In: *atp Automatisierungs-technische Praxis* 4 (2004), S. 1–31
- [Haa95] HAASIS, S.: *Wissens- und featurebasierte Unterstützung der Konstruktion von Stirnradgetrieben unter besonderer Berücksichtigung des Gußgehäuses*. Düsseldorf and Stuttgart, Fachhochschule für Technik Esslingen, Dissertation, 1995
- [Har14] HARTBRICH, I.: *Menschen haben es nicht so mit den Nachkommastellen. Interview mit Prof. Katharina Morik*: In: *VDI Nachrichten, Ausgabe 31 (2014)*. Dortmund, 01.08.2014
- [HE13] HOFFMANN, J.; ENGELKAMP, J.: *Lern- und Gedächtnispsychologie*. Heidelberg: Springer, 2013
- [Hel74] HELTEN, E.: *Statistische Forschungsstrategien*. Oldenbourg Verlag, 1974
- [Hen13] HENSE, R.: Using Chatter Vibrations in Milling to Create Tribological Functional Surfaces for Sheet-Bulk Metal Forming Tools. In: *Advanced Materials Research* 769 (2013), S. 263–270. <http://dx.doi.org/10.4028/www.scientific.net/AMR.769.263>. – DOI 10.4028/www.scientific.net/AMR.769.263. – ISSN 1662-8985
- [Her97] HERRMANN, J.: *Maschinelles Lernen und wissensbasierte Systeme: Systematische Einführung mit praxisorientierten Fallstudien*. Berlin: Springer, 1997

- [Her10] HERRMANN, C.: *Ganzheitliches Life Cycle Management. Nachhaltigkeit und Lebenszyklusorientierung in Unternehmen*. Berlin and Heidelberg: Springer, 2010
- [Her13] HERPER, J.: *Tribologische Untersuchungen von verschleiß- und reibungsarmen nanostrukturierten bionischen PVD-Beschichtungen*. Dortmund, TU Dortmund, Dissertation, 2013
- [Het14] HETZNER, H.: *Systematische Entwicklung amorpher Kohlenstoffschichten unter Berücksichtigung der Anforderungen der Blechmassivumformung*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 2014
- [Heu29] HEUVERS, A.: Was hat der Stahlgießer dem Konstrukteur über Lunker- und Rißbildung zu sagen? In: *Stahl und Eisen Zeitschrift für das Deutsche Eisenhüttenwesen* 49 (1929), Nr. 35, S. 1249–1256
- [HHF99] HOLMES, G.; HALL, M. ; FRANK, E.: Generating Rule Sets from Model Trees. In: *Working Paper Series of the Computer Science Department at Waikato University* 1999 (1999), Nr. 2, S. 1–10. – ISSN 1170–487X
- [Hin15] HINSCH, M.: *Die neue ISO 9001: 2015 - Ein Praxis-Ratgeber für die Normenumstellung*. 2. Auflage. Berlin, Heidelberg: Springer, 2015
- [Hit12] HITZ, R.: *Metamodellbasierte Methodenplanung von Umform- und Feinschneideprozessen*. Zürich, ETH Zürich, Dissertation, 2012
- [HK14] HOFMANN, M.; KLINKENBERG, R.: *Chapman & Hall/CRC data mining and knowledge discovery series*. Bd. 33: *RapidMiner - Data mining use cases and business analytics applications*. Boca Raton: CRC Press, 2014. – ISBN 978-1-4822-0550-3
- [HKL⁺15] HUBER, E.; KUSTER, J.; LIPPmann, R.; SCHMID, A.; SCHNEIDER, E.; WITSCHI, U. ; WÜST, R.: *Project Management Handbook*. Berlin: Springer, 2015
- [HKMW01] HIPPNER, H.; KÜSTERS, U.; MEYER, M. ; WILDE, K.: *Handbuch Data Mining im Marketing: Knowledge discovery in marketing databases*. Wiesbaden: Vieweg, 2001
- [HKP12] HAN, J.; KAMBER, M. ; PEI, J.: *Data mining: Concepts and techniques*. 3. Auflage. Amsterdam: Elsevier und Morgan Kaufmann, 2012
- [HKR70] HAHN, R.; KUNERTH, W. ; ROCKMANN, K.: Die Teileklassifizierung. In: *RKW Handbuch* (1970), Nr. 21
- [HKS02] HÄNLE, I. U.; KALKE, I. S. ; SEETHALER, D.-I. L.: Metallische Leichtbauwerkstoffe und Fertigungstechnologien im Automobilbau. In: *ATZ-Automobiltechnische Zeitschrift* 104 (2002), Nr. 3, S. 268–275
- [HLW06] HEROLD, H.; LURZ, B. ; WOHLRAB, J.: *Grundlagen der Informatik. Praktisch, technisch, theoretisch*. München: Pearson Studium, 2006
- [HMH10] HITZ, R.; MANOPULO, N. ; HORA, P.: Simulation and Knowledge Based Process Planning Through the Use of Metamodels. In: HORA, P. (Hrsg.): *Proceedings of NUMIFORM 2010* Bd. 1252. Melville, NY: American Institute of Physics, 2010 (AIP conference proceedings), S. 888–895
- [HNP05] HOTHÖ, A.; NÜRNBERGER, A. ; PAASS, G.: A Brief Survey of Text Mining. In: *Zeitschrift für Computerlinguistik und Sprachtechnologie* 20 (2005), Nr. 1, S. 19–62
- [Hop92] HOPPE, U.: *Methoden des knowledge engineering: Ein Expertensystem für das Wertpapiergeschäft in Banken*. Wiesbaden: Deutscher Universitätsverlag, 1992
- [Hua96] HUANG, G. Q.: *Design for X - Concurrent Engineering Imperatives*. London: Chapman & Hall, 1996

- [Hua08] HUA, J.: Study on knowledge acquisition techniques. In: *Intelligent Information Technology Application, 2008. IIITA '08. Second International Symposium on* Bd. 1, 2008, S. 181–185
- [Hub90] HUBKA, V.: Design for Properties - Eigenschaftsgerechtes Konstruieren. In: MEERKAMM, H. (Hrsg.): *Symposium Design for X*, 1990, S. 1–7
- [Hub95] HUBKA, V.: Design for X - DfX. In: MEERKAMM, H. (Hrsg.): *Beiträge zum 6. Symposium Design for X*, 1995, S. 1–6
- [IS10] ILSCHNER, B.; SINGER, R. F.: *Werkstoffwissenschaften und Fertigungstechnik: Eigenschaften, Vorgänge, Technologien*. 5. Auflage. Berlin: Springer, 2010
- [ISO12] ISO 14306: *ISO 14306:2012-12 - Industrial automation systems and integration. JT file format specification for 3D visualization*. Genf, 2012
- [JEWCO03] JARRATT, T. A. W.; ECKERT, C. M.; WEEKS, R. ; CLARKSON, P. J.: Environmental legislation as a driver of design. In: FOLKESON, A. (Hrsg.); GRALEN, K. (Hrsg.); NORELL, M. (Hrsg.) ; SELLGREN, U. (Hrsg.): *14th International Conference on Engineering Design*, Design Society, 2003, S. 231–240
- [JL13] JUNG, M.; LANGER, U.: *Methode der finiten Elemente für Ingenieure: Eine Einführung in die numerischen Grundlagen und Computersimulation*. 2., überarb. und erweiterte Aufl. Wiesbaden: Springer Vieweg, 2013 (SpringerLink : Bücher)
- [Joh01] JOHNSON, R. W.: An introduction to the bootstrap. In: *Teaching Statistics* 23 (2001), Nr. 2, S. 49–54
- [JS11] JARECKI, U.; SCHULZ, H. J.: *Dubbel Mathematik: Eine kompakte Ingenieurmathematik zum Nachschlagen*. Berlin: Springer, 2011
- [Jur04] JURKLIES, I.: *Generierung und Bewertung von Prozessketten für den Werkzeug- und Formenbau*. Aachen: Shaker, 2004
- [Kar88] KARBACH, W.: Methoden und Techniken des Knowledge Engineering. In: *Arbeitspapiere der GMD* (1988), Nr. 338, S. 126
- [KB90] KOLLER, R.; BERNS, S.: Strukturierung von Konstruktionswissen. In: *Konstruktion* 42 (1990), S. 85–90
- [KBK⁺13] KRUSE, R.; BORGELT, C.; KLAWONN, F.; MOEWES, C.; STEINBRECHER, M. ; HELD, P.: *Computational intelligence. A methodological introduction*. London, New York: Springer, 2013
- [KBR10] KRATZER, M.; BINZ, H. ; ROTH, D.: Wissensstruktur zur Integration von Konstruktionswissen in agentenbasierte Unterstützungsstysteme. In: KRAUSE, D. (Hrsg.); PAETZOLD, K. (Hrsg.) ; WARTZACK, S. (Hrsg.): *Design for X*. Hamburg: TuTech-Verl, 2010. – ISBN 978-3-941492-23-3, S. 235–246
- [KD15] KOTU, V.; DESHPANDE, B.: *Predictive analytics and data mining: Concepts and practice with RapidMiner*. Waltham, MA: Morgan Kaufmann, 2015. – ISBN 978-0-12-801460-8
- [Kir07] KIRCHNER, E.: *Leistungsübertragung in Fahrzeuggetrieben. Grundlagen der Auslegung, Entwicklung und Validierung von Fahrzeuggetrieben und deren Komponenten*. Berlin: Springer, 2007
- [KK12a] KIESEL, A.; KOCH, I.: *Lernen. Grundlagen der Lernpsychologie*. Wiesbaden: Verlag für Sozialwissenschaften, 2012

- [KK12b] KLÜVER, C.; KLÜVER, J.: *Lehren, Lernen und Fachdidaktik: Theorie, Praxis und Forschungsergebnisse am Beispiel der Informatik*. Wiesbaden: Vieweg+Teubner, 2012
- [KL90] KARBACH, W.; LINSTER, M.: *Wissensakquisition für Expertensysteme. Techniken, Modelle und Softwarewerkzeuge*. München: Hanser, 1990
- [Kle99] KLEIN, B.: Entwicklung einer leichten Sitzstruktur in Aluminiumbauweise. In: *ATZ-Automobiltechnische Zeitschrift* 101 (1999), Nr. 2, S. 74–79
- [Kle11] KLEUKER, S.: *Grundkurs Datenbankentwicklung: Von der Anforderungsanalyse zur komplexen Datenbankanfrage*. 2. Auflage. Wiesbaden: Vieweg+Teubner Verlag, 2011. – ISBN 3834814814
- [Klo07] KLOCKE, F.: *Fertigungsverfahren 4, Umformen*. Berlin: Springer, 2007
- [Koc00] KOCH, K.-R.: *Einführung in die Bayes-Statistik*. Berlin and New York: Springer, 2000. – ISBN 3-540-66670-2
- [Kög10] KÖGEL, G.: Von der Rolle. In: *Blech* (2010), Nr. 6, S. 100–103
- [Koh95] KOHAVI, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: DR. PERRAULT, C. R. (Hrsg.): *Proc. 14th Joint Int. Conf. Artificial Intelligence* Bd. 2, 1995, S. 1137–1145
- [Kol13] KOLLER, R.: *Konstruktionslehre für den Maschinenbau: Grundlagen zur Neu- und Weiterentwicklung technischer Produkte mit Beispielen*. Springer-Verlag, 2013
- [Kon90] KONONENKO, I.: Comparison of inductive and naive Bayesian learning approaches to automatic knowledge acquisition. In: *Current Trends in Knowledge Adquisition* (1990), S. 190–197
- [KR05] KLEINSCHMIDT, P.; RANK, C.: *Relationale Datenbanksysteme*. 3. Auflage. Berlin: Springer, 2005
- [Kra92] KRAUSE, D.: *Rechnerunterstütztes Konzipieren und Entwerfen mit Integration von Analysen, insbesondere Berechnungen*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 1992
- [Kra14] KRATZER, M.: *Anwendungsspezifische Entwicklung eines proaktiven Konstruktionsystems auf Basis von Softwareagenten*. Universität Stuttgart, Dissertation, 2014
- [KRBG11a] KRATZER, M.; RAUSCHER, M.; BINZ, H. ; GÖHNER, P.: An agent-based system for supporting design engineers in the embodiment design phase. In: MCALOONE, T. C. (Hrsg.); HOWARD, T. j. (Hrsg.): *18th International Conference on Engineering Design*, Technical University of Denmark, 2011, S. 178–189
- [KRBG11b] KRATZER, M.; RAUSCHER, M.; BINZ, H. ; GRÖHNER, P.: Konzept eines Wissensintegrationssystems zur benutzerfreundlichen, benutzerspezifischen und selbstdiagnos-tischen Integration von Konstruktionswissen. In: KRAUSE, D. (Hrsg.); WARTZACK, S. (Hrsg.); PAETZOLD, K. (Hrsg.): *Design for X*. Hamburg: TuTech Verl, 2011. – ISBN 9783941492370, S. 22–32
- [Kre12] KREHMER, H.: *Fortschritt-Berichte VDI : Reihe 1, Konstruktionstechnik, Maschinenelemente*. Bd. 416: *Vorgehensmodell zum Iterations- und Produktreifegradmanagement in der eigenschaftsbasierten Produktentwicklung*. Düsseldorf: VDI-Verlag, 2012. – ISBN 3183416018
- [KS11] KLETTI, J.; SCHUMACHER, J.: *Die perfekte Produktion: Manufacturing Excellence durch Short Interval Technology (SIT)*. Berlin: Springer, 2011

- [Kug09] KUGLER, H.: *Umformtechnik - Umformen metallischer Konstruktionswerkstoffe*. München: Carl Hanser, 2009
- [Kur92] KURBEL, K.: *Entwicklung und Einsatz von Expertensystemen: Eine anwendungsorientierte Einführung in wissensbasierte Systeme*. 2. Berlin: Springer, 1992. – ISBN 3-540-55237-5
- [KW⁺15] KESTEL, P.; WARTZACK, S. u. a.: Konzept für ein wissensbasiertes FEA Assistenzsystem zur Unterstützung konstruktionsbegleitender Simulationen. In: PAETZOLD, K. (Hrsg.); WARTZACK, S. (Hrsg.); KRAUSE, D. (Hrsg.): *DFX 2015: Proceedings of the 26th Symposium Design for X*, 2015, S. 1–12
- [La 11] LA ROCCA, G.: *Knowledge based engineering techniques to support aircraft design and optimization*. Delft: Delft University of Technology, 2011
- [La 12] LA ROCCA, G.: Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design. In: *Advanced Engineering Informatics* 26 (2012), Nr. 2, S. 159–179. <http://dx.doi.org/10.1016/j.aei.2012.02.002>. – DOI 10.1016/j.aei.2012.02.002. – ISSN 14740346
- [Läm03] LÄMMEL, U.: Data-Mining mittels künstlicher neuronaler Netze. In: *Wismarer Diskussionspapiere* (2003), Nr. 7, S. 1–37
- [Lan84] LANGE, K.: *Umformtechnik: Handbuch für Industrie und Wissenschaft - Band 1: Grundlagen*. 2. Berlin: Springer, 1984. – ISBN 3-540-13249-X
- [Lan93] LANGE, K.: *Umformtechnik. Handbuch für Industrie und Wissenschaft - Band 4: Sonderverfahren, Prozeßsimulation, Werkzeugtechnik, Produktion*. 2. Auflage. Berlin: Springer, 1993
- [LBH⁺12] LIZUNKOVA, Y.; BUROV, S.; HASSEL, T.; BIERBAUM, M. S.; BACH, F. W.: Temperature Control during the APN-Process on the Steel 1.3343. In: *Key Engineering Materials* 504 (2012), S. 1017–1022. <http://dx.doi.org/10.4028/www.scientific.net/KEM.504-506.1017>. – DOI 10.4028/www.scientific.net/KEM.504-506.1017. – ISSN 1662-9795
- [Len91] LENZ, A.: *Knowledge Engineering für betriebliche Expertensysteme: Erhebung, Analyse und Modellierung von Wissen*. Wiesbaden: Springer Fachmedien, 1991
- [LGE⁺15] LÖFFLER, M.; GROEBEL, D.; ENGEL, U.; ANDREAS, K.; MERKLEIN, M.: Analysis of Effectiveness of Locally Adapted Tribological Conditions for Improving Product Quality in Sheet-Bulk Metal Forming. In: *Applied Mechanics and Materials* 794 (2015), S. 81–88. <http://dx.doi.org/10.4028/www.scientific.net/AMM.794.81>. – DOI 10.4028/www.scientific.net/AMM.794.81
- [Lin09] LINDEMANN, U.: *Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden*. Berlin, Heidelberg: Springer, 2009
- [LK01] LAST, M.; KANDEL, A.: Data mining for process and quality control in the semiconductor industry. In: BRAHA, D. (Hrsg.): *Data Mining for Design and Manufacturing*. Boston, MA: Springer US, 2001 (Massive Computing). – ISBN 9781441952059, S. 207–234
- [LM98] LIU, H.; MOTODA, H.: *Feature selection for knowledge discovery and data mining*. New York: Springer, 1998
- [LR14] LÓPEZ, F.; ROMERO, V.: *Mastering Python regular expressions. Leverage regular expressions in Python even for the most complex features*. Birmingham: Packt Pub, 2014

- [Lut11] LUTZ, C.: *Rechnergestütztes Konfigurieren und Auslegen individualisierter Produkte. Rahmenwerk für die Konzeption und Einführung wissensbasierter Assistenzsysteme in die Konstruktion*. München: Dr. Hut Verlag, 2011
- [MAB⁺12] MERKLEIN, M.; ALLWOOD, J.; BEHRENS, B.-A.; BROSIAU, A.; HAGENAH, H.; KUZMAN, K.; MORI, K.; TEKKAYA, A.; WECKENMANN, A.: Bulk forming of sheet metal. In: *CIRP Annals - Manufacturing Technology* 61 (2012), Nr. 2, S. 725–745. <http://dx.doi.org/10.1016/j.cirp.2012.05.007>. – DOI 10.1016/j.cirp.2012.05.007. – ISSN 00078506
- [Mal10] MALLICK, P. K.: *Materials, design and manufacturing for lightweight vehicles*. Boca Raton: CRC Press, 2010
- [Man06] MANSOUR, M.: *Informations- und Wissensbereitstellung für die lebenszyklusorientierte Produktentwicklung*. Vulkan-Verlag, 2006
- [Mat05] MATHEWS, P.: *Design of experiments with MINITAB*. Milwaukee: ASQ Quality Press, 2005
- [Mee07] MEERKAMM, H.: *Fertigungsgerechtes Konstruieren*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Skriptum zur Vorlesung, 2007
- [MF88] MEERKAMM, H.; FINKENWIRTH, K.-W.: Fertigungsgerecht Konstruieren mit CAD - Anforderungen und Lösungsansätze. In: *Proceedings of ICED88* (1988), S. 259–266
- [MG15] MADENCI, E.; GUVEN, I.: *The finite element method and applications in engineering using ANSYS®*. Second edition. New York: Springer, 2015. – ISBN 978-1-4899-7549-2
- [MH16] MERKLEIN, M.; HAGENAH, H.: Introduction to sheet-bulk metal forming. In: *Production Engineering* 10 (2016), Nr. 1, S. 1–3
- [MHB⁺09] MOTTONEN, M.; HARKONEN, J.; BELT, P.; HAAPASALO, H.; SIMILA, J.: Managerial view on design for manufacturing. In: *Industrial Management & Data Systems* 109 (2009), Nr. 6, S. 859–872
- [Mie01] MIETZEL, G.: *Pädagogische Psychologie des Lernens und Lehrens*. 6. Auflage. Göttingen: Hogrefe, 2001
- [Mir06] MIRJAM, M.: *Erfahrungsmanagement mit fallbasierten Assistenzsystemen*. Berlin, Humboldt-Universität, Dissertation, 2006
- [MKOS11] MERKLEIN, M.; KOCH, J.; OPEL, S.; SCHNEIDER, T.: Fundamental investigations on the material flow at combined sheet and bulk metal forming processes. In: *CIRP Annals - Manufacturing Technology* 60 (2011), Nr. 1, S. 283–286. – ISSN 00078506
- [MKRS93] MEERKAMM, H.; KRAUSE, D.; RÖSCH, S.; STORATH, E.: Anforderungen an integrierte Konstruktionssysteme-Auswirkungen auf die Architektur des CAD-Referenzmodells. In: *VDI Berichte* 1079 (1993), S. 299
- [MKS⁺10] MERKLEIN, M.; KOCH, J.; SCHNEIDER, T.; OPEL, S.; VIERZIGMANN, U.: Manufacturing of complex functional components with variants by using a new metal forming process – sheet-bulk metal forming. In: *International Journal of Material Forming* 3 (2010), Nr. S1, S. 347–350
- [MM04] MCNAMEE, P.; MAYFIELD, J.: Character n-gram tokenization for European language text retrieval. In: *Information retrieval* 7 (2004), Nr. 1-2, S. 73–97

- [MMR06] McCARTHY, J.; MINSKY, M. L. ; ROCHESTER, N.: A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. In: *AI Magazine* 27 (2006), Nr. 4, S. 12–14
- [MSP05] MOLINARO, A. M.; SIMON, R. ; PFEIFFER, R. M.: Prediction error estimation: a comparison of resampling methods. In: *Bioinformatics* 21 (2005), Nr. 15, S. 3301–3307
- [MTB+11] MERKLEIN, M.; TEKKAYA, A. E.; BROSIAK, A.; OPEL, S.; KWIATKOWSKI, L.; PLUGG, B. ; SCHUNK, S.: Machines and tools for sheet-bulk metal forming. In: *Key Engineering Materials* 473 (2011), S. 91–98
- [Mür14] MÜRMANN, M.: *Wahrscheinlichkeitstheorie und Stochastische Prozesse*. Berlin, Heidelberg: Springer Spektrum, 2014
- [MW00] MEERKAMM, H.; WARTZACK, S.: Fertigungsgerechtes Konstruieren - Potenzial und aktuelle Tendenzen. In: *Werkstatt und Betrieb* 133 (2000), Nr. 1-2, S. 24–29
- [MW08] MOST, T.; WILL, J.: Metamodel of Optimal Prognosis-an automatic approach for variable reduction and optimal metamodel selection. In: *Proc. Weimarer Optimierungs- und Stochastiktage* 5 (2008), S. 20–40
- [MW11] MOST, T.; WILL, J.: Sensitivity analysis using the Metamodel of Optimal Prognosis. In: *Proc. Weimarer Optimierungs- und Stochastiktage* 8 (2011), S. 24–40
- [NBS16] NORTH, K.; BRANDNER, A. ; STEININGER, M. T.: *Wissensmanagement für Qualitätsmanager: Erfüllung der Anforderungen nach ISO 9001:2015*. 1. Auflage. Wiesbaden: Gabler, 2016
- [Nee15] NEE, A. Y. C.: *Handbook of manufacturing engineering and technology*. London: Springer, 2015
- [Nei90] NEITZEL, R.: *Fortschritte der CAD-Technik*. Bd. 2: *Entwicklung Wissensbasierter Systeme für die Vorrichtungskonstruktion*. Wiesbaden: Vieweg+Teubner Verlag, 1990. – ISBN 978-3-322-88805-1
- [NKNW96] NETER, J.; KUTNER, M. H.; NACHTSHEIM, C. J. ; WASSERMAN, W.: *Applied linear statistical models*. Bd. 4. Irwin Chicago, 1996
- [Non07] NONAKA, I.: The knowledge-creating company. In: *Harvard business review* 85 (2007), Nr. 7/8, S. 162
- [Nor11] NORTH, K.: *Wissensorientierte Unternehmensführung: Wertschöpfung durch Wissen*. 5., aktualisierte und erw. Aufl. Wiesbaden: Gabler, 2011 (Lehrbuch). – ISBN 978-3-8349-2538-1
- [Nor12] NORTH, M.: *Data mining for the masses*. 2012. – ISBN 978-0615684376
- [NT95] NONAKA, I.; TAKEUCHI, H.: *The knowledge creating company: How Japanese companies create the dynamics of innovation*. New York: Oxford Univ. Press, 1995. – ISBN 0-19-509269-4
- [NT12] NONAKA, I.; TAKEUCHI, H.: *Die Organisation des Wissens: Wie japanische Unternehmen eine brachliegende Ressource nutzbar machen*. Campus Verlag, 2012
- [Ope13] OPEL, S.: *Herstellung prozessangepasster Halbzeuge mit variabler Blechdicke durch die Anwendung von Verfahren der Blechmassivumformung*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 2013
- [Opi66] OPITZ, H.: *Werkstückbeschreibendes Klassifizierungssystem*. Aachen, RWTH, Dissertation, 1966

- [Ott04] OTTE, R.: *Data Mining für die industrielle Praxis*. 1. München: Hanser, 2004. – ISBN 3-446-22465-3
- [OW15] OLIVER, M. A.; WEBSTER, R.: *Basic steps in geostatistics: The variogram and kriging*. Cham: Springer, 2015
- [PBFG07] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K. H.: *Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung*. 7. Auflage. Berlin: Springer, 2007. – ISBN 9783540340607
- [PO13] PLETTKE, R.; OPEL, S.: Investigations on orbital forming of sheet metals to manufacture tailored blanks with a defined sheet thickness variation. In: *Advanced Materials Research* Bd. 769, 2013, S. 157–164
- [Pow00] POWERS, W. F.: Automotive materials in the 21st century. In: *Advanced materials and processes* 157 (2000), Nr. 5, S. 38–42
- [PRR03] PROBST, G.; RAUB, S.; ROMHARDT, K.: *Wissen managen. Wie Unternehmen ihre wertvollste Ressource optimal nutzen*. 4. Wiesbaden: Gabler, 2003
- [PS91] PIATETSKY-SHAPIRO, G.: Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop. In: *AI Magazine* 11 (1991), Nr. 5, S. 68–70
- [Pup90] PUPPE, F.: *Problemlösungsmethoden in Expertensystemen*. Berlin: Springer, 1990. – ISBN 9783540532316
- [Pup91] PUPPE, F.: *Einführung in Expertensysteme*. 2. Auflage. Berlin: Springer, 1991
- [Pyl99] PYLE, D.: *Data preparation for data mining*. Bd. 1. Morgan Kaufmann, 1999
- [Qui92] QUINLAN, J. R.: Learning with continuous classes. In: *5th Australian joint conference on artificial intelligence* Bd. 92, 1992, S. 343–348
- [Qui93] QUINLAN, J. R.: *C4.5. Programs for machine learning*. San Mateo: Morgan Kaufmann Publishers, 1993
- [Raa06] RAABE, D.: Einbaufertige und endkonturnahe Formteile durch Verfahrenskombinationender Kaltmassivumformung. In: *Jahrestreffen der Kaltmassivumformer*. Düsseldorf: VDI-Verlag, 2006, S. 1–10
- [Räs91] RÄSE, U.: *Gußgerechtes Konstruieren mit CAD: Möglichkeiten zur Beschreibung und Analyse von Gussteilen*. Erlangen, Universität Erlangen Nürnberg, Dissertation, 1991
- [RB01] RAY, E. T.; BRODACKI, O.: *Einführung in XML*. Köln: O'Reilly, 2001
- [RBW10a] RÖHNER, S.; BREITSPRECHER, T.; WARTZACK, S.: Anforderungen an die Wissensakquisitionskomponente eines selbstlernenden Assistenzsystems. In: BRÖKEL, K. (Hrsg.); FELDHUSEN, J. (Hrsg.); GROTE, K.-H. (Hrsg.); RIEG, F. (Hrsg.); STELZER, R. (Hrsg.): *8. Gemeinsames Kolloquium Konstruktionstechnik*, 2010, S. 1–6
- [RBW10b] ROTH, D.; BINZ, H.; WATTY, R.: Generic structure of knowledge within the product development process. In: MARJANOVIĆ, D. (Hrsg.): *Design 2010*, Design Society, 2010. – ISBN 978-953-7738-03-7, S. 1681–1690
- [RFHN14] RASCH, B.; FRIESE, M.; HOFMANN, W.; NAUMANN, E.: *Quantitative Methoden 1. Einführung in die Statistik für Psychologen und Sozialwissenschaftler*. 4. Auflage. Wiesbaden: Springer, 2014
- [Roo14] ROOCH, A.: *Statistik für Ingenieure: Wahrscheinlichkeitsrechnung und Datenauswertung endlich verständlich*. Berlin: Springer, 2014

- [Rot94] ROTH, K.: *Konstruieren mit Konstruktionskatalogen / Karlheinz Roth*. Bd. 1: *Konstruieren mit Konstruktionskatalogen. Band I: Konstruktionslehre*. 2. Berlin: Springer, 1994. – ISBN 3540573240
- [Rud98] RUDE, S.: *Wissensbasiertes Konstruieren*. Aachen: Shaker, 1998
- [RV95] RICHTER, F. J.; VETTEL, K.: Successful Joint Ventures in Japan. In: *International Journal of Long Range Planning* (1995)
- [SAKR05] SCHÄPPI, B.; ANDREASEN, M. M.; KIRCHGEORG, M. ; RADERMACHER, F. J.: *Handbuch Produktentwicklung*. Hanser München, 2005
- [SBF98] STUDER, R.; BENJAMINS, V. R. ; FENSEL, D.: Knowledge engineering - principles and methods. In: *Data & knowledge engineering* 25 (1998), Nr. 1, S. 161–197
- [SBM⁺12] SCHULZ, E.; BREITSPRECHER, T.; MUSAYEV, Y.; TREMMEL, S.; HOSENFELDT, T.; WARTZACK, S. ; MEERKAMM, H.: Interactions between amorphous carbon coatings and engine oil additives: prediction of the friction behavior using optimized artificial neural networks. In: *Advanced Ceramic Coatings and Materials for Extreme Environment II, Florida*, Wiley (2012)
- [SC08] STEINWART, I.; CHRISTMANN, A.: *Support vector machines*. New York: Springer, 2008
- [Sch94] SCHNEIDER, E.: *Reihe Wirtschaftsinformatik*. Bd. Bd. 12: *Der Prozess der Wissensakquisition und seine Integration in den Expertensystem-Entwicklungsprozess*. Bergisch Gladbach: J. Eul Verlag, 1994
- [Sch95] SCHMIDT, G.: *Dissertationen zur künstlichen Intelligenz*. Bd. 90: *Modellbasierte, interaktive Wissensakquisition und Dokumentation von Domänenwissen (MIKADO)*. Sankt Augustin: Infix, 1995
- [Sch00] SCHÖN, A.: *Konzept und Architektur eines Assistenzsystems für die mechatronische Produktentwicklung*, Lehrstuhl für Konstruktionstechnik der Universität Erlangen-Nürnberg, Dissertation, 2000
- [Sch02] SCHAEFFLER TECHNOLOGIES AG & Co. KG: *Zwischenringe für Mehrkonus-Synchronisationen: Automobil Produktinformation API 06*. Herzogenaurach, 01.11.2002
- [Sch10] SCHNEIDER, T.: *Automatisierte Akquisition von erfahrungsbasiertem Fertigungswissen im Werkzeug- und Formenbau*, Technische Universität Chemnitz, Dissertation, 2010
- [Sch13] SCHULZ, E.: *Wissensbasierte Vorhersage der Reibung in komplexen tribologischen Systemen am Beispiel des Kontakts Nockenwelle/ beschichteter Tassenstößel*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 2013
- [Sch14a] SCHERER, D.: *Methoden zur Automatisierung der inkrementellen Blechumformung*. München, Technische Universität München, Dissertation, 2014
- [Sch14b] SCHULZE, M.: Das virtuelle Stahlwerk. In: *VDI Nachrichten* (01.08.2014), Nr. 31/32. <http://www.vdi-nachrichten.com/Technik-Wirtschaft/Das-virtuelle-Stahlwerk>
- [SG87] SHAW, M. L. G.; GAINES, B. R.: KITTEN: Knowledge initiation and transfer tools for experts and novices. In: *International Journal of Man-Machine Studies* 27 (1987), Nr. 3, S. 251–280

- [SGS⁺07] SCHUH, G.; GOTTSCHALK, S.; SCHÖNING, S.; GULDEN, A.; RAUHUT, M. ; ZANCUL, E.: *Effizient, schnell und erfolgreich: Strategien im Maschinen- und Anlagenbau*. Frankfurt am Main: VDMA, 2007
- [SH03] SCHMALTZ, R.; HAGENHOFF, S.: Entwicklung von Anwendungssystemen für das Wissensmanagement: State of the Art der Literatur. In: *Hrsg.: Schumann, M.: Georg-August-Universität Göttingen, Institut für Wirtschaftsinformatik, Arbeitsbericht* (2003), Nr. 5
- [Sha91] SHAH, J. J.: Conceptual development of form features and feature modelers. In: *Research in engineering design* 2 (1991), Nr. 2, S. 93–108
- [Sha13] SHARAFI, A.: *Knowledge Discovery in Databases: Eine Analyse des Änderungsmanagements in der Produktentwicklung*. München, Technische Universität München, Dissertation, 2013
- [She00] SHEARER, C.: The CRISP-DM Model: The New Blueprint for Data Mining. In: *Journal of Data Warehousing* 5 (2000), Nr. 4
- [Sie32] SIEBEL, E.: *Die Formgebung im bildsamen Zustand*. Düsseldorf: Stahleisen, 1932
- [Sie14] SIEBENPFEIFFER, W.: *Leichtbautechnologien im Automobilbau. Werkstoffe, Fertigung, Konzepte*. Wiesbaden: Springer Vieweg, 2014
- [SK97] SPUR, G.; KRAUSE, F. L.: *Das virtuelle Produkt: Management der CAD-Technik*. München: Hanser, 1997. – ISBN 9783446191761
- [Ska07] SKARKA, W.: Application of MOKA methodology in generative model creation using CATIA. In: *Engineering Applications of Artificial Intelligence* 20 (2007), Nr. 5, S. 677–690. <http://dx.doi.org/10.1016/j.engappai.2006.11.019>. – DOI 10.1016/j.engappai.2006.11.019. – ISSN 09521976
- [SKH00] SANDER, S.; KASAN, D. ; HEYNEN, C.: Überlegungen zum Wissensmanagement in der Produktentwicklung. In: MEERKAMM, H. (Hrsg.): *DFX 2000: Proceedings of the 11th Symposium on Design for X, Schnaittach/Erlangen, Germany, 12.-13.20. 2000*, 2000, S. 51–95
- [SKH⁺13] SALFELD, V.; KRIMM, R.; HÜBNER, S.; MATTHIAS, T. ; VUCETIC, M.: Sheet-Bulk Metal Forming of Symmetric and Asymmetric Parts. In: *Advanced Materials Research* 769 (2013), S. 229–236. <http://dx.doi.org/10.4028/www.scientific.net/AMR.769.229>. – DOI 10.4028/www.scientific.net/AMR.769.229. – ISSN 1662–8985
- [Ski38] SKINNER, B. F.: *The Behavior of Organisms: An Experimental Analysis*. New York: Appleton-Century-Crofts, 1938
- [Ski78] SKINNER, B. F.: *Was ist Behaviorismus*. Reinbek: Rowohlt, 1978
- [SLV⁺11] SCHAPER, M.; LIZUNKOVA, Y.; VUCETIC, M.; CAHYONO, T.; HETZNER, H.; OPEL, S.; SCHNEIDER, T.; KOCH, J. ; PLUGGE, B.: Sheet-bulk metal forming a new process for the production of sheet metal parts with functional components. In: *Metallurgical and Mining Industry* 3 (2011), Nr. 7, S. 53–58
- [SM11] SCHNEIDER, T.; MERKLEIN, M.: Sheet-bulk metal forming of preformed sheet metal parts. In: *Key Engineering Materials* 473 (2011), Nr. 83–90
- [Smy96] SMYTH, P.: Clustering Using Monte Carlo Cross-Validation. In: *KDD*, 1996, S. 126–133
- [Sob15] SOBER, E.: *Ockham's razors - A user's manual*. Cambridge: Cambridge University Press, 2015. – ISBN 978-1-107-06849-0

- [Ste07] STEVENS, J.: *Intermediate statistics: A modern approach*. 3rd ed. New York: Lawrence Erlbaum Associates, 2007. – ISBN 978-0-8058-5466-4
- [Sto74] STONE, M.: Cross-validatory choice and assessment of statistical predictions. In: *Journal of the royal statistical society. Series B (methodological)* (1974), S. 111–147
- [Sto91] STOLL, H. W.: Design for manufacture: an overview. In: CORBETT, J. (Hrsg.): *Design for manufacture - Strategies, principles, and techniques*. Wokingham, England: Addison-Wesley, 1991, S. 107–129
- [Sto95] STORATH, E.: Entwicklung und Konstruktion auf der Datenautobahn -ein besserer Weg? In: MEERKAMM, H. (Hrsg.): *Beiträge zum 6. Symposium Design for X*, 1995, S. 48–53
- [Sto96] STORATH, E.: *Kontextsensitive Wissensbereitstellung in der Konstruktion: Wissensbasierte Konstruktionssysteme zur Assistenz in der Produktentwicklung beim TeleEngineering*. Lehrstuhl für Konstruktionstechnik der Universität Erlangen-Nürnberg, Dissertation, 1996
- [Sto01] STOKES, M.: *Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications*. London: Professional Engineering, 2001
- [Suh93] SUHM, A.: *Produktmodellierung in wissensbasierten Konstruktionssystemen auf der Basis von Lösungsmustern*. Karlsruhe, Universität Karlsruhe, Dissertation, 1993
- [SvH10] SIEBERTZ, K.; VAN BEBBER, D. ; HOCHKIRCHEN, T.: *Statistische Versuchsplanung: Design of Experiments (DoE)*. Berlin: Springer, 2010
- [SVLBG15] STJEPANDIĆ, J.; VERHAGEN, W.; LIESE, H. ; BERMELL-GARCIA, P.: Knowledge-Based Engineering. In: STJEPANDIĆ, J. (Hrsg.); WOGNUM, N. (Hrsg.); VERHAGEN, W. J. (Hrsg.): *Concurrent engineering in the 21st century*. Cham: Springer International Publishing, 2015, S. 255–286
- [SW03] SHAH, R.; WARD, P. T.: Lean manufacturing: context, practice bundles, and performance. In: *Journal of operations management* 21 (2003), Nr. 2, S. 129–149
- [SWG⁺16] SIECZKAREK, P.; WERNICKE, S.; GIES, S.; MARTINS, P. A. F. ; TEKKAYA, A. E.: Incipient and repeatable plastic flow in incremental sheet-bulk forming of gears. In: *The International Journal of Advanced Manufacturing Technology* (2016), S. 1–10. <http://dx.doi.org/10.1007/s00170-016-8442-6>. – DOI 10.1007/s00170-016-8442-6. – ISSN 1433–3015
- [TAN15] TSCHÖPE, S.; ARONSKA, K. ; NYHUIS, P.: „Was ist eigentlich Industrie 4.0?“: Eine quantitative Datenbankanalyse liefert einen Einblick. In: *Zeitschr. f. wirtsch. Fabrikbetrieb* (2015), Nr. 03, S. 145–149. – ISSN 0932-0482
- [Tho31] THORNDIKE, E. L.: *Human learning*. New York and London: The Century Co, 1931
- [Tho32] THORNDIKE, E. L.: *The fundamentals of learning*. New York: Columbia University, 1932
- [Tic97] TICHEM, M.: *A design coordination approach to design for X*. TU Delft, Delft University of Technology, 1997
- [Tot00] TOTOK, A.: *Modellierung von OLAP- und Data-Warehouse-Systemen*. Wiesbaden: Deutscher Universitätsverlag, 2000
- [Tre09] TREMMEL, S.: *Ein Beitrag zur Auslegung beschichteter Bauteile unter zyklischer Beanspruchung im Wälz-Cleit-Kontakt*. Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Dissertation, 2009

- [Tsc05] TSCHÄTSCH, H.: *Praxis der Umformtechnik: Arbeitsverfahren, Maschinen, Werkzeuge*. 8. Auflage. Wiesbaden: Vieweg und Teubner, 2005. <http://dx.doi.org/10.1007/978-3-322-99432-5>
- [TSK10] TAN, P. N.; STEINBACH, M. ; KUMAR, V.: *Introduction to data mining*. Boston: Pearson Addison-Wesley, 2010
- [UA92] UNRUH, V.; ANDERSON, D. C.: Feature-based modeling for automatic mesh generation. In: *Engineering with computers* 8 (1992), Nr. 1, S. 1-12
- [Vaj09] VAJNA, S.: *CAx für Ingenieure: Eine praxisbezogene Einführung*. 2. Auflage. Berlin: Springer, 2009
- [van01] VAN VLIET, J. W.: *Design for manufacturing. Development and application of a process-based design support methodology*. Delft: N.A., 2001. – ISBN 9037001912
- [VDI93] VDI 2221: *Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. Berlin, 1993
- [VDI97] VDI - VEREIN DEUTSCHER INGENIEURE: *VDI 2222 -Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. Berlin, 1997
- [VDI04a] VDI - VEREIN DEUTSCHER INGENIEURE: *VDI 2223 -Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. Berlin, 2004
- [VDI04b] VDI 2206: *Entwicklungsmeethodik für mechatronische Systeme*. Berlin, 2004
- [VDI09a] VDI - VEREIN DEUTSCHER INGENIEURE: *VDI 2209 - 3-D-Produktmodellierung, Technische und organisatorische Voraussetzungen, Verfahren, Werkzeuge und Anwendungen, Wirtschaftlicher Einsatz in der Praxis*. Berlin, 2009
- [VDI09b] VDI 5610: *VDI 5610 Blatt 1 - Wissensmanagement im Ingenieurwesen*. Berlin, 2009
- [VDI15] VDI 5610: *VDI 5610 Blatt 2 - Wissensbasiertes Konstruieren*. Berlin, 2015
- [VDM01] VDMA - VERBAND DEUTSCHER MASCHINEN- UND ANLAGENBAUER: *Kennzahlenkompass*. Frankfurt am Main: VDMA, 2001
- [Ver07] VERMEULEN, B.: *Knowledge based method for solving complexity in design problems*, TU Delft, Dissertation, 2007
- [VKME12] VIERZIGMANN, U.; KOCH, J.; MERKLEIN, M. ; ENGEL, U.: Material flow in sheet-bulk metal forming. In: *Key Engineering Materials* Bd. 504, 2012, S. 1035-1040
- [Vol41] VOLK, C.: *Der konstruktive Fortschritt. Ein Skizzenbuch*. Berlin, Heidelberg: Springer, 1941
- [War99] WARTZACK, S.: Durchgängige Rechnerunterstützung in der Produktentwicklung durch den Einsatz von semantisch hochwertigen Features. In: *VDI Berichte* 1497 (1999), S. 369-390
- [War01] WARTZACK, S.: *Predictive Engineering - Assistenzsystem zur multikriteriellen Analyse alternativer Produktkonzepte*. Düsseldorf: VDI, 2001. – ISBN 3183336014
- [War15] WARTZACK, S.: *Technische Produktgestaltung - Skriptum zur Vorlesung*. Erlangen, 2015
- [Wat86] WATERMAN, D. A.: *A guide to expert systems*. Reading: Addison-Wesley, 1986

- [Web92] WEBER, A.: *Ein relationsbasiertes Datenmodell als Grundlage für die Bauteil-Tolerierung*, Lehrstuhl für Konstruktionstechnik der Universität Erlangen-Nürnberg, Dissertation, 1992
- [Web05] WEBER, C.: CPM/PDD – An extended theoretical approach to modelling products and product development processes. In: BLEY, H. (Hrsg.): *Proceedings of the 2nd German-Israeli Symposium on Advances in Methods and Systems for Development of Products and Processes*, 2005, S. 159–179
- [Web12] WEBER, C.: Produkte und Produktentwicklungsprozesse abbilden mit Hilfe von Merkmalen und Eigenschaften – eine kritische Zwischenbilanz. In: KRAUSE, D. (Hrsg.); PAETZOLD, K. (Hrsg.); WARTZACK, S. (Hrsg.): *Design for X*. Hamburg: TuTech-Verl, 2012. – ISBN 978-3-941492-51-6, S. 25–62
- [WEH11] WITTEN, I. H.; EIBE, F. ; HALL, M. A.: *Data mining: Practical machine learning tools and techniques*. 3. Auflage. Burlington: Morgan Kaufmann, 2011. – ISBN 0123748569
- [Wei89] WEIDENMANN, B.: Lernen - Lerntheorien. In: LENZEN, D. (Hrsg.): *Pädagogische Grundbegriffe* Bd. 2. Reinbeck: Rowohlt Taschenbuch Verlag, 1989, S. 996–1010
- [Wei15] WEICKER, K.: *Evolutionäre Algorithmen*. 3. Auflage. Wiesbaden: Springer, 2015
- [WK07] WREMBEL, R.; KONCILIA, C.: *Data warehouses and OLAP. Concepts, architectures, and solutions*. Hershey: IRM Press, 2007
- [WM00] WARTZACK, S.; MEERKAMM, H.: Durchgängige Rechnerunterstützung in der Produktentwicklung aufgrund des Einsatzes semantisch hochwertiger Features. In: *VDI Berichte 1435* (2000), Nr. 3, S. 369–390
- [Wög42] WÖGERBAUER, H.: *Die Technik des Konstruierens*. München and Berlin: Oldenbourg Verlag, 1942
- [XL87] XU, C.-W.; LU, Y.-Z.: Fuzzy model identification and self-learning for dynamic systems. In: *IEEE Transactions on Systems, Man, and Cybernetics 4* (1987), Nr. 17, S. 683–689
- [Yeh07] YEH, I.-C.: Modeling slump flow of concrete using second-order regressions and artificial neural networks. In: *Cement and Concrete Composites 29* (2007), Nr. 6, S. 474–480
- [Zim67] ZIMMERMANN, D.: *ZAFO: Eine allgemeine Formenordnung für Werkstücke, Gestaltung, Handhabung und Rationalisierungserfolg*. Stuttgart-Vaihingen, TU Stuttgart, Dissertation, 1967
- [ZS01] ZIMMERMANN, W.; STACHE, U.: *Operations-Research: quantitative Methoden zur Entscheidungsvorbereitung*. Oldenbourg Verlag, 2001

Online-Buchshop für Ingenieure

■■■ VDI nachrichten

Online-Shops



Fachliteratur und mehr - jetzt bequem online recherchieren & bestellen unter: www.vdi-nachrichten.com/
Der-Shop-im-Ueberblick



Täglich aktualisiert: Neuerscheinungen VDI-Schriftenreihen



BUCHSHOP

Im Buchshop von vdi-nachrichten.com finden Ingenieure und Techniker ein speziell auf sie zugeschnittenes, umfassendes Literaturangebot.

Mit der komfortablen Schnellsuche werden Sie in den VDI-Schriftenreihen und im Verzeichnis lieferbarer Bücher unter 1.000.000 Titeln garantiert fündig.

Im Buchshop stehen für Sie bereit:

VDI-Berichte und die Reihe **Kunststofftechnik**:

Berichte nationaler und internationaler technischer Fachtagungen der VDI-Fachgliederungen

Fortschritt-Berichte VDI:

Dissertationen, Habilitationen und Forschungsberichte aus sämtlichen ingenieurwissenschaftlichen Fachrichtungen

Newsletter „Neuerscheinungen“:

Kostenfreie Infos zu aktuellen Titeln der VDI-Schriftenreihen bequem per E-Mail

Autoren-Service:

Umfassende Betreuung bei der Veröffentlichung Ihrer Arbeit in der Reihe Fortschritt-Berichte VDI

Buch- und Medien-Service:

Beschaffung aller am Markt verfügbaren Zeitschriften, Zeitungen, Fortsetzungsreihen, Handbücher, Technische Regelwerke, elektronische Medien und vieles mehr – einzeln oder im Abo und mit weltweitem Lieferservice

VDI nachrichten

BUCHSHOP

www.vdi-nachrichten.com/Der-Shop-im-Ueberblick

Die Reihen der Fortschritt-Berichte VDI:

- 1 Konstruktionstechnik/Maschinenelemente
- 2 Fertigungstechnik
- 3 Verfahrenstechnik
- 4 Bauingenieurwesen
- 5 Grund- und Werkstoffe/Kunststoffe
- 6 Energietechnik
- 7 Strömungstechnik
- 8 Mess-, Steuerungs- und Regelungstechnik
- 9 Elektronik/Mikro- und Nanotechnik
- 10 Informatik/Kommunikation
- 11 Schwingungstechnik
- 12 Verkehrstechnik/Fahrzeugtechnik
- 13 Fördertechnik/Logistik
- 14 Landtechnik/Lebensmitteltechnik
- 15 Umwelttechnik
- 16 Technik und Wirtschaft
- 17 Biotechnik/Medizintechnik
- 18 Mechanik/Bruchmechanik
- 19 Wärmetechnik/Kältetechnik
- 20 Rechnerunterstützte Verfahren (CAD, CAM, CAE CAQ, CIM ...)
- 21 Elektrotechnik
- 22 Mensch-Maschine-Systeme
- 23 Technische Gebäudeausrüstung

ISBN 978-3-18-344901-9