

Sebastian Bauer

Prozesssprachenbasiertes System zur Ansteuerung digitaler Menschmodelle
als Teilkomponente einer Software zur Planung und Visualisierung
menschlicher Arbeit in der Digitalen Fabrik

Sebastian Bauer

**Prozesssprachenbasiertes System
zur Ansteuerung digitaler Menschmodelle
als Teilkomponente einer Software
zur Planung und Visualisierung
menschlicher Arbeit in der Digitalen Fabrik**



**TECHNISCHE UNIVERSITÄT
CHEMNITZ**

Universitätsverlag Chemnitz

2016

Impressum

Bibliografische Information der Deutschen Nationalbibliothek

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

Technische Universität Chemnitz/Universitätsbibliothek

Universitätsverlag Chemnitz

09107 Chemnitz

<http://www.tu-chemnitz.de/ub/univerlag>

Herstellung und Auslieferung

Verlagshaus Monsenstein und Vannerdat OHG

Am Hawerkamp 31

48155 Münster

<http://www.mv-verlag.de>

ISBN 978-3-944640-72-3

<http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-195199>



TECHNISCHE UNIVERSITÄT
CHEMNITZ

**Prozesssprachenbasiertes System
zur Ansteuerung digitaler Menschmodelle
als Teilkomponente einer Software
zur Planung und Visualisierung
menschlicher Arbeit in der Digitalen Fabrik**

Dissertation
zur Erlangung des akademischen Grades

Dr.-Ing.

Herr Dipl.-Inf. Sebastian Bauer
geboren am 9. Februar 1982 in Karl-Marx-Stadt

Fakultät für Informatik
an der Technischen Universität Chemnitz

Gutachter:

Prof. Dr. Guido Brunnett, TU Chemnitz
Prof. Dr.-Ing. Egon Müller, TU Chemnitz

Tag der Verteidigung:

16. Dezember 2015

Bibliographische Beschreibung

Bauer, Sebastian

Prozesssprachenbasiertes System zur Ansteuerung digitaler Menschmodelle als Teilkomponente einer Software zur Planung und Visualisierung menschlicher Arbeit in der Digitalen Fabrik

Dissertation an der Fakultät für Informatik
der Technischen Universität Chemnitz,
Professur Graphische Datenverarbeitung und Visualisierung, Chemnitz

186 Seiten, 51 Abbildungen, 12 Tabellen, 90 Literaturquellen, 3 Anlagen

Referat

Die Werkzeuge und Methoden der Digitalen Fabrik sind bereits seit vielen Jahren in den einzelnen Phasen des Produktentstehungsprozess im Einsatz. Sie werden dabei u.a. auch zur Planung und Gestaltung menschlicher Arbeit eingesetzt. Mit Hilfe digitaler Menschmodelle können Aspekte des Arbeitsablaufs, der Zeitwirtschaft und der Arbeitsplatzgestaltung bereits frühzeitig untersucht und verbessert werden. Die Entwicklung effizienter Simulationssysteme steht auf diesem Gebiet jedoch insbesondere im Vergleich mit anderen Bereichen, wie beispielsweise der Robotersimulation, noch am Anfang.

Diese Arbeit beschäftigt sich zunächst mit der Analyse bereits existierender Menschmodell-Simulationssysteme. Aus den identifizierten Schwachstellen dieser Systeme und weiteren durch Experteninterviews sowie Literaturrecherche gewonnenen Erkenntnissen wird eine Anforderungsliste erstellt, die als Grundlage für die Konzeption und Umsetzung eines innovativen Ansatzes zur Ansteuerung und Bewegungsgenerierung für digitale Menschmodelle dient. Diese neuartige Methodik wird schließlich in ein Simulations-Softwaresystem implementiert und anschließend im Praxis-Einsatz evaluiert.

Schlagworte

Digitale Menschmodelle, Digitale Fabrik, Bewegungssimulation, menschliche Arbeit, 3D-Visualisierung, Fabrikplanung, Arbeitswissenschaft

Vorwort

Diese Arbeit entstand während der Konzeption und Entwicklung des Simulationssystems »ema« (Editor menschlicher Arbeit) der Chemnitzer Firma imk automotive GmbH. Die hierfür erforderliche wissenschaftliche Grundlagenforschung wurde im Verbundprojekt »System zur Bewegungssynthese für digitale Menschmodelle (eMAN)« erbracht und im Rahmen des Europäischen Fonds für regionale Entwicklung (EFRE) sowie vom Freistaat Sachsen durch die Sächsische Aufbaubank (SAB) gefördert. Ich bedanke mich deshalb bei den beteiligten Projektpartnern der Professur Graphische Datenverarbeitung & Visualisierung (TU Chemnitz), der Professur Arbeitswissenschaft (TU Chemnitz), des Instituts für Mechatronik e. V. sowie der imk automotive GmbH.

Herrn Prof. Guido Brunnett danke ich für die unkomplizierte und konstruktive Betreuung des Promotionsvorhabens. Herrn Prof. Egon Müller danke ich für die Möglichkeit zur Konsultation und zum Gedankenaustausch.

Mein ganz besonderer Dank gilt Herrn Dr. Jens Trepte und Herrn Dr. Wolfgang Leidholdt für ihren unerschütterlichen Glauben und aufopferungsvollem Einsatz ohne den die Idee, welche 2005 entstanden ist, heute nicht als anerkanntes Simulationssystem existieren würde. Weiterhin möchte ich mich auch noch einmal explizit bei allen anderen Mitarbeitern der imk automotive GmbH und dabei insbesondere bei Dr. Lars Fritzsche, Gerson Heuwieser, Thomas Jäckel und Alexander Hartenbach für ihre Unterstützung bedanken.

Meinen Eltern Ute und Günter Weitz danke ich für die Förderung, das Vertrauen und die Unterstützung auf meinem bisherigen Lebensweg.

Ich danke allen meinen Freunden für die Motivation und Inspiration die sie mir gegeben haben und hierbei insbesondere Madeleine Haubold für ihre Unterstützung in der finalen Phase der Arbeit.

Vorwort

Mein größter Dank geht an meine Frau, Julia Bauer, und meine beiden Kinder Emilia und Raphael. Ich danke Euch für die Motivation und Kraft die Ihr mir gegeben habt und insbesondere dafür, dass Ihr mir trotz der ohnehin zumeist knappen gemeinsamen Zeit, die notwendigen Freiräume zur Fertigstellung dieser Arbeit eingeräumt habt.

Inhaltsverzeichnis

Bibliographische Beschreibung.....	I
Vorwort.....	III
Inhaltsverzeichnis	V
Abkürzungsverzeichnis	IX
Abbildungsverzeichnis	XI
Tabellenverzeichnis	XIII
1 Einleitung.....	1
1.1 Die Gesellschaft ist im Wandel.....	1
1.2 Ein Planer ist kein Trickfilmersteller.....	3
1.3 Formulierung der Aufgabe	5
1.4 Ergebnisse.....	8
2 Einordnung und Forschungsumgebung	15
2.1 Planung, Gestaltung und Analyse menschlicher Arbeit.....	15
2.2 Prozesssprachen in der Fertigungsplanung.....	20
2.3 Die Digitale Fabrik	24
2.4 Digitale Menschmodelle	28
3 Methodische Vorarbeit	31
3.1 Analyse existierender Menschmodelle	31
3.2 Anforderungsanalyse.....	35
3.3 Identifikation der notwendigen Systemkomponenten.....	40
3.4 Eingangsinformationen für das System	43
3.4.1 Umgebungsinformationen.....	43
3.4.2 Tätigkeitsbeschreibung.....	48
3.5 Entwicklung der Absichtskomponente.....	52
3.5.1 Arten und Strukturierung des relevanten Wissens	52
3.5.2 Der Verrichtungs-Begriff.....	55

Inhaltsverzeichnis

3.5.3	Konzeption der Verrichtungsbibliothek.....	60
3.5.4	Beispiel-Verrichtung »Objekt(e) aufnehmen«.....	68
3.5.5	Zusammenführung der Erkenntnisse	72
3.6	Bewegungsgenerierung	76
3.6.1	Von der abstrakten zur konkreten Bewegungsbeschreibung.....	76
3.6.2	Verwendete Menschmodell-Struktur	79
3.6.3	Überblick zur Körperhaltungsgenerierung.....	83
3.6.4	Beispielverfahren zur Körperhaltungsgenerierung	86
4	Softwaretechnische Umsetzung.....	93
4.1	Vorgehensweise	93
4.2	Anforderungen, Anwendungsfälle und Arbeitsablauf	95
4.3	Systemarchitektur	99
4.4	Gestaltung der Benutzeroberfläche.....	105
4.5	Implementierung des Simulationskerns.....	111
4.5.1	Überblick	111
4.5.2	Objektstrukturen	113
4.5.3	Verhaltensstrukturen.....	118
4.5.4	Simulationsstrukturen	122
4.6	Spezielle geometrische Algorithmen.....	126
4.6.1	Laufwegs-Bestimmung	126
4.6.2	Arbeitspositions-Bestimmung	129
4.6.3	Objekthandhabung.....	133
4.7	Tests, Ergebnisse und produktiver Einsatz	136
5	Weiterführende Untersuchungen	145
6	Zusammenfassung.....	151
	Literaturverzeichnis	155
	Thesen.....	163

Inhaltsverzeichnis

Anhang A – Übersicht der vorliegenden Arbeit	A
Anhang B – Verwendete Menschmodell-Struktur	B
Anhang C – Morphologischer Kasten	C
Lebenslauf.....	I

Abkürzungsverzeichnis

2D / 3D.....	zweidimensional / dreidimensional
AAWS.....	Automotive Assembly Worksheet
CAD.....	Computer Aided Design
CAE.....	Computer Aided Engineering
CAM.....	Computer Aided Manufacturing
DHM.....	Digital Human Model
DIN.....	Deutsches Institut für Normung
DOF.....	Degree of Freedom
EAWS.....	European Assembly Worksheet
ema ⁵	Editor menschlicher Arbeit
ERP.....	Enterprise-Resource-Planning
KVP.....	Kontinuierlicher Verbesserungsprozess
MoCap.....	Motion Capturing
MTM.....	Methods Time Measurement
PEP.....	Produkt-Entstehungs-Prozess
PPS.....	Produktionsplanungs- und Steuerungssystem
PLM.....	Produkt-Lebenszyklus-Management
REFA.....	Reichsausschuss für Arbeitszeitermittlung
SvZ.....	Systeme vorbestimmter Zeiten
TCP.....	Tool Center Point
UAS.....	Universelles Analysier-System
VR.....	Virtuelle Realität

Abbildungsverzeichnis

Abbildung 1: Fachliche Einordnung als Säulenmodell	7
Abbildung 2: Referenzmodell für den Produktentstehungsprozess [17, p. 239]	17
Abbildung 3: Das Arbeitssystem als Modell [32, p. 31]	20
Abbildung 4: Modelle der »Digitalen Fabrik« [40, p. 11]	25
Abbildung 5: Relevante Software-Werkzeuge der Digitalen Fabrik [17, p. 3]	26
Abbildung 6: »Human« und »Jack« [7, pp. 30, Abb. 2.6]	31
Abbildung 7: Komponentenschema des Simulations-Systems [47, p. 269]	40
Abbildung 8: Greifpunkt und TCP an einem Akkuschrauber	45
Abbildung 9: Inhalte der Tätigkeitsbeschreibung	49
Abbildung 10: Bewegungsebenen [10, p. 22]	56
Abbildung 11: Häufigkeiten von Verrichtungen im Automobilbau [55, p. 36] ..	58
Abbildung 12: Verrichtungskomplexität vs. Modellierungsaufwand [55, p. 45].	59
Abbildung 13: Bestehende Gliederungen manueller Tätigkeiten [55, p. 51]	60
Abbildung 14: Übersicht über die Verrichtungsbibliothek V1.0	63
Abbildung 15: Körperhaltungsgraph [58]	64
Abbildung 16: Beispiel-Verrichtung »Laufen zum Ziel« [59]	67
Abbildung 17: Algorithmus der Verrichtung »Objekt(e) aufnehmen«	70
Abbildung 18: „Verrichtungsbibliothek“ (Morph. Kasten)	73
Abbildung 19: „Funktionen des Simulationsmodells“ (Morph. Kasten)	74
Abbildung 20: Verwendete Menschmodell-Struktur	80
Abbildung 21: Menschmodell Hilfsabmessungen	81
Abbildung 22: Topologien digitaler Menschmodelle [62, p. 7]	82
Abbildung 23: Menschmodell-Bewegungsgenerierung im Überblick [59]	85
Abbildung 24: Phasen der Posen-Generierung für die Beinstellung	86
Abbildung 25: Übersicht implementierter Griffarten	91
Abbildung 26: Editor menschlicher Arbeit (ema ⁵) im 2-Bildschirm-Betrieb	94
Abbildung 27: Benutzerrollen und Anwendungsfälle im Gesamtsystem	96
Abbildung 28: Arbeitsablauf für den Umgang mit dem Simulationssystem	97
Abbildung 29: Fachliche Architektur des Gesamtsystems (vereinfacht)	100
Abbildung 30: Komponenten und Schnittstellen im System	104
Abbildung 31: Die Benutzeroberfläche im Überblick*	106
Abbildung 32: Interaktive Objektmanipulation in der 3D-Ansicht	108
Abbildung 33: Ergebnis-Auswertung	110

Abbildungsverzeichnis

Abbildung 34: Klassendiagramm der Objektstrukturen.....	114
Abbildung 35: Klassendiagramm der Verhaltensstrukturen (Auszug).....	118
Abbildung 36: Varianten der Verrichtungszeitbestimmung.....	120
Abbildung 37: Klassendiagramm der Simulationsstrukturen.....	122
Abbildung 38: Visualisierung von Laufwegen in einem Beispiel-Layout.....	126
Abbildung 39: Raumunterteilung mit »gleichmäßigem Raster« [67, p. 39].....	127
Abbildung 40: Raumunterteilung mit »QuadTree-Struktur« [67, p. 40].....	127
Abbildung 41: Schematische Darstellung potentieller Arbeitspositionen.....	129
Abbildung 42: Arbeitspositionsbestimmung für Werkzeuge [69, p. 37].....	131
Abbildung 43: Algorithmus zur Werkstückklassifikation [70, p. A2].....	134
Abbildung 44: Testszenario zur Werkzeughandhabung.....	139
Abbildung 45: Testszenario für »Werkzeug auf Pfad bewegen«.....	142
Abbildung 46: Simulation einer Schweiß-Station.....	143
Abbildung 47: Simulation einer Station zur Nockenwellenmontage.....	143
Abbildung 48: Greifarten an verschiedenen filigranen Objekten [73, p. 82]....	146
Abbildung 49: Kooperative Tätigkeiten.....	147
Abbildung 50: Trainingssimulation für transportable Pontonbrücke [75].....	148
Abbildung 51: Tests zur Arbeit an der bewegten Karosserie.....	150

Tabellenverzeichnis

Tabelle 1: MTM-UAS-Simulation (Prototyp) [9, p. 59]	11
Tabelle 2: Editor menschlicher Arbeit (Virtools Prototyp) [10, p. 70]	11
Tabelle 3: Parametrisches Laufen (Prototyp)	11
Tabelle 4: Ergonomie-Visualisierung (Demonstrator).....	12
Tabelle 5: ema ⁵ DELMIA-Prototyp	12
Tabelle 6: ema ⁵ V5	13
Tabelle 7: ema ⁵	13
Tabelle 8: Relevante Umgebungsinformationen.....	47
Tabelle 9: Relevante Tätigkeitsinformationen.....	51
Tabelle 10: Verfahren zur Körperhaltungsgenerierung	92
Tabelle 11: Umsetzung der funktionalen Anforderungen.....	136
Tabelle 12: Verifizierung der nichtfunktionalen Anforderungen	140

1 Einleitung

1.1 Die Gesellschaft ist im Wandel

„Der demografische Wandel in Deutschland ist schon in vollem Gange. Seit fast vier Jahrzehnten reicht die Zahl der geborenen Kinder nicht aus, um die Eltern- generation zu ersetzen. Es sterben mehr Menschen, als Kinder geboren werden. Ohne Zuwanderung aus dem Ausland würde Deutschlands Bevölkerung bereits seit langem rapide schrumpfen.“ [1, p. 6] Der vom Statistischen Bundesamt festgestellte Rückgang der Gesamtbevölkerungszahl führt zudem zu einer Reduktion der Menge der Menschen im erwerbsfähigen Alter. Gleichzeitig wird durch die steigende Lebenserwartung der Anteil der Älteren im Segment der 20- bis 65-Jährigen immer mehr zunehmen. [2, p. 6]

Die produzierenden Unternehmen müssen sich zwangsweise auf diesen Wandel einstellen, wodurch sich der Fokus bei der Gestaltung von Fertigungsprozessen neben der effektiven Nutzung der Ressourcen auch immer mehr auf die ergonomische Gestaltung der Arbeitsplätze richtet. „Arbeitsbedingungen sind so zu gestalten, dass die Beschäftigten ihre Arbeit bis zu ihrer Rente gesund ausüben können“ [3, p. 97] ist ein Statement des Automobilkonzerns Daimler zu diesem Thema. Auch bei Volkswagen wird der Anpassung der Arbeitsbedingungen an den demografischen Wandel eine immer größer werdende Bedeutung zu teil: „Mit ergonomischen Verbesserungen entlang des gesamten Produkt- entstehungsprozesses gewährleistet Volkswagen, dass die Qualität der Arbeits- plätze und die aus der Produktion resultierenden Belastungen für die Mitarbeiter bereits in der Planungs- und Konstruktionsphase berücksichtigt werden.“ [4] An diesen beiden Beispielen wird deutlich, dass die Notwendigkeit, auf die Gesund- heit und Zufriedenheit der Arbeitskräfte zu achten, bereits in den jeweiligen Konzernstrategien verankert ist. Im Rahmen des sogenannten »Kontinuierlichen Verbesserungsprozess – KVP« [5] werden daher fortlaufend Maßnahmen durchgeführt, um vorhandene Arbeitsplätze zu optimieren und in der Ferti- gungsprozessplanung wird immer früher auf die ergonomischen Aspekte der Prozesse und Ressourcen geachtet. Teilweise haben die Untersuchungen sogar Rückwirkungen auf die Produktentwicklung.

Bereits seit Ende der 1970er Jahre optimieren rechnergestützte Methoden den Entstehungsprozess neuer Produkte. Daten, die in der Produktentwicklung ge-

1 Einleitung

neriert werden, können übernommen und in nachgelagerten Phasen der Fertigungsprozessgestaltung verwendet werden. Neben den Produktdaten liegen auch digital gespeicherte Informationen über Werkhallen, einzelne Arbeitsplätze und geplante Arbeitsabläufe vor. Selbst geometrische Informationen zu benötigten Ressourcen, seien es die Abmessungen von Regalen und Behältern oder der Platzbedarf von Anlagen und Robotern, sind mittlerweile in entsprechenden Datenverwaltungssystemen erfasst. Trotz all der zur Verfügung stehenden Daten finden diese Informationen für den Bereich der Planung menschlicher Arbeit nur sehr eingeschränkte Verwendung. Für die traditionellen Planungsmethoden (z.B.: Layout-Planung, Zeitanalysen) war bislang eine ungefähre Vorstellung über den zu gestaltenden Arbeitsplatz ausreichend. Doch aufgrund der neuen Anforderungen, welche durch die Berücksichtigung ergonomischer Gesichtspunkte in den Planungsprozess eingebracht werden, spielen auch komplexe geometrische Aspekte eine immer wichtigere Rolle. Digitale Menschmodelle werden zunehmend eingesetzt, um unter Verwendung der in der Digitalen Fabrik vorhandenen Daten eine neue Planungsqualität zu erreichen [6]. Zudem haben anhand digitaler Menschmodelle erstellte und verifizierte Planungsergebnisse den nicht zu unterschätzenden Vorteil, dass sie eine gemeinsame Diskussionsgrundlage zum verlustfreien Informationsaustausch zwischen Planungsingenieur, Facharbeiter und Management bilden können.

1.2 Ein Planer ist kein Trickfilmersteller

Der Faktor „Mensch“ spielt im Rahmen der Fertigungsprozessplanung eine bedeutsame Rolle. Dennoch reduziert sich die Planung menschlicher Arbeit im Kontext der Digitalen Fabrik noch immer auf vorwiegend textbasierte Auflistungen von sogenannten Sollzeit- oder Ergonomie-Analysen, ergänzt durch zweidimensionale Skizzen der Arbeitsumgebung. Dies ist das Handwerkszeug, welches der Planungsingenieur schon immer zur Verfügung hatte, worin er im Umgang geübt ist und was einen vertrauten, nachvollziehbaren Eindruck vermittelt. Bezüglich des Verständnisses und der technischen Realisierung der verbal beschriebenen Arbeitsaufgaben kann der Planer auf das Wissen und Verständnis des Facharbeiters aufbauen. Dieser führt die ihm übertragenen Aufgaben anhand seiner erworbenen Fähigkeiten und seiner Erfahrung aus. Aufgrund der menschlichen Vielseitigkeit ist der Facharbeiter sogar in der Lage, im Vorfeld begangene Planungsfehler bis zu einem bestimmten Grad auszugleichen – wobei dieses „Ausgleichen“ nicht selten negative Nebeneffekte hervorruft. Solche Effekte können beispielweise übermäßig hohe körperliche Belastungen des Mitarbeiters aufgrund nicht-ergonomischer Arbeitsbedingungen oder als Gegenstück ineffiziente Prozessfolgen aufgrund mangelhafter Auslastung sein. Die Anzahl derartiger Planungsfehler kann durch eine verbesserte Planungsabsicherung und den Einsatz neuer Methoden minimiert werden – wovon nicht nur der einzelne Mitarbeiter sondern auch das Unternehmen als Ganzes profitiert.

Dass die Nutzung von CAD/CAE-Systemen in der Planung eine ganze Bandbreite neuer Möglichkeiten, insbesondere im Bereich der Planungsabsicherung, bietet, ist zwar bekannt – zu oft scheitert der effektive Einsatz jedoch an der Komplexität, dem Funktionsumfang und der selten intuitiven Bedienung dieser Systeme. Hiervon betroffen sind auch 3D-Planungssysteme und insbesondere die vorhandenen 3D-Systeme zur Planung und Bewertung menschlicher Arbeit. Zu wenig vertraute Bedienmetaphern und eingeschränkte Bezüge zu vorhandenen und bewährten Planungsmethoden erschweren den Zugang und lassen die Akzeptanz im Keim ersticken. [7, p. 86]

Das zentrale Problem der Menschmodell-Simulationssysteme stellt die bisherige Herangehensweise beim Aufbau der Szenarien und insbesondere bei der Modellierung der menschlichen Bewegungen dar. Hier wird außer Acht gelassen, dass der Planungsingenieur zwar durchaus eine Vorstellung davon hat, wie ein Pro-

zess ablaufen soll. Allerdings fällt ihm (wie auch jedem anderen auf diesem Gebiet ungeübten Mitarbeiter) die Diskretisierung eines Gesamtprozesses auf menschliche Grundbewegungen bis hin zu einzelnen Körperhaltungen schwer. Da auch die einzelnen Körperhaltungen erst mühsam durch Manipulation der zahlreichen Gelenkwinkel des digitalen Menschmodells eingestellt werden müssen, ist der hierfür erforderliche Aufwand enorm. Untersuchungen hierzu haben einen ungefähren Modellierungsaufwand von 250 zu 1 ergeben. Dies bedeutet, dass für eine Minute Simulationszeit derzeit circa 250 Minuten Modellierungszeit aufgewendet werden müssen [8, p. 126]. Es ist gängige Praxis, die Bewegungen eines Menschmodells anhand aneinandergereihter Einzelposen zu modellieren – doch diese Aufgabe wird derzeit aufgrund mangelhafter Konzepte und fehlender alternativer Modellierungsebenen einfach auf den Benutzer übertragen.

„Der Planer ist kein Trickfilm-Ersteller“ – dieser prägnante Ausspruch fasst das gegenwärtige Problem zusammen, vor welchem jeder Planungsingenieur bei der Anwendung vorhandener Systeme zur Absicherung menschlicher Arbeit steht. Problematisch ist nicht nur, dass die Modellierung einzelner Posen sehr zeitaufwändig ist und damit Aufwand und Nutzen in einem nicht mehr tolerierbaren Ungleichgewicht stehen, sondern auch, dass zur Ausbildung und zum Wissen eines Planers eben nicht die Anatomie des menschlichen Bewegungsapparates und dessen Funktions- und Belastungsschemata gehören. Damit ist die traditionelle Modellierung menschlicher Tätigkeiten zur Planungsabsicherung nicht nur aus wirtschaftlicher Sicht fragwürdig. Es stellt sich auch die Frage, ob die durch einen Laien modellierten Posen und deren zeitliche Verläufe in ihrer anschließenden Bewertung überhaupt ein korrektes und reproduzierbares Ergebnis liefern können oder ob das Ergebnis vielmehr sehr stark von den „Modellierungsfähigkeiten“ des Bearbeiters abhängt. Diese Tatsache führt in einigen Unternehmen dazu, dass sich eigens auf die Menschmodellierung spezialisierte Mitarbeiter, ohne oder mit nur geringem Fertigungsprozesswissen, tagtäglich nur mit der Simulation beschäftigen müssen. Es wird also versucht, den Problemen der vorhandenen Systeme durch Manpower zu begegnen, was sicherlich weder ein zukunftsfähiges Konzept noch eine Lösung des Grundproblems darstellt.

1.3 Formulierung der Aufgabe

Trotz der Tatsache, dass bereits zahlreiche Systeme zur Nutzung digitaler Menschmodelle entwickelt worden sind, wird deren Potential, insbesondere für die Fertigungsprozessplanung, noch bei weitem nicht ausgeschöpft. Die Gründe hierfür lassen sich in fünf Hauptpunkten zusammenfassen. Aktuelle Simulationssysteme zur Modellierung menschlicher Arbeit ...

- ... sind Speziallösungen, deren Anwendungsfokus auf wenige Teilaspekte menschlicher Arbeit beschränkt ist. (Mangel an Universalität der Funktionalität)
- ... konzentrieren sich auf die Bewegungsmodellierung und vernachlässigen dabei den Planungsaspekt. (Mangel an Zielorientiertheit der Anwendung)
- ... sind nicht intuitiv bedienbar – ihre Handhabung setzt ein hohes Maß an Übung und Erfahrung des Anwenders voraus. (Mangel an Effizienz der Modellierung)
- ... liefern nur bedingt objektive Ergebnisse, da diese sehr stark von den Fähigkeiten des Anwenders abhängen. (Mangel an Reproduzierbarkeit der Ergebnisse)
- ... sind „Insellösungen“, die zwar in die Systemwelt des jeweiligen Herstellers eingebunden sind, denen es aber an Schnittstellen zu anderen Systemen mangelt. (Mangel an Übertragbarkeit der Methode)

Der Autor hat sich zum Ziel gesetzt, ein System zu konzipieren und umzusetzen, welches die oben beschriebenen Mängel nicht aufweist bzw. weitestgehend behebt. Hierbei ist das Kernelement der Forschung die Entwicklung einer Methode zur Ansteuerung digitaler Menschmodelle. Die Definition der Simulation soll in einer dem Fertigungsprozessplaner vertrauten Beschreibungsform erfolgen. Einzelne Körperhaltungen sollen ausschließlich Resultate algorithmischer Berechnungen sein und nicht mehr durch die Manipulation der Gelenkstellungen einzelner Körperteile hergestellt werden müssen. Durch die Vereinfachung und Formalisierung der Bewegungsgenerierung digitaler Menschmodelle soll deren effizienter Einsatz mit anschließender objektiver Ergebnisbewertung ermöglicht werden.

1 Einleitung

Das zu entwickelnde System soll sich in das vorhandene methodische Umfeld der Fertigungsprozessplanung und der Digitalen Fabrik integrieren. Wichtige Schnittstellen existieren dabei insbesondere zu den Bereichen 3D-Planungsabsicherung, Zeitwirtschaft und Ergonomie. Alle drei genannten Bereiche liefern sowohl Eingangsdaten, sollen im Gegenzug aber auch auf Ergebnisdaten des neuen Simulationssystems zurückgreifen können. So liefert der Bereich der 3D-Planungsabsicherung beispielsweise die relevanten geometrischen Informationen der Produkte bzw. Ressourcen und deren Anordnung am Arbeitsplatz in Form des sogenannten »Layouts«. Als verwertbares Ergebnis liefert die Simulation Erkenntnisse, welche zur Optimierung vorhandener Layouts und zur Auswahl möglicher Alternativ-Ressourcen dienen können. Der Bereich Zeitwirtschaft liefert Vorlagen zur Gliederung von Aufgaben und definiert gleichzeitig eine dem Planer vertraute Form der Beschreibung von Arbeitsinhalten. Gegenüber der tabellarischen Zeitanalyse liefert eine 3D-Simulation neben der Verifizierung der vorgesehenen Sollzeiten auch den Nachweis der Durchführbarkeit unter Berücksichtigung der Umgebung und hierbei insbesondere parallel stattfindender Maschinenbewegungen, relevanter Prozesszeiten und anderer am gleichen Arbeitsplatz tätiger Werker. Aus dem Fachbereich Ergonomie können Bewegungs- und Haltungsregeln zur Vermeidung körperlicher Schädigungen übernommen werden. Umgekehrt können die durch die Simulation erzeugten Bewegungsdaten als objektive Eingangsdaten für die nachfolgende Bewertung des simulierten Arbeitsplatzes verwendet werden.

Natürlich ist auch die eigentliche Ergebnis-Visualisierung der Simulation ein sehr bedeutsamer Aspekt. Hier sollen die bislang alphanummerisch gehaltenen Daten in Form von 3D-Animationen sichtbar gemacht und auch für Mitarbeiter ohne Fertigungsprozessplanungs-Hintergrund interpretier- und nachvollziehbar dargeboten werden. Mit dem angestrebten System soll somit ein Werkzeug zur Validierung und Optimierung von Planungsergebnissen bei gleichzeitiger Generierung von Dokumentations-, Präsentations- und Schulungsmaterialien entstehen. (Abbildung 1)

1 Einleitung

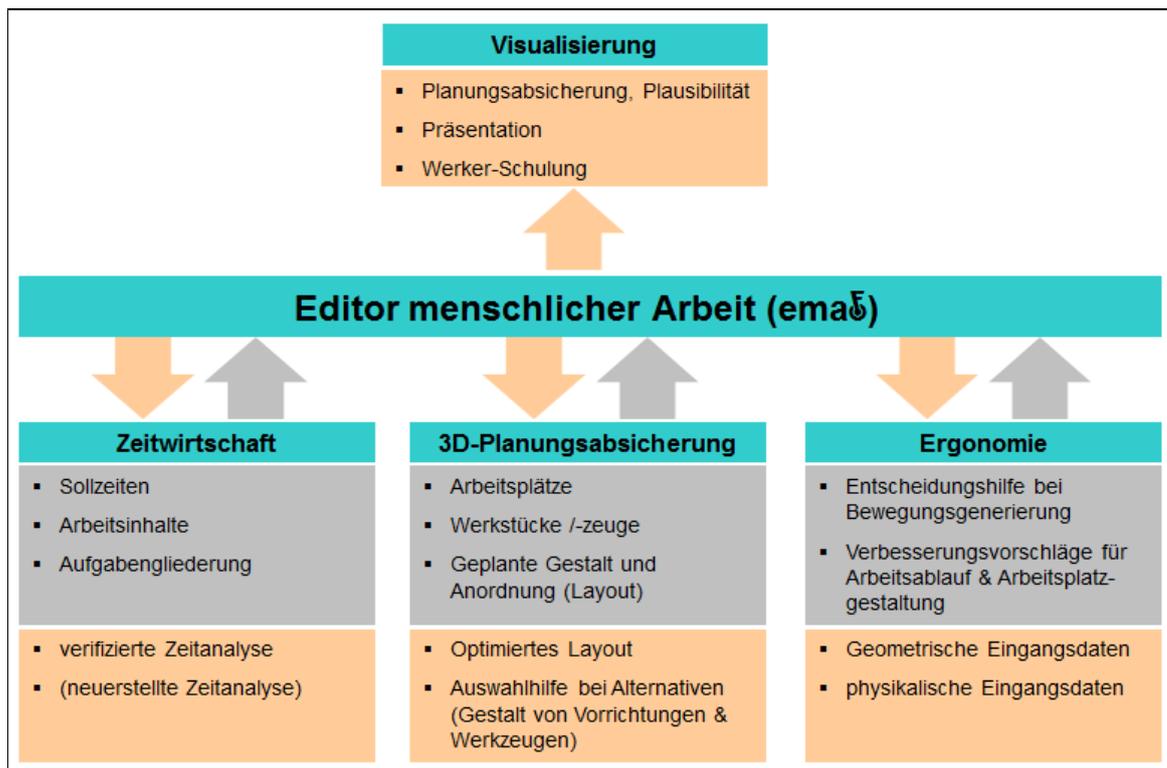


Abbildung 1: Fachliche Einordnung als Säulenmodell

1.4 Ergebnisse

Ein System zur Ansteuerung von digitalen Menschmodellen zu entwickeln, welches deren Einsatz zu Planungszwecken praktikabel machen sollte, war der Ausgangspunkt für die vorliegende Arbeit. Da auf dem Markt bereits eine Anzahl von Planungssystemen mit integrierten digitalen Menschmodellen existierte, beschäftigte sich der Autor zunächst mit deren Funktionsweisen, Leistungsumfängen und besonders intensiv mit deren Vor- und Nachteilen aus Anwendersicht. Im Laufe der fachlichen Untersuchung der Systeme konnten bereits Erkenntnisse über deren technische Realisierung und Limitationen gewonnen werden. Der Hauptkritikpunkt aus Anwendersicht bestand durchgängig in der sehr zeitaufwändigen und Expertenwissen voraussetzenden Modellierung von Bewegungen. Während die Generierung einzelner Körperhaltungen von nahezu allen Systemen durch relativ komfortable Funktionen unterstützt wird, ist es dem Benutzer zumeist allein überlassen, die einzelnen Körperhaltungen zu plausiblen, menschlichen Bewegungen zu verknüpfen. Der Zeitaufwand für die Erstellung und vor allem für die Anpassung von Simulationen digitaler Menschmodelle galt deswegen als größte Hürde für deren flächendeckenden Einsatz im produktiven Planungsgeschäft.

Somit war die Richtung der nachfolgenden Forschungs- und Entwicklungsaufgaben klar umrissen. Ein System sollte geschaffen werden, welches auf möglichst effiziente Art und Weise in der Lage ist, Bewegungen für digitale Menschmodelle zu erzeugen und sich darüber hinaus möglichst nahtlos in die tägliche Arbeit des Fertigungsprozessplaners integrieren lässt. Als nächstes galt es herauszufinden, wie die Schnittstellen eines solchen Systems gestaltet werden mussten, damit die angesprochene Integration erfolgreich vollzogen werden konnte. Die auf diesem Gebiet vorherrschende Methodik ist der Einsatz von Systemen vorbestimmter Zeiten und dabei insbesondere von Systemen nach den Standards von »MTM« (»Methods Time Measurement«). Die MTM-Verfahren lieferten detaillierte Grundlagen für die Strukturierung und zeitliche Bewertung von Bewegungsvorgängen für Werker und damit auch für deren digitale Abbilder. Darüber hinaus besteht die Methode »MTM« aus mehreren, aufeinander aufbauenden Teil-Verfahren, welche einen Ansatz für eine mögliche Art der Simulations-Ablaufstrukturierung lieferten. Die MTM-Bausteine sind in der Lage, sowohl Mikro-Bewegungen, wie beispielsweise einzelne Handgriffe

oder Armbewegungen, zeitlich abzubilden als auch komplexere Makro-Bewegungen, wie beispielsweise Aufnehmen und Platzieren, welche wiederum aus Mikro-Bewegungen zusammengesetzt sind. Trotz der Tatsache, dass die einzelnen MTM-Verfahren methodisch aufeinander aufsetzen, lassen sie sich nur eingeschränkt direkt zur Simulations-Generierung übernehmen. Da der Fokus von MTM primär auf die Zeitermittlung gerichtet ist, fehlen die für eine 3D-Simulation relevanten Positions- und Objektbezüge nahezu vollständig. Darüber hinaus sind die in der Automobilindustrie eingesetzten höherverdichteten MTM-Verfahren (MTM-UAS, MTM-MEK und OEM-spezifische Derivate) zum Teil schon so stark zusammengefasst und mit zahlreichen Regeln belegt, dass eine darauf aufbauende effiziente Bildung von Simulations-Algorithmen nicht möglich war. Darüber hinaus waren spezielle Bewegungen, wie bspw. ein „Einsteigen in ein Fahrzeug“ oder ein „Rutschen zu einem anderen Ort“ nicht vorgesehen, was die Anwendbarkeit weiter einschränkte. Daher entwarf der Autor eine auf die Anforderungen der Menschmodell-Simulation hin angepasste Prozesssprache und verwendete dabei den Begriff der »Verrichtung« als Abgrenzung von parametrischen und simulationsfähigen Arbeitsinhalten. Die Entwicklung dieses neuen, frei kombinierbaren und universellen Systems von aufeinander aufbauenden Verrichtungs-Bausteinen war einer der ersten großen Meilensteine auf dem Weg zu einer neuen Generation von Simulationssystemen für menschliche Arbeit.

Während die Entwicklung der sogenannten »Verrichtungsbibliothek« nach dem Bottom-Up-Prinzip und der damit verbundenen Kombination elementarer Bausteine zu komplexen Arbeitsabläufen folgte, beschäftigte sich der Autor zeitgleich mit der Frage, wie die Verrichtungen letztlich in Bewegungen digitaler Menschmodelle umgesetzt werden konnten. Der Ausgangspunkt dieser Untersuchungen waren Analysen hinsichtlich der Topologie und der Kinematik von existierenden Menschmodellen. Hierbei musste der Autor feststellen, dass sich sämtliche Menschmodelle trotz der gemeinsamen realen Vorlage teilweise in ihrem internen Aufbau unterscheiden, was wiederum einem problemlosen (Bewegungs-)Datentransfer zwischen den einzelnen Menschmodellen und den sie umgebenden Systemen erschwert. Hinzu kam die Anforderung, dass das zu entwickelnde Simulationssystem so flexibel sein sollte, dass es auch auf unterschiedliche Perzentile von digitalen Menschmodellen angewendet werden kann. Die festgestellte Varianz in den grundlegenden Strukturen vorhandener

Menschmodelle und die zusätzlich erwarteten Unterschiede durch die geplante Unterstützung beliebiger Perzentile führten den Autor zu dem Schluss, dass die Logik der Verrichtungs-Bausteine von der Logik zur Generierung von Bewegungen entkoppelt werden sollte. Darüber hinaus sollte hierdurch auch dem festgestellten Mangel an Zielorientiertheit anderer Systeme entgegengewirkt werden – der Anwender soll Tätigkeiten und Abläufe planen, während die Bewegungen automatisch aus dieser Planung abgeleitet werden.

Als Ergebnis dieser Entkoppelung entstand die Idee eines zweigeteilten Prozesses zur Bewegungsgenerierung. Die vom Benutzer getätigten Eingaben, welche sowohl den zu simulierenden Tätigkeitsverlauf als auch dessen Parametrisierung (Art der Tätigkeitsausführung, Objekt- und Geometriebezüge) beinhalten, werden zunächst in einer sogenannten »Absichtskomponente« verarbeitet. Als Kern der »Absichtskomponente« dienen dabei die im Aufbau befindliche »Verrichtungsbibliothek« und ein entsprechendes »Simulationsmodell«. Spezielle Algorithmen bilden intern die Abhängigkeiten und Beziehungen zwischen den einzelnen Verrichtungen ab, während vorwiegend geometrische Komplexfunktionen zur Nachbildung von Werker-Intelligenz (z.B. Laufpfad-, Greifpunkt- und Objekt-Handhabungs-Bestimmung) eingesetzt werden. Das Resultat dieser ersten Phase ist eine »abstrakten Bewegungsbeschreibung«. Hierbei handelt es sich um eine vom konkreten Menschmodell abstrahierte Folge von Bewegungszuständen, welche lediglich Informationen zu gewünschten Zielvorgaben oder Zuständen einzelner Körperteil-Gruppen und relevante Objektbezüge beinhaltet. In einem anschließenden Schritt werden diese Rahmenbedingungen genutzt, um sie durch einen »Bewegungsgenerator« auf ein konkretes Menschmodell anzuwenden und dabei konkrete Körperhaltungen, bestehend aus Konfigurationen von Gelenkwinkeln, zu berechnen. Der Bewegungsgenerator arbeitet vollautomatisch und stellt dabei die Reproduzierbarkeit der Ergebnisse sicher, was beim „traditionellen“ per Hand Modellieren einzelner Körperhaltungen bisher nicht der Fall gewesen ist. Die berechneten Körperhaltungen können abschließend als Ergebnis der Simulation visualisiert und ausgewertet werden. Um die theoretischen Erkenntnisse und die entwickelten Konzepte hinsichtlich ihrer Praxistauglichkeit zu verifizieren, entwickelte der Autor eine Reihe von Software-Prototypen (siehe Tabelle 1 bis Tabelle 5).

1 Einleitung

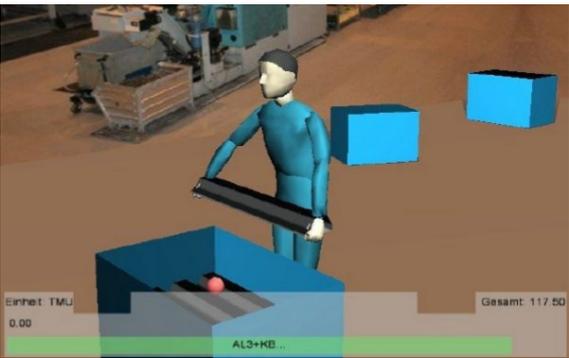
	Software	MTM-UAS-Simulation
	Jahr	2005
	Technologie	Virtools-Skript, C++
Getestete Konzepte	<ul style="list-style-type: none"> - Verrichtungsmethodik - Integration der MTM-Methodik 	

Tabelle 1: MTM-UAS-Simulation (Prototyp) [9, p. 59]

	Software	Editor menschlicher Arbeit (Virtools Prototyp)
	Jahr	2006
	Technologie	Virtools-Skript, C++
Getestete Konzepte	<ul style="list-style-type: none"> - Zeitleisten-Bedienmetapher mit Drag-and-Drop - simultane Simulation mehrerer Menschmodelle inklusive Synchronisierung 	

Tabelle 2: Editor menschlicher Arbeit (Virtools Prototyp) [10, p. 70]

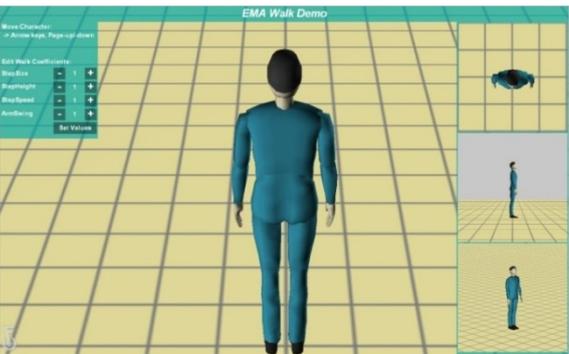
	Software	Parametrisches Laufen
	Jahr	2007
	Technologie	Virtools-Skript, C++
Getestete Konzepte	<ul style="list-style-type: none"> - Echtzeit-Bewegungsgenerierung (Laufbewegung) 	

Tabelle 3: Parametrisches Laufen (Prototyp)

Bei den ersten Tests diente dabei zunächst das 3D-Autorensystem »Virtools Dev« (Dassault Systèmes) als Werkzeug. »Virtools« lieferte die gesamte Infrastruktur, welche zum Laden und Verwalten von 3D-Modellen erforderlich war.

1 Einleitung

Darüber hinaus bot es durch die interne Skript-Sprache eine mächtige und effiziente Möglichkeit, um gemäß der Methodik des »Rapid-Prototyping« schnell zu Ergebnissen zu gelangen. Da es sich bei den »Virtools«-Prototypen um eigenständige Programme bzw. Technik-Demos handelte und bereits früh die Integration der Methodik in vorhandene 3D-Planungssysteme diskutiert wurde, folgte kurz darauf eine Beispiel-Implementierung für das PLM-System »DELMIA« (Dassault Systèmes). Hierbei konnte auch erstmals das Prinzip der zweiphasigen Bewegungsgenerierung in Anwendung auf ein weiteres Menschmodell (den »Human V5«) erfolgreich umgesetzt werden.

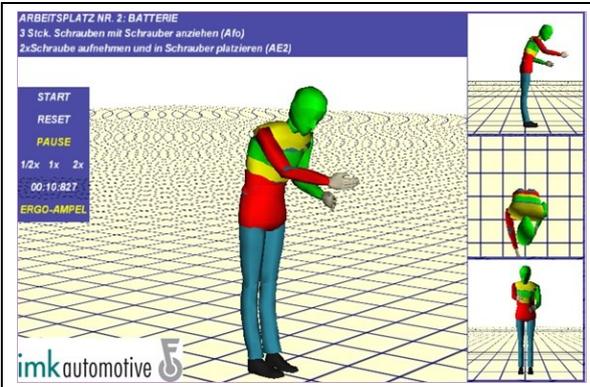
	Software Ergonomie-Visualisierung Jahr 2007 Technologie Virtools-Skript, C++ Getestete Konzepte <ul style="list-style-type: none"> -automatisches MTM-UAS-Baustein-Mapping -Integration von Ergonomie-Methoden
--	---

Tabelle 4: Ergonomie-Visualisierung (Demonstrator)

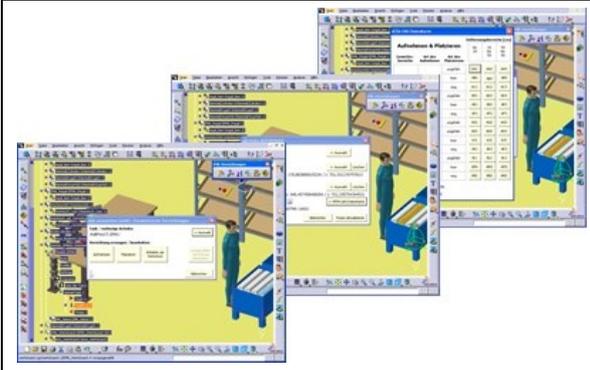
	Software ema DELMIA-Prototyp Jahr 2008 Technologie VBA Getestete Konzepte <ul style="list-style-type: none"> -Zweistufige Bewegungsgenerierung -Fremd-System-Integration
---	---

Tabelle 5: ema DELMIA-Prototyp

Die entwickelten Prototypen dienten jeweils nur zum Testen einzelner Aspekte des geplanten Systems. Mit dem Beginn der Entwicklungsarbeiten am Software-Produkt »Editor menschlicher Arbeit« (kurz: »ema«) wurden die gesammelten Erkenntnisse schließlich zusammengeführt. Als erster Schritt entstand dabei

1 Einleitung

»ema⁵V5«, ein funktionales Plug-In für »DELMIA V5«. (Tabelle 6) Die im Rahmen dieser Arbeit konzipierten Elemente »Verrichtungsbibliothek« und »Simulationsmodell« bilden dabei den elementaren Kern des Simulationssystems der Software. Darüber hinaus dient ein eigens entwickelter Bewegungsgenerator zur Berechnung visualisierbarer Simulationszustände. Über eine Schnittstelle zum PLM-System können die Simulationsergebnisse auf das digitale Menschmodell »Human V5« übertragen und somit ausgegeben werden.

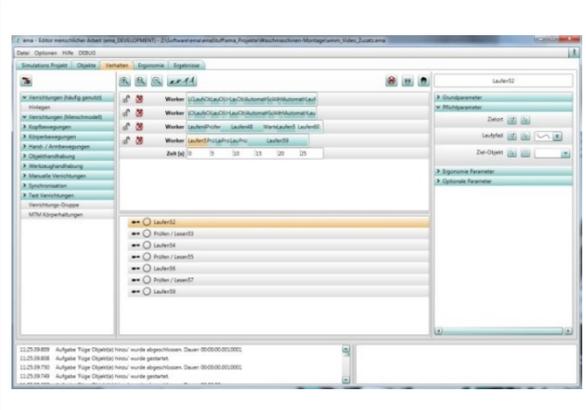
	Software	ema ⁵ V5
	Jahr	2010
	Technologie	.NET-Framework, C#
Getestete Konzepte	<ul style="list-style-type: none"> - Funktionales Plug-In für DELMIA V5 - Planung und Simulation menschlicher Arbeit - Ergonomie-Bewertung mit EAWS-Methodik 	

Tabelle 6: ema⁵V5

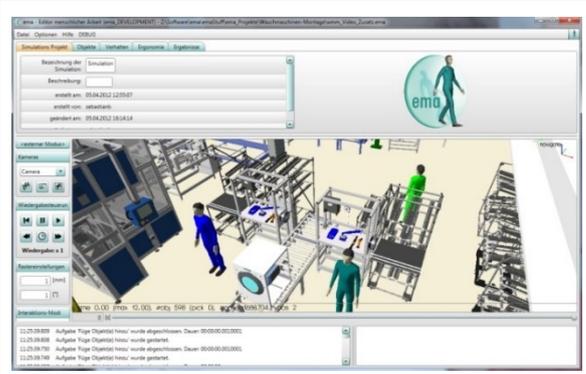
	Software	ema ⁵
	Jahr	2012
	Technologie	.NET-Framework, C#
Getestete Konzepte	<ul style="list-style-type: none"> - eigenständige Software mit 3D-Engine »eView« - Layout-Funktionalitäten - Menschmodell-Bibliothek - Collada-Schnittstelle 	

Tabelle 7: ema⁵

Ein weiterer Schritt war die Weiterentwicklung des »ema⁵V5« zur eigenständigen Software »ema⁵«. (Tabelle 7) Hierbei wurde die Fremdsystem-Integration durch die Anbindung der 3D-Engine »eView« der Firma »3DInsight« ersetzt und eigene Funktionalitäten zur einfachen Layout-Planung ergänzt. Dank der Flexibilität der verwendeten Systemarchitektur konnten die Kernkomponenten

1 Einleitung

und dabei insbesondere der Simulationskern von »ema5V5« bis auf die Anpassung einzelner Schnittstellen übernommen und dadurch in Zukunft gemeinsam weiterentwickelt werden. Die Ergebnisse dieser Arbeit konnten somit bereits in zwei kommerziell verfügbaren Software-Systemen angewendet und hinsichtlich ihrer Praxistauglichkeit verifiziert werden.

2 Einordnung und Forschungsumgebung

2.1 Planung, Gestaltung und Analyse menschlicher Arbeit

Trotz aller technologischer Fortschritte unserer Zivilisation ist es noch immer der Mensch, der im Zentrum der Leistungserstellung eines jeden Unternehmens steht. Die innerhalb eines Betriebes durch den Mensch erbrachten Arbeitsleistungen werden dabei in »dispositiv« und »operativ« unterschieden. [11, p. 358] Während erstgenannte Arbeiten hauptsächlich die leitenden und lenkenden Aufgaben im Betrieb umfassen, dienen die operativen bzw. objektbezogenen Aufgaben direkt der Leistungserstellung und -verwertung. [12] Die Art und Weise, wie diese Arbeitsleistungen erbracht und organisiert werden, hat sich im Laufe der Zeit grundlegend verändert. Mit der Erfindung von Dampfmaschinen, Motoren und der Anwendung von Informationstechnologie wandelte sich die Herstellung von Produkten vom Handwerk hin zur industriellen Fertigung. [13, p. 3] Am Anfang des 20. Jahrhunderts und mit der Entwicklung von Frederick Winslow Taylors „Taylorismus“ [14] standen dabei vor allem die Effizienzsteigerung und Rationalisierungsmaßnahmen in der Produktion im Vordergrund. Diese Maßnahmen wirkten sich jedoch negativ auf die Flexibilität der Produktionsprogramme und die Arbeitsbedingungen in den Betrieben aus. Henry Ford adaptierte Taylors Methoden und erweiterte diese im „Fordismus“ u.a. durch die Schaffung einer Sozialpartnerschaft zwischen Arbeitern und Unternehmen. [15, p. 5] Der damit einhergehende gesellschaftliche Aufstieg der Arbeiter und der wachsende Einfluss der Gewerkschaften führten schließlich zu einem Mitspracherecht der Arbeiter bei der Gestaltung der Arbeitsbedingungen. Neben dem gestiegenen Einfluss der Arbeitnehmer sind es heute jedoch auch die durch den demografischen Wandel veränderten gesellschaftlichen Rahmenbedingungen, die sich direkt auf die Organisation der menschlichen Arbeit auswirken. (siehe Kapitel 1.1)

Im Bereich der industriellen Massenfertigung sind es die Methoden und Verfahren der Fabrikplanung, welche die später in den jeweiligen Fabriken zu verrichtenden menschlichen Arbeiten vorbereiten. Dabei sind zahlreiche Rahmen- und Randbedingungen zu berücksichtigen, um die Arbeitsplätze auf der einen Seite optimal zu gestalten und auf der anderen Seite gleichzeitig die betrieblichen Investitionsvorhaben rationell umzusetzen. [16, p. 3] Die Aktivitäten zur Erreichung dieser Ziele lassen sich in strategische, auf langfristige Wirksamkeit hin

ausgerichtete, Elemente und in operative, zur Realisierung konkreter Produkte benötigte, Elemente gliedern. [13, p. 5]

Der Lebenszyklus eines Produkts beginnt bei der ersten Produktidee. Im Produktdefinitionsprozess (PDP) wird diese Idee in Produktvorschläge überführt. Aus den Vorschlägen können nun im Produktentstehungsprozess (PEP) detaillierte Konzepte und Umsetzungsvorschläge generiert werden. Parallel zu dieser Phase der Produktentwicklung kann beim Erreichen eines entsprechenden Reifegrades die Fertigungsplanung des Produktes begonnen werden. Die Planung, wie das Produkt in Zukunft gebaut werden soll, wird in ihrer Endphase vom Beginn der eigentlichen Produktion überlagert – hier werden bereits frühzeitig die vorgedachten Konzepte und Fertigungspläne in der Realität erprobt. Sobald die ersten vollfunktionsfähigen Produkte erzeugt worden sind, beginnt die Nutzungsphase. Hierbei spielen vor allem Instandhaltung, Verbesserungen und Anpassungen des Produkts an das Nutzungsumfeld eine wichtige Rolle. Wenn das Produkt das Ende seiner Lebenszeit erreicht hat, folgt schließlich die Außerbetriebnahme, die Beseitigung und das Recycling seiner Bestandteile.

Falls für das neue Produkt eine neue Fabrik gebaut werden muss, beginnt die Planung mit der Auswahl eines geeigneten Standortes (Standortplanung). Hierbei werden die relevanten Standortfaktoren analysiert und zur Selektion verwendet. Anschließend folgt die Generalbebauungsplanung, welche die Bebauung des gewählten Standortes und dessen Anbindung an die Infrastruktur vorbereiten soll. Die eigentliche Planung der Fertigung des neuen Produkts besteht danach hauptsächlich aus der Planung der sogenannten »Flusssysteme« der zukünftigen Produktion. Diese Planungsphase lässt sich wiederum in einzelne Planungsschritte untergliedern [13, p. 8]:

- Analyse des Produktionsprogramms, zur Festlegung sachlicher, mengenmäßiger, wertmäßiger und zeitlicher Aspekte
- Funktionsbestimmung, zur Definition der Funktionen, Prozesse und Elemente im Produktionssystem
- Dimensionierung, zur Bestimmung der Anzahl von benötigten Elementen mit dem Ziel der Abschätzung der notwendigen Investitionen und Betriebskosten
- Strukturierung, zur Bestimmung räumlicher und zeitlicher Abhängigkeiten von Flusssystemelementen mit dem Ziel, diese optimal anzuordnen

- Gestaltung, zur Einordnung der ermittelten Flusssystemelemente in die Produktionsumgebung (Layout-Erzeugung)

[13, p. 27ff]

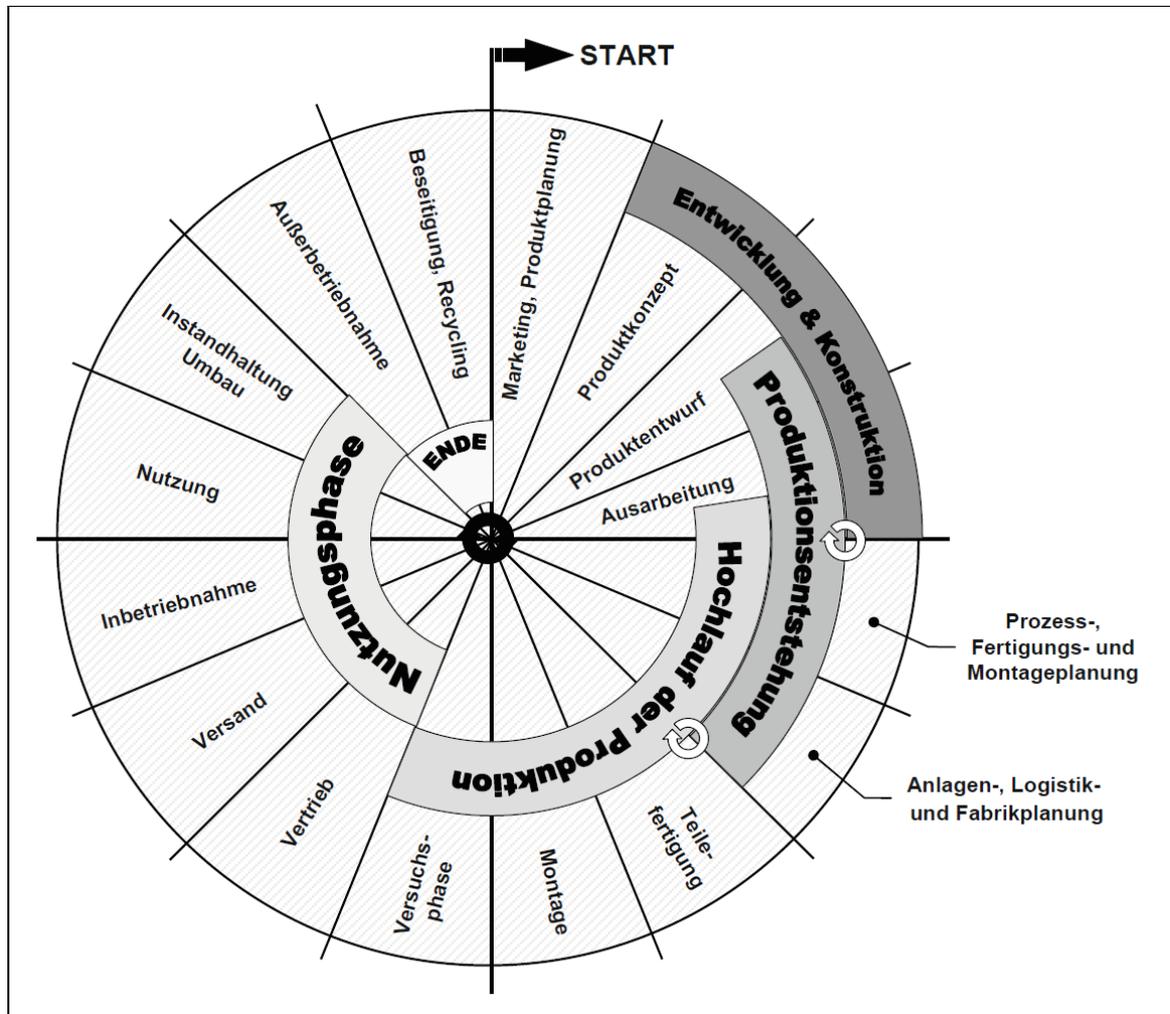


Abbildung 2: Referenzmodell für den Produktentstehungsprozess [17, p. 239]

Die Planung menschlicher Arbeit findet hauptsächlich in den drei letztgenannten Planungsschritten statt. Bei der Dimensionierung werden die Anzahl der benötigten Arbeitsplätze und Personen in der zukünftigen Produktion festgelegt. Als Bestimmungsgrundlage werden hierzu die benötigten »Soll-Zeiten« der zuvor bestimmten Abläufe ermittelt. Wenn diese Zeiten „... auch Anteile für nicht genau vorausbestimmbare Ablaufabschnitte [beinhalten], so nennt man

solche Soll-Zeiten [...] Vorgabezeiten“. [18, p. 227] Die Ermittlung der Soll-Zeiten kann dabei durch unterschiedliche Methoden erfolgen:

- Zusammensetzung unter der Verwendung von »Systemen vorbestimmter Zeiten« (kurz: »SvZ«) oder Planzeiten (resultieren aus ggf. vorhandenen »Ist-Zeiten«)
- Ermittlung durch Vergleichen und Schätzen
- Ermittlung durch Berechnung (für Betriebsmittel)

[18, pp. 246-247]

Im Planungsschritt »Strukturierung« erfolgt die Anordnung der Produktions-Elemente unter Berücksichtigung der zeitlichen und räumlichen Beziehungen der Elemente untereinander. Die zeitliche Strukturierung umfasst die Bestimmung der Arbeitsvorgänge und ihrer Abfolge, ebenso wie die Bestimmung von Fertigungs- und Losgrößen. Bei der räumlichen Strukturierung werden u.a. die Arbeitsaufgaben mit den an ihnen beteiligten (Teil-)Produkten und Ressourcen verknüpft. Anschließend wird entschieden, welche Elemente ortsfest bzw. ortsveränderlich sein werden. Für die ortsfesten Elemente werden dann die entsprechenden Standorte bestimmt, während für die beweglichen Elemente zugehörige Bewegungspfade definiert werden. [13, p. 41ff] Die Verknüpfung zwischen den Arbeitsplätzen und den Arbeitsgängen wird als »Abtaktung« bezeichnet. Dabei sind die wichtigsten Einflussfaktoren die Fügefolge des Produkts, die zur Verfügung stehende Taktzeit und Einschränkungen durch die an den jeweiligen Arbeitsplätzen verfügbaren Logistikflächen. [19, p. 239]

Im abschließenden Planungsschritt »Gestaltung« werden alle Elemente der Fertigung in die reale Werkhalle bzw. Fabrik eingeordnet. Dabei stehen ökonomische und ökologische Aspekte sowie der Arbeits- und Gesundheitsschutz im Fokus. Das Ergebnis dieses späten Planungsschrittes stellt das sogenannte »Layout« dar, was als „... grafische Darstellung der räumlichen Anordnung von Funktionseinheiten unterschiedlichen Abstraktionsgrades ...“ [20, p. 158] definiert ist. Während das Layout einen Überblick der geplanten Fertigung liefert, werden arbeitswissenschaftliche Methoden eingesetzt, um Arbeitsplätze und Tätigkeiten so zu gestalten, dass die Arbeitsbelastung für die Werker möglichst gering gehalten wird. Die dabei relevante Prozessergonomie zielt daher auf eine „optimale wechselseitige Anpassung zwischen dem Menschen und seinen Arbeitsbedingungen“ [21] ab. Um die auftretenden Belastung an Arbeitsplätzen zu

2 Einordnung und Forschungsumgebung

bestimmen, kommen »Screening-« oder Messverfahren zum Einsatz. Zu den verbreitetsten Verfahren der Analyse und Bewertung ergonomisch relevanter Belastungen zählen: [22, p. 10]

- BGI/GUV-I 7011 [23]
- BAuA Leitmerkmalmethoden [24]
- KoBRA-Methoden, wie z.B. Schwellenwerttabelle für Körperkräfte [25]
- NIOSH [26], OWAS [27], RULA [28]
- AAWS [29], EAWS [30]
- DIN-Normen (DIN EN 1005-2, DIN EN 1005-3, ISO 11228-1, ISO 11228-2)
- Messverfahren, z.B.: Elektromyographie [31]

Die für die »Screening-Verfahren« notwendigen Daten werden durch die Beobachtung realer Arbeitstätigkeiten ermittelt und dokumentiert. Im Gegensatz hierzu werden die Messverfahren aufgrund der hierzu notwendigen Hardware zumeist unter Laborbedingungen durchgeführt.

2.2 Prozesssprachen in der Fertigungsplanung

Das Ziel eines jeden produzierenden Unternehmens ist die Steigerung der eigenen »Produktivität«. Dabei ist mit Produktivität die Rentabilität der Faktoren Arbeit, Boden und Kapital im Betrieb gemeint. Auch wenn die Produktivität nicht als alleiniger Erfolgsgarant fungieren kann, so ist sie doch eine wichtige Voraussetzung für ökonomischen Erfolg. [32, p. 3f] Um die Produktivität sicherzustellen bzw. diese zu verbessern, ist es notwendig, mögliche Probleme zu identifizieren, deren Ursachen zu bestimmen und daraus geeignete Lösungen zu generieren. Die Dokumentation und Lösungsfindung bei der Planung menschlicher Arbeit erfolgt dabei mit Hilfe von Beschreibungsmodellen.

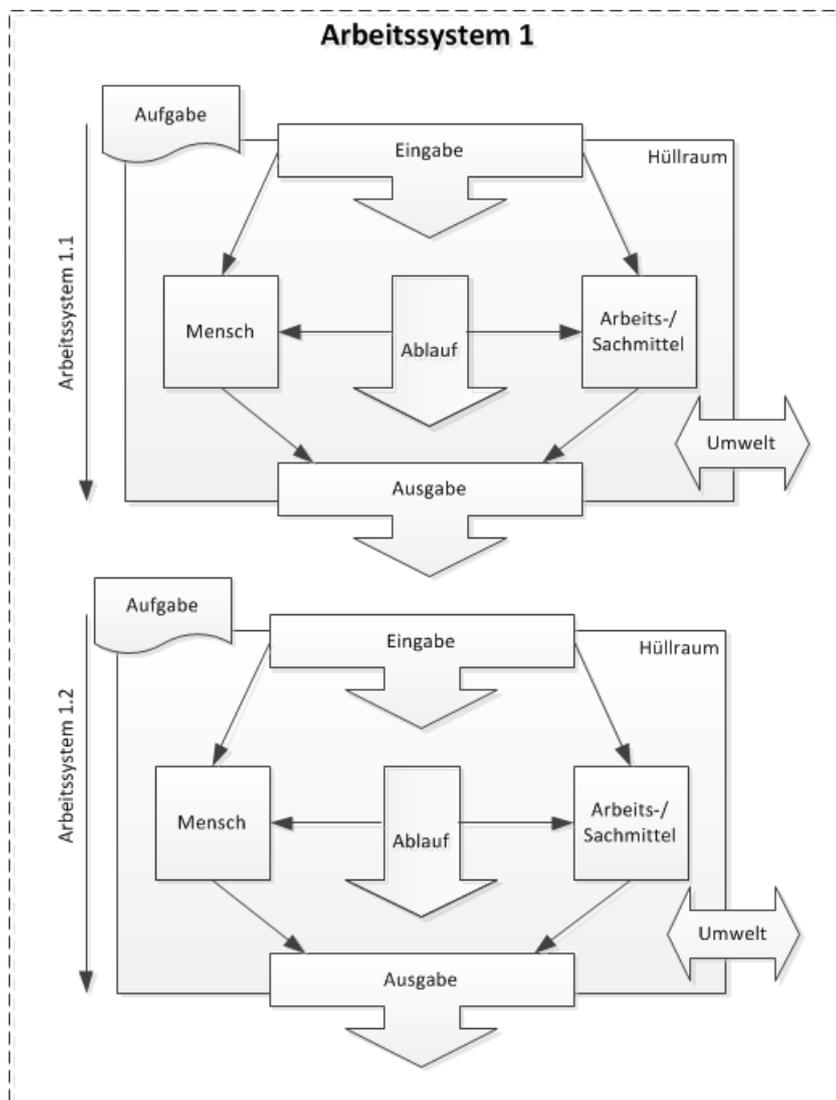


Abbildung 3: Das Arbeitssystem als Modell [32, p. 31]

Ein solches Modell ist das genormte Beschreibungsmodell »Arbeitssystem«. [33]. Die Bestimmungsgrößen des Modells und deren gegenseitigen Beziehungen sind in Abbildung 3 dargestellt. Zunächst zeigt die Abbildung, dass Arbeitssysteme im Allgemeinen in Teilsysteme untergliedert werden, welche sowohl hierarchisch als auch nebeneinander angeordnet sein können. Ein einzelnes Arbeitssystem hat eine explizite Aufgabe und wird durch einen sogenannten »Hüllraum« von seiner Umgebung getrennt. Diese Trennung reduziert die Interaktion des Arbeitssystems mit seiner Umgebung bzw. anderen Arbeitssystemen auf die beiden Hauptschnittstellen »Eingabe« / »Ausgabe« und möglicherweise auftretende Umwelteinflüsse (bidirektional). Der Kern des Arbeitssystems wird vom internen Arbeitsablauf und dessen Wechselwirkungen mit den Ressourcen »Mensch« und »Arbeits-/Sachmittel« gebildet. [34, p. 35f] Während der Ablauf jedoch nur für die Abfolge der Geschehnisse steht, wird „... das zeitliche und räumliche Zusammenwirken der Ressourcen Mensch und Arbeits-/Sachmittel, bei dem eine Transformation der Eingabe (Prozessinput) in die Ausgabe (Prozessoutput) vollzogen wird“ [32, p. 34], als »Prozess« bezeichnet.

Die Beschreibung von (Arbeits-)Prozessen kann auf vielfältige Art und Weise erfolgen. Beispiele hierfür sind u.a.:

- (Fluss-)Diagramme
- natürlich-sprachliche Beschreibungen
- (Prozess-)Bausteinsysteme

Während sich Flussdiagramme eher für den groben Überblick der Abfolge von Prozesselementen eignen und inhaltliche Details ausblenden, ist der Informationsgehalt natürlich-sprachlicher Beschreibungen sehr stark vom Verfasser selbiger abhängig. Um Probleme jedoch bestmöglich erkennen und beheben zu können, ist eine strukturierte, transparente und alle relevanten Informationen beinhaltende Beschreibung von Prozessen notwendig. Prozesssprachen haben genau diesen Zweck – sie sollen die Prozessstrukturtransparenz [35, p. 39f] sicherstellen, damit alle an der Prozessplanung oder -optimierung beteiligten Personen eine gemeinsame Diskussions- und Entscheidungsgrundlage haben. Darüber hinaus sollen Prozesssprachen durch formalisierte Beschreibungsstrukturen und Anwendungsregeln die Komplexität des Beschreibungsgegenstandes nur soweit reduzieren, dass dieser präzise und objektiv erfasst werden

kann. Hierdurch wird implizit auch die Vergleichbarkeit und Reproduzierbarkeit der Beschreibungen gewährleistet.

Als Begründer des systematischen Produktivitätsmanagements werden Adam Smith und Frederick Winslow Taylor angesehen. Jedoch gilt der Amerikaner Frank Bunker Gilbreth als Urheber des Bewegungsstudiums und der »Systeme vorbestimmter Zeiten«. [32, p. 59] Dabei handelt es sich um „... Verfahren, mit denen für die vom Menschen voll beeinflussbaren Bewegungselemente Einflussgrößen erfasst und Soll-Zeiten bestimmt werden können.“ [18, p. 478] Seit Ihrer Entstehung in der Mitte des 20. Jahrhunderts haben zwei Verfahren an Bedeutung gewonnen. Während das »Work-Factor-Verfahren« (WF) hauptsächlich messbare Einflussgrößen für die Zeitbewertung heranzieht, werden in den Bausteinsystemen des »Methods-Time-Measurement Verfahrens« (MTM) auch zu beurteilende Einflussgrößen verwendet. Insbesondere das MTM-Verfahren hat sich im Laufe der Zeit in Deutschland als Prozesssprache und Planungswerkzeug verankert. [18, p. 480] Die primären Eigenschaften des MTM-Bausteinsystems sind:

- Modellbildungsimmanenz
- Simulationsfähigkeit
- Komplexitätsvariation
- Bezugsleistungstreue / Normleistungsbezug

[36, p. 19]

Dabei beschreibt »Modellbildungsimmanenz« die Fähigkeit, (Arbeits-)Prozesse als Modell abzubilden. Dies wird durch die Verwendung von Prozessbausteinen und unter Einhaltung der »Grundsätze ordnungsgemäßer Modellierung« gewährleistet [37]. Die Eigenschaft der »Simulationsfähigkeit« bezieht sich auf die Möglichkeit zur Untersuchung von „Was wäre wenn?“-Szenarien. Durch die Umstrukturierung oder das Ersetzen von Bausteinen können die Auswirkungen im Hinblick auf den Zeitbedarf des Prozessverlaufs quantifiziert werden. Die durch MTM definierten Bausteinsysteme enthalten dabei unterschiedliche sogenannte »Methodenniveaus«. Diese ermöglichen es, Prozesse sehr feingliedrig, z.B. bis auf die Ebene von Fingerbewegungen, oder alternativ auch in komprimierter Form („Hilfsmittel handhaben“) zu beschreiben. Durch diese als »Komplexitätsvariation« bezeichnete Eigenschaft wird die Anwendbarkeit auf verschiedene Komplexitätsgrade und somit auch verschiedene Arten von Pro-

zessen sichergestellt. Aufgrund der Entwicklungsgeschichte und der angewendeten wissenschaftlichen Methoden der Bewegungsanalyse, welche die Grundlage aller Bausteinsysteme bildeten, kann MTM auch die Eigenschaft der »Bezugsleistungstreue« für sich in Anspruch nehmen. Diese besagt, dass die ermittelten Zeiten für durchschnittlich geübte Werker gelten, was eine fundierte und international anerkannte Grundlage für die Anwendung des Systems schafft.

Die gemeinsame Grundlage aller Prozesssprachen zur Beschreibung menschlicher Arbeit „... beruht auf der Erkenntnis, dass sich mit einer begrenzten Anzahl von Bewegungselementen [...] jeder manuelle Arbeitsablauf beschreiben lässt.“ [18, p. 478] Dabei sollten diese Grundelemente so gestaltet sein, dass sie eine möglichst vollständige Beschreibung der Prozessbestandteile liefern. Eine verbreitete Methode zur Sicherstellung der Vollständigkeit einer Prozessanalyse ist die »W-Frage-Technik« [35, p. 41ff], welche beispielsweise im »6W-Fragen Arbeitsblatt« [38] Anwendung findet:

- Maßnahme (Warum?): spezifische Beschreibung der Operation
- Personal (Wer?): Arbeitsbezeichnung, Anzahl der Arbeiter, Namen der Arbeiter, etc.
- Maschine / Betriebsmittel (Was?): Bezeichnung der Maschinen und Betriebsmittel, Bezeichnung der Vorrichtungen, Anzahl der Einheiten etc.
- Standort (Wo?): spezifische Beschreibung des Operationsbereichs
- Zeit (Wann?): Durchlaufzeit, Ausbringungsmenge (aufgewendete Zeit pro Einheit)
- Methode (Wie?): spezifische Beschreibung der Operationsabfolge

Trotz der Tatsache, dass verschiedenste Prozesssprachen bei der Planung menschlicher Arbeit zum Einsatz kommen, ist deren Anwendbarkeit nicht für alle Tätigkeitsarten gewährleistet. Insbesondere geistige Arbeiten lassen sich mit ihnen nur sehr eingeschränkt abbilden. Die sogenannte »informativische Arbeit«, welche durch das Aufnehmen von Informationen und deren Nutzung als Entscheidungsgrundlage charakterisiert ist, lässt sich nur abbilden, sofern diese auf binäre „Ja-Nein“-Entscheidungen reduziert werden kann (z.B. Prüftätigkeiten). Hingegen stoßen Prozesssprachen bei der Beschreibung »reflektiver Arbeit«, also dem Erzeugen neuer Informationen, an ihre Grenzen. [32, p. 518f]

2.3 Die Digitale Fabrik

Mit der Erfindung des Transistors, dessen Einsatz in der Digitaltechnik und der daraus resultierenden Entwicklung von Computern entstand die Möglichkeit, nahezu alle Arten von Informationen digital zu erfassen. Die digitalisierten Daten konnten nun beliebig verarbeitet und vervielfältigt werden. Besonders effektiv wirkt diese Technologie immer dann, wenn die abzubildenden Aspekte der realen Welt so komplex sind, dass sie durch traditionelle Dokumentations- und Rechenmethoden nicht mehr umfassend zu bewältigen sind. Die industrielle Fertigung von Produkten ist ein solch komplexes Anwendungsgebiet. Bereits seit mehr als 20 Jahren wird dieser Komplexität durch die Entwicklung und Einführung der »Digitalen Fabrik« begegnet. Dabei handelt es sich um einen „... Oberbegriff für ein umfassendes Netzwerk von digitalen Modellen, Methoden und Werkzeugen – u. a. der Simulation und der dreidimensionalen Visualisierung –, die durch ein durchgängiges Datenmanagement integriert werden. Ihr Ziel ist die ganzheitliche Planung, Evaluierung und laufende Verbesserung aller wesentlichen Strukturen, Prozesse und Ressourcen der realen Fabrik in Verbindung mit dem Produkt.“ [39, p. 3]

Das Hauptaugenmerk der »Digitalen Fabrik« liegt zwar auf dem Bereich der Produktionsplanung. Nichtsdestotrotz beinhaltet das Anwendungsfeld auch die vorgelagerten Prozesse der Produktentwicklung, die nachgelagerten Prozesse der eigentlichen Produktion, des Vertriebs und des Service sowie deren gemeinsame Schnittstellen. Generell soll der gesamte Produktlebenszyklus nach dem Prinzip „Von der Idee bis zum Produktlebensende“ [39, p. 5] möglichst vollständig unterstützt werden. Neben dem bereits angesprochenem Aspekt der Bewältigung der steigenden Komplexität von Produkten und Prozessen hat die »Digitale Fabrik« auch ganz konkrete Ziele, die durch den Einsatz der Technologie erreicht werden sollen. Hierzu zählt hauptsächlich die Verbesserung der:

- Wirtschaftlichkeit (Kosten- und Zeitverbesserungen)
- Qualität (Harmonisierung und Optimierung der Produktionsplanung)
- Kommunikation (Integration und Beteiligung aller Anwendergruppen)
- Standardisierung (Wiederverwendbarkeit von Planungsergebnissen)
- Möglichkeiten zum Wissenserwerb & -erhalt (Wiederverwendbarkeit von Know-how)

[39, pp. 9-13]

Zur Realisierung dieser Ziele wurde im Laufe der Zeit ein großer Vorrat an Modellen entwickelt, der zahlreiche Elemente des Planungsprozess abbilden kann. Die zur Verfügung stehenden Modelle lassen sich in drei Hauptkategorien einteilen: (Abbildung 4)

- Geometriemodelle (z.B.: 3D-Produktdaten, Layout-Informationen, Werkzeuge, Maschinen, Vorrichtungen, Umwelt, ...)
- Methodenmodelle (z.B.: Stücklisten, Arbeitsunterweisungen, FEM, Falschfarbendarstellungen, Werbung, Bedienungsanleitungen, ...)
- Bewegungsmodelle (z.B.: Robotersimulation, Maschinenbewegungen, Montageabfolgen, Produktfunktionen, digitale Menschmodelle, ...)

[40, p. 11ff]

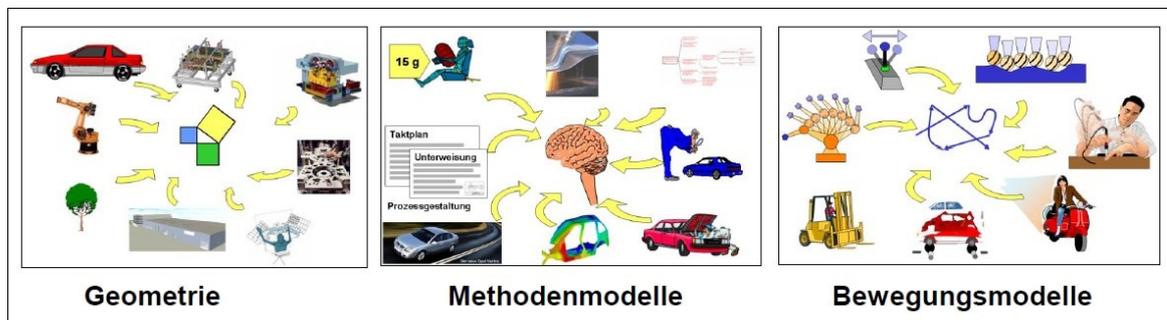


Abbildung 4: Modelle der »Digitalen Fabrik« [40, p. 11]

Um diese Modelle und deren Wechselwirkungen zu erfassen und deren Prozessintegration im Rahmen der Digitalen Fabrik zu ermöglichen, wurden auch zahlreiche Klassen von Software-Werkzeugen entwickelt. (Abbildung 5) So werden beispielsweise mit Hilfe von CAD-Systemen (Computer-aided Design) geometrische Informationen von Produkten und deren Produktionsumgebung erzeugt. Diese Daten können anschließend für diverse Untersuchungen herangezogen werden, um beispielsweise die erforderlichen Befehle für CNC-Maschinen daraus abzuleiten (CAM - Computer-aided Manufacturing) oder damit mechanische Festigkeitsberechnungen durchzuführen (FEM - Finite-Element-Methode). Als übergeordnete Software-System-Klasse fungieren die PLM-Systeme (Product Lifecycle Management). Dabei „... handelt es sich um einen Ansatz zur ganzheitlichen, unternehmensweiten Verwaltung und Steuerung aller Produktdaten und Prozesse des kompletten Lebenszyklus entlang der erweiterten Logistik-

kette – von der Konstruktion und Produktion über den Vertrieb bis hin zur Demontage und dem Recycling.“ [41]

Die eher geometriebezogenen Systeme werden durch Werkzeuge ergänzt, die sich vornehmlich der Verwaltung von alphanummerischen Informationen widmen. Hierbei sind vor allem die PPS-Systeme (Produktionsplanung und -steuerung) und die ERP-Systeme (Enterprise-Resource-Planning) zu nennen. Erstere gelten als „... das zentrale Informationssystem eines Produktionsunternehmens ...“ [42, p. 2], welche „... die gesamte technische Auftragsabwicklung von der Angebotsbearbeitung bis hin zum Versand“ umfassen. [43, p. 290] Demgegenüber handelt es sich bei ERP-Systemen um „...integrierte Software, die [...] alle oder wesentliche Teile der Geschäftsprozesse eines Unternehmens aus betriebswirtschaftlicher Sicht informationstechnisch unterstützt“. [44, p. 5]

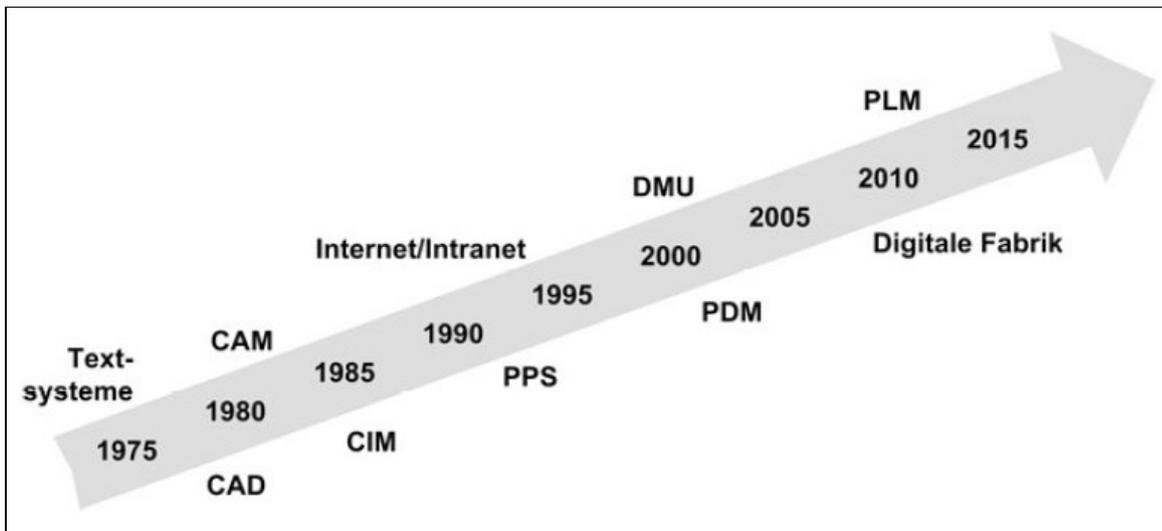


Abbildung 5: Relevante Software-Werkzeuge der Digitalen Fabrik [17, p. 3]

Die Zahl der Modelle und Methoden, die sich dem Themenkomplex »Digitale Fabrik« zuordnen lassen, ist groß. Auch die Entwicklung dieser Systeme erstreckt sich nunmehr über 35 Jahre. Dennoch hat sich insbesondere der Ansatz der durchgängigen Nutzung der Methoden über den gesamten Produktentstehungsprozess hinweg noch nicht durchgesetzt. Die Automobilindustrie nimmt zwar bei der Einführung der Digitalen Fabrik eine Vorreiterrolle ein, aber auch hier bestehen noch immer System- und Datenbarrieren zwischen den Bereichen der Produktentwicklung, Produktionsplanung und Produktion. Die

2 Einordnung und Forschungsumgebung

Ursachen dafür bestehen in zwei Kernproblemen, die in Zukunft gelöst werden müssen. Dies sind zum einen ein durchgängiges Datenmanagement, also die Nutzbarkeit der in der Entwicklung generierten Daten über den vollständigen Planungs- und Lebenszyklus des Produkts hinweg und zum anderen die Einführung von integrierten und standardisierten Prozessen für die Nutzung der Digitalen Fabrik. [17, p. 337ff] Gerade beim zweiten Punkt wird deutlich, dass die Herausforderungen nicht nur in der Bereitstellung geeigneter Software-Systeme und technischer Infrastrukturen liegen. Auch der konsequente und effiziente Einsatz in den passenden Geschäftsprozessen und Unternehmensunterorganisationen entscheidet letztlich darüber, ob die Technologie ihre gewünschte Wirkung entfalten kann.

2.4 Digitale Menschmodelle

Trotz der Tatsache, dass die Methoden und Werkzeuge der Digitalen Fabrik immer weiter reifen und miteinander vernetzt werden, gibt es immer noch Bereiche, in denen das vorhandene Potential bei weitem noch nicht ausgeschöpft wird. Nach wie vor zählen zumeist tabellen- und formularbasierte Methoden zum Standard der Planung menschlicher Tätigkeiten in der Digitalen Fabrik. Doch mit der Verfügbarkeit dreidimensionaler Produktdaten und Produktionsressourcen wurde die Tür zu etwas geöffnet, was im Bereich der Robotik schon heute Stand der Technik ist. Gemeint sind dabei leistungsfähige Simulationssysteme, welche statische und dynamische Untersuchungen anhand virtueller Abbilder des Menschen ermöglichen.

Die Entwicklung digitaler Menschmodelle begann in den sechziger Jahren zunächst in den USA und wenige Jahre später auch in Europa. Während in den Vereinigten Staaten die Luft- und Raumfahrtforschung als Triebfeder für die Entwicklung betrachtet werden können, war in Europa zunächst der Bereich der Produktentwicklung das erste Anwendungsgebiet. [45, pp. 3-5 - 3-12] Erste Untersuchungen mündeten 1967 in die Entwicklung des Modells »SAMMIE«, welches als erstes kommerziell verfügbares digitales Menschmodell betrachtet wird und sowohl in der Automobilindustrie als auch bei der (statischen) Planung von Industriearbeitsplätzen Verwendung fand. [46, p. 10/42] Im Laufe der Zeit entstanden zahlreiche weitere Modelle. Zum Teil bauten diese aufeinander auf, zum Teil hatten sie aber auch vollkommen konträre Ansätze. Eine aktuelle Übersicht aller historisch relevanten Menschmodelle und deren Funktionalitäten ist als »Katalog digitaler Menschmodelle« in Anlage A von Mühlstedt [7] zu finden.

In Abhängigkeit ihrer jeweiligen Anwendungsgebiete lässt sich die Vielzahl der existierenden Modelle in fünf Hauptkategorien mit folgenden Eigenschaften einteilen:

- Anthropometrische Modelle
 - möglichst exakte Nachbildung realer menschlicher Körpergrößen und -formen
 - Einsatz in Luftfahrt- und Automobilindustrie zur Gestaltung „mensch-zentrierter“ Räume (z.B. Fahrzeug-Cockpit, Sitzreihen in Flugzeugen, etc.)

2 Einordnung und Forschungsumgebung

- Modelle zur Produktionsgestaltung
 - (zum Teil) weniger exakte Anthropometrie und Gestalt im Vergleich zu anthropometrischen Modellen
 - Bewegungssimulation mit dem Fokus auf Arbeitsablaufplanung, Sollzeitbestimmung und Ergonomie
- Biomechanische Modelle
 - Modellierung anhand realistischer Massen, Momente, Feder- und Dämpfungselemente
 - dynamische Simulationen unter Verwendung der physikalischen Bewegungsgleichungen
 - Verwendung für Crash-Simulationen, Ergonomie-Betrachtungen und individuelle Leistungsuntersuchungen
- Anatomische Modelle
 - Berücksichtigung medizinischer Informationen (z.B. innerer Organe, Kreislaufsysteme, etc.)
 - Planung und Simulation chirurgischer Eingriffe, Einsatz zu Ausbildungszwecken
- Kognitive Modelle
 - Abbildung des Informationsflusses innerhalb und zwischen menschlichen Akteuren (Ursache-Wirkungs-Prinzipien und Wahrscheinlichkeitsmodelle)
 - Anwendungsgebiete sind z.B. Künstliche Intelligenz, Gestaltung von Mensch-Maschine-Schnittstellen, etc.

[45, pp. 3-13 - 3-25]

Beim konkreten Einsatz im Bereich der Digitalen Fabrik sind insbesondere die ersten drei genannten Modell-Kategorien relevant, um die Aspekte der Produkt- und Prozessergonomie frühzeitig überprüfen zu können. [6, p. 2]

Trotz der verschiedenen Einsatzfelder der digitalen Menschmodelle verfügen diese über grundlegende gemeinsame Eigenschaften. Jedes digitale Menschmodell hat beispielsweise eine Skelettstruktur. Diese besteht aus eindeutig bezeichneten, gekoppelten Segmenten, welche sich an der Anthropometrie des realen Menschen orientieren. Zur Visualisierung wird über das Skelett im Allgemeinen ein geometrisches Flächenmodell gelegt, welches mit der darunterliegenden Knochenstruktur verbunden ist und die Hautoberfläche repräsentieren

soll. Um die Menschmodelle für belastbare Auswertungen einsetzen zu können, müssen Skelett und Hautoberfläche an die Abmessungen realer Menschen angepasst werden. In verschiedenen Untersuchungsreihen wurden u.a. zu diesem Zweck Probanden vermessen und aus den dabei gewonnenen statistischen Informationen sogenannte »Perzentile« aggregiert. Diese Perzentile bestehen jeweils aus einem Satz von Körperabmaßen, die auf Menschmodelle übertragen werden können. [6, p. 10]

Um Bewegungen des menschlichen Körpers abbilden zu können, sind die Segmente der Skelettstruktur als sogenannte »kinematische Ketten« modelliert. Dabei sind die einzelnen Elemente des Skeletts durch Gelenke verbunden, deren mathematische Beschreibung Bewegungen gemäß den realen Freiheitsgraden menschlicher Gelenke ermöglichen sollen. Zu den grundlegenden Bewegungen, die auch für spätere Auswertungen relevant sind, gehören Bewegungen:

- ... des Hand-Arm-Systems
- ... des Fuß-Bein-systems
- ... der Wirbelsäule
- ... des Kopfes

Über entsprechende Schnittstellen wird der Anwender in die Lage versetzt, Gelenkstellungen einzugeben und damit Körperhaltungen zu erzeugen. Dennoch obliegt es derzeit dem Anwender, für die Realitätsnähe der eingegeben Haltung zu sorgen. [6, p. 15]

Der zuletzt genannte Fakt weist bereits auf die Grenzen des derzeitigen Standes der Technik bei digitalen Menschmodellen hin. Im Rahmen seiner Studien zu digitalen Menschmodellen und den dabei durchgeführten Anwenderbefragungen ermittelte Mühlstedt ebenfalls die Funktionen zur Erstellung von Animationen als diejenigen mit dem dringendsten Verbesserungswunsch. [7, p. 93] Darüber hinaus verwiesen die Anwender auf den aus ihrer Sicht zu hohen Zeitaufwand beim Einsatz und verbesserungswürdige Analyseverfahren, welche derzeit noch die Akzeptanz der Systeme einschränken. [7, p. 94]

3 Methodische Vorarbeit

3.1 Analyse existierender Menschmodelle

Als Einstieg in die Thematik und Grundlage für die nachfolgende Anforderungsanalyse untersuchte der Autor relevante digitale Menschmodelle und ermittelte deren Defizite aus Anwendersicht. Der hierzu notwendige Wissensaufbau erfolgte durch Diskussionen mit Anwendern, offizielle Schulungen (Human V5), im Selbststudium mit entsprechenden Testversionen und durch Literaturrecherchen. Es stellte sich schon sehr frühzeitig heraus, dass es im Bereich der Simulation menschlicher Tätigkeiten im Umfeld der Digitalen Fabrik zwei relevante Systeme gibt - »Jack« (Siemens) und »Human« (Dassault Systèmes). Bei beiden Systemen handelt es sich um kommerzielle Produkte, welche in die PLM-Systeme der jeweiligen Hersteller integriert sind. (Abbildung 6)

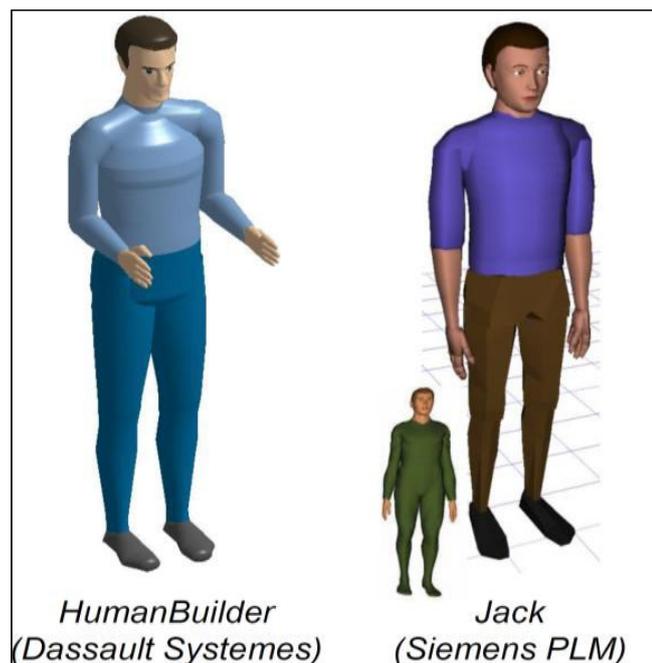


Abbildung 6: »Human« und »Jack« [7, pp. 30, Abb. 2.6]

Der französische Softwarehersteller »Dassault Systèmes« verdankt seinen hohen Bekanntheitsgrad insbesondere seinem CAD-System »CATIA« und seinem PLM-System »DELMIA«. Beide Systeme werden sowohl bei Automobil-OEMs als auch im Bereich der Luft- und Raumfahrtindustrie eingesetzt. Mit dem Menschmodell »Human«, welches als Zusatzmodul für beide Systeme angeboten

wird, brachte Dassault ab Version 5 (V5) seiner Produktreihe ein Werkzeug auf den Markt, welches für ergonomische Untersuchungen und zur Erstellung von Simulationen genutzt werden kann. Dabei liegt der Fokus bei der Integration in »CATIA« eher im Bereich der Produktergonomie, während die Integration in »DELMIA« den Schwerpunkt eher auf Prozessergonomie und die dafür notwendige Simulation von Tätigkeiten setzt. Der »Human« besteht aus 99 Körperteilen, welche über insgesamt 144 Freiheitsgrade vom Benutzer angesteuert werden können. Während die geometrische Hülle in Version 5 noch aus einzelnen unverbundenen Zylinder-Segmenten bestand, verfügt der »Human« ab Version 6 über ein geschlossenes Dreiecksnetz als virtuelle Hautoberfläche.

Das Programmpaket gliedert sich in einzelne Untermodule (sog. »Workbenches«), welche thematisch verwandte Funktionen zusammenfassen. So gibt es ein Modul zur parametrischen Erzeugung von Standard-Menschmodellen, innerhalb dessen unter Angabe von Geschlecht, Perzentil und Ethnie entsprechende Modelle erzeugt werden können (»Human Builder«). Diese Standard-Menschmodelle können danach bei Bedarf in ihren Abmaßen und Umfängen angepasst werden (»Human Measurement Editor«). Mit Hilfe des Moduls »Human Posture Editor« kann das Menschmodell anschließend in seiner Körperhaltung verändert werden. Hierzu stehen neben den klassischen Werkzeugen der inversen und Vorwärts-Kinematik auch etwas komplexere parametrische Mittel zur Verfügung, um beispielsweise sitzende oder kniende Haltungen einstellen zu können. Das Untermodul »Human Task Simulation« sorgt für die Integration in die Simulations-Infrastruktur von DELMIA. Dabei müssen vom Benutzer einzelne Arbeitsaufgaben des Menschmodells modelliert werden (sog. »Human Tasks«), welche anschließend durch die Verknüpfung mit einer übergeordneten Prozessstruktur in eine zeitliche Reihenfolge gebracht werden. Erwähnenswert hierbei ist die grundlegende Herangehensweise der Modellierung der Arbeitsaufgaben durch die Erzeugung und Aneinanderreihung einzelner Körperhaltungen. Einzelne Komplexfunktionen, um beispielsweise das Laufen auf einem Pfad oder das Nachverfolgen eines Objektes effizienter modellieren zu können, werden mitgeliefert. Für die Mehrheit der zu simulierenden Tätigkeiten bleibt jedoch lediglich die Einzelposen-Modellierung als adäquates Mittel.

Das digitale Menschmodell »Jack« wurde ursprünglich an der Universität von Pennsylvania entwickelt. Mitte der 90er Jahre wurde es zunächst von »UGS« übernommen, bevor es schließlich zu einem Modul von »Siemens PLM« reifte.

Neben der integrierten Version wird »Jack« im Gegensatz zum »Human« auch als eigenständige Anwendung vertrieben.

Das Menschmodell verfügt über 68 Körperteile mit insgesamt 135 Freiheitsgraden und seine Hülle wird ebenfalls durch ein geschlossenes Dreiecksnetz visualisiert. Im Grundfunktionsumfang steht »Jack« seinem „Verwandten“ von Dassault Systèmes in nichts nach. So kann das Menschmodell ebenfalls anhand anthropometrischer Maße und Parameter an den Anwendungsfall angepasst werden. Die Erstellung von statischen Körperhaltungen ist mit Hilfe vordefinierter Hand- und Körperstellungen ebenso möglich wie unter Zuhilfenahme von inverser und Vorwärts-Kinematik. Auch bei der Animation des Menschmodells wird der bekannte Weg der Einzelposen-Modellierung mit nachfolgender Interpolation selbiger beschritten. In einer Art Gant-Diagramm stehen hierfür mehrere Simulationsdaten-Spuren zur Verfügung, welche mit entsprechenden Key-Frames für einzelne Körperteile bestückt werden können. Eine Makro-Funktionalität zur Erstellung von Laufbewegungen steht ebenfalls zur Verfügung. Aus Sicht der vorliegenden Arbeit war neben den Standardfunktionen vor allem das Erweiterungsmodul »Task Simulation Builder« (kurz TSB) interessant. Aus einer Liste von grundlegenden Aktionen, den sogenannten »Tasks«, kann darin eine sequentielle Bewegungsfolge erstellt werden. Die zur Verfügung stehenden Aktionen reichen vom Gehen bis hin zum Aufnehmen, Platzieren und Betätigen. Darüber hinaus lassen sich die zur Verfügung stehenden Elemente auch per Interpolation mit benutzerdefinierten Körperhaltungen kombinieren.

Insgesamt bietet »Jack« bei der Simulations-Erstellung bereits einige fortschrittlichere Ansätze als das Menschmodell von Dassault Systemès. Dennoch konzentrieren sich beide Systeme aus Sicht des Autors zu sehr auf die Modellierung von Bewegungen und lassen den Planungsaspekt außen vor. Es ist durch die Verwendung der angebotenen Werkzeuge durchaus möglich auch komplexere technologische Abläufe abzubilden. Dabei stößt der Anwender allerdings sehr schnell an die Grenzen der angebotenen „Makro-Funktionalitäten“ (→ Mangel an Universalität der Funktionalität), wodurch schließlich doch wieder die Einstellung einzelner Körperhaltungen erforderlich wird. So sind beispielsweise keine Hilfsfunktionen vorhanden, um den Einsatz von Werkzeugen oder die Arbeit in „Nicht-Standard“-Körperhaltungen, wie zum Beispiel mit Körperhilfen („abstützen“), im Liegen oder unter geometrischen Zwangsbedingungen

auf „einfache“ Art und Weise zu simulieren. Sobald jedoch auf die Details einzelner Körperhaltungen und deren zeitliche Verläufe Einfluss genommen werden muss (→ Mangel an Zielorientiertheit der Anwendung), verliert die Menschmodell-Simulation als Planungs-Ergebnis an Objektivität und je nach Erfahrungsgrad des Anwenders ggf. auch an Qualität. Selbst wenn es sich um den gleichen Simulations-Ersteller und die gleichen Rahmenbedingungen (Szenario, Software) handelt, ist es nahezu unmöglich zwei identische Simulationen zu erzeugen (→ Mangel an Reproduzierbarkeit der Ergebnisse).

Aus Sicht des 3D-Datenaustauschs ist es zwar gut, dass sowohl »Human« als auch »Jack« in die jeweilige PLM-Lösung des Herstellers eingebunden sind. Andererseits sind die sehr stark der CAD-Welt entlehnten Eingabe-Metaphern für den normalen Planungsingenieur ein fremdes Betätigungsfeld (→ Mangel an Effizienz der Modellierung). Die Methoden der Digitalen Fabrik haben sich insbesondere im Bereich der Planung menschlicher Arbeit bislang noch nicht soweit durchsetzen können, wie es beispielsweise im Bereich der Roboter-Offline-Programmierung der Fall ist. Somit muss ein neues System auch dazu in der Lage sein, eine Brücke zwischen den traditionellen Planungsmethoden und den Möglichkeiten der rechnergestützten Planungswelt zu bilden. Der mangelnden Erfahrung der potentiellen Endanwender im CAD-Bereich sollte mit einem eingängigen Bedienkonzept und der Verwendung bekannter Begrifflichkeiten begegnet werden, um ein effizientes Arbeiten zu ermöglichen.

3.2 Anforderungsanalyse

Aus der Aufgabenstellung (Kapitel 1.3) und der Analyse existierender digitaler Menschmodelle (Kapitel 3.1) galt es zunächst konkrete Anforderungen für das zu entwickelnde Simulationssystem abzuleiten. Erste Ideen hierzu konnten wiederum beim Literaturstudium gewonnen werden. Sowohl Fritzsche [8, p. 112ff] als auch Mühlstedt [7, p. 62ff] beschäftigten sich in ihren Arbeiten u.a. mit den Grundlagen und Anwendungsgebieten digitaler Menschmodelle aus arbeitswissenschaftlicher Sicht. Einen umfassenden Überblick aller relevanten Anwendungen vermittelte zudem Duffy [45] in seinem »Handbook of Digital Human Modeling«. Der Autor nutzte außerdem die Möglichkeit zur direkten Diskussion mit den erfahrenen Planungsingenieuren der imk automotive GmbH in Chemnitz. Dabei konnte er mehrere Planungsprojekte begleiten, deren Fokus auf der Planung manueller Tätigkeiten lag. Bei diesen unterstützenden Tätigkeiten bestand auch die Gelegenheit zur Diskussion mit Kunden und potentiellen Anwendern des zu schaffenden Systems. Die Möglichkeit zur Beobachtung realer Automobilfertigungslinien und Montagearbeitsplätze bildete dabei eine Brücke zwischen den theoretischen Überlegungen im Vorfeld und der angewandten Praxis.

Als Startpunkt und zur Abgrenzung des Diskursbereichs formulierte der Autor zunächst einige grundsätzliche Fragen zum angestrebten System und dessen Eigenschaften. Anhand der gewonnenen Erkenntnisse konnten diese wie folgt beantwortet werden:

Wer wird das System wofür nutzen?

Das angestrebte Simulationssystem soll Planungsingenieure bei ihrer täglichen Arbeit und dabei insbesondere bei der Planung und Optimierung manueller Tätigkeiten unterstützen.

In welchem Umfeld soll das System zum Einsatz kommen?

Grundsätzlich soll das System in allen Industriebereichen angewendet werden können, in denen manuelle Tätigkeiten auftreten. Um für einen ersten Schritt eine Abgrenzung vornehmen zu können, fokussieren die nachfolgenden Untersuchungen zunächst auf den Bereich des Automobilbaus.

Welchen Vorteil soll das System bringen?

Die simulative Planung menschlicher Arbeit soll einen umfassenden Beitrag zur Planungsabsicherung beisteuern, die Planungsqualität erhöhen und dadurch zur Schaffung ergonomischer Arbeitsplätze sowie zur Effizienzsteigerung und Kostenreduktion beitragen. Darüber hinaus soll das System etablierte Planungsmethoden berücksichtigen und diese um den Aspekt der Visualisierung ergänzen. Die Visualisierung soll zur Anleitung von Arbeitskräften und zur Prozessdokumentation einsetzbar sein. Darüber hinaus soll das neue System Lösungsansätze für die festgestellten Mängel existierender Systeme (siehe Kapitel 3.1) aufzeigen.

Welche etablierten Planungsmethoden müssen berücksichtigt werden?

Das System soll möglichst nah an der Sprach- und Methodenwelt des Planungsingenieurs orientiert sein. Hierzu zählt die Strukturierung und prozesssprachliche Beschreibung von Abläufen ebenso wie die genutzten Systeme zur Sollzeitbestimmung (z.B. »MTM«) und Methoden zur ergonomischen Bewertung (z.B. »EAWS«).

Welche Fremdsysteme müssen berücksichtigt werden?

Die neuartige Methodik des zu entwickelnden Systems soll sich in nahezu jedes System integrieren lassen, welches ebenfalls bereits über ein digitales Menschmodell verfügt. Deswegen müssen entsprechende Schnittstellen zum Datenaustausch vorgesehen werden.

Welche Ergebnisse soll das System liefern und wie werden diese weiterverwendet?

Folgende Ergebnisse sollen durch die Arbeit mit dem zu entwickelnden Simulationssystem geliefert werden:

- Plausibilitätscheck → Unterstützung bei der Identifizierung von Planungsfehlern
- Zeitermittlung → zunächst Richtzeitermittlung, später belastbare Normzeitermittlung zur Unterstützung der Austaktung von Fertigungsbereichen
- Ermittlung ergonomisch relevanter Parameter → Durchführung der Ergonomie-Analyse

- Visualisierung → Erstellung von Dokumentationsmaterial zu Schulungs- und Präsentationszwecken

Was sind die Randbedingungen der Entwicklung? Gibt es Vorgaben für die Genauigkeit bzw. Detailliertheit der Simulation?

Da das Hauptaugenmerk auf der Simulation menschlicher Tätigkeiten liegt, ist es das Ziel, die Bewegungen realer Menschen möglichst detailliert und realistisch abzubilden. Hierzu zählt insbesondere:

- äußeres Erscheinungsbild (Habitus) → Die geometrische Hülle des verwendeten digitalen Menschmodells sollte durch ein geschlossenes Dreiecksnetz abgebildet werden. Aufgrund der Tatsache, dass es sich bei den Ergebnissen der Simulation um technische Visualisierungen handelt, besteht keine Notwendigkeit eine besonders detaillierte geometrische Hülle zu verwenden.
- innerer Aufbau (Anatomie) → Die verwendete innere „Knochen“-Datenstruktur (kinematische Struktur) muss so detailliert sein, dass das digitale Menschmodell alle erforderlichen Körperhaltungen einnehmen und gleichzeitig die Einnahme humanmotorisch unmöglicher Körperhaltungen verhindern kann.
- Körpermaße (Anthropometrie) → Bei der Dimensionierung der verwendeten digitalen Menschmodelle sollen beliebige statistisch ermittelte Werte zugrunde gelegt werden können (anthropometrische Perzentile).
- geometrische Randbedingungen → Die Bewegungen des Menschmodells sollen dem Gesetz der Schwerkraft folgen und in ihrer Geschwindigkeit der durchschnittlich erbringbaren Leistungsfähigkeit eines Menschen entsprechen.

Nachdem diese ersten groben Randbedingungen durch den Autor ermittelt worden waren, galt es daraus die funktionalen und nicht-funktionalen Anforderungen des Systems abzuleiten. Hierbei sollte auch noch einmal der Vergleich mit existierenden Systemen zur Simulation digitaler Menschmodelle Berücksichtigung finden. Entgegen der bislang verbreiteten Vorgehensweise des „per Hand Animierens“ digitaler Menschmodelle soll ein System geschaffen werden, welches dem Anwender eine effizientere Simulationserstellung ermöglicht. Die bisherige Vorgehensweise der Einzel-Körperhaltungs-Erzeugung bei

Menschmodellen wird allgemein als „Teach-in“ (engl. lehren, beibringen) bezeichnet. Hinderlich ist dabei jedoch die Tatsache, dass für jede zu erzeugende Simulation und auch bei jeder nachträglichen Änderung ein aufwändiges Anlernen erforderlich ist. Vorhandene digitale Menschmodelle verfügen im Gegensatz zu ihren realen Vorbildern über keinerlei Möglichkeiten, eine bereits „beigebrachte“ Tätigkeitsfolge an veränderte Umgebungsparameter anzupassen. Darüber hinaus ist auch die Ebene, auf der dem digitalen Menschmodell etwas beigebracht werden kann, in den vorhandenen Systemen (siehe Kapitel 3.1) so elementar, dass es für den Anwender schwierig bis unmöglich ist, korrekte und reproduzierbare Ergebnisse zu erzielen. Aus diesem Grund soll die Ansteuerung der digitalen Menschmodelle durch eine Prozesssprache erfolgen, welche sich an etablierten Methoden zur Planung menschlicher Arbeit orientiert.

Von den vorangegangenen Überlegungen und Umgebungsbedingungen ausgehend konnten die funktionalen Kernanforderungen des geplanten Systems identifiziert werden. Benötigt werden Funktionen zur:

- Definition bzw. Erzeugung digitaler Menschmodelle
- Definition der Arbeitsumgebung der digitalen Menschmodelle
- Definition der Tätigkeitsbeschreibung der digitalen Menschmodelle
- „Übersetzung“ der Tätigkeitsbeschreibung in Bewegungen
- Visualisierung der erzeugten Bewegungen
- Bewertung und Dokumentation der erzeugten Bewegungen

Gleichzeitig muss insbesondere folgenden nichtfunktionalen Anforderungen Rechnung getragen werden:

- **Benutzbarkeit:** Die angebotenen Funktionen und die verwendeten Begrifflichkeiten müssen mit der Denkwelt des Planungsingenieurs korrespondieren, um eine schnelle Erlernbarkeit und somit hohe Akzeptanz des Systems zu erreichen.
- **Effizienz:** Die Erstellung der Arbeitsumgebung und der Tätigkeitsbeschreibung sollte auf eine möglichst einfache und intuitive Art und Weise möglich sein, um eine hohe Produktivität beim Umgang mit dem System sicherzustellen.
- **Flexibilität:** Das System sollte vorhandene Standards der Planungswelt verwenden und gleichzeitig offen für die spätere Ergänzung / Anbindung zukünftiger Standards und Methoden sein.

- Portabilität: Die zu entwickelnde Methodik soll sich auf beliebige digitale Menschmodelle anwenden lassen; hierzu sind entsprechende Schnittstellen vorzusehen.
- Skalierbarkeit: Das System sollte prinzipiell in der Lage sein, eine beliebige Anzahl von digitalen Menschmodellen gleichzeitig anzusteuern.
- Korrektheit: Die Erzeugung plausibler und humanmotorisch korrekter Bewegungsabläufe für digitale Menschmodelle ist die Grundbedingung für alle nachfolgenden Aus- und Bewertungsfunktionalitäten des Systems.

Die enorme Komplexität menschlichen Handelns, welche durch das zu schaffende System abgebildet werden soll, veranlasste den Autor dazu, einige Einschränkungen zum Diskursbereich festzulegen:

- Zunächst soll das System nur zur Simulation von manuellen Tätigkeiten in der industriellen Produktion dienen. Dabei liegt der Fokus primär auf dem Bereich Automobilbau (Montage und Rohbau). Dieser Punkt schränkt sowohl die Art der zu simulierenden Tätigkeiten als auch die Gestalt der zugrundeliegenden Bewegungen ein.
- Es ist zunächst keine aufgabenbezogene Entscheidungsautonomie der digitalen Menschmodelle erwünscht, d.h. es gibt immer eine konkrete vom Anwender definierte Tätigkeitsbeschreibung (Aufgabe) und Abarbeitungsfolge, welche umgesetzt oder im Fehlerfall abgelehnt werden soll.
- Im ersten Entwicklungsschritt werden nur „Ein-Personen-Tätigkeiten“ unterstützt. Die Simulation von Aufgaben, welche die parallele räumliche und zeitliche Koordination mehrerer Menschmodelle erfordern (z.B. „gemeinsames Tragen eines großen Werkstücks“), ist nicht Bestandteil dieser Arbeit.
- Alle Objekte der Simulationsumgebung, mit denen das Menschmodell direkt interagiert, werden als statisch angenommen. D.h. Interaktions-Objekte werden nur durch das Menschmodell bewegt; sie bewegen sich nicht eigeninitiiert.

3.3 Identifikation der notwendigen Systemkomponenten

Basierend auf der vorangegangenen Anforderungsanalyse ließen sich nun die notwendigen Komponenten des angestrebten Software-Systems ableiten. Dabei gruppierte der Autor die ermittelten funktionalen (Kern-)Anforderungen und bildete aus ihnen eigenständige Systemkomponenten. Durch deren Unterteilung und Anordnung gemäß der klassischen softwaretechnischen Einstufung in Eingabe-, Verarbeitungs- und Ausgabe-Komponenten konnte darüber hinaus erstmalig der geplante Datenfluss des Systems dargestellt werden (siehe Abbildung 7).

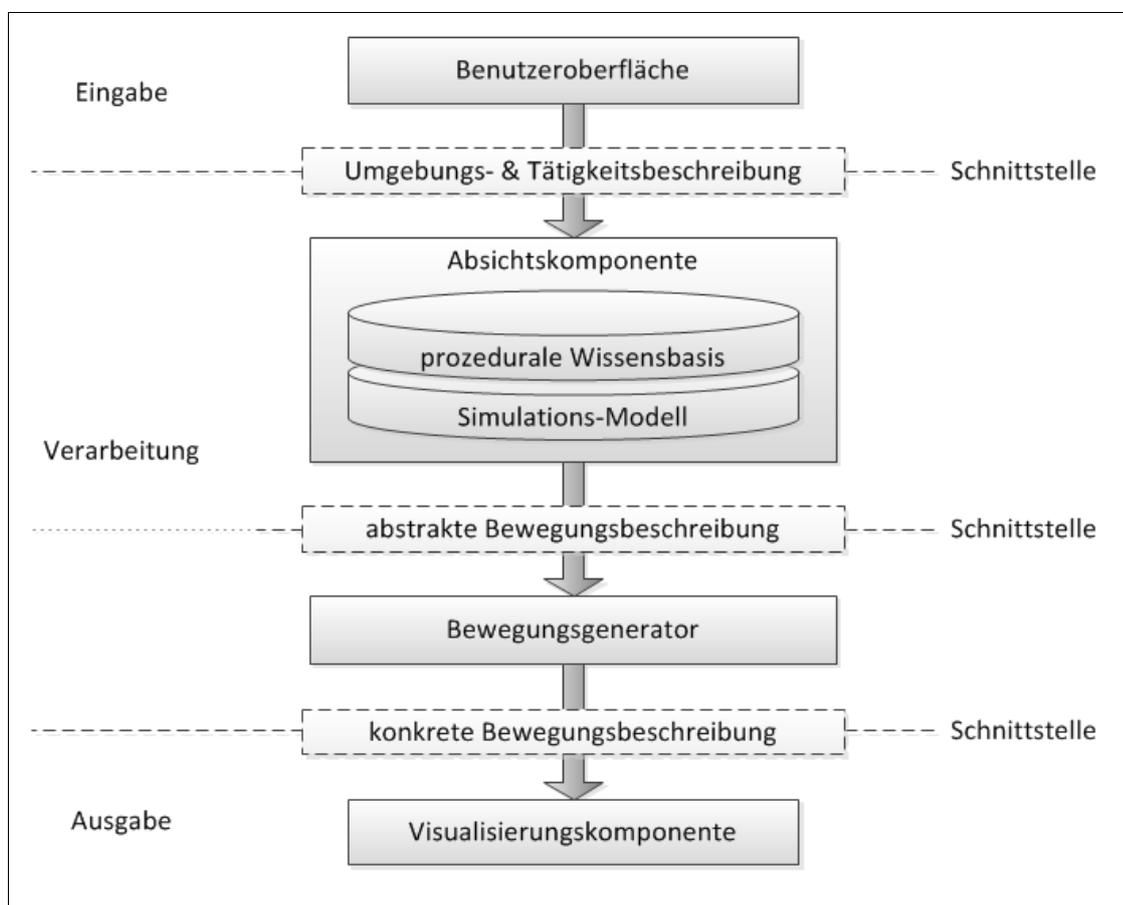


Abbildung 7: Komponentenschema des Simulations-Systems [47, p. 269]

Zunächst müssen dem System alle für die Simulation erforderlichen Informationen bekannt gegeben werden. Hierfür ist eine entsprechende Benutzeroberfläche erforderlich, welche die Verbindung zwischen den alpha-nummerischen

Eingaben und den der virtuellen 3D-Welt entlehnten Informationen schaffen muss. Zu deren Hauptaufgaben gehört die Konfiguration der Menschmodell-Instanzen und deren Arbeitsumfeld, inklusive aller für die Simulation relevanten Produkt- und Ressourceninformationen. Diese Eingaben werden danach durch die Definition des durchzuführenden Arbeitsablaufs ergänzt. Die Gesamtheit dieser Informationen wird über die Schnittstelle mit der Bezeichnung Umgebungs- und Tätigkeitsbeschreibung an den Verarbeitungskern des Simulationssystems weitergereicht.

In Bezug auf die Verarbeitung der Eingabedaten und deren Überführung in Bewegungen für digitale Menschmodelle entschied sich der Autor für einen zweistufigen Prozess. Beweggrund hierfür war primär die angestrebte Portierbarkeit der zu entwickelnden Methodik sowie die Anforderung, verschiedene anthropometrische Perzentile zu unterstützen (siehe Kapitel 3.2). Beide Ziele lassen sich durch eine Abstraktion der Verarbeitungslogik von der konkreten Struktur des zum Einsatz kommenden Menschmodells und der Bewegungsgenerierungsmethode realisieren. Zunächst sollen deshalb die Eingangsdaten durch eine sogenannte Absichtskomponente in eine Menschmodell-unabhängige Zwischenrepräsentation überführt werden. Die Absichtskomponente dient dabei der Auswertung der gegebenen Informationen aus Umgebungs- und Tätigkeitsbeschreibung und der Ermittlung der Bewegungsabsicht aus der Aufgabenstellung. Intern gliedert sich die Absichtskomponente in zwei Teilkomponenten – eine prozedurale Wissensbasis und das Simulationsmodell, welche eng miteinander verzahnt sind. Die prozedurale Wissensbasis soll dabei die Auflösung der zu simulierenden Tätigkeiten von der groben Ebene der Benutzereingaben bis auf eine feinere, durch einen nachfolgenden Bewegungsgenerator erfassbare Ebene vornehmen. Während der Verarbeitungsprozesse dient das Simulationsmodell als zentraler Anlaufpunkt für alle Arten von Statusabfragen und Berechnungen, welche das Menschmodell selbst oder seine Interaktion mit der Umgebung betreffen.

Als Ergebnis dieses Verarbeitungsschrittes soll eine sogenannte abstrakte Bewegungsbeschreibung entstehen. Den Zusatz „abstrakt“ trägt diese Schnittstelle deshalb, weil die Datenrepräsentation lediglich Zwangsbedingungen bzw. Bewegungsziele (»Constraints«) und diverse Randbedingungen für die Bewegungsausführung beinhaltet und somit weitestgehend unabhängig von der Struktur und Implementierung des verwendeten Menschmodells ist. Diese

„abstrakten“ Daten kann ein nachfolgender Bewegungsgenerator nun auf ein konkretes Menschmodell anwenden. Das Ergebnis in Form der konkreten Bewegungsbeschreibung enthält danach alle relevanten Gelenkwinkel und Positions-Informationen des digitalen Menschmodells aggregiert in sogenannten Schlüssel-Zuständen (»Key-Postures«).

Abschließend werden die generierten Daten an eine Visualisierungskomponente übergeben. Diese dient zum einen zur Ausgabe (Rendering) aller in der Simulation enthaltenen statischen 3D-Geometrien und zum anderen zur Anwendung der generierten Bewegungsbeschreibungen auf die dynamischen 3D-Objekte. Dabei müssen die erzeugten Schlüssel-Zustände durch die Generierung weiterer Zwischenzustände in fließende Bewegungen überführt werden, um eine realistische Bewegungs-Visualisierung zu ermöglichen.

In den nachfolgenden Kapiteln werden die Inhalte und Funktionsweisen der genannten Systemkomponenten weiter analysiert und detailliert.

3.4 Eingangsinformationen für das System

3.4.1 Umgebungsinformationen

Wie sein reales Vorbild muss auch das digitale Menschmodell zur Ausführung von Bewegungen und Tätigkeiten über grundlegende Informationen seiner (virtuellen) Umwelt verfügen. Je nach Einsatzfall und Stand der Planung können diese Informationen in mehr oder weniger detaillierter Form vorliegen. In der Praxis ist die Verfügbarkeit von 3D-Produktdaten und Layout-Informationen zumeist gewährleistet. Die Bereitstellung von Ressourcendaten, wie z.B. Werkzeugen, Regalen oder anderen Einrichtungsgegenständen gestaltet sich nach den Erfahrungen des Autors jedoch häufig problematisch.

Das absolute Minimum an Umgebungsinformation, für eine plausible Simulation, stellt die Definition einer Fußbodenhöhe oder besser noch die Vorgabe eines als Fußboden zu betrachtenden 3D-Objektes dar. In Kombination mit den sogenannten »Arbeitspositionen« auf dem Fußboden sind bereits Simulationen von Laufbewegungen, z.B. für einfache Materialflusssimulationen, realisierbar. Die Definition von »Arbeitspositionen« folgt dem »Arbeitssystem-Modell« [33] der Fertigungsprozessplanung, welches davon ausgeht, dass Tätigkeiten aufgrund der Abhängigkeit vom Vorhandensein entsprechender Ressourcen ortsgebunden (bezogen auf ein Bezugssystem) ausgeführt werden. Bei der Simulation menschlicher Arbeit sollte die Definition einer »Arbeitsposition« neben der geometrischen Position auch immer eine Orientierungsangabe beinhalten, welche die Ausrichtung des Menschmodells am Zielort definiert.

Zur Realisierung einfacher Interaktionen mit geometrischen Objekten nach dem Prinzip „Nimm ein Objekt und platziere es an einem Zielort“ sind neben den zur Ausführung der Tätigkeit einzunehmenden »Arbeitspositionen« auch Informationen dazu erforderlich, um welche konkreten Objekte es sich handelt und wo sich diese am Anfang und am Ende der Tätigkeit befinden sollen. Mit Hilfe der lokalen Objektkoordinatensysteme und der hierarchischen Beziehungen der Objekte im Szenengraph können die relevanten Positionen und Ausrichtungen bestimmt werden.

Da die Repräsentation der Geometrie von 3D-Objekten in Form von Dreiecksnetzen im Allgemeinen sehr komplex sein kann, hat sich als Vereinfachung die Verwendung von sogenannten »Bounding Volumes« bewährt. Hierbei handelt es sich um Hüllobjekte in Form von Kugeln, Quadern oder anderen Primitiv-

Geometrien, welche eine grobe Abstraktion von den tatsächlich zugrundeliegenden Objekten erlauben und eine einheitliche Repräsentationsform der Objekt-Geometrie liefern. Die »Bounding Volumes« sind für einen Großteil der angestrebten Einsatzfälle hinsichtlich ihrer Genauigkeit ausreichend und haben zugleich den Vorteil, dass sie aufgrund ihrer geringen Komplexität als effiziente Ausgangsbasis für nachfolgende geometrische Berechnungen dienen.

Zunächst soll also auf die Verwendung exakter Objekt-Geometrie verzichtet werden. Dennoch ist eine korrekte Handhabung von Objekten und insbesondere von Hilfsmitteln (z.B. Werkzeuge) für die angestrebte Plausibilität der resultierenden Bewegungs-Simulationen eminent wichtig. Dieser Widerspruch lässt sich durch eine ergänzende Attributierung der verwendeten Objekte auflösen. Der Begriff »Handhabung« beinhaltet bereits als Wortstamm den Hinweis auf den relevanten Bezugskörperteil »Hand«. Demnach sind die charakteristischsten Merkmale der Objekt-Handhabung die sogenannten »Greifpunkte«. Ein Greifpunkt entspricht dabei einer Position und Ausrichtung, welche im lokalen Koordinatensystem des entsprechenden Objekts definiert ist und damit einen Bezug zur Lage des Objekts in der Hand des Menschmodells repräsentiert. Darüber hinaus erwies sich die Ergänzung der Greifpunkt-Definition um ein weiteres Klassifikationsmerkmal – die Zuordnung zu einer oder beiden Händen – als notwendig. Der Grund hierfür lässt sich gut am Beispiel eines Werkzeuggriffs nachvollziehen (Abbildung 8). Unter der Annahme, dass die z-Achse des Greifpunkts auf die Handinnenfläche zeigt und die x-Achse in Fingerrichtung orientiert ist, liegen die Greifpunkte für die linke und rechte Hand auf den gegenüberliegenden Seiten des Griffs. Bei der Betrachtung der Anwendung von verschiedenen Handwerkzeugen konnte darüber hinaus die Notwendigkeit zur Vorgabe von Wirkpunkten ermittelt werden. Diese definieren in Anlehnung an die aus Robotersimulationen bekannten »Tool Center Points« (kurz: »TCP«) den Punkt samt Ausrichtung am Werkzeug, der bei der Anwendung an die vorgegebene Wirkstelle bzw. Zielposition am Werkstück geführt werden muss.

Die Handhabung von Objekten beinhaltet im Allgemeinen neben dem Aufnehmen auch ein Platzieren selbiger an einem bestimmten Zielort. Aus datentechnischer Sicht wird der Ablageort eines Objekts durch die Position und Ausrichtung des Objektkoordinatensystems am Zielort bestimmt. Um dem Anwender eine möglichst intuitive Möglichkeit zur Definition von Platzierzielen bieten

zu können, entstand die Idee zur Verwendung von frei positionierbaren Objektkopien – den sogenannten »Referenzobjekten«. Diese besitzen die identische geometrische Struktur ihres „Eltern-Objekts“, so dass die Definition von Zielorten durch einfaches Positionieren und Ausrichten der entsprechenden »Referenz« erfolgt.

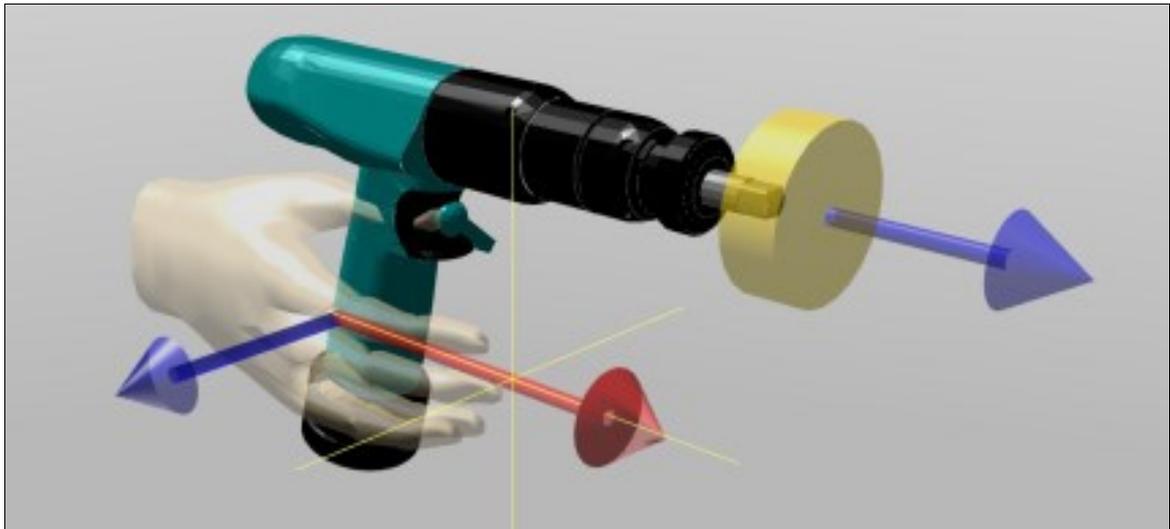


Abbildung 8: Greifpunkt und TCP an einem Akkuschauber

Neben den bereits ermittelten Informationen aktiver Komponenten – also solcher Objekte, welche direkt durch das Menschmodell manipuliert werden sollen – sind für realistischere Simulationen auch Informationen über passive Objekte erforderlich. Insbesondere automatische Funktionen zur Generierung von Bewegungen zum Ortswechsel und zur Vermeidung von Kollisionen bedürfen zusätzlicher Angaben. So sind für eine Kollisionsvermeidung ebenfalls Informationen bezüglich Gestalt und Lage der entsprechenden „Störobjekte“ erforderlich. Einmal mehr eignet sich dabei die Verwendung von »Bounding Volumes« als Abstraktion der tatsächlichen Objektgeometrie, um in einer ersten Ausbaustufe zu Ergebnissen zu gelangen. Die Betrachtung von „Real-Geometrie“ ist hierbei zwar grundsätzlich möglich; aufgrund des zu erwartenden Verarbeitungsaufwands und des vergleichsweise geringen Mehrwerts bei der Bewegungspfad-Ermittlung wird diese Variante aber zukünftigen Ausbaustufen vorbehalten bleiben und ist somit nicht Bestandteil dieser Arbeit.

Zur Ermittlung von Bewegungspfaden sind neben der Definition derjenigen Objekte, die als Hindernisse angesehen werden sollen, auch Informationen über „betretbare“ bzw. „passierbare“ Elemente notwendig. Insoweit wurde als absolutes Informations-Minimum zuvor bereits die Vorgabe einer Fußbodenhöhe genannt. Da jedoch nicht jedes abzubildende Simulations-Szenario immer nur einen ebenen Fußboden enthält, erwies sich die Vorgabe von sogenannten »Bodenobjekten« als sinnvoll. Als »Bodenobjekt« gelten dabei alle Objekte, welche vom Menschmodell betreten werden dürfen. Nicht nur einzelne Stufen, Schwellen und ganze Treppen können auf diese Art und Weise abgebildet werden. Auch die Möglichkeit zur Definition von Szenarien über mehrere Etagen wird dadurch eröffnet. In späteren Ausbaustufen ist sogar die Ergänzung komplexerer Bewegungsformen wie beispielsweise »Leiter klettern« denkbar. Voraussetzung dafür ist eine Möglichkeit zur Erkennung von „als Steighilfe verwendbaren“ Objekten. Diese Identifizierung kann zunächst durch Attributierung unterstützt werden oder gegebenenfalls später durch Geometrie-Erkennung erfolgen.

Neben der Visualisierbarkeit der Bewegungsabläufe zählt die Ableitbarkeit von Ausführungszeit- und Ergonomie-relevanten Parametern zu den Kernanforderungen des zu schaffenden Systems. Da für nachfolgende Ergonomie-Analysen [48] insbesondere die physikalische Eigenschaft der »Objekt-Masse« benötigt wird, sollte auch diese direkt als Attribut an Objekten mitgeführt werden. Neben dem Bereich der Ergonomie kann diese Eigenschaft in Zukunft auch zur Simulation von humanmotorischen Bewegungseinflüssen und physikalischen Aspekten verwendet werden.

Ein Grundgedanke der Konzeption des zu entwickelnden Systems besteht darin, möglichst viele der notwendigen Eingaben und Parameter automatisch zu bestimmen. Da die Implementierung von Algorithmen, welche diese „Intelligenz“ nach außen hin simulieren sollen, aufgrund deren Anzahl und Komplexität sehr aufwändig ist, werden zunächst entsprechende Schnittstellen vorgesehen. Die Bedienung dieser Schnittstellen erfolgt dabei anfangs durch direkte Benutzereingaben. In den dieser Arbeit nachfolgenden Entwicklungsschritten sollen die Benutzereingaben durch entsprechende Algorithmen ersetzt werden.

3 Methodische Vorarbeit

Eingangsinformationen	Datentyp	Beschreibung
Bodenhöhe / -objekte	Fließkommazahl	Definition von „begehbaren“ Elementen, relevant zur Berechnung von Fußpunkten
3D-Objekte	Transformationsmatrix & Bounding Volume	Definition von Interaktions-Objekten, relevant für Tätigkeiten, die den Transport von Objekten beinhalten
Arbeitspositionen	Transformationsmatrix	definiert Punkte samt Ausrichtungen, dient zur Festlegung von Standorten zur Ausführung von Tätigkeiten
Greifpunkte	Transformationsmatrix (lokal im Werkzeug-KOS)	definiert Punkte samt Ausrichtungen (lokal im Objektkoordinatensystem), dient zur Festlegung wo und wie die jeweilige Hand am Objekt „anfassen“ soll
Wirkpunkte	Transformationsmatrix (lokal im Werkzeug-KOS)	definiert einen Punkt samt Ausrichtung (lokal im Objektkoordinatensystem), dient zum Ausrichten eines Werkzeug-Objekts am Zielort
Referenzobjekte	Transformationsmatrix & Bounding Volume	eine Kopie eines anderen Objekts, dient zur intuitiven Definition dessen Ablageorts
Kollisionsobjekte	Transformationsmatrix & Bounding Volume	Definition von Hindernissen für Laufwegs-Bestimmung und Kollisionsvermeidung
Objekt-Massen	Fließkommazahl	Objektparameter für Ergonomie-Auswertung, später auch zur Simulation physikalischer Aspekte

Tabelle 8: Relevante Umgebungsinformationen

3.4.2 Tätigkeitsbeschreibung

Neben den Informationen über die Umgebung des Menschmodells muss dem zu schaffenden System eine Beschreibung der auszuführenden Tätigkeiten übergeben werden. In der Analyse existierender Systeme wurden Mängel in der Zielorientiertheit und der Effizienz der Modellierung festgestellt, da primär die Beschreibung von Bewegungen durch die Definition von Körperhaltungen im Vordergrund steht. Dies führt dazu, dass die erzielte Qualität der Ergebnisse mit den Fähigkeiten und Erfahrungswerten des Simulations-Erstellers in direktem Zusammenhang steht – was zudem ein Grund für die mangelhafte Objektivität und Reproduzierbarkeit von Menschmodell-Simulationen ist. In der Anforderungsanalyse wurde deshalb die Forderung erhoben, dass sich das zu entwickelnde System der Sprach- und Methodenwelt des Planungingenieurs bedienen soll.

Als geeignetes, weil weitverbreitetes Mittel in der Modellierung von Arbeitsprozessen gilt die Verwendung von Prozesssprachen. Diese abstrahieren Probleme der realen Welt soweit wie möglich, bewahren dabei aber gleichzeitig die relevanten Kerninformationen der beschriebenen Abläufe. Hierdurch wird gewährleistet, dass Prozesse vollständig dokumentiert werden können. Dabei wird durch die Eingrenzung auf einen beschränkten Begriffsvorrat und die Verwendung syntaktischer Konstrukte die Objektivität und Eindeutigkeit der Beschreibung sichergestellt. Somit war das Ziel klar umrissen – es sollte eine technologisch-orientierte Prozesssprache entwickelt werden, welche sich zur Modellierung von Simulationen für digitale Menschmodelle eignet. Die erste Frage hierbei war, welche Informationen diese Prozesssprache transportieren muss, um in nachfolgenden Verarbeitungsschritten als Grundlage einer Simulation dienen zu können. Hierzu verwendete der Autor die im Bereich der Prozessanalyse gebräuchliche Methode der »W-Frage-Technik« [35, p. 41ff]. Diese dient primär der Sicherstellung der Vollständigkeit der Beschreibung, wird zum Beispiel im sogenannten »6W-Fragen Arbeitsblatt« [38] angewendet und lässt sich wie folgt auf das vorliegende Problem übertragen (Abbildung 9):

Warum soll etwas getan werden?

Die zu simulierenden Prozesse ordnen sich in der Realität in übergeordnete Prozessfolgen ein und haben somit einen abgegrenzten Ausführungszweck in-

nerhalb dieser Strukturen. Existierende Beziehungen zu Nachbarprozessen können dabei sowohl die Startkonfiguration als auch den Verlauf der beschriebenen Tätigkeitsfolge beeinflussen.

Wer soll die Tätigkeit ausführen?

Neben der Festlegung, wie viele digitale Menschmodelle die Tätigkeiten in der Simulation ausführen sollen, ist auch eine eindeutige Zuteilung der Aufgaben zu den vorhandenen Menschmodell-Instanzen erforderlich.

Was soll (womit) ausgeführt werden?

Bestimmte Tätigkeiten müssen unter Verwendung bestimmter Objekte (aufnehmen, platzieren) oder unter Zuhilfenahme von Hilfsmitteln (Werkzeuge) ausgeführt werden. Die Angabe der „Art der Ausführung“ muss dabei um die entsprechenden Bezüge der zu verwendenden Objekte ergänzt werden.

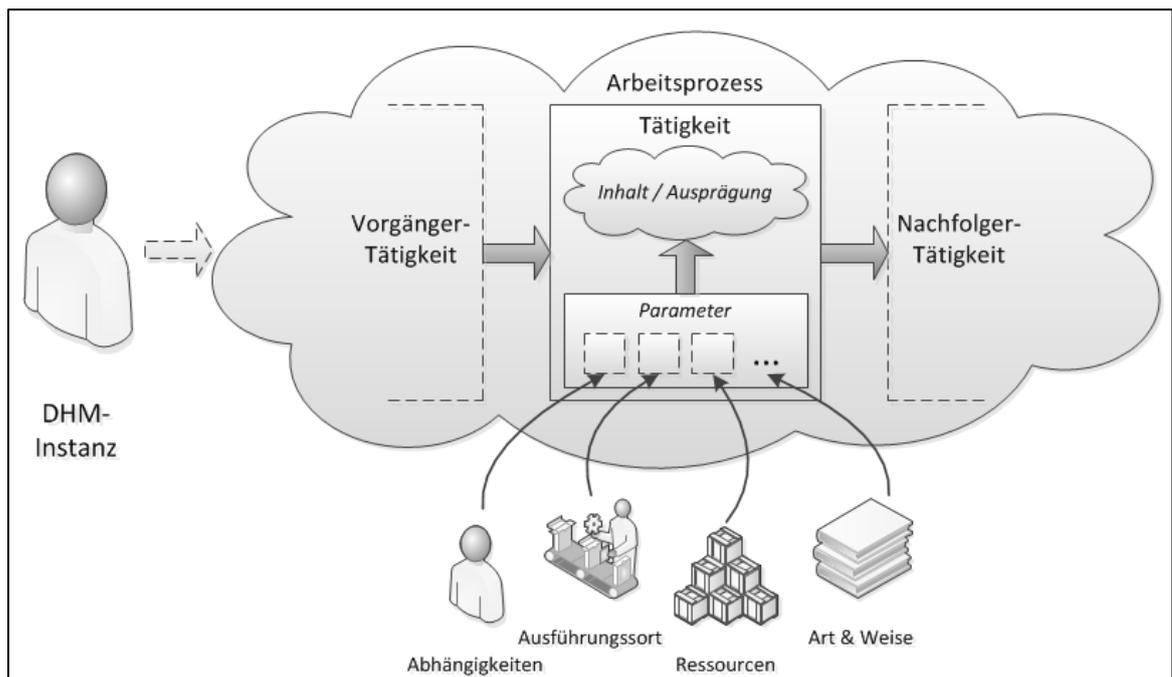


Abbildung 9: Inhalte der Tätigkeitsbeschreibung

Wo soll die Tätigkeit ausgeführt werden?

Da viele Tätigkeiten das Vorhandensein bestimmter Objekte (Ressourcen) voraussetzen und sich diese Objekte an den vom Planungsingenieur definierten

Positionen im Layout befinden, sind nahezu alle Tätigkeiten in der Produktion ortsgebunden. Dem digitalen Menschmodell muss somit entweder eine konkrete geometrische Position im 3D-Layout oder zumindest ein Objektbezug zur Durchführung der Tätigkeit vorgegeben werden. Darüber hinaus müssen entsprechende (Bewegungs-)Möglichkeiten zum Ortswechsel vorgesehen werden.

Wann soll die Tätigkeit ausgeführt werden?

Die Beschreibung des Tätigkeitsinhaltes wird durch die Definition der zeitlichen Abfolge vervollständigt. Wichtige Kenngrößen sind hierbei Startzeitpunkte, vorgegebene Tätigkeitsdauern und Möglichkeiten zur Definition von zeitlichen Abhängigkeiten. (z.B. durch Synchronisationspunkte)

Wie soll die Tätigkeit ausgeführt werden?

Die Art der auszuführenden Tätigkeit muss in einem für das System interpretierbaren Format angegeben werden. Eine entsprechende Menge möglicher „Teil-Tätigkeiten“ muss somit vordefiniert worden sein. Darüber hinaus können nahezu alle menschlichen Tätigkeiten auf vielfältige Art und Weise ausgeführt werden. Diese „Art der Ausführung“ macht die nähere Spezifizierung der zuvor genannten Art der Tätigkeit durch eine tätigkeitsabhängige Menge von Parametern notwendig.

Aus der Beantwortung der „W-Fragen“ konnten damit einige Erkenntnisse für die Konzeption der erforderlichen Schnittstelle und auch für das über die Schnittstelle anzusteuern System gewonnen werden. Als Hauptaufgabe wurde die Definition des „Tätigkeitsvorrats“ identifiziert. Hierzu zählen insbesondere der Inhalt und die Parametrisierung jeder einzelnen Tätigkeit, die von den digitalen Menschmodellen durchgeführt werden soll. Des Weiteren müssen die einzelnen Elemente bei der Übergabe an das System sowohl in einer festgelegten Reihenfolge als auch mit einer Zuordnung zu konkreten Instanzen von Menschmodellen versehen sein.

3 Methodische Vorarbeit

Frage	»6W-Fragen Arbeitsblatt« (Ishiwata, 2001)	Anwendung / Relevanz für System
Warum	spezifische Beschreibung der Operation	Abgrenzung gegenüber Nachbarprozessen, Berücksichtigung für Startzustand und Abhängigkeiten
Wer	Arbeitsbezeichnung, Anzahl der Arbeiter, Namen der Arbeiter, etc.	Aufgabenverteilung, Zuordnung der Tätigkeiten zu konkreten Instanzen digitaler Menschmodelle
Was	Bezeichnung der Maschinen und Betriebsmittel, Bezeichnung der Vorrichtungen, Anzahl der Einheiten etc.	Verknüpfung der Tätigkeiten mit Ressourcen, Objektbezüge für direkte Interaktion während der Tätigkeit
Wo	spezifische Beschreibung des Operationsbereichs	konkrete Ortsbezüge oder Ressourcen-Bezüge, Bewegungen zum Ortswechsel vorsehen
Wann	Durchlaufzeit, Ausbringungsmenge (aufgewendete Zeit pro Einheit)	Startzeitpunkte, Tätigkeitsdauern, Abhängigkeiten, Synchronisationsbedingungen
Wie	spezifische Beschreibung der Operationsabfolge	Definition eines Vorrats an Tätigkeitsbausteinen, Parametrisierung dieser Tätigkeitsbausteine

Tabelle 9: Relevante Tätigkeitsinformationen

3.5 Entwicklung der Absichtskomponente

3.5.1 Arten und Strukturierung des relevanten Wissens

Bei der Ermittlung und Betrachtung der notwendigen Systemkomponenten (siehe Kapitel 3.3) wurde die sogenannte Absichtskomponente als notwendig erachtet. Diese soll die Eingabeinformationen, bestehend aus Umgebungs- und Tätigkeitsbeschreibung, in eine Menschmodell-unabhängige Bewegungsbeschreibung übersetzen. Hierzu soll das digitale Menschmodell mit Kenntnissen und Fertigkeiten eines Facharbeiters ausgestattet werden, um entgegen dem bisherigen „per Hand Modellieren“ einzelner Körperhaltungen, Bewegungen automatisch aus den zuvor beschriebenen Eingaben generieren zu können. Das digitale Menschmodell soll damit zum „virtuellen Facharbeiter“ entwickelt werden. Die hierfür erforderlichen „Kenntnisse und Fertigkeiten“ gründen sich auf dem Wissens- und Erfahrungsschatz, den sich ein Mensch im Laufe seines Lebens aneignet. Somit galt es zunächst zu ermitteln, welche Elemente des Wissens für die Aufgabe der Ausführung von Tätigkeiten in der industriellen Produktion überhaupt relevant sind. Anschließend musste eine Möglichkeit geschaffen werden, um dieses Wissen für das zu entwickelnde Simulationssystem nutzbar zu machen.

Es gibt verschiedenste Versuche zu erklären, was »Wissen« ist. Eine solche Definition liefert beispielsweise Seiler: „Wenn man sowohl im Alltag, als auch in der Wissenschaft vom Wissen einer Person spricht, wird ihr dieses Wissen gleichsam als ihr eigener Besitz oder ein Gut zugeschrieben, welche sie durch Lernen erworben und im Gedächtnis gespeichert hat.“ [49, p. 10] Um die Bedeutung des Wissens für den Mensch zu unterstreichen, ergänzt er: „... dass er [der Mensch] ohne dieses intuitive Wissen weder leben, handeln, noch reden könnte.“ [49, p. 11] Zusammenfassend muss das Wissen somit als wichtige Grundlage für Entscheidungen und Handeln eines menschlichen Individuums angesehen werden. Ebenso existieren mehrere Ansätze, um das Wissen zu kategorisieren. Ein in den Humanwissenschaften verbreiteter Ansatz basiert auf der Unterscheidung von implizitem und explizitem Wissen. Als Vorreiter der Untersuchung des impliziten oder auch „stillen“ Wissens gilt Michael Polanyi, der sich 1966 in seinem Werk »The Tacit Dimension« [50] eingehend damit beschäftigte. Ein Ausgangspunkt seiner Betrachtungen lag in der Feststellung „... we can know more than we can tell ...“ [50, p. 4]. Dieses Zitat beschreibt sehr gut die

Tatsache, dass implizites Wissen als nicht direkt oder zumindest nicht einfach artikulierbar gilt. Im Gegensatz hierzu lässt sich explizites Wissen sehr gut beschreiben: „Explicit knowledge can be expressed in words and numbers, and easily communicated and shared in the form of hard data, scientific formulae, codified procedures, or universal principles.“ [51, p. 8]

Im Hinblick auf das zu entwickelnde Simulationssystem sind nach Ansicht des Autors beide zuvor beschriebenen Arten des Wissens relevant, um die Bewegungen für ein digitales Menschmodell unter den gegebenen Voraussetzungen erzeugen zu können. Ein Kernthema der Untersuchungen hierzu war die Frage „Wie bewegt sich ein Mensch?“. Bereits bei relativ einfachen Körperbewegungen wie zum Beispiel einem Abknien oder Hinlegen wurde bei einfachen Versuchen deutlich, dass zwar jede gesunde Testperson diese Bewegungen ausführen konnte. Eine detaillierte Beschreibung des Bewegungsablaufs und der Art und Weise, wie sich die einzelnen Körperteile dabei bewegen, konnte im Vorfeld jedoch keiner der Probanden geben. Außerdem fiel auf, dass die Bewegungsausführung zwischen den Probanden mitunter stark variierte (z.B. beim Hinlegen) und auch keine belastbaren Auskünfte darüber gegeben werden konnten, warum die Bewegungen gerade so und nicht anders ausgeführt wurden. In der Fachliteratur wird diese Art des offensichtlich impliziten Wissens als »Bewegungsintelligenz« beschrieben: „Die neuromuskulären Teile von Skelett und Organen reagieren auf Grund eines gesammelten gemeinsamen Wissens, das permanent erweitert wird und dabei ständig von Gefühl und Verstand überprüft wird.“ [52, p. 17] Zu den vom Simulationssystem abzubildenden Aspekten der »Bewegungsintelligenz« zählen Antworten auf die Fragen:

- Welche Teilbewegungen sind erforderlich, um eine bestimmte Körperhaltung einnehmen zu können? (Statomotorik und Sensomotorik)
- Welche (Rand-)Bedingungen sind zu erfüllen, um eine bestimmte Körperhaltung einnehmen oder Bewegung ausführen zu können?
- Wie gelangt das digitale Menschmodell von einem Ort zu einem anderen Ort? (Navigation und Lokomotorik)

Das Simulationssystem soll auch in der Lage sein, Tätigkeitsbeschreibungen aus der Planungswelt zu verarbeiten. Die aufgrund der Anwender-Zielgruppe »Planungsingenieur« dabei in Frage kommenden Formulierungen und Begrifflichkeiten entspringen etablierten Methoden aus der Fachwelt der Fertigungsprozess-

planung bzw. des Maschinenbaus. Da sich dieses Wissen sowohl auf Erfahrung als auch auf eine bewusst erlernte oder geschulte Basis zurückführen lässt, kann es als überwiegend explizites Wissen eingestuft werden. Hierzu zählen u.a.:

- Bedeutungen und Übersetzungen von technischen Fachbegriffen
- die Strukturierung von komplexen Aufgaben in Teilarbeitsschritte
- der Umgang mit Werkzeugen, Werkstoffen und weiteren speziellen Ressourcen der technischen Welt
- das „korrekte“, zielgerichtete Verhalten in der Produktionsumgebung

Die eindeutig in implizites oder explizites Wissen kategorisierbaren Elemente werden durch Aspekte ergänzt, bei denen die Zuordnung nicht so eindeutig möglich ist. Ein konkretes Beispiel hierfür sind die Aspekte der Objekt-Handhabung, also aller Teilschritte vom Herstellen des Kontaktes zwischen dem Mensch (bzw. seiner Hände) und einem Objekt bis zur Aufhebung dieses Kontaktes. Je nachdem, um welche Art von Objekt es sich handelt, kann die Art der Handhabung durch implizites Wissen bei einfachen Objekten oder durch explizites Wissen bei speziellen Objekten (z.B. Werkzeugen) bestimmt werden. Hier spielt neben der Art des Objekts dann auch die auszuführende Tätigkeit eine entscheidende Rolle. Während bei einfachen Aufnahme-, Trage- und Platziertätigkeiten kaum bis gar kein explizites Wissen erforderlich ist, wird dieses bei der Applikation eines speziellen Werkzeugs umso dringender benötigt. Grundsätzlich muss das zu entwickelnde System somit implizites und explizites Wissen in seiner prozeduralen Wissensbasis abbilden können, um dem Anspruch eines „virtuellen Facharbeiters“ gerecht zu werden.

3.5.2 Der Verrichtungs-Begriff

Ein zentrales Element aller Forschungs- und Entwicklungsarbeiten des im Rahmen dieser Arbeit vorgestellten Simulationssystems für digitale Menschmodelle stellt der Begriff der »Verrichtung« dar. Abgeleitet vom Verb „verrichten“ bezeichnet es im Arbeitsstudium die sichtbaren Bewegungen einer Arbeitsperson bei der Ausübung einer Tätigkeit [53, p. 191f]. Diese sehr allgemein gehaltene Definition und insbesondere deren Quelle skizzieren die Absicht, die hinter der Verwendung des Begriffs und der damit verbundenen Herangehensweise stand. Durch die Nutzung von Begrifflichkeiten, die der Gedankenwelt der potentiellen Anwender entlehnt sind, sollte die Zugänglichkeit und Akzeptanz des Systems gewährleistet werden. Darüber hinaus hat sich der reale Facharbeiter diese Art der Strukturierung von Arbeitsinhalten im Laufe seiner Ausbildung ebenfalls angeeignet. Somit sollte die Formulierung von Verrichtungen auch als geeignete Basis dienen, um sein virtuelles Pendant anzusteuern.

Ein weiteres Ziel war die Befreiung des Arbeitsgestalters von den ihm weniger bekannten Bereichen der Anthropometrie, Biomechanik und Humanmotorik. So fällt es einem Planungsingenieur leicht, sich die Arbeitsvorgänge von Werkern anhand verbaler Ablaufbeschreibungen vorzustellen oder diese umgekehrt verbal zu beschreiben. Zur Zerlegung menschlicher Bewegungen in diskrete Einzelposen fehlen sowohl die erforderlichen Kenntnisse als auch die notwendige Zeit im täglichen Planungsgeschäft. Deswegen sollte es Aufgabe der Verrichtungen bzw. des Simulationssystems sein, die Bewegungsgenerierung zu übernehmen.

Bereits in Bauer [10] wurde ein erster Entwurf eines Modells von sogenannten »Bewegungsebenen« vorgestellt (Abbildung 10). Das Modell besteht aus 5 Ebenen, wobei mit aufsteigender Nummerierung der Ebenen eine Detaillierung der Bewegungen erfolgt. Zu jeder Ebene wurde neben einer eindeutigen Bezeichnung auch ein charakteristisches Beispiel angegeben. Bemerkenswert sind dabei die Tatsachen, dass erst ab der Ebene 3 die Bezeichnung »Verrichtung« auftritt und dass es eine Zwischenebene (Ebene 4) bestehend aus Bausteinen des Sollzeit-Verfahrens MTM-UAS gibt. Das Modell diente in der Folge als Ausgangspunkt weiterer wissenschaftlicher Arbeiten und wurde zudem prototypisch erprobt. [10, p. 66ff] Darüber hinaus lieferte die Arbeit eine präzisere und auf das Simulationssystem abgegrenzte Begriffs-Definition: „Die Verrichtungen

wurden dabei als wiederkehrende mit Parametern belegbare Makros implementiert, welche sich bausteinartig zu Gesamtprozessen zusammensetzen lassen.“ [10, p. 2]

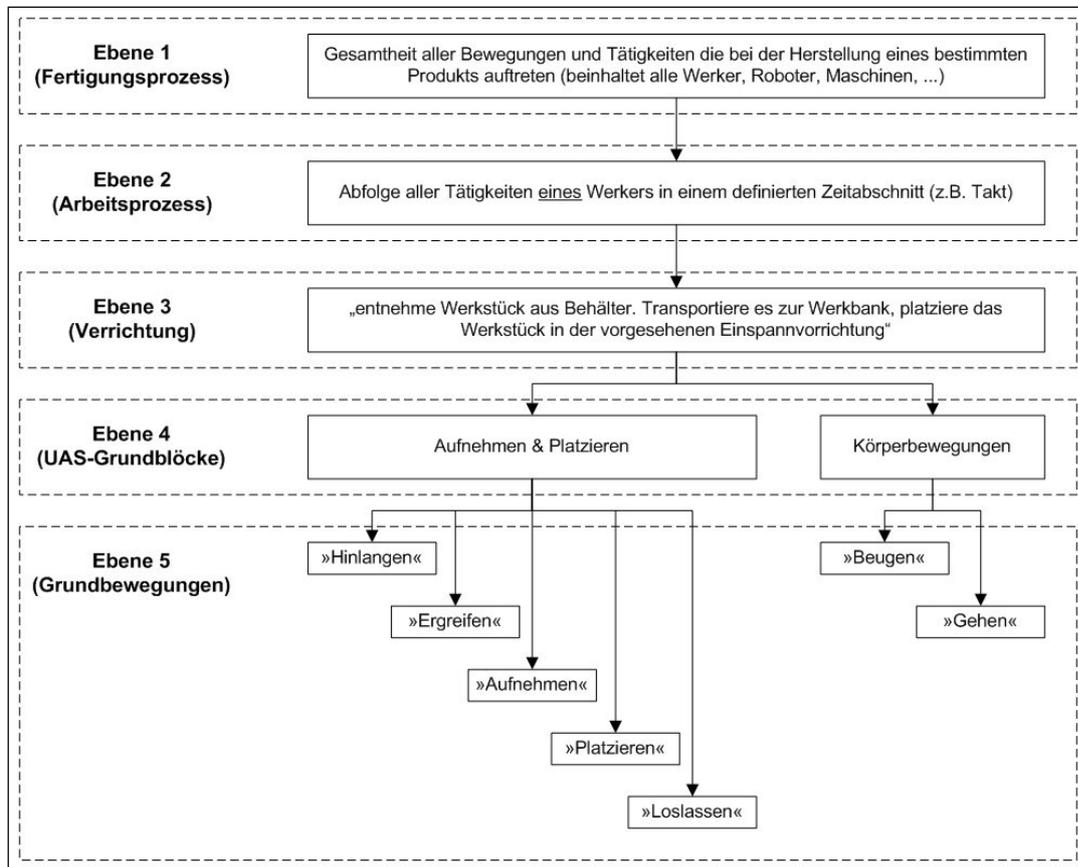


Abbildung 10: Bewegungsebenen [10, p. 22]

Ab dem Jahr 2008 beschäftigten sich erneut zwei wissenschaftliche Arbeiten mit der Strukturierung und Aufbereitung von Verrichtungen. In Speiser [54] wurde erstmals eine weitere Untergliederung von Verrichtung in sogenannte Komplex- und Teilverrichtungen diskutiert. Am Beispiel des Themenkomplexes „Schraubarbeiten“ wurden darüber hinaus Vorschläge einer möglichen Parametrisierung und die sich daraus ergebenden Probleme einer angedachten „Universal-Schraub-Verrichtung“ erarbeitet. Die Grundlagen für die Untersuchung waren einmal mehr die Unterlagen zur Strukturierung manueller Tätigkeiten von MTM [32] und die aus Experteninterviews gewonnenen Erkenntnisse typischer Parameter und deren Merkmalsausprägungen bei Schraubvorgängen. Im Ergebnis wurden 17 (technologisch geprägte) Verrichtungsparameter mit entsprechenden

Ausprägungen ermittelt. Allein durch deren mögliche Kombinationen hätte innerhalb dieser „Universal-Schraub-Verrichtung“ bereits eine Anzahl von Varianten im zweistelligen Millionenbereich abgebildet werden müssen. Aus der Konfrontation mit dem aufgetretenen Komplexitätsproblem entstand daher die Idee der Einführung einer weiteren Zwischenebene – der Ebene der sogenannten »Teilverrichtungen«, welche ihren Platz zwischen Ebene 3 und 4 (siehe Abbildung 10) finden sollte. Die neue Struktur der »Teilverrichtung« wurde nun als nicht mehr sinnvoll teilbares Segment einer (Komplex-)Verrichtung definiert, wobei sich komplexe Verrichtungen in Teilverrichtungen zerlegen und umgekehrt aus Teilverrichtungen wiederum Komplexverrichtungen erstellt werden können [54, p. 32]. Die „nicht mehr sinnvolle Teilbarkeit“ der Verrichtungen bezog sich hierbei jedoch ausschließlich auf den technologisch geprägten Aspekt der Tätigkeitsbeschreibung. Denn die Tatsache, dass sich auch unterhalb der Ebene der Teilverrichtungen noch weitere Ebenen befinden, deutet darauf hin, dass aus humanmotorischer Sicht sehr wohl weitere Untergliederungen möglich sind.

Parallel zur Arbeit von Speiser arbeitete Uhlig an seiner Diplomarbeit mit dem Titel »Analyse, Systematisierung und Vorbereitung der Implementierung parametrischer Verrichtungen zur Anwendung im Planungswerkzeug „Editor menschlicher Arbeit“« [55]. Als Einstieg wählte Uhlig dabei die Untersuchung mehrerer realer Fertigungsprozesse aus den Bereichen Karosserierohbau und Automobilendmontage. Im Vordergrund stand dabei zunächst das Ziel, der Ermittlung, welche Arten von Verrichtungen überhaupt relevant sind und mit welcher Häufigkeitsverteilung diese auftreten. Auch wenn aufgrund der beschränkten Anzahl der untersuchten Fallbeispiele keine allgemeingültige Aussage ableitbar war, so konnten zumindest Tendenzen für die Häufigkeiten der einzelnen Tätigkeitsgruppen im Bereich des Automobilbaus ermittelt werden. So kam den Gruppen »Schrauben«, »Manuelles Schweißen« und »Aufnehmen, Positionieren, Ablegen« mit jeweils rund 20% Häufigkeit die größte Bedeutung zu. Die Tätigkeitsgruppen »Stecken, Clipsen«, »Kleberauftrag manuell«, »Nieten«, »Maschinenbedienung« waren hingegen nur im einstelligen Prozentbereich vertreten (Abbildung 11). Diese Voruntersuchungen wurden nun genutzt, um einen Ansatz für die hierarchische Gliederung von Verrichtungen zu erarbeiten. Wie Speiser kam auch Uhlig zu dem Schluss, dass es zwei grundsätzlich gegensätzliche Ansätze zum Aufbau von Verrichtungen gibt. Das ist zum einen die

Beschreibung von Verrichtungen als komplexe Verrichtungsfunktionen, die mit entsprechend vielen Parametern ausgestattet die Abbildung zahlreicher Varianten ermöglichen und zum anderen ist es die Beschreibung durch ein Bausteinsystem, bestehend aus »Teilverrichtungen«.

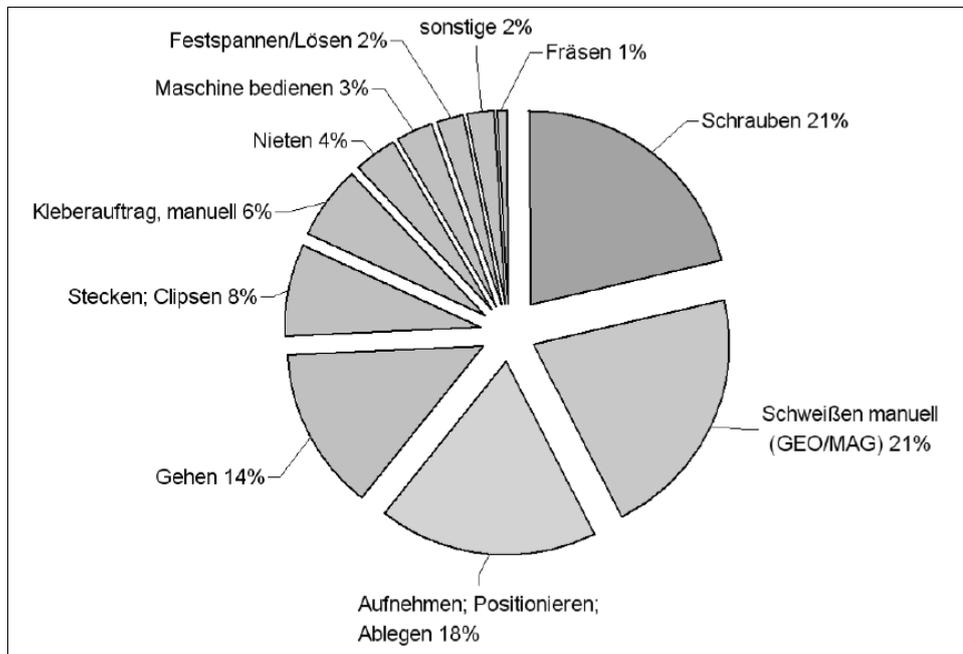


Abbildung 11: Häufigkeiten von Verrichtungen im Automobilbau [55, p. 36]

Bei der Entscheidung zwischen den beiden Ansätzen nutzte der Autor die aus Anwendersicht vorrangige Frage nach Aufwand und Nutzen. So verursacht die Suche nach passenden Verrichtungen in einem großen Katalog von »Teilverrichtungen« einen ähnlich hohen Aufwand wie die Konfiguration von Verrichtungsfunktionen bzw. Komplexverrichtungen mit zahllosen Parametern. (Abbildung 12) Außerdem musste die angestrebte universelle Einsetzbarkeit des zu entwickelnden Simulationssystems berücksichtigt werden. Hierbei erschien die Entwicklung eines feingliedrigen Systems von Elementarverrichtungen als erster sinnvoller Schritt. Auf diese Art und Weise konnte zunächst eine Grundmenge an Funktionalität geschaffen werden, welche sich später durch Kombinationen zu komplexeren Konstrukten aggregieren und sich dabei besser algorithmisch „beherrschen“ (Divide-and-Conquer) und pflegen lässt. Als positiver Nebeneffekt wurden dadurch auch die funktionalen Überdeckungen der einzel-

3 Methodische Vorarbeit

nen Verrichtungen durch möglichst häufige Wiederverwendung bereits implementierter Algorithmen auf ein Minimum reduziert.

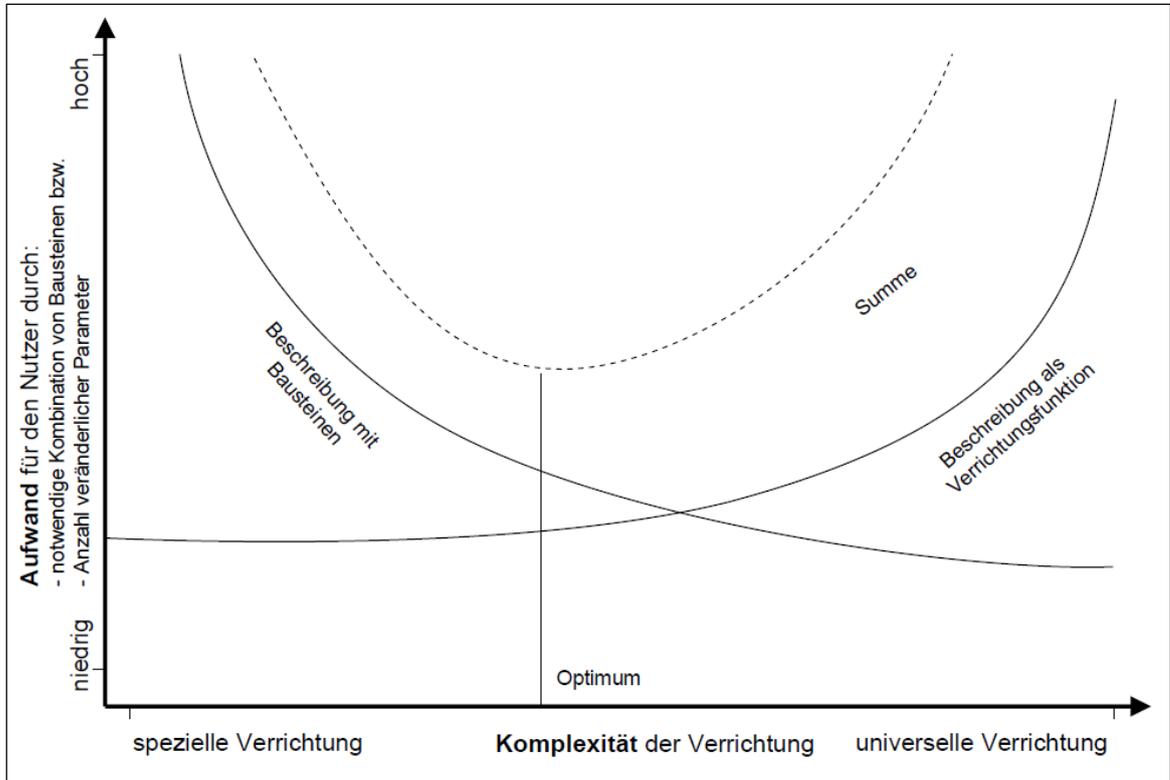


Abbildung 12: Verrichtungskomplexität vs. Modellierungsaufwand [55, p. 45]

3.5.3 Konzeption der Verrichtungsbibliothek

Beginnend bei den ersten frühen Entwürfen der Bibliotheksstruktur [10] über detailliertere Ebenen-Modelle [55] bis hin zur ersten fertigen Version entwickelte sich die Verrichtungsbibliothek in einem iterativen Prozess. Natürlich existierten seit vielen Jahren unterschiedlichste Gliederungssysteme für manuelle Tätigkeiten, welche auch zu Planungszwecken eingesetzt wurden. (Abbildung 13) Eine „fertige“ Vorlage ließ sich in den vorhandenen Systemen jedoch nicht finden: „... Fachliteratur, Normschriften oder andere etablierte Methodensammlungen bieten derzeit keine brauchbare Gliederung für manuelle Tätigkeiten (Verrichtungen) auf verschiedenen Ebenen an, um diese entsprechend den sich ergebenden Anforderungen, rechnerunterstützt zu planen, zu visualisieren und zu bewerten.“ [55, p. 65]

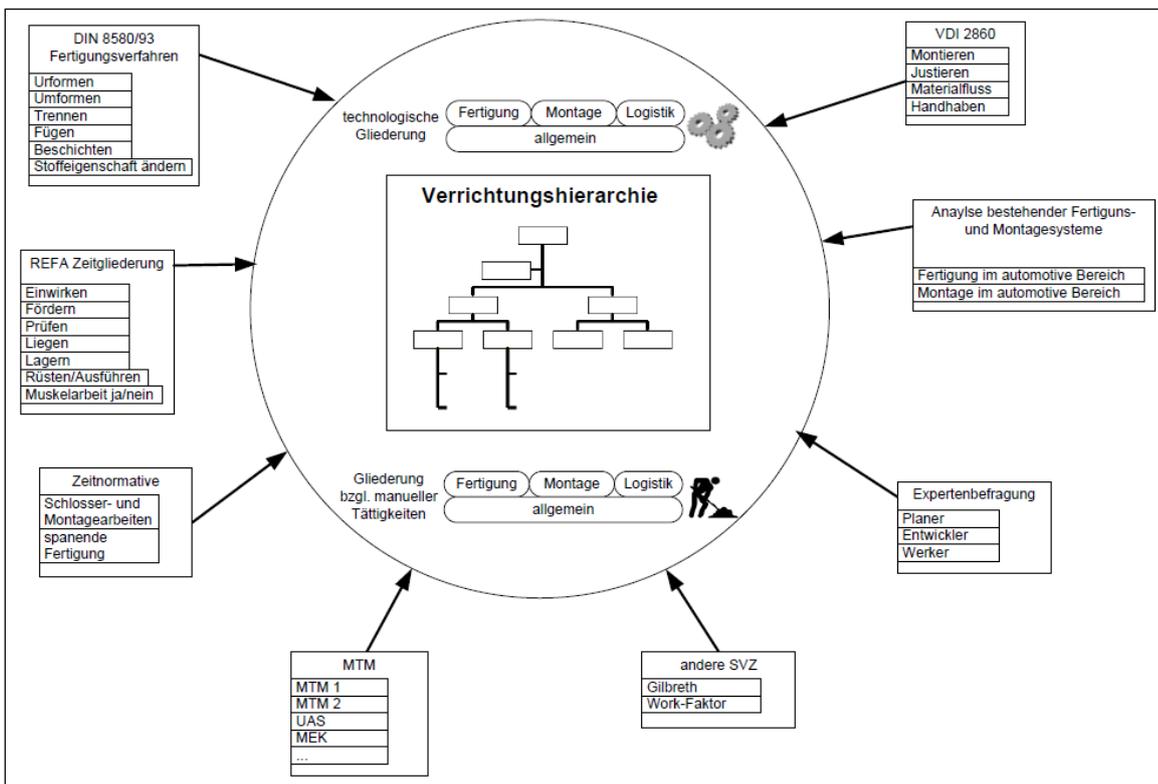


Abbildung 13: Bestehende Gliederungen manueller Tätigkeiten [55, p. 51]

Dennoch hatten die etablierten Systeme einen entscheidenden Vorteil. Sie prägen den typischen Sprachgebrauch des Planungsingenieurs und konnten somit ein entscheidender Faktor bei der Sicherstellung der Zugänglichkeit des neuen

Simulationssystems sein. Der Autor wählte deshalb das MTM-Verfahren als Ausgangspunkt der Entwicklung. Zum einen hat dieses im angestrebten Anwenderkreis eine hohe Verbreitung und zum anderen beinhaltet es bereits verschiedene Bausteinsysteme, welche die Beschreibung von Tätigkeiten auf unterschiedlichen Detaillierungsstufen (siehe Abschnitt 2.2) ermöglichen. Im Bereich des Automobilbaus werden aufgrund der auftretenden Taktzeiten und Arbeitsumfänge hauptsächlich das Verfahren »MTM-UAS« (kurz: UAS, Universelles Analysiersystem) oder kundenspezifische Derivate eingesetzt. Charakteristisch für MTM-UAS ist, dass die zur Verfügung stehenden Bausteine bereits relativ stark „verdichtet“ sind, d.h. Tätigkeiten und Bewegungen wurden darin bereits zusammengefasst. Zudem beinhalten die hinterlegten Zeitwerte zum Teil statistische Anteile. [56, p. 7] Was aus Sicht des Planers, der eine Zeitanalyse erstellen soll, ideal erscheint, barg jedoch beim Versuch der Ableitung einer Simulation mehrere Probleme. So wird beispielsweise das »Aufnehmen« und »Platzieren« von Objekten in einem Baustein zusammengefasst, selbst wenn es von anderen Bewegungen unterbrochen wird. Ebenso wird beim »Aufnehmen«, »Positionieren« und »Platzieren« von Werkzeugen verfahren. Daher bildet die Abfolge der analysierten Bausteine nicht zwangsläufig die Chronologie des Vorgangs ab, was wiederum für eine plausible Simulation zwingend erforderlich ist. Ein weiteres Problem stellen einige der zur Bestimmung der Baustein-Sollzeiten definierten »Einflussgrößen« dar. Hier werden stetige Wertebereiche, wie sie zum Beispiel bei Entfernungen auftreten, in sogenannte »Entfernungsbereiche« zusammengefasst und damit zum Zweck der Zeitbestimmung diskretisiert. Aus Simulations-Sicht ist dies jedoch eher ungünstig, da dadurch schon bei minimalen Veränderungen der Umgebungsbedingungen an den Unterteilungsintervallgrenzen „sprunghafte“ Geschwindigkeitsveränderungen der ausgeführten Bewegungen auftreten können. Zusammenfassend kam der Autor zu dem Schluss, dass MTM-UAS in Bezug auf seine Herangehensweise, also dem Zusammenfassen von Bewegungen, zwar grundsätzlich als Vorbild dienen konnte, aufgrund der festgestellten Mängel jedoch nicht als direkte Vorlage für die Verrichtungsbibliothek geeignet war.

In einem weiteren Schritt untersuchte der Autor nun das Bausteinsystem »MTM-1«. Dieses wird auch als »Grundverfahren« bezeichnet und bildet Tätigkeiten bis auf die Ebene von einzelnen Handgriffen und Fingerbewegungen ab. Es ermöglicht eine sehr detaillierte, wenn auch sehr aufwändige Art der Analyse,

da im Vergleich mit MTM-UAS bei gleichem Tätigkeitsinhalt deutlich mehr Bausteine verwendet werden müssen. Das Problem der zu starken Zusammenfassung und der dadurch verlorengehenden Chronologie besteht bei MTM-1 nicht. Dennoch werden auch im Grundverfahren diskrete Wertebereiche verwendet, welche aber zumindest bei den Armbewegungen feiner aufgelöst sind. Im Hinblick auf das geplante »Bottom-Up«-Verfahren bei der Entwicklung der Verrichtungsbibliothek erschien »MTM-1« dennoch als ein geeigneter Startpunkt.

Ein generelles Problem der Bausteine der Sollzeit-Bestimmungsverfahren ist die nur sehr rudimentär ausgeprägte Berücksichtigung von Objekt- bzw. Geometriebezügen. Dies ist nicht weiter verwunderlich, denn die Verfahren wurden schließlich nicht mit dem Ziel entwickelt, die Ableitung von Menschmodell-Simulationen zu unterstützen. Die Zuordnung, welche Objekte im jeweiligen Baustein verwendet werden sollen, ist kein fester Bestandteil der Prozesssprache MTM. Es ist dem Anwender selbst überlassen, Art und Anzahl der relevanten Objekte im benutzerdefinierten Beschreibungstext eines Bausteins zu spezifizieren. Ähnlich verhält es sich mit Ortsangaben. So enthalten einige Bausteine zwar Einflussgrößen, die geometrische Entfernungen repräsentieren. Angaben zu Bewegungsrichtungen oder Zielvorgaben fehlen dabei jedoch.

Basierend auf den gewonnenen Erkenntnissen entwickelte der Autor nun schrittweise eine eigene Prozesssprache bestehend aus Verrichtungen (Abbildung 14), welche als Eingabe für das Simulationssystem dienen sollte. Als Entwicklungsrahmen fungierten dabei die folgenden Vorgaben:

- Anlehnung an Struktur und Benennungen des Bausteinsystems von MTM-1
- Gewährleistung der Chronologie der Elemente
- Verzicht auf geometrische Einflussgrößen mit diskreten Wertebereichen
- Ergänzung relevanter Objekt- und Geometriebezüge
- Zusammenfassung von Bausteinen mit ähnlichem Bewegungsverläufen bzw. Einflussgrößen

Entgegen der vorgeschlagenen technologischen Gliederung von Verrichtungen vorangegangener Arbeiten (Abbildung 10) kam der Autor zu dem Schluss, dass aus Implementierungssicht die Unterscheidung in zwei grundsätzliche Arten von Verrichtungen ausreichend ist. Hierfür wurden die Bezeichnungen »Ele-

haltungen dienen. Der Graph wurde in mehreren Iterationen immer wieder vom Autor und seinen Kollegen bei imk automotive angepasst und verfeinert. Als Ergebnis entstand das Schema, welches in Abbildung 15 dargestellt ist.

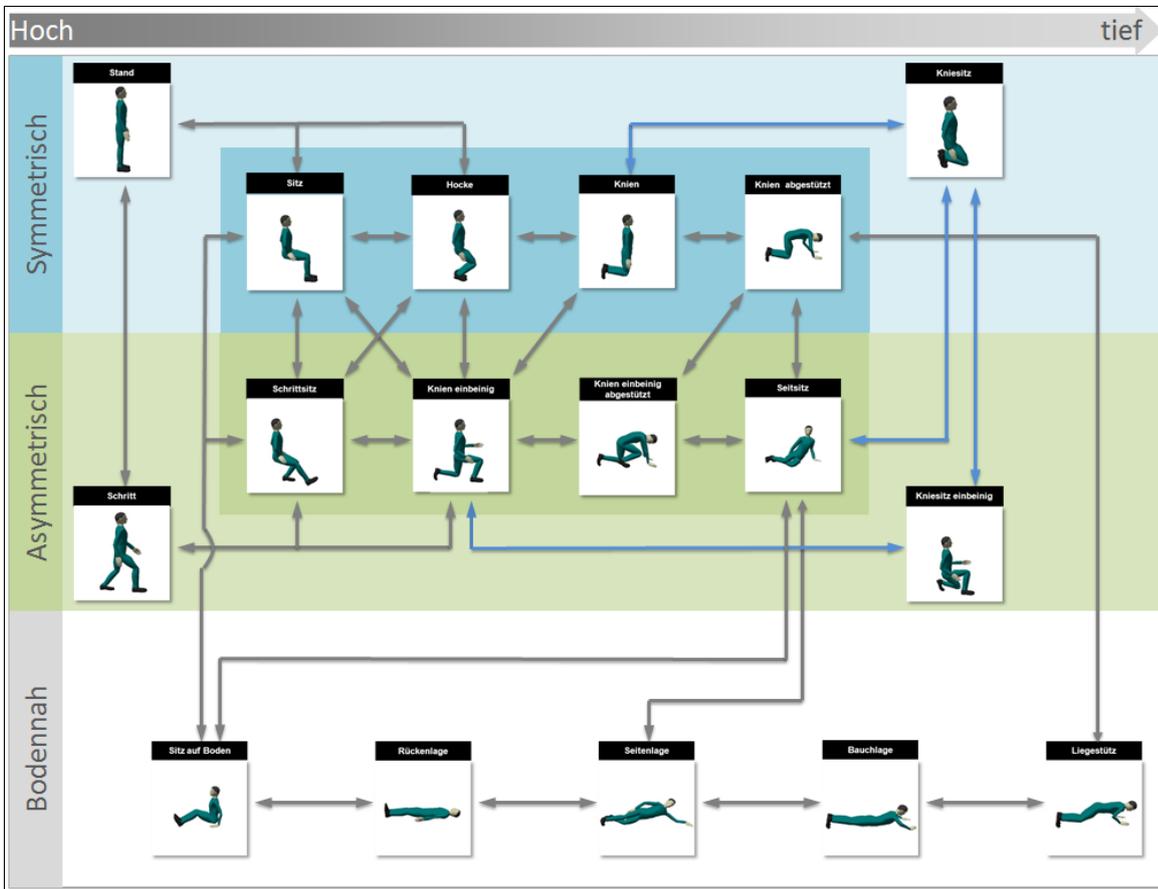


Abbildung 15: Körperhaltungsgraph [58]

Im nächsten Schritt wurden zu den Kanten im Körperhaltungsgraph und den zugehörigen Bewegungen entsprechende »Elementarverrichtungen« konzipiert. Jede Verrichtungsbeschreibung umfasst dabei drei grundlegende Aspekte:

- eine Menge von Eingabeparametern (Einflussgrößen)
- einen Abarbeitungsalgorithmus inklusive definierter Abbruchbedingungen
- die Ermittlung von Anforderungen an das Simulationsmodell

Das Grundprinzip bei der Bestimmung der Eingabeparameter war das Ziel, deren Anzahl zu minimieren. Trotzdem kam der Autor zu dem Schluss, dass die

einzelnen Bewegungen – so elementar sie auch sein mögen – aufgrund der hohen Beweglichkeit des menschlichen Körpers je nach Situation variieren müssen, damit realistische Simulationsergebnisse erzielt werden können. Beispielsweise muss es die Möglichkeit geben, das Aufnehmen eines Objektes aus einer bodennahen Lage sowohl durch vorheriges Abhocken als auch durch das Beugen des Oberkörpers zu realisieren. Die zuletzt genannte Variante sollte angesichts der hohen Beanspruchung der Wirbelsäule zwar im Allgemeinen vermieden werden, ist aber unumgänglich, wenn das Objekt durch geometrische Zwangsbedingungen (z.B. Objekt in Behälter) nicht anderweitig erreicht werden kann. Aus diesem Grund unterteilte der Autor die Parameter in »unbedingt erforderliche Pflichtparameter« und »optionale« Parameter, welche die Bewegungen näher beschreiben sollten. Für Letztere wurden darüber hinaus sinnvolle Standardwerte vordefiniert.

Im Abarbeitungsalgorithmus jeder Verrichtung steckt das eigentliche „Wissen“ des digitalen Menschmodells. Hier werden aus den eingegebenen Parametern und einer vordefinierten Abarbeitungsroutine die an realsprachliche Arbeitsanweisungen angelehnten Tätigkeitsbeschreibungen in die abstrakte Bewegungsbeschreibung für das digitale Menschmodell umgewandelt. Dies geschieht entweder durch den Aufruf untergeordneter Verrichtungen oder durch die direkte Erzeugung abstrakter Zustände. Dabei kann es je nach Verrichtung bestimmte Fälle geben, in denen die Abarbeitung abgebrochen werden muss. Abbruchbedingungen können im einfachsten Fall die fehlende Angabe von Pflichtparametern oder die Verletzung geometrischer Beschränkungen (Reichweiten bzw. Erreichbarkeiten) sein. Auch Plausibilitätsfehler im Ablauf, wie beispielsweise die Verwendung einer Hand, welche nicht mehr „frei“ ist oder das Platzieren eines Objekts, welches zuvor noch nicht aufgenommen worden ist, können zum Abbruch der Simulation führen.

Der Abarbeitungsalgorithmus lässt sich in drei Phasen einteilen, welche sich wiederum auf beliebige Komplexitätsgrade von Verrichtungen anwenden lassen:

- die Vorbereitungsphase, zur Herstellung eines fließenden Überganges zur vorhergehenden Verrichtung (beinhaltet Einstellung von Vorbedingungen wie z.B. Werkzeug aufnehmen, bestimmte Körperhaltung einnehmen)

- die Verrichtungsphase, zur Ausführung des eigentlichen Arbeits-/Aktionsinhaltes (kann wiederholbare / zyklische Inhalte beinhalten wie z.B. „n“ Nägel einschlagen)
- die Nachbereitungsphase, zum Abschluss der Tätigkeit (beinhaltet Bereitstellung von Meta-Informationen / statistischen Informationen für nachfolgende Auswertungen)

Bei der Definition der einzelnen Verrichtungen konnten zudem verrichtungsübergreifende Anforderungen an das zugrundeliegende Simulationsmodell herausgearbeitet werden. Neben elementaren Informationen über den aktuellen Zustand des Menschmodells (Position, Gelenkstellungen, aktuell kontrollierte Objekte, etc.) und den Zustand der Umgebung (Bodenhöhe, Hindernisse) zählen hierzu auch komplexe geometrische Funktionen wie beispielsweise die Ermittlung von Greifpunkten an Objekten oder die Bestimmung von kollisionsfreien Bewegungspfaden. Ein Hauptaugenmerk bei der Konzeption war es, den verrichtungsspezifischen Teil des jeweiligen Algorithmus so überschaubar wie möglich zu halten und möglichst viel Funktionalität auszulagern oder alternativ die Funktionalitäten anderer Verrichtungen zu verwenden. Durch diese algorithmische Verknüpfung der Verrichtungen untereinander wird während der Verrichtungs-Abarbeitung ein Baum untergeordneter Verrichtungsinstanzen erzeugt. Die Blatt-Ebene des Baums wird durch Instanzen von Elementarverrichtungen gebildet, welche die Schnittstelle zum Bewegungsgenerator bedienen. Um die Anzahl der erforderlichen Parameter pro Verrichtung nicht auf die Summe aller Parameter der untergeordneten Verrichtungen wachsen zu lassen, werden so viele Parameter wie möglich innerhalb des Abarbeitungsalgorithmus bestimmt und intern weitergereicht.

Ein Beispiel eines solchen Baums von Verrichtungsinstanzen ist in Abbildung 16 dargestellt. Die Verrichtung `Laufen zum Ziel` verwendet Funktionalitäten des Simulationsmodells, um zunächst einen Laufpfad zum angegebenen Ziel zu berechnen. Dieser intern berechnete Parameter wird anschließend an die hierarchisch untergeordnete Verrichtung `Laufen auf Pfad` übergeben. Es folgen die Abarbeitung der Verrichtungslogik und der erneute Aufruf untergeordneter »Elementarverrichtungen«, bis schließlich auf der untersten Ebene die abstrakten Zustände generiert werden können. Diese werden abschließend an

3 Methodische Vorarbeit

den Bewegungsgenerator übergeben, um die konkrete Bewegung berechnen zu lassen.

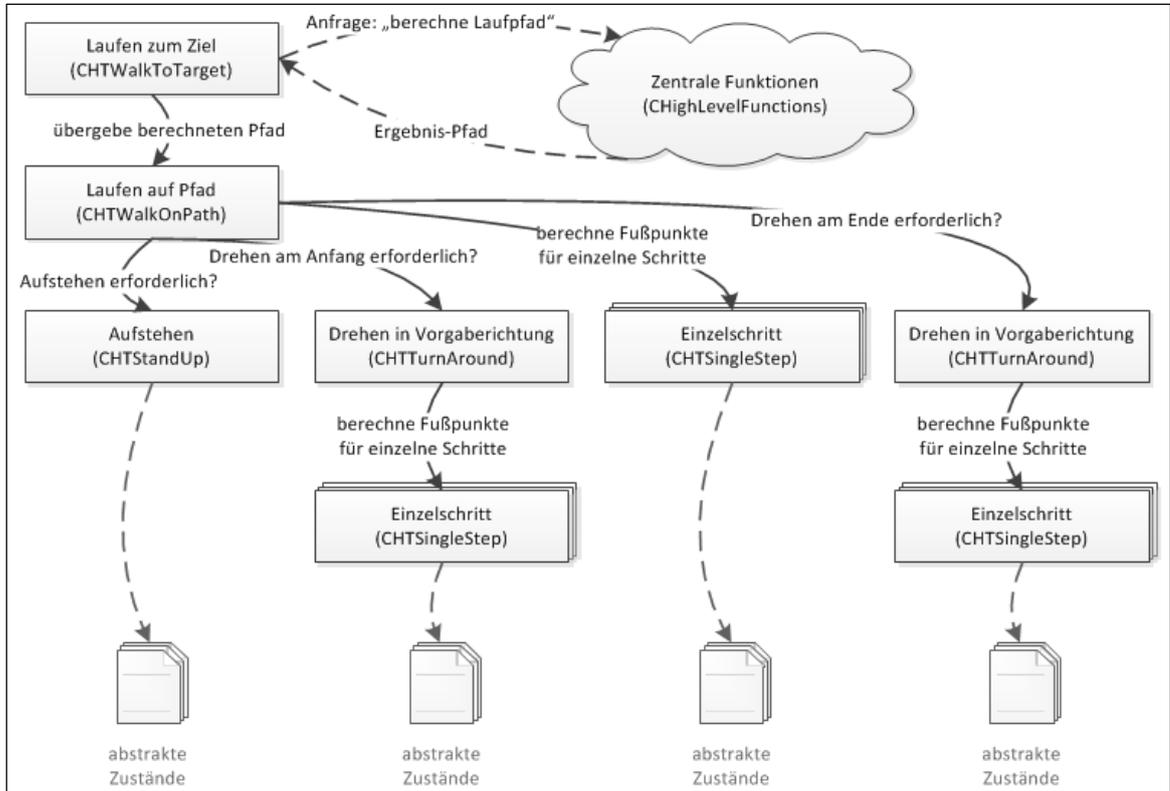


Abbildung 16: Beispiel-Verrichtung »Laufen zum Ziel« [59]

3.5.4 Beispiel-Verrichtung »Objekt(e) aufnehmen«

Um nun zu sehen wie die im vorherigen Kapitel beschriebenen theoretischen Sachverhalte sich in der praktischen Anwendung darstellen, wird nachfolgend die Verrichtung `Objekt(e) aufnehmen` anhand ihrer einzelnen Bestandteile erläutert. Die benannte Verrichtung dient je nach Konfiguration sowohl zur Abbildung der Körperbewegungen als auch der Arm- und Handbewegungen, die zur Kontaktaufnahme mit vorgegebenen Objekten erforderlich sind. Allerdings beschränkt sich die abgebildete Bewegung auf einen einzigen Griff, mit dem die Objekte am Ende an die Hand bzw. beide Hände gebunden werden.

Zunächst ein Blick auf die vom Anwender anzugebenden Pflichtparameter:

- [Objekt-Menge] `Objekt(e)` für die linke Hand
- [Objekt-Menge] `Objekt(e)` für die rechte Hand
- [boolescher Wert] `Automatisches Laufen`

Durch die Angabe der Objekte für die jeweilige Hand kann der Benutzer nicht nur vorgeben, ob ein oder mehrere Objekt(e) vom digitalen Menschmodell aufgenommen werden soll(en), sondern auch, ob das oder die Objekt(e) ein- oder beidhändig gehandhabt werden müssen. Natürlich könnte das Menschmodell die Entscheidung, mit welcher Hand bzw. ob ein Objekt ein- oder beidhändig aufgenommen wird, gegebenenfalls selbst treffen. Diese Funktionalität ist derzeit jedoch für eine spätere Ausbaustufe des Systems geplant, da hierzu die Auswertung der konkreten Objektgeometrie notwendig wäre, um detaillierte Informationen über dessen Gestalt und die daraus resultierende Handhabung zu erhalten. Der Parameter `Automatisches Laufen` dient als Möglichkeit, um dem Menschmodell vorzugeben, ob es sich zum Zweck des Aufnehmens des oder der Zielobjekt(e) von seiner derzeitigen Position entfernen darf oder nicht.

Ein Satz von optionalen Parametern ergänzt die Konfiguration der Verrichtung:

- [Constraint] `Greifpunkt für linke Hand`
- [Constraint] `Greifpunkt für rechte Hand`
- [Auswahltyp] `Oberkörperbewegung`
- [boolescher Wert] `Beinbewegung`

Während die Pflichtparameter die Antworten auf die geometrisch relevanten Fragen „Was?“ und „Wo?“ liefern, dienen die optionalen Parameter zur näheren

Erläuterung der Art der Ausführung der Tätigkeit („Wie?“). Auch diese Antworten könnten in einer späteren Ausbaustufe des Systems automatisch beantwortet werden und die Eingabe dieser Parameter dementsprechend entfallen. Aufgrund der Tatsache, dass das System jedoch von Grund auf neu entwickelt worden ist und dementsprechend noch nicht jedes Detail in der Wissensbasis Berücksichtigung finden konnte, hat der Autor die optionale Eingabe der Detailinformationen vorgesehen. Hierzu gehört auch der geometrische Ort, an dem der Kontakt zu den entsprechenden Objekten mit den Händen hergestellt werden soll – die sogenannten Greifpunkte. Darüber hinaus hat der Anwender die Möglichkeit, den Einsatz des Oberkörpers zu erlauben, zu verbieten oder zu erzwingen. Abschließend kann noch die Beinbewegung des Menschmodells am Aufnahme-Ort vom Benutzer beeinflusst werden. Auch wenn die genannten optionalen Parameter in der derzeitigen Version des Simulationssystems nicht ganz eliminiert werden konnten, so hat der Autor zumindest sinnvolle Standardwerte für Letztere hinterlegt.

Der Abarbeitungs-Algorithmus der Verrichtung (Abbildung 17) beginnt zunächst mit einer allgemeinen Überprüfung der vom Benutzer angegebenen Parameter. Getreu dem Prinzip, keiner Eingabe von außen „zu trauen“, wird jeder Parameter mehrfach auf unterschiedliche Aspekte hin überprüft. Während im späteren Verlauf der Bearbeitung primär die geometrischen Bedingungen geprüft werden, ist am Beginn des Algorithmus erst einmal das grundsätzliche Vorhandensein und korrekte Format der Eingaben relevant. Nach der ersten Grobprüfung folgen genauere Tests der angegebenen Objekte für die linke und rechte Hand. Dabei wird die für die weitere Abarbeitung relevante „Art der Handhabung“ ermittelt, welche sich in ein- oder beidhändiges Handhaben unterteilen lässt. Beim einhändigen Handhaben gibt es dann noch die Fallunterscheidung in „linke Hand“, „rechte Hand“ und „beidhändig einzeln“. Während der Ermittlung der Handhabung werden wiederum mehrere Plausibilitäts-Checks durchgeführt - diese prüfen ob:

- die benötigte (aktive) Hand bzw. beide Hände noch frei sind und die konfigurierte Handhabung somit überhaupt möglich ist
- die benannten Objekte mit einem Griff aufgenommen werden können (räumliche Distanz der Objekt(e) untereinander in Beziehung zur Anthropometrie der Hand)

3 Methodische Vorarbeit

- ein Objekt zu groß oder zu klein für die angegebene Art der Handhabung ist (siehe Kapitel 4.6.3)

Für den Fall, dass eine oder mehrere der vorangegangenen Überprüfungen ein negatives Ergebnis liefern, wird die weitere Abarbeitung mit einer Fehlermeldung abgebrochen.

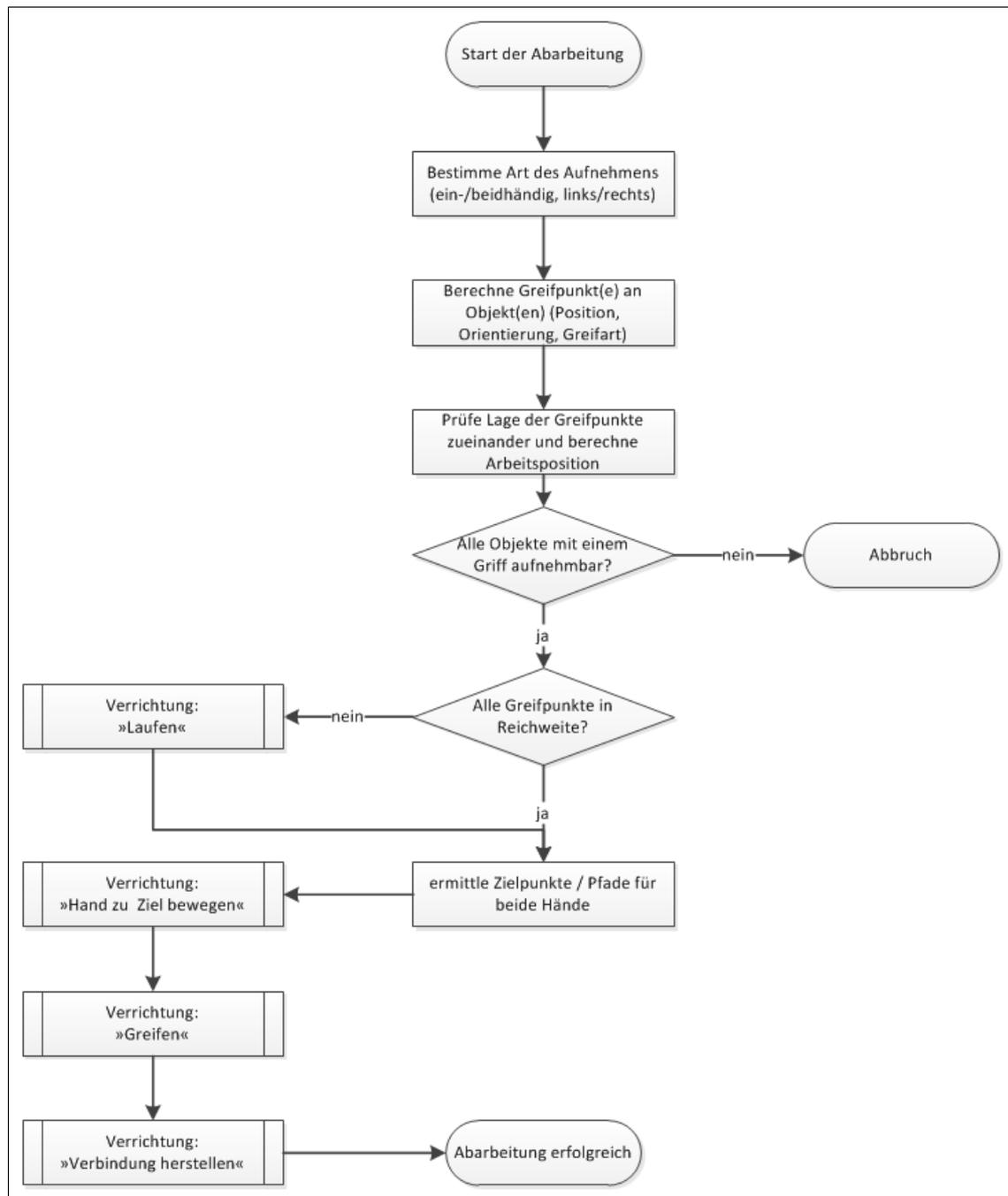


Abbildung 17: Algorithmus der Verrichtung »Objekt(e) aufnehmen«

Nach positiver Prüfung der Parameter und für den Fall, dass der Parameter Automatisches Laufen gesetzt wurde, folgt nun die Berechnung einer möglichst guten »Arbeitsposition« um das Aufnehmen ausführen zu können. „Gut“ bezieht sich dabei zum einen auf einen möglichst kurzen Weg und falls ein größeres, sperriges Objekt beidhändig aufgenommen werden soll, auf eine möglichst günstige Position relativ dazu, welche die Erreichbarkeit für beide Hände sicherstellt.

Wenn die zu benutzenden »Greifpunkte« nicht direkt vom Benutzer vorgegeben worden sind (Parameter: Greifpunkt für linke / rechte Hand), folgt anschließend die Ermittlung selbiger. Hierbei wird in Abhängigkeit der Ausrichtung des Menschmodells zu den Objekten, der zu verwendenden Hand und der Art und Anzahl der am Objekt automatisch berechneten oder manuell definierten »Greifpunkte« eine entsprechende Auswahl getroffen. Abschließend wird pro »Greifpunkt« noch eine Griff-Art und ein entsprechender Griffprozentsatz, welcher den Grad des Handschlusses repräsentiert, berechnet. Letztere dienen der Simulation des „Zugreifens“. (siehe Kapitel 3.6)

Es sind nun alle relevanten Vorbedingungen hinsichtlich der Plausibilität des Ablaufs und die geometrischen Abhängigkeiten überprüft worden. Was folgt, ist die Ansteuerung der untergeordneten Verrichtungen, welche in ihrer Abarbeitung die Informationen für die nachfolgende Bewegungsgenerierung im Format der abstrakten Bewegungsbeschreibung erzeugen. Dies sind im Einzelnen die untergeordneten Verrichtungen:

- Hand zu Ziel bewegen für die Hand-/Armbewegungen zu den »Greifpunkten« hin
- Greifen für die Fingerbewegung beim Schließen der Hand
- Verbindung herstellen für den Status-Wechsel des Anhängens der Objekte an die entsprechende Hand oder beide Hände

3.5.5 Zusammenführung der Erkenntnisse

Für jede einzelne Verrichtung, welche innerhalb des Simulationssystems entwickelt worden ist, wurde ein Abarbeitungsschema nach dem Vorbild des im vorangegangenen Kapitel erläuterten Beispiels konzipiert und umgesetzt. Als Ergebnis entstand eine Bibliothek mit derzeit 26 realisierten und noch einmal ebenso vielen konzipierten Verrichtungen. Ergänzt wird die Verrichtungsbibliothek um funktionale Anforderungen an das Simulationsmodell, welche für jeweils mehrere Verrichtungen relevant sind. Um einen Überblick über die somit erschaffene prozedurale Wissensbasis des digitalen Menschmodells zu erhalten und sowohl die Verrichtungen als auch die Funktionalitäten des Simulationsmodells in ihrer Einordnung und in ihren gegenseitigen Beziehungen darzustellen, entwickelte der Autor ein Schaubild in Form eines sogenannten Morphologischen Kastens (Kreativitäts-Technik). Dieses Schaubild gibt nicht nur einen grundsätzlichen Überblick über die Wissensbasis; es wurde darüber hinaus auch als Ideengeber und Aufgabenliste für die nachfolgende Implementierung verwendet.

Im linken Bereich sind die Bestandteile der Verrichtungsbibliothek dargestellt. Jedes Kästchen entspricht dabei einer einzelnen Verrichtung. Die vertikale Anordnung der Verrichtungen gibt einen Hinweis auf deren Komplexität, welche durch den Umfang an berechneter Simulationszeit und Anzahl der Verzweigungen im jeweils zugrundeliegenden Algorithmus definiert ist. Aus der vertikalen Anordnung lassen sich neben der Komplexität teilweise auch die direkten Abhängigkeiten zwischen den Verrichtungen erkennen. Der Autor hat aus Gründen der Übersichtlichkeit bewusst darauf verzichtet, jede einzelne Abhängigkeit (beispielsweise per Pfeil) darzustellen. Stattdessen ist die Anordnung so zu interpretieren, dass Elemente stets nur von anderen Verrichtungen verwendet werden, die in ihrer vertikalen Position neben oder über ersteren angeordnet sind. Ein Beispiel hierfür ist die Verrichtung `Laufen zu Ziel`. Diese verwendet die vertikal untergeordneten Elemente `Laufen auf Pfad` und `Drehen am Ort`. Im gleichen Zug kann die genannte Verrichtung jedoch von allen anderen vertikal höher befindlichen Verrichtungen verwendet werden. Neben den Abhängigkeiten und der Komplexität wurden die Verrichtungen auch hinsichtlich inhaltlicher Verwandtschaft gruppiert. Die Gruppierungen wurden entsprechend benannt und per grau schraffiertem Hintergrund kenntlich gemacht. Die-

3 Methodische Vorarbeit

schon durch manuelle Kombination der entsprechenden Verrichtungen simuliert werden können.

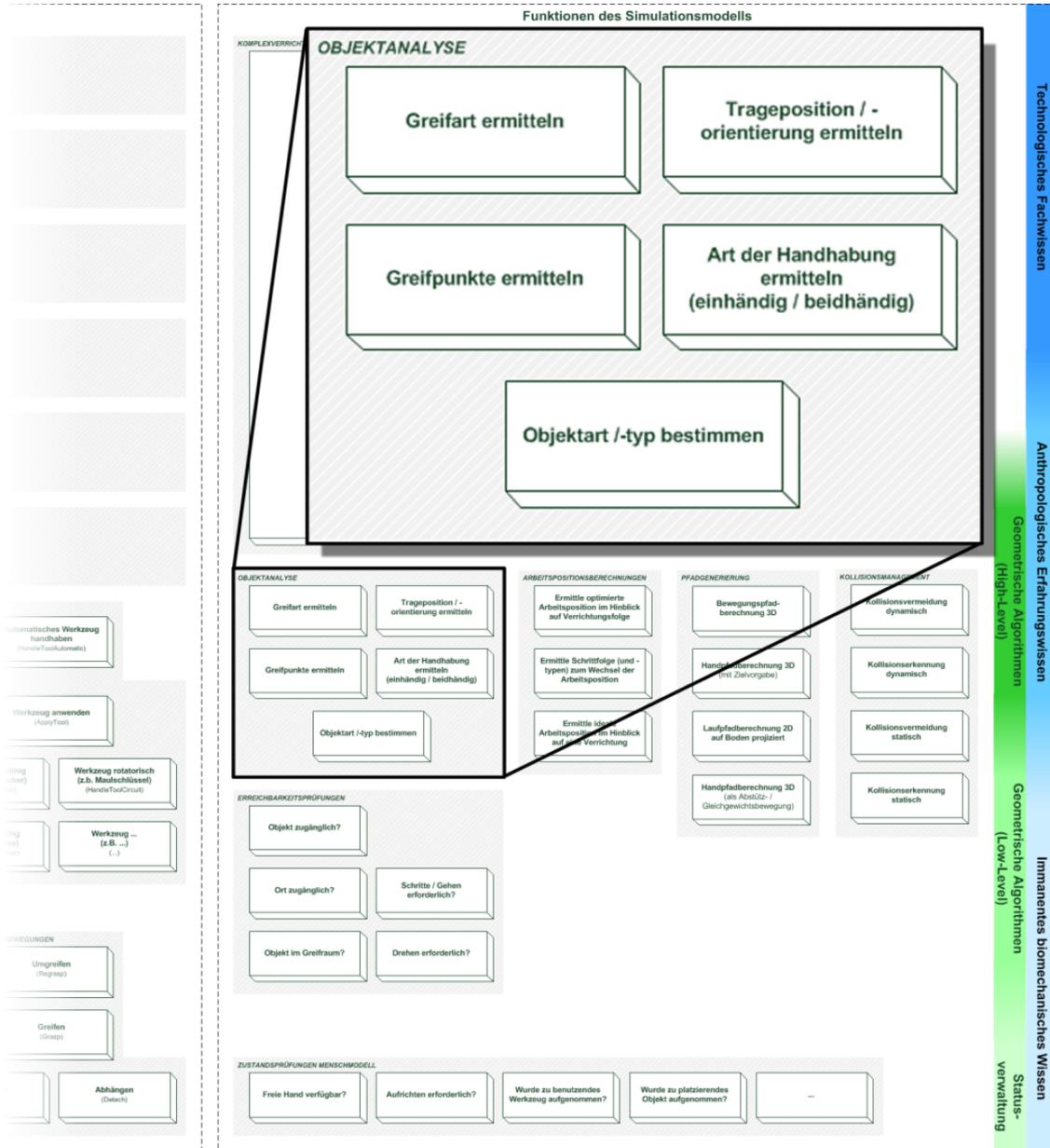


Abbildung 19: „Funktionen des Simulationsmodells“ (Morph. Kasten)

Auf der rechten Seite des morphologischen Kastens sind – ähnlich der Übersicht der Verrichtungen – die relevanten Funktionen des Simulationsmodells dargestellt. Auch diese wurden anhand ihrer algorithmischen Komplexität vertikal angeordnet. Während die Komplexität der Verrichtungen primär an der Dauer und Art des abgebildeten Tätigkeitsinhalts gemessen werden kann, wird die Komplexität der Funktionen des Simulationsmodells durch deren tatsächlichen Ressourcenverbrauch an Laufzeit und Speicherplatz bestimmt. Die Bandbreite reicht hierbei von einfachen Statusabfragen auf der untersten Ebene, über einfache geometrische Berechnungen bis hin zu aufwändigen Algorithmen für Pfadermittlung und Kollisionsvermeidung. Wie bereits bei den Verrichtungen gliederte der Autor die ermittelten Elemente in einer zusätzlichen horizontalen Gliederungsebene. Diese fasst die Funktionalitäten nach Themenbereichen zusammen und reicht vom Themenbereich »Zustandsprüfungen für das Menschmodell« über »Erreichbarkeitsprüfungen« und »Arbeitspositions-Ermittlung« bis hin zur »Pfadgenerierung« und zum »Kollisionsmanagement«.

Auch die Funktionen des Simulationsmodells bauen zum Teil aufeinander auf und Algorithmen höherer Ebenen nutzen die (Teil-)Ergebnisse von entsprechenden Algorithmen darunter befindlicher Ebenen. Ein Beispiel hierfür stellen die Elemente der Themengruppe »Objekt-analyse« dar. Als Basisalgorithmus dieser Gruppe dient die Funktion »Objektart/-typ bestimmen«, die Objekte entsprechend ihrer geometrischen Gestalt und Größe klassifiziert. Anschließend können die gewonnenen Erkenntnisse in den Algorithmen »Greifpunkte ermitteln« und »Art der Handhabung ermitteln« weiterverwendet werden. Daraus lassen sich in der Folge wiederum Erkenntnisse für die Funktionen »Greifart ermitteln« und »Trageposition/-orientierung ermitteln« gewinnen.

3.6 Bewegungsgenerierung

3.6.1 Von der abstrakten zur konkreten Bewegungsbeschreibung

Eines der Hauptziele bei der Entwicklung des Simulationssystems war die Separierung der Verarbeitung der eingegebenen Tätigkeitsbeschreibungen von deren Anwendung auf ein konkretes digitales Menschmodell. Die Gründe für diese Zielvorgabe lagen zum einen in der Gewährleistung der Übertragbarkeit der Methodik auf beliebige 3D-Simulations- und Planungssysteme. Darüber hinaus sollten die Ergebnisse der Absichtskomponente unabhängig von der Struktur und vor allem unabhängig von den geometrischen Abmessungen der damit angesteuerten Menschmodelle sein. Die letztgenannte Anforderung resultiert dabei hauptsächlich aus dem planerischen Ziel, anhand der 3D-Simulation gleicher Tätigkeitsabfolgen mit unterschiedlich großen Menschmodellen Informationen über die ergonomischen Gegebenheiten der untersuchten Arbeitsplätze zu gewinnen.

Bei der Identifikation der relevanten Systemkomponenten (siehe Abschnitt 3.3) wurde bereits die Unterscheidung in die Absichtskomponente und den Bewegungsgenerator vorgenommen. Für die Datenübergabe wurde eine Schnittstelle mit der Bezeichnung *abstrakte Bewegungsbeschreibung* vorgesehen. Um eine lose Koppelung zwischen der Bewegungsbeschreibung und dem konkreten Menschmodell zu erreichen, durften sich die zu übergebenden Daten lediglich auf allgemeingültige Grundstrukturen digitaler Menschmodelle beziehen und sollten dabei modellabhängige Spezifika ausblenden. Erreicht wurde dies durch die konsequente Verwendung von Bewegungszielvorgaben für die Endeffektoren des Modells und ergänzende Metainformationen für auszuführende Bewegungsmuster einzelner Körperregionen. Die Bewegungsziele werden dabei als sogenannte »Constraints« übergeben, welche jeweils aus Positions- und Orientierungsangaben bestehen. Dabei geht der Autor davon aus, dass jedes Menschmodell über die grundlegenden Endeffektoren des menschlichen Körpers (Füße, Hände, Kopf) verfügt, wohingegen die kinematische Strukturen des Rumpf- und Hüftbereichs mitunter stark variieren können. (vgl. Kapitel 3.1) Darüber hinaus sind die Metainformationen erforderlich, um beispielsweise eine eindeutige Unterscheidung zwischen Abhocken und Oberkörperbeugung beim Aufnehmen eines in Bodennähe befindlichen Objektes zu ermöglichen. Auch für die Ansteuerung der Hände bzw. der einzelnen Fingersegmente erwies sich

die Verwendung von vordefinierten Bewegungsmustern angesichts der zu erwartenden Anzahl der zu bewegenden Fingersegmente als sinnvolle Abstraktionsebene.

Die abstrakte Bewegungsbeschreibung besteht somit aus einer Folge von abstrakten Zuständen. Der dabei mindestens erforderliche Informationsgehalt pro Zustand wurde vom Autor wie folgt ermittelt:

- [Liste von Constraints] für relevante Körperteile (zunächst vorwiegend für die End-effektoren: Kopf, Hände, Füße später evtl. auch für Knie / Ellenbogen etc.)
- [Art der Oberkörperbewegung] zur Definition / Einschränkung der Bewegung des Oberkörpers
- [Art der Beinbewegung] zur Definition / Einschränkung der Bewegung der Beine
- [Art der Handbewegung] zur Definition / Einschränkung der Bewegung der Hände

Die Aufgabe der Komponente Bewegungsgenerator ist es nun, diese abstrakte Bewegungsbeschreibung auf ein digitales Menschmodell anzuwenden und sie somit in eine konkrete Bewegungsbeschreibung zu überführen. Die Visualisierung von Bewegungen basiert dabei auf einer Folge von Zuständen, welche aufeinanderfolgend wiedergegeben werden. Diese Einzelzustände beschreiben einzelne Körperhaltungen, sogenannte »Posen«, des digitalen Menschmodells. Ein solcher Simulationszustand wird allgemein durch zeitliche und geometrische Parameter charakterisiert. Anhand eines einzelnen Zeitwerts wird die Dauer angegeben, die zum Erreichen des Zustands erforderlich ist. Die Art und Anzahl der geometrischen Parameter variiert je nach Komplexität des anzusteuernenden Objekts. Im einfachen Fall eines starren Einzelobjekts genügt die Verwaltung der geometrischen Position und Orientierung des Objekts im Zielzustand. Handelt es sich um ein strukturiertes Objekt wie das digitale Menschmodell, muss zwangsläufig auch für jedes sich bewegende Unterobjekt eine entsprechende geometrische Zustandsbeschreibung verwaltet werden.

Jeder Simulationszustand baut auf dem vorangegangenen Zustand auf, wobei der Übergang zwischen den Zuständen je nach Erfordernis über beliebig viele interpolierte Zwischenzustände erfolgen kann. Da jeder Zustand also einen Vorgänger besitzt, ist das Vorhandensein eines ersten sogenannten »Initialzu-

stands« erforderlich. (siehe Kapitel 3.4.1) Der Initialzustand ist vor dem Beginn der Simulation festzulegen und wird persistent verwaltet, da jeder Simulationslauf die Zustände der bewegten Objekte und gegebenenfalls sogar deren hierarchische Beziehungen untereinander verändert und nur so jederzeit eine Rückkehr zum Ausgangspunkt sichergestellt werden kann.

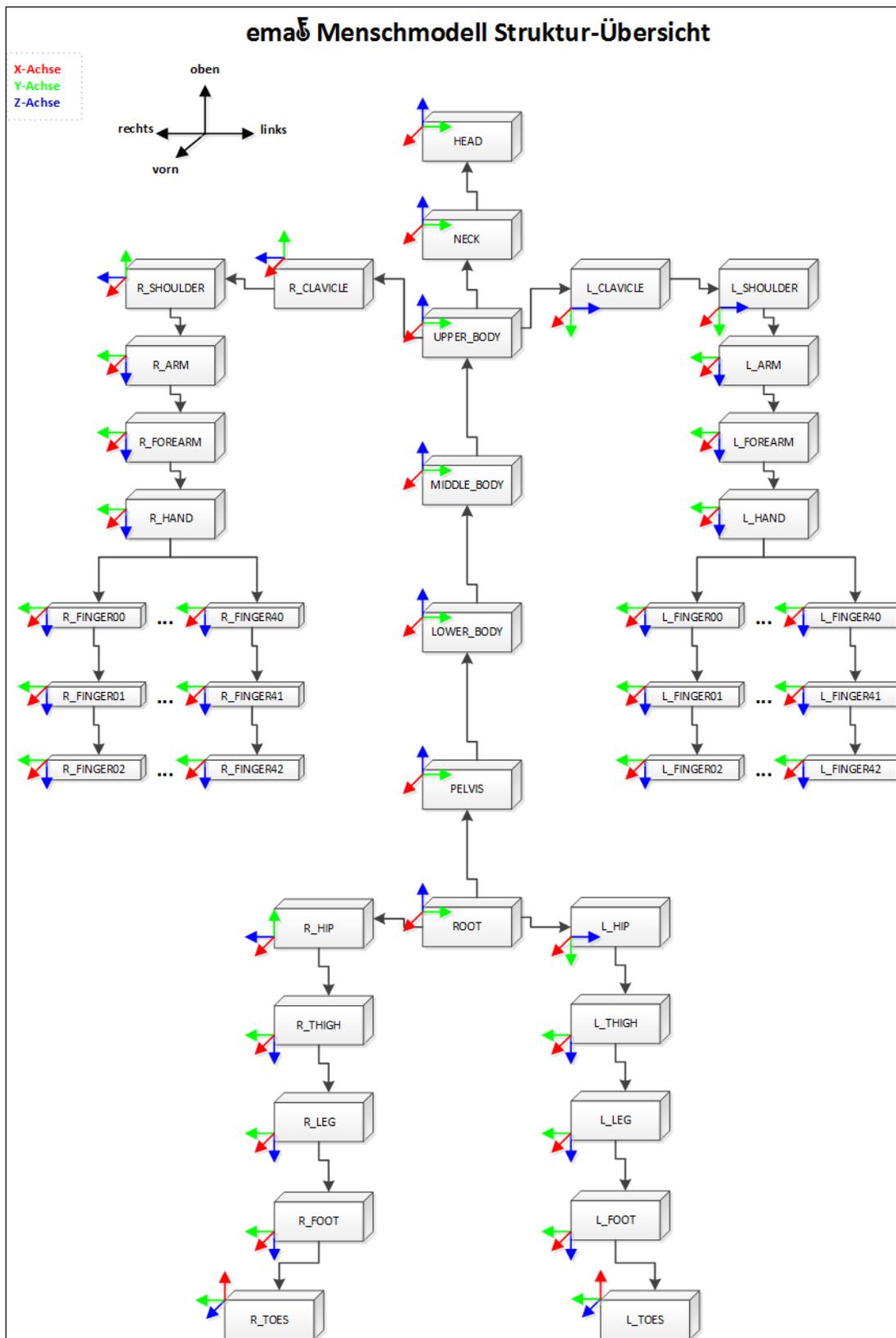
Auch wenn die tatsächliche Berechnung der einzelnen Körperhaltungen nicht zum unmittelbaren Kern der vorliegenden Arbeit gehörte, war sie dennoch ein unverzichtbares Mittel, um die erarbeiteten Konzepte der Absichtskomponente zu verifizieren. Denn erst nach der Anwendung der Bewegungsbeschreibung auf ein digitales Menschmodell war eine Bewertung der berechneten Ergebnisse, der vorgelagerten Komponenten, möglich. Zu diesem Zweck beschäftigte sich der Autor zunächst mit den bekannten Ansätzen zur Berechnung von Körperhaltungen. In der Literatur werden die Methoden in zwei grundlegende Klassen unterteilt – die datengestützten und die nicht-datengestützten Berechnungsansätze. [60, pp. 9-2] Die Verfahren des erstgenannten Typs erfordern jeweils eine Datenbasis (zumeist in Form einer zugrundeliegenden Datenbank), aus der mit den eingegebenen Suchkriterien eine passende Körperhaltung ermittelt wird. Die größte Einschränkung stellt hierbei naturgemäß der Inhalt der Datenbasis dar. Es können nur Körperhaltungen ermittelt werden, die in ähnlicher bzw. nahezu identischer Form in der Datenbasis enthalten sind. Dabei besteht zwar die Möglichkeit, „neue“ Körperhaltungen per Interpolation vorhandener Elemente zu erzeugen. Ein rein datengestützter Ansatz wird jedoch niemals so flexibel sein, dass er vollkommen neuartige und somit nicht in der Datenbasis vorhandene Körperhaltungen erzeugen kann. Mit Blick auf die zu erwartende hohe Varianz der Körperhaltungen bei den zu simulierenden menschlichen Tätigkeiten sah der Autor in dem zuletzt genannten Aspekt den schwerwiegendsten Nachteil für den alleinigen Einsatz einer datengestützten Methodik. Darüber hinaus stellt der Aufbau einer adäquaten Datenbasis und darauf operierender Algorithmen eine hochgradig komplexe und zeitintensive Aufgabe für sich dar. Vor diesem Hintergrund kam zur Konzeptverifizierung zunächst eine nicht-datenbasierte Körperhaltungsgenerierungs-Methode zum Einsatz, welche der Autor in den nachfolgenden Kapiteln näher erläutern wird.

3.6.2 Verwendete Menschmodell-Struktur

Als Ausgangspunkt der Untersuchungen diente die allgemeine Struktur eines digitalen Menschmodells. Da bereits zum Zeitpunkt der methodischen Untersuchungen das Menschmodell von »DELMIA V5« als potentielles „Ausgabemedium“ der Bewegungen in Betracht kam, wählte der Autor den »Human V5« als ersten Untersuchungsgegenstand. Die dabei ermittelte Übersicht der Körperteile des »Human V5« ist in Abbildung 20 dargestellt. Ein digitales Menschmodell ist ein durch Körperteile strukturiertes komplexes Objekt. Es stellt für sich betrachtet also einen Teil-Szenengraph dar, dessen Wurzel durch das Körperteil »ROOT« repräsentiert wird. Im Bezug zur Visualisierung des Modells (geometrische Hülle) befindet sich der Wurzel-Körperteil in der Mitte der Hüfte. Von dort aus verzweigen in Teil-Baumstrukturen die Beine und die Wirbelsäule des Modells. Das oberste Wirbelsäulensegment gliedert sich schließlich in die Arme und den Hals-Kopf-Bereich auf.

In der erarbeiteten Abbildung 20 lassen sich außerdem auch Regeln über die Lage der Koordinatensysteme der einzelnen Körperteile erkennen. So dient die z-Achse bei jedem Körperteil zur Definition der Längsachse, was bedeutet, dass die Länge eines Körperteils bei dem betrachteten Menschmodell immer entlang der z-Achse gemessen wird. Gleichzeitig ist mit ihr auch definiert, wo „Kind-Körperteile“ im Koordinatensystem ihrer „Eltern-Körperteile“ zu finden sind, nämlich ebenfalls entlang der z-Achse. Die Ausrichtung der x-Achsen folgt dem Prinzip „vom Menschmodell ausgehend nach vorn“, während die y-Achsen jeweils die rechtshändigen Koordinatensysteme vervollständigen.

Das betrachtete Menschmodell verfügt über eine detailliert ausmodellerte Wirbelsäule mit 18 beweglichen Wirbelsegmenten. Im Vergleich mit dem realen Vorbild, das über 24 bewegliche Wirbel verfügt [61, p. 3ff], wurde die Halswirbelsäule etwas vereinfacht. In der Übersicht zur Struktur des Menschmodells tauchen die 18 Wirbel jedoch nicht auf. Der Grund hierfür liegt in der Tatsache begründet, dass die Oberkörperhaltung des »Human V5« sowohl über die graphische Benutzeroberfläche als auch über die Programmierschnittstellen nur vereinfacht durch die beiden Bereiche der Brust- und Lendenwirbelsäule angesteuert werden kann. Die Umrechnung auf die einzelnen Wirbel erfolgt systemintern und kann damit nicht von außen beeinflusst werden.



Das Menschmodell beinhaltet darüber hinaus eine Reihe von »Hilfs-Körperteilen«, welche in die Szenengraph-Hierarchie eingeordnet sind. Diese »Hilfs-Körperteile« dienen innerhalb des Menschmodells dazu, statische Offsets bzw. Hilfspositionen mitzuführen, die während der Simulation für Untersuchungen herangezogen werden können (z.B.: Positionen von Ferse, Ballen, Augen, etc.). Ergänzt werden die »Hilfs-Körperteile« durch weitere vom Autor eingeführte »Hilfs-Zielpunkte«. Während die »Hilfs-Körperteile« ihre Relativposition innerhalb des Menschmodells während der Simulation beibehalten, können die »Hilfs-Zielpunkte« ihre Position und Ausrichtung je nach Situation verändern (z.B.: Hand-Offsets).

Als weitere Ergänzung zur Menschmodell-Struktur hat der Autor eine Reihe von internen Hilfsabmaßen definiert (Abbildung 21), welche sich zumeist aus aggregierten Abmaßen einzelner Körperteile ergeben und während der Simulation bei diversen komplexeren Berechnungen Verwendung finden. Da sich diese Abmessungen während der Simulation nicht ändern, könnten sie beispielsweise in einem Initialisierungsschritt berechnet und zur Verfügung gestellt werden (z.B.: gesamte Armlänge, gesamte Beinlänge, Schulterbreite, etc.).

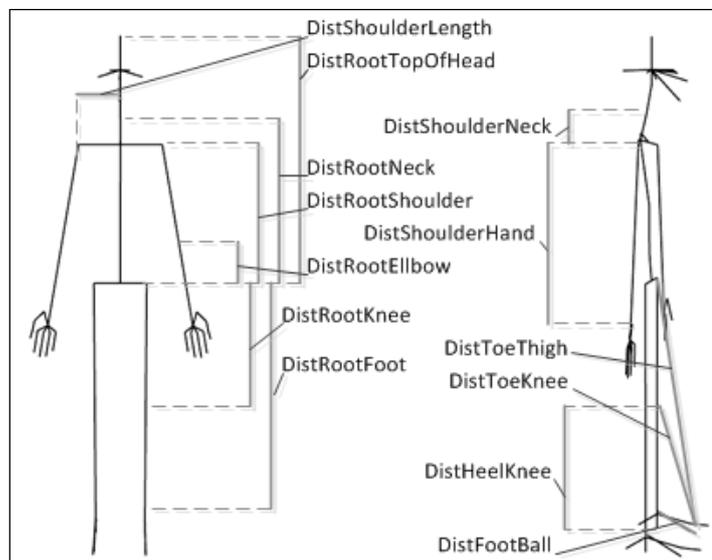


Abbildung 21: Menschmodell Hilfsabmessungen

Um die Universalität der gewählten Schnittstellen sicherzustellen, wurden die gewonnen Erkenntnisse auch im Kontext anderer digitaler Menschmodelle überprüft. Bei der Untersuchung der Modelle »Dynamicus«, »Ramsis«

(Abbildung 22) und »Santos« [45, pp. 29-5 / Figure 29.5] gelangte der Autor zu der Feststellung, dass sich der innere Aufbau der Menschmodelle aufgrund der gemeinsamen realen Vorlage grundsätzlich ähnelt. Insbesondere die Strukturierung der Gliedmaßen stimmt weitestgehend überein. Größere Abweichungen treten hauptsächlich beim Aufbau des Rumpfes und hierbei insbesondere bei der Anzahl der Segmente auf, welche die Wirbelsäule abbilden. Auch die Modellierung der Handwurzelknochen, des Beckens und die Anbindung des Schulterbereichs unterscheiden sich. Dennoch traten bei keinem der betrachteten Menschmodelle strukturelle Probleme auf, die die Anwendung der erarbeiteten Methodik zur Bewegungsgenerierung behindert hätten. Ganz im Gegenteil – die vergleichende Untersuchung hat die Korrektheit des Ansatzes der Unterscheidung zwischen abstrakter und konkreter Bewegungsbeschreibung noch einmal unterstrichen. Die definierten Schnittstellen erlauben somit theoretisch die Ankoppelung der erarbeiteten Methodik an nahezu jedes System, welches über ein eigenes digitales Menschmodell verfügt.

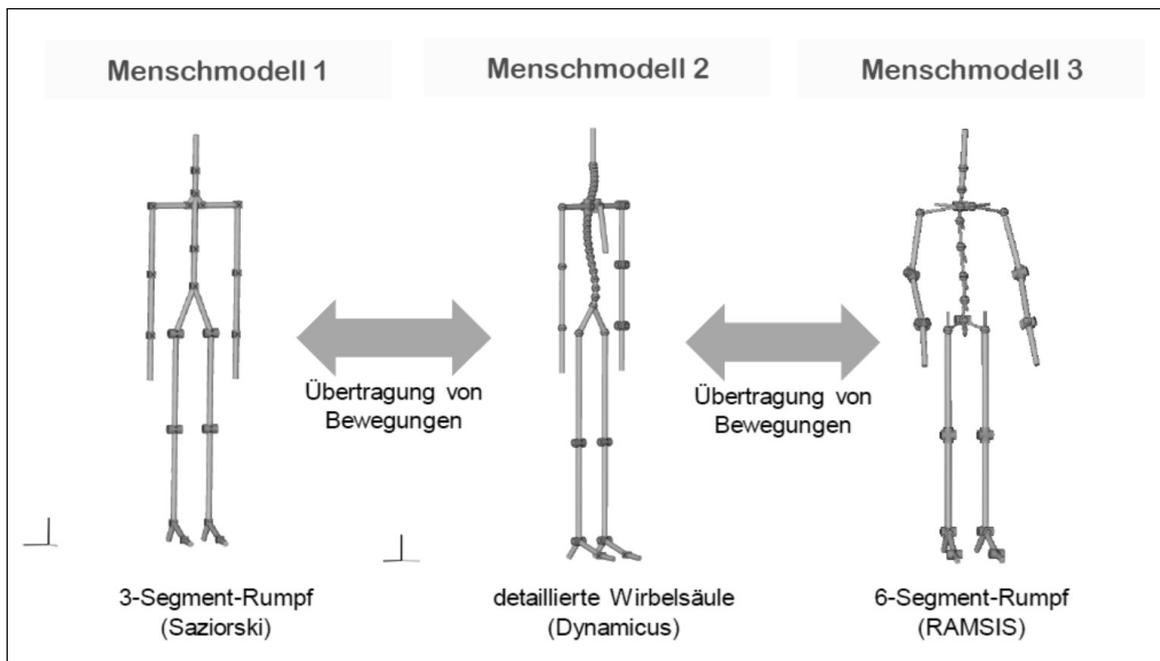


Abbildung 22: Topologien digitaler Menschmodelle [62, p. 7]

3.6.3 Überblick zur Körperhaltungsgenerierung

Bereits bei der Konzeption der Verrichtungsbibliothek wurden Untersuchungen hinsichtlich aller relevanten Körperhaltungen durchgeführt, die das Menschmodell einnehmen kann. (Abbildung 15) Die ermittelten Körperhaltungen werden primär durch die entsprechende Beinsetzung benannt und charakterisiert. Zwar tragen die Oberkörper- und Armhaltungen ebenfalls ihren Anteil zum Gesamterscheinungsbild des Menschmodells in der entsprechenden Körperhaltung bei. Dennoch bewirkt eine alleinige Veränderung der Armstellung keine Änderung der entsprechenden Grundkörperhaltung. Diese Tatsache führte den Autor zu der Idee, den Körper des Menschmodells in einzelne physiologische Funktionsgruppen zu unterteilen, um das komplexe Gesamtproblem der Körperhaltungsgenerierung nach dem »Divide-and-Conquer«-Prinzip in mehrere, nun weniger komplexe Teilprobleme aufzulösen. Als sinnvolle Untergliederung erwies sich dabei die Einteilung des Körpers des digitalen Menschmodells in folgende fünf Funktionsgruppen:

- Gruppe »Beine« (Hüfte, Oberschenkel, Unterschenkel, Füße, Zehen)
- Gruppe »Oberkörper« (Segmente der Wirbelsäule)
- Gruppe »Arme« (Oberarme, Unterarme, Hände)
- Gruppe »Hände« (alle Finger-Segmente)
- Gruppe »Kopf« (Hals, Kopf)

Die genannten „Körperteil-Gruppen“ haben gemein, dass jede Gruppe für sich Zustände einnehmen kann, die sich weitestgehend unabhängig von den Zuständen der anderen „Körperteil-Gruppen“ definieren lassen. So ist beispielsweise für die Definition einer Handstellung (Gruppe »Hände«) ohne Bedeutung, in welchem Zustand sich gerade die Gruppe »Beine« befindet. Dennoch besteht aufgrund der internen „Körperteil-Hierarchie“ die Notwendigkeit, bei der Teil-Posen-Generierung eine bestimmte Reihenfolge einzuhalten, nämlich bei den hierarchisch übergeordneten Gruppen beginnend hin zu den hierarchisch untergeordneten Gruppen. Die Berechnung der Armstellung kann beispielsweise erst erfolgen, wenn die der Oberkörperstellung und die der Hüftposition abgeschlossen sind. Bei Anwendung der umgekehrten Reihenfolge wäre eine zuvor für ein bestimmtes Ziel berechnete Armstellung mit nachfolgender Verschiebung der Hüfte nicht mehr korrekt.

In Abbildung 23 wird ein Überblick über die konzipierte Verarbeitung der Tätigkeitsbeschreibung und ihrer Zwischenzustände auf dem Weg hin zu konkreten (Simulations-)Zuständen gegeben. Dabei wurden im Hinblick auf die Realisierung bereits mögliche Ansätze objektorientierter Klassen-Strukturen vorgesehen. Zu Beginn eines Simulationslaufs wird zunächst die Instanz einer Simulationsprojekt-Klasse angelegt. Diese dient in der Folge als übergeordnetes Element, welches global benötigte Funktionalitäten (Logging, Konfiguration, etc.) zur Verfügung stellt. Unterhalb der Projekt-Klassen-Instanz werden in zwei nachfolgenden Schritten sowohl der Szenengraph in seinem Initialzustand als auch die Verrichtungs-Klassen initialisiert und miteinander verknüpft. Ergänzt werden die Objekt- und Tätigkeitsstrukturen schließlich durch die Initialisierung eines Objekts, welches übergeordnete Funktionen des Simulations-Modells als Dienste zur Verfügung stellt. Diese zentralen Funktionen können bei Bedarf von jeder Verrichtung aus angesprochen und verwendet werden.

Nachdem die Initialisierung der Kern-Strukturen erfolgreich abgeschlossen wurde, kann mit der eigentlichen Simulation begonnen werden. Hierzu wird innerhalb der Projekt-Instanz mit Hilfe einer entsprechenden Hauptsimulationsmethode über alle vorhandenen Instanzen der Verrichtungs-Klassen iteriert. Das Abbruchkriterium der Iteration kann dabei der erfolgreiche Abschluss der Berechnung aller Tätigkeiten, ein Fehler in der Simulation oder das Auftreten eines sogenannten »Dead-Locks« sein. Bei letzterem handelt es sich um eine Endlosschleife in der Iteration der noch nicht simulierten Verrichtungen, welche beispielsweise durch ungültige Synchronisations-Beziehungen verursacht werden kann.

Während der Simulation wird nun wiederum die interne Abarbeitungsroutine jeder einzelnen Verrichtung ausgeführt. Ziel der Verrichtungsabarbeitung ist es dabei, die Menge der abstrakten Zustände zu erzeugen. Dies kann entweder direkt durch die implementierte Logik in der Verrichtungs-Abarbeitungsroutine erfolgen – dann handelt es sich um eine »Elementarverrichtung« – oder durch Instanziierung untergeordneter Verrichtungen, die dann intern die abstrakten Zustände erzeugen (»Komplexverrichtung«). Somit entsteht aus der durch den Benutzer vorgegebenen Sequenz von Verrichtungen während der Simulation ein Baum aus Verrichtungsinstanzen mit abstrakten Zuständen als Blättern.

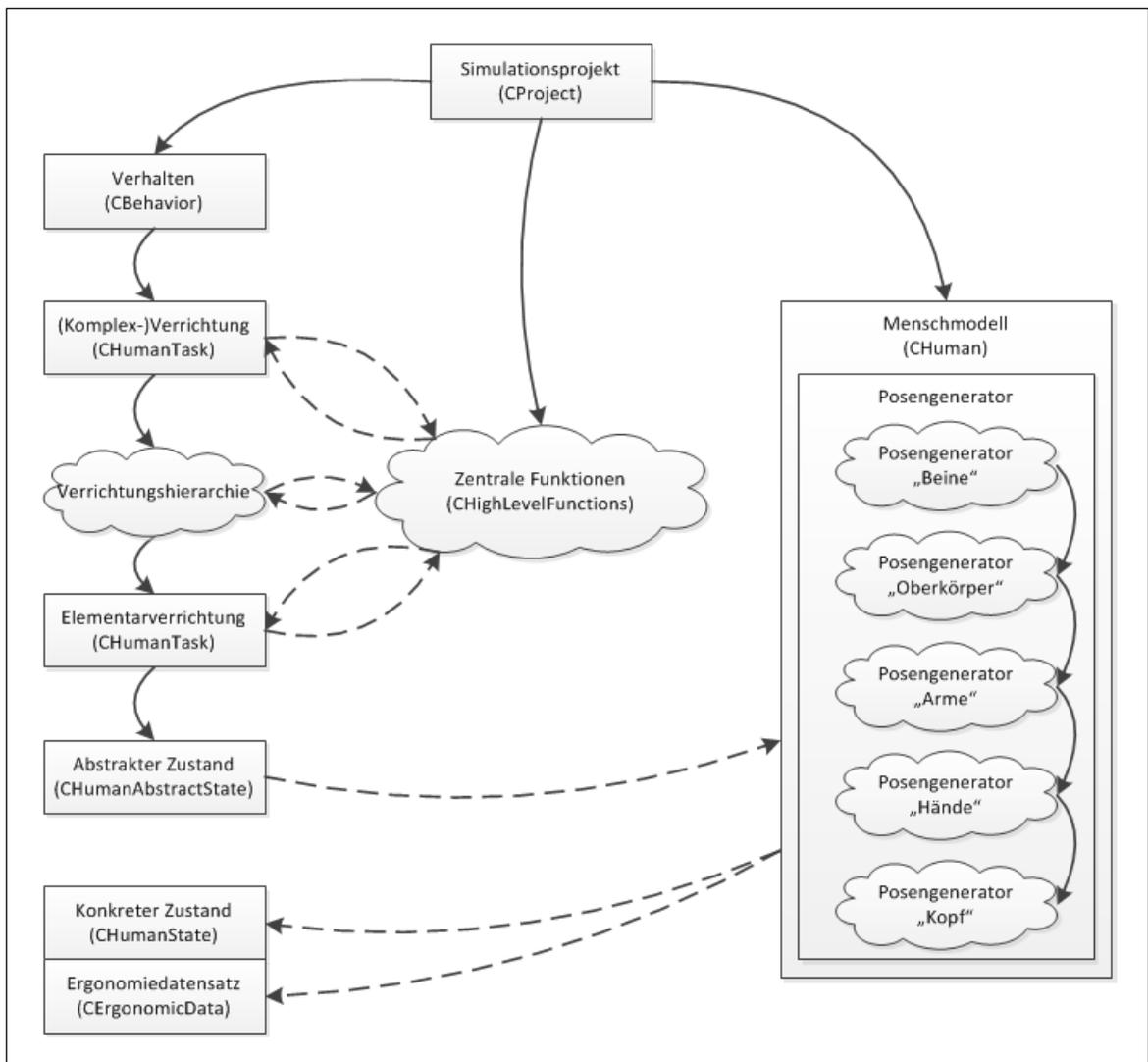


Abbildung 23: Menschmodell-Bewegungsgenerierung im Überblick [59]

Da die abstrakten Zustände aber lediglich zeitliche und geometrische Randbedingungen beschreiben, können sie noch nicht ohne weiteres zur Visualisierung verwendet werden. Zuvor müssen sie noch durch die Anwendung auf ein konkretes Menschmodell in ebenso konkrete Zustände für dieses Menschmodell überführt werden. Hierbei kommt die eingangs des Kapitels diskutierte gestaffelte Methodik der Körperhaltungs-Generierung zum Einsatz. Dabei werden nacheinander die entsprechenden (Teil-)Posengeneratoren für die oben genannten „Körperteil-Funktionsgruppen“ durchlaufen.

3.6.4 Beispielverfahren zur Körperhaltungsgenerierung

Die Verteilung des Gesamtproblems der Körperhaltungsgenerierung auf einzelne Posengeneratoren ermöglichte es, diese weitestgehend unabhängig voneinander zu entwickeln. Sowohl die Anzahl der beteiligten Körperteile und Freiheitsgrade als auch die Art der ausführbaren Teilbewegungen unterscheiden sich bei den gebildeten „Körperteil-Funktionsgruppen“ relativ stark. Aus diesem Grund wurde für jede Funktionsgruppe ein „maßgeschneidertes“ Verfahren entwickelt, wobei die Berechnungseffizienz durch die Berücksichtigung menschlicher Bewegungsmuster im Vordergrund stand.

Im ersten (Teil-)Posengenerator wird die „Körperteil-Gruppe“ »Beine« angesteuert. Da der zugehörige Körperteil »Hüfte« in der Hierarchie sehr weit „oben“ angeordnet ist, beeinflusst dessen geometrische Lage automatisch alle anderen Körperteile und muss somit zuerst bestimmt werden. Der Ausgangspunkt des analytischen Verfahrens, welches einzeln für das linke und rechte Bein durchlaufen wird, ist eine gegebene Hüftposition und ein gegebener Fußpunkt, welche zuvor berechnet bzw. direkt als Parameter übergeben wurden. Zur Reduktion der Komplexität werden die relevanten Körperteile (Oberschenkel, Unterschenkel, Fuß, Zehen) in einer Ebene abgebildet, welche durch die Positionen von Hüfte, Knie und Fußgelenk aufgespannt wird. Diese Vereinfachung ermöglicht die Berechnung in 2D unter Verwendung trigonometrischer Formeln. (Abbildung 24)

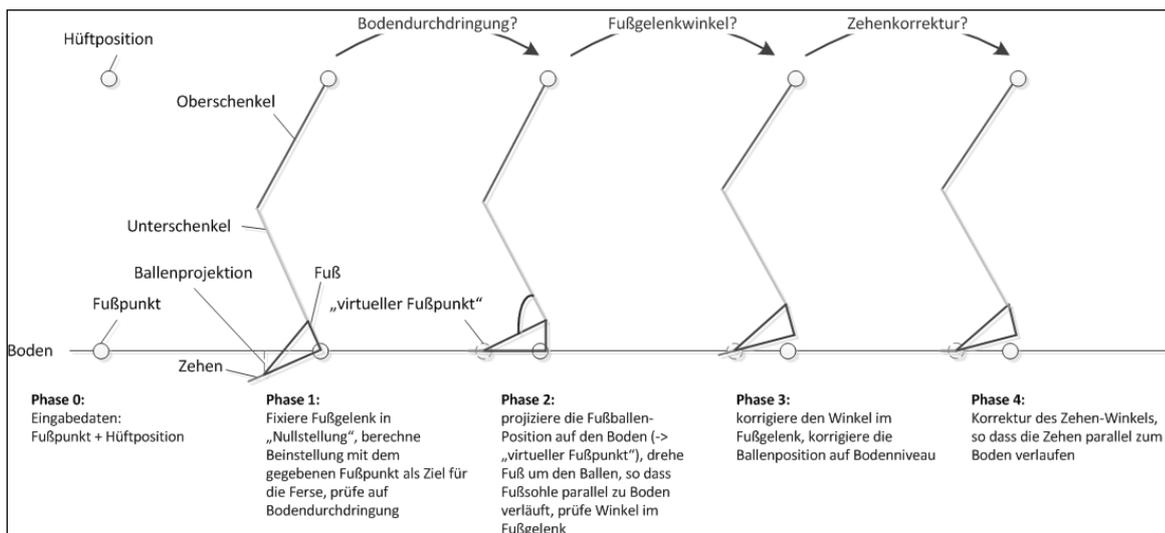


Abbildung 24: Phasen der Posen-Generierung für die BeinEinstellung

In der ersten Berechnungsphase wird der Fuß in seiner „Nullstellung“ (Winkel im Fußgelenk auf „null“ gesetzt) fixiert und es wird eine Beinstellung berechnet, bei der der gegebene Fußpunkt als Ziel für die Ferse verwendet wird. Anschließend erfolgt eine Prüfung, ob der Fuß den Boden durchdringt. Sollte dies nicht der Fall sein, ist die Berechnung beendet, andernfalls wird mit der Phase 2 fortgefahren. Diese beginnt mit der Verschiebung des Fußballens auf Bodenniveau und einer anschließenden Drehung des Fußes um den Ballen bzw. den »virtuellen Fußpunkt« (Ballenprojektion), so dass die Fußsohle parallel zum Boden verläuft. Durch die Drehung ergibt sich eine neue Position für das Fußgelenk, welche wiederum zur Berechnung einer neuen Ober- und Unterschenkelstellung benutzt werden kann. Als Grundlage zur Überprüfung der neuen Beinstellung dient der neue Winkel des Fußgelenks. Sollte dieser innerhalb der vorgegebenen Grenzen liegen, ist die Berechnung beendet; andernfalls muss eine weitere Korrektur durchgeführt werden. In der dritten Phase wird zunächst die eingetretene Fehlstellung korrigiert. Anschließend wird der Fuß mit seinem Ballen wieder auf das Bodenniveau verschoben – dabei gilt der „virtuelle Fußpunkt“ weiterhin als Referenz. Dadurch sollte es nun zum „Anheben der Ferse“ kommen, was wiederum zu einer korrigierten Position des Fußgelenks führt. Mit Hilfe dieser Position des Fußgelenks wird nun abermals eine neue Stellung für Ober- und Unterschenkel berechnet. In einer abschließenden vierten Phase wird dann lediglich noch die Stellung der Zehen korrigiert. Dabei werden die Zehen über ihren Freiheitsgrad so orientiert, dass sie parallel zum Boden ausgerichtet sind. Die Ansteuerung der Oberkörper-Segmente würde im Normalfall keine allzu großen Probleme bereiten, solange die betreffenden drei Segmente der vereinfachten Wirbelsäule ungeachtet aller Randbedingungen bewegt werden könnten. Allerdings erfordert die Ansteuerung des »Human V5« über die DELMIA-API die Angabe von Winkelwerten für den Lenden- und Brustwirbelsäulenbereich, was eine „Übersetzung“ der internen Winkelwerte in die externen DELMIA-Freiheitsgrade notwendig macht. DELMIA bildet die übergebenen Winkelwerte wiederum nach einem unbekanntem Verfahren auf alle 18 internen Segmente der Wirbelsäule des »Human V5« Menschmodells ab. Da das genaue Verfahren der DELMIA-internen Abbildungsvorschrift nicht ermittelt werden konnte, entschied sich der Autor für ein geeignetes Näherungsverfahren.

Als Berechnungsgrundlage für die Ansteuerung dient das übergebene Oberkörper-Constraint, welches in einem Vorverarbeitungsschritt aus einem oder beiden gegebenen Hand-Constraints berechnet werden kann. Auf Basis der Relativposition dieses Constraints (im Menschmodell-Koordinatensystem) lässt sich eine Einstufung in Entfernungsbereiche für die Objekthöhe und -entfernung in Bezug auf die Hüfte des Menschmodells vornehmen. In Abhängigkeit davon, in welchem Entfernungsbereich sich das Ziel befindet, werden nun die geeigneten Prozentwerte für Beugung, Verdrehung und Neigung bestimmt. Dies erfolgt in einem iterativen Verfahren, in dem (unter Berücksichtigung der Armlänge) die minimal erforderliche Hüft- bzw. Oberkörperbeugung berechnet wird. Zur Ausführung der Oberkörperbeugung stehen über eine interne Datenhaltungskomponente die drei parametrischen Standardposen für Oberkörperbeugung, -verdrehung und -neigung zur Verfügung. Hierbei ermöglicht die entwickelte Datenhaltungskomponente die bidirektionale Übersetzung der Prozentwerte in die Winkelwerte des »HumanV5«-Oberkörpers und die Winkelwerte für die vereinfachte 3-Segment-Wirbelsäule. Erreicht wird dies durch die Verwaltung und den Zugriff auf hinterlegte Abbildungstabellen (Mapping), welche im Vorfeld in einem ausgelagerten Initialisierungsprozess gewonnen wurden.

Neben dem Verfahren zur Berechnung der Beinstellung zählt die Berechnung der Armstellung zu den komplexeren der eingesetzten Verfahren. Im Gegensatz zur Beinstellung, bei der für die Fuß-Constraints derzeit nur die Positionen berücksichtigt werden, kann bei der Armstellung (Oberarm, Unterarm, Hand) optional auch die Ausrichtung des Endeffektors »Hand« relevant sein. Als Eingaben für die Berechnung der Armstellung dienen eine oder zwei Zielvorgabe(n) für die linke und / oder rechte Hand. Darüber hinaus wird ein Flag übergeben, das bestimmt, was bei Fehlen einer Zielvorgabe mit dem jeweiligen Arm geschehen soll. Insoweit besteht die einfachste Möglichkeit darin, den Arm in der Stellung zu belassen, in der er sich gerade befindet. Alternativ kann der Arm aber auch in seine „Normalstellung“ gebracht werden, die wiederum seinem ausgestreckten, der Schwerkraft folgenden Zustand entspricht.

Bei der Berechnung der Armstellung anhand von Zielvorgaben erfolgt zunächst eine Überprüfung, ob das jeweilige Constraint nur die Position oder auch die Ausrichtung am Ziel vorschreibt. Hierzu wird eine entsprechende Eigenschaft (Property) des »Constraints« ausgewertet. Für die ausrichtungsrelevante Berech-

nung kommt ein iteratives Verfahren zum Einsatz, was im Wesentlichen aus zwei Phasen besteht:

- Berechnung potentieller Handstellungen am Zielobjekt
- Berechnung und Bewertung jeweilig zugehöriger Unter- und Oberarmstellungen

Über das gegebene Constraint und den Hand-Offset lässt sich die ideale Handstellung am Ziel leicht ermitteln. Da diese ideale Handstellung angesichts der anderen Gelenkbeschränkungen allerdings nur in den seltensten Fällen direkt verwendet werden kann, wird in einem Vorverarbeitungsschritt – durch Variation eines Abweichungswinkels um jede der drei Koordinatenachsen – eine Liste potentieller Handstellungen am Zielobjekt bestimmt. Die Liste wird dabei nach den geringsten Abweichungen vom Idealzustand sortiert, d.h. die ideale Handstellung erscheint in der Liste immer an erster Position.

Nachdem die Liste der potentiellen Handstellungen erzeugt wurde, wird nun versucht, zu jeder Handgelenkstellung eine passende die definierten Gelenkbeschränkungen erfüllende Ober- und Unterarmstellung zu ermitteln. Innerhalb dieser Iteration wird zudem eine Bewertung der einzelnen Armstellungen vorgenommen, wodurch am Ende jeder Iteration für jedes Handgelenk immer die am besten bewertete Armstellung zurückgegeben wird. Als Bewertungskriterien dienen dabei nach derzeitigem Stand eine möglichst geringe Höhe der Ellenbogen und möglichst geringe Winkelwerte im Handgelenk. Der Abbruch der Iteration erfolgt, sobald eine passende Armstellung gefunden wurde.

Nun kann es aber auch Fälle geben, für die die obigen Iterationen kein Ergebnis liefern. Der Autor hat deshalb eine Rückfallvariante berücksichtigt, die so aussieht, dass das Verfahren für die Armstellungsberechnung ohne „Hand-Ausrichtungszwang“ verwendet wird. Hierbei kommt ein analytischer Ansatz zum Einsatz, welcher der Berechnung der Beinstellung ähnlich ist. Das Handgelenk wird in seiner „Nullstellung“ (= beide Handgelenks-Freiheitsgrade auf null Grad eingestellt) fixiert, wonach das Problem dann in einer 2D-Interpretation unter Zuhilfenahme trigonometrischer Berechnungen gelöst wird. Fingerstellungen bzw. Griffhaltungen tragen bei der Simulation mit Hilfe digitaler Menschmodelle zwar einerseits zu einem realistischerem Eindruck bei, werden aber andererseits angesichts ihrer Komplexität meist weitestgehend außen vor gelassen. Allein schon die Anzahl der relevanten Elemente eines detaillierten

Handmodells (15 bewegliche Segmente mit bis zu zwei Freiheitsgraden) lässt den Verarbeitungsaufwand erahnen, der bei einer Einzelansteuerung notwendig wäre. Hinzu kommt, dass von den Handbewegungen lediglich bei entsprechend körpernahen Kameraperspektiven überhaupt etwas zu erkennen ist. All das führt dazu, dass die meisten Arm-Hand-Ansteuerungs-Algorithmen am Körperteil Hand(-gelenk) enden, da dies letztendlich der Körperteil ist, der die Verbindung zu einem potentiellen Zielobjekt herstellt. Da das verwendete Menschmodell über einzelne Finger-Segmente verfügt, galt es, eine möglichst effiziente Möglichkeit zu finden, diese einzelnen Segmente anzusteuern.

In der Arbeitswissenschaft lassen sich verschiedenste Klassifikationen von „per Hand“ aus-geführten Tätigkeiten finden. Allen gemein ist eine Einstufung in verschiedene »Griffarten« bzw. »Handstellungen«. Um diese verwendbar zu machen, wurde in der bereits erwähnten internen Datenhaltungskomponente eine Reihe standardmäßiger Handposen hinterlegt, auf die über entsprechende Methoden zugegriffen werden kann. Dazu ist nicht nur die Angabe nötig, um welche Pose es sich handelt, sondern auch die Eingabe eines Prozentwerts, der den »Ausprägungs-Grad« der Hand-Pose bestimmt. Dabei entspricht eine Angabe von 0% immer der offenen Hand, während 100% immer die maximale Ausprägung der jeweils gewählten Pose definiert. Bei angegebenen Zwischenwerten findet eine Interpolation der in der Datenhaltung hinterlegten Winkelwerte der einzelnen Fingersegmente statt, wodurch auch Zwischenposen, wie beispielsweise »Hand halb geöffnet«, abbildbar sind.

Folgende Griffarten wurden bislang integriert (Abbildung 25 – von links nach rechts):

- offene Hand
- Kugelgriff
- Zylindergriff
- Pinzettengriff Typ 1 (Daumen auf 4 Finger)
- Pinzettengriff Typ 2 (Daumen auf 2 Finger)
- Pinzettengriff Typ 3 (Daumen auf Zeigefinger)
- Daumen auf Zeigefinger-Seite
- Zeigefingereinsatz
- Daumendruck
- Schreibgriff

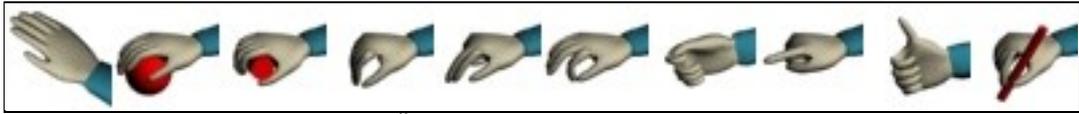


Abbildung 25: Übersicht implementierter Griffarten

Die Berechnung der Kopfstellung ist verglichen mit den vorgestellten Verfahren der anderen (Teil-)Posengeneratoren geradezu trivial. Als Ausgangspunkt der Berechnung fungiert ein Constraint, welches den Blickpunkt repräsentiert. Anschließend muss der Kopf über einfache Verfahren der 3D-Geometrie nur noch so ausgerichtet werden, dass er mit seiner „Blick-Achse“ in Richtung des Blickpunkts zielt. Nachdem diese Berechnung abgeschlossen ist, wird sicherheitshalber noch eine Überprüfung auf Einhaltung der Freiheitsgrade und bei Nicht-Einhaltung eine entsprechende Korrektur der berechneten Winkelwerte durchgeführt. Durch ein zusätzliches Flag, was dem „Kopf-Posengenerator“ mitgegeben wird, besteht die Möglichkeit, die Kopfstellung auf die „Kopf-Rotations-Achse“ (= z-Achse) einzuschränken. Sie wird primär bei der Bestimmung von Blickpunkten während des Laufens eingesetzt. Auf diese Weise ist es möglich, die berechneten Fußpunkte als Blickpunkte zu übergeben und diese ohne ihre integrierten Höheninformationen zu verwenden.

Bei den beschriebenen Verfahren für die (Teil-)Posengeneratoren handelt es sich um exemplarische Lösungsansätze, die nachweisen sollten, dass das vom Autor erdachte Prinzip der Trennung in abstrakte und konkrete Bewegungsbeschreibung im Zusammenspiel mit dem Menschmodell »Human V5« trotz dessen Eigenheiten und der speziellen Programmierschnittstelle funktioniert. Dieser Nachweis konnte erbracht werden. Bei der Anwendung der Methodik auf andere Menschmodelle kann sich unter Umständen der Einsatz anderer ggf. allgemeinerer Berechnungsverfahren zur Bestimmung der Körperhaltung als sinnvoll erweisen. (z.B.: allgemeine Verfahren der Inversen Kinematik) An den im vorliegenden Simulationssystem vorgesehenen Schnittstellen würde sich unter dieser Voraussetzung dann jedoch nichts ändern.

3 Methodische Vorarbeit

Gruppe	Eingangsdaten	Körperteile	Verfahren
»Beine«	[Constraint] Oberkörper-Referenz (optional) [Constraint] linker Fuß, rechter Fuß und Hüfte (jeweils optional) [Prozentwert] Gewichtsverteilung (zwischen linkem und rechtem Fuß) [Aufzählungstyp] Art der Beinbewegung	Hüfte Oberschenkel Unterschenkel Fuß Zehen	Analytisches Verfahren (3-stufig)
»Oberkörper«	[Constraint] Oberkörper-Referenz [Aufzählungstyp] Art der Oberkörperbewegung	Wirbelsäulen-Segmente	Iteratives Verfahren / Nutzung parametrischer Standardposen
»Arme«	[Flag] „einnehmen der Standardpose erlaubt / verboten“ [Constraint] linke Hand und rechte Hand (optional)	Schulter Oberarm Unterarm Hand	Iteratives Verfahren
»Hände«	[Flag] „einnehmen der Standardposen erlaubt / verboten“ [Aufzählungstyp] + [Prozentwert] Angaben zu den gewünschten Hand-Posen (Griffart + Prozentsatz der Ausprägung)	alle Finger-Segmente	Nutzung parametrischer Standardposen
»Kopf«	[Flag] „einnehmen der Standardpose erlaubt / verboten“ [Constraint] Kopf (Blickpunkt) [Flag] Einschränkung der Kopfbewegung auf die Rotationsachse	Hals Kopf	„einfache“ Look-At-Berechnung

Tabelle 10: Verfahren zur Körperhaltungsgenerierung

4 Softwaretechnische Umsetzung

4.1 Vorgehensweise

Nachdem der methodische Grundstein für das Simulationssystem gelegt worden war, galt es, die erarbeiteten Konzepte und Methoden auf ihre Praxistauglichkeit hin zu überprüfen. Der geeignetste Weg hierfür war die softwaretechnische Umsetzung des konzipierten Systems. Dabei folgte der Autor den klassischen Phasen der Softwareentwicklung von der Anforderungsanalyse bis hin zum Entwurf der technischen und fachlichen Architektur. Weil es sich bei dem vorliegenden System um einen völlig neuartigen Ansatz eines Simulationssystems für digitale Menschmodelle handelt, halfen insbesondere die Erkenntnisse, welche aus vorangegangenen und zum Teil parallel entwickelten Software-Prototypen gewonnen werden konnten. (siehe Kapitel 1.4) Ferner galt es zu berücksichtigen, dass die durchgeführten Forschungsarbeiten selbstverständlich auch das Ziel hatten, früher oder später in ein kommerzielles Softwareprodukt einzufließen. Diese Tatsache generierte weitere Rahmenbedingungen, die bei der Entwicklung berücksichtigt werden mussten. Ein erster grundlegender Entwurf des aus fachlicher Sicht zu schaffenden Systems wurde bereits im Abschnitt 3.3 diskutiert. (siehe Abbildung 7) Die dabei benannten Komponenten umfassen jedoch zunächst nur den fachlichen Kern der vorliegenden Arbeit. Für eine Umsetzung in die Praxis – und sei es nur zu Testzwecken – bestand die Notwendigkeit, den Fokus auch auf Elemente auszudehnen, die aus Sicht des Themas dieser Dissertation eher zum Bereich der Softwareumgebung gehören.

Die Implementierung einer alle Aspekte des geplanten Simulationssystems abdeckenden Lösung hätte zu Beginn der Entwicklung die verfügbaren Kapazitäten überstiegen. Daher entstand die Idee, sich zunächst in ein bestehendes System unter Verwendung dessen Infrastruktur zu integrieren. Dieses System war die PLM-Software DELMIA V5. Ausschlaggebend für diese Verfahrensweise war die Tatsache, dass DELMIA bereits über ein integriertes Menschmodell – den Human V5 (siehe Kapitel 3.1) – verfügte, das ein gutes Beispiel für die bislang sehr aufwändige Modellierung menschlicher Arbeit in Simulationssystemen ist. Die Anwendung der im Rahmen dieser Arbeit konzipierten Ansteuerungsmethode für digitale Menschmodelle bot damit die ideale Gelegenheit, um deren Vorteile im Vergleich zu existierenden Methoden darzustellen. Darüber hinaus war und ist DELMIA neben dem Konkurrenzprodukt von Siemens [63] eines

4 Softwaretechnische Umsetzung

der großen Planungssysteme im Bereich der Digitalen Fabrik und somit ein geeigneter Ausgangspunkt für eine kommerzielle Vermarktung der neuen Technologie.

Das Primärziel war damit also, die entwickelte Methodik zunächst in ein Plug-In für ein großes existierendes Planungssystem einfließen zu lassen. Dabei sollten allerdings gleichzeitig alle Optionen für die Weiterentwicklung zu einer eigenständigen Simulationslösung offen gehalten werden. Diese beiden grundlegenden Ziele galt es bei jedem Entwicklungsschritt zu berücksichtigen, was den Autor bei der Umsetzung mehrfach vor Herausforderungen stellte. So war es etwa notwendig, sich bei der Integration einerseits an die vorgegebenen DELMIA-APIs zu halten. Andererseits galt es zu beachten, sich nicht zu sehr in den Spezifika und Besonderheiten des DELMIA-Menschmodells und dessen vorhandener Ansteuerung zu verlieren. Die klare Abgrenzung von Funktionalitäten durch Modularisierung und die Entwicklung eigener interner Schnittstellen ermöglichte es schließlich, den Grundstein für ein System zu schaffen, welches 2010 produktreif als Plug-In erreichte, zwei Jahre später erstmals auch als eigenständige Software verkauft werden konnte und kontinuierlich weiterentwickelt wird. (Abbildung 26)

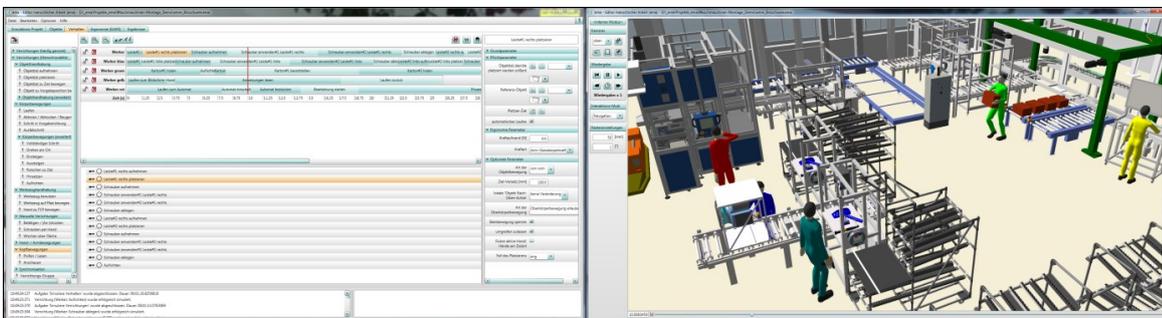


Abbildung 26: Editor menschlicher Arbeit (ema5) im 2-Bildschirm-Betrieb

4.2 Anforderungen, Anwendungsfälle und Arbeitsablauf

Den Kern der Anforderungen an das zu entwickelnde System bildet das Ziel, ein Simulationssystem zu schaffen, mit dessen Hilfe digitale Menschmodelle auf möglichst effiziente Art und Weise angesteuert werden können. Alle sich daraus ergebenden fachlichen Anforderungen wurden bereits in den Methodik-Abschnitten 3.1 bis 3.4 dargestellt. Im Rahmen der Anforderungsanalyse wurden darüber hinaus auch einige Einschränkungen und Abgrenzungen vorgenommen (siehe Ende des Abschnitts 3.1). Auf dem Weg zur softwaretechnischen Umsetzung galt es nun, die funktionalen Anforderungen in einem entsprechenden Konzept einer fachlichen Architektur adäquat abzubilden, während die nicht-funktionalen Anforderungen beim Entwurf der technischen Architektur berücksichtigt werden mussten.

Im ersten Schritt der Erarbeitung des Softwareentwurfs wurden die zahlreichen Wünsche und Ideen, welche im Laufe der Zeit vom Autor ermittelt oder an den Autor herangetragen wurden, in konkrete Funktionen und Anwendungsfälle für die zu entwickelnde Software umformuliert. Hierbei ging es in erster Linie um eine Systematisierung der zahllosen Ideen, in zweiter Linie aber auch um eine Abgrenzung des Funktionsumfangs einer ersten Version des Programms gegenüber möglichen Funktionalitäten von Folgeversionen. Zunächst galt es dabei, die potentiellen Anwendergruppen des Systems zu ermitteln, um darauf aufbauend die zu erfassenden Anwendungsfälle zu definieren. Folgende Benutzerrollen wurden im Rahmen der Anforderungsanalyse identifiziert:

- Administrator
- Linienkonzeptplaner
- Zeitplaner / Industrial Engineer
- Anlagenkonstrukteur
- Arbeitswissenschaftler

Mit Ausnahme der Rolle „Administrator“ könnten die übrigen Benutzerrollen auch zu einer übergeordneten Rolle „Planer“ bzw. „Anwender“ zusammengefasst werden. Zum einen vereinfachte die vorgenommene Untergliederung aber die spätere Einordnung der Software in die Geschäftsprozesse der Zielkunden, zum anderen erleichterte sie die Bildung und Zuordnung der groben Hauptanwendungsfälle:

4 Softwaretechnische Umsetzung

- UCM1: System anpassen
- UCM2: Grobgestaltung / -dimensionierung
- UCM3: Austaktung
- UCM4: Verifikation (Bewertung)
- UCM5: Dokumentation

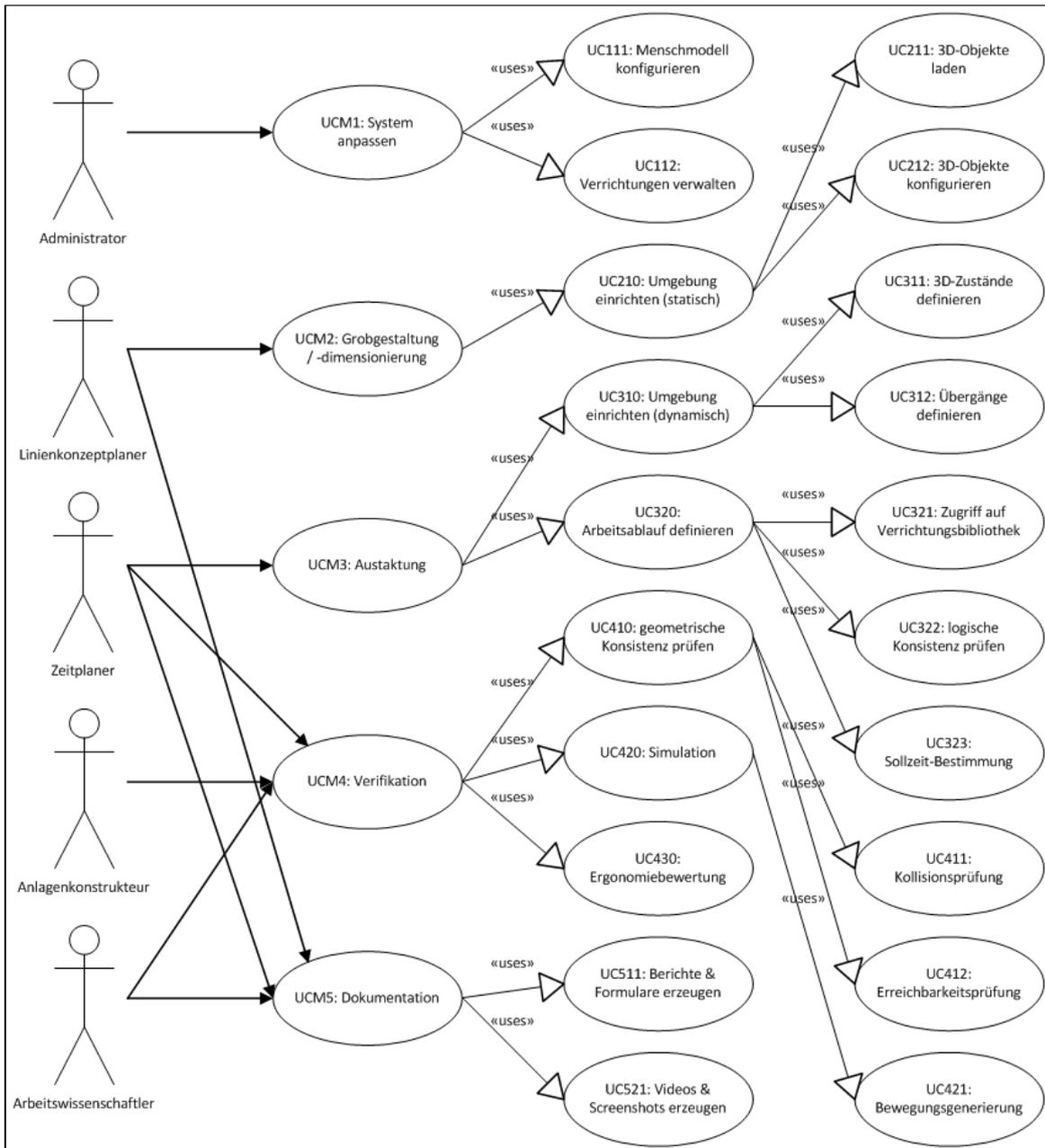


Abbildung 27: Benutzerrollen und Anwendungsfälle im Gesamtsystem

In einem hieran anschließenden Detaillierungsschritt wurden die ermittelten Hauptanwendungsfälle in Detailanwendungsfälle gegliedert (Abbildung 27), was die nachfolgende Analyse der für die Software erforderlichen Komponenten ermöglichte. Zunächst entwickelte der Autor jedoch ein Schema für den geplanten Arbeitsablauf beim Umgang mit dem zu schaffenden Gesamt-System. Dabei wurden die ermittelten Anwendungsfälle in einem Arbeitsablauf-Diagramm (Abbildung 28) angeordnet.

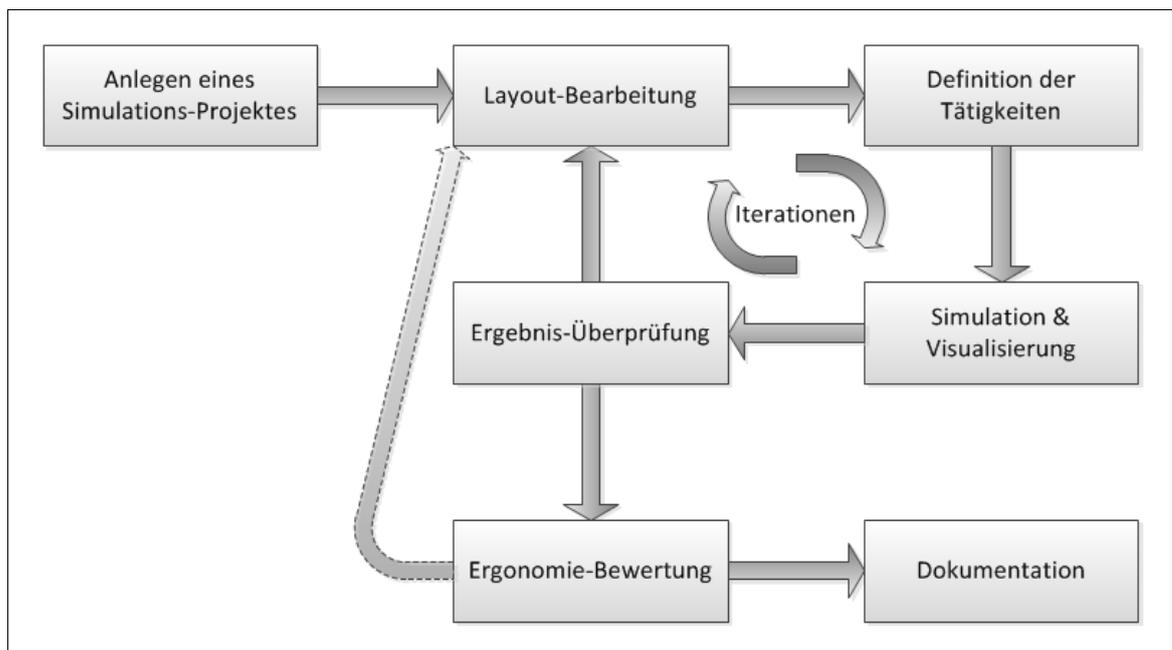


Abbildung 28: Arbeitsablauf für den Umgang mit dem Simulationssystem

Der Arbeitsablauf beginnt mit einem Initialisierungsschritt. Dabei wird die Simulationsumgebung an die Bedürfnisse des Benutzers angepasst und es werden relevante „Kopf-Daten“ der zu erstellenden Simulation erfasst. Dieser Arbeitsschritt orientiert sich an den gängigen auf „Papier-und-Stift“-basierenden Arbeitserfassungs- und Planungsmethoden, bei denen die Rahmenbedingungen der Untersuchung in einem Protokoll festgehalten werden. Es folgt die Erstellung und Bearbeitung des sogenannten 3D-Layouts. Alle für die Simulation benötigten Produkt- und Ressourceninformationen müssen in Form geometrischer Objekte in das Simulationsprojekt geladen werden. Anschließend müssen die Objekte im sogenannten »Start-Zustand« angeordnet werden. Die Anordnung umfasst dabei sowohl die räumliche Verteilung der Objekte als auch

deren hierarchische Struktur. Innerhalb der »Layout-Bearbeitung« werden bereits die zum Einsatz kommenden Menschmodelle platziert, denen anschließend die zu simulierenden Tätigkeiten zugeordnet werden können. Der Arbeitsschritt »Definition der Tätigkeiten« umfasst die Spezifizierung aller relevanten Antworten auf die „W-Fragen“ aus Abschnitt 3.4.2. Dabei kann der Benutzer auf Elemente der oben ebenfalls schon erläuterten Verrichtungsbibliothek (siehe Abschnitt 3.5.3) zurückgreifen, welche die Syntax und Semantik der Beschreibung liefern.

Nachdem der Aufbau des Simulationsszenarios erfolgt ist, kann die Simulation durchgeführt werden. Basierend auf den verwendeten Verrichtungen und dem Szenario werden die Bewegungsdaten für die digitalen Menschmodelle erzeugt. Hierbei finden implizite Plausibilitätstests statt, welche der zugrundeliegenden Absichtskomponente entstammen. Die Ergebnisse der »Simulation« werden anschließend visualisiert und in Form von textuellen Informationen, wie zum Beispiel Fehlermeldungen, Warnungen und Protokollen ausgegeben. Anhand der generierten Daten kann nun eine »Ergebnis-Überprüfung« durch den Benutzer erfolgen. In Anbetracht der zu erwartenden Komplexität der Arbeitsinhalte der digitalen Menschmodelle können im Anschluss noch weitere Iterationsschleifen folgen, beginnend mit Anpassungen des 3D-Layout und / oder der Tätigkeitsbeschreibung. Sobald die Simulation die vom Benutzer gewünschte Qualität erreicht hat und somit dem realen oder vorgedachten Beispiel hinreichend nahe kommt, kann die Bewertung der ergonomischen Aspekte der berechneten Bewegungen beginnen. Das bei der Ergonomiebewertung zum Einsatz kommende Verfahren »EAWS« [30] beruht auf der Auswertung geometrischer Bedingungen und Zeitanteile in Verbindung mit auftretenden Kräften und zu handhabenden Lasten. Daher ist es von besonderer Bedeutung, dass die vorangegangenen Iterationen zu einem möglichst realitätsnahen Simulationsergebnis geführt haben. Abschließend fließen die erzeugten Ergebnisse zur Weiterverwendung in eine entsprechende »Dokumentation« des Szenarios ein. Das Simulationssystem soll hierbei sowohl durch die Erzeugung entsprechender Formulare sowie die Möglichkeit der Weiterbearbeitung in externen Programmen wie auch durch die Generierung von Bild- und Videomaterial unterstützen.

4.3 Systemarchitektur

Nachdem die Vorarbeiten in Form der Anforderungsanalyse, der Definition der Anwendungsfälle und der Konzeption des Arbeitsablaufs abgeschlossen waren, widmete sich der Autor der Detailierung der fachlichen Systemarchitektur. Als Ausgangspunkt diente dabei das in Abschnitt 3.3 entworfene Komponentenschema. Da es lediglich die Komponenten des wissenschaftlichen Kerns dieser Arbeit enthielt, erweiterte der Autor das Schema unter Berücksichtigung der ermittelten Anforderungen zunächst um eine Datenhaltungskomponente und eine Auswertungskomponente (siehe Abbildung 29). Darüber hinaus wurden die Elemente der Verarbeitungslogik (Absichtskomponente und Bewegungsgenerator, siehe Abschnitt 3.3) in der neuen, übergeordneten Systemkomponente Simulationskern zusammengefasst. Schließlich wurden alle Komponenten entsprechend ihrer Rolle in einer klassischen 3-Schichten-Architektur [64, p. 10] angeordnet. Dabei wird die Präsentationsschicht durch die 2D-Benutzeroberfläche und die 3D-Visualisierungs-Komponente gebildet. In die Geschäftslogikschicht fallen sowohl der Simulationskern als auch die Auswertungskomponente, während die Datenzugriffsschicht durch die Komponenten Datenbereitstellung und Datenhaltung repräsentiert wird.

Im Hinblick auf den geplanten Arbeitsablauf beim Umgang mit dem System und die strikte Trennung zwischen den Arbeitsphasen »Konfiguration des Simulationsszenarios« und »Simulationsdurchführung« war es aus Sicht des Autors nicht sinnvoll, den Simulationskern direkt mit der Präsentationsschicht interagieren zu lassen. Stattdessen wurde ein datenzentrierter Ansatz [65, pp. 19, Tabelle 2.1] mit dem Ziel einer losen Koppelung der Daten von deren Verarbeitung verfolgt. Hierzu wurde ein Rahmensystem konzipiert, welches auf der einen Seite als Daten- und Befehlsvermittler und auf der anderen Seite als zentrale „Andock-Stelle“ innerhalb der Geschäftslogikschicht fungieren sollte. Das Rahmensystem dient sinnbildlich als „Rückgrat“ der Software. Es regelt alle Befehls- und Datenflüsse innerhalb des Systems, kommuniziert über entsprechende Schnittstellen mit den anderen Modulen und beinhaltet die interne „Arbeitsdaten-Haltung“. Darüber hinaus implementiert das Rahmensystem nicht nur die zentralen Schnittstellen für die Anbindung der Benutzeroberfläche, sondern stellt auch „Dienstleister-Funktionalitäten“ für alle anderen Module zur

Verfügung (Job-Management, Logging, Konfigurations-Management, Lizenzierung).

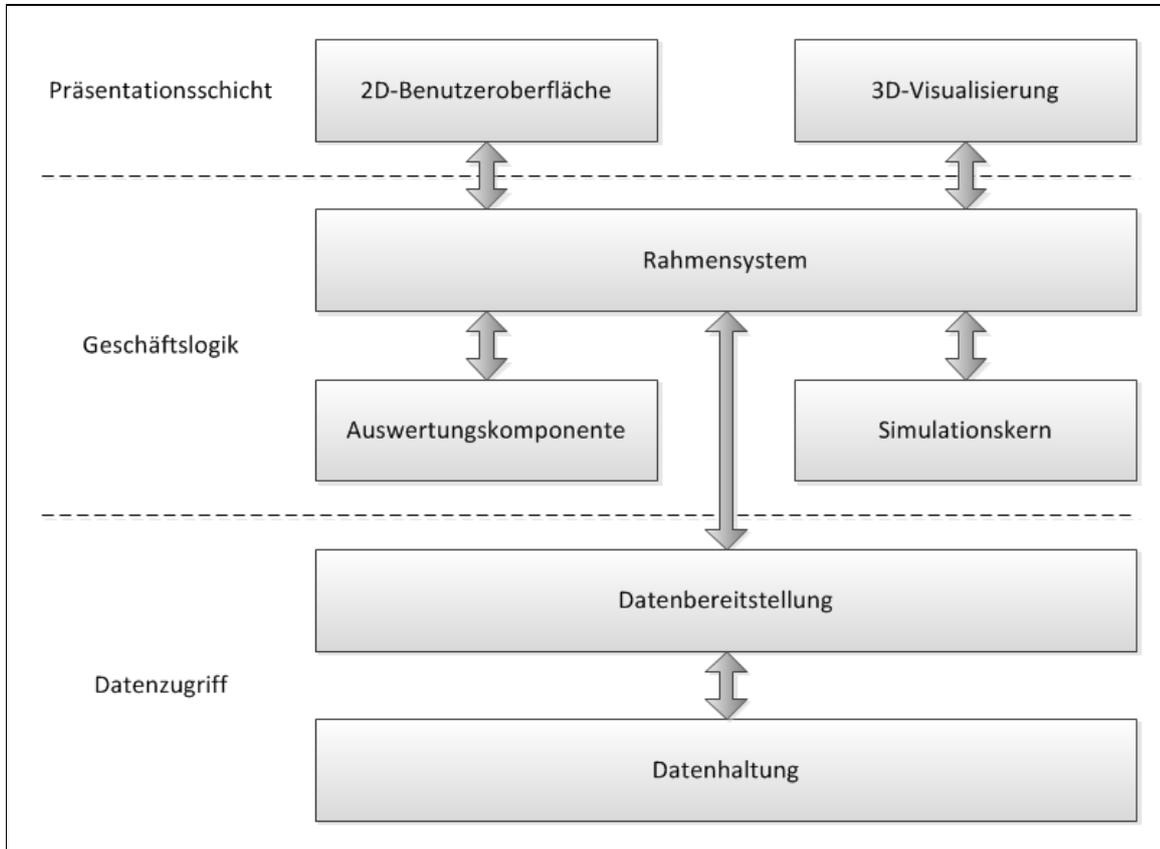


Abbildung 29: Fachliche Architektur des Gesamtsystems (vereinfacht)

Die Entkoppelung der Simulationslogik und deren indirekte Ansteuerung boten aus Sicht des Autors darüber hinaus noch einen weiteren Vorteil. Zum damaligen Zeitpunkt der Konzeption war noch nicht absehbar, welches Laufzeitverhalten die – zum Teil sehr komplexen – Algorithmen zur Bewegungsgenerierung im Zusammenspiel ergeben werden. Darüber hinaus bestand der Anspruch, das Simulationssystem später auch auf „schwächeren“ PC-Systemen einsetzen zu können, welche sich von „High-End-Hardware“ insbesondere im ohnehin immer relevanten Bereich der Leistungsfähigkeit von Hauptprozessor und Grafikberechnung unterscheiden. Eine parallele Simulation mit gleichzeitiger Wiedergabe des Simulationsergebnisses wäre auf solchen schwächeren Systemen wahrscheinlich generell nicht möglich gewesen und spätestens bei der Simulation mehrerer digitaler Menschmodelle zur gleichen Zeit hätten die verfügbaren Sys-

temressourcen dann nicht mehr ausreicht, wie erste Abschätzungen ergeben haben. Vorteil der Auslagerung der Simulationslogik war es insoweit auch, eine separate Berechnung der Ergebnisse zu ermöglichen, verbunden mit der Option, diese nach Abschluss der Simulation zu visualisieren oder je nach Bedarf anderweitig auszuwerten. Darüber hinaus bot dieser Ansatz auch die theoretische Möglichkeit, Verarbeitungskapazitäten auf andere Systeme auszulagern und somit bei Bedarf auch in einer Server-Client-Systemwelt arbeiten zu können.

Die den einzelnen Komponenten zugeordneten Aufgaben des oben dargestellten fachlichen Komponentenschemas (Abbildung 29) lauten wie folgt:

- 2D-Benutzeroberfläche:
 - Auswahl und Konfiguration der erforderlichen 3D-Objekte (Produkte, Ressourcen, Menschmodelle)
 - Konfiguration des Simulationsablaufs unter Verwendung der Verrichtungsbibliothek
 - Eingabevalidierung
 - Ergebnis-Darstellung (textuell, 2D-Diagramme, ...)
- 3D-Visualisierung:
 - Visualisierung des Simulationsprojekts
 - Interaktive Manipulation des 3D-Layouts
 - Interaktive Parameterauswahl
 - Wiedergabe der Simulationsergebnisse
- Rahmensystem:
 - Verknüpfung der Komponenten aus Geschäftslogik- und Datenzugriffsschicht
 - Bereitstellung aller erforderlichen Informationen für die Präsentationsschicht
 - Lizenzierung, Benutzer-/Rechteverwaltung
 - Log-Datei-Verwaltung
 - Verwaltung globaler Systemeinstellungen
- Simulationskern:
 - Konsistenz-, Plausibilitäts- und Fehlerprüfungen
 - Überführung des Initialzustands des 3D-Layouts und der Tätigkeitsbeschreibung in alle relevanten Simulations-Zustände (Bewegungsgenerierung für digitale Menschmodelle)

- Bestimmung auswertungsrelevanter Zusatzinformationen während der Simulation
- Auswertungskomponente:
 - Verarbeitung der Ergebnis-Rohdaten aus der Simulation und deren Auswertung hinsichtlich:
 - Dauer
 - Wertschöpfung
 - ergonomischer Aspekte
- Datenbereitstellung:
 - Speichern und Laden von Simulationsprojekten
 - Import-Funktionen für diverse Geometriedaten-Formate
 - Datenimport und -export-Funktionalitäten zur Interaktion mit Fremdsystemen
- Datenhaltung:
 - Sicherstellung der Datenkonsistenz
 - Datenkomprimierung und -verschlüsselung
 - Verwaltung der zum Simulationsprojekt zugehörigen Geometrie-Dateien

In einem weiteren Detailierungsschritt befasste sich der Autor mit der Verteilung der einzelnen Komponenten, deren Kommunikation untereinander und den vorzusehenden Schnittstellen zu Fremdsystemen. Ein Ausgangspunkt war dabei die Überlegung, bei welchen Komponenten des Systems es zu Überlappungen bei der Integration in ein mögliches Fremdsystem kommen könnte. Bei der Analyse der für eine Anbindung in Frage kommenden PLM-Systeme stellte sich die Verwendung der vorhandenen 3D-Visualisierungskomponenten und der systemspezifischen Datenhaltung als sinnvolle „Andockstellen“ heraus. Die Hauptgründe dafür lagen zum einen in den nativen 3D-Datenformaten, deren Spezifikationen zum Teil nicht öffentlich zugänglich sind und zum anderen in den umfangreichen Datenbanksystemen, welche die Datenhaltung im Hintergrund der PLM-Systeme realisieren. Darüber hinaus erlaubte die Auslagerung dieser Funktionalitäten die Fokussierung auf den innovativen Kern des zu entwickelnden Simulationssystems für digitale Menschmodelle.

In Abbildung 30 ist ein weiteres Komponentenschema dargestellt, diesmal jedoch unter Berücksichtigung der angesprochenen Schnittstellen zu Fremdsys-

temen. Nach wie vor dient das Rahmensystem als zentrales „Rückgrat“ des Systems. Daran angebunden sind die Geschäftslogik-Komponenten zur Simulationsberechnung und zur nachfolgenden Bewertung nach ergonomischen Gesichtspunkten, wobei die zuletzt genannte Komponente ein Teil der im fachlichen Schema angesprochenen Auswertungskomponente ist. Die fachlichen Kernkomponenten werden von den beiden Schnittstellen zur 3D-Visualisierung und zur Datenbereitstellung flankiert. Hierbei wurde die Möglichkeit vorgesehen, beide Schnittstellen durch eine Fremdsystem-Integration realisieren zu lassen. Als wahrscheinlichste Variante einer solchen Anbindung erschien dabei die Implementierung eines Fremdsystem-spezifischen Plug-Ins, welches die bidirektionale Übersetzung der zu übermittelnden Daten zwischen dem Simulationssystem und dem betreffenden PLM-System übernimmt. Die beiden Schnittstellen können durch die vorgeschlagene Architektur aber ebenso durch eigene Komponenten realisiert werden. Dies würde zum einen eine eigene Datenhaltungs-Komponente und zum anderen ein eigenes Modul zur 3D-Visualisierung erfordern.

Das Komponentenschema in Abbildung 30 stellt darüber hinaus zwei weitere Detaillierungen dar. So wurde im Bereich der Geschäftslogik der Simulationskern um eine Geometrie-Bibliothek erweitert, welche als „Fundament“ der Simulationsberechnung dienen soll. Hierfür wird die Bibliothek alle erforderlichen Klassen und Strukturen für die 3D-Geometrieberechnungen bereitstellen. Auch im Bereich der 2D-Benutzeroberfläche wurde eine Detaillierung vorgenommen. Dabei wurde eine weitere Zwischenschicht – die Bedienlogik – eingeführt. Diese zusätzliche Separierung von Logik und Präsentation hat das Ziel, die Flexibilität beim Umgang mit und der Auswahl von möglichen Technologien für Benutzeroberflächen zu erhöhen. Darüber hinaus erlaubt diese Erweiterung zur 4-Schicht-Architektur eine Trennung der gestalterischen Elemente (Farben, Grafiken und Formate) von den funktionalen Themen der Steuerung, Prüfung und Verarbeitung der Ein- und Ausgabe-Daten.

4 Softwaretechnische Umsetzung

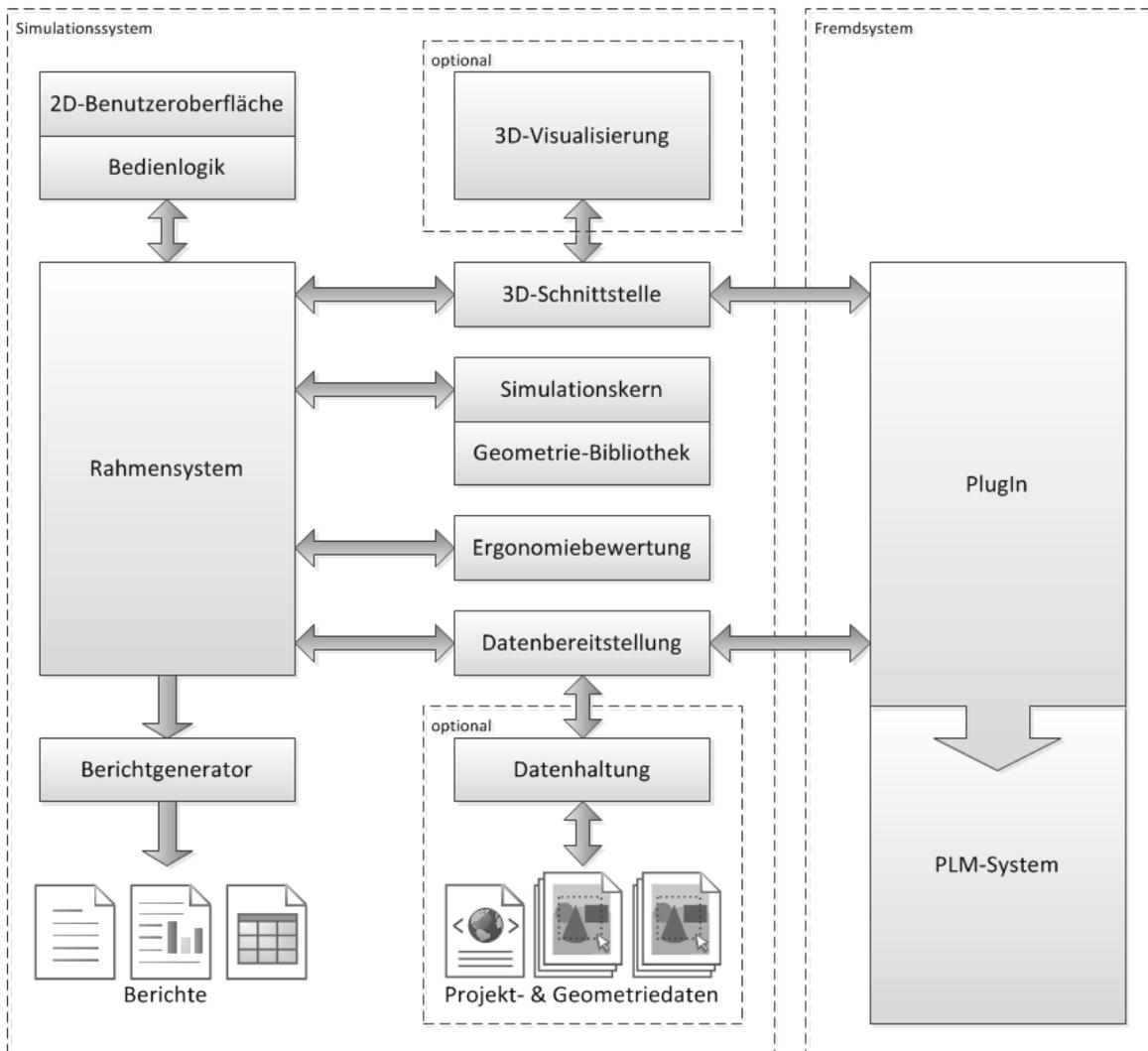


Abbildung 30: Komponenten und Schnittstellen im System

4.4 Gestaltung der Benutzeroberfläche

Die Benutzeroberfläche eines Software-Systems dient zur Interaktion des Anwenders mit dem System. Dabei gibt es die beiden grundlegenden Interaktionsrichtungen „Eingabe“ und „Ausgabe“. Ziel der Eingabe ist es, das vorliegende Simulationssystem mit allen für seine Ausführung notwendigen Daten zu versorgen (direkt durch entsprechende Bedienelemente bzw. indirekt durch Angabe eines Speicherortes), darüber hinaus aber auch die Steuerung der Abarbeitungsfolge. Aufgabe der „Ausgabe“ ist es demgegenüber, dem Benutzer alle Informationen zu geben, die er für die Interaktion mit dem System benötigt - hierzu zählt insbesondere die Bereitstellung und Aufbereitung der Simulationsergebnisse.

Beim praktischen Einsatz des vorliegenden Simulationssystems zählen folgende Aktionen zur „Eingabe“:

- Konfiguration des Systems (Spracheinstellung, Lizenzverwaltung, etc.)
- Konfiguration / Parametrisierung der Objekte bzw. Erstellung des 3D-Layouts
- Konfiguration / Parametrisierung der zu simulierenden Tätigkeiten und Zuweisung der Verrichtungen
- Starten der Simulationsberechnung
- Ergänzung von Ergonomie-relevanten Parametern
- Starten der Ergonomie-Berechnung

Die „Ausgabe“ umfasst folgende Aspekte:

- Darstellung aller relevanten Simulationsobjekte und deren Eigenschaften
- Darstellung aller Verrichtungen und deren Eigenschaften und Abhängigkeiten
- Darstellung der Ergebnisse der Ergonomie-Berechnung
- Darstellung der Ergebnisse der Simulation
- Ausgabe von relevanten Status- und Systeminformationen

Die Bedienung des Simulationssystems folgt einem klar gegliederten Arbeitsablauf (Abbildung 28) mit in sich abgeschlossenen Arbeitsschritten. Dementsprechend wurde die Benutzeroberfläche in Anlehnung an diese Arbeitsschritte in die Teilbereiche »Allgemeine Simulationseinstellungen«, »Layout-Erstellung«,

4 Softwaretechnische Umsetzung

»Definition des Arbeitsablaufs«, »Ergonomie-Auswertung« und »Ergebnis-Auswertung« untergliedert. Jedem Element des Arbeitsablaufs wurde ein Registerkarten-Bedienelement zugeordnet (Abbildung 31 #2), wodurch die Reihenfolge der Arbeitsschritte auch noch einmal in der Anordnung der Bedienelemente dokumentiert ist. Ergänzt werden die alphanumerischen Eingabebereiche (Abbildung 31 #3) durch einen »3D-Visualisierungsbereich« (Abbildung 31 #4), welcher sowohl zur Eingabe von geometrischen Informationen als auch zur Präsentation des Ergebnisses der Simulation dient. Schließlich befindet sich im unteren Teil der Benutzeroberfläche ein »Status-Bereich« (Abbildung 31 #5) der relevante Systemereignisse und Informationen zum Arbeitsstatus länger dauernder Berechnungen anzeigt.

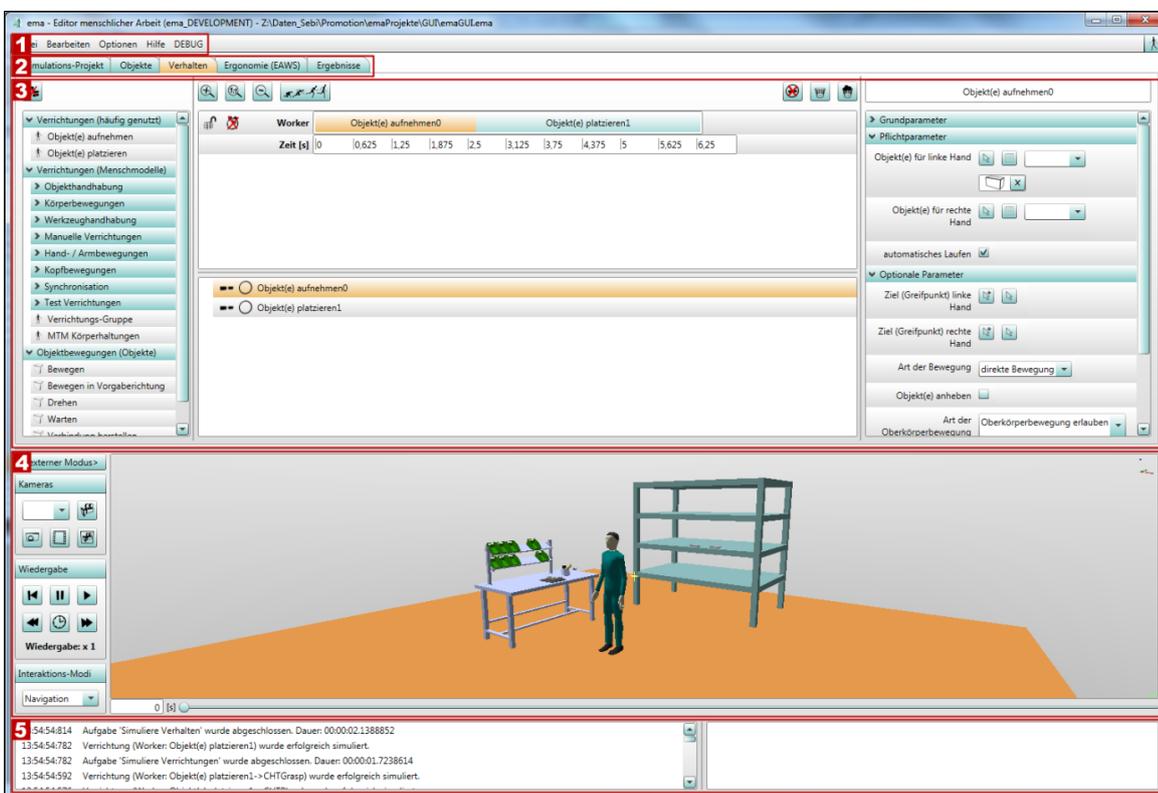


Abbildung 31: Die Benutzeroberfläche im Überblick*

*) 1 - Hauptmenü, 2 - Registerkarten, 3 - Eingabebereiche, 4 - 3D-Visualisierungsbereich, 5 - Status-Bereich

Der Bedienbereich der »Simulationseinstellungen« erstreckt sich sowohl über das bereits erwähnte Registerkarten-Bedienelement als auch über das Hauptmenü-Bedienelement im oberen Bereich der Benutzeroberfläche (Abbildung 31

#1). Im Menü können programmweite Einstellungen grundsätzlicher Natur vorgenommen werden, die damit dann nicht nur das aktuelle Simulationsprojekt betreffen, sondern auch alle anderen nachfolgend bearbeiteten Projekte. Hierzu gehören sowohl Informationen zur benutzerspezifischen Anpassung der Benutzeroberfläche, durch sogenannte „Themes“, als auch allgemeine Einstellungen zur Granularität und diversen anderen Aspekten der Simulationsberechnung (Berücksichtigung von Reichweitenbeschränkungen, Logging, etc.). Auf der Registerkarte mit der Bezeichnung `simulationsprojekt` können elementare Informationen zur gerade bearbeiteten Simulation hinterlegt werden. Neben der Bezeichnung des Projekts und einem Kommentar werden hier auch Informationen zum Ersteller sowie Zeitstempel der Erstellung und letzten Änderung gepflegt.

Für das Simulationssystem stellt der »3D-Visualisierungsbereich« (Abbildung 32) das zentrale Bedienelement zur Benutzereingabe und zur Ergebnispräsentation dar. Das Bedienelement teilt sich in zwei Bereiche: Während auf der linken Seite in einem schmalen Streifen untergeordnete Bedienelemente zur Steuerung des Visualisierungsbereichs angeordnet sind, befindet sich die eigentliche Visualisierung im rechten Bereich.

Die Einstellungen, welche im linken Bereich vom Benutzer gesetzt werden können, beziehen sich zum einen auf die Kamerasteuerung (Auswahl von Standard-Kameraperspektiven, Umschaltung zwischen perspektivischer Projektion und Parallelprojektion, etc.) und zum anderen auf Aspekte der Wiedergabesteuerung der Ergebnissimulation (Geschwindigkeit, Vor- und Rücklauf, Pause, etc.). Die eigentliche Interaktion mit dem 3D-Bereich erfolgt durch direkte Eingabe per Maussteuerung. Hierbei können einerseits Steuerungsfunktionen für die aktuelle 3D-Ansicht ausgeführt werden (rotieren, verschieben, zoomen) und andererseits Auswahlfunktionen benutzt werden, um beispielsweise Objekte oder 3D-Parameter zu selektieren. Da der »3D-Visualisierungsbereich« eine so zentrale Rolle spielt, besteht die Möglichkeit, ihn sowohl integriert als Teilelement der Gesamt-Benutzeroberfläche (siehe Abbildung 31) zu nutzen oder aber alternativ in einem separaten Fenster (Abbildung 32). Die zuletzt genannte Variante bietet mehr Platz für den eigentlichen 3D-Bereich und kann deshalb insbesondere bei der Verwendung eines zweiten Bildschirms sinnvoll eingesetzt werden.

4 Softwaretechnische Umsetzung

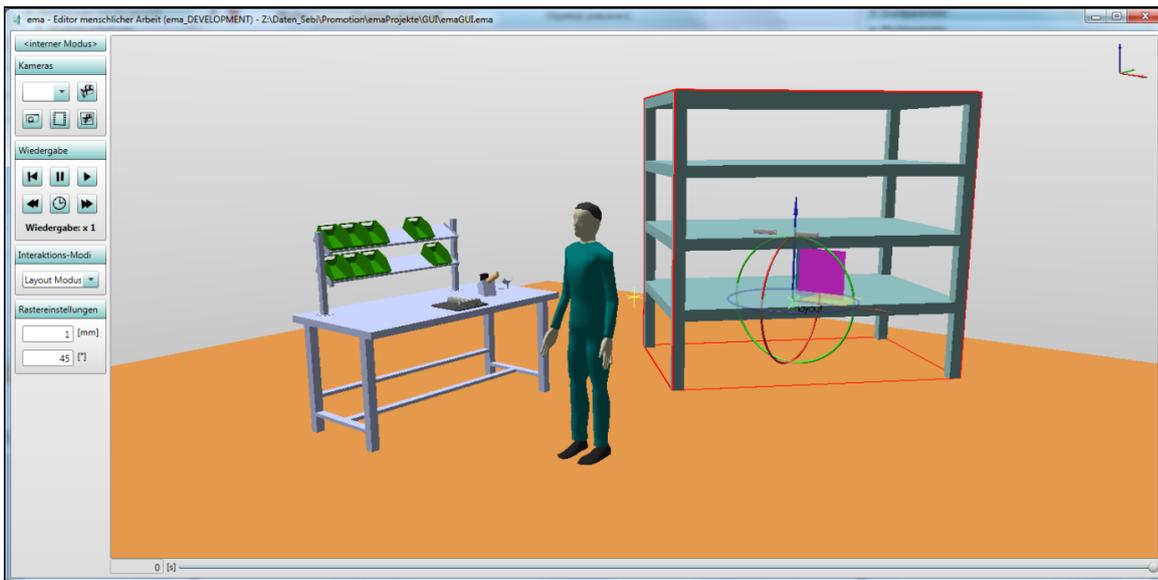


Abbildung 32: Interaktive Objektmanipulation in der 3D-Ansicht

Die Registerkarte, welche die Erstellung des Layouts und die Verwaltung der 3D-Geometrie unterstützen soll, trägt die Bezeichnung *Objekte*. Im linken Bereich befindet sich eine Übersicht über alle derzeit geladenen 3D-Objekte. Diese Übersicht kann wahlweise in einer Baum- oder einer Listendarstellung angezeigt werden. Bei Auswahl der Listendarstellung besteht die Möglichkeit, nach Objekttypen zu filtern oder Objekte nach (Teil-)Bezeichnungen zu suchen. Die Befehlsschaltflächen, welche sich oberhalb der Objektliste befinden, dienen zum Hinzuladen weiterer Objekte bzw. zum Löschen bereits vorhandener Objekte. Um tatsächlich gestalterisch tätig werden und die einzelnen Objekte zu einem Layout anordnen zu können, werden im rechten Bereich der Registerkarte die Eigenschaften des derzeit selektierten Objekts angezeigt. Neben Anpassungen der beschreibenden Eigenschaften (Name, Masse, etc.) können hier auch Änderungen an den aktuellen Translationen und Rotationen der 3D-Objekte vorgenommen werden, um sie wunschgemäß anzuordnen. Alternativ kann zur Anpassung der Objekttransformationen auch ein entsprechender interaktiver Modus des 3D-Visualisierungsbereichs genutzt werden. Hierbei können die 3D-Objekte per Maus und unter Verwendung von sogenannten Transformations-Hilfsobjekten verschoben oder gedreht werden (Abbildung 32). Die Veränderungen werden wechselseitig aktualisiert. Desgleichen wurde zur eindeutigen

Identifizierung und besseren Wiedererkennung der Objekte in ihren einzelnen Repräsentationen eine „Cross-Highlighting“-Funktionalität implementiert.

Zur Konfiguration der zu simulierenden Tätigkeiten dienen die Bedienelemente, die sich innerhalb der Registerkarte `verhalten` befinden. Hierbei wurde die im Rahmen der Diplomarbeit des Autors [10] konzipierte Bedienmetapher, welche die Definition von Arbeitsabläufen per Drag-and-Drop von Elementen aus einer Verrichtungsbibliothek in dafür vorgesehene Zeitleisten der jeweiligen digitalen Menschmodelle vorsieht, konsequent weiterverfolgt. Als Alternativdarstellung zur horizontalen Anordnung der Verrichtungs-Elemente in der Zeitleiste ist eine zusätzliche Auflistung derselben hinzugekommen. Sie erleichtert das Auffinden bestimmter Elemente insbesondere dann, wenn die Verrichtungen nach abgeschlossener Simulation ihre Bedienelement-Breite an die berechnete Simulationszeit anpassen.

Während im mittleren Bereich der Registerkarte die grundsätzliche Abfolge der Tätigkeiten definiert wird, können im rechten Bereich die sogenannten »Verrichtungs-Parameter« eingestellt werden (Abbildung 31 #3), die während der Simulation die tatsächliche Art der Ausführung der jeweiligen Verrichtung steuern. Jede Verrichtung beinhaltet dabei eine bestimmte Anzahl von Parametern, wobei sich diese in die Kategorien `Allgemeine`, `Pflicht-`, `Optionale` und `Ergonomie-Parameter` unterteilen lassen (siehe Kapitel 3.5.3). Die Bandbreite der Parametertypen reicht von Text-basierten Parametern (Bezeichnungen, Zahlen-Werte) über sog. Entscheidungs- und Auswahl-Parameter bis hin zu geometrischen Parametern mit Bezug zu Elementen des Layouts. Bereits auf der Benutzeroberfläche wird der Anwender dabei bei der Eingabe der Parameterwerte unterstützt und im Falle von Fehleingaben etwa durch einen entsprechenden Hinweistext auf Fehler aufmerksam gemacht, die andernfalls spätestens beim Start eines Simulationslaufs zu dessen Scheitern führen würden. Darüber hinaus werden insbesondere bei den etwas komplexeren Parametern mit Bezug zum Layout alternative Eingabemethoden angeboten (Auswahl aus Grafik, Auswahl aus Liste, Übernahme aus anderer Verrichtung), um den Benutzer in die Lage zu versetzen, die korrekten Werte situationsbedingt möglichst intuitiv einstellen zu können.

Nach einem erfolgreichen Simulationslauf besteht zudem die Möglichkeit zur Berechnung und Ausgabe statistischer Informationen zum Ablauf (Zeit, Werterschöpfung, Laufwege, etc.) sowie dessen Bewertung nach ergonomischen Kri-

4 Softwaretechnische Umsetzung

terien anhand der Methode »EAWS«. Beide Analysen werden durch jeweils eine Registerkarte (Ergonomie und Ergebnisse) repräsentiert. Die Ausgaben erfolgen je nach Art der Informationen entweder tabellarisch oder in Form von Diagrammen. Darüber hinaus werden insbesondere bei der Ergonomiebewertung einige Informationen auch in Form von Piktogrammen ausgegeben. Dies verbessert zum einen die Erfassbarkeit der Daten. Zum anderen erhöht es aber auch den Wiedererkennungswert und nimmt Bezug zu den sonst üblichen Formularen.

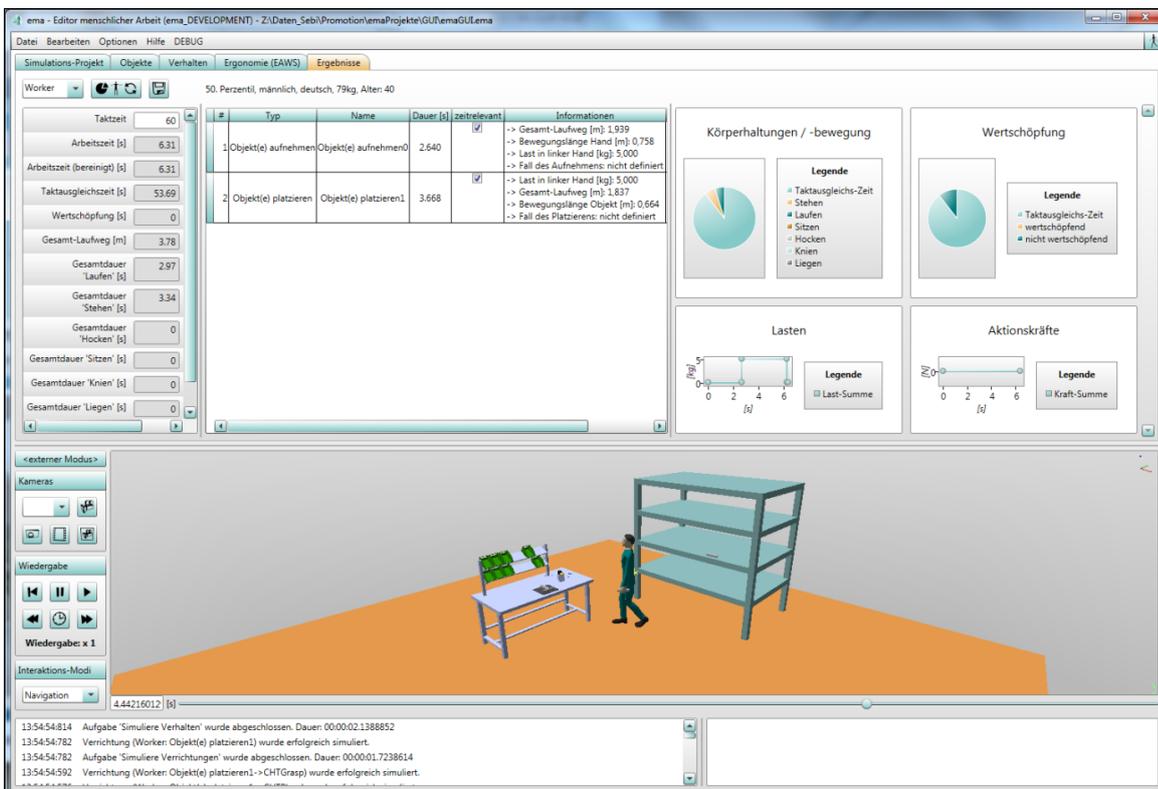


Abbildung 33: Ergebnis-Auswertung

4.5 Implementierung des Simulationskerns

4.5.1 Überblick

Aus Gründen der Modularisierung und der Kapselung von Teilfunktionalitäten des Simulationssystems wurde für den Simulationskern eine eigenständige Komponente entwickelt (siehe Abbildung 29 und Abbildung 30). Diese Programm-bibliothek beinhaltet alle Businesslogikklassen, die für die Durchführung der Simulation und der Bewegungsgenerierung erforderlich sind. Alle „Kern-Klassen“ werden nach Bedarf instanziiert, danach zur Durchführung der ihnen zugedachten Aufgaben eingesetzt und anschließend wieder verworfen - d.h. bei jedem Simulationslauf werden neue Instanzen der Klassen des Kerns generiert. Diese Vorgehensweise bietet unter anderem die Möglichkeit, Simulationsläufe gegebenenfalls asynchron in verschiedenen Varianten durchführen zu können, ohne dabei jeweils auf die vorherige Beendigung des vorangegangenen Simulationslaufes warten zu müssen.

Die Klassen und Strukturen im Simulationskern lassen sich anhand ihrer Funktionalitäten in fünf grundlegende Gruppen einordnen:

- Elemente zur Objekt- / Szenengraphverarbeitung (`Core.Objects`)
- Elemente zur Verhaltensbeschreibung und -verarbeitung (`Core.Behaviors`)
- Elemente zur Simulationsverarbeitung (`Core.Simulations`)
- Strukturen für den internen Datenzugriff (`Core.DataAccess`)
- Hilfsstrukturen (`Core.BasicStructures`)

Die drei wichtigsten Bereiche (Objekte, Verhalten, Simulation) werden in den nachfolgenden Kapiteln noch etwas genauer beschrieben.

Als zentraler Ausgangspunkt einer jeden Simulation dient eine Instanz der Klasse `CProject`. Diese Instanz wird vom Simulations-Rahmensystem angelegt und entsprechend initialisiert. Zu den Hauptaufgaben zählt die Verwaltung zentraler Hilfsdatenstrukturen wie beispielsweise Verzeichnisstrukturen aller Objekte und Vorrichtungen für einen schnellen und effizienten Zugriff während der Simulation. Als Hilfsobjekte stehen über die Instanz von `CProject` Verweise auf die zentralen Schnittstellen `ILogwriter` (Status- und Fehlermeldungsverwaltung), `IDataStreamManager` (Simulations-Ergebnis-Export) und `IConfig` (Simulationseinstellungen) zur Verfügung. Neben den Verwaltungsaufgaben hat die

Klasse `CProject` auch die Hoheit über die Simulations-Berechnung. Durch den Aufruf der Methode `ProcessBehaviors()` wird die Abarbeitung der einzelnen Objekt-Verhaltensstrukturen gestartet. Dabei wird nacheinander die Abarbeitung der einzelnen Verrichtungen jedes „aktiven“ Objektes angestoßen. Während der Simulation liegen die Hauptaufgaben von `CProject` bei der Status-Rückmeldung an das Simulations-Rahmensystem und der eventuell notwendigen Synchronisierung von Verrichtungen (z.B.: „warten auf Fertigstellung einer anderen Verrichtung“). Hierzu werden intern Synchronisierungssignale (Verrichtungs-Identifikator + Zeitstempel) verwendet, welche in einer separaten Datenstruktur gehalten und überprüft werden. Beim Eintreten eines sogenannten „Deadlocks“ in der Abarbeitung (Endlosschleife) ist es die Aufgabe von `CProject`, diesen zu erkennen und durch den Abbruch der Simulation und die Ausgabe einer Fehlermeldung darauf zu reagieren.

4.5.2 Objektstrukturen

Die Basisklasse und damit wichtigste Klasse innerhalb der Objektstrukturen ist `C3DNode`. Sie repräsentiert einen „geometrielosen“ Knoten des Szenengraphs. Um einen Szenengraph aufzubauen, bedarf es primär der Definition hierarchischer Beziehungen zwischen den einzelnen Knoten und für jeden Knoten der Definition dessen aktueller Transformation. `C3DNode` bietet hierfür alle benötigten Werkzeuge und stellt somit die Grundlage für alle Operationen zur Verfügung, die zur Manipulation des Szenengraphs dienen. Zusammenfassend beinhaltet die Klasse `C3DNode` folgende Funktionalitäten:

- Zustandsverwaltung (aktueller und Initialzustand)
- Manipulation der Objekthierarchie
- Manipulation der Transformation
- Verwaltung von verknüpften Objekten
- Verwaltung von Beschränkungen (»Constraints«)

Als Basisklasse aller 3D-Objekte beinhaltet `C3DNode` eine Reihe von Methoden, welche zur Abfrage oder Änderung des aktuellen Zustands des repräsentierten Objekts dienen. Hierzu zählen die aktuellen hierarchischen Beziehungen, Transformationsinformationen, zugewiesene Verrichtungen und eine Reihe von Hilfsattributen. Die Mehrheit der genannten Parameter muss aufgrund interner Wechselwirkungen durch entsprechende Methoden gesetzt werden, um die interne Konsistenz des Objekts sicherzustellen. Neben den aktuellen Informationen werden auch eine Reihe von relevanten Werten des Initialzustands (Hierarchie, Transformationsmatrix, etc.) vermerkt, um auf der Ebene der Objektstrukturen ein Zurücksetzen des Szenengraphs in den Initialzustand zu ermöglichen.

Generell ist es bei Bedarf möglich, beliebig viele Szenengraphen parallel anzulegen. Voraussetzung hierfür ist lediglich die Instanziierung eines Wurzelknotens (`Root`) durch Aufruf des Basis-Konstruktors ohne Argument. Unter diesen Wurzelknoten können nun beliebig viele Kindknoten „gehangen“ werden, in dem der Wurzelknoten den Kindknoten als Argument im Konstruktor übergeben wird. Jeder Knoten vermerkt intern sowohl seinen direkten Elternknoten (`Parent`) als auch seine hierarchisch untergeordneten Kindknoten (`ChildrenList`). Zusätzlich beinhaltet jedes Element des Szenengraphs einen Verweis auf

seinen Wurzelknoten (Root). Dieser dient später insbesondere zur schnelleren Prüfung, ob sich zwei Knoten im gleichen Szenengraph befinden. Die hierarchischen Beziehungen können jederzeit unter Verwendung der Methode `setParent()` geändert werden.

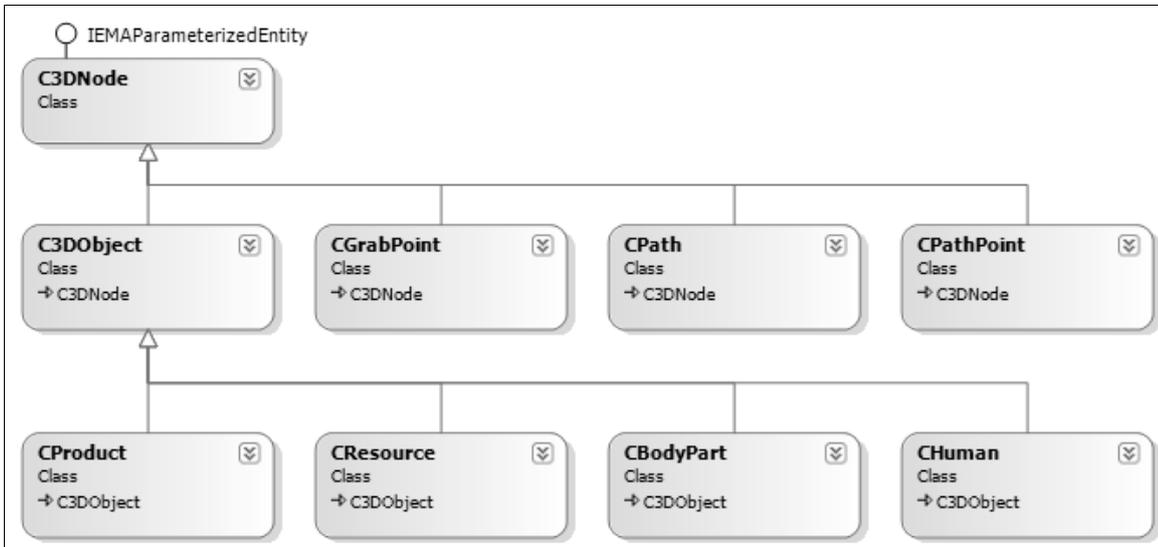


Abbildung 34: Klassendiagramm der Objektstrukturen

Während die Klasse `C3DNode` primär strukturbildende Funktionen beinhaltet, erweitert die von ihr abgeleitete Klasse `C3DObject` diese Funktionalität um einige gestaltbasierte Eigenschaften. Da in der aktuellen Version der Simulationssoftware noch keine Verwaltung von Dreiecksnetzen erfolgt, konzentrieren sich alle geometrischen Berechnungen zunächst auf Abstraktionen der Realgeometrie in Form von Hüllobjekten. Hierzu stehen in erster Linie sogenannte `BoundingBoxes` (Hüllquader) und `BoundingSpheres` (Hüllkugeln) als Member zur Verfügung. Zu beachten ist dabei, dass die Hüllkugel immer direkt aus dem aktuell vorhandenen Hüllquader berechnet wird. Da für manche Berechnungen und Auswertungsfunktionen neben der Form auch das Gewicht eines Objektes relevant sein kann, ergänzt `C3DObject` hierfür den Member `Mass`, welcher zur Verwaltung der Objekt-Masse dient.

Zur Modellierung komplexer strukturierter Objekte, die sich später in der Simulation bewegen sollen, kann es erforderlich sein, für Einzelteile des Objekts sogenannte „Freiheitsgrade“ (degrees of freedom, kurz `DOF`) zu definieren. `DOF` können als rotatorische oder translatorische Freiheitsgrade vorliegen und wer-

den in Form einer Listenstruktur direkt an dem Objekt verwaltet, an dem sie definiert sind. Sie dienen während der Simulation zur Prüfung, ob sich das strukturierte Objekt (z.B.: Menschmodell) mit seinen Bestandteilen innerhalb der definierten Bewegungsgrenzen befindet oder nicht.

Während der Simulation besteht darüber hinaus gelegentlich die Notwendigkeit, Objekte zu manipulieren und sie dazu von anderen Objekten „aufnehmen“ zu lassen. Dieser Sprachgebrauch ist sowohl bei der Arbeit mit Maschinen und Robotern als auch bei der Arbeit mit digitalen Menschmodellen passend. Um spezielle „Greifpunkte“ an Objekten definieren zu können, stellt die Klasse `C3DObject` eine weitere Listenstruktur zur Verfügung, welche mit Elementen vom Typ `CGrabpoint` gefüllt werden kann. Die Greifpunkte können sowohl von außen (z.B. Benutzereingabe) vorgegeben werden als auch bei Abwesenheit automatisch generiert werden (`CalculateGrabPoints()`). Bei Zuordnung bzw. Neuerzeugung eines Greifpunktes erfolgt eine bidirektionale Verknüpfung zwischen dem Greifpunkt und seinem Elternobjekt. Über diese Verknüpfung ist es möglich, von beiden Seiten aufeinander zuzugreifen und somit auch übergreifende Statusinformationen wie beispielsweise „wird verwendet“ (`CGrabPoint.IsUsed()`) und „wird verwendet von“ (`CGrabPoint.UsedByObject`) zu verwalten. Darüber hinaus wurde in `C3DObject` eine Methode implementiert, die die Bestimmung des unter den gegebenen Rahmenbedingungen (Erreichbarkeit, Verwendung der Greifpunkte, ...) „günstigsten“ Greifpunktes ermöglicht. Diese Methode beinhaltet neben der Auswahl des geeignetsten Greifpunktes bei Bedarf auch eine Optimierung der Ausrichtung des Greifpunkts.

Die derzeitige Implementierung der Greifpunkt-Klasse `CGrabPoint` ist bereits bezüglich der Besonderheiten eines menschlichen „Hand-Greifpunktes“ angepasst worden. Dies beinhaltet beispielsweise die Verwaltung der Information, ob der Greifpunkt von der linken oder der rechten Hand benutzt wird. Außerdem beinhaltet die Klasse `CGrabPoint` eine Methode zur automatischen Ermittlung der Art und Weise des Greifens und dabei insbesondere auch des Greif-Prozentsatzes basierend auf dem Hüllquader des Objektes (`GetGraspStyle()`).

Die beiden Klassen `CProduct` und `CResource` dienen in Anlehnung an ihre Namensvetter in DELMIA der weiteren Klassifizierung von allgemeinen Objekten. Während die Klasse `CProduct` derzeit nicht mehr als eine Platzhalterfunk-

tion einnimmt, führt die Klasse `CResource` mit ihrem Member `ToolCenterPoint` eine neue Eigenschaft ein. Wirkpunkte (`ToolCenterPoint`, kurz: TCP) definieren Referenzpunkte, mit deren Hilfe die jeweilige Ressource (z.B.: Werkzeug) an einer Wirkstelle platziert werden kann. Da diese Wirkstelle nicht immer im Objektkoordinaten-Ursprung liegen muss, handelt es sich beim TCP wiederum um ein Objekt vom Typ `C3DNode`, welches als Kindobjekt geführt wird und über sein eigenes Koordinatensystem hierarchisch unterhalb der „Eltern-Ressource“ platziert und ausgerichtet wird.

Als spezielles strukturiertes Objekt ist das digitale Menschmodell eine der wichtigsten Entitäten des Simulationskerns. Um nun ein DHM (digital human model) abzubilden, wurden die beiden Klassen `CHuman`, als Ausgangspunkt für das Menschmodell und `CBodyPart`, zur Unterstrukturierung, eingeführt. Während sich die Aufgaben der `CBodyPart`-Instanzen darauf beschränken, Informationen über Typ und Länge des repräsentierten Körperteils zu liefern, zählt zu den Hauptaufgaben der Klasse `CHuman` die Struktur- und Statusverwaltung während der Simulation. Zu diesem Zweck beinhaltet eine Instanz von `CHuman` eine Auflistung aller zugehörigen Körperteile. Die Körperteile selbst bilden anhand ihrer hierarchischen Abhängigkeiten im Ergebnis das „Skelett“ des Menschmodells (siehe Kapitel 3.6.2). Um von außen direkt auf bestimmte Körperteile zugreifen zu können, wurde eine entsprechende Zugriffsmethode implementiert (`GetBodyPartByType()`).

Zur Statusverwaltung für das digitale Menschmodell zählen auch überladene Methoden der Verwaltung angehangener Objekte (`attach / detach`). In Bezug auf das Menschmodell werden diese Funktionalitäten derzeit zur Abbildung von „Objekt mit der Hand aufnehmen“ verwendet. Die Implementierung ist dabei so allgemein gehalten, dass später auch Objekte mit anderen Körperteilen verknüpft werden können. Somit wäre beispielsweise das Tragen von Schutzbekleidung problemlos abbildbar.

In der aktuellen Version des Simulationssystems findet die komplette Zustandsgenerierung in der Menschmodell-Klasse statt (siehe Kapitel 3.6). Zu diesem Zweck wurde eine Vielzahl von Methoden zur Ansteuerung einzelner Körperteile und Körperteilgruppen implementiert. Die grundsätzliche Vorgehensweise bei der Ansteuerung des Menschmodells besteht in der Vorgabe von Constraints, welche einzelnen Endeffektoren des Menschmodells (Hände, Füße, Kopf, ...) als Zielvorgaben übergeben werden. Diese Constraints werden

durch Hilfsobjekte abgebildet, damit sie während der Simulation entsprechend in den Szenengraphen eingeordnet werden können. Neben den „externen Zielen“ werden dem Menschmodell bzw. dessen Körperteilen während der Initialisierung auch „interne Hilfsobjekte“ angehängt. Diese dienen beispielsweise zur Verwaltung von „Greif-Versatz-Punkten“, also zur korrekten Platzierung von Objekten in den Händen. Alle diese Hilfsobjekte können zentral über die Instanz des jeweiligen Menschmodells angesprochen und verwendet werden (`GetHelperTarget()`).

Mit `CPath` und `CPathPoint` stehen zwei Hilfsklassen zur Abbildung von Pfaden und Pfadpunkten zur Verfügung. Beide Klassen wurden von `C3DNode` abgeleitet und haben somit keinen Gestalt-Bezug. Nichtsdestotrotz erben beide Klassen natürlich die Eigenschaften ihrer Basis-Klasse was wiederum ihre Einordnung in den Szenengraph ermöglicht. Objekte vom Typ `CPath` dienen zur Definition und Arbeit mit Pfaden, die in der Simulation in erster Linie zur Beschreibung von Bewegungsbahnen genutzt werden. Ein Pfad beinhaltet dabei eine Menge von Pfadpunkten, auf die geordnet über eine entsprechende Listenstruktur (`PathPoints`) zugegriffen werden kann. Neben den obligatorischen Pfad-Längen-Berechnungen beinhaltet die Klasse `CPath` eine Reihe von komplexeren Methoden, die beispielsweise zur Glättung des Pfades (`GetSmoothedPath()`) oder zur Interpolation von Pfadpositionen genutzt werden können.

4.5.3 Verhaltensstrukturen

Wie die Zustands- und Bewegungsgenerierung im Detail abläuft, wurde bereits in Abschnitt 3.6 erläutert. Im vorliegenden Kapitel werden nun die hierfür implementierten Strukturen und deren Abhängigkeiten näher betrachtet.

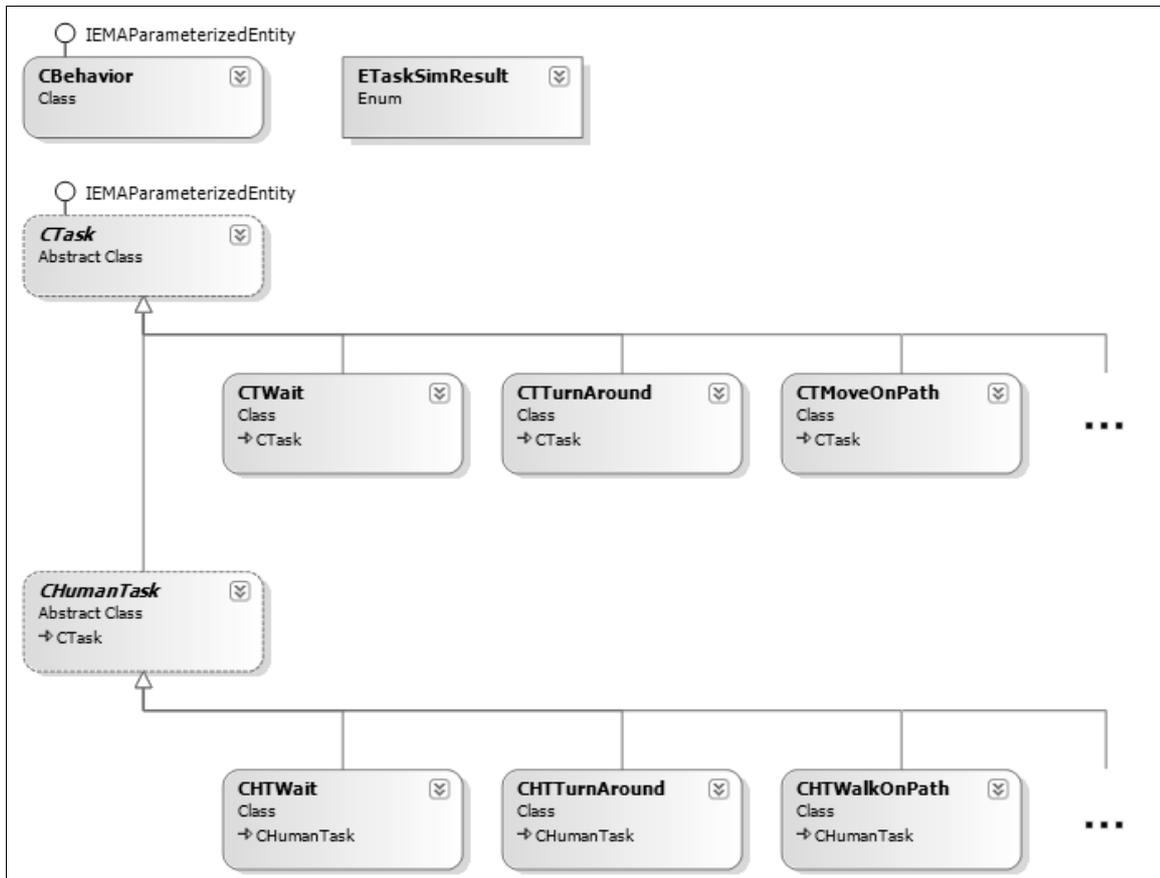


Abbildung 35: Klassendiagramm der Verhaltensstrukturen (Auszug)

Die Klasse `CBehavior` dient als verknüpfendes Element zwischen den Repräsentationen der zu simulierenden Objekte (z.B. Menschmodelle) und deren zugeordneten Verrichtungen. Dabei wurde die Implementierung absichtlich so allgemein gehalten, dass später neben den derzeit genutzten digitalen Menschmodellen auch das Verhalten allgemeiner oder anderer spezieller 3D-Objekte (Maschinen, Roboter, etc.) abgebildet werden kann. Die Verknüpfung zwischen einer Instanz der Klasse `CBehavior` und dem zugeordneten Objekt findet bidirektional statt. Hierzu beinhaltet `CBehavior` die Eigenschaft `AssignedObject`, während in der Klasse `C3DNode` das Pendant `AssignedBehavior`

implementiert wurde. Hauptaufgabe der Instanzen von `CBehavior` ist die Verwaltung aller dem Objekt zugeordneten Verrichtungen in Form einer Listenstruktur (`ListOfTasks`). Die Hauptverarbeitungsmethode, welche über die Instanz von `CProject` aufgerufen wird, durchläuft diese Liste während der Simulation und berechnet durch die jeweils implementierte Verrichtungslogik die Ergebnisse in Form von Zustands-Datensätzen. Auch mit der aktiven Instanz von `CProject` gehen die `CBehavior`-Objekte eine bidirektionale Verknüpfung ein. Dazu wird in `CProject` eine Listenstruktur mit allen vorhandenen `CBehavior`-Instanzen gehalten (`ListOfBehaviors`), während die Property `Project` in `CBehavior` umgekehrt den Zugriff auf die `CProject`-Instanz ermöglicht. Um nun während der Abarbeitung auf zentral implementierte Komplex-Funktionalitäten des Simulationsmodells zugreifen zu können, wurde auch der umgekehrte Zugriffsweg implementiert. Die angesprochenen Komplex-Funktionen, zu denen beispielsweise die Pfadfindung zählt, werden dabei über die Property `HighLevelFunctions` in `CProject` verfügbar gemacht.

Die Klassen `CTask` und `CHumanTask` stellen die funktionalen Grundgerüste für allgemeine bzw. Menschmodell-spezifische Verrichtungen zur Verfügung. Verrichtungen sind generell hierarchisch aufgebaut, d.h. eine Verrichtung kann aus einer beliebigen Anzahl von untergeordneten Verrichtungen bestehen und sich damit deren Funktionalität bedienen. Um durch die hierarchischen Ebenen navigieren zu können, beinhaltet `CTask` die Properties `ParentTask` und `ListOfSubTasks`. Natürlich beinhaltet jede Verrichtung auch eine Verknüpfung zur Instanz von `CBehavior`, in der sie eingefügt wurde. Hiermit werden auf Verrichtungsebene alle zentralen Funktionalitäten verfügbar gemacht. Am häufigsten werden dabei der `LogWriter`, die Komplex-Funktionen und der Zugriff auf den Datenstrom-Manager verwendet.

Das neben den berechneten Zustandsdaten wichtigste Ergebnis der Simulation ist die berechnete Simulationszeit. Für die Zeitverwaltung gibt es eine Reihe von einzelnen Members und Properties, die miteinander in Beziehung stehen. Es ist angedacht – und bereits mit `CTaskDuration` vorbereitet – diese Menge von Einzel-Members (`Duration`, `Duration-Calculated`, `DurationUsed`, `TimeStampStart`, `TimeStampEnd`, ...) in einer eigenen Verwaltungsdatenstruktur zu kapseln, um damit dann auch den Umgang mit den in `CHumanTask` ergänzten Sollzeitbestimmungs-Strukturen zu erleichtern. Das derzeitige Haupt-

4 Softwaretechnische Umsetzung

problem ist dabei aber, dass es mehrere Art und Weisen gibt, auf die eine Kommunikation der Simulationszeiten durch die verschiedenen Ebenen der Verrichtungshierarchie hindurch erfolgen kann. Alle derzeit implementierten Möglichkeiten sind in Abbildung 36 dargestellt.

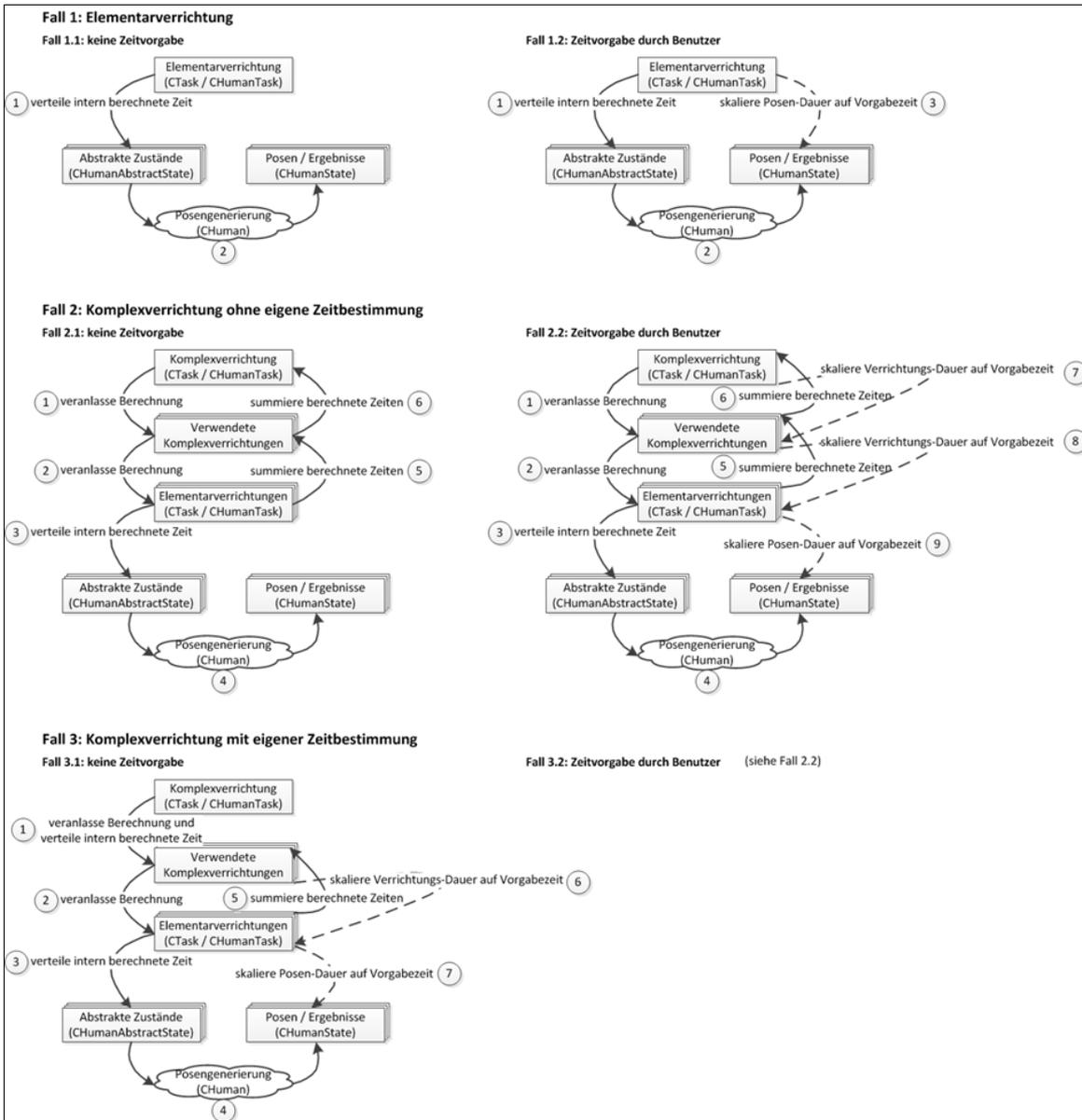


Abbildung 36: Varianten der Verrichtungszeitbestimmung

Für die Kommunikation mit der Umwelt (Framework) implementiert die Basis-klasse CTask die Schnittstelle IEMAPparameterizedEntity. Hierüber können

die einzelnen Verrichtungen dem Simulations-Rahmensystem ihre Parameter bekannt geben und diese bei der Simulation dann über die Konverter-Klasse `CParameterConverterCore` auf die internen Member abbilden. Alle Member, die in den Verrichtungsklassen von außen belegbare Parameter repräsentieren, wurden mit dem Benennungsschema „`m_pParameterName`“ versehen – dies erleichtert deren interne Unterscheidung.

Schließlich beinhalten die Verrichtungs-Basisklassen auch eine Reihe von Simulations-Grundfunktionen, welche durch die abgeleiteten Verrichtungsklassen mit der jeweiligen spezifischen Verrichtungslogik zu überladen sind. Nennenswert sind dabei insbesondere die beiden Methoden `VerifyInputParameters()` und `CalculateStates()`, welche zum einen die logische Parameterüberprüfung und zum anderen die tatsächliche Abarbeitungslogik implementieren. Alle von der Basisklasse abgeleiteten Klassen überschreiben die genannten Methoden mit eigener Logik. Dabei rufen diese aber in jedem Fall am Ende der überschriebenen Methode deren Pendant aus der Basisklasse auf, um abschließende verrichtungsunabhängige Verarbeitungsschritte durchzuführen.

4.5.4 Simulationsstrukturen

Die Bewegungsgenerierung bzw. Simulation beruht auf der Erzeugung von Einzelzuständen, die jeweils mit einer Zeit hinterlegt sind und die Bewegung eines Objektes beschreiben. Da es innerhalb des Simulationssystems verschiedene Objekttypen mit unterschiedlichen Komplexitätsgraden gibt, wurden zu deren Zustandsverwaltung auch verschiedene aufeinander aufbauende „Zustands-Klassen“ implementiert. Alle „Zustands-Klassen“ dienen in erster Linie der temporären Arbeitsdatenhaltung und implementieren darüber hinaus jeweils nur sehr eingeschränkte Funktionalitäten, wie beispielsweise Interpolations- und Kopiermethoden.

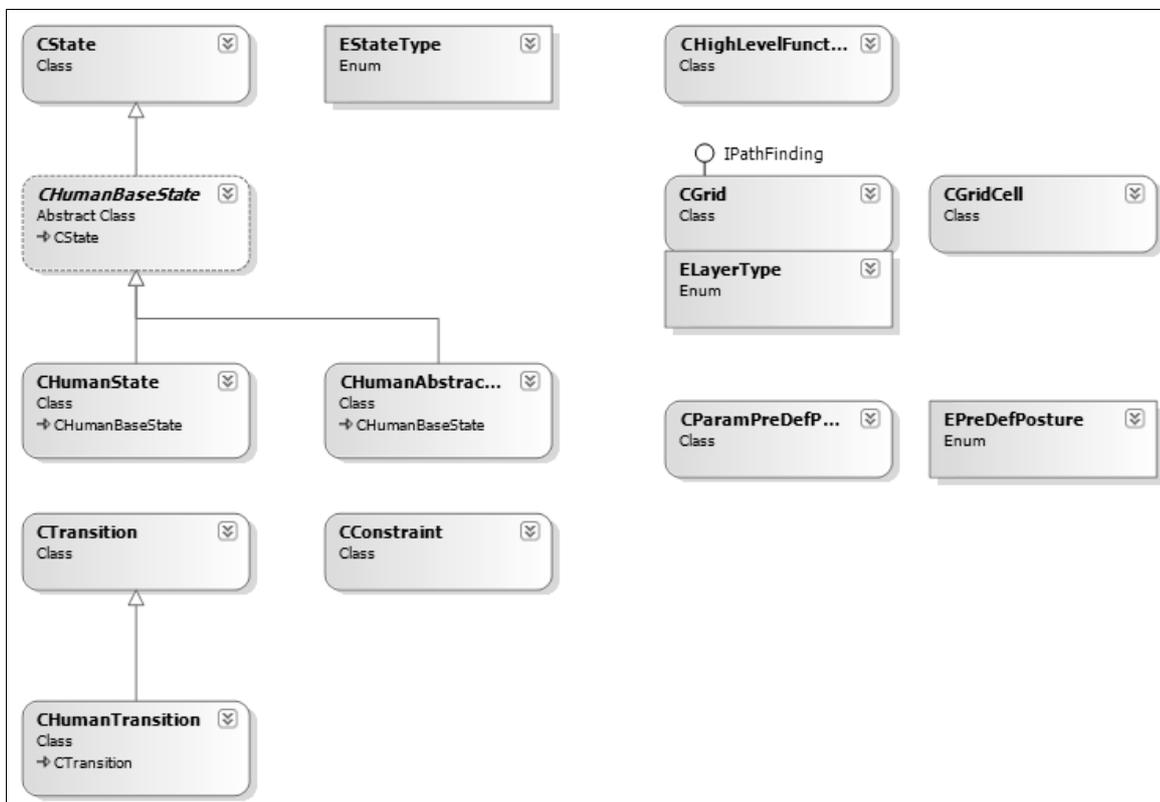


Abbildung 37: Klassendiagramm der Simulationsstrukturen

CState repräsentiert die elementarste Form einer Zustandsklasse. Sie implementiert neben den beiden Standardangaben für Ort (TargetMatrix) und Dauer (Duration) auch einige erweiterte Zustandseigenschaften, wie zum Beispiel an- oder abgehangene Objekte (m_DictionaryOfAttached-

(/DetachedObjects). Um Zustände, innerhalb derer Änderungen in den Objektverknüpfungen vorgenommen werden, zu kennzeichnen, wurde ein Member eingeführt, der den „Typ“ der jeweiligen Zustands-Instanz angibt. Diese Kennzeichnung des Typs erfolgt durch die Enumeration `EStateType` und erlaubt derzeit eine Markierung anhand der Zustandstypen `BASIC` für normale Bewegungszustände, `ATTACH` für Zustände, innerhalb derer Objekte angehängt werden und `DETACH` für Zustände, innerhalb derer Verknüpfungen gelöst werden. Diese Markierung wird insbesondere beim späteren Export der Zustände relevant, da die unterschiedlichen Zustandstypen bei der Simulationswiedergabe unterschiedlich gehandhabt werden.

Bemerkenswert bei allen Zustandsklassen ist die fehlende direkte Verknüpfung der Zustands-Instanzen zu dem Objekt, dessen Zustand sie repräsentieren. Diese Tatsache unterstreicht die Funktion der Zustandsklassen als reine Datenhaltungsklassen. Eine Beziehung zu einem Objekt wird lediglich beim Speichern seiner Daten bzw. beim Anwenden eines Zustandsdatensatzes auf ein Objekt (temporär) hergestellt. Diese Entkopplung von Objekten und Zuständen erlaubt die Mehrfachverwendung von Zuständen bzw. die gemeinsame Nutzung von Zuständen durch unterschiedliche Objekte.

Die Klasse `CHumanBaseState` wurde erst nachträglich eingeführt, um die gemeinsamen Eigenschaften der nachfolgend beschriebenen Klassen `CHumanState` und `CHumanAbstractState` in einer gemeinsamen Basisklasse zu vereinen. Wie dem Namenszusatz „Human“ zu entnehmen ist, handelt es sich jeweils um Zustands-Klassen zur Verwendung mit einem Menschmodell. Zu den Gemeinsamkeiten zählen zum einen die im Zustand einzunehmenden Handposen (bzw. Griffhaltungen). Zum anderen werden zwei Strukturen implementiert, die an- und abgehangene Objekte in Beziehung zu den jeweils verwendeten Körperteilen verwalten können.

Mit der Klasse `CHumanState` wurde das eigentliche Pendant in Sachen Zustandsverwaltung zur Klasse `CState` für Menschmodelle implementiert. Um zu vermeiden, dass Zustände für jedes Körperteil eines Menschmodells einzeln geführt werden müssen, besteht die Hauptaufgabe von `CHumanState` darin, die Teilzustände der Körperteile des Menschmodells zu aggregieren und sie in einer einzelnen Zustands-Instanz zu vereinen. Zu diesem Zweck beinhaltet `CHumanState` eine Reihe von Listenstrukturen, welche jeweils alle entsprechenden Werte eines Körperteils speichern. Hierbei sei explizit darauf hingewiesen, dass

die Zustands-Instanz jeweils alle Werte aller Körperteile speichert und nicht etwa nur diejenigen Werte, die sich geändert haben. Dadurch wird erreicht, dass die Zustände in ihrer Reihenfolge zueinander unabhängig sind, was deren Wiederverwendbarkeit ermöglicht.

Als Transformationseigenschaften der einzelnen Körperteile werden derzeit sowohl die Matrizen als auch die aktuellen DOF-Werte jedes Körperteils gespeichert. Die Verwaltung beider Transformationsangaben resultiert noch aus den Anfängen der Entwicklung und ist prinzipiell als redundant anzusehen.

Neben den körperteilspezifischen Informationen beinhaltet `CHumanState` noch einen Satz Metainformationen, welche zum einen für die Ansteuerung des Oberkörpers (siehe auch Kapitel 3.6.4) und zum anderen zur Zustandsverwaltung der Beine dienen. Die zuletzt genannte Information ist erforderlich, weil sich die Einordnung mancher Beinstellungen nur „von außen“ über die Verrichtungslogik vornehmen lässt (z.B. eindeutige Unterscheidung von Sitzen und Hocken). Schließlich beinhaltet `CHumanState` auch noch einen Member, der eine Struktur vom Typ `CErgonomicData` enthält. Hierüber können den einzelnen Zuständen ergänzende ergonomisch relevante Information mitgegeben werden, welche später über das Simulations-Rahmensystem bzw. ein angekoppeltes Ergonomie-Modul ausgewertet werden können.

Während die Klasse `CHumanState` zur Verwaltung von direkt auf das Menschmodell anwendbaren „Ergebnis-Zuständen“ dient, werden Instanzen der Klasse `CHumanAbstractState` für die Datenversorgung des Bewegungsgenerators verwendet. Das bedeutet, dass Verrichtungen zunächst Objekte vom Typ `CHumanAbstractState` generieren, die sodann auf ein Menschmodell angewendet und schließlich durch den Posen-Generator in Ergebnis-Zustände vom Typ `CHumanState` übersetzt werden. Instanzen vom Typ `CHumanAbstractState` stellen damit die Schnittstelle zwischen Verrichtung und Posen-Generator dar (siehe auch: Abschnitt 3.3 und Abbildung 23).

Den Kern eines Objektes vom Typ `CHumanAbstractState` stellt eine Auflistung von sogenannten »Constraints« (`CConstraint`) dar, die jeweils mit einem bestimmten Körperteil (momentan nur Endeffektoren, also: Kopf, Hände, Füße) verknüpft sind. Aufgabe des Posen-Generators ist es, später aus diesen Zielvorgaben die tatsächlichen Haltungen der einzelnen Körperteile zu berechnen.

Neben den »Constraints« beinhaltet jede Instanz von `CHumanAbstractState` noch eine Reihe von Steuerungsinformationen, die später ebenfalls vom Posen-

Generator ausgewertet werden können. Hierzu zählen insbesondere Status-Flags, die angeben, ob „unbenutzte Körperteile“ (= nicht per Constraint angesteuert) in ihren „Normal-Zustand“ überführt werden sollen oder ihre momentane Stellung beibehalten sollen.

Bei den beiden Klassen `CTransition` und `CHumanTransition` handelt es sich um zwei Hilfsklassen, die die Arbeit mit Interpolationen entlang von Pfaden erleichtern sollen. Für ihre Arbeit benötigen die Klassen vier Informationen:

- einen Pfad (`CTransition`) bzw. mehrere Pfade (`CHumanTransition`)
- eine Zeitvorgabe (Dauer)
- die Angabe eines Verlaufstyps in Form einer Bezierkurve
- eine Zeitschrittweite für die Interpolationsschritte

Mit Hilfe dieser Eingangsinformationen können automatisch Listen von interpolierten Einzel-zuständen generiert werden. Nach demselben Prinzip arbeitet auch die Klasse `CHumanTransition`, wobei diese in Erweiterung von `CTransition` statt einem einzigen mehrere Pfade verwaltet, welche wiederum einzelnen Körperteilen zugeordnet sind. Die Pfade werden intern gemäß der gegebenen Zeitschrittweite und des Geschwindigkeitsverlaufs unterteilt, um sie nachfolgend zur Zustandsgenerierung einsetzen zu können.

4.6 Spezielle geometrische Algorithmen

4.6.1 Laufwegs-Bestimmung

Nahezu jede Simulation menschlicher Arbeit (abgesehen von Sitz-Arbeitsplätzen) beinhaltet Ortswechsel des digitalen Menschmodells. Diese Ortswechsel werden zumeist durch Laufbewegungen ausgeführt. Neben der expliziten Vorgabe des zu wählenden Laufpfades erfordern komplexere Verrichtungen die Möglichkeit zur automatischen Pfadbestimmung. Die Eingangsinformationen für die entsprechenden Berechnungen sind dabei zum einen die derzeitige Startposition und die gewünschte Zielposition, zum anderen aber auch Informationen über die Umgebung des digitalen Menschmodells und dabei insbesondere Angaben zu relevanten Kollisions- und Bodenobjekten (siehe Kapitel 3.4.1).

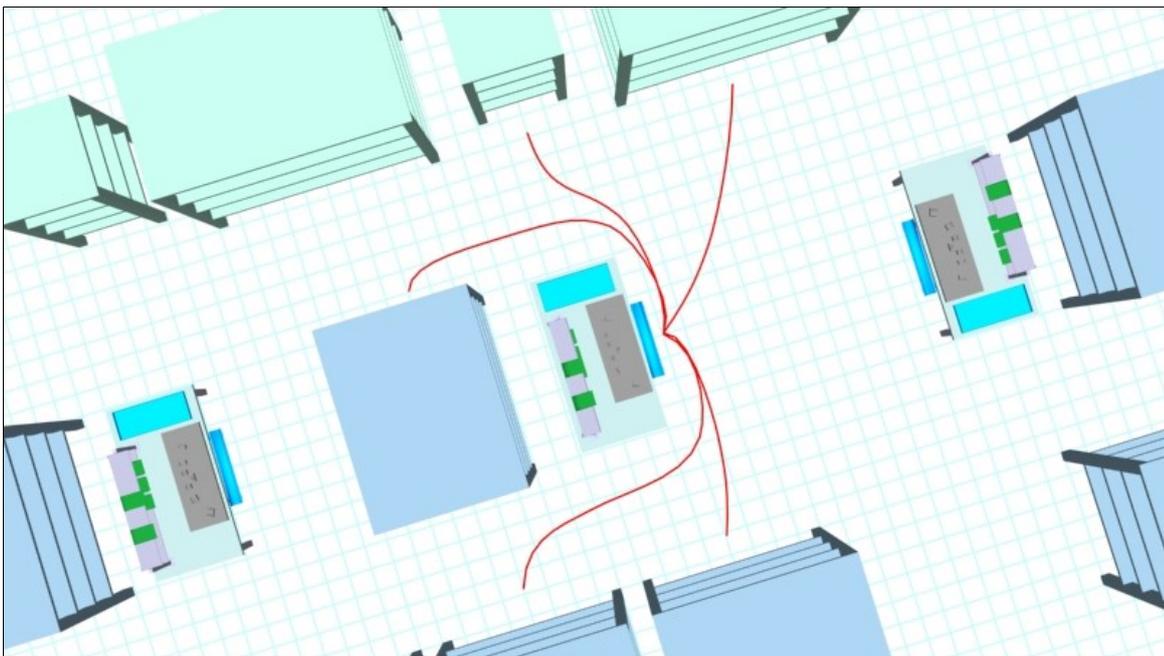


Abbildung 38: Visualisierung von Laufwegen in einem Beispiel-Layout

Der Lösungsraum der theoretisch möglichen Pfade zwischen einem gegebenem Start und Ziel ist unendlich groß. Da es sich beim vorliegenden Simulationssystem unter anderem auch um ein Werkzeug zur Prozessoptimierung handelt, sollte es auch dazu in der Lage sein, den jeweils kürzesten Pfad zu ermitteln. Bei der Betrachtung der möglichen Algorithmen entschied sich der Autor für den

»A*-Algorithmus«. Es handelt sich dabei um einen sogenannten heuristischen Suchalgorithmus, der zudem die optimale Lösung findet, sofern es denn eine Lösung gibt. [66, p. 104ff]

Der Suchalgorithmus arbeitet auf Graphenstrukturen, wobei den Kanten der Graphen entsprechende Gewichte zugeordnet sind, welche wiederum zur Bestimmung der Pfadlänge verwendet werden. Der Autor implementierte den A*-Algorithmus in einem ersten Schritt auf einem »gleichmäßigen Raster«. Dabei wurde die 3D-Umgebung zunächst in gleichgroße Gitterzellen unterteilt, welche anschließend als „frei“ oder durch „Kollisionsobjekte belegt“ markiert wurden. Die Nachbarschaftsbeziehungen zwischen den Gitterzellen, die in einem indizierten mehrdimensionalen Feld abgebildet wurden, sind trivial und auch die Zugriffszeiten auf bestimmte Zellen bewegen sich im Rahmen von $O(1)$. Dagegen ist die Speicherbelegung durch die verhältnismäßig hohe Anzahl von Gitterzellen, welche sich zudem negativ auf die Laufzeit des Algorithmus auswirkt, der größte Nachteil dieser Datenstruktur. Nichtsdestotrotz entstand eine erste Version eines funktionierenden Moduls zur Bestimmung des kürzesten kollisionsfreien Laufpfades.

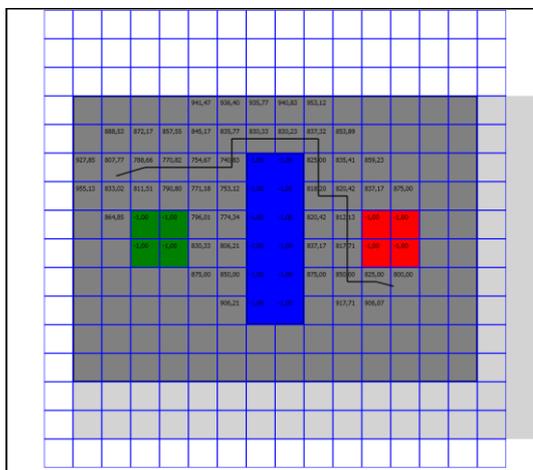


Abbildung 39: Raumunterteilung mit »gleichmäßigem Raster« [67, p. 39]

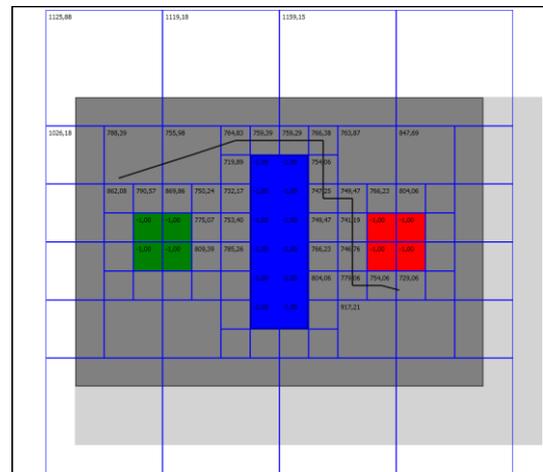


Abbildung 40: Raumunterteilung mit »QuadTree-Struktur« [67, p. 40]

In seiner Bachelorarbeit optimierte Knorr [67] den vorhandenen Algorithmus. Hierzu löste er zunächst die Abhängigkeit von Algorithmus und Datenstruktur durch die Einführung einer entsprechenden Schnittstelle. Den Algorithmus selbst verbesserte er durch die Implementierung einer Binärbaum-Struktur, wel-

che er zur Verwaltung der »Liste offener Knoten« verwendete. Der Binärbaum ermöglicht den Zugriff mit $O(1)$ auf den nächsten zu untersuchenden Knoten mit dem bis dato kürzesten Teil-Pfad. Darüber hinaus implementierte Knorr als Alternative zum gleichmäßigen Raster eine sogenannte »OctTree-Struktur« (bzw. »QuadTree-Struktur« im 2D-Raum). Diese behebt die Nachteile des zuerst implementierten Rasters (Anzahl der Zellen, Speicherbelastung) auf Kosten einer komplexeren Initialisierungsphase (aufgrund der rekursiven Zellunterteilungen) und aufwändiger zu bestimmender Nachbarschaftsbeziehungen zwischen den einzelnen Knoten. In abschließenden Tests zum Laufzeit- und Speicherbedarfsverhalten konnten die Verbesserungen zweifelsfrei nachgewiesen werden [67, p. 38ff].

Der Aufbau der entsprechenden Datenstruktur samt Einordnung der als Hindernisse markierten Objekte erfolgte zunächst ausschließlich in einem einzigen Initialisierungsschritt. Damit konnten anfangs nur statische Kollisionsobjekte berücksichtigt werden. Vor diesem Hintergrund ergänzte der Autor das Pfadbestimmungsmodul in einer weiteren Ausbaustufe um die Möglichkeit, dynamische Objekte zu verwalten. Hierzu wurde die bisherige Information, dass eine Zelle besetzt ist, um die Angabe von Zeitscheiben (Wann ist die Zelle besetzt?) erweitert. Zudem muss dem Aufruf der Pfadbestimmung nun ein entsprechender Zeitpunkt mitgegeben werden. Hierdurch können beispielsweise Szenarien am Fließband bzw. die Montage am bewegten Fahrzeug simuliert werden.

4.6.2 Arbeitspositions-Bestimmung

Ein weiterer wichtiger Aspekt der Simulation menschlicher Arbeit ist die Positionierung des Menschmodells zur nachfolgenden Ausführung einer bestimmten Tätigkeit. Dieser bestimmte Standort wird vom Autor im Rahmen des Simulationssystems als »Arbeitsposition« bezeichnet. Der einfachste Weg zur Bestimmung der Arbeitsposition war zunächst der einer Eingabe des Benutzers. Die Definition der Arbeitsposition erfolgte also schlicht durch einen Mausklick an den gewünschten Standort und einen zweiten Klick zur Bestimmung der Blickrichtung am Zielort. Dieses Prinzip funktionierte problemlos, ließ allerdings einiges an Flexibilität bei Änderungen des Umgebungs-Layouts vermissen. Insbesondere bei Tätigkeiten, die einen Objektbezug beinhalteten, musste die Arbeitsposition nach entsprechenden Änderungen des Layouts angepasst oder neu gesetzt werden.

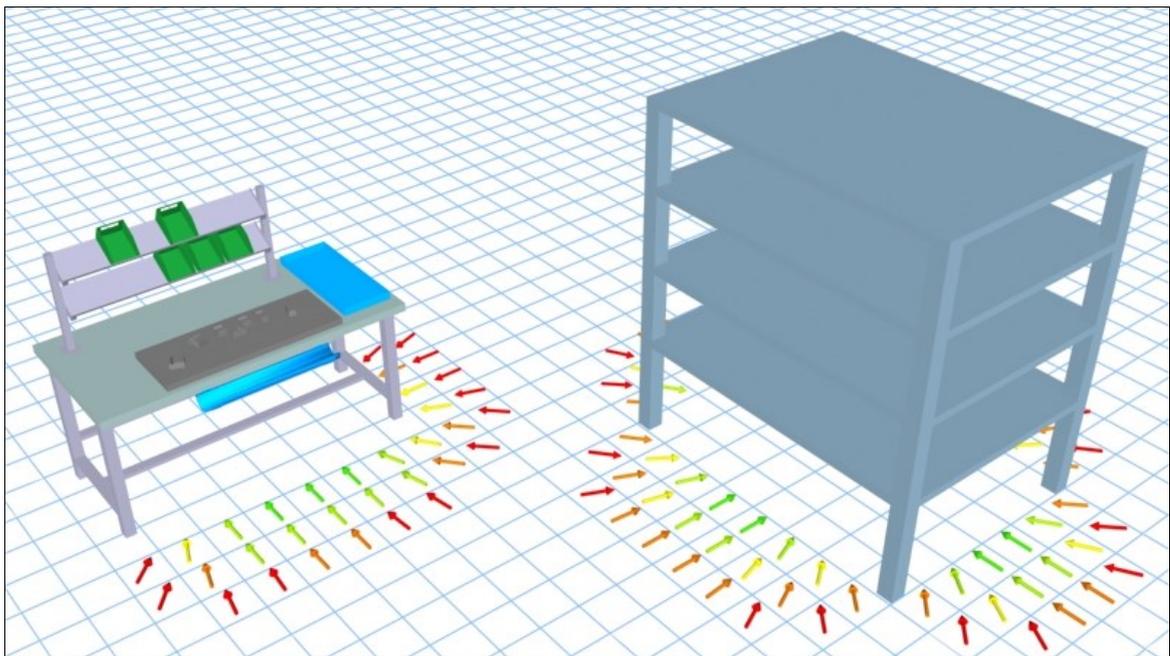


Abbildung 41: Schematische Darstellung potentieller Arbeitspositionen

Vor dem Hintergrund des Anspruchs, einen virtuellen Facharbeiter abbilden zu wollen, entwickelte der Autor daher einen ersten Algorithmus, mit dessen Hilfe ein „Laufen zum Objekt“ abgebildet werden konnte. Dieser beruht ebenfalls auf dem Prinzip des kürzesten Wegs zum Ziel. Da es um das Zielobjekt herum wieder unendlich viele potentielle Arbeitspositionen gibt, wählte der Autor aller-

dings die Ausrichtung des Objekt-Hüllquaders als einschränkendes Kriterium. In der Mitte jeder Seitenfläche des Hüllquaders wurde ein 3D-Punkt definiert. Anschließend wurden alle Punkte auf den Boden projiziert und um einen Sicherheitsabstand vom Zentrum des Hüllquaders ausgehend nach außen verschoben. Der verwendete Sicherheitsabstand ist dabei grundsätzlich abhängig von der Anthropometrie des verwendeten Menschmodells, kann aber auch vom Benutzer durch einen entsprechenden Skalierungsfaktor beeinflusst werden. Zur abschließenden Bestimmung der Ausrichtung der potentiellen Arbeitspositionen, genügten dann einfache Vektor-Operationen, um aus der globalen „Nach-Oben-Achse“ und der gegebenen Blickrichtung (Arbeitsposition \rightarrow Hüllquader-Mitte) die noch fehlende dritte Achse für die Transformationsachse zu bestimmen.

Die berechneten Arbeitspositionen werden hinsichtlich ihrer „Betretbarkeit“ geprüft und alle betretbaren Positionen werden sodann an das Modul zur Laufpfadbestimmung übergeben. Im Anschluss wird die Arbeitsposition gewählt, zu der der kürzeste Laufweg führt. Sollte kein einziger gültiger Pfad ermittelt werden können, wird die Prozedur noch einmal mit Punkten wiederholt, deren Ursprung in den Eckpunkten des Objekt-Hüllquaders liegt; im erneuten Fehlerfall erfolgt eine weitere Iteration mit einem erweiterten Sicherheitsabstand.

Mit dieser ersten Stufe konnte ein stark erhöhtes Maß an Flexibilität bei Anpassungen des 3D-Layouts erreicht werden. Durch den hergestellten Objektbezug war die direkte Vorgabe von Arbeitspositionen nur noch in Ausnahmefällen erforderlich. Alle Vorrichtungen, die einen Objektbezug beinhalteten, wurden nach und nach um eine Option zum „automatischen Laufen“ erweitert. Bei der Anwendung des Algorithmus zeigte sich im Übrigen, dass dieser sehr gut mit der Arbeitsweise der parametrischen Tätigkeitsbeschreibungen anhand von Vorrichtungen harmonisierte.

Darüber hinaus wurde allerdings auch deutlich, dass es durchaus Situationen gab, in denen noch etwas mehr „Intelligenz“ bei der Wahl der Arbeitsposition erforderlich war. Dies bewog den Autor zur Implementierung einer weiteren Ausbaustufe des Algorithmus, der es ermöglichen sollte, »kontextabhängige Arbeitspositionen« zu bestimmen. Gegenüber der bisherigen Arbeitspositions-Bestimmung sollte in dieser Ausbaustufe nun auch die Art der am Zielort auszuführenden Tätigkeit berücksichtigt werden. Die drei derzeit implementierten

erweiterten Algorithmen zur Arbeitspositions-Bestimmung betreffen die Verrichtungen:

- Hinsetzen und Einsteigen
- Aufnehmen und Platzieren sperriger Objekte
- Verrichtungen zur Werkzeughandhabung

Beim Hinsetzen liegt die Besonderheit hauptsächlich in der Ausrichtung mit dem Rücken zum Ziel, während beim Einsteigen eine seitliche Ausrichtung zur Einstiegsöffnung angestrebt wird. Aus diesen Rahmenbedingungen und den entsprechenden Sicherheitsabständen lassen sich zusätzliche Arbeitspositionen bestimmen, welche bei der nachfolgenden vergleichenden Bewertung bevorzugt behandelt werden.

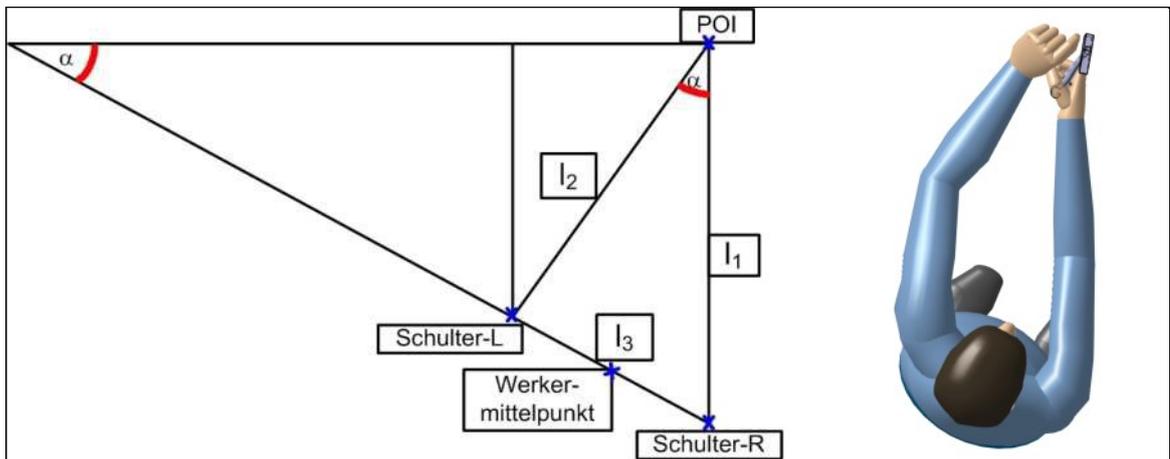


Abbildung 42: Arbeitspositionsbestimmung für Werkzeuge [69, p. 37]

Die beidhändige Handhabung sperriger Objekte stellt ebenfalls einen Sonderfall dar. Als sperrig werden dabei Objekte angesehen, bei denen eine Hauptabmessung größer 80cm ist oder zwei Hauptabmessungen größer 30cm sind. [68, p. 24] Durch die größeren Maße und die damit einhergehende tendenziell ungünstigere Masseverteilung empfiehlt sich als Arbeitsposition ein Standort, an dem die Hüfte des Menschmodells möglichst günstig zu den Greifpunkten des betreffenden Objekts bzw. Ablageortes ausgerichtet ist (z.B. Hüfte parallel zur längsten Objektseite). Dies gilt sowohl für das Aufnehmen als auch das Platzieren, wobei ein zeit- und kraftaufwändiges Umsetzen des Objekts vermieden wird.

Schließlich bildet auch die Werkzeughandhabung einen eigenen spezifischen Fall bei der Bestimmung der geeignetsten Arbeitsposition. In seiner Bachelorarbeit beschäftigte sich Geiler [69] unter anderem mit genau diesem Thema. Dabei bestimmte er sowohl die ideale Entfernung als auch die notwendige Ausrichtung für die Arbeitsposition in Abhängigkeit der Anthropometrie des verwendeten Menschmodells, der Beschaffenheit des jeweiligen Werkzeugs und der vorgegebenen Applikationsrichtung. [69, p. 36ff] Der von Geiler ermittelte Berechnungsalgorithmus wurde nahezu anpassungsfrei übernommen, da sich dieser bei den durchgeführten Tests als überaus realitätsnah und somit äußerst geeignet erwies.

4.6.3 Objekthandhabung

Im Laufe des Produktentstehungsprozesses muss der Werker zwangsläufig mit dem Produkt oder dessen Bestandteilen interagieren. Die Handhabung verschiedenster Objekte spielt somit auch bei der Simulation menschlicher Arbeit eine entscheidende Rolle. Dabei sind die beiden nach außen hin offensichtlichsten Merkmale der Objekthandhabung die Art und Weise, wie etwas gegriffen wird und die Art, wie ein kontrolliertes Objekt transportiert wird. Beide Merkmale sind wiederum abhängig von der Beschaffenheit des jeweiligen Objekts. Der erste Schritt, um Aussagen über die Handhabung eines Objekts tätigen zu können, bestand damit in dessen Analyse und anschließender Klassifizierung. Mit den theoretischen Grundlagen dieses Themas beschäftigte sich Tilch [70] in ihrer Bachelorarbeit.

Da sich vorhandene Klassifikationen nur bedingt für den Einsatz im vorliegenden Simulationssystem eigneten [70, p. 26], wurde ein neues Schema zur Klassifikation entworfen. Die neue Klassifikation stützt sich dabei primär auf die abstrahierten Abmessungen der Objekte, welche über die jeweiligen Hüllquader bestimmt werden. Tilch definierte eine Reihe von sog. Objekt-Klassen, welche sie anschließend in einen Entscheidungsbaum einordnete. Anhand der Baumstruktur dokumentierte sie nicht nur den Klassifikationsalgorithmus, sondern verknüpfte die Informationen auch mit Aussagen zur Machbarkeit der Handhabung und – so diese gegeben ist – dazu, mit wie viel Personen ein entsprechendes Objekt gehandhabt werden sollte. (Abbildung 43) Ihre Klassifikation reicherte Tilch darüber hinaus noch mit Aussagen zur bevorzugten Lage der Greifpunkte beim Tragen und der entsprechenden Greifart an [70, p. 64]. Schließlich untersuchte sie auch noch typische Tragehaltungen und Arten des Aufnehmens zu den einzelnen Objektklassen.

Natürlich muss die Vorarbeit von Tilch aufgrund der – in Abhängigkeit von der tatsächlichen Beschaffenheit der Objektgeometrie – mitunter stark vereinfachten Datengrundlage (Hüllquader) als eine erste Stufe für die Objektuntersuchungen betrachtet werden. Dennoch fand der Klassifikations-Algorithmus bereits seinen Platz in der Wissensbasis des vorliegenden Simulationssystems, wobei insbesondere die Aussagen über die Handhabbarkeit von Objekten verwendet werden, da sich diese bei ersten Tests als durchaus praktikabel erwiesen. Der Autor entschied sich in diesem Zusammenhang allerdings zunächst nur für

die Ausgabe von Warnungen, statt eines „harten“ Abbruchs der Simulation bei Verletzung der aufgestellten Handhabbarkeitsbedingungen. Denn je stärker sich das tatsächliche geometrische Volumen der verwendeten Testobjekte vom Volumen des Hüllquaders unterscheidet, desto ungenauer werden die von Tilch getroffenen Annahmen.

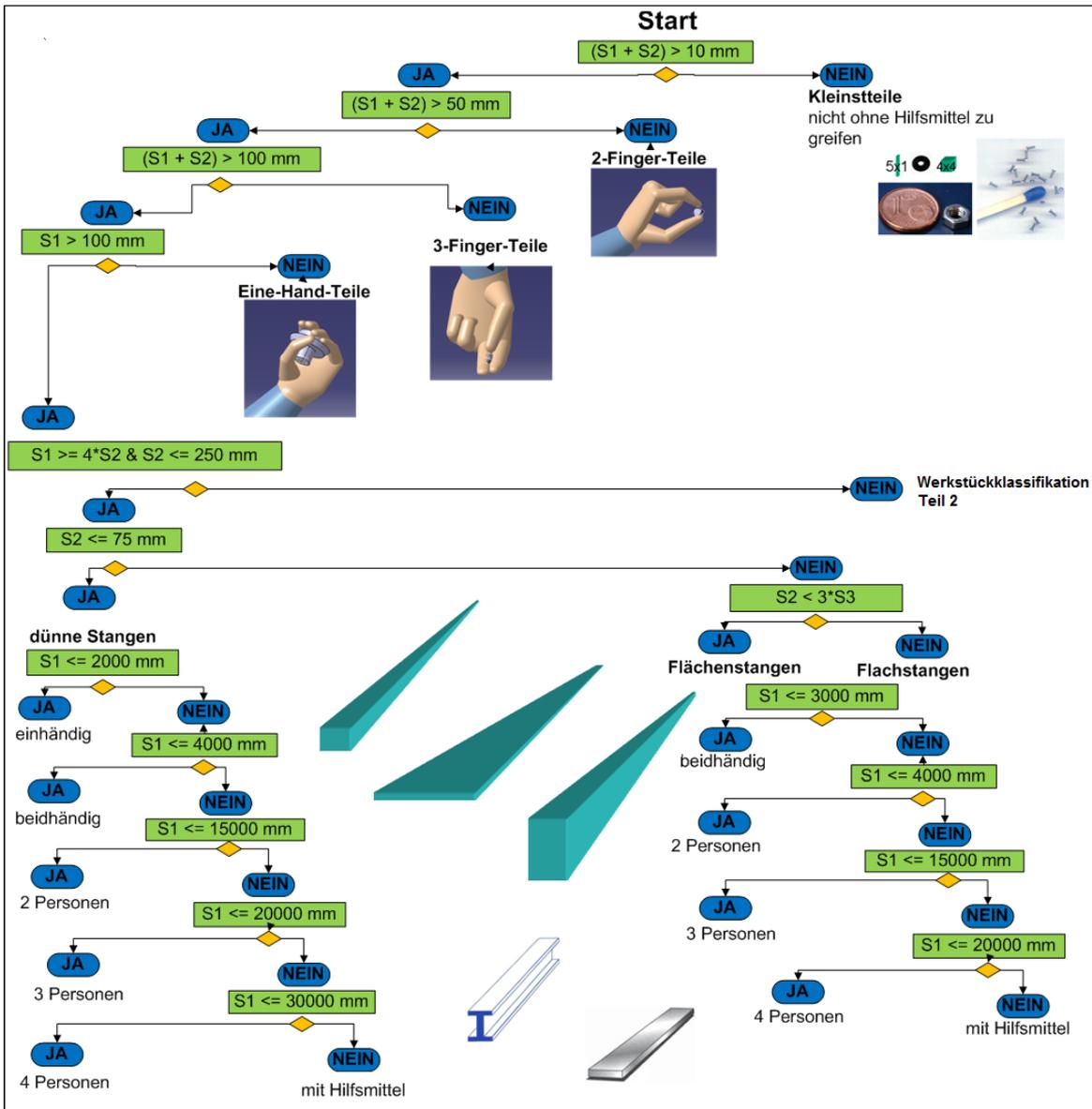


Abbildung 43: Algorithmus zur Werkstückklassifikation [70, p. A2]

Noch ausstehend ist die Implementierung der vorgeschlagenen Tragehaltungen, diese werden in Zukunft einen weiteren Beitrag zur realistischeren Anmutung

der Simulationsergebnisse leisten. Eine auf den vorliegenden Klassifikations-Algorithmus aufsetzende Untersuchung ist bereits in Vorbereitung. Hierbei soll dann vermehrt auf die tatsächliche Objektgestalt, Oberflächenbeschaffenheit und gegebenenfalls vorhandene Handhabungshilfen (Griffe, Henkel, etc.) eingegangen werden.

4.7 Tests, Ergebnisse und produktiver Einsatz

Nachdem das Simulationssystem in einer ersten stabil funktionierenden Version vorlag, stellte sich die Frage, auf welche Weise die berechneten Ergebnisse überprüft werden können.

Zunächst musste die Erfüllung der im Rahmen der Anforderungsanalyse (siehe Kapitel 3.1) ermittelten funktionalen und nichtfunktionalen Anforderungen nachgewiesen werden. Die funktionalen Anforderungen dienten als Grundlage des Systementwurfs und zur Ermittlung der Systemkomponenten. Demzufolge war die Überprüfung deren Erfüllung zunächst einmal „nur“ eine schlichte Prüfung auf Vollständigkeit in der Umsetzung. Es zeigte sich dabei, dass alle Anforderungen durch entsprechende Schritte im konzipierten Arbeitsablauf abgedeckt und durch entsprechende Systemkomponenten und Benutzeroberflächenelemente implementiert werden. In Tabelle 11 sind die funktionalen Kernanforderungen noch einmal aufgelistet und den umgesetzten Komponenten und Funktionen gegenübergestellt:

Funktionale Anforderung	Umsetzung
Definition bzw. Erzeugung digitaler Menschmodelle	Workflow: »Layout-Bearbeitung« Komponente: Datenhaltung
Definition der Arbeitsumgebung der digitalen Menschmodelle	GUI: Registerkarte Objekte
Definition der Tätigkeitsbeschreibung der digitalen Menschmodelle	Workflow: »Definition der Tätigkeiten« Komponente: Datenhaltung GUI: Registerkarte Verhalten
„Übersetzung“ der Tätigkeitsbeschreibung in Bewegungen	Workflow: »Simulation« Komponente: Simulationskern
Visualisierung der erzeugten Bewegungen	Workflow: »Visualisierung« Komponente: 3D-Visualisierung GUI: 3D-Visualisierungsbereich
Bewertung und Dokumentation der erzeugten Bewegungen	Workflow: »Definition der Tätigkeiten« Komponente: Auswertungskomponente GUI: Registerkarten Ergonomie, Ergebnisse

Tabelle 11: Umsetzung der funktionalen Anforderungen

Die Prüfung der nichtfunktionalen Anforderungen war im Vergleich dazu weniger einfach zu realisieren, da sich einige der zu überprüfenden Aspekte nur bedingt objektiv einschätzen ließen. Vor diesem Hintergrund wurden verschiedene Methoden genutzt, um die einzelnen Punkte zu überprüfen. Der Aspekt der »Benutzbarkeit« mit dem Ziel der schnellen Erlernbarkeit und hohen Akzeptanz des Simulationssystems ist ein konkretes Beispiel für ein überaus subjektives Kriterium. Zur Einschätzung nutzte der Autor in mehreren Entwicklungsstadien die Möglichkeit der Präsentation des Systems im Kreis der Planungsingenieure der imk automotive GmbH. Anhand der Erkenntnisse aus den anschließenden Diskussionen konnte das System fortwährend weiterentwickelt werden. Dabei wurden stets die Einfachheit der Bedienung und der methodische Ansatz des Systems gelobt. Alle Formen von Verbesserungsvorschlägen und aufgetretenen Fehlern wurden darüber hinaus in Form strukturierter Listen dokumentiert, was ebenfalls einen großen Beitrag zur iterativen Weiterentwicklung und Optimierung des Systems leistete.

Eine objektiv bewertbare nichtfunktionale Anforderung war der Aspekt der »Effizienz« im Umgang mit dem System. Das erklärte Ziel bei der Entwicklung war die drastische Reduktion des Aufwands zur Erstellung einer Simulation mit digitalen Menschmodellen. Im Durchschnitt lag das Verhältnis der Erstellungszeit zur simulierten Zeit bisher bei 250 zu 1, was einem Aufwand von reichlich 4 Stunden für 1 Minute Simulationszeit gleichkam. [8, p. 126] In Untersuchungen des vorliegenden Systems konnte eine Halbierung des ursprünglichen Aufwands der Bewegungsmodellierung ermittelt werden. [71, p. 333] Besonders große Zeiteinsparungen treten bei nachträglichen Änderungen einmal modellierter Szenarien auf. Hier genügt zumeist die einfache Anpassung des Layouts, um durch die vorhandenen Objektbezüge entsprechend angepasste Bewegungen für das Menschmodell zu erzeugen.

Die Anforderung der »Flexibilität« sollte von vornherein durch eine möglichst „offene“ Systemarchitektur sichergestellt werden. „Offen“ bedeutet dabei, dass die Integration weiterer Systeme und Methoden zur Planung menschlicher Arbeit ermöglicht werden sollte. Tatsächlich wurde das vorliegende System zunächst selbst als eine Plug-In-Lösung für das Planungssystem DELMIA V5 entworfen und erst später zu einer eigenständigen Software entwickelt. Die anfangs definierten Schnittstellen und die Aufteilung der Funktionen auf die entsprechenden Komponenten erwiesen sich dabei als überaus geeignet, um beide

Ansätze (Plug-In- und Standalone-Version) zu realisieren. Darüber hinaus konnten die vorhandenen Komponenten bereits durch weitere kundenspezifische Module ergänzt werden. Ein Beispiel hierfür ist die Einbindung der Ergonomie-Bewertungsmethode »APSA« der Audi AG. Das vorliegende System ist somit aus Sicht des Autors auch äußerst flexibel gestaltet worden.

Bei der Anforderung »Skalierbarkeit«, die zur Ansteuerung beliebig vieler digitaler Menschmodelle durch das Simulationssystem dient, war ein sogenannter „Stresstest“ das Mittel der Wahl, um den entsprechenden Nachweis zu erbringen. Hierbei wurden mehrere Testszenarien erstellt, welche jeweils mit unterschiedlichen Mengen von digitalen Menschmodellen und zugeordneten Verrichtungen konfiguriert wurden. Das komplexeste getestete Szenario beinhaltete 15 digitale Menschmodelle, die insgesamt mit über einhundert Verrichtungen angesteuert wurden. Diese Szenarien wurden nun auf mehreren Testsystemen ($n > 10$) simuliert, wobei sich die Testsysteme in Hardware- und Softwareausstattung deutlich unterschieden. Deren Bandbreite reichte dabei vom Standard-Office-PC mit Onboard-Grafikkarte bis zur CAD-Workstation mit High-End-Grafikkarte. Trotz der Vielzahl gleichzeitig zu visualisierender beweglicher Objekte ließ sich das Ergebnis der Simulation noch problemlos mit Bildwiederholraten von mehr als 18 Bildern pro Sekunde auf dem leistungstärksten Testrechner wiedergeben (Hardwareausstattung des Testsystems siehe Tabelle 12). Aus den durchgeführten Tests konnte die Erkenntnis abgeleitet werden, dass die geforderte »Skalierbarkeit« des Simulationssystems gegeben ist und die Anzahl der gleichzeitig simulierbaren und darstellbaren digitalen Menschmodelle nicht durch das Simulationssystem selbst, sondern primär durch die zugrundeliegende Hardware beschränkt ist.

Ein schwierig nachzuweisender Aspekt war die »Korrektheit« der Simulationsergebnisse. In der Anforderungsanalyse wurden insoweit korrekte Ergebnisse in Form „plausibler und humanmotorisch korrekter Bewegungsabläufe“ der digitalen Menschmodelle eingefordert, da sie Vorbedingungen für die nachfolgende Bewertung der Ergebnisse darstellen (siehe Abschnitt 3.2). Aufgrund des großen Umfangs der entstandenen Verrichtungsbibliothek und der nahezu beliebigen Kombinierbarkeit und Parametrisierbarkeit der entwickelten Verrichtungen schied ein umfassender alle Eventualitäten abdeckender Test nach dem Prinzip „jede Verrichtung in jeder Parametrisierung in Verknüpfung mit jeder anderen Verrichtung“ von vornherein aus. Stattdessen entwickelte der Autor eine Viel-

zahl konkreter und realitätsnaher Testszenarien, welche die Verrichtungen in kleineren logischen Gruppen mit typischen Parameterkonfigurationen in wiederum typischen 3D-Umgebungen zur Anwendung brachten. (Abbildung 44) Hauptaugenmerk bei der Erstellung der Szenarien war insbesondere der Praxisbezug und eine vollständige Abdeckung der Verrichtungsbibliothek durch die Summe aller Testszenarien (d.h. jede Verrichtung wird mindestens einmal verwendet). Nachdem die Testszenarien erstellt worden waren, erfolgten „Korrektheits-Tests“ in verschiedenen Ausprägungen. Zunächst wurden alle Simulationsergebnisse durch erfahrene Planungsingenieure erfolgreich auf visuelle Plausibilität hin überprüft. Dabei wurde insbesondere Wert auf die glaubhafte Anmutung der resultierenden Bewegungen gelegt, welche sich durch realistische Bewegungsgeschwindigkeiten und anatomisch-korrekte Körperhaltungen ergab. Zudem traten keinerlei Unstetigkeiten („Sprünge“) im zeitlichen Bewegungsverlauf auf. Als weiteres Mittel zur Verifizierung der Ergebnisse dienten darüber hinaus Videoaufnahmen und Motion-Capturing-Daten, welche zu bestimmten Tätigkeiten bereits vorlagen, im Übrigen aber extra angefertigt wurden. [57] Die Überprüfung anhand der Videoaufnahmen erfolgte durch Einzelbild-Vergleiche mit den entsprechenden Ergebnisdaten der Simulation.

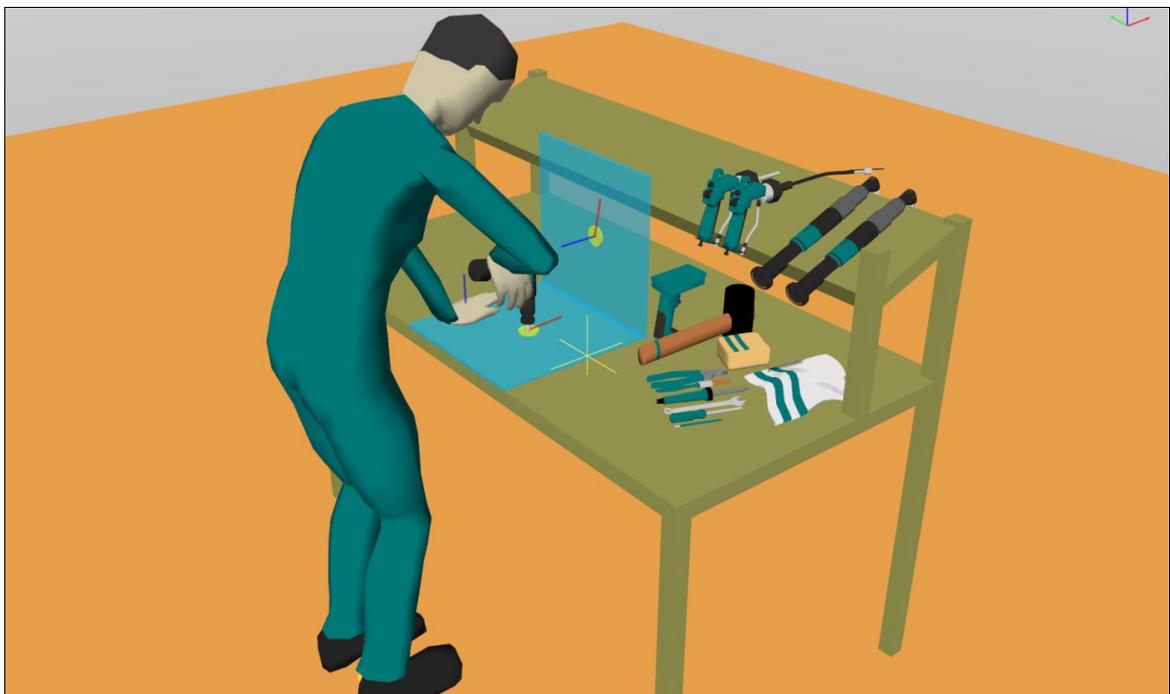


Abbildung 44: Testszenario zur Werkzeughandhabung

Nichtfunktionale Anforderungen	Methodik zur Überprüfung & Ergebnisse
<p>Benutzbarkeit: angebotene Funktionen und verwendete Begrifflichkeiten müssen mit Denkwelt des Planungsingenieurs korrespondieren (für schnelle Erlernbarkeit & hohe Akzeptanz)</p>	<p>Einschätzung durch Präsentationen mit Feedback-Runden, Experteninterviews, Pilotprojekte, Kundenzufriedenheitsbefragungen → intuitive Bedienbarkeit wurde bestätigt</p>
<p>Effizienz: Erstellung von Arbeitsumgebung und Tätigkeitsbeschreibung sollte möglichst einfach und intuitiv möglich sein (für hohe Produktivität)</p>	<p>Zeitlicher Vergleich zu bisherigen Methoden der Erstellung von Menschmodell-Simulationen → Halbierung des Modellierungsaufwands wurde erreicht [71, p. 333]</p>
<p>Flexibilität: System sollte vorhandene Standards der Planungswelt verwenden und offen für spätere Ergänzung / Anbindung zukünftiger Standards und Methoden sein</p>	<p>Konzeption und Umsetzung erster Zusatzfunktionen in Form von Plug-Ins, erfolgreiche Ein- und Anbindung in Fremdsysteme → Systemarchitektur erlaubte nahtlose Integration neuer Funktionalitäten</p>
<p>Skalierbarkeit: System sollte in der Lage sein eine beliebige Anzahl digitaler Menschmodelle gleichzeitig anzusteuern</p>	<p>erfolgreicher Stresstest mit 15 digitalen Menschmodellen in einer Simulation durchgeführt (Intel i7 Q82 mobile, 16GB RAM, Nvidia Quattro FX 1800M, Windows 7 64-Bit) → begrenzender Faktor ist allein die Hardwareausstattung des Simulationsrechners</p>
<p>Korrektheit: Erzeugung plausibler und humanmotorisch korrekter Bewegungsabläufe für digitale Menschmodelle ist Grundbedingung für nachfolgenden Aus- und Bewertungsfunktionalitäten</p>	<p>Überprüfung der Menschmodell-Bewegungen in zahlreichen Testszenarien durch visuelle Plausibilität, Einzelbild-Vergleich mit Videoaufzeichnungen und Motion-Capturing-Daten, Überprüfung der Bewegungen und zeitlichen Bewertung in Pilotprojekten → Bewegungsplausibilität konnte bestätigt werden</p>

Tabelle 12: Verifizierung der nichtfunktionalen Anforderungen

In einer weiteren Testphase wurden (durch die Simulation) ermittelte Richtzeiten mit per Hand erstellten MTM-Sollzeitanalysen verglichen. Hierbei lagen die Abweichungen beim Vergleich mit dem Verfahren MTM1 bei +/- 5% und beim Verfahren MTM-UAS bei +/-10%. Die Tatsache, dass es überhaupt Abweichungen gibt und diese bei MTM-UAS etwas größer ausfallen, liegt im MTM-Verfahren an sich. Denn einerseits resultieren die zugrundeliegenden Informationen aus statistisch gemittelten Zeitwerten; andererseits werden sie bei der Anwendung aus Gründen der Vereinfachung in diskrete Wertebereichsstufen unterteilt, was zu weiteren Ungenauigkeiten führen kann. (siehe Kapitel 3.5.3) Da neben der Ausführungszeit auch die Ergonomiebewertung ein wichtiges Auswertungsinstrument darstellt, wurden dazu ebenfalls entsprechende Tests durchgeführt. Das implementierte Ergonomie-Bewertungssystem »EAWS« [30] bewertet einen Tätigkeitsverlauf primär nach den auftretenden Körperhaltungen und deren zeitlichen Verteilungen. Ergänzend gehen Informationen zu aufgebrauchten Kräften und gehandhabten Lasten in die Bewertung ein. Da nun insbesondere die Körperhaltungen ein direktes Ergebnis der Bewegungsgenerierung darstellen, erschien auch die Ergonomiebewertung als ein weiteres geeignetes Instrument zur Verifizierung der Simulationsergebnisse. Erneut wurden entsprechende Testszenarien aufgebaut – diesmal jedoch mit dem Fokus auf die möglichst umfassende Abdeckung der ergonomischen Bewertungskriterien und Regeln. Auch in dieser nunmehr dritten Testphase konnte die Korrektheit der Simulationsergebnisse durch deren Vergleich mit per Hand erstellten Bewertungen nachgewiesen werden. Die vorhandenen Abweichungen in der Bewertung lagen aus Sicht der Tester weniger in der Software, als primär in der Methodik des Verfahrens »EAWS« begründet, welches zwar als „Screening-Verfahren“ gedacht ist, aufgrund der Ermittlung zahlreicher Schätzwerte (z.B.: „prozentualer Anteil einer bestimmten Körperhaltung in analysierten Zeitraum“) bei manueller Anwendung jedoch nur bedingt objektive und reproduzierbare Ergebnisse liefert. [72]

Dass es sich bei dem vorliegenden Simulationssystem um eine innovative Simulationssoftware handelt, welche die im Vorfeld aufgestellten Anforderungen erfüllt, konnte im Ergebnis aller beschriebenen Tests bestätigt werden. Ob das System jedoch auch die tatsächlichen Aufgaben bewältigen konnte, welche in der Praxis anfallen, war noch nachzuweisen. Insoweit kam dem Autor entgegen, dass die Firma imk automotive GmbH die Software bereits in einem sehr frühen

Status in Planungsprojekten eingesetzt hatte. Die erfahrenen Planungsingenieure von imk, die bereits an der Konzeption mitgearbeitet hatten, waren damit die ersten Anwender des Systems. Um Erfahrungen mit dem System zu sammeln und einen gewissen Bekanntheitsgrad zu erreichen, wurde der »Editor menschlicher Arbeit« zunächst in Projekten eingesetzt, innerhalb derer ohnehin manuelle Arbeitsplätze auf klassische Art und Weise geplant werden mussten. Die Ergebnispräsentationen anhand von 3D-Simulationen boten dabei auch nach Meinung der betreffenden Auftraggeber einen echten Mehrwert gegenüber den traditionellen Ergebnisformularen. Während der Bearbeitung der Projekte und dem Aufbau der Szenarien konnten außerdem zahlreiche Verbesserungsvorschläge und auch einige Systemfehler dokumentiert werden, welche im Zusammenspiel mit den vorliegenden Beispieldaten sehr schnell behoben werden konnten.

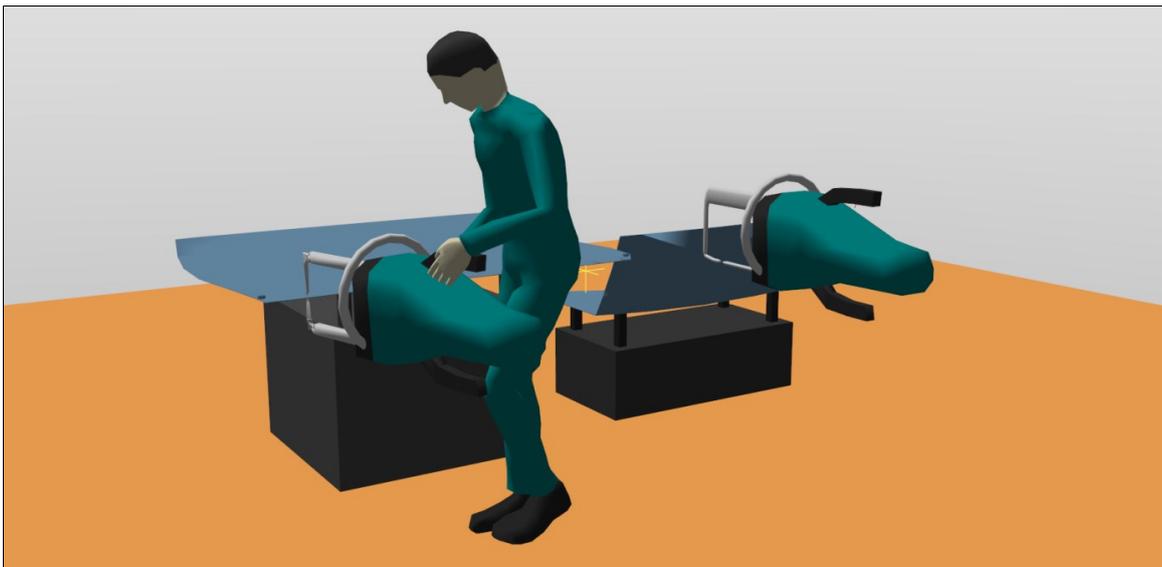


Abbildung 45: Testszenario für »Werkzeug auf Pfad bewegen«

Ohnehin musste das System aufgrund seines Anspruchs und seiner Komplexität einen ausgedehnten Reifeprozess durchlaufen, bis es schließlich erstmals auch in Form von Testinstallationen direkt beim Kunden eingesetzt wurde. Dieser Prozess bestand jedoch nicht nur in der Behebung von Fehlern oder der Integration von Komfortfunktionen, sondern auch in der Anpassung und Erweiterung der Verrichtungsbibliothek. Auch wenn die Vorbetrachtungen beim Aufbau der Bibliothek eine sehr gute Ausgangsbasis geschaffen hatten (siehe Kapitel 3.5.3), konnten bestimmte Details und die Anforderungen an Spezial-Verrichtungen

erst anhand praktischer Beispiele ermittelt werden. Ein konkretes Beispiel hierfür stellt dabei die Verrichtung Werkzeug auf Pfad bewegen (Abbildung 45) dar. Diese war zwar bereits bei der Konzeption vorgesehen und auch implementiert worden, musste jedoch im Zusammenhang mit einem konkreten Projekteinsatz noch einmal grundlegend überarbeitet werden, um unter anderem ein Mitlaufen des Menschmodells beim Werkzeugeinsatz abbilden zu können. Das ebengenannte Beispiel steht dabei stellvertretend für viele kleinere und größere Anpassungen, die auch nach der offiziellen Fertigstellung der ersten Produktiv-Version noch vorgenommen wurden.

Bei den Einsätzen in diversen Kundenprojekten (Abbildung 46 und Abbildung 47), der Durchführung von Anwenderworkshops und dem daraus resultierenden Verkauf von Softwarelizenzen wurde im Übrigen immer darauf geachtet, die Anwender in die Weiterentwicklung des Simulationssystems einzubinden. Die Firma imk automotive GmbH hat dazu eine Kundenzufriedenheitsbefragung und im Jahr 2012 erstmals ein sogenanntes „Anwenderforum“ zum Erfahrungsaustausch durchgeführt. Die Vielzahl und die durchweg positive Resonanz der Teilnehmer bestärkten die imk automotive GmbH den eingeschlagenen Weg und somit die Weiterentwicklung des in dieser Arbeit vorgestellten Simulationssystems auch in Zukunft fortzuführen.



Abbildung 46: Simulation einer Schweiß-Station



Abbildung 47: Simulation einer Station zur Nockenwellenmontage

5 Weiterführende Untersuchungen

Auch wenn das im Rahmen dieser Arbeit erschaffene System bereits eine sehr große Bandbreite von Einsatzszenarien abdecken kann, gibt es noch zahlreiche Aspekte mit Optimierungspotential und weiterem Forschungsbedarf. Der implementierte Bewegungsgenerator erlaubt beispielsweise die Abbildung grundsätzlicher Bewegungsmuster durch die Ansteuerung einzelner Körperteilgruppen. Abhängigkeiten zwischen den Körperteilgruppen und direkte aus physikalischen Gesetzmäßigkeiten resultierende Wechselwirkungen (Abstützung, Ausbalancieren, ...) können derzeit also grundsätzlich abgebildet, jedoch noch nicht automatisch berechnet werden. Hier besteht noch genügend Potential, welches in Form eines „Bewegungsgenerators 2.0“ umgesetzt werden sollte.

Die Ansteuerung der einzelnen Fingersegmente erfolgt derzeit noch über vordefinierte Griffhaltungen. Insoweit wurden bereits Untersuchungen hinsichtlich der Verwendung der einzelnen Finger bei der Ausführung von Filigranarbeit vorgenommen [74]. Dabei wurde sowohl der Umgang mit filigranen Werkzeugen als auch die Handhabung entsprechend kleiner Bauteile analysiert. Die daraus gewonnenen Erkenntnisse werden ebenfalls in den neuen Bewegungsgenerator einfließen.

Auch beim Thema des Greifens und der Objekthandhabung sieht der Autor noch deutliche Weiterentwicklungspotentiale, insbesondere im Bereich der gestaltbasierten Objekt- und Merkmalerkennung. Nach derzeitigem Stand des Simulationssystems wird beim Handhaben eines Objektes immer nur dessen Hüllquader berücksichtigt. In Verbindung mit benutzerdefinierten Zusatzinformationen (Greif- und Wirkpunkte) können zumindest schon grobe Erkenntnisse über die Handhabung gewonnen werden. Die Integration einer Gestalt- und Merkmalerkennung für 3D-Objekte könnte die derzeit noch vom Benutzer anzugebenden Zusatzinformationen überflüssig machen. Dies wäre ein weiterer Schritt zur Vervollständigung der Wissensbasis des Simulationssystems. Das digitale Menschmodell würde somit in die Lage versetzt werden, die Greifpunkte an Objekten anhand bestimmter Merkmale (z.B. Griffe, Henkel) selbstständig zu ermitteln und darüber hinaus von der Gestalt des Objekts auf dessen Art der Verwendung zu schließen (z.B. Werkzeugtyp). Wenn eine Gestalterkennung für 3D-Objekte implementiert werden würde, könnte diese darüber hinaus für eine weitaus genauere Kollisionsprüfung und -vermeidung herangezogen

werden. Dies betreffe neben der Objekthandhabung auch alle anderen Bewegungsabläufe des digitalen Menschmodells in seiner virtuellen Umgebung.

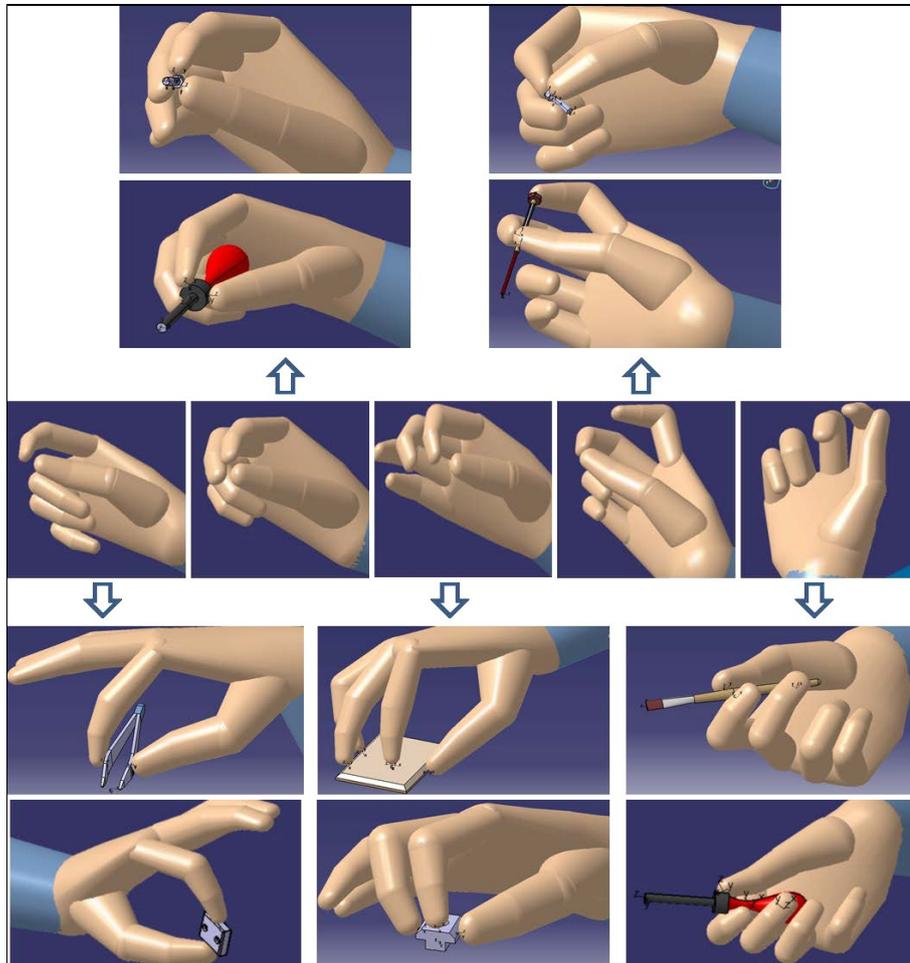


Abbildung 48: Greifarten an verschiedenen filigranen Objekten [73, p. 82]

Aufbauend auf die im Rahmen dieser Arbeit konzipierte und bereits umgesetzte Verrichtungsbibliothek sollen weitere komplexere Verrichtungen entwickelt werden. Primärziele sind dabei die Vereinfachung der Bewegungsmodellierung durch Gruppierung und Vorparametrisierung vorhandener Verrichtungen sowie die Ergänzung noch fehlender Basisverrichtungen. Erste „Kandidaten“ für eine Umsetzung sind beispielweise Verrichtungen, die ein „Mitlaufen“ bei gleichzeitiger Verrichtungsausführung erfordern. Aber auch die Ergänzung von kooperativen Tätigkeiten, also die Ausführung von Verrichtungen von mehreren digitalen Menschmodellen gleichzeitig (gemeinsames Tragen, etc.), stellt eine

potentielle Richtung für die zukünftige Weiterentwicklung dar. (Abbildung 49) Auch hierzu wurden bereits entsprechende Voruntersuchungen durchgeführt, welche in naher Zukunft umgesetzt werden.

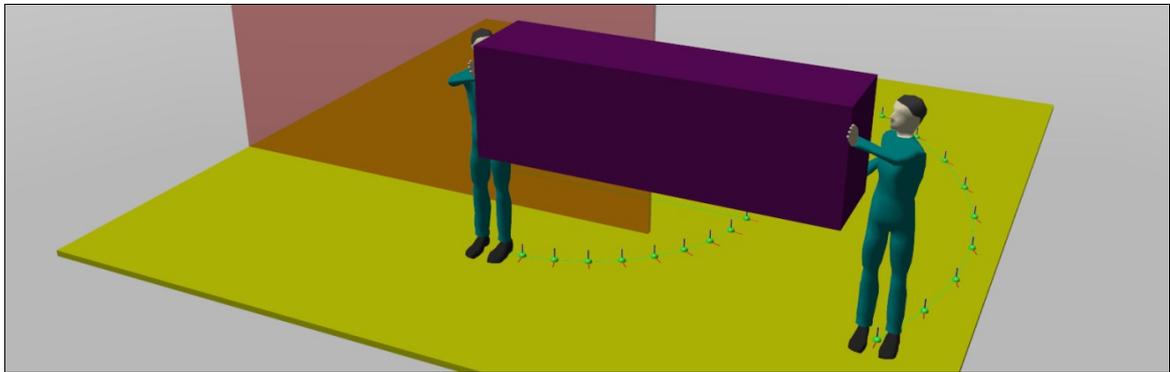


Abbildung 49: Kooperative Tätigkeiten

Da der Fokus der vorliegenden Arbeit auf Tätigkeiten der industriellen Produktion und dabei insbesondere auf solchen der Automobilfertigung lag, ist auch die Erweiterung des thematischen Bereichs der Verrichtungen eine sinnvolle Weiterentwicklungsrichtung. Zunächst sollten dabei verwandte Industriezweige betrachtet werden, für die sich der notwendige Aufwand der Anpassungen in Grenzen halten sollte. Im Lauf der Entwicklung gab es darüber hinaus Anfragen hinsichtlich der Nutzung des Systems für militärische Zwecke, wobei hier weniger die Simulation von Kampfeinsätzen als vielmehr die Nutzung des Systems zu Schulungs- und Ausbildungszwecken im Vordergrund steht, z.B. im Rahmen von Simulationen zur Maschinenbedienung. (Abbildung 50) Ziel ist es, die Abläufe und Abfolgen der einzelnen Tätigkeitsschritte in Einzel- und Mehrbenutzerumgebungen zu schulen. Insbesondere durch Mehrbenutzersimulationen, innerhalb derer die Schulungsteilnehmer gegenseitig ihre virtuellen Repräsentanten sehen, könnte das entwickelte System zur Ansteuerung der digitalen Menschmodelle den Realitätsgrad der Ausbildung deutlich erhöhen.

Jede Erweiterung des Anwendungsgebietes bringt freilich wieder neue Herausforderungen mit sich. Im Gebiet der Flugzeugmontage, das der Automobilfertigung gar nicht so unähnlich ist, wird deutlich, dass schon die aufgrund der Größenverhältnisse primär anzutreffende Arbeit „im Produkt“ eine solche Herausforderung darstellen kann. Bei Arbeiten, die außerhalb des Produkts ausgeführt werden, ist die Betrachtung von Kollisionen mit dem Produkt weitestgehend unkritisch – hierfür können entsprechend vereinfachte geometrische

5 Weiterführende Untersuchungen

Hilfsobjekte verwendet werden; darüber hinaus beschränkt das Produkt die körperliche Bewegungsfreiheit nur wenig. Arbeiten im Innenraum hingegen können die Bewegungsfreiheit des Körpers aufgrund ungünstiger Platzverhältnisse jedoch so massiv einschränken, dass zur Simulation derartiger Tätigkeiten vollkommen neue Bewegungsmuster und eine viel detailliertere Kollisionsvermeidung für das digitale Menschmodell implementiert werden müssen. Derartige Bewegungen werden in ihrer Komplexität nur noch durch die ebenfalls bereits angefragte Abbildung von Arbeit in Schwerelosigkeit übertroffen. Während die Arbeit unter dem Einfluss von Schwerkraft zumindest die Randbedingungen der möglichen Bewegungsformen und der Kontaktpunkte mit dem Boden vorgibt, bietet die Arbeit im Weltall um ein Vielfaches mehr Freiheitsgrade, die Berücksichtigung finden müssten.



Abbildung 50: Trainingssimulation für transportable Pontonbrücke [75]

Die Abbildung menschlicher Arbeit in einer 3D-Simulation liefert eine vollkommen neue Quantität und Qualität von Daten, die nach Ansicht des Autors durch Integration weiterer Methoden zur Bewertung und Beurteilung menschlicher Arbeit genutzt werden sollten. Dies können sowohl etablierte Methoden

aus den Bereichen Ergonomie (RULA, OWAS) und Zeitermittlung (MTM-Verfahren) sein. Ebenso können das aber auch neuartige, gerade in Entwicklung befindliche Ansätze wie das „Numerische Verhaltensmodell des Menschen“ [76] sein. Letzteres würde neben der Simulation in Normleistung erbrachter Tätigkeiten auch die Einflüsse des jeweiligen Werker-Typus, die auf ihn einwirkenden Umweltfaktoren und seine daraus resultierende Beanspruchung abbilden. Das Ergebnis wäre die simulative Abbildung von Arbeit, die von der Norm abweicht, weil der Arbeiter durch Überbeanspruchung geschwächt ist.

Einen weiteren Vorteil der Integration vielschichtiger Bewertungsverfahren stellt die daraus resultierende Möglichkeit dar, diese Ansätze zur Implementierung von Optimierungsalgorithmen zu nutzen. Übergeordnete Hilfsfunktionen könnten den Anwender in Zukunft unterstützen, um erstellte Simulationen ganz gezielt hinsichtlich bestimmter Aspekte (optimale Ergonomie, kürzeste Laufwege, höchste Wertschöpfung) zu verbessern. Diese Optimierung könnte anhand der Anpassung des Arbeitsplatzlayouts und / oder der Art und Abfolge der verwendeten Einrichtungen erfolgen. Dabei sollten diese Hilfswerkzeuge nicht das Ziel haben, den Planungsingenieur ersetzen zu wollen. Vielmehr sollten die Hilfsfunktionen alternative Gestaltungsvorschläge liefern, welche sich auf belastbares Fach- und Erfahrungswissen zurückführen lassen.

Zur Simulation menschlicher Arbeit gehört mitunter auch die Interaktion mit sich bewegenden Objekten. Beispiele hierfür sind die Arbeit an Werkstücken, welche über Fördersysteme bereitgestellt werden oder auch die Arbeit an und in bewegten Fahrzeugkarosserien. Die Berücksichtigung dynamischer Objekte und der sich daraus ergebenden Abhängigkeiten für Ort und Dauer der betreffenden Tätigkeiten stellt eine weitere Herausforderung für die Weiterentwicklung des vorliegenden Simulationssystems dar. Anders als Tätigkeiten in statischen Umgebungen lassen sich die dynamischen Zusammenhänge nur noch sehr schwer vom menschlichen Vorstellungsvermögen ablaufichtig erfassen. Umso wichtiger ist die Visualisierung und simulationsgestützte Auswertung eben jener komplexen Szenarien. Die Hauptthemen, welche im Zuge dessen bearbeitet werden müssen, betreffen die dynamische Laufwegs-Bestimmung und Kollisionsvermeidung ebenso wie erweiterte Möglichkeiten zur zeitlichen Synchronisation von menschlichen Tätigkeiten und natürlich auch die grundsätzliche Vorgehensweise zur Definition der Objektbewegungen. (Abbildung 51)

5 Weiterführende Untersuchungen



Abbildung 51: Tests zur Arbeit an der bewegten Karosserie

6 Zusammenfassung

Die vorliegende Arbeit beschäftigte sich mit der Konzeption und Umsetzung eines neuartigen Simulationssystems zur Planung, Visualisierung und Bewertung menschlicher Tätigkeiten. Während bereits für zahlreiche Aspekte der industriellen Fertigung adäquate Software-Werkzeuge im Umfeld der Digitalen Fabrik existierten, bestand bei der Simulation von manueller Arbeit noch erheblicher Forschungs- und Entwicklungsbedarf. Digitale Mensch-Modelle gibt es zwar bereits in großer Anzahl. Allerdings bedienen die vorhandenen Modelle lediglich bestimmte Nischen des digitalen Planungsprozess. Die Simulation menschlicher Bewegungen bei industriellen Produktionsprozessen mit dem Ziel, bereits frühzeitig Erkenntnisse über Plausibilität, Zeitbedarf und ergonomische Kriterien der geplanten Arbeitsplätze zu erhalten, war dabei bislang nur sehr eingeschränkt und auch nur mit großem Zeitaufwand möglich. Vor diesem Hintergrund hat es sich der Autor zum Ziel gesetzt, ein Simulationssystem zu entwerfen und umzusetzen, das einerseits auf möglichst intuitive und effiziente Art und Weise zu bedienen ist und andererseits belastbare Planungsergebnisse berechnen kann.

Ausgangspunkt der Untersuchungen war die Analyse existierender Software-Werkzeuge der Digitalen Fabrik sowie der Tätigkeiten und Ergebnisse während des Planungsprozess. Dabei konnte der Autor die Expertise eines mittelständischen Ingenieurdienstleisters und Einblicke in dessen laufende Planungsprojekte nutzen, um sich zunächst mit der Thematik, dem Umfeld und den gegebenen Vorbedingungen vertraut zu machen. Die daraus resultierenden Kontakte zu potentiellen Anwendern des Simulationssystems nutzte der Autor zur Durchführung einer entsprechenden Anforderungsanalyse. Basierend auf den ermittelten Anforderungen konnte anschließend ein erster Grob-Systementwurf erstellt werden. Dabei konnten sowohl die Schnittstellen zum Umfeld des Systems identifiziert und detailliert werden, als auch erste Erkenntnisse über eine potentielle interne Struktur gewonnen werden.

Eine Besonderheit stellt dabei der Ansatz einer zweistufigen Verarbeitung der Eingangsinformationen (Umgebungsinformationen und Tätigkeitsbeschreibung) zunächst in eine sogenannte abstrakte Bewegungsbeschreibung und erst anschließend in die konkrete Bewegungsbeschreibung dar. Während die abstrakte Bewegungsbeschreibung lediglich geometrische Bewegungszielvorgaben bein-

hältet, werden diese Informationen nachfolgend mit Hilfe eines speziellen Bewegungsgenerators und durch Anwendung auf ein konkretes digitales Menschmodell in eine konkrete Bewegungsbeschreibung (Gelenkwinkel und Positionsangaben) überführt. Diese Zweiteilung hat den Vorteil, dass der eigentliche Kern dieser Arbeit, nämlich die Komponente zur Überführung der realsprachlichen Arbeitsanweisungen in Bewegungsinformationen, vollkommen unabhängig von der Art und der Struktur des angesteuerten Menschmodells betrachtet werden konnte.

Für die Überführung der Eingangsinformationen in die abstrakte Bewegungsbeschreibung entwarf der Autor ein System, welches die Kenntnisse des Facharbeiters repräsentieren und für die Bewegungsgenerierung nutzen sollte. Kernelemente dieses Ansatzes sind die sogenannten »Verrichtungen«. Hierbei handelt es sich um parametrische Bewegungs- bzw. Tätigkeitsbausteine zur Ansteuerung der digitalen Menschmodelle. Darüber hinaus wurden die »Verrichtungen« in Form eines Bausteinsystems konzipiert, dessen Elemente sich beliebig aneinanderreihen und untereinander kombinieren lassen. Während die entstandene Verrichtungsbibliothek einen Teil der benötigten Funktionalität beisteuert, wurde ein anderer Teil in eine Bibliothek von übergeordneten Funktionen des Simulationsmodells ausgelagert. Diese beinhalten neben Funktionalitäten zur Objektidentifikation und zur Bestimmung des Objekthandlings auch Methoden zur Lauf- und Einbaupfadbestimmung. Funktionen, mit Hilfe derer die Art der Handhabung von Werkzeugen und anderer Hilfsmittel unter den gegebenen Rahmenbedingungen bestimmt werden können, wurden ebenfalls implementiert.

Um die erarbeiteten Konzepte zu überprüfen, mündeten diese in eine softwaretechnische Umsetzung. Hierbei verlief der Weg von einfachen Demonstratoren und Prototypen über erste Integrationsbeispiele bis hin zu einem vollwertigen Plug-In für das PLM-System DELMIA V5 und schließlich sogar zu einer „Standalone-Lösung“ des Simulationssystems. Getreu dem Ansatz, dass die (Software-)Technologie letztendlich nur ein Mittel zur Umsetzung der eigenen Ideen darstellt, entwickelte der Autor die Lösungen mit einer großen Bandbreite an Werkzeugen. Neben vollwertigen 3D-Autorensystemen und Spiele-Engines kamen einfachere Skript-Sprachen ebenso zum Einsatz wie das .NET-Framework und dessen objektorientierter Leistungsumfang.

Der Autor hatte die Möglichkeit, alle erarbeiteten Konzepte und Erkenntnisse durch seine Mitarbeit bei der Entwicklung der Simulationssoftware »ema⁵« (Editor menschlicher Arbeit) zu verifizieren. Dadurch wurde die praxisorientierte Lösungsfindung für die im Rahmen dieser Arbeit auftretenden Frage- und Problemstellungen erheblich vereinfacht. Schon sehr früh konnten Software-Stände in internen Projekteinsätzen getestet werden. Die sich hieraus ergebenden Iterationsschleifen des Testens und Verbesserns hatten großen Anteil an der Optimierung der entwickelten Algorithmen und der Strukturierung der Verrichtungsbibliothek.

Im Rahmen dieser Arbeit ist ein bislang beispielloses Simulationssystem zur Ansteuerung digitaler Menschmodelle entstanden. Erstmals ist es nun möglich, das digitale Abbild des Menschen auf einfache und intuitive Art in der Digitalen Fabrik zu simulieren und dabei Erkenntnisse über Zeitbedarf, Plausibilität und Ergonomie abzuleiten. Durch eine dem Planungsingenieur vertraute Prozesssprache können dem Menschmodell nahezu realsprachliche Anweisungen gegeben werden, welche direkt in entsprechende Körperbewegungen übersetzt werden. Diese Art der Ansteuerung reduziert den bisherigen Aufwand für die Erstellung von Simulationen mit Hilfe digitaler Menschmodelle von bislang 250 Minuten Erstellungsaufwand für eine 1 Minute Simulationszeit (siehe Abschnitt 4.7) auf weniger als die Hälfte. Noch größer wird die Zeitersparnis, wenn bereits erstellte Simulationen dazu genutzt werden können, um daraus Alternativen bzw. Varianten abzuleiten.

Das Simulationssystem »ema⁵« ist derzeit (Stand 2014) bei allen deutschen Automobilherstellern in den entsprechenden Planungsabteilungen im Einsatz. Im Volkswagen-Konzern wurde die Software sogar offiziell zertifiziert. Darüber hinaus verfügen mittlerweile auch zahlreiche deutsche Hochschulen über entsprechende Lizenzpakete, was die Integration in die Lehrpläne fachverwandter Studiengänge ermöglicht. Während vor einigen Jahren die simulationsgestützte Planung menschlicher Arbeit noch als „schwarzes Loch“ [77] der Digitalen Fabrik galt, wurde dieses durch zahlreiche Forschungs- und Entwicklungsprojekte mittlerweile geschlossen.

Literaturverzeichnis

- [1] Statistische Ämter des Bundes und der Länder, „Bevölkerungs- und Haushaltsentwicklung im Bund und in den Ländern (Heft 1),“ Statistisches Bundesamt, Wiesbaden, 2011.
- [2] Statistisches Bundesamt, „Bevölkerung Deutschlands bis 2060 - 12. koordinierte Bevölkerungsvorausberechnung,“ Statistisches Bundesamt, Gruppe ID, Pressestelle, Gruppe VIA "Demografische Modellrechnungen", Wiesbaden, 2009.
- [3] Daimler AG, „Daimler Nachhaltigkeitsbericht 2011,“ Daimler AG, Stuttgart, 2011.
- [4] Volkswagen Aktiengesellschaft, „Nachhaltigkeit - Bericht 2011,“ Volkswagen Aktiengesellschaft, Wolfsburg, 2011.
- [5] J. Witt und T. Witt, Werkzeuge des Qualitätsmanagements in der KVP-Praxis, Düsseldorf: Symposion Publishing GmbH, 2007.
- [6] VDI-Gesellschaft Fördertechnik Materialfluss Logistik, Fachbereich Modellierung und Simulation, Fachausschuss Digitale Fabrik, „Digitale Fabrik Ergonomische Abbildung des Menschen VDI-Richtlinie 4499, Blatt 4,“ in *VDI-Handbuch Materialfluss und Fördertechnik, Band 8: Materialfluss II (Organisation/Steuerung)*, Düsseldorf, Verein Deutscher Ingenieure e.V., 2012.
- [7] J. Mühlstedt, Entwicklung eines Modells dynamisch-muskulärer Arbeitsbeanspruchungen auf Basis digitaler Menschmodelle, Chemnitz: Technische Universität Chemnitz, Fakultät Maschinenbau, 2011.
- [8] L. Fritzsche, Work Group Diversity and Digital Ergonomic Assessment as New Approaches for Compensating the Aging Workforce in Automotive Production (engl. Übersetzung durch Autor), Dresden: Dissertation, Institut für Arbeits-, Organisations- und Sozialpsychologie, Technische Universität Dresden, 2010.
- [9] S. Bauer, Virtools Dev / Visualisierung manueller Tätigkeiten für Planungszwecke, Chemnitz: Studienarbeit, Technische Universität Chemnitz, 2006.

Literaturverzeichnis

- [10] S. Bauer, Entwicklung einer prototypischen Software zur Planung und Visualisierung manueller Tätigkeiten, Chemnitz: Diplomarbeit, Technische Universität Chemnitz, 2006.
- [11] J. R. Tiedtke, Allgemeine BWL - Betriebswirtschaftliches Wissen für kaufmännische Berufe - Schritt für Schritt, Wiesbaden: Betriebswirtschaftl. Verlag Dr. Th. Gabler | GWV Fachverlage GmbH, 2007.
- [12] E. Gutenberg, Grundlagen der Betriebswirtschaftslehre. Band 1: Die Produktion, Berlin, Heidelberg, New York: Springer, 1983.
- [13] A. Kobylka, Skript. Methoden der Fabrikplanung MBK510(I), Zwickau: Westsächsische Hochschule, 2009.
- [14] F. W. Taylor, Scientific Management, New York, London: Harper & Brothers, 1947.
- [15] T. Weingardt, Amerikanisierung der deutschen Wirtschaft vor und nach dem zweiten Weltkrieg, Leipzig: GRIN-Verlag, 2008.
- [16] H. Kettner, J. Schmidt und H.-R. Greim, Leitfaden der systematischen Fabrikplanung, Hanser Fachbuchverlag, 1984.
- [17] U. Bracht, D. Geckler und S. Wenzel, Digitale Fabrik - Methoden und Praxisbeispiele, Springer-Verlag, 2011.
- [18] REFA Bundesverband e.V., REFA Ausgewählte Methoden zur prozessorientierten Arbeitsorganisation, Hemsbach: Druckpartner Rübelmann GmbH, 2002.
- [19] M. Kropik, Produktionsleitsysteme in der Automobilfertigung, Berlin, Heidelberg: Springer, 2009.
- [20] C.-G. Grundig, Fabrikplanung: Planungssystematik - Methoden - Anwendungen, München: Carl Hanser Verlag GmbH CoKG, 2013.
- [21] Dudenverlag, „www.duden.de,“ Bibliographisches Institut GmbH, 2013. [Online]. Available: <http://www.duden.de/rechtschreibung/Ergonomie>.
- [22] D. Kern, „Ergonomische Bewertung von Montagetätigkeiten,“ BGHM Berufsgenossenschaft Holz und Metall, Villingen-Schwenningen, 2012.
- [23] Deutsche Gesetzliche Unfallversicherung e.V., BGI/GUV-I 7011 Belastungen für Rücken und Gelenke - was geht mich das an?, Berlin, 2013.

Literaturverzeichnis

- [24] U. Steinberg, A. Klußmann und F. Liebers, Leitmerkmalmethode Manuelle Arbeitsprozesse: Bericht über die Erprobung, Validierung und Revision, Dortmund: Bundesanstalt für Arbeitsschutz und Arbeitsmedizin, 2012.
- [25] M. Kugler, M. Bierwirth, K. Schaub, A. Sinn-Behrendt, A. Feith, K. Ghezel-Ahmadi und R. Bruder, KoBRA – Kooperationsprogramm zu normativem Management von Belastungen und Risiken bei körperlicher Arbeit, Ergonomie in der Industrie - aber wie?, München: Meindl Druck, 2010.
- [26] National Institute of Occupational Safety and Health NIOSH, Musculoskeletal disorders and workplace factors: A critical review of epidemiologic evidence for work-related musculoskeletal disorders of the neck, upper extremity, and low back, Cincinnati: NIOSH, 1997.
- [27] O. Karhu, P. Kansi und I. Kuorinka, „Correcting working postures in industry: A practical method for analysis,“ *Applied Ergonomics*, 8, pp. 199-201, 1977.
- [28] L. McAtamney und E. N. Corlett, „RULA: A survey method for the investigation of workrelated upper limb disorders,“ *Applied Ergonomics* 24, pp. 91-99, 1993.
- [29] K. Schaub, „Das "Automotive Assembly Worksheet" (AAWS),“ in *Montageprozesse gestalten - Fallbeispiele aus Ergonomie und Organisation*, Stuttgart, Ergonomia-Verlag, 2004, pp. 91-111.
- [30] K. Schaub und K. Ghezel-Ahmadi, „Vom AAWS zum EAWS – ein erweitertes Screening-Verfahren für körperliche Belastungen,“ in *Bericht zum 53. Arbeitswissenschaftlichen Kongress vom 28.2.-2.3.2007 an der Otto-von-Guericke-Universität Magdeburg*, Dortmund, GfA-Press, 2007, pp. 601-604.
- [31] C. Bischoff, R. Dengler und H. C. Hopf, EMG, NLG: Elektromyographie, Nervenleitungsuntersuchungen, Stuttgart: Georg Thieme Verlag KG, 2008.
- [32] R. Bokranz und K. Landau, Produktivitätsmanagement von Arbeitssystemen, Stuttgart: Schäffer-Poeschel Verlag, 2006.
- [33] DIN Deutsches Institut für Normung e. V., DIN EN ISO 6385 Grundsätze der Ergonomie für die Gestaltung von Arbeitssystemen, Beuth Verlag, 2004.

- [34] R. Bokranz und K. Landau, Einführung in die Arbeitswissenschaft. Analyse und Gestaltung von Arbeitssystemen, Stuttgart: Ulmer, 1991.
- [35] M. Gaitanides, R. Scholz und A. Vrohling, Prozessmanagement, München-Wien: Hanser-Verlag, 1994.
- [36] B. Britzke, MTM in einer globalisierten Wirtschaft. Arbeitsprozesse systematisch gestalten und optimieren, München: FinanzBuch Verlag, 2010.
- [37] J. Becker, M. Rosemann und R. Schütte, „Grundsätze ordnungsgemäßer Modellierung,“ in *Wirtschaftsinformatik* 37 Nr. 5, 1995, pp. 435-445.
- [38] J. Ishiwata, Die flexible Fabrik. Layout-Planung mit Prozess-Analyse, Landsberg: Mi-Wirtschaftsbuch, 2001.
- [39] VDI-Gesellschaft Fördertechnik Materialfluss Logistik, Fachbereich Modellierung und Simulation, Fachausschuss Digitale Fabrik, „Digitale Fabrik Grundlagen VDI-Richtlinie 4499, Blatt 1,“ in *VDI-Handbuch Materialfluss und Fördertechnik, Band 8: Materialfluss II (Organisation/Steuerung)*, Düsseldorf, Verein Deutscher Ingenieure e.V., 2008.
- [40] W. Leidholdt, Vorlesung "Digifab", Chemnitz: imk automotive GmbH, 2012.
- [41] WZL RWTH Aachen, „PLM - Product Lifecycle Management,“ 2013. [Online]. Available: <http://www.plm-info.de/de/default.html>.
- [42] K. Kurbel, Produktionsplanung und -steuerung im Enterprise Resource Planning und Supply Chain Management (6. Auflage), Oldenbourg Verlag, 2005.
- [43] M. Schenk und S. Wirth, Fabrikplanung und Fabrikbetrieb: Methoden Für die wandlungsfähige und vernetzte Fabrik, Springer-Verlag, 2006.
- [44] M. Hesseler und M. Görtz, Basiswissen ERP-Systeme: Auswahl, Einführung & Einsatz, W3L GmbH, 2007.
- [45] V. Duffy, Handbook of Digital Human Modeling, Boca Raton: CRC Press, 2008.
- [46] M. Schühle und N. Koller, Ergonomiesoftware, GRIN Verlag, 2004.
- [47] R. Jendrusch, W. Leidholdt, S. Bauer, H. Hermsdorf, N. Hofmann, A. Keil und B. Spanner-Ulmer, „Algorithmische Bewegungsgenerierung und

- semiautomatische Ergonomiebewertung für digitale Menschmodelle,“ *Zeitschrift für Arbeitswissenschaft* 65, pp. 259-271, 2011.
- [48] K. G. Schaub, G. Caragnano, B. Britzke und R. Bruder, „The European Assembly Worksheet,“ *Theoretical Issues in Ergonomics Science, Volume 14, Issue 6*, pp. 616-639, 2013.
- [49] T. B. Seiler, *Evolution des Wissens 1: Evolution der Erkenntnisstrukturen*, Berlin: Lit Verlag Dr. W. Hopf, 2012.
- [50] M. Polanyi, *The tacit dimension*, Garden City, New York: Doubleday, 1966.
- [51] I. Nonaka und H. Takeuchi, *The knowlegde creating company*, New York: Oxford University Press, 1995.
- [52] M. E. Todd, *Der Körper denkt mit: Anatomie als Ausdruck dynamischer Kräfte (2. überarbeitete Auflage, orig. 1937)*, Bern: H. Huber, 2003.
- [53] REFA Verband für Arbeitsstudien und Betriebsorganisation e.V., *Methodenlehre der Betriebsorganisation: Lexikon der Betriebsorganisation*, München: Carl-Hanser, 1993.
- [54] T. Speiser, *Aufbereitung von Verrichtungen der menschlichen Arbeit für die Planung auf visueller Basis in DELMIA*, Chemnitz: Projektarbeit, Technische Universität Chemnitz, 2008.
- [55] J. Uhlig, *Analyse, Systematisierung und Vorbereitung der Implementierung parametrischer Verrichtungen zur Anwendung im Planungswerkzeug "Editor menschlicher Arbeit"*, Chemnitz: Diplomarbeit, Technische Universität Chemnitz, 2009.
- [56] Deutsche MTM - Vereinigung e.V., *MTM-UAS Lehrgangsunterlage - A/JF -*, Hamburg: Deutsche MTM-Vereinigung e.V, 2007.
- [57] imk automotive GmbH, „eMAN - Verbundprojekt, Abschlußbericht (gefördert durch die Sächsische Aufbaubank SAB),“ imk automotive GmbH, Chemnitz, 2010.
- [58] imk automotive GmbH, „Konzept "Vertikales Bewegungssystem“,“ Chemnitz, 2013.
- [59] imk automotive GmbH, „Systemdokumentation emaV5 (Version 1.0),“ Chemnitz, 2011.

- [60] W. Park, „Data-Based Human Motion Simulation Methods,“ in *Handbook of Digital Human Modeling*, Boca Raton, CRC Press, 2008, pp. 9-1 bis 9-15.
- [61] W. Graumann und D. Sasse, *CompactLehrbuch Anatomie 2 - Bewegungsapparat*, Stuttgart: Schattauer, 2004.
- [62] Institut für Mechatronik e.V., „Biomechanische Menschmodelle Probleme der Kompatibilität,“ Chemnitz, 2011.
- [63] Siemens PLM Software, „Tecnomatix - Jack and Process Simulate Human,“ 2012. [Online]. Available: http://www.plm.automation.siemens.com/en_us/products/tecnomatix/assembly_planning/jack/. [Zugriff am 26 05 2012].
- [64] A. Olbrich, *Netze, Protokolle, Spezifikation*, Braunschweig / Wiesbaden: Vieweg-Verlag (Springer-Verlag), 2003.
- [65] M. Herrmann, *Service Orientierte Architektur (SOA) - Identifizierung äquivalenter Services in Form semantischer semiautomatischer Unterstützung des EMEO-Layers*, München: GRIN-Verlag, 2010.
- [66] P. E. Hart, N. J. Nilsson und B. Raphael, „A Formal Basis for the Heuristic Determination of Minimum Cost Paths,“ in *IEEE Transactions on Systems Science and Cybernetics SSC4 (2)*, 1968, pp. 100-107.
- [67] K. Knorr, *Auswahl, Konzeption und Implementierung eines geeigneten Verfahrens zur Pfadberechnung und Kollisionsvermeidung*, Mittweida: Bachelorarbeit, Hochschule Mittweida, 2011.
- [68] P. Geiler, *Analyse und Strukturierung ausgewählter Verrichtungen menschlicher Arbeit zur algorithmischen Abbildung*, Jena: Bachelorarbeit, Fachhochschule Jena, 2010.
- [69] Deutsche MTM-Vereinigung e.V., *MTM-UAS Lehrgangunterlagen*, Hamburg: Deutsche MTM-Vereinigung e.V., 2004.
- [70] J. Tilch, *Ausarbeitung eines Algorithmus zur automatisierten Werkstückhandhabung für ein Digitales Menschmodell*, Chemnitz: Bachelorarbeit, Technische Universität Chemnitz, 2010.
- [71] M. Manns und N. A. Arteaga Martín, „Automated DHM Modeling for Integrated Alpha-Numeric and Geometric Assembly Planning,“ in *Smart Product Engineering*, Berlin Heidelberg, Springer-Verlag, 2013, pp. 325-334.

- [72] L. Fritzsche und R. Jendrusch, „Prospektive Arbeitsgestaltung mit digitalen Menschmodellen - Validität der Ergonomiebewertung von simulierten Arbeitsplätzen in der Automobilmontage,“ in *Arbeit, Beschäftigungsfähigkeit und Produktivität im 21. Jahrhundert: Bericht zum 55. Kongress der Gesellschaft für Arbeitswissenschaft*, Dortmund, GfA Press, 2009, pp. 583-587.
- [73] M. Leckel, Formulierung und Klassifizierung von Verrichtungen filigraner Arbeit für den „Editor menschlicher Arbeit“, Jena: Fachhochschule Jena, Fachbereich Wirtschaftsingenieurwesen, 2011.
- [74] D. Lietze, Parametrisierung und Modellierung von Verrichtungen filigraner Arbeit für den „Editor menschlicher Arbeit“, Chemnitz: Masterarbeit, Technische Universität Chemnitz, 2012.
- [75] Szenaris GmbH, „Homepage der Firma Szenaris,“ 2012. [Online]. Available: <http://www.szenaris.com/>. [Zugriff am 27 12 2012].
- [76] W. Leidholdt und C. Wohlschläger, „Das »Numerische Verhaltensmodell des Menschen« - Simulation von Werkern unter erhöhter Beanspruchung,“ in *Tagungsband 15. IFF Wissenschaftstage (Magdeburg)*, Stuttgart, Fraunhofer Verlag, 2012.
- [77] W. Leidholdt und S. Bauer, „Der virtuelle Facharbeiter (Editor menschlicher Arbeit),“ in *Tagungsband 12. IFF Wissenschaftstage (Magdeburg)*, Stuttgart, Fraunhofer Verlag, 2009.
- [78] D. Reif, Methode zur Visualisierung manueller Tätigkeiten, Mittweida: Diplomarbeit, Hochschule Mittweida (FH), 2006.
- [79] imk automotive GmbH, „Gesamt-Projektvorstellung eMAN,“ Chemnitz, 2009.
- [80] J. Mühlstedt, H. Kaußler und B. Spanner-Ulmer, „Programme in Menschengestalt: digitale Menschmodelle für CAx- und PLM-Systeme,“ *Zeitschrift für Arbeitswissenschaft*, 02 2008.
- [81] Human Solutions Gruppe, „Produkte Mobility - RAMSIS,“ 2012. [Online]. Available: http://www.human-solutions.com/mobility/front_content.php?idcat=250. [Zugriff am 26 05 2012].
- [82] Dassault Systèmes, „Digitale Menschmodelle,“ 2012. [Online]. Available:

<http://www.3ds.com/de/products/delmia/solutions/human-modeling/overview/>. [Zugriff am 26 05 2012].

- [83] M. Haun, Wissensbasierte System - Eine praxisorientierte Einführung, Renningen: expert Verlag, 2000.
- [84] S. G. Tzafestas, Knowledge Based Systems - Advanced Concepts, Techniques & Applications, Singapur: World Scientific Publishing Co.Pte. Ltd., 1997.
- [85] E. Castillo, J. M. Gutiérrez und A. S. Hadi, Expert Systems and Probabilistic Network Models, New York: Springer-Verlag Inc., 1997.
- [86] R. A. Akerbar und P. S. Sajja, Knowledge-Based Systems, Sudbury, MA: Jones and Bertlett Publishers, LLC, 2010.
- [87] C. Helfrich, Praktisches Prozess-Management: vom PPS-System zum Supply chain Management, München: Hanser Verlag, 2002.
- [88] E. Gutenberg, Einführung in die Betriebswirtschaftslehre, Wiesbaden: Betriebswirtschaftlicher Verlag Dr. Th. Gabler, 1958.
- [89] H. Schmigalla, Fabrikplanung: Begriffe und Zusammenhänge, Hanser Verlag, 1995.
- [90] M. Schenk, S. Wirth und E. Müller, Fabrikplanung und Fabrikbetrieb: Methoden für die wandlungsfähige und vernetzte Fabrik, Berlin, Heidelberg: Springer-Verlag, 2004.

Thesen

Die vorliegende Arbeit stellt mehrere Thesen auf, welche sowohl durch Rapid Prototyping als auch durch die Integration der Ergebnisse in ein kommerzielles Softwaresystem und dessen Einsatz in realen Planungsprojekten evaluiert wurden.

These 1:

Eine mit Hilfe einer Prozesssprache dokumentierte Arbeitsanweisung lässt sich in Kombination mit vorhandenen geometrischen Umgebungsinformationen automatisch in eine Bewegungssimulation für digitale Menschmodelle überführen.

These 2:

Durch die Implementierung eines 2-stufigen Verfahrens zur Bewegungsgenerierung kann die Anwendbarkeit der Methodik auf digitale Menschmodelle, die sich in (Körper-)Größe und Struktur unterscheiden, sichergestellt werden.

These 3:

Zur Überführung der Tätigkeitsbeschreibung und Umgebungsinformationen in eine Menschmodell-unabhängige (abstrakte) Zwischenrepräsentation wird ein geeignetes Simulationsmodell und eine prozedurale Wissensbasis benötigt.

These 4:

Die abstrakte Bewegungsbeschreibung (geometrische Zwangsbedingungen bzw. Constraints) wird durch einen Bewegungsgenerator auf ein konkretes digitales Menschmodell angewendet, um eine konkrete Bewegungsbeschreibung abzuleiten (Gelenkwinkel & Positionen).

Thesen

These 5:

Mit weniger als 30 parametrischen Verrichtungen können nahezu alle auftretenden manuellen Tätigkeiten im Automobilbau (Rohbau & Montage) abgebildet werden.

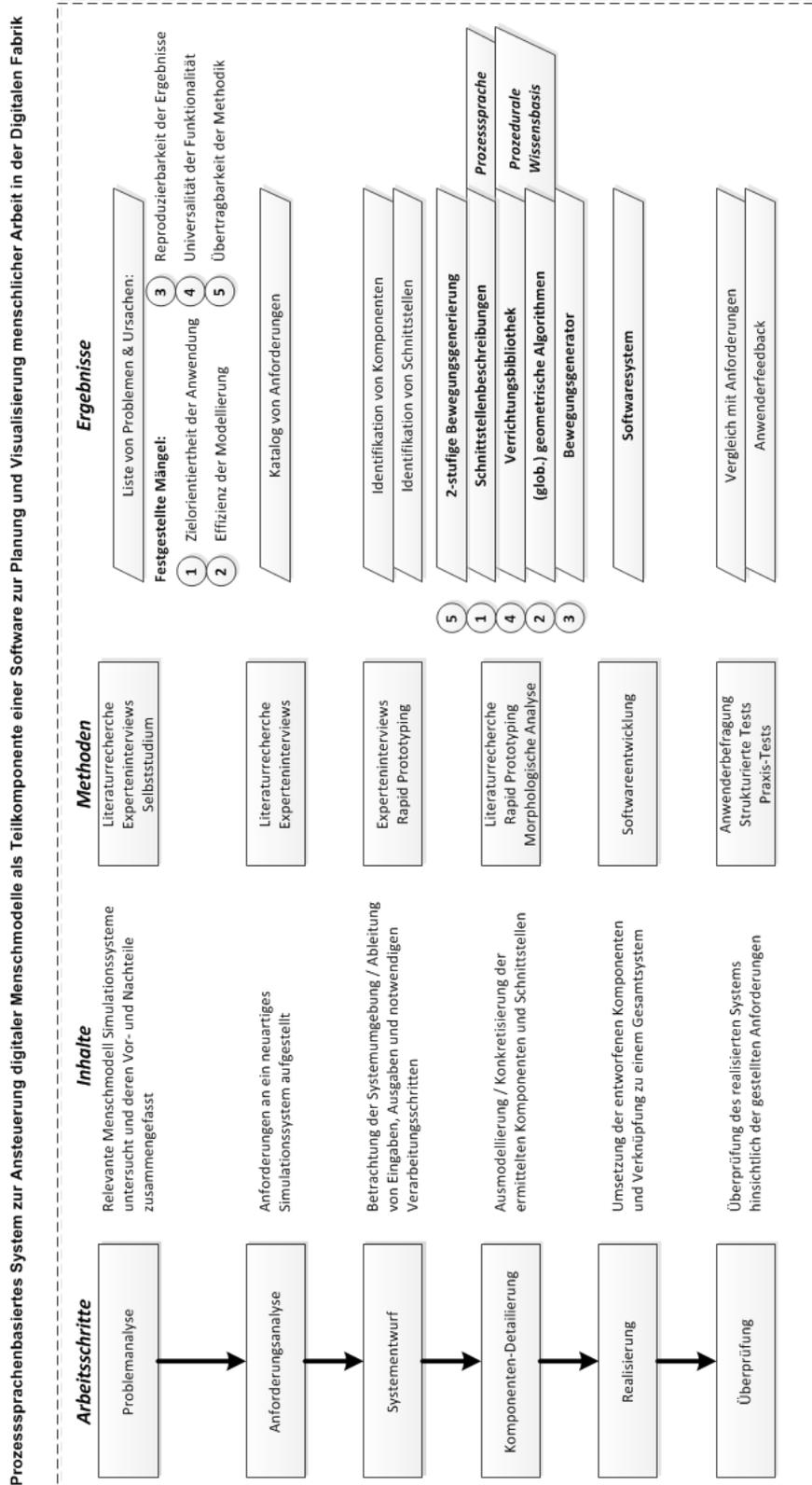
These 6:

Der Paradigmenwechsel von der Erzeugung einzelner Körperhaltungen hin zur Verwendung parametrischer Tätigkeitsbausteine steigert die Effizienz bei der initialen Erstellung von Menschmodell-Simulationen und reduziert den Aufwand bei nachträglichen Änderungen und Optimierungen.

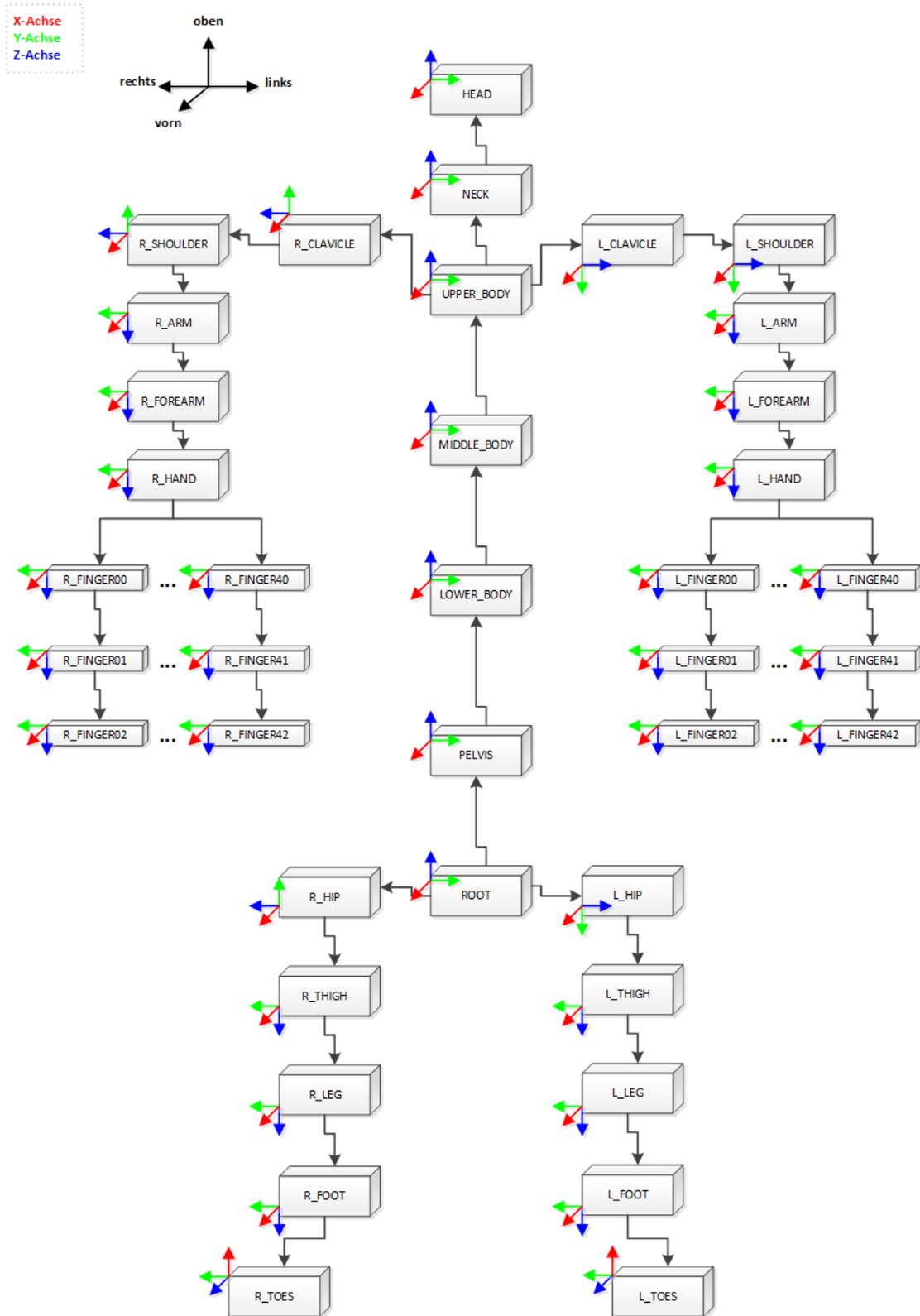
These 7:

Der Verzicht auf manuelle Anpassungen der Körperhaltungen bei der Bewegungssimulation von digitalen Menschmodellen führt bei der Auswertung der generierten Bewegungen zu objektiven und reproduzierbaren Ergebnissen.

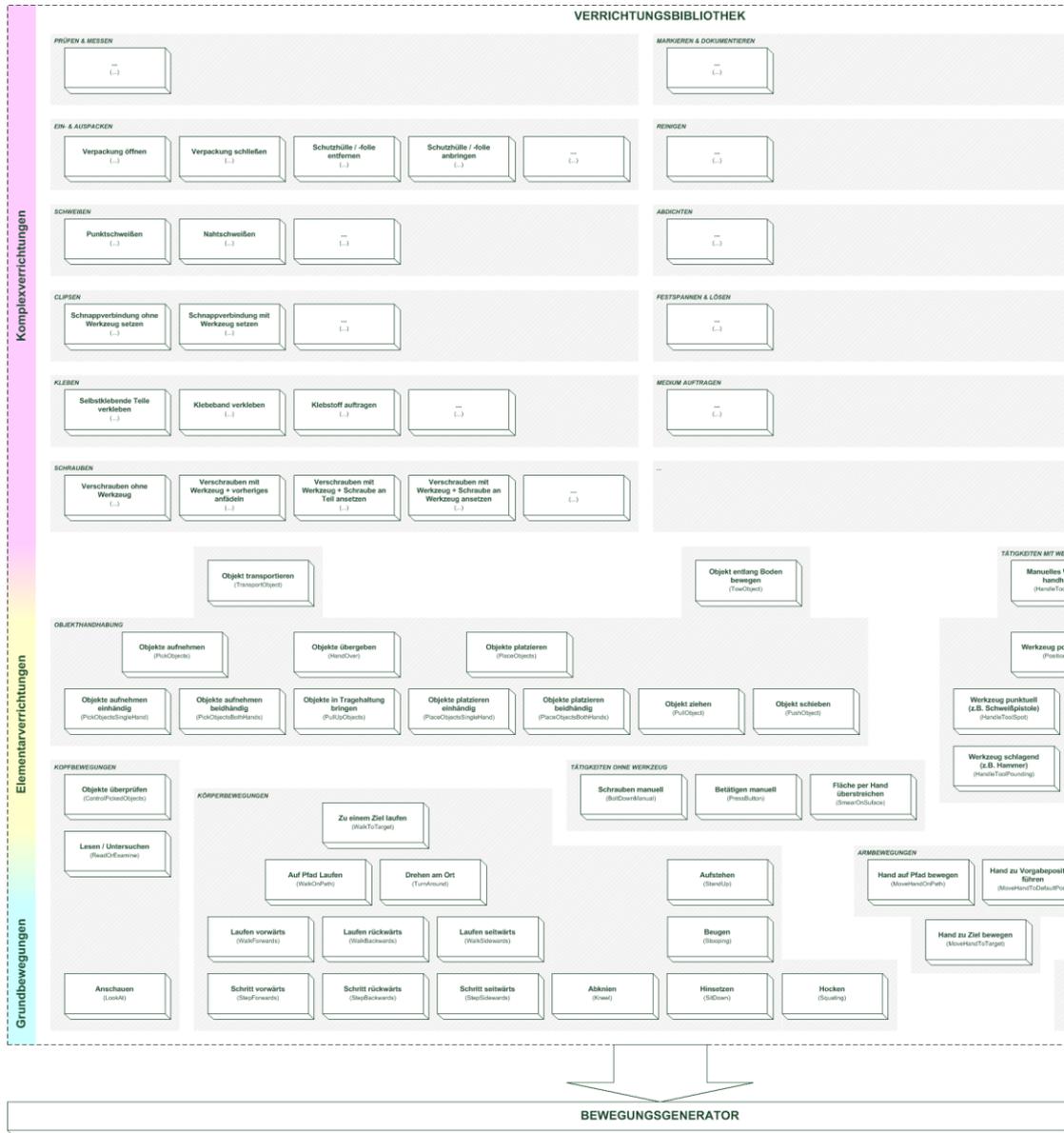
Anhang A – Übersicht der vorliegenden Arbeit



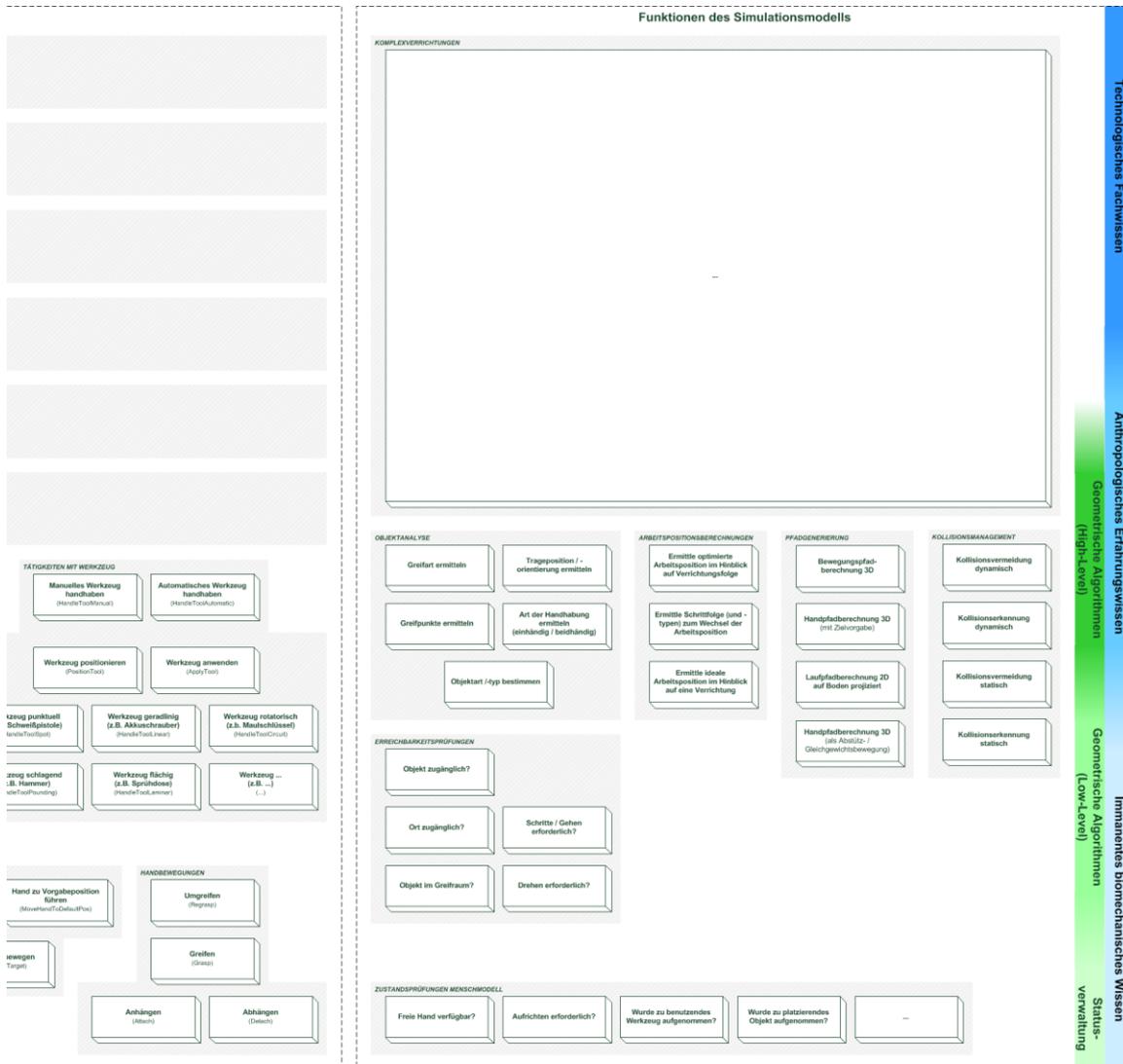
Anhang B – Verwendete Menschmodell-Struktur



Anhang C – Morphologischer Kasten



Anhang C – Morphologischer Kasten



Lebenslauf

Persönliche Daten

Name, Vorname: Bauer, Sebastian
Geburtsdatum: 9. Februar 1982
Geburtsort: Karl-Marx-Stadt, jetzt Chemnitz

Ausbildung

seit 04/2008 Doktorand bei imk automotive GmbH in Chemnitz

10/2001 - 02/2007 Studium „Angewandte Informatik“
an der Technischen Universität Chemnitz

Berufspraxis

seit 02/2010 Kompetenzteamleiter Softwareentwicklung
bei imk automotive GmbH in Chemnitz

11/2006 - 01/2010 Softwareentwickler
bei imk automotive GmbH in Chemnitz

Publikationen

- [1] W. Leidholdt und S. Bauer, Der virtuelle Facharbeiter (Editor menschlicher Arbeit), 12. IFF-Wissenschaftstage, Fraunhofer-Institut für Fabrikbetrieb und -automatisierung IFF, Magdeburg: Fraunhofer Verlag, 2009
- [2] R. Jendrusch und S. Bauer, Der virtuelle Facharbeiter nimmt seine Arbeit auf, 13. IFF-Wissenschaftstage, Fraunhofer-Institut für Fabrikbetrieb und -automatisierung IFF, Magdeburg: Fraunhofer Verlag, 2010
- [3] R. Jendrusch, W. Leidholdt, S. Bauer, H. Hermsdorf, N. Hofmann, A. Keil und B. Spanner-Ulmer, Algorithmische Bewegungsgenerierung und semi-automatische Ergonomiebewertung für digitale Menschmodelle, Zeitschrift für Arbeitswissenschaft 03/2011, 2011
- [4] L. Fritzsche, R. Jendrusch, W. Leidholdt, S. Bauer, T. Jäckel und A. Pirger, Introducing ema (Editor for Manual Work Activities) – A New Tool for Enhancing Accuracy and Efficiency of Human Simulations in Digital Pro-

- duction Planning. In V. Duffy, Digital Human Modeling, HCII 2011, LNCS 6777 (S. 272–281). Berlin: Springer, 2011
- [5] K. Schaub, J. Mühlstedt, B. Illmann, S. Bauer, L. Fritzsche, T. Wagner, A. Bullinger-Hoffmann und R. Bruder, Ergonomic assessment of automotive assembly tasks with digital human modelling and the ‘ergonomics assessment worksheet’ (EAWS), In J. Human Factors Modelling and Simulation, Vol. 3, Nos. 3/4, 2012
- [6] L. Fritzsche, W. Leidholdt, S. Bauer, T. Jäckel und A. Moreno, Interactive Production Planning and Ergonomic Assessment with Digital Human Models – Introducing the Editor for Manual Work Activities (ema), IEA 2012: 18th World congress on Ergonomics - Designing a sustainable future, In Work: A Journal of Prevention, Assessment and Rehabilitation, Vol. 41, Supplement 1/ 2012
- [7] L. Fritzsche, B. Illmann, W. Leidholdt, S. Bauer und A. Moreno, ema (Editor for Manual Work Activities) – A Holistic Method for Efficient Production Planning based on Digital Human Simulation, 2nd International Digital Human Modeling Symposium, The University of Michigan Transportation Research Institute, Ann Arbor, Michigan, 2013
- [8] B. Illmann, L. Fritzsche, W. Leidholdt, S. Bauer und M. Dietrich, Application and Future Developments of EMA in Digital Production Planning and Ergonomics, In V. Duffy, Digital Human Modeling, HCII 2013, Part II, LNCS 8026 (S. 66–75), Berlin: Springer, 2013
- [9] D. Gläser, L. Fritzsche, S. Bauer und W. Leidholdt, The quest to validate human motion
for digital ergonomic assessment – Biomechanical studies to improve the human-like behavior of the human model EMA, Applied Human Factors and Ergonomics Conference AHFE International, Krakau, 2014
- [10] G. Heuwieser, S. Bauer, B. Illmann, L. Fritzsche, Vollständige Integration von I.E.- Standards in digitale Menschmodelle - ema lernt fließend MTM-UAS -, 60. Frühjahrskongress der Gesellschaft für Arbeitswissenschaft, München, 2014

