



Künstliche Intelligenz für Ingenieure

von
Prof. Dr.-Ing. Jan Lunze

2., völlig überarbeitete Auflage

Oldenbourg Verlag München

Prof. Dr.-Ing. Jan Lunze studierte von 1970 bis 1974 Technische Kybernetik an der TU Ilmenau; 1980 folgte die Promotion auf dem Gebiet der dezentralen Regelung, 1983 die Habilitation über robuste Regelung. Von 1992 bis 2001 war Jan Lunze Professor für Regelungstechnik an der TU Hamburg-Harburg. Seit 2001 ist er Leiter des Lehrstuhls für Automatisierungstechnik und Prozessinformatik der Ruhr-Universität Bochum. Gegenstand seiner Forschungsarbeiten sind u.a. die Gebiete der Prozessüberwachung und der Fehlerdiagnose ereignisdiskreter und hybrider Systeme.

Bibliografische Information der Deutschen Nationalbibliothek

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

© 2010 Oldenbourg Wissenschaftsverlag GmbH
Rosenheimer Straße 145, D-81671 München
Telefon: (089) 45051-0
oldenbourg.de

Das Werk einschließlich aller Abbildungen ist urheberrechtlich geschützt. Jede Verwertung außerhalb der Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Bearbeitung in elektronischen Systemen.

Lektorat: Anton Schmid
Herstellung: Anna Grosser
Coverentwurf: Kochan & Partner, München
Gedruckt auf säure- und chlorfreiem Papier
Gesamtherstellung: Grafik + Druck GmbH, München

ISBN 978-3-486-70222-4

Vorwort

Zur Lösung vieler ingenieurtechnischer Probleme muss man Wissen logisch verarbeiten. So wird aus dem Aufbau eines zu montierenden Gerätes durch logisches Schlussfolgern die Reihenfolge abgeleitet, in der die Teile aneinander zu fügen sind, und daraus die Handlungsfolge für einen Roboter festgelegt. Diagnosesysteme bilden aus Messwerten diskrete Kenngrößen für den aktuellen Prozesszustand und ziehen daraus Schlussfolgerungen über die möglicherweise fehlerbehafteten Komponenten. Intelligente Verkehrsleiteinrichtungen steuern den Straßenverkehr, indem sie aus dem aktuellen Verkehrsaufkommen Geschwindigkeitsvorgaben ableiten und gegebenenfalls Verkehrsströme umleiten.

Um derartige Entscheidungen treffen zu können, ist Wissen über die Problemstellung erforderlich, Beurteilungsvermögen für die möglichen Entscheidungsalternativen sowie die Fähigkeit, Schlussfolgerungen zu ziehen. Die Künstliche Intelligenz liefert wichtige Grundlagen, um technische Anlagen dazu zu befähigen bzw. um Ingenieure beim Entwurf und der Überwachung solcher Anlagen zu unterstützen. Im Unterschied zu der im Ingenieurbereich verbreiteten Vorgehensweise, Aufgaben in numerisch lösbare Probleme zu überführen, eignen sich die in diesem Buch behandelten Methoden vor allem für diskrete Entscheidungsprobleme, die mit symbolischer Informationsverarbeitung gelöst werden. Für die Realisierung der eingangs genannten Systeme werden häufig beide Vorgehensweisen kombiniert.

Dieses Lehrbuch stellt die Grundlagen der Künstlichen Intelligenz für Ingenieure dar. Es gibt eine detaillierte Einführung in die wichtigsten Methoden der Wissensrepräsentation und der Wissensverarbeitung und zeigt, wie diese Methoden in ingenieurtechnischen Anwendungen eingesetzt werden können.

Damit der Brückenschlag von der Denkwelt der Ingenieure zur Herangehensweise der Künstlichen Intelligenz gelingt, wird auf zwei Dinge besonderer Wert gelegt. Erstens konzentriert sich dieses Lehrbuch auf die in der Technik einsetzbaren Methoden. Neben den logischen Grundlagen der Künstlichen Intelligenz werden die in den letzten Jahren entstandenen Verarbeitungsmethoden für Wissen mit Unsicherheiten behandelt und dabei Verbindungen zu den im Ingenieurbereich eingesetzten wahrscheinlichkeitstheoretischen Methoden hergestellt.

Zweitens wird an zahlreichen Beispielen gezeigt, dass es im Tätigkeitsfeld der Ingenieure viele Probleme gibt, die nicht mit den bewährten Methoden der Ingenieurwissenschaften gelöst werden können, sondern Darstellungs- und Verarbeitungsprinzipien für logisches Denken erfordern. Aus diesem Grunde werden in der ingenieurtechnischen Praxis heute Bayesnetze für die Beschreibung ungenau bekannter Ursache-Wirkungsbeziehungen eingesetzt, Fehlerdiagnosemethoden mit Fehlerbäumen und logischen Modellen, regelbasierte Verfahren zur Lösung von Planungsaufgaben, fuzzylogische Methoden für die Regelung verfahrenstechnischer Prozesse und heuristische Algorithmen für die Routenplanung und die Fahrplangestaltung. Diese und weitere Beispiele sollen die Leser in die Lage versetzen, die Wirksamkeit der hier behandelten

Lösungsansätze für ingenieurtechnische Aufgaben zu beurteilen und neue Anwendungsgebiete zu erschließen.

Inhalt. In den ersten zwei Kapiteln wird das Anliegen der Künstlichen Intelligenz erläutert und anhand eines einfachen Beispiels gezeigt, welche grundlegenden Probleme bei der Wissensverarbeitung zu lösen sind. Dann konzentriert sich das Lehrbuch auf drei Schwerpunkte:

- **Suche:** Eine wichtige Grundlage der Künstlichen Intelligenz bilden Algorithmen für die Suche in Grafen, die sich in allgemeinerer Form in der Wissensverarbeitung wiederfinden. Viele Begriffe sowie die grundlegende Architektur von Suchsystemen werden im Kapitel 3 für die Grafensuche eingeführt und in den nachfolgenden Kapiteln für regelbasierte und logikbasierte Systeme erweitert.
- **Logik:** In den Kapiteln 4, 6 und 7 werden Regeln und logische Formeln als grundlegende Repräsentationsformen für Wissen behandelt. Ausführlich wird das Resolutionsprinzip zur Verarbeitung derartigen Wissens erläutert und an Beispielen demonstriert.
- **Verarbeitung unsicheren Wissens:** Die Erweiterung der logikbasierten Methoden für die Darstellung und Verarbeitung von unsicherem Wissen in den Kapiteln 10 bis 12 ist für Ingenieur Anwendungen sehr wichtig und führt u. a. auf die in der Praxis vielfach eingesetzten Bayesnetze.

Das Lehrbuch schließt mit einer kurzen Einführung in die Wissensverarbeitung mit strukturierten Objekten und einer Zusammenfassung der Merkmale und der technischen Anwendungsgebiete der Wissensverarbeitung in den Kapiteln 13 und 14. Dabei werden auch Begriffe erläutert, die in der modernen Literatur zu finden sind, aber bei der Darstellung der Grundideen zunächst ausgespart wurden.

Für die Künstlichen Intelligenz gilt in besonderem Maße, dass eine Methode nur dann gut ist, wenn sie sich gut implementieren lässt. Dieser Aspekt wird dadurch hervorgehoben, dass die Erläuterung der Methoden stets in Algorithmen mündet, die problemlos in einer geeigneten Programmiersprache implementiert werden können. Heutige Programmiersprachen verfügen über alle diejenigen Konstrukte, die für Suchverfahren und Symbolmanipulationsmethoden in unterschiedlichen Domänen notwendig sind.

In den Text eingefügt sind zwei Kapitel zur **Softwaretechnik** der Künstlichen Intelligenz. Kapitel 5 zeigt mit einer Einführung in die Programmiersprache LISP, dass die symbolische Informationsverarbeitung in zweckmäßiger Weise auf eine Manipulation von Listen zurückgeführt werden kann. Mit PROLOG wird im Kapitel 8 eine Programmiersprache behandelt, bei der der Interpreter die Suche nach einer Lösung selbst organisiert, so dass die Anwender ihre Probleme nur noch deklarativ zu formulieren brauchen. Beide Kapitel sind keine erschöpfenden Programmieranleitungen, sondern zeigen mit der funktionalen und der logischen Programmierung, welche neuen Programmierstile für die Methoden der Künstlichen Intelligenz zweckmäßig sind. Leser, die sich nur für die Methoden der Künstlichen Intelligenz interessieren, können diese Kapitel ohne Weiteres überspringen.

Literaturhinweise am Ende jedes Kapitels weisen auf interessante Originalarbeiten sowie Monografien für ein vertiefendes Studium der behandelten Themen hin. **Übungsaufgaben** regen die Leser dazu an, sich über die hier behandelten Probleme hinaus das Anwendungsgebiet der Künstlichen Intelligenz zu erschließen. Die Lösung der wichtigsten Aufgaben ist im An-

hang 1 zu finden. Mit der **Projektaufgabe** (Anhang 4) können alle behandelten Methoden an einem durchgängigen Anwendungsbeispiel erprobt und verglichen werden. Die **Fragen zur Prüfungsvorbereitung** unterstützen die Wiederholung des behandelten Stoffes.

Leser. Dieses Lehrbuch wendet sich in erster Linie an Ingenieure, die die Methoden der Künstlichen Intelligenz in Kombination mit ingenieurtechnischen Verfahren einsetzen wollen. Die hier vermittelte fachübergreifende Sicht ist notwendig, weil „intelligente Maschinen“ nur durch die Kombination von Methoden der Ingenieurwissenschaften mit denen der Informatik entstehen können. Das Buch ist auch für die in der Praxis tätigen Ingenieure gedacht, die die Wissensverarbeitungsmethoden in ihrer Ausbildung nicht kennengelernt haben und sich für eine Erweiterung ihres Methodenspektrums interessieren.

Das Buch setzt nur mathematische Grundkenntnisse und eine gewisse Vertrautheit mit der Programmierung voraus. Die Querbezüge von den Methoden der Künstlichen Intelligenz zu systemtheoretischen Verfahren für die Modellierung und die Analyse kontinuierlicher und ereignisdiskreter dynamischer Systeme gehen über diesen Rahmen hinaus, sind aber für das Verständnis der restlichen Teile des Buches nicht notwendig.

Zweite Auflage. Die vorliegende zweite Auflage dieses Lehrbuchs entstand aus einer Neustrukturierung und Überarbeitung der ersten Auflage, in der die damals in zwei Bänden getrennt behandelten Methoden und Anwendungsbeispiele zusammengeführt wurden.

In der Zeit, die seit dem Erscheinen der ersten Auflage vergangen ist, hat sich die Auffassung der Ingenieure von der Künstlichen Intelligenz in mehrerer Weise verändert. In der Mitte der 1990er Jahre versprach man sich von den Methoden der Künstlichen Intelligenz nicht nur neue Repräsentations- und Verarbeitungsformen für Wissen, sondern auch eine neue Softwaretechnik, insbesondere das *rapid prototyping*, und Vorteile durch die avisierte spezielle Hardware, das automatische Programmieren und nichtalgorithmische Lösungsmethoden für Entscheidungsprobleme. Seither haben sich einige dieser Ideen im Ingenieurbereich etabliert, ohne dass man ihre Herkunft stets betont, beispielsweise in der objektorientierten Gestaltung von Softwaresystemen, durch den verbreiteten Einsatz von Klassifikationsmethoden oder im *Model Checking* zur Verifikation diskreter Steuerungen. Eine spezielle Hardware ist heute nicht mehr erforderlich, weil die moderne Rechentechnik auch symbolische Informationen in ausreichender Geschwindigkeit verarbeiten kann.

Allerdings werden die Methoden der Künstlichen Intelligenz noch nicht in der Breite eingesetzt, in der man es sich vor 15 Jahren erhofft hat. In technischen Anwendungsgebieten ist der Einsatz der Wissensverarbeitung auf die Bereiche konzentriert, in denen die Anwendung der Logik unumgänglich ist.

Dieser Wandel hat eine grundlegende Umgestaltung dieses Lehrbuchs erforderlich gemacht, in die auch meine Erfahrungen aus einer gleichnamigen Vorlesung eingegangen sind, die ich seit mehr als 20 Jahren an verschiedenen deutschen Universitäten gehalten habe. Die zweite Auflage des Buchs konzentriert sich auf die logischen Grundlagen der Künstlichen Intelligenz und geht nur noch punktuell auf die Softwaretechnik ein.

Danksagung. Ein Buch an der Grenze zwischen Künstlicher Intelligenz und Ingenieurwissenschaften kann nicht ohne einen intensiven Gedankenaustausch mit Vertretern beider Richtungen entstehen. Besonders erwähnen möchte ich gemeinsame Forschungsprojekte und intensive Diskussionen mit Prof. Dr. MARCEL STAROSWIECKI (Lille), Prof. Dr. MOGENS BLANKE (Lyngby), Prof. Dr.-Ing. VOLKER KREBS (Karlsruhe), Dr. LOUISE TRAVE-MASSUYES (Toulouse), Prof. Dr. PETER STRUSS (München) und Prof. Dr.-Ing. FRANK SCHILLER (München). Herr Dr.-Ing. JAN RICHTER (Nürnberg) hat über mehrere Jahre die Übungen zu meiner gleichnamigen Vorlesung gehalten und viele neue Ideen in die Gestaltung der Lehrveranstaltung eingebracht. Eine Reihe von Beispielen habe ich für Weiterbildungsveranstaltungen in der Industrie entwickelt, bei denen nicht primär die Methodik an sich, sondern vor allem die Anwendungsaspekte der hier behandelten Methoden im Mittelpunkt standen. Auch die Diskussionen mit den Industrievertretern haben wesentlich zur Gestaltung dieses Lehrbuchs beigetragen.

Mein Dank gilt weiterhin Frau ANDREA MARSCHALL, die mit großer Geduld und Sorgfalt die Bilder gestaltet hat, Frau Dipl.-Ing. KATRIN LUNZE und Frau HANNELORE HUPP, die die mehrfache Überarbeitung des Textes unterstützt haben, sowie dem Oldenbourg Wissenschaftsverlag für die stets gute Zusammenarbeit. Schließlich danke ich den „Hurtigruten“ für einen ruhigen Platz auf dem Panoramadeck der MS „Mitnatsol“, auf dem ich die letzten Korrekturen an dieser Auflage vorgenommen habe. Alle verbliebenen Fehler sind auf die herrliche norwegische Landschaft zurückzuführen, an der ich in dieser Zeit vorbeifuhr.

Münster, im Juli 2010

J. Lunze

Inhaltsverzeichnis

Verzeichnis der Anwendungsbeispiele	xvii
--	------

Hinweise zum Gebrauch des Buches	xxiii
---	-------

Teil 1: Problemlösen durch Suche

1 Das Fachgebiet Künstliche Intelligenz	1
1.1 Anliegen der Künstlichen Intelligenz	1
1.2 Ausgangspunkte	6
1.2.1 Mathematische Logik	6
1.2.2 Algorithmentheorie	8
1.2.3 Rechentechnik	12
1.3 Kurzer historischer Rückblick	13
1.3.1 Geburtsstunde: Dartmouth-Konferenz 1956	13
1.3.2 Die klassische Epoche: Spiele und logisches Schließen	13
1.3.3 Erste Erfolge: Verstehen natürlicher Sprache	15
1.3.4 Wissensbasierte Systeme und KI-Markt	15
1.3.5 Entwicklungstrend: Kognitive Systeme	16
1.4 Ingenieurtechnische Anwendungen der Künstlichen Intelligenz	17
1.4.1 Grundstruktur intelligenter technischer Systeme	17
1.4.2 Intelligente Agenten	21
1.4.3 Impulse der Künstlichen Intelligenz für die Lösung ingenieurtechnischer Probleme	23
1.5 Möglichkeiten und Grenzen der Künstlichen Intelligenz	26
Literaturhinweise	28
2 Einführungsbeispiel	31
2.1 Qualitative und quantitative Beschreibung eines Wasserversorgungssystems ..	31
2.2 Einfache Methoden zur Verarbeitung von Regeln	37
2.2.1 Umformung der Wissensbasis	37
2.2.2 Verschachtelung der Regeln in einem Entscheidungsbaum	38
2.2.3 Anordnung der Regeln als Wissensbasis	42
2.3 Probleme der Wissensverarbeitung	43
Literaturhinweise	44

3	Grafensuche	45
3.1	Grundbegriffe der Graphentheorie	45
3.1.1	Ungerichtete Grafen	45
3.1.2	Suchprobleme	47
3.2	Bestimmung von Erreichbarkeitsbäumen	48
3.2.1	Tremaux-Algorithmus	48
3.2.2	Geradeaussuche	49
3.2.3	Breite-zuerst-Suche	52
3.2.4	Tiefe-zuerst-Suche	56
3.2.5	Eigenschaften der Suchalgorithmen	59
3.3	Bestimmung von Pfaden	59
3.3.1	Tiefe-zuerst-Suche von Pfaden	59
3.3.2	Optimale Pfade	62
3.3.3	DIJKSTRA-Algorithmus	62
3.3.4	Gleiche-Kosten-Suche	72
3.4	Heuristische Suche	72
3.4.1	Erweiterungsmöglichkeiten der blinden Suche	72
3.4.2	A*-Algorithmus	74
3.5	Anwendungsbeispiel: Bahnplanung für Industrieroboter	82
3.5.1	Aufgabenstellung und Lösungsweg	82
3.5.2	Beschreibung kollisionsfreier Bahnen im Konfigurationsraum	83
3.5.3	Planungsalgorithmus	85
3.5.4	Erweiterungen	86
3.6	Zusammenfassung	87
3.6.1	Problemlösen durch Suche	87
3.6.2	Struktur und Eigenschaften von Suchsystemen	88
	Literaturhinweise	90
4	Regelbasierte Wissensverarbeitung	91
4.1	Zustandsraumdarstellung von Wissensverarbeitungsproblemen	91
4.1.1	Darstellung von Wissen in Form von Regeln	91
4.1.2	Zustandsraumdarstellung	93
4.1.3	Wissensverarbeitung als Grafensuche	97
4.2	Problemlösen durch Vorwärtsverkettung von Regeln	99
4.2.1	Vorwärtsverkettung	99
4.2.2	Verarbeitung von Schlussfolgerungsregeln	99
4.2.3	Verarbeitung von Aktionsregeln	103
4.2.4	Beispiel: Zusammenfassung von Widerstandsnetzwerken	103
4.2.5	Kommutative und nichtkommutative regelbasierte Systeme	110
4.2.6	Beispiel: Lösung von Packproblemen	113
4.3	Problemlösen durch Rückwärtsverkettung von Regeln	119
4.3.1	Rückwärtsverkettung	119
4.3.2	Anwendungsgebiete der Rückwärtsverkettung	122
4.4	Architektur und Einsatzgebiete regelbasierter Systeme	123
4.4.1	Allgemeiner Wissensverarbeitungsalgorithmus	123

4.4.2	Architektur regelbasierter Systeme	125
4.4.3	Einsatzcharakteristika regelbasierter Systeme	126
	Literaturhinweise	129
5	Funktionale Programmierung in LISP	131
5.1	Einführung in die funktionale Programmierung	131
5.1.1	Grundidee von LISP	131
5.1.2	Rekursive Funktionen	134
5.2	Syntax von LISP	136
5.2.1	Listen	136
5.2.2	LISP-Ausdrücke	139
5.2.3	Spezielle Auswertungsregeln	140
5.2.4	Verarbeitung von Listen	143
5.2.5	Definition von Funktionen	149
5.2.6	Bedingte Anweisungen und Let-Konstruktionen	150
5.3	Programmbeispiel: Tiefe-zuerst-Suche in Grafen	152
5.3.1	Programmelemente	152
5.3.2	Zusammenfassung zur Funktion „Erreichbarkeitsbaum“	157
5.4	Merkmale der Programmiersprache LISP	160
	Literaturhinweise	162

Teil 2: Logikbasierte Wissensverarbeitung

6	Aussagenlogik	163
6.1	Einführung in die logikbasierte Wissensverarbeitung	163
6.2	Grundlagen der Aussagenlogik	164
6.2.1	Aussagen und logische Ausdrücke	164
6.2.2	Semantik logischer Ausdrücke	168
6.2.3	Logische Gesetze	175
6.2.4	Logische Ausdrücke in Klauselform	177
6.3	Aussagenkalkül	179
6.3.1	Folgerungen	179
6.3.2	Ableitungsregeln der Aussagenlogik	181
6.3.3	Beweis aussagenlogischer Ausdrücke	185
6.3.4	Formale Systeme der Aussagenlogik	188
6.3.5	Eigenschaften des Aussagenkalküls	189
6.4	Problemlösen durch Resolution	190
6.4.1	Resolutionsprinzip der Aussagenlogik	190
6.4.2	Widerspruchsbeweis	192
6.4.3	Resolutionskalkül	194
6.4.4	Steuerung des Inferenzprozesses	197
6.5	Anwendungsbeispiel: Verifikation von Steuerungen	202
	Literaturhinweise	205

7	Prädikatenlogik	207
7.1	Grundlagen der Prädikatenlogik	207
7.1.1	Prädikate, logische Ausdrücke und Aussageformen	207
7.1.2	Prädikatenlogische Ausdrücke in Klauselform	212
7.1.3	Semantik prädikatenlogischer Ausdrücke	214
7.2	Prädikatenkalkül	216
7.2.1	Resolutionsregel der Prädikatenlogik	216
7.2.2	Resolutionskalkül	219
7.2.3	Merkmale von Resolutionssystemen	222
7.3	Resolutionswiderlegung in der logischen Programmierung	224
7.3.1	Resolutionsregel für Hornklauseln	224
7.3.2	Beweisverfahren der logischen Programmierung	227
7.4	Logik als Grundlage der Wissensrepräsentation und der Wissensverarbeitung	231
7.4.1	Modellierung technischer Systeme durch logische Ausdrücke	231
7.4.2	Beispiel: Prädikatenlogische Beschreibung von Planungsaufgaben	232
7.4.3	Vergleich von regelbasierter und logikbasierter Wissensverarbeitung	237
7.4.4	Erweiterungsmöglichkeiten der klassischen Logik	238
	Literaturhinweise	239
8	Logische Programmierung in PROLOG	241
8.1	Einführung in die logische Programmierung	241
8.2	Syntax von PROLOG	242
8.3	Abarbeitung logischer Programme	244
8.3.1	Semantik logischer Programme	244
8.3.2	Steuerfluss bei der Verarbeitung logischer Programme	245
8.3.3	Interpretation des Ergebnisses	249
8.4	Programmelemente	251
8.4.1	Verarbeitung von Listen	251
8.4.2	Rekursive Programmierung	253
8.4.3	Built-in-Prädikate	254
8.5	Programmbeispiele	259
8.5.1	Bestimmung von Pfaden in gerichteten Grafen	259
8.5.2	Zusammenfassung eines Widerstandsnetzwerkes	265
8.5.3	Handlungsplanung für Roboter	269
8.6	Anwendungsgebiete von PROLOG	274
	Literaturhinweise	276

Teil 3: Verarbeitung unsicheren Wissens

9	Nichtmonotones Schließen und ATMS	277
9.1	Probleme und Lösungswege für die Verarbeitung unsicheren Wissens	277
9.1.1	Quellen für die Unbestimmtheiten der Wissensbasis	278
9.1.2	Probleme der Darstellung und der Verarbeitung unsicheren Wissens	281

9.1.3	Überblick über die Behandlungsmethoden für unsicheres Wissen	282
9.2	Darstellung veränderlichen Wissens	283
9.3	Grundidee des ATMS	287
9.3.1	Begründungen	287
9.3.2	ATMS-Graf	288
9.3.3	Lokale und globale Umgebungen	290
9.4	Erweiterungen	294
9.4.1	Verwaltung logischer Ausdrücke	294
9.4.2	Behandlung logischer Widersprüche	296
9.4.3	Zusammenspiel von Problemlöser und ATMS	299
9.5	Anwendungsbeispiel: Analyse eines verfahrenstechnischen Prozesses	301
9.5.1	Aussagenlogisches Modell	301
9.5.2	Bildung des ATMS-Grafen	303
9.5.3	Analyse und Prozessüberwachung mit dem ATMS	307
9.6	Fehlerdiagnose mit ATMS	311
9.6.1	Modellbasierte Diagnose	311
9.6.2	Diagnoseprinzip GDE	314
9.6.3	Realisierung von GDE mit einem ATMS	315
9.6.4	Erweiterungen	319
	Literaturhinweise	324
10	Mehrwertige und unscharfe Logik	325
10.1	Mehrwertige Logiken	325
10.1.1	Logische Ausdrücke der dreiwertigen Logik	325
10.1.2	Ableitungsregel und Theorembeweisen	327
10.1.3	Erweiterung von dreiwertiger auf mehrwertige Logiken	329
10.2	Wissensverarbeitung mit unscharfen Mengen	330
10.2.1	Unscharfe Mengen	330
10.2.2	Unscharfe Mengen in der Wissensrepräsentation	334
10.2.3	Unscharfe Logik	337
10.2.4	Fuzzifizierung und Defuzzifizierung	342
10.2.5	Anwendungsbeispiel: Fuzzyregelung	345
	Literaturhinweise	354
11	Probabilistische Logik und Bayesnetze	355
11.1	Wahrscheinlichkeitstheoretische Modelle	355
11.1.1	Übersicht über die wahrscheinlichkeitstheoretische Behandlung unsicheren Wissens	355
11.1.2	Aussagenlogische Beschreibung zufälliger Ereignisse	356
11.1.3	Wahrscheinlichkeit logischer Ausdrücke	360
11.2	Probabilistische Logik	364
11.2.1	Modus Ponens der probabilistischen Logik	364
11.2.2	Fehlende Modularität der probabilistischen Logik	368
11.2.3	Bayessche Inferenzregel	372
11.2.4	Lösung von Diagnoseaufgaben	374

11.2.5	Aussagekraft probabilistischer Folgerungen	377
11.2.6	Anwendungsgebiete der probabilistischen Logik	379
11.3	Bayesnetze	380
11.3.1	Abhängige und unabhängige Ereignisse	380
11.3.2	Modellbildung mit Bayesnetzen	388
11.3.3	Kausales Schließen mit Bayesnetzen	397
11.3.4	Diagnostisches Schließen mit Bayesnetzen	403
11.3.5	Erweiterung der Bayesnetze	408
11.4	Zusammenfassung und Wertung	409
	Literaturhinweise	412
12	Heuristische Verfahren zur Darstellung und zur Verarbeitung unsicheren Wissens	413
12.1	Wissensverarbeitung auf der Grundlage der Evidenztheorie	413
12.1.1	Grundlagen der Evidenztheorie	414
12.1.2	Dempster-Regel	417
12.1.3	Erweiterung der Aussagenlogik mit Hilfe der Evidenztheorie	418
12.2	Heuristische Methoden	422
12.2.1	Beschreibung der Unbestimmtheit des Wissens durch Konfidenz- faktoren	422
12.2.2	Verarbeitung der Konfidenzfaktoren bei Ableitungen	424
12.3	Vergleichende Zusammenfassung der Methoden zur Verarbeitung unsicheren Wissens	426
	Literaturhinweise	428
13	Wissensverarbeitung mit strukturierten Objekten	429
13.1	Begriffsbildung und strukturierte Objekte	429
13.1.1	Begriffshierarchien und Vererbung von Eigenschaften	430
13.1.2	Multihierarchien und Sichten	432
13.2	Semantische Netze	433
13.2.1	Syntax und Semantik	433
13.2.2	Kausale Netze	435
13.3	Frames	436
13.3.1	Grundidee der Wissensrepräsentation mit Frames	436
13.3.2	Anordnung von Frames in Generalisierungshierarchien	439
13.3.3	Erweiterungsmöglichkeiten	440
13.3.4	Vergleich von Frames mit anderen Wissenrepräsentationsformen	442
	Literaturhinweise	443
14	Merkmale und technische Anwendungsgebiete der Wissensverarbeitung	445
14.1	Struktur wissensbasierter Systeme	445
14.2	Wissensrepräsentation	446
14.2.1	Modellbildung	446
14.2.2	Deklaratives und prozedurales Wissen	448
14.2.3	Anforderungen an die Wissensrepräsentation	449

14.2.4 Wissensrepräsentationsmodelle	451
14.2.5 Modularität der Wissensrepräsentation	453
14.2.6 Wissenserwerb	454
14.3 Wissensverarbeitung	455
14.3.1 Zusammenfassung der Verarbeitungsmethoden	455
14.3.2 Problemspezifikation und Algorithmmierung	457
14.4 Ingenieurtechnische Anwendungsgebiete	459
Literaturverzeichnis	465

Anhänge

Anhang 1: Lösung von Übungsaufgaben	471
Anhang 2: Grundlagen der Wahrscheinlichkeitsrechnung	493
Anhang 3: Aufgaben zur Prüfungsvorbereitung	499
Anhang 4: Projektaufgabe	503
Anhang 5: Fachwörter deutsch – englisch	507
Sachwortverzeichnis	511

Verzeichnis der Anwendungsbeispiele

Intelligente Roboter

Intelligente Robotersteuerung (Beispiel 1.1)	20
• Handlungsplanung und Bahnplanung für Industrieroboter	
Bahnplanung für Industrieroboter (Abschn. 3.5)	82
Bahnplanung für einen Roboter mit zwei Freiheitsgraden (Aufgabe 3.8)	86
Regeln für die Handlungsplanung von Robotern (Aufgabe 4.11)	118
Wissensbasis für die Handlungsplanung (Beispiel 13.2)	438
PROLOG-Programm für die Handlungsplanung (Abschn. 8.5.3)	269
Erweiterung des PROLOG-Programms (Aufgabe 8.8)	272
• Handlungsplanung für einen Containerbahnhof	
Zustandsraum der „Klötzchenwelt“ (Aufgabe 4.1 mit Lösung)	97, 474
Handlungsplanung für einen Containerbahnhof (Aufgabe 5.6)	160
Handlungsplanung mit dem Planungssystem STRIPS (Beispiel 7.5)	233
Handlungsplanung für zwei kooperierende Roboter (Aufgabe 7.8)	236
Handlungsplanung für einen Serviceroboter (Aufgabe 7.9)	236
PROLOG-Programm zum Stapeln von Containern (Aufgabe 8.9)	273
Strukturierung der Modellbibliothek für einen Containerbahnhof (Aufgabe 13.3)	443

Fertigungsautomatisierung

Ressourcenzuteilung in einer Fertigungszelle (Aufgabe 8.10)	273
Verifikation von Steuerungen (Abschn. 6.5)	202
Fehleranalyse einer Fertigungszelle (Aufgabe 11.4 mit Lösung)	379, 487
• Lösung eines Packproblems	
Regelbasierte Lösung (Abschn. 4.2.6)	113
Lösung bei unterschiedlichem Elementevorrat (Aufgabe 4.10)	118

Intelligente Messsysteme

Verarbeitung von Fehlern bei einer Druckmessung (Beispiel 10.2)	335
Wahrscheinlichkeitstheoretisches Modell eines Messgerätes (Beispiel 11.5)	384
Sensorfusion mit Hilfe der Evidenztheorie (Aufgabe 12.1)	421

Steuerung verfahrenstechnischer Prozesse

PROLOG-Programm zur Auswertung von Alarmen (Aufgabe 8.11 mit Lösung)	274, 481
Wissenbasis für ein Reglerentwurfssystem (Aufgabe 13.4)	443
• Fuzzyregelung eines Behälters	
Fuzzyregelung des Behälterfüllstandes (Beispiel 10.4)	347
Entwurf eines Fuzzyreglers (Aufgabe 10.3)	352
Erweiterung der Fuzzy-Füllstandsregelung (Aufgabe 10.4)	352
• Diagnose einer Flaschenabfüllanlage	
Modellierung einer Flaschenabfüllanlage mit Bayesnetz (Beispiel 11.8)	394
Bedingt unabhängige Ereignisse bei der Flaschenabfüllung (Aufgabe 11.5 mit Lösung) ...	397, 487
Diagnose mit einem Bayesnetz (Beispiel 11.10)	406
Erweiterung der Diagnose (Aufgabe 11.7)	407
• Modellierung und Überwachung eines Wasserversorgungssystems	
Qualitative und quantitative Beschreibung (Abschn. 2.1)	31
Entscheidungsbaum für die Verhaltensanalyse (Abschn. 2.2.2)	38
Erweiterung des Entscheidungsbaumes (Aufgaben 2.2 und 2.3)	41, 41
Regelbasierte Überwachung des Wasserversorgungssystems (Beispiele 4.1 und 4.2)	92, 100
Suchraum bei der Vorwärtsverkettung von Regeln (Aufgabe 4.3)	102
LISP-Programm zur Vorwärtsverkettung von Regeln (Aufgabe 5.5)	160
Rückwärtsverkettung von Schlussfolgerungsregeln (Beispiel 4.3)	120
Verhaltensvorhersage mit dem Aussagenkalkül (Beispiel 6.6)	186
ATMS-Graf zur Beschreibung des Wasserversorgungssystems (Beispiel 9.1)	309
Vorhersage des Verhaltens mit mehrwertiger Logik (Beispiel 10.1)	328
Fehlerdiagnose mit dreiwertiger Logik (Aufgabe 10.2)	330
Fehlerdiagnose mit unscharfer Logik (Beispiel 10.3)	339
Aussagenlogische Beschreibung zufälliger Ereignisse (Beispiel 11.1)	358
Vorhersage des Verhaltens mit probabilistischer Logik (Beispiel 11.2)	366

Verhaltensvorhersage mit einem Bayesnetz (Beispiel 11.9 und Aufgabe 11.6)	401, 403
Fehlerdiagnose bei niedrigem Wasserdruck (Aufgabe 11.9)	407
Erweitertes Bayesnetz für die Fehlerdiagnose (Aufgabe 11.11 mit Lösung)	409, 491
• Alarmauswertung für einen verfahrenstechnischen Prozess	
Überwachung eines verfahrenstechnischen Prozesses (Abschn. 9.5)	301
Einfache Auswertung von Alarmen (Aufgabe 8.11)	274
Logikbasierte Diagnose eines Behältersystems mit PROLOG (Aufgabe 8.12 mit Lösung) .	274, 483
Anwendung des ATMS auf ein Behältersystem (Aufgabe 9.2 mit Lösung)	309, 483

Fehlerdiagnose von Fahrzeugkomponenten

Fehlerdiagnose eines Motorkühlsystems mit probabilistischer Logik (Beispiel 11.4)	376
Diagnose einer Reifenpanne (Aufgabe 11.2 mit Lösung)	377, 485
Taxonomie für die Fehlersuche in einem Kraftfahrzeug (Aufgabe 13.2)	433
Realisierung eines Fahrzeugconfigurators (Projektaufgabe A4.1)	503
• Beschreibung und Diagnose einer Fahrzeugbeleuchtung	
Fehlerdiagnose einer Fahrzeugbeleuchtung (Aufgabe 1.3)	26
Regelbasierte Analyse durch Vorwärtsverkettung (Aufgabe 4.6 mit Lösung)	103, 475
Aussagenlogische Beschreibung der Fahrzeugbeleuchtung (Aufgabe 6.12)	196
Modellbasierte Diagnose mit einem ATMS (Beispiel 9.2)	313
Beschreibung der fehlerfreien Arbeitsweise durch einen ATMS-Grafen (Beispiel 9.3)	315
Diagnose mit dem Verfahren GDE (Beispiel 9.4 und Aufgabe 9.3)	317, 322
Erweiterung der Diagnose (Beispiel 9.5 und Aufgabe 9.4)	320, 323
Analyse der Fahrzeugbeleuchtung mit probabilistischer Logik (Aufgabe 11.1)	371
Fehlerdiagnose mit einem Bayesnetz (Aufgabe 11.8)	407

Wissensbasierte Systeme in der Verkehrstechnik

Planung einer Fahrtroute in Amsterdam (Aufgabe 3.6 mit Lösung)	80, 473
Routenplanung im Straßen- und Eisenbahnverkehr mit dem A*-Algorithmus (Aufgabe 3.7)	81
• Ampelsteuerung	
Regeln zur Steuerung einer Verkehrsampel (Aufgabe 4.15)	129
Verifikation einer Ampelsteuerung (Beispiel 6.10)	203

• Modellierung und Überwachung eines Parkhauses

Aussagenlogische Beschreibung eines Parkplatzes (Aufgabe 6.1)	164
Bestimmung der Modelle für den Parkplatz (Beispiel 6.3)	171
Interpretation der Parkplatzmodelle (Aufgabe 6.2)	175
Umformung der „Parkplatzformel“ (Aufgabe 6.5)	179
Folgerungen über die Nutzung des Parkplatzes (Beispiel 6.4)	181
Ableitungen über die Nutzung des Parkplatzes (Beispiel 6.5)	183
Stellplatzverwaltung in einem Parkhaus mit Prädikatenlogik (Aufgabe 7.3)	215

Analyse elektrischer und elektronischer Schaltungen

Generalisierungshierarchie von Bauelementen und Baugruppen (Aufgabe 13.1)	433
Beschreibung von elektrischen Bauelementen durch Frames (Beispiel 13.1)	437
Mehrebenenbetrachtungen beim Schaltkreisentwurf (Beispiel 14.1)	460

• Beschreibung und Analyse digitaler Schaltungen

Diagnose einer logischen Schaltung mit einem ATMS (Aufgabe 9.5)	323
Beschreibung einer digitalen Schaltung durch Hornklauseln (Aufgabe 7.7)	230
Prädikatenlogische Beschreibung einer logischen Schaltung (Aufgabe 7.6)	223
Beschreibung eines Inverters in PROLOG (Aufgabe 8.3)	259

• Beschreibung und Analyse einer Sicherheitsschaltung

Aussagenlogische Beschreibung einer 2-aus-3-Sicherheitsschaltung (Beispiel 6.2)	169
Zuverlässigkeitsanforderungen an eine Sicherheitsüberwachung (Aufgabe 11.10 mit Lösung)	408, 489
Analyse einer Sicherheitsschaltung mit 2-aus-3-Logik (Aufgabe 12.2)	425

• Zusammenfassung von Widerstandsnetzwerken

Regelbasierte Zusammenfassung von Widerstandsnetzwerken (Abschn. 4.2.4)	103
Erweiterungen (Aufgaben 4.7 und 4.8)	110, 110
Kommutativität des regelbasierten Systems (Aufgabe 4.9)	112
PROLOG-Programm (Abschn. 8.5.2)	265
Erweiterung des PROLOG-Programms (Aufgabe 8.7)	269

Analyse von Rechnernetzen

Prädikatenlogische Beschreibung eines Rechnernetzes (Beispiel 7.1)	208
Sicherheit von Rechnernetzen (Aufgabe 7.2)	215

Kommunikationswege in einem Rechnernetz (Beispiel 7.3)	216
Routing in einem Rechnernetz (Aufgabe 7.4)	223

Beispiele aus dem täglichen Leben

Was sind intelligente Systeme? (Aufgabe 1.1)	5
Verhalten eines Geldautomaten (Aufgabe 1.2)	25
Verifikation der Steuerung eines Geldautomaten (Aufgabe 6.17 mit Lösung)	205, 479
Dosieren einer Flüssigkeit (Aufgabe 4.2 mit Lösung)	99, 474
Stundenplanung (Aufgabe 4.14)	128
Fuzzyregelung eines aufrechtstehenden Pendels (Aufgabe 10.5)	353

Hinweise zum Gebrauch des Buches

Formelzeichen. Die Wahl der Formelzeichen hält sich an folgende Konventionen: Kleine Buchstaben bezeichnen Skalare und Funktionen, z. B. x, a, t , große Buchstaben Zufallsvariable oder Variable der Prädikatenlogik, z. B. X, Y . Mengen sind durch kalligrafische Buchstaben dargestellt: \mathcal{Q}, \mathcal{P} .

Programme. Programmausschnitte und die in den Programmen verwendeten Bezeichnungen sind in Schreibmaschinenschrift gesetzt. Für Anweisungen werden folgende Symbole verwendet. $S \leftarrow A$ bedeutet, dass der Wert der Variablen A der Variablen S zugewiesen wird. Für eine Menge \mathcal{M} und eine Variable X bedeutet die Relation $\mathcal{M} \Leftarrow X$, dass der Wert der Variablen X als neues Element in die Menge \mathcal{M} eingetragen wird, und die Relation $X \Leftarrow \mathcal{M}$, dass der Variablen X der Wert eines Elementes aus \mathcal{M} zugewiesen wird, wobei aus dem Text hervorgeht, um welches Element es sich handelt. Diese Anweisungen werden typischerweise für geordnete Mengen \mathcal{M} (Listen) angewendet.

Für die in diesem Buch behandelten Programmiersprachen LISP und PROLOG gibt es frei verfügbare Interpreter, die die Leser nutzen sollten, um die Konzepte beider Sprachen zu verstehen, u. a. *GNU Common LISP* (<ftp://ftp.gnu.org>), *CLISP* (clisp.cons.org) bzw. *GNU PROLOG* (www.gprolog.org).

Sprache. Sofern englische Bezeichnungen nicht als feststehende Fachbegriffe im Deutschen eingeführt sind, werden in diesem Lehrbuch deutsche Begriffe verwendet. Wichtige englische Fachbegriffe sind im deutsch-englischen Fachwörterverzeichnis (Anhang 5) aufgeführt.

Übungsaufgaben. Die angegebenen Übungsaufgaben sind ihrem Schwierigkeitsgrad entsprechend folgendermaßen gekennzeichnet:

- Aufgaben ohne Markierung dienen der Wiederholung und Festigung des unmittelbar zuvor vermittelten Stoffes. Sie können in direkter Analogie zu den behandelten Beispielen gelöst werden.
- Aufgaben, die mit einem Stern markiert sind, befassen sich mit der Anwendung des Lehrstoffes auf ein praxisnahes Beispiel. Für ihre Lösung werden vielfach außer dem unmittelbar zuvor erläuterten Stoff auch Ergebnisse und Methoden vorhergehender Kapitel genutzt. Die Leser sollen bei der Bearbeitung dieser Aufgaben zunächst den prinzipiellen Lösungsweg festlegen und erst danach die Lösungsschritte nacheinander ausführen. Die Lösungen dieser Aufgaben sind im Anhang 1 angegeben.

Weitere Informationen. Von der Homepage des Lehrstuhls für Automatisierungstechnik und Prozessinformatik der Ruhr-Universität Bochum können weitere Informationen sowie die Abbildungen dieses Buches in A4-Vergrößerung für die Verwendung in der Vorlesung bezogen werden:

`http://www.ruhr-uni-bochum.de/atp → Books`