

Rollenübernahme als Benutzermodellierungsmethode: Globale Antizipation in einem transmutierbaren Dialogsystem

Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
der Technischen Fakultät
der Universität des Saarlandes

vorgelegt von

Alassane Ndiaye

Saarbrücken

Januar 1998

Tag des Kolloquiums: 28. Januar 1998
Dekan: Prof. Dr.-Ing. Alexander Koch
Vorsitzender: Prof. Dr.-Ing. Gerhard Weikum
Berichterstatter: Prof. Dr. Wolfgang Wahlster
Prof. Dr. Karin Harbusch
(Universität Koblenz-Landau)
Akademischer Mitarbeiter: Dr. Anthony Jameson

Kurzzusammenfassung

In dieser Arbeit werden zwei Aspekte pragmatisch ausgerichteten Dialogverhaltens untersucht: Eine flexible Interaktion zwischen den Systemkomponenten und die Fähigkeit eines Dialogsystems, innerhalb einer gegebenen Dialogsituation mehr als eine der Dialogrollen zu übernehmen. Während das erste Ziel durch eine Multi-Agenten-Architektur unterstützt wird, bezieht sich das zweite Ziel auf die Fähigkeit, sich gedanklich in die Lage der am Dialog beteiligten Partner zu versetzen, um deren Reaktionen bzw. Intentionen zu antizipieren. Diese Strategie der *globalen Antizipationsrückkopplung* wurde in dem PRACMA-System realisiert, das in einem Verkaufsgespräch sowohl die Rolle des Verkäufers als auch die des Käufers übernehmen kann. Es wird gezeigt, wie die globale Antizipation zur Vorhersage des Käufers- bzw. Verkäufersverhaltens in PRACMA implementiert wurde. Es wird auch ein Ansatz präsentiert, der der Unsicherheit über das Dialogverhalten und die Präferenzen des Dialogpartners Rechnung trägt. Schließlich werden einige Überlegungen zur Effizienz bei der Verwendung der globalen Antizipationsrückkopplung in Abhängigkeit von den verfügbaren Systemressourcen erörtert.

Schlüsselwörter: Antizipationsrückkopplung, Rollenübernahme, Multi-Agenten-Architektur, Benutzermodellierung, natürlichsprachliche Verarbeitung, Transmutabilität.

Short Abstract

Two characteristics of pragmatically oriented dialog processing are investigated in this thesis: Flexible cooperation among the system's modules, which maximizes the system's exploitation of its knowledge and of its reasoning capabilities; and the ability of a system to take either (or any) of the dialog roles in its domain. While attainment of the first goal is supported by a multi-agent architecture, the second goal focuses on the ability of dialog participants to predict the responses of their dialog partners by hypothetically assuming the partner's role. This strategy, called *global anticipation feedback*, is investigated within PRACMA, a dialog system that is capable of taking either role (buyer or seller) in a sales talk. It is shown how global anticipation feedback can be used to anticipate either the seller's or the buyer's behavior. An extension of these techniques is discussed that addresses the limited predictability of a dialog partner's responses. Finally, several approaches to minimizing the computational cost of using global anticipation feedback according to the available system resources are addressed.

Keywords: Anticipation Feedback, Role Taking, Multi-Agent Architecture, User Modeling, Natural Language Processing, Transmutability.



Diese Arbeit entstand im Rahmen der von der Deutschen Forschungsgemeinschaft finanzierten Projekte PRACMA und READY. An dieser Stelle möchte ich mich bei allen bedanken, die zum Gelingen dieser Arbeit beigetragen haben.

Mein aufrichtiger Dank gilt vor allem Prof. Dr. Wolfgang Wahlster für die Aufnahme in seine Arbeitsgruppe, für das interessante Thema, für sein großes Interesse an der Arbeit und seine Hilfe bei allen kleinen und großen Problemen.

Sehr erfreut hat mich, daß Prof. Dr. Karin Harbusch sich trotz knapper Zeit spontan entschieden hat, das Koreferat dieser Arbeit zu übernehmen. Hierfür danke ich herzlich.

Ein großer Dank gebührt Dr. Anthony Jameson, bei dem ich immer ein offenes Ohr fand. Er hat diese Arbeit mit stetem Interesse begleitet und mir wichtige Denkanstöße und Hinweise gegeben.

Meinen Kollegen Ralph Schäfer und Thomas Weis danke ich für Ihre Unterstützung und Hilfsbereitschaft.

Ich bedanke mich auch bei Christopher Wirtz, der einen Teil der Implementation übernommen und durch zahlreiche Diskussionen zum Gelingen der Arbeit beigetragen hat. Ich danke auch Jan Amtrup, der die ICE-Software zur Interprozeßkommunikation entwickelte und Sylvia Bach für die Korrekturarbeit.

Ein herzliches Dankeschön für das angenehme Arbeitsklima geht an die Mitarbeiter am Lehrstuhl Wahlster, insbesondere an Doris Borchers.

Schließlich möchte ich mich bei allen bedanken, die hier nicht namentlich erwähnt wurden, von denen ich beim Verfassen dieser Arbeit Unterstützung erfahren habe.

Diese Arbeit möchte ich meiner verstorbenen Mutter Dieynaba Sy, meinem Onkel Amadou Ndiaye und meinem Vater Abou Alassane Ndiaye widmen, die mich stets unterstützt, gefördert und ermutigt haben.

Inhaltsverzeichnis

1	Motivation und Fragestellungen	1
1.1	Warum sollte ein System antizipieren?	1
1.2	Antizipation in interessenbasierten Dialogsystemen	3
1.3	Vier kanonische Fälle für globale Antizipation	5
1.4	Die zu untersuchenden Fragestellungen	6
1.5	Aufbau der Arbeit	7
2	Antizipation und Rückkopplung	9
2.1	Beiträge aus der Kommunikationstheorie	9
2.2	Beiträge aus der Psycholinguistik	11
2.3	Beiträge aus der Psychologie	12
2.3.1	Perspektivenübernahme	12
2.3.2	Beiträge aus dem Bereich “Theory of Mind”	17
2.3.3	Hörerorientierung	18
2.4	Simulation	19
2.5	Ansätze aus der Spiel- und Entscheidungstheorie	19
2.6	Systeme aus dem Bereich der Künstlichen Intelligenz	20
2.6.1	HAM-ANS: Ellipsengenerierung	22
2.6.2	IMP: Pragmatische Interpretation von Äußerungen	24
2.6.3	BILD: Bidirektionale Verarbeitung mit Hilfe reversibler Grammatiken	25
2.6.4	NAOS: Beschreibung von Straßenszenen	26
2.6.5	ANTLIMA: Visualisierung der Bildvorstellung des Hörers	27
2.6.6	ZORA: Zeige gestengenerierung	28
2.6.7	Antizipation in der Dialogübersetzung	29
2.6.8	WIP-GD: Antizipation bei der iterativen Graphikgenerierung	30
2.6.9	PEDRO: Vorhersage der Verständlichkeit einer Instruktion	31
2.6.10	Antizipation zur Verhinderung von falschen Inferenzen.	32
2.6.11	WISHFUL: Antizipation von Benutzerinferenzen	33
2.6.12	RMM: Koordination von Agenten-Handlungen	34
2.6.13	Zusammenfassung	35

3	Eine Architektur für Antizipationsrückkopplung	41
3.1	Architekturen von KI-Systemen	41
3.1.1	Die Basisarchitekturen in NL-Systemen	43
3.1.2	Die Blackboard-Architekturen	43
3.1.3	Die Aktoren-Architektur	45
3.1.4	Hybride Architekturen	46
3.2	CHANNELS: Eine Multi-Agenten-Architektur	46
3.2.1	Der Agenten-Begriff	46
3.2.2	Agenten und Nachrichten	47
3.2.3	Externe Agenten	53
3.2.4	Verschachtelung heterogener Architekturen	55
3.2.5	Nachrichtenaustausch vs. Blackboard	55
3.2.6	Mit CHANNELS verwandte Ansätze	56
3.2.7	Andere Anwendungen von CHANNELS	59
3.3	Die PRACMA-Agenten	60
3.3.1	Ein Überblick über das PRACMA-System	60
3.3.2	Der EVALUATION HANDLER	60
3.3.3	Der COMMENT AND QUESTION HANDLER	65
3.3.4	Der DIALOG PLANNER	67
3.3.5	Der EVALUATION EXPRESSION HANDLER	68
3.3.6	Der PRAGMATIC DIALOG MEMORY	68
3.3.7	Der DOMAIN BELIEF HANDLER	68
3.3.8	Der EGO	68
3.3.9	Der NATURAL LANGUAGE ANALYZER	69
3.3.10	Der NATURAL LANGUAGE GENERATOR	69
3.3.11	Der GLOBAL ANTICIPATOR	70
3.3.12	Der PRACMA SURFACE	70
3.4	Rollen-Transmutierbarkeit in PRACMA	70
3.4.1	Transmutierbarkeit mit Hilfe bidirektionaler Planoperatoren	71
3.5	Realisierung lokaler AFLs	72
3.6	Realisierung globaler AFLs	73
4	Antizipation der Käuferreaktion	77
4.1	Definition der Situation	77
4.2	Wie kann die Situation entstehen?	79
4.3	Angemessene Strategien	79
4.4	Realisierung innerhalb von PRACMA	80

4.5	Diskussion	82
4.5.1	Stärken und Schwächen dieses Ansatzes	82
4.5.2	Alternative Varianten dieses Ansatzes	83
5	Unsicherheitsbehandlung bei der globalen Antizipation	85
5.1	Definition der Situation	85
5.2	Wie kann die Situation entstehen?	86
5.3	Angemessene Strategien	86
5.4	Realisierung innerhalb von PRACMA	86
5.5	Diskussion	87
5.5.1	Stärken und Schwächen dieses Ansatzes	87
5.5.2	Alternative Varianten dieses Ansatzes: Der probabilistische Ansatz	88
6	Antizipation des Verkäuferverhaltens	91
6.1	Definition der Situation	91
6.2	Wie kann die Situation entstehen?	91
6.3	Angemessene Strategien	92
6.4	Realisierung innerhalb von PRACMA	92
6.5	Diskussion	93
6.5.1	Stärken und Schwächen dieses Ansatzes	93
7	Ressourcenadaptive Antizipation	95
7.1	Der Satisficing-Ansatz	95
7.1.1	Überblick über den Ansatz	95
7.1.2	Formalisierung	96
7.1.3	Realisierungsbeispiele	96
7.1.4	Diskussion	97
7.2	Der Iterationsansatz	98
7.2.1	Überblick über den Ansatz	98
7.2.2	Formalisierung	98
7.2.3	Realisierungsbeispiele	99
7.2.4	Diskussion	100
7.3	Der Ansatz der Antizipation im Nachhinein	100
7.3.1	Überblick über den Ansatz	100
7.3.2	Formalisierung	101
7.3.3	Realisierungsbeispiele	103
7.3.4	Diskussion	104
7.4	Der Ansatz der Antizipation nach Wert der Berechnung	106

7.4.1	Überblick über den Ansatz	106
7.4.2	Exkurs: Russell und Wefald's Ansatz zu "Value of Computation" .	106
7.4.3	Formalisierung	108
7.4.4	Realisierungsbeispiele	109
7.4.5	Diskussion	109
7.5	Der Ansatz der selektiven Antizipation	110
7.5.1	Überblick über den Ansatz	110
7.5.2	Formalisierung	110
7.5.3	Realisierungsbeispiel	111
7.5.4	Diskussion	111
7.6	Der Ansatz der selektiven Anpassung	112
7.6.1	Überblick über den Ansatz	112
7.6.2	Formalisierung	112
7.6.3	Realisierungsbeispiel	112
7.6.4	Diskussion	113
7.7	Der Ansatz der mehrstufigen Vorausschau	113
7.7.1	Überblick über den Ansatz	113
7.7.2	Formalisierung	113
7.7.3	Diskussion	115
7.8	Der Ansatz der Antizipation unter Annahme von Zeitdruck beim Dialogpartner	116
7.8.1	Überblick über den Ansatz	116
7.8.2	Formalisierung	116
7.8.3	Realisierungsbeispiele	116
7.8.4	Diskussion	117
7.9	Zusammenfassung	118
8	Zusammenfassung und Ausblick	121
8.1	Erzielte Ergebnisse	121
8.2	Erweiterungsmöglichkeiten	123
	Summary	125
	Literaturverzeichnis	133
	Autorenverzeichnis	145
	Sachverzeichnis	147

Abbildungsverzeichnis

1.1	Die Grundidee der Antizipationsrückkopplungsschleife.	2
1.2	Ein von PRACMA modelliertes Szenario.	4
1.3	Auszug aus einem Beispieldialog, in dem PRACMA sowohl die Rolle des Verkäufers als auch die des Käufers übernehmen kann.	5
2.1	Levelts Modell zur Selbstüberwachung bei der menschlichen Produktion gesprochener Sprache.	11
2.2	Beispiel eines Dialogs mit Generierung einer elliptischen Äußerung in HAM-ANS.	23
2.3	Auszug aus einem Beispieldialog, in dem IMP die Rolle des Immobilienmaklers übernimmt.	25
2.4	Beispiel einer Verzahnung von Parsing und Generierung.	26
2.5	Beispiele zur Verhinderung von falschen Inferenzen.	33
3.1	Die Blackboard-Architektur.	44
3.2	Die Architektur des PRACMA-Systems als Beispiel für die Kommunikation in CHANNELS.	47
3.3	Auszug aus der CHANNELS-Spezifikation in BNF.	48
3.4	Klassifikation von Sprechakten nach Vanderveken (1990).	50
3.5	Synchroner Nachrichtenaustausch.	51
3.6	Asynchroner Nachrichtenaustausch.	52
3.7	Beispiele von Nachrichtenaustausch zwischen den Modulen DIALOG PLANNER und COMMENT AND QUESTION HANDLER (CQH), dem Modul zur Äußerungsauswahl in PRACMA.	54
3.8	Übergänge der internen Zustände eines Objektes in ABCL.	57
3.9	Beispieldialog, in dem PRACMA sowohl die Rolle des Verkäufers (\mathcal{V}) als auch die des Käufers (\mathcal{K}) übernehmen kann.	61
3.10	Beispiel eines Interessenprofils.	62
3.11	Teil des Bewertungsbaumes zur Ermittlung der Gesamtbewertung eines Autos nach dem MAUT-Verfahren.	63
3.12	Auszüge eines Bayesschen Netzes, das PRACMA in der Rolle von \mathcal{V} verwendet, um bewertende Reaktionen zu Äußerungen des Verkäufers vorherzusagen bzw. zu interpretieren.	64

3.13	CQHs Algorithmus zur Auswahl eines Dialogzuges bei vorgegebenem Typ ohne Einsatz der globalen Antizipation.	65
3.14	Algorithmen zur Bewertung des unmittelbaren Nutzens eines Dialogzuges.	66
3.15	Auszug aus einem bidirektionalen Plan-Operator.	72
3.16	Die Architektur des PRACMA-Systems.	75
3.17	CQHs Algorithmus zur Auswahl eines Dialogzuges bei vorgegebenem Typ unter Verwendung der globalen Antizipation.	76
4.1	Entscheidungsbaum zur Illustration der Antizipation der Benutzerreaktion vor der Auswahl des nächsten Dialogzuges.	78
4.2	Gegenüberstellung des Interessenprofils in der Autodomäne von \mathcal{S} (links) mit dem Interessenprofil (rechts), das \mathcal{S} 's EVALUATION HANDLER dem jetzigen Käufer zuschreibt.	80
4.3	Algorithmen zur globalen Antizipation.	81
4.4	Ein Beispiel für die Antizipation der Käuferreaktion.	82
5.1	Verallgemeinerung des Entscheidungsbaums in Abbildung 4.1 (Seite 78) unter Berücksichtigung von Unsicherheit von \mathcal{S} über die Reaktionen von \mathcal{B}	85
5.2	Verallgemeinerung der Algorithmen zur globalen Antizipation (von Abbildung 4.3) durch Berücksichtigung der Unsicherheit des Verkäufers über die Käuferreaktion.	87
5.3	Ein Beispiel für die Antizipation der Käuferreaktion unter Berücksichtigung von Unsicherheit über die Inferenzen bzw. Bewertungskriterien des Käufers.	88
5.4	Entscheidungsbaum zur Illustration der Antizipation der Reaktionen des Dialogpartners unter Berücksichtigung der Wahrscheinlichkeiten, die den potentiellen Reaktionen des Dialogpartners beigemessen werden.	89
5.5	Algorithmus zur Ermittlung des Gesamtnutzens eines Dialogzugs nach dem probabilistischen Ansatz.	89
6.1	Ein Beispiel für die Antizipation der Verkäuferreaktion.	92
7.1	Algorithmus zur Auswahl eines geeigneten Dialogzuges nach dem Satisficing-Ansatz.	97
7.2	Beispiel zur Anwendung der runner-up-Strategie als Variante des Satisficing-Ansatzes.	98
7.3	Algorithmus zur Auswahl eines geeigneten Dialogzuges nach dem Iterationsansatz.	99
7.4	Zusammenstellung der Teilprozesse zur Äußerungsauswahl und -interpretation.	101
7.5	Darstellung des Entscheidungsprozesses von \mathcal{S} , wann antizipiert werden soll.	102
7.6	Die möglichen Fälle der Antizipation nach Wert der Berechnung.	108

7.7	Darstellung eines Entscheidungsbaumes der Entscheidungsstufe 2.	114
7.8	Darstellung des Übergangs von Stufe i zu $i + 1$	114
7.9	Stufenweise Analyse.	115
7.10	Algorithmus zur Bestimmung der potentiellen Dialogzüge in PRACMA unter Berücksichtigung von Zeitdruck.	117
7.11	Beispiel einer Antizipation unter Annahme von Zeitdruck beim Dialogpartner.	118
8.1	Illustration of the concept of global anticipation with decision trees.	126

Kapitel 1

Motivation und Übersicht der Fragestellungen

1.1 Warum sollte ein System antizipieren?

Bei der Realisierung intelligenter Benutzerschnittstellen in der Mensch-Maschine-Kommunikation spielt die Vorhersage der Reaktionen des Benutzers auf Ausgaben des Systems eine wesentliche Rolle, da intelligentes Dialogverhalten die Fähigkeit voraussetzt, das mögliche Verhalten des Benutzers vorhersagen zu können, um es von vornherein bei der Generierung zu berücksichtigen. Der wesentliche Vorteil einer solchen Vorgehensweise für ein System ist eine bessere Planung der eigenen Dialogbeiträge.

Im einfachsten Falle, beim Entwurf ergonomischer Menüschnittstellen bestimmt der Entwickler von vornherein die Menge der Reaktionsmöglichkeiten und präsentiert sie in Form von Menüs, in denen der Benutzer anklicken soll.

Intelligente Benutzerschnittstellen hingegen berücksichtigen explizit ein Benutzermodell – eine Wissensquelle, in der explizite Annahmen über den Benutzer repräsentiert sind, die für das Dialogverhalten des Systems relevant sind (Wahlster & Kobsa, 1989). So wird dem Benutzer durch Verbalisierung bzw. Visualisierung von Ergebnissen in Abhängigkeit von den Fähigkeiten des Benutzers und vom Dialogkontext geholfen. Hier besitzt das System die Fähigkeit, das Verhalten des Benutzers vorhersagen zu können. Ziel dieser Antizipation ist es beispielsweise, Verständnisprobleme bzw. redundante Informationen auszuschließen oder Fehlschlüsse seitens des Benutzers zu verhindern. Wenn der Benutzer eine Frage stellt, die auf falschen Annahmen basiert, soll das System ihn darauf aufmerksam machen. Darüber hinaus soll das System die eigenen Beiträge so generieren, daß der Benutzer nicht zu falschen Inferenzen verleitet wird. Joshi *et al.* (1984b) sprechen z.B. von “preventing false inferences”. Ein weiteres Anwendungsfeld betrifft die Generierung von eindeutigen anaphorischen Referenzen durch Simulation der Analyse der Äußerungen durch den Benutzer (Jameson & Wahlster, 1982). Hierbei antizipiert das System, ob eine vom ihm geplante elliptische Äußerung eindeutig vom Benutzer verstanden würde. In beiden Fällen versetzt sich das System in die Lage des Benutzers und simuliert dessen Verarbeitung, um diese von vornherein mit einzubeziehen (s. Abbildung 1.1).

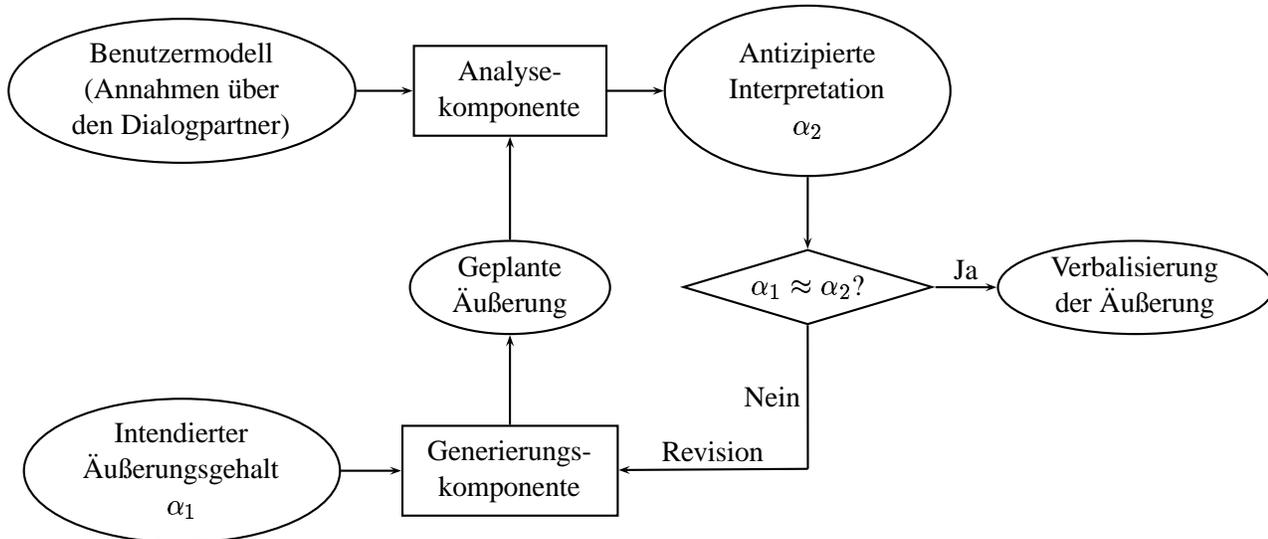


Abbildung 1.1: Die Grundidee der Antizipationsrückkopplungsschleife.

(Das System verwendet seine Analyse- bzw. Interpretationsfähigkeiten, um für einen von ihm geplanten kommunikativen Akt die Interpretation des Dialogpartners zu antizipieren.)

All diesen kooperativen Systemen ist der Versuch gemeinsam, die eigene Leistung und Äußerungen im alleinigen Dienste des Benutzers zu optimieren. Sie versuchen, die Ziele und das Wissen des Benutzers zu erkennen und zu berücksichtigen, um ihm die bestmögliche Antwort zu geben. Das Systemverhalten sollte, angepaßt an den Benutzer, möglichst objektiv und informativ sein.

Heutzutage werden aber zunehmend Systeme im Bereich des *electronic commerce* entworfen, die nicht ausschließlich kooperativ sind, sondern ein eigenes Interesse verfolgen. Man denke etwa an den seit der Internationalen Automobil-Ausstellung IAA 1995 bestehenden virtuellen Autosalon von Mercedes Benz im World Wide Web oder an dessen Nachfolger MBKS (Mercedes-Benz Kundenberatungs-System), in denen versucht wird, Information beim interessierten Kunden zu plazieren. Hier kommt der Antizipation eine große Rolle zu. Wenn das System das Profil des Benutzers kennt, wird es versuchen, das Objekt so zu präsentieren, daß ein positiver Eindruck beim Kunden entsteht und folglich dieser in seiner Kaufentscheidung beeinflusst wird.

Auch in der alltagssprachlichen Kommunikation spielt diese Idee eine Rolle. Man denke etwa an Redewendungen wie: "Er konnte in seiner Situation doch nicht anders handeln!"; "Versetzen Sie sich doch in meine Lage!"; "Was würden Sie denn an meiner Stelle tun?". Diese beispielhaften Redewendungen aus dem alltäglichen Leben illustrieren, wie wir uns die Lage anderer vergegenwärtigen, wie wir dazu aufgefordert werden, bzw. wie wir andere dazu auffordern, unsere Standpunkte nachzuvollziehen. In all diesen Fällen versetzt man sich virtuell in die Position einer anderen Person und übernimmt für einen Moment gedanklich die Perspektive bzw. die Rolle dieser Person, um deren Handlungen besser zu verstehen. Mit dieser Operation ist die Einsicht verbunden, daß wir dadurch Schlüsse darüber ziehen können, wie unser Gesprächspartner voraussichtlich handeln wird oder was ihn

dazu gebracht hat, so zu handeln, wie wir es beobachtet haben. Aus diesen Erkenntnissen erhoffen wir uns eine bessere Planung unseres eigenen Verhaltens und Handelns. Mit anderen Worten: Daß wir in unserer Interaktion mit anderen Personen erfolgreich sind, hängt nicht zuletzt von unserem Wissen darüber ab, wie diese Personen wahrscheinlich auf unser Verhalten und auf bestimmte Ereignisse antworten. Dabei orientieren wir uns nicht nur an ihrem offenkundigen Verhalten, sondern wir versuchen einerseits ihr geplantes Verhalten zu ergründen und andererseits uns ein Bild davon zu machen, welche ihre Absichten, Motive und Ziele sind. Diese Strategie wird als *Rollenübernahme* bzw. *Perspektivenübernahme* bezeichnet.

Mit dem Prozeß der Rollenübernahme ist einerseits *Antizipation* und andererseits *Rückkopplung* (*Feedback*) verbunden. Zum einen werden die Verarbeitungsprozesse des Gesprächspartners simuliert, um dessen künftige Handlungen zu antizipieren, und zum anderen werden diese gewonnenen Informationen in unsere eigene Planung vor unserer Handlung in einen Rückkopplungsprozeß mit einbezogen, um gegebenenfalls unsere Aktion *vor* deren Ausführung aufgrund der Antizipationsergebnisse anzupassen.

1.2 Antizipation in interessenbasierten Dialogsystemen

Sowohl in kooperativen als auch in konfliktären Situationen trägt die Antizipation in Form von Rollenübernahme dazu bei, latente Kooperationsmöglichkeiten zu entdecken und potentielle Konflikte zu vermeiden.¹ Insbesondere in nicht ausschließlich kooperativen Situationen — wie etwa in einem Verkaufsgespräch — bedarf es ein hohes Maß an Antizipation. Will man in einer solchen Situation erfolgreich und überzeugend sein, kommt es entscheidend darauf an, die Argumente und Reaktionen des Verhandlungspartners vorwegzunehmen.

In vielen alltäglichen Situationen verfolgen die Dialogpartner unterschiedliche Ziele und zeigen nicht ausschließlich kooperatives Verhalten. Beispiele einer solchen Situation sind Verkaufsverhandlungen, etwa beim Gebrauchtwagenhandel. In diesen Situationen kann es der Fall sein, daß die Dialogpartner Fakten verschweigen oder tendenziös entstellen, um das Verhandlungsobjekt so vorteilhaft wie möglich zu präsentieren. Hierzu kommt der Vorhersage der potentiellen Reaktionen des Dialogpartners während der Verhandlung eine entscheidende Rolle zu. Solche Interaktionen wurden innerhalb des Dialogsystems PRACMA (**P**rocessing **A**rguments between **C**ontroversially-**M**inded **A**ctors) (s. Jameson *et al.*, 1995; 1994) untersucht.²

PRACMA modelliert Verkaufsgespräche aus der Autodomäne, in denen eine Person \mathcal{V} versucht, ihren in der Zeitung annoncierten Gebrauchtwagen an einen potentiellen Käufer \mathcal{K} zu verkaufen, der sich telefonisch über den Wagen informieren will (s. Abbildung 1.2).

¹“Als kognitive Operation zur Erfassung der Handlungsorientierung anderer leistet Perspektivenübernahme darüber hinaus noch etwas, was keine andere Art der sozialen Wahrnehmung bieten kann, nämlich die Entdeckung und Analyse *latenter Kooperationsmöglichkeiten und Konflikte*.” (Geulen, 1982b, S. 66)

²Die Forschungen zu PRACMA wurden von der Deutschen Forschungsgemeinschaft im Rahmen des Sonderforschungsbereichs 314 “Künstliche Intelligenz — Wissensbasierte Systeme” gefördert.

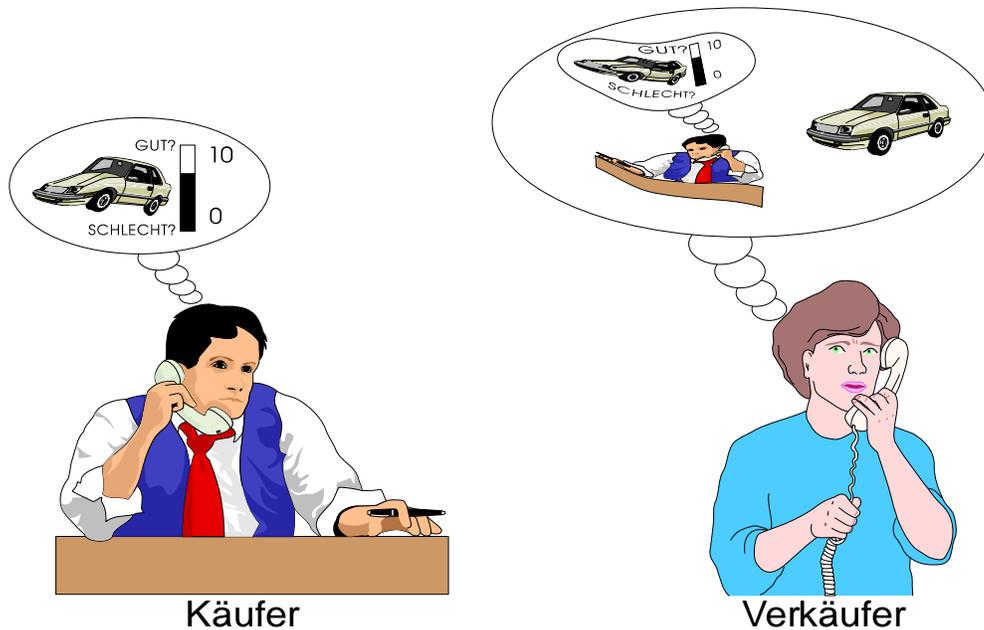


Abbildung 1.2: Ein von PRACMA modelliertes Szenario.

(Eine Person \mathcal{V} informiert telefonisch einen potentiellen Käufer \mathcal{K} über ihren annoncierten Gebrauchtwagen.)

Die Motivationen der beiden Akteure in dieser Beispieldomäne sind teilweise gegensätzlich: \mathcal{K} möchte dem Verkäufer möglichst detaillierte und korrekte Informationen entlocken, auf die er seine Kaufentscheidung gründen kann; dagegen möchte \mathcal{V} das Auto zu einem möglichst hohen Preis verkaufen, unabhängig davon, ob es für \mathcal{K} geeignet ist oder nicht. PRACMA ist in dem Sinne rollentransmutierbar, daß es in dieser Dialogsituation sowohl die Rolle des Autobesitzers oder auch die des potentiellen Käufers übernehmen kann. Abbildung 1.3 gibt einen Auszug aus einem Beispieldialog mit dem PRACMA-System wieder.³ Wie später gezeigt wird, kann insbesondere in teilweise nicht-kooperativen Dialogen wie in der PRACMA-Domäne Antizipation eine wesentliche Rolle spielen. Hierbei wird zwischen *lokaler* und *globaler* Antizipation unterschieden. Bei lokaler Antizipation wird lediglich vorausgesetzt, daß das System einige Aspekte der Verarbeitung des Dialogpartners — wie z.B. die syntaktische Analyse — simulieren kann. So können spezifische Auswirkungen einer Äußerung auf den Dialogpartner vorhergesagt werden. Geht es aber darum, den nächsten Dialogschritt des Partners vorherzusagen (z.B., ob er schweigt, eine Bewertung abgibt oder eine weitere Frage stellt), muß das System manchmal die gesamte Verarbeitung seiner geplanten Äußerung durch den Dialogpartner simulieren. Wir sprechen dann von globaler Antizipation.

³Im Kapitel 3 wird ein ausführlicherer Einblick in das PRACMA-System und in die dort eingesetzten Techniken gegeben.

- | |
|--|
| <ul style="list-style-type: none"> ▪ \mathcal{V}: Wohnen Sie in der Stadt oder auf dem Land? ▪ \mathcal{K}: Ich wohne auf dem Land. ▪ \mathcal{V}: Wie interessiert sind Sie an Komfort? ▪ \mathcal{K}: Ich bin ziemlich interessiert an Komfort. ▪ \mathcal{V}: Dieses Auto könnte etwas für Sie sein. ▪ \mathcal{V}: Wie in der Annonce beschrieben, handelt es sich um einen Golf, Baujahr 1991. ▪ \mathcal{V}: Er hat vier Türen. ▪ \mathcal{K}: Das ist gut. ▪ \mathcal{V}: Das Auto hat Automatikgetriebe. ▪ \mathcal{K}: [—] (Schweigen) |
|--|

Abbildung 1.3: Auszug aus einem Beispieldialog, in dem PRACMA sowohl die Rolle des Verkäufers als auch die des Käufers übernehmen kann.

1.3 Vier kanonische Fälle für globale Antizipation

Hier soll anhand von vier Fällen gezeigt werden, wie in einem Verkaufsgespräch die globale Antizipation verwendet werden kann. In den beiden ersten Fällen übernimmt das System die Rolle des Verkäufers, während in den zwei letzten Beispielfällen das System die Rolle des Käufers simuliert.

1. Das System als Verkäufer: Antizipation der nächsten Frage des Käufers nach einer freiwilligen Anmerkung des Systems.
Beispiel: Bevor das System unaufgefordert eine Anmerkung generiert, antizipiert es, zu welcher möglichen Frage des Käufers, als Reaktion zu dieser beabsichtigten Äußerung, dies führen kann. So kann eine Anmerkung über den neuen Zustand der Reifen eine Frage nach dem Kilometer-Stand des Autos zur Folge haben. Falls der Kilometer-Stand einen Schwachpunkt des Autos darstellt, wird der Verkäufer die freiwillige Äußerung über den Zustand der Reifen vermeiden oder vorsichtiger angehen.
2. Das System als Verkäufer: Antizipation der Käuferreaktion nach einer Frage vom System.
Beispiel: Neben direkten Antworten und freiwilligen Anmerkungen kann das System in der Rolle des Verkäufers durch direkte Fragen mit dem potentiellen Käufer interagieren. Es ist einem Verkäufer wichtig zu wissen, welche Reaktionen durch eine von ihm gestellte Frage beim Dialogpartner hervorgerufen werden. Beispiel einer solchen Überlegung ist: “Wird \mathcal{K} jetzt ungeduldig werden, wenn ich noch eine Frage zu seiner Person stelle?”
3. Das System als Käufer: Antizipation der freiwilligen Anmerkungen des Verkäufers aufgrund einer vom System gestellten Frage.

Beispiel: Das System kann sich erhoffen, daß \mathcal{V} nach der Beantwortung einer Frage darüber, ob das Auto eine Klimaanlage hat, etwas mehr über die Umweltfreundlichkeit des Autos sagen wird. Hier verzichtet das System absichtlich auf eine direkte Frage an den Käufer über den Aspekt der Umweltfreundlichkeit. Der Grund dafür ist, daß direkte Fragen dem Käufer starke Rückschlüsse über die eigentlichen Interessen des Systems ermöglichen und so die wahre Einstellung des Systems verraten würden.

4. Das System als Käufer: Antizipation der beim Verkäufer hervorgerufenen Auswirkungen einer vom System gemachten verbalen Reaktion.

Beispiel: Nach der Erwähnung eines besonders positiven Merkmals des Autos kann sich das System als \mathcal{K} überlegen, ob es mit einer neutralen oder eher negativen verbalen Reaktion antwortet. Grund einer solchen Überlegung ist: Zum einen kann der Käufer das Auto schlechter machen als es ist, um einen für ihn günstigeren Preis zu erreichen; zum anderen kann \mathcal{K} sich erhoffen, daß \mathcal{V} mehr über den betroffenen Aspekt des Autos sagt.

In beiden Fällen, in denen das System als Käufer fungiert, soll auch berücksichtigt werden, ob die gezeigten Reaktionen des Systems seiner tatsächlichen oder nur einer vorgespiegelten Einstellung entspricht.

1.4 Die zu untersuchenden Fragestellungen

Anhand folgender Fragestellungen soll untersucht werden, welches die potentiellen Vorteile der globalen Antizipation sind und welches die Hauptprobleme und -beschränkungen sind.

1. Rollen-Transmutierbarkeit

Ein System, das globale Antizipation verwenden soll, muß die Rolle des Gesprächspartners innerhalb eines Dialoges übernehmen können. Diese Fähigkeit, die Gesprächsrolle innerhalb eines Dialoges zu wechseln, ist eine Variante der Eigenschaft von Dialogsystemen, die Wahlster und Kobsa (1989, S. 30) als *Transmutierbarkeit* definieren.⁴ Dort werden transmutierbare Dialogsysteme wie folgt definiert: “dialog systems adaptable to applications that differ not only with respect to the domain of discourse, but also to dialog type, user type and intended system behavior”. (Bei Menschen wird Transmutierbarkeit durch die Übernahme verschiedener Rollen in alltäglichen Situationen erlernt. Beispielsweise hat sogar ein professioneller Verkäufer oft die Gelegenheit, selbst als Kunde aufzutreten.)

2. Architektur

Die Realisierung der globalen Antizipation erfordert die Simulation der Rolle des Dia-

⁴Man beachte die Unterscheidung zwischen transmutierbaren und transportablen Systemen: Während *transportable* Systeme die Adaption an neue Diskursbereiche bzw. Domänen ermöglichen, sind *transmutierbare* Systeme solche, die transportabel und an verschiedene Kommunikationssituationen (z.B. Verkäufer/Kunde, Lernender/Experte) anpaßbar sind.

logpartners. Hierzu ruft sich das PRACMA-System in einer weiteren kompletten Instantiierung des Systems rekursiv auf. Diese Unterinstantiierung des Systems wird mit der anderen Dialogrolle (Käufer oder Verkäufer) initialisiert. Die Unterinstantiierung ist nicht an der Interaktion mit dem Benutzer unmittelbar beteiligt. Sie erhält als Eingabe eine potentielle Äußerung des Systems, simuliert die Verarbeitung des Dialogpartners und liefert dessen potentielle Reaktionen an die Hauptinstantiierung zurück. Die Verwaltung beider System-Instantiierungen, ohne daß die jeweiligen Verarbeitungsprozesse beeinträchtigt werden, erfordert eine flexible Systemarchitektur.

3. *Unsicherheit über das Dialogverhalten des Dialogpartners*

Das System verfügt bei der Simulation der Rolle des Gesprächspartners nicht über ein präzises Modell über dessen tatsächliche Bewertungskriterien, Motivationen und Dialogverhalten. Im Falle globaler Antizipation wird diese Unsicherheit durch mehrere Faktoren beeinflusst, die dem System nicht genau bekannt sind. Das System weiß dann nicht exakt, wie es die Rolle des Dialogpartners aufgrund seiner Unsicherheit simulieren kann. Dies wirft die Frage der Unsicherheitsbehandlung auf, als Repräsentation und Inferenz über unsicheres Wissen.

4. *Ressourcenadaptive Antizipation*

Es ist für ein Dialogsystem relativ (zeit)aufwendig, bei jedem Dialogzug die gesamten Verarbeitungsprozesse des Benutzers zu antizipieren. Daher muß untersucht werden, wie die Kosten der globalen Antizipation gesenkt werden können bzw. wie weit und wie intensiv in Abhängigkeit von den verfügbaren Ressourcen das System vorausschauen kann und soll.

1.5 Aufbau der Arbeit

Zur Eingliederung dieser Arbeit in die Forschungslandschaft wird zunächst im Kapitel 2 eine interdisziplinäre Einordnung gegeben. Danach wird im dritten Kapitel auf einige Überlegungen zu Fragen der Systemarchitektur eingegangen, bevor das PRACMA-System und dessen Multi-Agenten-Architektur und zugrundeliegende Formalismen vorgestellt werden.

In den Kapiteln 4 bis 6 werden die Formalisierungen und Implementierungen der Ansätze zur Antizipation des Käufer- bzw. des Verkäuferverhaltens vorgestellt. Dabei wird in Kapitel 5 gezeigt, wie der Unsicherheit über das Verhalten des Käufers Rechnung getragen werden kann.

Anschließend wird im Kapitel 7 gezeigt, wie der mit der vorgestellten Technik der globalen Antizipation einhergehende Aufwand abhängig von den verfügbaren Ressourcen gemacht werden kann.

Abschließend werden im letzten Kapitel die vorgestellten Ansätze zusammengefaßt und die Grenzen und Erweiterungsmöglichkeiten der globalen Antizipation aufgezeigt.

Kapitel 2

Antizipation und Rückkopplung: Eine interdisziplinäre Einordnung

Grundlegend für die in dieser Arbeit behandelte Thematik sind die Begriffe “Antizipation” und “Rückkopplung” (Feedback). Rückkopplung ist nach DiStefano *et al.* (1976) einer der fundamentalsten Naturvorgänge und tritt in fast allen dynamischen Systemen auf, einschließlich in denen des menschlichen Körpers, in den zwischenmenschlichen Beziehungen und im Zusammenwirken von Mensch und Maschine.¹ In der zwischenmenschlichen Kommunikation führt Feedback zur qualitativen Verbesserung der kommunikativen Mitteilung (Flavell *et al.*, 1968).

Antizipation ist nach Piaget (1974) neben Rückkopplung eines der allgemeinsten Merkmale der kognitiven Funktionen. Eine der essentiellen Funktionen des Wissens — nach seiner Auffassung — ist, daß es zur Vorhersage führt, also zur Antizipationsfähigkeit. Antizipation ist wegen des Vorwegnahmecharakters *per se* Teil des menschlichen Denkens.

Antizipation und Rückkopplung spielen sowohl in der Kommunikationstheorie (in der Kybernetik), in der Psychologie, in der Spiel- und Entscheidungstheorie als auch in der Benutzermodellierung eine Rolle. In diesem Kapitel wird diese Rolle in einer interdisziplinären Einordnung aufgezeigt.

2.1 Beiträge aus der Kommunikationstheorie

Every communicator carries around with him an image of the receiver. He takes his receiver (as he pictures him to be) into account when he produces a message. He antici-

¹Die Begriffe Antizipation und Rückkopplung haben ihre Tradition in der Kybernetik, der Wissenschaft der Regelung, Steuerung und Nachrichtenübertragungstheorie. So dient nach de Latil (1953) die Antizipation der *groben Regulierung* und die Rückkopplung der *feinen Regulierung* in einem kybernetischen System. Beispiele solcher Systeme sind z.B ein Autopilot oder ein Angebot-Nachfrage-System, dessen Aufgabe die Erhaltung der Preisstabilität ist. In einem solchen System wird, analog zu den in dieser Arbeit verwendeten Antizipationsrückkopplungsschleifen, der Ausgang (oder eine andere zu regelnde Variable des Systems) mit dem Eingang des Systems verglichen (oder mit dem Eingang einer internen Komponente oder eines Teilsystems), so daß der zugehörige Regelvorgang als eine Funktion des Ausgangs und Eingangs entsteht (DiStefano *et al.*, 1976).

pates the possible responses of this receiver and tries to predict them ahead of time. These images affect his own message behaviors. (Berlo, 1960, S. 117)

Penser à la pensée d'autrui est une caractéristique essentielle de toute attitude sociale; chacun cherche à suivre et à devancer le progrès de la pensée de l'autre, l'avantage étant à celui qui devine une pensée de l'autre que celui-ci croit ignorée.² (Guillaume, 1954, S. 182)

Diese beiden Zitate verdeutlichen, daß der Idee der Antizipation im Bereich der Kommunikationstheorie und der sozialen Kognition eine wesentliche Rolle zukommt.³ Hierbei kann das aus der Informationstheorie stammende E-R-Schema von Shannon und Weaver (1976) als kanonisches Grundmodell dienen. Im E-R-Modell sendet ein Sender einem Empfänger ein Signal, dessen Übertragung von der Encodierung bzw. Decodierung abhängt, die auf der Sender- bzw. auf der Empfänger-Seite stattfinden, sowie von dem Rauschen des Übertragungskanal. Dann erfolgt gegebenenfalls Feedback als Regulierung.

Dieses E-R-Modell reicht jedoch nicht aus, um alle Phänomene der Kommunikationstheorie abzudecken. In einem E-R-Schema erfolgt eine Anpassung der Emission der Nachricht nach einer vom Rezipienten gesendeten Rückmeldung (Feedback, Rückkopplung). Dies ist eine reaktive Adaption (Mucchielli, 1991). Rückkopplung ("Feedback") ist die vom Interaktionspartner gegebene Rückmeldung. Dieses Konzept ist dennoch eng verzahnt mit dem Begriff der Antizipation, da Rückkopplung sich auch auf die *erwartete* Reaktion des Interaktionspartners beziehen kann:

Feedback refers to the information that the subject receives from the other participants [...], himself, or from his own expectations of responses that he might receive. (Silverstein, 1969, S. 17)

Dies verdeutlicht, daß der menschliche Akteur keine Rückmeldung (d.h. Reaktion zu seiner Aktion) braucht, um die Reaktion seines Gesprächspartners auf das, was er sagen wird oder dabei ist zu sagen, zu antizipieren. Da er weiß, daß der Gesprächspartner auch diese Fähigkeiten hat, kann er sogar dessen Antizipation antizipieren. Durch diese Antizipationsfähigkeit kann der Mensch stets die Wirkung seines Diskurses abwägen und ihn *a priori* adaptieren. Mucchielli (1991, S. 16) spricht von einer permanenten Antizipation der Akteure und Berlo (1960, S. 116) weiter: "All human communication involves predictions by the source and receiver about how other people will response to a message".

²Frei übersetzt: "Über die Gedanken des anderen nachzudenken, ist eine wesentliche Eigenschaft jedes sozialen Verhaltens; jeder versucht, dem Fortschreiten des Denkens des anderen zu folgen und es vorwegzunehmen. Der Vorteil liegt bei demjenigen, der einen Gedanken des anderen voraussagt, den dieser verborgen glaubte."

³Für eine ausführliche Übersicht über Theorien zur Kommunikation sei auf Littlejohn, 1991 und für eine Einführung in soziale Kognition auf Roloff & Berger, 1982 verwiesen.

2.2 Beiträge aus der Psycholinguistik

Antizipation und Rückkopplung spielen bei der Produktion spontaner gesprochener Sprache eine wesentliche Rolle, da der Mensch seine Sprachverstehenskomponente zur Überwachung der eigenen Sprachproduktion einsetzt (Levelt, 1989). Ein wesentlicher Grund hierfür ist folgende Beobachtung: Ein Sprecher kann sich selbst hören, während er spricht und er kann sich sozusagen auch hören, bevor er etwas geäußert hat.

Levelt (1989) schlägt ein psycholinguistisches Modell menschlicher Sprachproduktion vor, das sich auf die Produktion spontaner Sprache beschränkt (s. Abbildung 2.1). Das Modell besteht aus drei Komponenten: dem *Konzeptualisierer*, dem *Formulator* und dem *Artikulator*. Der Konzeptualisierer legt aufgrund der kommunikativen Intention des Sprechers den Inhalt der Äußerung fest. Ergebnis des Prozesses ist eine konzeptuelle Struktur (*preverbal message*), die an den Formulator weitergeleitet wird. Der Formulator übersetzt die konzeptuelle Struktur in eine linguistische Struktur in zwei Schritten. Bei der grammatischen Kodierung wird durch die Wortwahl und den Aufbau syntaktischer Strukturen die konzeptuelle Struktur in eine Oberflächenstruktur übersetzt. Diese wird dann während der phonologischen Kodierung durch Auswahl lexikalischer Formen und Prosodieplanung in einen Artikulationsplan überführt, der dem Artikulator übergeben wird. Dieser Plan wird auch *interne Sprache* (*internal speech*) genannt. Der Artikulator führt den Artikulationsplan aus und steuert die Atmung und die Sprechwerkzeuge.

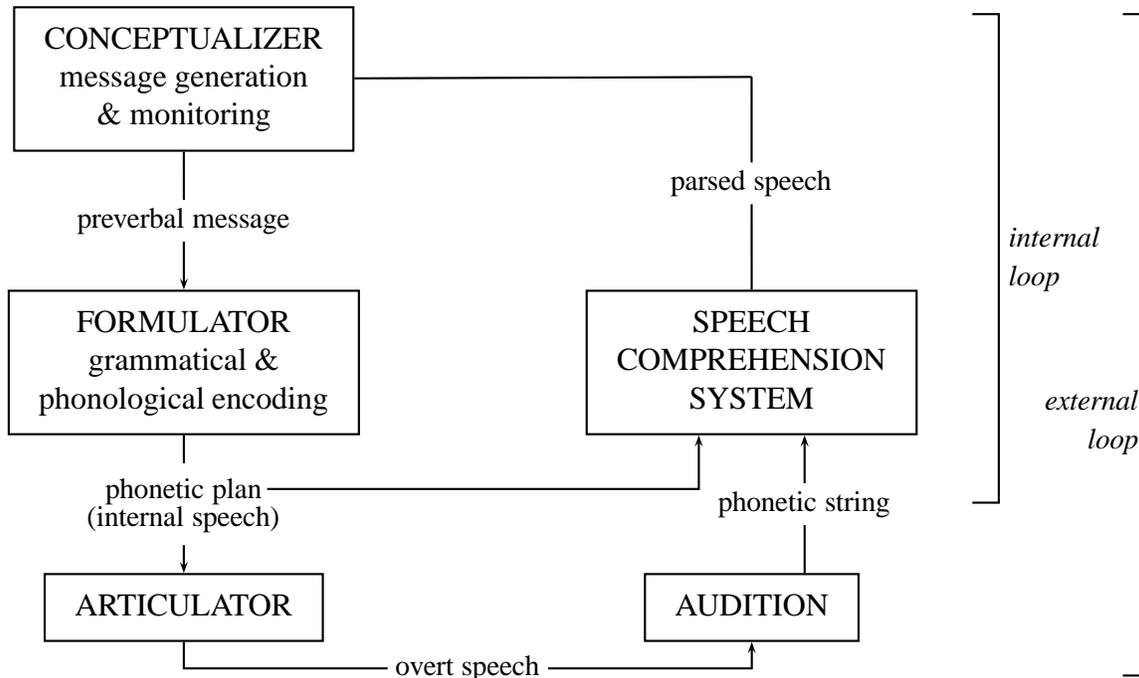


Abbildung 2.1: Levelts Modell zur Selbstüberwachung bei der menschlichen Produktion gesprochener Sprache.

Wie bereits erwähnt hat der Mensch die Eigenschaft, einerseits sich selbst zuzuhören, während er spricht, genauso wie er anderen zuhören kann; und andererseits kann er seiner internen Sprache "zuhören", d.h. bevor er etwas verbalisiert hat. Der Prozeß der Kontrolle der eigenen Sprachproduktion bedient sich einer doppelten Rückkopplungsschleife (*double perceptual-loop*, s. Abbildung 2.1): Eine interne Rückkopplungsschleife für die interne Sprache (antizipatorische Korrektur) und eine externe Rückkopplungsschleife für die Korrekturen von bereits Gesagtem, um auf die Reaktionen des Zuhörers etwa zu reagieren, falls die Äußerung des Sprechers mißverstanden wurde. Beide Schleifen dienen der Selbstüberwachung (*Self-monitoring*) und es wird die gleiche Sprachverstehenskomponente eingesetzt, die auch zur Analyse der Äußerung anderer verwendet wird.⁴ So kann der Sprecher bei der Analyse der internen Sprache die potentiellen Probleme antizipieren und sie beheben, bevor die Äußerung tatsächlich artikuliert wird. Durch Selbstüberwachung kann er auch die bereits artikulierten Äußerung mit der intendierten vergleichen und gegebenenfalls korrigieren. Nooteboom (1980) spricht in diesem Zusammenhang von *unspeaking* als Rücknahme von bereits Gesagtem.

2.3 Beiträge aus der Psychologie

Intelligentes Dialogverhalten in der zwischenmenschlichen Kommunikation setzt die Fähigkeit voraus, das Verhalten des Dialogpartners vorherzusagen. In der Denk- und Entwicklungspsychologie befassen sich verschiedene Ansätze mit dieser kognitiven Fähigkeit unter dem Begriff *Perspektivenübernahme* (Flavell *et al.*, 1968; Geulen, 1982a) und in den neueren Untersuchungen teilweise unter dem Begriff *Theory of mind* (Astington, 1993; Whiten, 1991).

In den 60er und 70er Jahren befaßten sich viele Forscher mit dem Themenkomplex Perspektivenübernahme, aufbauend auf die früheren Arbeiten Jean Piagets (s. Piaget, 1978 und Piaget & Inhelder, 1971) über die Psychologie und Intelligenz des Kindes sowie auf die Arbeiten George H. Meads (1968), der die kognitive Fähigkeit zur Antizipation der Reaktionen des anderen auf das eigene Verhalten und die Einbeziehung der antizipierten Reaktionen in die Planung des eigenen Verhaltens als den Kern sozialer Intelligenz sieht.

2.3.1 Perspektivenübernahme

Perspektivenübernahme ist nach Silbereisen (1987, S.706) ein hypothetischer Verstehensprozeß, bei dem versucht wird, sich die Situationsgebundenheit (also die *Perspektive*) des Handelns anderer Person zu vergegenwärtigen. Das Konzept der Perspektivenübernahme setzt eine Dezentrierung der eigenen Perspektive voraus. Diese mangelnde Dezentrierung entspricht dem bei Kindern beobachtbaren Phänomen egozentrischen Verhaltens (Piaget, 1978). Nach Flavell *et al.* (1968, S.16–17, 54) besteht kognitiver Egozentrismus in der Unfähigkeit, die Perspektive bzw. die Rolle des Gesprächspartners zu

⁴Vgl. die Ausführungen im Abschnitt 2.6 sowie die dort vorgestellten Systeme HAM-ANS bzw. BILD.

übernehmen sowie in der Unfähigkeit, die Rollenmerkmale eines anderen herauszufinden, sie mit den eigenen zu vergleichen und das Ergebnis des Vergleichs zu verschiedenen Anpassungsleistungen zu benutzen. Das egozentrische Kind differenziert nicht, sondern verwechselt seine eigene Perspektive mit der anderer. So redet es z.B. ohne zu versuchen den Hörer zu erreichen oder, wenn das Kind es dennoch versucht, dann mit Begriffen, die nicht geeignet sind. Piaget sieht die kognitive Entwicklung als zunehmende Dezentrierung der eigenen Perspektive.

Die Übernahme einer Fremdperspektive umfaßt folgende Unterteilung (Steins & Wicklund, 1993; s. auch Davis, 1994, S.46–51): *visuell-räumliche* Perspektivenübernahme, *affektive* bzw. emotionale Perspektivenübernahme und *konzeptuelle* Perspektivenübernahme.

Allen drei Ansätzen gemeinsam ist der Versuch über die eigene Sichtweise hinauszugehen und die Perspektive des anderen wahrzunehmen. Sie unterscheiden sich jedoch in dem jeweiligen Inhalt des Versuchs, sich die Rolle des anderen zu vergegenwärtigen. In visuell-räumlicher Perspektivenübernahme geht es um die Vorstellung, wie ein Objekt einer anderen Person erscheint, die sich an einer unterschiedlichen Position befindet; in affektiver Perspektivenübernahme werden die emotionalen Reaktionen einer anderen Person vorhergesagt; und in kognitiver Perspektivenübernahme werden Schlußfolgerungen aus den Gedanken, Motiven, Intentionen und Handlungen des anderen gezogen, um dessen Verhaltensweise zu antizipieren und in die eigene Verhaltensstrategie einzuplanen.

2.3.1.1 Visuell-räumliche Perspektivenübernahme

Dieser Aspekt der Perspektivenübernahme — auch perzeptuelle Perspektivenübernahme genannt — bezieht sich auf die Wahrnehmung räumlich und visuell erfaßbarer Merkmale, also eher auf die physikalischen. Die zugrundeliegende Frage hierbei ist: “Was sieht die andere Person von ihrer Position aus?” Der prominenteste Repräsentant hierbei ist der Drei-Berge-Versuch (Piaget & Inhelder, 1971) als Operationalisierung der visuell-räumlichen Perspektivenübernahme.

Beim Drei-Berge-Versuch werden vierjährigen Kindern ein 3D-Pappmodell mit drei Bergen vorgelegt, das drei deutlich unterschiedliche Perspektiven ergibt. Die Testpersonen sollten einschätzen, wie die Berge aus der Sicht eines Betrachters (einer an einem dieser Berge platzierten Puppe) aussehen. Die Mehrzahl der Kinder schrieben dem Betrachter die eigene Ansicht zu. Piaget und Inhelder (1971) sahen in dieser Beobachtung den Beweis des Egozentrismus von Kleinkindern in diesem Alter. Die Kinder waren nicht in der Lage zu erkennen, daß die eigene Perspektive nur eine unter vielen ist.

In Bezug auf visuell-räumliche Perspektivenübernahme weisen Herrmann und Grabowski (1994) auf einen interessanten Aspekt der Objektlokalisierung hin, bei dem der Gesprächspartner dazu aufgefordert wird, sich in die Lage des Sprechers hineinzusetzen. So sorgt der Sprecher dafür, daß er nicht die partnerbezogene Lokalisation verwendet, sondern daß er sprecherbezogen, also von seinem eigenen Blickpunkt aus, lokalisiert. Dies führt zu Beschreibungen der Art: “Von mir aus liegt der Ball links vom Stuhl”. Zu der Frage, wie nun lokalisiert werden sollte, machen es Herrmann und Grabowski (1994, S. 124) von den *mentalenen Kosten* der Koorientierung abhängig, die von demjenigen zu tragen sind, der die

eigene egozentrische Orientierung verläßt und die Perspektive des anderen übernimmt:

Der Partner ist im allgemeinen in der Lage, auch anhand von *sprecherbezogenen* Zwei- oder Dreipunktlokalisationen ein Objekt O_i zu identifizieren, doch entstehen ihm dabei zusätzliche mentale *Kosten*; denn er muß sich aus seiner eigenen egozentrischen Ausgangsorientierung herausbewegen und sich ‚in den Sprecher hineinversetzen‘. In diesem Fall hat es der Sprecher leicht, er muß seine eigene Orientierung nicht verlassen. Andererseits ist der *Sprecher* auch in der Lage, eine *partnerbezogene* Zwei- oder Dreipunktlokalisation zu produzieren, bei der nunmehr ihm die zusätzlichen Kosten entstehen; denn jetzt ist er es, der seine egozentrische Orientierung verlassen und sich zum Zwecke der Produktion einer partnerbezogenen Lokalisation ‚in den Hörer hineinversetzen‘ muß. Jetzt hat es wiederum der Hörer leicht: Er kann seine egozentrische Orientierung beibehalten. Die Angelegenheit gleicht einem Nullsummenspiel: Die Kosten, die der eine spart, muß der andere aufbringen.

2.3.1.2 Affektive Perspektivenübernahme

Diese Art der Perspektivenübernahme bezieht sich auf das Verstehen der emotionalen Zustände und Gefühle anderer Personen und wird *Empathie*⁵ genannt (Davis, 1994). Empathie ist die affektive Identifizierung mit einer Person oder einer Sache und setzt damit Einfühlungsvermögen voraus. Mit anderen Worten, es geht bei der Empathie um die Fähigkeit eines Individuums durch eine Art „affektive Ansteckung“ zu wissen bzw. nachzuempfinden, was ein anderer denkt, fühlt, wünscht, etc., auch wenn dessen Emotionen, Gefühle, Motivationen sich von den eigenen unterscheiden.

Empathie enthält sowohl eine kognitive Komponente zum Erkennen und Verstehen der Gefühle, Gedanken und Perspektiven anderer Personen als auch eine emotionale Komponente, die als affektive Reaktion auf die emotionale Lage eines anderen verstanden wird. Die zwei Aspekte unterscheiden sich darin, daß im ersteren keine Betroffenheit des Agenten vorausgesetzt wird, sondern lediglich seine Fähigkeit, sich die Lage des anderen vorzustellen (Friedlmeier, 1993).

2.3.1.3 Konzeptuelle Perspektivenübernahme

Die Idee der konzeptuellen Perspektivenübernahme wird in der entwicklungspsychologischen Literatur unterschiedlich bezeichnet: Rollenübernahme (Flavell *et al.*, 1968; Higgins, 1981), informationsbezogene Perspektivenübernahme (Geulen, 1982a), soziale Perspektivenübernahme (Selman, 1980), kognitive Perspektivenübernahme (Kurdek, 1977) und sozial-kognitive Perspektivenübernahme (Underwood & Moore, 1982). (In dieser Arbeit orientieren wir uns an dem Begriff *Rollenübernahme*.)

⁵Es gibt keine einheitliche Definition der Empathie in der entwicklungspsychologischen Literatur. In ihrer primitivsten Form ist Empathie nach Oléron *et al.* (1981, S. 72) die *affektive Ansteckung* („Gefühlsansteckung“), die dazu führt, daß beispielsweise Neugeborene weinen, weil sie andere weinen sehen oder ältere Kinder und Erwachsene lachen oder Angst empfinden, weil andere lachen oder Angst haben.

Definition. Rollenübernahme⁶ bezieht sich nach Flavell *et al.* (1968) einerseits auf die allgemeine Fähigkeit, die Rolle einer anderen Person im kognitiven Sinne einzunehmen, d.h. ihre Verhaltensweisen und -tendenzen in einer gegebenen Situation einzuschätzen; und andererseits auf die spezifische Fähigkeit, dieses Verständnis der Rolle einer anderen Person als Mittel zu nutzen, um effektiv mit ihr zu kommunizieren. Silbereisen (1975) definiert Rollenübernahme als antizipatorische soziale Kognition. Der wesentliche Vorteil dieses gedanklichen Versetzens in die Lage des anderen ist eine bessere Planung des eigenen Verhaltens. Shantz (1975) definiert Rollenübernahme als eine Menge kognitiver Prozesse, mit deren Hilfe eine Person das Verhalten einer anderen Person verstehen kann. Diese Konzeption setzt die Fähigkeit voraus, sich mental an die Stelle des anderen zu versetzen (“putting oneself in another person's shoes”) und dadurch dessen Denkweise zu inferieren: “To arrive at what another individual sees or believes, one puts oneself mentally in his position [...] and estimates what one would see or believe oneself in that situation” (Whiten & Perner, 1991, S. 9).

Rolle im Begriff *Rollenübernahme* verweist nach Shantz (1975, S. 264) nicht auf die definierten sozialen Rollen sondern bezieht sich auf die momentanen Dialogrollen während einer Kommunikationssituation. Zur Übernahme einer Rolle kann dennoch das Wissen über soziale oder berufliche Funktionen des Gegenübers mit einbezogen werden.⁷ :

The term “role” is not used as most social psychologists and sociologists use it to refer to a class of shared behavioral expectations defined by a set of functions or traits, such as sex role or occupational role. While role taking may use information such as sex, age, occupation upon which to make inferences about another's attributes, the term is use more generally to include even momentary positions or relations between two or more people. For example, at one moment a person is in the role of the listener, and in the next, as speaker.⁸

Der Sprecher, der effektiv und erfolgreich mit dem Dialogpartner kommunizieren will, muß in seinem Diskurs der Perspektive des Dialogpartners Rechnung tragen. Ophoff (1986) spricht von *Partnerjustierung* und sieht Rollenübernahme als sozial-kognitives Regulativ der kommunikativen Partneranpassung.

Zur vollständigen Rollenübernahmefähigkeit gehören unterschiedliche Teilfähigkeiten wie die Fähigkeit zur Informationsübermittlung unter Berücksichtigung der Informationsbedürfnisse des Zuhörers, die Fähigkeit zur Vorhersage des Verhaltens anderer (*kognitive*

⁶Der Begriff Rollenübernahme geht auf Mead (1968) zurück. Nach dessen Auffassung dient Rollenübernahme dem besseren Verständnis der eigenen Rolle: Indem man die Rolle des anderen übernimmt, sich von dessen Perspektive betrachtet, kann man sich auf sich selbst besinnen und so den eigenen Kommunikationsprozeß lenken.

⁷So kann zum Beispiel ein professioneller Verkäufer die Rolle des Käufers während einer Verhandlung übernehmen, um dessen Reaktionen vorwegzunehmen, da er selbst oft als Kunde anderswo auftritt.

Rollenübernahme ist auch von Rollenspiel zu unterscheiden. Im letzteren übernimmt man die Merkmale einer anderen Person und agiert wie diese Person, während es bei Rollenübernahme nur um das Verständnis der Rolle der anderen Person und nicht um die Durchführung geht. So übernimmt man z.B. bei nichtkooperativen Spielen die Rolle des Gegenspielers, um dessen Strategie zu verstehen; man macht aber nicht dessen Zug (vgl. auch Leichtman, 1971)

⁸Herrmann (1982, S. 31) spricht z.B. vom “Sprecher in seiner temporären Rolle als Hörer”.

Rollenübernahme) oder zur Vorhersage der Kognitionen anderer (rekursives Denken)⁹, die Fähigkeit zur Überzeugung anderer, die Fähigkeit zur Darstellung von Rollen und die Fähigkeit zur Übernahme der Wahrnehmungsposition anderer.

Damit ein Individuum zur Rollenübernahme in voller Kompetenz befähigt wird, sind nach Flavell *et al.* (1968, S. 208–211) fünf Aspekte beteiligt (vgl. auch Flavell, 1992, S. 112–113):

1. Vorhandensein (existence): Das Individuum muß zunächst die Einsicht haben, daß andere Personen in der Regel nicht die gleichen Verhaltensweisen aufweisen wie es selbst. Sie haben eigene Emotionen, Gedanken und Perzeptionen, sie besitzen also eine andere Perspektive als die des Individuums.
2. Voraussetzung (need): Das Individuum muß erkennen, daß in bestimmten Situationen eine Analyse der Perspektive anderer gefordert wird, um deren Motive, Gedanken und Positionen besser zu verstehen und folglich sich adäquat verhalten zu können.¹⁰
3. Inferenz (inference): Dieser Aspekt betrifft die tatsächliche Schlußfolgerung bzw. Vorhersage des Verhaltens der anderen Person auf der Grundlage des Modells, das man von ihr hat. Hierzu muß man wissen, wie man genau die Analyse der Perspektive des anderen durchführen kann. Das erfordert die Fähigkeit, die relevanten Rollenmerkmale des anderen genau diskriminieren zu können.
4. Bereithalten (maintenance): Nach der Vorhersage des potentiellen Verhaltens der anderen Person müssen die gewonnenen Verhaltensbereitschaften koordiniert werden. Man muß beispielsweise wissen, wie man die Ergebnisse der Analyse der Fremdperspektive im Gedächtnis behalten kann, um sie gegebenenfalls einsetzen zu können.
5. Anwendung (application): Diese Teilfähigkeit betrifft die Umsetzung des Wissens um die potentielle Verhaltensweise bzw. Sicht der anderen Person. Dazu muß man wissen, wie man die Kognitionen auf das geplante Ziel anwenden kann.

Selman (1980) schlägt ein Modell für die Entwicklung der Perspektivenübernahme vor, das in fünf strukturell unterschiedliche Stufen gegliedert ist:

- Stufe 0: egozentrische Perspektivenübernahme.
Es wird zunächst noch nicht zwischen der eigenen Perspektive und der einer anderen Person unterschieden, obwohl man sich der Existenz subjektiver Perspektiven bewußt ist.
- Stufe 1: sozial-informationsbezogene Perspektivenübernahme.
Der Person wird es deutlich, daß andere Personen ihre eigenen Perspektiven haben, die von der eigenen verschieden sein können.

⁹Vgl. auch die Arbeiten zu Meta-Kognition (Miller *et al.*, 1970; Flavell, 1974 sowie Flavell *et al.*, 1993).

¹⁰Diese Komponente führt z.B. beim Kind im Entwicklungsverlauf zur eigenständigen Motivation zur Rollenübernahme. Das Kind lernt im Laufe der Zeit die Signale zu deuten, wann es Rollenübernahme einsetzen muß, z.B. in Situationen, in denen Rollenübernahmeaktivitäten nicht explizit, jedoch implizit gefordert werden.

- Stufe 2: selbstreflexive Perspektivenübernahme.
Hier bewegt sich die Person aus ihrer Perspektive heraus und reflektiert über die eigene Perspektive bzw. über die anderer Personen. Dabei wird es der Person bewußt, daß andere die gleiche kognitive Leistung durchführen können. (Diese Reziprozität führt z.B. zu Überlegungen der Form: Ich weiß, daß Du weißt, daß ich weiß...)
- Stufe 3: wechselseitige Perspektivenübernahme.
Die Person kann gleichzeitig sowohl die eigene Perspektive als auch die fremde Perspektive einnehmen und koordinieren. Durch diese mutuelle Perspektivenübernahme kann sich die Person gleichzeitig als Akteur und Objekt betrachten.
- Stufe 4: Perspektivenübernahme mit dem sozialen und konventionellen System.
Die Person entdeckt, daß individuelle Perspektivenübernahme nicht ausreicht, sondern daß zum gegenseitigen Verständnis andere Aspekte wie soziale Konventionen miteinbezogen werden müssen.

Rollenübernahme ist notwendig, jedoch nicht hinreichend. Es gibt nach Shantz (1981) viele Möglichkeiten, wie man sich ein Bild von seinem Dialogpartner machen kann, u.a. durch Beobachten anderer, Nachahmung, Abstraktion, Stereotypen: Rollenübernahme ist nur eine davon. Für soziales Handeln ist Rollenübernahme eine notwendige Bedingung, zumal es dabei darum geht, die Handlungsorientierung eines anderen zu erkennen (Geulen, 1982b). Die meisten Autoren stimmen darin überein, daß Rollenübernahme zwar eine notwendige, jedoch sicher noch keine hinreichende Voraussetzung für soziales Verhalten und adäquate Kommunikation ist (Flavell *et al.*, 1968; Steins & Wicklund, 1993).¹¹

Rollenübernahme ist ressourcenaufwendig. Sich in die Lage einer anderen Person hineinzuversetzen, erfordert einen hohen Aufwand: “role taking or interaction requires the expenditure of large amount of energy. It is a time-consuming operation continually to interpret the world from someone else's point of view” (Berlo, 1960, S. 133). Hierzu sagt er weiter (S. 134): “The first weakness [...] as a basis for prediction is that it consumes too much energy”.

Schwalbe (1988, S. 419) sieht Rollenübernahme als Strategie, bei der abzuwägen ist, ob sich dessen Einsatz lohnt: “Role taking can [...] be thought of as a strategy for exercising power. The propensity to use it depends in part [...] on estimates of its costs and benefits relative to using other available strategies.”

2.3.2 Beiträge aus dem Bereich “Theory of Mind”

Rollenübernahme in der Entwicklungspsychologie betrifft in erster Linie die Frage, wie Kinder die Fähigkeit erlernen, die Position des Gegenübers einzunehmen. Hingegen wird seit den 80er Jahren unter dem Begriff *Theory of Mind* (Astington, 1993; Whiten, 1991;

¹¹Sogar zur adäquaten Vorhersage der Handlung einer anderen Person ist Rollenübernahme nur eine hilfreiche Methode. Es ist nicht gewährleistet, daß Rollenübernahme stets zum Erfolg führt. Man denke z.B. an folgendes Zitat aus Patrick Süskinds Roman “Das Parfum” (S. 232): “Daß er in der Lage war, sich gedanklich in die Lage des *künftigen* Mörders seiner Tochter zu versetzen, machte ihn dem Mörder nämlich haushoch überlegen.” (von uns hervorgehoben)

Wellman, 1990) der Fokus darauf gerichtet, wie Kinder dazu kommen, den konzeptuellen Rahmen zu erwerben, innerhalb dessen sie die Meinung und mentale Zustände anderer Personen sowie ihre eigenen auffassen und erkennen können. Unter Theory of Mind versteht man den Teil der Entwicklungspsychologie, der sich mit der Frage beschäftigt, welche mentalen Zustände anderen Personen zugeschrieben werden bzw. wie diese Zustände simuliert und vorhergesagt (“Mindreading”) werden. Diese mentalen Zustände betreffen Aspekte wie Glauben (Beliefs), Wünsche (Desires), Absichten (Intentions) und propositionale Einstellungen (propositional Attitudes). Hierbei stellt sich die Frage, ob Kinder eine Theorie über diese mentalen Zustände haben bzw. ab wann sie die Aktionen anderer durch Betrachtung ihrer Gedanken und Wünsche erklären und vorhersagen können: “An individual has a theory of mind if he imputes mental states to himself and others. A system of inferences of this kind is properly viewed as a theory because such states are not directly observable, and the system can be used to make predictions about the behavior of others.” (Premack & Woodruff, 1978, S. 515, zitiert nach Astington, 1993, S. 4). Astington (1993, S. 169) sagt hierzu:

Children are able to alter [...] their *default settings*, which form the background against which the child operates. They consists of the child's own current mental states and the current state of the world (as it is known to the child). Role taking, or simulation of another person's experience, involves alteration of these default settings. It involves, for example, setting aside one's one desires and/or beliefs and taking on those of the other person in order to simulate what the other person will do or how she'll feel. The more default settings that need to be suspended, the more difficult the simulation will be.

2.3.3 Hörerorientierung

Hörerorientierung dient der besseren Kommunikation und beinhaltet die Justierung der zu vermittelnden Informationen an die Bedürfnisse und die Merkmale des Hörers.¹² In monologischen Instruktionen bezieht sich Hörerorientierung darauf, wie Sprecher “die Art der Informationsübermittlung den Informationsbedürfnissen und Erwartungen (eines) [...] Interaktionspartners anpassen” (Roßnagel, 1994, S. 11). Hier dient das Partnermodell (Herrmann & Grabowski, 1994) als Grundlage sprachlicher Anpassungsprozesse. Jedoch ist Hörerorientierung nicht immer gegeben, da mit ihr eine gewisse Belastung aufgrund beschränkter Ressourcen beim Sprecher einhergeht. Beispiel einer solchen Ressourcenbeschränkung ist die Belastung des Arbeitsgedächtnisses (Roßnagel, 1995). Hierzu liegen folgende Ergebnisse vor:

- Die Hörerorientierung beim mündlichen Instruieren steigt nur, wenn vorher die kognitive Belastung (d.h. Belastung des Arbeitsgedächtnisses) vermindert worden ist (Roßnagel, 1993).
- Eine hohe kognitive Belastung impliziert eine Verringerung der Hörerorientierung (Roßnagel, 1995).

¹²So passen sich z.B. Ärzte der Sprache ihrer Patienten an, wenn sie ihnen bestimmte Sachverhalte und Zusammenhänge in einfachen Worten erklären.

- Die Verringerung der Belastung des Arbeitsgedächtnisses hat nicht eine Erhöhung der Hörerorientierung zur Folge (Roßnagel, 1994).

Die angesprochene Belastung des Arbeitsgedächtnisses hat Auswirkungen auf die Antizipation. Um die Folgen einer Belastung des Arbeitsgedächtnisses beim Dialogpartner durch Simulation abzuschätzen oder etwa zur adäquaten Realisierung einer Form von Selbstkorrektur beispielsweise nach einer Äußerung unter Ressourcenbeschränkung zu realisieren, sollte das System selber die spezifischen Aufgaben mit vergleichbaren Ressourcen durchführen. Dabei soll bei Simulation eines Sprechers mit hoher kognitiver Belastung beachtet werden, daß eine Belastung des Arbeitsgedächtnisses mit einer Minderung der Hörerorientierung einhergeht (Roßnagel, 1995). Diese Ergebnisse könnten z.B. zur Folge haben, daß bei Simulation durch die Belastung des Arbeitsgedächtnisses des Dialogpartners durch das System auf die globale Antizipation verzichtet wird.

2.4 Simulation

Zur Operationalisierung von Rollenübernahme kann Simulation eingesetzt werden.¹³ Simulation umfaßt im allgemeinen zwei Aufgabentypen: Vorhersage und Interpretation.

Bei der Vorhersage lassen sich zwei Ansätze unterscheiden:

- eine deskriptive Vorgehensweise, der die Frage zugrundeliegt: “Was würde unter gegebenen Umständen passieren?”
- eine präskriptive (vorschreibende) Vorgehensweise mit der Überlegung “Wie führe ich einen erwünschten Zustand herbei?”.

Simulation kann auch zur Interpretation eingesetzt werden. Die Überlegung hierzu lautet: “Wie passe ich das Modell aufgrund der beobachteten Ergebnisse an?”.

2.5 Ansätze aus der Spiel- und Entscheidungstheorie

Ein Paradigma der Handlungstheorie, bei dem die Vorhersage des Verhaltens des Interaktionspartners eine Rolle spielt, geht auf die ökonomische Theorie des Verhaltens am Markt und deren Weiterentwicklung in der Spieltheorie (s. z.B. Fudenberg & Tirole, 1991; Myerson, 1991; Brams, 1994) zurück. Die Spieltheorie wird als Methode zur Analyse strategischer Interaktionen bei der Analyse von ökonomischem Verhalten, politischen Verhandlungen sowie Gesellschaftsspielen eingesetzt. Bei diesem Ansatz der Spieltheorie suchen die an der Interaktion beteiligten Spieler nach optimalen Gewinnstrategien. Da anzunehmen ist, daß auch die Gegenspieler nach optimalen Strategien suchen, ist es entscheidend,

¹³Nach Auffassung von Harris (1991, S. 299) erfordert eine Simulation nur ein Modell des Interaktionspartners: “Children make predictions about other people's actions, thoughts, emotions by running a simulation. A simulation calls for a working model of the other person but not a theory”. Astington (1993, S. 168–171) spricht von Simulation als “mechanism by which role-taking might operate”.

vorherzusagen, wie sie sich verhalten werden. In diesem Zusammenhang spielt das Konzept des Gleichgewichtes als stabile Konstellation von eigenem Verhalten und Erwartungen bzgl. des Verhaltens der Gegenspieler eine wesentliche Rolle. Gleichgewichte sind stabile Konstellationen, in denen es sich für keinen der Handelnden lohnt, sein Verhalten einseitig zu verändern. Es können mehrere Gleichgewichte existieren und es wird die Annahme des gemeinsamen Wissens (Common Knowledge) gemacht. Das Gleichgewichtskonzept wird jedoch in der vorliegenden Arbeit nicht verwendet, da der Dialogpartner nicht als Gegner im spieltheoretischen Sinne angesehen wird, sondern als Quelle für bestimmte Ereignisse (z.B. Dialogzüge), die nur beschränkt vorhersagbar sind.

In der Entscheidungstheorie (Laux, 1991; Rapoport, 1989) werden Situationen betrachtet, in denen ein Agent sich durch seine Handlung in mindestens zwei unterschiedliche Zustände versetzen kann. Dabei wird angenommen, daß er sich in seiner Entscheidungsfindung rational verhält. Es gibt unterschiedliche Rationalitätsbegriffe: In der Entscheidungstheorie wird das sogenannte *Principle of Maximum Expected Utility* (PMEU) zugrundegelegt, das sich nach Russell und Norvig (1995, S. 419) folgendermaßen charakterisieren läßt: “an agent is rational if and only if it chooses the action that yields the highest expected utility, averaged over all the possible outcomes of the action”. Man unterscheidet auch zwischen normativer, präskriptiver und verhaltensorientierter Entscheidungstheorie.

2.6 Systeme aus dem Bereich der Künstlichen Intelligenz

Aus dem Bereich der KI wurde das Phänomen der Antizipation vor allem in den Spielbaum-Suchverfahren (Reinefeld, 1989) und der Benutzermodellierung realisiert.

Bei Spielbäumen handelt es sich um Zwei-Personen-Null-Summen-Spiele mit vollständiger Information: Am Spiel sind zwei Personen beteiligt, die abwechselnd jeweils einen Zug ausführen; die Gewinnauszahlung des einen Spielers entspricht genau dem Verlust seines Gegenspielers; den Spielern sind alle vorangehenden und nachfolgenden Züge in einer Spielsituation bekannt. Ein wesentliches Element hierbei ist die verwendete statische Bewertungsfunktion – auch Auszahlungsfunktion genannt – als quantitatives Maß für den Nutzen eines Spielzuges für den Spieler. Es wird angenommen, daß die Spieler sich rational verhalten, indem sie sich immer für den bestbewerteten Zug nach dem Minimimax-Verfahren entscheiden. An einem bestimmten Stand während des Spiels betrachtet ein Spieler seine Handlungsmöglichkeiten, versetzt sich in die Lage des Gegenspielers, um dessen folgende Aktion vorherzusagen. Er betrachtet wiederum seine Erwidierungen auf die Aktion des Gegenspielers und so weiter. Man spricht dann von einer Knotenexpansion. Da aber die Menge der zu berücksichtigenden Möglichkeiten gigantisch sein kann¹⁴, wird nicht versucht, einen vollständigen Spielbaum aufzubauen. Der Spielbaum wird lediglich bei der Bewertung des nächsten Zuges bis zu einer festgelegten Tiefe expandiert. Nach der Expansion werden zunächst den terminalen Knoten Werte zugewiesen, die durch die Anwendung der Bewertungsfunktion auf die durch den Knoten dargestellten Spielsituation ermittelt werden. Die

¹⁴Das Damespiel hat schätzungsweise 10^{78} Zugvarianten und das Schachspiel sogar 10^{120} (Reinefeld, 1989).

übrigen Knoten werden von unten nach oben “zurückbewertet”: An den Situationen, bei denen der Spieler (MAX) am Zuge ist, wird dem Knoten die höchste Bewertung aller direkten Nachfolgeknoten zugewiesen; wenn der Gegenspieler (MIN) am Zug ist, wird dem betrachteten Knoten die niedrigste Bewertung zugewiesen. Der Prozeß endet, wenn der Wert der Wurzel des betrachteten Baumes ermittelt wird. Das ist der Ausgangszustand, bei dem ein Spieler den weiteren Verlauf des Spiels antizipieren wollte. (Auf die effizienteren Varianten des Minimimax-Verfahrens, wie etwa das *Alpha-Beta* oder die entscheidungstheoretischen Verbesserungen, wird hier nicht eingegangen.)

In der Benutzermodellierung wurde das Phänomen der Antizipation mit Hilfe sog. *Antizipationsrückkopplungsschleifen* (Jameson & Wahlster, 1982; Wahlster & Kobsa, 1989) realisiert. Dabei wird die Analysefähigkeit des Systems dazu verwendet, die voraussichtliche Interpretation eines geplanten Dialogbeitrages durch den Dialogpartner vorwegzunehmen. Dabei wird angenommen, daß die Verarbeitungsprozesse des Systems und die des Benutzers ähnlich sind. Hierzu stellt sich das System die Frage:

“Wenn ich die von mir geplante Äußerung bzw. Aktion, bezogen auf das Wissen und die Überzeugungen, die ich dem Dialogpartner zuschreibe, analysieren würde, welche Wirkung hätte dann die Äußerung auf mich?”

Bei dieser Vorgehensweise ist allerdings vorausgesetzt, daß das System über ähnliche Verarbeitungsprozesse wie denen des Dialogpartners verfügt oder zumindest weiß, wie es diese Prozesse adäquat simulieren kann.¹⁵ Im ersteren Fall kann sich das System selbst als Modell nehmen, um die Verarbeitung seitens des Dialogpartners zu simulieren. Im zweiten Fall ist es notwendig, daß es über ein Modell des Dialogpartners verfügt und dieses beispielsweise in einer separaten Instantiierung des Systems in der Rolle des Dialogpartners verwalten kann.

Dual zu der Antizipation der Interpretation, die der Dialogpartner machen wird, kann auch gefragt werden, was den Dialogpartner dazu veranlaßt hat, eine bestimmte Äußerung zu machen. Hierbei stellt man sich die Frage:

“Wenn ich die vom Dialogpartner gemachte Äußerung bzw. Aktion gemacht hätte, aufgrund welcher Überzeugungen und Ziele hätte ich dies gemacht?”

Ferner kann auch die Antizipationsrückkopplung präskriptiv verwendet werden, um der Frage nachzugehen, wie ein erwünschter Zustand oder Eindruck herbeigeführt werden kann. Die Frage hierbei ist:

“Was muß ich vorher gemacht haben bzw. was muß vorher gegolten haben, damit ich einen bestimmten Zustand erreiche?”

Antizipationsrückkopplungsschleifen lassen sich in zwei Kategorien unterteilen (vgl. Wahlster & Kobsa, 1989, S. 22–26):

¹⁵Vgl. die Ausführung im Abschnitt 2.3.1.3 und die Auffassung von Whiten und Perner (1991, S. 10) über das Verständnis der mentalen Zustände anderer: “It requires the tacit assumption that the other person does have similar mental states in similar conditions.”

- Bei lokalen Antizipationsrückkopplungsschleifen wird nur ein Teil des Verstehens- bzw. des Verarbeitungsprozesses antizipiert. Hierfür wird dementsprechend nur ein Teil des Systems verwendet. Um beispielsweise die Interpretation einer Ellipse zu antizipieren, wird in HAM-ANS (Jameson & Wahlster, 1982) lediglich das zur Ellipsengenerierungskomponente duale Modul der Ellipsenrekonstruktion verwendet.
- Bei globalen Antizipationsrückkopplungsschleifen wird eine Instantiierung des kompletten Systems oder zumindest sämtlicher Teile des Systems verwendet, um den kompletten Verstehens- bzw. den Verarbeitungsprozeß vorwegzunehmen.

Antizipationsrückkopplung wurde z.B. verwendet zur Ellipsengenerierung (Jameson & Wahlster, 1982), zur Beschreibung von Straßenszenen (Novak, 1987b; 1987a), zur pragmatischen Interpretation von Äußerungen (Jameson, 1989), zur Generierung von Zeigegeesten in einer multimodalen Schnittstelle (Jung *et al.*, 1989), zur Antizipation von Benutzerinferenzen (Joshi *et al.*, 1984a; 1984b; Zukerman & McConachy, 1993a; 1993b) und zur Visualisierung der Bildvorstellung des Hörers anhand von verbalen Beschreibungen durch den Sprecher (Schirra, 1995; Blocher & Schirra, 1995).

Im Folgenden werden einige Systeme präsentiert, die die Idee der Antizipationsrückkopplung eingesetzt haben. Dabei orientieren wir uns, nachdem wir die Grundrisse jedes Systems beschrieben haben, an folgenden Punkten (soweit es nötig ist):

- Zweck der Antizipationsrückkopplungsschleife (abgekürzt als AFL für Anticipation Feedback Loop): Welche Ziele vom System werden durch die Antizipationsrückkopplung unterstützt? Handelt es sich um lokale oder globale AFLs bzw. werden die AFLs für die Generierung oder auch zur Analyse verwendet?
- Partnermodellierung: Wie (wenn überhaupt) werden Unterschiede zwischen dem System und dem Partner berücksichtigt und modelliert?
- Realisierung: Wie wurde die AFL im System realisiert? Welche sind die zugrundeliegenden Formalismen?
- Unsicherheitsbehandlung: Wie (wenn überhaupt) wird unsicheres Wissen über die Repräsentationen und Inferenzen des Dialogpartners berücksichtigt?
- Welche sind weitere Besonderheiten des Systems; welche sind die Stärken und Schwächen der Verwendung von AFLs im System?

2.6.1 HAM-ANS: Ellipsengenerierung

Zu einem kooperativen und kommunikativ-adäquaten natürlichsprachlichen System gehört auch ein expliziter Elliptifizierungsprozeß. Bei der Generierung von elliptischen Äußerungen in HAM-ANS¹⁶ (Jameson & Wahlster, 1982), einem transportablen natürlichsprachlichen Zugangssystem, wurde die Technik der Antizipationsrückkopplung eingesetzt.

Zweck der AFL. Die AFL im HAM-ANS dient der Generierung eindeutig interpretierbarer elliptischer Äußerungen. Die Ellipsengenerierung vollzieht sich in zwei Schritten,

¹⁶HAMburger Anwendungsorientiertes Natürlichsprachliches System

- **User: Haben fünf Mitarbeiter mehr als drei Wochen Urlaub?**
Wenn fünf Mitarbeiter weniger als drei Wochen, jedoch nicht mehr als zwei Wochen, oder wenn nur zwei Mitarbeiter mehr als drei Wochen Urlaub haben, dann kann ein kooperatives System statt nur mit “Nein” zu antworten die folgende Antwort geben:
- **System: Nein, zwei.**
Jedoch antizipiert das System durch Einsatz seiner Ellipsenrekonstruktionskomponente, daß diese Ellipse den Benutzer zu einer mehrdeutigen Interpretation verleiten kann. Es entscheidet sich dann abhängig vom Kontext eine der beiden folgenden Ellipsen zu generieren.
- **System: Nein, zwei Wochen.**
- **System: Nein, zwei Mitarbeiter.**

Abbildung 2.2: Beispiel eines Dialogs mit Generierung einer elliptischen Äußerung in HAM-ANS.

nämlich Identifikation möglicher Ellipsen und Überprüfung dieser Ellipsen in einer Antizipationsrückkopplungsschleife. Durch die AFL werden solche Teile der semantischen Repräsentation der vollständigen Antwort eliminiert, die identisch mit denen der im semantischen Dialoggedächtnis abgespeicherten Benutzerfrage sind.

Partnermodellierung. In Bezug auf die Ellipsengenerierung findet keine explizite Modellierung von \mathcal{U} statt. Es werden lediglich *a priori* Existenzannahmen des Benutzers verwendet.

Realisierung. Das System verwendet sich selbst als Modell, um die Rekonstruierbarkeit der Ellipse zu überprüfen. Hierzu setzt das System die Komponente ein, die es zur Interpretation der vom Benutzer geäußerten Ellipsen verwendet. Für jede mögliche Form der Elidierung wird durch Verwendung der Ellipsenrekonstruktionskomponente überprüft, ob die erzeugte Ellipse für das System selbst eindeutig rekonstruierbar und verständlich wäre, falls es diese Ellipse in dem betrachteten Kontext interpretieren sollte. Die kürzeste verständliche Ellipse wird generiert.

Unsicherheitsbehandlung. Das System setzt voraus, daß der Benutzer sich genau so verhält, wie das System es auch täte. Es macht keine individuellen Annahmen über die Präferenzen des Benutzers.

Diskussion. Wenn HAM-ANS eine potentielle Ellipse ablehnt, weil diese mehrdeutig ist, wird der nächste Kandidat gewählt und auf Eindeutigkeit geprüft. Es findet keine Diagnose der Ablehnungsursachen statt. Es werden nur kontextuelle Ellipsen behandelt, d.h. solche, die die Existenz einer vorhergehenden Ellipse voraussetzen.

2.6.2 IMP: Pragmatische Interpretation von Äußerungen

IMP¹⁷ (Jameson, 1989; 1983) liegt die Idee zugrunde, daß Sprecher in einer bewertungsorientierten Dialogsituation sich bemühen das Bild zu beeinflussen, das der Hörer von ihren Dialogmotivationen und ihren Hörermodellen hat. Wenn ein Hörer die Äußerungen eines Sprechers interpretiert, berücksichtigt er nicht nur dessen Dialogmotivation sondern auch dessen Modell von ihm als Hörer. Bei der Generierung einer Äußerung berücksichtigt der Sprecher, daß seine Äußerungen so vom Hörer interpretiert werden.

Zweck der AFL. So wurde IMP zur pragmatischen Interpretation von Äußerungen in einem Vorstellungsgespräch (Jameson, 1989) und in früheren Arbeiten bei Zimmerreservierungen (Jameson, 1983) eingesetzt. In diesen sog. *bewertungsorientierten* Dialogen führen zwei Teilnehmer, ein *Evaluator* und ein *Informant*, ein Gespräch zur Bewertung eines *Objektes*, das eine Sache, eine Person oder eine Aktion sein kann, bezüglich einigen Kriterien. In den von IMP behandelten Situationen übernimmt der Informant die Rolle des Sprechers (\mathcal{S}) und der Evaluator die Rolle des Hörers (\mathcal{H}). In diesen Situationen gibt \mathcal{S} aufgrund seiner Dialogmotivation und seines Modells vom Hörer unaufgefordert Kommentare bzw. Zusatzkommentare ab, die Auswirkung auf die Bewertung des Objektes durch \mathcal{H} haben. Die Auswahl solcher Kommentare wird neben \mathcal{S} s jeweiliger Einstellung davon beeinflußt, wie \mathcal{H} \mathcal{S} s Dialogmotivation bei der Interpretation von \mathcal{S} s Äußerungen berücksichtigt und wie \mathcal{H} \mathcal{S} s Modell von ihm (\mathcal{H}) wahrnimmt. Hierzu antizipiert IMP \mathcal{H} s Interpretation seiner Zusatzkommentare, wobei es die eventuell verschiedene Einstellung berücksichtigt, von der es annimmt, daß sie ihm vom Benutzer zugeschrieben wird.

Partnermodellierung. IMP verwaltet neben der tatsächlichen Systemeinstellung (positiv, negativ oder objektiv) eine vorgespiegelte Einstellung.

Realisierung. Bei der Generierung eines Kommentars vergleicht IMP die Eindrucksveränderung, die vom geplanten Kommentar bewirkt wird, mit der erwünschten Eindrucksveränderung beim Hörer. Hierzu wird eine lokale Antizipationsrückkopplungsschleife eingesetzt, um die Interpretation des geplanten Kommentars durch \mathcal{H} vorwegzunehmen und ggf. die Kommentare wegzufiltern, die nicht zur angestrebten Eindrucksveränderung beitragen bzw. diejenigen, die die tatsächliche Einstellung des Systems verraten würden. Abhängig von den Ergebnissen der lokalen Antizipationsrückkopplungsschleife ergeben sich folgende Fälle:

1. Der antizipierte entstehende Eindruck ist kompatibel mit dem vorgespiegelten Eindruck; dann wird der geplante Kommentar verbalisiert;
2. Der geplante Kommentar ist inkonsistent mit dem vorgespiegelten Eindruck, dann wird er verworfen und es wird nach einem anderen gesucht; oder
3. Der geplante Kommentar soll ergänzt werden: Zu dem geplanten Kommentar können eventuell durch adäquate Konjunktionen oder Adverbialen eingeleitete Zusatzkommentare hinsichtlich der vorgespiegelten Einstellung hinzugefügt werden. Einleiten-

¹⁷Der Name ist aus dem dem System zugrundeliegenden zentralen Begriff des Eindrucks (*impression*) über den Gesprächspartner abgeleitet.

de Konjunktionen oder Adverbialen sind z.B. *leider*, *unglücklicherweise*, *glücklicherweise*, *ich möchte Sie darauf hinweisen* (by the way). Abbildung 2.3 zeigt ein solches Beispiel.

- **User:** What about facilities outside the room?
- **System:** There's kitchen and there's a bathtub and *by the way*, the room is very large.
- **User:** Is there a washing machine and a dryer?
- **System:** No, *but* there is a washing machine.
- **User:** Is the room on a quiet street?
- **System:** No, in fact *unfortunately* there's a lot of noise from the street.

Abbildung 2.3: Auszug aus einem Beispieldialog, in dem IMP die Rolle des Immobilienmaklers übernimmt.

(Die im Dialog hervorgehobenen Äußerungen werden vom IMP zu den ursprünglich geplanten Kommentaren hinzugefügt, um beim Benutzer die erwünschten Eindrucksverschiebungen herbeizuführen.)

Unsicherheitsbehandlung. In IMP werden zwei Fälle der Unsicherheit berücksichtigt: Einerseits die Unsicherheit von S in U s Bewertungskriterien und andererseits die Unsicherheit von S über seine projizierten Merkmale. Die Gewichte werden durch Intervalle ersetzt.

2.6.3 BILD: Bidirektionale Verarbeitung mit Hilfe reversibler Grammatiken

Dem System BILD¹⁸ (Neumann, 1998; 1994 sowie Neumann & van Noord, 1994) liegen zwei Prinzipien zugrunde: Bidirektionalität und Selbstüberwachung. Zur Realisierung der Bidirektionalität werden *reversible Grammatiken* eingesetzt, die sich durch die Verwendung ein und derselben Grammatik sowohl bei der Analyse als auch bei der Produktion natürlichsprachlicher Ausdrücke zeigen. Das resultierende bidirektionale Modell verzahnt die Module zur Analyse und Generierung in einer uniformen Verarbeitung. Diese enge Verzahnung ermöglicht die Selbstüberwachung des Systems: der Parser wird zur Unterstützung des Generierungsprozeß eingesetzt und umgekehrt.

Zweck der AFL. In BILD wird eine AFL sowohl bei der Generierung als auch bei der Analyse eingesetzt. Beide AFLs dienen dazu, eindeutig interpretierbare natürlichsprachliche Äußerungen zu generieren bzw. die Mehrdeutigkeit in einer Benutzeräußerung aufzulösen. Hierbei wird eine Verarbeitungsrichtung als Monitoring (Überwachung) für die andere eingesetzt.

¹⁸**B**idirektionale **L**inguistische **D**eduktion

Realisierung. Es sind zwei Fälle zu unterscheiden:

- Bei der Analyse geht es um eine benutzergesteuerte Disambiguierung: Der Generator wird während des Parsingprozesses zur Auflösung von Ellipsen bzw. zum Generieren von Paraphrasen eingesetzt, die den unterschiedlichen Lesarten der Eingabe entsprechen. Der Benutzer wählt dann z.B. in einem Klärungsdialog die der intendierten Lesart entsprechende Paraphrase aus (s. Abbildung 2.4).

- **User: Die Männer haben die Frau mit dem Fernglas gesehen.**
Falls der Benutzer eine solche Eingabe macht, stellt das System fest, daß die Benutzereingabe mehrdeutig ist. Nach der Analyse ergeben sich zwei semantische Repräsentationen, die an den Generator weitergeleitet werden. Die mehrdeutigen Teile werden ersetzt und der Generator erzeugt danach die eindeutigen Paraphrasen.
Für die erste Lesart wird folgende Paraphrase erzeugt:
- **System: Die Männer haben mit dem Fernglas die Frau gesehen.**
Für die zweite Lesart präsentiert das System dem Benutzer einer der beiden folgenden Paraphrasen.
- **System: Die Frau mit dem Fernglas haben die Männer gesehen.**
- **System: Die Männer haben die Frau, die das Fernglas hat, gesehen.**
Der Benutzer wählt dann aus den vorgeschlagenen Paraphrasen die von ihm intendierte Lesart aus.

Abbildung 2.4: Beispiel einer Verzahnung von Parsing und Generierung.

- Bei der Generierung geht es um die Vermeidung ambiger und damit potentiell mißverständlicher Äußerungen. Um herauszufinden, ob der Satz mehrdeutig ist, bzw. um die Quellen dieser Ambiguität festzustellen, wird der Parser aufgerufen. In einem Revisionsschritt werden dann die Teile der Äußerung durch Heranziehen kontextueller Informationen umformuliert, so daß sie eindeutig verstanden werden.

Diskussion. BILD ermöglicht die Wiederverwendung von Teilergebnissen. So können in der Generierungsrichtung bereits analysierte bzw. durch Parsing berechnete Strukturen verwendet werden und umgekehrt.

2.6.4 NAOS: Beschreibung von Straßenszenen

In NAOS¹⁹ (Novak, 1987b; 1987a), einem System zur Textgenerierung aus visuellen Daten, wurde auch die Idee der Antizipationsrückkopplung realisiert. Das System erzeugt ausgehend von einer aufgenommenen Bildfolge über eine zeitveränderliche Straßenverkehrsszene eine kohärente natürlichsprachliche Beschreibung der Szene.

¹⁹NA natürlichsprachliche Beschreibung von Objektbewegungen in einer Straßenverkehrsszene

Zweck der AFL. Die AFL in NAOS dient der Generierung kohärenter Beschreibungen von Objektbewegungen in einer Straßenverkehrsszene, um dem Hörer, der die Szene nicht sieht, den Aufbau einer Vorstellung dieser Szene zu ermöglichen.

Partnermodellierung. Es findet keine explizite Modellierung des Verstehensprozesses, der Ziele, Pläne und Überzeugungen des Hörers statt. In NAOS wird lediglich das *a priori* Wissen des Hörers über die Szenen repräsentiert und es erfolgt eine Anpassung nach jeder Systemäußerung. Darüber hinaus wird ein Koreferenzennetz zur Verwaltung der Tiefenkasus eingesetzt.

Realisierung. Bei der Generierung von Beschreibungen visueller Daten wird eine Vorwegnahme des Verstehensprozesses des Hörers vor jeder Äußerung vorgenommen, so daß er sich ein Bild der Szene machen kann. Er selbst sieht die Szene nicht, kennt aber deren statischen Hintergrund. Ziel der Beschreibung ist es, daß der Hörer die verbal beschriebenen Trajektorien der Objekte in sein Modell des statischen Hintergrundes integriert. Diese Vorwegnahme des Hörerverständnisses wird in NAOS als **antizipierte Visualisierung** bezeichnet.

Die AFL wird bei der Auswahl optionaler Tiefenkasus eingesetzt. Es wird jedoch keine Analyse des geplanten Beitrages analysiert und es handelt sich nicht um eine echte globale Antizipation: "Der [...] Auswahlalgorithmus berücksichtigt den Verstehensprozeß in gewissem Grade und vermeidet eine explizite Rückkopplung durch z.B. eine Parsingkomponente. Die Vorgehensweise ist dadurch zielgerichtet, daß bei der Auswahl jeweils geprüft wird, ob die [zu generierende] Information schon bekannt ist." (Novak, 1987a, S. 87)

Unsicherheitsbehandlung. Aus der Repräsentation der Szene kann Unsicherheit resultieren. In NAOS wird diese jedoch nicht weiterbetrachtet.

Diskussion. Anders als in dem System ANTLIMA, das als nächstes vorgestellt wird, findet keine inkrementelle Generierung bzw. Analyse statt. In NAOS erfolgt die Generierung erst nach dem Abschluß der Ereigniserkennung für die gesamte Szenenfolge.

2.6.5 ANTLIMA: Visualisierung der Bildvorstellung des Hörers

Das System ANTLIMA²⁰ (Schirra, 1995; Blocher & Schirra, 1995; Blocher, 1994) simuliert einen Radioreporter, der für Hörer von einem Fußballspiel berichtet.

Zweck der AFL. Die AFL in ANTLIMA wird zur Generierung kommunikativ adäquater natürlichsprachlicher Beschreibungen eingesetzt, so daß der Hörer sich eine genaue Vorstellung eines Spielgeschehens machen kann. ANTLIMA geht von einem typischen Hörer aus.

Partnermodellierung. Für die Antizipation ist die Repräsentation des Hörerverständnisses nach der Ereigniserkennung als Menge von Propositionen von Interesse, da später die Differenzpropositionen den Unterschied zwischen der intendierten und der erzielten Wirkung darstellen. Im Hörermodell werden auch die räumlichen Relationen repräsentiert.

²⁰ANTicipation of the Listners' IMAgery

Diese räumlichen Relationen treten als Belegung optionaler Tiefenkasus auf. Deren Anwendbarkeit wird durch sog. Typikalitätspotentialfelder beschrieben.

Realisierung. Die in ANTLIMA verfolgte Strategie ist die Vorwegnahme der Bildvorstellung der Hörer durch Vergleich des intendierten und imaginierten Äußerungsgehalts. Hierzu stellt sich die Hörermodellierungskomponente die Frage: “Wie würde der Hörer die geplante Äußerung im gegebenen Kontext verstehen?” bzw. “Wie soll gegebenenfalls die geplante Äußerung verbessert werden?”. Zur Realisierung dieses sog. **antizipierten Verständnisses** setzt ANTLIMA in einem Re-Analyse-Prozeß des geplanten Beitrages einen Teil der Analysekomponente des SOCCER-Systems ein, um zu überprüfen, ob die aus der Interpretation des geplanten Beitrags resultierende räumliche Vorstellung des Hörers der eigentlichen Szene entspricht (Blocher, 1994). Falls ja, wird die Äußerung ausgegeben. Ansonsten müssen Korrekturmaßnahmen eingeleitet werden. Die Re-Analyse liefert eine Propositionenmenge, die dem vom Hörer erwarteten Verständnis der geplanten Äußerung sowie den aus dem aufgrund des Verständnisses aus dem Gesagten gezogenen Implikaturen entspricht. Falls es Abweichungen zwischen diesem Verständnis und der vom Sprecher intendierten Wirkung gibt, muß der Unterschied zwischen dem erwünschten und der bezüglich des Hörerverständnisses erzielten Wirkung korrigiert werden. Hierfür wird die geplante Äußerung etwa durch zusätzliche Propositionen als Belegung eines Tiefenkasus modifiziert. Dies führt z.B. zu Äußerungen der Form “Er spielt Moll, dem Verteidiger, den Ball *nahe der hinteren Außenlinie* zu” statt “Er spielt Moll, dem Verteidiger, den Ball zu”.²¹

Diskussion. ANTLIMA ermöglicht die Wiederverwendung der während eines Antizipationszyklus gewonnenen Ergebnisse. Falls die Interpretation r_1 einer potentiellen Äußerung z_1 nicht zufriedenstellend ist, wird ein neues z_2 unter Berücksichtigung der bei z_1 entstandenen Schwierigkeiten erzeugt. z_2 wird unter der Annahme generiert, daß durch die Verbesserung von z_1 sämtliche Mißverständnisse ausgeräumt sind.

2.6.6 ZORA: Zeigegestengenerierung

Das System ZORA²² (Jung *et al.*, 1989, Wahlster, 1991) unterstützt die Generierung von Zeigegesten in dem Generator POPEL des XTRA-Systems. XTRA (Allgayer *et al.*, 1989) ist ein multimodales Zugangssystem zu Expertensystemen, das natürliche Sprache mit Graphik und Zeigegesten sowohl für die Ein- als auch für die Ausgabe kombiniert. Ausgehend von einer natürlichsprachlichen Eingabe –eventuell mit Zeigegesten– extrahiert das System relevante Daten für das Expertensystem, bearbeitet die Benutzereingabe und generiert anschließend Antworten aus den Ergebnissen und Erklärungen des Expertensystems. Die generierte natürlichsprachliche Ausgabe kann wiederum auch Zeigegesten enthalten.

²¹ ANTLIMA kann neben der zusätzlichen Spezifizierung durch Angabe optionaler räumlicher Tiefenkasus andere korrigierende Eingriffe in die geplanten Äußerungen realisieren: Initiierung von Klärungseinschüben, Korrektur des aktuellen Kontextes, Absichern unterspezifischer Nominatoren, Beeinflussung der Mitteilungswürdigkeit, Absicherung des Gebrauchs von Indikatoren.

²² ZeigegestengeneratORprogrAmm

Zweck der AFL. In der von XTRA behandelten Domäne kann die generierte natürlichsprachliche Ausgabe Zeigegesten auf ein Formular für den Lohnsteuerjahresausgleich enthalten. Diese Zeigegesten sind bestimmt durch das Zeigemittel (Bleistift, Finger, Hand) sowie den Gestenverlauf, der sich aus autonomen Bewegungen zusammensetzt. Die Antizipationsrückkopplung wird zur Überprüfung potentieller Zeigegesten durch die Analysekomponente eingesetzt, um zu überprüfen, ob angenommen werden kann, daß die Zeigegesten vom Benutzer auch richtig interpretiert werden.

Partnermodellierung. In ZORA wird von einem Standardbenutzer ausgegangen und es findet keine explizite Modellierung des Benutzers statt.

Realisierung. Der ZORA-Algorithmus besteht aus vier Schritten: einer Vorlauf-, einer Klassifikations-, einer Auswahl- und einer Visualisierungsphase. In der Vorlaufphase werden die geometrischen Gegebenheiten des zu zeigenden Objekts bzgl. Ort, Gestalt und Größe berechnet. Dann erfolgt die Klassifikationsphase zur Bestimmung des Objekttyps, auf den gezeigt werden soll. Während dieser Phase findet die Auswahl der für diesen Objekttyp in Frage kommenden Zeigegesten statt. Eine Zeigegeste ist durch das Zeigemittel (Bleistift, Finger) und den Gestenverlauf definiert, der sich aus atomaren Bewegungen zusammensetzt. In der Selektionsphase wird die adäquateste Zeigegeste durch eine lokale Antizipationsrückkopplungsschleife bestimmt. Hierzu wird TACTILUS-II (Wille, 1989), dem Pendant für die deiktische Analyse von Zeigegesten, eingesetzt. TACTILUS-II²³ simuliert dann die Benutzerreaktion, indem getestet wird, ob die selektierte Zeigegeste korrekt interpretiert wird. Dabei geht das Modul folgendermaßen vor: Es berechnet die Regionen, auf die gezeigt werden soll; propagiert diese Region durch die Formularhierarchie; und sortiert sie nach Güte. Eine Region gilt als identifiziert, wenn sie in der zurückgelieferten Liste enthalten ist. Die dadurch gewährleistete Antizipation der Interpretation der Zeigegeste durch den Benutzer erlaubt eine Korrektur der Zeigegeste, falls das intendierte Zielobjekt durch TACTILUS-II nicht eindeutig identifiziert wird. Wenn die Geste das intendierte Objekt eindeutig spezifiziert, wird sie in der Visualisierungsphase auf dem Bildschirm simuliert. Andernfalls werden alternative Zeigegesten solange getestet und gegebenenfalls modifiziert, bis eine eindeutige Zeigegeste gefunden wird oder alle Gesten erfolglos getestet worden sind. Diese Überprüfung gewährleistet, daß die ausgewählte Zeigegeste kommunikativ adäquat ist.

2.6.7 Antizipation in der Dialogübersetzung

Antizipationsrückkopplung kann auch in der Dialogübersetzung bzw. Dolmetschen eingesetzt werden (Wahlster, 1989). Vorbild dieser Vorgehensweise ist die Tatsache, daß professionelle Dolmetscher bei der Dialogübersetzung sowohl die Effekte ihrer Äußerungen in der Zielsprache als auch die Intentionen der Sprecher antizipieren.

²³Für die Analyse von Zeigegesten bedient sich TACTILUS-II einer Formularhierarchie, die die geometrischen und logischen Strukturen der Regionen des Lohnsteuerjahresausgleichsformulars sowie die Verbindungen der Regionen untereinander und zum konzeptuellen Domänenwissen beschreibt. So muß etwa beschrieben werden, wo sich die Region "Werbungskosten" auf dem Formular befindet und welche die Verbindung zur konzeptuellen Wissensbasis ist.

Zweck der AFL. Ziel des Einsatzes von AFLs bei der Dialogübersetzung ist es, eine präzise Übersetzung bzw. Dolmetschen einer Äußerung von der Quell- in die Zielsprache zu erreichen.

Partnermodellierung. Bei der Dialogübersetzung werden zwei Benutzermodelle eingesetzt: Einerseits ein Modell des Sprechers bei der Analyse von Äußerungen in der Quellsprache und andererseits ein Modell des Hörers bei der Generierung von Äußerungen in die Zielsprache.

Realisierung. Hierzu werden zwei Analysekomponenten (für die Quell- bzw. für die Zielsprache) sowie zwei Generierungsmodule (jeweils für die Quell- bzw. die Zielsprache)²⁴ benötigt. Es wird von zwei Benutzermodellen ausgegangen: Ein Modell des Sprechers bei der Analyse von Äußerungen in der Quellsprache und ein Modell des Hörers bei der Generierung von Äußerungen in die Zielsprache. In dem vorgeschlagenen radikalen Ansatz handelt es sich um eine doppelte Antizipation durch Ausnutzung des Benutzermodells über den Sprecher und der Generierungsfähigkeiten des Systems für die Quellsprache während der Analyse und Ausnutzung des Benutzermodells über den Hörer und der Analysefähigkeiten des Systems für die Zielsprache während der Generierung. Die zwei Antizipationsrückkopplungsschleifen werden dazu benutzt, um zum einen während des Analyseprozesses unter Einsatz des Benutzermodells über den Sprecher die Frage zu beantworten "Aufgrund welcher Annahmen, Ziele und Pläne hat der Sprecher die zu übersetzende Äußerung produziert?" und zum anderen während des Generierungsprozesses unter Verwendung des Modells über den Hörer die dazu duale Frage zu beantworten "Welche Wirkung wird die von mir geplante Übersetzung auf den Hörer haben, wenn ich ihm bestimmte Annahmen, Ziele und Pläne unterstelle?". Die Analyse-AFL hat einen abduktiven Hintergrund, während die Generierungs-AFL einem deduktiven Ansatz folgt. Beide AFLs steuern den Übersetzungsprozeß und führen gegebenenfalls zur Revision der geplanten Übersetzung.

2.6.8 WIP-GD: Antizipation bei der iterativen Graphikgenerierung

WIP²⁵ (Wahlster *et al.*, 1993), ein System zur multimedialen Informationspräsentation in Abhängigkeit von verschiedenen Generierungsparametern, verfügt neben einem Textgenerator, einem Layoutmanager und einer Komponente zur Verwaltung des Domänenwissens, über einen Graphikgenerator (WIP-GD). In WIP-GD (Rist, 1996) sind die Design- und Realisierungskomponenten in einer Kaskadenarchitektur verzahnt. Innerhalb von WIP-GD kommt auch die Technik der Antizipationsrückkopplung zum Einsatz.

Zweck der AFL. Die Graphikgenerierung durch WIP-GD ist ein iterativer Prozeß gemäß einer Entwurf-Realisierung-Evaluation-Schleife. Diese Schleife entspricht einer Antizipationsrückkopplungsschleife. Ziel der AFL in WIP-GD ist es, dazu beizutragen, daß die Graphikgenerierung auf das Benutzerprofil des Betrachters zugeschnitten ist.

²⁴Dies entspricht der Tatsache, daß ein professioneller Dolmetscher sowohl die Quellsprache als auch die Zielsprache fließend und gut spricht.

²⁵Wissensbasierte InformationsPräsentation

Partnermodellierung. Es werden Benutzerstereotypen verwendet, sowie eine regelbasierte Repräsentation von Betrachterprofilen.

Realisierung. Die Präsentationsoption in WIP ist mit einer Menge von graphischen Constraints assoziiert, die sich auf die Layoutvorlage, auf die zu verwendenden Graphik-elemente und auf die Abbildungstechniken beziehen. Die Graphikgenerierung wird als iterativer Prozeß aufgefaßt, bei dem die Design-, Realisierungs- und Evaluierungsphasen verzahnt sind. Die Umsetzung eines Präsentationsziels in eine graphische Präsentation erfolgt in mehreren Schritten in einer Entwurf-Realisierung-Evaluation-Schleife. Ausgehend von den Visualisierungszielen wird nach Ausführung von Konstruktionsoperatoren in einer Evaluationsphase überprüft, ob die mit den Konstruktionsoperatoren intendierten Effekte erreicht wurden oder ob unerwünschte Seiteneffekte entstanden sind. Falls eine Korrektur erforderlich ist, werden zur Behebung des entstandenen Mangels oder Seiteneffekts weitere Operationen angewandt oder die Designentscheidungen so revidiert, daß die unerwünschten Ergebnisse nicht mehr auftreten. Die Rückkopplungsanpassung durch die Evaluationsphase soll dazu beitragen, daß die erzeugten Instruktionen auf den individuellen Betrachter zugeschnitten sind, damit dieser zu den vom System intendierten Implikaturen veranlaßt wird.

2.6.9 PEDRO: Vorhersage der Verständlichkeit einer Instruktion

PPP²⁶ (André & Rist, 1996, André *et al.*, 1993) produziert aus einem kommunikativen Ziel und mehreren Generierungsparametern eine multimodale Präsentation, die Text, Graphik und Animationen enthalten kann. Diese Präsentation besteht aus technischen Instruktionen. Wie der Benutzer die Instruktionen verstehen wird, hängt von seinen individuellen Merkmalen ab. Um noch vor der tatsächlichen Präsentation potentielle Verständlichkeitsprobleme vorwegzunehmen, wird die Benutzermodellierungskomponente PEDRO²⁷ (van Mulken, 1996; 1997) eingesetzt.

Zweck der AFL. Um die von PPP generierte Instruktion richtig zu interpretieren, also zu dekodieren, muß der Benutzer einige Inferenzen machen. PEDRO wird zur Antizipation der potentiellen Interpretationsschwierigkeiten des Benutzers eingesetzt.

Partnermodellierung. Die Merkmale des Benutzers sowie die kausalen Zusammenhänge in der Domäne werden mit dem Formalismus der Bayesschen Netze modelliert.

Realisierung. Mit Hilfe von Bayesschen Netzen wird die Verständlichkeit der Präsentationen eingeschätzt; die Vorhersage wird durch die Interpretation von Evidenzen aus Benutzerreaktionen verfeinert. PEDRO verwendet Schwellwerte, um zu bestimmen, wann auf die intendierte Präsentation verzichtet werden muß und nach einer Alternativpräsentation zu suchen ist, bei der die bei der Antizipation festgestellten Probleme behoben werden. Dabei werden die Ergebnisse der Antizipation zur Diagnose der vermuteten Interpretationsschwierigkeiten verwendet.

²⁶Personalized Plan-Based Presenter

²⁷Präsentation Decodability pRedictOr

Unsicherheitsbehandlung. In der betrachteten Domäne resultiert die Unsicherheit daraus, daß einerseits keine sicheren Aussagen zu den Merkmalen des Benutzers gemacht werden und daß andererseits seine Reaktionen nicht immer als sichere Evidenzen interpretiert werden können. Durch die Verwendung der Bayesschen Netze wird diesen Aspekten der Unsicherheit Rechnung getragen.

Diskussion. PEDRO verwendet im Gegensatz zu den meisten Systemen, die Antizipationsrückkopplung eingesetzt haben, kein explizites Modul, das die potentiellen Reaktionen simuliert, sondern Bayessche Netze. Ein Vorteil des Einsatzes von Bayesschen Netzen in PEDRO ist ihre Verwendung zum Diagnostizieren des Verständlichkeitsproblems. Dabei basiert die Diagnose auf den bei der Antizipation aufgebauten Bayesschen Netzen. Gegebenenfalls wird eine Alternativpräsentation generiert, um die diagnostizierten Schwierigkeiten zu beheben. Mit Hilfe dieser Technik der Diagnose kann PEDRO im Laufe einer Sitzung immer präzisere Antizipationen machen: dadurch kann PPP im Laufe der Interaktion seine Präsentationen an den Benutzer adaptieren. Zur Realisierung des Ansatzes von PEDRO müssen jedoch die kausalen Zusammenhänge in der betrachteten Domäne angegeben werden.

2.6.10 Antizipation zur Verhinderung von falschen Inferenzen.

In dem von Joshi *et al.* (1984a; 1984b) entwickelten System geht es darum, Äußerungen zu vermeiden, die den Benutzer dazu verleiten könnten, falsche Schlüsse zu ziehen (vgl. auch Webber & Mays, 1983).

Zweck der AFL. Die Antizipation in diesem System dient der Generierung adäquater Antworten auf Benutzerfragen, so daß dieser nicht zu falschen Schlußfolgerungen aus der vom System gegebenen Antwort verleitet werden kann.

Partnermodellierung. Im Partnermodell werden die Überzeugungen und Erwartungen des Benutzers in einem logikbasierten Formalismus repräsentiert.

Realisierung. Die zusätzliche Information, die das System \mathcal{S} einsetzt, um den Benutzer \mathcal{U} davon abzuhalten, falsche Inferenzen zu ziehen, beschränkt sich auf den Teil der Wissensbasis, auf den das System zur Beantwortung der Benutzerfrage fokussiert. \mathcal{S} antizipiert die Inferenzen, die \mathcal{U} aus \mathcal{S} s Antwort ableiten könnte. Falls \mathcal{U} etwas schlußfolgern könnte, was falsch wäre, d.h. nicht kompatibel mit \mathcal{S} s "current beliefs" (und nicht \mathcal{S} s Gesamtwissensbasis) — d.h. aus r kann \mathcal{U} g ableiten und in den Beliefs von \mathcal{S} gilt $\neg g$ —, dann generiert \mathcal{S} eine zusätzliche Information, um dieser potentiellen Schlußfolgerung zuvorzukommen (s. Abbildung 2.5). Es werden sowohl Antworten vermieden, die \mathcal{U} dazu verleiten, Defaults anzunehmen, von denen \mathcal{S} weiß, daß sie nicht anwendbar sind, als auch Antworten, die den Erwartungen von \mathcal{U} nicht entsprechen.

Diskussion. Kern dieser Strategie ist die Modifizierung der Grice'schen Qualitätsmaxime (Wahrheitsmaxime) "Be truthful", die geändert wurde zu: "If you, the speaker, plan to say anything which may imply for the hearer something that you believe to be false, then provide further information to block it." (Joshi *et al.*, 1984a, S. 169).

- **User: Is Sam an associate professor?**
- **System: Yes, but he doesn't have tenure.**
- **User: Can I drop CIS577?**
- **System: No, but you can take an incomplete and ask for more time to finish the work.**

Abbildung 2.5: Beispiele zur Verhinderung von falschen Inferenzen.

2.6.11 WISHFUL: Antizipation von Benutzerinferenzen

Beim Prozeß der Textgenerierung berücksichtigen kompetente Sprecher und Schreiber die potentielle Auswirkung ihrer Äußerungen auf ihre Zuhörer bzw. Leser, indem sie versuchen Äußerungen zu generieren, die sowohl ihren Kommunikationszielen als auch den Anforderungen der Zuhörer gerecht werden (s. z.B. Hovy, 1988).

Die meisten Generierungssysteme gehen davon aus, daß die Hörer nur direkte Inferenzen ziehen. Die Annahmen bringen zwei Probleme mit sich: Potentielle indirekte Inferenzen aus dem Diskusbereich werden nicht behandelt, und der generierte Text gibt alle Überzeugungen des Hörers wieder, so daß der Text Teile enthalten kann, die der Benutzer leicht ableiten kann. Um diesen Problemen entgegenzuwirken, verwendet das System WISHFUL (Zukerman & McConachy, 1993a; 1993b) einen hierarchischen Planungsprozeß für die Generierung sog. *Rhetorical Devices* (RDs) für die Erfüllung bestimmter Kommunikationsziele.²⁸ Dabei wird bei der Generierung von Erklärungen aus der Algebra das Studentenmodell konsultiert, das drei Aspekte des Studenten modelliert: seine Überzeugungen und sein Wissen, die Inferenzregeln, die er wahrscheinlich anwenden wird, sowie seine Einstellungen und Fähigkeiten.

Zweck der AFL. Ziel der Antizipation der Inferenzen des Studenten ist es, algebraische Erklärungen zu generieren, so daß \mathcal{U} keine falschen Inferenzen schlußfolgert, und ihm keine Information mitgeteilt wird, die er bereits weiß oder leicht ableiten kann.

Partnermodellierung. Das Studentenmodell besteht aus drei Aspekten: (a) der Repräsentation des Wissens und der Überzeugungen, die dem Studenten vom System zugeschrieben werden, (b) der Repräsentation der potentiellen Inferenzenregeln, die der Student anwenden kann, und (c) der Repräsentation des Studentenprofils. Das System verfügt über fünf vordefinierte Profile (excellent, good, average, mediocre, bad).

Realisierung. Der Inhaltsplaner erhält als Eingabe ein zu erklärendes Konzept und eine Liste von zu berücksichtigenden Merkmalen. Zunächst werden die Informationen bestimmt, die mitgeteilt werden sollten. Die Antizipation kommt im zweiten Schritt zum Zuge. Da werden die bereits vorgeschlagenen Informationen gefiltert und ggfs. komplettiert. Falls anzunehmen ist, daß der Student einen Teil der neuen Informationen nicht verstehen wird, wird der Planer rekursiv mit diesem Teil aufgerufen.

²⁸Siehe hierzu auch die früheren Arbeiten von Zukerman (1990b, 1990a).

2.6.12 RMM: Koordination von Agenten-Handlungen

Im RMM²⁹ (Gmytrasiewicz & Durfee, 1997; Noh & Gmytrasiewicz, 1997) wird ein entscheidungstheoretischer Ansatz verwendet, damit Agenten unter Berücksichtigung des erwarteten Nutzens aller ihrer Handlungsmöglichkeiten und der Modellierung der Entscheidungsprozesse der anderen Agenten sich rational entscheiden. Unter rational wird hier die Verwendung des *Principle of Maximum Expected Utility* zur Entscheidungsfindung verstanden. Mit Hilfe von RMM wurden z.B. Szenarios betrachtet, in denen eine Koordination von Luftabwehr durch Bodengeschütze modelliert wird (Noh & Gmytrasiewicz, 1997). Die einzelnen Batterien stellen die Agenten dar und verfügen über unterschiedliche Raketen. Die Agenten haben das gemeinsame Ziel, ein bestimmtes Gebiet zu verteidigen bzw. den eventuell nach einem Angriff entstehenden Schaden zu minimieren. Dabei dürfen die einzelnen Agenten nicht miteinander kommunizieren.

Zweck der AFL. Ziel der Antizipation der Aktionen der anderen Agenten ist die Koordination der Agenten-Handlungen ohne explizite Kommunikation zwischen den Agenten zu erlauben, um ein gemeinsames Ziel zu erfüllen.

Partnermodellierung. In RMM gibt es drei Modellierungsmöglichkeiten: das *rationale* Modell, das *no-info* Modell und das *sub-intentional* Modell. Im ersteren wird der Partner gemäß dem PMEÜ modelliert. Das no-info Modell bedeutet z.B., daß ein Agent nicht weiß, wie die anderen Agenten tatsächlich sind. In diesem Fall werden die potentiellen Aktionen des Agenten als gleich wahrscheinlich betrachtet. Das no-info Modell tritt auch auf, falls ein Agent nicht weiß, wie er selbst von einem anderen modelliert wird. Das sub-intentional Modell gibt die Rationalitätsannahme auf und resultiert in eine probabilistische Einschätzung des Verhaltens des anderen Agenten.

Realisierung. Zunächst wird der andere Agent modelliert, damit seine Aktion antizipiert werden kann. Hierzu wird die Auszahlungsmatrix errechnet. Dann wird die Aktion gewählt, die dem handelnden Agenten seinen erwarteten Nutzen maximiert.

Unsicherheitsbehandlung. Es gibt mehrere Unsicherheitsquellen: Ein Agent kann unsicher über die Welt sein; er kann unsicher über sein Modell vom anderen Agenten sein (z.B.: Hat die andere Batterie noch genügend Kurzstrecken-Geschosse?); oder er kann darüber unsicher sein, wie er vom anderen modelliert wird. Die verschiedenen Handlungsmöglichkeiten werden mit Wahrscheinlichkeiten versehen, die der Unsicherheit Rechnung tragen und in den Entscheidungsprozeß einfließen.

Diskussion. Prinzipiell ist im RMM die Anzahl der Antizipationen unbeschränkt. Jedoch entsteht kein Regressionsproblem. Die rekursive Modellierung endet entweder in einem no-info Modell oder in einem sub-intentional Modell.

In RMM kann Ressourcenbeschränkungen durch ein Anytime-Verfahren Rechnung getragen werden. Hierzu wird ein *iterative deepening* Ansatz realisiert: Zunächst wird nur ein Teil der Wissensbasis verwendet, um eine (Vor)Antwort zu haben, die jederzeit als Ergebnis geliefert werden kann; dann wird mehr Wissen eingesetzt, um eine bessere Antwort zu erzielen.

²⁹Recursive Modeling Method

2.6.13 Zusammenfassung

In den letzten Abschnitten wurden die Systeme besprochen, die die Idee der Antizipationsrückkopplung eingesetzt haben. Die Tabellen 2-1 und 2-2 geben eine zusammenfassende Übersicht der diskutierten Systeme, anhand folgender Kriterien: Wozu wird die AFL eingesetzt? Wie sehen die Aktionen des Systems und die Reaktionen des Interaktionspartners aus? Wie läßt sich die realisierte AFL charakterisieren? Wie wird der Partner zum Zwecke der Antizipation modelliert? Wird die Antizipation von einem ausgewählten Modul realisiert und werden dabei die Ergebnisse früherer Antizipationsläufe wiederverwertet? Wird im System Unsicherheit berücksichtigt? Wieviele AFLs werden eingesetzt bzw. wie weit wird antizipiert?

Tabelle 2-1: Zusammenstellung von Systemen, die Antizipationsrückkopplung verwendet haben. (Teil 1)

	Zweck der AFL	Ss Züge	Us Züge	Art der Antizipation	Lokal vs. global
HAM-ANS (Jameson & Wahlster, 1982)	Generierung eindeutig interpretierbarer Ellipsen	Elliptische Äußerungen	Rekonstruktion der elliptischen Äußerung	Konzeptuelle Antizipation	Lokal
IMP (Jameson, 1989; 1983)	Generierung von Äußerungen, die eine erwünschte Eindrucksverschiebung hervorrufen und die mit dem Image von <i>S</i> verträglich sind	Aussagen über ein zu bewertendes Objekt	Pragmatische Interpretation der Systemäußerung	Konzeptuelle Antizipation	Lokal
NAOS (Novak, 1987b; 1987a)	Generierung kohärenter Beschreibungen von Objektbewegungen in einer Straßenverkehrsszene, um <i>U</i> den Aufbau einer Vorstellung der Szene zu ermöglichen.	Generierung von Beschreibungen visueller Daten	Visualisierung einer Szene	Visuell-räumliche Antizipation	Semi-global
ANTLIMA (Schirra, 1995; Blocher & Schirra, 1995; Blocher, 1994)	Generierung kommunikativ adäquater Beschreibungen, so daß der Hörer eine genaue Vorstellung eines Spielgeschehens entwickeln kann.	Verbale Beschreibung eines Spielzugs	Visualisierung des Spielgeschehens	Visuell-räumliche Antizipation	Semi-global
ZORA (Jung <i>et al.</i> , 1989; Wahlster, 1991)	Auswahl der besten Zeigegesten – Ort, Bewegung und Zeigemittel – bei der Generierung multimodaler Äußerungen	Zeigegesten	Deiktische Analyse von Zeigegesten	Visuell-räumliche Antizipation	Lokal
BILD (Neumann, 1998; 1994; Neumann & van Noord, 1994)	Generierung eindeutig interpretierbarer natürlichsprachlicher Ausdrücke und korrekte Analyse der Benutzeräußerungen.	Generierung eindeutig interpretierbarer Äußerungen bzw. Generierung von Paraphrasen zur Auswahl der Lesart	Verstehen der generierten Äußerung bzw. Auswahl einer Lesart bei Ambiguität der Benutzeräußerung	Konzeptuelle Antizipation	Lokal

wird fortgesetzt. . .

Zweck der AFL		Ss Züge	Us Züge	Art der Antizipation	Lokal vs. global
WIP-GD (Rist, 1996)	Anpassung der Graphikgenerierung auf individuellen Betrachter, damit dieser zu den von S intendierten Implikaturen veranlaßt wird.	Präsentation von Illustrationen	Implikaturen	Konzeptuelle Antizipation	Lokal
PEDRO (van Mulken, 1996; 1997)	Verständlichkeit für \mathcal{U} einer multimodalen Präsentation.	Präsentation von technischen Instruktionen	Dekodierung der Präsentation	Konzeptuelle Antizipation	Lokal
[Course Advisor] ^a (Joshi <i>et al.</i> , 1984a; 1984b)	Generierung adäquater Antworten zu Anfragen von \mathcal{U} , so daß \mathcal{U} nicht zu falschen Schlußfolgerungen aus der von S gegebenen Antwort verleitet werden kann.	Antwort bzw. Überbeantwortung einer direkten Frage von \mathcal{U} .	Ableiten von Inferenzen aus S s Antwort	Konzeptuelle Antizipation	Lokal
WISHFUL (Zukerman & McConachy, 1993a; 1993b)	Generierung benutzeradaptierter Erklärungen von Konzepten aus der Algebra, so daß \mathcal{U} keine falschen Inferenzen schlußfolgert und ihm keine Information mitgeteilt wird, die er bereits weiß oder leicht ableiten kann.	Erklärungen von algebraischen Konzepten	Ableiten von Inferenzen aus S s Erklärungen	Konzeptuelle Antizipation	Lokal
RMM (Gmytrasiewicz & Durfee, 1997; Noh & Gmytrasiewicz, 1997)	Koordination von Agenten-Handlungen ohne explizite Kommunikation zwischen den Agenten	Eine Handlung, die mit der von \mathcal{U} koordiniert, ohne daß S und \mathcal{U} kommunizieren.	Eine Handlung, die mit der von S koordiniert, ohne daß S und \mathcal{U} kommunizieren.	Konzeptuelle und visuell-räumliche Antizipation	Global
GAF ^b (Ndiaye, 1996; Ndiaye & Jameson, 1996b; 1996a)	Antizipation des nächsten Dialogschrittes des Dialogpartners durch Rollenübernahme in einem bewertungsorientierten Dialog.	Dialogzüge in Form von Fragen, Kommentaren und bewertenden Äußerungen	Dialogzüge in Form von Fragen, Kommentaren und bewertenden Äußerungen	Konzeptuelle Antizipation	Global

^a In dem zitierten Artikel angegeben war kein Systemname.^b Name des innerhalb von PRACMA implementierten Systems. Das System wird ab Kapitel 3 beschrieben.

Tabelle 2-2: Zusammenstellung von Systemen, die Antizipationsrückkopplung verwendet haben. (Teil 2)

	Partnermodellierung	Modul zur Antizipation	Wiederverwendung der Ergebnisse	Unsicherheitsbehandlung	Anzahl der AFLs	Anzahl der Ebenen	Weitere Eigenarten
HAM-ANS (Jameson & Wahli-ster, 1982)	Keine explizite Modellierung von \mathcal{U}	Bestehende Systemkomponente zur Ellipsenanalyse	Keine	Keine	1	1	Für jede mögliche Form der Elidierung wird durch Verwendung der Ellipsenrekonstruktionskomponente überprüft, ob die erzeugte Ellipse von \mathcal{S} eindeutig rekonstruierbar ist. Die kürzeste verständliche Ellipse wird generiert.
IMP (Jameson, 1983)	Das Partnermodell enthält Annahmen über die Standards und die Erwartungen des \mathcal{U} . \mathcal{S} unterhält auch ein Modell von \mathcal{U} bezüglich seiner Eindrücke über das zu bewertende Objekt sowie Annahmen darüber, wie \mathcal{U} ihn einschätzt	Verwendung der Interpretationskomponente	Keine	Ja	1	1	Verwendung eines Bewertungsformular. Für jede potentielle Äußerung zu einem Merkmal dieses Formulars wird \mathcal{U} s Bewertung dieses Merkmals berechnet.
NAOS (Novak, 1987a)	Keine explizite Modellierung von \mathcal{U} : Repräsentation des a priori Hörerwissens über die Szene, Anpassung nach jeder Beschreibung und Verwendung eines Ko-referenznetzes.	Nachschlagen im Partnermodell	Keine	Keine	1	1	Einsatz des Wissens über den Verstehensprozess des \mathcal{U} bereits bei der Generierung, damit schlechte Äußerung von vornherein vermieden werden.

wird fortgesetzt...

	Partnermodellierung	Modul zur Antizipation	Wiederverwendung der Ergebnisse	Unsicherheitsbehandlung	Anzahl der AFLs	Anzahl der Ebenen	Weitere Eigenarten
ANTLIMA (Schirra, 1995; Blocher & Schirra, 1995; Blocher, 1994)	Hörermodell; Aufbau der mentalen Bilder des Hörers mit Hilfe von Typikalitäts-Potentialfeldern.	Bestehende Komponente zur Analyse der Ereignisse	Ja	Keine	1	1	Vergleich des intendierten und imaginierten Äußerungsgehalts durch Re-Analyse.
ZORA (Jung <i>et al.</i> , 1989)	Keine explizite Modellierung von \mathcal{U} .	Verwendung von TACTILUS-II, dem Modul zur deiktischen Analyse	Keine	Keine	1	1	Testen der Ambiguität der zu generierenden Zeigegeste.
BILD (Neumann, 1998; Neumann & van Noord, 1994)	Keine explizite Modellierung von \mathcal{U}	Verwendung der eigenen Generierungskomponente bei der Analyse bzw. Einsatz des eigenen Parsers bei der Generierung	Ja	Keine	2	1	Uniforme Verarbeitung durch Verwendung von ein und derselben Grammatik sowohl bei der Analyse als auch bei der Generierung.
WIP-GD (Rist, 1996)	Regelbasierte Repräsentation von Betrachterprofilen bzw. Betrachterpräferenzen. Verwendung von Benutzerstereotypen.	Verwendung von Regeln	Keine	Keine	1	1	
PEDRO (van Mulken, 1996; 1997)	Bayessche Netze und probabilistisches Einschätzen der Inferenzen von \mathcal{U} .	Kein Modul. Simulation in Bayesianischen Netzen enthalten.	Ja, zur Diagnose	Ja	1	1	Diagnostizieren des Verständlichkeitsproblems, damit gegebenenfalls eine alternative Präsentation zur Beseitigung des diagnostizierten Problems generiert werden kann.

wird fortgesetzt...

	Partnermodellierung	Modul zur Antizipation	Wiederverwendung der Ergebnisse	Unsicherheitsbehandlung	Anzahl der AFLs	Anzahl der Ebenen	Weitere Eigenarten
[Course Advisor] (Joshi <i>et al.</i> , 1984a; 1984b)	Repräsentation der Überzeugungen und Erwartungen des Benutzers in Logik.	Kein eigenes Modul. Es werden Inferenzen gezogen.	Keine	Keine	1	1	Revision der Grice'schen Qualitätsmaxime, so daß potentiell falsche Inferenzen seitens von \mathcal{U} vermieden werden.
WISHFUL (Zukerman & McConachy, 1993a; 1993b)	Im einem Studentenmodell werden \mathcal{U} s Überzeugungen und Wissen, die Inferenzregeln, die \mathcal{U} anwenden könnte und \mathcal{U} s Fähigkeiten und Einstellungen repräsentiert.	Kein eigenes Modul. Es werden Inferenzen gezogen.	Keine	keine	1	1	Forward Reasoning zur Bestimmung von \mathcal{U} s potentiellen Inferenzen und Backward Reasoning zur Bestimmung der Erklärungsform.
RMM (Gmytrasiewicz & Durfee, 1997; Noh & Gmytrasiewicz, 1997)	Drei Möglichkeiten zur Partnermodellierung: das Rational Model, das No-Info Model und das Sub-intentional Model.	Berechnen der Auszahlungsmatrizen	Keine	Ja	variabel	∞^c	Rekursive Modellierung; Dynamische Programmierung; Berücksichtigung von Ressourcenbeschränkungen
GAF (Ndiaye, 1996; Ndiaye & Jameson, 1996b; 1996a)	Modellierung des Bewertungsprozesses bei \mathcal{U} mit Bayesschen Netzen.	Verwendung der bestehenden Komponenten. Instantiierung des Systems	Keine	Ja	variabel ^d	variabel ^d	Verwendung eines entscheidungstheoretischen Ansatzes und der Transmutierbarkeit des PRACMA-Systems. Aufruf einer kompletten Instantiierung in der Rolle des \mathcal{U} , um dessen Dialogverhalten zu simulieren. Berücksichtigung von Ressourcenbeschränkungen.

^c ∞ bedeutet, daß die entsprechende Klassifikation nicht anwendbar ist: Eine Auszahlungsmatrix entspricht der kompletten Expansion eines Spielbaumes. Daher sind in de Einträgen der Matrix alle Ebenen des Spiels bereits berücksichtigt.

^dDie Anzahl der AFLs und Antizipationsebenen ist prinzipiell variabel; jedoch ist nur eine Ebene implementiert worden.

Kapitel 3

Eine Architektur für Antizipationsrückkopplung

In diesem Kapitel werden nach einem Überblick in die Konzepte der Architektur zwei Voraussetzungen zur Realisation der globalen Antizipation im System PRACMA erörtert, nämlich die Multi-Agenten-Architektur CHANNELS und die Transmutierbarkeit.

Hierzu werden zunächst im Abschnitt 3.1 die wesentlichen Architekturmodelle präsentiert, bevor auf das Paradigma der Multi-Agenten-Architekturen eingegangen wird. Anschließend wird im Abschnitt 3.2 die Architektur CHANNELS vorgestellt, sowie das PRACMA-System mit seinen Modulen. CHANNELS wurde als verteilte Architektur für natürlichsprachliche Systeme entwickelt und erwies sich als besonders geeignet für globale Antizipation. Die Fähigkeit von PRACMA, sowohl die Rolle des Käufers als auch die des Verkäufers in einem Verkaufsgespräch, wird im Abschnitt 3.4 über Rollen-Transmutierbarkeit besprochen. Nach der Darstellung der Realisierung lokaler Antizipationsrückkopplungsschleifen in PRACMA wird im letzten Abschnitt gezeigt, wie die CHANNELS-Architektur und die Transmutierbarkeit die Realisierung globaler Antizipationsrückkopplung unterstützen können.

3.1 Architekturen von KI-Systemen

Um die in diesem Kapitel vorgeschlagene Architektur CHANNELS besser einordnen zu können, geben wir zunächst eine Übersicht über Architekturen von KI-Systemen.

In erfolgreichen KI-Systemen resultiert das intelligente Verhalten nicht nur aus der Leistung einzelner Systemkomponenten, sondern in entscheidender Weise durch die komplexe Interaktion und die Integration der einzelnen Komponenten. Die Architekturen für KI-Systeme lassen sich nach Gasser (1993) in drei Kategorien unterteilen:

- Architekturen für individuelle Agenten

Solchen intelligenten Universalsysteme basieren auf dem GPS-Ansatz. Beispiele solcher Systeme sind PRODIGY (Carbonell *et al.*, 1991), THEO (Mitchell *et al.*, 1991) und SOAR (Rosenbloom *et al.*, 1991). Sie zeichnen sich durch eine einheitliche Wis-

sensrepräsentation, eine uniforme Such- und Inferenzkomponente und eine einheitliche Systemarchitektur aus. In diesen Architekturen werden verschiedene kognitive Fähigkeiten wie z.B. Planung, Sprachverstehen, Bildverstehen, Lernen in einem Gesamtsystem vereint.

- Architekturen für verteiltes Problemlösen und Multi-Agenten-Systeme
Dieser Architektur liegt die Metapher zugrunde, daß Menschen kooperieren bzw. sich in Gruppen organisieren, um ein gemeinsames Ziel zu lösen. Die Teilgebiete *Verteiltes Problemlösen* (Distributed Problem Solving) und *Multi-Agenten-Systeme* bilden das Gebiet der *Verteilten Künstlichen Intelligenz* (Distributed AI, s. Bond & Gasser, 1988; Gasser, 1992). Bei einem Multi-Agenten-System handelt es sich um autonome Agenten, die eventuell bereits existieren und eine heterogene Struktur aufweisen. Die Agenten können kooperieren und ihre Aktionen koordinieren, um eine gemeinsame Aufgabe zu lösen. Sie können aber auch konkurrieren, da die Agenten nicht unbedingt ein gemeinsames Ziel verfolgen müssen, sondern durchaus unterschiedliche bis hin zu konträren Zielen anstreben können. Hingegen ist beim verteilten Problemlösen ein gemeinsames Ziel vorhanden, das in Teilaufgaben gleichmäßig auf die Agenten in einer Top-Down Vorgehensweise verteilt wird. Die Agenten haben eine homogene Struktur und kooperieren zur Lösung des gemeinsamen Problems.
- “Soziale” Architekturen.¹
Das sind flexible Makroarchitekturen, die sich an die Organisationsformen menschlicher Gesellschaften orientieren. Die Komponenten (Agenten) haben soziale Regeln zur Bewertung des eigenen Handelns sowie des Handelns der anderen Agenten.

Zur Definition einer Architektur für ein KI-System gehören unterschiedliche Aspekte (Wahlster, 1992; siehe auch von Hahn, 1991):

- die Dekomposition des Systems in Module,
- die Spezifikation von Kommunikationskanälen, Kommunikationsprotokollen und Interaktionssprachen zwischen den Modulen,
- die Spezifikation des Daten- und Kontrollflusses zwischen den Modulen,
- Verteilungsstrategien für Ressourcen
- Synchronisationsstrategien

Während die beiden letzten Aspekte optional sind, sind die drei ersten Aspekte in jedem KI-System notwendig. Die wesentlichen Architekturen können in Basisarchitekturen mit und ohne Steuerungskomponente sowie deren Erweiterungen durch Verschachtelung und Rückkopplung klassifiziert werden.²

¹“This type of architecture introduces several entirely new themes for intelligence, including emergent, self-organizing, and higher-order structures of knowledge and action; mechanisms for dynamic aggregation of agents and knowledge; organizations as intelligent systems, large-scale intelligent system dynamics; an the mutual construction of agents and societies” (Gasser, 1993, S. 2).

²Auf Architekturen wie etwa die *Subsumptionsarchitektur* von Brooks oder das Integrationsmodell wird hier nicht eingegangen.

3.1.1 Die Basisarchitekturen in NL-Systemen

Die **sequentielle** Architektur (auch *Phasenmodell* genannt) wie sie z.B. in XTRA (Allgayer *et al.*, 1989) realisiert wurde, zeichnet sich aus durch eine unidirektionale Hintereinanderschaltung der Systemkomponenten ohne Rückkopplung zwischen den Modulen. Durch die starre geradlinige Anordnung der Module, die nur beschränkten Informationsaustausch zwischen den Modulen erlaubt, resultiert der Nachteil dieser Architektur. (Z.B. bei der Analyse in natürlichsprachlichen Systemen können die in jedem Modul entstehenden Mehrdeutigkeiten zu kombinatorischen Effekten führen.)

Im **hierarchischen** Modell steuern ausgewählte Module andere ihnen untergeordnete Module in einer baumförmigen Struktur. Die Steuerung der gesamten Verarbeitung unterliegt einem Modul, dem "Supervisor". Beispiel eines solchen Systems ist SHRDLU (Winograd, 1972). Der Vorteil dieses Modells liegt in der potentiellen Parallelisierbarkeit, da verschiedene Zweige des Baumes simultan verarbeitet werden können. Jedoch kann durch die Steuerungskomponente ein Flaschenhals entstehen.

Das **heterarchische** Modell zeichnet sich dadurch aus, daß alle im System vorhandenen Module autonom arbeiten und miteinander kommunizieren können. Beispiel eines Systems mit einer solchen Architektur ist etwa der Wortexperten-Parser von Small und Rieger (1982).

Das **Kaskaden-Modell** besteht wie das Phasenmodell aus einer Hintereinanderschaltung von Systemkomponenten, jedoch mit bidirektionalem Kommunikationsfluß zwischen zwei aufeinanderfolgenden Modulen. Rückkopplungen über mehrere Komponenten sind nur bedingt möglich: Die Information muß durch die Komponenten weitergereicht werden, die dazwischen liegen. Diese Architektur unterstützt inkrementelle und parallele Verarbeitung. Sie wurde daher insbesondere in natürlichsprachlichen Generierungssystemen, etwa POPEL (Reithinger, 1992), realisiert.

Zu den Basisarchitekturen gehören auch die Blackboard- und Aktor-Modelle, auf die nachfolgend eingegangen wird.

3.1.2 Die Blackboard-Architekturen

Die Blackboard-Architektur (Engelmore & Morgan, 1988; Velthuisen, 1992; sowie das System HEARSAY-II³ von Erman *et al.* (1981)) besteht aus drei Hauptkomponenten: dem Blackboard (Nachrichtentafel), den Wissensquellen und dem Steuerungsmodul (s. Abbildung 3.1).⁴ Blackboard-Systeme zeichnen sich auf folgende Weise aus: Einerseits entsteht die Lösung der anstehenden Probleme inkrementell und andererseits wird das Wissen opportunistisch angewandt, abhängig von den vom Steuerungsmodul verwendeten Heuristiken.

³HEARSAY-II ist ein natürlichsprachliches System, das die Eingabe in Form von gesprochenen Sätzen versteht und als Anfrage an eine Datenbasis interpretiert.

⁴Man bemerke die Analogie zu Produktionssystemen, die als rudimentäre Blackboard-Systeme betrachtet werden können. Produktionssysteme bestehen aus drei Komponenten: der Datenbasis, der die Produktionsregeln enthaltenden Regelbasis und dem Regelinterpretier. Diese drei Komponenten entsprechen in der Reihenfolge dem Blackboard, den Wissensquellen und der Steuerungskomponente.

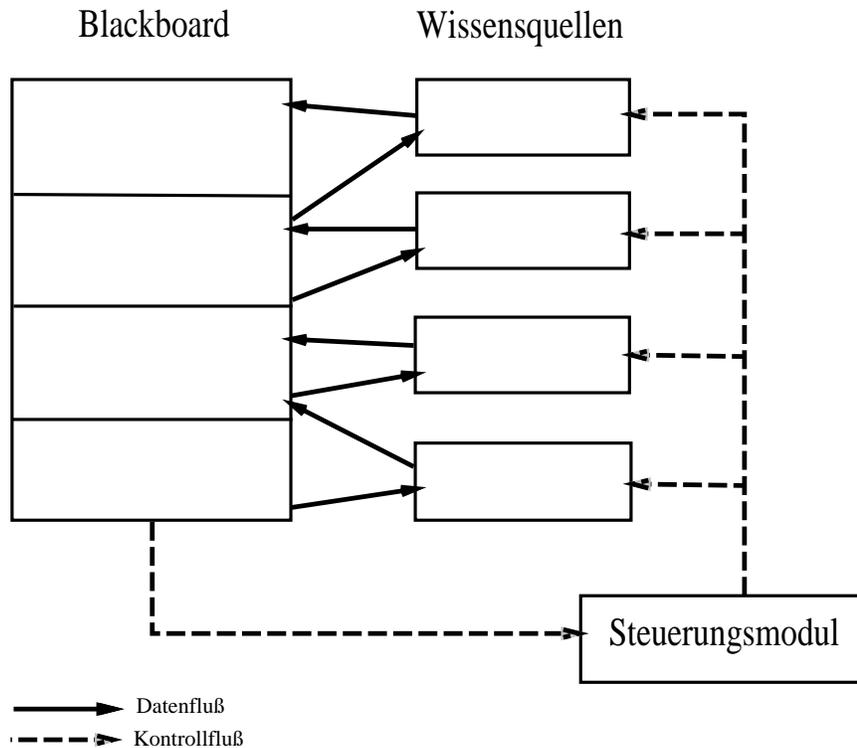


Abbildung 3.1: Die Blackboard-Architektur.

In den Wissensquellen ist das zur Problemlösung nötige Anwendungswissen repräsentiert. Sie arbeiten unabhängig und kommunizieren nicht direkt, sondern nur indirekt miteinander durch das gemeinsame Blackboard und steuern somit Informationen bei, die zur Lösungsfindung notwendig sind. Jede Wissensquelle hat Aktivierungsbedingungen und ein Repertoire an Aktionen, mit deren Ausführen sie zur Lösungsfindung beitragen kann. Das Blackboard ist eine globale strukturierte Datenbasis, in der die Teillösungen während des Lösungsprozesses abgelegt werden. Die Wissensquellen können lesend bzw. schreibend auf die abgelegten Daten zugreifen. Das Blackboard kann in mehrere Ebenen strukturiert werden — Man spricht von horizontaler Unterteilung. Diese Blackboard-Ebenen entsprechen den verschiedenen Abstraktionsebenen. Beispielsweise ist bei der Analyse natürlichsprachlicher Äußerungen eine mögliche Dekomposition die Unterteilung in folgenden Ebenen: Morphologie, Syntax, Semantik und Pragmatik. Zu jeder Ebene gehört eine Beschreibung der Daten, die dort zu schreiben bzw. zu finden sind. Jede Ebene kann wiederum intern unterteilt werden, um etwa eine Zeitachse einzuführen — Man spricht von vertikaler Unterteilung.

Das Steuerungsmodul ist für die Aktivierung der verschiedenen Wissensquellen zuständig. Die Komponente beobachtet die Veränderungen im Blackboard. Die meisten Blackboard-Systeme werden ereignisgetrieben gesteuert, d.h. jede Änderung im Blackboard stellt ein Ereignis dar, das eine oder mehrere Wissensquellen aktiviert.

Die Kommunikation im Blackboard-Modell erfolgt asynchron durch einen gemeinsa-

men Datenbereich (*data sharing*), auf den jede Wissensquelle jederzeit lesen und schreiben darf. Einige Arbeiten integrieren Elemente der verteilten und parallelen Verarbeitung in das Blackboard-Modell.

3.1.3 Die Aktoren-Architektur

Die Aktoren-Architektur verwendet die Metapher des objekt-orientierten Verarbeitungsmodells, bei dem die einzelnen Systemkomponenten durch den wechselseitigen Versand von Nachrichten kommunizieren, um gemeinsam an der Lösung der entstehenden Aufgaben zu arbeiten.

Stellvertretend für diese nachrichtenbasierten Ansätze wird nun das Aktorenmodell präsentiert. Beim Aktorenmodell (Agha, 1991; 1986) handelt es sich um autonome potentiell nebenläufige Prozesse, die Aufgaben delegieren oder in Kooperation mit anderen lösen können. Die Kommunikation und Steuerung der Aktoren erfolgt *ausschließlich* über das Versenden und Empfangen von Nachrichten (engl. *message passing*) zwischen den Aktoren. Ein Aktor ist ein nebenläufig arbeitendes Objekt mit wechselnden internen Zuständen, das durch folgende Eigenheiten beschrieben wird:

- seine Bekanntschaften (engl. *Acquaintances*): Liste der dem Aktor bekannten Aktoren, an die er Nachrichten adressieren kann.
- ein Skript, das das Verhalten des Aktors beim Eintreffen einer Nachricht definiert. Das Skript besteht aus Methoden, die auf gewisse Muster der Nachrichten reagieren. Diese Methoden können neue Aktoren erzeugen, Nachrichten senden und das Verhalten des Aktors modifizieren.
- einen "Briefkasten": Das ist ein Puffer, in dem einlaufende Nachrichten gesammelt werden, bis der Aktor sie bearbeitet und die Resultate an andere Aktoren sendet.
- ein *Proxy*: Wenn ein Aktor eine Nachricht erhält, die er nicht versteht bzw. nicht bearbeiten kann, gibt er diese an seinen Proxy weiter, der sich um die Verarbeitung der Nachricht kümmert.⁵

Aktoren arbeiten *nebenläufig*. Ihre Kommunikation ist asynchron: ein sendender Aktor muß nicht den internen Zustand des Empfängers kennen und muß nicht warten, bis der Empfänger Kommunikationsbereitschaft signalisiert. Im Aktorenmodell findet kein *Broadcasting* statt: Ein Aktor kann nur an ihm bekannte Aktoren Nachrichten über eine Punkt-zu-Punkt-Verbindung senden.

In der natürlichsprachlichen Verarbeitung wurde das Aktorenmodell z.B. beim verteilten Parsing natürlicher Sprache verwendet (Hahn, 1993). Durch sog. Wortaktoren auf der Grundlage eines linguistisch validen Protokolls werden natürlichsprachliche Eingaben analysiert.

⁵Mit dieser Technik, die auch *Delegation* genannt wird, kann beispielsweise Vererbung simuliert werden (Stoyan, 1988, S. 224).

3.1.4 Hybride Architekturen

Die Basis-Architekturen lassen sich durch Verschachtelung oder Rückkopplung zwischen den einzelnen Komponenten zu hybriden Architekturen kombinieren. Dabei werden verschiedene Modelle zu größeren komplexeren Architekturen zusammengefaßt. So wurden z.B. im XTRA-System in eine sequentielle Architektur ein Generator mit Kaskadenarchitektur eingebunden. (Siehe auch die späteren Ausführungen zur Verschachtelung heterogener Architekturen in den Abschnitten 3.2.4 und 3.2.5.)

3.2 CHANNELS: Eine Multi-Agenten-Architektur

In diesem Abschnitt wird die innerhalb von PRACMA entwickelte Multi-Agenten-Architektur CHANNELS präsentiert. Multi-Agenten-Systeme benutzen die Metapher, daß mehrere spezialisierte autonome Module kommunizieren und kooperieren, um eine gemeinsame Aufgabe zu lösen, die keines der Module allein lösen kann. Bevor auf die einzelnen Aspekte der Architektur eingegangen wird, wird der Begriff "Agent" erläutert.

3.2.1 Der Agenten-Begriff

Es gibt in der Literatur keine einheitliche Definition des Begriffs "Agent". Man findet in der englischsprachigen Literatur auch andere Begriffe wie "actor", "node", "problem solver", "softbots", etc. Es handelt sich eher um die Sichtweise gemäß dem Paradigma sozialer Organisationen wie beispielsweise in menschlichen Gesellschaften oder etwa in einem Bienenschwarm. Da es keine Standardisierung in Hinblick auf die Inferenz- und Kommunikationsfähigkeiten gibt, variieren Agenten in ihrer Granularität und Komplexität.⁶

Cutkosky *et al.* (1993, S. 31) definieren einen Agenten als "a computer program that communicates with external programs exclusively via a predefined protocol. An agent is capable of responding to all message defined by the protocol, and uses the protocol to invoke the services of other agents.". Wooldridge und Jennings (1994) definieren die schwache Spezifikation des Begriffs "Agent" (weak notion of agency) durch folgende Eigenschaften: Die Agenten sollen autonom sein, sozial (sie kommunizieren untereinander gemäß einem Protokoll), reaktiv (sie reagieren auf die Veränderungen in ihrer Umgebung) und proaktiv (sie können auch die Initiative ergreifen). Weitere optionale Eigenschaften können sogar die Modellierung von mentalen und emotionalen Zuständen sowie Unkooperativität, Beschränkung des Rationalitätsbegriffes oder Lernfähigkeiten umfassen (siehe auch die Agentenklassifikation nach Fischer (1993) in fünf ineinander verschachtelte Kategorien: primitive, technische, technisch-intelligente, kognitive und soziale Agenten.).

In der Literatur werden die Begriffe "Agent" wie auch "Akteur" mit unterschiedlichen Bedeutungen verwendet. Im Kontext von PRACMA steht "Agent" für ein aktives System-

⁶Für Ansätze zur Standardisierung des Agentenbegriffes sei auf die Forschungsarbeiten zur Wissensrepräsentation KIF (**K**nowledge **I**nterchange **F**ormalism) (Genesereth & Fikes, 1992) und zur Agentenkommunikation KQML (**K**nowledge **Q**uery and **M**anipulation **L**anguage) (Finin *et al.*, 1995) verwiesen.

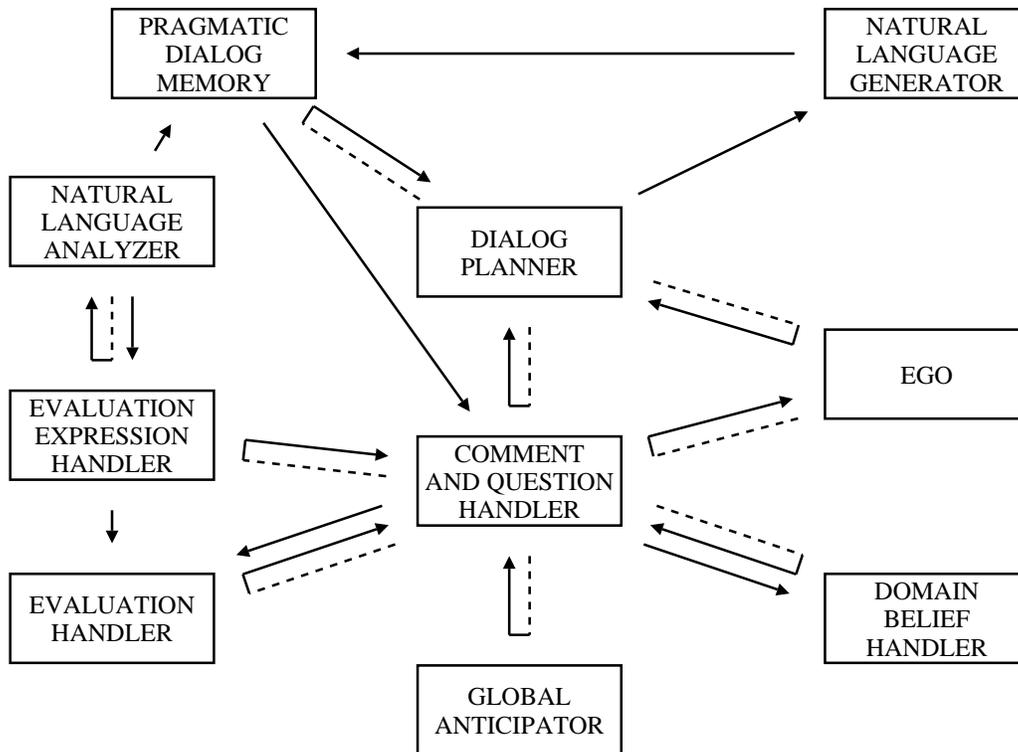


Abbildung 3.2: Die Architektur des PRACMA-Systems als Beispiel für die Kommunikation in CHANNELS.

(Ein einfacher Pfeil stellt eine INFORM-Nachricht dar, die unaufgefordert Informationen von einem Agenten zu einem anderen übergibt. Eine gestrichelte Linie gefolgt von einem Pfeil stellt einen Kommunikationsakt vom Typ ASK gefolgt von REPLY dar.)

modul, während “Akteur” einen Dialogteilnehmer (z.B. einen Verkäufer) bezeichnet, dessen Rolle von einem Menschen oder von PRACMA übernommen werden kann.

3.2.2 Agenten und Nachrichten

Die Dialogverarbeitung in PRACMA erfordert eine Systemarchitektur, die eine flexible Interaktion zwischen den einzelnen Systemkomponenten mit ihren unterschiedlichen Repräsentations- und Verarbeitungsformalismen unterstützt. Die für PRACMA entwickelte Multi-Agenten-Architektur CHANNELS (**C**ooperating **H**eterogeneous **A**gents for a **N**atural **L**anguage **S**ystem) verwendet Techniken der verteilten KI und der objektorientierten Programmierung (Ndiaye & Jameson, 1994). Die einzelnen Systemmodule werden als autonome, spezialisierte Problemlöser — Agenten — realisiert, die ausschließlich durch Nachrichtenaustausch kommunizieren und kooperieren. Diese Kommunikation wird durch ein *kommunikationsakt-basiertes Protokoll* gesteuert (s. Abbildungen 3.2 und 3.3).

```

<agent> ::= <standard-agent>
<standard-agent> ::= (<name> <acquaintances> <status> <job queue> <replies> ...)
<acquaintances> ::= (<agent>*)
<job queue> ::= (<job>*)
<job> ::= (<agent> <communicative act> <content>)
<reply> ::= (<job> <content>)
<status> ::= idle | active | waiting
<message> ::= (<sender> <recipients> <communicative act> <communication form>
<content>)
<sender> ::= agent
<recipients> ::= (<agent>+) | <class of agents>
<communicative act> ::= inform | ask | reply
<communication form> ::= synchronous | asynchronous
<content> ::= <LISP expression>

```

Abbildung 3.3: Auszug aus der CHANNELS-Spezifikation in BNF.

3.2.2.1 Struktur der Agenten

Ein CHANNELS-Agent wird durch folgende Eigenschaften definiert:

- Bekanntschaften (Acquaintances)

Das ist die Liste der anderen Agenten, die dem Agenten bekannt sind. Nur die Mitglieder der Liste dürfen dem Agenten Nachrichten senden. Die bekannten Agenten haben Zugriff auf den "Briefkasten" des Agenten, wo sie Nachrichten hinterlassen dürfen.
- Status

Der Status *idle*, *active* oder *waiting* definiert den internen Zustand des Agenten. Eine Funktion bestimmt den nächsten Zustand des Agenten in Abhängigkeit von dem vorigen Zustand und der Eigenschaft der erhaltenen oder versandten Nachricht.
- Wissen/Fähigkeiten

Das sind diejenigen Prozeduren bzw. Funktionen, die das Verhalten des Agenten definieren. Welche Funktion angewandt wird, hängt von dem Inhalt der empfangenen Nachricht ab.
- Puffer (Job queue)

Hier werden Nachrichten abgespeichert, bis sie behandelt werden können. Sobald sie komplett bearbeitet wurden, werden sie gelöscht.
- Historie

Hier wird Buch über die Verarbeitung bzw. über die empfangenen und gesendeten

Nachrichten geführt

- Wirtsrechner (Host)
Die Agenten bzw. die dazugehörigen Prozesse sind verteilbar auf mehrere Rechner. Hier wird angegeben, auf welchem Rechner im lokalen Netz der Agent bzw. der induzierte Prozeß angesiedelt ist.
- Selbstpräsentationsmöglichkeiten:
Die Agenten sind so konzipiert, daß sie ihren Verarbeitungszustand präsentieren können. Dazu gehören der momentane interne Zustand, die Ergebnisse und die Kommunikationsverbindungen zu anderen Agenten. Sowohl zum Verstehen der Funktionsweise des Systems als auch zur Fehlersuche ist diese Eigenschaft besonders nützlich.

3.2.2.2 Struktur der Nachrichten

Die Agenten kommunizieren ausschließlich durch Nachrichten, die durch folgende Eigenschaften charakterisiert werden:

- Sender
Der Absender ist der Agent, der die Dienste eines oder mehrerer anderer Agenten braucht und Nachrichten zu diesen initiiert. An ihn werden die Ergebnisse der Verarbeitung durch die Empfänger der Nachrichten zurückgeschickt.
- Empfänger
Das ist die nichtleere Liste der Agenten, an die Nachrichten geschickt werden sollen. Diese Agenten müssen dem Absender bekannt sein.
- Kommunikationsakt
Die Kommunikationsakte bestimmen die Art des Kommunikationsprotokolls zwischen den Agenten. In CHANNELS wurden folgende implementiert:
 - INFORM
Mit dem Kommunikationsakt INFORM werden unaufgefordert Informationen von einem Agenten zu einem anderen oder mehreren mitgeteilt.
 - ASK
Mit ASK fordert ein Agent andere auf, den übergebenen Inhalt der Nachricht zu verarbeiten und die Ergebnisse an den fragenden Agenten zurückzuschicken. Der fragende Agent kann dann seine Verarbeitung solange zurückstellen, bis er eine Antwortnachricht (REPLY) erhält oder mit der Verarbeitung fortfahren und das Ergebnis zu einem späteren Zeitpunkt empfangen und verarbeiten kann.
 - REPLY
Eine REPLY-Nachricht folgt bei erfolgreicher Verarbeitung einem ASK. Der Inhalt ist das Ergebnis der Anfrage, das als Antwort an den Agenten zurückgeschickt wird, der ein ASK initiiert hatte.
 - SELF-PRESENT
Streng genommen ist SELF-PRESENT kein Kommunikationsakt. Hierdurch wird ein Agent dazu veranlaßt, den Stand seiner internen Verarbeitung sowie seine Ergebnisse graphisch darzustellen.

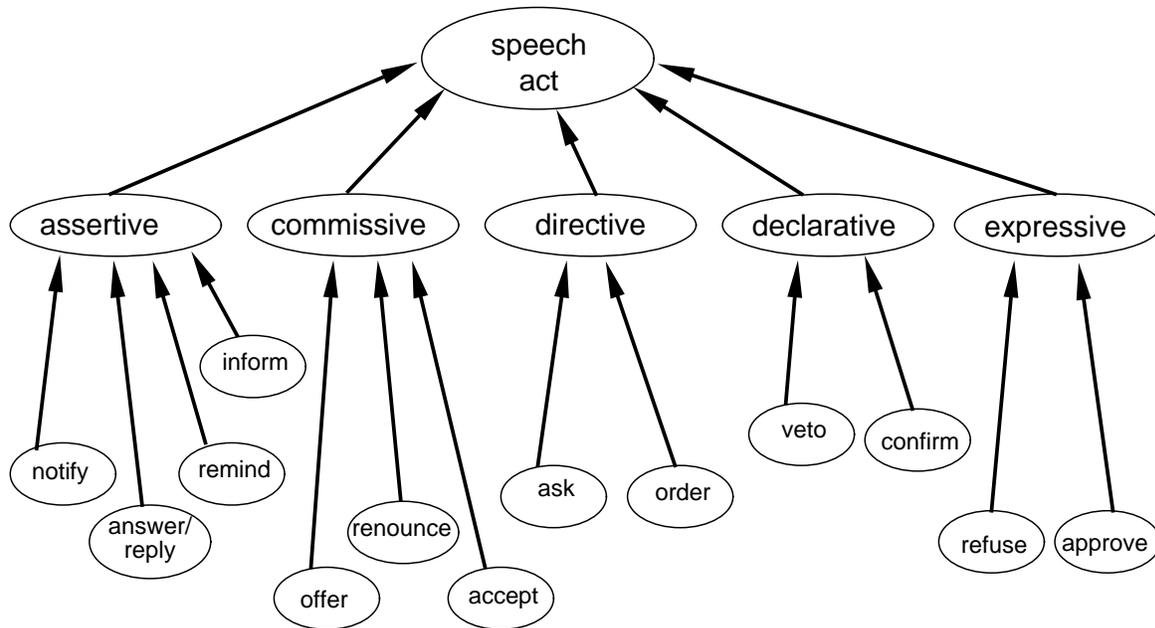


Abbildung 3.4: Klassifikation von Sprechakten nach Vanderveken (1990).

(Es wurden nur einige Sprechakte als Beispiele angeführt.)

Inspiziert wurden diese Kommunikationsakte durch die Arbeiten aus der Sprechakttheorie (Vanderveken, 1990; Searle, 1983). Sprache dient dem interpersonalen Handeln und sprachliche Äußerungen haben eine kommunikative Funktion, die durch die illokutionäre Kraft des Äußerungsinhalts gegeben ist. Mit unseren Äußerungen rufen wir beim Adressaten gewisse Wirkungen bzw. Gefühle, Stimmungen, etc. hervor. Beispiele sind Behauptungen (z.B. “Es regnet.”), Fragen (z.B. “Kannst Du mir den Bericht besorgen?”), Befehle (z.B. “Schließen Sie die Tür!”), Versprechungen (z.B. “Ich werde morgen pünktlich kommen.”). Die illokutionäre Kraft einer Äußerung liegt in der spezifischen Handlung, die dadurch induziert wird. Beispielsweise will man mit der Äußerung “Es zieht.” erreichen, daß der Gesprächspartner das Fenster schließt. Sprechakte werden nach Vanderveken (1990) in fünf Klassen klassifiziert: Assertiva, Kommissiva, Direktiva, Deklarativa und Expressiva (s. Abbildung 3.4).

INFORM, ASK und REPLY sind Basiskommunikationsakte, mit deren Hilfe weitere Kommunikationsakte definiert werden können. (Siehe auch die Unterteilung in KQML in sog. Core Performatives — Nachrichten-Typen —, die benutzt werden, um weitere Performatives zu definieren). Beispielsweise könnte REMIND wie folgt modelliert werden: ein Agent *A* sendet ein ASK an einen Agent *B*, der nach einer Weile immer noch kein REPLY als Antwort geschickt hat; dann sendet *A* ein INFORM mit dem gleichen Inhalt wie das vorangegangene ASK evtl. mit einigen der ersten Nachricht spezifischen Informationen. Der Typ CANCEL wäre nach dem selben Schema wie REMIND modellierbar. Der Typ REQUEST gleiche ASK jedoch mit der Möglichkeit für den Adressaten das Durchführen der Anfra-

ge abzulehnen oder zu akzeptieren. Durch Definition von Autoritätsbeziehungen zwischen den Agenten kann ORDER mit Hilfe der Basiskommunikationsakte definiert werden; und PROPOSE wäre wie INFORM, gefolgt von einer Nachricht, deren Inhalt angibt, ob der Vorschlag akzeptiert, abgelehnt oder modifiziert wurde.

- Kommunikationsmodus

Die Nachrichten zwischen den Agenten werden in CHANNELS synchron oder asynchron übertragen. Der Kommunikationsmodus hat Auswirkungen auf die Steuerung der Verarbeitung der Agenten.

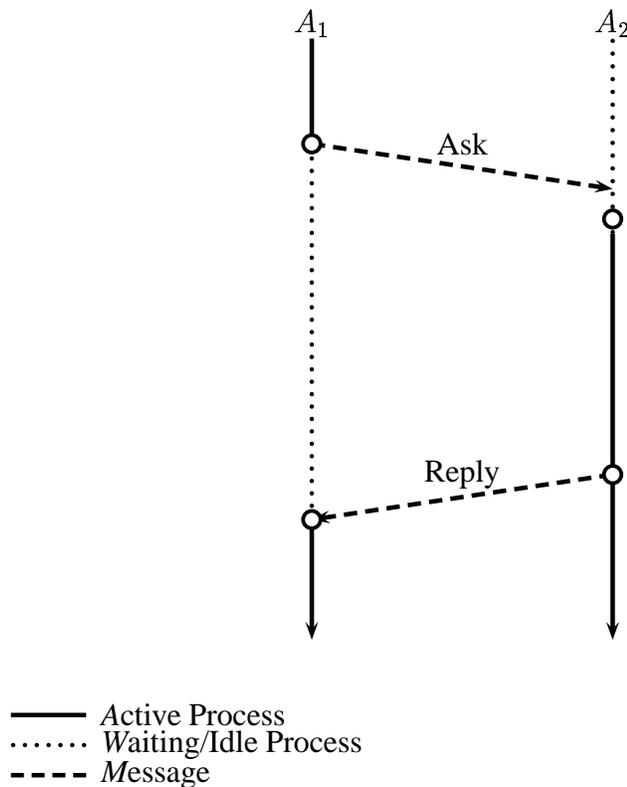


Abbildung 3.5: Synchroner Nachrichtenaustausch.

- Eine Kommunikation von A_1 nach A_2 heißt *synchron*, wenn A_1 nach dem Abschicken der Nachricht in einen wartenden Zustand übergeht, seine Berechnungen suspendiert und solange wartet, bis er wieder durch die Antwort von A_2 auf die Nachricht reaktiviert wird (s. Abbildung 3.5).⁷ Dann geht A_1 in einen aktiven

⁷Die hier verwendete Definition weicht ab von der Verwendung des Begriffs "synchron" etwa in Bereich "Verteilte Betriebssysteme" (Tanenbaum, 1995, S. 82-85): Dort wird "synchron" lediglich definiert, als die Bereitschaft des empfangenden Prozesses, die Nachricht anzunehmen. Man spricht in diesem Zusammenhang auch von blockierenden Primitiven: Während der Nachrichtenübertragung ist der sendende Prozeß blockiert. Der Prozeß rechnet erst weiter, wenn er vom Empfänger eine Antwort-Nachricht bekommt. Hingegen sind

Zustand über und setzt seine Berechnungen fort. Diese Definition der synchronen Kommunikation wird auch in der Programmiersprache ORIENT84/K verwendet (Ishikawa & Tokoro, 1987). Stoyan (1988, S. 194) bezeichnet diese Kommunikationsart als *Synchronisierung beim Sender*, da, wenn ein Objekt während seiner Arbeit ein anderes zur Ausführung einer Operation aktivieren muß, es auf das Ergebnis zu warten hat.

- Beim *asynchronen* Kommunikationsmodus geht es um eine Nachricht mit einem Auftrag, dessen Ergebnis nicht unmittelbar vorliegen muß. A_1 sendet eine Nachricht an A_2 , merkt sich den dazugehörigen Job (Auftrag) und setzt seine Berechnung fort, um z.B. Anfragen anderer Agenten zu beantworten. Wenn A_2 zu einem späteren Zeitpunkt die Antwort auf die Nachricht zurückschickt, setzt A_1 die Bearbeitung an diesem Teil fort (s. Abbildung 3.6).

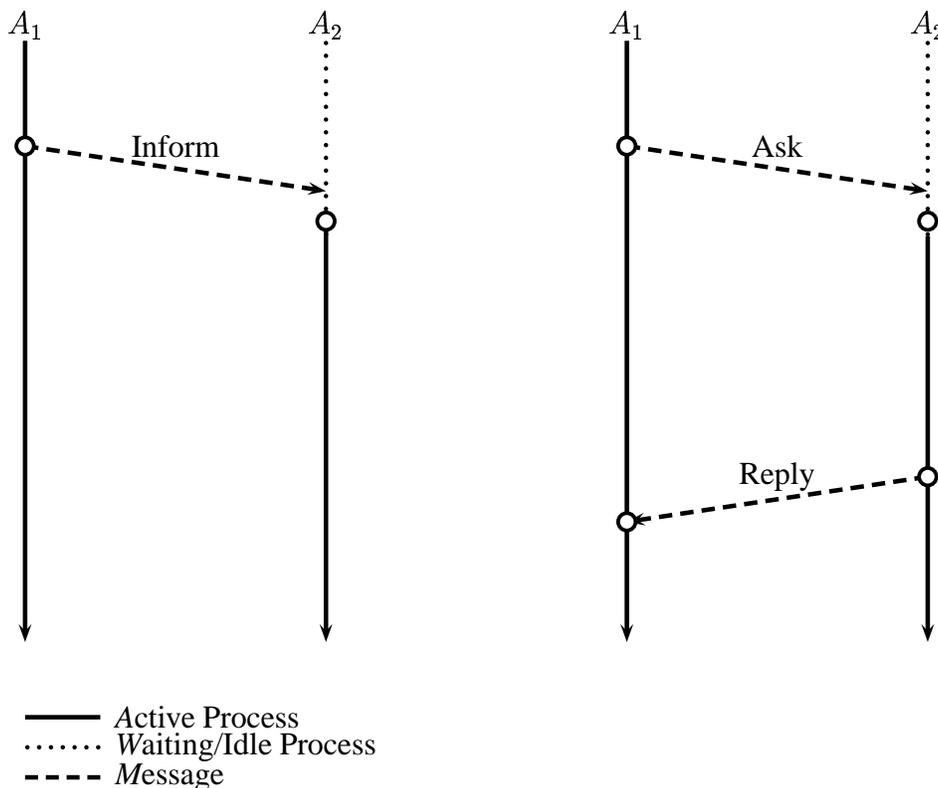


Abbildung 3.6: Asynchroner Nachrichtenaustausch.

asynchrone Primitive nicht-blockierend: Die Steuerung bleibt beim sendenden Prozeß. Wenn der Empfänger nicht bereit ist, die Nachricht zu empfangen, wird sie gepuffert, bis sie angenommen werden kann.

Agha (1986, S. 136) definiert im Akteurenmodell synchrone Kommunikation als “communication between two actors requiring the sender to wait until the recipient acknowledges or otherwise responds to the communication before continuing with further processing.”

- **Inhalt**

Hier wird spezifiziert, was den anderen Agenten in der Nachricht mitgeteilt wird. Aufgrund dieses Inhalts weiß der Adressat einer Nachricht, welche Aktionen unternommen werden sollen.

Nachrichten erzeugen sog. *jobs*, die neben einem eindeutigen Bezeichner durch folgende Slots charakterisiert werden: Absender der Nachricht, Typ der Nachricht und Inhalt.

CHANNELS ist in der Programmiersprache COMMONLISP (Steele, Jr., 1990) mit dem *Common Lisp Object System* (CLOS) implementiert. Die Agenten, die Nachrichten sowie die Jobs werden als Klassen implementiert. Hierzu werden die von COMMONLISP bereitgestellten Multi-Tasking-Facilities (LUCID, 1992) verwendet. Zu jedem Agenten gehört ein Prozeß.

Die Agenten kooperieren durch Nachrichtenaustausch, wenn ein Agent Dienste eines anderen braucht. Hierbei wird davon ausgegangen, daß die Agenten alle ihnen geschickten Anfragen beantworten.⁸ Diese Art von Kooperation ist explizit, indem die einzelnen Agenten mit denjenigen Nachrichten austauschen, von denen sie Dienste haben wollen.⁹ Abbildung 3.7 gibt ein Beispiel von Nachrichtenaustausch zwischen zwei Modulen des PRACMA-Systems.

3.2.3 Externe Agenten

Wie in der Spezifikation in Abschnitt 3.2.2.1 gesagt wurde, gehört zu jedem Agenten die Angabe darüber, auf welchem Rechner bzw. in welchem LISP-Image sein zugehöriger Prozeß abläuft. Die Agenten in der CHANNELS-Architektur sind so konzipiert, daß ihre Prozesse nicht alle auf dem selben Rechner ablaufen müssen. Beispielsweise können bei der Instantiierung des PRACMA-Systems einige Agenten auf zwei Rechner bzw. auf zwei

⁸In der realisierten Implementierung werden keine Konflikte zwischen den Agenten betrachtet. Diese könnten z.B. dann entstehen, wenn ein Agent *A* eine bestimmte Information von Agent *B* benötigt, der die Anfrage aber derzeit nicht beantworten kann; oder auch, wenn mehrere Agenten eine definierte Teilaufgabe übernehmen können (z.B., wenn das System über drei verschiedene Aktionsplaner verfügt). In diesen Fällen verhandeln die Agenten miteinander z.B. mittels Contract Net (Davis & Smith, 1983), um festzulegen, wer letztendlich den Auftrag erhält oder wie die Anfrage von *A* bearbeitet werden kann. Hierbei ist zu beachten: Aus Sicht der Systemarchitektur sind diese Verhandlungen (negotiation) kooperative Prozesse zur Konfliktlösung und somit nicht zu verwechseln mit den Verhandlungen, die zwischen dem potentiellen Käufer und dem Verkäufer stattfinden. Beim Contract Net informiert ein Agent (der Manager) die anderen über die zu lösenden Teilaufgabe. Agenten, die potentiell zur Lösung dieser Teilaufgabe beitragen können, sog. Bidders, antworten mit Angeboten (Bids), die vom Manager bewertet werden. Nach der Bewertung wählt der Manager das beste Angebot aus und vergibt den Auftrag an den/die anbietenden Agenten (Contractors), der/die sich zur Bearbeitung der Teilaufgabe verpflichtet/verpflichten und über den Fortgang bzw. Ergebnisse der Bearbeitung an den Auftraggeber berichtet/berichten. Die jeweiligen Rollen als Manager, Contractor oder Bidder werden dynamisch angenommen. So kann ein Agent in verschiedenen Contract Nets verschiedene Rollen als Contractor und Manager gleichzeitig einnehmen.

⁹Prinzipiell ist, wie Genesereth *et al.* (1986) gezeigt haben, Kooperation auch ohne Kommunikation möglich. Wenn die betrachtete Welt vollständig modelliert ist, ist Kooperation ohne Informationsaustausch möglich und wird zu einer lokalen Angelegenheit.

```

(send-message *dialog-planner* '(*cqh*) 'ask 'asynchronous
  (:GIVE-APPROPRIATE-REACTION
    ((:CLASSIFICATIONS (STATEMENT (FOUR_DOORS DOORS SPACE COMFORT SPORTINESS))
      (:CONTENT NIL)
      (:SURFACE-NLL (THE ?X1 (SIZE (INST ?X1))
        (HAS_ATTRIBUTE ((AGENT ?X1) (THEME BIG))))))
    (:DEEP-NLL (THE ?X1 (SIZE (INST ?X1))
      (HAS_ATTRIBUTE ((AGENT ?X1) (THEME BIG))))))
    (:SPEAKER SELLER)
    (:UTTERANCE-STRING "It has four doors.")
    (:THRESHOLD 1))))

(send-message *cqh* '(*dialog-planner*) 'reply 'asynchronous
  '(JOB123 ((:CLASSIFICATIONS (VERBAL-REACTION (FOUR_DOORS DOORS SPACE COMFORT SPORTINESS))
    (:CONTENT :POSITIVE)
    (:SURFACE-NLL (:POSITIVE ((AGENT SPEAKER) (PROP ?X))))
    (:DEEP-NLL (:POSITIVE ((AGENT SPEAKER) (PROP ?X))))
    (:SPEAKER BUYER)
    (:UTTERANCE-STRING "That's good.")
    (:THRESHOLD 5))))

```

Abbildung 3.7: Beispiele von Nachrichtenaustausch zwischen den Modulen DIALOG PLANNER und COMMENT AND QUESTION HANDLER (CQH), dem Modul zur Äußerungsauswahl in PRACMA.

LISP-Images innerhalb eines Rechners verteilt werden.¹⁰ Falls ein Agent in einem anderen LISP-Image (auf dem selben oder auf einem anderen Rechner) angesiedelt ist als in der betrachteten Instantiierung, handelt es sich um einen *externen Agenten*. Ein externer Agent ist ein virtueller Agent innerhalb eines Lisp-Image mit dem die anderen Systemmodule der Instantiierung durch Nachrichten kommunizieren können, der jedoch in einem separaten Lisp-Image angesiedelt ist. Das Lisp-Image kann auf demselben oder auf einem anderen Rechner ablaufen.

Man beachte hierzu folgendes: Eine Instantiierung bezieht sich immer auf eine Initialisierung des PRACMA-Systems mit der Rolle des Käufers bzw. des Verkäufers. Eine Instantiierung kann selbst auf zwei LISP-Images verteilt werden. Wir reden von Hauptinstantiierung, wenn das PRACMA-System mit einer Rolle initialisiert wird und der Benutzer die duale Rolle übernimmt; wir sprechen von Unterinstantiierung, wenn PRACMA die Rolle des Dialogpartners simuliert und hierzu die einzelnen Komponenten mit der Rolle des Gesprächspartners initialisiert. Diese externen Agenten sind die Voraussetzung zur Realisierung globaler Antizipationsrückkopplung in PRACMA, da sie die Verbindung zwischen der Hauptinstantiierung und der Simulation der Rolle des Dialogpartners in der Unterinstantiierung realisieren.

Die Interprozeßkommunikation sowie die Kommunikation zwischen den Instantiierungen wird durch ICE¹¹ (Amtrup, 1997; 1994) unterstützt. ICE basiert auf der Kommunikati-

¹⁰In der jetzigen Implementation können die Instantiierungen auf zwei Rechner verteilt werden. Die Agenten sind jedoch leicht erweiterbar, so daß sie innerhalb eines lokalen Netzes auf mehr als zwei Rechnern ablaufen können.

¹¹Intarc (INteractive Architecture) Communication Environment

onssoftware PVM¹² (Geist *et al.*, 1994), wodurch ein Netzwerk heterogener UNIX-Rechner virtuell zu einem einzelnen großen Parallelrechner gemacht werden kann. Während ICE die Schnittstelle zwischen den Komponenten der verteilten Anwendung bildet, übernimmt PVM die physikalische Kommunikation. ICE regelt die Interprozeßkommunikation durch Bereitstellung einer Schnittstelle zu verschiedenen Programmiersprachen¹³ und mit Hilfe eines sog. Intarc License Servers (ILS). Der ILS ist eine Art Meldezentrale, bei der sich alle Komponenten anmelden müssen. Der ILS hat Informationen über alle am System beteiligten Komponenten und kann z.B. Anfragen nach Adressen von Komponenten beantworten. Nach der Anmeldung beim ILS wird den beiden beteiligten Komponenten ein Basiskanal zugeteilt. Dieser Basiskanal ist eine Standard-Kommunikationsschnittstelle, die eine bidirektionale Punkt-zu-Punkt-Verbindung zwischen den beiden Komponenten realisiert. ICE bietet Kommunikationsfunktionen zur Initialisierung des ILS, zur Registrierung und Abmeldung einer Komponente beim ILS, zum Aufbau und Schließen des Basiskanal einer Komponente, zum Senden und Empfangen von Nachrichten sowie zum Aufbau und Schließen zusätzlicher Kommunikationskanäle an.

In dem ICE-Modell werden die Komponenten namentlich identifiziert, und es besteht eine bidirektionale Datenverbindungen zwischen zwei Komponenten.

3.2.4 Verschachtelung heterogener Architekturen

Die CHANNELS- Agenten können in ihre Granularität und Komplexität von einfachen Modulen (z.B. EGO) bis hin zu komplexeren Architekturen (z.B. das Modul zur natürlichsprachlichen Analyse) variieren. Eine flexible Architektur muß dieser Anforderung gerecht werden: "Flexible implementation support for DAI systems must provide ways of integrating heterogeneous problem-solvers of different granularity" (Gasser & Briot, 1992, S. 94). Daher können in einem System die einzelnen Module unterschiedliche lokale Architekturen aufweisen. Beispielsweise ist das Modul zur natürlichsprachlichen Analyse (NATURAL LANGUAGE ANALYZER) als Blackboard realisiert, und das Modul zur Generierung natürlichsprachlicher Ausgaben (VM-GEN) bedient sich einer Kaskadenarchitektur.

Die Verschachtelung heterogener Architekturen erlaubt die Wiederverwendung von bereits definierten Modulen (cf. auch Wittig, 1992). Hierzu werden die Agenten um eine Ebene erweitert, die die Kommunikation und Kooperation mit den anderen Agenten innerhalb des Systems definiert. Das ARCHON (Wittig, 1992) unterstützt z.B. eine solche Vorgehensweise.

3.2.5 Nachrichtenaustausch vs. Blackboard

Bezüglich der Kommunikation innerhalb eines Multi-Agenten-Systems lassen sich zwei Richtungen unterscheiden, je nachdem, ob die Kommunikation über ein Blackboard oder

¹²Parallel **V**irtual **M**achine

¹³Die aktuelle Version von ICE unterstützt mehrere Hardware-Plattformen und Betriebssysteme mit Schnittstellen zu C, C++, ALLEGRO Common LISP, Quintus und Sictus Prolog, Tcl/Tk und wurde im Rahmen von CHANNELS auf LUCID Common LISP portiert.

durch direkte Nachrichtenadressierung erfolgt. Prinzipiell bieten die nachrichtenbasierten und blackboardbasierten Modelle die gleiche Funktionalität: Jedes Modell kann durch das andere simuliert werden. Zur Simulation einer Blackboard-Architektur durch das nachrichtenbasierte Modell kann ein Agent oder eine Gruppe von Agenten verwendet werden, um das Blackboard zu simulieren. Die Agenten senden Nachrichten an diesen Agenten, der das Blackboard repräsentiert. Der wiederum leitet die Nachrichten an die anderen Agenten, die durch diese Nachrichten getriggert werden sollen.

Die beiden Modelle schließen sich nicht gegenseitig aus; sie sind eher als Extreme zu betrachten, da die meisten Systeme Elemente beider Ansätze einbeziehen (s. Rich & Knight, 1991, S. 444): Es kann der Fall sein, daß ein Agent nicht genau weiß, an welchen anderen Agenten er eine Nachricht adressieren soll; dennoch weiß er z.B. daß es sich um einen Agenten handeln muß, der zu einer bestimmten Klasse von Agenten gehört. In einer nachrichtenbasierten Architektur kann dies dadurch implementiert werden, daß die Agenten in einer Klassenhierarchie organisiert werden und daß Nachrichten an Klassen adressiert werden können, die die Nachricht an alle Mitglieder der Klasse zustellen. In einer Blackboard-Architektur kann dies durch Hierarchien realisiert werden.

Cawsey *et al.* (1992) haben in einer empirischen Studie den blackboardbasierten Ansatz mit dem aktorbasierten Paradigma bei Multi-Agenten-Systemen verglichen. Ziel ihrer Studie war herauszufinden, welche Systemarchitektur für welchen Anwendungsbereich geeignet ist. Hierzu haben sie evaluiert, welche Auswirkungen die Anzahl der Agenten, die Verteilung der Problemlösungsfähigkeiten unter den einzelnen Agenten sowie die Genauigkeit der Modelle über die Kompetenzen der anderen Agenten haben. Sie zeigten, daß die Auswahl einer Architektur in erster Linie von den Eigenschaften des Anwendungsbereichs abhängt. Sie kommen in dieser empirischen Studie zu folgendem Ergebnis:¹⁴

- (a) Der aktorbasierte Ansatz ist überlegen in Kontexten, in denen das Wissen gleichmäßig über alle Agenten verteilt ist und diese um die Kompetenzen der anderen Agenten wissen, also je präziser das Agentenmodell der einzelnen Agenten ist.
- (b) Wenn es aber Überlappungen in den einzelnen Spezialgebieten der Agenten gibt oder wenn diese wenig über die Kompetenzen der anderen Agenten wissen, dann ist der blackboardbasierte Ansatz geeigneter.

3.2.6 Mit CHANNELS verwandte Ansätze

Wie bereits erwähnt hat CHANNELS Prinzipien aus den objektorientierten nebenläufigen Sprachen (Gasser & Briot, 1992) wie ABCL (Yonezawa, 1990; Yonezawa & Tokoro, 1987) integriert. Die Nützlichkeit verteilter Architekturen zur natürlichsprachlichen Verarbeitung

¹⁴“If knowledge is well distributed, with each agent having distinct types of knowledge they can reason about, and having accurate models of the areas of expertise of other agents, an actor model is appropriate, especially where large numbers of agents are involved. ... However, if there is more overlap in the areas of expertise, or if the agents have only limited knowledge of each others expertise areas, then the blackboard model may be more appropriate.” (Cawsey *et al.*, 1992, S. 251)

wurde auch in anderen NL-Systemen wie ASL (Pyka, 1991; von Hahn, 1992), CAMEL (Sabah, 1990; Sabah & Briffault, 1993), TALISMAN (Stefanini & Demazeau, 1995; Stefanini *et al.*, 1992) sowie in den Arbeiten von Fum *et al.* (1988) gezeigt. Auf diese Systeme wird im Rest dieses Abschnittes kurz eingegangen.

■ ABCL

In ABCL (Yonezawa, 1990; Yonezawa & Tokoro, 1987) werden die Objekte als autonome informationsverarbeitende Einheiten aufgefaßt, die ausschließlich durch Nachrichtenaustausch interagieren. ABCL erlaubt keine Broadcasting, sondern nur eine bidirektionale Punkt-zu-Punkt Kommunikation zwischen den Objekten, wobei ein Objekt zu jeder Zeit Nachrichten an ein anderes schicken kann, unabhängig vom internen Zustand des Empfängers. Die ankommenden Nachrichten werden gepuffert und in der Reihenfolge ihrer Ankunft in die Nachrichtenwarteschlange des Objekts eingetragen. Die Objekte laufen nebenläufig. Ein ABCL-Objekt wird definiert durch seinen internen Zustand (*dormant*, *active*, *waiting*) und durch sein Verhaltensskript, das die Aktionsfolge definiert, die nach Eintreffen von Nachrichten ausgeführt werden soll. Neben

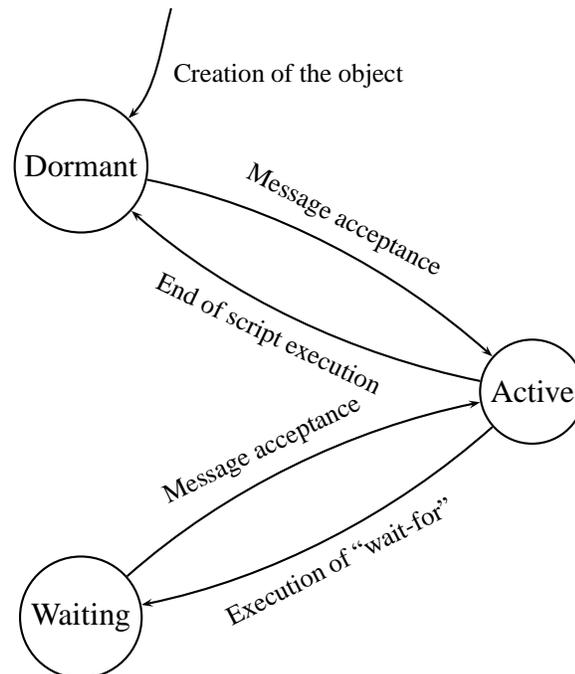


Abbildung 3.8: Übergänge der internen Zustände eines Objektes in ABCL.

den drei internen Zuständen *active*, *dormant* und *waiting*, gibt es drei Nachrichtentypen in ABCL:

- *past*: Wenn ein Objekt O eine Nachricht M an ein Objekt T sendet, dann wartet O nicht darauf, daß T die Nachricht empfangen hat oder eine Antwort auf diese Nachricht gibt. A setzt unabhängig davon seine Berechnungen unmittelbar fort.
- *now* (send and wait): Wenn ein Objekt O eine Nachricht M an ein Objekt T sendet, wartet O auf die Verarbeitung der Nachricht durch T. Bis zum Eintreffen der Ant-

wort auf die abgeschickte Nachricht ist O im Wartezustand. O Kann Nachrichten empfangen aber keine bearbeiten. Der zu O gehörige Prozeß ist suspendiert und kann nur durch die Rückantwort von T reaktiviert werden.

- *future*: Wenn ein Objekt O nicht unmittelbar die Ergebnisse einer Anfrage an ein Objekt T benötigt, kann O eine Nachrichtenübertragung vom Typ *future* verwenden. Nachdem O die Nachricht abgeschickt hat, wartet es nicht auf die Antwort, sondern setzt seine Berechnungen unmittelbar fort. Beim Abschicken der Anfrage wird ein privates Objekt spezifiziert, in dem das Ergebnis abgespeichert wird. Wenn O zu einem späteren Zeitpunkt die Antwort von T benötigt, schaut es in seinem privaten Objekt nach, ob die Antwort bereits vorliegt. (Wenn die Antwort immer noch nicht da ist, setzt O seine Berechnung fort und fragt zu einem späteren Zeitpunkt wieder nach.)

So korrespondiert die *past* Nachrichtenübertragung mit *inform*, die *now* Nachrichtenübertragung mit *synchronous ask* und die *future* Nachrichtenübertragung mit *asynchronous ask*.

ABCL bietet darüber hinaus einen Expreßmodus bei dem ein Objekt den Empfänger der Nachricht veranlassen kann, seine Berechnungen zu unterbrechen und die Nachricht sofort zu verarbeiten. In ABCL gibt es jedoch nur eine Prioritätsstufe für Unterbrechungen. Die Verarbeitung einer Expreßnachricht kann nicht unterbrochen werden. Wird dem Objekt eine weitere Expreßnachricht geschickt, so wird diese zurückgestellt, bis die Verarbeitung der Expreßnachricht abgeschlossen ist.

Yonezawa *et al.* (1990, S. 37–39) haben ein minimales Modell aufgezeigt, wie einerseits *now* als Kombination von *past* und einem selektiven Empfangsmodus im Wartezustand und andererseits *future* auch als Kombination von *past* und *now* reduziert werden kann.

- ASL¹⁵

Grundlage der ASL-Architektur (Pyka, 1991; von Hahn, 1992) ist eine deterministische, inkrementelle und zeitsynchrone Verarbeitung. In ASL wurde das Verstehen kontinuierlich gesprochener Sprache untersucht. Die Architektur integriert zwei Aspekte der natürlichsprachlichen Kommunikation: die Sprachsignalenverarbeitung und die Sprachstrukturanalyse. In ASL werden die verschiedenen Systemkomponenten als Module, die, im Unterschied zu CHANNELS, über eine Blackboard-Architektur ohne Ebenhierarchie und zentrale Kontrolle durch Nachrichten kommunizieren.

- CAMEL¹⁶

Das System CAMEL (Sabah, 1990) verfolgt ebenfalls den blackboardbasierten Ansatz mit einer zentralen Steuerungskomponente, dem sog. *Supervisor*, der die gesamte Interaktion und Kooperation zwischen den Agenten verwaltet. CAMEL hat mehrere Wissensquellen, die den einzelnen Abstraktionsebenen bei der Sprachverarbeitung entsprechen. Der Supervisor verwendet Metaregeln zur Steuerung der Komponenten. In den späteren Entwicklungen wurden Ansätze der Aktorensysteme aufgenommen, um

¹⁵Architecture for integrated Speech and Language Systems

¹⁶Comprehension Automatique de Récits, Apprentissage, Modélisation des Echanges Langagiers

zusätzliche Flexibilität zu gewähren (Sabah & Briffault, 1993).

■ **TALISMAN**

Das System TALISMAN (Stefanini & Demazeau, 1995; Stefanini *et al.*, 1992) ist ein Beispiel für eine Multi-Agenten-Architektur zur natürlichsprachlichen Verarbeitung mit direkter Nachrichtenadressierung. TALISMAN ist eine natürlichsprachliche Schnittstelle, in der die Agenten zu den verschiedenen linguistischen Ebenen (Morphologie, Syntax, Semantik) und zu den komplexeren sprachlichen Phänomenen (z.B. Disambiguierung, Ellipsen, Negation, Koordination) korrespondieren. Die Kommunikation zwischen den 8 Agenten wird dezentral mit Hilfe sog. linguistischer Regeln¹⁷ (engl. *linguistic laws*) gesteuert. Die Nachrichten in TALISMAN werden auch wie in CHANNELS neben den Angaben über den Sender, die Empfänger und den Inhalt der Nachricht durch den Übermittlungsmodus (synchron oder asynchron) sowie durch Kommunikationsakte (*assert*, *inform*, *request*, *answer*, ...) definiert. Letztere basieren auf zwei Primitiven *send* und *deliver*. Ein Agent in TALISMAN kann auch einen anderen anhalten, um Serialisierbarkeit zu realisieren.

■ Das Modell von Fum *et al.* (1988)

Fum *et al.* (1988) stellten ein Modell einer Multi-Agenten-Architektur für natürlichsprachliche Verarbeitung vor. In ihrem Ansatz wurden grobkörnige (engl. *large-grained*) heterogene Agenten (Spezialisten genannt) modelliert, die zur Lösung von NL-Aufgaben kooperieren. Das Modell zeichnet sich durch zentrale Steuerung und eine Kombination von ereignisgesteuerten (*bottom-up*, *event-driven*) und zielgesteuerten (*top-down*, *goal-driven*) Vorgehensweisen aus. Im Gegensatz zu CHANNELS wurde dort die gesamte Interagenten-Kommunikation durch einen zentralen Kooperationsmanager (*cooperation manager*) gesteuert.

3.2.7 Andere Anwendungen von CHANNELS

Die CHANNELS-Architektur wurde in MOSES (Maaß, 1996; Paul, 1996) eingesetzt, einem System, das ausgehend von visuellen Daten des Saarbrücker Universitätscampus inkrementelle multimodale Wegbeschreibungen in einer dreidimensionalen Umgebung erzeugt. Das System besteht aus sechs Komponenten: dem visuellen Objektauswahlmodul, dem Pfadsuchmodul, dem Sprachproduktions- und Animationsmodul, dem Navigationsmodul, dem Modul zur Integration raumbezogenen Wissens und dem Steuerungsmodul für die Kontrolle und Adaption des Systems an Ressourcenbeschränkungen. Diese einzelnen Systemmodule wurden mit Hilfe der CHANNELS-Architektur als Agenten modelliert, die ausschließlich durch direkten Nachrichtaustausch kommunizieren. Die zwischen einigen Modulen notwendigen Sequentialisierungsbeziehungen wurden durch synchronen Nachrichtenaustausch implementiert.

Durch CHANNELS wurde eine Anbindung eines Lisp-Systems mit einer Java-Anwendung realisiert. Das System JALCOSTE¹⁸ erlaubt es, mit Hilfe von Agenten von

¹⁷S. die Arbeiten zu den sog. Law governed systems (Minsky, 1991).

¹⁸Java-Lisp Connection System

einem LISP-Image aus auf eine Benutzeroberfläche zuzugreifen, die aus einem Java-Applet besteht. (Applets sind Java-Anwendungen oder Anwendungsbausteine.) Die Verbindung zwischen dem LISP- und dem Java-System wird über Sockets realisiert. Die anderen Agenten im LISP-Image, die etwas auf die Benutzeroberfläche generieren wollen, schicken Kommandos bzw. Anfragen durch einen ausgewählten Agenten an das Java-Applet. Dieses schickt Antworten auf Anfragen der Agenten auf der LISP-Seite sowie aufgetragene Events (z.B. angeklickte Buttons) zurück, die aus diesen Anfragen resultieren. Auf der LISP-Seite leitet der Agent die aus dem Java-Applet eintreffenden Antworten und Events an die betroffenen Agenten weiter. Die Agenten in der LISP-Seite benutzen lediglich die von dem Java-System durch JALCOSTE bereitgestellten Bibliotheksfunktionen. Sie schicken bei Bedarf Nachrichten an den ausgewählten Agenten, der für die Verbindung zum Java-Applet verantwortlich ist.

3.3 Die PRACMA-Agenten

3.3.1 Ein Überblick über das PRACMA-System

In den meisten bisher existierenden natürlichsprachlichen Dialogsystemen wurden kooperative Dialogsituationen modelliert. In diesen Systemen basiert die pragmatische Verarbeitung auf den Konversationsmaximen von Grice (1975): Das Verhalten des Systems sollte möglichst an die Bedürfnisse des Benutzers angepaßt und möglichst informativ sein.¹⁹

Systeme, die nicht ausschließlich kooperative Dialogsituationen modellieren, verfolgen hingegen ein eigenes Interesse. Die Motivationen des Systems und die des Dialogpartners sind teilweise gegensätzlich. Das System PRACMA (s. Jameson *et al.*, 1995; 1994 sowie 1997) modelliert bewertungsorientierte Dialoge am Beispiel von Verkaufsgesprächen, die nicht ausschließlich kooperative Dialogsituationen darstellen: Der Käufer möchte dem Verkäufer möglichst detaillierte und korrekte Informationen entlocken und das Auto gegebenenfalls zu einem niedrigen Preis erwerben, während der Verkäufer Fakten verschweigen oder schmücken kann, um das Auto in ein günstiges Licht zu stellen und somit seine Preisvorstellungen durchzusetzen. Abbildung 3.9 gibt einen Beispieldialog (s. Wahlster *et al.*, 1997) zwischen einem Verkäufer (\mathcal{V}) eines in der Zeitung annoncierten Gebrauchtwagens und einem potentiellen Käufer (\mathcal{K}), der sich telefonisch über das Auto informieren will, wieder. PRACMA kann jeweils eine der Dialogrollen übernehmen.

3.3.2 Der EVALUATION HANDLER

Der Agent EVALUATION HANDLER (Schäfer, 1997) verarbeitet die Bewertungen des Käufers sowohl wenn PRACMA die Rolle des Käufers als auch wenn es die Rolle des Verkäufers übernimmt. In der Rolle des Käufers nimmt der EVALUATION HANDLER die Bewertung des Verkaufsobjekts vor; In der Rolle des Verkäufers schätzt er die vermuteten Bewertungskriterien des Käufers ein.

¹⁹Ausnahmen sind z.B. die Systeme PAULINE (Hovy, 1988) und IMP (Jameson, 1989)

1. \mathcal{V} : Wohnen Sie in der Stadt oder auf dem Land?
Obwohl solche Fragen nach Eigenschaften von \mathcal{K} diesen irritieren können, stellt \mathcal{V} sie, falls der zu erwartende Informationsgewinn hoch genug erscheint.
2. \mathcal{K} : Ich wohne auf dem Land.
Aus dieser Angabe zieht \mathcal{V} Schlüsse über \mathcal{K} s Interesse an Zuverlässigkeit.
3. \mathcal{V} : Wie interessiert sind Sie an Komfort?
Auch diese direkte Frage nach einer Interessensdimension scheint \mathcal{V} die Mühe wert zu sein.
4. \mathcal{K} : Ich bin ziemlich interessiert an Komfort.
Diese Antwort läßt direktere Schlüsse zu, auch wenn sie nicht für bare Münze genommen werden darf.
5. \mathcal{V} : Dieses Auto könnte etwas für Sie sein.
6. \mathcal{V} : Wie in der Annonce gesagt, handelt es sich um einen Golf, Baujahr 1991.
Stünde mehr als ein Auto zur Auswahl, würde \mathcal{V} an dieser Stelle ein geeignetes auswählen und vorschlagen. Da hier nur eins zur Verfügung steht, wird lediglich an die in der Zeitungsannonce genannten Eigenschaften erinnert.
7. \mathcal{V} : Er hat vier Türen.
Von allen möglichen Aussagen über einzelne Attribute des Autos erwartet \mathcal{V} , daß diese die positivsten Auswirkungen auf die Bewertung von \mathcal{K} haben wird.
8. \mathcal{K} : Das ist gut.
Diese bewertende Reaktion spiegelt die *Bewertungsverschiebung* wider, die die Aussage hervorruft. \mathcal{V} interpretiert sie vor allem als Zeichen von Interesse an Komfort.
9. \mathcal{V} : Das Auto hat Automatikgetriebe.
Angesichts des angepaßten Benutzermodells und des aktuellen Dialogkontextes scheint \mathcal{V} diese Aussage am nützlichsten.
10. \mathcal{K} : [—]
Da \mathcal{K} s Nutzen aller hier möglichen Äußerungen unter dem von Schweigen liegt, gibt \mathcal{K} zu erkennen, daß \mathcal{V} weiterreden soll.
11. \mathcal{K} : Ist der Kilometerstand hoch?
Jetzt findet \mathcal{K} , daß es Zeit ist, eine Frage zu stellen, auf die er vermutlich eine Antwort mit hohem Informationswert erhalten wird.
12. \mathcal{V} : Der Kilometerstand ist nicht besonders niedrig.
Da eine präzise Antwort hier eine besonders ungünstige Bewertungsverschiebung bei \mathcal{K} bewirkt hätte, wählt \mathcal{V} eine vage Formulierung.
13. \mathcal{K} : Das ist schlecht.
Die Reaktion von \mathcal{K} deutet sowohl auf ein starkes Interesse an Zuverlässigkeit als auch auf eine ursprünglich optimistische Erwartung bzgl. des Kilometerstandes hin.
14. \mathcal{K} : Wie groß ist die Chance, daß er den nächsten TÜV besteht?
Diese Frage erzeugt eine starke Dialogverpflichtung, etwas über diese Chance zu erzählen.
15. \mathcal{V} : Die Zeit bis zum nächsten TÜV ist 1,5 Jahre.
 \mathcal{V} will diese Verpflichtung nicht erfüllen, da die Wahrheit ungünstig ist. Sie wählt eine Antwort, die nicht als eindeutige Verletzung der Verpflichtung gilt.
16. \mathcal{K} : Ich bin an Sicherheit sehr interessiert.
Diese Äußerung stellt eine schwache Verpflichtung auf, etwas über die sicherheitsrelevanten Aspekte des Autos zu sagen.
17. \mathcal{V} : Es hat einen rechten Außenspiegel.
Diesmal liegt es im Interesse von \mathcal{V} , der Verpflichtung nachzukommen.
18. \mathcal{K} : Aha.
Da \mathcal{K} keine bewertungsrelevanten Implikationen zu dieser Aussage erkennt, reagiert er neutral.
19. \mathcal{V} : Das ist gut für die Sicherheit.
 \mathcal{V} schätzt jetzt, daß \mathcal{K} die Implikationen der vorigen Äußerung nicht erkannt hat. Es erscheint ihr die Mühe wert, auf die Implikationen hinzuweisen.

Abbildung 3.9: Beispieldialog, in dem PRACMA sowohl die Rolle des Verkäufers (\mathcal{V}) als auch die des Käufers (\mathcal{K}) übernehmen kann.

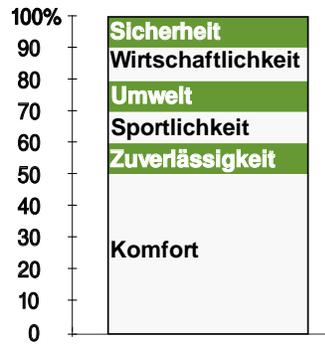


Abbildung 3.10: Beispiel eines Interessenprofils.

(Das System unterhält durch den EVALUATION HANDLER ein solches Profil über den Dialogpartner sowie ein eigenes Interessenprofil. Die Höhe der Segmente gibt die Wichtigkeit wieder, die der entsprechenden Bewertungsdimension beigemessen wird. In dem Beispiel handelt es sich um eine Person, der Komfort sehr wichtig ist und alle anderen Dimensionen gleich wichtig sind.)

In PRACMA wird für die Bewertung der einzelnen Attribute sowie für die Gesamtbewertung des Autos die *Multi-Attribute-Utility-Theory* (MAUT, Multiattributive Objektbewertung) (von Winterfeldt & Edwards, 1986) verwandt. Dabei wird das Auto auf *Bewertungsdimensionen* bewertet, die durch eine individuell gewichtete Summe zu einer Gesamtbewertung aggregiert werden. In der von PRACMA modellierten Autodomäne handelt es sich um sechs Dimensionen: Komfort, Sicherheit, Wirtschaftlichkeit, Sportlichkeit, Umweltfreundlichkeit und Zuverlässigkeit. Die Höhe der Gewichtungen für die einzelnen Bewertungsdimensionen ergibt das Interessenprofil²⁰ (s. Abbildung 3.10). Die Bewertung auf den einzelnen Bewertungsdimensionen wird durch die gewichtete Summe der Bewertung der Attribute des Autos errechnet. Die Bewertungen der relevanten Attribute werden mit Hilfe einer Bewertungsfunktion vorgenommen, die jede Ausprägung auf einen Wert abbildet. Abbildung 3.11 gibt ein Beispiel zur Ermittlung der Gesamtbewertung nach dem MAUT-Verfahren.

In der Käufer-Rolle

In der Rolle von \mathcal{K} muß das System durch den EVALUATION HANDLER zwei Aufgaben erfüllen: die Ermittlung der Gesamtbewertung des Autos und die Auswahl von Fragen, die an \mathcal{V} gestellt werden sollten sowie die Auswahl einer geeigneten bewertenden Reaktion nach einer Äußerung des \mathcal{V} über ein Merkmal des Autos.

Oft sind die Einschätzungen von Bewertungen sowie deren Beziehungen untereinander mit Unsicherheit behaftet, da bei der Bewertung eines Objektes nicht alle Ausprägungen der einzelnen Attribute bekannt sind. Um diese Unsicherheit über die Ausprägungen je-

²⁰Das Interessenprofil eines Käufers ist ein Vektor von Verteilungen, die angeben, wie interessiert er an den einzelnen Bewertungsdimensionen ist. Dabei ist die Hauptbewertungsdimension die dem Käufer wichtigste Bewertungsdimension. Das Interessenprofil des Dialogpartners wird probabilistisch eingeschätzt. Diese Einschätzung wird durch Wahrscheinlichkeitsverteilungen dargestellt.

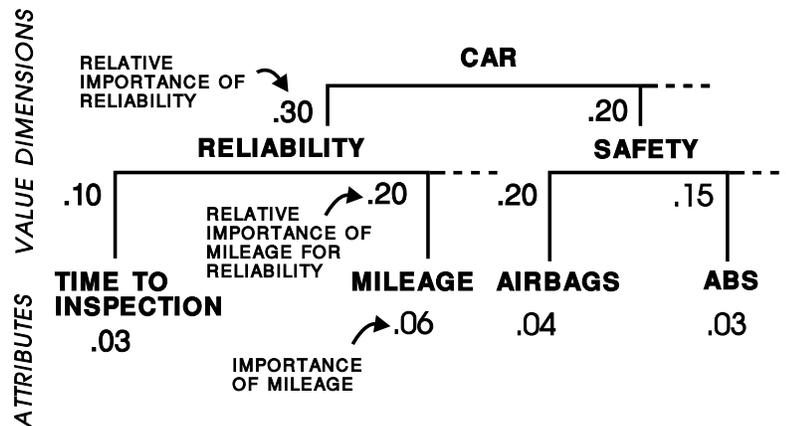


Abbildung 3.11: Teil des Bewertungsbaumes zur Ermittlung der Gesamtbewertung eines Autos nach dem MAUT-Verfahren.

(Diese Bewertung wird vom System in der Rolle des Käufers vorgenommen. Jede Ausprägung eines Attributs wird durch die Bewertungsfunktion auf einen Wert abgebildet. Die gewichtete Summe der Bewertung der einzelnen Attribute bzgl. den für diese Attribute relevanten Bewertungsdimensionen ergibt die Gesamtbewertung des Autos.)

des Attributs zu berücksichtigen, werden diese in Form einer Wahrscheinlichkeitsverteilung geschätzt. Die Bewertung jedes einzelnen Attributs kann dann mit Hilfe der Bewertungsfunktion in Form eines *Einflußdiagramms*²¹ (s. z.B. Neapolitan, 1990) erstellt werden. Die Gesamtbewertung läßt sich dadurch errechnen, daß die Attributbewertungen additiv aggregiert werden (s. Schäfer, 1997).

Für die Auswahl von Fragen an \mathcal{V} muß \mathcal{K} für jedes Attribut des Autos ermitteln, inwieweit es im Vergleich zu anderen zur Unsicherheit über die Gesamtbewertung beiträgt. Aufgrund dieses Vergleiches wird die nächste zu stellende Frage von \mathcal{K} ausgewählt.

Nachdem der Verkäufer Informationen über ein Merkmal des Autos gegeben hat, muß sich das System (\mathcal{K}) überlegen, wie es auf die erhaltenen Informationen mit einer bewertenden Äußerung reagiert. Dabei muß \mathcal{K} die *Bewertungsverschiebung* berechnen. Diese ergibt sich aus der Differenz zwischen den Erwartungswerten der jetzigen Bewertung und der Bewertung vor Erhalt der neuen Information. Die ausgewählte verbale Reaktion wird dann an den Agenten COMMENT AND QUESTION HANDLER geschickt, der für die Äußerungsauswahl zuständig ist (s. den späteren Abschnitt 3.3.3 sowie Abbildung 3.14).

In der Verkäufer-Rolle

Wenn PRACMA die Rolle von \mathcal{V} übernimmt, muß das System unter Berücksichtigung des Bewertungsmodells — wie es in der Rolle von \mathcal{K} verwendet wurde — eine Einschätzung von \mathcal{K} s Bewertungen durchführen. Dazu werden die Interessen des \mathcal{K} an den verschiedenen

²¹Einflußdiagramme sind Erweiterungen von Bayesschen Netzen (s. z.B. Pearl, 1991) und werden zur Berechnung von Entscheidungen und Bewertungen unter Unsicherheit eingesetzt.

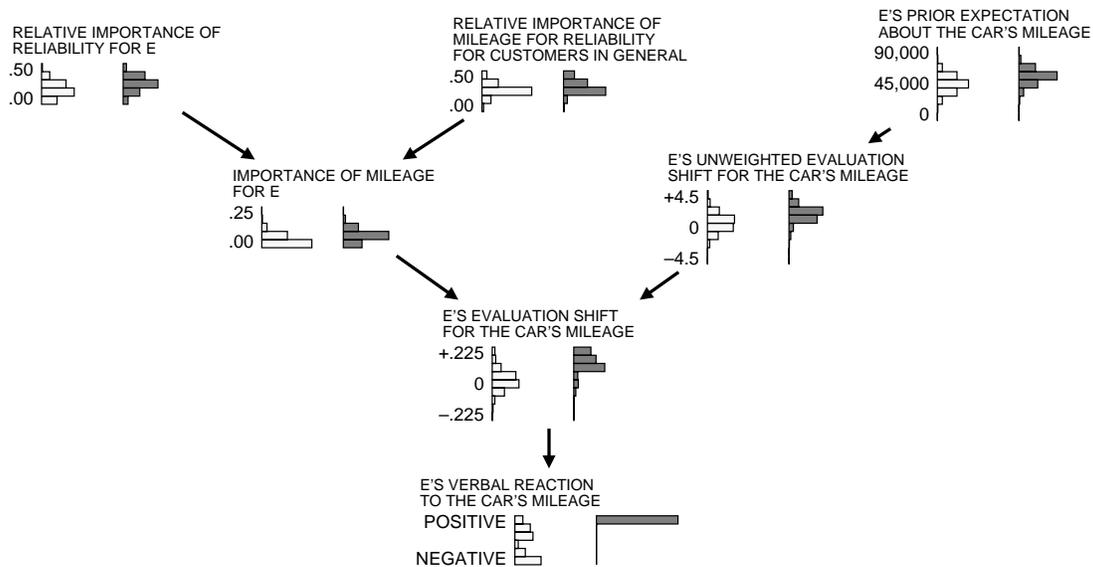


Abbildung 3.12: Auszüge eines Bayesschen Netzes, das PRACMA in der Rolle von \mathcal{V} verwendet, um bewertende Reaktionen zu Äußerungen des Verkäufers vorherzusagen bzw. zu interpretieren.

(Ein Pfeil von A nach B bedeutet, daß A B beeinflusst. In der Abbildung steht E für Evaluator, also für den Käufer.)

Bewertungsdimensionen eingeschätzt. (Es ergibt sich dann ein Interessenprofil wie in Abbildung 3.10.) Anhand von \mathcal{K} s Antworten, Fragen und bewertenden Reaktionen kann das System mit Hilfe eines Bayesschen Netzes Schlüsse über \mathcal{K} s Bewertungskriterien ziehen. Zum Beispiel läßt eine Frage nach einem Attribut hohe Unsicherheit bei der Attributbewertung vermuten und aufgrund dessen auf Interesse an der betroffenen Dimension schließen. Um \mathcal{K} s Schätzungen über die einzelnen Attributsausprägungen zu bestimmen, werden sog. *Meta-Knoten* (Ndiaye & Jameson, 1994) eingesetzt. Diese Meta-Knoten entsprechen Parametern, die verwendet werden, um Wahrscheinlichkeitsverteilungen zu beschreiben. Solche Parameter können z.B. der Erwartungswert und die Ausdehnung einer Verteilung sein.

In der Rolle von \mathcal{V} muß das System die Reaktionen des \mathcal{K} s auf die vom System gemachten Äußerungen vorhersagen bzw. interpretieren, um das Modell über \mathcal{K} s Bewertungen anzupassen. Abbildung 3.12 gibt ein Beispiel einer solchen Verarbeitung von Reaktionen des \mathcal{K} s auf eine Äußerung über den Kilometer-Stand des Autos. Als \mathcal{V} muß das System antizipieren, wie ein Käufer auf eine bestimmte Beschreibung eines Attributs reagieren wird. Hierzu sagt es die Bewertungsverschiebung bei \mathcal{K} voraus, indem es die vermutete jetzige Bewertung des \mathcal{K} mit der *nach* Erhalt der neuen Information entstehenden Bewertung vergleicht. Die vorausgesagte Bewertungsverschiebung ergibt sich aus der Differenz (s. die Funktion PREDICTED-EVALUATION-SHIFT in Abbildung 3.14 im nächsten Abschnitt.)

3.3.3 Der COMMENT AND QUESTION HANDLER

Der Agent COMMENT AND QUESTION HANDLER (abgekürzt als CQH) ist für die Modellierung und Manipulation von Eindrücken zuständig und verwendet hierzu entscheidungstheoretische Verfahren zur Behandlung bewertungsorientierter Dialoge.

Der CQH ist einerseits für die detaillierte Interpretation der Äußerungen des Dialogpartners und andererseits für die Auswahl der Systemäußerungen zuständig, wenn der Typ der Äußerung vom Dialogplaner vorgegeben ist. Der CQH übernimmt folgende Aufgaben:

- In der Verkäuferrolle: Der CQH bestimmt dann vage Aussagen, die über das Auto bzw. dessen Attribute gemacht werden können sowie die Implikationen dieser Aussagen auf die Bewertung. Der CQH in der Rolle von \mathcal{V} wählt Fragen über \mathcal{K} s persönliche Merkmale und Interessen an bestimmten Attributen oder Dimensionen aus und interpretiert \mathcal{K} s bewertende Äußerungen auf \mathcal{V} s Mitteilungen.
- In der Käuferrolle: Der CQH ist für die Auswahl der Fragen nach Eigenschaften des Autos sowie für die Auswahl der bewertenden Reaktionen auf \mathcal{V} s Aussagen verantwortlich. Das Modul entscheidet auch, ob \mathcal{K} dem Verkäufer sein Interesse an bestimmten Attributen oder Bewertungsdimensionen bekunden soll.

Der COMMENT AND QUESTION HANDLER verwendet einen entscheidungstheoretischen Ansatz. Für jede mögliche Äußerung werden die erwarteten Kosten und Nutzen von potentiellen Äußerungen geschätzt, um den Gesamtnutzen für das System zu ermitteln. Hierzu fließen die Ergebnisse des Agenten EVALUATION HANDLER ein, der für die Verarbeitung von Bewertungen zuständig ist. Zur Bestimmung des Dialogzuges mit dem größten Gesamtnutzen für das System wird nach dem Schema in Abbildung 3.13 verfahren. Die potentiellen Züge und die darauf folgenden Reaktionen werden folgendermaßen quantifiziert (s. Abbildung 3.14):²²

```

function BEST-MOVE(type, constraints) returns a dialog move of type
type that fulfills constraints
  /* Executed by CQH */
  possible-moves ← ALLOWABLE-MOVES(type, constraints)
  for m in possible-moves do
    UTILITY[m] ← UTILITY-OF-MOVE(m)
  end
  return the m in reasonable-moves with the highest UTILITY[m]

```

Abbildung 3.13: CQHs Algorithmus zur Auswahl eines Dialogzuges bei vorgegebenem Typ ohne Einsatz der globalen Antizipation.

²²Die Beispieläußerungen beziehen sich auf den Beispieldialog von Abbildung 3.9. Die Zahl in Klammern entspricht der Numerierung der Dialogbeiträge in dem Beispieldialog. Wir geben jeweils ein Beispiel (eine Äußerung) und das verwendete Kriterium zur Quantifizierung des Nutzens der Äußerung.

```

function UTILITY-OF-MOVE(move) returns a utility
/* Executed by CQH */
case TYPE[move]
  Moves possible in both roles:
    silence
    return 0
  ...
  Moves possible in role of seller:
    comment-on-attribute:
    return ASK(EVALUATION HANDLER, "PREDICTED-
      EVALUATION-SHIFT(move)")
  ...
  Moves possible in role of buyer:
    question-about-attribute:
    return ASK(EVALUATION HANDLER, "CURRENT-EVALUATIVE-
      UNCERTAINTY(TOPIC[move])")
    evaluative-reaction:
    return INFORMATIVENESS(move)
  ...

```

```

function PREDICTED-EVALUATION-SHIFT(comment) returns an
estimate of the shift in  $\mathcal{U}$ 's evaluation that comment would lead to
/* Executed by the EVALUATION HANDLER */
construct and evaluate a Bayesian network to predict  $\mathcal{U}$ 's evaluation
shift, taking into account  $\mathcal{S}$ 's uncertain beliefs about  $\mathcal{U}$ 's interests and
knowledge
prediction  $\leftarrow$  the probability distribution representing the resulting
belief concerning  $\mathcal{U}$ 's evaluation shift
return EXPECTED-VALUE(prediction)

```

Abbildung 3.14: Algorithmen zur Bewertung des unmittelbaren Nutzens eines Dialogzuges. (In Abhängigkeit von der vom System übernommenen Rolle und vom Typ des potentiellen Dialogzuges werden die entsprechenden Anfragen an den Agenten EVALUATION HANDLER gestellt. Z.B. PREDICTED-EVALUATION-SHIFT entspricht der erwähnten Bewertungsverschiebung in Abschnitt 3.3.2.)

■ Potentielle Züge

- Kommentar von \mathcal{V} über ein Attribut des zu bewertenden Objekts
Beispiel: "Der Kilometerstand ist nicht besonders niedrig" (12).
Kriterium: Welche positiven oder negativen Auswirkungen hat die Aussage auf \mathcal{K} 's Bewertung?
- Frage über eine Eigenschaft von \mathcal{K}
Beispiel: "Wie interessiert sind Sie an Komfort" (3) oder "Wohnen Sie auf dem Land?" (1).
Kriterium: Wie wird meine Unsicherheit über \mathcal{K} 's Bewertungsmaßstäbe reduziert bzw. wie hoch ist der Informationsgewinn?
- Frage von \mathcal{K} nach einem Attribut
Beispiel: "Ist der Kilometerstand hoch?" (11)
Kriterium: Wieviel Unsicherheit über den Käufer wird bei \mathcal{V} reduziert?

- Bewertende Äußerung von \mathcal{K}
Beispiel: “Das ist gut.” (8).
Kriterium: Welche bewertungsrelevanten Implikationen hat diese Äußerung?
- Potentielle Reaktionen
 - Frage von \mathcal{K} nach einem Attribut
Beispiel: “Ist der Kilometerstand hoch?”
Kriterium: Was ist der Nutzen einer ehrlichen Antwort auf diese Frage?
 - Bewertende Äußerung von \mathcal{K}
Beispiel: “Das ist gut.” (8)
Kriterium: Wie positiv ist die Reaktion?
 - Kommentar von \mathcal{V} über ein Attribut des zu bewertenden Objekts Beispiel: “Der Kilometerstand ist nicht besonders niedrig.” (12)
Kriterium: Wieviel Unsicherheit wird im Modell von \mathcal{V} über \mathcal{K} reduziert?

3.3.4 Der DIALOG PLANNER

Der Agent DIALOG PLANNER ist in Koordination mit dem COMMENT AND QUESTION HANDLER für die Interpretation der Benutzeräußerungen und für die Generierung von Systemäußerungen verantwortlich. Der Dialogplaner bestimmt, *was für einen* Dialogzug das System als nächstes machen kann. Hierzu stellt der Planer Anfragen an CQH — den Agenten zur Äußerungsauswahl —, ob es sich grundsätzlich lohnt, eine Anmerkung eines bestimmten Typs zu machen.

PRACMA verwendet einen um iterative Konstrukte erweiterten hierarchischen inkrementellen Dialogplaner (Weis, 1994). Die Dialogverarbeitung in PRACMA ist in Dialogphasen strukturiert: Dialogeröffnung, Akquisition, Informationspräsentation, die wiederum in Unterphasen untergliedert sind. In der Akquisitionsphase versucht das System in der Rolle des Verkäufers Informationen etwas über die persönlichen Merkmale (etwa Beruf, Wohnort) sowie über die Interessen bzgl. der Hauptbewertungsdimensionen (etwa Komfort) zu erfahren. Die Eröffnungs- und die Akquisitionsphase dienen dem Aufbau bzw. der Validierung eines Modells des Dialogpartners. Die Informationspräsentation umfaßt das Anpreisen des Autos sowie den Informationsaustausch mit dem Käufer. Das Systemverhalten in der Informationspräsentationsphase hängt von verschiedenen Aspekten ab, die die Dialogstrategie und die Dialogtaktik bestimmen. Die Dialogstrategie hängt ab von Systemeinstellungen wie dem Grad der Beredsamkeit (im EGO repräsentiert), den Einschätzungen aus dem aktuellen Benutzermodell, etc. Zur Erfüllung eines Dialogzieles kann das System i.a. verschiedene Dialogtaktiken verwenden, die die Sprechaktwahl steuern. Das Dialogverhalten hängt auch von den Erwartungen des Dialogpartners über das Dialogverhalten des Systems ab, den sog. Dialogverpflichtungen (Jameson & Weis, 1996; Weis, 1996). (Stellt \mathcal{K} eine Frage, erzeugt dies eine Verpflichtung für \mathcal{V} , diese Frage zu beantworten; das Ignorieren oder Zurückstellen der Frage geht mit gewissen “sozialen” Kosten für die Verletzung dieser Dialogverpflichtung einher.) Die Dialogverpflichtungen werden vom PRAGMATIC DIALOG MEMORY verwaltet und vom DIALOG PLANNER bei der Auswahl von Dialogtaktiken berücksichtigt.

Der vom Dialogplaner festgelegte Sprechakt-Typ und Inhalt des nächsten Dialogbeitrages wird dem Agenten NATURAL LANGUAGE GENERATOR übergeben, der für die Formulierung in eine natürlichsprachliche Äußerung zuständig ist.

3.3.5 Der EVALUATION EXPRESSION HANDLER

Dieser Agent wird bei der Interpretation der Benutzeräußerungen zur semantischen Verarbeitung zusammengesetzter bewertender Äußerungen eingesetzt (Werner, 1997). (Da die Einzelheiten dieser Komponente nicht entscheidend für die globale Antizipation sind, wird nicht näher auf diesen Agenten eingegangen.)

3.3.6 Der PRAGMATIC DIALOG MEMORY

Dieser Agent verwaltet die Dialoghistorie, eine Wissensquelle, in der Wissen über den bisherigen Verlauf des Dialoges abgespeichert ist. Darüber hinaus werden von den Sprechakten Dialogverpflichtungen abgeleitet, die vom Dialogplaner verarbeitet werden (Jameson & Weis, 1996; Weis, 1996). Der PRAGMATIC DIALOG MEMORY verwaltet mit Hilfe eines regelbasierten Ansatzes die Dialogverpflichtungen für \mathcal{V} und \mathcal{K} jeweils in einer Agenda. Die Information über den Dialogverlauf und die Dialogverpflichtungen werden den anderen Systemmodulen auf Anfrage zur Verfügung gestellt bzw. diesem unmittelbar mitgeteilt.

3.3.7 Der DOMAIN BELIEF HANDLER

Der DOMAIN BELIEF HANDLER verarbeitet das Domänenwissen des Käufers, wenn PRACMA die Rolle des Käufers übernimmt, bzw. das dem Käufer zugeschriebene Domänenwissen, wenn PRACMA die Rolle des Verkäufers übernimmt. Hierzu wird das modallogische Wissensrepräsentationssystem MOTEL (Hustadt & Nonnengart, 1993) verwendet. Zu den Aufgaben des DOMAIN BELIEF HANDLER gehört die Rekonstruktion der Inferenzketten beim Dialogpartner. In MOTEL wird dem Dialogpartner bei dieser Rekonstruktion ein gewisses allgemeines Domänenwissen zugeschrieben. Man weiß jedoch nicht immer mit Sicherheit, ob er tatsächlich über dieses Wissen verfügt. Daher wurde bei dem Einsatz von MOTEL in PRACMA eine Integration modallogischer und probabilistischer Ansätze (Jameson, 1995) ausgearbeitet.

3.3.8 Der EGO

Im EGO werden die Motivationsparameter und Verhaltenspräferenzen des Systems repräsentiert. Der EGO stellt eine Wissensbasis dar, in der die Einstellungen des von PRACMA modellierten Käufers oder Verkäufers abgespeichert sind. Diese Einstellungen können von den anderen Agenten abgefragt werden und betreffen Aspekte wie:

- Dialogrolle:
Die in der Dialogsituation vom System modellierte Rolle des Verkäufers oder die des potentiellen Käufers.

- **Beredsamkeit**
Dieses Merkmal betrifft die Neigung des Systems gesprächig zu sein oder nicht. Je nachdem wie die Einstellung ist, bestimmt dieses Merkmal, wie ausführlich die Systemäußerungen (Antworten und freiwillige Kommentare) ausfallen.
- **Frageneigung:**
Hierdurch wird die Neigung des Systems gesteuert, in der Rolle des Käufers Fragen an den Verkäufer über die Eigenschaften des Autos bzw. in der Rolle des Verkäufers Fragen an den potentiellen Käufer über dessen Interessen zu stellen.
- **Kooperativität:**
Dies ist ein Maß, wie weit die Ziele des Systems und die des Dialogpartners auseinanderklaffen.

Die im EGO abgespeicherten Einstellungen haben Auswirkung auf die Dialogstrategie des Systems. So hat z.B. die Frageneigung Einfluß auf die Auswahl der Dialogpläne.

3.3.9 Der NATURAL LANGUAGE ANALYZER

Die Analyse der natürlichsprachlichen Eingabe des Dialogpartners wird vom Agenten NATURAL LANGUAGE ANALYZER übernommen. Dieser Agent ist für die morphologische, syntaktische und semantisch-pragmatische Analyse der natürlichsprachlichen Eingaben zuständig. Diese Eingaben können vage Ausdrücke, Modalverben und Adverbien enthalten. Es gibt zwei Realisierungen:

- Verarbeitung eingetippter deutschsprachiger Texte. Sie ist in einer Blackboard-Architektur realisiert (s. Kipper, 1994) und macht Gebrauch von dem Parser DISCO (Uszkoreit *et al.*, 1994). Diese Realisierung zeigt anhand von Beispielen, wie die pragmatischen Agenten von PRACMA mit Agenten für oberflächennahe Verarbeitung gekoppelt werden können.
- Eine schnellere, technisch einfachere Realisierung durch Verwendung einer hierarchisch aufgebauten Menü-Schnittstelle. Mit Hilfe dieser Menü-Schnittstelle kann der Benutzer in seiner entsprechenden Rolle aus einer Vielzahl möglicher Äußerungen eine auswählen.

Die Ergebnisse der natürlichsprachlichen Analyse sind zunächst in \mathcal{NLL} (Laubsch & Nerbonne, 1991) repräsentiert. \mathcal{NLL} ist eine logische Sprache zur semantischen Repräsentation der Bedeutung natürlichsprachlicher Ausdrücke. Diese \mathcal{NLL} -Ausdrücke werden von dem Modul zur Erkennung propositionaler Einstellungen (Kipper, 1995) in eine semantische Repräsentation in MOTEL überführt.

3.3.10 Der NATURAL LANGUAGE GENERATOR

Dieser Agent ist für die Verbalisierung der Ergebnisse der Dialogplanung. Der Agent erhält von Agent DIALOG PLANNER Strukturen, die er in natürlichsprachlichen Äußerungen formuliert. Analog zum NATURAL LANGUAGE ANALYZER liegen zwei Realisierungen vor:

- Verwendung des Generators VM-GEN (Kilger & Finkler, 1995), ein inkrementeller Textgenerator. VM-GEN weist eine Kaskadenarchitektur auf, in der die Module zum Textdesign und die Textrealisierung die Komponenten der Kaskade bilden. Dem Generator liegt eine Baumadjunktgrammatik (*tree adjoining grammar*, TAG) zugrunde.
- Verwendung von schablonenbasierten Techniken (Templates) zur natürlichsprachlichen Ausgabe.

3.3.11 Der GLOBAL ANTICIPATOR

Der GLOBAL ANTICIPATOR dient der globalen Antizipation der Benutzerreaktionen und verwaltet die zweite Instantiierung des PRACMA-Systems zur Simulation der Rolle des Dialogpartners. Im Abschnitt 3.6 wird auf dieses Modul ausführlich eingegangen.

3.3.12 Der PRACMA SURFACE

Zuständig für die Realisierung der Interaktion zwischen dem Benutzer und dem System ist der in dem *Common Lisp Interface Manager* (CLIM, 1992) implementierte PRACMA SURFACE. Dieser Agent realisiert die graphische Schnittstelle. Zu seinen Aufgaben gehören die graphischen Ausgaben, das Bereitstellen des Eingabemenüs, das Anzeigen von den Nachrichten zwischen den Agenten und das Anzeigen vom internen Systemzustand. Darüber hinaus unterstützt der PRACMA SURFACE die Selbstpräsentationsmöglichkeiten der Agenten.

3.4 Rollen-Transmutierbarkeit in PRACMA

Im Abschnitt 3.3, bei der Präsentation von PRACMA und seinen Agenten, wurde bei den betroffenen Agenten jeweils gezeigt, wie das System sich verhält, wenn es die Rolle des Käufers oder die des Verkäufers übernimmt. Die Transmutierbarkeit ist lokal bei den Agenten angesiedelt, durch jeweilige Unterscheidung nach der übernommenen Rolle. Zusammenfassend können die Agenten hinsichtlich der Rollen-Transmutierbarkeit in drei Gruppen unterteilt werden:

1. Die Module COMMENT AND QUESTION HANDLER (CQH) und DIALOG PLANNER sind für die Interpretation und Generierung der Systemäußerungen verantwortlich. Sie arbeiten auf die gleiche Art und Weise in beiden Rollen, verwenden aber unterschiedliches deklaratives Wissen zur Behandlung der rollenspezifischen Verarbeitung.
2. Die Module EVALUATION HANDLER und DOMAIN BELIEF HANDLER verarbeiten in beiden Rollen die Bewertungen bzw. das Wissen des Benutzers. Sie verwenden in beiden Rollen dieselben Formalismen, nämlich probabilistische Verarbeitung durch Bayessche Netze (Schäfer, 1997) bzw. modallogische Wissensrepräsentation (Hustadt & Nonnengart, 1993).

3. Die übrigen Module arbeiten auf die gleiche Art und Weise in beiden Rollen.

3.4.1 Transmutierbarkeit mit Hilfe bidirektionaler Planoperatoren

In diesem Abschnitt stellen wir eine alternative Realisierung der Transmutierbarkeit vor. In den früheren Entwicklungen von PRACMA wurde die Fähigkeit in einer Dialogsituation, verschiedene Rollen zu übernehmen, mit Hilfe sog. *bidirektionaler Dialogoperatoren* (Ndiaye & Jameson, 1994) erreicht. Dort wurden ein Aktions- und ein Dialogplaner benötigt, um den aufgaben- und zielorientierten Dialog zwischen dem Verkäufer und dem Käufer zu führen. Der Aktionsplaner steuerte das Sammeln von Fakten über die verschiedenen Attribute des Autos sowie die Preisverhandlungen. Der Dialogplaner steuerte hingegen das Dialogverhalten des Systems, wie die Auswahl der Sprechakte zur Überbeantwortung einer Frage, die Erkennung und Berücksichtigung der Dialogschritte des Partners. Beide Planungsprozesse interagierten: Wenn PRACMA z.B. die Rolle des Käufers übernommen hat, aktivierte der Aktionsplaner den Dialogplaner, um Fakten über das Auto, die es sich nicht anderweitig durch Ableitung aus dem Akteurmodell besorgen kann, vom Verkäufer zu erfragen. Der Dialogplaner eröffnete zuerst das Gespräch und veranlaßte den Generator, eine bestimmte Frage zu stellen. Im anderen Falle, wenn der Benutzer der Käufer ist, wird der Dialogplaner durch dessen Eingabe aktiviert und stößt seinerseits den Aktionsplaner an.

Wir diskutieren diese alternative Realisierung am Beispiel des Dialogplaners. Hierzu wurde eine Erweiterung des in POPEL verwendeten hierarchischen inkrementellen Dialogplaners eingesetzt (Reithinger, 1992). Die Idee der hierarchischen Planung ist es, zuerst im groben auf einer hohen Ebene zu planen und die Details, die sich aus den Teilzielen ergeben, später einzufügen. Die Planoperatoren wurden soweit wie möglich *bidirektional* formuliert. Abbildung 3.15 zeigt einen Teil eines bidirektionalen Planoperators. Der Operator hat drei Subziele (Subgoals):

- (1) den Dialog initialisieren
Dazu gehören: Den Dialogpartner begrüßen und Aufmerksamkeit erlangen z.B. durch konventionalisierte Sprechakte wie "Guten Tag", "Was kann ich für Sie tun?", "Ich möchte mich informieren."
- (2) die Verhandlungen über das Auto führen
Dazu gehören in Abhängigkeit der gespielten Rolle: Beschreibung von Merkmalen des Autos, Fragen über das Auto, Bewertung bzw. Anpreisen des Autos, Preisverhandlung mit Preisfestlegung und Argumentation. Die Schritte können solange wiederholt werden, bis sich der Käufer und der Verkäufer geeinigt haben oder keine Einigungsmöglichkeit mehr sehen.
- (3) den Dialog abschließen.
Neben der Wiederholung und Bestätigung der Vereinbarungen bzgl. Objekt und Preis gehört dazu die Verabschiedung des Dialogpartners z.B. durch konventionalisierte Sprechakte.

Die Variablen `?person1` und `?person2` enthalten die Informationen über die Rollen Käufer bzw. Verkäufer, die jeweils von den Dialogpartnern übernommen werden.

```
(define-plan-operator
:NAME negotiation-dialog
:GOAL
  (AM ((sb_facts am (:list (b believe all))
        (isa ?d dialog)(irole actor ?d ?person1)(irole counteractor ?d ?person2))))
:PRECONDITIONS
  ((AM ((assert_ind (:list (b believe all)) ?d dialog)
        (assert_ind (:list (b believe all)) ?d ?person1 actor)
        (assert_ind (:list (b believe all)) ?d ?person2 counteractor)
        (sb_facts am (:list (b (believe ?person1)(b want all))
        (isa ?d dialog)(irole actor ?d ?person1)(irole counteractor ?d ?person2))))))
:SUBGOALS
  ((AM ((sb_facts am (:list (b believe all))
        (isa ?init initialize)(irole actor ?init ?person1)
        (irole counteractor ?init ?person2)) *optional*))
  (AM ((sb_facts am (:list (b believe all))
        (isa ?n negotiate)(irole actor ?n ?person1)
        (irole counteractor ?n ?person2)(irole topic ?n ?negoitem))) *optional*)
  (AM ((sb_facts am (:list (b believe all))(isa ?f finish)
        (irole actor ?f ?person1)(irole counteractor ?f ?person2))) *optional*))
...)
```

Abbildung 3.15: Auszug aus einem bidirektionalen Plan-Operator.

(Jeder mit “(AM)” beginnende Ausdruck entspricht einer ASK-Nachricht, den der DIALOG PLANNER an die Komponente **A**ctor **M**odel, ein Agent zur Modellierung der Wünsche und Überzeugungen des jeweiligen Akteurs (Käufer oder Verkäufer) in der Sprache MOTEL sendet. Dieser Agent wurde in den späteren PRACMA-Entwicklungen durch den Agenten DOMAIN BELIEF HANDLER ersetzt.)

Der Vorteil der Verwendung bidirektionaler Planoperatoren statt zwei getrennter Pakete von Planoperatoren, eins für jede Rolle, liegt in erster Linie in der Repräsentation des Systemsplanungswissens, das konsistent und nicht redundant modelliert werden kann. Ferner kann jede vom System in einer Rolle verwendete Dialogstrategie auch in der anderen Rolle handhabbar sein. Allerdings sind die Planoperatoren nicht leicht erweiterbar und sie können komplexer und weniger handhabbar sein als zwei unidirektionale Planoperatoren. Es ist auch nicht einfach, den Unterschieden in den teilweise konfliktären Zielen des Käufers und des Verkäufers Rechnung zu tragen.

3.5 Realisierung lokaler AFLs

Bei der Antizipation der Verarbeitungsprozesse des Dialogpartners wird zwischen lokalen und globalen Antizipationsrückkopplungsschleifen unterschieden (siehe Abschnitt 2.6). Bei lokalen AFLs wird nur ein Teil des Systems eingesetzt, um ein Teil des Verstehens- bzw. des Verarbeitungsprozesses des Dialogpartners vorherzusagen. In einem Verkaufsgespräch geht es bei einer lokalen AFL z.B. um die Auswirkung eines Kommentars über die Bewertung des Verhandlungsobjekts.

Wenn PRACMA die Rolle des Verkäufers übernimmt, muß das System oft entscheiden, ob etwas über die Attribute des Autos gesagt werden kann, bzw. was gesagt werden soll. (Beispielsweise, ob das Auto über einen Tempomat verfügt.) Hierzu antizipiert das System die potentiellen Auswirkungen einer Äußerung über ein bestimmtes Merkmal des Autos auf die Verarbeitungsprozesse des Käufers \mathcal{K} . In PRACMA wurden zwei Arten von lokalen AFLs implementiert, eine einfache und schnelle bzw. eine komplexere, die aber auch zeitaufwendiger ist.

Bei der einfachsten lokalen AFL ruft das System die Prozeduren auf, die es in der Rolle des \mathcal{K} verwenden würde, um die Implikationen einer Äußerung auf den Bewertungsprozeß des Käufers vorherzusagen. \mathcal{V} fragt sich hierzu: "Welche Folgen hätte der von mir geplante Kommentar über ein Attribut des Autos auf *meine* Bewertung des Autos?"

Bei der komplexeren lokalen AFL berücksichtigt \mathcal{V} die möglichen Unterschiede zwischen seinem eigenen Wissen und Bewertungskriterien und denen des Käufers, mit dem \mathcal{V} verhandelt. Beispielsweise kann \mathcal{V} bei der Antizipation berücksichtigen:

- wie wahrscheinlich ist es, daß \mathcal{K} glaubt, daß das Auto einen Tempomat hat
- ob \mathcal{K} weiß, daß der Tempomat nicht nur Auswirkungen auf komfortables Fahren hat, sondern auch auf den Benzinverbrauch
- welche Wichtigkeit mißt \mathcal{K} den Bewertungsdimension "Komfort" und "Wirtschaftlichkeit" bei.

PRACMA setzt die einfachere lokale AFL ein, wenn das System die Rolle von \mathcal{V} übernimmt und nicht viel Zeit der Antizipation in einem bestimmten Dialogphase widmen will. Beide AFLs sind insofern lokal, als sie bei der Vorhersage der Implikationen eines Kommentars auf \mathcal{K} s Bewertung nur ein Teil der Interpretationsfähigkeiten des Systems verwenden.

Die angesprochenen lokalen AFLs zur Antizipation eines Kommentars \mathcal{C} auf den Käufer \mathcal{K} erfordern lediglich eine Kommunikation zwischen COMMENT AND QUESTION HANDLER und EVALUATION HANDLER (s. Abbildung 3.14). In der einfacheren Variante sendet CQH eine Nachricht an EVALUATION HANDLER: EVALUATION HANDLER wird gefragt, *inwieweit* würden die *eigenen* Bewertungen des Systems durch den Kommentar \mathcal{C} verändert. In der komplexeren Variante sendet CQH eine Meta-Level-Anfrage: EVALUATION HANDLER wird aufgefordert, auf der Meta-Ebene über die Bewertungsprozesse des \mathcal{K} nachzudenken.

3.6 Realisierung globaler AFLs

Durch lokale AFLs werden die Auswirkungen einer Äußerung auf den Dialogpartner vorhergesagt. Geht es jedoch darum, über diese Konsequenzen hinaus, den nächsten Dialogschritt des Interaktionspartners vorherzusagen, zeigen lokale Schleifen ihre Grenzen. Wenn das System nun in der Rolle von \mathcal{V} den Kommentar macht, daß das Auto einen Tempomat hat, kann \mathcal{K} z.B. folgendermaßen reagieren:

- (a) er kann schweigen und auf eine weitere Anmerkung des Verkäufers warten.
- (b) er kann eine bewertende Äußerung machen, wie "Das ist gut."
- (c) er kann eine weitere Frage stellen, die im Zusammenhang mit dem Attribut Tempomat steht oder ein anderes Merkmal des Autos betrifft.

Es kann für das System vorteilhaft sein, den nächsten Dialogschritt zu antizipieren. Das ist z.B. der Fall, wenn das System in der Rolle von \mathcal{V} vorhersagt, daß \mathcal{K} eine Frage über ein Merkmal des Autos stellen wird, die \mathcal{V} am liebsten vermeiden möchte. Beispielsweise, wenn dieses Merkmal ein Schwachpunkt des Autos ist. In diesem Falle könnte \mathcal{V} abwägen, ob er auf den ursprünglich geplanten Kommentar verzichtet oder ihn zumindest aufgrund der Antizipationsergebnisse abändert.

In PRACMA wurden durch die flexible Systemarchitektur und die Rollen-Transmutierbarkeit innerhalb des Systems zwei wichtige Mittel zur Realisierung globaler Antizipationsrückkopplung geschaffen (Ndiaye & Jameson, 1994). Die CHANNELS-Architektur ermöglicht, daß sich PRACMA in einer weiteren kompletten Instantiierung des Systems in der Rolle des Dialogpartners (Käufer oder Verkäufer) rekursiv aufrufen kann. Die Kommunikation zwischen den beiden Instantiierungen wird durch *externe Agenten* gesteuert.

Zur Realisierung der globalen Antizipation verwaltet der Agent GLOBAL ANTICIPATOR (abgekürzt als GAF für Global Anticipation Feedback) die mit der anderen Dialogrolle initialisierte *Unterinstantiierung* von PRACMA in einem separaten COMMONLISP-Prozeß (s. den rechten Teil von Abbildung 3.16). Die Unterinstantiierung ist nicht an der Interaktion mit dem Benutzer unmittelbar beteiligt. Sie dient lediglich der Verarbeitung der von GAF übergebenen Eingaben. Diese Eingaben sind die Äußerungen, die in der Hauptinstantiierung zur engeren Auswahl für die nächste Systemäußerung in Frage kommen. GAF stellt sicher, daß in der Unterinstantiierung ein, hinsichtlich der in der Hauptinstantiierung verfügbaren Informationen, realistisches Modell des Benutzers verwendet wird, und daß eine Anpassung des Dialogkontextes an die jeweils aktuelle Dialogsituation gewährleistet ist.

Die Aufgabe der Auswahl des nächsten Dialogzuges des Systems wird hierarchisch zwischen den Modulen DIALOG PLANNER und COMMENT AND QUESTION HANDLER aufgeteilt. Zunächst entscheidet der DIALOG PLANNER, ein inkrementeller Planer, welcher *Typ* von Dialogzug gemacht wird. Dazu berücksichtigt er verschiedene Faktoren wie den Dialogverlauf (repräsentiert im PRAGMATIC DIALOG MEMORY) und die Motivationsparameter des Systems (gespeichert im EGO). Sobald der Planer entschieden hat einen bestimmten Zugtyp zu machen, fragt er CQH nach einer geeigneten Anmerkung dieses Typs. Dazu verwendet CQH die Strategie der globalen Antizipation (s. Abbildung 3.17). Er bestimmt alle in der Dialogsituation möglichen Anmerkungen des betreffenden Typs sowie deren Nutzen. In PRACMA wird ein Bewertungsformular verwendet. Für jede potentielle Anmerkung wird der zu erwartende Nutzen mit den damit verbundenen Kosten verglichen. Die Anmerkungen sollen den Eindruck des Käufers über das Auto beeinflussen.

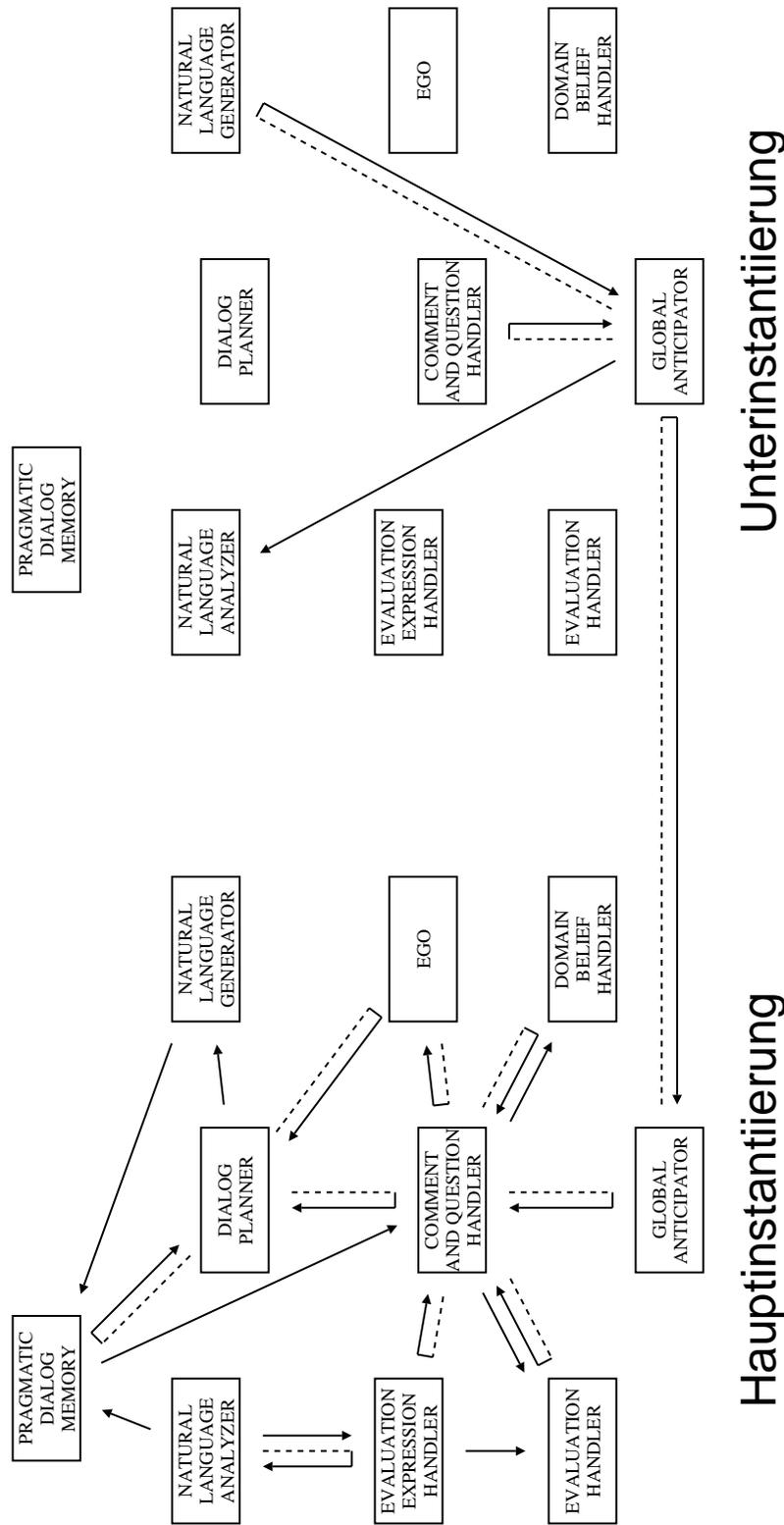


Abbildung 3.16: Die Architektur des PRACMA-Systems.

(Der linke Teil der Abbildung entspricht eine Instantiierung des PRACMA-System wie in Abbildung 3.2. Rechts ist die Unterinstantiierung zur Simulation der Rolle des Gesprächspartners. Diese Unterinstantiierung wird ausschließlich zum Zwecke der globalen Antizipation verwendet. Darin tauschen die Agenten Nachrichten der gleichen Typen wie in der Hauptinstantiierung aus sowie zusätzlich die im rechten Teil der Abbildung dargestellten Nachrichten.)

Dieser Eindruck wird bestimmt durch \mathcal{K} s vermutete Bewertung eines Attributs des Autos (z.B. “Wie wichtig ist eine Klimaanlage?”) und die Wahrscheinlichkeit, die der Käufer dessen Vorhandensein zuschreibt (z.B. “Hat das Auto eine Klimaanlage?”). Der Nutzen einer Anmerkung hängt daher von der Veränderung ab, die sie im Eindruck des Dialogpartners bewirkt. Für alle Anmerkungen, deren Nutzen über einem bestimmten Schwellwert δ liegen, werden zusätzlich die jeweiligen Nutzen der potentiellen Reaktionen des Käufers antizipiert (Ndiaye & Jameson, 1996b). Dazu schickt CQH diese Anmerkungen an das Modul zur globalen Antizipation (GAF). In jedem Durchlauf wird in der Unterinstantiierung unter Berücksichtigung des aktuellen Dialogkontextes und des Partnermodells die Rolle des Dialogpartners mit der übergebenen Äußerung als Eingabe simuliert.

```

function BEST-MOVE(type, constraints) returns a dialog move of type
type that fulfills constraints
  /* Executed by CQH */
  possible-moves  $\leftarrow$  ALLOWABLE-MOVES(type, constraints)
  for m in possible-moves do
    UTILITY[m]  $\leftarrow$  UTILITY-OF-MOVE(m)
  end
  reasonable-moves  $\leftarrow$  subset of possible-moves with UTILITY >  $\delta$ 
  if *global-anticipation?* = True then
    for m in reasonable-moves do
      UTILITY[m]  $\leftarrow$  UTILITY[m] + UTILITY-OF-ANTICIPATED-
        RESPONSE(m)
    end
  return the m in reasonable-moves with the highest UTILITY[m]

```

Abbildung 3.17: CQHs Algorithmus zur Auswahl eines Dialogzuges bei vorgegebenem Typ unter Verwendung der globalen Antizipation.

Kapitel 4

Antizipation der Käuferreaktion

Nachdem im letzten Kapitel die wesentliche Voraussetzung zur Realisierung der globalen Antizipation in PRACMA, nämlich die Transmutierbarkeit und die flexible Multi-Agenten-Architektur, und die technische Realisierung vorgestellt wurden, betrachten wir in diesem sowie in den Kapiteln 5 und 6 konkrete Fälle der Antizipation in einem Verkaufsgespräch. Das System spielt die Rolle des Verkäufers bzw. die des Käufers. Dabei wird jede Situation nach folgender Struktur besprochen:

- **Definition der Situation**
Hier werden die Annahmen dargelegt, die die besprochene Situation charakterisieren.
- **Wie kann die Situation entstehen?**
In diesem Abschnitt werden konkrete Situationen im Kontext von PRACMA beschrieben.
- **Angemessene Strategien**
Hier werden Ansätze präsentiert, wie das System die in dieser Situation beschriebenen Probleme lösen kann. (Die in diesem Abschnitt aufgeführten Strategien müssen nicht alle realisiert worden sein.)
- **Realisierung innerhalb von PRACMA**
Dieser Abschnitt beschreibt, wie die angemessenen Strategien innerhalb von PRACMA implementiert wurden.
- **Diskussion**
Hier werden die Stärken und Schwächen des Ansatzes sowie vergleichbare Ansätze bei anderen Autoren und alternative Varianten des Ansatzes besprochen.

4.1 Definition der Situation

In einem Dialogsystem kann die Rolle der Antizipationsrückkopplungsschleifen mittels *Entscheidungsbaumen* dargestellt werden. Nach Bühlmann *et al.* (1975, S. 18) ist ein Entscheidungsbaum ein Baum, dessen Knoten in drei disjunkte Mengen E , Z und B eingeteilt

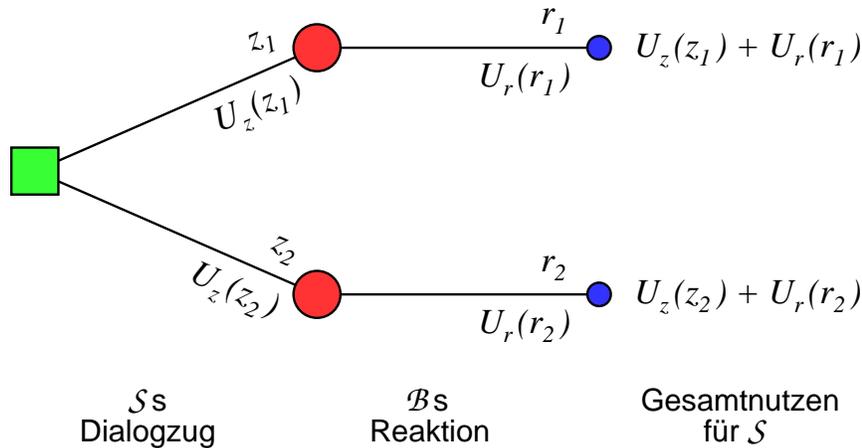


Abbildung 4.1: Entscheidungsbaum zur Illustration der Antizipation der Benutzerreaktion vor der Auswahl des nächsten Dialogzuges.

(S s Auswahl einer Äußerung durch CQH ist durch den *decision node* repräsentiert, die potentiellen Antworten des Benutzers (B) entsprechen den *chance nodes* und der Gesamtnutzen der Äußerung für S wird durch die *utility nodes* dargestellt. Die an den Kanten abgebildeten $U_z(z_i)$ bzw. $U_r(r_i)$ stehen für die jeweiligen Nutzen der Züge z_i bzw. r_i .)

werden können. Die Menge E und Z dürfen leer sein, die Menge B nicht. E heißt Menge der Entscheidungsknoten, Z Menge der Zufallsknoten und B die Menge der Blätter. Die Unterbäume eines Entscheidungsbaums sind wiederum Entscheidungsbäume.

Abbildung 4.1 stellt AFLs als Entscheidungsbaum dar. Dabei ist ein *Entscheidungsknoten* (durch ein kleines Quadrat bezeichnet) ein Punkt, in dem der Entscheider (das System) sich für eine der möglichen Handlungsalternativen (Aktionen, Dialogzüge) entscheiden muß. Die durch kleine Kreise bezeichneten Knoten sind die *Zufallsknoten* (*chance nodes*), an denen sich der Dialogpartner (die Welt, der Gegenspieler) für einen Dialogzug entscheiden muß. Allen von diesem Knoten ausgehenden Kanten wird die gleiche Wahrscheinlichkeit zugeordnet (s. die spätere Diskussion). Die durch kleinere Kreise dargestellten Endknoten (*utility nodes*, *Endknoten*) sind Knoten, an denen das Endergebnis der potentiellen Entscheidungsfolgen ermittelt wird. Eine Entscheidungsfolge ist eine von dem Entscheidungsknoten ausgehenden Kante, die bis zu dem Endknoten hinführt.

Das System (S) hat zwischen verschiedenen Dialogzügen $z_1 \dots z_n$ zu wählen, die an den Benutzer (B) adressiert werden könnten. Jeder Dialogzug z_i hat einen bestimmten Nutzen (Utility) $U_z(z_i)$ für S . Dieser Nutzen ist eine Quantifizierung der unmittelbaren Vorteile eines Zuges für S . Anstatt den Dialogzug mit dem höchsten Nutzen $U_z(z_i)$ auszuwählen, antizipiert S für jeden der z_i die Reaktion r_i , die der Benutzer B wahrscheinlich als Antwort auf den Dialogzug z_i zeigen wird; und jede dieser potentiellen Reaktionen hat wiederum einen bestimmten Nutzen $U_r(r_i)$ für S . S wählt dann letztendlich den Dialogzug mit dem höchsten Gesamtnutzen $U_z(z_i) + U_r(r_i)$.¹ So wird eine AFL für jeden Dialogzug z_i aufgerufen, um

¹Das Zusammenaddieren beruht auf der Annahme, daß die beiden Typen vom Nutzen unabhängig sind.

die Reaktion r_i von \mathcal{B} vorherzusagen. Dabei wird angenommen, daß die Berücksichtigung von $U_r(r_i)$ zusätzlich zu $U_z(z_i)$ das System \mathcal{S} bei der Auswahl des nächsten Dialogzuges beeinflussen kann. Hierzu reflektiert \mathcal{S} zur Ermittlung der voraussichtlichen Reaktion bzw. Interpretation des Dialogpartners wie folgt: “Wie würde ich auf diesen Dialogzug reagieren, wenn ich in der Lage von \mathcal{B} wäre?” bzw. “Wie würde der nächste Dialogzug mein Modell von \mathcal{B} beeinflussen, wenn ich in seiner Lage wäre?”. Dazu instantiiert sich \mathcal{S} rekursiv in der Rolle von \mathcal{B} .

4.2 Wie kann die Situation entstehen?

Wenn das System die Rolle des Verkäufers (\mathcal{V}) übernimmt, kommt es vor, daß es z.B. nach der Beantwortung einer Frage des Käufers (\mathcal{K}) eine spontane Anmerkung über das Auto machen will. Es könnte beispielsweise das Auto durch die Erwähnung der neuen Reifen anpreisen. Dies könnte aber den Käufer dazu animieren, eine diesem Thema naheliegende Frage, nämlich die nach dem Kilometerstand, zu stellen. Da das System einen genauen Kenntnisstand über den Zustand des Autos hat, wird es dann auf diese Anmerkung verzichten, falls der Kilometerstand einen Schwachpunkt des Autos darstellt. Das System simuliert also in diesem Fall die Verarbeitungsprozesse des Käufers. Dazu verwendet GAF in der Unterinstantiierung ein Modell des Käufers mit denselben Bewertungskriterien und Dialogstrategien, wie \mathcal{S} sie einsetzen würde, wenn \mathcal{S} ein Auto kaufen würde.

4.3 Angemessene Strategien

Eine einfache mögliche Strategie zur Antizipation der Reaktionen des Dialogpartners besteht darin, bei der Simulation der Verarbeitungsprozesse des Dialogpartners *sich selbst als Modell* zu nehmen (Ndiaye & Jameson, 1995). \mathcal{S} geht davon aus, daß der Dialogpartner dieselben Bewertungskriterien, Wissen und Dialogstrategien wie \mathcal{S} verwenden würde, wenn das System ein Auto kaufen sollte. \mathcal{S} geht davon aus, daß der Dialogpartner sich genau so verhält, wie \mathcal{S} das in einer entsprechenden Situation tun würde. In dieser Strategie (genannt *situational role taking* in Higgins, 1981) sagt \mathcal{S} die Antworten des Gesprächspartners voraus, indem \mathcal{S} sich die Frage stellt: “Wie würde ich auf diese Äußerung reagieren, wenn ich in der Lage meines Dialogpartners wäre?”. Die Anwendung einer solchen Strategie resultiert aus der Ähnlichkeit zu dem Dialogpartner. In diesem Zusammenhang unterscheidet Higgins (1981) zwischen *assumed similarity* und *inferred similarity*. Beim ersteren handelt es sich quasi um eine egozentrische Sichtweise. Der Handelnde berücksichtigt nicht oder nicht genügend die Möglichkeit, daß er sich in einigen Aspekten vom Dialogpartner unterscheiden kann. Er kann aber annehmen, daß die Unterschiede zu dem Dialogpartner nicht so groß bzw. relevant sind, als daß sie berücksichtigt werden sollten. Beim *inferred similarity* nimmt der Handelnde an, daß, gerade weil der Gesprächspartner ähnliche Eigenschaften wie er selbst hat, auch ähnliche Antworten geben wird. Die Methode von *situational role taking* eignet sich, wenn die Interessen der Dialogpartner sich deutlich unterscheiden. So könnte ein an Komfort interessiertes \mathcal{S} eine Anmerkung über die Klimaanlage des Auto machen.



Abbildung 4.2: Gegenüberstellung des Interessenprofils in der Autodomäne von S (links) mit dem Interessenprofil (rechts), das S 's EVALUATION HANDLER dem jetzigen Käufer zuschreibt.

Ein an Umweltfreundlichkeit interessierter Käufer würde dieses Attribut des Autos eventuell negativ bewerten.

Im Gegensatz zu *situational role taking* gibt es auch *individual role taking*. Dort wird angenommen, daß beide Dialogpartner sich (u.U. sogar radikal) in ihren individuellen Eigenschaften (etwa Interessen, Überzeugungen, Einstellungen) unterscheiden (s. Abbildung 4.2). Hierzu ermittelt S zunächst über das Benutzermodell die Parameter, die den Dialogpartner charakterisieren. Gegebenenfalls kann S die Parameter aus Erwartungswerten zusammensetzen. Entscheidend für die Strategie *individual role taking* ist es, ein realistisches Modell des Dialogpartners zu ermitteln. Die Unterinstantiierung zur Simulation der Verarbeitungsprozesse wird dann mit den Parametern angepaßt, die den Dialogpartner charakterisieren.

4.4 Realisierung innerhalb von PRACMA

Abbildung 4.3 gibt die Algorithmen wieder, wie die jeweiligen Nutzen der antizipierten Reaktionen des Käufers antizipiert werden können. Der Gesamtnutzen ergibt sich aus der Summe des Nutzens des betrachteten Dialogzuges und des antizipierten Nutzens der Antwort des Käufers auf diesen Dialogbeitrag. S wählt dann den Zug mit dem höchsten Nutzen. (Zur Erinnerung: $UTILITY[m] := UTILITY[m] + UTILITY-OF-ANTICIPATED-RESPONSE(m)$ in BEST-MOVE aus Abbildung 3.17).

Die potentiellen Reaktionen des Käufers auf die Dialogzüge des Verkäufers werden dabei wie folgt bewertet:²

- Fragen nach einem Attribut:

Der Nutzen einer Frage nach einem Merkmal des Autos wird gemäß den CQH-Kriterien wie im letzten Kapitel dargestellt.

²Im Algorithmus ist bei UTILITY-OF-RESPONSE der Fall mit aufgeführt, wenn das System die Reaktionen des Verkäufers antizipieren muß. Dieser Fall wird erst im Kapitel 6 besprochen.

```

function UTILITY-OF-ANTICIPATED-RESPONSE (move) returns a
utility
/* Executed by CQH */
return UTILITY-OF-RESPONSE(ASK(GAF, "ANTICIPATED-
RESPONSE(move"))

```

```

function ANTICIPATED-RESPONSE(move) returns an anticipated
response to move
/* Executed by GAF */
INFORM(NATURAL LANGUAGE ANALYZER, "UTTERANCE-
INTERPRETED(move)")
return ASK(NATURAL LANGUAGE GENERATOR, "INTERNAL-
REPRESENTATION(LATEST-UTTERANCE())")

```

```

function UTILITY-OF-RESPONSE(response) returns a utility
/* Executed by CQH */
case TYPE[response]
  Responses possible in both roles:
  silence:
    return 0
  ...
  Possible responses from buyer:
  question-about-attribute:
    return UTILITY-OF-MOVE(BEST-MOVE("comment-on-
attribute", "(topic = TOPIC[response]"))
  evaluative-reaction:
    return POSITIVENESS[response]
  ...
  Possible responses from seller:
  comment-on-attribute:
    return ASK(EVALUATION HANDLER, "RESULTING-
UNCERTAINTY-REDUCTION(response)")
  ...

```

Abbildung 4.3: Algorithmen zur globalen Antizipation.

(Die interne Repräsentation S_s geplanter Äußerung wird in der Unterinstantiierung verwendet, um die potentiellen Antworten des Benutzers durch Simulation von dessen Rolle vorherzusagen. UTILITY-OF-RESPONSE bewertet dann den Nutzen der potentiellen Antworten aus der Sicht von S_s .)

- **Bewertende Äußerung:**
Der Nutzen hiervon ergibt sich daraus, wie positiv die vom Käufer gezeigte Reaktion ist.
- **Schweigen:**
Der Nutzen von Schweigen als Reaktion auf eine Äußerung wird mit 0 bewertet.

In der Rolle des Verkäufers \mathcal{V} schätzt das System durch das Modul CQH zu jedem Merkmal des Autos (z.B. Benzinverbrauch) ein, ob es sich lohnt, zu diesem Attribut eine Anmer-

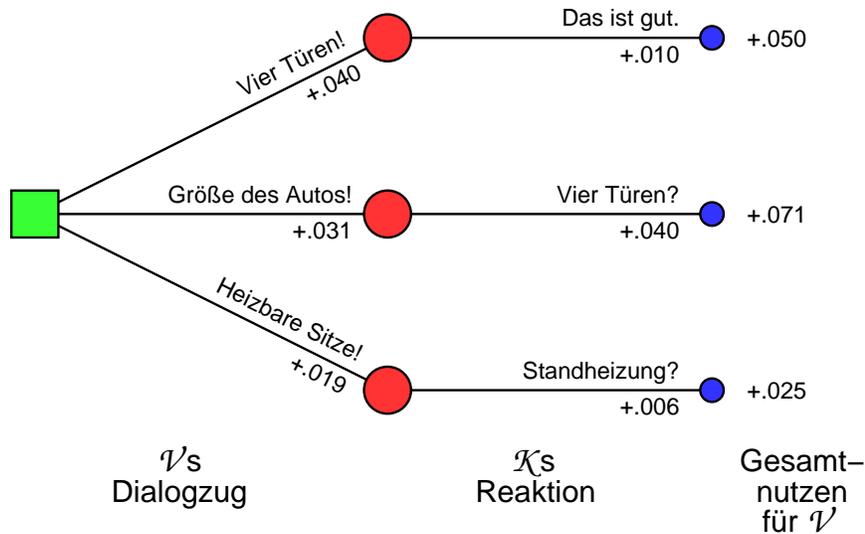


Abbildung 4.4: Ein Beispiel für die Antizipation der Käuferreaktion.

(Die unter den Äußerungen abgebildeten Zahlen drücken den Nutzen der entsprechenden Äußerungen für das System aus. Rechts jeden Pfades im Entscheidungsbaum steht der Gesamtnutzen als Summe.)

kung zu machen. CQH ermittelt dazu für jedes Attribut den zu erwartenden Nutzen. Beispielsweise würde CQH in der Situation des Beispieldialoges von Abbildung 3.9 aufgrund der jeweiligen Nutzen und ohne Verwendung der globalen Antizipation eine Anmerkung über das Attribut “vier Türen” auswählen (s. Abbildung 4.4). Durch die zusätzliche Berücksichtigung der potentiellen Reaktionen von \mathcal{K} erweist sich eine Anmerkung von \mathcal{S} über die Größe des Autos als lohnender, da der errechnete Gesamtnutzen für dieses Attribut höher liegt als der jeweilige Gesamtnutzen von “vier Türen” und “heizbare Sitze”.

4.5 Diskussion

4.5.1 Stärken und Schwächen dieses Ansatzes

\mathcal{S} antizipiert genau einen nächsten Dialogzug des Käufers. \mathcal{S} könnte auch weitere Züge vorausschauen. Dies würde den Entscheidungsbaum von Abbildung 4.1 weiter expandieren. Jedoch muß \mathcal{S} irgendwann die Vorausschau beenden, um seinen Dialogzug auszuführen. Die Durchführbarkeit einer solchen Antizipation wird in Abschnitt 7.7 erörtert.

Bei der Vorhersage der Reaktionen des Käufers kann es sein, daß ein Dialogzug mit einem hohen Nutzen nach der Antizipation bei der Berücksichtigung des Gesamtnutzens nicht gemacht wird. Grund hierfür kann es sein, daß die Erwähnung des zugehörigen Merkmals zu einer ungünstigen Reaktion des Käufers führt. (Beispielsweise kann die Erwähnung der neuen Reifen eines Autos zu einer Frage über den Kilometer-Stand führen.) Wenn das

entsprechende Merkmal des Autos aber deutlich hervorragend ist, kann \mathcal{S} dennoch die Äußerung gefolgt von einer anderen Anmerkung machen, um den Dialogfokus zu verschieben.

4.5.2 Alternative Varianten dieses Ansatzes

In den Fällen, in denen \mathcal{S} kein adäquates Modell vom Käufer hat, kann es sich dazu entscheiden, einige Extremfälle zu betrachten. Diese Extremfälle entsprechen starken Interessen in den jeweiligen Bewertungsdimensionen. So kann \mathcal{S} die Reaktionen des Käufers mit einer Unterinstantiierung realisieren, bei der ein extrem an Umweltfreundlichkeit interessierter Käufer simuliert wird. Die zugrundeliegende Annahme ist, daß \mathcal{S} dadurch Dialogbeiträge vermeiden kann, die einem solch radikalen Käufer mißfallen könnten.

Behandlung von Unsicherheit bei der globalen Antizipation

5.1 Definition der Situation

Die Verwendbarkeit eines Entscheidungsbaumes wie in Abbildung 4.1 (Seite 78) setzt voraus, daß \mathcal{S} zu jedem Zug z_i die exakte Reaktion r_i von \mathcal{B} präzise vorhersagen kann. Der allgemeinere Fall, der der Unsicherheit von \mathcal{S} über das Verhalten von \mathcal{B} Rechnung trägt, ist in Abbildung 5.1 dargestellt. Zu jedem Zug z_i bestimmt \mathcal{S} eine Menge potentieller Reaktionen $\{r_{ij}\}$. Im Falle globaler AFLs wird diese Unsicherheit durch mehrere Faktoren beeinflusst, die \mathcal{S} nicht genau bekannt sind. \mathcal{S} weiß dann nicht exakt, wie es die Rolle von \mathcal{B} aufgrund seiner Unsicherheit simulieren kann. Dies wirft die Frage der Unsicherheitsbehandlung auf.

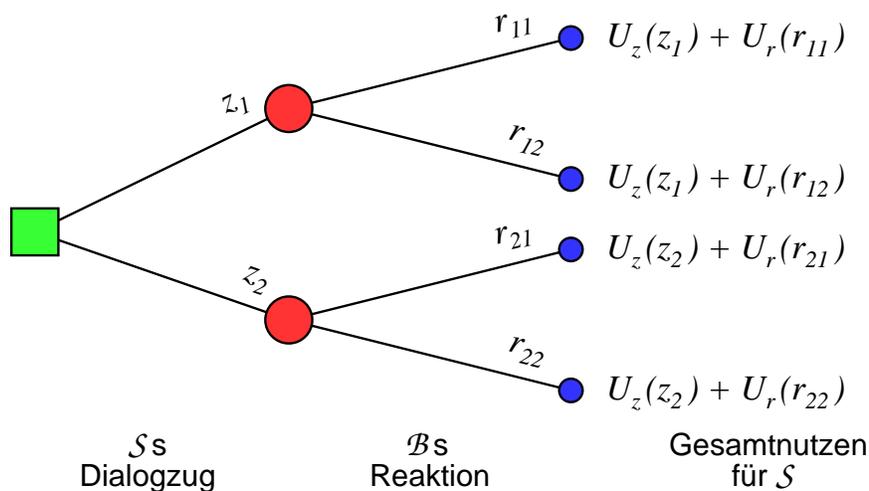


Abbildung 5.1: Verallgemeinerung des Entscheidungsbaums in Abbildung 4.1 (Seite 78) unter Berücksichtigung von Unsicherheit von \mathcal{S} über die Reaktionen von \mathcal{B} .

5.2 Wie kann die Situation entstehen?

Die im vorherigen Beispiel vorgestellte Strategie berücksichtigt für jede geplante Äußerung genau eine potentielle Reaktion des Dialogpartners. Die verwendeten Algorithmen setzen voraus, daß die Folgereaktion des Käufers stets korrekt und präzise antizipiert wird. Diese Annahme trägt aber der Unsicherheit des Systems über das tatsächliche Dialogverhalten und die Präferenzen des Käufers keine Rechnung. Da es verschiedene Quellen für die Unsicherheit des Systems über den Dialogpartner gibt, wird die verfolgte Strategie dahingehend geändert, daß für jede geplante Äußerung statt einer einzigen Reaktion eine Menge möglicher Reaktionen berücksichtigt wird.

5.3 Angemessene Strategien

Bei diesem Ansatz — *runner-up strategy* genannt (Ndiaye & Jameson, 1996b) — reflektiert \mathcal{S} wie folgt: “Welche *Reaktionen* würde ich *in Betracht ziehen*, wenn ich in der Lage des Dialogpartners wäre?”. Diese Strategie legt folgende Überlegung zugrunde: “Auch wenn ich Unsicherheit über die Einstellungen meines Interaktionspartners habe, so unterscheidet er sich wahrscheinlich nicht so sehr von mir, als daß er einen Zug auswählt, der nicht unter meinen N vorausgesagten Zügen ist”.¹ Diese Strategie beruht auf folgender Annahme: Die vorhergesagten Züge, deren Nutzen über einem bestimmten Schwellwert ε liegen, sind diejenigen, die beim Käufer am wahrscheinlichsten sind.

5.4 Realisierung innerhalb von PRACMA

Die Grundidee dieser Strategie ist die Nutzung der Vorgehensweise von CQH bei der Auswahl der nächsten Äußerung eines von DIALOG PLANNER vorgegebenen Typs. Im vorherigen Fall wurde für jeden Zug vom \mathcal{S} in der Unterinstantiierung genau die Anmerkung mit dem höchsten Nutzen zurück geliefert. Da CQH zur Ermittlung der besten Anmerkung sämtliche dem Dialogkontext angemessenen Alternativen berücksichtigt, können hier nun alle Anmerkungen aus der Unterinstantiierung, deren Nutzen über einem bestimmten Schwellwert ε liegen, an die Hauptinstantiierung zurück geliefert werden. Abbildung 5.2 stellt die dazugehörigen Algorithmen dar.

Abbildung 5.3 zeigt die Ergebnisse zu dem Beispiel aus Abbildung 4.4 unter Berücksichtigung der Unsicherheit über die Bewertungskriterien und Inferenzen des Käufers. Im Gegensatz zur einfachen Variante (Abbildung 4.4) zeigt sich jetzt die Anmerkung über die Größe des Autos nicht mehr als die günstigste, da der Käufer danach eine Frage über den Benzinverbrauch stellen könnte, der einen Schwachpunkt des Autos darstellt.

¹Vgl. die Annahme im Abschnitt 2.6 über die Ähnlichkeit der Verarbeitungsprozesse des Systems mit denen des Dialogpartners.

```

function UTILITY-OF-ANTICIPATED-RESPONSE (move) returns a
utility
/* Executed by CQH */
possible-utilities ← set of utilities obtained by applying UTILITY-
OF-RESPONSE to the result yielded by ASK(GAF, “ANTICIPATED-
RESPONSES(move)”)
return AVERAGE(possible-utilities)

```

```

function ANTICIPATED-RESPONSES(move) returns a set of possible
responses to move
/* Executed by GAF */
INFORM(NATURAL LANGUAGE ANALYZER, “UTTERANCE-
INTERPRETED(move)”)
return ASK(CQH, “BEST-MOVES-CONSIDERED( $\epsilon$ )”)

```

```

function BEST-MOVES-CONSIDERED( $\epsilon$ ) returns a set of moves
/* Executed by CQH */
reasonable-moves ← the set of reasonable moves considered during the
most recent execution of BEST-MOVE
best-move ← the m in reasonable-moves with the highest UTILITY[m]
return the set of all m in reasonable-moves such that
UTILITY[m] ≥ UTILITY[best-move] −  $\epsilon$ 

```

Abbildung 5.2: Verallgemeinerung der Algorithmen zur globalen Antizipation (von Abbildung 4.3) durch Berücksichtigung der Unsicherheit des Verkäufers über die Käuferreaktion. (In ANTICIPATED-RESPONSES erhält GAF statt nur einer Antwort vom NATURAL LANGUAGE GENERATOR von CQH mehrere Antwortmöglichkeiten vom Käufer. Diese müssen auf jeden Fall von CQH ermittelt werden. Falls die Wahrscheinlichkeiten in ANTICIPATED-RESPONSES(*move*) geschätzt werden können, gehen die Schätzungen in die Berechnung von UTILITY-OF-ANTICIPATED-RESPONSE ein (s. Abschnitt 5.5.2): Hier wird lediglich der Mittelwert von *possible-utilities* als Schätzung verwendet.)

5.5 Diskussion

5.5.1 Stärken und Schwächen dieses Ansatzes

Die Berücksichtigung der Unsicherheit über den Käufer erhöht die Genauigkeit der Antizipation. Diese Vorgehensweise erhöht jedoch nicht wesentlich die Kosten, da sämtliche Züge als Nebenprodukt der Äußerungsauswahl durch CQH ohnehin betrachtet werden.

Ein weiterer Vorteil dieser Strategie liegt in der Möglichkeit für die Wahl der Schwellwerte. Durch ein geeignetes Setzen der Schwellwerte (δ bzw. ϵ) für die Berücksichtigung der Anmerkungen in der Haupt- bzw. Unterinstantiierung wird die Verhandlungsstrategie des Systems variiert. So kann es z.B. im Laufe eines Dialoges vorsichtiger werden als am Anfang.

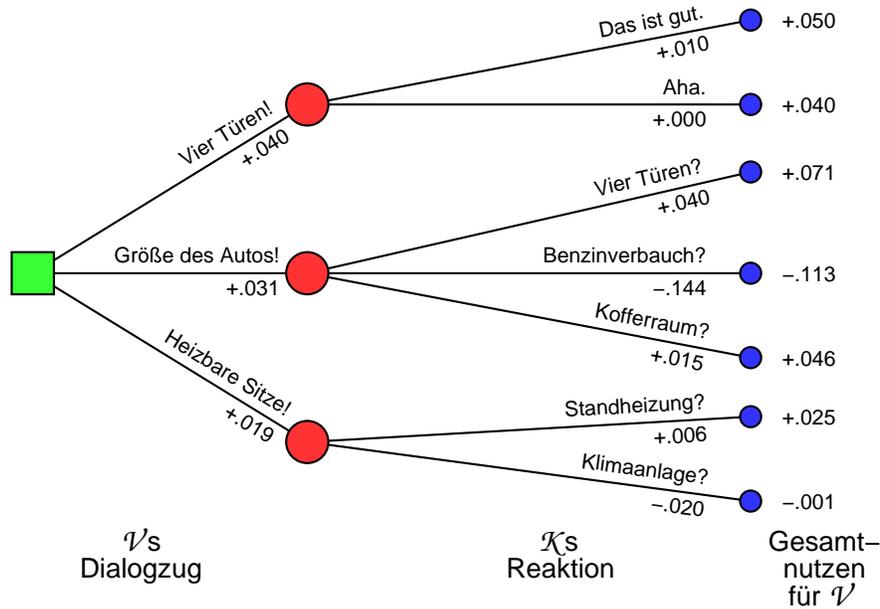


Abbildung 5.3: Ein Beispiel für die Antizipation der Käuferreaktion unter Berücksichtigung von Unsicherheit über die Inferenzen bzw. Bewertungskriterien des Käufers.

(Die unter den Äußerungen abgebildeten Zahlen drücken den Nutzen der entsprechenden Äußerungen für das System aus. Rechts jeden Pfades im Entscheidungsbaum steht der Gesamtnutzen als Summe.)

5.5.2 Alternative Varianten dieses Ansatzes: Der probabilistische Ansatz

Überblick über den Ansatz. In den in diesem Kapitel bis jetzt betrachteten Situationen der Antizipation unter Berücksichtigung von Unsicherheit wurden zu jedem vom System geplanten Dialogzug statt einer einzigen Reaktion eine Menge potentieller Reaktionen untersucht. Allerdings wurde bis jetzt davon ausgegangen, daß die betrachteten potentiellen Reaktionen des Dialogpartners alle gleich wahrscheinlich sind. (So wurde dann wie in der Funktion UTILITY-OF-ANTICIPATED-RESPONSE der Abbildung 5.2 der Mittelwert zur Ermittlung des Nutzens des Zufallsknotens verwendet.) Diese Vorgehensweise trägt jedoch nicht der Tatsache Rechnung, daß, auch wenn der Dialogpartner nicht als Gegner im spieltheoretischen Sinne angesehen wird, seine Reaktionen nicht alleine dem Zufall überlassen werden. Manche Reaktionen sind, abhängig vom Dialogkontext und -zustand, wahrscheinlicher als andere.

Formalisierung und Bewertung. Die sich ergebende Situation ist in Abbildung 5.4 wiedergegeben. Zu jedem vom System in Erwägung gezogenen Dialogzug m_i betrachten wir die Menge der antizipierten Reaktionen r_{ij} des Dialogpartners mit den entsprechenden Wahrscheinlichkeiten (s. Abbildung 5.5). $p(r_{ij})$ gibt an, mit welcher Wahrscheinlichkeit das System \mathcal{S} annimmt, daß der Dialogpartner (\mathcal{B}) die Reaktion r_{ij} zeigen wird. (Bis jetzt wurde lediglich angenommen, daß $p(r_{ij}) = \frac{1}{|r_{ij}|}$).

Diese probabilistische Sichtweise führt zu einem neuen Entscheidungsprozeß an den Zu-

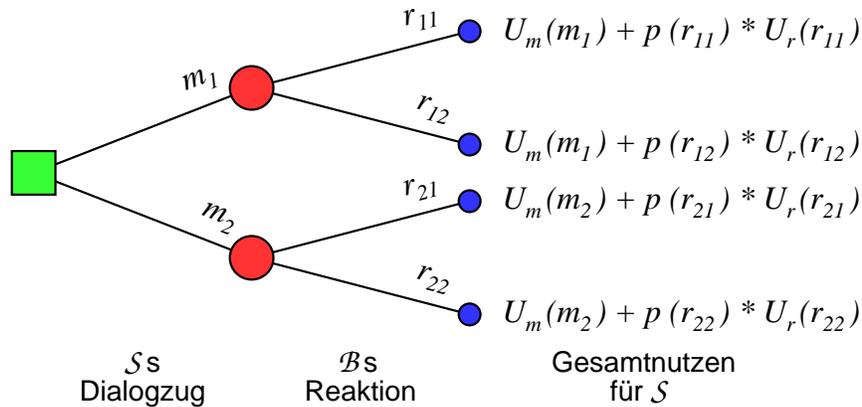


Abbildung 5.4: Entscheidungsbaum zur Illustration der Antizipation der Reaktionen des Dialogpartners unter Berücksichtigung der Wahrscheinlichkeiten, die den potentiellen Reaktionen des Dialogpartners beigemessen werden.

```

function UTILITY-OF-ANTICIPATED-RESPONSE-PROBABILISTIC
(move) returns the expected utility
/* Executed by CQH */
possible-responses-and-probabilities ← the set corresponding to the
result yielded by ASK(GAF, "ANTICIPATED-RESPONSE-WITH-
PROBABILITY(move)")
for each  $r_i, p(r_i)$  in possible-responses-and-probabilities do
    UTILITY[ $r_i$ ] ←  $p(r_i) * UTILITY-OF-RESPONSE(r_i)$ 
end
return  $\sum_i UTILITY(r_i)$ 

```

```

function ANTICIPATED-RESPONSE-WITH-PROBABILITY(move) re-
turns an anticipated response to move with the corresponding probability
/* Executed by GAF */
INFORM(NATURAL LANGUAGE ANALYZER, "UTTERANCE-
INTERPRETED(move)")
predicted-move ← ASK(NATURAL LANGUAGE GENERATOR,
"INTERNAL-REPRESENTATION(LATEST-UTTERANCE())")
return (predicted-move, PROBABILITY(predicted-move))

```

Abbildung 5.5: Algorithmus zur Ermittlung des Gesamtnutzens eines Dialogzugs nach dem probabilistischen Ansatz.

(Dieser Algorithmus berücksichtigt zu jeder potentiellen Reaktion des Dialogpartners die dazugehörige Wahrscheinlichkeit und bildet den Nutzen des Zufallsknotens als gewichtete Summe der Nutzen dieser Reaktionen.)

fallsknoten und somit am Entscheidungsknoten. Diesem Prozeß liegt das *Maximum Expected Utility* (Russell & Norvig, 1995, S. 419 und 472) zugrunde. Hierzu betrachten wir $U_{\text{GAF}}(m_i)$ den Erwartungswert des Gesamtnutzens nach Anwendung der globalen Antizi-

pation

$$U_{\text{GAF}}(m_i) = U(m_i) + \sum_j p(r_{ij}) * U(r_{ij}).$$

\mathcal{S} wählt dann den Zug m_i aus, der den erwarteten Gesamtnutzen maximiert.

Diskussion. Die den antizipierten Reaktionen des Dialogpartners zugeschriebenen Wahrscheinlichkeiten hängen von der Anzahl der antizipierten Reaktionen ab und somit von dem Schwellwert ε in der Unterinstantiierung. (Zur Erinnerung: Es werden in der Unterinstantiierung diejenigen Züge, deren Nutzen über einem bestimmten Schwellwert ε liegen, als diejenigen betrachtet, die beim Käufer am wahrscheinlichsten sind.) Daher kommt der Wahl dieses Schwellwerts eine große Bedeutung zu.

Kapitel 6

Antizipation des Verkäuferverhaltens

6.1 Definition der Situation

In der Rolle des Verkäufers — wie in den beiden vorherigen Fällen — kennt das System den genauen Zustand des Autos und schätzt, wie der Käufer die Attribute des Autos bzgl. der Bewertungsdimensionen bewertet. So baut es in dieser Rolle ständig Eindrücke über die Bewertungen des Käufers auf. Diese Eindrücke beeinflussen sein Verhalten als Verkäufer.

6.2 Wie kann die Situation entstehen?

Wenn das System nun die Rolle des Käufers \mathcal{K} übernimmt, macht der Verkäufer oft konkrete Angaben zum Auto, die \mathcal{K} im einzelnen nicht antizipieren kann. Deswegen kann es für das System in der Rolle des Käufers nicht darum gehen — wie in den vorherigen Fällen —, die nächste Äußerung des Dialogpartners zu antizipieren. Vielmehr ist der Käufer daran interessiert, wie seine nächste Äußerung die Eindrücke manipulieren kann, die der Verkäufer von ihm hat. Dazu stellt sich das System in der Rolle von \mathcal{K} die Frage: “Wie würde meine nächste Äußerung meine Eindrücke vom Käufer beeinflussen, wenn ich sie als Verkäufer in dieser Dialogsituation hören würde?”. Zum Beispiel kann \mathcal{K} denken, daß \mathcal{V} ihn nach einer Frage über eine Klimaanlage oder nach einer positiven bewertenden Äußerung auf die Nennung des Attributs “Klimaanlage” hin als einen an Komfort interessierten Käufer einschätzen wird. \mathcal{V} könnte sich dann dazu ermutigt fühlen, mehr über die Aspekte des Autos zu sagen, die die Bewertungsdimension “Komfort” betreffen. Wenn der Käufer diese Verschiebung des Eindrucks, den \mathcal{V} von ihm hat, als nützlich betrachtet, kann er die geplante Äußerung machen. Wenn \mathcal{K} diese Verschiebung nicht wünscht, kann er beispielsweise entscheiden, die Frage über die Klimaanlage lieber zu verschieben, bis der Verkäufer ein genaueres Modell über seine Bewertungsstandards hat und ihn nicht aufgrund der Frage als einen an Komfort interessierten Käufer erkennt.

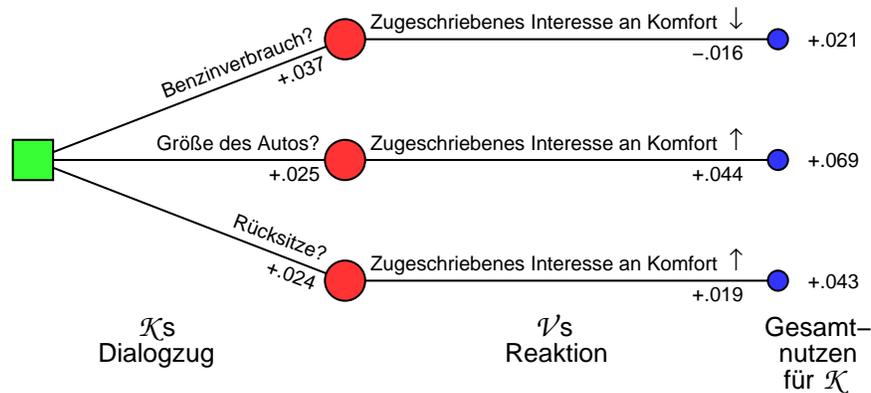


Abbildung 6.1: Ein Beispiel für die Antizipation der Verkäuferreaktion.

(Bei der Antizipation der Reaktionen von \mathcal{V} drücken die Zahlen den Nutzen für das System in der Rolle von \mathcal{K} aus. Eine positive (bzw. negative) Zahl besagt, daß nach der entsprechenden Äußerung von \mathcal{K} das Interesse an Komfort, das \mathcal{V} ihm zuschreibt, größer (bzw. geringer) wird.)

6.3 Angemessene Strategien

In der Rolle des Verkäufers unterhält PRACMA ein Modell der Eindrücke, die der Käufer vom Auto hat und wählt seine Äußerungen so, daß sie bestimmten Veränderungen der Eindrücke beim Käufer herbeiführen. Diese Eindrücke sind Einschätzungen von \mathcal{K} 's Interesse an den verschiedenen Bewertungsdimensionen. \mathcal{K} 's Frage nach einem Attribut oder \mathcal{K} 's bewertende Äußerung nach der Erwähnung eines Attributs läßt in der Regel auf Interesse bzw. Desinteresse in den einzelnen Bewertungsdimensionen schließen.¹

Wenn \mathcal{S} nun in der Rolle des Käufers \mathcal{K} die Reaktion des Verkäufers \mathcal{V} vorhersagen will, simuliert es die Verarbeitungsprozesse von \mathcal{V} und schätzt \mathcal{V} 's Eindrücke über seine Interessen in den einzelnen Bewertungsdimensionen vor und nach einer Äußerung ein. Für jede potentielle Äußerung antizipiert \mathcal{K} , ob diese Äußerung bei \mathcal{V} die Interesseneinschätzung in der Hauptbewertungsdimension erhöhen wird, die \mathcal{V} ihm zuschreibt. Der Nutzen der Antizipation der Verkäuferreaktion ist eine Funktion der bei \mathcal{V} induzierten Eindrucksverschiebung über die Interessen von \mathcal{K} .

6.4 Realisierung innerhalb von PRACMA

In der Rolle von \mathcal{K} reflektiert das System durch das Modul CQH zu jeder von \mathcal{V} geäußerten Anmerkung, ob es sich lohnt, dazu einen Kommentar (z.B. als bewertende Äußerung) abzugeben oder ob eine weitere Frage über ein bestimmtes Attribut nützlicher ist. Dazu ermittelt CQH zu jeder bewertenden Äußerung bzw. zu jedem Attribut den zu erwartenden Nutzen. In dem in Abbildung 6.1 dargestellten Beispiel würde CQH ohne Antizipation

¹Zur Erinnerung: Jedes Attribut im Bewertungsformular hat Implikationen für eine oder zwei der sechs Bewertungsdimensionen (Schäfer, 1997). So hat z.B. das Attribut "Klimaanlage" eine positive Auswirkung auf die Dimension "Komfort" und eine negative auf "Umweltfreundlichkeit".

in der Situation des Beispieldialoges aufgrund der jeweiligen Nutzen eine Frage nach dem Benzinverbrauch höher bewerten als Fragen nach der Größe des Autos bzw. nach den Rücksitzen. Wenn das System die Rolle eines an Komfort interessierten Käufers übernimmt, der auch als solcher erkannt werden möchte, werden die von CQH ausgewählten nächstmöglichen Äußerungen an die Unterinstantiierung zur Vorhersage ihrer Folgen auf die Hauptbewertungsdimension von \mathcal{K} (hier z.B. "Komfort") übergeben. Vor und nach der Antizipation werden jeweils die Eindrücke von \mathcal{V} über die Interessen des \mathcal{K} eingeschätzt. Die Eindrucksverschiebung bzgl. der Interessen von \mathcal{K} ergibt sich als die Differenz. Eine positive Verschiebung bedeutet aus der Sicht von \mathcal{K} , daß \mathcal{V} ihm nach seiner Äußerung mehr Interesse in der entsprechenden Hauptdimension zuschreiben wird. Eine negative Verschiebung läßt auf eine geringere Interessenzuschreibung schließen. So erweist sich dann in dem Beispiel (s. Abbildung 6.1) der Gesamtnutzen der Frage nach der Größe des Autos höher als der Gesamtnutzen der Frage nach dem Benzinverbrauch.

6.5 Diskussion

6.5.1 Stärken und Schwächen dieses Ansatzes

In dem vorgestellten Beispiel handelt es sich um einen an Komfort interessierten Käufer, der als solcher erkannt werden möchte. In alltäglichen Situationen kommt es jedoch häufig vor, daß ein Dialogpartner ein bestimmtes Bild seiner Interessen von sich geben möchte, das nicht notwendigerweise seiner tatsächlichen Einstellung entspricht. Beispielsweise könnte ein Käufer, der nicht nur als an Komfort interessiert gelten möchte, dem Verkäufer vorspielen, daß er auch an Umweltfreundlichkeit interessiert ist. Durch eine Unterscheidung zwischen tatsächlichen und projizierten Interessen (vgl. Jameson, 1989) kann diesem Sachverhalt Rechnung getragen werden. \mathcal{K} würde dann seine Äußerungen mittels Antizipation so auswählen, daß das ihm von \mathcal{V} zugeschriebene Interesse an Umweltfreundlichkeit wächst. Die Berücksichtigung der projizierten Einstellung findet erst bei der Antizipation der Reaktion von \mathcal{V} statt. Im obigen Beispiel würden daher Äußerungen weggefiltert, die das zugeschriebene Interesse an Umweltfreundlichkeit verringert hätten.

Ressourcenadaptive Antizipation

Die globale Antizipation setzt den rekursiven Aufruf des ganzen Systems bei der Auswahl der nächsten Dialogäußerung voraus. Verglichen mit lokalen Antizipationsrückkopplungsschleifen stellt ein solcher rekursiver Aufruf einen hohen Rechenaufwand dar. So stellt sich aus Effizienzgründen die Frage, *ob*, *wie*, *wann* und *inwieweit* die globale Antizipation eingesetzt werden sollte.

In diesem Kapitel werden die den Einsatz und die Verwendung der globalen Antizipation betreffenden Effizienzfragen erörtert. Die ersten beiden Abschnitte 7.1 und 7.2 stellen Ansätze dar, wie die Anzahl der Antizipationsrückkopplungsschleifen variiert werden kann. Der darauf folgende Abschnitt 7.3 behandelt die Frage, *wann* — d.h. vor oder nach einer Äußerung — die Folgen dieser Äußerung antizipiert werden sollen. Abschnitt 7.4 zeigt Strategien auf, *ob* es in einer bestimmten Situation aufgrund des zu erwartenden Mehrwerts der Antizipation lohnend ist, zu antizipieren. In Abschnitt 7.5 wird untersucht, wann während der Dialogplanung auf die Antizipation verzichtet werden kann und in Abschnitt 7.6 wird ein Ansatz diskutiert, wie die Unterinstantiierung zur Simulation des Dialogpartners selektiv angepaßt werden kann. Im Anschluß daran wird aufgezeigt, wie die Antizipation zu mehrstufigem Vorausschauen eingesetzt werden kann. Schließlich wird in Abschnitt 7.8 der Frage nachgegangen, wie antizipiert werden kann in den Fällen, in denen der Dialogpartner unter Zeitdruck ist.

Zur Besprechung der Ansätze wird jeweils nach einem kurzen Überblick aufgezeigt, wie der Ansatz formalisiert werden kann. Anschließend werden Realisierungsbeispiele gegeben und Besonderheiten des Ansatzes diskutiert.

7.1 Der Satisficing-Ansatz

7.1.1 Überblick über den Ansatz

Eine Methode, bei Zeitbeschränkungen den Rechenaufwand beim Einsatz der globalen Antizipation zu reduzieren, besteht darin, die Anzahl der Antizipationsrückkopplungsschlei-

fen zu verringern. Hierzu kann das System beispielsweise zunächst die Züge erzeugen, die es in einer gegebenen Situation machen könnte. Auf die potentiellen Züge wird der Antizipationsalgorithmus iterativ angewandt, bis ein akzeptabler bzw. *befriedigender Mindestgesamtnutzen* eines dieser Züge erreicht ist. Bei diesem Ansatz wird keine Maximierung des Gesamtnutzens sondern eine *Satisfizierung* (*Satisficing*¹, s. Simon, 1981, sowie Laux, 1991, S. 52–56) angestrebt, da die Bestimmung des maximal erreichbaren Gesamtnutzens mit Zeitaufwand einhergeht.

7.1.2 Formalisierung

Die Grundidee des Satisficing-Ansatzes besteht darin, den nächsten Dialogzug vom System so zu wählen, daß ein Mindestgesamtnutzen U_{SAT} für das System erreicht wird (s. Abbildung 7.1). Falls ein solcher befriedigender Zug gemacht werden kann, betrachtet das System keinen weiteren Dialogzug mehr und muß somit nicht mehr die Reaktionen des Dialogpartners antizipieren.

7.1.3 Realisierungsbeispiele

Eine Variante dieses Ansatzes ist in PRACMA realisiert worden (Ndiaye & Jameson, 1995). In dieser Implementierung (s. Abbildung 7.2) werden nach dem von CQH zur Äußerungsauswahl verwendeten Algorithmus sämtliche Züge erzeugt. Für jeden dieser potentiellen Dialogzüge werden N mögliche Reaktionen antizipiert. Für jede dieser potentiellen Reaktionen überprüft das System, ob es hierzu eine Antwort geben kann, dessen Nutzen über einem bestimmten Schwellwert liegt. Falls keine derartige Antwort für eine der antizipierten Reaktionen gefunden werden kann, wird der geplante Dialogzug abgelehnt; und das System betrachtet den nächsten von CQH vorgeschlagenen Dialogzug. Das hier verwendete Kriterium, um einen bestimmten Dialogzug nach der Antizipation zu akzeptieren oder zu verwerfen ist wie folgt: Wenn für jede der potentiellen Reaktionen des Käufers auf einen Dialogzug des Verkäufers eine befriedigende Antwort von CQH gegeben werden kann, wird die Antizipation beendet und die geplante Äußerung verbalisiert; ansonsten wird diese Äußerung verworfen und die nächste betrachtet. Falls keine der potentiellen Äußerungen ausgewählt werden konnte, schweigt das System und bewirkt damit einen Sprecherwechsel.

¹Das *satisficing*-Konzept geht auf Simons Arbeiten in den Wirtschaftswissenschaften zu *bounded rationality* zurück. Dem Konzept liegt die Überlegung zugrunde, daß in der realen Welt selten optimale Entscheidungen getroffen werden können; daher begnügen sich Entscheidungsträger oft mit suboptimalen Entscheidungen. "The decision maker has a choice between optimal decisions for an imaginary simplified world or decisions that are 'good enough', that satisfy, for a world approximating the complex real one more closely." (Simon, 1981, S. 35).

```

function BEST-MOVE-SATISFICING(type, constraints) returns a dialog
move of type type that fulfills constraints
  static:  $U_{SAT}$  the satisficing level
            $\delta$  the threshold

  possible-moves  $\leftarrow$  ALLOWABLE-MOVES(type, constraints)
  for each m in possible-moves do
    UTILITY[m]  $\leftarrow$  UTILITY-OF-MOVE(m)
  end
  reasonable-moves  $\leftarrow$  subset of possible-moves with UTILITY  $>$   $\delta$ 
  if *global-anticipation?* then
    UPDATE-SUBORDINATE-INSTANTIATION(ASK(EVALUATION
    HANDLER, "Parameter-Estimates"))
    for each m in reasonable-moves do
      UTILITY[m]  $\leftarrow$  UTILITY[m] + UTILITY-OF-ANTICIPATED-
      RESPONSE(m)
      if UTILITY[m]  $\geq U_{SAT}$  then return m
    end
  else return the m in reasonable-moves with the highest UTILITY[m]

```

Abbildung 7.1: Algorithmus zur Auswahl eines geeigneten Dialogzuges nach dem Satisficing-Ansatz.

(Dieser Algorithmus ist eine Variante von BEST-MOVE. Für jeden der potentiellen Züge werden die Reaktionen des Dialogpartners solange antizipiert, bis ein befriedigender Gesamtnutzen eines Zuges erreicht ist — also $UTILITY[m] \geq U_{SAT}$. Dann hört der Algorithmus sofort auf und liefert den entsprechenden Zug zurück.)

7.1.4 Diskussion

Ein Nachteil der eben vorgestellten Implementierung besteht darin, daß das System eine potentielle Äußerung verwirft, wenn auch nur auf eine der potentiellen Reaktionen keine befriedigende Antwort gegeben werden kann. Es kann jedoch der Fall sein, daß das System nicht allen sondern nur einer antizipierten Reaktion des Dialogpartners mit einer sehr positiven Antwort begegnen kann und daß der Dialogpartner diese Reaktion wahrscheinlicher wählen wird.

Generell hängt die Zeitersparnis bei dem Satisficing-Ansatz davon ab, ob die nach Verwendung der globalen Antizipation erfolgreichen Züge früh betrachtet wurden. Die Züge mit hohem Nutzen vor der Antizipation können sich nach der Antizipation als nicht mehr günstig erweisen. Daher braucht man *Heuristiken* bzw. *Metawissen*, um abzuschätzen, welche Züge als erste betrachtet werden.

Der Wahl des befriedigenden Mindestgesamtnutzens kommt eine große Bedeutung zu: Analog zu den Ausführungen für die Schwellwerte δ bzw. ϵ kann durch ein geschicktes Setzen von U_{SAT} die Verhandlungsstrategie des Systems variiert werden. So kann es z.B. bei großer Unsicherheit über die Bewertungskriterien bzw. über das Dialogverhalten des Käufers zunächst U_{SAT} relativ hoch setzen und im Laufe des Dialoges den Schwellwert verringern. Hierbei wird angenommen, daß eine Reduzierung der Unsicherheit über den Käufer zu einer besseren Verhandlungsstrategie führt. Die Reduzierung der Unsicherheit bedeutet jedoch nicht die Erhöhung der Wahrscheinlichkeit, daß der Dialogpartner letzten Endes das Auto kauft.

<p>— Anticipating reaction to possible comment —</p> <p>Seller: Its average mileage of all tires is very low. Buyer: That's OK. Buyer: Is the car's overall mileage high?</p> <p>— Anticipating next most likely questions —</p> <p>Buyer: Does the car have a central locking mechanism? Buyer: Does the car have child locks? Buyer: Does the car have ABS?</p> <p>— Planned comment rejected! —</p> <p>Da COMMENT AND QUESTION HANDLER keine befriedigende Antwort zu \mathcal{K}s Frage wahrscheinlichste nächste Frage "Is the car's overall mileage high?" geben kann, wird die vorgeschlagene Äußerung über den Kilometer-Stand der Reifen nicht gemacht. Hier macht eine weitere Antizipation von \mathcal{K}s anderen potentiellen Reaktionen keinen Unterschied.</p> <p>— Anticipating reaction to alternative comment —</p> <p>Seller: It has self-tightening seat belts. Buyer: That's OK. Buyer: How big is the car?</p> <p>— Anticipating next most likely questions —</p> <p>Buyer: Does the car have ABS? Buyer: Does the car have a tinted windshield? Buyer: Does the car have an automatic transmission?</p> <p>— Planned comment rejected! —</p>	<p>Der COMMENT AND QUESTION HANDLER in der Rolle von \mathcal{V} könnte befriedigende Antworten zu \mathcal{K}s zweitwahrscheinlichsten nächsten Fragen: Es handelt sich in dem modellierten Beispiel tatsächlich um ein Auto mit Automatik-Getriebe und \mathcal{V} könnte eine vage Antwort über die getönten Windschutzscheiben geben. Nun müßte das System bei der Beantwortung von \mathcal{K}s drittwahrscheinlichster Frage aber zugeben, daß das Auto kein ABS hat. Um zu vermeiden, daß das System in diese Situation kommt, verwirft der COMMENT AND QUESTION HANDLER die zweite potentielle Äußerung über die Sicherheitsgurte.</p> <p>— Anticipating reaction to alternative comment —</p> <p>Seller: It ought to pass the next inspection. Buyer: <No evaluative response> Buyer: Is its time to the next inspection long?</p> <p>— Anticipating next most likely questions —</p> <p>Buyer: Is its average mileage of all tires high? Buyer: Does it have an automatic transmission? Buyer: Does it have an air cooling system?</p> <p>— Planned comment accepted! —</p> <p>Die wahrscheinlichsten Reaktionen des \mathcal{K} zu der dritten potentiellen Äußerung können alle zufriedenstellend von COMMENT AND QUESTION HANDLER in der Rolle von \mathcal{V} behandelt werden. Daher wird die Äußerung "It ought to pass the next inspection" als nächste vom System gemacht.</p>
--	--

Abbildung 7.2: Beispiel zur Anwendung der runner-up-Strategie als Variante des Satisficing-Ansatzes.

(Das Beispiel ist gemäß der ursprünglichen Version abgebildet (Ndiaye & Jameson, 1995).)

7.2 Der Iterationsansatz

7.2.1 Überblick über den Ansatz

Um bei der globalen Antizipation Zeit einzusparen, kann es für ein System nützlich sein, die Anzahl der Antizipationsrückkopplungsschleifen dadurch zu verringern, daß die potentiellen Dialogbeiträge iterativ erzeugt und durch Verwendung der Antizipation überprüft werden. Analog zu dem Satisficing-Ansatz soll das System aufhören zu antizipieren, wenn ein Mindestgesamtnutzen des geplanten Dialogzuges erreicht werden kann. Falls eine Äußerung diesen Nutzen nicht erreicht, dann wird nach einer alternativen Äußerung unter Berücksichtigung der erzielten Ergebnisse der Antizipation gesucht.

7.2.2 Formalisierung

Kern dieses Ansatzes bildet die iterative Erzeugung der potentiellen Dialogzüge, die das System in einer gegebenen Situation machen könnte. Das System fängt mit einem Dialogzug an² und antizipiert die Reaktionen auf diesen Dialogzug. Falls der Gesamtnutzen

²Falls die Domäne es erlaubt, können hier Heuristiken eingesetzt werden, damit das System mit dem Zug beginnt, bei dem die besten Ergebnisse zu erwarten sind.

dieses Dialogzuges für das System über einem bestimmten Schwellwert (U_{SAT}) liegt, wird dieser verbalisiert (s. Abbildung 7.3). Ansonsten wird ein anderer Dialogzug vorgeschlagen unter Berücksichtigung der Gründe, die dazu geführt haben, daß der Schwellwert nicht erreicht werden konnte. Die Iteration geht solange bis entweder ein geeigneter Zug ausgewählt werden kann oder alle potentiellen Kandidaten abgearbeitet sind. Voraussetzung dieses An-

```

function BEST-MOVE-ITERATION(type, constraints) returns a dialog
move of type type that fulfills constraints
  static:  $U_{SAT}$  the statisficing level
            $\delta$  the threshold
   $m \leftarrow$  PROMISING-MOVE(type, constraints)
  loop do
    UTILITY[ $m$ ]  $\leftarrow$  UTILITY-OF-MOVE( $m$ )
    if *global-anticipation?* then
      UPDATE-SUBORDINATE-INSTANTIATION(ASK(EVALUATION
HANDLER, "Parameter-Estimates"))
      UTILITY-OF-ANTICIPATION[ $m$ ], result-of-anticipation  $\leftarrow$  UTILITY-
OF-ANTICIPATED-RESPONSE+( $m$ )
      UTILITY[ $m$ ]  $\leftarrow$  UTILITY[ $m$ ] + UTILITY-OF-ANTICIPATION[ $m$ ]
      if UTILITY[ $m$ ]  $\geq U_{SAT}$  then return  $m$ 
      else  $m \leftarrow$  IMPROVE-MOVE(move, result-of-anticipation)
    else return  $m$ 
  end

```

```

function UTILITY-OF-ANTICIPATED-RESPONSE+ (move) returns a
utility and internal results of  $U$ 's anticipated reactions to guide the
selection of the next candidate move.
  return UTILITY-OF-RESPONSE(ASK(GAF, "ANTICIPATED-
RESPONSE(move)")), ANTICIPATED-RESPONSE(move)

```

Abbildung 7.3: Algorithmus zur Auswahl eines geeigneten Dialogzugs nach dem Iterationsansatz.

(Dieser Algorithmus stellt eine iterative Prüfung der potentiellen Kandidaten dar. Für jede weitere Iteration werden die Ergebnisse der vorherigen Antizipation berücksichtigt. Für jeden der potentiellen Züge werden die Reaktionen des Dialogpartners solange antizipiert, bis ein geeigneter Zug m gefunden ist — also UTILITY[m] $\geq U_{SAT}$. Dann hört der Algorithmus sofort auf und liefert den entsprechenden Zug zurück.)

satzes ist allerdings, daß jeder Zug eine Verbesserung des vorigen darstellt. Hierzu müssen bei der nächsten Auswahl des Kandidaten die Ergebnisse der Anwendung der Antizipation auf den vorigen Kandidaten berücksichtigt werden.

7.2.3 Realisierungsbeispiele

In ANTLIMA (Schirra, 1995; Blocher & Schirra, 1995, s. auch Abschnitt 2.6.5) wurde ein Spezialfall dieser Strategie realisiert. Wenn das System eine Äußerung z_1 zur Beschreibung

einer Szene erzeugt hat, antizipiert die AFL-Komponente von ANTLIMA, wie der Benutzer die Äußerung interpretieren wird. Falls die antizipierte Interpretation r_1 von dem vom Sprecher intendierten Verständnis abweicht (d.h. ein bestimmter Schwellwert wurde nicht erreicht), wird die geplante Äußerung z_1 korrigiert. Die neue Äußerung z_2 wird unter Berücksichtigung der Gründe erzeugt, die zur Ablehnung von z_1 geführt hatten, z.B. durch Hinzunahme eines optionalen Tiefenkasus modifiziert. ANTLIMA könnte dann wie im Algorithmus von Abbildung 7.3 erneut die Interpretation r_2 zu z_2 voraussagen, um zu überprüfen, ob der Schwellwert erreicht wird; aber ANTLIMA generiert z_2 unter der Annahme, daß durch die Verbesserung von z_1 eine adäquate Äußerung generiert wird.

7.2.4 Diskussion

Der Erfolg des Iterationsansatzes hängt entscheidend davon ab, daß jeder Zug während der Iteration eine Verbesserung des vorigen darstellt. Hierbei stellt sich die Frage, wie die Ergebnisse früherer Antizipationen wiederverwendet werden können. Die Methode eignet sich für Domänen wie Szenenbeschreibungen oder multimodale Präsentationen. Wenn es z.B. darum geht, unaufgefordert eine Anmerkung in einem bewertungsorientierten Dialog zu machen, läßt sich die Methode nicht unmittelbar einsetzen: beispielsweise werden in Dialogsystemen PRACMA und IMP zunächst sämtliche Züge erzeugt, um aus diesen denjenigen mit dem höchsten Nutzen auszuwählen. Bei jedem Iterationsschritt müssen sämtliche Züge erneut berechnet werden. Im vorigen Abschnitt wurde eine Verallgemeinerung dieser Methode — der Satisficing-Ansatz — vorgestellt, die den Besonderheiten bewertungsorientierter Dialoge gerecht wird.

7.3 Der Ansatz der Antizipation im Nachhinein

7.3.1 Überblick über den Ansatz

Eine Variante ressourcenadaptiver Antizipation bei Zeitbeschränkungen besteht darin, die Entscheidung, *wann* antizipiert werden sollte, in Abhängigkeit von dem zu erwartenden Mehrwert der Antizipation zu treffen. Der Mehrwert der Antizipation hängt von der Wahrscheinlichkeit ab, daß die geplante Äußerung von Anfang an adäquat ist und keiner Verbesserung bedarf. Je nachdem, wie der Mehrwert eingeschätzt wird, kann das System auf die Antizipation gänzlich verzichten, sie vor der Äußerung einsetzen oder erst im Nachhinein. Im letzteren Fall kann ein System unter Zeitdruck zunächst auf die globale Antizipation verzichten, da diese nicht gerade kostengünstig ist. Das System kann dann *nach* der Verbalisierung der Äußerung unter Zeitdruck die globale Antizipation dazu verwenden, um gegebenenfalls eine *Korrektur* zu machen. Dabei wird folgendermaßen vorgegangen: (a) verwende vor der Äußerung nur lokale Antizipationsschleifen, um minimal verständliche Äußerungen zu ermöglichen und (b) verwende globale Schleifen erst *nach* der Äußerung, um eventuelle Nachteile aufzuspüren und durch spontane Äußerungen zu korrigieren.

Kern des Ansatzes ist es, die Parallelität auszunutzen, die zwischen der Interpretation der Äußerung durch \mathcal{U} und der Antizipation durch \mathcal{S} nach der Verbalisierung der Äußerung besteht: Man antizipiert, während der Dialogpartner die bereits gemachte Äußerung interpretiert.

7.3.2 Formalisierung

Dem Ansatz liegt die Annahme zugrunde, daß, wenn im Nachhinein antizipiert wird, es \mathcal{S} gelingt, sich zu verbessern, bevor \mathcal{U} die gemachte Äußerung interpretieren und darauf reagieren kann. Je nachdem, wann antizipiert werden soll und wie die Antizipation ausfällt, ergeben sich 4 Fälle (s. Abbildung 7.4):

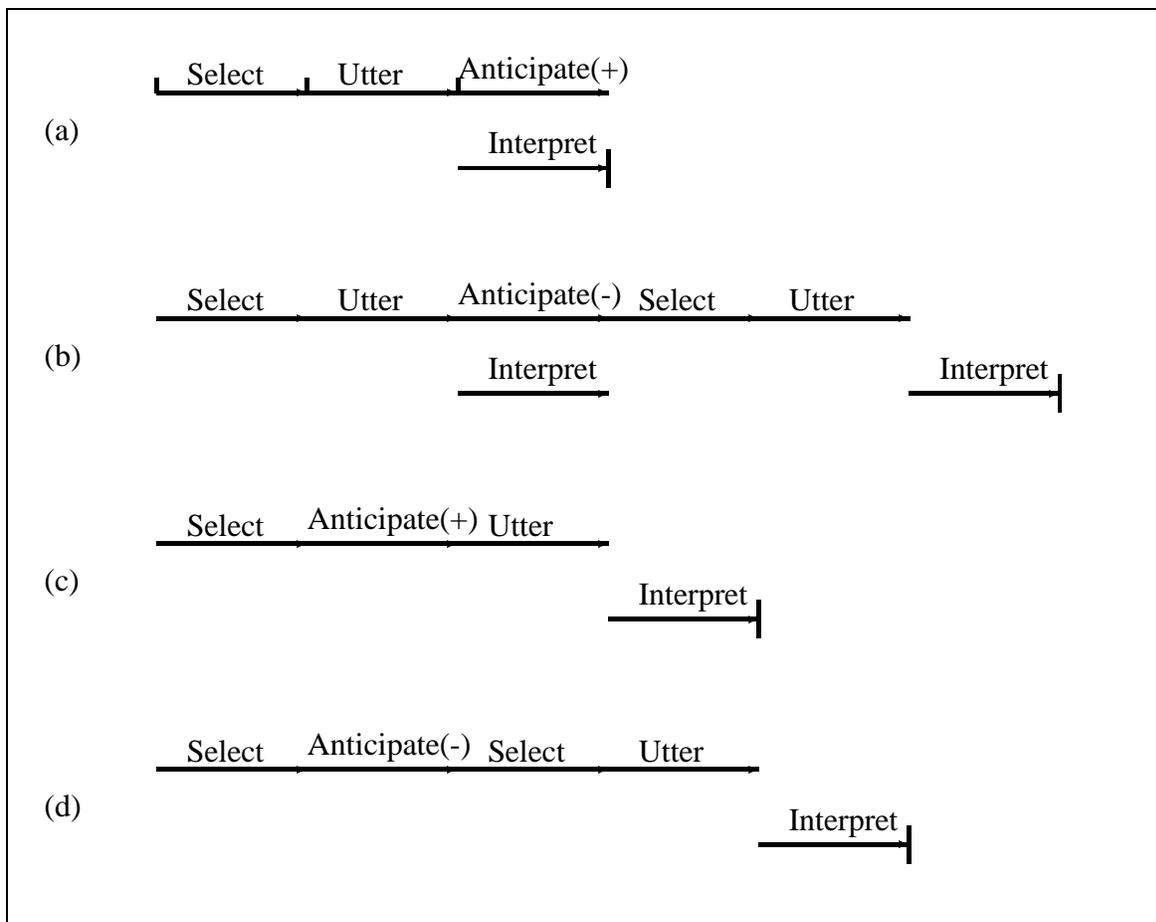


Abbildung 7.4: Zusammenstellung der Teilprozesse zur Äußerungsauswahl und -interpretation.

(Die Teilprozesse sind in der Abbildung gleich lang gezeichnet, obwohl sie in der Realität unterschiedlich lang dauern. Die Bezeichnungen (+) bzw. (-) hinter "Anticipate" drücken aus, daß die Antizipation positiv bzw. negativ ausfällt.)

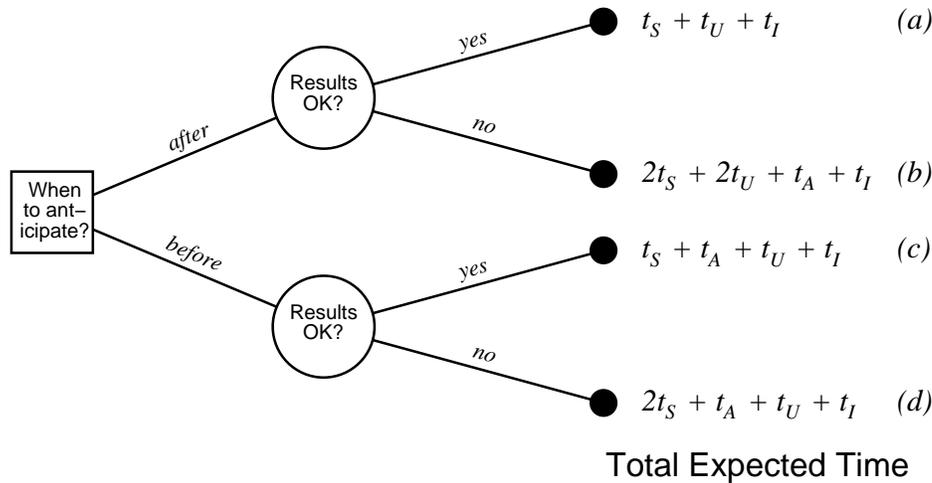


Abbildung 7.5: Darstellung des Entscheidungsprozesses von \mathcal{S} , wann antizipiert werden soll.

- Fall (a): \mathcal{S} verzichtet auf die Antizipation vor der Verbalisierung und gibt seine bereits selektierte Äußerung aus. Während \mathcal{U} die gemachte Äußerung interpretiert, antizipiert \mathcal{S} die Reaktionen auf seine Äußerung und stellt fest, daß die Äußerung keiner Verbesserung bedarf.
- Fall (b): \mathcal{S} verhält sich wie im Fall (a), stellt jedoch aufgrund der negativen Ergebnisse der Antizipation fest, daß Verbesserungsmaßnahmen eingeleitet werden müssen: \mathcal{S} selektiert dann eine zusätzliche Äußerung und verbalisiert sie.
- Fall (c): \mathcal{S} setzt die Antizipation vor der Verbalisierung der Äußerung ein, stellt jedoch keinen Verbesserungsbedarf fest.
- Fall (d): \mathcal{S} verhält sich wie im Fall (c), muß aber nach der Antizipation nach einer neuen Äußerung bzw. nach einer Verbesserung der von der Antizipation abgelehnten Äußerung suchen.

In der Abbildung 7.4 wurde angenommen, daß die Teilprozesse gleich lang dauern. In der Realität dauern sie aber unterschiedlich lang. Wie Abbildung 7.4 es verdeutlicht, kann \mathcal{S} in den Fällen (a) und (b) antizipieren, während \mathcal{U} eine Äußerung interpretiert. Es ergeben sich folgende *kritische Pfade*³, die die maximale Dauer widerspiegeln: (a) S–U–I; (b) S–U–A–S–U–I; (c) S–A–U–I; (d) S–A–S–U–I. Die vier Fälle lassen sich als Entscheidungsbaum darstellen (s. Abbildung 7.5).

³Kritische Pfade (*Critical Path Method*) sind aus der Netzplantechnik bzw. dem Projektmanagement bekannt. Dort wird ein Projekt durch eine Menge von Vorgängen definiert, zu deren Durchführung jeweils eine vorgegebene Zeitspanne, die Dauer des Vorganges, erforderlich ist. Zwischen den einzelnen Vorgängen bestehen u.U. Abhängigkeiten derart, daß ein Vorgang frühestens unmittelbar nach Ende eines anderen Vorganges begonnen werden kann. Aus den Abhängigkeiten zwischen den Ereignissen ergibt sich der *kritische Pfad* von Beginn zum Ende des Projektes als die maximale Länge, die zugleich die Mindestdauer des Projektes festlegt.

S basiert seine Entscheidung, ob er vor oder nach der Äußerung antizipiert, auf dem Mehrwert von GAF. Sei p die Wahrscheinlichkeit, daß die Resultate der Antizipation positiv ausfallen. Sei C ein Maß für die Kosten der Verbesserung. (C drückt beispielsweise aus, wie peinlich es ist, wenn man sich nach einer Äußerung verbessern muß.) Wenn ansonsten die Zeitkosten als die einzig anfallenden Kosten berechnet werden, ergeben sich folgende Erwartungswerte für die Antizipation nach bzw. vor der Äußerung:

$$\text{Erwartungswert}[a | b] = p(t_S + t_U + t_I) + (1 - p)(2t_S + 2t_U + t_A + t_I + C)$$

$$\text{Erwartungswert}[c | d] = p(t_S + t_U + t_A + t_I) + (1 - p)(2t_S + t_U + t_A + t_I)$$

Die kritische Wahrscheinlichkeit p_c für den Erfolg der Strategie $a | b$, d.h. die minimale Wahrscheinlichkeit, oberhalb der die Strategie der Antizipation nach der Äußerung eingesetzt werden soll, beträgt:⁴

$$p_c = \frac{t_U + C}{t_A + t_U + C} = 1 - \frac{t_A}{t_U + C + t_A}$$

Je höher die Zeit der Antizipation ausfällt, desto kleiner wird p_c . In diesem Fall ist der Erfolg der Antizipation im Nachhinein gering. Je höher die Kosten für die Verbesserung ausfallen, um so weniger eignet sich die Antizipation im Nachhinein. Bei hohen Kosten für die Verbesserung empfiehlt sich, von vorneherein zu antizipieren (s. hierzu die spätere Diskussion).

7.3.3 Realisierungsbeispiele

7.3.3.1 Realisierung in PRACMA

Die Antizipation im Nachhinein ist in dem jetzigen PRACMA-System nicht implementiert; wir geben jedoch an, wie sie in PRACMA implementiert werden kann. PRACMA verfügt über verschiedene Parameter, um das Verhalten des Systems unter Zeitdruck zu modellieren. Beispielsweise wird das System, wenn es unter Zeitdruck ist, die Anzahl der zu berücksichtigenden Dimensionen beschränken (s. Abschnitt 7.8). Da S sich unter Zeitdruck schnell äußern soll, verzichtet es bei der Äußerungsauswahl wegen den Zeitbeschränkungen zunächst auf die Überprüfung der potentiellen Kandidaten durch die globale Antizipation. Erst nach der Verbalisierung der Systemäußerung wird die globale Antizipation aufgerufen. Falls die Antizipationsergebnisse negativ ausfallen, wird eine Korrektur durchgeführt. Die Verbesserung bestand darin, bei Bedarf spontan eine zusätzliche Äußerung zu machen.

7.3.3.2 Die Studie von Horton und Keysar (1996)

Erkenntnisse zu dem in diesem Abschnitt vorgestellten Ansatz liefern z.B. die Arbeiten von Horton und Keysar (1996), die untersucht haben, wie und wann Sprecher *common*

⁴Die Wahrscheinlichkeit ergibt sich durch Gleichsetzung der Erwartungswerte für die beiden Antizipationsstrategien

ground (dem Sprecher und dem Hörer gemeinsame Kontextinformationen) in der referentiellen Kommunikation einsetzen. Die Analogie zu den in GAF betrachteten Situationen gilt insofern, als die Einbeziehung des *common ground* bei der Sprachproduktion eine unmittelbare Auswirkung auf die Antizipation hat. Die Berücksichtigung vom gemeinsamen Kontextwissen führt z.B. zu adäquateren Beschreibungen, die der Zweck der Antizipation in NAOS oder ANTLIMA sind (s. Abschnitt 2.6.4 und 2.6.5).

Horton und Keysar (1996) zeigen, daß Sprecher auf *common ground* zugreifen, erst nachdem ein Initialplan für die Sprachproduktion formuliert ist. Aus der Studie geht hervor: (a) Ohne Zeitbeschränkung basieren Beschreibungen vom Sprecher in der referentiellen Kommunikation eher auf *common ground* (dem Sprecher und dem Hörer gemeinsame Kontextinformationen) als auf privilegierte Kontextinformationen; (b) unter Zeitbeschränkung basieren die Beschreibungen gleichermaßen auf privilegierten Kontextinformationen. Nach dieser Studie ist die Einbeziehung von *common ground* in der Äußerungsplanung abhängig von den verfügbaren Ressourcen: "Speakers' descriptions relied on privileged context information less than on shared context information, but when speeded they relied on shared and privileged context to the same degree. We interpret this pattern of results as initial evidence against a model that assumes that common ground is part of an initial utterance design." (Horton & Keysar, 1996, S. 111). Unter Zeitdruck wird erst in einer späteren Phase versucht, gegebenenfalls den Initialplan zu verbessern: "According to the Monitoring and Adjustment model, while the initial plan does not take common ground into account, speakers do monitor and attempt to correct and revise utterances which violate common ground." (Horton & Keysar, 1996, S. 111).

Nach der Studie von Horton und Keysar (1996, S. 112) bedarf der Plan in vielen Fällen keiner *post hoc* Korrektur, obwohl in der Initialphase nur privilegierte und keine dem Hörer und dem Sprecher gemeinsamen Informationen explizit verwendet wurden. Dieses Modell ist im Vergleich zum Initial Planning Modell weniger ressourcenaufwendig. Im Falle von Ressourcenbeschränkung ist diese Vorgehensweise von Vorteil.

Bezogen auf die Situation von Abbildung 7.4 korrespondiert die Initialplanung zu der Phase "Select" und das Monitor and Adjustment den Phasen "Anticipate, Select, Utter". Wenn die Situation auf GAF übertragen wird, wird bei Zeitbeschränkungen das System bei der Äußerungsplanung auf die globale Antizipation verzichten und u.U. in einer späteren Phase eventuell entstandene Nachteile der Äußerung durch globale Antizipation entdecken und durch weitere freiwillige Äußerungen aufheben. Nach der Studie wäre der Mehrwert der globalen Antizipation in einigen Situationen gering, da man *häufig* von Anfang an eine gute Äußerung auswählt. So würde PRACMA unter Zeitdruck zunächst auf den Einsatz der globalen Antizipation verzichten. Die Studie von Horton und Keysar (1996) läßt jedoch die Frage der Verallgemeinerbarkeit des Ansatzes auf andere Domänen als die der referentiellen Kommunikation offen.

7.3.4 Diskussion

Bei der Entscheidung, erst im Nachhinein zu antizipieren, hängt es u.a. von zwei Fragen ab: Wie peinlich ist es, wenn man sich nach einer Äußerung verbessern muß bzw.

wieviel kostet die Verbesserung? In einigen Situationen, etwa Vorstellungsgesprächen wie sie beispielsweise in IMP modelliert wurden, müssen Äußerungen gründlich überlegt werden. Nach einer unüberlegten Äußerung kann es viel Mühe kosten, den beim Interviewer erweckten Eindruck zu korrigieren. Hingegen gibt es Situationen, in denen es vorteilhaft sein kann, von Anfang an auf eine Korrektur zu setzen. Dies eignet sich in Situationen, in denen beispielsweise aufgrund von Arbeitsgedächtnisbelastung dem Dialogpartner nur leicht verständliche Äußerungen zugemutet werden. Man denke z.B. an die Äußerung eines Mechanikers an eine Person unter Zeitdruck in einer Vorstudie für das READY-Projekt⁵ (Wahlster *et al.*, 1995a; 1995b): *Im Fußraum, das heißt unterm Lenkrad links ist ein Hebel. Dadran ziehen.* Hier verzichtet der Sprecher auf eine längere Ausführung und setzt von vornherein auf die Selbstkorrektur.⁶ Der Vorteil in einer Situation mit Zeitdruck ist, daß der Sprecher bald mit der Verbalisierung der Äußerung anfangen kann und daß der Hörer früher mit der Interpretation anfangen kann und evtl. zwischendurch Rückmeldung machen kann. Mit Einsatz der Antizipation vor der Artikulation der Äußerung hätte die Äußerung folgendermaßen aussehen können: *Unterm Lenkrad links ist ein Hebel.* Ohne Antizipation (*Im Fußraum ist ein Hebel*) hätte schnell mit der Verbalisierung begonnen werden können, aber die Chancen auf erfolgreiche Interpretation der Äußerung wären geringer.

Der vorgestellte Ansatz, bei dem in Abhängigkeit von der zu erwartenden Zeit entschieden wird, geht von festen Zeitabschnitten für die einzelnen Teilprozesse aus. Hierbei ergibt sich ein Unterschied zu den *Anytime-Algorithmen* (Dean & Boddy, 1988), die unterbrechbare Algorithmen darstellen, deren Ausgabequalität sich bei längerer Rechenzeit verbessert. Sie geben Näherungslösungen aus, falls sie vor einem vollständigen Durchlauf unterbrochen werden. Das Ergebnis ist eine Funktion der investierten Verarbeitungszeit bzgl. einer Qualitätsmetrik, die in sog. Performanzprofilen ausgedrückt wird (s. Russell & Wefald, 1991a). Während dort eine geringe Überschreitung bei einer flexiblen Zeitschranke noch hinnehmbar ist, ist der Rückgabewert eines Moduls nach dem Ablauf einer absoluten Zeitschranke unbrauchbar. In dem hier vorgestellten Ansatz ist das Ergebnis bei Überschreitung der Zeitschranke brauchbar.

Horton und Keysar (1996) haben in ihrem Aufsatz Zeitdruck untersucht, jedoch nicht explizit den Anytime-Ansatz berücksichtigt. Jedes Experiment in der Untersuchung dauerte 10 Sekunden. Zur Realisierung eines Anytime-Ansatzes könnte man eine stochastische Zeitschranke zufällig innerhalb der verfügbaren Zeit wählen, und die Äußerungen der Testpersonen untersuchen.

In PRACMA kann das Anytime-Verhalten folgendermaßen realisiert werden: Das CQH-Modul bestimmt als erstes das beste Ergebnis ohne den Einsatz der Antizipation und ruft parallel das GAF-Modul auf. Falls bis zum Ablauf der vorgegebenen Zeitschranke die Er-

⁵Die Forschungen zu READY werden von der Deutschen Forschungsgemeinschaft im Rahmen des Sonderforschungsbereichs 378 "Ressourcenadaptive kognitive Prozesse" gefördert.

⁶Vgl. die Arbeit von Finkler (1997) zu Selbstkorrektur. Dort wurde ein situativer Generator realisiert, der je nach ausgesprochenem Äußerungsanfang unterschiedliche Korrekturfortsetzungen ansteuern kann und auch in der Lage ist, an einer geeigneten Stelle zu unterbrechen, damit eine günstige (im Sinne von in der umfangreichen empirischen Analyse menschlicher Korrekturen häufig aufgetretenen Konstruktion) Äußerungsfortsetzung erfolgen kann.

gebnisse der Antizipation vorliegen, werden diese noch für die Äußerungsauswahl von CQH berücksichtigt. Falls sie erst nach dem Ablauf der vorgegebenen Zeitschranke verfügbar sind, und falls die Zeit bis zum Verstehen des bereits geäußerten Dialogbeitrags ausreicht, kann das System gegebenenfalls eine *Selbstkorrektur* durchführen.

Die Abschätzung der erwarteten Zeit für die Äußerungsauswahl mit bzw. ohne Antizipation erfordert jedoch ein Modell zur Repräsentation von Ressourcenbeschränkungen und zur Allokation der Systemressourcen (s. Z.B. Wahlster *et al.* 1995a, 1995b).

7.4 Der Ansatz der Antizipation nach Wert der Berechnung

7.4.1 Überblick über den Ansatz

Während bei dem Ansatz der Antizipation im Nachhinein (s. Abschnitt 7.3) auf jeden Fall antizipiert wurde, wird in diesem Abschnitt eine Strategie vorgestellt, bei der die Entscheidung, *ob* antizipiert werden sollte, in Abhängigkeit von dem zu erwartenden Mehrwert der Antizipation getroffen wird. Hier stellt sich die Frage, wie der Nutzen der Antizipation *geschätzt* werden kann, ohne sie explizit durchzuführen.

Bevor auf den Ansatz der Antizipation nach Wert der Berechnung eingegangen wird, wird der Ansatz von Russell und Wefald zum Wert einer Berechnung, der als Grundlage dient.

7.4.2 Exkurs: Russell und Wefald's Ansatz zu "Value of Computation"

In Bereich von Suchalgorithmen untersuchten Russell und Wefald (1991a, 1991b), wie in Abhängigkeit von dem Zustand und dem zu erwartenden Nutzen, ein Knoten expandiert werden soll oder nicht. Die Kernidee hierbei ist, daß die Entscheidung getroffen wird, ohne daß der Nutzen der Expansion des Knoten explizit berechnet wird. Daher kann bestenfalls eine Entscheidung aufgrund des *erwarteten Wertes* der Berechnung getroffen werden.⁷

Sei zu einem bestimmten Zeitpunkt α die Aktion ("current best action"), die das System ausführen kann, falls es nicht weiter vorausschaut. Das Ziel jeder weiteren Berechnung S_j vor der Durchführung einer Aktion dient der Verfeinerung der Auswahl der Defaultaktion. Der Mehrwert (V) der weiteren Berechnung ist der Zuwachs des Nutzens, verglichen mit dem Nutzen der Defaultaktion α , die hätte gemacht werden können.

⁷Berechnungen im Sinne von Russell und Wefald (1991a) sind Aktionen, unter denen auf der Basis ihrer erwarteten Nutzen selektiert wird. Diese Nutzen sind aus den erwarteten Effekten der Berechnungen abgeleitet, unter anderem hinsichtlich des Verbrauchs von Zeit und Speicher und der möglichen Revision der intendierten Aktion des Agenten. Der Nutzen einer Berechnung und der resultierenden Aktion ist eine Funktion der Qualität der resultierenden Lösung und der zur Auswahl dieser Lösung benötigten Zeit.

$$V(S_j) = U([S_j]) - U([\alpha])$$

Der Nutzen von S_j geht jedoch mit Kosten der Expansion einher. Durch Trennung des *Nutzens* in eine zeitunabhängige Funktion U_I und die *Kosten der Berechnung* (TC , Time Cost) läßt sich der erwartete Wert der Expansion eines Knotens (*value of computation*) folgendermaßen definieren:

$$V(S_j) = \Delta(S_j) - TC([S_j])$$

Dabei ist

$$\Delta(S_j) = U([\alpha_{S_j}]) - U([\alpha])$$

der Zuwachs des Nutzens

$$\hat{V}(S_j) = \Delta(S_j) - TC([S_j])$$

Russell und Wefald (1991a) verwenden einen entscheidungstheoretischen Ansatz. Dabei ist die optimale Aktion diejenige, die den erwarteten Nutzen für den Agenten maximiert:

$$U(A_i) = \sum_k P(W_k)U([A_i, W_k])$$

wobei gilt: A_i ist eine Menge potentieller Aktionen zum betrachteten Zeitpunkt; $[A_i, W_k]$ ist das Ergebnis der Anwendung von A_i im Weltzustand W_k (d.h. in unserem Kontext $[A_i, W_k]$ beschreibt die potentiellen Reaktionen des Dialogpartners auf die vom System geplanten Äußerungen A_i); $P(W_k)$ ist die Wahrscheinlichkeit, daß das System/der Agent im Weltzustand W_k ist. Weiterhin wird eine Kostenfunktion C eingeführt, die die Differenz zwischen dem Gesamtnutzen und dem eigentlichem Nutzen ("intrinsic utility") beschreibt. Der erwartete eigentliche Nutzen der Aktion A_i nach Beendigung von S_j ergibt sich aus der Differenz zwischen dem Gesamtnutzen der Aktion A_i und den *Kosten der Expansion* von S_j :

$$U([S_j]) = U([A_i, [S_j]]) = U_I([A_i]) - C(A_i, S_j)$$

Durch Trennung des *Nutzens* und der *Kosten einer Berechnung* (TC , Time Cost) läßt sich der erwartete Wert einer Expansion eines Knotens (*value of computation*) folgendermaßen definieren:

$$\hat{V}(S_j) = \Delta(S_j) - TC([S_j])$$

wobei

$$\Delta(S_j) = U([\alpha_{S_j}]) - U([\alpha])$$

den geschätzten Mehrwert der Berechnung darstellt. Da die Nutzen geschätzt werden sollen, ergibt sich Unsicherheit.

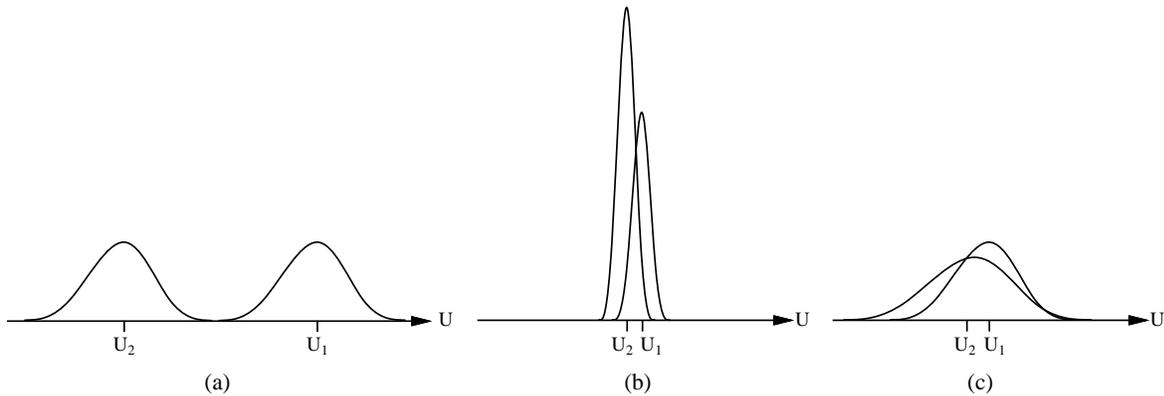


Abbildung 7.6: Die möglichen Fälle der Antizipation nach Wert der Berechnung.

7.4.3 Formalisierung

Während in den bis jetzt betrachteten Fällen immer antizipiert wurde, betrachten wir nun einen Ansatz, bei dem zunächst geschätzt wird, was die Antizipation ergeben würde.

Seien A_1 die Äußerung mit dem höchsten errechneten Nutzen ohne Einsatz der globalen Antizipation und A_2 die beste Alternativäußerung, d.h. mit dem zweithöchsten Nutzen. (Ohne Einsatz der Antizipation wird das System die Äußerung A_1 selektieren und verbalisieren.) Man betrachte nun die zu erwartenden Nutzen, wenn antizipiert würde. Hierzu verwendet man beispielsweise die Varianz der gelernten Nutzen aus früheren Antizipationen. Seien A_1^{GAF} und A_2^{GAF} die Anwendung der globalen Antizipation Äußerungen, die sich nach Anwendung der Antizipation auf A_1 und A_2 ergeben würden. Seien $\hat{U}(A_1^{\text{GAF}})$ und $\hat{U}(A_2^{\text{GAF}})$ die zu erwartenden Nutzwerte. Drei Fälle sind zu behandeln (s. Abbildung 7.6):

- (a) $\hat{U}(A_2^{\text{GAF}}) \gg \hat{U}(A_1^{\text{GAF}})$
 Wenn $\hat{U}(A_1^{\text{GAF}})$, der geschätzte Wert einer Aktion, und $\hat{U}(A_2^{\text{GAF}})$, der des besten Alternativkandidaten, weit auseinanderliegen und sich die beiden Kurven nicht überlappen, dann verzichtet das System auf eine tatsächliche Vorausschau der Auswirkung dieser Aktionen. Das System hört dann auf und wählt die Aktion mit dem größeren geschätzten Nutzen aus. In diesem Falle wird davon ausgegangen, daß es unwahrscheinlich ist, daß eine tatsächliche Antizipation mehr Informationen bringt. Die Wahrscheinlichkeit, eine andere Aktion nach der Antizipation zu wählen, ist sehr gering.
- (b) $\hat{U}(A_2^{\text{GAF}}) \approx \hat{U}(A_1^{\text{GAF}})$
 Wenn die geschätzten Werte zweier potentieller Aktionen sehr nah beieinander liegen und die Varianz der Differenz klein ist, lohnen sich auch keine weiteren Antizipationen. In diesem Fall ist es unwahrscheinlich, daß eine weitere Antizipation einen signifikanten Unterschied zwischen beiden Alternativen offenbart.
- (c) $\hat{U}(A_2^{\text{GAF}}) \approx \hat{U}(A_2^{\text{GAF}})$

Falls sich die Verteilungen überlappen und es viel Unsicherheit über den geschätzten Nutzen der potentiellen Aktionen gibt, soll eine weitere Antizipation in Betracht gezogen werden. Da man in diesem Fall nicht weiß, welche der beiden potentiellen Aktionen die beste wäre, muß tatsächlich antizipiert werden. Man ruft dann GAF auf die beiden Aktionen auf und wählt letztendlich die Aktion mit dem tatsächlich errechneten größeren Nutzen. Hier könnte man sich auch dazu entscheiden, dieselbe Prozedur anzuwenden, statt nur die Aktion mit dem errechneten größeren Nutzen zu nehmen (z.B., wenn mehr Zeit in Anspruch genommen werden kann).

7.4.4 Realisierungsbeispiele

Dem Ansatz liegt die Idee zugrunde, daß man für gewisse Klassen von Äußerungen bzw. Situationen weiß, wieviel Varianz im Nutzen der Reaktionen es gibt. So kann man beurteilen, wie wahrscheinlich es ist, daß der zweitbeste Kandidat den besten “überholt”. In PRACMA wird nach der Selektion durch CQH der Kandidaten zur Äußerungsauswahl diese Vorgehensweise angewandt. Die Varianz im Nutzen der Reaktion des Dialogpartners kann z.B. pro Sprechakttyp bestimmt werden. (So kann man z.B. die Varianz bei einer bewertenden Äußerung angeben.)

Seien A_{GAF} und $A_{\text{-GAF}}$ die beiden Äußerungskandidaten. Das Kriterium kann dann wie folgt lauten: verzichte auf Antizipation, wenn gilt

$$U(A_{\text{GAF}}) - U(A_{\text{-GAF}}) \geq k * \sqrt{\sigma_{A_{\text{GAF}}}^2 + \sigma_{A_{\text{-GAF}}}^2}$$

7.4.5 Diskussion

Zur Realisierung des in diesem Abschnitt vorgestellten Ansatzes müssen die Varianzen bestimmt werden. Hierzu lassen sich Lernmethoden einsetzen. Russell und Wefald (1991a, S. 161) sprechen in diesem Zusammenhang von “learn[ing] to predict the outcome of computational actions and learn[ing] the value of computation directly”. Es lassen sich zwei Fälle unterscheiden:

- Das System könnte auch nach der Antizipation (*post hoc*) lernen. Hierzu merkt es sich die Aktion, die es ohne Anwendung von GAF ausgewählt hätte, und vergleicht sie nach der Antizipation mit der tatsächlich nach der Verwendung von GAF gewählten Äußerung.
- Man könnte z.B. folgende Vorgehensweise benutzen:⁸ Zwei Systeme, das eine in der Rolle des Verkäufers und das andere in der Rolle des Käufers, simulieren “offline” mehrere Verkaufssituationen und lernen somit vorherzusagen, wie der jeweilige Dialogpartner reagieren wird. Nach der Lernphase können die Ergebnisse kompiliert werden.

Im ersteren Fall ist das Lernen in echten Dialogen bzw. simulierten Dialogen, aufwendig und man braucht hierzu ein Maß für den letztendlichen Erfolg eines Dialogzuges. Beim

⁸Ich danke Stuart Russell für diesen Vorschlag.

zweiteren werden die Varianzen nur aufgrund von Ausprobieren der Antizipation gelernt. Dabei geht es um die Frage, wie wahrscheinlich es ist, daß die Antizipation die Einschätzung von S ändert.

Wie bereits angesprochen, führt eine Reduzierung der Unsicherheit über den Dialogpartner nicht notwendigerweise zu einer Erhöhung der Wahrscheinlichkeit des Verkaufs des Autos. Durch den Lernprozeß entwickelt das System Wissen über die Wahrscheinlichkeit, daß ein bestimmter Dialogzug es seinem Ziel näher bringt, das Auto zu verkaufen. (In Spielsituationen erhöhen bzw. vermindern manche Züge die Wahrscheinlichkeit des Sieges.)

7.5 Der Ansatz der selektiven Antizipation

7.5.1 Überblick über den Ansatz

In Systemen mit getrennten Modulen zur Dialogplanung und Äußerungsauswahl wird das Modul zur Äußerungsauswahl u.U. mehrfach gefragt, ob es eine Anmerkung eines *bestimmten Typs* liefern kann. Hierbei kann das System auf die Antizipation bei manchen der Aufrufe des Moduls zur Äußerungsauswahl verzichten.

7.5.2 Formalisierung

Ein System, das wie PRACMA getrennte Module zur Dialogplanung und Äußerungsauswahl hat, geht bei der Bestimmung des nächsten Dialogbeitrags folgendermaßen vor: Zunächst wird in einer ersten Phase bestimmt, was gesagt werden kann — etwa, ob eine Anmerkung eines bestimmten Typs lohnend ist —, und in einer zweiten Phase, falls der Dialogplaner sich für diesen Anmerkungstyp entschieden hat, liefert das Modul zur Äußerungsauswahl eine Anmerkung dieses Typs. So ergeben sich in solchen Systemen vier Möglichkeiten der Anwendung der globalen Antizipation:

1. Das System setzt in beiden Planungsphasen die globale Antizipation ein.
2. Das System verwendet während der initialen Planungsphase die globale Antizipation und verzichtet später, wenn es sich für diesen Anmerkungstyp entschieden hat und das Modul zur Äußerungsauswahl die gleiche Anmerkung vorschlägt, darauf, erneut die Reaktionen des Dialogpartners vorherzusagen.
3. Das System verzichtet während der initialen Planungsphase auf die Verwendung der globalen Antizipation, da es hierbei nur darum geht, ob es sich grundsätzlich lohnt, eine Anmerkung eines bestimmten Typs zu machen. Erst nachdem der Dialogplaner sich für einen bestimmten Anmerkungstyp entschieden hat — d.h. in der zweiten Planungsphase —, wird das Modul zur Äußerungsauswahl dazu aufgefordert, eine Anmerkung abzuliefern.

4. Das System verzichtet in den beiden Phasen der Dialogplanung auf die Antizipation, wenn abzusehen ist, daß sie keinen Nutzen bringt oder wendet die Antizipation auf die erfolgversprechenden Anmerkungen an (s. hierzu die spätere Diskussion).

7.5.3 Realisierungsbeispiel

In PRACMA fragt der DIALOG PLANNER bei der Bestimmung des nächsten Dialogzuges oft CQH — das Modul zur Äußerungsauswahl —, ob es sich grundsätzlich lohnt, eine Anmerkung eines bestimmten Typs zu machen. Auch wenn CQH eine geeignete Anmerkung findet, kann sich der Planer nachfolgend noch aufgrund anderer relevanter Kriterien für einen anderen Dialogzug entscheiden. CQH kann deshalb bei solchen Anfragen während der initialen Planungsphase auf die globale Antizipation der Reaktionen des Dialogpartners verzichten. Erst wenn der DIALOG PLANNER nachfolgend tatsächlich nach einer Anmerkung dieses Typs fragt, ruft CQH das Modul für die globale Antizipation auf.

In dem beschriebenen Fall, daß zunächst während der initialen Planungsphase auf die Antizipation verzichtet wird und sie erst in der zweiten Phase zum Einsatz kommt, kann es gelegentlich vorkommen, daß die von CQH vorgeschlagene Anmerkung sich nach der globalen Antizipation als nicht mehr günstig erweist. In diesem Fall muß CQH entweder eine alternative Anmerkung gleichen Typs liefern oder gegebenenfalls den Mißerfolg an den DIALOG PLANNER melden. Der Planer könnte dann z.B. neu planen oder auf den intendierten Zug verzichten und das Wort abgeben.⁹

7.5.4 Diskussion

Falls die Bewertung bei der Äußerungsauswahl mit Hilfe der Antizipation nicht oft anders ausfällt als die Bewertung ohne Antizipation, kann man oft auf GAF verzichten oder zumindest GAF zunächst nur auf die beste Äußerung anwenden. Wenn beispielsweise in der initialen Planungsphase der Äußerungsauswahl eine bestimmte Anmerkung mit Abstand den größten Wert aufweist, kann dann in der späteren Planungsphase die globale Antizipation nur auf diese Anmerkung angewendet werden.

Der Ansatz der selektiven Antizipation weist eine gewisse Analogie zur *approximativen Planung* (Ginsberg, 1995) auf. Während die meisten Planer korrekte und vollständige Pläne liefern. Approximationsplaner garantieren jedoch nicht, daß die gelieferten Pläne vollständig und korrekt sind.

⁹In der zwischenmenschlichen Kommunikation erinnert dies an die Situation einer Person, die den Mund aufmacht, um etwas zu sagen, jedoch nach genauerem Nachdenken über die Folgen schweigt. Diesem Problem kann man durch *conditional planning* (Russell & Norvig, 1995, S. 392–400) begegnen. Der Dialogplaner sollte dann beim Verzicht auf die Antizipation während der Initialplanungsphase einen alternativen Plan für den Fall, daß sich nachher eine Anmerkung als nicht mehr günstig erweist, haben.

7.6 Der Ansatz der selektiven Anpassung

7.6.1 Überblick über den Ansatz

Für eine präzise Antizipation wird das Modell des Gesprächspartners in der Unterinstantiierung mit Hilfe der in der Hauptinstantiierung verfügbaren Informationen über die Einschätzung seines Wissens, seiner Bewertungskriterien und weiteren Merkmalen stets auf den neuesten Stand gebracht. Eine ständige Anpassung in der Unterinstantiierung kann daher zeitaufwendig werden. Jedoch kann dieser Prozeß selektiv erfolgen, zumal nicht jede Anpassung Auswirkungen auf die Antizipation des nächsten Dialogzuges hat.

7.6.2 Formalisierung

Durch das in der Hauptinstantiierung verwaltete Benutzermodell über den Gesprächspartner kann bestimmt werden, welche Anmerkungen des Benutzers bzw. welche vom System abgeleiteten Informationen eine Auswirkung auf das Benutzermodell in der Unterinstantiierung haben. Dieses Benutzermodell ist maßgebend für die Genauigkeit der in der Unterinstantiierung ausgewählten Anmerkungen und folglich für die Genauigkeit der Ergebnisse der globalen Antizipation. Zur Realisierung des Ansatzes der selektiven Anpassung kann das System über eine Prozedur DECIDE-IF-UPDATE-NEEDED verfügen, die besagt, wann die Unterinstantiierung angepaßt werden soll. Folgendes Kriterium kann benutzt werden für die Entscheidung, ob angepaßt wird oder nicht:

$$\sum_i (w_{S_i} - \hat{w}_{U_i}) \leq \theta$$

Dabei ist w_{S_i} ein Gewicht der Dimension i für \mathcal{S} und \hat{w}_{U_i} der Erwartungswert dieses Gewichtes für \mathcal{U} . Diese Prozedur kann dann gelegentlich aufgerufen werden, bevor der EVALUATION HANDLER die Einschätzung der Parameter \mathcal{U} s an die Unterinstantiierung übergibt. θ ist z.B. ein Maß dafür, wie sehr \mathcal{U} sich in seinen Merkmalen von \mathcal{S} unterscheidet, als daß die Unterinstantiierung angepaßt werden sollte.

Ein System sollte über eine gute Heuristik verfügen, um zu bestimmen, wann auf jeden Fall die Unterinstantiierung angepaßt werden sollte. Eine Anpassung könnte z.B. nach jeder Äußerung des Benutzers erfolgen, aus der das System relevante Information für das Benutzermodell ableitet. Die Unterinstantiierung könnte auch nach einer Äußerung von \mathcal{S} angepaßt werden, bei der das System annimmt, daß relevante Änderungen in den Eindrücken des Benutzers herbeigeführt werden.

7.6.3 Realisierungsbeispiel

Obwohl es für GAF kein großer Aufwand ist, N potentielle Reaktionen von \mathcal{U} statt nur einer zu liefern, ist es für \mathcal{S} zeitaufwendig, zu betrachten, wie es zu jeder dieser Reaktionen antworten würde. Wenn \mathcal{S} dabei alle Fälle betrachten sollte, wie sich \mathcal{U} von ihm unterscheiden kann, würde dies sämtliche Anpassungen der Unterinstantiierung zur Simulation

von \mathcal{U} erfordern. Zur Reduzierung des Aufwandes der globalen Antizipation könnte auf die aufwendigen Anpassungen verzichtet und sich mit einer einfacheren Strategie begnügt werden. Wenn das System beispielsweise anfangs zu der Einschätzung kommt, daß es mit seinen Bewertungsstandards und Dialogverhalten dem Dialogpartner sehr ähnelt und daß die Folgen des Verzichtes auf die Antizipation nicht gravierend werden.

7.6.4 Diskussion

Der Ansatz der selektiven Anpassung illustriert im Extremfall das von Higgins (1981) verwendete Konzept von *inferred similarity* (s. die Besprechung in Abschnitt 4.3). Hierbei geht der Sprecher davon aus, daß er ähnliche Merkmale und Präferenzen hat wie der Dialogpartner. Auch wenn er sich vom Dialogpartner unterscheidet, ist der Unterschied nicht so groß, als daß er in der Situation beträchtenswert ist. Das System geht in diesem Fall davon aus, daß die Anpassung der Unterinstantiierung die Ergebnisse der Antizipation nicht wesentlich beeinflussen wird.

In der zwischenmenschlichen Kommunikation beschränkt man sich manchmal bei der Anpassung des Modells nur auf die Faktoren, die einen Einfluß auf die Bewertungen haben könnten. Das sind die Fälle, in denen “judges decide not to consider all possible factors relevant to the judgment, but, instead, select those factors that are the best predictors or meet a certain threshold of predictive power because consideration of additional factors with less predictive power would lower the overall predictive accuracy” (Higgins, 1981, S. 147).

7.7 Der Ansatz der mehrstufigen Vorausschau

7.7.1 Überblick über den Ansatz

In den bis jetzt betrachteten Fällen hat das System die Rolle des Benutzers übernommen, um dessen potentielle Reaktion(en) vorherzusagen. Die Antizipation beschränkte sich auf eine Ebene, nämlich auf die Vorhersage eines Zuges. Das System könnte weitere Züge antizipieren. Zwei Fälle sind zu unterscheiden: die mehrstufige und die rekursive Antizipation.

Im Falle der mehrstufigen Antizipation sagt \mathcal{S} seine Antwort auf die bereits antizipierte Reaktion von \mathcal{U} voraus und antizipiert die Reaktion von \mathcal{U} auf diese Antwort, etc. (s. Abbildung 7.7)

7.7.2 Formalisierung

Durch *stufenweise Analyse* (Bühlmann *et al.*, 1975, S. 73–74) läßt sich der Erwartungswert des Gesamtnutzens eines Zuges bestimmen. Der Erwartungswert bezieht sich auf die Antizipation der Reaktionen des Dialogpartners auf eine bestimmte Antizipationsstufe, gefolgt von der potentiellen Antwort des Systems bis zu einer vorgegebenen Antizipationsstufe. Abbildung 7.8 stellt die Antizipationsstufe i dar. Die Abbildung 7.9 zeigt den ent-

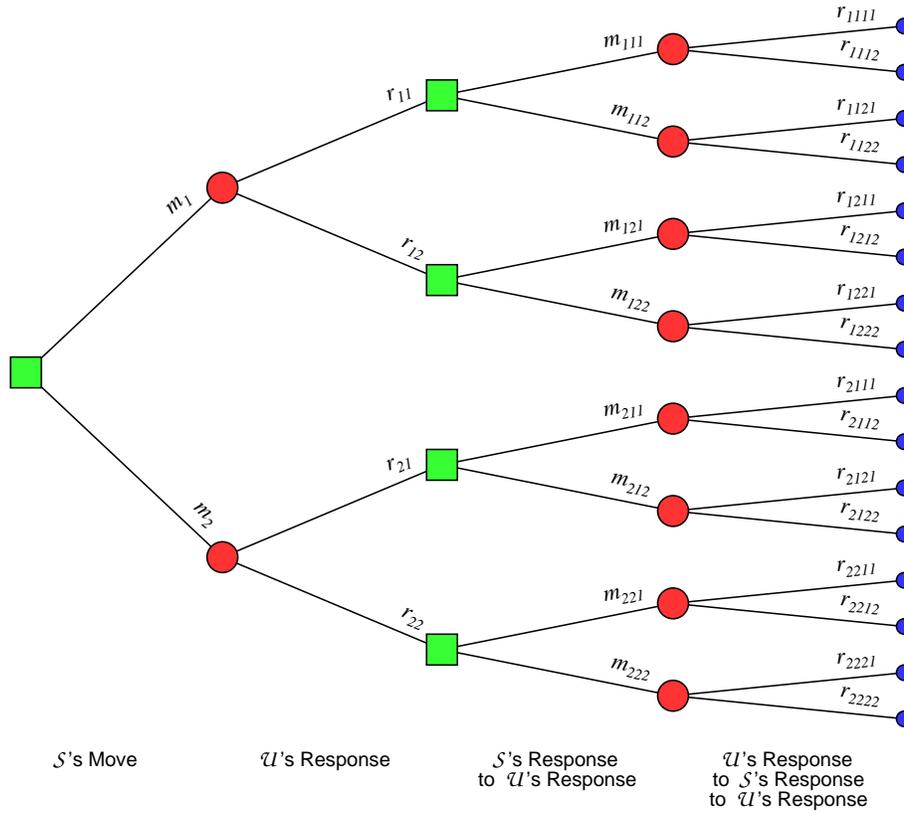


Abbildung 7.7: Darstellung eines Entscheidungsbaumes der Entscheidungsstufe 2.

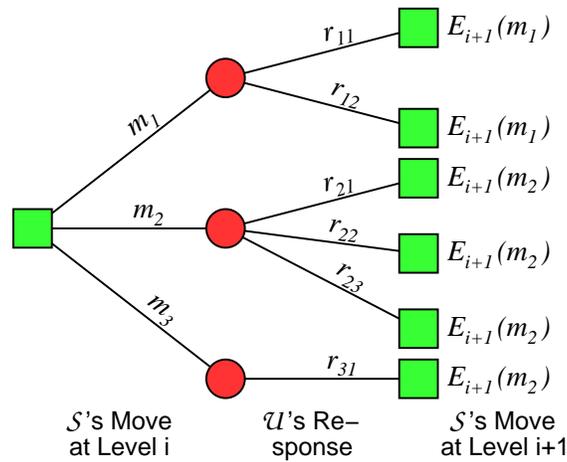


Abbildung 7.8: Darstellung des Übergangs von Stufe i zu $i + 1$.
 (Das Bild wurde (Bühlmann *et al.*, 1975) nachempfunden.)

sprechenden Algorithmus. Für jede Antizipationsstufe wird das bisherige Antizipationsverfahren angewandt: Antizipation der Reaktionen des Dialogpartners gefolgt von der Systemantwort. Der Gesamtnutzen jeder Verzweigung im Antizipationsbaum ergibt sich aus der Summe der Nutzen für die einzelnen Stufen. Dabei stellt $D(w)$ den Nutzen des Zuges w

```

function BEST-MOVE-LOOK-AHEAD(type, constraints, n) returns a
dialog move of type type that fulfills constraints after looking ahead up to
anticipation level n

for  $i \leftarrow n$  downto 1 do
   $N(z_i^j) = \sum_k p_i^j [D(w_i^j) + N(E_{i+1}^k)]$ 
  /* Für alle  $z_i$  wird der Erwartungswert berechnet und festgehalten */
   $N(E_i^k) = \max_j [D(a) + N(z_i^j)]$ 
  /* Für alle  $E_i$  wird der Erwartungswert berechnet, den maximalen Wert
ermittelt und das indice gemerkt */
end

```

Abbildung 7.9: Stufenweise Analyse.

dar; $N(w)$ den Erwartungswert des Nutzens von w ; p_i^j die Wahrscheinlichkeit der Reaktion r_i^j und E_i^k den Entscheidungsknoten der i -ten Stufe, welche dem Ast mit Index k folgt. Es wird die Verzweigung mit dem höchsten Erwartungsnutzen gewählt.

7.7.3 Diskussion

Die Antizipation der Reaktion(en) des Dialogpartner auf einen geplanten Zug weist viele Analogien mit den heuristischen Suchverfahren auf wie z.B. bei Spielbäumen. Dort wird ein Baum bis zu einer bestimmten Tiefe expandiert, eine statische Bewertungsfunktion auf die Blätter angewandt und die Werte der Blätter werden nach oben zu den Elternknoten bis zum Wurzel propagiert. Bei der Propagierung der Werte wird z.B. bei Spielen der *Mini-max*-Algorithmus¹⁰ verwendet. Ein wesentlicher Nachteil des Minimax-Verfahrens ist — wie in der Grundversion des Antizipationsalgorithmus —, daß dort *alle* Knoten betrachtet werden müssen. Der Effizienz halber wurden Verfahren entwickelt z.B. *Alpha-Beta*, um den Suchraum dadurch zu begrenzen, daß nicht alle Verzweigungen der Suchbäume durchsucht werden müssen.

Eine Erweiterung dieses Ansatzes besteht darin, die Antizipationstiefe variabel zu halten. Je nachdem, welche Knotenexpansionen Erfolg versprechen, wird bei diesen Knoten mehrstufig vorausgeschaut. Durch Einschätzung des erwarteten Wertes einer potentiellen Berechnung wird selektiv entschieden, ob ein Knoten weiter expandiert werden soll oder nicht.

¹⁰Zur Erinnerung: Beim Minimax wählt ein Spieler (MAX), der am Zuge ist, unter einer endlichen Menge potentieller Züge denjenigen, der seine Auszahlung (Payoff) maximieren wird; der Gegenspieler (MIN) wird einen Zug wählen, um den Payoff (Auszahlung) von MAX zu minimieren.

Hierzu können die Verfahren der Antizipation nach Wert der Berechnung (Abschnitt 7.4) eingesetzt werden.

Der Ansatz der mehrstufigen Antizipation ist zeitaufwendig. Je weiter vorausgeschaut wird, desto unsicherer werden die antizipierten Reaktionen. Glücklicherweise ist menschliche Kommunikation nicht so starr wie die meisten Spiele und bietet eine gewisse Flexibilität, die es erlaubt, erweckte Eindrücke im Laufe eines Dialoges zu korrigieren. Bei der Entscheidung darüber, wie weit vorausgeschaut werden soll, ist ein "Trade-off" zwischen dem Nutzen der weiteren Antizipation und dem damit einhergehenden Aufwand zu beachten. So kann die Entscheidung abhängig von den verfügbaren Ressourcen und der Wichtigkeit genauer Antizipation getroffen werden. Hierzu wird im nächsten Abschnitt ein Ansatz vorgestellt, wie selektiv bei der weiteren Vorausschau vorgegangen werden kann.

7.8 Der Ansatz der Antizipation unter Annahme von Zeitdruck beim Dialogpartner

7.8.1 Überblick über den Ansatz

Das System PRACMA bietet unabhängig von GAF Möglichkeiten, zeitadaptiv zu arbeiten, indem es z.B. auf manche Teilschritte verzichtet oder seine Qualitätsanforderungen an das Modul zur Äußerungsauswahl an die verfügbaren Ressourcen sowie an der Wichtigkeit des zu produzierenden Dialogbeitrags anpaßt. Der Ansatz der Antizipation unter Annahme von Zeitdruck beim Dialogpartner besteht darin, PRACMAS zeitadaptive Methoden in der Unterinstantiierung zu verwenden und der Unterinstantiierung den Grad des Zeitdrucks mitzuteilen, so daß diese sich entsprechend verhält.

7.8.2 Formalisierung

Das System kann z.B. den Zeitdruck beim Dialogpartner dadurch berücksichtigen, daß der Faktor *Zeitdruck* als Parameter beim Updaten der Unterinstantiierung vor der Antizipation neben den Parametern, wie der Dialogpartner eingeschätzt wird, mit übergeben. Der Zeitdruck beim Dialogpartner kann sich beispielsweise dadurch bemerkbar machen, daß während der Antizipation nicht alle möglichen Züge untersucht werden, sondern nur diejenigen in Betracht gezogen werden, die im unmittelbaren Zusammenhang mit dem Attribut im Dialogfokus stehen. Abbildung 7.10 gibt den Algorithmus zur Auswahl der potentiellen Dialogzüge unter Annahme vom Zeitdruck beim Dialogpartner an. Hierbei ist zu beachten, daß nur der Dialogpartner und nicht das System unter Zeitdruck steht. In der Unterinstantiierung wird dieser Zeitdruck simuliert.

7.8.3 Realisierungsbeispiele

Zur Simulation des Zeitdrucks beim Dialogpartner wurden in PRACMA drei Fälle betrachtet, an denen der Zeitdruck simuliert werden kann:

function ALLOWABLE-MOVES (*type, constraints*) **returns** a set of potential moves

restrict the selection process to moves whose topics are sufficiently close to the selected item in the dialog focus.

select all potential moves within an interval $\Delta(\frac{1}{p})$

where p is the current time pressure

return the resulting set of moves

Abbildung 7.10: Algorithmus zur Bestimmung der potentiellen Dialogzüge in PRACMA unter Berücksichtigung von Zeitdruck.

(Dieser Algorithmus wird aufgerufen, wenn das System unter Zeitdruck steht. In diesem Fall werden nur diejenigen Züge betrachtet, die in unmittelbarem Zusammenhang mit dem erwähnten Attribut stehen.)

- Beschränkung der Breite des Fokus
In diesem Fall werden nur diejenigen Züge betrachtet, die in unmittelbarem Zusammenhang mit dem erwähnten Attribut stehen (s. Abbildung 7.10). Dieser Vorgehensweise liegt folgende Annahme zugrunde: “Wenn der Dialogpartner sich schnell äußern muß, wird er nicht viel Zeit zur Untersuchung aller potentiellen Züge aufwenden, sondern er wird nur wenige relevante Züge betrachten.”
- Beschränkung der Anzahl der zu berücksichtigenden Bewertungsdimensionen
In PRACMA hat jedes Attribut im Bewertungsformular Implikationen für eine oder zwei der sechs Bewertungsdimensionen (Schäfer, 1997). Im Extremfall kann während der Antizipation unter Annahme von Zeitdruck beim Dialogpartner, statt beide Dimensionen, nur die Hauptdimension berücksichtigt werden. (Beispielsweise hat “Motorleistung” Auswirkung auf die Dimensionen “Sportlichkeit” und “Umweltfreundlichkeit”.) Während im Dialog ohne Zeitdruck für die Simulation des Dialogpartners sämtliche Attribute in der Unterinstanziierung auf zwei Dimensionen bewertet werden, kann die Bewertung auf die Hauptdimension beschränkt werden, falls der Dialogpartner unter Zeitdruck ist. (So wird beispielsweise das Attribut “Motorleistung” nur auf die Hauptdimension “Sportlichkeit” bewertet.) Abbildung 7.11 stellt ein Beispiel der Implementierung dieser Strategie dar.
- Für die Interpretation der Äußerungen des Verkäufers kann die Interpretation in der Unterinstanziierung auch auf eine Dimension beschränkt werden.

7.8.4 Diskussion

Die in PRACMA realisierten Fälle der Simulation der Antizipation unter Zeitdruck können auch kombiniert werden. Hierbei kann in Abhängigkeit von einer Skala der Dringlichkeit des Zeitdrucks entschieden werden, wieviele dieser Fälle kombiniert werden sollen. Nach der Verbalisierung einer Äußerung unter Annahme von Zeitdruck beim Partner kann auch die Strategie der Antizipation im Nachhinein angewandt werden.

<p>Buyer: Does the car have wide tires? Seller: It doesn't have wide tires. Nach der Verbalisierung der Antwort, plant das System eine zusätzliche freiwillige Anmerkung, um dem beim Dialogpartner durch die negative Antwort hervorgerufenen Eindruck entgegenzuwirken. Dazu wird GAF aufgerufen, um vorherzusagen, wie der unter Zeitdruck stehende Dialogpartner auf diese geplante Äußerung des Systems reagieren wird. — Anticipating reaction to possible comment — Seller: Its average mileage of all tires is very low. Buyer: That's OK. Buyer: Is the car's overall mileage high? — Planned comment rejected! —</p>	<p>— Anticipating reaction to alternative comment — Seller: It has self-tightening seat belts. Buyer: That's OK. Buyer: Does the car have side impact protection? — Planned comment accepted! —</p>
---	--

Abbildung 7.11: Beispiel einer Antizipation unter Annahme von Zeitdruck beim Dialogpartner.

(Ohne Zeitdruck hätten die Dimensionen "Sicherheit" und "Sportlichkeit" berücksichtigt werden sollen. Bei Annahme von Zeitdruck beim Dialogpartner wird jedoch nur die Dimension "Sicherheit" berücksichtigt.)

7.9 Zusammenfassung

In diesem Kapitel wurde aufgezeigt, wie die Antizipation ressourcenadaptiv angewandt werden kann. Die präsentierten Ansätze lassen sich in drei Gruppen untergliedern: ob antizipiert werden soll, wann antizipiert werden soll und wie antizipiert werden soll. Sie können auch kombiniert werden.

Bei dem Satisficing- und Iterationsansatz versucht das System durch Setzen eines Mindestgesamtnutzens die Anzahl der AFLs zu variieren. Sobald ein akzeptabler Zug (also ein Dialogbeitrag, dessen Gesamtnutzen über einem bestimmten Schwellwert liegt) gemacht werden kann, verzichtet das System auf weiteres Vorausschauen. Bei dem Ansatz der Antizipation im Nachhinein wurde die Frage behandelt, wie das System in Abhängigkeit von den zu erwartenden Zeitkosten der Antizipation vor bzw. nach einer Äußerung entscheidet, ob vor oder nach der Verbalisierung eines Dialogbeitrages die Folgen dieser Äußerung antizipiert werden. Der Ansatz der Antizipation nach Wert der Berechnung untersuchte die Frage, ob die Auswahl eines Dialogbeitrages tatsächlich antizipiert werden soll oder ob die Entscheidung nur in Abhängigkeit vom erwarteten Mehrwert der Antizipation getroffen werden kann.

In PRACMA ist die Bestimmung des nächsten Dialogbeitrages in zwei Phasen aufgeteilt, in denen jeweils globale Antizipation eingesetzt werden kann. Bei dem Ansatz der selektiven Antizipation wurde gezeigt, wann es vorteilhaft ist, die globale Antizipation in beiden, nur in der ersten oder erst in der späteren Phase der Äußerungsauswahl einzubeziehen. Die Genauigkeit der Antizipation hängt von der präzisen Modellierung des Dialogpartners in der Unterinstanziierung ab. Bei dem Ansatz der selektiven Anpassung wurde untersucht, wann die Unterinstanziierung angepaßt werden sollte und wann darauf verzichtet werden könnte.

Auf die potentiellen Antworten zu vorausgesagten Zügen des Dialogpartners kann wiederum globale Antizipation angewandt werden. Je weiter vorausgeschaut wird, desto ungenauer werden die antizipierten Reaktionen. Der Ansatz der mehrstufigen Vorausschau

untersuchte, inwieweit Vorausschau notwendig ist. Im letzten Abschnitt wurde gezeigt, wie bei der Antizipation ein Dialogpartner simuliert werden kann, der unter Zeitdruck steht.

Zusammenfassung und Ausblick

Ziel dieser Arbeit war es, im Bereich der Benutzermodellierung am Beispiel nicht ausschließlich kooperativer Dialoge einen allgemeinen Rahmen für die Anwendung der globalen Antizipation zu untersuchen. Wir geben in diesem abschließenden Kapitel eine Zusammenfassung der Ergebnisse und zeichnen einige Erweiterungsmöglichkeiten auf.

8.1 Erzielte Ergebnisse

Die Erforschung und Verwendung globaler Antizipationsrückkopplungsschleifen ist für Dialogsysteme wegen der expliziten Modellierung des Interaktionspartners und der Vorwegnahme von dessen Reaktionen vorteilhaft. Der wesentliche Vorteil der globalen Antizipation liegt in einer besseren Planung des eigenen Verhaltens. In dieser Arbeit wurde ein Rahmen für globale Antizipation in einem rollentransmutierbaren Dialogsystem durch Rollenübernahme vorgestellt. Anhand einer prototypischen Implementierung im System PRACMA wurde die Realisierbarkeit einer solchen Technik in Dialogsystemen belegt. Es wurden auch Möglichkeiten zum effizienten Einsatz der globalen Antizipation erörtert. Die vorgestellten Erfahrungen aus PRACMA haben gezeigt, daß es viele potentielle Verwendungen und Erscheinungsformen der globalen Antizipation in Dialogsystemen gibt.

- *Rollen-Transmutierbarkeit*

Diese Eigenschaft beschreibt die Fähigkeit eines Systems innerhalb einer Dialogsituation, die Gesprächsrollen zu wechseln. So kann das PRACMA-System die Rolle des Käufers oder die des Verkäufers übernehmen. Darüber hinaus kann das System während einer Dialogsituation, in der es eine bestimmte Rolle übernimmt, in einer Simulation gleichzeitig die andere Rolle übernehmen. Somit kann das System die Reaktion des Gesprächspartners auf die geplanten Systemäußerungen vorhersagen.

- *Architektur* Die pragmatische Verarbeitung in PRACMA erfordert eine Systemarchitektur, die eine flexible Interaktion zwischen den einzelnen Agenten mit ihren unterschiedlichen Repräsentations- und Verarbeitungsformalissen unterstützt. Für PRACMA wur-

de zu diesem Zweck die Multi-Agenten-Architektur CHANNELS entwickelt. CHANNELS verwendet Techniken der verteilten KI und der objektorientierten Programmierung. Die einzelnen Systemmodule werden als Agenten realisiert, die ausschließlich durch Nachrichtenaustausch kommunizieren und kooperieren. Diese Kommunikation wird durch ein kommunikationsakt-basiertes Protokoll gesteuert. Kommunikationsakte bestimmen die Art und den Modus der Kommunikation. CHANNELS verwendet die Basisakte INFORM, ASK, REPLY und die Modi *synchron* und *asynchron*. Die Agenten arbeiten als nebenläufige Prozesse. Sie können sowohl innerhalb eines Rechners angesiedelt als auch in einem lokalen Netzwerk verteilt werden.

Es wurden *externe Agenten* eingeführt, die die Kommunikation zwischen zwei Instanzierungen des PRACMA-Systems verwalten.

- *Antizipation*

In PRACMA wurden durch die flexible Systemarchitektur und die Rollentransmutierbarkeit zwei wesentliche Voraussetzungen zur Realisierung globaler Antizipationsrückkopplung geschaffen. Die CHANNELS-Architektur ermöglicht, daß sich PRACMA in einer weiteren kompletten Instantiierung des Systems in der Rolle des Dialogpartners rekursiv aufruft. In dieser mit der anderen Dialogrolle initialisierten Unterinstanzierung wird das Verhalten des Dialogpartners simuliert und antizipiert.

Die realisierte Implementierung erlaubt es, analog zu der Fähigkeit von PRACMA beide Rollen innerhalb eines Dialoges einzunehmen, sowohl das Verhalten des Käufers als auch das des Verkäufers vorherzusagen.

- *Unsicherheit über das Dialogverhalten des Dialogpartners*

Eine präzise Vorhersage der Reaktion des Dialogpartners setzt ein präzises Modell über das Dialogverhalten und die Präferenzen des Dialogpartners voraus. Das System verfügt in aller Regel nicht über ein solches Modell. Der resultierenden Unsicherheit des Systems über die vielen Faktoren, die das Dialogverhalten des Käufers bestimmen, wurde Rechnung getragen. Es wurden Methoden aufgezeigt, wie mit dieser Unsicherheit umgegangen werden kann. In der *runner-up* Strategie wurde für jede vom System geplante Äußerung statt einer einzigen Reaktion eine Menge möglicher Reaktionen vorhergesagt, in der Annahme, daß der vom Dialogpartner letztendlich ausgewählte unter den vorhergesagten enthalten ist. Diese Menge enthält die Reaktionen, die das System *in Betracht ziehen* würde, wenn es in der Lage des Dialogpartners wäre.

Da, auch wenn der Dialogpartner nicht als Gegner im spieltheoretischen Sinne angesehen wird, seine Züge nicht alle gleich wahrscheinlich sind, wurde im probabilistischen Ansatz aufgezeigt, wie diejenigen Züge verstärkt berücksichtigt werden, die wahrscheinlicher sind. Grund einer solchen Vorgehensweise ist, daß manche Reaktionen, abhängig vom Dialogkontext und -zustand, wahrscheinlicher sind als andere.

- *Ressourcenadaptive Antizipation*

Da der rekursive Aufruf eines Systems für jeden potentiellen Dialogzug teuer sein kann, wurden verschiedene Methoden aufgezeigt, wie in Abhängigkeit von den verfügbaren Systemressourcen antizipiert werden kann. So wurden Verfahren entwickelt, *ob, wie, wann* und *inwieweit* die globale Antizipation in einer gegebenen Situation eingesetzt werden sollte.

8.2 Erweiterungsmöglichkeiten

- *Empirische Evaluation*

Eine Verbesserung der in dieser Arbeit implementierten Technik der globalen Antizipationsrückkopplung betrifft die empirische Auswertung der von GAF erzielten Ergebnisse. Hierzu sollten Experimente durchgeführt werden, in denen in einer gegebenen Dialogsituation Versuchspersonen vorhersagen, wie sie selbst reagieren würden. Diese Äußerungen werden dann mit denen von GAF verglichen, und das System lernt gegebenenfalls aus diesem Vergleich.

- *Lernen*

Wie im Abschnitt 7.4 besprochen wurde, kann die Anwendung der globalen Antizipation in Abhängigkeit von dem zu erwartenden Mehrwert getroffen werden. Um eine vernünftige Abschätzung dieses Mehrwertes zu haben, könnten zwei Instantiierungen des PRACMA-System, die eine in der Rolle des Verkäufers und die andere in der Rolle des Käufers, "off-line" mehrere Verkaufssituationen simulieren und somit lernen, vorherzusagen, wie der jeweilige Dialogpartner reagieren wird. Nach der Lernphase können die Ergebnisse kompiliert werden, um bei bestimmten Situation, abzuschätzen, ob es sich lohnt zu antizipieren.

- *Anwendung auf andere Domänen*

In der vorliegenden Arbeit wurden globale Antizipation in nicht ausschließlich kooperativen Dialogen am Beispiel eines Verkaufsgespräches zwischen einem Verkäufer und einem potentiellen Käufer untersucht. Aber auch bei völlig kooperativen anwendungsorientierten Dialogsystemen bietet die globale Antizipation Vorteile. Darüber hinaus läßt sich das AFL-Modell verwenden, um vorherzusagen, wie etwas erreicht werden soll bzw. warum etwas vom Dialogpartner gesagt oder getan wurde. Die Idee der globalen Antizipation läßt sich auf weitere Domänen übertragen, denn "das AFL-Architekturmodell ist für jegliche Art von interaktivem System wegen der expliziten Modellierung des Interaktionspartners und der Kombination vom abduktivem und deduktivem Vorgehen sehr attraktiv." (Wahlster, 1992).

Summary

In man-machine communication, as in everyday dialogs, intelligent dialog behavior of a natural language (NL) system involves the ability of the system to shift its cognitive perspective in order to take the role of the dialog partner and to simulate his or her dialog behavior. One particular way in which a system can anticipate a user's responses is to make use of the system's own comprehension (and perhaps generation) capabilities, temporarily taking the role of the user and simulating his or her behavior. In the user modeling literature, such cases are characterized as so called *anticipation feedback loops (AFL)*.

To date, almost all implemented systems that have employed anticipation feedback have used a limited part of the system to realize a *local* AFL. There are few examples of systems that use a *global* AFL. With a global AFL, a large part of the system's own understanding capabilities is used to anticipate the user's responses. **Chapter 1** motivates the thesis and introduces the issues raised by global anticipation feedback. **Chapter 2** presents an interdisciplinary view of the concept of anticipation. The similarity to the concept of *role-taking* in developmental psychology is reflected in the theoretical framework. Relations to psycholinguistics are also pointed out.

In this thesis, we model AFLs with decision trees: A system \mathcal{S} has to choose among several possible dialog moves $m_1 \dots m_n$ that will have some effect on the user \mathcal{U} (cf. the left-hand side of Figure 8.1). Each m_i has some immediate degree of appeal for \mathcal{S} , which can be conceived of as a utility $U_m(m_i)$. But instead of selecting the move with the highest $U_m(m_i)$, \mathcal{S} anticipates the response r_i that \mathcal{U} is likely to make to each m_i ; and each r_i is itself associated with a utility $U_r(r_i)$. \mathcal{S} chooses the move with the highest total utility $U_m(m_i) + U_r(r_i)$. An AFL can be invoked in the step where \mathcal{S} anticipates \mathcal{U} 's response r_i . The point of doing so is that the determination of $U_r(r_i)$ in addition to $U_m(m_i)$ may affect \mathcal{S} 's choice of a move. This simplicity and generality is, however, associated with a number of limitations and challenges, as the following issues raised by global AFLs show:

1. *Within-dialog transmutability.* A system that uses a global AFL must be able to take the role of the other participant in the type of dialog it conducts. By contrast, a local AFL presupposes only that the system be able to do some part of the processing required for the other role; and this common processing may involve a generic subtask, such as syntactic analysis, which is relatively independent of any particular dialog role. For human beings, transmutability is often given, because people learn to take many different roles in dialogs in the course of their everyday experience. (For example, even a professional salesperson often has the opportunity to act as a customer.) But systems that employ user modeling techniques are typically designed to play a particular role. It may therefore require a considerable additional investment to enable them to switch to the role of their dialog partner.

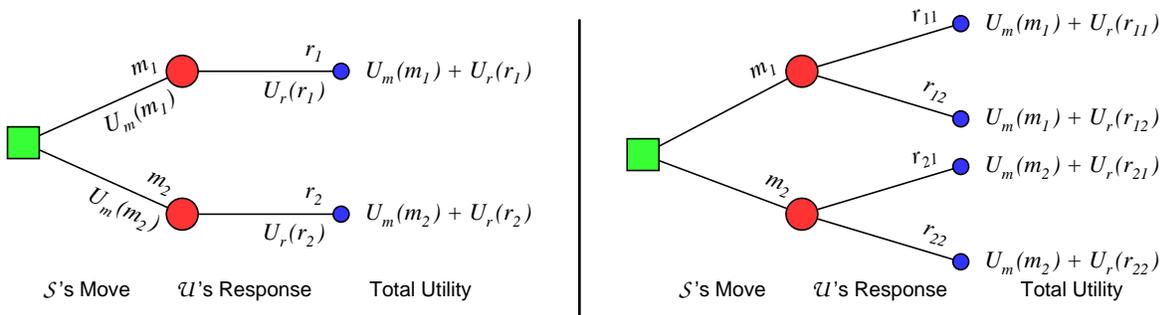


Figure 8.1: Illustration of the concept of global anticipation with decision trees.

(The left-hand side shows a simple decision tree that illustrates the strategy of anticipating the user's response before selecting a move; the right-hand side is a generalization of the decision tree in the left-hand side that takes into account uncertainty about the user's responses.)

2. *Communication between system instantiations.* The dialog system has to be able to invoke itself in another system instantiation, in the role of the dialog partner, without interfering with its workings in its original role.

3. *Uncertainty about factors that determine the responses of the dialog partner.* The decision tree in the left-hand side of Figure 8.1 presupposes that \mathcal{S} can predict \mathcal{U} 's response to any given move m_i with certainty. The more general case is shown in the right-hand side of the figure: When considering a move m_i , \mathcal{S} can at best narrow the possible responses of \mathcal{U} down to a set $\{r_{ij}\}$. In the case of a global AFL, this uncertainty is due to the fact that \mathcal{U} 's response will be influenced by some factors that are not entirely known to \mathcal{S} . In other words, \mathcal{S} does not know exactly how to pretend to be the user. The question arises of how \mathcal{S} can best deal with this uncertainty.

4. *Resource-adaptive Anticipation.* In general, it is relatively time-consuming for a dialog system to anticipate the complete processing by the user of a possible dialog move by the system. The question therefore arises of how the computational cost of using global AFLs can be minimized.

The present thesis explores the relatively uncharted area of global anticipation feedback with the help of the listed issues. These questions are investigated within the NL dialog system PRACMA (Jameson *et al.*, 1994; 1995), which we use here as a testbed.

Flexible Architecture and Transmutability

Chapter 3 introduces some basic architectures for NL systems and then focuses on the CHANNELS architecture, which uses techniques from distributed AI and from concurrent object-oriented programming. (PRACMA is implemented within this architecture.) In CHANNELS (**C**ooperating **H**eterogeneous **A**gents for a **N**atural-**L**anguage **S**ystem), the system's modules, realized as agents, interact cooperatively through a *communication-act-based protocol* that governs the exchange of messages. Each message is characterized by attributes including: the sender, the recipient(s), the type of communication act, the mode

of communication (synchronous or asynchronous), and the actual content of the message. The communication acts (INFORM, ASK, and REPLY) define the nature of the interaction among the agents. For instance, an ASK requests the recipient(s) to send information back to the originator of the message, while an INFORM passes information from one module to another. Messages communicate information between a sender agent and a receiver agent either *synchronously* or *asynchronously*. The communication is synchronous if the sender requires a response before continuing processing; until a response is received, it remains in the state *waiting*. With asynchronous communication, the sender can engage in further processing before receiving a response. The agents run concurrently as simulated processes over a local network using PVM (Geist *et al.*, 1994) and ICE (Amtrup, 1994). For the distribution of the computation of the agents over a local network and the realization of the communication between various instantiations of the PRACMA system, the CHANNELS architecture supports *external agents*. (An external agent is a virtual agent within a PRACMA instantiation to which the agents within the system can refer and send messages but which is actually located in another instantiation of the system on the same or on another machine.) These external agents are used for the realization of global anticipation feedback

Most natural language systems are restricted to cooperative dialogs, whereas PRACMA (**P**rocessing **A**rguments between **C**ontroversially-**M**inded **A**ctors) models partly noncooperative dialogs, which are widely represented in everyday situations. PRACMA models dialogs in which a person (to be called the *seller*) is trying to sell his or her used car to a potential *buyer*. PRACMA is able to take the role of either the seller or the buyer within its dialog situation. This ability to switch roles can be seen as a particular variant of the property of dialog systems that Wahlster and Kobsa (1989, p. 30) define as *transmutability*: the property of being adaptable to applications that differ with respect to “dialog type, user type, and intended system behavior”. This within-dialog transmutability in PRACMA enables the system to realize various types of AFLs—in particular, truly global AFLs, in which the system consults a complete instantiation of itself in order to anticipate the dialog partner's responses.

The goals of the two dialog participants in PRACMA's example domain conflict to a certain degree: The buyer wants to get the best possible information on which to base a decision about the car, whereas the seller would like to sell the car, whether or not it is really suitable for the buyer. When \mathcal{S} is the seller, this conflict increases the importance of anticipation feedback for \mathcal{S} , because it increases the range of utility that \mathcal{U} 's responses can have for \mathcal{S} . For example, if \mathcal{U} decides to ask about an attribute of the car, it makes a big difference to a noncooperative seller whether this attribute happens to be one on which the car rates highly or poorly; for a cooperative seller, this difference would not be so important. (The car is modeled in terms of six evaluation dimensions: Reliability, Economy, Safety, Environment, Comfort and Sportiness.)

PRACMA uses a simple local AFL when taking the role of a seller who is not able or willing to devote much attention to the dialog. This AFL is local in that the only part of the system's comprehension capabilities that \mathcal{S} makes use of is \mathcal{S} 's capability to derive the evaluative implications of a given comment about a car. The use of a global AFL in PRACMA becomes necessary when \mathcal{S} tries to anticipate a more complex response by \mathcal{U} : What \mathcal{U} 's next

dialog move will be if \mathcal{S} makes a given dialog move. It can be important for \mathcal{S} to anticipate \mathcal{U} 's next move. For example, if \mathcal{U} asks a question, it might concern some topic that \mathcal{S} would prefer to avoid (e.g., an attribute with respect to which the car is weak); if this seems likely, \mathcal{S} should consider not making the comment that it originally intended to make.

Realization of Global Anticipation Feedback

The job of choosing the system's next dialog move is divided hierarchically between the PRACMA agents DIALOG PLANNER and COMMENT AND QUESTION HANDLER (nicknamed CQH). The DIALOG PLANNER, an incremental planner, decides what *type* of move to make. In doing so, it takes into account a variety of factors, including the dialog history (stored in the agent PRAGMATIC DIALOG MEMORY) and various motivational parameters (stored in the agent EGO). Once it has decided on a particular type of move, it asks CQH to choose a specific move of that type.

A global AFL cannot be realized as a query to one of agents that make up the system, as it requires an invocation of the entire system. Therefore, a special agent, the GLOBAL ANTICIPATOR (nicknamed GAF for Global Anticipation Feedback) maintains a *subordinate instantiation* of PRACMA in a separate COMMONLISP image which may be located on another computer. GAF initializes this instantiation to take the other dialog role. This instantiation does not engage in any direct interaction with the user but rather responds to inputs from the GAF that controls it. (Recall that the CHANNELS architecture provides *external agents* that make possible communication between different instantiations.) This subordinate image contains instantiations of all of PRACMA's agents, including GAF. The GAF in the main instantiation accepts and responds to queries from agents that require global anticipation feedback. One task of GAF is to ensure that the subordinate instantiation constitutes as realistic a model of the actual user as is possible given the information available to the main instantiation. To accomplish this, GAF regularly queries the agents EVALUATION HANDLER and DOMAIN BELIEF HANDLER in the main instantiation, asking them for their assessments of \mathcal{U} 's evaluation criteria and knowledge; the resulting estimates are used to initialize the corresponding agents in the subordinate instantiation.

Anticipation of the Buyer's Processing

In **Chapter 4** we present a straightforward use of global anticipation feedback for the prediction of the reaction of the buyer. In this approach, GAF uses an instantiation of \mathcal{B} (the buyer) with exactly the same evaluation standards, domain knowledge, and dialog strategies that \mathcal{S} herself would have if she were buying a used car. In this approach, (called *situational role taking* by Higgins (1981)), a communicator predicts the responses of the communication partner by asking, "What would I do, think, feel, see, etc., if I were in that situation?". The use of this approach may be based on *assumed similarity*, where the communicator does not even consider the possibility that he or she may differ in important ways from the dialog partner; or on *inferred similarity*, a considered judgment that such differences are not important enough to have to be taken into account.

\mathcal{S} anticipates \mathcal{B} 's next dialog move by temporarily pretending that it is the buyer in the dialog. More concretely, \mathcal{S} consults a simultaneously active instantiation of PRACMA in which the system is taking the role of the buyer. \mathcal{S} uses the strategy described in the left-hand side of Figure 8.1. For each potential utterance m_i whose utility $U_m(m_i)$ is above a given threshold δ , m_i is sent to the subordinate instantiation (where the system takes the role of the buyer) to anticipate the potential reaction r_i of \mathcal{B} . The resulting utility $U_r(r_i)$ is then added to the already computed utility $U_m(m_i)$. \mathcal{S} eventually chooses the move with the highest expected overall utility $U_m(m_i) + U_r(r_i)$.

Taking Uncertainty into Account

In **Chapter 5** we present an approach in which \mathcal{S} takes into account his or her uncertainty about the factors that determine the responses of the dialog partner.

The algorithms used in the simple variant have presupposed that, if \mathcal{S} takes the trouble to use a global AFL, \mathcal{S} will always anticipate \mathcal{U} 's response correctly. This assumption is less realistic than the conceptualization shown in the right-hand side of Figure 8.1. But the question remains: How can global anticipation, given a possible move m_i , return not just a single anticipated response r_{i1} , but rather a *set* of possible responses $\{r_{i1} \dots r_{in}\}$? This question is difficult to answer in a general way. But within the framework presented here, the problem is manageable if \mathcal{S} considers only other responses *of the same type* as the most likely response r_{i1} . The basic idea is to exploit the way in which CQH chooses moves of a given type, namely by evaluating all reasonable moves of that type. Although this algorithm has been discussed so far only with respect to its use in the main instantiation, it is of course also used in the subordinate instantiation, when \mathcal{U} is being simulated. For example, when the subordinate instantiation, in the role of the buyer, chooses a specific question to ask, the CQH of the subordinate instantiation first considers all questions that have some relevance to the current dialog focus and then chooses the one with the highest UTILITY-OF-MOVE. A consequence is that when the subordinate instantiation has produced a move r_{i1} for \mathcal{U} as a response to the move m_i by \mathcal{S} , GAF can ask CQH which moves it considered that had a UTILITY-OF-MOVE that was *almost as high* as that for r_{i1} . The assumption underlying this query is the following one: The moves that rated almost as high as r_{i1} for the simulated \mathcal{U} represent the most likely alternative hypotheses about how \mathcal{U} will respond to m_i .

By applying the described strategy, \mathcal{S} becomes more cautious, asking not just “What response would I make in this situation?” but rather “What responses would I *consider* making in this situation?”. The resulting N responses that would rate highest for \mathcal{S} if she took the role of \mathcal{B} can be viewed as the N most likely responses for \mathcal{B} to make. The assumption underlying this *runner-up strategy* is: “Even though I have a good deal of uncertainty about the \mathcal{B} I'm talking with, he is probably not so different from me that he would choose a move that was not among my N most preferred moves”. (Actually these N moves are those whose overall utility exceeds a given threshold ε .) The higher N (i.e. the lower ε) is set, the more likely it is that \mathcal{B} 's actual response will be anticipated. If \mathcal{S} checks that she could in turn respond satisfactorily to all of these N possible responses, she is relatively unlikely to overlook potential problems. (Note that anticipating a set of possible responses

is in itself no more time-consuming than anticipating a single one, since CQH always has to consider several moves. The list of almost-chosen moves is already available to the CQH in the subordinate instantiation as a by-product of its processing.)

Anticipation of the Seller's Processing

A buyer in general knows that the seller is maintaining a model of his interests and knowledge, and the buyer may try to anticipate how this model—and the seller's dialog moves based on this model—will be affected by particular behaviors of \mathcal{B} . **Chapter 6** shows how PRACMA, when taking the role of the buyer, anticipates the behavior of the seller and the seller's modeling of the buyer.

An important topic in this approach is the notion of an *impression*. In the role of the seller, \mathcal{S} maintains a model of the buyer. This model contains the impressions that the buyer has about the car. The seller chooses his dialog contributions so that these impressions of the buyer are altered. These impressions are estimates of \mathcal{B} 's interests in the six evaluation dimensions. To anticipate the reactions of the seller, \mathcal{S} simulates the processing of the seller and estimates the seller's impressions about the buyer's interests in the evaluation dimensions before and after an utterance. (Recall that six evaluation dimensions are considered by PRACMA: Reliability, Economy, Safety, Environment, Comfort and Sportiness.) For each potential utterance, \mathcal{S} in the role of the buyer anticipates whether this utterance increases the estimate of his interest in the main evaluation dimension that the seller ascribes to the buyer. The utility is a function of the resulting impression shift concerning the buyer's interests. \mathcal{S} ultimately chooses his dialog contribution according to its utility.

Resource-Adaptive Anticipation

Because global anticipation feedback is computationally expensive, a system must be selective in applying it. **Chapter 7** deals with resource-adaptive anticipation and discusses several possible types of taking into account resource limitations. The presented approaches can be classified according to criteria *whether*, *how*, *when* global anticipation should be used and *how much* look-ahead is required.

The strategies of global anticipation presuppose that all branches of the considered decision trees are to be processed completely. But if a system is willing to sacrifice some decision quality according to the available system resources, computation can be done more selectively. For example, the satisficing approach shows how the search for a move can be terminated as soon as one as an acceptable overall utility has been found—i.e. the overall utility of an intended move exceeds some threshold. A special case of the satisficing strategy is the iterative approach: The system does not generate all possible moves m_i at once but rather generates and evaluates them one by one, using the result of the evaluations $U_r(r_{ij})$ to guide the generation of the next move m_{i+1} .

The approach of *post-utterance anticipation* decides under time pressure *when* to anticipate according to the expected probability that the move selected without global anticipation is

adequate or not. Depending on this probability, the system might decide to use global anticipation before generating an utterance or to refrain from using global anticipation beforehand and use it after the generation of the selected move while the dialog partner is analyzing the system utterance. If necessary, the system initiates additional utterances to compensate for the undesired consequences of the verbalized utterance.

In the approach of *anticipation according to the value of computation*, the system first has to decide whether it is worthwhile to anticipate before the verbalization of a move or not. The system first selects the two best moves without use of global anticipation. It then considers the expected improvement in utility compared to choosing a move without applying global anticipation to these moves. According to the variance of the distribution of the utilities the system chooses the move which is expected to be the best or it decides to actually anticipate in the cases where there is a good deal of uncertainty.

The next approach deals with the restriction of the use of global anticipation feedback to a late stage in the utterance planning process. For example, in PRACMA, when deciding what type of dialog move to make next, the dialog planner often asks the module for selecting utterances whether it is possible to make a worthwhile move of a given type. But even if the answer is positive, the planner may end up choosing a different type of dialog move, since other criteria are also relevant. Because this type of query by the DIALOG PLANNER comes frequently, it would be impractical for the selection module to invoke the global anticipation module (perhaps repeatedly) every time it answers such a query. Instead, it simply checks whether there is some move of the type in question that is acceptable. It is only when (and if) the planner subsequently asks the selection module actually to select a move of this type that it takes the trouble to invoke the anticipation module. When it does so, it may of course discover that all of the possible moves rate poorly. In such cases the system's behavior is similar to that of a person who begins to say something and then has second thoughts about the wisdom of doing so. The occasional appearance of this phenomenon seems to be a necessary consequence of the limited time that the system can spend anticipating the user's responses during the early planning of a dialog contribution.

The next strategy handles the *selective updating of the subordinate instantiation* on the basis of estimates of the user's knowledge, evaluation criteria, and other characteristics. Performing this updating frequently can be not only time-consuming but also wasteful. For example, only a small part of the updates may actually have any effect on the anticipation of \mathcal{U} 's next move. A simplified approach is to do the updating only occasionally—or even only once, at the beginning of a dialog, on the basis of the initially available information about the user.

The next approach deals with the *minimizing look-ahead*. In principle \mathcal{S} could expand the nodes of the used decision trees so as to look at least several moves ahead as is done, e.g., in game-playing programs. (Note that the system views the user not as an adversary but simply as a cause of events that have a limited degree of predictability.) Another type of look-ahead can be achieved if global anticipation is allowed to occur within the subordinate instantiation: When anticipating \mathcal{U} 's next response, \mathcal{S} considers how \mathcal{U} will anticipate \mathcal{S} 's subsequent move, etc. An important limitation of both of these types of look-ahead is their relatively high computational cost. For example, to extend a decision tree beginning with

one of the right-most nodes, \mathcal{S} has to go through the whole process of generating possible moves, a process which can involve all of the agents which make up the system. Note also that as the tree gets deeper, the additional expansions become less worthwhile, as they concern dialog moves which are increasingly unlikely ever to occur. One reasonable approach is to make the amount of look-ahead dependent on (a) the resources available to the system and (b) the assessed importance of correct anticipation.

The last approach deals with the anticipation assuming that the dialog partner is under time pressure. The simulation of this time pressure involves e.g. the restriction of the selection process in the subordinate instantiation to moves whose topics are sufficiently close to the selected item in the dialog focus.

Conclusion

Chapter 8 looks back on the themes developed in the previous chapters to see what has been contributed: a general framework for realizing and comparing variants of global anticipation feedback; one solution to the problem of realizing communication between system instantiations; a manageable approach to the handling of uncertainty about the user; a bidirectional anticipation ability; and some approaches to resource-adaptive anticipation. The chapter points out what remains to be done as future work.

Literaturverzeichnis

- Agha, G. (1986). *ACTORS: A model of concurrent computation in distributed systems*. Cambridge, MA: MIT Press.
- Agha, G. (1991). The structure and semantics of actor languages. In J. W. de Bakker, W. P. de Roever & G. Rozenberg (Hrsg.), *Foundations of object-oriented languages* (S. 1–59). Berlin, Heidelberg: Springer.
- Allgayer, J., Jansen-Winkel, R., Reddig, C. & Reithinger, N. (1989). Bidirectional use of knowledge in the multi-modal NL access system XTRA. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, Detroit, MI, 1492–1497.
- Amtrup, J. W. (1994). *ICE: INTARC Communication Environment – Design und Spezifikation*. Technical Report VM-Memo 48, Universität Hamburg, Hamburg.
- Amtrup, J. W. (1997). ICE: A communication environment for natural language processing. *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA-97)*, Las Vegas, NV, 69–77.
- André, E., Graf, W., Heinsohn, J., Nebel, B., Profitlich, H.-J., Rist, T. & Wahlster, W. (1993). *Personalized Plan-Based Presenter*. Technical report, German Research Institute for Artificial Intelligence (DFKI), Saarbrücken. DFKI Document D-93-05.
- André, E. & Rist, T. (1996). Coping with temporal constraints in multimedia presentation planning. *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, Portland, OR, 142–147.
- Astington, J. W. (1993). *The child's discovery of the mind*. Cambridge, MA: Harvard University Press.
- Berlo, D. K. (1960). *The process of communication: An introduction to theory and practice*. New York: Holt, Rinehart and Winston.
- Blocher, A. & Schirra, J. R. J. (1995). Optional deep case filling and focus control with mental images: ANTLIMA-KOREF. In C. S. Mellish (Hrsg.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (S. 417–423). San Mateo, CA: Morgan Kaufmann.
- Blocher, A. (1994). *KOREF: Zum Vergleich intendierter und imaginer Äußerungsgehalte*. Diplomarbeit, Fachbereich Informatik, Universität des Saarlandes, Saarbrücken.
- Bond, A. H. & Gasser, L. (Hrsg.) (1988). *Readings in distributed artificial intelligence*. San Mateo, CA: Morgan Kaufmann.
- Brams, S. J. (1994). *Theory of moves*. Cambridge, UK: Cambridge University Press.

- Bühlmann, H., Loeffel, H. & Nievergelt, E. (1975). *Entscheidungs- und Spieltheorie: Ein Lehrbuch für Wirtschaftswissenschaftler*. Berlin: Springer-Verlag.
- Carbonell, J. G., Knoblock, C. A. & Milton, S. (1991). PRODIGY: An integrated architecture for planning and learning. In K. VanLehn (Hrsg.), *Architectures for intelligence* (S. 241–278). Hillsdale, NJ: Erlbaum.
- Cawsey, A., Galliers, J. R., Reece, S. & Sparck Jones, K. (1992). A comparison of architectures for autonomous multi-agent communication. *Proceedings of the Tenth European Conference on Artificial Intelligence*, Wien, 249–251.
- CLIM (1992). *HP/SPARC Common Lisp Interface Manager 1.0: User's Guide*.
- Cutkosky, M., Englemore, R., Richard Fikes, M. G., Gruber, T., Mark, W., Tenenbaum, J. & Weber, J. (1993). Pact: An experiment in integrating concurrent engineering systems. *IEEE Computer*, 26 (1), 28–37.
- Davis, M. H. (1994). *Empathy: A social psychological approach*. Madison, WI: Brown & Benchmark.
- Davis, R. & Smith, R. G. (1983). Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence*, 20 (1), 63–109.
- Dean, T. L. & Boddy, M. (1988). An analysis of time-dependent planning. *Proceedings of the Seventh National Conference on Artificial Intelligence*, St. Paul, MN, 49–54.
- de Latil, P. (1953). *La pensée artificielle: Introduction à la cybernétique*. Paris: Gallimard.
- DiStefano, J. J., Stubberud, A. R. & Williams, I. J. (1976). *Regelsysteme: Theorie und Anwendung mit Beispielen aus Technik, Physik und Biologie*. Düsseldorf: McGraw-Hill.
- Englemore, R. & Morgan, T. (1988). *Blackboard systems*. Wokingham, England: Addison-Wesley.
- Erman, L. D., Hayes-Roth, F., Lesser, V. R. & Reddy, D. R. (1981). The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. In B. L. Webber & N. J. Nilsson (Hrsg.), *Readings in artificial intelligence* (S. 349–389). Los Altos, CA: Kaufmann.
- Finin, T., Labrou, Y. & Mayfield, J. (1995). KQML as an agent communication language. In J. Bradshaw (Hrsg.), *Software agents*. Cambridge, MA: MIT Press.
- Finkler, W. (1997). *Automatische Selbstkorrektur bei der inkrementellen Generierung gesprochener Sprache unter Realzeitbedingungen; ein empirisch-simulativer Ansatz unter Verwendung eines Begründungsverwaltungssystems*. DISKI – Dissertationen zur KI. Band 165. Sankt Augustin: Infix Verlag.
- Fischer, K. (1993). *Verteiltes und kooperatives Planen in einer flexiblen Fertigungsumgebung*. DISKI – Dissertationen zur KI. Band 26. Sankt Augustin: Infix Verlag.
- Flavell, J. H., Botkin, P. T., Fry Jr., C. L., Wright, J. W. & Jarvis, P. E. (1968). *The development of role-taking and communication skills in children*. New York: Wiley.

- Flavell, J. H., Miller, P. H. & Miller, S. A. (1993). *Cognitive development* (3. Aufl.). Englewood Cliffs: Prentice-Hall.
- Flavell, J. H. (1974). The development of inferences about others. In T. Mischel (Hrsg.), *Understanding other persons* (S. 66–116). Totowa, NJ: Rowman and Littlefield.
- Flavell, J. H. (1992). Perspectives on perspective taking. In H. Beilin & P. B. Pufall (Hrsg.), *Piaget's theory: Prospects and possibilities*. Hillsdale, NJ: Lawrence Erlbaum.
- Friedlmeier, W. (1993). *Entwicklung von Empathie, Selbstkonzept und prosozialem Handeln in der Kindheit*. Konstanz: Hartung-Gorre.
- Fudenberg, D. & Tirole, J. (1991). *Game theory*. Cambridge, MA: MIT Press.
- Fum, D., Guida, G. & Tasso, C. (1988). A distributed multi-agent architecture for natural language processing. *Proceedings of the Twelfth International Conference on Computational Linguistics*, Budapest, Hungary, 812–814.
- Gasser, L. & Briot, J.-P. (1992). Object-based concurrent programming and distributed artificial intelligence. In N. M. Avouris & L. Gasser (Hrsg.), *Distributed artificial intelligence: Theory and praxis* (S. 81–107). Dordrecht: Kluwer.
- Gasser, L. (1992). An overview of DAI. In N. M. Avouris & L. Gasser (Hrsg.), *Distributed Artificial Intelligence: Theory and praxis* (S. 9–30). Dordrecht: Kluwer.
- Gasser, L. (1993). *Architectures and environments for AI – Individual-agent, Distributed, and social architectures for intelligent system*. Fifth Advanced Course on Artificial Intelligence. Capri, Italy.
- Geist, A., Beguelin, A., Dongorra, J., Jiang, W., Manchek, R. & Sunderman, V. (1994). *PVM: Parallel Virtual Machine. A user's guide and tutorial for networked parallel computing*. Cambridge, MA: MIT Press.
- Genesereth, M. & Fikes, R. (1992). *Knowledge Interchange Format, version 3.0 reference manual*. Technical Report Logic-92-1, Computer Science Department, Stanford University.
- Genesereth, M., Ginsberg, M. & Rosenschein, J. (1986). Cooperation without communication. *Proceedings of the Fifth National Conference on Artificial Intelligence*, St. Paul, MN, 51–57.
- Geulen, D. (Hrsg.) (1982a). *Perspektivenübernahme und soziales Handeln: Texte zur sozial-kognitiven Entwicklung*. Frankfurt am Main: Suhrkamp.
- Geulen, D. (1982b). Soziales Handeln und Perspektivenübernahme. In D. Geulen (Hrsg.), *Perspektivenübernahme und soziales Handeln: Texte zur sozial-kognitiven Entwicklung*. Frankfurt am Main: Suhrkamp.
- Ginsberg, M. (1995). Approximate planning. *Artificial Intelligence*, 76, 89–124.
- Gmytrasiewicz, P. J. & Durfee, E. H. (1997). *Rational interaction in multiagent environments: Coordination*. Zur Veröffentlichung eingereichtes Manuskript.
- Grice, H. P. (1975). Logic and conversation. In P. Cole & J. L. Morgan (Hrsg.), *Syntax and semantics, vol. 3: Speech acts* (S. 41–58). New York: Academic Press.

- Guillaume, P. (1954). *Introduction à la psychologie*. Paris: J. Vrin.
- Hahn, U. (1993). An actor model for distributed natural language parsing. In G. Adriaens & U. Hahn (Hrsg.), *Parallel natural language processing*. Norwood NJ: Ablex.
- Harris, P. L. (1991). The work of the imagination. In A. Whiten (Hrsg.), *Natural theories of mind: Evolution, development and simulation of everyday mindreading* (S. 283–304). Oxford, UK: Basil Blackwell.
- Herrmann, T. & Grabowski, J. (1994). *Sprechen: Psychologie der Sprachproduktion*. Heidelberg: Spektrum Akademischer Verlag.
- Herrmann, T. (1982). *Sprechen und Situation: Eine psychologische Konzeption zur situationsspezifischen Sprachproduktion*. Berlin: Springer-Verlag.
- Higgins, E. T. (1981). Role taking and social judgment: Alternative developmental perspectives and processes. In J. H. Flavell & L. Ross (Hrsg.), *Social cognitive development: Frontiers and possible futures* (S. 119–153). Cambridge, England: Cambridge University Press.
- Horton, W. S. & Keysar, B. (1996). When do speakers take into account common ground? *Cognition*, 59, 91–117.
- Hovy, E. H. (1988). *Generating natural language under pragmatic constraints*. Hillsdale, NJ: Erlbaum.
- Hustadt, U. & Nonnengart, A. (1993). Modalities in knowledge representation. In *Proceedings of the Sixth Australian Joint Conference on Artificial Intelligence* (S. 249–254). Singapore: World Scientific Publishing.
- Ishikawa, Y. & Tokoro, M. (1987). Orient84/K: an object-oriented concurrent programming language for knowledge representation. In A. Yonezawa & M. Tokoro (Hrsg.), *Object-oriented concurrent programming* (S. 159–198). Cambridge, MA: MIT Press.
- Jameson, A., Kipper, B., Ndiaye, A., Schäfer, R., Simons, J., Weis, T. & Zimmermann, D. (1994). Cooperating to be noncooperative: The dialog system PRACMA. In B. Nebel & L. Dreschler-Fischer (Hrsg.), *KI-94: Advances in artificial intelligence* (S. 106–117). Berlin: Springer.
- Jameson, A., Schäfer, R., Simons, J. & Weis, T. (1995). Adaptive provision of evaluation-oriented information: Tasks and techniques. In C. S. Mellish (Hrsg.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (S. 1886–1893). San Mateo, CA: Morgan Kaufmann.
- Jameson, A. & Wahlster, W. (1982). User modelling in anaphora generation: Ellipsis and definite description. *Proceedings of the Fifth European Conference on Artificial Intelligence*, Orsay, Frankreich, 222–227.
- Jameson, A. & Weis, T. (1996). How to juggle discourse obligations. In R. Meyer-Klabunde & C. von Stutterheim (Hrsg.), *Proceedings des Symposiums 'Konzeptuelles und semantisches Wissen in der Sprachproduktion'* (S. 171–185). Heidelberg/Mannheim: Arbeiten aus dem Sonderforschungsbereich 245 'Sprache und Situation'. Bericht Nr. 92.

- Jameson, A. (1983). Impression monitoring in evaluation-oriented dialog: The role of the listener's assumed expectations and values in the generation of informative statements. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, 616–620.
- Jameson, A. (1989). But what will the listener think? Belief ascription and image maintenance in dialog. In A. Kobsa & W. Wahlster (Hrsg.), *User models in dialog systems* (S. 255–312). Berlin: Springer.
- Jameson, A. (1995). Logic is not enough: Why reasoning about another person's beliefs is reasoning under uncertainty. In A. Laux & H. Wansing (Hrsg.), *Knowledge and belief in philosophy and artificial intelligence* (S. 199–229). Berlin: Akademie Verlag.
- Joshi, A., Webber, B. L. & Weischedel, R. M. (1984a). Living up to expectations: Computing expert responses. *Proceedings of the Fourth National Conference on Artificial Intelligence*, Austin, TX, 169–175.
- Joshi, A., Webber, B. L. & Weischedel, R. M. (1984b). Preventing false inferences. *Proceedings of the Tenth International Conference on Computational Linguistics*, Stanford, CA, 134–138.
- Jung, J., Kresse, A., Reithinger, N. & Schäfer, R. (1989). Das System ZORA: Wissensbasierte Generierung von Zeigegeesten. In D. Metzger (Hrsg.), *GWAI-89: Proceedings of the Thirteenth German Workshop on Artificial Intelligence* (S. 190–194). Berlin: Springer.
- Kilger, A. & Finkler, W. (1995). *Incremental generation for real-time applications*. Research Report 95-11, DFKI, Saarbrücken.
- Kipper, B. (1994). A blackboard architecture for natural language analysis. *Proceedings of the Seventh Florida AI Research Symposium*, Pensacola Beach, FL, 231–235.
- Kipper, B. (1995). *Repräsentation und Verarbeitung propositionaler Einstellungen in natürlichsprachlichen Systemen*. Dissertation, Universität des Saarlandes.
- Kurdek, L. (1977). Structural components and intellectual correlates of cognitive perspective-taking in first- through four-grade children. *Child Development*, 48, 1503–1511.
- Laubsch, J. & Nerbonne, J. (1991). *An overview of \mathcal{NLL}* . Technical report, Hewlett Packard Laboratories, Palo Alto, CA.
- Laux, H. (1991). *Entscheidungstheorie I: Grundlagen*. Berlin: Springer-Verlag.
- Leichtman, S. R. (1971). *Correlates of role-taking and communication skill in children*. Dissertation, University of North Carolina.
- Levelt, W. J. M. (1989). *Speaking: From intention to articulation*. Cambridge, MA: MIT Press.
- Littlejohn, S. W. (1991). *Theories of human communication* (4 Aufl.). Belmont: CA: Wadsworth.

- LUCID (1992). *HP/SPARC Lucid Common Lisp 4.1: User's Guide and Advanced User's Guide*.
- Maaß, W. (1996). *Von visuellen Daten zu inkrementellen Wegbeschreibungen in dreidimensionalen Umgebungen: Das Modell eines kognitiven Agenten*. Dissertation, Universität des Saarlandes.
- Mead, G. H. (1968). *Geist, Identität und Gesellschaft aus der Sicht des Sozialbehaviorismus*. Frankfurt am Main: Suhrkamp.
- Miller, P. H., Kessel, F. S. & Flavell, J. H. (1970). Thinking about people thinking about people think about . . . : A study of social cognitive development. *Child Development* (41), 613–623.
- Minsky, N. H. (1991). The imposition of protocols over open distributed systems. *IEEE Transactions on Software Engineering*, 17(2), 183–195.
- Mitchell, T., Allen, J., Chalasani, P., Cheng, J., Etzioni, O., Ringuette, M. & Schlimmer, J. C. (1991). THEO: A framework for self-improving systems. In K. VanLehn (Hrsg.), *Architectures for intelligence* (S. 323–355). Hillsdale, NJ: Erlbaum.
- Mucchielli, A. (1991). *Les situations de communication: Approche formelle*. Paris: Eyrolles.
- Myerson, R. B. (1991). *Game theory: Analysis of conflict*. Cambridge, MA: Harvard University Press.
- Ndiaye, A. & Jameson, A. (1994). Supporting flexibility and transmutability: Multi-agent processing and role-switching in a pragmatically oriented dialog system. In P. Jorrand & V. Sgurev (Hrsg.), *Proceedings of the Sixth International Conference on Artificial Intelligence: Methodology, Systems, Applications* (S. 381–390). Singapore: World Scientific Publishing.
- Ndiaye, A. & Jameson, A. (1995). Global anticipation feedback in a transmutable dialog system. In U. Malinowski (Hrsg.), *ABIS 95, Adaptivität und Benutzermodellierung in interaktiven Systemen: Workshop-Beiträge*. München: Siemens Corporate R & D.
- Ndiaye, A. & Jameson, A. (1996a). Global anticipation feedback as a user modeling technique for dialog systems. *Actes du 3^e Colloque Africain sur la Recherche en Informatique (CARI'96)*, Libreville, Gabun, 730–744.
- Ndiaye, A. & Jameson, A. (1996b). Predictive role taking in dialog: Global anticipation feedback based on transmutability. In S. Carberry & I. Zukerman (Hrsg.), *Proceedings of the Fifth International Conference on User Modeling* (S. 137–144). Boston, MA: User Modeling, Inc.
- Ndiaye, A. (1996). Rollenübernahme in einem Dialogsystem. *Künstliche Intelligenz*, 4/96, 34–40.
- Neapolitan, R. E. (1990). *Probabilistic reasoning in expert systems: Theory and algorithms*. New York: Wiley.

- Neumann, G. & van Noord, G. (1994). Reversibility and self-monitoring in natural language generation. In T. Strzalkowski (Hrsg.), *Reversible grammar in natural language processing* (S. 59–96). Dordrecht: Kluwer.
- Neumann, G. (1994). *A uniform computational model for natural language parsing and generation*. Dissertation, Universität des Saarlandes.
- Neumann, G. (1998). Interleaving natural language parsing and generation through uniform processing. *Artificial Intelligence*. To appear.
- Noh, S. & Gmytrasiewicz, P. J. (1997). Agent modeling in anti-air defense. In A. Jameson, C. Paris & C. Tasso (Hrsg.), *User modeling: Proceedings of the Sixth International Conference, UM97* (S. 389–400). Wien, New York: Springer Wien New York.
- Nooteboom, S. G. (1980). Speaking and unspeaking: Detection and correction of phonological and lexical errors in spontaneous speech. In V. A. Fromkin (Hrsg.), *Errors in linguistic performance: Slips of the tongue, ear, pen, and hand* (S. 87–95). London: Academic Press.
- Novak, H.-J. (1987a). *Textgenerierung aus visuellen Daten: Beschreibungen von Straßenszenen*. Heidelberg: Springer-Verlag. Informatik-Fachberichte 142.
- Novak, H.-J. (1987b). Strategies for generating coherent descriptions of object movements in street scenes. In G. Kempen (Hrsg.), *Natural language generation: New results in artificial intelligence, psychology, and linguistics* (S. 117–132). Dordrecht, Niederlande: Nijhoff.
- Oléron, P., Beaudichon, J., Cartron, A., Danset-Léger, J., Melot, A.-M., Nguyen-Xuan, A. & Winnykamen, F. (1981). *Savoirs et savoir-faire psychologiques chez l'enfant*. Bruxelles: Pierre Mardaga.
- Ophoff, H.-W. (1986). *Partner-adaptive Aspekte referentieller Kommunikation: Ein Theorie-Entwurf und empirische Prüfung von Basis-Annahmen*. Frankfurt am Main: Peter Lang.
- Paul, J. (1996). *MOSES-SPEAK - Adaptierte raumsprachliche Beschreibungen in 3D-Umgebungen am Beispiel inkrementeller Wegbeschreibungen*. Diplomarbeit, Fachbereich Informatik, Universität des Saarlandes, Saarbrücken.
- Pearl, J. (1991). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan Kaufmann.
- Piaget, J. & Inhelder, B. (1971). *Die Entwicklung des räumlichen Denkens beim Kinde*. Stuttgart: Klett.
- Piaget, J. (1974). *Biologie und Erkenntnis: Über die Beziehungen zwischen organischen Regulationen und kognitiven Prozessen*. Frankfurt am Main: S. Fischer Verlag.
- Piaget, J. (1978). *Das Weltbild des Kindes*. Stuttgart: Klett-Cotta.
- Premack, D. & Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1, 515–526.

- Pyka, C. (1991). *Architektur von ASL-Nord*. Technical Report ASL-TR-4-91/UHH, University of Hamburg.
- Rapoport, A. (1989). *Decision theory and decision behaviour: Normative and descriptive approaches*. Dordrecht, The Netherlands: Kluwer.
- Reinefeld, A. (1989). *Spielbaum-Suchverfahren*. Heidelberg: Springer-Verlag. Informatik-Fachberichte 200.
- Reithinger, N. (1992). *Eine parallele Architektur zur inkrementellen Generierung multimodaler Dialogbeiträge*. DISKI – Dissertationen zur KI. Band 1. Sankt Augustin: Infix Verlag.
- Rich, E. & Knight, K. (1991). *Artificial Intelligence*. New York: McGraw-Hill.
- Rist, T. (1996). *Wissensbasierte Verfahren für den automatischen Entwurf von Gebrauchsgaphik in der technischen Dokumentation*. DISKI – Dissertationen zur KI. Band 139. Sankt Augustin: Infix Verlag.
- Roloff, M. E. & Berger, C. R. (Hrsg.) (1982). *Social cognition and communication*. Beverly Hills, CA: Sage Publications.
- Rosenbloom, P. S., Newell, A. & Laird, J. E. (1991). Towards the knowledge level in Soar: The role of the architecture in the use of knowledge. In K. VanLehn (Hrsg.), *Architectures for intelligence* (S. 75–111). Hillsdale, NJ: Erlbaum.
- Roßnagel, C. (1993). Hörerorientierung beim Instruieren als “Opfer” deklarativer Vereinfachung im Arbeitsgedächtnis? *Abstractband der 35. Tagung experimentell arbeitender Psychologen*, 232.
- Roßnagel, C. (1994). *Mündliche Instruktionen als nicht-gesprochene Sprechprodukte: Zur Hörerorientierung geübter Instruktoren*. Dissertation, Ruprecht-Karls-Universität Heidelberg.
- Roßnagel, C. (1995). Kognitive Belastung und Hörerorientierung beim monologischen Instruieren. *Zeitschrift für Experimentelle Psychologie*, 42, 94–110.
- Russell, S. J. & Norvig, P. (1995). *Artificial Intelligence: A modern approach*. Englewood Cliffs, NJ: Prentice-Hall.
- Russell, S. J. & Wefald, E. H. (1991a). *Do the right thing: Studies in limited rationality*. Cambridge, MA: MIT Press.
- Russell, S. J. & Wefald, E. H. (1991b). Principles of metareasoning. *Artificial Intelligence*, 49, 361–395.
- Sabah, G. & Briffault, X. (1993). CAMEL: A step towards reflection in natural language understanding systems. *Proceedings of the Fifth IEEE International Conference on Tools with AI*, Boston.
- Sabah, G. (1990). CAMEL: A computational model of natural language understanding using a parallel implementation. *Proceedings of the Ninth European Conference on Artificial Intelligence*, Stockholm, 563–565.

- Schäfer, R. (1997). *Inferences about evaluation processes in dialog with bayesian networks and multi-attribute utility theory*. Zur Veröffentlichung eingereichtes Manuskript.
- Schirra, J. R. J. (1995). *Bildbeschreibung als Verbindung von visuellem und sprachlichem Raum – Eine interdisziplinäre Untersuchung von Bildvorstellungen in einem Hörermodell*. DISKI – Dissertationen zur KI. Band 71. Sankt Augustin: Infix Verlag.
- Schwalbe, M. L. (1988). Role taking reconsidered: Linking competence and performance to social structure. *Journal for the Theory of Social Behaviour*, 18(4), 411–436.
- Searle, J. R. (1983). *Sprechakte – Ein sprachphilosophischer Essay*. Frankfurt am Main: Suhrkamp.
- Selman, R. L. (1980). *The growth of interpersonal understanding: Developmental and clinical analyses*. New York: Academic Press.
- Shannon, C. E. & Weaver, W. (1976). *Mathematische Grundlagen der Informationstheorie*. München: Oldenbourg Verlag.
- Shantz, C. U. (1975). The development of social cognition. In E. M. Hetherington (Hrsg.), *Review of child development research, volume five* (S. 257–323). Chicago: The University of Chicago Press.
- Shantz, C. U. (1981). The role of role-taking in children's referential communication. In W. P. Dickson (Hrsg.), *Children's oral communication skills* (S. 85–102). New York: Academic Press.
- Silbereisen, R. K. (1975). *Antezedente Bedingungen der Rollenübernahme bei Kindern*. Dissertation, Technische Universität Berlin.
- Silbereisen, R. K. (1987). Soziale Kognition. In R. Oerter & L. Montada (Hrsg.), *Entwicklungspsychologie* (Kapitel 14, S. 696–737). Weinheim: Psychologie Verlags Union.
- Silverstein, S. (1969). *Attitudinal congruency and anticipation of feedback as variable affecting the comparability of deception and role-playing experimental procedures*. Dissertation, University of Connecticut.
- Simon, H. A. (1981). *The sciences of the artificial* (2 Aufl.). Cambridge, MA: The MIT Press.
- Small, S. & Rieger, C. (1982). Parsing and comprehending with word experts (A theory and its realization). In W. G. Lehnert & M. H. Ringle (Hrsg.), *Strategies for natural language processing* (S. 89–147). Hillsdale, NJ: Erlbaum.
- Steele, Jr., G. L. (1990). *Common LISP – The Language, Second Edition*. Digital Press.
- Stefanini, M.-H., Berrendonner, A., Lallich, G. & Oquendo, F. (1992). TALISMAN: Un système multi-agents gouverné par des lois linguistiques pour le traitement de la langue naturelle. *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes, France.
- Stefanini, M.-H. & Demazeau, Y. (1995). TALISMAN: A multi-agent system for natural language processing. In J. Wainer & A. Carvalho (Hrsg.), *Advances in artificial*

- intelligence: Proceedings of the twelfth brazilian symposium on artificial intelligence* (S. 312–322). Berlin: Springer-Verlag. LNAI 991.
- Steins, G. & Wicklund, R. A. (1993). Zum Konzept der Perspektivenübernahme: Ein kritischer Überblick. *Psychologische Rundschau*, 44, 226–239.
- Stoyan, H. (1988). Nachrichtenverarbeitung – Objekt-orientierte Programmierung (1). In *Programmiermethoden der künstlichen Intelligenz, band 2* (S. 181–244). Berlin: Springer.
- Tanenbaum, A. S. (1995). *Verteilte Betriebssysteme*. München: Prentice Hall.
- Underwood, B. & Moore, B. (1982). Perspective-taking and altruism. *Developmental Psychology*, 12, 98–203.
- Uszkoreit, H., Backofen, R., Busemann, S., Diagne, A. K., Hinkelman, E. A., Kasper, W. et al. (1994). *DISCO—an HPSG-based NLP system and its application for appointment scheduling*. Research Report 94-38, DFKI, Saarbrücken.
- Vanderveken, D. (1990). *Meaning and speech acts*. Cambridge: Cambridge University Press.
- van Mulken, S. (1996). Reasoning about the user's decoding of presentations in an intelligent multimedia presentation system. In S. Carberry & I. Zukerman (Hrsg.), *Proceedings of the Fifth International Conference on User Modeling* (S. 67–74). Boston, MA: User Modeling, Inc.
- van Mulken, S. (1997). Inferenzen über die Verständlichkeit einer Präsentation: Benutzermodellierung für Multimedia-Dialogsysteme. In R. H. Kluwe (Hrsg.), *Kognitionswissenschaft: Strukturen und prozesse intelligenter systeme* (S. 133–155). Wiesbaden: Deutscher Universitätsverlag.
- Velthuisen, H. (1992). *The nature and applicability of the blackboard architecture*. Maastricht: Leidschendam: PTT Research-I11. Doctoral Dissertation.
- von Hahn, W. (1991). System architectures as the key issue for speech understanding. In E. Klein & F. Veltman (Hrsg.), *Natural language and speech: Symposium proceedings* (S. 184–188). Berlin, Heidelberg: Springer.
- von Hahn, W. (1992). Von der Verknüpfung zur Integration: Kontrollstrategie oder kognitive Architektur? *Proceedings of KONVENS-92*, 1–10.
- von Winterfeldt, D. & Edwards, W. (1986). *Decision analysis and behavioral research*. Cambridge, England: Cambridge University Press.
- Wahlster, W., André, E., Finkler, W., Profitlich, H.-J. & Rist, T. (1993). Plan-based integration of natural language and graphics generation. *Artificial Intelligence*, 63, 387–427.
- Wahlster, W., Jameson, A., Ndiaye, A., Schäfer, R. & Weis, T. (1995a). READY: Ressourcenadaptive Dialogführung. In W. H. Tack (Hrsg.), *Ressourcenadaptive kognitive Prozesse: Finanzierungsantrag 1996 – 1997 – 1998*. Saarbrücken: Universität des Saarlandes.

- Wahlster, W., Jameson, A., Ndiaye, A., Schäfer, R. & Weis, T. (1995b). Ressourcenadaptive Dialogführung: Ein interdisziplinärer Forschungsansatz. *Künstliche Intelligenz*, 9(6), 17–21.
- Wahlster, W., Jameson, A., Ndiaye, A., Schäfer, R. & Weis, T. (1997). PRACMA: Processing Arguments between Controversially-Minded Agents. In P. Deussen (Hrsg.), *Künstliche Intelligenz – Wissensbasierte Systeme: Arbeits- und Ergebnisbericht für die Jahre 1994 – 1995*.
- Wahlster, W. & Kobsa, A. (1989). User models in dialog systems. In A. Kobsa & W. Wahlster (Hrsg.), *User models in dialog systems* (S. 4–34). Berlin: Springer.
- Wahlster, W. (1989). Anticipation feedback in dialog systems. *Proceedings of the ATR Symposium on Basic Research for Telephone Interpretation*, Kyoto, Japan, S. 7.2.1–7.2.23.
- Wahlster, W. (1991). User and discourse models for multimodal communication. In J. W. Sullivan & S. W. Tyler (Hrsg.), *Intelligent user interfaces* (S. 45–67). New York / Reading, MA: ACM Press / Addison-Wesley.
- Wahlster, W. (1992). *Architekturen zur Koordination kognitiver Prozesse in KI-Systemen*. OHP-Folien zum Vortrag im Rahmen der Ringvorlesung des Graduiertenkollegs Kognitionswissenschaften.
- Webber, B. L. & Mays, E. (1983). Varieties of user misconceptions: Detection and correction. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, 650–652.
- Weis, T. (1994). *VIPER: Ein verteilter inkrementeller Dialogplaner für eine Multi-Agenten-Umgebung*. Diplomarbeit, Fachbereich Informatik, Universität des Saarlandes.
- Weis, T. (1996). Die Rolle von Diskursverpflichtungen in bewertungsorientierten Informationsdialogen. *Natural Language Processing and Speech Technology: Results of the 3rd KONVENS Conference, Bielefeld, October 1996*, Berlin, 280–292.
- Wellman, H. M. (1990). *The child's theory of mind*. Cambridge, MA: MIT Press.
- Werner, A. (1997). *Erkennung von Bewertungen anhand natürlichsprachlicher Ausdrücke*. Diplomarbeit, Fachbereich Informatik, Universität des Saarlandes. In Vorbereitung.
- Whiten, A. & Perner, J. (1991). Fundamental issues in the multidisciplinary study of mindreading. In A. Whiten (Hrsg.), *Natural theories of mind: Evolution, development and simulation of everyday mindreading* (S. 1–17). Oxford, UK: Basil Blackwell.
- Whiten, A. (Hrsg.) (1991). *Natural theories of mind: Evolution, development and simulation of everyday mindreading*. Oxford, UK: Basil Blackwell.
- Wille, M. (1989). *TACTILUS-II: Evaluation und Ausbau einer Komponente zur Simulation und Analyse von Zeigegesten*. Diplomarbeit, Universität Saarbrücken, Saarbrücken.
- Winograd, T. (1972). *Understanding natural language*. Edinburgh: Edinburgh University Press.

- Wittig, T. (Hrsg.) (1992). *ARCHON: An architecture for multi-agent systems*. London: Ellis Horwood.
- Wooldridge, M. & Jennings, N. R. (1994). Agent theories, architectures, and languages: A survey. In M. Wooldridge & N. R. Jennings (Hrsg.), *Intelligent agents : Proceedings of workshop on agent theories, architectures, and languages (ECAI-94)* (S. 1–32). Amsterdam: Springer-Verlag. LNAI 890.
- Yonezawa, A., Shibayama, E., Honda, Y., Takada, T. & Briot, J.-P. (1990). An object-oriented concurrent computation model ABCM/1 and its description language ABCL/1. In A. Yonezawa (Hrsg.), *ABCL: An object-oriented concurrent system* (S. 13–43). Cambridge, MA: MIT Press.
- Yonezawa, A. & Tokoro, M. (Hrsg.) (1987). *Object-oriented concurrent programming*. Cambridge, MA: The MIT Press.
- Yonezawa, A. (Hrsg.) (1990). *ABCL: An object-oriented concurrent system*. Cambridge, MA: MIT Press.
- Zukerman, I. & McConachy, R. (1993a). Consulting a user model to address a user's inferences during content planning. *User Modeling and User-Adapted Interaction*, 3, 155–185.
- Zukerman, I. & McConachy, R. (1993b). Generating concise discourse that addresses a user's inferences. In R. Bajcsy (Hrsg.), *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (S. 1202–1207). San Mateo, CA: Morgan Kaufmann.
- Zukerman, I. (1990a). Anticipating a listener's response in text planning. In M. C. Golumbic (Hrsg.), *Advances in artificial intelligence, natural language and knowledge-based systems* (S. 145–170). New-York: Springer-Verlag.
- Zukerman, I. (1990b). A predictive approach for the generation of rhetorical devices. *Computational Intelligence*, 6, 25–40.

Autorenverzeichnis

- Agha, G. 45, 52
Allen, J. 41
Allgayer, J. 28, 43
Amtrup, J. W. 54, 127
André, E. 30, 31
Astington, J. W. 12, 17, 18, 19
- Backofen, R. 69
Beaudichon, J. 14
Beguelin, A. 55, 127
Berlo, D. K. 10, 17
Berrendonner, A. 57, 59
Bühlmann, H. 77, 113, 114
Blocher, A. 22, 27, 28, 36, 39, 99
Boddy, M. 105
Botkin, P. T. 9, 12, 14, 15, 16, 17
Brams, S. J. 19
Briffault, X. 57, 59
Briot, J.-P. 55, 56, 58
Busemann, S. 69
- Carbonell, J. G. 41
Cartron, A. 14
Cawsey, A. 56
Chalasanani, P. 41
Cheng, J. 41
Cutkosky, M. 46
- Danset-Léger, J. 14
Davis, M. H. 13, 14
Davis, R. 53
de Latil, P. 9
Dean, T. L. 105
Demazeau, Y. 57, 59
Diagne, A. K. 69
DiStefano, J. J. 9
Dongorra, J. 55, 127
Durfee, E. H. 34, 37, 40
- Edwards, W. 62
Engelmore, R. 43, 46
Erman, L. D. 43
Etzioni, O. 41
- Fikes, R. 46
Finin, T. 46
Finkler, W. 30, 70, 105
Fischer, K. 46
Flavell, J. H. 9, 12, 14, 15, 16, 17
Friedlmeier, W. 14
Fry Jr., C. L. 9, 12, 14, 15, 16, 17
Fudenberg, D. 19
Fum, D. 57, 59
- Galliers, J. R. 56
Gasser, L. 41, 42, 55, 56
Geist, A. 55, 127
Genesereth, M. 46, 53
Geulen, D. 3, 17
Ginsberg, M. 53, 111
Gmytrasiewicz, P. J. 34, 37, 40
Grabowski, J. 13, 18
Graf, W. 31
Grice, H. P. 60
Gruber, T. 46
Guida, G. 57, 59
Guillaume, P. 10
- Hahn, U. 45
Harris, P. L. 19
Hayes-Roth, F. 43
Heinsohn, J. 31
Herrmann, T. 13, 15, 18
Higgins, E. T. 14, 79, 113, 128
Hinkelman, E. A. 69
Honda, Y. 58
Horton, W. S. 103, 104, 105
Hovy, E. H. 33, 60
- Hustadt, U. 68, 70
- Inhelder, B. 12, 13
Ishikawa, Y. 52
- Jameson, A. 1, 3, 21, 22, 24, 36, 37, 38, 40, 47, 60, 64, 67, 68, 71, 74, 76, 79, 86, 93, 96, 98, 105, 106, 126
Jansen-Winkel, R. 28, 43
Jarvis, P. E. 9, 12, 14, 15, 16, 17
Jennings, N. R. 46
Jiang, W. 55, 127
Joshi, A. 1, 22, 32, 37, 40
Jung, J. 22, 28, 36, 39
- Kasper, W. 69
Kessel, F. S. 16
Keysar, B. 103, 104, 105
Kilger, A. 70
Kipper, B. 3, 60, 69, 126
Knight, K. 56
Knoblock, C. A. 41
Kobsa, A. 1, 6, 21, 127
Kresse, A. 22, 28, 36, 39
Kurdek, L. 14
- Labrou, Y. 46
Laird, J. E. 41
Lallich, G. 57, 59
Laubsch, J. 69
Laux, H. 20, 96
Leichtman, S. R. 15
Lesser, V. R. 43
Levelt, W. J. M. 11
Littlejohn, S. W. 10
Loeffel, H. 77, 113, 114
- Maaß, W. 59
Manchek, R. 55, 127

- Mark, W. 46
 Mayfield, J. 46
 Mays, E. 32
 McConachy, R. 22, 33, 37, 40
 Mead, G. H. 12, 15
 Melot, A.-M. 14
 Miller, P. H. 16
 Miller, S. A. 16
 Milton, S. 41
 Minsky, N. H. 59
 Mitchell, T. 41
 Moore, B. 14
 Morgan, T. 43
 Mucchielli, A. 10
 Myerson, R. B. 19

 Ndiaye, A. 3, 37, 40, 47, 60, 64, 71, 74, 76, 79, 86, 96, 98, 105, 106, 126
 Neapolitan, R. E. 63
 Nebel, B. 31
 Nerbonne, J. 69
 Neumann, G. 25, 36, 39
 Newell, A. 41
 Nguyen-Xuan, A. 14
 Nievergelt, E. 77, 113, 114
 Noh, S. 34, 37, 40
 Nonnengart, A. 68, 70
 Nooteboom, S. G. 12
 Norvig, P. 20, 89, 111
 Novak, H.-J. 22, 26, 27, 36, 38

 Oléron, P. 14
 Ophoff, H.-W. 15
 Oquendo, F. 57, 59

 Paul, J. 59
 Pearl, J. 63
 Perner, J. 15, 21
 Piaget, J. 9, 12, 13
 Premack, D. 18
 Profitlich, H.-J. 30, 31
 Pyka, C. 57, 58

 Rapoport, A. 20

 Reddig, C. 28, 43
 Reddy, D. R. 43
 Reece, S. 56
 Reinefeld, A. 20
 Reithinger, N. 22, 28, 36, 39, 43, 71
 Rich, E. 56
 Richard Fikes, M. G. 46
 Rieger, C. 43
 Ringuette, M. 41
 Rist, T. 30, 31, 37, 39
 Rosenbloom, P. S. 41
 Rosenschein, J. 53
 Roßnagel, C. 18, 19
 Russell, S. J. 20, 89, 105, 106, 107, 109, 111
 Sabah, G. 57, 58, 59
 Schäfer, R. 3, 22, 28, 36, 39, 60, 63, 70, 92, 105, 106, 117, 126
 Schirra, J. R. J. 22, 27, 36, 39, 99
 Schlimmer, J. C. 41
 Schwalbe, M. L. 17
 Searle, J. R. 50
 Selman, R. L. 14, 16
 Shannon, C. E. 10
 Shantz, C. U. 15, 17
 Shibayama, E. 58
 Silbereisen, R. K. 12, 15
 Silverstein, S. 10
 Simon, H. A. 96
 Simons, J. 3, 60, 126
 Small, S. 43
 Smith, R. G. 53
 Sparck Jones, K. 56
 Steele, Jr., G. L. 53
 Stefanini, M.-H. 57, 59
 Steins, G. 13, 17
 Stoyan, H. 45, 52
 Stubberud, A. R. 9
 Sunderman, V. 55, 127
 Takada, T. 58

 Tanenbaum, A. S. 51
 Tasso, C. 57, 59
 Tenenbaum, J. 46
 Tirole, J. 19
 Tokoro, M. 52

 Underwood, B. 14
 Uszkoreit, H. 69

 van Mulken, S. 31, 37, 39
 van Noord, G. 25, 36, 39
 Vanderveken, D. vii, 50
 Velthuijsen, H. 43
 von Hahn, W. 42, 57, 58
 von Winterfeldt, D. 62

 Wahlster, W. 1, 6, 21, 22, 28, 29, 30, 31, 36, 38, 42, 60, 105, 106, 123, 127
 Weaver, W. 10
 Webber, B. L. 1, 22, 32, 37, 40
 Weber, J. 46
 Wefald, E. H. 105, 106, 107, 109
 Weis, T. 3, 60, 67, 68, 105, 106, 126
 Weischedel, R. M. 1, 22, 32, 37, 40
 Wellman, H. M. 18
 Werner, A. 68
 Whiten, A. 15, 21
 Wicklund, R. A. 13, 17
 Wille, M. 29
 Williams, I. J. 9
 Winnykamen, F. 14
 Winograd, T. 43
 Woodruff, G. 18
 Wooldridge, M. 46
 Wright, J. W. 9, 12, 14, 15, 16, 17

 Yonezawa, A. 58

 Zimmermann, D. 3, 60, 126
 Zukerman, I. 22, 33, 37, 40

Sachverzeichnis

- ABCL, 57–58
- AFL
 - als Entscheidungsbäume, 77–79, 85
 - globale, 22, 73–76
 - lokale, 22, 72–73
 - Zusammenstellung von Systemen mit AFL, 36–40
- Agent
 - Definition, 46–47
- Algorithmen
 - ALLOWABLE-MOVES, 117
 - ANTICIPATED-RESPONSE-WITH-PROBABILITY, 89
 - ANTICIPATED-RESPONSES, 87
 - ANTICIPATED-RESPONSE, 81
 - BEST-MOVE-ITERATION, 99
 - BEST-MOVE-LOOK-AHEAD, 115
 - BEST-MOVE-SATISFICING, 97
 - BEST-MOVES-CONSIDERED, 87
 - BEST-MOVE, 65, 76
 - PREDICTED-EVALUATION-SHIFT, 66
 - UTILITY-OF-ANTICIPATED-RESPONSE+, 99
 - UTILITY-OF-ANTICIPATED-RESPONSE-PROBABILISTIC, 89
 - UTILITY-OF-ANTICIPATED-RESPONSE, 81, 87
 - UTILITY-OF-MOVE, 66
 - UTILITY-OF-RESPONSE, 81
- Antizipation, 1–4
 - des Käufersverhalten, 77–83
 - des Käufersverhaltens
 - unter Unsicherheit, 85–90
 - des Verkäufersverhaltens, 91–93
 - Fragestellungen, 6–7
 - in der Benutzermodellierung, 21–22
 - in der Dialogübersetzung, 29–30
 - in der Kommunikationstheorie, 10
 - in der Psycholinguistik, 11–12
 - Kanonische Fälle in PRACMA, 5–6
 - ressourcenadaptive, 95–119
 - Antizipation im Nachhinein, 100–106
 - Iterationsansatz, 98–100
 - mehrstufige, 113–116
 - nach Wert der Berechnung, 106–110
 - Satisficing-Ansatz, 95–97
 - selektive, 110–111
 - selektive Anpassung, 112–113
 - unter Zeitdruck, 116–117
 - zur Verhinderung falscher Inferenzen, 32
- Antizipationsrückkopplungsschleife, *siehe* AFL
- ANTLIMA, 27–28
- Architektur
 - CHANNELS, 46–60
 - Aktoren, 45
 - Blackboard, 43–45
 - Definition, 42
 - heterarchische, 43
 - hierarchische, 43
 - hybride, 46
 - kaskadierte, 43
 - Nachrichtenaustausch vs. Blackboard, 55–56
 - sequentielle, 43
- Benutzermodell, 1, 30, 67, 112
- Bewertungsdimensionen, 62–65, 67, 73, 83, 91, 92
- BILD, 25–26
- CHANNELS
 - Agenten, 48–49
 - asynchrone Kommunikation, 52
 - externe Agenten, 53–55
 - Kommunikationsakte, 49–51
 - Kommunikationsmodus, 51–52
 - Nachrichten, 49–53
 - synchrone Kommunikation, 51–52
 - Syntax, 48
- Eindrucksverschiebung, 92
- Empathie, *siehe* affektive Perspektivenübernahme
- Entscheidungstheorie, 19

- Feedback, 3, 10
- Gleichgewicht, 20
- HAM-ANS, 22–23
- Hörerorientierung, 18
- ICE, 54
- IMP, 24–25
- Interessenprofil, 62, 64
- Kommunikationsakte, 49–51
 - ASK, 49
 - INFORM, 49
 - REPLY, 49
 - weitere, 49–51
- MAUT, 62
- Minimax-Verfahren, 20
- Multi-Agenten-Systeme, 42
- Multi-Attribute-Utility-Theory, *siehe* MAUT
- Multiattributive Objektbewertung, *siehe* MAUT
- NAOS, 26–27
- Nutzen von Dialogzügen, 65–67, 74, 76, 78–80
- PEDRO, 31–32
- Perspective taking, *siehe* Perspektivenübernahme
- Perspektivenübernahme, 12
 - affektive, 14
 - konzeptuelle, 14
 - Stufen der, 16–17
 - visuell-räumliche, 13
- Planoperatoren, bidirektionale, 71
- PMUE, *siehe* Principle of Maximum Expected Utility
- PRACMA
 - COMMENT AND QUESTION HANDLER, 65–67
 - DIALOG PLANNER, 67–68
 - EGO, 68–69
 - EVALUATION HANDLER, 60–64
 - Beispieldialog, 61
 - Beschreibung der Agenten, 60–70
 - Rollen-Transmutierbarkeit, 6, 70–72
- Principle of Maximum Expected Utility, 20
- Protokoll, kommunikationsakt-basiertes, 47
- PVM, 55
- Rückkopplung, *siehe* Feedback
- Rückmeldung, *siehe* Feedback
- RMM, 34
- Role taking, *siehe* Rollenübernahme
- Rolle, Definition, 15
- Rollenübernahme, 14
 - Aspekte, 16
 - assumed similarity, 79
 - Definition, 15
 - individual role taking, 80
 - inferred similarity, 79
 - situational role taking, 79
- Rollenspiel, 15
- Schwellwert, 76, 86, 87, 90, 96, 97, 99, 100, 118
- Selbstüberwachung, 12
- Self-Monitoring, *siehe* Selbstüberwachung
- Spieltheorie, 19
- Sprechakte, 49–51
- Theory of Mind, 17
- Transmutierbarkeit, 6, 70–72
- Unsicherheit in der Antizipation, 85–90
 - der probabilistische Ansatz, 88–90
- Verteilte KI (Distributed AI), 42
- Verteiltes Problemlösen, 42
- WIP-GD, 30–31
- WISHFUL, 33
- ZORA, 28–29