

THEORIEN DER KUNSTLICHEN INTELLIGENZ
ALS PSYCHOLOGISCHE THEORIEN

Inauguraldissertation
der Philosophisch-historischen Fakultät
der Universität Bern
zur Erlangung der Doktorwürde
vorgelegt von
Walter F. Bischof
von Grub-Eggersriet SG

Selbstverlag, Bern, 1982

Von der Philosophisch-historischen Fakultät auf Antrag der
Herren Proff. Dr. N. Foppa und R. Groner angenommen.

Bern, den 11. Februar 1982

Der Dekan: Prof. Dr. A. Esch

Bei der Entstehung dieser Arbeit habe ich aus den Diskussionen mit vielen Leuten gelernt, denen ich an dieser Stelle herzlich danken möchte, vor allem aber

meinem Lehrer Prof. Dr. K. Foppa für seinen Vorschlag, eine Arbeit über die hier diskutierten Fragen zu schreiben und seine anregende Kritik

Dr. Marina Groner für die unzähligen Diskussionen, in denen sie mir half, meine Argumente durchzudenken und auszuformulieren,

Prof. Dr. R. Groner für seine kritischen Bemerkungen zu meiner Arbeit meiner Mutter, die einen Teil der Reinschrift meiner Arbeit besorgte.

Vor allem aber möchte ich meinen Eltern danken, die mir mein Studium ermöglicht haben.

INHALTSVERZEICHNIS

	<u>Seite</u>
I Einleitung	1
II Semantische Repräsentation	3
1. Anforderungen an eine Theorie des Sprachverstehens	3
2. Semantik der Semantischen Repräsentation	5
3. Ambiguitätsfreiheit der Semantischen Repräsentation	9
3.1 Typen von Ambiguitäten	9
3.2 Das Paradigma des systematischen Desambiguierens	10
3.3 Eigenschaften von Desambiguierungsprozessen	11
3.4 Grenzen der Forderung SR2	11
4. Lexikalische Dekomposition	13
4.1 Herleitung der lexikalischen Dekomposition	13
4.2 Motivation der Forderung SR3	17
4.3 Vertreter der lexikalischen Dekomposition in der AI	20
4.4 Eigenschaften der lexikalischen Dekomposition	24
5. Kasussysteme	26
III Parsen	31
1. Woods' Augmented Transition Network Grammatik	33
2. Marcus' Wait-and-See-Parser	40
3. Riesbecks Parser	45
4. Wilks' Parser	50
5. Verstehen gesprochener Sprache	54
6. Partielles Parsen	58
IV Repräsentation von Wissen	62
1. Inferenzen	64
1.1 Welche Inferenzregeln sind notwendig?	65
1.2 Wann werden Inferenzen gemacht?	66
2. Einige Repräsentationsformalismen	68
3. Semantische Netze	71
3.1 Erste Formen der Semantischen Netze	71
3.2 Weiterentwicklungen in der Künstlichen Intelligenz	76
3.3 Weiterentwicklungen in der Psychologie	84
V Organisation des Wissens	88
1. Kombinatorische Explosion	89
2. Minskys Frameansatz	92
3. Charniaks Frameansatz	96
3.1 Framesformalismus	96
3.2 Framesanwendung	98

	<u>Seite</u>
4. Der Framesansatz von Schank und Abelson	102
5. Grundlagen der Wissensorganisation	105
5.1 Frametypen	105
5.2 Repräsentationsfragen des Framesansatzes	105
5.3 Framesansatz des Sprachverstehens	106
5.4 Wissensorganisation in Frames	108
VI Entwicklung von Theorien kognitiver Leistungen	113
1. Einleitung	113
2. Das Paradigma der Kognitiven Wissenschaft	115
2.1 Wissenschaftsgeschichtliche Hintergründe	116
2.2 Die Trennung von Berechnung und Mechanismus	122
2.3 Komplexität von Algorithmen und funktionale Architektur	124
3. Gegenstandsbereich, Erklärung, Ueberprüfung	129
3.1 Gegenstandsbereich	130
3.2 Ueberprüfung von Theorien	134
3.3 Produkt- und Indikatorphänomene	140
3.4 Erklärung	142
4. Entwicklung von Theorien	146
4.1 Sparsamkeit von Theorien	146
4.2 Grenzen einer schrittweisen Verfeinerung von Theorien	147
4.3 Formalisierung von Theorien	152
4.3.1 Ableitung von Voraussagen	153
4.3.2 Einfachheit von Formalismen	154
4.3.3 Theorie und Programm	156
VII Zusammenfassung	159
Literatur	160

I. E I N L E I T U N G

In dieser Arbeit soll gezeigt werden, wie die in der Künstlichen Intelligenz (AI) verwendeten Methoden dazu eingesetzt werden können, um eine leistungsfähige Theorie des Sprachverstehens zu entwickeln.

Es gibt eine ganze Reihe von Gründen, weshalb es sinnvoll erscheint, die Methoden der Künstlichen Intelligenz auch in der Entwicklung einer psychologischen Theorie einzusetzen. Die wichtigsten unter ihnen leiten sich ab aus Forderungen, die wir an eine Theorie stellen und aus Eigenschaften, die wir von einer Theorie erwarten.

So glauben wird, dass eine gute Theorie des Sprachverstehens (oder anderer intelligenter Leistungen) viel zu komplex sein wird, als dass sie sich leicht und überschaubar axiomatisieren lässt. Dabei bleibe vorläufig dahingestellt, ob eine Theorie des Sprachverstehens prinzipiell komplex sein muss, oder ob sich Sprachverstehen letztlich nicht doch einfach erklären lässt. Tatsache bleibt, dass wir heute noch zu wenig darüber wissen, wie eine einfache und leistungsfähige Theorie des Sprachverstehens aussehen könnte. Wie in dieser Arbeit noch gezeigt wird, erlauben es die Methoden der Künstlichen Intelligenz, sowohl Modelle von sehr grosser Komplexität zu entwickeln und auf ihre Leistungsfähigkeit zu überprüfen, als auch verschiedene Modelle miteinander zu kombinieren, vorausgesetzt, dass sie genügend weit formalisiert werden können.

Im weiteren fordern wir von einer Theorie des Sprachverstehens, dass sie als Prozessbeschreibung konzipiert sein soll und dass dieser Prozess unter zeitlichen und räumlichen (d.h. Gedächtniskapazitäts-) Beschränkungen operieren soll. Aus dieser Forderung ergeben sich zwei wichtige Konsequenzen. Erstens einmal können in einem solchen Unterfangen reine "Strukturmodelle" nicht genügen. Wir werden aber ab Beispielen sehen, welche Bedeutung ihnen im Rahmen von Prozessmodellen zukommt und welche Implikationen sich aus Strukturmodellen auf den Design von Prozessmodellen ergeben können. In der Forderung, dass der Prozess des Sprachverstehens unter zeitlichen und räumlichen Beschränkungen operieren soll, liegt der entscheidende Unterschied zwischen dem AI Paradigma des Sprachverstehens und dem linguistischen Paradigma, wie es sich seit Chomsky entwickelt hat. So können sich

verschiedene Prozessmodelle, die ohne zeitlich-räumlich Beschränkungen gleich leistungsfähig sind und daher aus der Sicht der Linguistik als gleichwertig zu beurteilen sind, in ihrer Effizienz sehr stark unterscheiden, wenn sie unter diesen Beschränkungen operieren müssen. Die in der Künstlichen Intelligenz gegebene Möglichkeit der Effizienzbeurteilung von Prozessmodellen scheint uns daher auch aus psychologischer Sicht von entscheidender Bedeutung, da auch der Mensch solchen Beschränkungen unterworfen ist.

Die vorliegende Arbeit gliedert sich in fünf Teile. Im ersten Teil werden Randbedingungen definiert, unter denen die Modelle des Sprachverstehens zu beurteilen sind und insbesondere werden Eigenschaften des Produkts des Verstehensprozesses, der semantischen Repräsentation, definiert. Im zweiten Teil werden verschiedene Parsermodelle diskutiert und einem detaillierten Leistungsvergleich unterzogen. Im dritten und vierten Teil wird das Problem der Repräsentation und der Organisation von Wissen diskutiert, was weiter zum Verständnis des Verstehensprozesses selbst, aber auch seinem Produkt, der semantischen Repräsentation, hilft. Im letzten Teil wird schliesslich das Problem der Theorienentwicklung in der kognitiven Psychologie und in der interdisziplinären Zusammenarbeit zwischen der kognitiven Psychologie und der AI diskutiert. Dazu wird vom Paradigma der Kognitiven Wissenschaft ausgegangen und gezeigt, welche Vorteile es gegenüber den beiden andern Disziplinen hat, aber auch welche Implikationen sich aus seiner Position für die Eigenschaften von Theorien des Sprachverstehens und intelligenter Leistungen im allgemeinen ergeben.

II. SEMANTISCHE REPRESENTATION

1. ANSPRUECHE AN EINE THEORIE DES SPRACHVERSTEHENS

Eine Theorie des Sprachverstehens soll erklären, wie sprachliche Äußerungen auf Gedächtnisstrukturen abgebildet werden. Mit dieser Definition wird Sprachverstehen von andern Leistungen wie etwa Sprechen, d.h. Spracherzeugung, oder Sprechsteuerung, etwa im Dialog, abgegrenzt. Eine Abgrenzung verschiedener sprachlicher Leistungen innerhalb des gesamten sprachlichen Verhaltens ermöglicht es wohl, Theorien mit einem bescheideneren Leistungsanspruch zu entwickeln, ob dies aber im Hinblick auf eine Theorie des gesamten sprachlichen Verhaltens oder sogar des ganzen kommunikativen Verhaltens sinnvoll ist, bleibe hier dahingestellt.

Im weiteren sollte diese Theorie des Sprachverstehens nicht auf menschliches Sprachverstehen allein beschränkt bleiben, sondern sollte auch zeigen, welche Leistungen irgendein, ob biologisches oder künstliches System erbringen muss, um Sprache verstehen zu können.

Im folgenden werden einige Modelle des Sprachverstehens aus der Künstlichen Intelligenz besprochen. Bei den meisten dieser Modelle wird der Leistungsanspruch noch weiter reduziert, um die Komplexität der Modelle auf ein vernünftiges Mass zu reduzieren. Auf der Seite der sprachlichen Eingabe sind vor allem zwei Einschränkungen von Bedeutung.

Erstens enthalten die meisten der besprochenen Modelle weder optische noch auditive Spracheingabe mit der Ausnahme von sehr wenigen Spracherkennungssystemen. Damit wird es unnötig, auditive oder graphische Erkennungsprozesse modellieren zu müssen, andererseits hat damit aber die sprachliche Eingabe auch die unnatürliche Eigenschaft, dass sie fehlerfrei und vollständig (d.h. ohne fehlende Wort- oder Textfragmente) ist. Es kann gezeigt werden, dass diese Einschränkung wichtige Konsequenzen auf die erarbeiteten Modelle hat und darf daher diese Modelle immer nur mit Rücksicht auf diese gesetzten Randbedingungen beurteilen.

Eine zweite Einschränkung besteht darin, dass die meisten Modelle nur Sätze oder einfache Text- oder Dialogfragmente verarbeiten. Diese Einschränkung ist darauf zurückzuführen, dass bereits kurze, auch inhaltlich abgeschlossene Textfragmente so viele Probleme aufwerfen, dass eine Verarbeitung ganzer Texte oder Dialoge heute noch kaum machbar wäre. (Eine Ausnahme dazu stellt das Projekt von Rieger (1977, 1979) dar).

Auf der Seite der erzeugten Gedächtnisstrukturen, im folgenden als semantische Repräsentation (SR) bezeichnet, lässt sich nur sehr ungenau umschreiben, welche Leistungen ein Sprachverstehensmodell zu erbringen hat. Die semantische Repräsentation ist weder a priori gegeben noch gibt es heute ein einigermaßen genügendes Modell einer semantischen Repräsentation. Für die erzeugte SR können daher nur einige Charakteristiken angegeben werden, die - so glauben wir - von jeder möglichen semantischen Repräsentation erfüllt werden müssen.

Im folgenden soll auf vier Forderungen an die SR eingegangen werden, nämlich:

SR 1) Die SR soll die Bedeutung von Sätzen repräsentieren.

SR 2) Die SR soll keine Ambiguitäten enthalten.

SR 3) Sätze mit gleicher Bedeutung sollen die gleiche SR haben.

SR 4) Die SR soll die funktionale Struktur von Sätzen aufzeigen.

Diese Forderungen sind, wie im folgenden noch gesehen wird, keineswegs voneinander unabhängig, betonen aber jeweils etwas andere Aspekte. Eine befriedigende Erfüllung aller dieser Forderungen würde bedingen, dass man genau wüsste, wie sprachliche Äußerungen formal repräsentiert werden können und dass die Semantik verwendeter Formalismen eindeutig festgelegt ist. Die Probleme der ersten Bedingung sind allgemein und insbesondere aus der Linguistik bekannt. Auf sie soll hier aber nicht weiter eingegangen werden. Mit dem zweiten Problem, der Semantik der verwendeten Repräsentationsformalismen hat sich Pat Hayes (1977a) kritisch auseinandergesetzt. Auf seine Argumente werden wir im folgenden noch eingehen.

Da die Bedingungen für eine formale und befriedigende Festlegung der semantischen Repräsentation nicht gegeben sind, müssen wir uns mehr oder weniger auf intuitive Argumente verlegen. Einerseits erlaubt dies, Probleme zu ignorieren, mit denen sich schon andere Disziplinen mit mässigem Erfolg beschäftigen, und uns auf Probleme der Konstruktion einer mehr oder weniger befriedigenden semantischen Repräsentation zu konzentrieren. Andererseits muss einschränkend gesagt werden, dass damit vielleicht eine ganze Reihe von zusätzlichen Problemen im Prozess des Sprachverstehens verdeckt werden.

2. SEMANTIK DER SEMANTISCHEN REPRÄSENTATION

Als erste Forderung an die semantische Repräsentation stellen wir die folgende auf:

SRL: die SR SOLL DIE BEDEUTUNG VON SÄTZEN REPRÄSENTIEREN.

Das Hauptproblem dieser Forderung liegt sicher darin, festzulegen, was mit der Bedeutung von Sätzen gemeint ist. Eine Einschränkung, die für die hier besprochenen Ansätze gemacht werden kann, ist die, dass auf die Repräsentation von konnotativen Bedeutungsaspekten verzichtet wird. Aber auch im denotativen Bedeutungsaspekt werden entscheidende Einschränkungen gemacht und man kann sie am ehesten verdeutlichen, wenn man vom Begriff des 'zentralen Bedeutungsaspekts' ausgeht.

Wilks (1976) hat die Unterscheidung zwischen strukturell motivierten und inhaltlich motivierten Systemen eingeführt. Strukturell motivierte Systeme sind solche Systeme, die in erster Linie dazu entwickelt wurden, um nicht-linguistische Probleme zu lösen. Sie brauchen die Bedeutung von Sätzen nur so weit zu analysieren, wie sie mit den Problemen, die das System lösen kann, in Verbindung gebracht werden kann. Wenn man etwa dem Informationssuchsystem LUNAR (Woods et al., 1972, S.G.15) den Befehl (1)

(1) Can you give me all chromite analyses for those samples eingibt, dann ist für das System einzig der Bedeutungsaspekt von "give" im Sinne von "herausdrücken" wichtig, während es für das System völlig unwichtig ist, dass "give" auch als physisches Uebergeben eines Objektes oder in weiteren Bedeutungen interpretiert werden kann. Der zentrale und einzige Bedeutungsaspekt, den das System verwendet, beschränkt sich auf "give" im Sinne von "herausdrücken".

Im Gegensatz dazu befasst man sich bei inhaltlich orientierten Systemen mit den Problemen des Sprachverstehens selbst. Dazu gehört in einem allgemeinen Sinne auch die Auseinandersetzung mit dem Bedeutungsreichtum der Sprache und den daraus folgenden potentiellen Ambiguitäten in der Sprache (Auf den Begriff der Ambiguität wird im nächsten Abschnitt genauer eingegangen). Es zeigt sich aber auch bei diesen Systemen, dass nur einzelne Bedeutungsaspekte erfasst werden. Wenn etwa Wilks (1973 b) das Wort "Polizist" als "eine Person, die böse Personen aus der Menge von Leuten herausnimmt" repräsentiert, dann wird dies nur allzu deutlich. Ähnlich geht auch Schank vom Begriff des zentralen Bedeutungsaspektes aus, wenn er argumentiert, dass in der semantischen Repräsentation nur sichere Inferenzen (z.B. "Eine Frau ist ein Mensch") explizit repräsentiert werden. Obwohl dieses Argument an und für sich einleuchtend ist, ist in der Praxis unklar, wie 'zentrale Bedeu-

tung' besser als ad hoc definiert werden kann.

Andererseits muss gewährleistet werden, dass solche Bedeutungsaspekte, die als wichtig beurteilt werden, in der semantischen Repräsentation auch wirklich repräsentiert werden können, d.h., dass die SR gewissen Anforderungen an die Ausdrucksmächtigkeit genügt. Auf dieses Problem werden wir im Zusammenhang mit der Quantifikation in semantischen Netzwerken noch eingehen und zeigen, dass die Erfüllung einer solchen Forderung schwierig sein kann.

Die Beurteilung der Leistungsfähigkeit einer SR (im Sinne ihrer Ausdrucksmächtigkeit) bedingt aber, dass über die Semantik der SR selbst Klarheit besteht. Es muss also klar sein, was ein Ausdruck wie

(2) (ATRANS FRITZ FRITZ PETER BUCH)

bedeuten soll. Wir wollen hier auf drei verschiedene Ansätze eingehen, nämlich den prozeduralen Ansatz von Woods (1973, Woods et al. 1972), den Abbildungsansatz und den modelltheoretischen Ansatz (z.B. Pat Hayes, 1977 a).

Im prozeduralen Ansatz von Woods wird die Semantik eines Ausdrucks durch die Prozedur, auf die sich dieser Ausdruck bezieht, definiert.

Die semantische Repräsentation (3)

(3) (FOR EVERY X13 / (SEQ SAMPLES) : (CONTAIN X13 (NPR* X15 / (QUOTE CHROMITE)) (QUOTE NIL)) ; (FOR EVERY X18 / (SSUNION X1 / (SEQ MAJORELTS) : T ; (DATAFILE (WHQFILE X 3) X13 (NPR* X20 / (QUOTE CHROMITE)) X1)) : T ; (PRINTOUT X18)))

die dem Eingabesatz (1) entspricht, ist definiert durch die wohldefinierte Sequenz von Operationen, die das System aufgrund dieses Ausdrucks ausführt und zum Ausdruck einer Liste von Mondprobenbezeichnungen führt. In gleicher Weise ist im Programm SHRDLU von Winograd (1972) der Ausdruck

(4) (THAND (THGOAL (\$PICKUP :BLOCK23))

(THGOAL (\$PUTIN :BLOCK23 :BOX7)))

der dem Befehl

(5) Pick up the block and put it in the box.

entspricht, durch die Prozeduren, die zur Ausführung dieses Befehls führen, definiert. Einerseits ist die Semantik der SR in solchen Systemen wohl definiert, andererseits ist sie aber auch beschränkt auf die abgeschlossene Mikrowelt, in der das System Operationen ausführen kann.

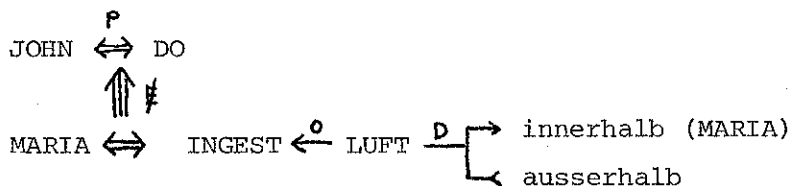
Schwieriger wird die Bestimmung der Semantik von Ausdrücken dann, wenn ein System nicht auf geschlossenen Mikrowelten mit wohldefinierten Objekten und Handlungen beschränkt bleibt, wie dies bei den hier diskutierten Systemen der Fall ist. Es gibt nun keinen unmittelbar einleuchtenden Weg mehr, um zu zeigen, wie sich Ausdrücke der semantischen Repräsentation auf bestimmte Konfigurationen der Welt (oder einer möglichen Welt)

beziehen. Ein anderer Weg, den wir Abbildungsansatz bezeichnen wollen, besteht nun darin, dass man Regeln angibt, wie man einen Ausdruck der SR in die natürliche Sprache übersetzen kann. So hat der Satz

(6) Hans würgte Maria.

in Schanks (1972) System die semantische Repräsentation

(7)



die aufgrund von (informalen) Interpretationsregeln wiederum umschrieben werden kann als

(8) Hans tat etwas, das es Maria nicht ermöglichte, Luft von aussen ins Innere von Maria zu nehmen.

Die Beurteilung darüber, ob (7) eine adäquate Repräsentation von (6) ist, hängt davon ab, ob man die beiden Sätze (8) und (6) als gleichbedeutend beurteilt oder nicht. In dieses Urteil spielt aber auch wieder hinein, was man als den zentralen Bedeutungsaspekt des Satzes (6) betrachtet. Dieser Ansatz hat mehrere Schwächen. Erstens ist die Beurteilung der Gleichbedeutung von Sätzen sehr vage und geht nicht über eine intuitive Ebene hinaus. Zweitens müssten für eine präzisere Bestimmung der Semantik von Ausdrücken die informalen Interpretationsregeln durch formale Regeln ersetzt werden, die angeben, wie Ausdrücke der semantischen Repräsentation auf die natürliche Sprache abgebildet werden. Wilks (1976) argumentiert, dass diese Regeln im Prozess gegeben sind, der natürlichsprachliche Ausdrücke auf Ausdrücke der semantischen Repräsentation abbildet, nämlich dem Sprachverstehensprozess. Die Regeln wären aber dann am einfachsten, wenn man gleich die natürlichsprachlichen Ausdrücke als ihre semantische Repräsentation einsetzen würde. In dieser Richtung argumentiert denn auch Wilks, wenn er sagt, dass die SR und natürlichsprachliche Ausdrücke keinen verschiedenen ontologischen Status haben und die SR nur eine reduzierte Mikrosprache ist, in der nur noch grobe Bedeutungsunterschiede ausgedrückt werden können (dass die natürlichen Sprachen und die SR eine verschiedene Syntax haben können, spielt in diesem Zusammenhang keine Rolle). Wenn nun aber die Semantik eines SR-Ausdrucks über den Sprachanalyseprozess selbst definiert ist, dann gibt es kein unabhängiges Kriterium mehr, um die Korrektheit des Sprachanalyseprozesses zu beurteilen und in diesem Sinne argumentiert denn auch

Wilks, dass ein solches System nur noch auf der Basis seines gesamten Verhaltens beurteilt werden kann.

Dieser 'Abbildungsansatz' wurde von Pat Hayes (1977 a) heftig kritisiert mit dem bereits erwähnten Argument der Vagheit des Begriffs "bedeutungsgleich", aber auch mit dem Argument, dass damit keine Analyse der Bedeutung eines Textfragmentes gemacht werden kann, weil die SR nicht mehr unabhängig vom Sprachanalyseprozess gerechtfertigt werden kann.

Ein anderes wichtiges Argument besteht darin, dass durch die mangelnde Axiomatisierung der Primitiva in der SR (Primitiva im logischen, nicht im linguistischen Sinn) es nicht möglich ist, die Semantik der SR präzise festzulegen. Deshalb ist es auch immer wieder schwierig gewesen, Unterschiede zwischen verschiedenen Formalismen (etwa semantische Netzwerke versus Prädikatenkalkül versus Frames) exakt festzulegen. Andererseits ist die Formalisierung einer Repräsentation und die saubere Ausarbeitung der Repräsentation von Alltagswissen eine aufwendige Arbeit, wie die Arbeit von Pat Hayes (1978) über die Repräsentation von Wissen über Flüssigkeiten zeigt. Wichtig ist aber in diesem Zusammenhang die Tatsache, dass sich die Probleme des Sprachverstehens grundsätzlich auch dann untersuchen lassen, wenn die SR diesen Ansprüchen nicht genügt, d.h., dass die präzise Formalisierung der SR nicht eine notwendige Bedingung für die Entwicklung eines Modells des Sprachverstehens ist. Hingegen ist einleuchtend, dass man dadurch häufig in Gefahr läuft, sich auf irrelevante Probleme zu konzentrieren oder in einem andern Formalismus bereits gelöste Probleme nochmals löst. (Ein Beispiel des letzteren ist die "Wiederentwicklung" der Quantifizierung in semantischen Netzwerken, obwohl dieses Problem im Prädikatenkalkül bereits gelöst war und man durch geeignete Formalisierung der semantischen Netzwerke ihre Äquivalenz mit dem Prädikatenkalkül leicht nachweisen konnte).

In dieser Arbeit werden wir aber trotz dieser Einwände die SR durch intuitive Kriterien festlegen, um uns dafür auf die eigentlichen Probleme des Sprachverstehens konzentrieren zu können.

3. AMBIGUITÄTSFREIHEIT DER SEMANTISCHEN REPRÄSENTATION

Ein wichtiges Charakteristikum natürlicher Sprachen besteht darin, dass sie systematisch Ambiguitäten enthalten. Wir wollen hier nicht auf die Gründe eingehen, weshalb die natürlichen Sprachen so viele Ambiguitäten enthalten, sondern wollen dies einfach als empirische Tatsache dahingestellt lassen. Genau diese Eigenschaft soll aber in der semantischen Repräsentation nicht enthalten sein. Deshalb stellen wir als zweite Forderung an die semantische Repräsentation auf:

SR2: DIE SR SOLL KEINE AMBIGUITÄTEN ENTHALTEN.

Aus dieser Forderung leitet sich unmittelbar ab, dass eine der wichtigsten Aufgaben eines sprachverstehenden Systems darin besteht, alle in der natürlichen Sprache enthaltenen Ambiguitäten zu entdecken und zu desambiguieren.

3.1. Typen von Ambiguitäten

Wir wollen zuerst eine einfache Klassifikation von Ambiguitätstypen geben. Wir übernehmen diese Klassifikation der Uebersicht halber, werden uns aber später noch mit den Motiven, die dieser Klassifikation traditionell zugrundeliegt, kritisch auseinandersetzen.

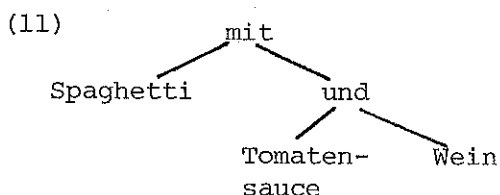
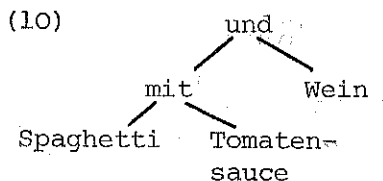
a) lexikalische Ambiguitäten: Das Wort "Hahn" kann sowohl Wasserhahn, als auch Gewehrhahn, als auch ein männliches Tier bedeuten.

b) strukturelle Ambiguitäten: Der Satz

(9) Ich möchte gern Spaghetti mit Tomatensauce und Wein.

kann zwei verschiedene syntaktische Lesungen haben.

Er kann besagen, dass ich Spaghetti mit Tomatensauce gemischt und dazu Wein möchte (10) oder dass ich die Spaghettis, die Tomatensauce und den Wein alles vermischt möchte (11).



c) Ambiguität der zugrundeliegenden Struktur: Der Satz

(12) Der Hauptmann befahl den Polizisten das Trinken zu beenden
kann entweder besagen, dass die Polizisten aufhören sollen zu trinken,
oder dass sie verhindern sollen, dass andere Leute trinken.

Im Gegensatz zu Satz (9) haben die beiden Lesungen von (12) die gleiche Tiefenstruktur. Sie unterscheiden sich dadurch, dass sie verschiedene Kasuszuordnungen haben: Im ersten Fall sind die Polizisten Agens von "trinken", im zweiten Fall sind es unspezifizierte, andere Personen.

d) Referenzambiguität: Im Satz

(13) Hans verkaufte seinen Hund, da er krank war.

kann sich das Pronomen "er" sowohl auf Hans als auch auf Hund beziehen.

3.2. Das Paradigma des systematischen Desambiguierens

In diesem Zusammenhang muss darauf hingewiesen werden, welche Rolle Ambiguitäten und damit der Prozess des Desambiguierens im Sprachverstehen spielen. Dazu wollen wir zwischen "echten Ambiguitäten" und "potentiellen Ambiguitäten" unterscheiden. Potentielle Ambiguitäten sind solche Ambiguitäten, die aus ihrem Kontext mit Hilfe syntaktischer, semantischer und pragmatischer Information desambiguiert werden können. Echte Ambiguitäten können aus ihrem Kontext nicht desambiguiert werden.

Die meisten Ambiguitäten der natürlichen Sprachen sind nur potentiell ambig. Wir sind in unserem täglichen Sprachgebrauch so gewöhnt, diese Ambiguitäten zu lösen, dass sie uns gar nicht mehr auffallen. Im Gegenteil, es braucht einige Erfahrung, um in einem Text alle potentiellen Ambiguitäten entdecken zu können. Sogar echte Ambiguitäten werden meist nicht entdeckt, da wir uns immer, wenn auch oft "unbegründet", für eine der Lesungen entscheiden (Bever, Garrett & Hurtig, 1973). So ist es denn auch nicht verwunderlich, wenn die Betonung der systematischen Desambiguierung in der AI unserer Intuition anfänglich widerspricht.

Aber wenn die Ambiguitäten der natürlichen Sprachen in der SR nicht vorhanden sein sollen, dann ist es unmittelbar evident, dass im Verlaufe des Sprachverstehens irgendwo eine Desambiguierung stattfinden muss.

Es ist unserer Ansicht nach das Verdienst der AI, dass sie, basierend auf den Misserfolgen der frühen Versuche maschineller Uebersetzung, die zentrale Rolle der Desambiguierung innerhalb des Sprachverstehens hervorgehoben hat.

3.3. Eigenschaften von Desambiguierungsprozessen

Desambiguierungen können nie mit Sicherheit gemacht werden. Es können zu jeder Zeit im Text neue Informationen auftauchen, die der bisher plausibelsten Interpretation widersprechen und nur mit einer bisher verworfenen Interpretation in Einklang stehen. Dies bedingt jedoch, dass jedes sprachverstehende System fähig sein muss, seine Hypothesen über die Interpretation von Sätzen zu korrigieren.

Die eingeführte Ambiguitätstypologie ist traditionell häufig mit einem hierarchischen Modell des Sprachverstehens verbunden gewesen, in dem der Prozess des Sprachverstehens aus einer Sequenz von syntaktischen, semantischen und pragmatischen Analyseeinheiten besteht. Es muss aber betont werden, dass diese Korrespondenz zwischen Ambiguitätstypen und identifizierbaren Prozesskomponenten nicht notwendigerweise bestehen muss. So erfüllt denn diese Ambiguitätstypologie in erster Linie ein didaktisches Ziel. Ob sie sich aber auch aus der Sicht der Prozessstruktur rechtfertigen lässt, bleibe hier dahingestellt.

3.4. Grenzen der Forderung SR 2

Im Zusammenhang mit der systematischen Desambiguierung muss zwischen ambigen und vagen Ausdrücken unterschieden werden (siehe z.B. Wahlster, 1977), auch wenn diese Unterscheidung nicht scharf abgegrenzt werden kann. Ein ambiger Ausdruck kann in mehreren Bedeutungen verwendet werden. Bei einem vagen Ausdruck hingegen gibt es keine eindeutigen Konventionen, ob er in einer bestimmten Situation anwendbar ist, oder bestimmte relevante Aspekte der Aussage bleiben offen. So sind zum Beispiel "jung" und "Onkel" vage Ausdrücke. Im Satz

(14) Fritz ist jung.

ist nicht festgelegt, wie alt Fritz genau ist, es wird nur ein ungefähre Bereich festgelegt (vielleicht "jünger als 20"?). Ähnlich ist beim Wort "Onkel" nicht festgelegt, ob es sich um den Bruder des Vaters oder den Bruder der Mutter handelt.

Wenn man die Unterscheidung zwischen ambigen und vagen Ausdrücken nicht macht, müsste (14) als ambiger Satz angesehen werden, dessen eine Lesung vielleicht

(15) Fritz ist 17 Jahre und 2 Monate alt.

wäre. Ähnlich musste auch im Verwandtschaftssystem von Lindsay (1963) "Onkel" als ambiger Ausdruck angesehen werden, da in jener SR "Onkel" auf "Bruder des Vaters" oder "Bruder der Mutter" reduziert werden musste. Durch die Einführung vager Konzepte und Prädikate in der SR wird aber verhindert,

dass solche unplausible Desambiguierungen wie (15) vorgenommen werden müssen. Damit stösst die Forderung SR2 auch an ihre Grenzen.

4. LEXIKALISCHE DEKOMPOSITION

Wenn wir von der unter SRI vertretenen Position ausgehen, dass die SR nur den zentralen Bedeutungsaspekt von Sätzen erfassen soll, d.h. jenen Teil der Bedeutung, der für das Verstehen am wichtigsten ist, dann haben die folgenden Satzpaare die gleiche Bedeutung:

(16a) Fritz erkältete sich.

(16b) Fritz holte sich eine Erkältung.

(17a) Fritz küsste Maria

(17b) Maria wurde von Fritz geküsst.

Als dritte Forderung an die SR stellen wir nun die folgende auf:

SR3: SAETZE MIT GLEICHER BEDEUTUNG HABEN DIE GLEICHE SR.

Was unter "gleicher Bedeutung" verstanden werden soll, kann nicht mit einem einfachen Kriterium festgelegt werden. Der Begriff der logischen Aequivalenz hilft im Falle der natürlichen Sprache nicht viel, da ja nicht a priori festgelegt ist, wie die sprachlichen Aeusserungen auf die logischen Prädikate abgebildet werden. Die folgenden Beispiele werden uns aber einen intuitiven Begriff davon geben.

4.1. Herleitung der lexikalischen Dekomposition

Im folgenden werden die Folgen der Forderung SR3 für die semantische Repräsentation diskutiert. Dazu gehen wir von einigen wenigen Beispielsätzen aus und leiten aus einer oberflächlichen Repräsentation eine immer tiefere Repräsentation ab. Wir nehmen nun an, dass die Sätze

(18) Fritz gab Peter ein Buch.

(19) Ein Buch wurde Peter von Fritz gegeben.

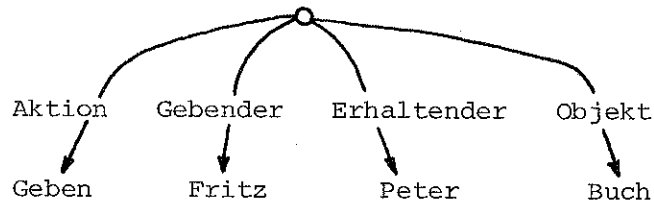
das Gleiche bedeuten. Die Vernachlässigung der Bedeutungsveränderung durch die Aktiv-Passiv-Transformation stehe hier stellvertretend für alle syntaktischen Regeln (siehe jedoch, Katz & Fodor, 1964, Hall-Partee, 1971). Es ist damit bereits klar, dass jene Aspekte, die mit Hilfe syntaktischer Transformationen ausgedrückt werden können, wie Focus und Thema, in der SR verloren gehen (siehe dazu aber Kay, 1975).

Den beiden Sätzen (5) und (6) wollen wir vorläufig die folgende semantische Repräsentation zuordnen:

(20) (GEBEN FRITZ PETER BUCH)

oder in Kasusnotation

(21)



Zur Vereinfachung unserer Argumentation lassen wir die Zeitbestimmung in unseren Beispielrepräsentationen weg.

Wir nehmen nun weiter an, dass auch (22) das Gleiche bedeutet wie (18) und (19)

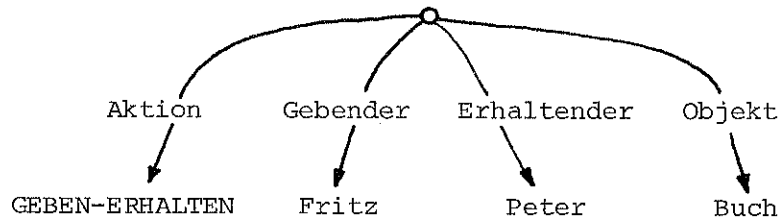
(22) Peter erhielt von Fritz ein Buch.

und allen drei Sätzen die gleiche SR zuordnen.

(23) (GEBEN-ERHALTEN FRITZ PETER BUCH)

oder in Kasusnotation

(24)



Mit der Aktion "GEBEN-ERHALTEN" ist in unserer SR ein Element aufgetaucht, das von den lexikalischen Items "geben" und "erhalten" prinzipiell verschieden ist. Es ist ein Kunstwort, oder in linguistischer Terminologie ein lexikalisches Primitiv. Es taucht nur in der SR auf und kann sich auf der Oberfläche eines Satzes verschieden realisieren. Wie es realisiert wird, wird durch Abbildungsregeln festgelegt. Wenn man als Subjekt des einfachen, deklarativen und aktiven Satzes den Kasus "Gebender" wählt, realisiert sich GEBEN-ERHALTEN im lexikalischen Item "geben", wenn hingegen "Erhaltender" als Subjekt gewählt wird, realisiert sich GEBEN-ERHALTEN als "erhalten". Wir haben also folgende Abbildungsregeln:

(25) GEBEN-ERHALTEN

Gebender = Subjekt —————> Verb=geben

Erhaltender = Subjekt —————> Verb=erhalten

In der Repräsentation des Satzes

(26) Fritz tauschte ein Buch gegen Peters Ball.

kann argumentiert werden, dass "tauschen" eigentlich aus zwei Aktionen besteht, den beiden Uebergaben des Tauschgegenstandes, und dass sich diese beiden Aktionen gegenseitig verursachen (begründen). Demnach könnte (26) folgen-

dermassen repräsentiert werden:

- (27) a (GEBEN-ERHALTEN FRITZ PETER BUCH)
- b (GEBEN-ERHALTEN PETER FRITZ BALL)
- c (VERURSACHEN a b)
- d (VERURSACHEN b a)

In diesem Fall wird ein lexikalisches Item nicht mehr durch ein einziges lexikalisches Primitiv repräsentiert, das für eine ganze Klasse von lexikalischen Items verwendet wird, sondern durch eine Kombination von lexikalischen Primitiva. Das lexikalische Item "tauschen" wird also in eine Kombination von lexikalischen Primitiva aufgegliedert, die alle einen Teil der Bedeutung von "tauschen" repräsentieren, daher auch die Bezeichnung lexikalische Dekomposition. Dass in diesem Fall ein einziges Primitiv mehrmals in der SR vorkommt, ist nur ein Spezialfall.

Der Satz (28) führt uns zu einer weiteren Eigenschaft der SR.

(28) Fritz kaufte ein Brot beim Bäcker.

Wir können "kaufen" als einen Spezialfall von "tauschen" auffassen, bei dem einer der Tauschgegenstände Geld ist. Demnach würde (28) folgendermassen repräsentiert:

- (29) a (GEBEN-ERHALTEN BAECKER FRITZ BROT)
- b (GEBEN-ERHALTEN FRITZ BAECKER GELD)
- c (VERURSACHEN a b)
- d (VERURSACHEN b a)

Mit dem Tauschgegenstand Geld ist nun etwas in der semantischen Repräsentation aufgetaucht, das im ursprünglichen Satz (28) gar nicht spezifiziert war. Offensichtlich haben wir aus der Bedeutung von "kaufen" abgeleitet, dass einer der Tauschgegenstände Geld war, oder in linguistischer Terminologie, wir haben eine lexikalische Präsupposition gemacht (Kiparsky & Kiparsky, 1970; King, 1977).

Nun folgt aber nicht mit logischer Notwendigkeit, dass einer der Tauschgegenstände Geld war. Wenn wir nach dem Satz (28) erfahren, dass Fritz mit einem Scheck bezahlte, oder anschreiben liess, führt dies nicht zu einer Inkonsistenz. Wir nehmen nur in Abwesenheit spezifischer gegenteiliger Information an, dass Fritz mit Geld bezahlte.

Wir kehren nun noch einmal zu den Sätzen (18) und (19) zurück und vergleichen sie mit

(30) Peter nahm ein Buch von Fritz.

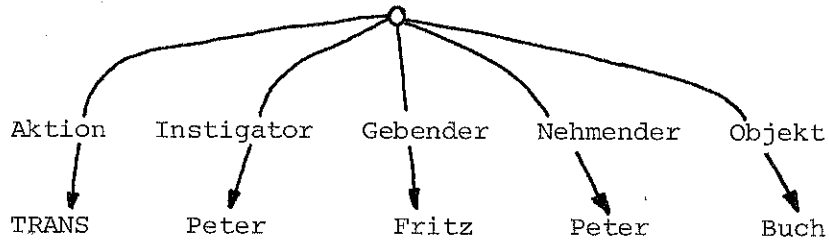
In (30) ist es Peter, der die Aktion des Besitzwechsels auslöst, während es in (18) und (19) Fritz ist. Um alle drei Sätze gleich repräsentieren zu

können, führen wir ein neues Primitiv TRANS ein, in dem zusätzlich zu den übrigen Kasus auch noch der Instigator der Aktion spezifiziert wird. Demnach wird (30) folgendermassen repräsentiert:

(31) (TRANS PETER FRITZ PETER BUCH)

oder in der zweiten Notation

(32)



Entsprechend würden (18) und (19) durch

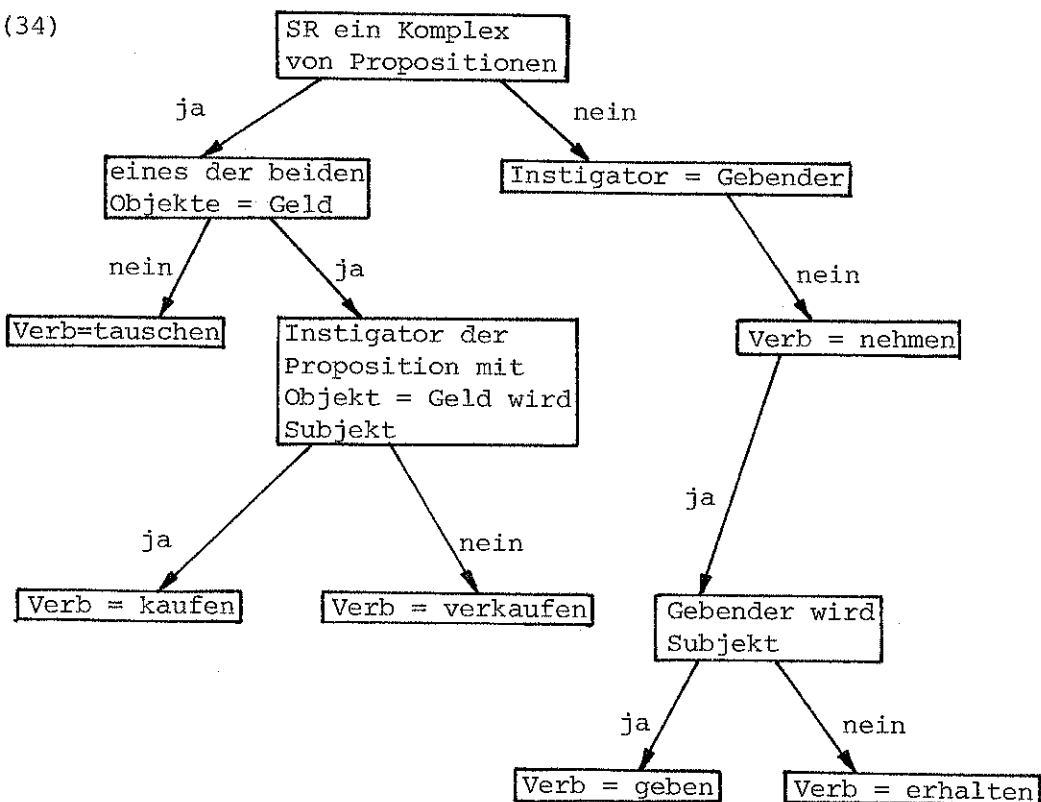
(33) (TRANS FRITZ FRITZ PETER BUCH)

repräsentiert. (31) unterscheidet sich von den vorherigen Repräsentationen dadurch, dass das gleiche Objekt (die gleiche Nominalphrase in unserem Beispiel) mehrmals in der SR auftaucht. Damit ist die Abbildung der SR-Kasus auf die Oberflächenkasus nicht mehr eineindeutig.

Bei der Diskussion von Schanks Ansatz in 4.3. wird der Leser bemerken, dass das in (31) eingeführte Primitiv TRANS Schanks ATRANS entspricht.

Abschliessend wollen wir zur Uebersicht zeigen, wie sich das Primitiv TRANS oder Kombinationen davon in den lexikalischen Items "geben", "erhalten", "nehmen", "geben" und "erhalten" realisiert. Wir verwenden dazu ein binäres Diskriminationsnetz (Feigenbaum, 1963; Hintzman, 1968) wie es von Goldman (1974) in einem Spracherzeugungsprogramm verwendet wurde.

(34)



4.2. Motivation der Forderung SR3

Im folgenden wollen wir etwas systematischer einige Motive diskutieren, die der Forderung SR3 zugrundeliegen.

Paraphrasierung

Betrachten wir das folgende Beispiel aus einem Frage-Antwort-System:

(35) Text: Peter erhielt von Fritz ein Buch.

Frage: Gab Fritz dem Peter ein Buch?

Werden "geben" und "erhalten" in der SR verschieden repräsentiert, dann muss ihr Zusammenhang durch eine Regel ausgedrückt werden, wenn die Frage beantwortet werden soll. Diese Technik wurde von Simmons (1973) verwendet. Angenommen, "geben" werde durch

(36) (GEBEN agens rezipient objekt)

repräsentiert und "erhalten" durch

(37) (ERHALTEN agens ursprung objekt)

dann drückt Simmons den Zusammenhang zwischen (36) und (37) mit der folgenden Paraphrasierungsregel aus:

(38) (GEBEN (AGENS URSPRUNG) (REZIPIENT AGENS) (OBJEKT OBJEKT) ERHALTEN)

d.h. zwischen "geben" und "erhalten" gibt es die folgenden eineindeutigen Abbildungen:

(39) agens-von-geben \longleftrightarrow ursprung-von-erhalten
rezipient-von-geben \longleftrightarrow agens-von-erhalten
objekt-von-geben \longleftrightarrow objekt-von-erhalten

die die Kasusbildungen beim Uebergang von "geben" nach "erhalten" und umgekehrt definieren.

Wenn hingegen "geben" und "erhalten" in der semantischen Repräsentation gleich repräsentiert werden, dann werden solche Paraphrasierungsregeln unnötig. Gegen die Verwendung von solchen Paraphrasierungsregeln sprechen zwei Tatsachen: Erstens wird Paraphrasieren im Sprachgebrauch häufig verwendet und sollte daher effizient implementiert werden. Die identische Repräsentation von Paraphrasen ist effizienter als die Anwendung von zusätzlichen Paraphrasierungsregeln. Zweitens sind für n Verben $\binom{n}{2}$ Paraphrasierungsregeln notwendig, was bei grossem n zu einem enormen Speicherplatzbedarf führt.

Inferenzökonomie

Wir möchten ausdrücken, wer vor und nach einem Akt des "geben" bzw. "erhaltens" der Besitzer des Objekts war. Werden "geben" und "erhalten" gleich repräsentiert, d.h. durch

(40) (GEBEN-ERHALTEN gebender erhaltender objekt)

dann können wir das in etwa durch folgende Inferenzregeln ausdrücken:

(41) Wenn (GEBEN-ERHALTEN gebender erhaltender objekt)_t
dann (HABEN gebender objekt)_{t-1}
und (HABEN erhaltender objekt)_{t+1}

d.h. wenn zur Zeit t ein Akt des GEBEN-ERHALTENS stattfand, dann hatte der "gebende" das Objekt zur Zeit t-1 und der "erhaltende" hat das Objekt zur Zeit t+1.

Wenn hingegen "geben" und "erhalten" in der semantischen Repräsentation erhalten bleiben, dann gibt es zwei Alternativen, um diese Inferenzregeln auszudrücken.

In der ersten Alternative werden für jedes lexikalische Item alle Inferenzregeln definiert. So hätten wir z.B. die folgenden beiden Inferenzregeln:

(42) Wenn (GEBEN agens rezipient objekt)_t
dann (HABEN agens objekt)_{t-1}
und (HABEN rezipient objekt)_{t+1}

(43) Wenn (ERHALTEN agens ursprung objekt)_t
dann (HABEN ursprung objekt)_{t-1}
und (HABEN agens objekt)_{t+1}

In der zweiten Alternative gibt es nur noch eine Inferenzregel (mit dem Namen INF37), aber vor der Anwendung dieser Inferenzregel müssen zuerst die Kasus abgebildet werden:

(44) INF37 Wenn (??? von nach obj)_t
dann (HABEN von obj)_{t-1}
und (HABEN nach obj)_{t+1}

Zur Anwendung der Inferenzregel INF 37 werden die beiden Kasusabbildungen definiert (nach Simmons Paraphrasierungsregeln):

(45) (GEBEN (AGENS VON) (REZIPIENT NACH) (OBJEKT OBJ) INF37)
und (46) (ERHALTEN (AGENS NACH) (URSPRUNG VON) (OBJEKT OBJ) INF 37)

Beide Alternativen haben Nachteile gegenüber der Version (41). Im ersten Fall müssen äusserst viele redundante Inferenzregeln formuliert werden, die viel Speicherplatz benötigen. Im zweiten Fall muss vor jeder Anwendung einer Inferenzregel die Kasus abgebildet werden und die Kasusabbildungen brauchen zusätzlichen Platz. Wenn man bedenkt, wie häufig Inferenzen gemacht werden, dann erweist sich die lexikalische Dekomposition als äusserst

vorteilhaft.

Das Argument der Inferenzökonomie kann noch an einem anderen Beispiel gezeigt werden. Betrachten wir das folgende Beispiel:

(47) Text: Fritz kaufte Brot beim Bäcker.

(48) Frage: Erhielt Fritz Brot vom Bäcker?

Wenn man (47) gleich repräsentiert wie in (49) in der Einleitung, also

- (49) a (GEBEN-ERHALTEN BAECKER FRITZ BROT)
- b (GEBEN-ERHALTEN FRITZ BAECKER GELD)
- c (CAUSE a b)
- d (CAUSE b a)

und die Frage (48) entsprechend als

(50) (GEBEN-ERHALTEN BAECKER FRITZ BROT)?

dann kann die Frage (48) sehr einfach beantwortet werden, da (50) und (49a) genau gleich sind.

Verzichtet man hingegen auf lexikalische Dekomposition, dann muss eine zusätzliche Inferenzregel formuliert werden, damit (48) beantwortet werden kann. Diese Inferenzregel sähe in etwa so aus:

- (51) Wenn (KAUFEN agens ursprung objekt)
- dann (GEBEN ursprung agens objekt)

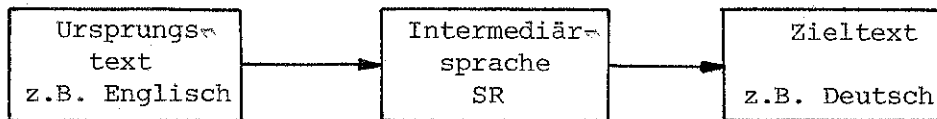
Beim Ansatz der lexikalischen Dekomposition sind bereits einige - nach Schank die für das Sprachverstehen wichtigsten - Inferenzresultate in der semantischen Repräsentation enthalten.

Relative Sprachfreiheit der SR

Das Argument, dass die SR möglichst wenige - oder keine - sprachspezifische Elemente (syntaktische und lexikalische Informationen) enthalten soll, leitet sich aus drei Gesichtspunkten ab.

a) Im Paradigma der maschinellen Sprachübersetzung ist es vorteilhaft, wenn die Intermediärsprache (siehe Abbildung (52)) möglichst keine sprachspezifischen Eigenheiten der Ursprungssprache enthält, die nicht in die Zielsprache übersetzt werden können.

(52)



Je weniger einzelsprachliche Eigenheiten diese Intermediärsprache enthält, umso eher ist sie geeignet, um aus einer beliebigen Sprache in eine beliebige andere Sprache zu übersetzen.

b) In einem integrierten System, das nicht nur verbale, sondern auch optische Informationen aufnimmt und nicht nur verbales, sondern auch physi-

sches Verhalten (Roboter) produziert, ist es von grossem Vorteil, diese Informationen unabhängig von einzelnen Sinnesmodalitäten zu repräsentieren. Eine relative Sprachfreiheit der semantischen Repräsentation ist ein erster Schritt dazu. Allerdings muss hinzugefügt werden, dass gerade dieses Argument auch gegen Schank geführt wird, dessen "Conceptual dependency" Repräsentation immer noch primär linguistisch orientiert ist (Pat Hayes, 1977 a).

c) Der dritte Gesichtspunkt stammt aus der Psycholinguistik, in der viele Befunde dafür sprechen, dass nur die Bedeutung von Sätzen (der konzeptuelle Inhalt) relativ überdauernd behalten wird, der exakte Wortlaut, d.h. syntaktische und lexikalische Informationen hingegen viel weniger genau behalten wird.

4.3. Vertreter der lexikalischen Dekomposition in der AI

Die beiden prominentesten und profiliertesten Vertreter der lexikalischen Dekomposition innerhalb der Künstlichen Intelligenz sind Schank und Wilks. Im folgenden werden die wichtigsten Elemente ihrer Repräsentation charakterisiert und einige Unterschiede zwischen den beiden Ansätzen herausgearbeitet. Dies wird insofern erschwert, als nur Schank seine Theorie relativ ausführlich und vollständig beschrieben hat, während Wilks sein System immer nur beispielhaft vorgestellt hat.

Schanks Conceptual Dependency Theorie

Schank hat sowohl seine Repräsentationstheorie als auch ihre Anwendungen, vor allem in den Systemen MARGIE und SAM in mehreren Arbeiten detailliert präsentiert (Schank, 1972; Schank, 1973; Schank, 1975; Schank und Abelson, 1977). Schanks Ziel ist es, einen Formalismus zur Repräsentation von Bedeutung zu entwickeln, der weitgehend sprachfrei ist, eine kanonische Repräsentation von Bedeutung erlaubt (d.h. der Forderung SR3 Rechnung trägt) und zur Modellierung von Gedächtnis- und Inferenzprozessen geeignet ist. Formal gesehen ist Schanks Conceptual Dependency Theorie eine Dependenzgrammatik im Sinne von Hays (1964).

Die Basiseinheit jeder Conceptual Dependency Struktur ist eine Konzeptualisation, die um eine Handlung (Act), einen Zustand (State) oder eine Zustandsänderung (Statechange) zentriert ist.

Schank geht davon aus, dass jede Handlung in eine kleine Menge primitiver Handlungen zerlegt werden kann. So behauptet er, dass alle Handlungen durch Kombinationen von 11 (oder einigen mehr) primitiven Acts reprä-

sentiert werden können, zu denen unter anderen gehören: ATRANS (transfer of possession, ownership or control), PTRANS (transfer of physical location), PROPEL (application of physical force), INGEST (internalization of an external object into the body of an animal), EXPEL (pushing an object out of the body) usw. (Schank, 1975). Die genaue Anzahl der von Schank verwendeten primitiven Acts ist völlig unwichtig, wenn man bedenkt, dass er fast nur Handlungen von Menschen repräsentiert. Würde man seine Repräsentation auf beliebige Texte und Sachverhalte ausdehnen, würde sich die Zahl der primitiven Acts sicher erhöhen. Wichtig ist aber, dass die Anzahl der Acts trotzdem sehr klein bleiben würde.

Der zweite Baustein von Konzeptualisationen sind die Zustände, wie etwa der Ort eines Objektes oder der mentale Zustand eines Lebewesens. Wie bei den Acts postuliert Schank auch bei den States eine sehr kleine Anzahl notwendiger primitiver States. Aber auch hier ist nicht eindeutig, wieviele zusätzliche States eingeführt werden müssten, wollte man beliebige Sachverhalte repräsentieren.

Als dritten Baustein schliesslich braucht Schank Zustandsänderungen, die für jeden Zustand definiert werden. So wird etwa "sterben" als Veränderung des Gesundheitszustandes zum minimalen Wert hin repräsentiert. In einigen Fällen lassen sich nun primitive Acts auf eine unspezifische Handlung plus eine Zustandsänderung reduzieren. So könnte PTRANS repräsentiert werden als eine unspezifische Handlung (DO), die eine Veränderung im Ort des bewegten Objektes verursacht. Diese Reduzierung ist aber nicht mit allen Acts sinnvoll (z.B. mit ATTEND, das umschrieben wird mit "focusing a sense organ toward a stimulus") und man würde mit einem solchen Schritt nur eine kleinere Anzahl von Primitiva gegen eine erhöhte Komplexität der Repräsentationsstruktur eintauschen.

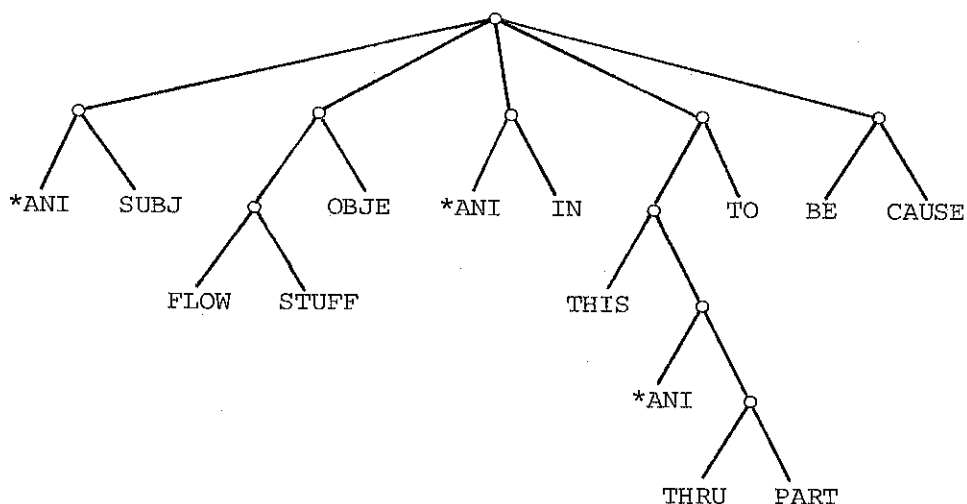
Wilks' Präferenzsemantik

Wilks' System der Präferenzsemantik (Wilks, 1973a, 1973b, 1975a) ist aus mehreren Gründen schwieriger zu präsentieren als Schanks Conceptual Dependency Theorie. Es liegt bereits daran, dass er sein System nie so systematisch beschrieben hat wie Schank, sondern es nur immer beispielhaft erläutert hat. Einige Gründe dazu liegen aber im prinzipiell verschiedenen Ansatz von Schank und Wilks. Schank geht es darum, eine Theorie der "Bedeutungsrepräsentation" zu entwickeln, während Wilks' System Teil eines integrierten maschinellen Uebersetzungssystems ist. Schank geht es also mehr um den theoretischen Aspekt seiner Arbeit, während bei Wilks der technologische Aspekt vorherrscht. Zweitens vertritt Schank in seinen Ar-

beiten die Ansicht, dass die Bedeutung von Sätzen auf einer Stufe repräsentiert wird, der Stufe der Conceptual Dependency (allerdings mit gewissen Einschränkungen, Schank & Abelson, 1977), während Wilks System aus einer Hierarchie von Repräsentationsstufen besteht, von denen wir uns im folgenden nur mit einer beschäftigen werden. Drittens ist Wilks aus theoretischen Gründen, die wir im folgenden noch diskutieren werden, viel flexibler in der Form seiner Repräsentation, und in der Auswahl seiner lexikalischen Primitiva.

Im Abschnitt über Parsen werden wir uns systematisch mit dem ganzen System von Wilks auseinandersetzen. In diesem Zusammenhang aber wollen wir nur den Teil seines Systems betrachten, der in bezug auf die semantische Repräsentation am wichtigsten ist. Der Grundbaustein in Wilks' semantischer Repräsentation ist die 'Formel', die aus einer kleinen Menge von Einheiten, den semantischen Primitiva aufgebaut ist. Die Formeln werden unter anderem dazu verwendet, die Bedeutung von Wörtern zu repräsentieren. So wird zum Beispiel die Bedeutung des Wortes "trinken" durch (53) repräsentiert.

(53)



Die Formel (die von rechts nach links zu lesen ist) besagt ungefähr: "Trinken" ist eine Aktion, vorzugsweise von einem belebten Subjekt getan (*ANI SUBJ), auf Flüssigkeiten angewandt ((FLOW STUFF) OBJE), die verursacht (CAUSE BE), dass die Flüssigkeit im belebten Ding ist (*ANI IN), durch eine bestimmte Oeffnung des Lebewesens - dem Mund - hindurch (THIS (*ANI (THRU PART))) TO).

Die Terminale dieser Formel sind semantische Primitiva, deren es etwa 80 gibt. Sie können (der Uebersicht halber) in 5 Klassen eingeteilt werden:

- a) Entitäten: MAN (menschliches Wesen), STUFF (Substanzen), ACT (Aktionen), FOLK (menschliche Gruppen) usw.
- b) Aktionen: PICK (wählen), BE (existieren), CAUSE (verursachen) usw.
- c) Typ-Indikatoren: KIND (irgendeine Qualität), HOW (irgendeine Aktion) usw.
- d) Sorten: CONT (Behälter, GOOD (moralisch akzeptierbar) usw.
- e) Kasus: TO (Richtung, SOUR (Ursprung), GOAL (Ziel, Ende) LOCA (Ort), SUBJ (Aktor) usw.

Zusätzlich werden Elemente verwendet, die für Klassen von Primitiva stehen, z.B. steht *ANI für die Primitiva MAN, BEAST und FOLK.

Einer der entscheidenden Unterschiede zwischen Schank und Wilks liegt nun im Begriff der Präferenz. Bei Schank wird etwa für den Kasus Actor die Restriktion +lebend (im Sinne von Katz und Fodor, 1963) verlangt, d.h. ein Satz wie

(54) Mein Auto trinkt Benzin.

kann nicht repräsentiert werden, da Auto die Restriktion +lebend verletzt. Im Gegensatz dazu besteht in Wilks System nur eine Präferenz für ein lebendes Subjekt, d.h. bei zwei verschiedenen Lesungen wird die Lesung mit +lebend vorgezogen. Wenn, wie in (54), kein lebendes Subjekt vorhanden ist, dann wird diese - metaphorische - Lesung akzeptiert.

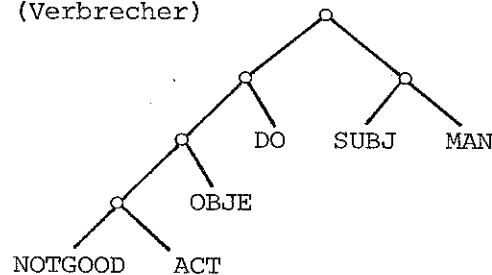
An einem Beispiel wollen wir zeigen, wie dieser Präferenzmechanismus funktioniert. Im Satz

(55) "The big policeman interrogated the crook."

ist das Wort "crook" mehrdeutig: es kann entweder Verbrecher oder Hirtenstab bedeuten. Die beiden Lesungen von "crook" werden repräsentiert durch

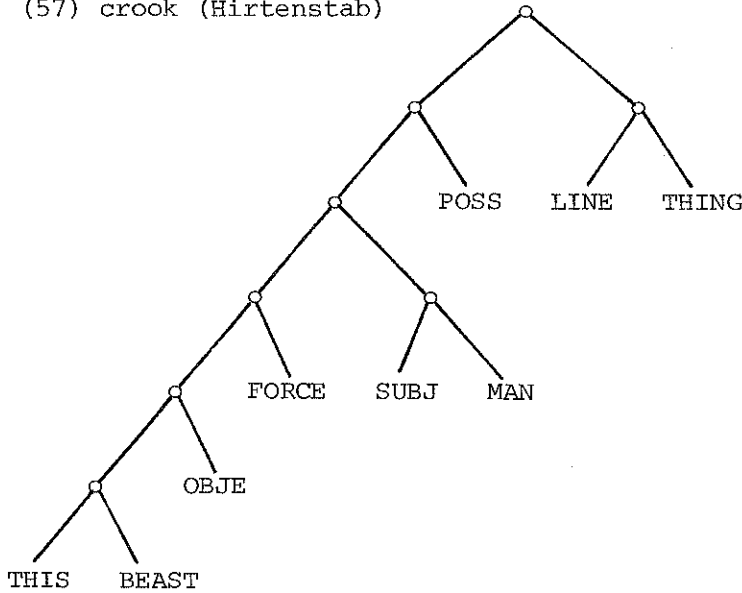
(56)

(56) crook (Verbrecher)



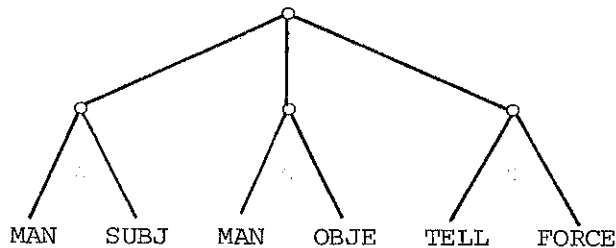
d.h. ein Mensch, der schlechte Aktionen tut, und (57),

(57) crook (Hirtenstab)



d.h. ein langes, gerades Objekt, das von einem Menschen besessen wird, der eine bestimmte Art von Tieren überwacht. Das Element ganz rechts in der Formel, der Kopf, gibt die grobe Kategorie der Wortbedeutung an, also im Falle von Verbrecher MAN und im Falle von Hirtenstab THING. Das Verb 'interrogate', dessen semantische Repräsentation (58)

(58)



ist, zieht als Objekt eine Formel mit MAN-Kopf vor. Die Formel (56) erfüllt diese Präferenz, während es die Formel (57) nicht tut. Daher wird auch die Lesung (56) der Lesung (57) vorgezogen.

4.4. Eigenschaften der lexikalischen Dekomposition

Wir haben bereits in 4.1. gezeigt, dass man dem Ansatz der lexikalischen Dekomposition verschieden weit folgen kann. Der Ansatz ist einerseits begrenzt durch die Position der vollständigen Dekomposition, wie sie von Schank und Wilks vertreten wird, und andererseits durch die Position der vollständigen Individualität, in der nur lexikalische Items in der semantischen Repräsentation vorkommen (siehe z.B. Simmons, 1973). Es gilt

nun, zwischen diesen beiden extremen Positionen die geeignetste Stufe der lexikalischen Dekomposition zu finden.

Gegen die Position der vollständigen Individualität sprechen vor allem die bereits erwähnten Motive der Forderung SR3.

Gegen die extreme Position der lexikalischen Dekomposition sprechen ebenfalls einige Gründe. Aus der Sicht der Inferenzprozesse ist es vor allem die Tatsache, dass dort gewisse Inferenzen nur durch sehr komplexe Muster ausgelöst werden können. Dies ist unter anderem der Fall bei wortspezifischen Inferenzen von Wörtern, deren semantische Repräsentation sehr komplex ist (z.B. "schwitzen", das in Schanks System durch eine Konzeptualisation mit EXPEL repräsentiert wird.) Durch die Einführung von nichtprimitiven Konzepten in die semantische Repräsentation können solche Inferenzen viel leichter ausgelöst werden.

Aber auch aus der Sicht der Paraphrasierung sind gewisse Vorbehalte gegen die extreme Position der lexikalischen Dekomposition anzubringen. Das Problem tritt dann auf, wenn man fordert, dass alle Paraphrasierungen eines Satzes die gleiche semantische Repräsentation haben sollen, oder in anderen Worten, eine kanonische Repräsentation für alle Paraphrasen fordert. Es ist aus der mathematischen Linguistik bekannt, dass für gewisse formale Sprachen (die eingeschränkter sind, als für die SR notwendig ist) keine Funktion existiert, die alle kanonischen Formen eines Ausdruckes berechnet (siehe z.B. das Wortproblem für Semi-Thue Systeme, Gross & Lentin, 1971, Davis 1958)

Abgesehen von solchen formalen Gründen ist es unklar, wie weit man der Forderung nach kanonischer Repräsentation von Paraphrasen folgen soll, da nicht eindeutig ist, welchen Stellenwert man der Paraphrasierungsfähigkeit innerhalb des Sprachverstehens zuordnen soll.

5. KASUSSYSTEME

In diesem Abschnitt wird auf eine vierte und letzte Forderung an die semantische Repräsentation eingegangen:

SR4: DIE SR SOLL DIE FUNKTIONALE STRUKTUR VON EREIGNISSEN AUFZEIGEN.

Damit ist gemeint, dass in der SR die Funktion von Objekten in einer Aktion so repräsentiert werden können, dass Fragen wie "Wer hat die Aktion durchgeführt?", "Was wurde durch die Aktion betroffen?" oder "Womit wurde die Aktion durchgeführt?" möglichst einfach beantwortet werden können. Diese Funktionen sind weder in der Oberflächen- noch Tiefenstruktur von Sätzen direkt reflektiert. Dass dies für die Tiefenstruktur nicht der Fall ist, illustrieren die Beispiele (60) - (62), in denen die Funktion der Objekte (ihre semantische Rolle) identisch ist,

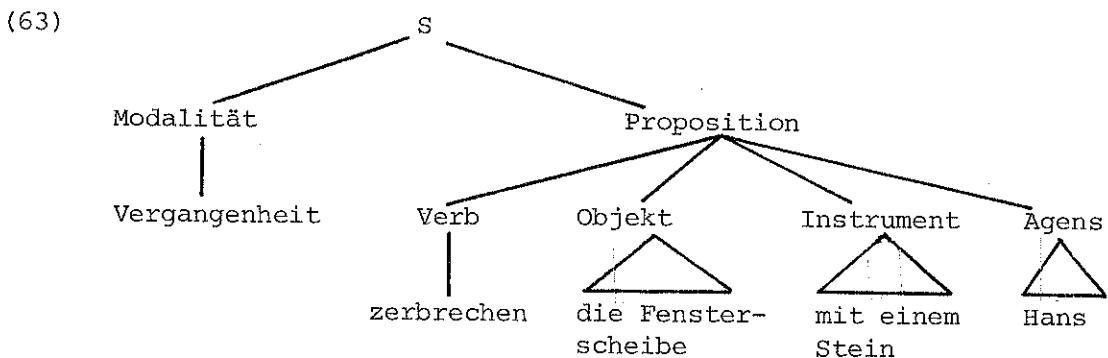
(60) Hans zerbrach die Fensterscheibe mit einem Stein.

(61) Der Stein zerbrach die Fensterscheibe.

(62) Die Fensterscheibe zerbrach.

ihre syntaktische Rolle sich jedoch ändert. Damit folgt trivial, dass die semantische Rolle auch in der Oberfläche nicht direkt reflektiert sein kann.

Fillmores (1968, 1971) Kasusgrammatik ist ein Ansatz zur Repräsentation genau dieser semantischen Rollen. So wird der Satz (60) bei Fillmore durch (63) repräsentiert,



während in der Repräsentation von (61) das Agens und in der Repräsentation von (62) das Agens und das Instrument fehlen.

Auf die Frage nach den Motiven für die Forderung SR4 findet man in der Literatur verschiedene, zum Teil sogar inkonsistente Antworten. In einer Analyse der linguistischen Literatur und der Literatur der Künstlichen Intelligenz haben Rosner und Somers (1980) gezeigt, dass dies vor allem darauf zurückzuführen ist, dass der Begriff des Kasusrahmens in verschiedenen Bedeutungen gebraucht wird.

So definiert Fillmore einen Kasusrahmen für jedes Verb, in dem festgehalten ist, welche Rollen (Kasus) für dieses Verb spezifiziert werden können. Der Kasusrahmen (64) des Verbs 'zerbrechen' besagt, dass mit dem Verb 'zerbrechen'

(64) [_____ Objekt (Instrument) (Agens)]

ein Objekt obligatorisch angegeben werden muss, während ein Instrument und ein Agens fakultativ angegeben werden können. Der Kasusrahmen (64) definiert daher alle möglichen Kombinationen von Kasusrealisierungen (65)

(65) [_____ Objekt]

[_____ Objekt + Instrument]

[_____ Objekt + Agens]

[_____ Objekt + Instrument + Agens]

für das Verb 'zerbrechen'. Die spezifizierten Kasus werden eineindeutig auf Nominalphrasen in der Tiefenstruktur abgebildet. Daraus lassen sich verschiedene linguistische Phänomene erklären. So ist (66) ungrammatisch,

(66) *Hans zerbrach mit einem Hammer.

weil das obligatorische Objekt fehlt. Ebenso sind (67) und (68) ungrammatisch,

(67) *Hans und der Hammer zerbrachen die Scheibe mit einem Hammer.

(68) *Der Hammer zerbrach die Scheibe mit dem Hammer.

weil die eineindeutige Abbildung verletzt wird: In (67) werden zwei verschiedene Kasus auf eine NP der Tiefenstruktur und in (68) ein Kasus auf zwei verschiedene NP in der Tiefenstruktur abgebildet.

Mit der Subjektauswahlregel wird festgelegt, welcher Kasus Subjekt in der Tiefenstruktur wird (69):

(69) "If there is an Agent, it becomes subject; otherwise if there is an Instrument it becomes subject; otherwise the subject is the object." (Fillmore, 1968, S. 33).

Mit dieser Regel lässt sich daher zwischen den Sätzen (60), (61) und (62) ein einfacher Zusammenhang formulieren.

Mit der Kasusgrammatik lässt sich auch die semantische Identität verschiedener Verben einfach erfassen. So sind die Verben 'geben' und 'erhalten' semantisch identisch, unterscheiden sich aber durch verschiedene Kasusrealisierungen (und bilden somit eine Ausnahme zur Standardkasusrealisierung (69)). Während in 'geben' das Agens Subjekt wird, ist es in 'erhalten' der Dativ. Ähnlich liegt auch der Unterschied zwischen 'sehen' und 'anschauen' darin, dass das Subjekt in 'sehen' Dativ, in 'anschauen' hingegen Agens ist.

Diese Argumente beruhen kritisch auf der Annahme, dass eine kleine, universelle (d.h. von den einzelnen Verben unabhängige) Menge von Kasus genügen, um alle möglichen Rollen von Entitäten in einem Verb zu beschreiben. So postuliert denn auch Fillmore, dass dazu eine beschränkte Menge von Kasus, wie Agens, Instrument, Dativ, Faktitiv, Objekt und Lokativ genügen sollte.

Neben den möglichen Kasuskombinationen in (64) wird auch beschränkt, welche Entitäten einen bestimmten Kasus füllen können. Diese Beschränkungen werden mit Katz & Fodor-Markierungen festgelegt. So muss etwa das Agens die Markierung +belebt haben und damit kann ein Satz wie (70)

(70) Die Freiheit zerbrach die Fensterscheibe.

als semantisch unakzeptabel klassiert werden. Aus der beschränkten Anzahl von Kasus folgt aber, dass die semantischen Beschränkungen sich nur auf sehr allgemeine Eigenschaften beziehen können.

Aus diesen wenigen Beispielen zeigt sich, dass mit der Kasusgrammatik verschiedene linguistische Verallgemeinerungen einfach erfasst werden können. Als solche sind sie aber nicht nur für die Linguistik interessant, indem sie eine sparsame Beschreibung linguistischer Phänomene ermöglichen. Wie Charniak (1976 b) gezeigt hat, ist es eine plausible Annahme, dass linguistische Verallgemeinerungen in einer guten, d.h. sparsamen, effizienten und leistungsfähigen Theorie des Sprachverstehens mindestens in einer analogen Form auch enthalten sein müssen.

Fillmores Kasusidee ist in der Künstlichen Intelligenz rasch sehr populär geworden und in einer Reihe von SR-Formalisten verwendet worden (Norman und Rumelhart, 1975; Schank, 1972;1975; Simmons, 1973; Wilks, 1975a), vor allem in den semantischen Netzwerken, die die Verwendung der Kasusnotation geradezu prädestinieren. Wir werden aber im folgenden zeigen, dass mit der Uebernahme der gleichen Notation nicht auch die gleichen Motive übernommen wurden.

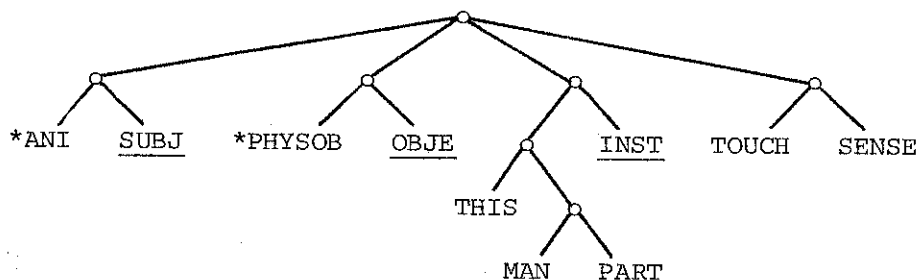
In der SR geht es nicht so sehr darum, Sätze zu repräsentieren, als die von den Sätzen beschriebenen Ereignisse. Daher geht es in der SR auch nicht so sehr darum, den Kasusrahmen von Verben zu repräsentieren, als den Kasusrahmen von Ereignissen. Während nun in einem Satz ein Kasus fehlen kann, ist dies in einem Ereignis nicht möglich. Ein Ereignis des Oeffnens ohne ein Agens kann es nicht geben, das Agens kann höchstens unbekannt sein. Dies führt aber dazu, dass eine Subjektauswahlregel wie (69) nicht direkt auf die SR angewandt werden kann. Im weitern gibt es auch keine prinzipiellen Argumente gegen die Einführung beliebiger Kasus in der SR.

Im Gegenteil, die Leistungsfähigkeit einer Repräsentation kann erhöht werden, wenn ein häufig verwendeter Teilaspekt eines Sachverhaltes durch einen Kasus spezifiziert wird. Aus diesen Argumenten ergibt sich aber, dass die Motive für die Verwendung der Kasusnotation in der SR nicht die gleichen sein können, die Fillmores Kasusgrammatik zugrundeliegen. Damit lässt sich auch die Abbildung von Kasus der SR auf Teile der Oberflächenstruktur von Sätzen nicht mehr in der gleichen einfachen Art wie bei Fillmore definieren.

So müssen denn in Schanks Conceptual Dependency Theorie die Abbildungen von Kasus der SR auf Oberflächenkasus für jedes Verb einzeln definiert werden (Riesbeck, 1974; Goldman, 1974). Diese Abbildungsregeln sind nicht aus der SR ableitbar und müssen daher sowohl im Parser als auch im Generator separat enthalten sein.

Im System von Wilks, das erst im Kapitel über Parsen eingehend diskutiert wird, finden sich kasusgrammatische Elemente auf zwei Ebenen. In der semantischen Repräsentation (den Templates) findet man nur die beiden Kasus Agens und Objekt. Interessanter ist das Kasussystem in seiner Repräsentation von Wortbedeutungen, das im Gegensatz zu Schank viel stärker oberflächenorientiert ist. So beziehen sich die Kasus SUBJ, OBJ und INST in seiner Repräsentation von "grasp" (im Sinne von "aufheben eines physischen Gegenstandes") auf das

(71)



Oberflächenverb "grasp" und die syntaktischen Konstruktionen, die zum Ausdruck von Subjekt, Objekt und Instrument gebraucht werden und nicht auf das in der semantischen Repräsentation enthaltene Primitiv SENSE. Durch die Kombination von syntaktisch und semantisch orientierten Elementen in der Repräsentation ist es daher möglich, syntaktisch orientierte Parserstrategien direkt aus der SR abzuleiten. Aber auch in der Wilksschen Repräsentation können die Kasusabbildungen nicht in der allgemeinen und einfachen Form wie bei Fillmore erfasst werden. Es fragt sich daher aus dieser Sicht, ob nicht damit bereits unsere frühere Behauptung widerlegt worden ist, dass die in Fillmores Ansatz erfassten linguistischen Generalisierungen für eine

gute Theorie des Sprachverstehens wichtig sein würden. Es scheint ja, dass die Anforderungen der SR und die Annahmen, die der Fillmoreschen Kasusgrammatik zugrundeliegen, miteinander unvereinbar sind. Die Frage kann aber nicht definitiv entschieden werden und es scheint nicht ausgeschlossen, dass sich die beiden Ansätze miteinander kombinieren lassen. So schlägt denn auch Marcus (1980) vor, dass beim Parsen eine Fillmore-ähnliche Repräsentation als Zwischenstufe zu einer tieferen SR konstruiert wird. Inwieweit dieser Zweistufenansatz aber den Ansätzen von Schank und Wilks überlegen ist, wird sich erst in der Zukunft zeigen.

III. P A R S E N

Man kann beim Prozess des Sprachverstehens zwei Teilprozesse unterscheiden, das Parsen und das Integrieren. Beim Parsen wird die den Sätzen zugrundeliegende Struktur aufgefunden und in der Integration werden diese dann in einen weiteren Kontext (zum Beispiel den bis dahin gelesenen Text) integriert. In diesem Kapitel wollen wir über den Prozess des Parsens sprechen.

Unter Parsen verstehen wir die Auffindung der zugrundeliegenden Struktur von Sätzen (der semantischen Repräsentation) unter Ausnützung aller im Satze selbst enthaltenen Informationen als auch der im linguistischen und nichtlinguistischen Kontext enthaltenen Informationen. Bevor wir mit der Diskussion einzelner Parsermodelle beginnen, wollen wir einige Unterscheidungen einführen, mit denen sich die Parserstrategien charakterisieren lassen.

a) Syntaktisches versus semantisches Parsen

Die Parserstrategien können sowohl auf syntaktischem Wissen als auch auf semantischem Wissen basieren und es lässt sich zeigen, dass beide Komponenten in einem idealen Parser gebraucht werden. In den frühen Parsermodellen wurde meistens auf eine der beiden Komponenten das Hauptgewicht gelegt und die andere Komponente nur dann, wenn überhaupt, verwendet, wenn die erste nicht zum Erfolg führte. Nur in wenigen Ansätzen ist versucht worden, beide Komponenten gleichzeitig und gleichwertig zu berücksichtigen.

b) Hypothesengesteuertes versus stimulusgesteuertes Parsen

Hypothesengesteuerte (top-down) Parser beginnen mit einer erwarteten Satzstruktur, die mit Hilfe der Grammatikregeln weiter aufgegliedert wird, bis die Wortebene erreicht ist, die dann mit dem Eingabesatz verglichen wird. Stimmen Hypothese und Eingabesatz nicht überein, wird die Hypothese geändert. Beim stimulusgesteuerten (bottom-up) Parsen andererseits wird auf der Wortebene begonnen. Die Wörter des Satzes werden in immer grössere Konstituenten zusammengefasst, bis die Satzebene erreicht ist.

c) Depth-first versus breadth-first Parsen

Im Falle mehrerer möglicher Interpretationen wird bei der Depth-first-Strategie (Nilsson, 1971) eine einzige Alternative weiterverfolgt bis sie zum Erfolg führt oder in einer Sackgasse endet. Im letzteren Falle wird

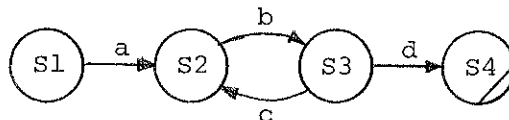
zum letzten Wahlpunkt mit mehreren Alternativen zurückgesprungen und eine neue Alternative ausprobiert (sogenanntes "Backtracking"). Bei der breadth-first-Strategie werden alle Interpretationsmöglichkeiten gleichzeitig verfolgt, indem eine parallele Verarbeitung simuliert wird.

1. WOODS' AUGMENTED TRANSITION NETWORK GRAMMATIK

Eines der meistverwendeten Parsermodelle der Künstlichen Intelligenz geht zurück auf eine Arbeit von Woods (1970). Darin wird ein Formalismus vorgestellt, der es erlaubt, grammatikalische Regeln einfach zu repräsentieren und der von einem Interpreter dazu verwendet werden kann, Sätze zu parsen. Woods' Augmented Transition Network (ATN) Grammatik ist eine Erweiterung der aus der Automatentheorie stammenden Zustandsübergangsgraphen (siehe dazu zum Beispiel Böhling & Indermark, 1969; Gross & Lentin, 1971).

In diesen Graphen repräsentieren die Knoten die Zustände eines Parsers, der einen Satz parst und die Wörter auf den gerichteten Kanten geben an, welches Wort als nächstes im Satz folgen muss, damit der Parser in den nächsten Zustand übergehen kann. Ein Satz wird dann vom Parser akzeptiert, wenn es einen entsprechenden Pfad durch den Zustandsübergangsgraphen vom Anfangszustand zu einem der Endzustände gibt. So akzeptiert zum Beispiel der Zustandsübergangsgraph in (1) Ketten der Form $abcabc\dots bcabcd$. S1 ist dabei der Anfangszustand, S4 der (mit einem Schrägstrich gekennzeichnete) Endzustand.

(1)



Wood's ATN-Grammatik enthält drei Erweiterungen dieser Zustandsübergangsgraphen. Damit unbeschränkt viele Einbettungen geparkt werden können, muss ein Rekursionsmechanismus eingebaut werden. Dies wird ermöglicht, indem in sogenannten PUSH-Kanten andere Teilnetze wie Subroutinen (möglicherweise rekursiv) aufgerufen werden können. Wenn auch die Notwendigkeit unbeschränkter Rekursion zum Verstehen natürlicher Sprache umstritten ist, so ermöglicht dieser Mechanismus andererseits auch, dass zum Beispiel die syntaktischen Regeln von Nominalphrasen nur einmal kodiert werden müssen.

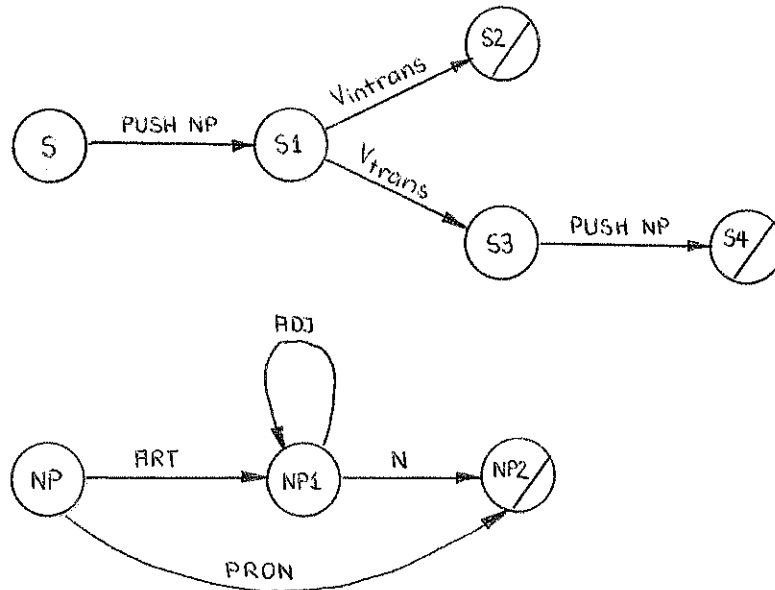
Als zweite Erweiterung können in ATN-Grammatiken beliebige Bedingungen angegeben werden, die erfüllt sein müssen, damit der Kante gefolgt werden darf und drittens können mit jedem Uebergang beliebige strukturbildende Aktionen verbunden sein, die die syntaktische Repräsentation des geparkten Eingabesatzes bilden. Die Uebergangsnetzwerke, die in der ursprünglichen Form endliche Sprachen erkennen können (siehe dazu Chomsky, 1963), erreichen durch die Einführung der PUSH-Kanten die Mächtigkeit von kon-

text-freien Grammatiken, durch die Einführung beliebiger Bedingungen die Mächtigkeit von kontextsensitiven Grammatiken und durch die zusätzliche Einführung beliebiger Aktionen die Mächtigkeit von Transformationsgrammatiken (Typ 0 Grammatiken).

Im folgenden soll die Funktionsweise eines ATN-Parsers vereinfacht illustriert werden. Beispiel (2) zeigt eine einfache Phrasenstrukturgrammatik, Abbildung (3) die entsprechende ATN-Grammatik.

- (2) S \rightarrow NP + VP
- NP \rightarrow PRON | ART + MN
- MN \rightarrow N | ADJ + MN
- VP \rightarrow Vintrans | Vtrans + NP

(3)



In der abgebildeten ATN-Grammatik bezeichnen S und NP Anfangszustände, die mit einem Schrägstrich gekennzeichneten Knoten S2, S4 und NP2 Endzustände. Die mit "Push NP" bezeichnete Kante besagt, dass an dieser Stelle in das ATN-Netz mit dem Anfangszustand NP übergegangen werden soll. Wird in diesem NP-Netz ein Endzustand erreicht (NP2), dann darf im S-Netz zum nächsten Zustand übergegangen werden. Die übrigen Etiketten auf den Kanten geben an, zu welcher Wortklasse das laufende Wort gehören muss, damit der Kante gefolgt werden darf.

Der ATN-Parser beginnt nun im Anfangszustand des ATN-Netzes und folgt den Kanten des Graphen, bis ein Endzustand erreicht wird. Ist dies der Fall und bleiben keine nichtanalytierten Wörter im Eingabesatz übrig, dann ist der Parsingprozess erfolgreich, sonst nicht. Die Tabelle (5) zeigt die Schritte, die beim Parsen des Satzes (4) gemacht werden.

(4) Er schreibt ein Buch.

(5)

Zustand	Bedingung	aktuelles Wort	Bedingung erfüllt
S	Push NP	ER	
NP	ART	ER	nein
NP	PRON	ER	ja
NP2	-	SCHREIBT	
S1	Vintrans	SCHREIBT	nein
S1	Vtrans	SCHREIBT	ja
S3	Push NP	EIN	
NP	ART	EIN	ja
NP1	ADJ	BUCH	nein
NP1	N	BUCH	ja
NP2	-	-	
S4	-	-	

Ein interessanter Fall ist die Analyse des Satzes (6), deren Schritte in (7) gezeigt werden.

(6) Die Grossen gruben ein Loch.

(7)

Zustand	Bedingung	aktuelles Wort	Bedingung erfüllt
S	Push NP	DIE	
NP	ART	DIE	ja
NP1	ADJ	GROSSEN	ja
NP1	ADJ	GRUBEN	nein
NP1	N	GRUBEN	ja
NP2	-	EIN	
S1	Vintrans	EIN	nein
S1	Vtrans	EIN (*)	nein
NP1	N	GROSSEN	ja
NP2	-	GRUBEN	
S1	Vintrans	GRUBEN	nein
S1	Vtrans	GRUBEN	ja
S3	Push NP	EIN	
NP	ART	EIN	ja
NP1	ADJ	LOCH	nein
NP1	N	LOCH	ja
NP2	-	-	
S4	-	-	

An der mit einem Stern gekennzeichneten Stelle ist der Parser am Ende einer Sackgasse angelangt, da von S1 aus keine mit dem aktuellen Wort vereinbaren Kanten weiterführen. Der Parser geht dann zum letzten Zustandsknoten zurück, wo er zwischen mehreren Kanten wählen konnte und folgt von dort aus einem neuen Pfad. Dazu müssen die seit diesem Zustand gemachten Aktionen rückgängig gemacht werden und insbesondere auch das aktuelle Wort zurückgesetzt werden. Diesen Prozess des Wiederherstellens eines alten Zustandes nennt man "Backtracking" oder "Backup".

Diskussion

In diesem Abschnitt wird jedes Parsermodell mit den eingangs aufgeführten Unterscheidungen und mit den übrigen Parsern verglichen.

a) syntaktisches versus semantisches Parsen

Der ATN-Parser ist ein typischer syntaktischer Parser und die verwendeten semantischen Informationen sind in fast allen Anwendungen dieses Modells vernachlässigbar klein geblieben. Eine Ausnahme dazu ist der Parser von Burton (1976), in dem die syntaktischen Kategorien nach semantischen Gesichtspunkten noch weiter aufgegliedert werden.

b) Hypothesengesteuertes versus stimulusgesteuertes Parsen

Die rein hypothesengesteuerte (top-down) Strategie der ATN-Grammatik führt zu einer Reihe miteinander verwandter Probleme:

Agrammatische Konstruktionen: Normalerweise können mit einer ATN-Grammatik keine agrammatischen Konstruktionen geparkt werden. Wenn man die ATN-Grammatik so aufweicht, dass auch leicht agrammatische Konstruktionen zugelassen sind, dann müssen diese zuletzt ausprobiert werden, da der Parser sonst falschen Pfaden für korrekte Sätze folgen könnte. Das heisst mit anderen Worten, dass ein leicht agrammatischer Satz erst geparkt werden kann, wenn alle möglichen grammatischen Konstruktionen ausprobiert worden sind.

Interdependenz syntaktischer Konstruktionen: Die Repräsentation und Verwendung kontext-sensitiver Regeln ist in der ATN-Grammatik nicht immer effizient, wie an einem Beispiel gezeigt werden soll. Im Deutschen hängt die Anwesenheit gewisser Verbkonstruktionen wie "geschlagen worden" von der Anwesenheit eines Hilfsverbs ab. Es wäre daher effizienter, wenn die Teilnetze zum Parsen dieser Konstruktionen nur dann vorhanden wären, wenn bereits ein Hilfsverb aufgetaucht ist, da sonst die Anfangskanten dieser Teilnetze häufig abgeprüft werden. Eine solche stimulusgesteuerte, dynamische Veränderung ist aber in den statischen ATN-Netzen nicht möglich.

Verarbeiten spezifischer syntaktischer Konstruktionen: Konstruktionen wie "x und y" oder "entweder x oder y" können an sehr vielen, verschiedenen Stellen in einem Satz auftauchen. In einer reinen ATN-Grammatik muss an all diesen Stellen explizit die Erwartung für solche Konstruktionen eingebaut werden, wodurch die Effizienz des ATN-Parsers und die Einfachheit des ATN-Netzes stark leidet. Solche Konstruktionen werden denn auch typischerweise von externen, stimulusgesteuerten Mechanismen verarbeitet wie sie später noch diskutiert werden.

Alle drei erwähnten Probleme der rein hypothesengesteuerten Strategie legen nahe eine gemischte hypothesengesteuerte und stimulusgesteuerte Strategie zu verwenden.

a) Depth-first- versus breadth-first-Strategie

Wie im Beispiel (3) gezeigt wurde, probiert der ATN-Parser eine Alternative nach der andern aus und falls er an das Ende einer Sackgasse gelangt ist, springt er zum letzten Wahlpunkt zurück und probiert eine neue Alternative aus. Diese Depth-first-Strategie ist ineffizient und es stellt sich die Frage, wie das häufige Backtracking vermieden oder zumindest seine Häufigkeit vermieden werden kann. Es bieten sich dabei verschiedene Lösungen an:

Vermindern der Häufigkeit von Backups: Zum Vermindern der Häufigkeit von Backups können die von einem Zustand aus führenden alternativen Kanten nach einem Kriterium geordnet und in dieser Reihe ausprobiert werden. So können die Kanten so angeordnet werden, dass die häufigsten syntaktischen Konstruktionen zuerst probiert werden. Man kann sie auch nach der Leichtigkeit ihrer Falsifizierbarkeit ordnen. Dabei werden jene syntaktischen Konstruktionen zuerst probiert, die am leichtesten falsifiziert werden können: So wird die Präpositionalnominalphrase (PNP) vor der Nominalphrase (NP) probiert, da es zur Falsifikation der PNP-Hypothese schon genügt, wenn das nächste Wort keine Präposition ist, während eine NP mit vielen verschiedenen Wortklassen beginnen kann. Mit dieser Ordnung wird die mittlere Länge der Backups verkürzt.

Gezieltes Backtracking: Man kann am Ende einer Sackgasse versuchen, die Ursache des Misserfolgs zu diagnostizieren und aufgrund dieser Diagnose gezielt an einem andern Ort im ATN-Netz weiterfahren. So gelangt der Parser im Beispiel (8) nach S2 in der Grammatik in (3), weil er hier fälschlicherweise "singen" als

(8) Peter singt ein Lied.

intransitives Verb klassiert hat. Da "singen" aber auch transitiv sein

kann, wird nun direkt nach S3 gesprungen und dabei die Verbeigenschaft transitiv/intransitiv geändert. Die Methode des gezielten Backtrackings ist von Winograd (1972) verwendet worden. Es hat sich aber gezeigt, dass diese Methode sehr kompliziert und undurchsichtig werden kann.

Breadth-first-Strategie: Man kann den ATN-Interpreter leicht ändern so, dass er nach der breadth-first Methode arbeitet. Man gewinnt dabei aber nicht an Effizienz, da von allen erzeugten Interpretationen doch nur eine richtig ist.

Aus der bisherigen Diskussion ist zu erkennen, dass das Problem des Backtrackings in ATN-Netzen nicht zufriedenstellend lösbar ist.

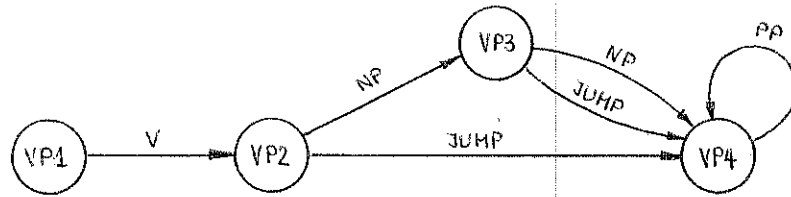
Psychologische Bedeutung der ATN-Grammatik

In den Sechzigerjahren wurde in der Psycholinguistik versucht, die Unterschiede in der perzeptuellen Komplexität von Sätzen auf die Ableitungskomplexität in Chomskys Theorie zurückzuführen (Johnson, 1965; Savin und Perchonok, 1965). Auch wenn einige der ersten Befunde für diese Theorie sprachen, ist dieses Unterfangen im grossen und ganzen doch gescheitert, und Fodor und Garrett (1966) schlossen schliesslich, dass die psychologische Evidenz für die Theorie der Ableitungskomplexität zu schwach war, als dass diese Theorie beibehalten werden sollte. Andererseits argumentiert Kaplan (1972), dass die ATN-Grammatik sich eigne, um Unterschiede in der perzeptuellen Komplexität von Sätzen zu erklären. Aus der bisherigen Diskussion geht aber nicht hervor, dass es sich dabei nicht um eine Reinkarnation der Theorie der Ableitungskomplexität handeln kann. Der Grund dazu liegt in einem wichtigen Unterschied zwischen der ATN-Grammatik und einer simplen Anwendung von Grammatikregeln, wie sie sowohl in den frühen psychologischen Theorien als auch in den frühen syntaktischen Analyseprogrammen (Matthews, 1961; Petrick, 1965) verwendet wurden. Der Unterschied liegt in der sogenannten Faktorisierung von Grammatikregeln in der ATN-Grammatik. Dabei werden grammatische Regeln, die teilweise übereinstimmen, in der ATN-Grammatik verschmolzen. So werden etwa die Regeln (9)

- (9) $VP \rightarrow V$
- $VP \rightarrow V + NP$
- $VP \rightarrow V + NP + NP$
- $VP \rightarrow V + NP + NP$

in der ATN-Grammatik in die nichtredundante Repräsentation (10) verschmolzen.

(10)



Der Vorteil dieser Repräsentation liegt nicht nur darin, dass gewisse syntaktische Komponenten nur einmal kodiert werden müssen, sondern es kann auch die Anzahl der Backups gegenüber einem simplen Regelanwendungsalgorithmus entscheidend reduziert werden. Dies liegt daran, dass im simplen Regelanwendungsalgorithmus bereits vor dem Parsen des Verbs entschieden werden muss, welche VP-Regel angewendet werden soll. In der ATN-Grammatik hingegen entfallen solche Entscheidungen durch Verschmelzung der Regeln. Die Verschmelzung von Regeln bleibt aber nicht auf einzelne Regeln beschränkt. In der gleichen Weise werden zum Beispiel auch die Regeln für die Aktiv- und Passivform von Sätzen verschmolzen (siehe Woods, 1970). Damit wird aber die Ableitungskomplexität in der ATN-Grammatik nicht mehr reflektiert.

Nach diesen Präliminarien können wir uns der eigentlichen Argumentation von Kaplan zuwenden. Er postuliert, dass die perzeptuelle Komplexität eines Satzes direkt proportional zur Anzahl der beim Parsen gemachten oder versuchten Uebergänge in einer ATN-Grammatik ist. Er kann dies auch in der Tat belegen. So ist etwa der perzeptuelle Komplexitätsunterschied in (11)

(11 a) The editor authors the newspaper hired liked laughed

(11 b) The editor the authors the newspaper hired liked laughed.

darauf zurückzuführen, dass im Verlaufe des Parsens von (11 a) einem viel längeren falschen Pfad (bis 'hired') gefolgt wird, als in (11 b), wo diesem falschen Pfad nur bis zu 'the' der zweiten Nominalphrase gefolgt wird. Nun hat bereits Bever (1970) solche Unterschiede mit einer Reihe von verschiedenen perzeptuellen Strategien erklären können. Mit der ATN-Grammatik ist es aber möglich geworden, die verschiedenen Strategien durch ein einheitliches Prinzip erklären zu können.

Nun haben wir in der Diskussion der ATN-Grammatik geschlossen, dass das Problem des Backtrackings in der ATN-Grammatik nicht befriedigend gelöst sei und andererseits mit Hilfe des Backtracking-Mechanismus die psychologische Relevanz der ATN-Grammatik zeigen wollen. Man könnte daher hier einen Widerspruch zwischen einem technologischen Motiv gegen das Backtracking und einem psychologischen Motiv für das Backtracking vermuten. In der Diskussion des Parsers von Marcus werden wir aber sehen, dass dieser Widerspruch gerade bei Sätzen wie (11) nicht besteht.

2. MARCUS' WAIT-AND-SEE-PARSER

Marcus (1974, 1975, 1979) * hat ein Parsermodell entwickelt, in dem die beschriebenen Probleme der ATN-Grammatik vermieden werden können. Sein Parser beruht auf der fundamentalen Annahme, dass die Syntax einer Sprache strikt deterministisch geparkt werden kann, in dem Sinne, dass alle vom Parser gebildeten Teilstrukturen in der endgültigen Satzstruktur enthalten sind. Dies ist bei der "Guess-and-then-Backup"-Strategie eines ATN-Parsers nicht der Fall, wo beim Backup Teilstrukturen fortgeworfen oder verändert werden müssen. Die Schlüsselidee zu Marcus' Parsermodell ist die, dass bei jedem Wahlpunkt im Parsingprozess eine kleine Anzahl diagnostischer Fragen über nachfolgende Konstituenten darüber entscheiden kann, welche Alternative zum Erfolg führt.

Wenn die Mächtigkeit dieser diagnostischen Fragen dadurch beschränkt wird, dass nur eine beschränkte Anzahl von nachfolgenden Konstituenten überprüft werden kann, dann ergibt sich eine ganz einfache Erklärung sogenannter Sackgassensätze wie (12),

(12) I told the boy the dog bit Sue would help him.

in denen die meisten Leser zuerst zu einer falschen Teillesung gelangen. Wird nämlich die Anzahl der nachfolgenden überprüfbaren Konstituenten entsprechend gewählt, dann können Alternativen in üblichen Sätzen diagnostiziert werden, Alternativen in Sackgassensätzen jedoch nicht. Die Beschränkung der Mächtigkeit der diagnostischen Regeln folgt notwendigerweise aus der Determinismushypothese. Damit wird verhindert, dass in den diagnostischen Regeln ein Backtracking- oder pseudoparalleler Mechanismus implementiert werden kann, der der Determinismushypothese zuwiderlaufen würde. Man bemerke, dass Marcus nicht behauptet, dass sein Parser alle Sätze deterministisch parsen kann. Die deterministische Strategie versagt aber nur in Gartenpfadsätzen, bei denen auch der Mensch den starken Eindruck hat, dass er eine erste Analyse korrigieren musste.

Der Marcus-Parser besteht aus einer grossen Anzahl von Produktionsregeln, die auf zwei Ebenen arbeiten. Die sogenannten "Group level"-Regeln suchen den Eingabesatz ab und versuchen lokale syntaktische Strukturen (wie zum Beispiel Nominalphrasen) zu bilden, die dann in einem Buffer ab-

* Eine ausführliche Beschreibung des Marcusparsers findet sich in (Marcus, 1980).

gelegt werden. Die "Clause constructor"-Regeln verarbeiten die im Buffer erscheinenden Konstituenten und versuchen daraus Phrasen zu konstruieren. Die "Clause constructor"-Regeln warten auf bestimmte syntaktische Muster. So wartet die PARTICIPIAL-MODIFIER-Regel auf die beiden Konstituenten NP und VP mit Partizip. Ist dieses Muster im Buffer vorhanden, wird diese Regel ausgelöst und konstruiert daraus eine Nominalphrase mit verkürztem Relativsatz, die dann zur weiteren Verarbeitung wieder in den Buffer zurückgelegt wird. Wenn nun zu einem bestimmten Zeitpunkt mehrere Regeln ausgelöst werden, dann wird gleichzeitig auch eine diagnostische Regel mit höherer Priorität ausgelöst. Diese diagnostische Regel überprüft weitere Konstituenten, um zu entscheiden, welche der möglichen Regeln angewendet werden soll. Aufgrund dieser Entscheidung wird dann die Kontrolle an die richtige "Clause constructor"-Regel weitergegeben, die mit der Phrasenkonstruktion weiterfährt.

Marcus (1974) illustriert die Arbeitsweise seines Parsers am Beispiel der Sätze (13) und (14).

(13) Is the block sitting in the box?

(14) Is the block sitting in the box red?

"Sitting in the box" ist in (13) Teil des Prädikates, in (14) hingegen ein verkürzter Relativsatz. Ein ATN-Parser müsste in mindestens einem der Sätze ein Backup machen, da die gleiche Phrase in den beiden Sätzen verschiedene Rollen innehat. Der interessanteste Punkt im Parsingprozess ist nach (15).

(15) Is the block sitting in the box...

Zu diesem Zeitpunkt enthält der Buffer die folgenden Konstituenten: Ein Hilfsverb (is), eine bestimmte NP (the box) und eine VP mit Partizip (sitting in the box). Durch dieses Muster werden zwei Regeln ausgelöst. Einerseits wird die PARTICIPIAL-MODIFIER-Regel ausgelöst, die Konstruktionen wie in (14) bildet und auf eine NP und eine VP mit Partizip wartet. Andererseits wird die Regel YES/NO-QUESTION ausgelöst, die Konstruktionen wie in (13) bildet und auf ein Hilfsverb, eine NP und eine VP wartet. In einer ATN-Grammatik würde nun blind eine der beiden Möglichkeiten ausprobiert. Im Marcus-Parser hingegen wird nun auch eine Diagnostik-Regel, der PARTICIPIAL-DIAGNOSTICIAN, ausgelöst, die auf ein Hilfsverb, eine NP und eine VP mit Partizip wartet. Da die diagnostische Regel die höchste Priorität hat, übernimmt sie die Kontrolle. Zuerst prüft sie, ob die Features der vorhandenen Konstituenten mit beiden Interpretationen vereinbar sind. So schliesst zum Beispiel eine NP mit Eigennamen zum vorne herein eine Konstruktion wie (14) aus. Falls immer noch beide Interpretationen

lässig sind, muss PARTICIPIAL-DIAGNOSTICIAN nur prüfen, ob alle nachfolgenden Konstituenten entweder mit ADVERB (wie zum Beispiel "for sure") oder mit TIME (wie zum Beispiel "this year") markiert sind. Ist dies der Fall, so handelt es sich um eine Konstruktion wie (13) und YES/NO-QUESTION wird aktiviert, andernfalls handelt es sich um eine Konstruktion wie (14) und PARTICIPIAL-DIAGNOSTICIAN wird aktiviert.

Zu jedem Zeitpunkt im Parsingprozess ist nur ein kleiner Teil der "clause constructor"-Regeln aktiv, d.h. nur ein kleiner Teil der Regeln kann überhaupt angewendet werden. Dazu werden die Regeln in Paketen organisiert und in den Regeln kann angegeben werden, welche der Regelpakete aktiviert bzw. deaktiviert werden sollen. So enthält zum Beispiel das Regelpaket PARSE-AUX (Marcus, 1979, S. 219) eine Regel, in der spezifiziert wird, dass das Regelpaket PARSE-AUX deaktiviert und das Reglpaket PARSE-VP aktiviert werden soll. Die Regelpakete lassen sich mit den Zuständen in der ATN-Grammatik vergleichen und die in einem Regelpaket enthaltenen Regeln entsprechen dann den von den ATN-Zuständen weggehenden Kanten. Der Unterschied der beiden Modelle liegt darin, dass in Marcus' Parser mehrere Regelpakete gleichzeitig aktiv sein können und dass Regelpakete dynamisch aktiviert bzw. deaktiviert werden können. In den Begriffen der ATN-Grammatik würde das heissen, dass nicht einem statischen Graphen gefolgt wird, sondern dass der Graph dynamisch auf den bereits geparsten Input zugeschnitten wird.

Diskussion

a) syntaktisches versus semantisches Parsen

Der Marcusparser ist in erster Linie auf syntaktischer Ebene erarbeitet worden, d.h. er verwendet vorläufig nur syntaktische Informationen. Er ist aber so ausgelegt, dass die syntaktischen Regeln in Kooperation mit semantischen Regeln arbeiten können. Wie eine solche Strategie aber im Detail realisiert werden kann, werden erst die zukünftigen Arbeiten zeigen.

b) Hypothesengesteuertes versus stimulusgesteuertes Parsen

Der Marcusparser verwendet eine gemischte Strategie. Man kann zeigen, dass dies in einem deterministischen Parsermodell notwendig ist. Die Notwendigkeit einer teilweisen Steuerung durch die Stimuli ist aus den folgenden Sätzen ersichtlich.

(15a) John bought a book.

(15b) Did John buy a book?

Ein rein hypothesengesteuertes System müsste in mindestens einem der beiden Sätze eine falsche Hypothese aufstellen. Entsprechend kann man mit den

Sätzen

(16a) I called (NP John) (S to make Sue feel better).

(16b) I wanted (S John to make Sue feel better).

zeigen, dass ein rein stimulusgesteuerter Parser mindestens einen der Sätze falsch parsen muss, da "John to make Sue feel better" auf zwei verschiedene Arten interpretiert werden kann. Die richtige Interpretation kann nur dann aufs erste Mal richtig geparkt werden, wenn der Parser aufgrund des Hauptverbs die richtige Hypothese aufstellt.

c) Depth-first versus Breadth-first Strategie

Der Marcusparser verwendet keine dieser beiden nichtdeterministischen Strategien. Der Marcusparser ist ein deterministischer Parser, der eine Ähnlichkeit mit den aus der Automatentheorie bekannten LR (K)-Grammatiken zeigt (Hopcroft & Ullman, 1969). Der Unterschied dieser beiden Modelle liegt darin, dass bei den LR(K)-Grammatiken K Wörter vorausgeschaut werden kann, beim Marcusparser hingegen K Konstituenten.

Psychologische Bedeutung des Marcusparsers

Der Marcusparser ist als psychologisches Modell vor allem aus zwei Gründen interessant. Erstens gibt er als erster Parser Evidenz dafür, dass Sprache effizient geparkt werden kann, d.h. dass der Aufwand zur Verarbeitung von Sätzen mit der Länge der Sätze nur linear ansteigt. Marcus zeigt auch, dass diese Effizienz mit einem sehr beschränkten Mechanismus erreicht werden kann, nämlich mit der gleichzeitigen Verfügbarkeit von maximal drei Konstituenten. Nun hat bereits Bever (1970) gezeigt, dass mit Hilfe von perzeptuellen Strategien Sätze effizient geparkt werden können und hat dies anhand einiger Beispiele demonstriert. Mit dem Marcusparser ist es aber möglich, diese Strategien auf einen einheitlichen und einfachen Mechanismus zu reduzieren. Marcus' Parsertheorie ist derjenigen von Bever also insofern überlegen, als sie noch detaillierter ausformuliert (und damit besser testbar ist) und dass sie sparsamer ist.

Im Marcusparser kann auch der Backtrackmechanismus (oder allgemeiner ausgedrückt, ein Korrekturmechanismus) - als psychologischer Mechanismus postuliert - neu interpretiert werden. Er muss nicht mehr, wie bei der ATN-Grammatik, als tiefer, nicht bewusster Mechanismus interpretiert werden. In den Sätzen, in denen der Marcusparser versagt und die erste Interpretation eines Satzes korrigieren muss, hat auch der Mensch den starken Eindruck, dass er eine Korrektur vornehmen musste. Andererseits lässt sich die perzeptuelle Komplexität von Sätzen nicht mehr auf die gleiche Art definieren, wie dies mit der ATN-Grammatik möglich war. Vorläufig

lässt sich aus dem Marcusparser nur ableiten, dass Gartenpfadsätze perzeptuell komplexer sind. Für eine Erklärung der andern Unterschiede, wie sie von Bever (1970) und Kaplan (1972) diskutiert wurden, ist der Marcusparser noch zu wenig weit entwickelt.

3. RIESBECKS PARSER

Riesbeck (1974, 1975) hat ein Parsermodell entwickelt, das in mehreren sprachverstehenden Systemen verwendet wurde, die auf der Conceptual Dependency Theorie von Schank (1975) beruhen. Im Gegensatz zu den bisher besprochenen Parsern basiert Riesbecks Parserstrategie vor allem auf semantischem Wissen.

Im einfachsten Modell von Riesbecks Parsern wird nur Wissen verwendet, das mit einzelnen Wörtern assoziiert ist. Dieses Wissen ist in Form von Produktionsregeln kodiert, die von Riesbeck Requests genannt werden. Der Parser verwendet nun folgende Strategie. In einem ersten Schritt werden alle Requests, die mit dem laufenden Wort assoziiert sind, aktiviert und in die Menge der aktiven Requests eingefügt. Im zweiten Schritt werden alle aktiven Requests so lange interpretiert, bis kein Test mehr erfüllt ist, worauf zum nächsten Wort im Eingabesatz weitergegangen wird. Am Beispiel des Satzes (17)

(17) John gave Mary a beating.

soll dieser Prozess illustriert werden. Die Tabelle in (18, adaptiert aus Riesbeck, 1974, S. 31) zeigt die einzelnen Schritte im Parsingprozess. In der Kolonne "Aktive Requests" wird nur der Test der Requests gezeigt, in der Kolonne "Aktion" nur die Aktion des ausgelösten Requests.

In der einfachsten Form des Riesbeck-Parsers wird der Parser nur durch Informationen gesteuert, die mit einzelnen Wörtern und vor allem mit Verben assoziiert sind. Diese in Requests formulierten Informationen werden aktiviert, sobald das entsprechende Wort im Eingabesatz auftaucht und entfernt, sobald der Test des Requests erfüllt und die Aktion ausgeführt worden ist.

Dieses Grundschema hat Riesbeck in zwei Richtungen erweitert. Im erweiterten Parser, der für das Verstehen von zusammenhängenden Texten ausgelegt ist, können Requests auch durch den vorangegangenen Text aktiviert werden. Dazu muss der Mechanismus der Aktivierung bzw. Deaktivierung von Requests verallgemeinert werden.

Ausgangspunkt und wichtigstes Konzept dieses erweiterten Parsers ist das sogenannte "Kontextcluster". Ein Kontextcluster besteht aus einer Menge von Erwartungen über den Inhalt des nächsten Satzes, etwa der Art "Hans wird Maria etwas geben". Aus den inhaltlichen Erwartungen des Kontextclusters werden nun Erwartungen über bestimmte sprachliche Konstruktionen abgeleitet.

Dazu werden die Requests nach zwei Gesichtspunkten organisiert, nach ihrem "Need" und nach ihrem "Focus". Der Need eines Requests bezeichnet

(18) Schritte beim Parsen des Satzes "John gave Mary a beating."

Schritt	laufendes Wort	Tests der wartenden Requests	ausgelöste Requests	Aktionen der ausgelösten Requests
0	-	1) Ist laufende Konstituente eine NP?	-	-
1	John	1	1	Nimm an, die NP sei Subjekt des Satzes.
2	gave	2) Ist laufende Konstituente eine NP und menschlich? 3) Ist laufende Konstituente eine NP und ein Objekt? 4) Ist laufende Konstituente eine NP und eine Aktion? 5) Wahr	5	Bilde die Konzeptualisation Nimm an, dass das Wort "to" den Rezipienten von "geben" einleitet. (Request 8) Nimm an, "Mary" sei der Rezipient. Somit ist die bis jetzt aufgebaute Konzeptualisation:
3	Mary	2,3,4 8) Ist laufendes Wort das Wort "to"?	2	
4	a	3,4,8 6) Wahr	6	behalte die aktiven Requests und ersetze sie durch Request 7.
5	beating	7) Beendet das laufende Wort eine NP?	-	-
6	PERIOD	7	7	Bilde die NP "a beating" und setze die alten Requests von Schritt 4 wieder ein.
7	-	3,4,8	4	Nimm an, die Aktion der NP sei die Hauptaktion des Satzes, das vorherige Subjekt (John) werde Aktor und der vorherige Rezipient (Mary) werde Objekt. Damit ist die endgültige Konzeptualisation:

jenes Fragment der semantischen Repräsentation, das durch das Request aufgebaut wird. Der Focus eines Requests bezeichnet jenes Fragment des Eingabesatzes oder der semantischen Repräsentation, auf die das Request wartet.

Alle Requests mit dem gleichen Need werden in einem Cluster organisiert. Wird nun ein bestimmtes Request ausgelöst, dann werden alle andern Requests im gleichen Need-Cluster deaktiviert. So ist zum Beispiel im Satz (9) nach dem Wort "give" vorausgesagt worden, dass eine menschliche NP (Request 2) oder eine mit "to" eingeleitete NP (Request 8) der Rezipient der Aktion sein wird. Mit dem Auftauchen des Wortes "Mary" in Schritt 3 ist das Request 3 erfüllt. Im einfachen Parsermodell blieb Request 8 weiterhin aktiv, während es im erweiterten Modell nun deaktiviert wird. Dies ist umso notwendiger als das unnötige Aktivbleiben eines Requests zu Missinterpretationen führen kann, wie etwa im Satz (19)

(19) John gave Mary a book to read.

in dem nach dem Parsen des Wortes "book" weiterhin erwartet wird, dass eine mit "to" eingeleitete NP den Rezipienten der Aktion spezifiziert.

Der zweite Gesichtspunkt, nachdem die Requests organisiert werden, ist der Focus, der das Fragment der semantischen Repräsentation oder des Eingabesatzes bezeichnet, auf die das Request wartet. Dadurch ist es möglich, dass beim Auftauchen eines bestimmten Satzfragmentes all jene Requests ignoriert werden können, die auf ein anderes Fragment warten.

Durch den Need und den Focus ist es möglich, die Requests in einem Netz von Abhängigkeiten zu organisieren. Der Need-Richtung folgend kann man feststellen, welche andern Requests von einem bestimmten Request abhängen, der Focus-Richtung folgend kann man feststellen, von welchen andern Requests ein bestimmtes Request selbst abhängt. Somit können über die Need-Focus-Ketten die in den Kontextclustern enthaltenen inhaltlichen Erwartungen mit den sprachspezifischen Erwartungen in Verbindung gebracht werden. Die Strategie des erweiterten Riesbeck-Parsers läuft nun darauf hinaus, dass laufend die Need-Focus-Ketten auf ihre Vereinbarkeit überprüft werden. Dadurch können aufgrund sprachspezifischer Informationen allgemeine, inhaltliche Erwartungen bestätigt bzw. widerlegt werden.

Es würde zu weit führen, diese Strategie mehr als nur beispielhaft zu erklären. Nach dem Parsen des ersten Satzes in (20)

(20) John hated Mary. He gave her a sock.

enthält das Kontextcluster zwei inhaltliche Erwartungen, nämlich a) dass John Mary schlagen würde und b) dass Mary John verletzt hatte. Nach dem Parsen von "give" im zweiten Satz kann nun eine Need-Focus-Kette aufgebaut

werden. "Give" kann in mindestens zwei Formen auftreten, einmal im üblichen Sinn des Uebergabens eines Gegenstandes und andererseits zusammen mit einer nominalisierten Aktion (wie im Deutschen "einen Kuss geben") zum Ausdruck dieser Aktion (d.h. "küssen"). Die Requests für beide Interpretationen haben als Need eine Konzeptualisation, die der Parser mit dem Kontextcluster vergleicht. Dabei findet er, dass das aufgebaute Konzeptualisationsfragment der ersten Interpretation (John gibt jemandem etwas) mit keiner der Erwartungen im Kontextcluster vereinbar ist, hingegen ist das Konzeptualisationsfragment der zweiten Interpretation mit der Hypothese a) des Kontextclusters vereinbar. Daher wird die zweite Interpretation vorgezogen. Der Need-Focus-Kette wieder Richtung sprachspezifische Requests folgend, stellt der Parser nun Präferenzen auf für a) das Request, das nominalisierte Aktionen behandelt und b) das Request, das Wörter im Sinne einer Aktion statt eines Objektes interpretiert. Gelangt daher der Parser zum Wort "sock", wird es im Sinne von "Ohrfeige" interpretiert und entsprechend in die semantische Repräsentation eingebaut.

Diskussion

Riesbecks Parser ist Teil eines integrierten sprachverstehenden Systems (Schank and Yale AI Project, 1975) und daher kommt der Interaktion des Parsers mit den übrigen Systemkomponenten eine viel wichtigere Bedeutung zu, als dies bei den übrigen besprochenen Parsermodellen der Fall ist. Aus diesem Grund wird die folgende Diskussion Riesbecks Parser insofern nicht ganz gerecht, als nur auf die bereits besprochenen Unterscheidungen eingegangen wird.

a) syntaktisches versus semantisches Parsen

Riesbecks Parser beruht auf der Schankschen Auffassung, dass syntaktische Informationen zum Verstehen natürlicher Sprache eine untergeordnete Rolle spielen. So arbeitet denn sein Parser vornehmlich aufgrund semantischer Informationen, die mit den einzelnen Wörtern oder den kontextuellen Erwartungen assoziiert sind. Allerdings muss einschränkend gesagt werden, dass Riesbecks Parser nur eine beschränkte Menge syntaktischer Varietäten verarbeiten kann und es bleibt abzuklären, wie sich sein Modell beim Parsen komplizierterer syntaktischer Formen verhalten würde.

b) Hypothesengesteuertes versus stimulusgesteuertes Parsen

Wie Marcus verwendet Riesbeck eine gemischte Strategie. Einerseits arbeitet sein Parser stimulusgesteuert, indem er Wissen verwendet, das mit einzelnen Wörtern assoziiert ist und nur dann verwendet wird, wenn diese

Wörter im Text auftauchen. Andererseits werden aus den bereits im Satz erschienen Wörtern und aus dem Kontext Erwartungen über weitere im Satz auftauchende Komponenten aufgestellt. Im übrigen ist sein Ansatz nahe mit dem Regelpaketansatz von Marcus verwandt, wobei die Pakete nach erwarteten und aufgebauten Repräsentationsfragmenten zusammengestellt werden.

c) Depth-first versus breadth-first Parsen

Riesbeck selbst behauptet, dass sein Parser ohne Backup-Mechanismen auskommt. Eine Analyse seiner Programme (vor allem Riesbeck, 1974) zeigt aber, dass er in Wirklichkeit einen depth-first Algorithmus mit gerichtetem Backup-Mechanismus verwendet. Dazu ist vielleicht einschränkend zu sagen, dass Riesbeck in neueren Arbeiten (Riesbeck, 1977) ansatzweise versucht, mit gewissen Interpretationen zuzuwarten, um Backups vermeiden zu können.

4. WILKS' PARSER

Wilks' Parser ist Teil eines maschinellen Uebersetzungssystems, das während mehreren Jahren entwickelt wurde (Wilks, 1973a, 1975; Charniak & Wilks, 1976). Der Parser ist insofern schwierig zu diskutieren, als er nur fragmentarisch dokumentiert ist (siehe aber King und Wilks, in Vorbereitung). Ferner ist der Wilkssche Ansatz so andersartig, dass es schwierig ist, Unterschiede beziehungsweise Gemeinsamkeiten mit andern Parsen herauszuarbeiten.

In Wilks' semantischer Repräsentation können vier Stufen identifiziert werden. An unterster Stelle stehen die semantischen Elemente (Primitiva), aus denen die semantischen Formeln zur Repräsentation von Wortbedeutungen aufgebaut sind (siehe dazu Abschnitt I.4.3). Auf der nächsthöheren Stufe stehen die semantischen "Templates", die etwa der intuitiven Idee einer einfachen Mitteilung (basic message) oder einem einfachen Satz entsprechen. Sie enthalten drei Elemente, einen Agenten, eine Aktion und ein Objekt, die je durch eine Formel repräsentiert werden. Die semantischen Templates sind eine der Schlüsselideen zu Wilks semantischer Repräsentation, da sie auf der Annahme beruht, dass eine relativ kleine Anzahl solcher Templates genügt, um die in der Sprache ausdrückbare Inhalte abzudecken. Die semantischen Templates ihrerseits werden durch Kasusrelationen (Paraplates) und Inferenzregeln miteinander verbunden zur Repräsentation eines ganzen Textes in einem semantischen Block.

Man kann die allgemeine Strategie des Wilksschen Parsers etwa so charakterisieren, dass mit allen möglichen Interpretationen des Eingabesatzes begonnen wird, die dann parallel durch eine Reihe von Filtern gefiltert werden, bis nur noch eine einzige Interpretation übrigbleibt. Im folgenden wollen wir eine kurze Uebersicht über die einzelnen Phasen des Parsers geben und parallel dazu die semantische Repräsentation weiter erläutern.

Phase 1: Fragmentierung In dieser Phase wird zu jedem Wort im Eingabesatz seine Formel aus dem Lexikon geholt (im Falle eines ambigen Wortes alle Formeln) und dann der Eingabesatz vorwiegend aufgrund syntaktischer Hinweise (Interpunktionen, Präpositionen usw.) in einzelne Fragmente zerlegt. Die Fragmentierung sollte es ermöglichen, dass sich in jedem Fragment ein Template identifizieren lässt. Im Falle des Satzes (21)

(21) I saw the boy with a dog in the park.

sähe die Fragmentierung folgendermassen aus:

(22) (I saw the boy) (with a dog) (in the park).

Phase 2: Bare Template Matching In dieser Phase wird versucht, in jedem Fragment ein oder mehrere Templates zu identifizieren, indem die Köpfe aller Formeln mit einer Liste von akzeptablen Templates (bare templates) verglichen werden. Zu der Menge der akzeptablen Templates gehören u.a. (MAN MAKE THING), (MAN CAUSE MAN) und (MAN BE KIND), während (MAN MAN MAN) keines ist, da es der intuitiven Idee einer einfachen Mitteilung nicht entspricht. Mit dieser Methode des bare template matchings können bereits einige falsche Interpretationen ausgefiltert werden, wie im folgenden Satz:

(23) Small men sometimes father big sons.

KIND	<u>MAN</u>	HOW	<u>MAN</u>	<u>KIND</u>	<u>MAN</u>
KIND	<u>MAN</u>	HOW	<u>CAUSE</u>	KIND	<u>MAN</u>

Unter den Wörtern sind die Köpfe der semantischen Formeln geschrieben. Nur in der Kopfsequenz mit der Bedeutung von "father" als "erzeugen" lässt sich ein bare template identifizieren, nämlich (MAN CAUSE MAN).

Phase 3: Preferential Expansion In gewissen Fragmenten lassen sich mehrere bare templates identifizieren, wie im Satz

(24) The big policeman interrogated the crook.

Es sind dies (MAN FORCE MAN) für die Interpretation von "crook" als Verbrecher und (MAN FORCE THING) für die Interpretation von "crook" als Hirtenstab. In der Expansionsphase wird nun geprüft, wieviele der in den Formeln spezifizierten Präferenzen erfüllt sind. Bei diesen beiden Interpretationen liegt der Unterschied darin, dass "interrogate" ein menschliches Objekt bevorzugt und nur "crook" im Sinne von Verbrecher diese Präferenz erfüllt. Mit der Wahl von "crook" im Sinne von Verbrecher ist in der semantischen Repräsentation ein Zusammenhang (eine Abhängigkeit) mehr gefunden oder in andern Worten, eine dichtere semantische Repräsentation gefunden.

Phase 4: Paraplate matching In dieser und der nächsten Phase werden zwischen allen Templates eines semantischen Blocks Verbindungen konstruiert. Eines dieser Verbindungskonstrukte ist das Paraplate, das normalerweise Präpositionalfragmente mit dem Rest des Satzes verbindet, wie etwa im Satz

(25) John left Zürich by the highway.

In den Templates dieses Satzes repräsentiert □ den Scheinagens der Präposition, die wie eine Aktion behandelt wird. (Der Einfachheit halber werden im folgenden in den Templates die ursprünglichen Wörter anstelle der semantischen Formeln ausgesetzt.)

(26a) (JOHN LEFT ZUERICH)

(26b) (□ BY THE+HIGHWAY)

Für jede Präposition gibt es nun eine Reihe von Paraplates, die angeben,

wie Konstruktionen mit dieser Präposition interpretiert werden müssen. Eines der Paraplates von "by" ist (27).

(27) (*ANI MOVE (WHERE POINT)) $\xrightarrow{\text{DIRE}}$ (□ BY (WHERE LINE))

Es kann auf (26) angewendet werden, da John unter alles Belebte (*ANI) fällt, leave eine Bewegung (MOVE) ist, Zürich ein räumlicher Punkt (WHERE POINT) und Highway eine räumliche Linie (WHERE LINE) ist. Das Paraplate (27) besagt daher, dass (26a) und (26b) durch die Kasusrelation DIRE (Richtung) miteinander verbunden werden können. Falls mehrere Paraplates anwendbar sind, wird wie in der Expansionsphase jenes Paraplate angewandt, das die meisten Präferenzen erfüllt und daher zur dichtesten semantischen Repräsentation führt.

Phase 5: Extended Inferential Mode Falls mit den bisherigen Filtern noch nicht alle Mehrdeutigkeiten gelöst werden konnten, muss mit Hilfe von Inferenzen versucht werden, einen einzigen kohärenten semantischen Block zu konstruieren. Es würde zu weit führen, diese Phase detailliert zu beschreiben. Deshalb soll sie nur anhand einer einfachen Pronomenreferentialisierung illustriert werden, nämlich im Satz (28),

(28) John bought a car and liked it immediately.

dessen Repräsentation ungefähr so aussieht:

(29) (JOHN BOUGHT A+CAR)
(JOHN IMMEDIATELY+LIKED IT)

Die Inferenzregel, die zur Lösung der Pronomenreferenz in (29) führt, ist

(30) ((*ANI 1) ((SELF HAVE) CAUSE) (*REAL 2)) \rightarrow (1 *JUDGE 2).

Sie besagt, dass wenn ein belebter Agens (*ANI) ein Objekt (*REAL) erwirbt ((SELF HAVE) CAUSE), dann ist es oft der Fall, dass er es auch beurteilt ("JUDGE). Die Zahlen in den Formeln geben an, welche Formeln miteinander übereinstimmen müssen. Mit dieser Inferenzregel kann nun abgeleitet werden, dass sich "it" auf "car" beziehen muss, damit die beiden Formeln übereinstimmen.

Diskussion

a) syntaktisches versus semantisches Parsen

Wie der Riesbeck-Parser ist auch der Wilks-Parser vorwiegend semantisch orientiert. Im Unterschied zu Riesbeck anerkennt Wilks aber die Informationsträchtigkeit syntaktischer Hinweise und es ist klarer, wie weitere syntaktische Strategien im Wilksschen Parser eingebaut werden könnten, nämlich in der bis jetzt äusserst einfach gehaltenen Fragmentierungsphase.

b) Hypothesengesteuertes versus stimulusgesteuertes Parsen

Wilks' Parser verwendet eindeutig eine stimulusgesteuerte Strategie. Die richtige Interpretation wird nicht aufgrund inhaltlicher oder syntaktischer Erwartungen selegiert, sondern jene Interpretation wird selegiert, die zur dichtesten semantischen Repräsentation führt. Wenn man die Formeln und Templates im Wilks-Parser mit den wortassoziierten Requests von Riesbeck vergleicht, dann fällt auf, dass Wilks' Parser viel weniger verbzentriert ist. Während bei Riesbeck das Verb praktisch alle Information zur Bildung der semantischen Repräsentation assoziiert hat, ist diese bei Wilks auf alle Wörter verteilt: Ein Nomen kann eine ebenso aktive Rolle zur Bildung der semantischen Repräsentation spielen wie ein Verb.

Ein zweiter interessanter Unterschied der beiden Parser liegt darin, dass bei Riesbeck die semantische Repräsentation (Conceptual Dependency Repräsentation) und die Informationen, die zum Parsen gebraucht werden, getrennt sind, während sie bei Wilks nicht trennbar sind: Bei Wilks kann die Parserstrategie direkt aus der semantischen Repräsentation abgeleitet werden.

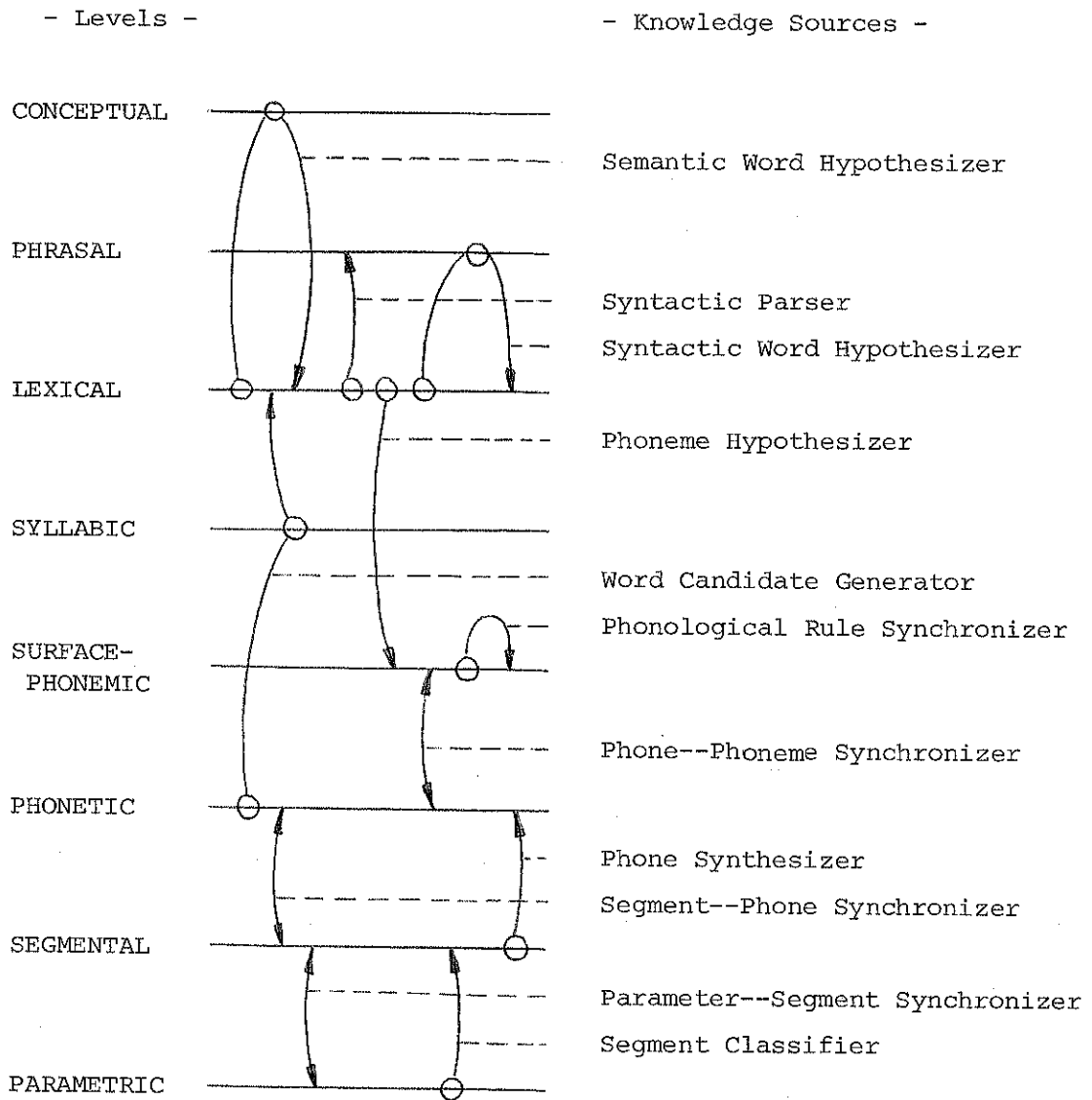
c) Depth-first versus breadth-first Parsen

Eines der wichtigsten Charakteristiken des Wilks-Parser im Vergleich zu den bisher besprochenen Parsern ist seine Breadth-first-Strategie. Es wird von allen möglichen Interpretationen des Eingabesatzes (allen möglichen Templates) ausgegangen, die so lange parallel analysiert oder gefiltert werden, bis nur noch eine einzige Interpretation überlebt.

5. VERSTEHEN GESPROCHENER SPRACHE

Obwohl hier nur auf Parser eingegangen wird, die geschriebene Texte verarbeiten, wollen wir ganz kurz auf einige Speech Understanding Systeme eingehen, die etwas andere Schwerpunkte in bezug auf Parserorganisation und Parserstrategien legen. Wir ignorieren dabei diejenigen Probleme, die für die Erkennung gesprochener Sprache spezifisch sind, wie etwa die Erkennung von Schallwellenmustern.

Anhand des HEARSAY-II Systems (Reddy, Erman, Fennell & Neely, 1973; Erman & Lesser, 1975; Lesser, Fennell, Erman & Reddy, 1975; Lesser & Erman, 1977) lässt sich ein Aspekt illustrieren, der bei den bisher besprochenen Parsern nur rudimentär entwickelt ist, nämlich die gleichzeitige und gleichwertige Verwendung verschiedener Wissensquellen. Damit ist gemeint, dass HEARSAY-II gleichzeitig akustisches Wissen (z.B. über die Identifikation von Phonemen in Schallwellenmustern), syntaktisches Wissen (über die Grammatik der Sprache), semantisches Wissen (über die Bedeutung von Wörtern und Segmenten) und pragmatisches Wissen (über die Wahrscheinlichkeit bestimmter Äußerungen im laufenden Kontext) verwendet. Jede dieser Wissensquellen besteht aus einer Menge von Prozeduren, die einerseits Hypothesen über gewisse Fragmente des Eingabesatzes aufstellen und andererseits solche Hypothesen prüfen und bewerten können. Die Wissensquellen kommunizieren miteinander über eine sogenannte "Wandtafel", auf die jede ihre Hypothesen schreibt und andererseits die auf der Wandtafel vorhandenen Hypothesen beurteilt, falls sie dafür zuständig sind. Ein Kontrollprogramm überwacht den Ablauf des ganzen Systems und übergibt jeweils die Kontrolle derjenigen Wissensquelle, die an der wahrscheinlichsten Hypothese arbeitet. Der Parsingprozess stoppt dann, wenn eine als sehr wahrscheinlich beurteilte Hypothese über die Interpretation des ganzen Satzes vorliegt. In (31) ist illustriert, wie die verschiedenen Wissensquellen miteinander interagieren, wobei die Wissensquellen noch weiter als bisher differenziert werden. Dabei zeigen die Pfeile von derjenigen Wissensquelle, die eine Hypothese aufstellt auf die Wissensquelle, die diese Hypothese beurteilt. Die Abbildung illustriert, dass diese Interaktionen sehr kompliziert sind und dass daher auch der Ablauf des Parsingprozesses sehr kompliziert sein muss. Es würde zu weit führen, diesen Ablauf auch nur an einem Beispiel zu zeigen und wir verweisen daher auf die Beschreibung von Rumelhart (1977, S. 103ff). Für unsere Diskussion sind zwei Folgerungen aus Hearsay-II wichtig:



(31) A Set of Knowledge Sources for Hearsay-II

(aus: Erman & Lesser, 1975, S. 489)

Erstens war unsere bisherige Unterscheidung von syntaktischen und semantischen Parserstrategien vereinfachend. In Speech Understanding Systemen, in denen die Unsicherheit über die Interpretation der Eingabe viel grösser ist, kann nicht darauf verzichtet werden, neben dem syntaktischen und semantischen Wissen auch alle andern möglichen Informationsquellen zu berücksichtigen. Dies wird aber auch bei textverarbeitenden Parsern notwendig werden, wenn sie auch Texte minderer Qualität (d.h. mit Orthographiefehlern, agrammatischen Konstruktionen usw.) verarbeiten müssen. Leistungsfähige Parser können aber auch nicht mehr auf eine einzige Wissensquelle ihr Schwergewicht legen, wie dies bei den bisher besprochenen Parsern der Fall war. Nur durch die gleichwertige Verwendung aller im Text enthaltenen Informationen kann effizient und leistungsfähig geparst werden.

Die zweite wichtige Folgerung aus Hearsay-II ist die, dass die gleichwertige Verwendung verschiedener Wissensquellen auch eine andere Organisation des ganzen Parsingprozesses bedingt. In der hierarchischen Organisation der bisherigen Parser steht für jede Prozedur fest, wann und von wem sie aufgerufen wird, d.h. von wem sie die Kontrolle übernimmt. So wird zum Beispiel im Wilksschen Parser der "Extended inferential mode" nur dann aufgerufen, wenn nach dem "Paraplate matching" immer noch mehrere mögliche Interpretationen vorhanden sind. Im Gegensatz dazu muss in einer Organisation wie beim HEARSAY-II-Parser jede Prozedur jederzeit die Kontrolle übernehmen, unterbrechen und wiederaufnehmen können. Es ist daher nicht mehr unbedingt möglich, Phasen einer bestimmten Verarbeitungsweise, etwa einer syntaktischen Analyse, zu identifizieren oder sogar vorauszusagen.

Eine andere Erweiterung der bisher besprochenen Parserstrategien lässt sich anhand des HWIM Speech Understanding Systems (Woods, Bates, Brown, Bruce, Cook et al., 1976; Woods, 1977; Woods, 1978) illustrieren. Während die meisten andern Parser eine strikte links-nach-rechts-Strategie verfolgen, wendet HWIM die sogenannte "Insel-getriebene" oder "Mittenach-aussen" Strategie an. Dabei beginnt der Parser mit jenem Wort des Satzes, das er am sichersten erkennen kann und bestimmt dann, mit welchen andern Wörtern sich dieses erste Wort (das sogenannte "seed") links und rechts kombinieren lässt. Diese Hypothesen werden nun überprüft bzw. beurteilt und resultieren in einer erweiterten Hypothese über ein Satzfragment. Alle Hypothesen (und ihre Beurteilungen) werden in eine Warteschlange gelegt, aus der laufend die am wahrscheinlichsten beurteilte Hypothese herausgeholt und weiterverfolgt wird. So konkurrieren die neuen Hypothesen sowohl untereinander als auch mit den übrigen "seed"-Hypothesen und

veranlassen somit den Parser, manchmal eine Hypothese über ein Satzfragment weiter zu verfolgen und manchmal auf eine andere kompetierende Hypothese überzuwechseln. Im Verlaufe des Parsingprozesses werden nun laufend die besten Hypothesen erweitert und miteinander kombiniert, bis eine genügend gute Hypothese über den ganzen Satz vorliegt, die als die endgültige Interpretation des ganzen Satzes betrachtet wird.

Wie schon beim HEARSAY-II-System haben wir das HWIM-System nur erwähnt, um einen neuen Aspekt in unsere bisherige Diskussion der verschiedenen Parserstrategien zu bringen. Das System ist bei seiner noch so bescheidenen Leistung so komplex, dass eine nur annähernd angemessene Beschreibung den Rahmen dieser Diskussion völlig sprengen würde. Für unsere Diskussion sind zwei Punkte wichtig: Erstens ist strikte links-nach-rechts-Strategie bei fehlerhaften und vor allem bei unvollständigen Sätzen ineffizient. Gerade in solchen Fällen ist aber die inselgetriebene Strategie optimal und es ist gut vorstellbar, dass eine Kombination dieser beiden Strategien zu einem sehr leistungsfähigen Parser führen wird.

Zweitens besteht wie schon HEARSAY-II auch HWIM aus einer Menge unabhängiger Prozesse, die je eine Hypothese verfolgen und die heterarchisch organisiert sind. Dies legt nahe, dass bei Systemen der Komplexität von HEARSAY-II oder HWIM eine nichthierarchische Kontrollstruktur von Vorteil ist. Der Grund liegt darin, dass solchen Systemen die Interaktionen zwischen allen Systemkomponenten nicht mehr voraussagbar ist und daher eine unabhängige Konzipierung der einzelnen Systemkomponenten einfacher zu erfassen ist.

6. PARTIELLES PARSEN

Neben den bisher besprochenen Parsern gibt es noch eine ältere Tradition von Parsern, die nicht eine vollständige Beschreibung des Eingabesatzes konstruieren, sondern nur versuchen, die im Zusammenhang mit dem Kontext wichtigsten Schlüsselwörter und Phrasen zu identifizieren. Dies soll anhand von Bobrows (1968) Programm STUDENT illustriert werden, einem Programm, das algebraische Probleme wie (32) lösen kann.

(32) The number of customers Tom gets is twice the square of 20 per cent of the number of advertisements he runs, and the number of advertisements he runs is 45, what is the number of customers Tom gets?

Der Text (32) wird mit Hilfe von Transformationsregeln in ein System von Gleichungen (33) umgeformt, das dann vom Programm gelöst wird.

(33) (EQUAL X (NUMBER OF CUSTOMERS TOM GETS))
(EQUAL (NUMBER OF ADVERTISEMENTS HE RUNS) 45)
(EQUAL (NUMBER OF CUSTOMERS TOM GETS)
(Times 2
(EXPT (TIMES 0.2
(NUMBER OF ADVERTISEMENTS HE RUNS))
2)))

Die Transformationsregeln zur Transformation des Textes in Gleichungen haben alle die gleiche Form wie die Beispiele in (34).

(34) (x IS y) → (EQUAL x y)
(TWICE x) → (TIMES 2 x)
(Zahl PERCENT OF x) → (Zahl/100 OF x)
(Zahl OF x) → (TIMES Zahl x)

STUDENT führt weder eine syntaktische Analyse durch noch produziert es eine allgemeine semantische Repräsentation. Es sucht im Eingabetext nur nach bestimmten Schlüsselwörtern und Phrasen, die zur Konstruktion der Gleichungen notwendig sind, während die übrigen Satzfragmente wie "number of customers Tom gets" nicht weiter analysiert werden.

Die gleiche Strategie wird auch in den Programmen von Colby, Weber & Hilf (1971) und Colby (1975) verwendet, in denen ein Psychotherapeut oder ein paranoider Patient simuliert wird. Der folgende Ausschnitt aus einem Dialog (Colby, 1975, S. 75f), in dem das Programm einen paranoiden Patienten simuliert, soll die Leistungsfähigkeit dieser Programme illustrieren.

- (35) Dr. How did you come to be in hospital?
Pt. I am upset.
Dr. How long have you been in hospital?
Pt. About a week.
Dr. In what way are you upset?
Pt. People make me nervous.
Dr. In what way do they make you nervous?
Pt. People give me a funny look sometimes.
Dr. Is there anything about people in particular?
Pt. Certain persons.
Dr. Who are the certain persons?
Pt. Do you know how the underworld operates?

Auch diese Programme suchen den Eingabetext nach bestimmten Schlüsselwörtern und Phrasen ab und versuchen den Text aufgrund dieser Hinweise so lange zu transformieren, bis er mit einem der abgespeicherten abstrakten Muster in der Datenbasis übereinstimmt. Zu diesem Muster wird dann eine Antwort gesucht und zurückgegeben. So wird zum Beispiel in der Frage

(36) What is your current occupation?

zuerst das Wort "current" eliminiert, da es unbekannt ist, dann "is" durch "be", "your" durch "you" und "occupation" durch "job" ersetzt. Auf das resultierende Muster (37) existiert in der Datenbasis eine Antwort, die dann zurückgegeben wird. Es ist nicht schwierig, die Schwächen eines solchen

(37) (WHAT BE YOU JOB)

Parsers herauszufinden, der nur nach bestimmten Schlüsselwörtern und Phrasen sucht. Ein echtes Verständnis des Gesagten ist oftmals verunmöglicht und zum Beispiel würde das Programm auf die Frage (38)

(38) What is your fathers occupation?

die gleiche Antwort wie auf (36) geben.

Auf der andern Seite sind Parser wie der von Bobrow oder Colby deshalb interessant, weil sie illustrieren, welche Bedeutung dem Gebrauch von stereotypen Wendungen und Schlüsselwörtern in der Sprache zukommt. Für eine Theorie des Parsens ist aber nicht nur die Tatsache wichtig, dass solche Wendungen wichtige Informationen in sich tragen können. Ebenso wichtig ist, dass solche Wendungen zu einem Problem für einen üblichen Parser werden können, wenn sie nicht als Einheit erkannt werden. So ist es unglaubhaft, dass Wendungen wie

(39) Wären Sie bitte so freundlich und würden ...

oder

(40) Ich komme nicht umhin zu sagen, dass ...

in der üblichen Art geparst werden.

Die Idee des partiellen Parsens wurde von Schank, Lebowitz & Birnbaum (1978) wieder aufgegriffen. Er geht davon aus, dass die Prozesse des

Parsens, Inferierens und der Gedächtnissuche völlig integriert und untrennbar sind. Würden diese Prozesse aber gleichzeitig und vollständig ablaufen, dann käme es zu einer Ueberlastung der Verarbeitungskapazität (Norman & Bobrow, 1975). Daraus kann geschlossen werden, dass die tieferen Verarbeitungen zum Teil auf Kosten der Verarbeitung unwichtiger Wörter ablaufen müssen.

Die Entscheidung darüber, welchen Wörtern Aufmerksamkeit geschenkt wird, und welche überflogen werden, hängt von verschiedenen Faktoren ab. Wichtig sind zum Beispiel Wörter, die bestimmte Erwartungen erfüllen, seien es nun Erwartungen des Parsers über bestimmte Konstituenten oder inhaltliche Erwartungen, die etwa aus Frames abgeleitet werden. Unabhängig davon kann man auch von einem einfachen, themaspezifischen Interessanzwert ausgehen. Man denke nur an die sogenannten "Buzzwords", die dem Leser schon beim raschen Ueberfliegen eines Textes in die Augen springen.

Unter diesen Voraussetzungen gelangt man zu einer modifizierten Parserstrategie, die gewisse Aehnlichkeiten mit den Parsern von Colby und Bobrow aufweist. Im Text werden nicht mehr alle Wörter gleichzeitig von links nach rechts verarbeitet. Uninteressante Wörter werden entweder völlig ignoriert oder unverarbeitet in einem STM-Speicher abgelegt. Erst interessante Wörter lösen tiefere Verarbeitungsprozesse aus, in denen wichtige Informationen in die Textrepräsentation eingefügt werden und neue Erwartungen aufgestellt werden, die die weitere Verarbeitung steuern. Falls es diese Verarbeitung verlangt, können auch die im STM-Speicher behaltenen Wörter nachträglich noch in die Textrepräsentation eingearbeitet werden.

Es soll hier darauf verzichtet werden, die Arbeitsweise des partiellen Parsers an einem Beispiel zu illustrieren, da daraus kaum auf seine allgemeine Leistungsfähigkeit geschlossen werden kann. Einerseits ist es aber einleuchtend, dass ein solcher Parser viel robuster gegen unbekannte Wörter und Wendungen ist. Während ein partieller Parser auch aus einem unvollständigen Text ein (partiell) Resultat produzieren kann, versagen die andern Parser vollständig.

Andererseits hängt die Fähigkeit des Parsers, wichtige Informationen aus einem Text extrahieren zu können, ganz entscheidend von den abgespeicherten Wissensstrukturen ab. Mit einem solchen Parser ist es unmöglich, prinzipiell neue Wissensinhalte zu parsen, da sie wegen der fehlenden entsprechenden Wissensstrukturen einfach ignoriert werden. Diese Parserphilosophie ist sehr nahe verwandt mit dem sogenannten Frames-Ansatz des Verstehens, in dem ein Text - vereinfachend gesagt - dadurch verstanden wird, dass er mit

bestehenden Wissensstrukturen in Verbindung gebracht wird. Sachverhalte, die nicht auf bestehende Wissensstrukturen bezogen werden können, werden daher nicht verstanden. Beim partiellen Parsen wird diese Philosophie auf das Parsen übertragen. Texte, die nicht auf bestehende Wissensstrukturen bezogen werden können, werden ignoriert.

IV. R E P R A E S E N T A T I O N V O N W I S S E N

Im letzten Kapitel wurden einige Parsermodelle diskutiert, wobei vor allem auf die Eigenschaften der verschiedenen Parserstrategien eingegangen wurde. Gleichzeitig wurde aber auch immer wieder diskutiert, wie das Wissen, auf das die Parser zurückgreifen, repräsentiert werden kann. In diesem und dem nächsten Kapitel soll auf die Probleme der Wissensrepräsentation weiter eingegangen werden, wobei in diesem Kapitel vor allem auf Fragen der Wissensrepräsentation am Beispiel der semantischen Netzwerke eingegangen wird und im nächsten Kapitel am Beispiel des Framesansatzes Probleme der Wissensorganisation diskutiert werden.

Es wurde bereits mehrmals darauf hingewiesen, dass ein sprachverstehendes System sich auf verschiedene Arten von Wissensquellen stützt, für die die traditionelle Unterscheidung zwischen syntaktischem, semantischem und pragmatischem Wissen (das letztere wird auch als Alltagswissen bezeichnet) übernommen wurde. Unter syntaktischen Regeln werden Regeln wie etwa die Regel über Subjekt-Hauptverb-Konkordanz verstanden, unter semantischen Regeln etwa diejenige, dass das Objekt von "töten" belebt sein muss und unter Alltagswissen werden Regeln wie etwa die, dass Kühe Gras fressen, verstanden. Hinter dieser Unterscheidung verschiedener Wissenstypen liegt vor allem ein historisches Motiv, indem hinter der Repräsentation syntaktischen Wissens eine lange Tradition, vor allem in der Linguistik, steht, während die Repräsentation semantischen und vor allem pragmatischen Wissens erst in neuerer Zeit in der Linguistik und vor allem in der Künstlichen Intelligenz in den Vordergrund getreten ist. Es muss aber darauf hingewiesen werden, dass diese Wissenstypen weder scharf voneinander abgegrenzt werden können, noch dass ihre Unterscheidung in einem Prozessmodell unbedingt gerechtfertigt werden kann.

Es wurden im Laufe der Zeit eine ganze Reihe von Repräsentationsformalismen eingeführt, über die im folgenden eine kurze Uebersicht gegeben wird und von denen einer, die semantischen Netzwerke, anschliessend eingehend diskutiert wird. Die Unterschiede zwischen diesen Formalismen, beziehungsweise ihre Vor- und Nachteile können oft nur oberflächlich angegeben werden. Ein Grund dazu liegt in der, bereits im ersten Kapitel erwähnten, mangelnden Formalisierung der Semantik dieser Formalismen. Oftmals spielen daher in der Popularität einzelner Formalismen Modetrends

und/oder persönliche Präferenzen mit. Oft sind sie aber auch auf irrelevanten Unterscheidungen begründet, wie dies vor allem Pat Hayes (1975) eingehend kritisiert hat. Und gerade im Hinblick auf die Verwendung in einer psychologischen Theorie müssen die Dispute um die verschiedenen Formalismen einer kritischen Betrachtung unterworfen werden. Es ist nicht selten passiert, dass Repräsentationsformalismen (unter anderen die semantischen Netzwerke) anhand akzidenteller Eigenschaften auf ihre psychologische Plausibilität überprüft und als psychologisch unplausibel verworfen wurden. Aber nicht nur in der Psychologie ist das passiert, auch in der Künstlichen Intelligenz selbst. So sind etwa die Argumente, die anfangs der Siebzigerjahre gegen die Verwendung des Prädikatenkalküls als Repräsentationsformalismus angeführt wurden, vor allem gegen die Syntax und die damals noch wenig entwickelten Deduktionstechniken gerichtet gewesen. Sie sind aber nur akzidentielle Eigenschaften eines Prädikatenkalküls und konnten auch, wie sich seither gezeigt hat, entscheidend verbessert werden.

1. INFERENZEN

In der Diskussion von Repräsentationsformalisten wurde und wird immer wieder von Inferenzen gesprochen. Darunter wird, wie in der Logik, die Anwendung einer Inferenzregel verstanden, mit dem Unterschied, dass die Inferenzen nicht logisch strigent zu sein brauchen.

Einer der wichtigsten Beiträge der Künstlichen Intelligenz zu einer Theorie des Sprachverstehens liegt darin, dass sie überzeugend gezeigt hat, dass zum Sprachverstehen eine grosse Menge von Alltagswissen notwendig ist und dass im Verlaufe des Verstehensprozesses eine grosse Anzahl von Inferenzen gemacht werden müssen. Eines der bekanntesten Beispiele dazu ist die folgende Geschichte aus (Charniak, 1972, S. 32):

- (1) Today was Jack's birthday. Janet and Peggy went to the store. They had to get presents. "I will get a top", said Janet. "Don't do that", said Peggy. "Jack has already a top. He will make you take it back.

Um zu entscheiden, dass sich das letzte Pronomen "it" nicht auf den Kreislauf bezieht, den Jack bereits hat, sondern auf den, den er erhalten soll, genügt es nicht, nur auf syntaktische oder semantische Informationen zurückzugreifen. Die richtige Entscheidung kann nur mit Hilfe von Alltagswissen gemacht werden. In einer umfassenden Analyse hat Charniak (1974) dieses Wissen in Inferenzregeln zu formulieren versucht und gezeigt, dass zur Bestimmung der Pronomenreferenz mehrere Inferenzschritte notwendig sind, wenn nicht spezifische ad-hoc-Regeln verwendet werden sollen. So wären zur Lösung der Pronomenreferentialisierung von (1) unter anderen die folgenden vier Regeln (aus Charniak, 1974) notwendig:

- (2) If PERSON gets OBJECT in order to give it to PERSON 2 and PERSON 2 wants (not wants) OBJECT then PERSON wants (not wants) OBJECT.
- (3) If PERSON not-wants OBJECT then PERSON might want re-exchanging OBJECT.
- (4) If it is plausible, that PERSON wants action A and PERSON does A1, a significant subaction of A, then PERSON does A.
- (5) Taking back anOBJECT to the location where it has been before is a significant subaction of re-exchanging it.

Im Entwurf eines Systems, das die Fähigkeit hat, solche Inferenzen zu ziehen, tauchen nun eine ganze Reihe von Fragen auf, Fragen, die die Re-

präsentation und Anwendung dieser Inferenzregeln betreffen. Dazu gehören auch die folgenden Fragen:

- 1) Welche Inferenzregeln sind notwendig?
- 2) Wann werden Inferenzen gemacht?
- 3) Wie werden Inferenzregeln repräsentiert?
- 4) Wie werden Inferenzregeln angewendet?
- 5) Wie werden Inferenzregeln organisiert und aufgefunden?

In diesem und dem nächsten Kapitel werden diese Fragen immer wieder auftauchen und - wenigstens teilweise - beantwortet.

1.1. Welche Inferenzregeln sind notwendig?

Nur in sehr wenigen Arbeiten wurde versucht, das zum Verstehen eines bestimmten Themas notwendige Wissen systematisch in Regeln zu erfassen (etwa Charniak, 1974; Denofsky, 1976; Pat Hayes, 1978). Dabei zeigt sich, dass auch für ein noch so eingeschränktes Thema - etwa die Geschichte (1) - es sehr schwierig sein kann, gute und relativ allgemeine Inferenzregeln aufzustellen. Und wenn man andererseits sieht, wieviele Regeln etwa das Programm DENDRAL braucht (Buchanan, Sutherland & Feigenbaum, 1969; Feigenbaum, Buchanan & Lederberg, 1970; Feigenbaum, 1977; Buchanan & Mitchell, 1978), um "nur" Spektrogramme zu interpretieren, kann man ersehen, dass hier auch die AI vor einem äusserst schwierig zu lösenden Problem steht.

In bezug auf zwei Aspekte dieses Problems sind aber wenigstens teilweise Antworten gefunden worden.

Spezifität der Inferenzregeln: In frühen Arbeiten (etwa Charniak, 1972; Rieger, 1974) wurde versucht, die beim Verstehen auftauchenden Probleme mit möglichst allgemein gültigen Inferenzregeln zu lösen. Die extremste Position wurde von Rieger (1974) eingenommen, in dessen auf Schanks Conceptual Dependency Theorie beruhendem Inferenzsystem nur sehr allgemeine Inferenzregeln verwendet werden, die mit den primitiven Acts assoziiert sind. Dabei zeigte sich, dass bereits für relativ einfache Probleme mehrere Inferenzschritte notwendig werden und es ist nicht abschätzbar, mit wieviel Aufwand ein Desambiguierungsproblem wie in (1) gelöst werden könnte.

Auf der andern Seite führt die unsophistizierte Verwendung spezifischer Inferenzregeln noch rascher zu Problemen, da diese typischerweise nur auf wenige Situationen anwendbar sind, und in den übrigen Situationen zu falschen Schlüssen führen.

Ein möglicher Ausweg aus diesem Dilemma wird durch den unten diskutierten Frames-Ansatz angedeutet, der einfach gesagt darin besteht, dass zusammenhängende, situationsspezifische Inferenzregeln in einem Frame zusammengefasst werden. Diese werden erst dann aktiviert (d.h. können erst dann angewendet werden), wenn die entsprechende Situation auftaucht. So wäre zum Beispiel das Wissen über "Schenken" unter anderem im "Geburtstags"-Frame enthalten und würde erst aktiviert, wenn die Rede auf das Thema Geburtstag kommt.

Taxonomie von Inferenzregeln: Gegeben die Tatsache, dass aus jedem Text eine unbeschränkte Anzahl von Inferenzen gemacht werden können, erscheint es notwendig, zwischen wichtigen und unwichtigen Inferenzen unterscheiden zu können. So wäre etwa die Inferenz, dass Janet Lungen hat, in der Geschichte (1) eine unwichtige Inferenz, da sie zur Lösung der Pronomen-referentialisierung nichts beitragen kann. Da die Unterscheidung zwischen unwichtigen und wichtigen Inferenzen von der entsprechenden Situation abhängt, wird es notwendig, eine feinere Taxonomie der Inferenzregeln aufzustellen, die die durch die Inferenzregeln hergestellten Zusammenhänge beschreibt und so zur Steuerung des Inferenzprozesses verwendet werden kann.

Im Laufe der Zeit sind nun eine ganze Reihe von Taxonomien von Inferenzen und Zusammenhängen aufgetaucht (insbesondere Schank, 1973, 1974; Rieger, 1974, 1975; Charniak, 1975a; Wilks, 1975b), die zum Teil sehr elaboriert und differenziert gehalten sind. So braucht Rieger (1975) nicht weniger als 25 verschiedene Relationen, um den Zusammenhang zweier Ereignisse auszudrücken, Relationen, die alle auf einer naiv-physikalischen Beschreibung von Ereignissen beruhen.

Schwieriger hingegen gestaltet sich in diesen Arbeiten in einem Prozessmodell die Motivation einer - intuitiv einleuchtenden - Taxonomie. Damit ist gemeint, dass eine verwendete Taxonomie von Inferenzregeln auch tatsächlich zur Steuerung des Inferenzprozesses verwendet wird, wie dies etwa in Charniak (1975a) versucht wurde.

1.2. Wann werden Inferenzen gemacht?

In einem System, das Texte liest und dann Fragen über diesen Text beantworten kann, gibt es grundsätzlich zwei Möglichkeiten, wann das System Inferenzen macht. Es kann einerseits den Text unverarbeitet aufnehmen und erst dann Inferenzen machen, wenn eine Frage gestellt wird (Fragezeit-Inferenzen) oder es kann bereits beim Lesen Inferenzen machen (Lesezeit-Inferenzen). Der Vorteil eines Fragezeit-Inferenzsystems

liegt darin, dass die Anzahl der gemachten Inferenzen auf ein Minimum reduziert wird. Andererseits kann aber ein solches System die Forderung nach einer eindeutigen semantischen Repräsentation nicht erfüllen, da bereits zur Desambiguierung Inferenzen gemacht werden müssen, wie die Geschichte (1) illustriert.

Interessanter und lange Zeit umstrittener war die Frage, ob Lesezeit-Inferenzen nur dann gemacht werden sollen, wenn ein Desambiguierungsproblem auftaucht (sogenanntes problemausgelöstes Inferieren) oder ob auch in andern Situationen Inferenzen gemacht werden sollen (sogenanntes problemunabhängiges Inferieren). Die Frage war deshalb umstritten, weil sich die Positionen je nach Zielsetzung, die dem sprachverstehenden System zugrundeliegt, mehr oder weniger gut vertreten lassen.

Wenn ein System jene Leistung erbringen soll, die wir unter "echtem, tiefem Verständnis" verstehen, dann lässt sich die Anwendung problemunabhängiger Inferenzen einfach vertreten, da es zu diesem Ziel gehört, dass das System zu einem Text eine umfassende und kohärente Interpretation finden kann. Hier leuchtet es daher ein, dass bereits während des Lesens zusätzlich Inferenzen gemacht werden müssen, unabhängig davon, ob sie für Desambiguierungen gebraucht werden oder nicht.

In aufgabenorientierten Anwendungen hingegen ist das Hauptziel die Konstruktion einer eindeutigen semantischen Repräsentation, die die Grundlage für die weiteren Aufgaben des Systems bilden, sei es nun in einem Uebersetzungssystem die Erzeugung des Textes in der Zielsprache oder in einem Frage-Antwort-System die Suche nach der Antwort auf die gestellte Frage. Hier basierten die Argumente um die Notwendigkeit von problemunabhängigen Inferenzen meist auf Beispielen, in denen der zu einer Desambiguierung notwendige Hinweis im Text weit zurückliegt (etwa Charniak & Wilks, 1975, S. 12f). In solchen Situationen versagen Systeme, die nur problemausgelöste Inferenzen durchführen und daher einen viel oberflächlicheren und lokaleren Kontext aufbauen.

Wenn man hingegen die Position des problemunabhängigen Inferierens vertritt, dann taucht das zusätzliche Problem auf, wie die Zahl der Lesezeitinferenzen vernünftig beschränkt werden kann. Dazu müssen einerseits Kriterien festgelegt werden, die die Richtung des Inferenzprozesses bestimmen, aber es braucht auch Kriterien, wie weit in einer bestimmten Richtung Inferenzen gezogen werden sollen. Auf diese Fragen wird im nächsten Kapitel über die Organisation des Wissens noch teilweise eingegangen.

2. EINIGE REPRÄSENTATIONSFORMALISMEN

Der Prädikatenkalkül ist wohl der traditionsreichste Repräsentationsformalismus und dominierte auch in der Künstlichen Intelligenz bis zum Ende der Sechzigerjahre. Dann büsste er allerdings an Popularität ein, die er erst in neuerer Zeit wieder am Zurückgewinnen ist. Im Prädikatenkalkül (PK) werden Fakten durch atomare Ausdrücke repräsentiert, etwa

(6) Mensch (Fritz)

oder "Alle Menschen sind fehlbar" durch

(7) $\forall x(\text{Mensch}(x) \supset \text{fehlbar}(x))$.

Inferenzen werden durch die Anwendung formaler Inferenzregeln gewonnen.

So kann etwa mit der "Modus Ponens"-Regel, die besagt, dass aus A und $A \supset B$ B gewonnen werden kann, aus (6) und (7) die Aussage

(8) Fehlbar (Fritz)

gewonnen werden. Ein PK-System inferiert daher eine bestimmte Aussage, indem es diese Aussage aufgrund seines Wissens beweist. Die Aufgabe eines solchen Systems liegt daher darin, den Beweis einer Aussage zu finden und es kann gezeigt werden, dass es für jede wahre Aussage (d.h. mit dem Wissen vereinbare Aussage) schliesslich einmal einen solchen Beweis finden muss. Die verschiedenen beweisenden Systeme unterscheiden sich aber stark im Aufwand, den sie zum Auffinden eines solchen Beweises treiben müssen. Gerade die mangelnde Effizienz der damals verwendeten Beweisverfahren führte anfangs der Siebzigerjahre zu einem Niedergang der Popularität des Prädikatenkalkülansatzes.

In neuerer Zeit sind aber ganz entscheidende Fortschritte in der Entwicklung effizienter Beweisverfahren gemacht worden (z.B. Bledsoe, 1977), die denn auch mit ein Grund sind, dass der Prädikatenkalkül einen neuerlichen Aufschwung in der Künstlichen Intelligenz erlebt (Bobrow, 1980; Pat Hayes, 1978a, 1978b; McDermott & Doyle, 1979; McDermott, 1977).

Ein anderer populärer Repräsentationsformalismus sind die von Newell und Simon entwickelten Produktionssysteme (Newell & Simon, 1972; Davis & King, 1977), die historisch aber bis auf Post (1943) zurückgehen. Ein einfaches Produktionssystem besteht aus einer linear geordneten Menge von Produktionsregeln der Form "Test - Aktion", einer Datenbasis, häufig Short term memory genannt, auf die die Produktionsregeln angewandt werden und einem Interpreter, der die Produktionsregeln anwendet. Der Interpreter sucht die Produktionsregeln linear ab, bis er eine Produktionsregel, deren Muster der Testseite in der Datenbasis vorhanden ist, findet.

Dann wird dieses Muster durch das Muster auf der Aktionsseite der Produktionsregel ersetzt. Diese Schritte führt nun der Interpretierer so lange durch, bis kein Test einer Produktionsregel mehr erfüllt ist.

Zwei der im letzten Kapitel besprochenen Parser waren Produktionssysteme, nämlich der Parser von Riesbeck und derjenige von Marcus. Bei Riesbeck heißen die Produktionsregeln Requests und die Datenbasis besteht aus dem laufenden Wort und der bis dahin konstruierten semantischen Repräsentation. Bei Marcus ist der Buffer die Datenbasis.

Diese einfachen (sogenannten "reinen") Produktionssysteme wurden im Laufe der Zeit immer mehr verallgemeinert. So wurden in allgemeineren Konfliktlösungsschemata neue Kriterien gegeben, nach denen eine Produktionsregel ausgelöst (angewandt) wird, falls mehrere Regeln anwendbar wären. Während bei den reinen Produktionssystemen die Regelpräzedenz durch die lineare Ordnung der Produktionsregeln gegeben ist, gibt es nun auch eine Allgemeinheitsordnung (die spezifischste Regel wird gewählt), Recencyordnung (die zuletzt benützte Regel oder die Regel, die sich auf zuletzt veränderte Daten bezieht, wird gewählt), Präzedenzordnung (den Regeln werden explizite Prioritäten gegeben wie im Marcusparser), Regelpakete (die Regeln werden in Paketen organisiert, die unabhängig voneinander aktiviert bzw. deaktiviert werden (wie im Marcusparser) usw.

Auch auf der Seite der Datenbasis wurden Erweiterungen gemacht. Während sie bei den reinen Produktionssystemen aus einer einfachen Liste von Items besteht, ist sie nun häufig sehr komplex strukturiert.

Die derzeitige Popularität der Produktionssysteme ist auf einige erfolgreiche Anwendungen auf den Gebieten der Spektralanalyse (DENRAL von Feigenbaum, Buchanan & Lederberg, 1971), der medizinischen Diagnostik (MYCIN von Shortliffe, 1976), der Mathematik (AM von Lenat, 1976) und anderen Gebieten zurückzuführen.

Neben diesen Anwendungen in Expertensystemen sind die Produktionssysteme aber auch als Instrument zur psychologischen Modellbildung populär geworden (Newell & Simon, 1972; Anderson, 1976).

In der Künstlichen Intelligenz werden immer wieder neue Programmiersprachen entwickelt, die die Ansprüche und die Flexibilität der Datenrepräsentation und der Kontrollstruktur besser erfüllen als die konventionellen Programmiersprachen. Prominente Vertreter dieser Sprachen sind PLANNER (Hewitt, 1972), MICROPLANNER (Sussman, Winograd & Charniak, 1971) und CONNIVER (McDermott & Sussman, 1972).

Aehnlich wie bei den Produktionssystemen wird in diesen Programmiersprachen eine Datenbasis manipuliert, die aus Behauptungen wie (AUF LAMPE TISCH) besteht. Anstelle von Produktionsregeln braucht man in PLANNER standardisierte Programme, genannt Theoreme, bei denen man drei wichtige Typen unterscheidet. Antezedenstheoreme wie (9) werden dann ausgeführt, wenn

(9) (ANTEZEDENS (X Y) ; X und Y sind Variablen
(AUF X Y) ; Muster des Theorems
(ERASE (LEER Y)) ; Auszuführender Teil des Theorems

eine ihrem Muster entsprechende Behauptung in der Datenbasis auftaucht.

Konsequenztheoreme wie (10) werden dann ausgeführt, wenn eine ihrem Muster

(10) (KONSEQUENS (X Y Z) ; X, Y und Z sind Variablen
(UEBER X Y) ; Muster des Theorems
(GOAL (UEBER X Z))
(GOAL (UEBER Z Y))) ; Auszuführender Teil des Theorems

entsprechende Behauptung in einem GOAL-Befehl steht, aber nicht in der Datenbasis vorhanden ist. ERASING-Theoreme werden dann aufgerufen, wenn eine ihrem Muster entsprechende Behauptung aus der Datenbasis entfernt wird.

Diese Programmiersprachen bieten etwas andere Möglichkeiten als die Produktionssysteme darin, wie die Theoreme (resp. Produktionsregeln) ausgewählt werden. Die Wahl des Theorems kann entweder völlig dem Interpreter überlassen werden, oder es können durch Empfehlungslisten Gruppen von Theoremen bis hin zu spezifischen Theoremen ausgewählt werden. Aehnlich wie bei den Produktionssystemen konnte auch die Datenbasis, die bei Microplanner noch einfach gehalten war, in den neueren Programmiersprachen immer komplexer werden.

Neben diesen Repräsentationsformalismen werden im folgenden noch zwei andere eingehend besprochen, nämlich die semantischen Netzwerke und der Framesansatz in einigen seiner Spielvarianten.

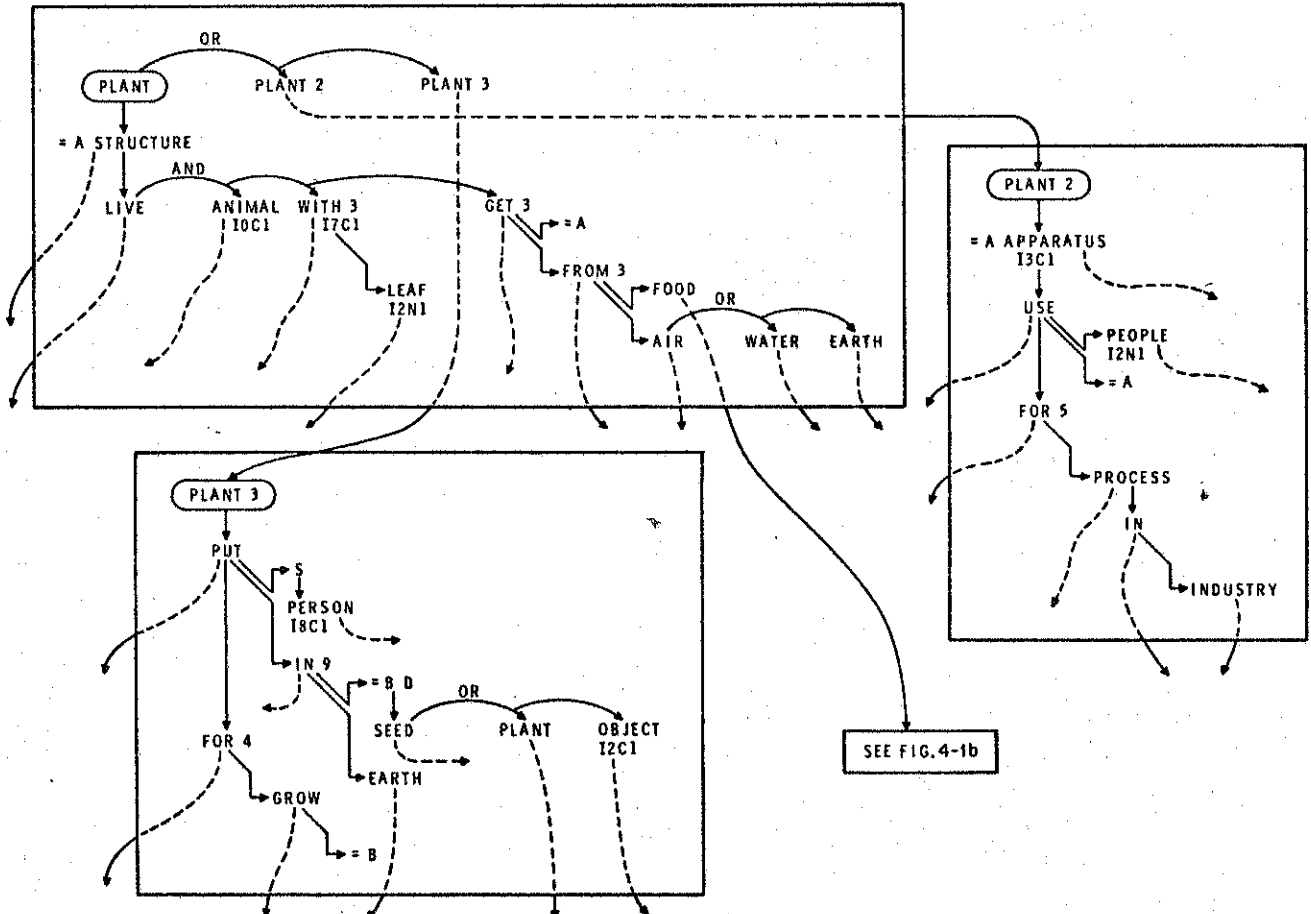
Diese verschiedenen Repräsentationsformalismen sind zum Teil oft mit bestimmten Schulen oder bestimmten Anwendungen verbunden gewesen. Ihre respektiven Vor- und Nachteile sind oft zum Objekt harter Auseinandersetzungen gewesen, wie etwa die Prozedural-Deklarativ-Kontroverse zwischen den Planner-Anhängern und den Prädikatenkalkülanhängern (um es etwas vereinfacht auszudrücken). Mit der fortschreitenden Entwicklung der Formalismen sind aber viele Unterschiede verschwunden und einige Unterschiede haben sich als irrelevant erwiesen. So hat man heute denn oft den Eindruck, dass viele der Unterschiede zwischen den Formalismen eigentlich nicht mehr als Notationsvarianten sind.

3. SEMANTISCHE NETZE

3.1. Erste Formen der semantischen Netze

Ein weiterer Formalismus zur Repräsentation von Wissen ist das semantische Netzwerk, dessen Entwicklung allgemein Quillian (1968, 1969) zugeschrieben wird, obwohl bereits frühere Arbeiten (Lindsay, 1963; Raphael, 1968) entscheidende Aspekte vorweggenommen haben. Dieser Formalismus sollte nach Quillian ermöglichen, die Definition von Wörtern zu repräsentieren und ein "humanlike use of those meanings" ermöglichen (Quillian, 1968, S. 227). Im einfachsten Fall ist ein semantisches Netz ein gerichteter Graph, in dem die Knoten Konzepte und die gerichteten, etikettierten Kanten (Zeiger) Relationen zwischen diesen Konzepten repräsentieren. Die von einem Konzept ausgehenden Zeiger zeigen auf andere Konzepte, die zusammen die Definition (Bedeutung) dieses Konzeptes ausmachen. So kann in Quillians Netz jeder Konzeptknoten als Kopf einer Ebene betrachtet werden, die seine Definition enthält. Abbildung (11) aus (Quillian, 1968, S. 236) zeigt drei solche Ebenen für die drei Bedeutungen von "plant".

- PLANT. 1. Living structure which is not an animal, frequently with leaves, getting its food from air, water, earth.
 2. Apparatus used for any process in industry.
 3. Put (seed, plant, etc.) in earth for growth.



Die ausgezogenen Zeiger innerhalb der Ebenen formen die Struktur der Bedeutung des Kopfkongzeptes (PLANT1, PLANT2 und PLANT3). So besagt etwa die Ebene von PLANT1, dass PLANT1 eine lebende Struktur ist, die kein Tier ist, häufig Blätter hat und ihre Nahrung aus der Luft, dem Wasser und der Erde bezieht. Quillian braucht eine kleine Menge von Etiketten für die Zeiger innerhalb der Ebenen, nämlich Untermenge (der Zeiger von PLANT1 nach STRUCTURE), Modifikation (APPARATUS wird modifiziert durch USE), Disjunktion (OR in der Ebene von PLANT1), Konjunktion (AND in der Ebene von PLANT1) und Subjekt/Objekt (in der Ebene von PLANT3 ist as Subjekt S mit dem Objekt IN9 durch die Relation PUT verbunden).

Die gestrichelten Zeiger aus der Ebene heraus zeigen von einer spezifischen Instanz (Token) eines Wortes auf sein Allgemeinkongzept (Type), das Kopf einer anderen Ebene ist.

Die Länge des Pfades zwischen zwei Kongzepten, auch als semantische Distanz bezeichnet, kann als Indikator für die inhaltliche Nähe dieser beiden Kongzpte verwendet werden und, wenigstens beschränkt, für Wortdesambiguierungen verwendet werden unter der Annahme, dass in einem Satz inhaltlich nahe verwandte Wörter eher zusammen auftreten als inhaltlich entfernt verwandte.

Eine weitere wichtige Eigenschaft des Quillianschen Gedächtnismodells ist die Art und Weise, wie Eigenschaften von Kongzepten abgespeichert werden.

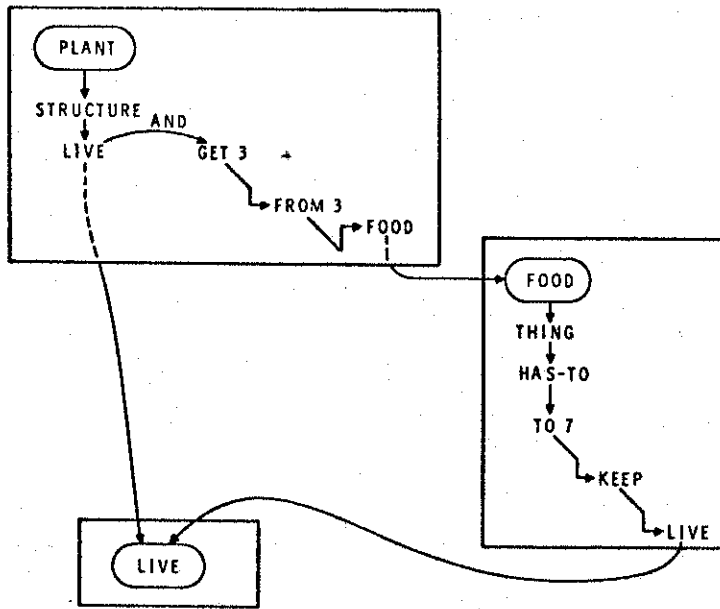
Es wird angenommen, dass eine mit einem Kongzept assoziierte Eigenschaft auch für alle Untermengen dieses Kongzeptes gilt, also zum Beispiel alle Eigenschaften von "Tier" auch für "Vogel" gelten. Deshalb wird eine Eigenschaft nur mit dem oberstmöglichen Kongzept in einer Ober/Untermengenhierarchie assoziiert und für alle Untermengen deduziert.

Quillians Idee des semantischen Netzwerkes wurde von verschiedenen Forschern aufgenommen und weiterentwickelt, deren profilierte Vertreter die Gruppe Lindsay, Norman und Rumelhart waren (Rumelhart, Lindsay und Norman, 72). Ihr Hauptbeitrag in der Entwicklung liegt in zwei Aspekten, erstens in der Kombination Fillmorescher Kasusgrammatik mit der SN Idee, und zweitens in der einheitlichen Repräsentation deklarativer Wissensrepräsentation à la Quillian mit prozeduraler Wissensrepräsentation in Programmen.

Mit Hilfe eines solchen semantischen Netzes ist es Quillians Programm möglich, gewisse Inferenzen zu ziehen und z.B. den inhaltlichen Zusammenhang zwischen Wörtern anzugeben oder auch beschränkt Wortdesambiguierungen vorzunehmen. Die erste Aufgabe wird durch die sogenannte

"Durchschnittssuche" gelöst, d.h. von beiden Konzepten aus wird gleichzeitig allen Zeigern gefolgt, bis sich die beiden Mengen der so erreichten Knoten überschneiden. Der Pfad, der die beiden Konzepte über diesen gemeinsamen Knoten verbindet, spezifiziert dann den einfachsten ihrer Zusammenhänge. Abbildung (12) aus (Quillian, 1968, S. 250) illustriert, welcher Zusammenhang so zwischen "plant" und "food" gefunden wird, nämlich, dass (a) Pflanzen Nahrung aus der Erde beziehen und (b), dass Pflanzen Lebewesen sind und Nahrung von Lebewesen eingenommen werden muss.

(12)



Die Länge des Pfades zwischen zwei Konzepten, in der Literatur als semantische Distanz bezeichnet, kann etwas über die inhaltliche Nähe dieser beiden Konzepte aussagen und so können etwa in einem Satz Wörter aufgrund ihrer semantischen Distanz zu den andern Wörtern desambiguiert werden. Die inferentiellen Fähigkeiten von Quillians Programm liegen darin, dass Eigenschaften von Konzepten auch aus Obermengenkonzepten hergeleitet werden können (inheritance of properties), die es erübrigen, dass diese Eigenschaft bei allen untergeordneten Konzepten wiederholt werden muss.

Die Idee des semantischen Netzwerkes wurde von verschiedenen Autoren übernommen und in verschiedenen Richtungen weiterentwickelt. Eine dieser Weiterentwicklungen stammt von der Gruppe Lindsay, Norman und Rumelhart (Rumelhart, Lindsay & Norman, 1972; Norman & Rumelhart, 1975). In ihrer Arbeit sind drei Aspekte von besonderer Wichtigkeit.

1. Sie haben die Repräsentation von Verben und Prädikaten systematisiert, indem sie die Idee des semantischen Netzes mit der Fillmoreschen Kasusgrammatik kombiniert haben. Prädikate werden dabei nicht mehr als Relationen zwischen Knoten, sondern als Konzepte repräsentiert und die Argumente dieser Prädikate durch Kasuslinks im Sinne Fillmores verbunden. Dadurch wird die Repräsentation von Objekten und Prädikaten im semantischen Netz vereinheitlicht. Damit ist es auch möglich, Prädikate gleich wie Objekte zu beschreiben, und die Type/Token Unterscheidung ist auch auf Prädikate anwendbar. (Allerdings ist dazu beizufügen, dass die Type/Token Unterscheidung von Norman & Rumelhart nicht identisch ist mit derjenigen von Quillian. Während sie bei Quillian der Unterscheidung zwischen allgemeinem Konzept und spezifischer Instanz entspricht, entspricht sie bei Norman & Rumelhart der Unterscheidung zwischen einer Klasse von Objekten und einem Element aus dieser Klasse).
2. Die zweite Erweiterung des LNR-Netzwerkes liegt darin, dass nicht nur Definitionen von Konzepten (Prädikaten und Objekten) möglich ist, sondern auch von Ereignissen und Episoden, d.h. Ketten von Ereignissen, die durch zeitliche Relationen miteinander verbunden sind.
3. Im LNR-Netz kann nicht nur deklarative Information repräsentiert werden wie im Quillianschen Netz, sondern auch prozedurale Information, d.h. ausführbare Prozeduren sind ein integraler Bestandteil ihres semantischen Netzes. So können etwa neue Operationen oder Inferenzregeln im semantischen Netz definiert und dann angewendet werden (siehe Norman & Rumelhart, 1975, S. 165 ff).

Der Hauptbeitrag von Norman & Rumelhart zu den semantischen Netzwerken liegt aber eindeutig darin, dass sie gezeigt haben, dass semantische Netze nicht nur zur Repräsentation von Sprache geeignet sind, sondern ebenso auch für die Repräsentation von Prozessen (Scragg, 1975), visuelle Wahrnehmung (Palmer, 1975), Problemlösen (Eisenstadt & Kareev, 1975) usw.

Einen ähnlichen Ansatz wie LNR hat auch Simmons (1973) entwickelt. Sein Ansatz ist weniger deshalb interessant, weil er neue grundlegende Beiträge zur Idee des semantischen Netzes geliefert hat, als darin, dass er semantische Netze in einem integrierten sprachverarbeitenden System eingesetzt hat, einem System, das sowohl einen (ATN-) Parser, als auch einen (ATN-) Generator enthält.

Ein weiterer interessanter Ansatz innerhalb der semantischen Netzwerke findet sich auch im SCHOLAR-Programm von Carbonell (1970), der ein weiterentwickeltes Quillian'sches Netz verwendet. In diesem Programm wird ein Inferenztyp verwendet, den man am ehesten als "Selbstwissen"-Inferenz bezeichnen könnte. Das CAI-Programm kennt unter anderem die wichtigsten Fakten über die Industrieproduktion verschiedener Länder. So kann man zum Beispiel das Programm fragen, ob die Oelproduktion ein wichtiger Industriezweig von Frankreich sei. Eine solche "Selbstwissen"-Inferenzregel besagt nun, dass dies nicht der Fall ist, weil Oelproduktion nicht unter den Industriezweigen von Frankreich aufgeführt ist, das Programm aber alle wichtigen Industriezweige kennt.

Diese semantischen Netzwerke der ersten Generation bildeten die Grundlage für Weiterentwicklungen sowohl in der AI als auch in der Psychologie. Es ist nun interessant zu vergleichen, wie sich die Weiterentwicklungen in beiden Disziplinen voneinander unterscheiden. In der AI standen zwei Ziele im Vordergrund: 1. Die semantischen Netzwerke zu einem adäquaten Repräsentationsformalismus weiterzuentwickeln und 2. die Semantik der semantischen Netzwerke selbst zu klären. Die Entwicklungen in der Psychologie andererseits basieren auf den Experimenten von Collins & Quillian (1969), auf die weiter unten eingegangen wird, und beruhen darauf, alle in diesem experimentellen Paradigma gefundenen experimentellen Effekte zu erklären. Wir wollen im folgenden die Arbeiten in beiden Disziplinen näher beschreiben und sie dann miteinander vergleichen.

3.2. Weiterentwicklungen in der Künstlichen Intelligenz

Eines der hartnäckigsten Probleme der semantischen Netzwerke war die Quantifizierung und im besonderen die Möglichkeit, den Gültigkeitsbereich von Quantoren im Netz auszudrücken. Dieses Problem ist insofern interessant, als das semantische Netzwerk nicht als adäquater Repräsentationsformalismus betrachtet werden kann, solange dieses Problem nicht gelöst ist. Dies soll an einem Beispiel illustriert werden. Der Satz

(13) Alle Schweizer besitzen eine Uhr.

hat zwei Lesungen, nämlich

(14) $\forall x (\text{Schweizer}(x) \supset \exists y (\text{Uhr}(y) \ \& \ \text{besitzen}(x,y)))$

d.h. jeder Schweizer besitzt seine Uhr, und

(15) $\text{Uhr}(y) \ \& \ \forall x (\text{Schweizer}(x) \supset \text{besitzen}(x,y))$

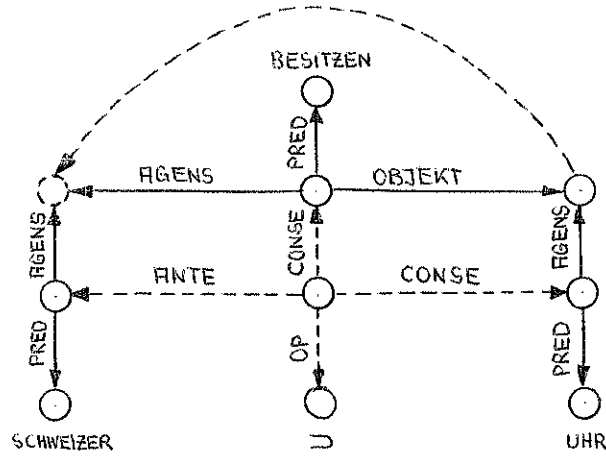
d.h. es gibt eine Uhr, die allen Schweizern gehört. Die beiden Lesungen (14) und (15) unterscheiden sich durch unterschiedlichen Gültigkeitsbereich der Quantoren und in keinem Netz der ersten Generation war es möglich, diesen Unterschied auszudrücken. Sie sind also insofern ein inadäquater Repräsentationsformalismus als sie zwei verschieden Propositionen nicht verschieden repräsentieren.

Erst Schubert (1976) hat eine Lösung zu diesem Problem gefunden. Sie besteht, kurz gesagt, darin, dass Prädikatenkalkülformeln in die sogenannte Skolemnormalform umgeschrieben werden, die sich dann in einem semantischen Netz repräsentieren lässt. In der Skolemnormalform werden alle existentiellen Quantoren durch eine Skolemfunktion ersetzt. Was eine Skolemfunktion ist, kann in (14) illustriert werden. Dort ist in anderen Worten ausgedrückt, dass man mit jedem Schweizer die Uhr assoziieren kann, die er besitzt. So wird mit einer Skolemfunktion, die wir Des-Schweizers-Uhr bezeichnen wollen, jedem Schweizer seine Uhr zugeordnet und (14) kann nun in der Skolemnormalform wiedergeschrieben werden als

(16) $\forall x (\text{Schweizer}(x) \supset (\text{Uhr}(\text{Des-Schweizers-Uhr}(x)) \ \& \ \text{besitzen}(x, \text{Des-Schweizers-Uhr}(x))))$

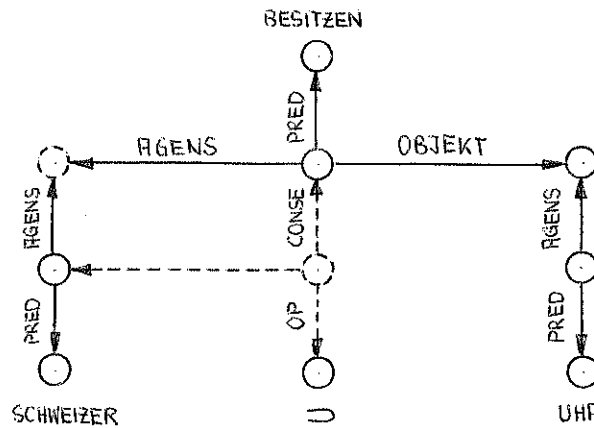
(16) könnte umschrieben werden als "Wenn x ein Schweizer ist, dann hat die Funktion Des-Schweizers-Uhr als Wert diejenige Uhr, die x besitzt". Die Repräsentation von (16) in einem semantischen Netz ist in (17) gegeben. Darin werden alle universell quantifizierten Knoten (alle Schweizer) als gestrichelte Knoten gezeichnet. Die in der Skolemfunktion ausgedrückte Abhängigkeit zwischen jeder Uhr und "ihrem" Schweizer wird durch einen gestrichelten Zeiger repräsentiert.

(17)



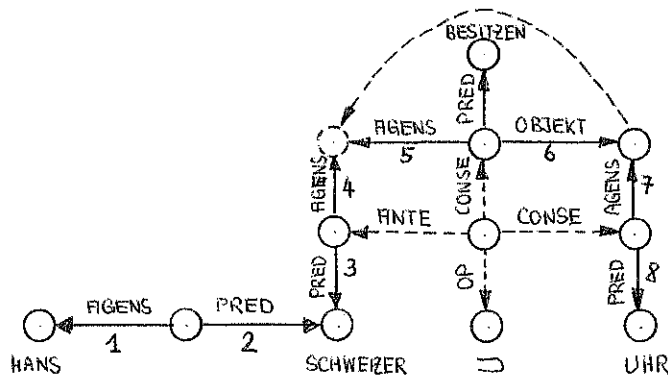
Entsprechend sieht die Repräsentation von (15) im semantischen Netz folgendermassen aus:

(18)



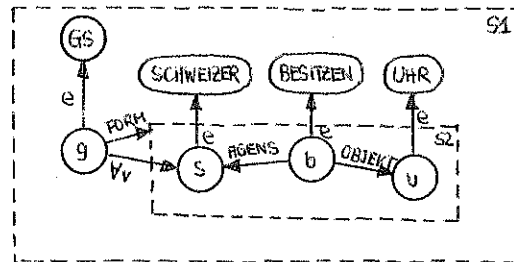
Nun ist wohl damit das Problem der Quantifizierung in semantischen Netzen gelöst, aber andererseits ist die Anzahl von Knoten und Zeigern zur Repräsentation einer Proposition gegenüber den Netzen der ersten Generation stark angestiegen. So muss zum Beispiel zur Beantwortung der Frage, ob der Schweizer Hans eine Uhr besitzt, nicht weniger als acht Zeigern gefolgt werden, wie dies in Abbildung (19) illustriert ist.

(19)

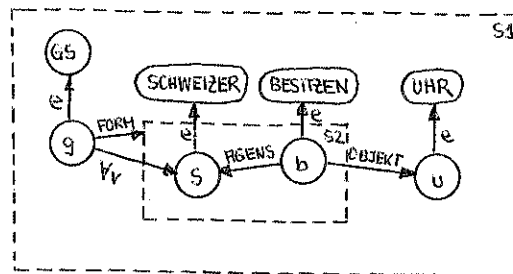


Ein anderer Mechanismus zur Lösung des Quantifikationsproblems wurde von Hendrix (1975) vorgeschlagen, nämlich das sogenannte "Network partitioning", bei dem alle Knoten eines Netzes in disjunktive Klassen, genannt "Ebenen" eingeteilt werden. So gehören etwa in Abbildung (20) die Knoten s, b und u in die Ebene S2, alle anderen in die Ebene S1. Eine der Anwendungen des "Network partitioning" ist nun, dass damit der Gültigkeitsbereich von Quantoren angegeben werden kann. Ohne auf die Details der Darstellung eingehen zu wollen, zeigen wir die beiden Repräsentationen (20) und (21), die den Interpretationen (14) respektive (15) entsprechen.

(20)



(21)



Bei der Präsentation dieser beiden Arbeiten ging es darum, zu zeigen, dass die semantischen Netze der ersten Generation bedeutend verändert werden mussten, um das Quantifikationsproblem zu lösen oder allgemeiner, um sie zu einem adäquaten Repräsentationsformalismus weiterzuentwickeln. Im Falle von Schuberts Lösung musste die Annahme fallengelassen werden, dass zusammenhängende Knoten durch sehr wenige Links miteinander verbunden sind, im Falle von Hendrix' Lösung wurde ein ganz neues Konzept, das des "Network partitioning" eingeführt.

Am Beispiel des Quantifikationsproblems wurde gezeigt, warum die Netzwerke der ersten Generation kein adäquater Repräsentationsformalismus sind. Ein für die Weiterentwicklung der semantischen Netze noch viel wichtigerer Fragenkomplex ist aber die Frage, welche Entitäten in diesen

Netzen gebraucht werden müssen und was ihre Semantik ist. Wir wollen nun kurz auf einige dieser Probleme eingehen und zeigen, wie sie gelöst wurden. Dabei zeigt sich, dass der einfache Formalismus der frühen semantischen Netzwerke auch in anderer Hinsicht nicht genügte.

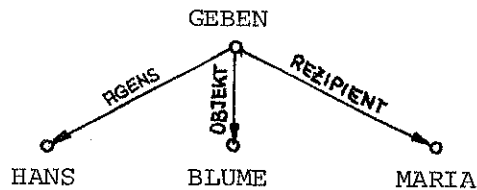
Ziemlich früh wurde erkannt, dass es nicht genügt, Objekte als Knoten und Prädikate als Links zu repräsentieren, wie in Abbildung (22).

(22)



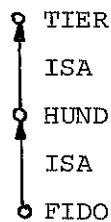
Es genügt nämlich dort nicht, wo ein Prädikat mehr als zwei Argumente nimmt, wie etwa die Relation "geben", die ein Agens, ein Objekt und einen Rezipienten nimmt. Man ist dann dazu übergegangen, auch Prädikate als Knoten zu repräsentieren und die Argumente mit Hilfe von Kasuszeigern mit dem Prädikatknoten zu verbinden (23).

(23)



Aus dieser notwendigen Erweiterung folgt, dass im semantischen Netz Objekte und Prädikate gleich repräsentiert werden. Eine nächste Erweiterung liegt darin, dass man zwischen verschiedenen Knotentypen hat unterscheiden müssen. Die erste Unterscheidung ist die Type/Token-Unterscheidung von Quillian, nämlich die Unterscheidung zwischen Individuum und allgemeinem Konzept. Eine andere Unterscheidung ist die Type/Token-Unterscheidung von Norman & Rumelhart, nämlich die Unterscheidung zwischen Individuum und Klasse. Die Type/Token-Unterscheidung wurde üblicherweise durch den Zeiger "ISA" indiziert, in dem Sinne, als ISA von einem Tokenknoten auf einen Typeknoten zeigt. Anhand dreier Netzfragmente wollen wir nun zeigen, dass wir sowohl die Quilliansche als auch die Norman & Rumelhart-Type/Token-Unterscheidung brauchen (24a, b, c).

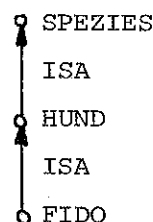
(24)



(a)



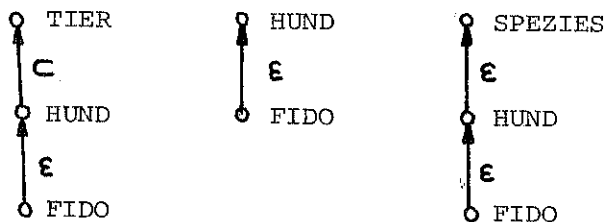
(b)



(c)

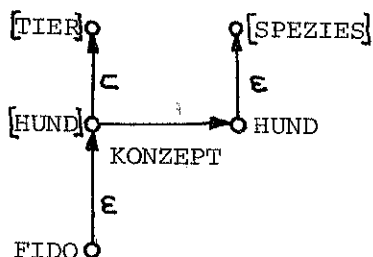
Während uns (24a) korrekt erscheint, ist (24b) daher falsch, weil "Fido" kein Type sein kann. Das kann gelöst werden, wenn die Type/Token-Unterscheidung von Norman & Rumelhart übernommen wird und "ISA" durch zwei Zeiger, nämlich "ε" (Element) und "⊂" (Teilmenge) ersetzt wird wie in Abbildung (25) illustriert ist.

(25)



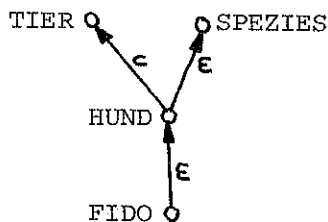
Mit dieser Unterscheidung ist es dann klar, warum (24) illegal ist. "Fido" ist ein spezifisches Individuum und keine Menge, und daher darf keine Relation ε auf Fido zeigen. Dieser neuen Konvention widerspricht aber (25c), obwohl es uns korrekt erscheint. Zur Lösung dieses Problems gibt es zwei Möglichkeiten. Entweder unterscheidet man wie Anderson (1976) zwischen Klasse und Konzept. Demnach wäre "Fido" Element der Klasse "Hund" und das Konzept "Hund" Element der Klasse "Spezies" wie in Abbildung (26).

(26)



Man kann auch nach Leveque & Myopoulos (1979) zwischen Klassen und Metaklassen, d.h. Klassen von Klassen, unterscheiden. Demnach wäre "Fido" Element der Klasse "Hund" und die Klasse "Hund" Element der Metaklasse "Spezies" wie in Abbildung (27), in der "Fido" ein Individuum, "Hund" und "Tier"

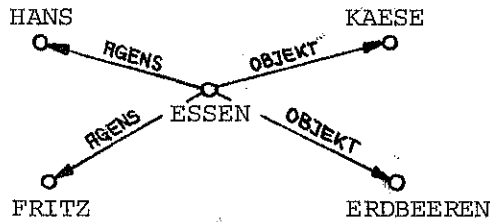
(27)



Klassen und "Spezies" eine Metaklasse sind.

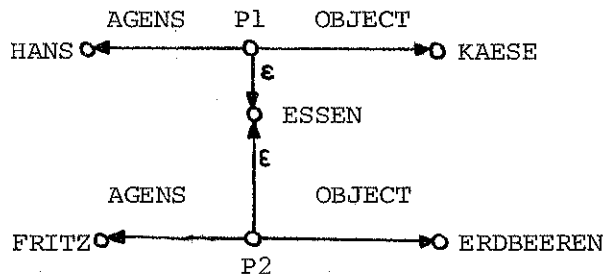
Wir haben vorher gesehen, dass Prädikate und Objekte in semantischen Netzwerken gleich repräsentiert werden. Demnach würde man auch bei Prädikaten zwischen Type und Token unterscheiden. Man kann zeigen, dass man dies in der Tat tun muss. Angenommen, wir würden Prädikate ohne Type/Token-Unterscheidung wie in (23) repräsentieren und wir wollten die beiden Propositionen "Hans isst Käse" und "Fritz isst Erdbeeren" repräsentieren. Das resultierende Netzwerk ist in Abbildung (28) gezeigt.

(28)



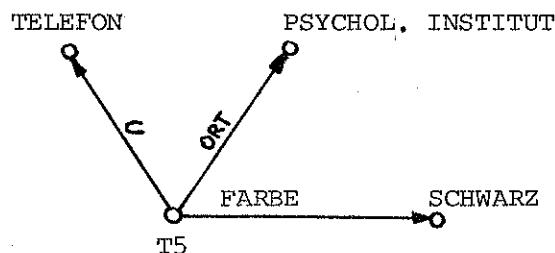
Darin kann man aber nicht mehr entscheiden, ob Hans Käse oder Erdbeeren gegessen hat, weil nicht angegeben ist, welche Agens-Objekt Paare zusammengehören. Dieses Problem kann mit der Einführung der Type/Token-Unterscheidung für Prädikate gelöst werden und die neue Version von (28) sähe dann folgendermassen aus:

(29)



Nicht nur bei den Knoten, auch bei den Attributen müssen einige Unterscheidungen eingeführt werden. Das Netzwerkfragment (30) kann auf mindestens zwei verschiedene Arten interpretiert werden. Entweder repräsentiert der Knoten T5 die Menge der schwarzen Telefone im Psychologischen Institut oder er repräsentiert die Proposition "Alle Telefone im Psychologischen Institut sind schwarz".

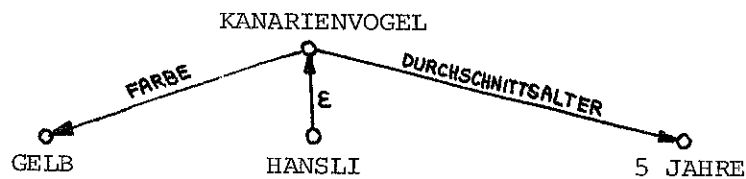
(30)



Der Unterschied liegt darin, dass in den beiden Interpretationen das Attribut "Farbe" verschieden interpretiert wird. In der ersten Interpretation ist das Attribut "Farbe" wie die beiden andern Attribute ein Attribut, das den Knoten T5 spezifiziert. Würden wir zum Beispiel den Wert dieses Attributs auf "Weiss" ändern, so würde T5 nicht mehr die Menge der schwarzen Telefone des Psychologischen Institutes repräsentieren, sondern die Menge der weissen Telefone des psychologischen Institutes. In der zweiten Interpretation hingegen sind nur die Attribute Teilmenge und Ort Attribute, die das Konzept T5 bestimmen, d.h. T5 repräsentiert die Menge der Telefone des Psychologischen Instituts. Würden wir den Wert des Attributs "Farbe" auf "Weiss" ändern, würde die Interpretation des Knotens T5 nicht ändern. Attribute, die den Inhalt eines Konzeptes spezifizieren, werden in der Literatur als strukturelle Attribute bezeichnet (Woods, 1975), die andern Attribute werden als assertive Attribute bezeichnet.

Im weiteren muss zwischen Attributen unterschieden werden, die Individuen modifizieren und solchen, die Klassen modifizieren. So ist in Abbildung (31)

(31)



Das Attribut "Farbe" so zu interpretieren, dass alle Instanzen der Klasse "Kanarienvogel" gelb sind, oder im Prädikatenkalkül ausgedrückt:

$$(32) \quad \forall x(\text{Instanz}(x, \text{KANARIENVOGEL}) \supset \text{Farbe}(x, \text{GELB}))$$

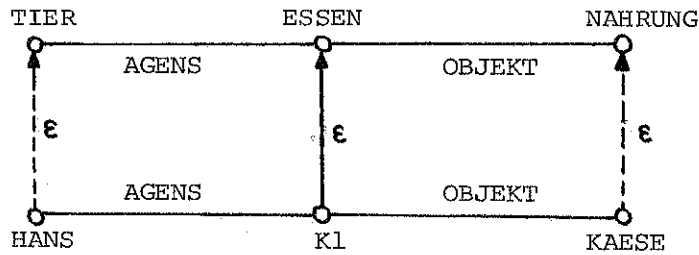
Auf der andern Seite modifiziert das Attribut "Durchschnittsalter" die Klasse "Kanarienvogel" und muss wie (33) interpretiert werden.

$$(33) \quad \text{Durchschnittsalter}(\text{KANARIENVOGEL}, \text{5JAHRE})$$

Es ist aber nicht der Fall, dass ein bestimmter Kanarienvogel, zum Beispiel Hansli, dieses Attribut ererbt.

Angenommen, wir definieren ein Prädikat "Essen" mit den beiden Argumenten AGENS und OBJEKT. Wir wollen nun angeben, dass die Menge der möglichen Objekte auf Nahrungsmittel beschränkt sein müssen. Es zeigt sich, dass dies leicht in den bisher entwickelten Formalismus eingebaut werden kann (Abbildung (34)).

(34)



In Abbildung (34) werden die beiden Attribute und OBJEKT als Rollen bezeichnet, die beiden Knoten TIER und NAHRUNG als Rollenbeschränkungen (oder Wertbeschränkungen), die beiden Knoten HANS und KAESE sind Rollenfüller (Role filler) und der Knoten Kl repräsentiert die Proposition "Hans isst Käse". Die beiden gestrichelten Elementzeiger illustrieren, wie die Rollenbeschränkungen zu interpretieren sind, nämlich der AGENS jedes Tokens von ESSEN muss Element der Klasse TIER sein und entsprechend muss jedes OBJEKT Element der Klasse NAHRUNG sein.

Es konnte nur angedeutet werden, wie die semantischen Netze der ersten Generation in der AI weiterentwickelt wurden, um damit einen adäquaten Repräsentationsformalismus mit präziser Semantik zu erhalten. Wir haben dabei verschiedene Aspekte ignorieren müssen, wie zum Beispiel die Anwendung der semantischen Netzwerke zur Repräsentation komplexer Wissensinhalte oder wie die semantischen Netzwerke in den Arbeiten von Phil Hayes (1976, 1977a, 1977b), Hendrix (1975, 1976), Brachman (1978) oder Levesque & Mylopoulos (1979) weiterentwickelt wurden. Ebenso wenig sind wir auf die neueren Anwendungen dieses Formalismus eingegangen.

Es ging uns in dieser Diskussion vor allem um zwei Dinge. Erstens wollten wir zeigen, dass semantische Netzwerke sich nicht prinzipiell von anderen Formalismen unterscheiden. So hat Schubert (1976) gezeigt, dass es eine eineindeutige Abbildung vom Prädikatenkalkül in semantische Netzwerke geben kann. Zweitens wollten wir zeigen, inwiefern die semantischen Netze der ersten Generation zu einfach konzipiert waren und wie sie erweitert werden mussten, um einen adäquaten und eindeutigen Formalismus zu erhalten.

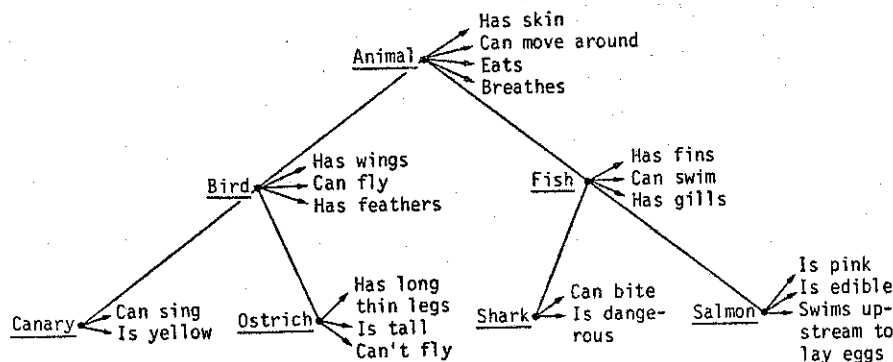
Für uns ist diese Weiterentwicklung insofern interessant, als die Weiterentwicklungen der semantischen Netzwerke in der Psychologie ebenfalls in den semantischen Netzwerken der ersten Generation ihren Ausgangspunkt genommen hat. Interessanterweise hat die Psychologie am ursprünglichen Formalismus kaum Veränderungen vorgenommen, wie weiter unten noch

illustriert wird, obwohl in der AI gezeigt wurde, dass der ursprüngliche Formalismus zur Wissensrepräsentation inadäquat ist.

3.3. Weiterentwicklungen in der Psychologie

Die semantischen Netzwerke sind in der experimentellen Psychologie zum ersten Mal aufgetaucht in einer Arbeit von Collins & Quillian (1969). In dem dort beschriebenen Experiment mussten die Versuchspersonen Sätze wie "Ein Kanarienvogel kann fliegen" oder "Ein Kanarienvogel ist ein Tier" nach ihrem Wahrheitsgehalt beurteilen und es wurde die Reaktionszeit zwischen der (visuellen) Präsentation des Satzes und dem Urteil gemessen. Zur Erklärung der Reaktionszeitenunterschiede verwendeten Collins & Quillian eine vereinfachte Version von Quillians Netzwerk (Abbildung 35).

(35)

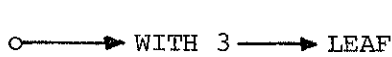


Von Bedeutung für die Erklärung der Reaktionszeitenunterschiede sind vier Annahmen. Drei davon beziehen sich auf die Repräsentationsstruktur und eine auf den Suchprozess. Während die Annahmen 1, 2 und 4 bereits im Quillianschen Modell enthalten sind, musste Annahme 3 neu hinzugenommen werden.

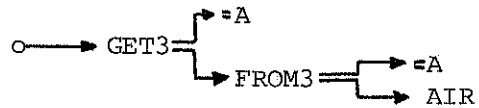
1. Hierarchie-Annahme: Es wird angenommen, dass Konzepte hierarchisch in Obermengenketten (ISA-Ketten) organisiert sind. So ist "Vogel" die Obermenge von "Kanarienvogel" und diejenige von "Vogel" ist "Tier". Um nun zu entscheiden, dass ein Kanarienvogel ein Vogel ist, muss nur einem Zeiger gefolgt werden, während für den Satz "Ein Kanarienvogel ist ein Tier" zwei sukzessiven Zeigern gefolgt werden muss. Daher, so sagen Collins & Quillian voraus, wird die Reaktionszeit für den zweiten Satz länger sein.

2. Oekonomie-Annahme: Es wird angenommen, dass Eigenschaften mit dem allgemeinsten Konzept verbunden sind, auf das sie zutreffen. So ist die Eigenschaft "can sing" mit Kanarienvogel verbunden, während die Eigenschaft "can fly" mit Vogel verbunden ist. Die Reaktionszeit für die zweite Eigenschaft wird daher länger sein, weil zusätzlich einem ISA-Zeiger gefolgt werden muss.

3. Attribut-Enkodierungs-Annahme: Es wird angenommen, dass alle Attribute eines Konzeptes in einem oder gleich vielen Schritten erreicht werden können und uniform enkodiert werden. Dies ist gegenüber Quillian (1968) insofern eine vereinfachende Annahme, als Quillian Attribute gerade nicht einheitlich kodiert, wie anhand der beiden Attribute "with leaves" und "gets food from air" (in der Definition von PLANT1, Quillian (1968), S. 236) illustriert werden kann (Abbildung 36a und 36b):



(36a)

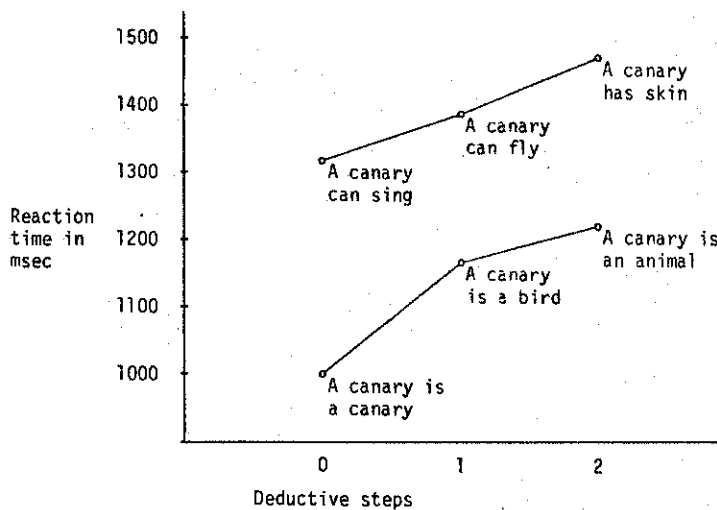


(36b)

4. Annahme der parallelen Suche: Es wird angenommen, dass in der Gedächtnissuche zwischen zwei Konzepten parallel allen Verbindungen gefolgt wird (in der Literatur als "spreading activation" bezeichnet, Collins & Loftus, 1975).

Die Voraussagen von Collins & Quillian scheinen in der Tat durch die Daten bemerkenswert gut bestätigt zu werden (Abbildung 37).

(37)



Als einziger Ausreisser kann die Reaktionszeit für den Satz "A canary is a canary" bezeichnet werden, für den sie annahm, dass er eher durch einen "Pattern matching"-Prozess als durch einen Deduktionsprozess entschieden werde.

In der Folge von Collins & Quillians Experiment wurde eine ganze Reihe von Untersuchungen durchgeführt, die das experimentelle Paradigma von Collins & Quillian verwendeten. Aus der Reihe der früheren Arbeiten wollen wir einige auswählen, die Zweifel an der Validität von Collins & Quillians Annahmen aufkommen liessen und werden dann kurz auf einige Alternativmodelle eingehen.

1. Es ist problematisch, von einer strikten (biologischen) Hierarchie auszugehen. So tendieren nämlich Versuchspersonen dazu, zum Beispiel "Hund" direkt als "Tier" zu klassifizieren, obwohl eine striktere Klassifikation "Hund-Säugetier-Tier" wäre. Entsprechend fanden auch Rips et al. (1973), dass die Reaktionszeit für "Ein Hund ist ein Tier" kürzer war als für "Ein Hund ist ein Säugetier".

2. Das Collins & Quillian Modell sagt voraus, dass die Reaktionszeiten für die Sätze "Ein Spatz ist ein Vogel" und "Eine Gans ist ein Vogel" gleich lang sind, da beide einen Deduktionsschritt zur Entscheidung brauchen. Wilkins (1971) fand, dass aber die Reaktionszeit für den ersten Satz kürzer war und allgemein, dass die Reaktionszeit für typischere Instanzen einer Kategorie (siehe dazu auch Smith et al. (1974), S. 218) kürzer war als diejenige für relativ untypische Instanzen. In einer Antwort auf frühe Experimente zur Ueberprüfung von Collins & Quillians Modell haben Collins & Loftus (1975) darauf hingewiesen, dass bereits in Quillians (1968) Modell verschiedene Links verschieden gut erreicht werden können (indiziert durch sogenannte "criteriality tags"). Damit sind die Ergebnisse von Wilkins mit dem ursprünglichen Modell von Quillian nicht vereinbar.

3. Conrad (1972) bat ihre Versuchspersonen in einem Vorexperiment Konzepte wie "Kanarienvogel" zu definieren und berechnete aus der Häufigkeit der produzierten Attribute einen Assoziationswert. In der Replikation des Collins & Quillian Experimentes fand sie dann, dass die Reaktionszeitenunterschiede vollständig durch die Assoziationsstärkenunterschiede erklärt werden konnten und dass die semantische Distanz im Sinne von Collins & Quillian keinen weiteren Einfluss hatte. In einer Analyse der Collins & Quillian Daten fand sie auch, dass alle Attribute mit höherer semantischer Distanz einen kleineren Assoziationswert hatten. In ihrer Replikation auf

Conrad haben Collins & Loftus (1975) die Annahme der ökonomischen Speicherung insofern abgeschwächt, als eine häufig verwendete Eigenschaft nicht nur mit dem allgemeinsten Konzept assoziiert, sondern auch mit spezifischen Instanzen (redundant) assoziiert werden kann.

Mit der Annahme der "schwachen kognitiven Oekonomie" einerseits und der unterschiedlichen Verfügbarkeit der Links andererseits wird die semantische Distanz zu einem abgeleiteten Konzept, d.h. die Ursache der Reaktionszeitenunterschiede kann nicht mehr eindeutig lokalisiert werden. Da sich Oekonomisierung und Verfügbarkeit der Links experimentell nicht mehr trennen lassen, muss die semantische Distanz neu operational definiert werden, wie dies zum Beispiel Rips et al. (1975) mit Hilfe von Ratings gemacht haben.

Das Collins & Quillian Modell ist noch mit einer ganzen Reihe von andern experimentellen Paradigmata untersucht worden, unter anderem mit dem Priming-Paradigma von Freedman & Loftus (1971). Darin mussten die Versuchspersonen Instanzen einer Kategorie produzieren, die mit einem bestimmten Buchstaben beginnen oder durch ein bestimmtes Adjektiv charakterisiert sind (z.B. Früchte, die mit "A" beginnen oder Früchte, die rot sind). Die Experimente sollten alle zeigen, dass das Collins & Quillian Modell in einer eingegengten Interpretation nicht haltbar ist. Collins & Loftus (1975) haben aber gezeigt, dass die Experimente mit der erweiterten Interpretation vereinbar sind. Sie haben auch gezeigt, dass die in der Folge von Collins & Quillian entwickelten Alternativmodelle wie das mengentheoretische Modell von Conrad oder das Attribut-Modell von Rips et al. (1973) als Spezialfälle von Quillians ursprünglichem Modell aufgefasst werden können.

So liegt denn das eigentliche Problem der semantischen Netzwerke als psychologische Modelle nicht darin, dass sie experimentelle Daten nicht erklären könnten. Das Problem liegt darin, dass man keine experimentellen Paradigmata gefunden hat, um entweder das Modell widerlegen zu können oder um es weiter differenzieren zu können. So sind denn auch seit 1975 keine wichtigen Arbeiten mehr erschienen, die zur Entwicklung der semantischen Netzwerke als psychologisches Modell beigetragen hätten. Die Zahl der Freiheitsgrade, die in diesem Modell noch enthalten sind, scheint zu gross, als dass sie innerhalb der Psychologie angemessen weiterentwickelt werden könnten.

V. ORGANISATION DES WISSENS

In diesem Kapitel wird vom Problem der kombinatorischen Explosion ausgegangen, dem Problem, dass der Aufwand zur Lösung eines Problems mit dem Anwachsen einer unorganisierten Wissensbasis exponentiell ansteigt. Dann wird eine kurze historische Uebersicht über den sogenannten Framesansatz gegeben, den bekanntesten Ansatz zur Organisation von Wissen. Schliesslich werden ausgehend von einer Diskussion dieses Ansatzes Grundprobleme und andere Ansätze der Wissensorganisation diskutiert.

1. KOMBINATORISCHE EXPLOSION

Wir wollen davon ausgehen, dass ein sprachverstehendes System über tausend verschiedene Inferenzregeln verfügt, eine Annahme, die ohne weiteres gerechtfertigt ist, da man schätzt, dass ein gutes sprachverstehendes System über Millionen von verschiedenen Fakten zu verfügen hat. Wenn wir nun weiter annehmen, dass zur Lösung eines bestimmten Problems, etwa eines Ambiguierungsproblems fünf Inferenzschritte benötigt werden, dann ergeben sich 1000^5 oder 10^{15} verschiedene Möglichkeiten, die Inferenzregeln anzuwenden, von denen meist nur eine Kombination zum Erfolg führt. Dieses rasche Anwachsen von Möglichkeiten nennt man "kombinatorische Explosion". Die kombinatorische Explosion ist ein in der künstlichen Intelligenz allgegenwärtiges Problem, nicht nur im Sprachverstehen, sondern auch etwa in den Gebieten des Spielens, des Theorembeweisens, des Sehens und des Problemlösens. Gerade etwa beim Schachspiel ist es ein Leichtes, zu zeigen, dass ein schachspielendes Programm unmöglich alle Kombinationen ausprobieren kann. Wenn man nämlich davon ausgeht, dass eine Schachpartie aus durchschnittlich 35 Zügen besteht und bei jedem Zug durchschnittlich 100 verschiedene Zugsmöglichkeiten bestehen, dann ergeben sich 35^{100} oder 2.5×10^{154} verschiedene Zugsmöglichkeiten. Ein Schachprogramm, das daher alle Züge ausprobiert und pro Zug nur eine Milliardstel Sekunde braucht, müsste daher 10^{138} Jahre rechnen.

Weniger drastisch, aber doch auch eindrucklich, hat sich das Problem der kombinatorischen Explosion in einer frühen Arbeit über die Verwendung von Alltagswissen im Sprachverstehen gezeigt, nämlich in der Arbeit von Rieger (1974, 1975).

Riegers Programm ist Teil des Programms MARGIE, das auf der Conceptual Dependency Theorie von Schank basiert (Schank, 1975) und mit dem Parser von Riesbeck und dem Generator von Goldman zusammenarbeitet. Riegers Teil enthält den Inferenzteil dieses Programms. Es basiert auf der Annahme, dass in einem Text immer nur ein kleiner Teil des gesamten Kontextes explizit ausgedrückt ist. Ein Ziel des Programms besteht daher darin, weitere Fakten über einen plausiblen Kontext von Sätzen zu inferieren. So schliesst denn sein Programm aus einem Eingabesatz wie (1)

(1) John sagte Maria, dass Bill ein Buch wolle.
eine grosse Menge von plausiblen Fakten, wie

- (2a) John glaubt, dass Bill ein Buch wolle.
- (2b) Maria weiss nun, dass Bill ein Buch will.
- (2c) Bill will ein Buch.
- (2d) Bill will ein Buch besitzen.
- (2e) Bill will wahrscheinlich ein Buch lesen.
- (2f) Bill will vielleicht die im Buch enthaltenen Ideen kennen.
- (2g) Bill wird sich vielleicht ein Buch verschaffen.
- (2h) John könnte Bill ein Buch geben.
- (2i) Maria könnte Bill ein Buch geben.
- (2k) John möchte vielleicht, dass Maria Bill ein Buch gibt.
- (2l) John und Maria waren kürzlich zusammen.

Durch das Inferieren plausibler Kontexte soll es dem Programm möglich werden, Zusammenhänge zwischen den Sätzen eines Textes zu finden. Diese sind dort gegeben, wo sich die Inferenzräume zweier Sätze "berühren", ganz ähnlich wie es schon bei Quillian für den Zusammenhang zweier Konzepte formuliert wurde. Dazu wird auf einen Eingabesatz hin reflexartig ein Inferenzprozess ausgelöst, in dem 25 verschiedene Inferenztypen angewandt werden, die aber in ihrer Richtung nicht beeinflusst werden. Zu diesen Inferenztypen gehören u.a. Resultatinferenzen (2b), Motivationsinferenzen (2k), Ermöglichungsinferenzen (2l), Funktionsinferenzen (2e) usw.

Rieger repräsentiert seine Inferenzregeln als LISP-Programme, die spontan ablaufen, sobald das ihrem Muster entsprechende Fragment eine Conceptual Dependency Struktur im Gedächtnis auftaucht (Sie funktionieren daher ähnlich wie PLANNER-Antezedenstheoreme). Sie können als niedere Inferenzen bezeichnet werden, in dem Sinne als sie mit primitiven Conceptual Dependency Fragmenten assoziiert sind und nicht mit höheren Konzepten wie "schwitzen", "Zirkus" oder gar ganzen Situationen. Riegers Inferenzen sind daher auch insofern interessant, als durch sie die Semantik der Schankschen Primitiva, bis anhin nur vage umschrieben, genauer festgelegt werden. Sie ergänzen jene Inferenzen der Conceptual Dependency Theorie, die als sicher gelten und daher bereits explizit in der Conceptual Dependency Repräsentation repräsentiert werden (wie etwa die Tatsache, dass das Objekt von "töten" stirbt).

So interessant und leistungsfähig Riegers Programm auch ist, es hat ein prinzipielles und mit seinem allgemeinen Ansatz kaum lösbares Problem. Er konzipierte den Inferenzprozess als spontan und ungerichteten Prozess auf einer tiefen Stufe. Dies führt unweigerlich zu einer kombinatorischen

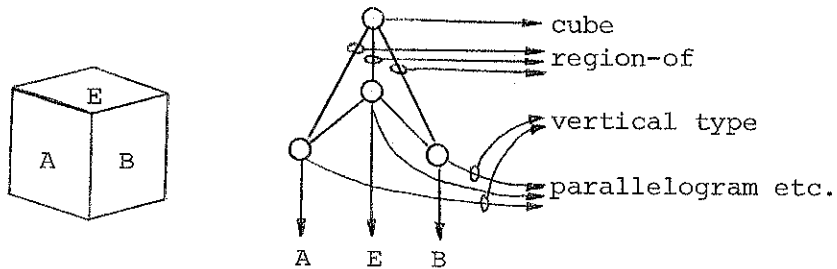
Explosion insbesondere da auch bereits einfache Inferenzen wie in (2) mehrere Inferenzschritte erfordern. Er konzipiert denn auch seinen Inferenzprozess als parallel ablaufenden Prozess (wie dies ja auch schon unter ähnlichen Voraussetzungen bei der Suche in semantischen Netzwerken gemacht wird). Dagegen sprechen aber mindestens zwei Argumente. Erstens ist unser heutiges Verständnis paralleler Prozesse noch nicht sehr tief und aus dieser Sicht wäre deshalb die Konzeption des Inferenzprozesses als serielltem Ablauf vorzuziehen. Zweitens ist es wohl plausibel, dass gewisse tiefe Prozesse wie etwa die periphere visuelle Verarbeitung parallel ablaufen, höhere Prozesse verlaufen aber wahrscheinlich kaum parallel (Wir können nicht gleichzeitig hundert Multiplikationen durchführen). Sprachliche Inferenzprozesse scheinen aber eher ein höherer Prozess zu sein und sollten daher seriell ablaufen. Wenn man aber einen Prozess seriell konzipiert, dann muss man einen Weg finden, um die kombinatorische Explosion zu verhindern oder mindestens abzuschwächen.

Auf eine kurze Formel gebracht, besteht nun das Ziel der Wissensorganisation darin, die kombinatorische Explosion so zu vermeiden, indem in einer bestimmten Problemsituation möglichst alle irrelevanten Inferenzregeln ausgeschaltet werden, während aber immer noch garantiert ist, dass die in dieser Situation notwendigen Inferenzregeln verfügbar bleiben. Die im folgenden diskutierten Ansätze zur Wissensorganisation versuchen, dies zu ermöglichen.

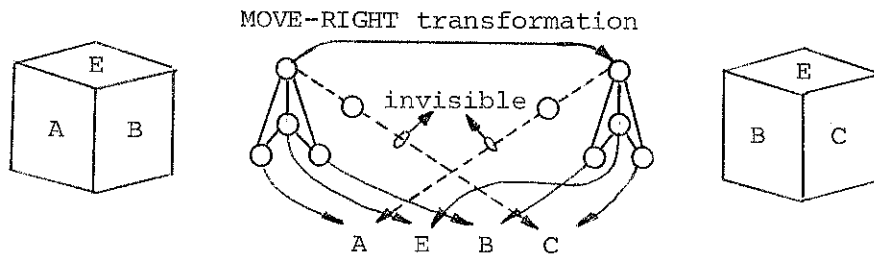
2. MINSKYS FRAMEANSATZ

Im Jahre 1974 veröffentlichte Minsky einen Artikel (Minsky, 1975), in dem er verschiedene Arbeiten aus den Gebieten des Sprachverstehens und des Sehens referierte und zeigte, dass sie im Grunde genommen alle an einem ähnlichen Problem arbeiteten: Wie hängen einzelne Wissensfakten zusammen und wie können diese Zusammenhänge mit in der Lösung eines bestimmten Problems angewendet werden. Ein solcher Zusammenhang besteht zum Beispiel in der Beschreibung von zwei verschiedenen Würfelansichten, wie sie in (3) und (4) (nach Minsky, 1975) gegeben sind. Wie kann das "Verschwinden" und "Wiederhervortreten" von Würfelseiten unter verschiedenen Rotationen einheitlich erfasst werden?

(3)

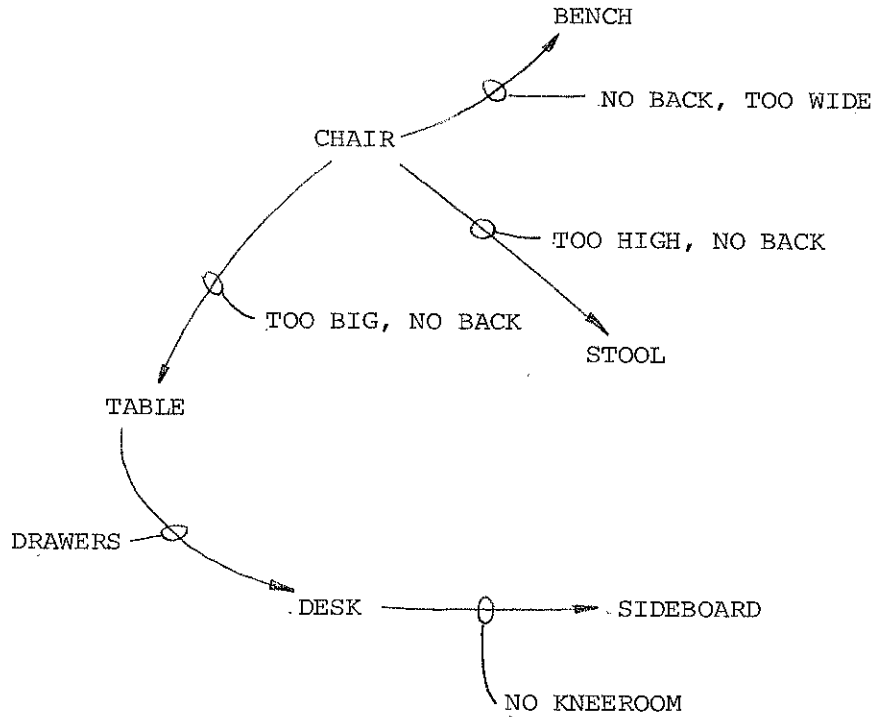


(4)



Ein anderer Zusammenhang zwischen Fakten ist im Ähnlichkeitsnetzwerk von Winston (1975) gegeben, in dem die Unterschiede zwischen verschiedenen Objekten erfasst wird (Abbildung 5, nach Minsky, 1975).

(5)



Minsky beschreibt nun den Kern seiner Frame-Theorie folgendermassen (Minsky, 1975):

"Wenn man einer neuen Situation begegnet (oder wenn man seine Sicht eines Problems erheblich verändert), dann wählt man aus dem Gedächtnis eine Struktur, die Frame genannt wird. Dabei handelt es sich um ein erinnertes Gerüst, das der Wirklichkeit angepasst werden kann, indem Einzelheiten daraus verändert werden.

Ein Frame ist eine Datenstruktur zur Darstellung stereotyper Situationen. Dabei kann es sich um den Aufenthalt in einem Wohnzimmer oder um den Besuch einer Geburtstagsfeier für Kinder handeln. Jedem Frame sind verschiedene Arten von Informationen zugeordnet. Einige dieser Informationen geben an, wie das Frame gebraucht werden kann, andere beschreiben, welches Ereignis als nächstes erwartet werden muss, und wieder andere Informationen handeln davon, was man unternehmen muss, wenn diese Erwartungen nicht eintreffen."

Wir können uns ein Frame als ein Netzwerk von Knoten und Relationen vorstellen. Die 'obersten Stufen' sind fixiert und repräsentieren Dinge, die

in der angenommenen Situation immer wahr sind. Die unteren Stufen haben viele 'Terminals' - 'Schlitze', die durch spezifische Instanzen von Daten gefüllt werden müssen. Jedes Terminal kann Bedingungen spezifizieren, die durch seine Zuordnungen erfüllt werden müssen. (Diese Zuordnungen sind üblicherweise kleinere 'Unterframes'.)" ...

"Gruppen von verwandten Frames werden miteinander in ein Frame-System verbunden. Die Effekte wichtiger Aktionen sind in den Transformationen zwischen Frames eines Systems reflektiert" ...

"Die Terminals eines Frames sind normalerweise mit 'Default'-Zuordnungen gefüllt. So kann ein Frame eine grosse Anzahl von Details enthalten, deren Annahme durch die Situation nicht spezifisch garantiert ist."

Im weiteren geht Minsky auf verschiedene Beispiele ein, in denen Frames sinnvoll angewandt werden können. In seinen linguistischen Beispielen zeigt sich, dass er eigentlich verschiedene Frames verwendet, die grob gesagt, der syntaktischen Repräsentation, der semantischen Repräsentation, der Repräsentation stereotyper Ereignisse (Thematische Frames) und der Repräsentation kommunikativer Konventionen (Narrative Frames) unterscheidet.

Der für die weiteren Entwicklungen auf dem Gebiet des Sprachverstehens wichtigste Frametyp ist das thematische Frame, in dem stereotype Situationen und Handlungen repräsentiert werden. Minsky zeigt anhand zweier Ausschnitte, welche Informationen in ihnen enthalten sein sollen: Einerseits enthalten sie statische Informationen, zu denen die Default-Zuordnungen der Terminals gehören. So enthält das "Geburtstagsparty"-Frame Defaults resp. Bedingungen der Terminals wie

(6) Kleid Sonntagskleid.

Geschenk Muss Gastgeber erfreuen. Muss gekauft und in
Geschenkpapier eingewickelt sein.

Essen Kuchen, Eiskrem, Soda, Hotdogs

Andererseits enthalten sie auch Informationen, die die Form von Handlungsanweisungen haben für jemanden (Y), der an einer Geburtstagsparty (für X) teilnimmt, wie etwa

(7) Y muss P für X haben Wähle P!

X muss P mögen Wird X P mögen?

Kaufe P Wo P kaufen?.

Minskys Artikel ist eigentlich mehr eine Ansammlung von Ideen über mögliche Formen und Anwendungen von Frames, als ein einheitlicher und ausgearbeiteter Formalismus. Aber nicht zuletzt wegen dieser Vagheit ist der Framesansatz so erfolgreich geworden, da er sich leicht auf jede Si-

tuation und (fast) jeden bestehenden Formalismus übertragen liess. Viele von den späteren Arbeiten über Frames waren denn auch bestrebt, die Idee des Framesansatzes zu präzisieren und formalisieren.

Zum Schluss seien nochmals die wichtigsten Ideen in Minskys Frameansatz zusammengefasst:

- Frames repräsentieren stereotype Situationen und Ereignisse.
- Die Frameterminals werden durch spezifische Objekte/Propositionen gefüllt.
- Die Terminals können Bedingungen angeben, die durch jeden "Füller" erfüllt sein müssen und "Entschuldigungen" angeben, die bei nur teilweiser Erfüllung der Bedingungen gegeben werden können.
- Die Terminals können einen Default angeben, der bei Abwesenheit eines Füllers angenommen wird.
- Verwandte Frames können in einem Framesystem organisiert werden.
- Die in einem Framesystem vorhandenen Relationen enthalten u.a. Ähnlichkeiten zwischen Objekten (Winstons Ähnlichkeitsnetzwerk), Transformationen zwischen verschiedenen Sichten der gleichen Situation / Objekt (Würfelrotation), hierarchische Relationen (im Sinne einer ISA-Relation in semantischen Netzen) oder Terminal-Subframe-Verbindungen.

3. CHARNIAKS FRAMEANSATZ

Charniak hat Minskys Framesidee aufgegriffen und in einer Reihe von Arbeiten (Charniak, 1975a, 1975b, 1976a, 1977, 1978b) konkretisiert. In seiner wichtigsten Arbeit (Charniak, 1977) hat er einen Teil des Alltagswissens über das Bemalen von Gegenständen repräsentiert, wobei er sich auf die darin enthaltenen Teilhandlungen, ihre Konsequenzen und Begründungen beschränkt hat.

3.1. Framesformalismus

Das folgende Fragment seines PAINTING-Frames (nach Charniak, 1977) soll eine gewisse Ahnung über seinen Formalismus vermitteln.

(8) PAINTING (COMPLEX-EVENT)

```

    VARS:  (AGENT must be animate)
           (OBJECT must be solid)
           (PAINT must be a liquid, usually PAINTING4: (a paint))
           (INSTRUMENT must be a solid, usually is either a
            roller or a paint brush, and should be
            absorbent)

    ...
    GOAL:  PAINTING-GOAL (OBJECT has a coat of PAINT on it)
           COMES-FROM (PAINTING22 via rules which say that
            paint on INSTRUMENT will stick to
            OBJECT, partially fulfilling the
            goal)
    EVENTS: PAINTING14 (NOT (object dirty)
              LEADS-TO: (DIRTY-OBJECT3))
    PAINTING15 (everything nearby is covered with paper)
    PAINTING16
      (LOOP
        PAINTING17 (paint is on instrument)
          COMES-FROM: (STICK10)
        PAINTING19 (NOT (GREATER amount of paint on in-
          strument than some constant)
          LEADS-TO (DRIP3))
          COMES-FROM:
            (a) (rule which states that
              surface volume varies
              directly with pressure)
            (b) (apply pressure to instrument)
        PAINTING22 (bring instrument in contact with object)
          LEADS-TO: (STICK6)
          (DURING PAINTING26 PAINTING22)
        PAINTING26 (amount of paint on instrument is greater
          than some constant)
          LEADS-TO: (STICK8))
    PAINTING37 (NOT (paint on instrument)
              LEADS-TO: (PAINT-DRY2))
              COMES-FROM: (WASH-GOAL)

```

Das abgebildete Frame ist nur ein Fragment des vollständigen Frames, wie etwa aus der Nummerierung der Frame-Terminals ersichtlich ist. Die in Grossbuchstaben geschriebenen Elemente sind im vollständigen Frame genauso enthalten, das übrige ist nur eine natürlichsprachliche Umschreibung des Formalismus.

In diesem Frame bezeichnet VARS die Variablen des Frames, GOAL das Ziel der Handlung und EVENT die zeitlich geordnete Reihe von Teilhandlungen, die zum Teil wiederholt ausgeführt werden können (LOOP von PAINTING17 bis PAINTING26).

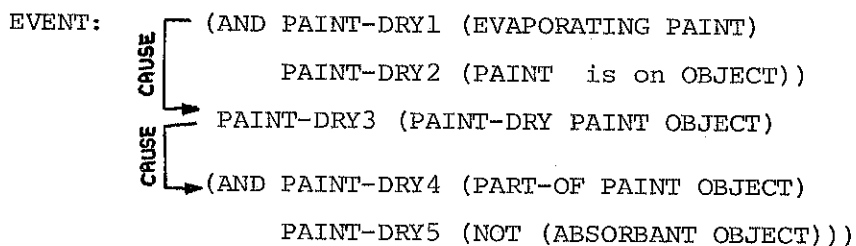
Die Relation LEADS-TO verbindet eine Teilhandlung resp. einen Teilzustand mit dem Terminal eines Frames, in dem die Konsequenzen dieser Teilhandlung bzw. dieses Teilzustandes beschrieben werden. So zeigt in PAINTING14 die Relation LEADS-TO in das Frame DIRTY-OBJECT, das beschreibt, dass die auf ein schmutziges Objekt aufgetragene Farbe mit der Zeit abblättert.

Die Relation COMES-FROM verbindet eine Teilhandlung resp. einen Teilzustand mit dem Terminal eines Frames, in dem beschrieben wird, wie diese Teilhandlung ausgeführt, resp. dieser Teilzustand erreicht werden kann. So zeigt in PAINTING37 die Relation COMES-FROM in das WASH-Frame, das beschreibt, wie Schmutz durch Waschen von einem Objekt entfernt werden kann.

Die Pfeile skizzieren die in den Frames enthaltenen Informationen über das zeitliche Nacheinander der einzelnen Teilhandlungen resp. Teilzustände.

Charniak unterscheidet vier verschiedene Frametypen: COMPLEX-EVENT, SIMPLE-EVENT, STATE und OBJECT. Ein COMPLEX-EVENT wie PAINTING beschreibt eine aus Teilhandlungen bestehende Tätigkeit. Ein SIMPLE-EVENT entspricht einer Zustandsänderung in der Conceptual Dependency Theorie von Schank. So beschreibt das leicht vereinfachte SIMPLE-EVENT-Frame PAINT-DRY (9)

(9) PAINT-DRY (SIMPLE-EVENT)



dass das Austrocknen von Farbe auf einem Objekt durch die Verflüchtigung der Farbe verursacht wird und selbst wiederum bewirkt, dass das Objekt selbst nicht mehr absorbieren kann. Ein STATE-Frame beschreibt einen Zu-

stand. So beschreibt etwa (10)

(10) CLOSED (STATE)

 VARS: (CONT must be a CONTAINER)

 (FLU must be a FLUID)

 RELATIONS: (AND CLOSED1 (CLOSED CONT)

 CLOSED2 (CONT contains FLU)

 IMPLIES

 CLOSED3 (NOT (FLU has contact with atmosphere))

dass eine Flüssigkeit in einem geschlossenen Behälter (z.B. einem Farbtopf) keinen Kontakt mit Luft hat. Schliesslich werden durch OBJECT-Frames Objekte beschrieben, wie das PAINT-Frame in (11), das besagt,

(11) PAINT (OBJECT)

 DESCRIPTION: (LIQUID PAINT)

 LEADS-TO: (PAINTING4)

dass Farbe eine Flüssigkeit ist, die beim Bemalen von Gegenständen verwendet wird. Charniak's Formalismus ist deswegen interessant, als es ihm gelungen ist, eine einheitliche Repräsentation von Aktionen, Zuständen und Objekten zu entwickeln, die mit sehr wenigen Relationen (COMES-FROM, LEADS-TO, Zeitrelationen und einigen wenigen andern) auskommt.

3.2. Framesanwendung

Charniak's Framesformalismus ist dazu geeignet, den von Minsky nur skizzierten Framesansatz des Sprachverstehens zu illustrieren. Danach werden Texte insofern verstanden, als sie mit den im Gedächtnis abgespeicherten Repräsentationen stereotyper Situationen und Handlungen in Bezug gebracht werden. Dies wird anhand der Geschichte (12) illustriert,

(12) Jack was going to paint a chair. He washed it.

deren semantische Repräsentation in erster Approximation wie (13) aussieht.

(13) S1 (INTEND JACKS2)

 S2 (PAINTING JACK CHAIR)

 S3 (WASH JACK CHAIR)

S2 wird als eine Instanz des PAINTING-Frames betrachtet mit AGENS=JACK und OBJECT=CHAIR (vgl. PAINTING-Frame in (8)). S3 ist eine Instanz des WASH-Frames, dessen WASH-GOAL (NOT(Object dirty)) mit PAINTING14 übereinstimmt. Damit ist über das PAINTING-Frame der Zusammenhang zwischen S2 und S3 gegeben und die Aktion S3 kann so begründet werden, dass damit der Stuhl nicht mehr schmutzig ist (WASH-GOAL) und dass dies notwendig ist, weil sonst die auf den Stuhl aufgetragene Farbe mit der Zeit

abblättert (PAINTING14 LEADS-TO DIRTY-OBJECT). Schon diese kleine Passage zeigt, dass das Programm von Charniak ein tieferes Textverständnis zeigt, als es sonst bei sprachverstehenden Programmen üblich ist. Allerdings ist es aus Platzgründen nicht möglich, diese Aussage an weiteren Beispielen zu belegen.

Auch in Charniaks Frames sind mit Terminals Defaultwerte und Restriktionen assoziiert, wie dies am Beispiel des (Mal-) Instruments illustriert wird, das folgendermassen beschrieben wird

(14) (INSTRUMENT (SOLID INSTRUMENT)

NORMAL: PAINTING1 (PAINT-BRUSH INSTRUMENT)

PAINTING2 (ROLLER INSTRUMENT)

PAINTING3 (ABSORBENT INSTRUMENT)

und besagt, dass das Instrument ein fester Körper sein muss und normalerweise ein Pinsel oder Farbroller ist und normalerweise die Eigenschaft "absorbierend" hat. Wird nun in einer Geschichte das Instrument nicht explizit erwähnt, dann nimmt das Programm an, dass ein Pinsel oder (etwas weniger wahrscheinlich) ein Farbroller gebraucht wurde. Die strikte Eigenschaft SOLID muss von jedem Instrument erfüllt werden und daher würde weder "idea" in (15)

(15) Jack painted a chair with an idea.

noch "green paint" in (16)

(16) Jack painted a chair with green paint.

als Instrument akzeptiert. Interessanter ist allerdings der Fall, in dem das Instrument nur die "normale Eigenschaft" erfüllt, dass es absorbierend ist. Dies ist bei "Schwamm" in (17) der Fall.

(17) Jack was going to paint a chair. He dipped a sponge into the paint.

In diesem Falle würde das Programm nach weiterer Evidenz suchen, die für die Anerkennung von "Schwamm" als Instrument spricht, wie dies bei einer Fortsetzung von (17) in (18) der Fall wäre.

(18) Then he wiped the sponge across a leg of the chair.

Hinter der Aussage, dass das Programm nach weiterer Evidenz sucht, steckt ein fundamentales Prinzip des Framesansatzes des Sprachverstehens. In einem Frame werden normale Situationen und Handlungsabläufe beschrieben. Jede Abweichung davon ist für das Programm ein Indiz dafür, dass hier etwas Interessantes ausgesagt wird, für dessen Interpretation sich ein vermehrter Aufwand lohnt. Dies ist auch der Fall in (19),

(19) Jack was painting a wall. At one point he pressed the brush too hard against the wall.

Normalerweise würden nur die Terminals der aktiven Frames nach einer Uebereinstimmung mit dem Eingabetext abgesucht. Im PAINTING-Frame (PAINTING19) wird aber nur die Menge der Farbe auf dem Instrument reguliert, nicht aber die Stärke des Drucks. Normalerweise würde daher auch die richtige Interpretation des zweiten Satzes nicht gefunden. Die Tatsache aber, dass mit 'too much' eine Abweichung von der Norm indiziert ist, ist mit ein Grund dazu, dass das Programm in diesem Falle in den Unterframes nach den genauen Zusammenhängen zwischen dem Druck auf den Pinsel und der Menge der Farbe auf dem Pinsel sucht. Dies ermöglicht ihm schliesslich eine korrekte Interpretation zu finden (ohne dass hier auf die Details eingegangen wird. (Siehe dazu Charniak, 1976a).

Der Framesansatz des Sprachverstehens, so weit er bis jetzt besprochen wurde, gibt uns ein Instrument, um möglichst viele der zur Lösung eines Problems irrelevanten Inferenzregeln auszuschalten, da nur Inferenzregeln der gegenwärtig aktiven Frames (und in speziellen Situationen ihrer Unterframes) in Betracht gezogen werden. Auf der andern Seite muss aber auch garantiert werden, dass die in einer bestimmten Situation notwendigen Inferenzregeln auch verfügbar oder in andern Worten, dass die in einer Situation anwendbaren Frames auch aktiviert werden.

Eine Frameaktivierung in einer hypothesengesteuerten Top-down-Manier wäre unplausibel, denn es gibt kein gutes Kriterium, wie man den Raum von vielleicht hunderttausenden von möglichen Frames a priori genügend einschränken könnte, um eine gute erste Hypothese aufstellen zu können. So ist man sich denn auch einig, dass die Frameaktivierung ein vorwiegend datengesteuerter Prozess sein muss.

In einem ersten Versuch könnte man davon ausgehen, dass Frames durch Schlüsselwörter im Text aktiviert werden. So wird etwa, wenn das Wort "Restaurant" im Text auftaucht, das RESTAURANT-Frame aktiviert. Wenn man aber nach diesem simplen Schema verfährt, dann wird im Satz (20) (aus Charniak, 1978)

(20) The lawyer took the cab to the restaurant near the university.
eine ganze Reihe von Frames aktiviert, die über "lawyer", "cab", "restaurant" und "university" gefunden werden.

In der "Durchschnittstechnik" von Fahlman (1979) wird der Durchschnitt von allen mit den Schlüsselwörtern assoziierten Frames gebildet werden und damit die Menge von aktivierten Frames entscheidend eingeschränkt werden. Zwei Probleme, die mit dieser Technik verbunden sind, bestehen darin, dass man erstens kein gutes a priori Kriterium hat, über wieviele Schlüs-

selwörter hinweg man diesen Durchschnitt bilden soll (über den ganzen Text hinweg?) und zweitens darin, dass dieser Durchschnitt häufig auch leer werden kann, wenn nicht eine ganze stereotype Geschichte gelesen wird.

Charniak (1978a) schlägt ein Schema vor, in dem sowohl mit Prädikaten (Aktionen und Zustände) als auch mit Objekten eine kleine Liste von Frames assoziiert sind, in denen sie eine wichtige Rolle spielen. Allgemeine Wörter wie "gehen" haben keine Frames assoziiert, da sie in zu vielen Frames auftauchen, als dass sie zu einem vernünftigen kleinen Hypothesenraum führen könnten. Sind nun diese Listen von assoziierten Frames sehr klein oder die Kosten für die Framesaktivierung resp. Korrekturen im Falle einer falschen Hypothese klein (d.h. brauchen nur einen kleinen Verarbeitungsaufwand) dann genügt dieses Aktivationschema. Im andern Falle können nun beliebig komplizierte Vortests eingebaut werden, die zwischen den Kandidatenframes in den assoziierten Frames listen oder zwischen den mit den verschiedenen Wörtern assoziierten Frames entscheiden. Einige Eigenschaften dieser Tests werden in Charniak (1978a) skizziert, aber sie sind noch zu wenig ausgearbeitet, als dass hier eingehender auf sie eingegangen werden sollte.

4. DER FRAMESANSATZ VON SCHANK UND ABELSON

Der Ansatz von Schank und Abelson (Schank, 1975; Schank & Abelson, 1977) setzt etwas andere Akzente als derjenige von Charniak. Charniak ist daran interessiert, einen eingeschränkten Wissensbereich möglichst vollständig und in einem sauber ausgearbeiteten Formalismus zu repräsentieren. Schank und Abelson andererseits haben wohl auch einen einfachen Formalismus zur Repräsentation von Frames (genannt Skripts) entwickelt, haben sich aber viel mehr mit der Repräsentation von allgemeinerem Wissen auseinandergesetzt.

Sie sind von der Tatsache ausgegangen, dass Wissensstrukturen in Form von Skripts, die spezifische, stereotype Situationen und Handlungen beschreiben, nur in einem kleinen Teil von Situationen anwendbar sind. So ist es unplausibel, dass der Text (21)

(21) Hans wusste, dass die Operation seiner Frau sehr teuer werden würde. Da war immer noch Onkel Harry. Er holte das Telefonbuch. mit Hilfe eines 'Bezahlen-einer-Operation'-Skripts verstanden wird. Andererseits ist diese Situation aber auch nicht vollkommen neu und das Verstehen dieser Passage wäre nicht signifikant verschieden, wenn "Operation seiner Frau" durch "Ausbildung seines Sohnes" oder "Abzahlung einer Hypothek" ersetzt würde. Schank und Abelson argumentieren nun, dass solche Texte aufgrund von Wissen über allgemeine Pläne verstanden werden, in denen ein Zielzustand (Verfügen über viel Geld) mit einer Menge von möglichen Aktionen (ausleihen, stehlen, in Lotterie gewinnen usw.) verbunden ist, die zu diesem Zustand führen.

Nun sind Pläne nicht grundsätzlich verschieden von Skripts und der Uebergang von Plänen zu Skripts kann als fliessend betrachtet werden. Pläne haben nur allgemeinere Ziele als Skripts und dementsprechend ist auch die Menge der möglichen Teilhandlungen oft viel grösser. Wird nun eine bestimmte Plansequenz sehr häufig gebraucht, dann ist es aus Effizienzgründen vorteilhaft, wenn diese Sequenz in Form eines Skripts abgespeichert wird und nicht jedesmal wieder neu abgeleitet werden muss (Diese Idee ist schon von Sussman (1973) in einem Programm realisiert worden).

Am Beispiel des Planes mit dem Ziel "Brauchen eines Objektes" wie das Essen von Nahrung oder das Lesen eines Buches soll skizziert werden, wie Pläne im Sprachverstehen verwendet werden. Dieser Plan wird durch (21) repräsentiert

(21) USE(X) = D-KNOW(LOC(X)) + D-PROX(X) + DCONT(X) + I-PREP(X) + DO

und besagt ungefähr, dass wenn man X brauchen will (USE(X)), dann muss man zuerst den Ort von X herausfinden (D-KNOW(LOC(X))), dann am gleichen Ort wie X sein (PROX(X)), dann die Kontrolle über X haben (D-CONT(X)), dann X zum Gebrauch vorbereiten (I-PREP(X)) und dann X brauchen (DO).

Eine mögliche Realisation dieses Planes im Falle von "Essen" könnte daher die folgende sein: Zuerst fragt man, wo es etwas zu essen gibt (falls man es nicht schon weiss) und geht dann, zum Beispiel, in ein Restaurant (falls man nicht schon dort ist), gewinnt Kontrolle über die Nahrung (die im Falle eines Restaurants durch Bezahlung ermöglicht wird), bereitet sich zum Essen vor (Tisch finden, bestellen, usw.) und führt dann die Zielhandlung aus, indem man isst.

Auf der rechten Seite von (21) stehen die Teilziele dieses Planes. Mit jedem dieser primitiven Teilziele sind Skripts und Plankästen assoziiert, die zur Erfüllung dieses Teilziels führen können. So sind mit dem Ziel D-CONT, dem Kontrollgewinn über ein physisches Objekt, die folgenden Plankästen assoziiert: ASK, INVOKE THEME, INFORM REASON, BARGAIN OBJECT, BARGAIN FAVOR, THREATEN, OVERPOWER und STEAL. Jeder dieser Plankästen beschreibt nun einen Weg, um das primitive Teilziel D-CONT zu erreichen. Das folgende Beispiel (nach Schank & Abelson, 1977, S. 92) soll dies illustrieren.

- (22) Hans wollte eine der Whiskyflaschen von Harry.
- (22a) Er sagte: "Wie wär's mit einer für mich?". (ASK)
- (22b) Er erinnerte Harry daran, dass sie schon immer Freunde gewesen waren. (INVOKE THEME)
- (22c) Er sagte, dass der Chef noch kommen würde und dass der Whisky ausgegangen sei. (INFORM REASON)
- (22d) Er offerierte Harry eine Flasche Montrachet. (BARGAIN OBJECT)
- (22e) Er offerierte ihm Holz zu spalten. (BARGAIN FAVOR)
- (22f) Er tönte an, dass er nicht genug Anerkennung für seine Geschäfte mit Harry erhielt. (THREATEN)
- (22g) Er boxte Harry ins Gesicht. (OVERPOWER)
- (22h) Er stellte beiläufig seine Tasche neben die Bar und wartete auf seine Gelegenheit. (STEAL)

Es soll hier nicht über ein illustrierendes Beispiel hinausgegangen werden. Die zugrundeliegende Idee ist aber klar: Das sprachverstehende System verfügt über eine Menge von mehr oder weniger standardisierten Plänen, mit deren Teilzielen Skripts und Plankästen assoziiert sind. Daraus ergibt sich das folgende Modell des Sprachverstehens von plan-

orientierten Geschichten. Zuerst muss das Ziel bestimmt werden, das der Handelnde vor sich hat. Daraus können dann die primitiven Teilziele abgeleitet werden. Jeder der nachfolgenden Sätze wird nun danach geprüft, ob einem der Plankästen von einem dieser Teilziele entspricht. Ist dies der Fall, dann ist damit auch eine Verbindung zum vorangegangenen Text gegeben und somit eine kontextuelle Interpretation des Satzes gefunden.

5. GRUNDLAGEN DER WISSENSORGANISATION

In diesem Abschnitt werden die wichtigsten Eigenschaften des Framesansatzes rekapituliert und im Hinblick auf eine allgemeinere Theorie der Wissensorganisation diskutiert. Daran anschliessend wird auf einige allgemeinere Probleme der Wissensorganisation eingegangen.

5.1. Frametypen

Bereits Minsky unterscheidet zwischen verschiedenen Frames, was aus seiner Klassifikation in syntaktische, Kasus-, semantische, thematische und narrative Frames hervorgeht. Auf eine weitere Unterscheidung geht er nicht ein, braucht sie aber in seinen Beispielen, die Unterscheidung zwischen Objektframes (z.B. seine Repräsentation eines Zimmers), deren Terminals mit Objekten gefüllt werden (z.B. Fenster) und propositionalen Frames (z.B. das An-eine-Geburtstagsparty-gehen-Frame), deren Terminals mit Propositionen oder Handlungen gefüllt werden. Es ist nun wichtig, dass in der Anwendung des Framesansatzes kein Unterschied zwischen den verschiedenen Frametypen besteht und somit insbesondere zwischen der Verarbeitung von Objekten und der Verarbeitung von Handlungen kein Unterschied gemacht wird. Es soll hier nicht auf die philosophischen Implikationen dieser Haltung eingegangen werden, aber einige direkte Konsequenzen aus dieser Haltung sind schon bemerkenswert. Wie Objekte können auch Handlungen typische Eigenschaften haben wie etwa typische Teilhandlungen oder typische Objekte, die in dieser Handlung involviert sind. Ebenso werden auch bei Handlungen Defaulteigenschaften angenommen, die bis zum Eintreten widersprüchlicher Information beibehalten werden. Es ist nicht ganz klar, wie weit die Aehnlichkeiten im Erkennen von Objekten und im Erkennen von Handlungen gehen und erst die Zukunft kann zeigen, wie fruchtbar dieser Ansatz ist.

5.2. Repräsentationsfragen des Framesansatzes

Es wurde oft argumentiert, dass der Framesformalismus ein ausdrucksstärkerer Repräsentationsformalismus ist als etwa der Prädikatenkalkül. Pat Hayes (1977b) hat aber gezeigt, dass dies nicht der Fall ist und sich alle wichtigen Eigenschaften von Frames auch mit dem Prädikatenkalkül repräsentieren lassen. Zwei dieser Eigenschaften sollen hier wiedergegeben

werden, nämlich die Repräsentation von Frameterminals und die Repräsentation von Defaultwerten. Dies ist insofern interessant, als im Prädikatenkalkül diese im Framesansatz nur ungenau definierten Konzepte präziser erfasst werden können.

Aus der bisherigen Diskussion ist hervorgegangen, dass ein Frame ein Ausdruck zur Beschreibung einer stereotypen Situation ist, und dass mit jedem Frame eine Anzahl von Terminals assoziiert sind. Die Entitäten, die auf Frames und auf Frameterminals bezogen werden (und als Instanzen bezeichnet werden wie bei den semantischen Netzwerken) sind Individuen. Jedes Frameterminal definiert eine Relation zwischen der Frameinstanz und der Frameterminalinstanz. So wird zum Beispiel im AUTO-Frame durch das Frameterminal BESITZER eine Relation zwischen der Instanz des AUTO-FRAMES (z.B. Al) und der Instanz des BESITZER-Frameterminals (z.B. Pl) definiert, eine Relation, die am besten als BESITZER-VON bezeichnet wird, also BESITZER-VON(Al,Pl). Somit kann ein Frame F mit den Frameterminals T_1, \dots, T_n durch den Prädikatenkalkülausdruck (23) beschrieben werden.

$$(23) \quad \forall x (F(x) \equiv \exists y_1, \dots, y_n (T_1(x, y_1) \& \dots \& T_n(x, y_n)))$$

Dieser Ausdruck besagt ungefähr, dass zu jeder Instanz x eines Frames F die Entitäten y_1, \dots, y_n existieren, die mit x durch die Relationen T_1, \dots, T_n verbunden sind.

Etwas schwieriger ist der Begriff des Defaults zu erfassen, das heisst, dass für ein Frameterminal eine spezifische Instanz angenommen wird, vorausgesetzt, dass keine anderslautende Informationen vorhanden sind. So wird zum Beispiel im PAINTING-Frame in Abwesenheit andersartiger Information angenommen, dass das INSTRUMENT ein Pinsel war. Wenn mit Φ die Menge aller Propositionen des Systems bezeichnet wird, dann kann dieser Default im Prädikatenkalkül folgendermassen definiert werden (\vdash = ableitbar).

$$(24) \quad (\neg \exists y (\Phi \vdash \text{Instrument(malen)=y})) \supset (\Phi \vdash \text{Instrument(malen)=Pinsel})$$

5.3. Framesansatz des Sprachverstehens

Im Framesansatz des Sprachverstehens werden eine Reihe von Annahmen über das Sprachverstehen gemacht. Von diesen Annahmen werden in diesem Abschnitt drei Annahmen etwas eingehender betrachtet und in ihrem Geltungsanspruch relativiert.

Erstens wird einmal angenommen, dass Texte dadurch verstanden werden, dass sie auf ein oder mehrere Frames bezogen werden, denn damit werden Zusammenhänge zwischen den einzelnen Sätzen des Textes definiert. Als einfachstes Argument könnte man gegen diese Annahme anführen, dass damit

nur spezifische Situationen, die mit Skripts im Sinne von Schank beschrieben werden, verstanden werden können. Diese Einschränkung kann aber mindestens teilweise aufgehoben werden, wenn man auf Pläne im Sinne von Schank zurückgreift, die auf ein viel breiteres Spektrum von Situationen und damit auch beschränkt auf neue Situationen anwendbar sind. Trotzdem bleibt aber das Problem bestehen, wie mit dem Framesansatz völlig neuartige Situationen verstanden werden können. So hat denn auch Wilks (1977) in einer Kritik des Framesansatzes dagegen argumentiert, dass die folgende Passage aus einem Text über Initiationsriten nur dadurch verstanden werden kann, dass sie

(25) Little Kimathi's mother accidentally touched his arm during the puberty rite. The crowd drew back in horror.

auf ein PUBERTY-RITE-Frame bezogen wird, in dem etwa enthalten ist, dass das Berühren des Arms des Kindes eine verbotene Handlung ist und dass daraus mit Hilfe weiterer Inferenzregeln eine Verbindung zum zweiten Satz gefunden wird. Viel plausibler ist es, dass wir in solchen Situationen, in denen wir kaum über spezifische Kenntnisse verfügen, mit allgemeineren Inferenzregeln wie etwa (26, nach Wilks, 1977) arbeiten.

(26) (Humans display alarm) → (other humans perform a bad action)

In andern Worten wird in dieser Situation der wichtige Zusammenhang zwischen den beiden Sätzen vom zweiten Satz via allgemeine Inferenzregeln gefunden, und nicht vom ersten Satz aus, dessen tieferes Verstehen genauere Kenntnisse über Initiationsriten bedingen würde.

Eine zweite Annahme, die im Framesansatz des Sprachverstehens gemacht wird, besteht darin, dass Frames den normalen Ablauf einer Handlung definieren, so wie sie auch ein normales Zimmer definieren. Das führende Motiv hinter dieser Annahme ist nun nicht, dass ein Frame damit einen möglichst grossen Bereich von Texten über eine bestimmte Situation abdeckt, denn Texte wie (27) sind nicht nur äusserst selten und langweilig, sondern widersprechen, grob gesagt, grundlegenden kommunikativen Gesetzen.

(27) Hans wollte einen Stuhl malen. Er bedeckte den Boden mit Zeitungspapier. Dann tauchte er wiederholt den Pinsel in den Farbkübel und bestrich dann den Stuhl. Am Schluss reinigte er den Pinsel, indem er ihn mit Seifenwasser wusch.

Die Festlegung eines normalen Handlungsablaufes in einem Frame dient dazu um Abweichungen vom normalen Ablauf zu entdecken, die wie in Charniaks Frameansatz einen erhöhten allgemeinen Verarbeitungsaufwand

rechtfertigen. Daneben lassen sich aber für solche Abweichungen spezifische Regeln formulieren wie etwa die Regel von Schank, dass zwei Abweichungen vom normalen Ablauf einer Handlung miteinander kausal verbunden sein könnten. Im Textfragment (28)

(28) Hans ass in einem Restaurant. Der Hamburger war kalt.

Hans liess ein kleines Trinkgeld zurück.

könnte daher mit einem relativ einfachen Mechanismus festgestellt werden, dass Hans wegen dem kalten Hamburger ein kleines Trinkgeld zurückliess, da sowohl "kalte Speise" als auch "kleines Trinkgeld" Abweichungen vom normalen Handlungsablauf darstellen und das frühere Ereignis (Servieren der kalten Speise, die warm sein sollte) das spätere Ereignis (Zurücklassen des kleinen Trinkgelds) verursacht. Diese einfache Verursachung von abweichenden Ereignissen ist aber nicht notwendig und in einem etwas plausibleren System würde daher diese Inferenz nur als Hypothese zur genaueren Ueberprüfung aufgestellt. Dies kann zum Beispiel damit erleichtert werden, als im Frame selbst "Vorschläge" enthalten sind, wie solche Abweichungen erklärt werden können oder welche Erwartungen aus diesen Abweichungen ableitbar sind.

Mit der Strukturierung von Handlungen in Hierarchien von Handlungen und Teilhandlungen mit assoziierten Zielen und Teilzielen geht nun auch eine neue Interpretation von Handlungen einher. Eine Handlung wie das Drehen des Zündschlüssels wird nun nicht mehr isoliert verstanden, sondern ist Teilhandlung der höheren Handlung Autofahren, die wiederum nur eine Teilhandlung einer übergeordneten Handlung ist. Daraus ergeben sich zwei unmittelbare Probleme. Wie weit ist es erstens sinnvoll, in einem sprachverstehenden System nach übergeordneten Handlungen zu suchen, etwa bis zu einem obersten LEBENS-Frame mit den Teilhandlungen Geburt-Kopulation-Tod wie es Yolks (1975) karriert hat. Zweitens fragt sich, wie weit sich aus solchen übergeordneten Frames noch Erwartungen ableiten lassen, die spezifisch genug sind, um im Verstehensprozess noch sinnvollerweise verwendet werden zu können. So unwichtig diese Probleme auch heute im Vergleich mit den aktuellen viel primitiveren Problemen zu sein scheinen, einmal werden auch sie beantwortet werden müssen.

5.4. Wissensorganisation in Frames

Die grösste Bedeutung des Framesansatzes liegt darin, dass damit ein Formalismus gegeben ist, um Inferenzregeln einfach organisieren zu können. Um dies zu illustrieren, wird der Framesansatz mit dem Dämonenansatz von Charniak (1972) verglichen.

Die wichtigste Idee dieses geschichtenverstehenden Programms war, dass die einzelnen Sätze einer Geschichte erst in ihrem Kontext adäquat interpretiert werden können, etwa der Satz "There was no sound" im Kontext eines Kindes, das sein Sparschwein schüttelt. (Diese Idee ist in der bisherigen Diskussion über Frames immer wieder aufgetaucht). Dazu war im Modell mit jedem thematischen Konzept wie Sparschwein oder Geburtstag eine "Basisroutine" assoziiert, die beim Auftauchen dieses Konzepts eine Reihe von Dämonen (gleich Antezedenstheoremen in PLANNER) aufstellt. Diese Dämonen warten nun auf Sätze wie "There was no sound" und wissen, falls sie auftauchen, wie diese Sätze interpretiert werden müssen (z.B., dass kein Geld im Sparschwein ist). So weit entspricht der Dämonensatz genau dem Framesansatz: Bei der Aktivierung eines Frames werden eine Reihe von Erwartungen (Frame-terminals) aufgestellt und im Falle einer Instantiierung eines Frame-terminals ist eine Interpretation dieser Instanz gegeben im Sinne wie es im Abschnitt 5.2. diskutiert wurde.

Der Framesansatz bietet nun aber gegenüber dem Dämonenansatz zwei wichtige Vorteile, die beide damit zu tun haben, dass die Dämonen wohl gemeinsam (durch die Basisroutine) aktiviert werden, dann aber unabhängig voneinander operieren. Der erste Vorteil ist relativ geringfügig und besteht darin, dass im Dämonenansatz jeder Dämon unabhängig vom andern prüfen muss, ob auftauchende Textfragmente alle für die Anwendbarkeit des Dämonen notwendigen Bedingungen erfüllt, während im Frameansatz diese Überprüfung für alle Dämonen (Frame-terminals) nur einmal gemacht werden muss. Wenn daher aus dem vorangegangenen Text bekannt ist, dass zum Beispiel das Malinstrument (im PAINTING-Frame) ein Farbroller ist, dann gilt diese Beschränkung für alle Framestatements und muss nicht von jedem Frame-terminal neu überprüft werden. Der zweite und entscheidende Vorteil des Framesansatzes liegt aber darin, dass sich im Framesansatz zwischen den Dämonen im Prinzip beliebige Relationen definieren lassen, die dazu verwendet werden können um zwischen den Dämonen Prioritäten setzen zu können. Aufgrund dieser relativen Prioritäten können die in einem Frame aktivierten Dämonen verschieden leicht verfügbar sein oder es kann im Falle gleichzeitiger Anwendbarkeit verschiedener Dämonen Konfliktlösungsschemata verwendet werden, wie sie bei den Produktionssystemen verwendet wurden.

Damit hängt der entscheidende Vorteil des Frameansatzes von der Definition nützlicher Relationen zwischen den Dämonen und ihrer Verwendung in einem Prozessmodell ab (siehe dazu die Diskussion über Taxonomien von Inferenzregeln). Wir wollen dies anhand von zwei Beispielen diskutieren.

Eine erste Relation ist in der hierarchischen Organisation in Frames und Unterframes gegeben, wie dies bei Charniak's Frameansatz der Fall ist. Dort werden genauere (detailliertere) Zusammenhänge zwischen bestimmten Ereignissen in Unterframes beschrieben und diese wiederum werden nur dann abgesucht, wenn die gegenwärtige Situation einen erhöhten Verarbeitungsaufwand rechtfertigt, wie dies in (19) der Fall war. Damit ist die simple Aktiv/Inaktiv-Dichotomie des Dämonenansatzes im Framesansatz durch ein feiner aufgegliedertes Aktivierungsschema ersetzt worden.

Eine zweite Relation ist in der zeitlichen Ordnung der Dämonen im Framesansatz von Charniak gegeben. Diese werden im PAINTING-Frame unter anderem so eingesetzt, dass Dämonen, die sich auf zukünftige Handlungen beziehen, die Priorität haben über solche Dämonen, die sich auf vergangene Handlungen beziehen und dass Dämonen, die sich auf zeitlich nähere Handlungen (in der Zukunft) die Priorität haben über jene Dämonen, die sich auf zeitlich entferntere Handlungen (in der Zukunft) beziehen. Mit diesen Regeln ist es Charniak's Programm möglich, die plausibelste Antwort auf die beiden folgenden Textfragmente (29) und (30) (aus Charniak, 1978b) zu geben.

(29) Jack was going to paint. He got some newspaper.

Question: What for? Answer: To cover nearby things.

(30) Jack finished painting. He got some newspaper.

Question: What for? Answer: To clean the paint brush.

Es würde zu weit führen, hier mehr als nur beispielhaft auf die verschiedenen Ansätze der inhaltlichen Wissensorganisation einzugehen, wie dies hier geschehen ist. Dies ist eines der heute aktuellen Themata im Gebiet des Sprachverstehens und es wird noch einige Zeit dauern, bis man die verschiedenen Ansätze unter einem einheitlichen Gesichtspunkt betrachten können. Es ist auch schwer abzusehen, wie gross der Nutzen der verschiedenen Organisationsprinzipien in der aktuellen Anwendung im Sprachverstehen objektiv sein wird. Nicht zuletzt soll aber auf ein weiteres Organisationsprinzip hingewiesen werden, nämlich dass Inferenzen auch nach rein idiosynkratischen Interessantheitsgraden organisiert werden können (Schank, 1978), einer Idee, der wir schon bei der Diskussion der partiellen Parserstrategien begegnet sind.

6. FRAMESANSATZ IN DER PSYCHOLOGIE

Obwohl "Frames" schon kurz nach dem Erscheinen von Minskys Arbeit auch in der Psychologie zu einem Schlagwort wurde, gibt es eigentlich nur sehr wenige Arbeiten, die sich mit dem Framesansatz im engeren Sinne beschäftigen. Denn im Framesansatz ist mehr enthalten als nur die allgemeine Idee, dass Wissen in "Clusters" im Sinne von Bousfield (1953) gruppiert ist oder in einer fixen Sequenz von Operationen im Sinne von Bartletts (1932) Schemata repräsentiert ist. Ebensowenig treffen jene Arbeiten den Kern des Framesansatzes, die sich nur mit dem Einfluss des Kontextes auf die Interpretation einzelner Sätze befassen. Im Framesansatz sind zusätzliche, spezifische Ideen enthalten. Dazu gehören etwa die Idee des Defaultwertes, oder die Idee, dass ein Text mit Hilfe von Frameinstantiierungen verstanden wird. Oder auch die Idee, dass auf der Basis von Frames ungewöhnliche oder untypische Ereignisse erkannt werden. Die Arbeiten aber, die sich mit diesen spezifischen Eigenschaften des Framesansatzes beschäftigen, sind in der psychologischen Literatur äusserst spärlich vertreten.

Auf zwei dieser Arbeiten soll kurz eingegangen werden, nämlich auf die Arbeit von Bower, Black und Turner (1979) und die Arbeit von Graesser, Gordon und Sawyer (1979).

Bower et al. haben in einer Reihe von Experimenten einige Eigenschaften des Skriptansatzes von Schank & Abelson auf ihre psychologische Plausibilität überprüft. Aus diesen Experimenten sind drei Resultate erwähnenswert, nämlich erstens einmal das Resultat, dass typische (und notwendige) Teilhandlungen eines Skripts dazu tendieren, in der Wiedergabe eines Textes fälschlicherweise wiedergegeben zu werden. Die Falschalarmrate ist noch erhöht, wenn die Vpn noch eine zweite Geschichte (auf der Basis des gleichen Skripts) gelesen haben, in der diese Teilhandlung explizit erwähnt worden war. Bower et al. konnten diesen Effekt auf der Basis des Frameinstantiierungsmechanismus und der zusätzlichen Annahme, dass instantiierte Frameterminals markiert werden, plausibel erklären. In einem andern Experiment zeigte sich, dass Ereignisse, die an einer falschen Position im Handlungsablauf auftauchen, bei der Wiedergabe häufig an der im Skript definierten Position wiedergegeben werden. In einem dritten Experiment fanden Bower et al., dass untypische (d.h. im Skript nicht enthaltene) Ereignisse mit erhöhter Wahrscheinlichkeit reproduziert werden und führen dies darauf zurück, dass für diese Ereignisse ein erhöhter Verarbeitungsaufwand notwendig war, der zu einer stabileren Gedächtnisspur führte. Diesen letzten

Effekt konnten auch Graesser et al. (1979) in einem ähnlichen Experiment nachweisen.

Ein grundsätzliches Problem der psychologischen Überprüfung des Framesansatzes besteht darin, dass man keine gesicherten Daten darüber erheben kann, welche Teilhandlungen in einem Skript enthalten sind, aber auch nicht von welchem Allgemeingrad ein Skript ist, das eine Versuchsperson in einem Experiment verwendet. Als wahrscheinlich einziger Ausweg bleibt daher die Annahme (sowohl von Bower et al. als auch Graesser et al.), dass die Versuchspersonen ein einheitliches Skript verwenden, und dass dieses Skript die in der freien Verbalisierung am häufigsten genannten Ereignisse als Teilhandlungen enthält. Die Annahme eines universellen Skripts ist völlig unplausibel, da die von einem Menschen verwendeten Skripts auf eine individuelle Lerngeschichte zurückgeführt werden müssen. Zweitens würden in der Anwendung von Skripts beim Sprachverstehen mit an Sicherheit grenzender Wahrscheinlichkeit viel mehr Fakten verwendet, als Versuchspersonen je frei verbalisieren würden.

Schon aus diesem ersten Problem lässt sich ableiten, dass es mit Hilfe der heutigen experimentellen Techniken der Psychologie wohl kaum möglich sein wird, mehr als nur die größten Eigenschaften einer Theorie des Sprachverstehens, wie sie der Framesansatz darstellt, zu überprüfen. Feinere Unterschiede zwischen den verschiedenen Framesansätzen lassen sich experimentell wohl kaum mehr gesichert erfassen.

VI. E N T W I C K L U N G V O N T H E O R I E N
K O G N I T I V E R L E I S T U N G E N

1. Einleitung

In diesem Kapitel wird die Entwicklung von Theorien kognitiver Leistungen diskutiert. Auch wenn wir dazu auf der Grundlage der bisher diskutierten Theorien des Sprachverstehens aufbauen, so glauben wir, dass die Schlüsse, die daraus gezogen werden können, nicht auf Theorien des Sprachverstehens beschränkt sind, sondern sich auf alle kognitiven Leistungen verallgemeinern lassen. In dieser Diskussion wollen wir insbesondere auf die Rolle zu sprechen kommen, welche die Künstliche Intelligenz darin spielen kann. Dazu muss aber zuerst einmal gezeigt werden, dass die Erkenntnisse der Künstlichen Intelligenz für die Entwicklung von Theorien kognitiver Leistungen überhaupt relevant sein können. Man kann ja die Zielsetzung der Künstlichen Intelligenz rein technologisch auffassen, d.h. ihr einziges Ziel darin sehen, eine Maschine zu konstruieren, die intelligente Leistungen möglichst effizient erbringt, unabhängig davon, welche Strategien etwa vom Menschen verwendet werden. Unter dieser Zielsetzung wäre es allerdings schwer zu zeigen, dass Erkenntnisse der Künstlichen Intelligenz von Bedeutung für die Kognitive Psychologie sein können. Die Situation ist zu vergleichen mit der Bedeutung der Erkenntnisse aus dem Flugzeugbau für das Verständnis des Vogelfluges. Es ist nicht abzustreiten, dass dort gewisse Erkenntnisse, etwa der Aerodynamik, auch zum Verständnis des Flugapparats der Vögel beitragen können, aber auf Grund verschiedener Zielsetzungen in der technischen Anwendung und den biologischen Gegebenheiten bleibt dieser Transfer doch sehr beschränkt. Eine analoge Einschränkung kann auch für die Künstliche Intelligenz gemacht werden, solange sie rein technologisch orientiert bleibt.

Nun ist aber in neuerer Zeit eine zweite Sicht der Künstlichen Intelligenz immer populärer geworden, in der sie als Teil einer umgreifenden Disziplin, der Kognitiven Wissenschaft aufgefasst wird. Die Kognitive Wissenschaft beschäftigt sich mit intelligenten Leistungen,

unabhängig davon, ob diese Leistungen von einem natürlichen oder einem künstlichen System erbracht werden.

Die Gründe, die zur Entwicklung der Kognitiven Wissenschaft führten, sind vielfältig und wir werden darauf später eingehen. Ein Grund, der auf der Hand liegt, sei aber vorweggenommen, nämlich die wachsende Erkenntnis, dass die Gegenstandsbereiche der Künstlichen Intelligenz und der kognitiven Psychologie (und auch der Linguistik im Falle des Sprachverstehens) sehr nahe miteinander verwandt sind. Aus der immer stärker werden interdisziplinären Zusammenarbeit und auf der Basis der konvergierenden Grundvorstellungen hat sich die neue Leitvorstellung entwickelt, die zur Basis der Kognitiven Wissenschaft wurde. Es zeigen sich Hinweise darauf, dass sich die Kognitive Wissenschaft zu einem eigentlichen Paradigma im Sinne von Kuhn(1962) entwickelt, mit eigenen Anforderungen an Theorien und eigenen methodischen Ansätzen, die sich langsam gegenüber einer technologisch orientierten Künstlichen Intelligenz, aber auch gegenüber der klassisch orientierten Kognitionspsychologie abgrenzen.

So wollen wir denn im folgenden auch vom Paradigma der Kognitiven Wissenschaft ausgehen und es mit der kognitiven Psychologie vergleichen. Wir wollen damit betonen, dass wir im Vergleich die Position einer rein technologisch orientierten Künstlichen Intelligenz vernachlässigen wollen.

Im ersten Teil werden wir auf die Philosophie der Kognitiven Wissenschaft näher eingehen und ihre Konsequenzen für die Bildung von Theorien intelligenter Leistungen untersuchen. Im nächsten Teil werden wir uns mit verschiedenen Fragen befassen, die alle den Zusammenhang zwischen Theorien und den Phänomenen berühren. Schliesslich wollen wir die Strategien der Theorienentwicklung diskutieren, die sich aus der Position der Kognitiven Wissenschaft ableiten lassen und sie mit den Strategien in der kognitiven Psychologie vergleichen.

2. Das Paradigma der Kognitiven Wissenschaft

In den letzten Jahren hat sich auf dem Grenzgebiet zwischen Psychologie, Künstlicher Intelligenz, Linguistik und anderen Gebieten ein neues wissenschaftliches Paradigma entwickelt, die Kognitive Wissenschaft. Ziel der Kognitiven Wissenschaft ist es, die Bedingungen und Prozesse, die zu intelligenten Leistungen führen, zu erfassen, unabhängig davon, ob diese Leistungen von einem natürlichen (biologischen) oder einem künstlichen System erbracht werden. Dabei werden sowohl Mensch als auch Maschine einheitlich als Systeme aufgefasst, die symbolische Beschreibungen der Umwelt konstruieren und diese symbolischen Beschreibungen manipulieren. Es ist eine der Grundannahmen der Kognitiven Wissenschaft, dass es einen mehr oder weniger wohldefinierten Bereich psychologischer Fragestellungen gibt, die auf dieser Ebene der symbolischen Manipulationen beschrieben werden können und zwar durch die Angabe 1) eines Algorithmus, 2) eines Systems von Symbolstrukturen (Datenstrukturen) und 3) eines einheitlichen Interpretationsschemas, das die Anwendung des Algorithmus auf die Datenstrukturen definiert (Pylyshyn, 1980). Die Auffassung von Mensch und Maschine als Symbolsysteme geht auf Newell und Simon zurück, die diese Position mehrfach ausformuliert haben (Newell & Simon, 1976, Newell, 1980).

Sie gehen aus vom Begriff eines physischen Symbolsystems, das sie folgendermassen definieren (Newell & Simon, 1976):

"A physical symbol system consists of entities called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure). Thus, a symbol structure is composed of a number of instances (or tokens) of symbols related together in some physical way (such as one token being next to another). At any instant of time the system will contain a collection of these symbol structures. Beside these structures, the system also contains a collection of processes that operate on expressions to produce other expressions: processes of creation, modification, reproduction and destruction. A physical symbol system is a machine that produces through time an evolving collection of symbol structures. Such a system exists in a world of objects wider than just these symbolic expressions themselves."

Das Adjektiv physisch hebt hervor, dass physische Symbolsysteme den Gesetzen der Physik unterworfen sind und durch technische Systeme realisierbar sind und dass sie nicht auf menschliche Systeme beschränkt bleiben.

Zwei Begriffe, die für das Verständnis physischer Symbolsysteme von zentraler Bedeutung sind, sind 'Designation' und 'Interpretation', die Newell und Simon folgendermassen definieren:

Designation: An expression designates an object if, given the expression, the system can either affect the object itself or behave in ways dependent on the object.

Interpretation: The system can interpret an expression if the expression designates a process and if, given the expression, the system can carry out the process.

Der Kern der Position von Newell und Simon ist in der Hypothese des 'physikalischen Symbolsystems' formuliert, nach der physische Symbolsysteme die notwendigen und hinreichenden Voraussetzungen für intelligentes Verhalten haben, d.h. dass jedes intelligente System sich bei genauerer Analyse als ein physisches Symbolsystem erweist und dass jedes physische Symbolsystem von genügender Grösse so organisiert werden kann, dass es allgemeine und dem Menschen entsprechende Intelligenz zeigt. Es ist nicht das Ziel dieser Diskussion, hier die Fakten zu diskutieren, die für diese Hypothese sprechen. Wir verweisen dazu auf Newell & Simon(1976). Wir wollen uns viel mehr damit beschäftigen, welche Konsequenzen sich aus der Annahme dieser Position ergeben, und wie nützlich sich diese Position für die Theoriebildung innerhalb der Psychologie erweist. Es scheint uns aber sinnvoll, zu zeigen, dass die Position der Kognitiven Wissenschaft nicht völlig neuartig ist, sondern aus einer Reihe historischer Entwicklungen in diesem Jahrhundert hervorgegangen ist.

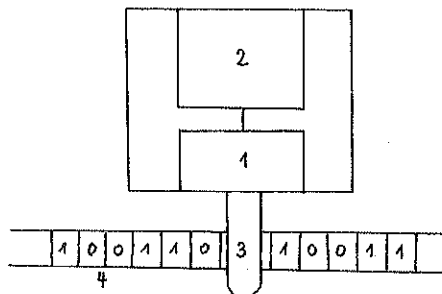
2.1 Wissenschaftsgeschichtliche Hintergründe

Die ersten Wurzeln liegen in der Entwicklung der formalen Logik, die auf den Arbeiten von Frege und von Whitehead & Russell(1910) basieren. Darin wird die Logik als ein rein formales System von Symbolen aufgefasst,

das mit syntaktischen Transformationsregeln manipuliert wird, die losgelöst von der Bedeutung der Symbole angewendet werden. Hinter der Loslösung der Logik von der natürlichen Sprache steht eine grosse Leistung, widerspricht sie doch - so meinen wir - der Intuition. Was bis dahin als eine nur durch die menschliche Intelligenz ermöglichte Leistung gegolten hatte (nämlich das Beweisen von Sätzen) konnte so durch ein mechanisches Spiel mit Symbolen ersetzt werden; ein Spiel, über dessen Regelinterpretation keine Zweifel bestehen. Damit war es auch möglich geworden, die Technik des Beweisens von Sätzen auf eine sichere Grundlage zu stellen. Entscheidend ist, dass dies nur damit erreicht werden konnte, dass die Manipulation der Symbole losgelöst von der Bedeutung der Symbole geschah. Ein Indiz dafür, welche Leistung hinter dieser Auffassung steckt, und wie schwer sie es hatte, sich durchzusetzen, zeigt sich in den späteren langanhaltenden Disputen zwischen den Formalisten (Hilbert) und Intuitionisten (Brouwer, Wittgenstein) in der Mathematik, deren Auseinandersetzung auf eben dieser Trennung basiert, haben doch die Formalisten diese Trennung vervollständigt und die mathematischen Kalküle als Kalküle mit sinnlosen Symbolen betrachtet.

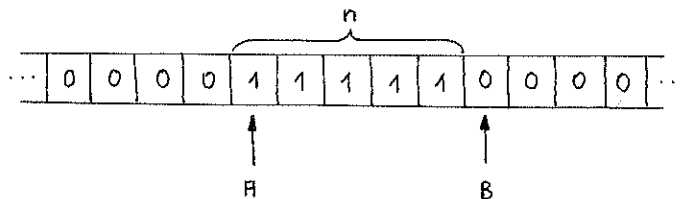
Ein weiterer wichtiger Schritt auf dem Wege zur Kognitiven Wissenschaft war die Entwicklung der Automatentheorie und der digitalen Computer, zu deren wichtigsten Begründer der britische Mathematiker Alan Turing gehört. Seine Arbeiten sind insofern wichtig, als es ihm gelungen ist, den früher nur intuitiv erfassten Begriff des Algorithmus und der Berechenbarkeit präzise zu formulieren. Die Konsequenzen dieser Errungenschaft sind so vielfältig, dass es sich lohnt, näher darauf einzugehen. Möglich wurde dies mit dem Konzept der Turingmaschine, dem Schema eines einfachen Computers. Eine Turingmaschine besteht 1) aus einer Steuereinheit, 2) aus einem endlich grossen Programmspeicher, 3) aus einem Band-Lese-, Schreib- und Steuerkopf und 4) einem unbeschränkt grossen Bandspeicher (1).

(1)



Auf dem in Felder eingeteilten Band werden Informationen in einem bestimmten Alphabet von Symbolen, zum Beispiel $A=(0,1)$ abgespeichert, während im Programmspeicher die Turingprogramme abgespeichert sind. Diese bestehen aus einer Menge von Quintupeln der Form $(S_i, Z_j, Z_{ij}, S_{ij}, B_{ij})$, die in der folgenden Weise interpretiert werden: Befindet sich die Maschine im Zustand S_i und steht unter dem Lese-/Schreibkopf das Zeichen Z_j , dann schreibt die Maschine das Zeichen Z_{ij} an die Stelle unter dem Lese-/Schreibkopf, verschiebt das Band um B_{ij} (eine Position nach rechts, eine Position nach links oder keine Verschiebung) und geht dann in den Zustand S_{ij} über. Dann beginnt der Interpretationszyklus von neuem. Befindet sich die Maschine im Zustand S' und steht das Zeichen Z' unter dem Lese-/Schreibkopf, existiert aber kein Quintupel (S', Z', \dots) im Programmspeicher, so stoppt die Maschine. An einem einfachen Beispiel, einem Programm, das die Funktion $f(n)=n+1$ mit n einer natürlichen Zahl berechnet, soll die Arbeitsweise der Maschine illustriert werden. Die Zahlen seien auf dem Band mit dem Alphabet $A=(0,1)$ dargestellt, wobei die Zahl n durch n aufeinanderfolgende '1' repräsentiert sei (2).

(2)



Das mit 'A' bezeichnete Feld befinde sich im Anfangszustand S_1 unter dem Lese-/Schreibkopf. Das folgende Turingprogramm berechnet dann die Funktion $f(n)=n+1$:

Q1: $(S_1 \ 1 \ 1 \ S_1 \ R)$

Q2: $(S_1 \ 0 \ 1 \ S_2 \ R)$

Das Programm stoppt, sobald es in den Zustand S_2 übergeht, da es kein Quintupel (S_2, \dots) gibt. Der Lese-/Schreibkopf befindet sich dann an der Position B auf dem Band, und auf dem Band stehen (zwischen A und B) $n+1$ '1'-Zeichen.

Damit ist es nun möglich, den Begriff des Algorithmus präzise zu fassen: Er wird mit einem Turingprogramm identifiziert. Ebenfalls lässt

sich der Begriff der Berechenbarkeit exakt erfassen: Eine Funktion f heisst berechenbar genau dann, wenn es ein Turingprogramm gibt, das diese Funktion berechnet.

Mit der Arbeit von Turing allein hätte man aber noch nicht sicher sein können, ob die intuitiven Begriffe des 'Algorithmus' und der 'Berechenbarkeit' adäquat erfasst worden waren, konnten sie doch als nur eine der möglichen Präzisierungen aufgefasst werden. So war es denn von entscheidender Bedeutung, dass etwa gleichzeitig zu Turings Arbeiten, aber unabhängig von ihm noch andere, ähnliche Formalisierungen entwickelt wurden, nämlich die Postschen Produktionssysteme (Post, 1943), die Markov-Algorithmen (Markov, 1951, Brauer und Indermark, 1968) und Churchs Lambdakalkül (Church, 1941). Mit den Beweisen, dass sich alle diese unabhängigen Ansätze als äquivalent erwiesen, konnte man aber überzeugt sein, dass die Begriffe des 'Algorithmus' und der 'Berechenbarkeit' adäquat erfasst worden waren.

Von ebenfalls weitreichender Bedeutung ist das Konzept der universellen Turingmaschine, die auf dem Band sowohl die Beschreibung eines Programmes T als auch die Argumente für T verlangt. Die universelle Turingmaschine kann die Funktion, für die T ein Algorithmus ist, berechnen, indem sie die Ausführung von T simuliert. Damit war erstens das Konzept der heutigen allgemeinen, d.h. zweckungebundenen Computer geboren. Zweitens konnte damit bewiesen werden, dass die universelle Turingmaschine jede Funktion berechnen kann, die von irgendeiner Maschine berechnet werden kann. Diese Folgerung ist äusserst wichtig, besagt sie doch, dass die Menge der berechenbaren Funktionen nicht von der Architektur der berechnenden Maschinen abhängt. Dies wiederum ist für die Kognitive Wissenschaft wichtig, kann doch nicht mehr argumentiert werden, dass die Leistungen der menschlichen Intelligenz nur vom Menschen oder von einem System mit der gleichen Architektur erreicht werden kann. Damit ist die wichtigste Rechtfertigung gegeben für die Trennung der Ebene der Berechnung von der Ebene des Mechanismus, der diese Berechnung unterstützt.

So weitreichend auch die Konsequenzen aus Turings Arbeiten sind, durchsetzen konnte sich diese Auffassung erst mit der Entwicklung von konkreten digitalen Computern, in der von Neumann eine grosse Rolle spielte. Im von Neumannschen Konzept eines Allgemeinzweck-Computers wurde Turings Idee der universellen Maschine konkretisiert. Die einfache Turingmaschine arbeitet mit einem fixen Programm, das eine bestimmte

Funktion berechnet. Im Gegensatz dazu enthält die universelle Turingmaschine nur ein einfaches 'Interpretations'-Programm im Programmspeicher, während die 'Berechnungs'-Programme dynamisch vom Band gelesen und interpretiert werden. Während also eine einfache Turingmaschine, zum Beispiel mit einem 'Additions'-Programm nur als eine 'Additionsmaschine' operiert, kann die universelle Turingmaschine für beliebige Anwendungen gebraucht werden. Diese Unterscheidung wurde im von Neumannschen Konzept realisiert. Das Interpretationsprogramm ist festverdrahtet (oder in modernen Computern mikroprogrammiert), während die Anwendungsprogramme dynamisch in den Hauptspeicher geladen und ausgeführt werden.

Ein weiterer wichtiger Schritt auf dem Wege zu der von der Kognitiven Wissenschaft vertretenen Position war die Entwicklung der listenverarbeitenden Programmiersprachen, und dabei insbesondere die Entwicklung der Programmiersprache LISP. LISP wurde von McCarthy (1961) auf der Basis von Churchs Lambdakalkül entwickelt und war ursprünglich nur als ein reiner Kalkül zum Ausdruck von Funktionen gedacht. LISP hat sich aber bald als eine geeignete Programmiersprache für die Künstliche Intelligenz erwiesen und sich in den Sechziger- und Siebzigerjahren als praktisch einzige Sprache durchgesetzt. LISP weist einige Charakteristika auf, die für die Entwicklung der Position der Kognitiven Wissenschaft wichtig sind. So werden Programme und Daten auf die gleiche Weise repräsentiert. Dies erlaubt, Programme durch Programme zu modifizieren und ermöglicht so die dynamische Anpassung von Programmen an ihre Anwendung. Damit ist die letzte Trennung von der Konzeption einer Maschine mit festgelegtem Programm vollzogen: Die Programme der Maschine können sich dynamisch an die Umwelt der Maschine anpassen. Damit ist aber auch die Idee noch schärfer als bisher hervorgehoben, dass die Ebene der Berechnung von der Ebene des Mechanismus, die in einem gegebenen System fixiert bleibt, getrennt werden kann.

Alle diese Entwicklungen führten schliesslich zur Auffassung der Maschinen als physische Symbolsysteme, wie sie von Newell & Simon formuliert wurde. Würde diese Sicht nur das Gebiet der Computerwissenschaft betreffen, wäre sie für die Psychologie nicht unbedingt von Bedeutung. Newell und Simon aber gehen mit ihrer physischen Symbolsystem-Hypothese einen Schritt weiter und stellen eine Verbindung zwischen dieser Sicht und jedem intelligenten System und damit auch dem Menschen her. Diese Hypothese

ist von entscheidender Wichtigkeit, gelten doch mit ihrem Zutreffen alle Resultate aus der Berechenbarkeitstheorie auch für den Menschen. Dies gilt etwa für die vielen Unlösbarkeitsbeweise, die auf diesem Gebiet gefunden worden sind. Im Gegensatz zu den früheren Unlösbarkeitsbeweisen der Mathematik und Geometrie des letzten Jahrhunderts, die die Unlösbarkeit gewisser Probleme mit bestimmten Hilfsmitteln (z.B. mit Zirkel und Lineal) zeigten, weisen die Unlösbarkeitsbeweise der Berechenbarkeitstheorie auf eine prinzipiellere Unlösbarkeit hin. Mit der physischen Symbolsystem-Hypothese wird dieser Typ von Unlösbarkeitsbeweisen präzisiert: Wenn die Hypothese zutrifft, dann sind diese Probleme für jedes intelligente Wesen unlösbar, wenn sie für den Menschen zutrifft, dann sind sie für den Menschen unlösbar. Die Frage bleibt daher, ob die Hypothese zutrifft, oder inwieweit es alternative Konzeptionen intelligenter Systeme gibt. Die erste Frage kann nicht beantwortet werden, zur zweiten ist zu sagen, dass es bis jetzt keine alternativen Ansätze in der Psychologie gibt, für die gezeigt werden konnte, dass sie für die Erklärung intelligenten Verhaltens prinzipiell genügen und die den gleichen Grad von Spezifität aufweisen wie die Theorien der physischen Symbolsysteme.

In diesem Zusammenhang muss auf den Unterschied zwischen Algorithmus und Heuristik eingegangen werden. Eine Funktion ist dann nicht berechenbar, wenn es keinen Algorithmus gibt, der für alle Argumente des Argumentbereichs den Wert der Funktion berechnen kann. Dies heißt aber nicht, dass es kein Verfahren gibt, das für eine Teilmenge des Argumentbereichs eine Teillösung liefert. Es heißt aber, dass diese Teilmenge notwendig eine echte Teilmenge ist, mit andern Worten, dass eine Lösung nicht garantiert ist. In der klassischen Terminologie heißt ein solches Verfahren ein heuristisches Verfahren. Heuristische Verfahren können auch von Maschinen angewendet werden. Diese Feststellung ist wichtig, wird doch häufig argumentiert, dass Maschinen ein bestimmtes Problem nicht lösen können, weil es nur mit Hilfe heuristischer Regeln gelöst werden könne. Wir wollen hier aber auf eine eingehendere Diskussion heuristischer Verfahren und ihrer Anwendung auf Maschinen verzichten und verweisen dazu auf Nilsson(1971) und Groner, Bischof und Groner(1982).

2.2 Die Trennung von Berechnung und Mechanismus

In der bisherigen Diskussion wurde bereits argumentiert, dass man zwischen der Ebene der Berechnung und der Ebene des diese Berechnung unterstützenden Mechanismus unterscheiden könne. Das Argument beruht darauf, dass sich keine eindeutige Beziehung zwischen der Architektur des Mechanismus und Berechnungen eines Systems herstellen lässt. Mit andern Worten kann aus der Architektur des Mechanismus nicht geschlossen werden, welche Berechnungen das System durchführt und aus den Berechnungen des Systems kann nicht geschlossen werden, welche Architektur dem System zugrundeliegen muss. Im Prinzip entspricht diese Position der Position gegen den Reduktionismus in der Psychologie. Für unseren Zweck ist es wichtig zu sehen, dass ganz ähnliche Argumente auch in der Kognitiven Wissenschaft geltend gemacht werden können.

Ein anderes Argument für die Trennung von Berechnung und Mechanismus beruht darauf, dass in die Beschreibung der beiden Ebenen grundsätzlich verschiedene Entitäten eingehen, die sich nicht ineinander überführen lassen.

Auf der Ebene des Mechanismus lässt sich ein solches System physikalisch beschreiben, falls es sich um ein künstliches System handelt, oder physiologisch (biochemisch) im Falle eines biologischen Systems. Dazu gehören unter anderem die Beschreibung der elektrischen Potentiale an den verschiedenen Stellen eines elektronischen Schaltkreises oder die biochemischen Zustände in neuronalen Schaltungen. Die Zustände des Systems sind definiert durch identische physikalische (physiologische) Beschreibungen und die Zustandsübergänge sind durch physikalische resp. physiologische Kausalzusammenhänge festgelegt. Auf dieser Ebene lassen sich durch Abstraktionen funktionale Zusammenhänge formulieren, etwa ein ODER-Gatter, ein Schieberegister in elektronischen Schaltkreisen, oder ein einfacher Reflexbogen oder lateraler Hemm-Mechanismus in neuronalen Schaltungen. Diese Beschreibungen lassen sich auf rein physikalische resp. physiologische Beschreibungen reduzieren, falls die Menge der verschiedenen, einer funktionalen Beschreibung entsprechenden physikalischen Beschreibungen beschränkt ist. Ist diese Bedingung nicht erfüllt, dann ist eine Reduktion, d.h. eine Ersetzung der funktionalen Beschreibung durch die physikalische Beschreibung nicht sinnvoll. Die

Reduktion der Ebene der Berechnung auf die Ebene des Mechanismus wird, wie wir noch sehen werden, unter anderem auch an dieser Forderung scheitern.

In die vollständige Beschreibung eines Mechanismus gehen aber auch Entitäten ein, die sich nicht sinnvoll auf die Ebene der Berechnung übertragen lassen. Dies ist darauf zurückzuführen, dass sie für einen bestimmten Mechanismus spezifisch sind und daher nicht in eine Abstraktion über alle möglichen Mechanismen hinweg übernommen werden können, für die Beschreibung dieses Mechanismus aber unabdinglich sind. Auf dem Gebiete der elektronischen Schaltkreise wären dies zum Beispiel die Beschreibung der Stromversorgung, der Wärmeeffekte oder etwa das optimale Verhältnis zwischen der Anzahl von Schaltelementen und der Länge der Leiterbahnen. Auf dem Gebiet der Neurophysiologie wären dies zum Beispiel die zeitlichen Verhältnisse und Stärken der verschiedenen Phasen von Aktionspotentialen in der neuronalen Übertragung oder die Interaktion zwischen transienten und anhaltenden (sustained) Kanälen der sensorischen Informationsübertragung (Breitmeyer&Ganz, 1976).

Von der Ebene der Berechnungen lässt sich nun zeigen, dass eine Reduktion auf die Ebene des Mechanismus nicht sinnvoll ist. Eine Reduktion der Berechnung auf die Ebene der Mechanismen müsste daher alle möglichen Mechanismen berücksichtigen. Es ist aber unplausibel, dass sich alle möglichen Mechanismen durch einheitliche physikalische Beschreibungen charakterisieren lassen, andererseits müsste dies für eine Reduktion vorausgesetzt werden. Ebenso wichtig ist aber auch die Tatsache, dass es auch auf der Ebene der Berechnung Entitäten gibt, die sich nicht sinnvoll auf die Ebene des Mechanismus übertragen lassen. Die Frage, ob eine bestimmte Funktion berechenbar ist, oder ob ein bestimmtes Problem entscheidbar ist, lässt sich nicht durch Rückgriff auf den Mechanismus eines Systems und damit durch physikalische Argumente beantworten: Diese Ebene von Problemen ist nicht durch physikalische Gesetze bestimmt.

Wir haben jetzt zwei wichtige Argumente diskutiert, die für eine Trennung der Ebenen der Berechnung und des Mechanismus sprachen, nämlich das Argument der Unabhängigkeit der Menge der berechenbaren Funktionen von der Architektur des zugrundeliegenden Mechanismus und das Argument, dass die Beschreibungen auf den beiden Ebenen verschiedene, miteinander unvergleichbare Entitäten enthalten. Damit lässt sich aber im Vergleich zwischen der Psychologie und der Künstlichen Intelligenz nicht für prinzipiell verschiedene Leistungsbereiche aufgrund der verschiedenen

Architekturen argumentieren. Die Ebene der Berechnung und die Ebene des Mechanismus hängen aber doch enger zusammen, als bisher argumentiert wurde. Darauf wird im nächsten Abschnitt eingegangen.

2.3 Komplexität von Algorithmen und funktionale Architektur

Algorithmen können dadurch charakterisiert werden, welchen Aufwand sie brauchen, um ein bestimmtes Problem zu lösen. Dieser Aufwand kann sowohl zeitlich - wieviele Schritte braucht der Algorithmus bis zur Lösung - als auch räumlich - wieviel Speicherplatz braucht der Algorithmus für die Speicherung von Zwischendaten - charakterisiert werden. Dieser Aufwand, die zeitliche und räumliche Komplexität der Algorithmen, kann als Funktion der Problemgrösse betrachtet werden, die etwa durch die Menge der Eingabedaten erfasst wird. Diese kann zum Beispiel beim Sortieren von n Zahlen mit n festgelegt werden. Für eine exakte Erfassung der Komplexität der Algorithmen müssten die erwähnten Grössen, d.h. die Schritte des Algorithmus, der Speicherbedarf und die Problemgrösse exakter definiert werden. Für einen Vergleich von Algorithmen ist man aber meist nur an einer groben Klassifizierung der Funktionen interessiert, etwa in welche Familie von Funktionen sie gehören (Polynom-, Exponentialfunktionen usw.) und bei Polynomfunktionen etwa, von welchem Grad sie sind. So haben von den Sortieralgorithmen zum Beispiel der 'Comparison Count'-Algorithmus eine Zeitkomplexitätsfunktion $O(N^2)$, der 'Shellsort'-Algorithmus $O(N^{3/2})$ oder der 'Heapsort'-Algorithmus $O(N \cdot \log N)$ (Knuth, 1975, Aho, Hopcroft und Ullman, 1974). Wie wichtig der Vergleich der Komplexitätsfunktionen ist, zeigen die Algorithmen mit exponentieller Zeitkomplexitätsfunktion, die in der Komplexitätstheorie eine grosse Rolle spielen (Garey und Johnson, 1979), steigen doch die Werte solcher Funktionen so rasch an, dass sich die Anwendung solcher Algorithmen für grössere N aus Zeitgründen verbietet. Man vergleiche dazu nur das Zeitverhalten eines $O(N^2)$ und eines $O(2^N)$ Algorithmus für verschiedene N , wenn man von einer Schrittdauer von 10^{-6} sek/Schritt ausgeht (3):

(3)

N=	10	30	50
N^2	0.0001 Sek	0.0009 Sek	0.0025 Sek
2^N	0.001 Sek	17.9 Min	35.7 Jahre

Uns interessiert die Frage, wie die Komplexität von Algorithmen durch die zugrundeliegende Architektur des Mechanismus und damit durch die Menge der zur Verfügung stehenden Primitivoperationen bestimmt wird.

Wir wollen dazu die bereits diskutierte Turingmaschine mit einer zweiten Maschine von anderer Architektur, der sogenannten 'Random Access'-Maschine (RAM) vergleichen (Aho, Hopcroft & Ullman, 1974). Die RAM verfügt über zwei Bänder, von denen sie von einem nur lesen und auf das andere nur schreiben kann. Neben dem Programmspeicher hat sie zusätzlich eine (unbeschränkt grosse) Anzahl von internen Speicherplätzen R_i , auf die sie direkt zugreifen kann. Zu den Operationen der Maschine gehören u.a.

LOAD i Inhalt von R_i nach R_0
STORE i Inhalt von R_0 nach R_i
ADD i Inhalt von R_i + Inhalt von R_0 nach R_0
MULT i Inhalt von R_i x Inhalt von R_0 nach R_0
READ i Inhalt der laufenden Position auf dem
 Leseband nach R_i
WRITE i Inhalt von R_i nach der laufenden Position
 auf dem Schreibband
JUMP l Nächste Operation in der
 Programmspeicherstelle l

Wie man sieht, entspricht RAM einer einfachen Version eines heutigen Computers. Nun unterscheiden sich die Turingmaschine und RAM nicht in ihrer Berechnungsleistung, wie auch aus der Diskussion der Turingmaschine folgt, sie unterscheiden sich aber in der Geschwindigkeit, mit der Berechnungen ausgeführt werden können. So kann man unter gewissen Einschränkungen zeigen, dass ein Algorithmus, mit Zeitkomplexitätsfunktion $O(n)$ auf der

RAM-Maschine, auf einer Turingmaschine mit der Zeitkomplexitätsfunktion $O(n^3)$ simuliert werden kann (Garey & Johnson, 1979).

Wir werden später argumentieren, dass in der Evaluierung von Theorien auf ihre psychologische Plausibilität die Komplexität der Verarbeitungsprozesse in Betracht gezogen wird. Aus der bisherigen Diskussion könnte man dann aber schliessen, dass dazu für die beiden zu vergleichenden Systeme, den Menschen und die simulierende Maschine, die gleiche Architektur vorausgesetzt werden müsste, da sich sonst die Komplexitätsfunktionen bei gleicher Berechnungsart aufgrund der verschiedenen Architekturen unterscheiden könnten. Um dieses Problem umgehen zu können, geht man in zwei Schritten vor: Um einen Prozess auf dem System M zu simulieren, wird auf dem System C erst ein System M' mit der Architektur von M simuliert, und dann auf M' die Prozesse von M simuliert. Damit sind die Prozesse auf M und M' in bezug auf ihre Komplexität wieder vergleichbar. Eine solche Simulation einer Maschine B auf einer Maschine A nennt man in der Computerwissenschaft Emulation.

Damit wird aber zwischen die Ebene des Mechanismus und die Ebene der Berechnung eine weitere Ebene eingeschoben, nämlich die Ebene der emulierten Maschine mit einer die Berechnung bestimmenden, eigenen Architektur, die von Pylyshyn (1980) als funktionale Architektur bezeichnet wird. In dieser Arbeit wurden mehrere solcher funktionaler Architekturen erwähnt. So kann die Breadth-first-Suche von Wilks' Parser (Abschnitt III.4) als eine Emulation einer parallelen Maschine betrachtet werden. Und die Philosophie des Marcus-Parsers ist entscheidend mit der Definition eines eingeschränkten Automaten verbunden, dient er doch dazu, das Konzept des deterministischen Parsens zu präzisieren. Aber bereits auf tieferer Stufe werden bestimmte funktionale Architekturen emuliert, nämlich auf der Stufe der Programmiersprachen (LISP, PLANNER, PASCAL usw.). So beruht zum Beispiel die Programmiersprache PASCAL auf der Definition einer hypothetischen Stackmaschine, der sogenannten P-Maschine. Die Uebertragung von PASCAL auf einen neuen Computer wird dadurch sehr vereinfacht: Man braucht auf der Zielmaschine nur die P-Maschine zu emulieren und kann dann PASCAL auf dieser emulierten Maschine ausführen. LISP und PLANNER und einige andere in der Künstlichen Intelligenz verwendete Programmiersprachen definieren ihrerseits Maschinen, die für die Zwecke der Künstlichen Intelligenz besonders geeignet sind. Sie unterscheiden sich von den übrigen Programmiersprachen dadurch, dass ihre Architektur vom von Neumannschen

Konzept zum Teil beträchtlich abweicht.

Für unsere Diskussion ergibt sich daraus die Konsequenz, dass man im Vergleich zwischen den menschlichen Leistungen und den Leistungen eines entsprechenden Programms auf einem Computer nicht einfach vom äusseren, zum Beispiel dem reellen zeitlichen, Verhalten der Maschine ausgehen kann. Das Verhalten der Programme muss immer in Relation zu der ihnen zugrundeliegenden funktionalen Architektur analysiert werden, und der Erklärungswert von Theorien, die auf solchen Programmen basieren, ist immer relativ zur funktionalen Architektur bestimmt. Obwohl wir der Diskussion etwas vorgreifen, wollen wir ein Beispiel geben. Die Erklärung des Phänomens der Gartenpfadsätze im Parser von Marcus ist nur dann als psychologische Erklärung akzeptabel, wenn der menschliche Mechanismus der Sprachverarbeitung die gleiche Architektur aufweist wie der einfache Automat, der dem Parser von Marcus zugrundeliegt. Dies soll die Leistung von Marcus nicht schmälern, sagt er doch in erster Linie aus, dass unter den von ihm gesetzten Randbedingungen (d.h. seiner postulierten Architektur) eine einfache Erklärung dieses Phänomens möglich ist. Die Frage aber, ob es die richtige psychologische Erklärung ist, hängt auch davon ab, ob die von ihm angenommene funktionale Architektur sich auch im Menschen nachweisen lässt.

Während wir bei künstlichen Systemen eine Trennung zwischen der durch den Mechanismus festgelegten Architektur und der emulierten Architektur befürworten, halten wir eine solche Trennung bei biologischen, durch Evolution entwickelten Systemen nicht für sinnvoll. Bei Maschinen ist diese Trennung eine Konsequenz aus der von Neumannschen Konzeption einer Allgemeinzweck-Maschine. Bei biologischen Systemen ist es aber eine sinnvolle Annahme, dass sich die Struktur der Organe, besonders auch des Gehirns, im Verlaufe der Evolution an ihre Funktion angepasst hat, oder dass jene Systeme mit der am besten angepassten Architektur überlebt haben. Ein System, das ein anderes System mit verschiedener Architektur emuliert, ist aber benachteiligt, weil die Emulation selbst schon zeitaufwendig sein kann.

Wenn man aber zur Beschreibung biologischer Systeme auf das Konzept der emulierten Architektur verzichten kann, dann sind Untersuchungen der Architektur, wie sie von Biologen, Neuroanatomen und Neurophysiologen gemacht werden, wieder sinnvoll. Wenn man auch die Ebene der Berechnung von der Ebene des Mechanismus trennen kann, so lassen sich doch gewisse, wenn

auch sehr beschränkte Uebertragungen von einer Ebene auf die andere machen. In diesem Sinne wollen wir denn auch keineswegs die Wichtigkeit neurophysiologischer Untersuchungen für die Psychologie, und bedingt auch für die Kognitive Wissenschaft, bestreiten.

Aber auch von Seiten der Künstlichen Intelligenz muss langfristig eine etwas andere Position eingenommen werden. Die Emulation einer andern Maschine hat den Vorteil, dass sie billig ist, weil sie relativ schnell programmiert werden kann. Sie hat aber den Nachteil, dass sie langsam ist, ein Nachteil, der mit zunehmender technologischer Ausbeutung der Errungenschaften der Künstlichen Intelligenz immer schwerer ins Gewicht fällt. So zeigt sich heute ein immer stärkerer Trend dahin, spezifische, nicht-von Neumannsche Architekturen zu realisieren, ausgehend von LISP-Maschinen bis zu spezifischen Systemen zur Wissensrepräsentation (Fahlman, 1979). Die Entwicklungen in dieser Richtung sind aber aus psychologischer Richtung zurückhaltend zu bewerten. Einerseits ermöglichen sie die prinzipielle Möglichkeit direkter Mensch-Maschine-Vergleiche, andererseits muss aber berücksichtigt werden, dass die Leitlinien für die Entwicklung dieser Maschinen nur am Rande durch psychologische Befunde beeinflusst sind und so das Probleme des Architekturvergleichs zwischen Mensch und Maschine bestehen bleibt.

3. Gegenstandsbereich, Erklärung, Ueberprüfung

Ein wissenschaftliches Paradigma ist dadurch gekennzeichnet, 1) welche Fragen an die Natur gestellt werden, 2) welche Antworten als akzeptabel beurteilt werden und 3) welche Methoden verwendet werden, um Antworten zu finden.

In der Künstlichen Intelligenz ergeben sich die Fragen aus Aufgaben, die zu ihrer Lösung Intelligenz verlangen. Die Frage lautet, wie ein System aussieht, das diese Aufgaben lösen kann. Die Antwort besteht in einem Programm, das die Aufgaben lösen kann und auf einem Computer implementierbar ist. Die Methode zum Finden dieser Antwort besteht in einem iterativen Zyklus von Programmentwurf, Beobachtung des Programmverhaltens mit verschiedenen Aufgabeninstanzen, Diagnose von Programmfehlern und Leistungsschwächen und Entwurf einer verbesserten Programmversion.

In der Kognitiven Wissenschaft lautet die Frage allgemeiner, nämlich wie ein beliebiges, ob biologisches oder künstliches, System aussehen muss, das diese Aufgabe lösen kann. Die Antwort besteht in einem Algorithmus oder Algorithmenschema, das auf jedem symbolmanipulierenden System implementiert werden kann. Die Kognitive Wissenschaft baut unter anderem auf der Künstlichen Intelligenz und der Psychologie auf und übernimmt frei aus diesen Disziplinen angemessene Methoden.

In diesem Abschnitt wollen wir auf der Grundlage dieser einfachen Beschreibung aufbauen und versuchen, die Arbeitsweise der Kognitiven Wissenschaft im Vergleich zur Kognitiven Psychologie genauer zu charakterisieren. Eine präzise Charakterisierung ist allerdings nicht möglich, ist doch das Paradigma der Kognitiven Wissenschaft erst im Aufbau begriffen. Aber ebenso wenig ist dies in der kognitiven Psychologie möglich, die eine unübersehbare Vielfalt von methodischen und theoretischen Ansätzen aufweist. Dasselbe gilt für die Künstliche Intelligenz, in der es fließende Uebergänge von einer theoretisch orientierten Disziplin zu einer rein technologisch orientierten Disziplin gibt. In der Kognitiven Wissenschaft ist eine präzise Charakterisierung auch deshalb schwierig, weil sie auf den andern Disziplinen aufbaut und weil es noch völlig offen ist, welche der übernommenen methodischen und theoretischen Ansätze sich bewähren werden. Trotzdem ist es sinnvoll, wenigstens eine ansatzweise Charakterisierung zu versuchen.

3.1 Gegenstandsbereich

Das erste Problem, das wir angehen wollen, ist die Frage nach dem Gegenstandsbereich der Kognitiven Wissenschaft. Die Frage beinhaltet mehrere Teilaspekte. Erstens ist es eine Frage danach, mit welchen Leistungen eines Systems sich die Kognitive Wissenschaft auseinandersetzt. Zweitens ist es eine Frage danach, in welcher Form ein Problem gegeben sein muss, damit eine Antwort überhaupt gefunden werden kann. Drittens ist es eine Frage danach, welche empirisch beobachtbaren Phänomene in der Kognitiven Wissenschaft eine Rolle spielen.

Als Probleme werden in der Kognitiven Wissenschaft Probleme gewählt, die zu ihrer Lösung Intelligenz verlangen. Eine präzise Umschreibung des Gegenstandsbereichs der Kognitiven Wissenschaft würde daher eine präzise Definition von 'Intelligenz' voraussetzen. Es ist aber erst ein Ziel der Kognitiven Wissenschaft, den Begriff 'Intelligenz' näher zu präzisieren. Sie will also nicht auf einen von der Psychologie übernommenen Intelligenzbegriff aufbauen. Daraus folgt, dass der Gegenstandsbereich der Kognitiven Wissenschaft nicht präzise definiert sein kann.

Ein anderer Weg der Umschreibung des Gegenstandsbereichs besteht darin, dass man schaut, was für Aufgaben gewählt werden. Waren es in den Anfängen der Künstlichen Intelligenz noch Aufgaben wie das Lösen von Schach-Endproblemen, das Lösen von Integralen oder das Finden einfacher Beweise in der Logik, wurden mit der Zeit immer schwierigere Aufgaben angegangen, wie die Interpretation von Massenspektrogrammen, das Finden interessanter mathematischer Konzepte, das Verstehen und Zusammenfassen von Texten oder die Interpretation visueller Daten. Wären nun alle diese Aufgaben voneinander unabhängig und könnten durch je verschiedene, spezifische Methoden gelöst werden, ohne dass sich Gemeinsamkeiten zeigten, hätte sich kaum das Paradigma der Kognitiven Wissenschaften entwickeln können. Denn es ist ja nicht ihr Ziel, Theorien einzelner isolierter Leistungen aufzustellen. Bestimmend für die Herausbildung des Paradigmas der Kognitiven Wissenschaft war es daher, dass immer wieder die gleichen Probleme auftauchten und immer wieder gleiche Techniken zum Erfolg führten. Dazu gehören unter anderem die Beobachtung, dass zur Erreichung der

geforderten Leistungen auf vielfältiges Wissen zurückgegriffen wird, das Problem, wie dieses Wissen repräsentiert, organisiert und verwendet wird, die Entdeckung, dass viele intelligente Leistungen als Suchprozesse charakterisiert werden können, und dass die Suchräume im allgemeinen so gross sind, dass ein Auffinden einer Lösung nur unter Verwendung heuristischer Regeln möglich ist.

Es zeigt sich, dass der Gegenstandsbereich der Künstlichen Intelligenz viel eher durch die gemeinsamen Probleme und die gemeinsamen Lösungswege charakterisiert werden kann. Das führt nun aber zu einem Zirkel: Die Fragen, die gestellt werden, werden durch die Antworten, die auf diese Fragen gefunden werden, bestimmt. Dies ist nicht untypisch für ein sich entwickelndes Paradigma: Für einige erste Fragen werden Antworten gefunden, deren Allgemeinheit und Güte an neuen Fragen überprüft werden. Dann beschäftigt man sich immer mehr mit solchen Fragen, für die die entwickelten Methoden besonders geeignet sind.

Die Algorithmisierbarkeit eines Prozesses stellt gewisse Anforderungen an die Form, in der Probleme gestellt werden können. Dies ist in unserer Diskussion der Anforderungen an eine Theorie des Sprachverstehens deutlich geworden. So ist das Verstehen ein Prozess, in dem eine natürliche Sprache in eine interne (semantische) Repräsentation übersetzt wird. Diese ist aber einer empirischen Untersuchung nur sehr schwer zugänglich und ihre Eigenschaften können nur unvollständig und indirekt über äusseres Verhalten erschlossen werden. Die Entwicklung eines Algorithmus des Verstehensprozesses aber erfordert eine präzise Formulierung des Prozessproduktes, der semantischen Repräsentation. Dies führt dazu, dass in einer solchen Theorie eine ganze Reihe von Annahmen steckt, die empirisch kaum oder nur sehr schwer zu überprüfen sind. Im Gegensatz dazu ist man in der Psychologie gezwungen, möglichst alle Annahmen, die hinter einer Theorie stecken, empirisch zu belegen, oder wenigstens aufzuweisen, wie diese Annahmen überprüft werden können, will man sich nicht den Vorwurf gefallen lassen, man betreibe Metaphysik. Man steht damit dem Problem gegenüber, wie man den Konflikt zwischen dem Anspruch der Kognitiven Wissenschaft nach Algorithmisierung eines Prozesses und dem Anspruch der Psychologie nach empirischer Belegbarkeit der zugrundeliegenden Annahmen lösen kann.

Wir werden im folgenden anhand von vier Punkten argumentieren, dass sich die Position der Kognitiven Wissenschaft, in der viele empirisch nur

schwer überprüfbare Annahmen notwendig sein können, auch vom Standpunkt der Psychologie aus vertreten lässt.

1. Wir haben argumentiert, dass es das Ziel der Kognitiven Wissenschaft sei, die notwendigen Eigenschaften von Prozessen für bestimmte Klassen von Leistungen zu finden. Es ist unplausibel, dass dieses Ziel in einem Schritt erreicht werden kann. Als erstes Teilziel kann daher die Formulierung eines spezifischen, für diese Leistungen suffizienten Modells gesetzt werden. Erst wenn man suffiziente Modelle entwickelt hat, kann man durch Vergleich von Modellen Schlüsse auf allgemeine Eigenschaften der Prozesse ziehen. Wir glauben nicht, dass dies mit partiell ausformulierten und nur für spezifische Leistungen ausgelegten Modellen möglich ist. Dies trifft aber in gleichem Masse auch auf die kognitive Psychologie zu. Es scheint auch dort wenig sinnvoll, zwischen Alternativmodellen entscheiden zu wollen, deren Suffizienz für bestimmte Leistungen nicht nachgewiesen ist. Die Erarbeitung von spezifischen, suffizienten Modellen geht aber meist mit der Einführung von unüberprüfbaren Annahmen einher.

2. In der Diskussion der psychologischen Experimente zu den semantischen Netzwerken haben wir gesehen, dass die Ursachen von Reaktionszeitunterschieden an verschiedenen Stellen in den Modellen lokalisiert werden können, etwa in der Struktur der internen Repräsentation (Collins & Quillian, 1969), im Prozess der Vergleichsoperation (Conrad, 1972) oder in den Eigenschaften der Repräsentationsprimitiva selbst (Rips et al., 1973). Eine Entscheidung zwischen diesen Alternativmodellen scheint auf rein empirischer Basis kaum möglich. Trotzdem ist man in der Entwicklung eines suffizienten Modelles gezwungen, auf spezifischen Teilmodellen aufzubauen. Man darf dann aber den Status solcher Teilmodelle nicht falsch bewerten. Ihre psychologische Validität ist in dieser Phase der Entwicklung weniger wichtig als ihre Leistungsfähigkeit und ihre Eignung, eine Basis für allgemeinere Modelle zu bilden. Gerade hierin werden in der neueren kognitiven Psychologie die Modelle der Künstlichen Intelligenz oft falsch bewertet. Zu Modellen, die in spezifischen experimentellen Paradigmata überprüft wurden, werden dann Alternativmodelle entwickelt, die wohl mit den in diesem Paradigma gefundenen Daten besser übereinstimmen, die aber aus der Sicht allgemeiner Theorien äusserst fragwürdig sind. Dies ist etwa der Fall mit Meyers (1970) Alternativmodell zu Collins & Quillian, in dem z.T. exhaustive Suchen im semantischen Knotenraum durchgeführt werden. Andererseits werden in diesen Alternativmodellen Vereinfachungen

eingeführt, die im Kontext eines spezifischen experimentellen Paradigmas wohl gerechtfertigt sind, aus der Sicht der Entwicklung eines leistungsfähigen und suffizienten Modells aber einen Rückschritt bedeuten. Dies ist eindrücklich klar geworden im Vergleich der Arbeiten über semantische Netzwerke in der Künstlichen Intelligenz und der Psychologie. In der Künstlichen Intelligenz wurden immer komplexere Formalismen eingeführt, um eine adäquate Repräsentation zu ermöglichen. In der kognitiven Psychologie hingegen wurden immer einfachere Modelle verwendet, mit denen empirische Effekte wohl besser vorausgesagt werden konnten, die an Repräsentationskraft im Vergleich zu Quillians ursprünglichem Formalismus aber ungleich viel schwächer waren. Kurzfristig gesehen, d.h. im Hinblick auf die Erklärung von spezifischen, experimentellen Effekten, bedeuten sie einen Fortschritt, langfristig, d.h. im Hinblick auf die Entwicklung eines suffizienten Sprachmodells, bedeuten sie einen Rückschritt.

3. Im Abschnitt über die Ueberprüfung von Modellen werden wir argumentieren, dass eine vollständige empirische Ueberprüfung von Modellen unter gewissen Randbedingungen prinzipiell nicht gewährleistet werden kann. Es ist schwer abzuschätzen, wie weit diese Bedingungen in der Kognitiven Wissenschaft erfüllt sind. Wo dies aber der Fall ist, ist man gezwungen, gewisse Annahmen nach nicht-empirischen Kriterien zu setzen, etwa nach ihrer intuitiven Plausibilität oder nach ihrer Einfachheit. Eine Theorie wird dann relativ zu diesen gesetzten Annahmen evaluiert, so wie wir es in der Diskussion der verschiedenen Parsermodelle in Bezug auf die gesetzten Forderungen an eine Theorie des Sprachverstehens gemacht haben. Dies ist der Fall mit der Entscheidung darüber, ob ein Verarbeitungsprozess seriell oder parallel abläuft. So haben unter anderen Townsend (1971, 1972) und Anderson (1976) gezeigt, dass man allein auf Grund von chronometrischen Studien nicht entscheiden kann, ob ein Prozess seriell oder parallel ablaufen muss. Wenn nämlich die Parameter eines seriellen und eines entsprechenden parallelen Modells beliebig gewählt werden können, dann ist es möglich, sie so festzulegen, dass die beiden Modelle zu gleichen zeitlichen Voraussagen führen. So ist denn die Entscheidung darüber, ob der Prozess seriell oder parallel abläuft, davon abhängig, welche zusätzlichen Annahmen über diesen Prozess als plausibler beurteilt werden.

4. In einem späteren Abschnitt über die schrittweise Verfeinerung von Theorien werden wir ein weiteres Argument anführen, das gegen eine

Entwicklung von Theorien mit minimalen und nur empirisch belegbaren Annahmen spricht.

Alle diese Argumente führen zum Schluss, dass der Gegenstandsbereich der Kognitiven Wissenschaft nicht durch eine wohldefinierte Menge von empirischen Phänomenen gegeben ist. Erstens hat sich gezeigt, dass der Bereich der Phänomene, auf die das Paradigma der kognitiven Wissenschaft anwendbar ist, nicht oder zumindest noch nicht umrissen werden kann. Zweitens kommt man unter den Gegebenheiten der Kognitiven Wissenschaft nicht darum herum, in der Theorienbildung zu den empirischen Fakten zusätzliche Annahmen in Betracht zu ziehen, deren empirische Ueberprüfung im einzelnen oder innerhalb der Gesamtheit von Annahmen nicht mehr gewährleistet werden kann. Wie weit dies auch für die kognitive Psychologie zutrifft, ist nicht klar. Zumindest gibt es aber Hinweise, dass in einigen Fällen auch hier die gleiche Situation gegeben ist.

3.2 Ueberprüfung von Theorien

Theorien dienen zwei Zwecken, erstens Voraussagen für Experimente abzuleiten, und zweitens Phänomene erklären zu können. Mit dieser Unterscheidung sind unterschiedliche Anforderungen verbunden, die an eine Theorie gestellt werden müssen. Zum Zwecke der Voraussage von Experimenten genügt eine Theorie dann, wenn zu einem bestimmten Zustand Z1 eines Prozesses der richtige Folgezustand Z2 abgeleitet werden kann. Dabei ist nebensächlich, ob die Kausalkette des Prozesses, die von Z1 nach Z2 führt und die Ableitung dieses Uebergangs in der Theorie miteinander übereinstimmen. In andern Worten, es wird nur gefordert, dass die Theorie für die gleichen Input-Bedingungen Z1 die gleichen Output-Bedingungen Z2 wie die Natur liefert. In Anlehnung an die formale Grammatiktheorie wird diese Bedingung 'schwache Aequivalenz'-Bedingung genannt, zu vergleichen mit zwei Grammatiken, die die gleiche Sprache erzeugen, deren Tiefenstruktur (deren Ableitungen) aber nicht übereinstimmen müssen.

Im Gegensatz dazu wird zum Zwecke der Erklärung an eine Theorie eine schärfere Forderung gestellt. Nicht nur soll sie das gleiche Input-Output-Verhalten zeigen wie die Natur; die Ableitung der Folgezustände soll auch auf die gleiche Art geschehen. In Anlehnung an die

formale Grammatiktheorie wird diese Forderung 'starke Aequivalenz'-Forderung genannt.

In diesem Abschnitt wollen wir zwei Problemkreise diskutieren, nämlich erstens, wie die schwache Aequivalenz von Theorien festgelegt werden kann und zweitens, inwieweit die Forderung nach starker Aequivalenz erfüllt werden kann.

In der Kognitiven Wissenschaft stellt sich das Problem in der Form, dass eine Aufgabe gewählt wird, die zu ihrer Lösung Intelligenz verlangt und dass ein Algorithmus gesucht wird, der das gleiche Verhalten zeigt, wie ein Mensch, der mit dieser Aufgabe konfrontiert wird. Für einen Vergleich des Verhaltens muss zuerst einmal festgelegt werden, wie das Verhaltensrepertoire der beiden Systeme beschrieben und welche Verhaltenselemente für den Vergleich in Betracht gezogen werden. In der Terminologie der Grammatiktheorie muss also zuerst das Alphabet der erzeugten Sprachen festgelegt werden. Eine eingehende Diskussion dieses Problems wird auf später verlegt. Hier wollen wir das Problem des Verhaltensvergleichs zuerst aus allgemeinerer Sicht betrachten.

Das Problem des Verhaltensvergleichs liegt darin, Kriterien festzulegen, wann man sinnvoll von gleichem Verhalten sprechen soll. In einigen Fällen kann dies einfach sein, etwa bei Aufgaben wie dem Lösen mathematischer Probleme oder der Interpretation von Massenspektrogrammen. Gleiches Verhalten liegt dann vor, wenn für das gleiche Problem die gleiche Lösung produziert wird. Unterschiede in den Formulierungen des zu produzierenden Verhaltens, bei mathematischen Problemen etwa umgeformte Lösungen, sind trivial und die Menge der äquivalenten Lösungen ist formal definierbar.

Schwieriger gestaltet sich der Vergleich von Verhaltensweisen dort, wo sehr verschiedene Verhaltensweisen möglich sind und wo keine scharfen Kriterien angegeben werden können, welche Verhaltensweisen als akzeptabel beurteilt werden sollen. So ist beim Schachspiel wohl die Syntax der Züge definiert - ein Bauer darf nicht rückwärts gezogen werden - innerhalb der syntaktisch möglichen Spielkombinationen gibt es aber kein scharfes Kriterium, welche Spielweisen als intelligent beurteilt werden sollen. Im Schachspiel gibt es ziemlich genaue Konventionen darüber, wie die Stärke eines, auch künstlichen, Spielers beurteilt wird. Wann ist aber das Ziel eines intelligent spielenden Algorithmus erfüllt? Wenn das Programm so gut spielt wie ein guter Amateur, wie dies heute der Fall ist, oder erst wenn

es Weltmeister ist? Es lässt sich kein scharfes Kriterium angeben, ein jedes wäre beliebig.

Noch schwieriger wird der Leistungsvergleich etwa bei sprachlichem Verhalten. Aus der Linguistik ist bekannt, dass bereits das Kriterium der syntaktischen Korrektheit nicht exakt erfasst werden kann. Aber vor allem lässt sich zu syntaktisch korrekten Äußerungen nicht wie beim Schachspiel ein mehr oder weniger gut umschriebenes Kriterium angeben, nach dem die relative Leistungsfähigkeit von Programmen beurteilt werden kann. Als einziges Kriterium kann dann noch gelten, ob die sprachlichen Äußerungen in einer bestimmten Situation akzeptabel sind. Dies mag wohl in eingeschränkten Situationen, in denen nur wenige Äußerungen sinnvoll sind, angemessen sein, wie etwa dort, wo inhaltliche Fragen über einen gelesenen Text beantwortet werden müssen (Charniak, 1976a). In Situationen aber, in denen die Menge der akzeptablen Verhaltensweisen relativ gross ist, wie etwa in offenen Dialogen, wird das Akzeptabilitätskriterium zu einem sehr schwachen Kriterium.

Etwas stärker ist das Diskriminationskriterium, das dem klassischen Turingtest (Turing, 1950) zugrunde liegt. Im Turingtest interagiert ein unabhängiger Beobachter in zufälliger Reihenfolge mit einem Menschen und mit einem Programm. Ist es dem Beobachter nicht möglich, signifikant häufiger richtig zu entscheiden, ob er mit dem Programm oder mit dem Menschen interagiert, erfüllt das Programm die erforderliche Leistung intelligenten Verhaltens. Mit dem Turingtest ist aber eine ganze Reihe von Problemen verbunden.

Die Interaktionen im Turingtest müssen so gewählt werden, dass sie für intelligentes Verhalten symptomatisch sind. Dies illustriert Gunderson (1964) mit seinem modifizierten Turingtest, in dem entschieden werden soll, ob Steine denken können. Einem unabhängigen Beobachter tritt entweder ein Mensch auf den Fuss oder es wird ihm ein Stein auf den Fuss fallen gelassen. Ist es dem Beobachter nicht möglich, signifikant häufig richtig zu entscheiden, ob ihm der Mensch oder der Stein auf den Fuss tritt, erfüllt der Stein die erforderliche Leistung und man kann schliessen, dass Steine denken können.

Der Turingtest kann durch geeignete Manipulationen zu einem schwachen Testkriterium werden. So ist es nicht von ungefähr, dass Colby (1975) für seine Simulationen einen Paranoiker und einen Gesprächstherapeuten wählt, von denen man vom einen ohne weiteres deviante Äußerungen und vom anderen

keine grosse Eigeninitiative im Gespräch erwartet. Mit der Manipulation der Erwartungen des Beobachters ist es möglich, sein Akzeptabilitätskriterium herabzusetzen. Und die fehlende Erfahrung im Gespräch mit Parnaoikern oder Gesprächstherapeuten kann dazu führen, dass ein Beobachter auch nur über ein schwaches Diskriminationskriterium verfügt.

Die mangelnde Interaktionsfähigkeit dieser Programme wird deutlich zu einem Symptom einer Leistungsschwäche, wenn man sie gegeneinander (oder miteinander) laufen lässt, wie dies, leider nicht publiziert, mehrfach gemacht wurde. Dort zeigt sich, dass die beiden Programme nur über ein äusserst beschränktes Repertoire von Aeusserungen verfügen und dieses Repertoire in stereotyper Weise anwenden. Damit wird ein fundamentales Problem sichtbar, nämlich wie man die schwache Aequivalenz des Verhaltens von zwei Systemen festlegen soll. Um die schwache Aequivalenz von zwei Grammatiken festzulegen, genügt es nicht, zu prüfen, ob endlich viele gleiche Zeichenketten erzeugt werden. Noch weniger schlüssig wird der Vergleich, wenn mit beiden Grammatiken quasizufällig Zeichenketten erzeugt werden, aufgrund derer man eine Aussage über die Aequivalenz der Grammatiken abgeben soll. In der formalen Grammatiktheorie basiert der Aequivalenzbeweis von zwei Grammatiken, falls er überhaupt möglich ist, auf der Analyse der Grammatikregeln. In analoger Weise kann ein Leistungsvergleich von zwei Systemen nur über eine Analyse ihrer inneren Funktionsweise, ihrer Regeln und Teilprozesse geschehen.

Im besonderen ist aber der Turingtest schlecht geeignet, partiell ausformulierte Modelle zu prüfen und dies aus zwei Gründen. Erstens kann es schwierig sein, eine Situation herzustellen, in der für den unabhängigen Beobachter realistische Bedingungen herrschen, die ihm eine adäquate Beurteilung erlauben. Zweitens kann man dazu neigen, Komponenten in ein Teilmodell einzubauen, die nur die Funktion haben, das Verhalten des Programms natürlicher erscheinen zu lassen, die sich aber in der zugrundeliegenden Theorie nur schwer rechtfertigen lassen.

Die angeführten Probleme des Turingtests wollen nicht besagen, dass er ein wertloses Instrument der Kognitiven Wissenschaft ist. Er ist nur als Instrument für die Ueberprüfung der psychologischen Validität von Programmen schlecht geeignet. Hingegen kann eine dem Turingtest gleiche Situation geeignet sein, die Leistung eines Programms zu beurteilen. Dies zeigen unter anderem die Studien mit dem medizinischen Diagnoseprogramm MYCIN (Shortliffe,1976, Feigenbaum,1982), in denen Experten die Leistung

von Aerzten und von MYCIN beurteilten, in Unkenntnis darüber, von wem eine jede Diagnose stammt. Damit konnte erreicht werden, dass mögliche Vorurteile der Experten in der Leistungsbeurteilung ausgeschaltet werden.

Die Probleme, die sich in der Ueberprüfung der schwachen Aequivalenz einer Theorie ergeben, treffen in gleichem Masse auch auf die Ueberprüfung der starken Aequivalenz zu, auf die wir nun eingehen wollen. Wir haben gesagt, dass eine Theorie zum Zwecke der Erklärung mit dem erklärten Prozess stark äquivalent sein muss. Wir wollen zuerst zeigen, dass unter gewissen Bedingungen diese Forderung prinzipiell nicht erfüllt werden kann. Dann werden wir eine abgeschwächte Forderung für den Erklärungswert einer Theorie einführen, die Komplexitätsäquivalenz, und ihre Probleme diskutieren.

Aus der formalen Grammatiktheorie ist bekannt, dass die starke Aequivalenz nicht für alle Paare von kontext-sensitiven Grammatiken bewiesen werden kann (Gross und Lentin, 1971). Falls daher ein Verhalten in seiner Komplexität einer kontext-sensitiven Grammatik entspricht, ist das Finden eines starken Aequivalenzbeweises für ein Modell nicht mehr garantiert. Der Nachweis, dass ein bestimmtes Verhalten nur durch eine kontext-sensitive Grammatik beschrieben werden kann, dürfte schwierig sein, würde er doch eine vollständige Formalisierung der Verhaltensbeschreibung verlangen. Ein etwas schwächeres Argumente geht dahin, dass die Regeln, die das Verhalten steuern, zum Nachweis der starken Aequivalenz vollständig bekannt sein müssten, eine Forderung, die aus der Sicht der heutigen Situation in der kognitiven Psychologie mit den relativ schwachen, zur Verfügung stehenden Instrumentarien als unhaltbar zu erachten ist.

Als ein abgeschwächtes Kriterium schlägt Pylyshyn(1980) die sogenannte 'Komplexitäts-Aequivalenz' vor. Danach sind zwei Systeme dann komplexitätsäquivalent, wenn sie erstens das gleiche Input-Output-Verhalten zeigen und zweitens die Komplexität zur Produktion eines bestimmten Verhaltens bei beiden Systemen gleich ist. Theoretisch kann man die Komplexität einer erzeugten Zeichenkette mit der Anzahl von Ableitungsregeln zur Produktion dieser Zeichenkette identifizieren. In der kognitiven Psychologie taucht dieses Konzept verschiedentlich auf, so etwa unter dem Begriffe des 'processing load' (Norman und Bobrow,1975),der als Mass der Komplexität und Grösse der Datenstrukturen, auf denen ein Modell operiert, aufgefasst wird (Pylyshyn,1980). Ein frühes Beispiel sind die Untersuchungen zur Theorie der Ableitungskomplexität in der

Psycholinguistik (Savin und Perchonok, 1965, McKean, Slobin und Miller, 1962). In diesen Arbeiten wurde geprüft, ob sich ein Zusammenhang zwischen der Ableitungskomplexität von Sätzen im Rahmen des Chomskyschen Ansatzes und der Belastung der menschlichen Verarbeitungsmechanismen durch diese Sätze nachweisen lässt.

Die Interpretation der Verarbeitungskomplexität in einer Theorie hängt kritisch davon ab, ob sie zuverlässig gemessen werden kann und beruht daher auf einer empirischen Hypothese über den Zusammenhang zwischen der Verarbeitungskomplexität und einem beobachtbaren Phänomen. Als ein solches Indikatorphänomen wird häufig die Reaktionszeit aufgefasst, die als eine Funktion der Anzahl von Verarbeitungsschritten in einem Prozess angesehen wird. Anhand der Reaktionszeit lassen sich die Probleme aufzeigen, die mit der Erfassung der Komplexität eines Prozesses verbunden sind.

Erstens einmal kann die Reaktionszeit nur als ein relatives Mass aufgefasst werden, d.h. relativ zu einer hypothetischen funktionalen Architektur eines Systems mit einer festgelegten Menge von Primitivoperationen. Ein direkter Modellvergleich auf der Basis von Reaktionszeiten ist daher nur möglich, wenn beide Modelle die gleiche Architektur mit den gleichen Operationen postulieren. Unter weniger scharfen Bedingungen gilt nicht einmal unbedingt, dass Reaktionszeiten bis auf eine lineare Transformation vergleichbar sind, wie es etwa Pylyshyn(1980) formuliert. Abhängig von der zugrundeliegenden funktionalen Architektur können die Zeiten für den gleichen Algorithmus auf zwei verschiedenen Systemen in einer nichtlinearen Beziehung stehen, wie die Diskussion in 2.3 gezeigt hat. Ein absoluter Zeitvergleich ist daher für zwei beliebige Systeme nur möglich, wenn erstens die funktionalen Architekturen beider Systeme bekannt sind und zweitens die Zeiten für die Primitivoperationen auf beiden Systemen gleich geeicht werden können. In der Praxis wird die Interpretation von Reaktionszeiten noch dadurch erschwert, dass sie nicht als Funktion eines bestimmten Teilprozesses, sondern immer nur als Funktion eines ganzen Prozessablaufs erfasst werden können, d.h. von der Wahrnehmung eines Stimulus bis zur Auslösung einer Reaktion. Ihre Verwendung für die Bestimmung von Teilprozessen beruht daher auf der Annahme, dass es keine Interaktionen mit andern Prozesskomponenten gibt, die sich auf die Reaktionszeiten auswirken.

3.3 Produkt- und Indikatorphänomene

Die Auffassung von Reaktionszeiten als Indikatoren für die Komplexität von Verarbeitungsprozessen führt uns zurück zur Diskussion des Gegenstandsbereichs der Kognitiven Wissenschaft und zur Frage, welche Phänomene für die Kognitive Wissenschaft Daten sind. Dazu muss die Unterscheidung zwischen Produkten eines Prozesses und Indikatoren von Eigenschaften des Prozesses eingeführt werden. Ein kognitiver Verarbeitungsprozess operiert auf einer bestimmten Eingabe-Situation und erzeugt eine bestimmte Ausgabe-Situation oder eine Ausgabe-Verhalten, das zu einer neuen Situation führt. So kann im Schachspiel die Eingabe-Situation in einer bestimmten Spielsituation, das Ausgabeverhalten in einem bestimmten Zug, der zur nächsten Spielposition führt, gesehen werden. Beim Sprachverstehen besteht die Eingabesituation in einem vorgegeben Text, die Ausgabesituation in einer semantischen Repräsentation des Satzes. Während ein kognitiver Prozess abläuft, können Phänomene beobachtet werden, die üblicherweise nicht dem Ausgabe-Verhalten oder der Ausgabe-Situation subsumiert werden. Dazu gehören die vorher erwähnten Reaktionszeiten, dann auch etwa Hautwiderstandsmessungen oder Augenbewegungen, aber auch etwa Lautlöseprotokolle. Phänomene, die durch das Ausgabeverhalten oder die Ausgabesituation beschrieben oder impliziert werden, wollen wir im folgenden als 'Produktphänomene', die andern als 'Indikatorphänomene' bezeichnen. Die Interpretation der beiden Klassen von Phänomenen muss unterschieden werden.

Die Leistung eines Modells unabhängig von seiner psychologischen Validität wird daran gemessen, inwieweit es unter gegebenen Eingabebedingungen die gleiche Ausgabesituation resp. das gleiche Ausgabeverhalten erzeugt wie ein Mensch, so wie es in der Diskussion der schwachen Äquivalenzprüfung formuliert wurde. Dieser Teil der Modellprüfung basiert primär auf Prozessprodukten. Andererseits wurde argumentiert, dass zum Zwecke einer psychologischen Validierung von Modellen schwärfere Kriterien gesetzt werden müssen und hier spielen dann die Indikatorphänomene die primäre Rolle. Allerdings muss hinzugefügt werden, dass sich mit unserer Definition die Trennung in Produktphänomene und Indikatorphänomene nicht streng durchführen lässt. So werden die einer Fehleranalyse zugrundeliegenden Daten durch das Ausgabeverhalten eines

Prozessmodells beschrieben, sie dienen aber dazu, Eigenschaften des ablaufenden Prozesses zu inferieren. Trotzdem glauben wir, dass diese Unterscheidung sinnvoll ist, und wir können damit das Verhältnis von Künstlicher Intelligenz, kognitiver Psychologie und Kognitiver Wissenschaft in ihrem methodischen Vorgehen deutlicher charakterisieren.

Die Künstliche Intelligenz ist primär daran interessiert, Modelle zu bilden, die in bezug auf die Prozessprodukte leistungsfähig sind. Die psychologische Validierung spielt nur eine untergeordnete Rolle, trotzdem kann man aber kaum argumentieren, dass Indikatorphänomene für die Künstliche Intelligenz, auch wenn sie technologisch aufgefasst wird, irrelevant sind. Sie erlauben nämlich, den riesigen Raum von möglichen Theorien, wie leistungsfähig sie auch immer sein mögen, entscheidend einzuschränken. So war denn auch die Imitation schon immer eine wichtige Strategie in der Entwicklung neuer Technologien. So drängt sich auch in der Künstlichen Intelligenz auf, die vom Menschen verwendeten Strategien zu imitieren, ist er doch, wie Minsky einmal gesagt hat, das Produkt eines Forschungsprojektes über Millionen von Jahren. Man könnte sich nun aber einfach auf die eigene Intuition und Introspektion verlassen, um nach einer Theorie zu suchen. Allerdings wird man auch als Nichtpsychologe feststellen, dass die Introspektion eine unzuverlässige Quelle von Evidenz ist und dass sich gewisse psychische Prozesse, wie z.B. die peripheren Prozesse der visuellen Wahrnehmung, der Introspektion völlig entziehen. Aber auch wenn man akzeptiert, dass die Künstliche Intelligenz solche Fakten berücksichtigen soll, kann man immer noch argumentieren, dass sie nicht mit der gleichen Rigorosität an sie gebunden ist wie die Psychologie.

In der kognitiven Psychologie liegen die Verhältnisse etwas anders. Hier wird der psychologischen Validierung natürlich grosses Gewicht beigemessen und damit spielen Indikatorphänomene auch eine viel wichtigere Rolle als in der Künstlichen Intelligenz. Andererseits spielen die Produktphänomene, so scheint es uns, eine weniger wichtige Rolle im Vergleich zu den Indikatorphänomenen und damit auch im Vergleich zur Künstlichen Intelligenz. Dabei spielen zwei Gründe eine Rolle. Erstens fehlt in der kognitiven Psychologie ein Äquivalent zur technologischen Motivation der Künstlichen Intelligenz und damit eine primäre Orientierung auf die Prozessprodukte. Die kognitive Psychologie setzt sich nicht zum Ziel, einen künstlichen Menschen zu entwickeln. Der zweite Grund liegt in einer methodischen Orientierung der kognitiven Psychologie, in der Theorien

vorwiegend durch eine schrittweise Verfeinerung von Theorien entwickelt werden. Auf der Suche nach einem adäquaten psychologischen Modell stehen damit auch nicht spezifische Leistungen und damit die Prozessprodukte im Vordergrund. Vielmehr gilt es zu entscheiden, welche Klassen von möglichen Theorien psychologisch valide sind. Damit hat auch die Analyse von Indikatorphänomenen eine höhere Priorität. Auch hier wird nicht argumentiert, dass die kognitive Psychologie auf die Analyse von Produktphänomenen verzichtet oder gar verzichten kann. Prozessmodelle können nur im Hinblick auf ihre Leistungen und damit im Hinblick auf ihre Produkte entworfen werden. Im Grade der Berücksichtigung dieser Leistungen unterscheiden sie sich aber deutlich, von der unbelegten Annahme einer prinzipiellen Suffizienz bis hin zu ausformulierten Suffizienzbeweisen für gewisse formalisierte Modelle. Tatsache bleibt aber, dass die Frage der Modellsuffizienz in der kognitiven Psychologie eine vergleichsweise untergeordnete Rolle spielt. Im Abschnitt über die schrittweise Verfeinerung von Theorien werden wir noch einmal eingehend auf diese Problematik eingehen.

In der Kognitiven Wissenschaft, die sich die Analyse der Bedingungen intelligenten Verhalten für beliebige symbolmanipulierende Systeme zum Ziel setzt, tritt eine ausgewogenere Berücksichtigung beider Phänomenklassen. Aus dem Ziel der Bedingungsanalyse leitet sich direkt die Forderung nach der Formulierung von Suffizienznachweisen ab und damit die Notwendigkeit der Analyse von Produktphänomenen. Eine systemunabhängige Formulierung dieser Prinzipien erfordert aber auch die Erkennung systemspezifischer Eigenheiten und beinhaltet damit in bezug auf den Menschen auch eine psychologische Fragestellung. In dieser Hinsicht werden dann auch psychologische Indikatorphänomene für die Kognitive Wissenschaft von Bedeutung.

3.4 Erklärung

In den Naturwissenschaften besteht die Erklärung von Phänomenen darin, sie auf mechanistische Gesetze zurückzuführen. In der Kognitiven Wissenschaft wird ein etwas geändertes Konzept der Erklärung verwendet. Im Abschnitt über das Paradigma der Kognitiven Wissenschaften haben wir

gesehen, dass Theorien der Kognitiven Wissenschaft unabhängig vom ausführenden Mechanismus formuliert werden. Damit ist aber auch eine Erklärung durch Rückgriff auf kausale mechanistische Zusammenhänge ausgeschlossen.

In den Sozial- und Humanwissenschaften besteht die Erklärung von Phänomenen vorwiegend darin, sie auf ein System von Regeln zurückzuführen. Dies trifft im Prinzip auch auf die Kognitive Wissenschaft zu, allerdings erhält hier der Begriff der 'Regel' eine exakt definierte Bedeutung, wird er doch auf den Begriff des 'Algorithmus' zurückgeführt. Aus dem Vorgehen der Kognitiven Wissenschaft lassen sich nun zwei verschiedene Auffassungen vom Begriff der Erklärung ableiten, die wir im folgenden als 'spezifische Erklärung' und als 'allgemeine Erklärung' bezeichnen wollen.

Eine spezifische Erklärung eines Phänomens besteht in einem Algorithmus auf einem spezifischen symbolmanipulierenden System, der unter angemessenen Bedingungen dieses Phänomen produzieren kann. Unter diesen Voraussetzungen sind Programme der Künstlichen Intelligenz als spezifische Erklärungen für die produzierten Phänomene zu betrachten. Das Ziel der Kognitiven Wissenschaft ist aber allgemeiner. Sie will ja nicht nur einen Suffizienzbeweis für eine Leistung auf einem bestimmten System erbringen, sondern will die notwendigen und hinreichenden Bedingungen unabhängig von bestimmten Systemen formulieren. Eine systemunabhängige Formulierung von solchen Bedingungen und Prinzipien wollen wir als 'allgemeine Erklärung' bezeichnen. In der heutigen Situation ist es allerdings kaum sinnvoll, nur allgemeine Erklärungen von Phänomenen suchen zu wollen, ist es doch erst in einzelnen Fällen gelungen, suffiziente Bedingungen für spezifische Systeme zu formulieren, geschweige denn, sie systemunabhängig zu formulieren.

Unter diesen Voraussetzungen folgt, dass eine spezifische Erklärung primär nur für das System gilt, für das der Algorithmus konzipiert wurde. Dieses System ist, wie in Abschnitt 2 formuliert wurde, durch die funktionale Architektur bestimmt. Eine Erklärung ist damit nur relativ zu den in der funktionalen Architektur gesetzten Annahmen gültig, d.h. kann auf ein anderes System nur übertragen werden, wenn diesem die gleiche funktionale Architektur zugrundeliegt. In diesem Sinne können auch Erkenntnisse aus der Künstlichen Intelligenz nur dann als psychologische Erklärungen gelten, wenn das künstliche und das natürliche System auf der gleichen funktionalen Architektur basieren. So müssen auch die Eigenschaften etwa des Parsers von Marcus interpretiert werden, auf den wir

bereits im Abschnitt 2 hingewiesen haben. Die Leistung von Marcus besteht unter anderem darin, gezeigt zu haben, dass es für das Phänomen der sogenannten Gartenpfadsätze eine spezifische Erklärung gibt, die sich aus Eigenschaften einer einfachen, funktionalen Architektur ableiten lassen. Der Wert seiner Erklärung für die kognitive Psychologie wird dadurch bestimmt, inwieweit sich seine Annahmen über die funktionale Architektur auch psychologisch rechtfertigen lassen. Die heutigen Erkenntnisse der kognitiven Psychologie, insbesondere diejenigen aus den Untersuchungen über den Kurzzeitspeicher und aus den Arbeiten über die Tiefe der Verarbeitung (Craik & Lockhart, 1972) deuten darauf hin, dass Architektur des Marcusparsers mit psychologischen Modellvorstellungen nicht unvereinbar ist. Somit ist das Modell von Marcus auch eine mögliche psychologische Erklärung, die aber den Vorteil hat, dass sie nach den Prinzipien der Kognitiven Wissenschaft streng ausformuliert ist. Als eine psychologische Erklärung im strengen Sinne kann es aber nur dann gelten, wenn der Nachweis erbracht worden ist, dass die zugrundeliegende funktionale Architektur und die damit gesetzten Annahmen auch psychologisch valide sind.

Die Erklärung von Phänomenen durch Algorithmen bedingt eine bestimmte Stufe der Beschreibung dieser Phänomene, damit diese überhaupt auf einen Algorithmus bezogen werden können. Und die Erklärung der Phänomene bleibt beschränkt auf die dem Algorithmus entsprechende Stufe der Beschreibung. So werden wohl gewisse Muskelbewegungen eines Schachspielers durch seine kognitiven Aktivitäten bestimmt, etwa wenn er eine Figur auf ein neues Feld stellt, sie würden aber nur dann erklärt, wenn die Modellierung eines Schachspielers bis auf die Stufe der motorischen Aktivitäten hinunter ausformuliert wäre. Für die Kognitive Wissenschaft ergeben sich daraus zwei Konsequenzen. In der Analyse von Phänomenen und in der Beurteilung ihrer Erklärbarkeit durch Theorien der Kognitiven Wissenschaft gilt es zu beachten, auf welcher Stufe sie für eine Algorithmisierung beschrieben werden müssen. Dies wird auch dadurch bestimmt, welche Beschreibungsprimitiva in der Formalisierung zur Verfügung stehen, eine Frage, auf die wir später noch genauer eingehen wollen.

Ebenso wichtig wie die Stufe der Beschreibung ist auch die Bedingung, dass Phänomene formalisiert repräsentierbar sein müssen und damit ein wohldefiniertes Beschreibungssystem erfordern. So enzieht sich etwa Verhalten, das nur durch intuitiv erfasste Kategorien von Verhaltenselementen erfasst ist, einer Erklärung durch Theorien der

Kognitiven Wissenschaft.

Algorithmen werden in der Kognitiven Wissenschaft als Erklärungen aufgefasst. Wir haben gesehen, dass hinter einer solchen Erklärung eine ganze Reihe von Annahmen steckt, die durch die funktionale Architektur des Systems gesetzt werden. Eine Uebertragung einer Erklärung auf ein anderes System ist daher nur unter der Bedingung gültig, dass die gesetzten Annahmen mit diesem zweiten System verträglich sind. Im weiteren zeigt sich, dass die Algorithmisierung gewisse Anforderungen an die Stufe und die Formalisierbarkeit der Phänomenbeschreibungen stellt und eine Erklärung von Phänomenen auf den Fall beschränkt bleibt, wo diese Bedingungen erfüllt sind.

4. Entwicklung von Theorien

In diesem Abschnitt werden einige Eigenschaften von Theorien und ihre Implikationen für die Strategien der Theorienentwicklung diskutiert. Die Argumente werden letztlich die Ansicht unterstützen, dass die Verwendung von Methoden der Künstlichen Intelligenz sowohl für die Ziele der Kognitiven Wissenschaft als auch der kognitiven Psychologie unter bestimmten Voraussetzungen notwendig sind und in andern Fällen das Finden einer geeigneten Theorie entscheidend vereinfachen kann. Mit 'Methoden der Künstlichen Intelligenz' ist dabei vor allem die Formulierung von Modellen in der Form von Programmen gemeint. Wie die Diskussion zeigen wird, kann die Uebernahme dieser Methode etwa in der kognitiven Psychologie nicht als eine einfache Erweiterung des Methodeninstrumentariums aufgefasst werden, ergeben sich daraus doch wichtige Konsequenzen dafür, wie weit die damit entwickelten Theorien für den Menschen überschaubar und manipulierbar sind.

4.1 Sparsamkeit von Theorien

Es gilt seit Ockham als eines der wichtigsten Prinzipien, dass eine Theorie möglichst sparsam zu sein hat, im Sinne, dass sie möglichst wenige Entitäten und möglichst wenige Annahmen über diese Entitäten enthalten soll. Dieses Kriterium kann sowohl für relative als auch absolute Bewertungen von Theorien verwendet werden. Mit der relativen Bewertung von Theorien ist gemeint, dass von zwei Theorien gleicher Leistungsfähigkeit jene vorgezogen wird, die sparsamer ist. Es gibt kaum plausible Argumente, die gegen die Verwendung des Sparsamkeitsprinzips in solchen Situationen spricht. Mit der Verwendung des Sparsamkeitsprinzips für absolute Bewertungen einer Theorie ist gemeint, dass eine Theorie nur dann akzeptabel sein soll, wenn sie eine bestimmte Grenze der Komplexität nicht überschreitet.

Die Gründe, die zu dieser Ansicht führen, sind sehr verschiedenartig. Ein Grund dazu liegt sicher im Erfolg der klassischen Physik, der es gelungen ist, die scheinbare Komplexität der Naturphänomene auf einige wenige allgemeine Gesetze zurückzuführen. Die Kenntnis dieser wenigen

Gesetze ermöglicht es uns, auf relativ einfache Weise die Umwelt zu manipulieren und die Konsequenzen dieser Manipulationen vorausszusagen. Dazu gehört auch, dass die Konsequenzen der Veränderungen von Parametern in der Theorie leicht ableitbar sind und, noch wichtiger, dass sich die entscheidenden Parameter in der Theorie überhaupt identifizieren lassen. Im folgenden wird unter anderem argumentiert, dass die Phänomene, mit denen sich die Kognitive Wissenschaft befasst, einen viel grösseren Komplexitätsgrad aufweisen als die Phänomene der klassischen Physik. Eine erhöhte Komplexität der Phänomene allein genügt jedoch noch nicht, um eine erhöhte Komplexität der Theorie der intelligenten Leistungen zu rechtfertigen. Dies kann am Beispiel von Simons Ameise (nach H.A. Simon) illustriert werden. Wenn man beobachtet, wie sich eine Ameise auf dem Boden fortbewegt und sieht, wie sie bei Hindernissen nach Umwegen sucht, diese Umwege ausprobiert und wieder aufgibt, wie sie Hindernisse aus dem Weg räumt usw., dann ist man leicht dazu geneigt, der Ameise die Fähigkeit zu komplexen Planungs-, Prüf- und Korrekturprozessen zuzuschreiben. Dies ist aber eine Täuschung: Die Ameise reagiert mit einem relativ einfachen Mechanismus auf lokale Eigenschaften einer komplexen Umwelt. Die komplexen Handlungen der Ameise sind also nicht auf einen komplexen Verarbeitungsmechanismus zurückzuführen, sondern auf eine komplexe Umwelt, in der sich die Ameise bewegt. In diesem Sinne muss man daher auch zeigen, dass in intelligenten Leistungen mehr steckt als in den Leistungen von Simons Ameisen, will man eine komplexe Theorie des Menschen rechtfertigen.

Im nächsten Abschnitt werden wir Argumente anführen, die diese Ansicht unterstützen. Es wird aber nicht argumentiert werden, dass Theorien des intelligenten Verhaltens unbedingt komplex sein müssen. Es wird aber gezeigt werden, dass sie erstens komplex sein können und zweitens, dass es auf dem Wege zu einer einfachen Theorie sinnvoll sein kann, auch komplexe Theorien zu entwerfen.

4.2 Grenzen einer schrittweisen Verfeinerung von Theorien

Eine mögliche Strategie der Theorienbildung, die wir schrittweise Verfeinerung nennen wollen (siehe auch Groner, 1978), besteht darin, dass eine Theorie über eine Kette von alternativen Hypothesen und ihrer

experimentellen Uebeprüfung entwickelt wird. Ausgegangen wird von Hypothesen über möglichst allgemeine Eigenschaften eines Prozesses, die dann sukzessive mit Hypothesen über immer spezifischere Eigenschaften des Prozesses erweitert werden. Diese Strategie ermöglicht es, Annahmen, die zu falschen Voraussagen führen, relativ leicht zu identifizieren. Würde man andererseits von einer völlig ausspezifizierten Theorie ausgehen, wäre es bedeutend schwieriger, die Ursache einer falschen Voraussage in der grossen Menge von Annahmen zu lokalisieren. In der Strategie der schrittweisen Verfeinerung ist die Idee enthalten, dass Prozesse schematisch beschrieben werden können und dass auf die Spezifikation jedes Details von Prozessen verzichtet werden kann. Zu einem Schema lassen sich dann verschiedene Algorithmen spezifizieren, die dieses Schema implementieren; die Spezifikation eines Algorithmus bringt aber keine grundsätzlich neuen Erkenntnisse mit sich. Das heisst mit andern Worten, dass die Menge der möglichen Algorithmen durch das Schema bereits festgelegt ist und dass von den Details der Algorithmen sinnvoll abstrahiert werden kann.

Die Strategie der schrittweisen Verfeinerung schliesst die Entwicklung einer Theorie eines Teilprozesses nicht aus. Sie bedingt aber, dass Teilprozesse voneinander isoliert werden können. Diese Annahme besagt im Prinzip das gleiche wie die Annahme, dass ein Prozess in einem Schema sinnvoll beschrieben werden kann. Die Teilprozesse eines so beschriebenen Prozesses dürfen nur eine lokale Wirkung haben, d.h. der Teilprozess A darf nur beschränkt mit dem Teilprozess B interagieren. Ist dies nicht der Fall, dann können die beiden Teilprozesse nicht sinnvollerweise voneinander isoliert werden und mit der Spezifikation der Teilprozesse können grundlegend neue Erkenntnisse gewonnen werden.

Es stellt sich daher die Frage, ob diese Annahme immer gemacht werden kann. Erkenntnisse in der Künstlichen Intelligenz, aber zum Teil auch in den Naturwissenschaften zeigen, dass dies nicht der Fall ist. Ein Beispiel ist der Prozess der Faltung von Eiweissketten (Levitt und Warshel, 1975; Marr, 1976b). Der komplizierte Faltungsprozess wird durch eine grosse Anzahl von Einflüssen, wie der Geometrie der Aminosäuren, ihren hydrophobischen Interaktionen oder den thermalen Perturbationen bestimmt und führt zu einer, für jede Eiweisskette typischen, stabilen Konfiguration. In jedem Moment des Faltungsprozesses sind nur wenige der Einflüsse wichtig, diese Einflüsse sind aber für den weiteren Faltungsprozess entscheidend. In einer schematischen Theorie müsste von

diesen lokalen Einflüssen abstrahiert werden. Da aber alle Einflüsse zu einem Zeitpunkt im Faltungsprozess entscheidend sein können, muss eine schematische Theorie notwendigerweise inadäquat bleiben. Das zur Zeit beste Modell der Faltung von Eisweissketten von Levitt und Marshel ist daher auch ein Programm, in dem alle diese Einflüsse beschrieben sind und das den Faltungsprozess so lange simuliert, bis eine stabile Konfiguration erreicht ist. Marr(1976b) bezeichnet eine solche Theorie als Typ 2 Theorie und kontrastiert sie mit den schematischen Theorien, den Typ 1 Theorien.

Auch in der Künstlichen Intelligenz gibt es eine ganze Reihe von Typ 2 Theorien, von denen wir eine, nämlich Lenats(1976, 1977) Programm AM erwähnen möchten. Dieses Programm entdeckt mit Hilfe hunderter heuristischer Regeln interessante mathematische Konzepte wie die Multiplikation, die Division, die Primzahlen oder die Goldbachsche Vermutung, ausgehend von nur wenigen, allgemeinen Konzepten wie Prädikat, Operation, Menge, Vereinigung und Durchschnitt. Die Frage, ob und wann im Verlaufe des Entdeckungsprozesses ein bestimmtes Konzept entdeckt wird, hängt von der Präsenz bestimmter ursprünglicher Konzepte und heuristischer Regeln ab, aber auch von bestimmten Konfigurationen des Entdeckungsprozesses selbst, wie etwa dem zeitlich nahe aufeinanderfolgenden Entdecken verwandter Konzepte. Diese Zusammenhänge sind aber so komplex, dass es bisher noch nicht gelungen ist, die Bedingungen, die zur Entdeckung eines interessanten Konzepts führen, in einer schematischen Typ 1 Theorie geeignet zu beschreiben. In seiner neuesten Arbeit hat Lenat(1980) erst Ansätze zu einer möglichen Formulierung einer solchen Theorie aufgezeigt. Um daher beurteilen zu können, welche Leistungen AM erbringen kann, bleibt daher vorläufig nur ein experimentelles Vorgehen, in dem die verschiedenen Parameter des Programms variiert und das Verhalten des Programms beobachtet wird.

Auf ein anderes Beispiel einer Typ 2 Theorie sind wir bereits im Zusammenhang mit Systemen zum Verstehen gesprochener Sprache eingegangen, nämlich auf das System Hearsay-II (III.5). Wie Lenats Programm arbeitet auch dieses System auf eine charakteristische Weise, die mit dem Arbeiten einer Gesellschaft von Experten verglichen werden kann (Minsky,1977). In dieser Gesellschaft (z.B. einem Produktionssystem) verfügen die einzelnen Experten (Produktionsregeln) nur über ein beschränktes Wissen und kommunizieren miteinander, indem sie Meldungen auf einer Wandtafel (Datenbasis) lesen und schreiben. Entdeckt ein Experte eine Meldung, die in

seinen Zustandsbereich fällt (ein Item in der Datenbasis, das seinem Muster entspricht), dann kann er die Kontrolle übernehmen und seinerseits neue Meldungen auf die Wandtafel schreiben. Welcher Experte jeweils als nächster aktiviert wird, hängt also von den Eigenschaften der einzelnen Experten und vom aktuellen Zustand im Problemlösungsprozess ab. Es zeigt sich, dass in einer solchen Gesellschaft von Experten das Verhalten des gesamten Systems durch komplexe Interaktionen zustandekommen kann, die sich nicht sicher in einer schematischen Typ 1 Theorie erfassen lassen.

Es stellt sich daher die Frage, ob im Bereich intelligenten Verhaltens Phänomene existieren, die sich nur mit einer Typ 2 Theorie erfassen lassen. Sie lässt sich nicht eindeutig beantworten, obwohl es unseres Erachtens im Bereich kognitiver Prozesse ziemlich wahrscheinlich ist. Aber auch wenn die Frage nicht sicher entschieden werden kann, lassen sich daraus zwei wichtige Konsequenzen ziehen. Falls in einem bestimmten Bereich intelligenten Verhaltens nur eine Typ 2 Theorie möglich sein sollte, dann müssen notwendigerweise alle Versuche, eine Typ 1 Theorie zu entwickeln, scheitern. Solange aber der Nachweis für die prinzipielle Suffizienz einer Typ 1 Theorie fehlt, ist es ratsam, nicht nur nach Typ 1 Theorien zu suchen, wie dies etwa in der kognitiven Psychologie der Fall ist.

Andererseits kann man gegen die Entwicklung von Typ 2 Theorien damit argumentieren, dass eine Typ 2 Theorie nicht dem Ziel jeder Wissenschaft entspricht, einfache und damit einfach anwendbare Gesetze über die Phänomene der Natur und des Menschen zu finden. Diesem Argument kann man nichts entgegenhalten, wenn man eine nur teilweise adäquate, aber einfache Theorie einer adäquaten, aber komplexen und nicht mehr überschaubaren Theorie vorzieht. Falls man aber als oberstes Ziel der Wissenschaft die Entwicklung von empirisch adäquaten Theorien setzt, kann man gezwungen sein, eine Typ 2 Theorie zu akzeptieren. Ein solcher Sachverhalt ist nicht aussergewöhnlich, gibt es in diesem Jahrhundert doch bereits eine ganze Reihe solcher, zum Teil schwerwiegender Ernüchterungen. Man erinnere sich nur an die Entdeckung, dass viele Fragen grundsätzlich nicht entscheidbar sind oder an die Entdeckung, dass sich gewisse Probleme ziemlich sicher nicht effizient lösen lassen und für die auch bei kleiner Problemkomplexität die schnellsten Computer Jahrtausende zur Lösung brauchen würden (Siehe Abschnitt 2 in diesem Kapitel). Es ist daher nicht von der Hand zu weisen, dass es vielleicht auch Phänomene gibt, die sich nur mit einer komplexen Theorie beschreiben lassen.

Als weiteres Argument gegen die Entwicklung von Typ 2 Theorien kann man anführen, dass sie dazu führen kann, die einem Problem inhärenten Gesetzmässigkeiten zu übersehen. Dies wird dann problematisch, wenn für dieses Phänomen eine Typ 1 Theorie existiert. Wie weit sich diese Befürchtung historisch belegen lässt, ist nicht klar. Am ehesten könnte sie in einigen Arbeiten der Künstlichen Intelligenz über die peripheren Prozesse der visuellen Wahrnehmung gezeigt werden. Darin wurde versucht, auch so tiefe Prozesse wie die Segmentierung mit Hilfe von Alltagswissen zu steuern, während Analysen des Problems in neuerer Zeit gezeigt haben, dass diese Leistung durch sehr einfache Prozesse ohne Rückgriff auf Alltagswissen erreicht werden kann.

Allerdings kann man aber auch umgekehrt argumentieren, nämlich dass das Experimentieren mit einer Typ 2 Theorie und insbesondere die Analyse von Fehlern und Leistungsschwächen zu neuen Erkenntnissen führen kann. So ist Lenats Programm ein wichtiger Schritt auf dem Weg zu einer Typ 1 Theorie des heuristischen und kreativen Denkens, falls sie überhaupt existiert, und dies aus zwei Gründen. Erstens konnte damit herausgefunden werden, welche Leistungen heuristische Regeln und der einfache Mechanismus, der diese Regeln anwendet, erbringen. So hat AM Konzepte entdeckt, die selbst dem Autor des Programms unbekannt waren, wie das Konzept der maximal teilbaren Zahlen. Andererseits aber hat es auch wichtige Konzepte wie die rationalen Zahlen nicht entdeckt und hat mit zunehmender Komplexität der Konzepte immer weniger interessante entdeckt. Ein komplexes Konzept in diesem Sinne war bereits die Division, die von AM mit Hilfe von Mengen und Mengenoperationen repräsentiert werden musste. Daraus kann man schliessen, dass zum heuristischen Denken auch gehört, dass zu neu entwickelten Konzepten neue, adäquate heuristische Regeln entwickelt werden müssen, die eine effiziente Verwendung dieser Konzepte ermöglichen. Dies ist denn auch von Lenat, Hayes-Roth und Klahr(1979) versucht worden. Die Erfahrungen mit diesem Programm haben es wiederum ermöglicht, erste Prinzipien für eine Typ 1 Theorie der Heuristik zu formulieren (Lenat,1980). Es ist kaum vorstellbar, dass diese Entwicklung ohne das Experimentieren mit dem Typ 2 Programm AM möglich gewesen wäre.

Der historische Nachweis, dass die Formulierung einer Typ 1 Theorie erst durch eine vorangegangene Formulierung einer entsprechenden Typ 2 Theorie ermöglicht wurde, ist auch deswegen schwierig durchzuführen, weil reine Typ 1 oder Typ 2 Theorien nur selten vorkommen. Viel eher enthalten

Theorien gewisse Typ 1 und gewisse Typ 2 Elemente. So enthalten die vielen Varianten von ATN-Parsern verschiedene, mehr oder weniger elegante Mechanismen, um die Häufigkeit von Backups zu vermindern. Diese können als Typ 2 Elemente bezeichnet werden, während die ATN-Grammatik in ihrer ursprünglichen Form einer Typ 1 Theorie entspricht. Gerade die Analyse der verschiedenen Mechanismen hat zur Entwicklung der diagnostischen Regeln im Marcusparser geführt. Marcus seinerseits hat über lange Zeit mit seinem Parser experimentiert und die Mechanismen immer weiter einschränken und vereinfachen können, ohne dass die Leistungen des Parsers abnahmen. Erst damit konnte die Ansicht erhärtet werden, dass ein deterministisches Parsen von Sätzen überhaupt möglich ist. Man kann sich aber fragen, ob das lange Experimentieren, sowohl mit ATN-Grammatiken als auch mit dem Marcusparser notwendig waren, um zu dieser Erkenntnis zu gelangen. Im Prinzip muss diese Frage verneint werden, hat doch schon Bever(1970) mit seinen perzeptuellen Strategien ganz ähnliche Ideen formuliert. Unseres Erachtens ist aber Marcus' Theorie der Beverschen in mehrfacher Hinsicht überlegen. Sie ist nicht nur präziser und einfacher formalisiert, ihre Leistungsfähigkeit und Grenzen lassen sich mit Hilfe des Programms auch genau festlegen.

Wir glauben gezeigt zu haben, dass es im Bereiche intelligenten Verhaltens Phänomene gibt, die entweder prinzipiell nur durch Typ 2 Theorien erfasst werden können oder für die es zumindest sinnvoll ist, im Verlaufe ihrer Analyse auch Typ 2 Theorien in Betracht zu ziehen. Welche Konsequenzen sich daraus auf die Theorienentwicklung ergeben, wollen wir im nächsten Abschnitt diskutieren.

4.3 Formalisierung von Theorien

Programme können als Theorien und Programmiersprachen als Formalismen zur Beschreibung von Theorien aufgefasst werden. Wir wollen in diesem Abschnitt auf diese Auffassung näher eingehen und drei Aspekte näher diskutieren, nämlich erstens auf das Problem der Ableitung von Voraussagen aus einer solchen Theorie, zweitens auf die Frage der subjektiven Einfachheit der Formalismen, und schliesslich werden wir uns grundsätzlich mit der Frage auseinandersetzen, inwieweit Programme als Theorien aufgefasst werden können.

4.3.1 Ableitung von Voraussagen

Ein Vorteil der Formalisierung einer Theorie wird darin gesehen, dass Voraussagen auf eindeutige Weise abgeleitet werden können. Dies ist dort nicht gegeben, wo eine Theorie nur natürlichsprachlich formuliert ist. Während in einer formalen Theorie die Regeln zur Ableitung von Sätzen explizit gegeben sind, sind sie in der natürlichsprachlichen Formulierung nur implizit gegeben und hängen auch von der Interpretation (im Sinne von Verstehen) der Theoriesätze ab. Es ist daher in einer solchen Theorie nicht immer eindeutig, ob eine bestimmte Voraussage aus der Theorie folgt. Aber auch in formalisierten Theorien, insbesondere der Kognitiven Wissenschaft, ist die Ableitung von Voraussagen nicht immer trivial, wie im folgenden gezeigt werden soll.

Im Prädikatenkalkül ist der Beweis erbracht, dass alle Voraussagen (Theoreme), die in einem gegebenen Axiomensystem wahr sind, algorithmisch abgeleitet werden können (Vollständigkeit des Prädikatenkalküls erster Ordnung). Aber bei Theorien von grösserer Komplexität, etwa der elementaren Zahlentheorie und damit allen Theorien, die sie enthalten, ist diese Garantie nicht mehr gegeben (Suppes, 1957). Das bedeutet, dass praktisch bei allen mathematischen Theorien nicht mehr alle Theoreme algorithmisch abgeleitet werden können und dass man darum zum Finden von Voraussagen auf heuristische Regeln angewiesen ist. Aber auch beim Beweisen von Voraussagen müssen Einschränkungen gemacht werden. Es ist nicht in jedem Fall garantiert, dass ein Beweis gefunden werden kann. Dies ist etwa der Fall bei Systemen von Differentialgleichungen, die sich nicht mehr in jedem Fall analytisch lösen lassen. Hier ist man dann auf numerische Approximationen angewiesen und in der Praxis werden komplexe Systeme von Differentialgleichungen häufig nur noch simuliert, um Voraussagen abzuleiten.

In der Kognitiven Wissenschaft ist die Situation noch schwieriger. Typischerweise sind die Modelle so komplex, dass Voraussagen nur durch Simulation auf einer Maschine abgeleitet werden können. Ob daher die einem Programm zugrundeliegende Theorie alle Anforderungen erfüllt, kann nur empirisch, d.h. durch Beobachtung des Programmverhaltens unter verschiedenen Aufgabeninstanzen festgestellt werden. In diesem Sinne kann denn auch die Künstliche Intelligenz als eine empirische Wissenschaft

aufgefasst werden: Ein Programm wird als eine Instantiierung einer Theorie aufgefasst, und es ist eine empirische Frage, ob das Programm die an die Theorie gestellten Leistungsforderungen erfüllt. Dieses Vorgehen als 'empirisch' zu bezeichnen mag auf den ersten Blick etwas ungewöhnlich erscheinen, ist es doch zumindest denkbar, dass bewiesen werden kann, erstens ob ein Programm eine korrekte Instantiierung einer Theorie ist, und zweitens ob eine Theorie die an sie gestellten Forderungen erfüllt. Die erste Bedingung ist identisch mit der Forderung nach einem Korrektheitsbeweis für Programme. Die Arbeiten der letzten Jahre auf diesem Gebiet zeigen, dass Korrektheitsbeweise bereits für sehr kleine Programme äusserst aufwendig sein können und daher in der Praxis kaum durchführbar sind. Zur zweiten Bedingung ist zu sagen, dass die gleichen Einschränkungen wie für andere formale Theorien gelten, falls es sich um eine Typ 1 Theorie handelt. Falls es sich aber um eine Typ 2 Theorie handelt, sind Ableitungen von Voraussagen ohne Zuhilfenahme eines Computers praktisch unmöglich.

Der Zweck des Experimentierens mit Programmen bleibt aber nicht auf die Ableitung von Voraussagen beschränkt. Erstens kann man damit beurteilen, wie robust ein Programm gegenüber Variationen der Eingabe ist und kann damit abschätzen, wie leicht sich eine Theorie verallgemeinern lässt. Zweitens lässt sich damit feststellen, wie stark die einzelnen Parameter das Verhalten des Programms beeinflussen, wie dies etwa Lenat(1976) in seinen Experimenten mit dem Programm AM gemacht hat.

4.3.2 Einfachheit von Formalismen

Eine Theorie kann durch verschiedene Kalküle, wie Prädikatenkalkül, Produktionssysteme, Automaten, mathematische Strukturen usw. formalisiert werden. Diese unterscheiden sich meist nicht in ihrer Mächtigkeit (mit Ausnahme etwa der endlichen Automaten) hingegen stark in der subjektiven Einfachheit, mit der bestimmte Sachverhalte repräsentiert werden können. So liegt denn die heutige Unpopularität der S-R-Theorien nicht in der mangelnden Mächtigkeit des S-R-Formalismus begründet. Da der S-R-Formalismus einem endlichen Automaten entspricht (Suppes, 1969), der Mensch aber auch nur über einen endlich grossen Speicher verfügt, muss der S-R-Formalismus prinzipiell genügen. Vielmehr liegt einer der Gründe darin,

dass sich Modelle, wie sie heute in der kognitiven Psychologie üblich sind, im S-R-Formalismus nicht einfach darstellen lassen.

Es wurde bereits angedeutet, dass mit der Einfachheit von Formalismen eine subjektive Kategorie angesprochen ist, deren Beurteilung sowohl vom Anwender als auch von der Anwendung bestimmt ist. Kein Formalismus ist für alle Anwendungen gleich gut geeignet. Im folgenden wird argumentiert, dass in der Kognitiven Wissenschaft laufend neue Formalismen entwickelt werden, die sich für die Repräsentation kognitiver Prozesse besonders gut eignen.

Die Flexibilität der Programmiersprache LISP erleichtert die Entwicklung neuer Formalismen und ihre Einbettung in LISP. Damit sind gute Bedingungen für die Entwicklung anwendungsspezifischer Formalismen gegeben. Wichtig ist aber, dass diese Formalismen nicht in der Luft stehen, sondern durch ihre Einbettung in LISP - mit den in II.2 gemachten Einschränkungen - wohldefiniert sein können.

Ein bestimmender Faktor für die Einfachheit eines Formalismus ist die Flexibilität der zur Verfügung stehenden Datenstrukturen und der zugehörigen Operationen. Dies wird klar am Beispiel der klassischen Simulationstechniken, in denen praktisch nur numerische Repräsentationen und numerische Operationen zur Verfügung stehen. Im Gegensatz dazu fördert die Kognitive Wissenschaft die Sicht von Maschinen als allgemeinen symbolmanipulierenden Systemen und damit auch die Entwicklung von Formalismen, die für beliebige Symbolmanipulationen geeignet sind. Wie anpassungsfähig diese Formalismen sind, haben wir eingehend im Kapitel über die Repräsentation von Wissen diskutiert.

Die Künstliche Intelligenz und die Computerwissenschaft im allgemeinen sind im Laufe der Zeit immer wieder mit dem Problem der wachsenden Komplexität von Programmen konfrontiert worden. Dies führte dazu, dass mit den zur Verfügung stehenden Formalismen immer wieder Grenzen erreicht wurden. Berühmt wurde so die Softwarekrise Ende der Sechzigerjahre, in der mehrere grosse Programmierprojekte an der Komplexität der Programme scheiterten und zur Entwicklung des Konzepts der strukturierten Programmierung führte. Auch heute stösst man in der Künstlichen Intelligenz immer wieder an Grenzen der Komplexität. Schon aus technologischer Sicht ist man dann gezwungen, Formalismen zu entwickeln, mit denen komplexe Prozesse subjektiv einfach beschrieben werden können (Winograd, 1975). Die so entwickelten Formalismen erlauben es immer wieder, neue Komplexitätsgrenzen zu überwinden und erlauben es der Kognitiven

Wissenschaft, immer komplexere Modelle intelligenter Leistungen zu implementieren.

Aber das Komplexitätsargument allein genügt noch nicht, um die Formalismen der Künstlichen Intelligenz auch für die andern kognitiv orientierten Disziplinen als wertvoll zu beurteilen. Ebenso wichtig ist, dass diese Formalismen im Hinblick auf die Repräsentation kognitiver Sachverhalte und Prozesse entwickelt wurden. Es ist eines der Verdienste der Künstlichen Intelligenz, dass sie immer wieder versucht, auch solche Sachverhalte zu repräsentieren, deren Formalisierung in andern Disziplinen gescheitert oder nur unbefriedigend gelöst werden konnte, wie Glaubenssysteme, Planungsprozesse oder die Repräsentation von Alltagswissen. Dass diese Formalisierungen unseren (subjektiven) Adäquatheitskriterien oftmals nicht genügen, kann nicht zum Argument gegen die Formalisierbarkeit dieser Sachverhalte gemacht werden. Denn aus der Inadäquatheit eines Formalismus kann noch nicht auf die prinzipielle Unmöglichkeit der Formalisierung geschlossen werden.

Unsere Argumentation geht also nicht dahin, spezifische Formalismen der Kognitiven Wissenschaft für die Repräsentation kognitiver Prozesse als besonders geeignet hinzustellen. Welche es sind, kann nur über den Erfolg ihrer Anwendung beurteilt werden. Wir glauben aber, dass die in der Künstlichen Intelligenz und der Computerwissenschaft entwickelten Formalismen sich deswegen als mögliche Formalismen für alle kognitiv orientierten Wissenschaften aufdrängen, weil sich diese Disziplinen mit dem gleichen Problem der Komplexität konfrontiert sehen.

4.3.3 Theorie und Programm

Ueber das Verhältnis von Theorien und Programmen sind immer wieder neue Auffassungen formuliert worden, auf die wir im folgenden eingehen wollen. Nach einer Auffassung werden Programm mit Theorien gleichgesetzt. Diese Auffassung kann damit gerechtfertigt werden, dass Programme in einem wohldefinierten Kalkül definiert sind und auch die Semantik des Kalküls über die Grundoperationen des Computers wohldefiniert ist. Gegen diese Auffassung spricht, dass gewisse Programmteile nur schlecht als Elemente der Theorie gerechtfertigt werden können, weil sie nur in einer

spezifischen Implementation mit ihren zufälligen Eigenheiten eine Funktion erfüllen. Dies ist der Fall mit Programmteilen, die eine bestimmte Architektur emulieren, aber auch mit graphischen Aufbereiterprogrammen für die Ausgabe, Terminalsteuerprogrammen, Trace-Routinen usw.

Die Trennung von relevanten und irrelevanten Programmteilen bringt aber etwelche Schwierigkeiten mit sich, weil nicht klar ist, wo die Grenze zu ziehen ist. Wird diese Grenze informal festgelegt, dann unterliegt die Festlegung darüber, welche Programmteile die Theorie beschreiben, einem subjektiven Ermessenskriterium und es ist nicht auszuschliessen, dass das physisch gleiche Programm für verschiedene Personen eine verschiedene Theorie sein kann.

Eine konventionelle Abgrenzung der relevanten, d.h. zur Theorie gehörigen Programmteile, bedingt aber die Möglichkeit einer abstrakten Programmbeschreibung durch einen andern Kalkül. In sehr vielen Fällen ist dies auch möglich, wie wir in der Diskussion der Typ 1 Theorien gesehen haben. Danach wird die abstrakte Beschreibung als 'die Theorie' aufgefasst, die durch verschiedene Algorithmen implementiert werden kann. Diese wiederum haben den Status von Modellen im engeren Sinne der formalen Logik. Nun haben wir auch gesehen, dass bei gewissen Problemen eine solche abstrakte Beschreibung praktisch nicht möglich ist, und die Beschreibung des Programms selbst die einfachstmögliche Theorie - eine Typ 2 Theorie - ist. Damit ist der Zirkel geschlossen: Eine formale Abgrenzung der Theorie scheint ausgeschlossen. Aus dieser Situation zeigen sich drei mögliche Auswege. Erstens kann man ablehnen, dass Programme als Theorien aufgefasst werden. Diese Position entspricht der bereits diskutierten Ablehnung von Typ 2 Theorien mit all ihren Konsequenzen, zu denen gehört, dass gewisse Phänomene nicht durch eine Theorie erfassbar sind. Zweitens kann man subjektive Abgrenzungskriterien für Theorien akzeptieren. Diese Position entspricht zwar nicht dem Ideal der Wissenschaftstheorie mit ihren scharfen Kriterien an die Theoriebeschreibung, ist aber, wenn man von den tatsächlichen Gegebenheiten in den Wissenschaften ausgeht, nicht unüblich. Drittens kann man darauf hoffen, dass die noch junge Disziplin der Programmabstraktion (durch Computer) uns ein methodisches Instrument zur Verfügung stellen wird, das eine präzise, maschinenunabhängige Programmbeschreibung erlaubt (siehe z.B. Rich & Shrobe, 1976). Ob und wie weit dies möglich sein wird, kann aber erst die Zukunft zeigen.

Aus allen angeführten Argumenten geht hervor, dass mit dem

methodischen Instrumentarium der Kognitiven Wissenschaft keineswegs ein wissenschaftstheoretisches Ideal erreicht werden kann. Es sollte aber auch klar geworden sein, dass die Methoden der Kognitiven Wissenschaft in Situationen angewandt werden können, in denen die traditionelle Vorgehensweise der Kognitionspsychologie versagt. Die Uebernahme von Methoden der Kognitiven Wissenschaft drängt sich dann aus praktischen Gründen auf. Diese Arbeit sollte gezeigt haben, dass sich aus einer solchen Zusammenarbeit von Psychologie und Kognitiver Wissenschaft sehr fruchtbare Ansätze ergeben, die zu berechtigten Hoffnungen Anlass geben, dass wir in der Zukunft die komplexen Leistungen des Menschen viel besser erfassen können, als dies heute noch der Fall ist.

VII. Z U S A M M E N F A S S U N G

In detaillierten Analysen werden Arbeiten der Künstlichen Intelligenz auf dem Gebiet des Sprachverstehens und der Wissensrepräsentation auf ihre Verwendbarkeit als psychologische Theorien diskutiert. Dabei wird vor allem auf ihre Voraussetzungen, ihre Leistungsfähigkeit und ihre Ueberprüfbarkeit eingegangen und mit Arbeiten der kognitiven Psychologie verglichen. Es wird argumentiert, dass eine Synthese der beiden Disziplinen in der Kognitiven Wissenschaft zu leistungsfähigeren Theorien kognitiver Leistungen führen kann, dass diese Theorien unter Umständen aber äusserst komplex werden können.

- Abelson, R.P. (1968) Simulation of Social Behavior, in: Lindzey, G. und Aronson, E. (Hrsg.) The Handbook of Social Psychology. Vol. 2, Addison-Wesley, Reading, Mass.
- Aho, A.V., J.E. Hopcroft und J.D. Ullman (1974) The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, Mass.
- Anderson, J.R. (1976) Language, Memory and Thought, Lawrence Erlbaum, Hillsdale, New Jersey
- Bartlett, F.C. (1932) Remembering. Cambridge University Press
- Becker, J.D. (1975) The Phrasal Lexikon, in: Proceedings of the Workshop on Theoretical Issues in Natural Language Processing 1, Cambridge, Mass., 70-73
- Bever, T. (1970) The Cognitive Basis for Linguistic Structures, in: Hayes, J. (Hrsg.) Cognition and the Development of Language. Wiley, New York, 279-353
- Bever, T., M.F. Garrett und R. Hurtig (1973) The Interaction of Perceptual processes and Ambiguous Sentences. Memory and Cognition, 1 (3), 277-286
- Bledsoe, W.W. (1977) Non Resolution Theorem Proving. Artificial Intelligence, 9 (1) 1-36
- Bobrow, D.G. (1968) Natural Language Input for a Computer Problem-Solving System, in: M. Minsky (Hrsg.) Semantic Information Processing, MIT Press, Cambridge, Mass.
- Bobrow, D.G. (1976) Dimensions of Representation, in: Bobrow, D.G. und A. Collins (Hrsg.) Representation and Understanding. Studies in Cognitive Science, Academic Press, New York
- Bobrow, D.G. (1980) Special Issue on Non-Monotonic Logic. Artificial Intelligence, Vol. 13, No. 1 & 2, 1-172
- Böhling, K.H. und Indermark, K. (1969) Endliche Automaten I. BI Hochschulschriften H 703, Hochschultaschenbücherverlag, Mannheim
- Bousfield, W.A. (1953) The Occurrence of Clustering in the Recall of Randomly arranged Associates, J. Genetic Psychol., 49, 229-240
- Bower, G., J.D. Black und T.J. Turner (1979) Scripts in Memory for Text. Cognitive Psychology, 11 (2).
- Brachman, R.J. (1978) A Structural Paradigm for Representing Knowledge. Bolt, Beranek and Newman Inc. Report No. 3605, Cambridge, Mass.

- Brauer, W. und K. Indermark (1968) Algorithmen, Rekursive Funktionen und Formale Sprachen. BI Hochschulschriften 817, Bibliographisches Institut, Mannheim
- Breitmeyer, B. und L. Ganz (1976) Implications of Sustained and Transient Channels for Theories of Pattern Masking, Saccadic Suppression and Information Processing. Psychological Review, 83 (1), 1-36
- Bruce, B. (1975) Case Systems for Natural Language. Artificial Intelligence, Vol. 6, 327-360
- Buchanan, B.G. and T.M. Mitchell (1978) Model-Directed Learning of Production Rules, in: Waterman, D.A. und F. Hayes-Roth (Hrsg.) Pattern-Directed Inference Systems. Academic Press, New York, 297-312
- Buchanan, B.G., G. Sutherland und E.A. Feigenbaum (1969) Heuristic Dendral: A Program for Generating Explanatory Hypotheses in Organic Chemistry, American Elsevier, New York, 209-254
- Burton, R.R. (1976) Semantic Grammar: An Engineering Technique for Constructing Natural Language Understanding Systems. Bolt, Beranek and Newman Inc., Cambridge, Mass., Report No. 3453
- Carbonell, J.R. (1970) Mixed-Initiative Man-Computer Instructional Dialogues. Bolt, Beranek and Newman Inc., Cambridge, Mass. Report No. 1971
- Chang, C. und R.C. Lee (1973) Symbolic Logical and Mechanical Theorem Proving. Academic Press, New York.
- Charniak, E. (1972) Toward a Model of Children's Story Comprehension. MIT Artificial Intelligence Laboratory, Cambridge, Mass. AI-TR-266
- Charniak, E. (1974) He will make you take it back: A Study in the Pragmatics of Language. Institute for Semantic and Cognitive Studies, Genf, Working Paper 5
- Charniak, E. (1975a) A Partial Taxonomy of Knowledge about Actions. Proceedings of the 4th International Joint Conference on Artificial Intelligence, Tbilissi, UdSSR, Vol. 1, 91-98
- Charniak, E. (1975b) Organization and Inference in a Framelike System of Common Sense Knowledge. Proceeding of the Conference on Theoretical Issues in Natural Language Processing 1, Cambridge, Mass., 46-55
- Charniak, E. (1975c) A Brief on Case. Institute for Semantic and Cognitive Studies, Genf, Working Paper 22
- Charniak, E. (1976a) Ms. Malaprop, A Language Comprehension Program. Institute for Semantic and Cognitive Studies, Genf, WP 31

- Charniak, E. (1976b) Syntax in Linguistics, in: Charniak, E. und Y. Wilks (Hrsg.) Computational Semantics. North-Holland, Amsterdam, 23-40
- Charniak, E. (1977) A Framed Painting: The Representation of a Common Sense Knowledge Fragment. Cognitive Science, 1 (4), 355-394
- Charniak, E. (1978a) With a Spoon in Hand it must be the Eating Frame. Proceedings of the Workshop on Theoretical Issues in Natural Language Processing 2, Urbana-Champaign, Illinois, 187-193
- Charniak, E. (1978b) On the Use of Framed Knowledge in Language Comprehension. Artificial Intelligence, 11 (3), 225-266
- Charniak, E. und Y. Wilks (1976, Hrsg.) Computational Semantics. North Holland, Amsterdam
- Chomsky, N. (1963) Formal Properties of Grammars, in: Luce, R.D., R.R. Bush und E. Galanter (Hrsg.) Handbook of Mathematical Psychology, Vol. 2, Wiley, New York, 323-418
- Church, A. (1941) The Calculi of Lambda-Conversion. Annals of Mathematics Studies 6, Princeton
- Colby, K.M (1973) Simulations of Belief Systems, in: Schank, R. und K.M. Colby (Hrsg.) Computer Models of Thought and Language, Freeman, San Francisco, California
- Colby, K.M. (1975) Artificial Paranoia: A Computer Simulation of Paranoid Processes. Pergamon Press, New York
- Colby, K.M., R.C. Parkinson und B. Faught (1974) Pattern-Matching Rules for the Recognition of Natural Language Dialogue Expressions. American Journal of Computational Linguistics, Mikrofilm 5
- Colby, K.M., S. Weber und F.D. Hilf (1971) Artificial Paranoia. Artificial Intelligence, Vol. 3, 199-221
- Collins, A. und E.F. Loftus (1975) A Spreading-Activation Theory of Semantic Processing. Psychological Review, 82 (6), 407-428
- Collins, A. und M.R. Quillian (1969) Retrieval from Semantic Memory. J. Verb. Learn. Verb. Behavior, 8, 240-247
- Conrad, C. (1972) Cognitive Economy in Semantic Memory, J. Exp. Psychol., 92, 149-154
- Craik, F.I.M. und R.S. Lockhart (1972) Levels of processing: A Framework for Memory Research, J. verb. learn. verb. behav., 11, 671-684
- Davis, M. (1958) Computability and Unsolvability. McGraw-Hill, New York

- Davis, R. und J. King (1977) An Overview of Production Systems, in: Elcock, E.W. und Michie, D. (Hrsg.) Machine Intelligence 8. Edinburgh University Press
- Denofsky, M.D. (1976) How Near is Near? A Near Specialist. MIT Artificial Intelligence Laboratory, Cambridge, Mass., Memo 344
- Erman, L.D. und V.R. Lesser (1975) A Multi-Level Organization for Problem Solving Using Many Diverse Cooperating Sources of Knowledge. Advance Papers of the 4th International Joint Conference on Artificial Intelligence, Tbilissi, UdSSR, 483-490
- Eisenstadt, M. und Y. Kareev (1975) Aspects of Human Problem Solving: The Use of Internal Representations, in: Norman, D.A. und D.E. Rumelhart, Explorations in Cognition. Freeman, San Francisco, 308-346
- Fahlman, S. (1974) A Planning System for Robot Construction Tasks. Artificial Intelligence, 5, 1-49
- Fahlman, S. (1979) Net1: A System for Representing and Using Real-World Knowledge. MIT Press, Cambridge, Mass.
- Feigenbaum, E.A. (1977) The Art of Artificial Intelligence: Themes and Case Studies of Knowledge Engineering. Advance Papers of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass., 1014-1029
- Feigenbaum, E.A. (1982) Heuristic Knowledge and its Applications, in: Groner, R., W.F. Bischof und M. Groner (Hrsg.) Methods in Heuristics, Lawrence Erlbaum Associates, (in Vorbereitung)
- Feigenbaum, E.A., B.G. Buchanan und J. Lederberg (1971) On Generality and Problem Solving: A Case Study using the Dendral Program, in: Meltzer, B. und D. Michie (Hrsg.) Machine Intelligence 6. American Elsevier, New York, 165-190
- Fillmore, C. (1971) Some Problems for Case Grammar. Working Papers in Linguistics, 10, Ohio State University, 245-265
- Fodor, J. und M. Garrett (1966) Some Reflections on Competence and Performance, in: Lyons, J. und R. Wales (Hrsg.) Psycholinguistic Papers. Edinburgh University Press, Edinburgh, 135-183
- Freedman, J.L. und e.F. Loftus (1971) Retrieval of Words from Long-Term Memory. J. Verb. Learn. Verb. Behav., 10, 107-115
- Garey, M.R. und Johnson, D.S. (1979) Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco

- Goldman, N.M. (1974) Computer Generation of Natural Language from a Deep Conceptual Base. Institute for Semantic and Cognitive Studies, Genf, Working Paper 2
- Graesser, A.C., S.E. Gordon und J.D. Sawyer (1979) Recognition Memory for Typical and Atypical Actions in Scripted Activities: Test of a Script Pointer + Tag Hypothesis. *J. Verb. Learn. Verb. Behav.*, 18, 319-332
- Groner, R. (1978) Hypothesen im Denkprozess. Huber, Bern
- Groner, R., W.F. Bischof und M. Groner (1982, Hrsg.) *Methods in Heuristics*. Lawrence Erlbaum Associates, New Jersey (in Vorbereitung)
- Gross, M. und A. Lentin (1971) *Mathematische Linguistik*. Springer, Berlin
- Gunderson, K. (1964) The Imitation Game. in: Anderson, S. (Hrsg.) *Minds and Machines*. Prentice-Hall, Englewood Cliffs, N.J.
- Hall-Partee, B. (1971) On the Requirement that Transformations Preserve Meaning, in: Fillmore, C. und D. Langendoen (Hrsg.) *Studies in Linguistic Semantics*. Holt, Rinehart & Winston, 1-22
- Hayes, Pat (1975) Some Problems and Non-Problems in Representation Theory. *Proceedings of the 1st AISB Conference, Sussex*, 63-79
- Hayes, Pat (1977a) In Defense of Logic. *Advance Papers of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass.*, 559-565
- Hayes, Pat (1977b) The Logic of Frames. Technical Report, Department of Computer Science, University of Essex. Auch in: Metzging, D. (Hrsg.) *Frame Conceptions and Text Understanding*. de Gruyter, Berlin, 1979
- Hayes, Pat (1978) Naive Physics I: Ontology for Liquids. Institute for Semantic and Cognitive Studies, Genf, Working Paper 35
- Hayes, Phil (1976) A Process to Implement some Word-Sense Disambiguations. Institute for Semantic and Cognitive Studies, Genf, Working paper 23
- Hayes, Phil (1977a) On Semantic Nets, Frames and Associations. *Proceeding of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass.*, 99-107
- Hayes, Phil (1977b) Some Association-Based Techniques for Lexical Disambiguation by Machine. Doctoral Thesis. Ecole Polytechnique Federale de Lausanne, Lausanne

- Hays, D. (1964) Dependency Theory: A Formalism and some Observations. Language, 40, 511-525
- Hendrix, G. (1975) Expanding the Utility of Semantic Networks through Partitioning. Advance Papers of the 4th International Joint Conference on Artificial Intelligence, Tbilissi, UdSSR, 115-121
- Hendrix, G. (1979) Encoding Knowledge in Partitioned Networks, in: Findler, N.V. (Hrsg.) Associative Networks, Academic Press, New York
- Hewitt, C. (1972) Description and Theoretical Analysis (Using Schemata) of Planner: A Language for Proving Theorems and Manipulating Models in a Robot. MIT Artificial Intelligence Laboratory, Cambridge, Mass., MIT-TR-258
- Hintzman, D. (1968) Explorations with a Discrimination Net Model for Paired Associate Learning. J. Math. Psychol., 5
- Hopcroft, J.E. und Ullman, J.D. (1969) Formal Languages and their Relation to Automata. Addison-Wesley, Reading, Mass.
- Johnson, N.F. (1965) Linguistic Models and Functional Units of Language Behavior, in: Rosenberg, S. (Hrsg.) Directions in Psycholinguistics. McMillan, London, 29-65
- Kaplan, R.M. (1972) Augmented Transition Networks as Psychological Models of Sentence Comprehension. Artificial Intelligence, 3, 77-100
- Kay, M. (1973) A General Syntactic Processor, in: R. Rustin (Hrsg.) Natural Language Processing. Algorithmics Press, New York, 193-241
- Kay, M. (1975) Syntactic Processing and Functional Sentence Perspective. Proceedings of the Workshop on Theoretical Issues in Natural Language Processing 1, Cambridge, Mass., 6-9
- Katz, J.J. und J. Fodor (1963) The Structure of Semantic Theory. Language, 39 (2), 170-210
- Katz, J.J. und J. Fodor (1964) An Integrated Theory of Linguistic Description. MIT Press, Cambridge, Mass.
- King, M. (1977) Several Ways to be Suggestive: An Examination of Presupposition. Institute for Semantic and Cognitive Studies, Genf, Working Paper 25
- Kiparsky, P. und D. Kiparsky (1970) Fact, in: Bierwisch, M. und K. Heidolph (Hrsg.) Progress in Linguistics. Mouton, The Hague
- Knuth, D.E. (1975) The Art of Computer Programming: Sorting and Searching. Vol. 3, Addison-Wesley, Reading, Mass.

- Kuhn, T. (1962) The Structure of Scientific Revolutions. University of Chicago Press, Chicago
- Lesser, V.R., R.D. Fennell, L.D. Erman und D.R. Reddy (1975) Organization of the Hearsay-II Speech Understanding System. IEEE Transactions on Acoustics, Speech and Signal Processing, 23 (1), 11-23
- Lenat, D.B. (1976) AM: An Artificial Intelligence Approach to Discovery in Mathematics as Heuristic Search. Stanford Artificial Intelligence Laboratory, Memo AIM-286
- Lenat, D.B. (1977) The Ubiquity of Discovery. Artificial Intelligence, 9, 257-285
- Lenat, D.B. (1980) The Nature of Heuristics. Stanford Artificial Intelligence Laboratory, Technical Report
- Lenat, D.B., F. Hayes-Roth und P. Klahr (1979) Cognitive Economy in Artificial Intelligence. Proceedings of the 6th International Joint Conference on Artificial Intelligence, Tokyo, Japan, 531-536
- Leveque, H. und J. Myopoulos (1979) A Procedural Semantics for Semantic Networks, in: Findler, N.V. (Hrsg.) Associative Networks. Academic Press, New York, 93-121
- Levitt, M. und A. Warshel (1975) Computer Simulation of Protein Folding. Nature, Vol. 253, 694-698
- Lindsay, R.K. (1963) Inferential Memory as the Basis of Machines which Understand Natural Language, in: Feigenbaum E. und J. Feldman (Hrsg.) Computers and Thought. McGraw-Hill, New York, 217-236
- Marcus, M. (1974) Wait-and-See Strategies for Parsing Natural Language. MIT Artificial Intelligence Laboratory, Cambridge, Mass., WP 75
- Marcus, M. (1975) Diagnosis as a Notion of Grammar. Proceedings of the Workshop on Theoretical Issues in Natural Language Processing 1. Cambridge, Mass., 10-14
- Marcus, M. (1979) A Theory of Syntactic Recognition for Natural Language, in: Winston, P.H. und R.H. Brown (Hrsg.) Artificial Intelligence: An MIT Perspective. MIT Press, Cambridge, Mass.
- Markov, A.A. (1951) Theorie der Algorithmen. Trudy Mat. Inst. Steklova, 38, 176-189
- Marr, D. (1976a) Early Processing of Visual Information. Philosophical Transactions of the Royal Society of London, 275 (942), 483-534
- Marr, D. (1976b) Artificial Intelligence - A Personal View. MIT Artificial Intelligence Laboratory, Memo 335

- Matthews, G. (1961) Analysis by Synthesis of Sentences in a Natural Language. Proceedings of the International Conference on Machine Translation and Applied Linguistic Analysis, H.M.S.O., London
- McCarthy, J. (1961) A Basis for a Mathematical Theory of Computation, in: Proceedings of the Western Joint Computer Conference
- McDermott, D.V. (1977) Flexibility and Efficiency in a Computer Program for Designing Circuits. MIT Artificial Intelligence Laboratory, Cambridge, Mass., TR-402
- McDermott, D.V. und J. Doyle (1979) Non-Monotonic Logic I. MIT Artificial Intelligence Laboratory, Cambridge, Mass., Memo 486a
- McDermott, D.V. und G. Sussman (1972) The Conniver Reference Manual. MIT Artificial Intelligence Laboratory, Cambridge, Mass., TR-402
- McKean, K., D. Slobin und G.A. Miller (1962) cited in Miller, G.A., Some Psychological Studies of Grammar, American Psychologist, 17, 748-762
- Meyer, D.E. (1970) On the Representation and Retrieval of stored Semantic Information, Cognitive Psychology, 1, 242-300
- Minsky, M. (1975) A Framework for Representing Knowledge, in: Winston, P.H. (Hrsg.) The Psychology of Computer Vision. McGraw-Hill, New York
- Minsky, M. (1977) Plain Talk about Neurodevelopmental Epistemology. Advance Papers of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass., 1083-1092
- Newell, A. (1980) Physical Symbol Systems. Cognitive Science, 4 (2), 135-183
- Newell, A. und H. Simon (1972) Human Problem Solving. Prentice-Hall, Englewoods Cliffs, New Jersey
- Newell, A. und H. Simon (1976) Computer Science as Empirical Inquiry: Sumbols and Search, Communications of the ACM, 19 (3), 113-126
- Nilsson, N.J. (1971) Problem-Solving Methods in Artificial Intelligence. McGraw-Hill, New York
- Norman, D.A. und D.G. Bobrow (1975) On Data-Limited and Resource-Limited Processes. Cognitive Psychology, Vol. 7, 44-64
- Norman, D.A. und D.E. Rumelhart (1975) Explorations in Cognition. Freeman, San Francisco
- Palmer, S.E. (1975) Visual Perception and World Knowledge: Notes on a Model of Sensory-Cognitive Interaction, in: Norman, D.A. und D.E. Rumelhart, Explorations in Cognition. Freeman, San Francisco, 279-307

- Paxton, W.H. (1974) A Best-First Parser. Stanford Research Institute, Menlo Park, California, Technical Note 92
- Petrick, S. (1965) A Recognition Procedure for Transformational Grammars. Unpublished Dissertation, MIT, Cambridge, Mass.
- Post, E. (1943) Formal Reductions of the General Combinatorial Problem. American Journal of Mathematics, 65, 197-268
- Pylyshyn, Z.W. (1980) Computation and cognition: issues in the foundations of Cognitive Science, The Behavioral and Brain Sciences, 3 (3), 111-132
- Quillian, M.R. (1968) Semantic Memory, in: Minsky, M. (Hrsg.) Semantic Information Processing. MIT Press, Cambridge, Mass., 227-270
- Quillian, M.R. (1969) The Teachable Language Comprehender: A Simulation Program and Theory of Language. Communications of the ACM, 12, 459-476
- Quine, W.V. (1951) Mathematical Logic. Harpe Torchbooks, Harper&Row, NY
- Raphael, R. (1968) SIR: Semantic Informaton Retrieval, in: Minsky, M. (Hrsg.) Semantic Information Processing. MIT Press, Cambridge, Mass., 33-145
- Reddy, R., L.D. Erman, R.D. Fennell und R.B. Neely (1973) The Hearsay Speech Understanding System: An Example of the Recognition Process. Advance Papers of the 3rd International Joint Conference on Artificial Intelligence, Stanford, Cal., 185-193
- Reddy, R. und A. Newell (1974) Knowledge and its Representation in a Speech Understanding System, in: Gregg, L.W. (Hrsg.) Knowledge and Cognition. Lawrence Erlbaum Associates, Potomac, Md. 252-285
- Rich, C. und H.E. Shrobe (1976) Initial Report on a LISP Programmer's Apprentice. MIT Artificial Intelligence Laboratory, Cambridge, Mass., AI-TR-354
- Rieger, C. (1974) Conceptual Memory: A Theory and Computer Program for Processing the Meaning Content of Natural Language Utterances. Ph.D. Thesis, Stanford Artificial Intelligence Laboratory, Memo 233
- Rieger, C. (1975) The Commonsense Algorithm as a Basis for Computer Models of Human Memory, Inference, Belief and Contextual Language Comprehension. Proceedings of the Workshop on Theoretical Issues in Natural Language Processing 1, Cambridge, Mass., 199-214
- Rieger, C. (1977) GRIND-1: First Report on the Magic Grinder Story Comprehension Project. Computer Science Technical Report Series, University of Maryland, TR-588

- Rieger, C. (1979) Five Aspects of a Full-Scale Story Comprehension Model, in: Findler, N.V. (Hrsg.) Associative Networks. Academic Press, New York, 425-462
- Riesbeck, C. (1974) Computer Analysis of Natural Language in Context. Institute for Semantic and Cognitive Studies, Genf, WP 4
- Riesbeck, C. (1975) Conceptual Analysis, in: Schank, R., Conceptual Information Processing. North-Holland, Amsterdam
- Riesbeck, C. (1977) Delayed Interpretation Using Processing Notes. Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass., 83
- Riesbeck, C. (1978) An Expectation-Driven Production System for Natural Language Understanding, in: Waterman, D.A. und F. Hayes-Roth (Hrsg.) Pattern-Directed Inference Systems. Academic Press, New York, 415-430
- Riesbeck, C. und R. Schank (1976) Comprehension by Computer: Expectation-Based Analysis of Sentences in Context. Yale University, Department of Computer Science, Research Report 78
- Rips, L.R., E.J. Shoben und E.E. Smith (1973) Semantic Distance and the Verification of Semantic Relations. J.verb.Learn.verb.Behav., 12, 1-20
- Rosner, M. und H. Somers (1980) Case in Linguistics and Cognitive Science. Institute for Semantic and Cognitive Studies, Genf, WP 40
- Rumelhart, D.E. (1977) Introduction to Human Information Processing. Wiley, New York
- Rumelhart, D., P. Lindsay und D.A. Norman (1972) A Process Model for Long Term Memory, in: Tulving, E. und W. Donaldson (Hrsg.) Organization of Memory, Academic Press, New York, 198-221
- Savin, H. und E. Perchonok (1965) Grammatical Structure and the Immediate Recall of English Sentences. J.verb.Learn.verb.Behav., 4, 348-353
- Schank, R. (1972) Conceptual Dependency: A Theory of Natural Language Understanding. Cognitive Psychology, 3 (4), 552-631
- Schank, R. (1973) Identification of Conceptualizations Underlying Natural Language, in: Schank, R. und K.M. Colby (Hrsg.) Computer Models of Thought and Language. Freeman, San Francisco
- Schank, R. (1978) Interestingness: Controlling Inferences. Artificial Intelligence, 12 (3), 273-297
- Schank, R. und R.P. Abelson (1977) Scripts, Plans, Goals and Understanding. Lawrence Erlbaum, Hillsdale, New Jersey

- Schank, R. und K.M. Colby (1973) Computer Models of Thought and Language. Freeman, San Francisco
- Schank, R., M. Lebowitz und L.A. Birnbaum (1978) Integrated Partial Parsing. Research Report 143, Department of Computer Science, Yale University, New Haven
- Schank, R. and the Yale AI Project (1975) SAM - A Story Understander. Report 43, Department of Computer Science, Yale University, New Haven
- Schubert, L.K. (1976) Extending the Expressive Power of Semantic Networks. Artificial Intelligence, 7, 163-198
- Scragg, G.W. (1975) Answering Questions about Processes, in: Norman, D.A. und D.E. Rumelhart (Hrsg.) Explorations in Cognition. Freeman, San Francisco, 349-375
- Shortliffe, E.H. (1976) Computer-Based Medical Consultations: MYCIN. American Elsevier, New York
- Simmons, R.F. (1973) Semantic Networks: Their Use for Understanding English Sentences, in: Schank, R. und K.M. Colby (Hrsg.) Computer Models of Thought and Language. Freeman, San Francisco
- Smith, E.E., E.J. Shoben und L.J. Rips (1974) Structure and Process in Semantic Memory: A Feature Model for Semantic Decisions. Psychological Review, 81, 214-241
- Suppes, P. (1957) Introduction to Loic. Van Nostrand, Princeton, NJ
- Suppes, P. (1969) Stimulus-response theory of finite automata. Journal of Mathematical Psychology, 6, 327-355
- Sussman, G. (1973) A Computational Model of Skill Acquisition. MIT Laboratory of Artificial Intelligence, Cambridge, Mass., TR-297
- Sussman, G., T. Winograd und E. Charniak (1971) Microplanner Reference Manual. MIT Laboratory of Artificial Intelligence, Report 203a
- Townsend, J.T. (1971) A Note on the Identifiability of Parallel and Serial Processes. Perception and Psychophysics, 10, 161-163
- Townsend, J.T. (1972) Some Results on the Identifiability of Parallel and Serial Processes. British Journal of Mathematical and Statistical Psychology, 10, 168-199
- Turing, A.M. (1950) Computing Machinery and Intelligence, Mind, 59, 433-460. Nachgedruckt in: Feigenbaum, E. und J. Feldman (Hrsg.) Computers and Thought. Mc Graw-Hill, New York, 11-38

- Ullman, S. (1979) The Interpretation of Visual Motion. MIT Press, Cambridge, Mass.
- Wahlster, W. (1977) Die Repräsentation von vagem Wissen in natürlich-sprachlichen Systemen der Künstlichen Intelligenz. Institut für Informatik, Universität Hamburg, Bericht Nr. 38
- Wilkins, W.A. (1971) Conjoint Frequency, Category Size and Categorization Time, *J. verb. learn. verb. behav.*, 10, 338-385
- Wilks, Y. (1968) On-Line Semantic Analysis of English Texts. Machine Translation and Computer Learning.
- Wilks, Y. (1973a) An Artificial Intelligence Approach to Machine Translation, in: Schank, R. und K.M. Colby (Hrsg.) *Computer Models of Thought and Language*. Freeman, San Francisco
- Wilks, Y. (1973b) Preference Semantics. Stanford Artificial Intelligence Laboratory, AIM-206
- Wilks, Y. (1975a) A Preferential, Pattern-Seeking Semantics for Natural Language Inference. *Artificial Intelligence*, Vol. 6, 53-74
- Wilks, Y. (1975b) Seven Theses on Artificial Intelligence and Natural Language. Institute for Semantic and Cognitive Studies, Genf, WP 17
- Wilks, Y. (1976) Parsing English II, in: Charniak, E. und Y. Wilks (Hrsg.) *Computational Semantics*. North-Holland, Amsterdam, 155-184
- Wilks, Y. (1977) Frames, Scripts, Stories and Fantasies. Department of Language and Linguistics, University of Essex
- Winograd, T. (1972) *Understanding Natural Language*. Academic Press, New York
- Winograd, T. (1975) Breaking the Complexity Barrier again. *Sigplan Notices of the Association for Computing Machinery*, 10 (1), 13-30
- Winston, P.H. (1975) Learning Structuralal Descriptions from Examples, in: Winston, P.H. (Hrsg.) *The Psychology of Computer Vision*. McGraw-Hill, New York, 157-210
- Woods, W. (1970) Transition Network Grammars for Natural Language Analysis. *Communications of the ACM*, 13, 591-602
- Woods, W. (1973) Progress in Natural Language Understanding: An Application to Lunar Geology. *AFIPS Proceedings of the 1973 National Computer Conference and Exposition*, 441-450

- Woods, W. (1975) What's in a Link: Foundations for Semantic Networks, in: Bobrow, D.G. und A. Collins (Hrsg.) Representation and Understanding. Academic Press, New York, 35-82
- Woods, W. (1977) Shortfall and Density Scoring Strategies for Speech Understanding Control. Advance Papers of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass., 18-26
- Woods, W. (1978) Multiple Theory Formation in High-Level Perception, in: Spiro, R.J., B.C. Bruce und W.F. Brewer (Hrsg.) Theoretical Issues in Reading Comprehension. Lawrence Erlbaum Associates, Hillsdale, New Jersey
- Woods, W., M. Bates, G. Brown, B. Bruce, C. Cook, J. Klovstad, J. Makoul, B. Nash-Webber, R. Schwartz, J. Wolf und V. Zue (1976) Speech Understanding Systems - Final Technical Progress Report, BBN Report 3438, Vols 1-4, Bolt, Beranek and Newman Inc., Cambridge, Mass.
- Woods, W., R.M. Kaplan und B. Nash-Webber (1972) The Lunar Sciences Natural Language Information System. Final Report. Bolt, Beranek and Newman Inc., Cambridge, Mass.
- Yolks, W. (1975) There's always Room at the Top or how Frames gave my Life Meaning. Sigart Newsletter, 53, 19-20

Lebenslauf

Am 2. Juli 1951 wurde ich in St. Gallen geboren, wo ich während 6 Jahren die Primarschule, während 2 Jahren die Katholische Kantonsrealschule und während 5 Jahren das Gymnasium besuchte. Im Herbst 1969 legte ich die Maturitätsprüfung B ab.

Im Herbst 1969 begann ich das Studium der Psychologie an der Universität Bern, zuerst an der Philosophisch-naturwissenschaftlichen Fakultät mit den Nebenfächern mathematische Statistik und Psychopathologie und ab 1973 an der Philosophisch-historischen Fakultät mit den Nebenfächern Philosophie und Psychopathologie. Im Sommer 1981 legte ich das Lizentiatsexamen bei Prof. Dr. N. Foppa ab.

Während meines Studiums habe ich an verschiedenen Forschungsprojekten mitgearbeitet, so von 1974 bis 1978 am Institute for Semantic and Cognitive Studies Lugano/Genf auf dem Gebiet des automatischen Sprachverstehens, von 1978 bis 1981 an einem europäischen Projekt über maschinelle Uebersetzung sowie in mehreren Nationalfondsprojekten über kognitive Prozesse und über visuelle Wahrnehmung.