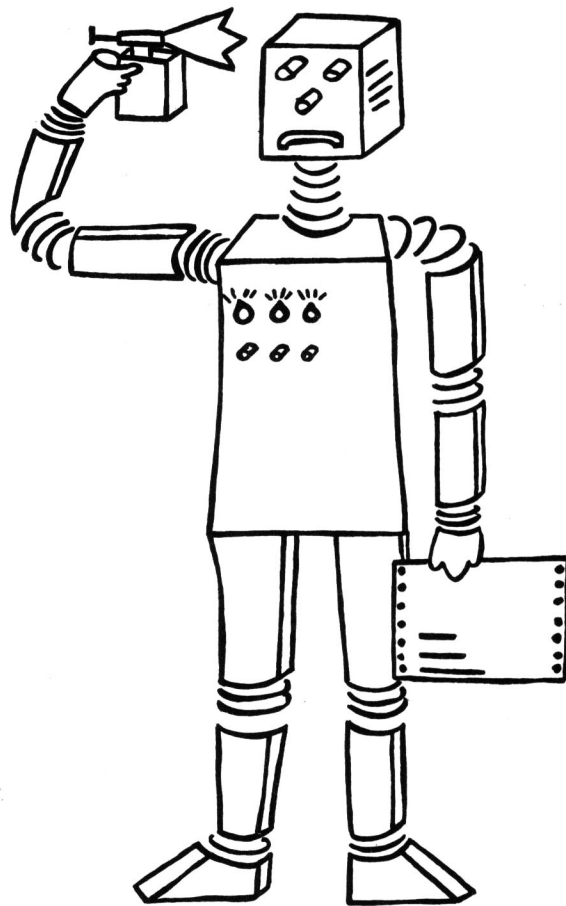


SEH-Working Paper

Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
D-6750 Kaiserslautern 1, W. Germany



Qualitative Modelle in
Wissensbasierten Systemen

Frank Puppe
Hans Voss

März 1986

SWP-86-01

QUALITATIVE MODELLE IN
WISSENSBASIERTEN SYSTEMEN

Frank Puppe

Hans Voss

Abstrakt: Diese Arbeit gibt einen Überblick über die Verwendung von qualitativen Modellen des Gegenstandsbereichs in wissensbasierten Systemen. Die Arbeit besteht aus zwei Teilen. In dem ersten Teil werden einige Systeme diskutiert (CADUCEUS, ABEL, Long's System, Davis' System), die verschiedene Typen von kausalen Modellen benutzen.

In dem zweiten Teil werden die wichtigsten Aspekte der entwickelten Modellierungsmethoden dargestellt. Die zwei wesentlichen Strukturierungsmethoden des objektorientierten und des prozeßorientierten Ansatzes werden gegenübergestellt. Constraint-Techniken als häufig verwendete Implementierungsmethode werden ausführlich behandelt. Neben Darstellungsmitteln für zeitliche und kausale Beziehungen wird die Einführung von Objektzuständen beschrieben. Anschließend wird die Unterscheidung zwischen stetigen und digitalen Variablen motiviert, und die Notwendigkeit einer Hierarchie von Modellbildungen erläutert. In einem ausführlichen Ausblick wird aufgezeigt, was bisher noch nicht erreicht wurde.

INHALTSVERZEICHNIS

1. Einleitung

Teil I:

2. Überblick über Modellbasierte Expertensysteme

2.1 CADUCEUS

2.2 ABEL

2.3 Long's System

2.4 Davis' System

2.5 Diskussion

Teil II: Methoden der Modellierung

3. Der Situationskalkül

4. Der Common Sense Algorithmus

5. Objektorientierte Strukturbeschreibungen

6. Prozeßorientierte Strukturbeschreibungen

6.1 Überblick

6.2 Die Entwicklung der Prozeßtheorie

7. Verhaltensbeschreibung und Implementierung

7.1 Einfache Relationen

7.2 Weitgehend einfache Relationen

7.3 Constraint Propagierung mit symbolischen
Ausdrücken

7.4 Destruktive Constraint Propagierung

8. Kausale Beziehungen

9. Zeitliche Beziehungen

10. Zustände von Objekten

11. Digitale und stetige Variablen

- 12. Hierarchien von Modellen
- 13. Simulation und Envisioning
- 14. Zusammenfassung
- 15. Ausblick
- 16. Literatur

ANHANG A: Das System von Davis

ANHANG B: Das System von de Kleer

1. EINLEITUNG

Beschreibungen technischer und biologischer Systeme sind ein wichtiges Teilgebiet der Informatik. Programmsysteme für

- Simulationsmethoden
- Computerunterstütztes Konstruieren und Fertigen (CAD/CAM)
- medizinische Diagnostik und Therapie
- Beschreibung/Diagnostik/Design/Verifikation von Hardware

haben eine längere bis lange Tradition. Die meisten in der Informatik verwendeten Wissensrepräsentationen und Algorithmen haben jedoch häufig den Nachteil, sehr unflexibel auf einen bestimmten Zweck unter sehr harten Randbedingungen ausgerichtet zu sein. Die Inkaufnahme dieses Nachteils ermöglicht andererseits oft erst eine genügend effiziente Anwendung.

Seit ca. zehn Jahren gibt es in der Künstlichen Intelligenz (KI) ernsthafte Bemühungen, allgemeinere und vielseitigere Repräsentationsmethoden für physikalische Systeme zu entwickeln.

Beispiel: Der klassische D-Algorithmus [Roth 66] dient zum Erkennen von einfachen stuck-at Fehlern in digitalen Schaltkreisen. Es werden dabei nur Fehler betrachtet, bei denen Leitungen permanent den konstanten Wert 0 oder 1 haben. Zudem wird angenommen, daß höchstens genau ein solcher Fehler vorliegen kann.

Da das vom D-Algorithmus zu lösende Problem der Testgenerierung NP-vollständig ist, wird man nicht mit einer besonders effizienten Implementierung rechnen können. Dennoch ist man zurecht sehr bemüht, die vorhandenen klassischen Algorithmen zu verbessern (z.B. [Blum 85]).

In [Davis 84] wird ein KI-Ansatz vorgestellt, der eine ganzheitliche Betrachtung einer großen Klasse von Fehlern zur Diagnostik von digitalen Schaltkreisen erlaubt. Obwohl über die Komplexität der dort verwendeten Verfahren keine klaren Aussagen gemacht werden, so werden doch neue Wege zu umfassenderen Lösungen aufgezeigt. Auf allgemeinere Eigenschaften dieses und einiger anderer Systeme werden wir im Laufe der folgenden Kapitel näher eingehen.

Man sollte die in der KI entwickelten Methoden gewiß nicht als ein Wundermittel darstellen. Es ist jedoch eine objektive Beobachtung, daß KI-Methoden zur Diagnostik, Planung u.ä. in immer weiteren Anwendungen im technischen und biologischen Bereich er-

probt bzw. für eine Verwendung in Betracht gezogen werden. In einigen Fällen wird dabei bezogen auf den gegenwärtigen Entwicklungsstand (noch) zu viel erwartet. Die Entwicklung allgemeiner Verfahren zur Modellierung technischer Systeme z.B. als Informationsquelle für die Folgerungsprozesse eines diagnostischen Expertensystems steckt noch in den Kinderschuhen. Von einer einheitlichen, weithin anerkannten Theorie oder Methodik über Modellierungen dieser Art ist man zur Zeit noch ein gutes Stück entfernt.

Die noch fehlende Systematik hindert allerdings nicht daran, mit einzelnen Modellierungen für spezielle Anwendungen erfolgreiche Systeme zu konstruieren. Die Vielfalt von Vorschlägen kann in der jetzigen Experimentierphase nur von Vorteil sein. Sowohl im medizinischen Bereich für physiologische und pathophysiologische Modelle als auch bei der Modellierung technischer Systeme gilt:

Es gibt nicht eine einzige adäquate Modellierung eines Systems für jeden Zweck, sondern es gibt im allgemeinen ein kontinuierliches Spektrum von Modellierungen auf allen Ebenen der Detaillierung. Auf jeder Detaillierungsebene wiederum gibt es möglicherweise verschiedene Beschreibungen, die das modellierte Objekt unter jeweils besonderen Gesichtspunkten betrachten.

Dieses Spektrum von Beschreibungen reicht von exakten Modellen z.B. mittels quantitativer Differentialgleichungen bis hin zu stark abstrahierenden, assoziativen Regeln z.B. eines Diagnostik-Expertensystems. Bisher ist keine Modellierung bekannt, in der das gesamte Spektrum einheitlich auf genügend vielen Ebenen erfaßt werden kann. Eine Hauptaufgabe für die Zukunft besteht daher darin, einen integrativen Ansatz zur Beschreibung eines Systems auf verschiedenen Abstraktionsebenen zu finden, der exakte quantitative Modelle - soweit vorhanden - genauso umfaßt wie möglicherweise ungenauere qualitative oder assoziative Modelle.

TEIL I2. ÜBERBLICK ÜBER MODELLBASIERTE EXPERTENSYSTEME

Eine wesentliche Motivation für die Entwicklung von kausalen Modellen für biologische und technische Systeme kommt von den bisherigen Erfahrungen mit assoziativen Expertensystemen (XPSe). Trotz ihrer großen Erfolge wird zunehmend klarer, daß ein System, das eine große Menge von Regeln auf ein Problem anwendet, noch keineswegs alle Fähigkeiten eines Experten simulieren kann. So können Experten nicht nur Routineprobleme lösen (wie derzeitige XPSe), sondern sind auch bei neuen Problemen nicht hilflos, können scheinbare Widersprüche häufig durch Berücksichtigung von Zusatzwissen auflösen, sind in der Lage, ihre Regeln zu hinterfragen und zu begründen, überblicken Randgebiete ihres Spezialgebietes, besitzen die Fähigkeit zur Einschätzung der eigenen Kompetenz und, was am wichtigsten ist, lernen durch Erfahrung. In einem stark beachteten Übersichtsartikel [Davis 82] hat Davis illustriert, daß viele dieser Probleme sich durch explizite Repräsentation von kausalen Modellen des Anwendungsbereiches zumindest teilweise lösen lassen.

Kausale Modelle sind eine eher deklarative Wissensrepräsentation, mit denen Struktur, Verhalten und Funktion eines Systems relativ unabhängig von einer konkreten Problemstellung beschrieben werden. Zur Illustration geben wir ein Beispiel eines einfachen pathophysiologischen Modelles einer Krankheit (Fig. 2.1), das die Folge von Zuständen beschreibt, die von einer zugrundeliegenden Ursache zu den mit der Krankheit assoziierten Symptomen führen. (Wir orientieren uns im folgenden an der medizinischen Terminologie, die jedoch direkt auf technische Systeme übertragbar ist: Physiologie = Funktionsweise eines Systems und Pathophysiologie = fehlerhafte Funktionsweise eines Systems).

Einige Regeln für Angina Pectoris ohne probabilistische Bewertung (Schmerz = Brustschmerz):

Schmerzverstärkung = körperliche Anstrengung

Schmerzverstärkung = Aufregung

Schmerzabschwächung = körperliche Ruhe

Schmerzabschwächung = Nitroglycerin

Anzahl der Risikofaktoren (Rauchen etc.) = hoch

Ein einfaches pathophysiologisches Modell zur Erklärung dieser Regeln:



Erläuterung: Die Risikofaktoren Übergewicht, Bewegungsmangel, Rauchen, Diabetes und Bluthochdruck führen in Abhängigkeit vom Alter zur Arteriosklerose (Verhärtung der Blutgefäße), wovon insbesondere auch die das Herzen versorgenden Koronararterien betroffen sind. Aufgrund des verminderten inneren Durchmessers und der geringeren Elastizität können die sklerotischen Koronararterien einen erhöhten Blutbedarf des Herzens nicht befriedigen, was die Brustschmerzen verursacht.

Ursache für einen erhöhten Blutbedarf des Herzens sind z.B. körperliche Anstrengung und Aufregung. Die Unterversorgung des Herzens mit Blut kann kurzfristig durch Nitropräparate wie Nitroglycerin ausgeglichen werden, die eine Verringerung der Herzarbeit bewirken.

Fig. 2.1: einfaches pathophysiologisches Modell zur Erklärung assoziativer Regeln von Angina Pectoris

Ein komplizierteres Modell für Angina Pectoris würde von dem physiologischen Modell des gesunden Kreislaufs ausgehen und zeigen, welche pathologischen Veränderungen zur Angina Pectoris führen.

Ein physiologisches Modell hat den Vorteil, daß Krankheiten, die in dem pathophysiologischen Modell explizit beschrieben werden müssen, sich als Fehlverhalten bestimmter Komponenten

beschreiben lassen. Selbstverständlich gibt es nicht nur ein (patho)physiologisches Modell, sondern beliebig viele Abstraktionsebenen. Je detaillierter ein kausales Modell ist, desto mehr Wechselwirkungen kann es berücksichtigen und daher manche Erklärungen ausschließen, die mit einem einfachen Modell plausibel erscheinen.

Wenn in einem kausalen Modell verschiedene Erklärungen für gegebene Symptome existieren, wird als Entscheidungskriterium "Occam's Razor" benutzt, d.h. die einfachste Erklärung (z.B. die mit der geringsten Anzahl unabhängiger pathologischer Prozesse) ist die beste.

Ein kausales Modell kann als ein System von miteinander in Beziehung stehenden Objekten (Komponenten) beschrieben werden. Die Komponenten können Attribute (Variablen) besitzen, die ihren Zustand charakterisieren und ihr Verhalten beeinflussen. Das Verhalten läßt sich häufig als Relation (Constraint) zwischen ihren Input- und Output-Werten beschreiben. Auf einer niedrigen Abstraktionsebene ist es möglich, daß Verhalten einer Komponente aus seiner physikalischen (geometrischen) Struktur herzuleiten. In einem Gesamtsystem hat eine Komponente meist eine bestimmte Funktion, die ihr Sollverhalten beschreibt. Die Feststellung der Unterschiede zwischen dem Sollverhalten und dem tatsächlichen Verhalten ist ein wichtiges Problem in vielen Anwendungsbereichen.

Die Beschränkungen kausaler Modelle liegen im unvollständigen Wissen und in ihrer Ineffizienz. Kein kausales System kann alle Wechselwirkungen berücksichtigen, die in einem System auftreten können (weil das Wissen unzureichend ist und weil die Beschreibung zu komplex würde). Insbesondere fehlt in kausalen Modellen meistens Wissen über die Wahrscheinlichkeit des Auftretens möglicher Ereignisse, die unter anderem auch von der Vorgeschichte des Systems abhängt (z.B. den Umweltbedingungen, denen es ausgesetzt war). Daher können kausale Modelle nur mögliche Erklärungen liefern, die durch Erfahrungswissen validiert werden müssen.

Selbst in Bereichen, wo sehr gute kausale Modelle für ein System existieren, z.B. in der KFZ-Diagnostik, spricht die Ineffizienz der Simulation eines detaillierten Modelles gegen seine routinemäßige Verwendung. Die Ineffizienz bezieht sich sowohl auf die Menge der benötigten Daten über den genauen Systemzustand als auch auf den Rechenaufwand bei der Simulation selbst. Wenn z.B. ein KFZ-Meister zweimal hintereinander bei einem bestimmten Motortyp denselben Fehler festgestellt hat, wird er beim dritten Motor mit ähnlichen Symptomen sofort diesen Fehler vermuten, ohne die kausalen Mechanismen zu berücksichtigen.

In der Praxis sind die existierenden modellbasierten Expertensysteme noch ausschließlich Forschungsprototypen, die in kleinen, sorgfältig ausgewählten Anwendungsgebieten getestet werden - im Gegensatz zu den teilweise schon industriell einsetzbaren assoziativen Expertensystemen. Kriterien für die Einteilung von modellbasierten Expertensystemen sind die Art des verwendeten

Modells (pathophysiologisch oder physiologisch) und die Darstellungsebene (qualitativ oder quantitativ; Bestimmung von Zuständen oder von Zustandsänderungen). Die Komplexität des Modells ergibt sich aus der Berücksichtigung verschiedener Aspekte kausaler Beziehungen. Dazu gehören:

- multiple gleichzeitige Ursachen für einen Zustand (erfordert Repräsentation von Schweregraden von Zuständen und die Fähigkeit zur Summation der Einzeleffekte)
- Verschiedene Arten von Beziehungen: z.B. linear, multiplikativ, Schwellwerteffekt, reversibel, irreversibel
- Abhängigkeiten von verstärkenden oder abschwächenden Faktoren (Katalysatoreffekt) und von Randbedingungen
- Rückkopplungsschleifen
- zeitliche Beziehungen zwischen Ursache und Wirkung (z.B. gleichzeitig, verzögert, überlappend, etc.)
- verschiedene Abstraktionsebenen

Im folgenden diskutieren wir einige Systeme, die verschiedene Typen von kausalen Modellen benutzen:

- CADUCEUS (Diagnostik in der Inneren Medizin): einfaches qualitatives, pathophysiologisches Modell [Pople 82]
- ABEL (Diagnostik von Elektrolyt und Säure/Base Störungen): quantitatives, pathophysiologisches Modell [Patil 81]
- Long's System (Vorhersage von Therapieeffekten bei Herzinsuffizienz): qualitatives, physiologisches Modell [Long 86]
- Davis' System (Diagnostik von einfacher Hardware): quantitatives, physiologisches Modell [Davis 84]

2.1 CADUCEUS

Der Unterschied zwischen CADUCEUS und seinem rein assoziativem Vorgängersystem INTERNIST ist die Repräsentation und Verwendung einer umfassenden Diagnoseheterarchie, in der es zwei Typen von Beziehungen zwischen Pathokonzepten gibt:

- nosologische (hierarchische) Beziehungen: A ist eine Unterdiagnose von B, wenn A spezieller ist und mindestens ein wichtiges Symptom mit B gemeinsam hat
- pathophysiologische Beziehungen: A ist durch B verursacht

Weitere Aspekte der Beziehungen (s.o.) werden nicht dargestellt; es werden auch keine Schweregrade von Pathokonzepten repräsentiert.

Die Inferenzstrategie von CADUCEUS besteht darin, zunächst einige allgemeine Pathokonzepte direkt aufgrund der vorhandenen Symptomatik zu etablieren. Zwar gibt es kaum Symptome, die typisch (pathognomonisch) für Feindiagnosen sind, aber durchaus

pathognomonische Symptome für Grobdiagnosen (z.B. Gelbsucht für Lebererkrankung oder akutes Fieber für Infektion). In der zweiten Phase versucht CADUCEUS, eine möglichst einfache Erklärung für die so etablierten Pathokonzpte zu finden. Dazu verwendet es sechs verschiedene Syntheseoperatoren, die die verschiedenen primitiven Verknüpfungsmöglichkeiten zweier Knoten (Symptom oder Pathokonzpt) in dem Netzwerk über nosologische und pathophysiologische Beziehungen repräsentieren:

- (1,2) A ist Unterdiagnose bzw. Ursache von B
- (3,4) A und B haben gemeinsame Ursache bzw. Unterdiagnose
- (5) A hat als Ursache eine Unterdiagnose von B
- (6) A und B haben Ursachen, von denen die eine die Unterdiagnose der anderen ist

Zur Effizienzsteigerung bei der Suche nach Verbindungen zwischen Knoten, die die Anwendung mehrerer Syntheseoperatoren erfordern, verfügt CADUCEUS noch über "Planungsbeziehungen". Sie verbinden einen Knoten mit allen Grobdiagnosen, bei denen es vom Knoten zu mindestens einer Unterdiagnose der Grobdiagnose einen kausalen Pfad gibt. Planungsbeziehungen werden beim Aufbau der Wissensbasis automatisch generiert.

Durch die Verwendung pathophysiologischer Beziehungen ist CADUCEUS wesentlich besser als INTERNIST in der Lage, kohärente Erklärungen für Symptome aus verschiedenen Organsystemen zu finden. Da jedoch die Repräsentation der kausalen Beziehungen äußerst grob ist (Existenz oder Nichtexistenz ohne die Qualifikation durch zusätzliche Bedingungen; s.o.), ist die endgültige Entscheidung über die Etablierung eines Pathokonzptes probabilistisch wie in INTERNIST. Die pathophysiologische und nosologische Heterarchie dient hauptsächlich zu einer im Vergleich zu INTERNIST wesentlich verbesserten Verdachtsgenerierung.

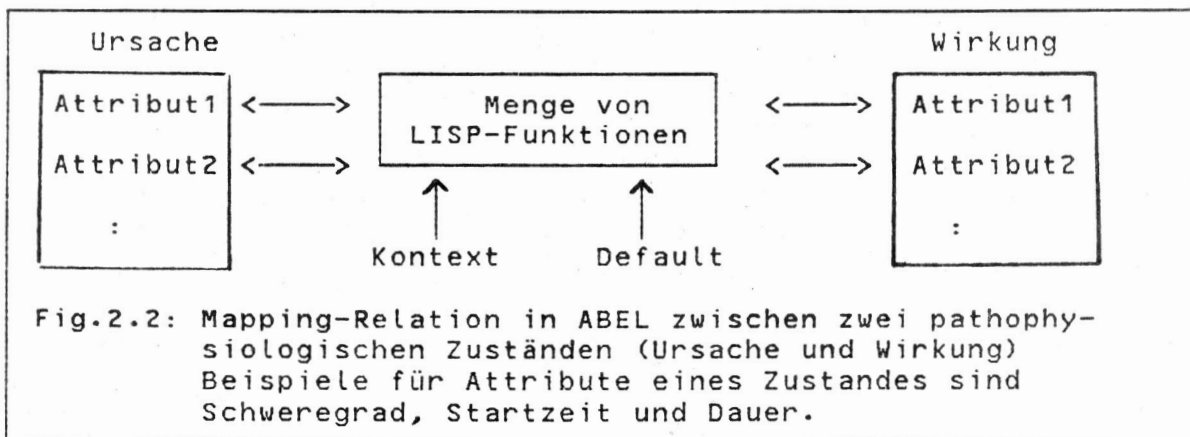
2.2 ABEL

ABEL benutzt ein wesentlich präziseres pathophysiologisches Modell als CADUCEUS. Seine besonderen Merkmale umfassen Krankheitsbeschreibungen auf verschiedenen Abstraktionsebenen und die Entwicklung von Techniken zur quantitativen Summation verschiedener Effekte auf den Schweregrad eines Zustandes. Die Verwendung einer quantitativen Beschreibungsebene ist in ABEL möglich, da sein Anwendungsbereich, nämlich Störungen im Säure/Base und Elektrolythaushalt, eines der wenigen medizinischen Gebiete ist, in dem sich die pathophysiologischen Zustände mittels Laborwerten quantitativ errechnen lassen und auch die kausalen Beziehungen hinreichend präzise verstanden sind.

Die globale Vorgehensweise von ABEL ist wie folgt: Aufgrund der Eingabelabordaten (PH-Wert und Konzentration verschiedener Elektrolyte: Na, K, Cl, HC03, pC02) konstruiert ABEL alle

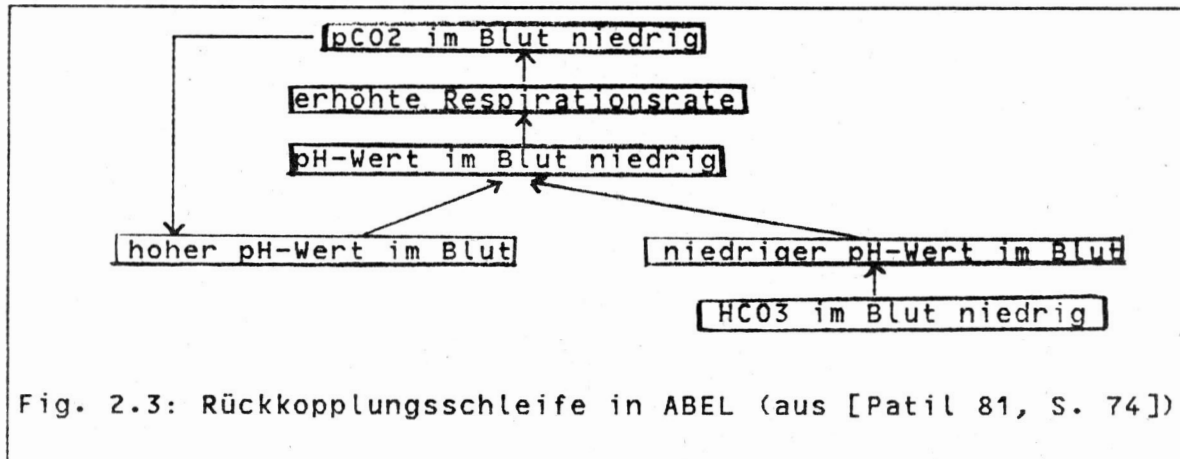
konsistenten patientenspezifischen Modelle (PSM's). Ein PSM repräsentiert eine Menge von pathophysiologischen Zuständen einschließlich ihren Schweregraden und den kausalen Beziehungen zwischen ihnen, die insgesamt eine konsistente Erklärung für die beobachteten Symptome des Patienten sind. PSM's werden nach der kleinsten Anzahl der unerklärten Zustände bewertet, die sie enthalten. Anschließend sucht ABEL für die besten ein oder zwei PSM's Diagnosen (coherent hypotheses, CH's), die die unerklärten pathophysiologischen Zustände verursachen können. Die CH's postulieren meistens noch weitere, bisher nicht im PSM vorhandene Zustände, deren Symptome zur Unterscheidung zwischen den möglichen Hypothesen erfragt werden. Die Bewertung von Hypothesen geschieht nach strukturellen Kriterien ohne Gebrauch probabilistischer Evidenzwerte: für eine Hypothese sprechen die Anzahl der Zustände im PSM, die die Hypothese erklären kann; gegen sie die Anzahl von Zuständen, die postuliert, aber nicht im PSM instanziiert sind.

Ein pathophysiologischer Zustand besitzt verschiedene Attribute wie Schweregrad, Startzeit und Dauer. Die kausalen Beziehungen zwischen verschiedenen Zuständen werden durch Constraints (mapping relations) dargestellt, die in beiden Richtungen (Ursache --> Wirkung und Wirkung --> Ursache) ausgewertet werden können. Ein Constraint (Fig. 2.2) ist als eine Menge von LISP-Funktionen implementiert, die die Attribute des einen Zustandes aus den Attributen des anderen Zustandes berechnen. Zusätzlich können Kontext und Default-Bedingungen für einen Constraint angegeben werden.



Die kombinierte Wirkung von verschiedenen Ursachen auf einen Zustand bestimmt ABEL durch "Komponentensummutation". Jede Komponente repräsentiert den Anteil des Gesamtzustandes, der von einer Ursache kommt. Die Verknüpfung erfolgt durch eine "mapping-relation", die typischerweise die Schweregrade addiert sowie das Minimum der Startzeiten und das Maximum der jeweiligen Dauer der einzelnen Komponenten nimmt. Durch die Fähigkeit zur Komponentensummutation kann ABEL auch Rückkopplungsschleifen repräsentieren (kausale Beziehungen, bei denen ein Zustand über verschiedene Zwischenzustände auf sich selbst verstärkend oder

abschwächend zurückwirkt), die in der Funktionsweise von biologischen Systemen eine überragende Rolle spielen (s. Fig. 2.3). Allerdings kann die Stärke einer Rückkopplungsschleife nur approximiert berechnet werden.



Die Repräsentation des pathophysiologischen Modells auf verschiedenen Abstraktionsebenen erreicht ABEL dadurch, daß es spezielle Zustände als Landmarken (focal nodes) auszeichnet. Auf der höchsten Abstraktionsebene werden nur die kausalen Beziehungen zwischen den Landmarken dargestellt, die auf niedrigeren Ebenen durch schrittweises Hinzufügen von Zwischenzuständen und ihrer Beziehungen einschließlich Rückkopplungsschleifen immer mehr verfeinert werden, bis sie auf der untersten Ebene vollständig dargestellt sind.

Die Integration einer neuen kausalen Beziehung ist nur auf der untersten Abstraktionsebene möglich, da nur auf dieser Ebene die (quantitative) Komponentensummutation durchführbar ist. Ein neuer Befund wird wie folgt in das PSM aufgenommen: Zunächst dient er zur Etablierung eines pathophysiologischen Zustandes auf der obersten Abstraktionsebene. Dieser wird bis auf die unterste Ebene "elaboriert", damit seine Auswirkungen auf andere Zustände berechnet werden können, indem die zugehörigen kausalen Beziehungen instanziiert und mit den Wirkungen der bereits vorhandenen kausalen Beziehungen durch Komponentensummutation verrechnet werden. Die Änderungen auf der untersten Abstraktionsebene werden anschließend auf den oberen Ebenen "aggregiert".

Insgesamt besitzt ABEL folgende fünf Operatoren zur Erweiterung und Konsistenzerhaltung seines PSM's:

- Initial formulation: Formulierung des PSM aufgrund der initialen Labordaten
- Aggregation: Zusammenfassung der Daten von einer niedrigen zur höheren Abstraktionsebene
- Elaboration: Umkehrung der Aggregation
- Component summation: Verrechnung der Wirkungen verschiedener Ursachen auf einen Zustand

Projection: Erweiterung des PSM durch Hinzufügen von Hypothesen zur Erklärung unerklärter Zustände im PSM

ABEL's Stärke ist seine Fähigkeit zur quantitativen Simulation der pathophysiologischen Vorgänge in seinem Anwendungsbereich. Es bewertet Hypothesen ausschließlich nach ihrem Erklärungswert in dem aktuellen PSM. Seine Vorgehensweise hängt entscheidend davon ab, daß die Schweregrade der Zustände quantitativ bestimmt werden können. Das kann auch nicht durch die höheren Abstraktionsebenen umgangen werden, die nur eine Zusammenfassung der detaillierten Simulation repräsentieren und keine eigenständigen, qualitativen Operatoren zur Propagierung kausaler Effekte enthalten.

2.3 LONGS SYSTEM

In den meisten medizinischen Teilgebieten lassen sich die internen Zustände nicht so leicht quantitativ bestimmen wie in ABEL. Wenn trotzdem die kausalen Beziehungen zwischen den Zuständen hinreichend gut bekannt sind, ist es möglich, eine qualitative Simulation durchzuführen, bei der nur die Richtung der Änderung eines Zustandes (verstärkend oder abnehmend) und nicht sein absoluter Wert hergeleitet wird. Ein entsprechendes physiologisches Modell mit qualitativer Simulation ist in Long's System zur Vorhersage von Therapieeffekten bei Herzinsuffizienz implementiert.

Seine globale Vorgehensweise besteht darin, daß zunächst einige wichtige Parameter des physiologischen Modells durch Daten des Patienten (z.B. Pulsfrequenz, Arteriosklerose der Koronararterien) abgeleitet werden. Dann wird eine mögliche Therapie der Beschwerden (z.B. Betablocker) auf ihre positiven Effekte und Nebenwirkungen bei dem speziellen Patienten überprüft. Die Therapie beeinflusst primär bestimmte Parameter in dem physiologischen Netzwerk, deren Änderungen (und nicht deren Absolutwerte wie in ABEL) propagiert werden. Das Ergebnis sind die Auswirkungen der Therapie auf alle repräsentierten Zustände im Modell.

Die Stärke der Änderung eines Zustandes wird durch positive oder negative Zahlenwerte repräsentiert, die durch Summierung aller Einflüsse von anderen Zuständen berechnet werden. Der Einfluß von einem anderen Zustand wird durch eine positive oder negative Stärke X qualifiziert ($X = \{-3/2, -1, -1/2, +1/2, +1, +3/2\}$ mit der Bedeutung: der Gewinn des Zustandes beträgt das X -fache der Änderung der Ursache). Das Hauptproblem ist die Repräsentation der verschiedenen Typen von Beziehungen zwischen den Parametern; sie umfassen:

- Mehrfach-Beziehungen (z.B. zwischen Pulsfrequenz und Herzminutenvolumen: 1) Pulsfrequenz x Herzschlagvolumen = Herzminutenvolumen. 2) je höher die Pulsfrequenz, desto geringer das Schlagvolumen)
- additive Beziehungen
- multiplikative Beziehungen
- nicht-lineare Beziehungen (z.B. Frank-Sterling-Beziehung: Wenn der Blutdruck des einlaufenden Blutes in die Herzkammer niedrig ist, dann hat steigender Blutdruck starken Einfluß auf das Schlagvolumen. Wenn der Blutdruck des einlaufenden Blutes in die Herzkammer hoch ist, dann hat steigender Blutdruck wenig zusätzlichen Einfluß auf das Schlagvolumen)
- Beziehungen, die Zeit erfordern (z.B. die Verstärkung der Herzmuskulatur wegen Herzinsuffizienz)

In Long's System werden alle Typen von Beziehungen linear approximiert und nichtlineare Beziehungen in annähernd lineare Abschnitte aufgeteilt. Die Aufteilung hängt von den Werten der beteiligten Parameter ab (in den obigen Beispielen von der Pulsfrequenz bzw. von dem Blutdruck des in die Herzkammer einlaufenden Blutes). Um bei einer Simulation den gültigen Abschnitt der Beziehung zu ermitteln, von dem die Stärke der Beziehung abhängt, muß daher der tatsächliche Zustand (und nicht nur die Zustandsänderung der relevanten Parameter bekannt sein; es reichen qualitative Angaben wie normal, hoch, niedrig, etc.). Long gibt an, daß diese Parameter gerade solche sind, die dem Arzt geläufig sind und deren Werte herleitbar sind. Alle Beziehungen haben ein Attribut, das die Größenordnung der Zeit angibt, die zur Ausprägung des Effektes erforderlich ist. Dabei wird angenommen, daß bei wesentlich kleineren Zeitintervallen der Effekt vernachlässigbar ist.

Die große Menge von Rückkopplungsschleifen (über 150) und der hohe Vernetzungsgrad (zwischen zwei Zuständen kann es über 70 Pfade geben) erfordern einen effizienten Inferenzmechanismus. Dazu adaptiert Long einen Algorithmus zur Analyse von Signalflüssen mit mehrfachem Input und Output [Mason 56] zur Simulation seines Netzwerkes, was möglich ist, da er nur (abschnittsweise) lineare Beziehungen verwendet. Die Details des Algorithmus sind in [Long 86] beschrieben. Kausale Beziehungen, die Zeit erfordern, werden nur dann berücksichtigt, wenn das Zeitintervall, in dem die Simulation stattfindet, entsprechend groß gewählt wurde.

Die Simulation hat nur geringen Erklärungswert, wenn alle vorhandenen Einflüsse zwischen den Zuständen gezeigt werden, da die meisten wegen der vielen Rückkopplungsschleifen stark gedämpft sind oder weil sie sich wechselseitig aufheben. Die Erklärungskomponente von Long's System ist deshalb in der Lage, die wichtigsten Pfade zwischen zwei Zuständen (der stärkste Pfad und weitere Pfade, deren Stärke einen wählbaren Prozentsatz des stärksten Pfades nicht unterschreitet) graphisch hervorzuheben. Weiterhin wird die Stärke einer Zustandsänderung nicht durch den Zahlenwert, sondern durch einen daraus hergeleiteten, qualitativen Wert (schwach erhöht, erhöht, stark erhöht bzw. erniedrigt) angegeben.

Ein Nachteil von Long's System ist, daß der Ableitungsprozeß selbst nicht erklärbar ist, was teilweise durch den Detaillierungsgrad des Modelles bedingt ist, der sicherlich wesentlich höher ist, als der Detaillierungsgrad der Modelle von Ärzten. Insbesondere Pfade, die sehr viele Rückkopplungsschleifen berühren, könnten ignoriert werden, da ihr Effekt äußerst gering ist. Dies würde die Überschaubarkeit des Modells verbessern. Ein anderes offenes Problem ist die Approximation der nichtlinearen in abschnittsweise lineare Beziehungen, da der anfangs zutreffende Abschnitt nicht konstant bleiben muß und sich der den Abschnitt bestimmende Parameter insbesondere über Rückkopplungsschleifen selbst beeinflussen kann.

2.4 DAVIS SYSTEM

Der Anwendungsbereich von Davis' System ist die Diagnostik von (einfacher) digitaler Hardware, die sich (besser als die bisher beschriebenen medizinischen Bereiche) sehr präzise modellieren ist. Davis' System verfügt über ein Modell der korrekt funktionierenden Hardware (physiologisches Modell), das bei der Diagnostik so manipuliert wird, daß es das beobachtete Verhalten der defekten Hardware simulieren kann.

Die Basiskonzepte zur Repräsentation der Hardware zeigt Fig. 2.4.

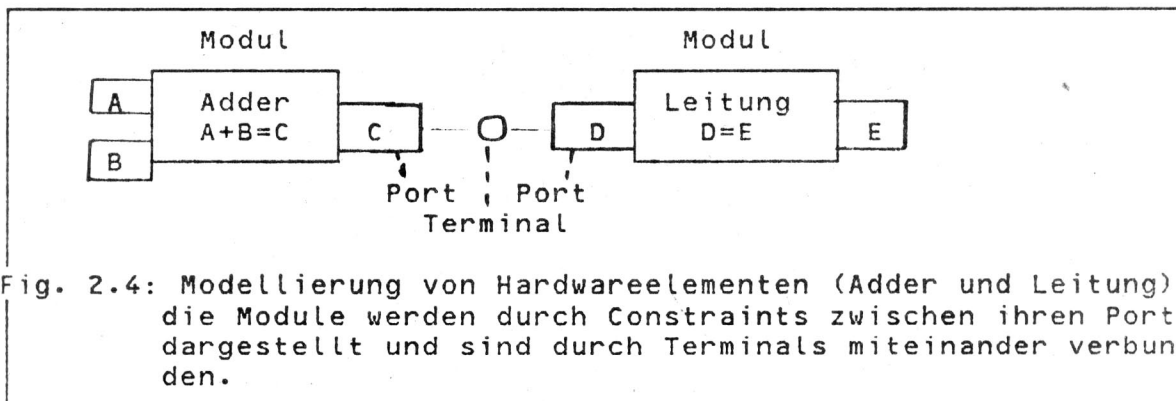


Fig. 2.4: Modellierung von Hardwareelementen (Adder und Leitung): die Module werden durch Constraints zwischen ihren Ports dargestellt und sind durch Terminals miteinander verbunden.

Die physikalischen Objekte der Hardware (Gatter, Leitungen, etc.) werden als "Module" dargestellt, deren Verhalten durch Constraints zwischen ihren Eingabe- und Ausgabe-"Ports" spezifiziert ist. Die Module sind durch Überlagerung ihrer Terminals miteinander verbunden. Terminals sind im Gegensatz zu Ports, die eine innere Struktur haben können (s.u.), primitive Konzepte. Ihre Werte (bits oder Zahlen) können von außen abgefragt werden. Das Verhalten der Module läßt sich typischerweise durch logische bzw. numerische Constraints (für

Gatter oder Leitungen bzw. Adder oder Multiplier) beschreiben.

Die Module werden auf verschiedenen Abstraktionsebenen repräsentiert. Innerhalb einer Ebene ist ein Modul eine Black-Box, von dem nur sein durch ein Constraint beschriebenes Verhalten bekannt ist. Auf der nächsten konkreteren Ebene wird die innere Struktur des Modules dargestellt und sein Verhalten ergibt sich aus dem Verhalten der Submodule. Die Umwandlung der Input und Output-Werte beim Übergang von einer Abstraktionsebene zu einer anderen (z.B. von Zahlen in Bits) wird durch die Ports bewerkstelligt. Im Gegensatz zu ABEL ist das Modell in Davis's System auf jeder Abstraktionsebene vollständig spezifiziert, weswegen eine tiefere Abstraktionsebene erst dann berücksichtigt werden braucht, wenn zusätzliche Details erforderlich sind.

Die Diagnostikstrategie von Davis' System ist wie folgt: Sobald an den Input-Terminals des Modells Werte angegeben werden, werden durch Constraint-Propagierung die Werte an allen übrigen Terminals (der obersten Abstraktionsebene) einschließlich der Output-Terminals des Modells ermittelt. Wenn die so vorhergesagten Output-Werte nicht mit den an dem zu diagnostizierenden Gerät beobachteten Werte übereinstimmen, wird die Ursache der Inkonsistenz wie folgt gesucht:

- i) Kandidatengenerierung (durch discrepancy detection): die fehlerhaften Werte werden rückwärts propagiert. Ein falscher Wert kann entweder in dem Modul liegen, das den fehlerhaften Wert berechnet oder ein Input zu diesem Modul war bereits fehlerhaft etc. Um die Anzahl der zu überprüfenden Kandidaten überschaubar zu halten, werden die Fehler in Fehlertypen eingeteilt, und erst alle Kandidaten eines Typs (auf allen Hierarchieebenen) überprüft, bevor Kandidaten des nächsten Fehlertypes in Betracht gezogen werden. Die beiden wichtigsten Fehlertypen sind:
 - (1) Ein Modul ist defekt, d.h. es verhält sich anders als es seiner durch ein Constraint beschriebenen Spezifikation entspricht. Dies wird durch Aufhebung des dem Modul zugeordneten Constraints simuliert (constraint suspension). Ein aufgehobenes Constraint kann jedes Verhalten simulieren.
 - (2) Zwei benachbarte Leitungen sind durch eine "Brücke" verbunden. Dies wird durch eine Erweiterung des Modells dargestellt, indem die entsprechenden Terminals der Leitungen durch ein zusätzliches Modul, das die Brücke repräsentiert, verbunden werden.
- ii) Kandidatenüberprüfung: Ein Kandidat (d.h. die zugehörige Modelländerung) ist konsistent, wenn das geänderte Modell das beobachtete Verhalten simulieren kann.

Um für Brückenfehler effizient Kandidaten generieren zu können, reicht ein rein funktionales Modell der Hardware nicht aus. Deswegen verfügt Davis' System über eine explizite Repräsentation der physikalischen Anordnung der Hardware, mit der es u.a.

überprüfen kann, welche Leitungen physikalisch benachbart sind. Eine methodisch orientierte, ausführlichere Beschreibung von Davis' System findet sich im Anhang.

2.5 DISKUSSION

Die vier diskutierten, modellbasierten Expertensysteme unterscheiden sich sehr stark in ihren Anwendungsbereichen und ihren Modellen. Am auffälligsten ist der Unterschied zwischen Davis' System und den medizinischen XPSen. Letzere basieren auf der Annahme, daß sich ihr Modell nicht verändert: CADUCEUS und ABEL suchen eine minimale Menge von externen Ursachen (Krankheiten) für pathophysiologische Zustände (die sich aufgrund der beobachteten Symptome relativ leicht herleiten lassen), wobei ABEL auch multiple Einflüsse auf einen Zustand und Rückkopplungsschleifen berücksichtigen kann. Long's System simuliert die Auswirkungen von Therapiemaßnahmen auf ein physiologisches Modell, das sich durch einige patientenspezifische Parameter einstellen läßt.

Davis' System hingegen versucht, die vorhandenen Symptome durch eine Änderung seines Modells zu erklären, so daß eine Simulation des geänderten Modelles die Symptome voraussagen bzw. bestätigen kann. Dieser prinzipielle Unterschied ist durch die verschiedenen Anwendungsgebiete bedingt: in den medizinischen Anwendungsgebieten geht es vor allem darum, externe Ursachen (Krankheiten bzw. Therapien) zu finden, mit denen die vorhandenen Symptome erklärt bzw. beeinflußt werden können. Die prinzipielle Funktionsweise des Organismus ist dabei kaum gestört. Dies ist ein generelles Merkmal von analogen Systemen, bei denen Schwankungen von Parametern einerseits durch Rückkopplungsschleifen stark gedämpft werden, andererseits innerhalb gewisser (unscharfer) Grenzen tolerierbar sind.

Bei der Modellierung digitaler Hardware gilt im Gegensatz zur "analogen Technik" das "Alles-oder-Nichts"-Gesetz: ein Teil funktioniert entweder korrekt oder gar nicht. Da Toleranzgrenzen und Rückkopplungsschleifen fehlen, verändert der Fehler eines Bauteils die Funktionsweise des Gesamtsystems, was bei der Modellierung berücksichtigt werden muß.

Die unterschiedliche Funktionsweise analoger und digitaler Systeme wirkt sich auch auf die Repräsentation verschiedener Abstraktionsebenen in einem Modell aus: in digitalen Systemen sind die verschiedenen Abstraktionsebenen jeweils vollständig spezifizierbar, ohne daß ein Rückgriff auf die niedrigeren Ebenen erforderlich ist (Beispiel in Davis' System: Die Daten der Zahlen- und Bit-Ebene sind jeweils in sich vollständig und lassen sich bijektiv ineinander überführen). In analogen Systemen zeichnen sich die verschiedenen Abstraktionsebenen jedoch durch einen unterschiedlichen Informationsgehalt aus (zwischen quanti-

tativen und den immer gröberen qualitativen Ebenen gibt es keine Bijektion, sondern es findet eine echte Datenabstraktion mit Informationsverlust statt). Während es relativ leicht ist, die Modelle der höheren Ebenen aus denen der unteren herzuleiten (wie die Aggregation in ABEL), ist es wesentlich schwieriger und bisher in keinem XPS realisiert, mit einem groben Modell eines analogen Systems zu beginnen und dieses bei Bedarf partiell zu verfeinern, da die Ergebnisse der Verfeinerung nicht notwendigerweise mit der groben Modellierung konsistent sind. Dieses und weitere Probleme bei der Darstellung kausaler Modelle werden ausführlicher im zweiten Teil besprochen.

TEIL II: METHODEN DER MODELLIERUNG

In den folgenden Kapiteln sollen die wichtigsten Gemeinsamkeiten und Unterschiede in den existierenden KI-Ansätzen über Methoden der Modellierung diskutiert werden. Wegen ihrer gemeinsamen Ursprünge und ihrer guten Vergleichbarkeit beschränken wir uns auf Modellierungen technischer Systeme. (Modelle biologischer Systeme sind historisch weitgehend unabhängig entstanden.)

Mit Ausnahme der Arbeiten von Patrick J. Hayes [Hayes 78,85a,85b] sind in [AI-Journal 84] die wesentlichsten und, bezogen auf 1984, aktuellen Arbeiten auf diesem Gebiet enthalten. Wir werden die wichtigsten Merkmale dieser Systeme allgemein darstellen, und in einigen Fällen auf konkrete Ausprägungen und Unterschiede hinweisen. Im Anhang sind die Systeme von Davis und das ENVISION-System von Brown und de Kleer [de Kleer/Brown 84] ausführlicher dargestellt. Diese Beschreibungen kann man je nach Verständnis jederzeit als Anschauungsmaterial zur Hilfe nehmen.

In den einzelnen Kapiteln wird zum Teil auf die historische Entwicklung der Ansätze eingegangen. Bild 2.1 zeigt grob die Zusammenhänge auf. Bei der Interpretation dieses Graphen sollte man beachten, daß die Kanten zum Ausdruck der Abhängigkeiten nicht immer dasselbe Gewicht haben. So ist der Einfluß des EL-Systems [Stallman/Sussman 77] auf das CONSTRAINTS-System [Steele/Sussman 80] höher zu bewerten als etwa die Beziehung zwischen EL und ENVISION. Naturgemäß ist der Graph nicht vollständig, sondern stellt eine subjektive Auswahl der wesentlichen Einflüsse dar.

Ab Kapitel 5 werden wir auf die allgemeinen Eigenschaften der heute aktuellen Modellierungen eingehen. In den Kapiteln 3 und 4 werden zwei frühere Ansätze skizziert, die sich zwar nicht durchsetzen konnten, von denen aber zumindest der erste Ansatz von John Mc Carthy lange Zeit einen wesentlichen Einfluß ausüben konnte. In Kapitel 14 werden die wesentlichsten Punkte noch einmal zusammengefaßt.

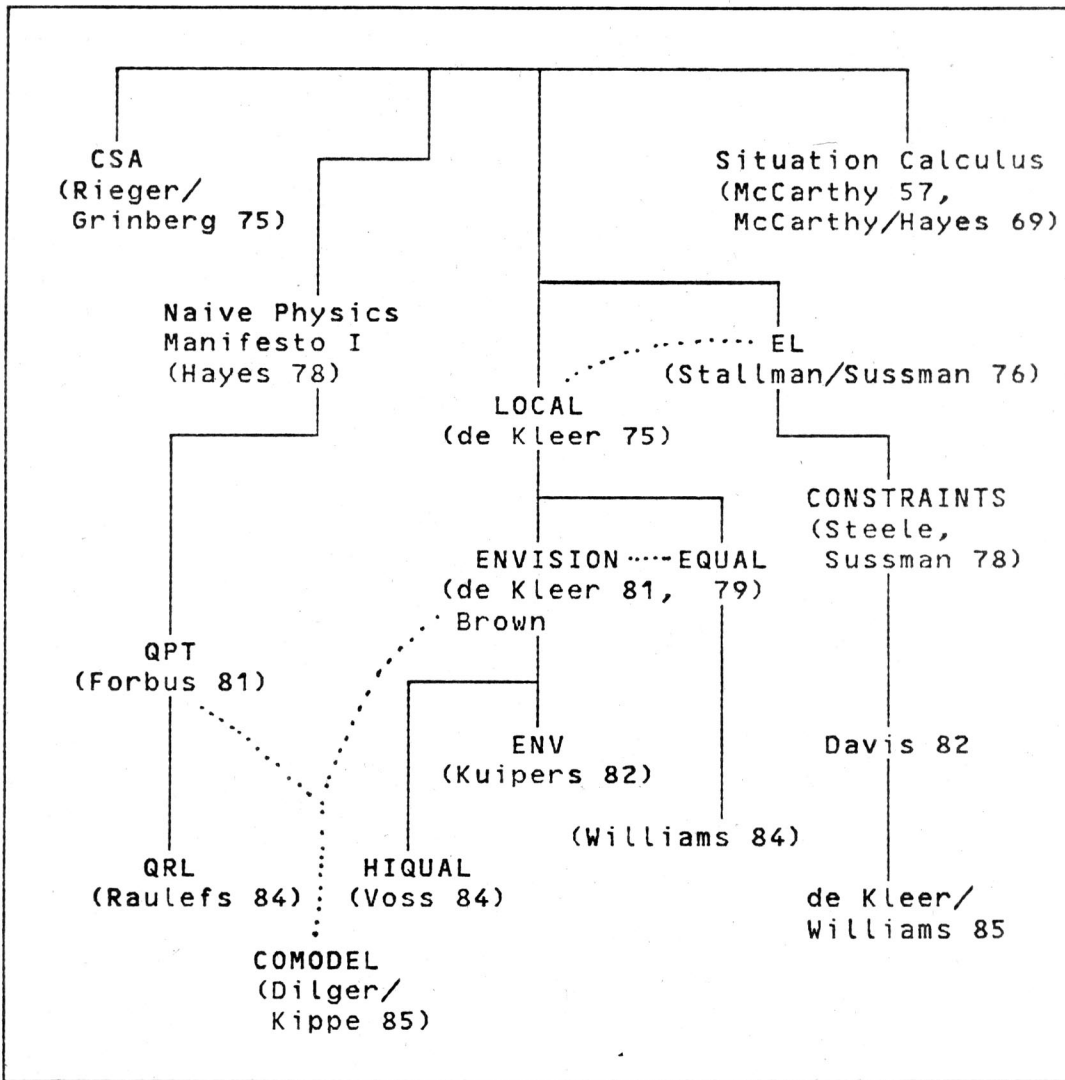


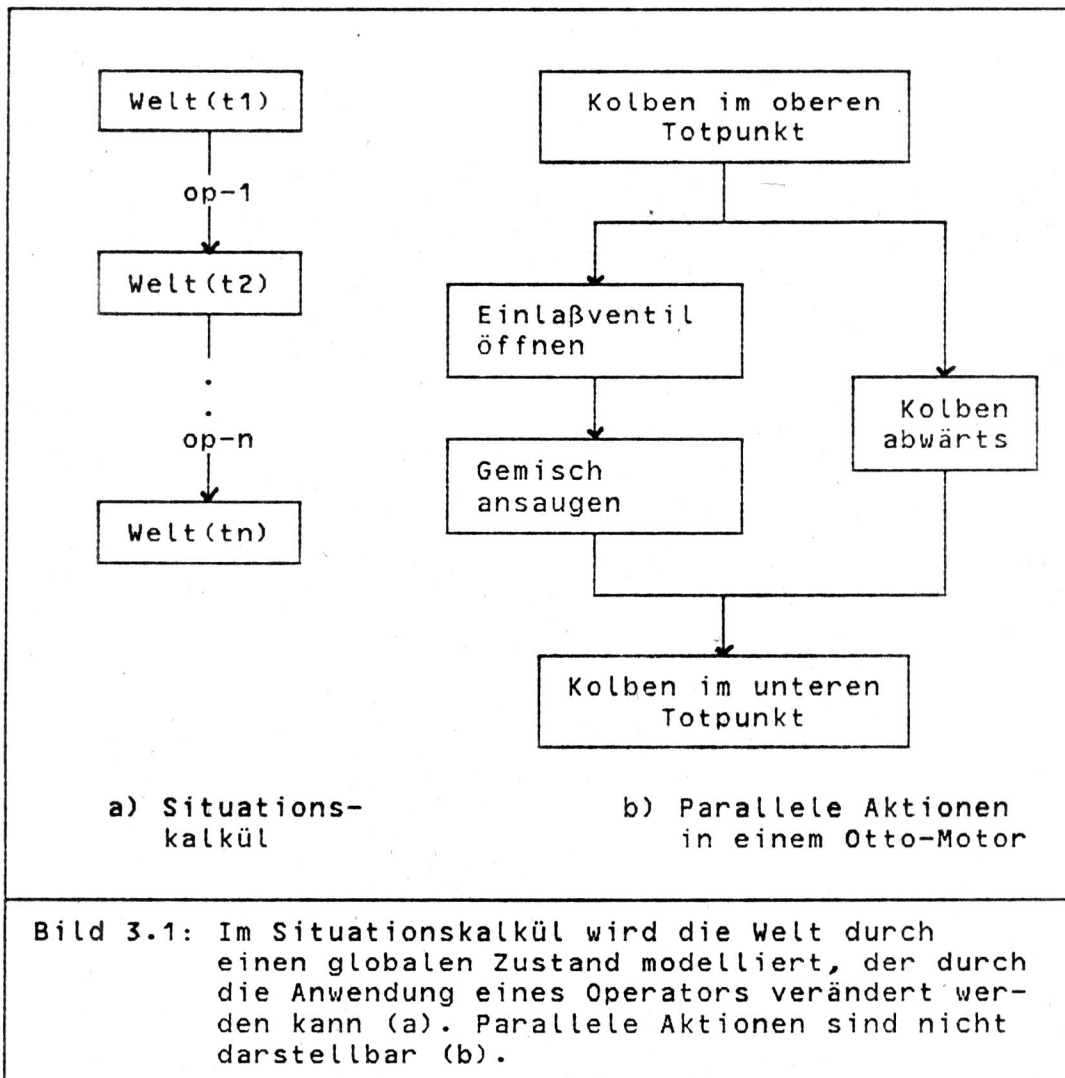
Bild 2.1: Historische Entwicklung von Modellierungen. Die Jahreszahlen geben (soweit bekannt) die ungefähre Zeit der Forschungsarbeit wieder, und liegen demnach früher als die später datierten Veröffentlichungen. Diese sind dem Literaturverzeichnis mit Jahreszahlen \geq den hier angegebenen zu entnehmen.

3. DER SITUATIONSKALKÜL

Der bereits 1958 von John McCarthy entwickelte [McCarthy 68] und noch 1969 gemeinsam mit Patrick J. Hayes [McCarthy/Hayes 69] vertretene Situationskalkül erklärt Abläufe in der Welt als eine Sequenz von Veränderungen eines **globalen** Zustands. Der globale Zustand wird modelliert als eine globale Datenbasis von Fakten; die Anwendung eines Operators kann die Menge der gültigen Aussagen verändern (Bild 3.1). Der Situationskalkül ist für die Beschreibung von technisch-physischen Systemen aus zwei wesentlichen Gründen nicht geeignet:

1. Es lassen sich keine parallelen Aktionen darstellen.
In einem technischen Gerät z.B. existieren meistens verschiedene, voneinander relativ unabhängige Pfade von zeitlichen und kausalen Verknüpfungen. Es wäre wünschenswert, diese netzwerkartigen Ablaufstrukturen adäquat darstellen zu können. Dies ist im Situationskalkül mit seinen streng sequenzialisierten Transformationen eines globalen Zustands nicht möglich.
2. Das bekannte **Frame-Problem** macht eine praktische Anwendung des Situationskalküls zur Modellierung realistischer Systeme unmöglich.
Praktisch ausgedrückt besagt das Frame-Problem folgendes: wie können die Effekte von Aktionen explizit so beschrieben werden, daß einerseits diese Beschreibung möglichst kurz ist, andererseits aber auch die Anwendung der Aktion keinen zu großen Aufwand bedeutet.
Die DELETE- und ADD-Listen von STRIPS-Operatoren zum Beispiel erlauben eine knappe Beschreibung der Effekte der Operatoren bezüglich einiger (primitiver) Prädikate. Die Wirkung auf nicht-primitive Prädikate muß zu gegebener Zeit vom System berechnet werden.

Dasselbe, was für die Beschreibung der Operator-Effekte gilt, trifft auch auf die Beschreibung der Vorbedingungen eines Operators zu. Hier besagt das sogenannte **Qualifikations-Problem**: wie kann einerseits die Anwendbarkeit eines Operators möglichst prägnant beschrieben werden, andererseits aber auch so vollständig sein, daß (möglichst) keine inkorrekte Anwendung erfolgt. In vielen praktischen Fällen wird sich eine vollständige Beschreibung nicht erzielen lassen, wenn es zu den allgemeinen Anwendbarkeitsregeln sehr viele nicht explizit aufzählbare Ausnahmeregelungen gibt. Diese Situation ist eine der Motivationen für Systeme zur Verwaltung von Abhängigkeitsnetzen und für nicht-monotone Logik. [Reinfrank 85a] gibt einen ausgezeichneten Überblick über dieses Gebiet und die zugrundeliegende Motivation. Auch das hier angedeutete Frame-Problem und das Qualifikations-Problem wird dort ausführlich diskutiert. Eine sehr gute Darstellung der aufgezeigten Problematik findet sich auch in [Charniak/McDermott 85].



4. DER COMMON SENSE ALGORITHMUS

Mit dem Konzept des Common Sense Algorithmus (Rieger/Grinberg 77) wird eine umfangreiche Liste von primitiven kausalen Beziehungen bereit gestellt. Mit diesen Relationen lassen sich Übergänge zwischen den konzeptuellen Zuständen des betrachteten Systems explizit formulieren. Das Ergebnis ist ein semantisches Netz von Zuständen, die über verschiedenartige kausale Verbindungen zusammenhängen. Auf dieser Darstellung können z.B. Simulationen des Systems durchgeführt werden.

Mit anderen Darstellungen, die auf ad hoc Beschreibungen mittels assoziativer Verknüpfungen beruhen, hat die CSA-Darstellung einen gemeinsamen, gravierenden Nachteil: Die Analyse der möglichen Verhaltensweisen wird nicht aus der Struktur des betrachteten Systems abgeleitet, sondern wird über die Zustandsübergänge explizit so eingegeben, wie der Benutzer bzw. der Experte das Verhalten versteht.

Eine solche Orientierung an der Struktur des betrachteten Systems ist das Hauptmerkmal aller heute aktuellen Modellierungsansätze. In den folgenden beiden Kapiteln werden die zwei wichtigsten Arten von Strukturbeschreibungen näher beleuchtet: objektorientierte und prozeßorientierte Strukturbeschreibungen.

5. OBJEKTORIENTIERTE STRUKTURBESCHREIBUNGEN

Objektorientierte Strukturbeschreibungen werden in allen in Bild 2.1 genannten Systemen verwendet, die direkt oder indirekt das EL System als Vorfahren haben. Die objektorientierten Repräsentationen von EL sind maßgeblich für die Techniken der Constraint-Propagierung, auf die wir in Kapitel 7 zur Darstellung des Verhaltens zurückkommen werden.

In objektorientierten - oder in Verbindung mit technischen Systemen besser: komponentenorientierten - Strukturbeschreibungen ist das **Objekt** die zentrale Einheit, von der Aktionen ausgehen und auf die Aktionen wirken. Für jede Zustandsänderung sind ein oder mehrere Objekte verantwortlich.

Objekte können miteinander verbunden werden, und über die Verbindungen kann ein Austausch von Informationen bzw. - je nach Sichtweise - von Materie stattfinden. Dieses **lokale Aufeinander-einwirken** von Objekten über fest vorgegebene Kanäle schränkt die Anzahl der möglichen Interaktionen sehr stark ein. Allein durch die Struktur des Systems sind den möglichen kausalen Beziehungen Grenzen gesetzt. Dieser lokale Wirkungsraum eines jeden Objekts ist somit ein Beitrag zur Lösung des Frame-Problems.

Eine wesentliche Unterscheidung zwischen den bestehenden Ansätzen besteht darin, ob die Verbindungen zwischen den Objekten gerichtet sind oder nicht. Eine solche Entscheidung hat gravierende Auswirkungen auf die in einer Verhaltensanalyse zu berücksichtigenden Wirkungen bzw. kausalen Zusammenhänge. So sind in EL, ENVISION [de Kleer/Brown 84] und ENV [Kuipers 84] alle Verbindungen **ungerichtet**. De Kleer und Brown argumentieren, daß in dem von Ihnen hauptsächlich betrachteten Bereich mechanischer Geräte die meisten Abläufe umkehrbar sind. Diese Aussage trifft im allgemeinen in dem Gebiet von elektronischen Bauteilen weniger oft zu. Das in [Williams 84] beschriebene und in enger

Anlehnung an ENVISION definierte System zur qualitativen Analyse von elektronischen Bauteilen erlaubt dann auch neben ungerichteten auch **gerichtete** Verbindungen für alle die Fälle, wo nur eine Richtung der Beeinflussung auftreten kann.

[Davis 84] und das HIQUAL System [Voss 86] gehen in dieser Beziehung noch weiter und ordnen jeder Verbindung eine Richtung zu. Falls Interaktionen in beiden Richtungen vorkommen sollen, müssen daher zwei Verbindungen - eine für jede Richtung - vorhanden sein. Dieses Vorgehen ist dann unproblematisch, wenn Verbindungen als rein konzeptuelle Einheiten betrachtet werden, und wenn ihnen keine Interpretation in Form von real existierenden Komponenten (Leitungen, Schläusche, mechanische Kupplungen usw.) unterlegt werden. Solche physikalisch existierenden Verbindungen müssen je nach Abstraktionsebene entweder explizit als eigene Objekte spezifiziert werden, oder als auf der aktuellen Ebene noch nicht betrachtete Komponenten den betrachteten Komponenten zugerechnet werden. Eine solche Zuordnung wiederum ist unproblematisch, wenn sie intuitiv eindeutig erfolgen kann. Andernfalls könnte der Zwang zu einer womöglich willkürlichen Auswahl eine gewisse Unnatürlichkeit der Darstellung hervorrufen.

6. PROZEBORIENTIERTE STRUKTURBESCHREIBUNGEN

6.1 ÜBERBLICK

[Forbus 84] weicht mit der "Qualitative Process Theory" (QPT) als einziger der bekannten Ansätze von der objektorientierten Strukturbeschreibung ab. Im Gegensatz zu objektorientierten Beschreibungen sind die aktiven, handelnden Einheiten seiner Modellierung nicht die Objekte, sondern ausschließlich die sogenannten **Prozesse**. Diese Aussage läßt sich sogar verschärfen: für jede beobachtete Wirkung bzw. Änderung ist unmittelbar ein Prozeß verantwortlich (Closed World Assumption).

In der Ontologie von QPT existieren auch Objekte. Diese sind aber rein passive Einheiten, die ausschließlich durch ihre bloße Existenz oder durch die konkreten Ausprägungen der ihnen zugeordneten Eigenschaften in Erscheinung treten. Die Existenz von bestimmten Objekten bzw. die Erfüllung spezifischer Bedingungen über den Eigenschaften der existierenden Objekte kann eine Voraussetzung für den Start oder das Ende von Prozessen sein. Andererseits kann die Aktivität eines Prozesses die Eigenschaften von Objekten verändern, Objekte erzeugen oder zerstören. Diese Veränderungen in der Objektwelt wiederum können die Menge der aktiven Prozesse verändern. können.

Beispiele und Bemerkungen:

1. Als Prozesse können auftreten: Kochen; Schmelzen; Korrodieren; Fluß von Wärme, Flüssigkeit o.ä.; ...
2. Beim Prozeß des Kochens wird auch Wärme an die Umgebung abgegeben. Der Prozeß des Kochens tritt also in Verbindung mit dem Prozeß des Wärmeflusses auf. Im Sinne eines modularen Aufbaus tut man als "Prozeßdesigner" jedoch gut daran, diese beiden Vorgänge als eigene Prozesse zu definieren und nicht zu einem Prozeß zu verschmelzen.
3. Der Prozeß des Schmelzens ist beendet, wenn das betreffende Objekt völlig geschmolzen ist, d.h. seinen Aggregatzustand vollständig von fest nach flüssig verändert hat. Es ist sicher eine nicht ganz einfache Design-Entscheidung, ob das betreffende Objekt damit seine Existenz verloren hat, oder ob es nur in dem neuen Zustand weiter existiert. Schmilzt ein Eiswürfel in der Nordsee, wird das kaum eine Nachwirkung haben. Das flüssige Speiseeis in der Einkaufstasche dagegen läßt seine Existenz wahrscheinlich nicht leugnen.

Festzuhalten bleibt jedoch, daß solche Veränderungen in der Objektwelt in QPT einfacher und natürlicher zu modellieren sind als in objektorientierten Strukturbeschreibungen, in denen die Objektstruktur für die meisten Systeme fest vorgegeben und unveränderbar ist. Lediglich von Davis wird eine Strategie vorgeschlagen, nach der zur Fehlersuche auch Veränderungen an der gegebenen Struktur in Betracht gezogen werden.

6.2 DIE ENTWICKLUNG DER PROZESSTHEORIE

Die Arbeiten von Forbus basieren wesentlich auf den Vorschlägen von Patrick J. Hayes zur Schaffung einer **naiven Theorie der Physik** [Hayes 78,85a]. Hayes selbst hat seine Vorschläge weiter konkretisiert und an dem schwierigen Problem des Verhaltens von Flüssigkeiten getestet [Hayes 85b]. Die Faszination, die von diesem Beispiel ausgeht, darf nicht darüber hinweg täuschen, daß der Weg von der gegebenen Beschreibung zu einer ökonomischen und von einem Programm unterstützbaren Analyse der erzeugten Beschreibungen noch weit ist. Hayes betont denn auch, daß es ihm in erster Linie auch gar nicht um effiziente Methoden geht, sondern vielmehr um das Aufzeigen der Probleme, der prinzipiellen Lösbarkeit und um gewisse, ihm wesentlich erscheinende Elemente der Modellierung.

Die wichtigste Einheit in dieser Modellierung ist die **Geschichte**. Eine Geschichte beschreibt die Entwicklung eines begrenzten, aber kontinuierlich veränderbaren Ausschnittes des Weltmodells während einer bestimmten ununterbrochenen Zeitspanne. Bild 6.1 illustriert den Begriff der Geschichte, wobei aus Gründen der

Darstellbarkeit der zweidimensionalen Raum gewählt wird.

Als konkretes Modell könnte man sich ein Zoomobjektiv vorstellen, durch das man eine Szene verfolgt. Die stetige Abfolge aller Raumausschnitte, die vom Auge durch das Objektiv wahrgenommen werden, hat die Gestalt einer Geschichte. Gibt es nun zwei Personen mit solchen Zoomobjektiven, von denen die eine einen Adler am Himmel, die andere eine Maus am Boden verfolgt, so können die beiden betrachteten Geschichten in dem Augenblick in Wechselwirkung treten, wo der Adler sich anschickt, die Maus zu ergreifen. Je nach Einstellung der Zooms werden sich während dieses Intervalls die Szenen überschneiden, oder eine Szene ist vollständig in der anderen enthalten, oder beide Szenen sind identisch. In der Abfolge können sich die beiden Geschichten wieder trennen, wenn das Unterfangen des Adlers mißlingt, oder gemeinsam fortfahren, wenn der Adler mit der Maus in den Fängen von dannen zieht.

Forbus hat den Begriff der Geschichte mit dem Begriff des Prozesses verallgemeinert. Ein Prozeß ist ein Schema, aus dem durch Aktivierung in Abhängigkeit von den Gegebenheiten der Umwelt bestimmte Manipulationen in der für den Prozess definierten Umgebung ausgeführt werden können. Jede konkrete Ausführung ist eine Geschichte im Sinne von Hayes.

Tatsächlich kennt QPT den Begriff der Geschichte nicht nur zur Beschreibung der Semantik von Prozessen, sondern auch als explizites Konzept der Sprache in Form einer **eingeschlossenen Geschichte** (encapsulated history). Mit eingeschlossenen Geschichten beschreibt Forbus solche Abläufe, die sich nur schwer durch das doch relativ starre Muster eines Prozesses abbilden lassen. Zum Beispiel wird eine Explosion als eine eingeschlossene Geschichte modelliert, die zur Zerstörung der situationsgemäß betroffenen Objekte führt. Die Schwierigkeit besteht hier darin, den exakten Wirkungsraum des zur Explosion führenden Ereignisses anzugeben. Zur genauen Festlegung dieses Raumes ist ein im allgemeinen sehr umfangreiches Wissen notwendig. Z.B. müssen die genauen Daten zur Feststellung der Explosionsstärke vorhanden sein, und für jedes Objekt muß bekannt sein, ob es von der Druckwelle bzw. durch herumfliegende Teile erreichbar ist. Wenn ja, dann stellt sich weiter die Frage, ob die Druckwelle oder andere Teile zur Zerstörung bzw. signifikanten Beschädigung ausreichen. Dies ist eventuell abhängig von der Lage oder von der momentanen Unterstützung des Objekts, usw.

Sieht man einmal von solchen Katastrophenfällen ab, die in jeder Modellierung schwer zu erfassen sind, dann wird der Wirkungsraum eines Prozesses durch eine für den Prozeß charakteristische Menge von beteiligten Objekten bzw. einem vorgegebenen Raumausschnitt begrenzt. [Forbus 84] enthält ausführliche Betrachtungen zu diesem von uns bereits erwähnten Beitrag zur Lösung des Frame-Problems. Interessant ist insbesondere im Zusammenhang mit Geschichten und Prozessen die Analyse von Forbus, daß durch den lokalen Charakter der Prozesse das Frame-Problem substituiert wird durch zwei - hoffentlich - einfachere Probleme:

-
- **das Problem der lokalen Entwicklung** (von Prozessen)
[local evolution problem]
 - **das Problem der Wechselwirkungen** (zwischen Prozessen)
[intersection/interaction problem]

Das erstgenannte Problem bezieht sich auf die geschickte Auswahl der Prozesse als möglichst unabhängige Vorgänge mit wenigen Wechselwirkungen. Man könnte dies auch als ein Bestreben um eine möglichst hohe Modularität des gewählten Systems von Prozessen bezeichnen. Natürlich kann ein interessantes System von Prozessen nicht völlig ohne Wechselwirkungen auskommen.

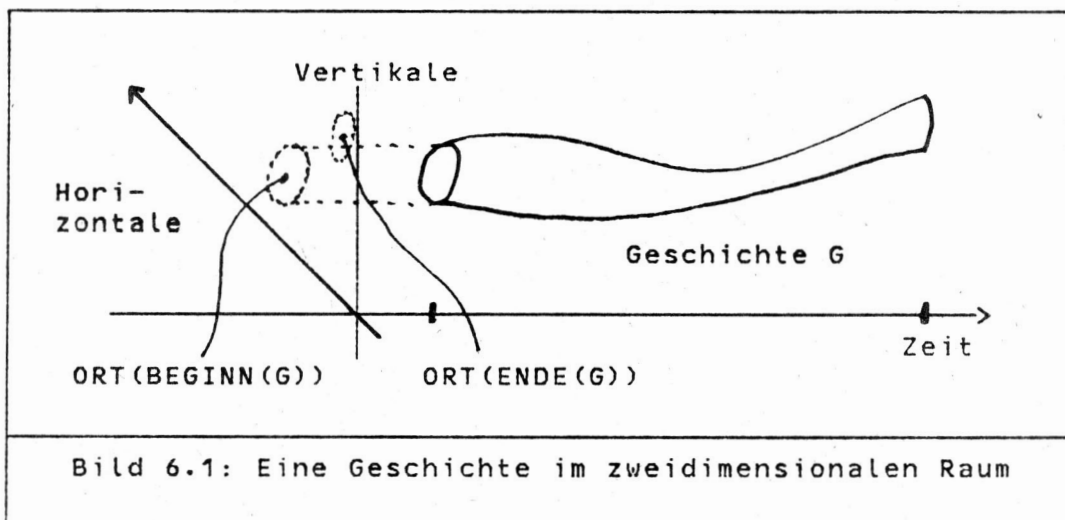
Das zweite Problem erfordert Methoden

1. Zum Feststellen, ob eine Wechselwirkung potentiell stattfinden könnte; hierzu muß die zeitliche und räumliche Interferenz zweier Prozesse festgestellt werden.
2. Zur Prüfung, ob bei Vorliegen einer potentiellen Wechselwirkung eine solche auch tatsächlich stattfindet. Wenn sie stattfindet, ist sie dann auch qualitativ signifikant?

Das Beispiel der Explosion hat die Begriffe bereits teilweise veranschaulicht. Ein weiteres Beispiel soll insbesondere die Schwierigkeit des zweiten Problems aufzeigen.

Beispiel: Ein Ball trifft auf die Oberfläche eines Tennisschlägers auf, der vom Spieler gegen die Flugrichtung des Balles geführt wird. Diese einfache Situation enthält eine Vielzahl von Parametern, die gemeinsam den Ausgang dieses Ereignisses bestimmen.

Die Beschreibung der Prozeßtheorie kann in dieser Arbeit aus Zeitgründen leider nicht weiter vertieft werden. Dies ist um so mehr zu bedauern, weil eine eingehendere Analyse mit Sicherheit sehr viele Gemeinsamkeiten zwischen den objektorientierten und den prozeßorientierten Beschreibungen herausstellen würde.



7. VERHALTENSBESCHREIBUNG UND IMPLEMENTIERUNG MIT CONSTRAINTS

In objektorientierten Strukturbeschreibungen für Modelle technischer Systeme werden die möglichen Verhaltensweisen eines Objekts meistens durch eine Menge von Relationen beschrieben. Jede solche Relation ist eine Beziehung zwischen den Werten von Eigenschaften, die den Objekten zugeordnet sind, und von Werten, die an den Verbindungen des Objekts "anliegen" können. Werden bestimmte Werte initial vorgegeben, so kann die Analyse der möglichen Interaktionen zwischen den Objekten durch sukzessive Berechnung und anschließende Weitergabe von Einschränkungen für bisher unbekannte Ausprägungen von Eigenschaften erfolgen. Ein Verfahren, das nach diesem Prinzip arbeitet, wird als **Constraint-Propagierung** bezeichnet. Obwohl es genau genommen die neuen Erkenntnisse über die Werte von Variablen sind, die (als neue Constraints) propagiert werden, werden meistens die Relationen als statische Einschränkungen über möglichen Wertekombinationen der Variablen als Constraints bezeichnet. Diese Relationen sind auch der Ausgangspunkt für die jeweiligen Einschränkungen der Variablenwerte.

Mit dem für technische Modellierungen und für die Constraint-Propagierung allgemein sehr einflußreichen EL-System stellten Stallman und Sussman eine Methode zur Darstellung und Analyse von elektronischen Schaltkreisen vor, die auf dem Constraint-Propagierungsverfahren beruhte. Alle Systeme, die EL als direkten oder indirekten Vorgänger in dem Graphen aus Bild 2.1 haben, verwenden an zentralen Punkten ihrer Analyse ein Constraint-Propagierungsverfahren. Dasselbe gilt für die Analyse in QPT, wo

nach dem Modell der Constraint-Verfahren die Prozesse als Objekte fungieren, und die Wechselwirkungen zwischen den Prozessen durch Constraint Propagierungstechniken ermittelt werden.

Die Schwierigkeiten im Propagierungsverfahren hängen von den Eigenschaften der Relationen und von der Vollständigkeit der zur Verfügung stehenden Information ab. In den folgenden Abschnitten werden wir einige verschiedene Typen von Relationen und die damit verbundenen Propagierungstechniken betrachten. Einen kurzen Überblick über die aufgezeigten Unterschiede gibt auch [Reinfrank 85b].

7.1 EINFACHE RELATIONEN

Im einfachsten Fall gibt es in jedem Propagierungsschritt mindestens eine Relation, die bis auf eine beteiligte Variable die Werte aller anderen beteiligten Variablen kennt. Eine Relation heißt **einfach**, wenn in jeder solchen Situation der Wert der unbekannt Variablen aus der Definition der Relation eindeutig ableitbar ist.

Allgemeiner heißt eine Relation **anwendbar**, wenn höchstens eine Variable einen unbekannt Wert besitzt, oder wenn sich der Wert einer unbekannt Variablen eindeutig aus den bekannten Werten ergibt. Sind für eine Relation alle Werte bekannt, so wird überprüft, ob das hiermit gegebene Wertetupel in der Relation enthalten ist. Wenn das nicht der Fall ist, so existiert für die vorgegebene Menge von Anfangswerten keine Ergänzung für die unbekannt Werte, mit denen alle Relationen gleichzeitig erfüllt werden können.

Beispiel: PLUS := { (x,y,z) | x+y=z, (x,y,z) ∈ ℝ³ }.

PRODUKT := { (x,y,z) | x*y=z, (x,y,z) ∈ ℝ³ }

Aus der Kenntnis von zwei Werten läßt sich der dritte Wert eindeutig ableiten.

Spezialfälle für zwei unbekannt Variablen:

1. x*y=z mit x=0 ⇒ z=0

2. x*y=z mit z=1 ⇒ x=1 ∧ y=1

Soll PLUS z.B. ein ADDER-Glied in einem digitalen Schaltkreis modellieren, so könnte man zumindest für Simulationszwecke die Anwendung der Relation auf den Fall beschränken, wo sowohl x als auch y bekannt sind. "Ein ADDER-Glied kann nicht subtrahieren." Für Inferenzzwecke - z.B. wie muß der Wert von x gewesen sein, wenn y=6 und z=10 gesichert sind? - könnte man auch die andere Richtung zulassen. Dies führt auf die bereits begonnene Diskussion über die Gerichtetheit von Verbindungen zurück. Hier und in den folgenden Abschnitten wollen wir davon ausgehen, daß alle

Verbindungen ungerichtet sind. Eine grundsätzlichere Diskussion über diesen Punkt werden wir in Kapitel 8 über kausale Beziehungen wieder aufnehmen.

7.2 WEITGEHEND EINFACHE RELATIONEN

In praktisch relevanten Problemen treten stets auch Relationen auf, die die Eigenschaft der Einfachheit nicht oder nur unvollständig besitzen.

Beispiel: Die Systeme ENVISION, ENV [Kuipers 84], und [Williams 84] abstrahieren die reellen Zahlen auf die Menge VZ (Vorzeichen) der drei qualitativen Werte

$$VZ := \{+, -, 0\}$$

Die in den Modellierungen betrachteten physikalischen Größen sind Variablen, die stets Werte aus VZ annehmen. Relationen werden als Gleichungen über Variablen und Konstanten in VZ notiert. Jede Variable ist eindeutig einem Objekt zugeordnet. Eine Relation darf dabei nur Variablen des eigenen Objekts und derjenigen Objekte benutzen, mit denen das eigene Objekt verbunden ist. In ENV gibt es gegenüber ENVISION die Besonderheit, daß die Objekte mit den Variablen identifiziert werden, ein wesentliches Strukturierungsmerkmal also fehlt.

Zur Interpretation der Gleichungen, d.h. zur Berechnung von bisher unbekannt Variablen, müssen Rechenoperationen auf VZ definiert werden:

		y →		
	+	0	+	-
x	0	0	+	-
↓	+	+	+	? ₁
	-	-	? ₂	-

		y →		
	*	0	+	-
x	0	0	0	0
↓	+	0	+	-
	-	0	-	+

Hier gibt es bei der Operation + zwei Fälle ?₁ und ?₂, wo die Kenntnis der Werte von x und y nicht ausreicht, um z gemäß z=x+y zu bestimmen. Dies könnte nur in einer Verfeinerung der Werte geschehen, in der man die "tatsächlichen" reellen Werte r.x und r.y von x und y berücksichtigen kann, z.B.:

```
?1 := if r.x > r.y then '+'
      else if r.x = r.y then '0'
      else '-'
```


In diesem oder ähnlich gelagerten Fällen kommt es entscheidend darauf an, wieviele Werte für die bisher unbekannte Eigenschaft (Variable) abgeleitet werden können. Handelt es sich nur um einige wenige Werte (in dem Beispiel die Werte +, - und 0 für ?), so kann man versuchen, mit einer beliebigen Auswahl zum Erfolg zu kommen. Eine Relation, die nur an relativ wenigen Stellen nicht einfach ist, und an den nicht einfachen Stellen nur wenige Werte zuläßt, wollen wir als **weitgehend einfach** bezeichnen (almost simple, vgl. [Reinfrank 85b]).

Nach einer oder mehrerer Auswahlen kann sich später herausstellen, daß die Menge der insgesamt getätigten Auswahlen **inkonsistent** ist, d.h. mit den gewählten Werten werden nicht mehr alle Relationen erfüllt. Eine solche Konsistenzprüfung findet auf jeden Fall immer dann statt, wenn eine Relation angewendet wird, die keine unbekanntes Variablen mehr enthält. Ist die hiermit definierte Wertekombination aller Variablen nicht in der Relation enthalten, so muß für eine der Variablen mit einem gewählten Wert eine andere Wahl getroffen werden, falls es noch eine Alternative gibt. Dieser Vorgang kann durch einfaches Backtracking realisiert werden, oder wie in fast allen hier betrachteten Systemen durch ein abhängigkeitsgesteuertes Backtracking (**dependency directed backtracking**, vgl. [Stallman/Sussman 77]).

Beispiel: Betrachte die Lösungen des folgenden Gleichungssystems in VZ, wenn $x=+$ und $y=-$ vorgegeben werden:

$$(1) \ x+y=z \quad (2) \ x+u=w \quad (3) \ y*w=z \quad (4) \ x*z-w=0$$

Zur Lösung wird angenommen, daß anwendbare Gleichungen in der angegebenen Reihenfolge überprüft werden.

Zu Beginn ist die einzige Gleichung mit genau einer Variablen mit unbekanntem Wert (1). Der Wert von z muß gewählt werden, da $x+y = (+)+(-)$ nicht eindeutig ist. Ein möglicher Ablauf ist der folgende:

1. Annahme: $z=(-)$ (1)
2. $w=(+)$ (3)
3. Annahme: $u=(-)$ (2)
4. Widerspruch: $x*z-w=0$; backtracking
5. Annahme: $u=(+)$ (2)
6. Widerspruch: (4); backtracking
7. Annahme: $u=(0)$ (2)
8. Widerspruch: (4); backtracking
9. Annahme: $z=(+)$ (1)
10. $w=(-)$ (3)
11. Annahme: $u=(-)$ (2)
12. Widerspruch: (4); backtracking

13. Annahme: $u=(+)$ (2)
 14. Widerspruch: (4); backtracking
15. Annahme: $u=(0)$ (2)
 16. Widerspruch: (4); backtracking
17. Annahme: $z=(0)$ (1)
 18. $w=(0)$ (3)
19. Annahme: $u=(-)$ (2)
 20. Gleichung (4) ist erfüllt.
 Keine weitere Gleichung anwendbar.

Zu diesem Beispiel sind die folgenden Bemerkungen zu machen:

- Der "menschliche Experte" erkennt sofort, daß die jeweils letzten beiden Annahmen über den Wert von u (Zeilen 6,8,14,16) überflüssig sind, da die Erfüllbarkeit der Gleichung (4) völlig unabhängig von der Wahl von u ist. Ein abhängigkeitsgesteuertes Backtracking würde hier erkennen, daß im Falle einer Inkonsistenz in $x*z-w=0$ nur eine andere Wahl von z eine Änderung bewirken kann. Die Ausführung würde dann sofort von Zeile 4 zur Zeile 9 und von Zeile 12 zur Zeile 17 verzweigen.

Ein Vorziehen der Überprüfung von (4) vor die Auswahl von u würde die Ineffizienz auch beseitigen. Dies ist jedoch keine Argumentation für den allgemeinen Fall.

- Man kann sich mit dem Auffinden einer einzigen Lösung zufrieden geben (Ende mit Schritt 20.), oder aber nach weiteren Lösungen suchen. Alle Lösungen erhält man, wenn man das backtracking solange fortsetzt, bis keine Alternativen für andere Auswahlen mehr vorhanden sind. Beachte, daß ein abhängigkeitsgesteuertes Backtracking von Schritt 20 ausgehend nicht sofort stoppen darf (keine weitere Alternative für z), sondern erkennen muß, daß außer der bisherigen Lösung noch zwei weitere existieren, die sich insgesamt nur durch die Wahl von u voneinander unterscheiden:

	1	2	3
x	+	+	+
y	-	-	-
z	0	0	0
u	+	-	0
w	0	0	0

7.3 CONSTRAINT-PROPAGIERUNG MIT SYMBOLISCHEN AUSDRÜCKEN

Die bisherige Voraussetzung war, daß zur Anwendung eines Constraints von n Variablen mindestens $n-1$ Variablen einen eindeutigen Wert hatten. Wenn diese Voraussetzung nicht erfüllt ist, d.h. es gibt nur $n-2$ Variablen mit bekanntem Wert und keine der unbekannt Variablen ist ableitbar, dann könnte man sich zumindest zwei mögliche Fortsetzungsstrategien vorstellen. Zu ihrer Darstellung nehmen wir an, daß v_{n-1} und v_n die einzigen unbekannt Variablen sind.

1. Man verfährt analog zu dem im letzten Abschnitt besprochenen Verfahren. Es wird also eine Annahme über den Wert von v_{n-1} gemacht und anschließend eine Annahme über den Wert von v_n . Damit steigt natürlich die Anzahl der möglichen Annahmen auf das Produkt aus der Anzahl der möglichen Werte für v_{n-1} und den Werten für v_n . Dieses Produkt darf also nicht sehr groß sein.
2. Man entwickelt einen symbolischen Ausdruck, der eine der unbekannt Variablen - hier v_{n-1} - in Abhängigkeit von der verbleibenden Variablen - hier v_n - und den bekannten Werten darstellt. An Stelle von v_n wird implementatorisch ein neuer symbolischer Wert, z.B. $U-v_n$, erzeugt und in den Ausdruck eingesetzt. Dieser Ausdruck wird propagiert, wobei die Hoffnung darin besteht, daß irgendwann eine sogenannte Koinzidenz (coincidence, [Stallman/Sussman 77]) auftritt.

Eine Koinzidenz ist eine Situation, in der einer Variablen ein symbolischer oder nicht-symbolischer Wert zugewiesen wird, obwohl die Variable bereits einen Wert hatte (symbolisch oder nicht-symbolisch).

Wenn der neue Wert oder der alte Wert nicht-symbolisch ist, also mindestens eine Variable enthält, so kann man beide Werte gleichsetzen und nach einer der Variablen auflösen. Enthält die Gleichung nur eine Variable, dann sollte man den für die Variable durch Lösen der Gleichung berechneten Wert bei Eindeutigkeit in allen symbolischen Ausdrücken substituieren und die Ausdrücke gegebenenfalls vereinfachen.

Das erste Verfahren ist indiskutabel bei zu vielen möglichen Wertekombinationen. In dem zweiten Verfahren wird die durch einen Constraint gegebene Information in der Bildung des symbolischen Ausdrucks optimal ausgenutzt, und es ist kein backtracking notwendig. Allerdings wird dies durch einen i.a. sehr hohen und im konkreten Fall eventuell nicht zu bewältigenden Aufwand zur Manipulation symbolischer Ausdrücke erkauft. Im Gegensatz zu EL ist die Propagierung symbolischer Ausdrücke in den meisten der späteren Systeme (noch) nicht möglich, und es wird teilweise das 1. Verfahren verwendet (ENVISION). Wir wollen nun die Anwendung des zweiten Verfahrens an einem Beispiel demonstrieren.

Beispiel: Wir gehen von folgenden Gleichungen aus, die

zusätzlich in Bild 7.1 mittels eines sogenannten Constraint-Netzwerks illustriert sind. Das Beispiel wurde von Steele benutzt, um die Grenzen seines CONSTRAINTS Systems aufzuzeigen [Steele 80].

$$(1) v_2 = v_1 + v_4$$

$$(2) v_3 = v_2 + v_4$$

Zwischen den drei Variablen v_1 , v_2 und v_3 über den reellen Zahlen gilt die Beziehung $v_2 - v_1 = v_3 - v_2$, d.h. v_2 hat den gleichen Abstand zu v_1 und v_3 . Werden z.B. für v_1 und v_2 Werte vorgegeben, so lassen sich die Werte von v_3 und v_4 durch einfache Propagierung berechnen. Werden jedoch nur die Werte von v_1 und v_3 vorgegeben, so ist jedem der 3-stelligen Constraints nur eine Größe bekannt. Wir werden das Beispiel konkretisieren und mit den zwei initialen Werten $v_1 = 2$ und $v_3 = 6$ fortfahren. Es ergeben sich z.B. die folgenden Schritte:

1. Ersetze v_4 durch den symbolischen Wert $U - v_4$.
 $v_1 + U - v_4 = 2 + U - v_4$ ist der neue Wert von v_2 .
2. Gleichung (2) ergibt nach Einsetzung der bekannten Werte
 $6 = v_2 + U - v_4$
Damit kann auch v_2 aufgelöst werden zu $v_2 = 6 - U - v_4$.
Nun ist eine Koinzidenz aufgetreten, denn für v_2 war bereits $2 + U - v_4$ berechnet worden. Also:
 $2 + U - v_4 = 6 - U - v_4 \Leftrightarrow U - v_4 = 2$
3. Substitution der Lösung für $U - v_4$ in einem Ausdruck für v_2 ergibt $v_2 = 4$.

7.4 DESTRUKTIVE CONSTRAINT PROPAGIERUNG

Alle bisher diskutierten Propagierungstechniken werden als konstruktiv bezeichnet, da aufgrund der vorhandenen Informationen neue Werte berechnet werden. Wenn die Anzahl der für eine Auswahl in Frage kommenden Werte bei nur weitgehend einfachen Constraints für praktische Zwecke zu groß oder gar unendlich ist, kann evtl. eine als destruktiv bezeichnete Propagierungstechnik die auszuwählenden Werte einschränken. Möglicherweise könnte es sogar gelingen, die Werte bis auf ein paar wenige so weit einzugrenzen, daß man wieder mit einer konstruktiven Propagierungstechnik und Annahmen über die verbleibenden Werte fortfahren kann. Prinzipiell ist also ein geschicktes Zusammenwirken beider Techniken denkbar. Wir kennen jedoch kein System, das hiervon bisher Gebrauch gemacht hat.

Es ist ein anscheinend weit verbreiteter Irrtum, daß die Variablen für konstruktive Propagierungsalgorithmen höchstens einen einzigen Wert annehmen können. In den Beispielen und auch in den Constraint-Anwendungen in [AI Journal 84] werden zwar immer nur einelementige Wertemengen betrachtet. Die in HIQUAL [Voss 85] verwendete Zeitanalyse jedoch basiert auf einem Propagierungsverfahren von Allen [Allen 83] und ist ein Beispiel für ein kon-

struktives Propagierungsverfahren, in dem die Belegung von Variablen zu jedem Zeitpunkt eine Menge von möglichen Werten ist.

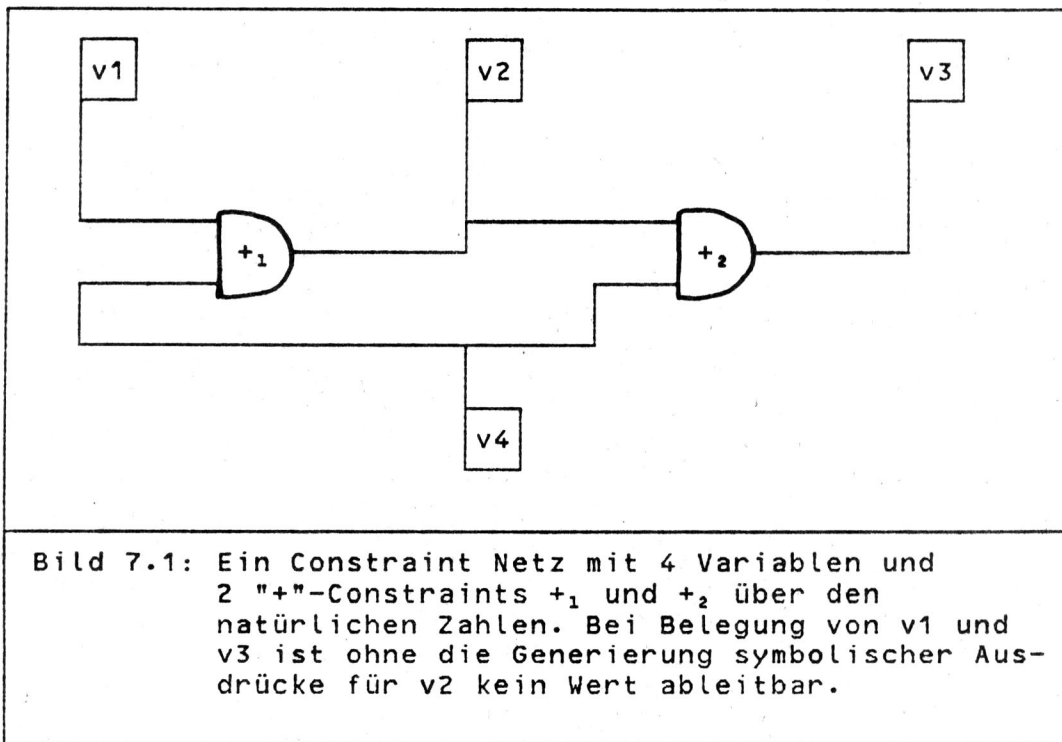
Diese Erweiterung des Verfahrens auf Mengen wird klarer, wenn man die Unkenntnis über den Wert einer Variablen v so darstellt, daß v jeden Wert w ihres Wertebereichs W annehmen kann, also etwa $werte(v) = W$, oder deutlicher

$$werte(v) = \bigvee_{w \in W} v = w$$

Die genaueste Kenntnis von v als einelementige Wertemenge ist dann nur ein Spezialfall dieser Darstellung, und es sind beliebige Zwischenlösungen $werte(v) = tw \subseteq W$ denkbar. Die generelle Arbeitsweise eines Constraint-Propagierungsverfahrens besteht somit darin, ausgehend von einer initialen Belegung der Variablen mit den maximal möglichen Werten die Information über die Constraints so auszunutzen, daß die einzelnen Belegungen minimiert werden. In der konkreten Form dieser Ausnutzung von Informationen unterscheidet sich die destruktive Propagierung (dP) von der konstruktiven Propagierung (kP).

- In kP ist die Ableitung einer neuen Wertemenge konstruktiv, weil sie aus der Analyse eines Constraints aufgrund des **Vorhandenseins** von Werten der am Constraint beteiligten Variablen erfolgt. Es wird jeweils die aufgrund der lokalen Information neu berechnete Wertemenge propagiert.
- In dP ist die Ableitung einer neuen Wertemenge destruktiv, weil sie aufgrund der **Abwesenheit** von Werten der am Constraint beteiligten Variablen erfolgt. Es wird jeweils die Information propagiert, daß bestimmte Werte für eine Variable nicht in Frage kommen.

Ein bekanntes Beispiel für eine destruktive Propagierung ist der Waltz-Algorithmus [Waltz 75] zur Erkennung von Szenen in der Blocks-World. Eine ausführliche Beschreibung dieses Verfahrens und einer asynchron parallelen Version hiervon findet man in [Reinfrank 85]. Eine Analyse der verschiedenen Constraint-Verfahren, insbesondere auch der Beziehungen zwischen konstruktiven und destruktiven Propagierungsverfahren, ist Gegenstand eines künftigen Berichts [Voss 86b].



8. KAUSALE BEZIEHUNGEN

Im Zusammenhang mit Strukturbeschreibungen wurde bereits einiges über kausale Beziehungen gesagt. Die zwei entscheidenden Punkte aus den Kapiteln 5 und 6 sollen hier kurz wiederholt werden:

1. Die Strukturbeschreibung legt die prinzipiell möglichen Pfade kausaler Wechselwirkungen bereits statisch fest.
2. Modellierungen unterscheiden sich wesentlich dadurch, ob man Verbindungen zwischen Objekten (auch) gerichtet einführen kann oder nicht.

Der zweite Punkt soll hier etwas vertieft werden. Es gibt nämlich eine angeregte Diskussion darüber, ob und wie weit die Beschränkung auf das ursprüngliche Constraint-Modell mit ausschließlich ungerichteten Verbindungen die tatsächlich möglichen kausalen Abläufe adäquat erfasst. Konkret ist die Frage, ob nicht durch das Modell Verhaltensweisen ermöglicht werden, die keine physikalische Entsprechung haben oder dem menschlichen Kausalverständnis widersprechen, also auch keine akzeptable Erklärung für ein Verhalten sein können.

Die Beschreibung physikalischer Zusammenhänge bzw. Abläufe geschieht üblicherweise mittels Gleichungen über physikalischen Größen. Eine Gleichung kann jeweils so manipuliert werden, daß sich jede Variable in Abhängigkeit von den übrigen darstellen läßt. Dies entspricht ganz dem theoretischen Modell der Constraint-Propagierung. Dennoch spricht man z.B. von funktionalen Abhängigkeiten zwischen Variablen oder unterscheidet zwischen unabhängigen und abhängigen Variablen eines physikalischen Modells. Abhängige Variablen sind dabei solche, die man nicht direkt, sondern nur über den Umweg über unabhängige Variablen indirekt verändern kann. [Forbus 84] gibt dazu das folgende Beispiel:

Beispiel: (a) Wärme(Substanz) = Menge(Substanz)
* Temperatur(Substanz)

(b) Menge(Natrium,Lösung) =
Konzentration(Natrium,Lösung)
* Menge(Lösung)

Die erste Gleichung (Constraint) besagt, daß die in einer Substanz enthaltene Wärme proportional dem Produkt ihrer Temperatur und ihrer Menge ist. Der zweite Constraint formuliert die strukturell analoge Situation für den Anteil von Natrium in einer Lösung.

Für (a) wäre die folgende Aussage ein korrekter kausaler Schluß:

"Die erhöhte Wärme verursacht einen Anstieg der Temperatur."

Ein inkorrektter Schluß wäre dagegen:

"Die erhöhte Wärme verursacht einen Anstieg der Menge der Substanz."

Noch deutlicher und bei Verwendung in einem medizinischen Expertensystem möglicherweise noch tragischer wäre die kausal falsche Interpretation in (b):

"Eine Erhöhung des Natriumgehalts in einer Lösung führt zu einem Anstieg der Menge der Lösung."

Die Natriumkonzentration als der entscheidende Faktor für die korrekte Funktion der Niere beispielsweise würde sich nach dieser Interpretation nicht verändern.

In Forbus' QPT und den meisten anderen Systemen werden deshalb nicht Constraints im strengen Sinne, sondern i.a. eingeschränkte Constraints betrachtet, die nur nach bestimmten Richtungen aufgelöst werden können. Forbus unterscheidet z.B. zwischen den stets gerichteten Wirkungen (influences) und den Proportionalitäten (proportionalities). Eine Wirkung beschreibt den direkten Effekt eines Prozesses; eine so beeinflusste Größe ist sozusagen eine unabhängige Variable des Systems. In Proportionalitäten werden dagegen Einflüsse auf bzw. zwischen abhängigen Größen erfaßt.

Davis verwendet **Simulationsregeln** (simulation rules) zur Darstellung von kausalen Abhängigkeiten und **Ableitungsregeln** (inference rules) zur Herleitung von Inputs einer Komponente bei gegebenen Outputs und evtl. anderen Inputs. Die Ausführung eines Modells darf ausschließlich Simulationsregeln verwenden.

De Kleer's Beschreibungen in ENVISION beruhen ausschließlich auf ungerichteten Constraints. Ein Teil der in [Iwasaki/Simon 85] formulierten Kritik an diesem System fußt letztendlich auf der Tatsache, daß mit den Beschreibungen von ENVISION auch Erklärungen generiert werden können, die nicht der intuitiven menschlichen Erfahrungswelt entsprechen.

Abschließend möchten wir betonen, daß die Einschränkung von Constraints auf bestimmte Richtungen kein technisches, sondern ein rein konzeptionelles Problem ist.

9. ZEITLICHE BEZIEHUNGEN

Kausale und zeitliche Beziehungen sind miteinander durch die folgende Aussage verknüpft:

Wenn B durch A verursacht wird, dann darf B nicht vor A beginnen.

Dies ist die allgemeinste Form, in der man den Zusammenhang formulieren kann. Es wäre z.B. eine starke Einschränkung der möglichen Beziehungen, wenn man nur die folgende Festlegung träfe:

Wenn A durch B verursacht wird, dann finden A und B gleichzeitig statt oder B findet nach A statt.

Diese Formulierung hat nur dann Gültigkeit, wenn A und B punktförmige Ereignisse mit der Dauer null sind. Punktförmige Ereignisse sind wichtig, um relevante instabile Zustände beschreiben zu können. Der Vorgang des Gehens beim Menschen besteht z.B. aus einer stetigen Folge von ausschließlich instabilen Gleichgewichtszuständen. Der Durchgang eines Kolbens durch einen Totpunkt ist ein Beispiel für einen signifikanten Zeitpunkt in der Beschreibung eines Motors.

Im allgemeinen sollte man jedoch davon ausgehen, daß sowohl Ursachen als auch Wirkungen einer Kausalbeziehung echte Zeitintervalle mit einer positiven Zeitdauer zugeordnet werden müssen. Zwischen zwei Zeitintervallen können viele mögliche zeitliche Beziehungen existieren. Eine in gewissem qualitativen Sinn vollständige Menge von Zeitrelationen wird z.B. in [Allen 83] definiert (Bild 9.1). Man könnte diese Menge von Relationen noch erweitern um qualitative und auch quantitative

Beschreibungen der Intervalllängen und der Längen zwischen den Intervallen. Zum Beispiel könnte man die Relation "A < B" verfeinern zu "A R< B" mit:

$R< := \{\text{sehr-weit-vor, weit-vor, vor, kurz-vor, sehr-kurz-vor}\}$

Allen selbst beschreibt eine Methode, seine Relationen um quantitative Aussagen über Intervalllängen kanonisch zu erweitern.

Relation	Symbol	Inverse	Graphik
X before Y	<	>	XXX YYY
X equal Y	=	=	XXX YYY
X meets Y	m	mi	XXXYYY
X overlaps Y	o	oi	XXX YYY
X during Y	d	di	XXX YYYYY
X starts Y	s	si	XXX YYYYY
X finishes Y	f	fi	XXX YYYYY
Bild 9.1: Die Zeitrelationen von Allen.			

In den hier betrachteten Systemen haben zeitliche Betrachtungen in dieser Vielfalt der Darstellungsformen bisher kaum Eingang gefunden. Ernsthaftige Ansätze gibt es bei Forbus zumindest auf der Ebene der Semantikbeschreibung von QPT. In HIQUAL können Zeitrelationen von Allen auf der Syntaxebene zur Beschreibung zeitlicher Beziehungen zwischen Vorbedingungen (Ursachen) und Nachbedingungen (Wirkungen) in Regeln, und zwischen diesen und den inneren Zuständen von Objekten (s. Kapitel 10) explizit spezifiziert werden. Die Beschreibung der Semantik besteht in der Analyse der Zeitrelationen zwischen den möglichen Ereignissen und den Zuständen, wobei ohne explizite Angabe in einer Objektdefinition von den allgemeinsten Standardbeziehungen ausgegangen wird. Die explizit angegebenen Relationen brauchen dann nur als zusätzliche Einschränkungen den Analyseinformationen hinzugefügt werden.

Die Hardware-Diagnostiksysteme von Davis und von Genesereth [Genesereth 84] erlauben lediglich eine digitale und synchrone globale Uhr, gemäß der alle Ereignisse mit einem Zeitpunkt markiert werden können. Es ist klar, daß es von dieser Seite

starke Bemühungen geben muß, auch schwierigere Zusammenhänge für zeitlich komplexere Geräte darstellen zu können.

Über diese beiden Systeme hinausgehend ist die von ENVISION durchgeführte Analyse in der Lage, zwischen instabilen globalen Zuständen, die als Zeitpunkte interpretiert werden, und zwischen stabilen globalen Zuständen zu unterscheiden. Instabile Zustände sind daran zu erkennen, daß eine qualitative Variable über VZ einen 0-Durchgang ausführt.

Wichtig ist dabei, daß auch ENVISION noch an einem globalen Zustand festhält, der in einer globalen Sicht für jedes einzelne Objekt seinen aktuellen inneren Zustand aufführt. Ein Ablauf des Systems ist eine Folge solcher globaler Zustände. Diese Darstellungsform des Analyseergebnisses hat wie der Situationskalkül den wesentlichen Nachteil, daß voneinander unabhängige, asynchron parallele Zweige von Aktivitäten nicht adäquat erkannt werden können. Der Vorteil, den man durch die ursprüngliche modulare Beschreibung einzelner Komponenten gewonnen hatte, geht somit in der Darstellung des Analyseergebnisses wieder verloren. Aufgrund der während der Analyse gespeicherten Abgängigkeiten könnte man vielleicht die verteilte Ablaufstruktur durch eine Nachbearbeitung als Ergebnis wieder gewinnen. Es bleibt jedoch dabei, daß über die Unterscheidung zwischen punktförmigen und dauerhaften Zuständen und ihren Nachfolgebeziehungen keine weiteren zeitlichen Relationen spezifiziert bzw. analysiert werden können.

Eine andere zu dem Thema dieses Kapitels wichtige Arbeit ist [McDermott 82]. Einen groben Überblick über Darstellungen von Zeit und auch allgemeiner über Qualitatives Schließen enthält [Charniak/McDermott 85]. [Schwind 85] gibt eine ausführliche Übersicht über Zeitrepräsentationen in der Künstlichen Intelligenz.

10. ZUSTÄNDE VON OBJEKTEN

Objekte reagieren im allgemeinen auf denselben Reiz nicht immer mit demselben Verhalten. Das in Anhang B beschriebene Ventil des Druckreglers weist zum Beispiel die drei Bereiche OFFEN, GESCHLOSSEN und TÄTIG auf, in denen völlig unterschiedliche Reaktionen auftreten. Im Bereich TÄTIG erfüllt das Ventil seine eigentliche Funktion als Regler. Im Bereich GESCHLOSSEN wird keine Druckänderung mehr übertragen, während im Bereich OFFEN jede Änderung entgegen der Reglerfunktion direkt übertragen wird.

Die Wahl solcher Bereiche wird typischerweise dadurch motiviert, daß eine entsprechende das Verhalten beschreibende mathematische Funktion an den Grenzen dieser Bereiche Unstetigkeitspunkte besitzt, oder an dieser Stelle ihr Verhalten qualitativ signifi-

kant verändert (Wendepunkte, Minima, Maxima). Anstatt eine einzige nichtlineare Funktion zu verwenden, versucht man, die nicht lineare Funktion durch eine Reihe linearer Funktionen zu approximieren (Bild 10.1).

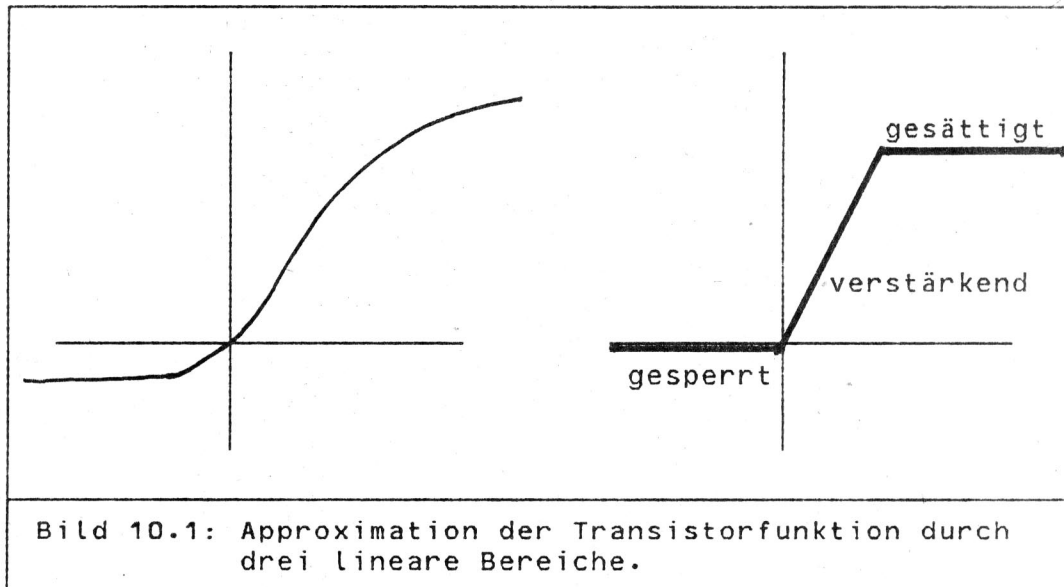


Bild 10.1: Approximation der Transistorfunktion durch drei lineare Bereiche.

Ein Zustand ist somit ein Name für einen in der "Realität" möglicherweise nur annähernd linearen Bereich, in dem das Verhalten des Objekts qualitativ durch eine lineare Funktion beschrieben werden kann.

Zustände dieser Art wurden bereits in EL u.a. zur Modellierung von Dioden und Transistoren verwendet. Zu jedem Zeitpunkt ist das Verhalten eines Objekts durch genau einen zustandsspezifischen Constraint charakterisiert. Da die Information über den Zustand von Objekten bei Beschreibung der gesamten Ausgangssituation in EL nicht vorgegeben werden muß, gehört es zu den Aufgaben der Constraint-Propagierung, den Objekten jeweils einen möglichen Zustand zuzuordnen. Dabei kann es vorkommen, daß Kombinationen von Zuständen der Objekte zu Inkonsistenzen führen. Solche Inkonsistenzen werden in EL im wesentlichen durch sogenannte **Monitore** erkannt. Eine Inkonsistenz kann nur dadurch beseitigt werden, daß für mindestens ein Objekt ein anderer Zustand gewählt wird.

Zur Entscheidung, welche Zustandsauswahlen überhaupt an der Inkonsistenz beteiligt sind, werden in EL alle wichtigen Entscheidungen und Abhängigkeiten dokumentiert. In Abschnitt 7.2 wurde bereits allgemein die hiernach benannte Technik des abhängigkeitsgesteuerten Backtracking diskutiert. Diese Technik läßt sich auf das Beispiel von EL und speziell auf die Repräsentation von Zuständen anwenden, wenn man anstelle der Zustände eines Objekts eine zusätzliche Variable einführt, die als mögliche Werte gerade die Zustandsbezeichner hat. Die für die Verhaltensweisen des Objekts gegebenen Constraints müssen

dann so erweitert werden, daß sie nur bei entsprechender Belegung der Zustandsvariablen anwendbar sind.

Jon Doyle hat später die in EL verwendeten und in EL's Implementationssprache ARS angebotenen Repräsentationen für Abhängigkeiten um nicht-monotone Abhängigkeiten erweitert und verallgemeinert [Doyle 79].

ENVISION, HIQUAL und QPT erlauben ebenfalls die Definition von Objektzuständen. In ENVISION werden die den Zuständen zugeordneten Bereiche durch Ungleichungen über Variablen charakterisiert. Zum Beispiel wird der Zustand TÄTIG des Ventils durch die Ungleichung

$$0 < A < A\text{-max}$$

spezifiziert, wobei A die von dem Ventil freigelassene Durchlaßfläche ist. Während der Analyse muß festgestellt werden, ob sich die qualitativen Werte der Variablen so verändern können, daß die Ungleichung für den momentanen Zustand verletzt und ein neuer Zustand etabliert werden muß. Ist z.B. in der aktuellen Situation $A=+$, $A < A\text{-max}$ und $\Delta A = -$ (ΔA ist die qualitative Ableitung von A), so könnte in der nächsten Situation A den Wert 0 erreichen, und damit der Zustand GESCHLOSSEN erreicht werden.

In HIQUAL gibt es keine solche Charakterisierung von Zuständen durch Ungleichungen, sondern der Wechsel eines Objekts in einen neuen Zustand wird als Aktion einer Regel explizit angegeben. Hiermit wird zwar die Analyse der Zustandsübergänge erleichtert, andererseits hat die Methode von ENVISION aber den Vorteil der höheren Modularität bzgl. der Objektbeschreibungen. Nimmt man z.B. Änderungen an der Anzahl der Zustände oder an der Menge der Constraints eines Zustands vor, so müssen die Verhaltensbeschreibungen in unveränderten Zuständen in ENVISION nicht angepaßt werden, wohl aber in HIQUAL, wenn aus den unveränderten Zuständen veränderte Zustände erreicht werden sollen.

11. DIGITALE UND STETIGE VARIABLEN

Die Systeme zur Modellierung lassen sich danach qualifizieren, ob die auftretenden Variablen (Parameter, Terminals, ...) nur stetige oder beliebige (digitale) Wertänderungen ausführen können.

Sei W der Wertebereich einer Variablen V, p_0 eine partielle Ordnung auf V, $\text{inv}(p_0)$ die inverse Relation von p_0 , und $w \in W$ der aktuelle Wert von V.

Dann heißt V **stetig**, wenn V einen neuen Wert w' nur dann annehmen kann, falls w' in p_0 zu w benachbart ist, also

$(w, w') \in po \cup inv(po)$.

Wenn V nicht stetig ist, dann wird V als **digital** bezeichnet.

Im allgemeinen ist auf den betrachteten Wertebereichen eine totale Ordnung definiert, z.B. auf VZ . Eine stetige Variable darf also nicht von $+$ ($-$) nach $-$ ($+$) wechseln, ohne zwischendurch den Wert 0 gehabt zu haben. Eine ähnliche Konsequenz wird z.B. in der "ordering rule" von ENVISION beschrieben:

E und B seien zwei Objekte; E habe einen Zustand ZE charakterisiert durch $X > m$ und B habe einen Zustand ZB charakterisiert durch $X > n$; $n > 0, m > 0$. In dem aktuellen globalen Zustand sei bekannt, daß $\Delta X = +$ und $X = 0$ gilt. ΔX bezeichnet die qualitative Ableitung von X , X wird also grösser. Der globale Zustand kann nun entweder E mit ZE oder B mit ZB enthalten.

Angenommen, man weiß zusätzlich, daß $n < m$ gilt. Aus der Stetigkeit kann man nun schließen, daß in dem neuen globalen Zustand B in ZB sein muß, also $n \leq X < m$. B ändert seinen Zustand vor E .

Diese Einschränkungen der Wertänderungen spielen z.B. in QPT, ENVISION und HIQUAL eine große Rolle. In dem nicht an den Modellen der Physik orientierten System von Davis gibt es nur digitale Variablen, die für die Modellierung digitaler Hardware auf den höheren Abstraktionsebenen auch adäquat sind. Stetige Variablen werden jedoch interessant, wenn eine Beschreibungsebene der Elektronik erreicht wird, wo die Bits durch Potentiale o.ä. dargestellt werden. Prinzipiell ist es wünschenswert, in einer einzigen Repräsentationssprache sowohl stetige als auch digitale Variablen zur Verfügung zu stellen. Eine andere Frage ist, ob diese dann auch in einer Beschreibungsebene gemeinsam verwendet werden können bzw. sollen.

12. HIERARCHIEN VON MODELLEN

Es herrscht allenthalben eine große Einigkeit, daß Modellierungen auf verschiedenen Abstraktionsebenen möglich sein sollen. Auf die Bedeutung hierarchischer Beschreibungen ist in Teil I und an verschiedenen Stellen dieses zweiten Teils hingewiesen worden.

Trotz dieser Bedeutung sind für ENVISION, QPT und die hiermit verwandten Systeme zur Beschreibung einer qualitativen Physik (wie [Williams 84], [Kuipers 84]) noch keine Erweiterungen zur hierarchischen Modellierung bekannt geworden. Diese Tatsache ist sicherlich nicht auf mangelndes Interesse zurückzuführen. Es ist vielmehr so, daß ein derartiger Ansatz für diese Art von Systemen nicht einfach anzugeben ist. Zudem sind die zu lösenden Pro-

bleme zur Analyse einer einzigen Abstraktionsebene groß genug und relevant genug, um sich zur Zeit noch allein darauf zu konzentrieren.

Die bisher mit den objektorientierten Systemen aus der oben genannten Kategorie erzeugten Modellierungen sind alle noch relativ detailliert auf einer Ebene "unmittelbar" über der Ebene physikalischer Differentialgleichungen angesiedelt. Wie teilweise demonstriert wird, läßt sich diese Art von qualitativen Beschreibungen beinahe automatisch aus den konkreten Differentialgleichungen erzeugen. Hier ist zumindest nicht direkt zu sehen, welche Art von Abstraktion die Modellierung auf einer höheren Ebene enthalten soll. Es gibt zwar bereits innerhalb der heute "gültigen" Physik bereits verschiedene Ebenen der Detailierung, die etwa von der modernen Quantenmechanik bis zu dem physikalischen Weltbild des Isaac Newton reichen. Die meisten Beschreibungen der qualitativen Physik entstehen auch als Abstraktionen aus Newton'schen Gleichungen, die Frage bleibt jedoch: welche adäquaten Modelle der Physik gibt es jenseits einer qualitativen Newton'schen Physik? Konkreter: wie können noch abstraktere Beschreibungen gewonnen werden, die nicht einfach losgelöst existieren, sondern strukturelle und verhaltensmäßige Beziehungen zu den konkreteren Beschreibungen besitzen?

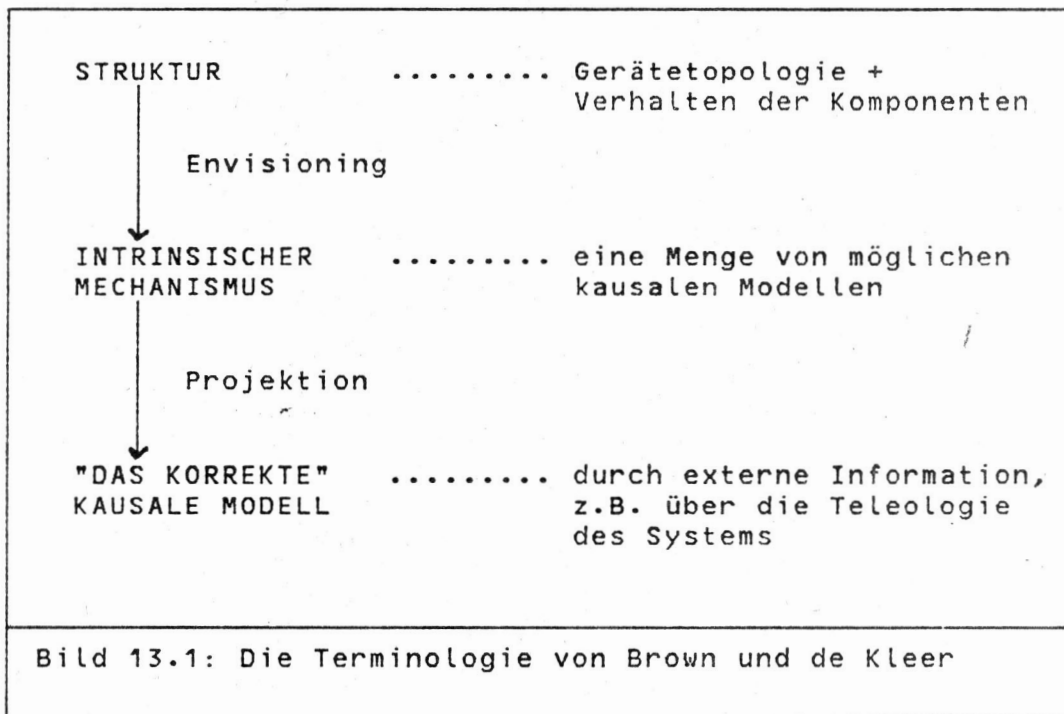
Diejenigen Systeme, die sich nicht unbedingt einer physikalischen Beschreibung des zu modellierenden Systems verschrieben haben, wie etwa Davis, DART oder HIQUAL, bieten dem Benutzer Möglichkeiten zur Bildung von Modellen auf verschiedenen Ebenen an. Anhang A zeigt z.B. Davis' hierarchische Beschreibung eines Addierers. Bis auf für diesen Überblick mehr oder weniger unbedeutende Unterschiede können ähnliche Strukturbeschreibungen auch in HIQUAL und in DART erzeugt werden.

13. SIMULATION UND ENVISIONING

Einige Systeme sind in der Lage, über die Simulation hinaus bzw. anstatt einer Simulation ein **Envisioning** durchzuführen. In der Terminologie von [de Kleer/Brown 83] wird der Begriff des Envisioning in Bild 13.1 präzisiert. Demnach versteht man unter Envisioning den Prozeß des Erzeugens von allen möglichen kausalen Verhaltensweisen eines Gesamtsystems, wenn Struktur und Verhalten der Einzelkomponenten vorgegeben sind. (Der Begriff der Struktur umfaßt bei de Kleer und Brown sowohl die Struktur, wie sie in dieser Übersicht eingeführt wurde, als auch das Verhalten der Komponenten.)

Aus dem möglicherweise nicht eindeutigen Ergebnis des Envisioning können später die "unerwünschten", nicht als anwendbar erachteten Verhaltensweisen aufgrund von Kenntnissen eliminiert werden, die nicht explizit in die Modellierung eingebracht wor-

den sind. Diesen Vorgang bezeichnen de Kleer und Brown als **Projektion**. Wie z.B. teleologische, also auf die Bestimmung des betrachteten Gegenstands zielende Informationen sinnvoll eingebracht werden können, wird in [de Kleer 84] beschrieben.



Die Ursachen für mehrdeutige kausale Interpretationen sind in folgenden Beobachtungen begründet, die mehr oder weniger für alle hier betrachteten Modellierungen Gültigkeit haben:

- Die Werte von Variablen sind qualitativ, d.h. Abstraktionen über den meist reellen Werten der exakten physikalischen Beschreibungen.
- Die in der Analyse erfaßten Ereignisse sind zeitlich nur partiell geordnet. An einigen Punkten fehlt also Information darüber, welches von zwei möglichen Ereignissen zuerst stattfindet.
Vergleiche hierzu etwa die in Kapitel 11 beschriebene ordering rule für den Fall, daß über die Beziehung zwischen n und m nichts bekannt ist. In diesem Fall enthält die durch Envisioning generierte Struktur zwei mögliche Ausführungspfade, einen Pfad für jede mögliche Fortsetzung.
- In den einzelnen Objekten sind keine globalen Informationen über die Beziehungen zu anderen Objekten enthalten. Die aufgrund dieser Maßnahme gewonnene Modularität der Komponentenbeschreibungen muß also bezahlt werden mit einem möglichen Mangel an Informationen, die zur Klärung einiger Mehrdeutigkeiten führen könnten. De Kleer und Brown fordern diese Modularität etwas spezifischer mit dem **no-function-**

in-structure Prinzip: bei der Definition einer Komponente dürfen keine Voraussetzungen über die Objektumgebung gemacht werden, in der die Komponente einmal konkret integriert werden wird.

Wenn man nicht alle Umgebungen antizipieren kann bzw. aus Komplexitätsgründen auch nicht will, dann bleibt die Objektdefinition möglicherweise zu allgemein und umfaßt nicht alle Einschränkungen des Verhaltens, das sich durch konkrete Einbettungen ergeben könnte. De Kleer und Brown schlagen vor, die hierdurch entstehenden Mehrdeutigkeiten erst einmal in Kauf zu nehmen, um sie bei Bedarf später durch Informationen über globalere Zusammenhänge in der konkreten Anwendung auszuräumen.

Wie in Kapitel 8 zur Kausalität diskutiert, ist speziell für das ENVISION System zu klären, ob nicht einige Mehrdeutigkeiten auch durch eine gezielte Einschränkung der Anwendungsrichtung von Constraints beseitigt werden könnten.

Jedes durch Envisioning erzeugte Verhalten kann man als eine explizite Repräsentation einer Simulation des Systems unter den gewählten Randbedingungen betrachten. Insofern kann man sagen, daß mit der Existenz eines Envisioning-Verfahrens auch Simulationen umfaßt werden, ohne daß eine Interpretation oder Ausführung der beschriebenen Komponenten explizit **ablaufen** muß.

Während die Systeme mit ausschließlich digitalen Variablen (Davis und DART) nur Simulationen ausführen können, erzeugen die Systeme mit stetigen Variablen (ENVISION, QPT, HIQUAL usw.) eine Repräsentation aller möglichen Verhaltensweisen.

14. ZUSAMMENFASSUNG

Wir haben in diesem zweiten Teil die wichtigsten Eigenschaften von KI-Systemen zur Modellierung von technisch-physikalischen Gegenstandsbereichen diskutiert. Die wesentlichen Punkte werden im folgenden noch einmal kurz aufgezählt.

- Der von McCarthy vorgeschlagene Situationskalkül kann aus zwei Gründen keine befriedigende Lösung sein:
 1. Parallele Abläufe lassen sich nicht darstellen.
 2. Die unstrukturierten globalen Zustände machen das Frame-Problem zu einem unüberwindlichen technischen Hindernis.

- Der Common Sense Algorithmus CSA erlaubt eine differenzierte Darstellung vieler Arten zeitlicher und kausaler

Zusammenhänge. Die Beschreibung eines komplexen, aus einzelnen Komponenten zusammengesetzten Systems hat jedoch einen ad hoc Charakter. Es fehlt an der Modularität der Beschreibung, d.h. an geeigneten Strukturierungsmitteln, um ein Gesamtverhalten weitgehend aus der Summe der Einzelverhalten der beteiligten Komponenten erzeugen zu können.

- Eine solche Modularität ist kennzeichnend für spätere Entwicklungen. Erreicht wird dies entweder durch objektorientierte oder durch prozeßorientierte Strukturbeschreibungen.

Forbus' Qualitative Theorie von Prozessen ist eine Verallgemeinerung und teilweise Präzisierung der von Hayes entwickelten Grundidee der Geschichten. Eine Geschichte ist ein zeitlich und räumlich begrenzter Ausschnitt aus dem Gesamtgeschehen. Durch diese Begrenzung des Kontexts wird das Frame-Problem entschärft und nach Forbus' Ausführungen ersetzt durch die einfacheren Probleme der lokalen Entwicklung von Prozessen und dem Problem der Wechselwirkungen zwischen Prozessen. Für alle Veränderungen in der betrachteten Welt sind Prozesse verantwortlich zu machen.

In objektorientierten Strukturbeschreibungen sind die Objekte die aktiven Einheiten. Objekte können zur Kommunikation von Informationen miteinander verbunden werden, um so komplexere Strukturen zu bilden.

- In allen Systemen findet der Austausch von Wechselwirkungen zwischen verschiedenen Einheiten (Objekte oder Prozesse) implementatorisch durch die Technik der Constraint-Propagierung statt. Bis zu einem gewissen Grad kann man also alle Systeme als Constraint-Systeme bezeichnen. Dies gilt in den Systemen QPT, ENVISION, ENV und Williams nur für die von einer globalen Analyse der Zustandsänderungen unterschiedenen "Inkrementellen (Störfall-) Analyse", in der die Reaktion des Systems auf eine lokal ausgeführte Störung eines Gleichgewichtszustands beobachtet wird.

Constraint-Systeme unterscheiden sich in ihrer Mächtigkeit durch die Eigenschaften der Constraint-Relationen und der Anwendbarkeit von Constraints bei unvollständiger Information. Für die praktisch relevanten Techniken haben wir unterschieden zwischen einfachen und weitgehend einfachen Relationen. Weitgehend einfache Relationen erfordern aufgrund ihrer teilweisen Mehrdeutigkeit ein Backtracking-Verfahren.

Sind zur Anwendung eines Constraints mindestens zwei Variablen unbekannt, und kann man diesen Variablen keine eindeutigen Werte zuordnen, so kann man mit der Erzeugung von symbolischen Werten und der Propagierung von symbolischen Ausdrücken Fortschritte erzielen.

Sind die vorliegenden Einschränkungen zu ungenau, d.h. bleiben für eine Variable zu viele mögliche Werte übrig, so könnte man möglicherweise einen destruktiven Propagierungsalgorithmus bzw. eine Kombination aus konstruktiver und

destruktiver Propagierung anwenden.

- Die verwendeten Strukturierungsmechanismen schränken die Pfade der möglichen kausalen Wechselwirkungen bereits syntaktisch ein. Ein Unterschied zwischen den Systemen besteht darin, ob Verbindungen gerichtet sind oder nicht. In ENVISION werden z.B. nur ungerichtete Verbindungen erlaubt, d.h. Constraints sind stets in allen Richtungen anwendbar. Forbus hingegen demonstriert, daß eine uneingeschränkte Anwendung von Constraints (hier speziell Gleichungen) zu nicht-kausalen Erklärungen führen kann.
- Die Repräsentation von genaueren, aber immer noch möglichst qualitativen zeitlichen Zusammenhängen ist im allgemeinen noch nicht befriedigend gelöst. Die Analyse in ENVISION und den artverwandten Repräsentationen besteht in der Erzeugung einer Struktur, die zwar für alternative Pfade Verzweigungen enthält, in der aber jeder Knoten einen globalen Zustand des Gesamtsystems repräsentiert. In der Analyse der möglichen Tätigkeiten von Prozessen kann Forbus das asynchron parallele Verhalten eines zusammengesetzten Systems besser abbilden. In HIQUAL können zeitliche Beziehungen am genauesten spezifiziert und analysiert werden.
- In exakteren Modellierungen nicht-lineare Verläufe werden zur Handhabbarkeit für qualitative Schlüsse meistens in approximativ lineare Abschnitte zerlegt. Einem solchen linearen Teilbereich wird ein Bezeichner für einen Zustand zugeordnet, unter dem lineare Beziehungen das für diesen Abschnitt typische Verhalten beschreiben.
- Zur praktikablen Modellierung und Folgerungsfähigkeit über komplexen Systemen ist eine Beschreibung auf mehreren Abstraktionsebenen notwendig. In den Systemen mit der Intention einer qualitativen Physik haben Hierarchien von Modellen bisher kaum Eingang gefunden. In den von physikalischen Modellen mehr losgelösten Systemen wie Davis, HIQUAL und DART gibt es Methoden zur Definition von Hierarchien, die ihre Analogie in Spezifikationen von Hierarchien von Datentyprepräsentationen im Bereich des Software-Engineering finden (vgl. z.B. [Silverberg 81]).
- Einige Systeme zielen darauf ab, alle möglichen Verhaltensweisen des repräsentierten Mechanismus zu analysieren (Envisioning), während andere sich auf die Möglichkeit der Simulation beschränken. Dafür bieten die "Simulationssysteme" wie z.B. Davis und DART die zusätzliche Möglichkeit einer Rückwärtsinterpretation für Zwecke der Diagnostik an.

15. AUSBLICK

Alle in diesem Überblick betrachteten Systeme mit Ausnahme der frühen Arbeiten von McCarthy sind im Jahr 1986 nicht älter als 10 Jahre, die meisten sogar erheblich jünger. Zumindest seit den beherzten Aufforderungen in [Hayes 79] hat die Beschäftigung mit der hier behandelten Thematik zurecht einen enormen Zulauf zu verzeichnen. Wie in allen "Modegebieten" (womit wir in keiner Weise ausdrücken wollen und auch nicht hoffen, daß die Thematik bald out-of-date ist, im Gegenteil) gibt es auch hier eine teilweise noch ungestüme Euphorie. Die hieraus resultierenden zum Teil überzogenen Erwartungen werden sich aber recht bald einer realistischeren Sicht der bisher erreichten Lösungen und der noch zu bewältigenden Probleme fügen.

Ein generelles Hindernis zur praktischen Anwendbarkeit wohl aller bisherigen Ansätze ist die ungenügende Effizienz. Diese Ineffizienz ist teilweise den gestellten Problemen immanent, ließe sich aber in Einzelfällen durch problemspezifischere Methoden verbessern. So bietet es sich z.B. an, die Annahmen über die Wahl des Wertes einer Variablen wissensgesteuert auszuführen. Heuristiken dieser Art werden in noch größerem Umfang in die dargestellten Verfahren integriert werden müssen.

Es gibt zur Zeit noch keine realistischen Anwendungen der betrachteten Techniken. Im Bereich der Hardwarebeschreibungen wird man wahrscheinlich dieses Ziel zuerst erreichen, wie insbesondere die Arbeiten von Davis vermuten lassen. Das größte Hindernis auf diesem Weg dürfte in der noch mangelnden Ausdrucksfähigkeit zur Darstellung von komplexeren zeitlichen Abläufen liegen. Speziell auf dem Gebiet des VLSI-Entwurfs oder Teilen davon sind einige hier nicht weiter beschriebene Arbeiten angesiedelt, die sicherlich auch nicht mehr lange auf erste praktische Erfolge warten lassen.

In vielen Forschungsgruppen denkt man an Anwendungen im Bereich der Konstruktion, der Prozeßüberwachung, der Fertigungsplanung oder ähnlichen technischen Bereichen. Auch hier ist die mangelnde Ausdrucksfähigkeit für zeitliche Beziehungen oft noch ein schwieriges Problem. Aber es kommen auch andere Schwierigkeiten dazu:

- In dynamischen Umgebungen müssen Änderungen der vorhandenen Daten effizient verarbeitet werden können. Die vorhandenen Techniken mit abhängigkeitsgesteuertem Backtracking und der Verwaltung von Abhängigkeitsnetzen sind zum Teil weitaus besser als die vor 10 Jahren benutzten Verfahren, doch für einen praktischen Einsatz meistens auch noch nicht effizient genug. Die intensiven Bemühungen in dieser Richtung, z.B. die Entwicklung des ATMS (assumption based truth maintenance) von de Kleer [de Kleer 86] weisen dies deutlich aus.

- Für realistische Anwendungen müssen qualitative und exakte quantitative Modelle ohne viele "Kanten" ineinander übergreifen. Insbesondere Probleme mit unvollständigen Informationen stellen hohe Anforderungen an die Qualität der Symbolverarbeitung in Constraint-Systemen. Eine Beschränkung auf lineare Funktionen wird sich wahrscheinlich auch nicht in allen Fällen einhalten lassen, sodaß auch hier mit einem erhöhtem Aufwand gerechnet werden muß.

Die Behandlung von Unsicherheiten wird durch eine Integration von Wahrscheinlichkeitsmodellen ergänzt werden müssen. Die von de Kleer vorgeschlagene Hardware-Diagnostik [de Kleer/Williams 85] z.B. verwendet unmittelbar den ATMS und berücksichtigt auch probabilistische Werte über Ausfallhäufigkeiten der Komponenten.

Generell kann man zur Problematik der Modellhierarchien sagen, daß Techniken aus dem Bereich des Software-Engineering insbesondere zur Spezifikation abstrakter Datentypen in Verbund mit KI-Arbeiten zur Darstellung von Hierarchien (Frames) wertvolle Beiträge zur Präzisierung der strukturellen Beziehungen leisten könnten. Aus einer Kombination dieser Methoden werden sich möglicherweise mächtigere Techniken zur Darstellung von Hierarchien von Constraint-Systemen entwickeln.

- In vielen Bereichen des CAD/CAM spielen geometrische Beziehungen eine große Rolle. Zu qualitativen Darstellungen der Geometrie ist in diesem Überblick noch nichts gesagt worden. Dies hat seinen Grund darin, daß es hierüber bzgl. der betrachteten Systeme auch nicht viel zu berichten gibt. Erst seit jüngster Zeit existieren einige wenige Ansätze, in denen geometrischen Beziehungen eine größere Bedeutung beigemessen wird. Dies ist z.B. der Fall in COMODEL [Dilger/Kippe 85] oder in dem Expertensystem GARI zur Fertigungsplanung [Descotte/Latombe 85]. GARI ist auch aus zwei anderen Gründen besonders interessant:

1. Das Planungsproblem wird als ein Constraint-Problem aufgefaßt und realisiert. Dabei werden die relevanten Constraints aufgrund der aktuellen Parameter des zu fertigenden Teils, der zur Verfügung stehenden Maschinen und der geometrischen Struktur des Teils zunächst dynamisch erzeugt. Zusätzlich können Constraints mit Gewichten versehen werden, die bei Fertigungsalternativen entsprechend berücksichtigt werden müssen.
2. Zur Behebung von auftretenden Konflikten wurden in einer ersten Version sämtliche Planungsentscheidungen in Abhängigkeitsnetzen a la Doyle protokolliert. Diese Lösung war jedoch aufgrund des enorm hohen Platzverbrauchs zu ineffizient. Man entschied sich daher für ein uninformiertes Rücksetzungsverfahren, das aber die Gewichte der getroffenen Entscheidungen

berücksichtigt und die Optimalität der gefundenen Lösung bzgl. der gesetzten Kriterien garantiert.

- Besonders für Konstruktionsprobleme ist die Integration von Wissen über die Funktion des betrachteten Systems ausschlaggebend. Bedeutsam ist hier sicherlich das System EQUAL [de Kleer 84], weitere Arbeiten in diesem Bereich sind jedoch rar.

Für den Konstruktionsbereich ist eine Hierarchisierung der Modelle besonders wichtig, und es wäre gerade hier von großer Bedeutung, Techniken der Softwarekonstruktion mit den Begriffen der "Spezifikation" und der "Implementierung" (vgl. z.B. [Klaeren 83]) in die Betrachtungen einfließen zu lassen.

Ein anderes Erfordernis besteht in der Protokollierung der wichtigsten Konstruktionsentscheidungen, um auftretende Inkonsistenzen insbesondere zwischen Spezifikation und Implementierung effizient beheben zu können.

Leider unvollendet gebliebene Ansätze zu den beiden letzten Punkten sind in [Raulefs 84] präsentiert worden.

Diese abschließende Diskussion soll aber nicht den Eindruck erwecken, daß es hier einzig und allein darum gehe, möglichst schnell möglichst viele bedeutende Anwendungen anbieten zu können. Die Betrachtung von Anwendungen soll vielmehr dazu anregen, den Blick für die Realität und die noch ausstehenden Probleme nicht zu verlieren.

Besonders den Arbeiten zur qualitativen Physik muß ein hoher Erkenntniswert zugesprochen werden, der ohne jede konkrete Anwendung allein für sich eine große Bedeutung erlangen wird. Dabei sind die Arbeiten in diesem Bereich längst nicht abgeschlossen. Alle oben genannten Probleme treffen natürlich auch hier zu. Zusätzlich wird in nächster Zukunft einiges investiert werden, um die zugrundeliegenden mathematischen Modelle zu erweitern und zu präzisieren. Diese Arbeiten werden den Grundstock zu einer qualitativen Mathematik, insbesondere der Analysis, bilden.

Wie bereits in den bisherigen Modellen, werden zur Beschreibung höherer Abstraktionsebenen Erkenntnisse aus der Kognitionswissenschaft in die Modellbildungen verstärkt einfließen (vgl. etwa [Gentner/Stevens 83]). Dies wiederum wird einen fruchtbaren Rückkopplungseffekt auf die Kognitionswissenschaft haben, da die erzeugten Modelle zur Überprüfung der mentalen Modelle dienen können. Auch hier starten momentan viele Aktivitäten.

Anhang A DAS SYSTEM VON DAVIS

A1 ÜBERBLICK

Das System von Davis [Davis 84], im folgenden kurz mit "Davis" bezeichnet, dient zur Fehlersuche in digitaler elektronischer Hardware. Ein betrachtetes Gerät wird in Hierarchien von Modellen sowohl auf logischer als auch auf physikalischer Ebene repräsentiert. Zur Diagnostik wird eine geordnete Folge von Fehlerkategorien betrachtet, wobei jede Kategorie mit einem bestimmten Typ von möglichen Störungen assoziiert ist (interactions, z.B. elektrische, magnetische usw.). Ein Gerät wird bzgl. einer Fehlerkategorie vollständig untersucht, bevor die in der Ordnung folgende Kategorie berücksichtigt wird.

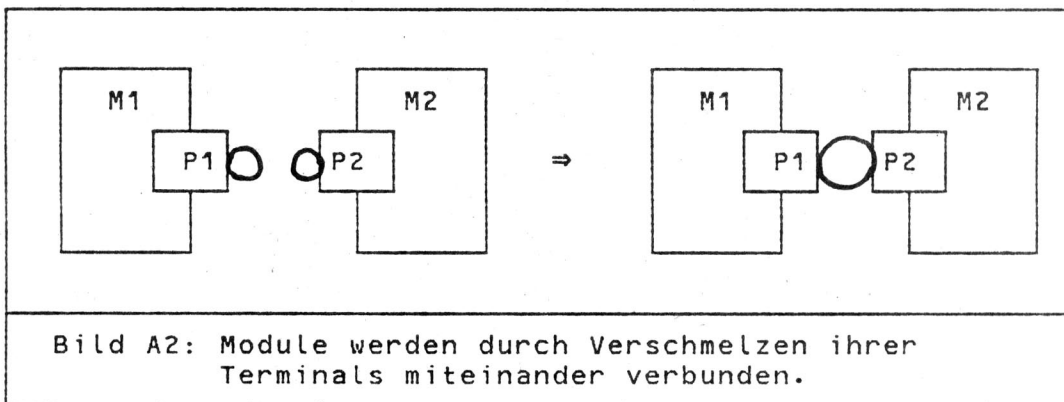
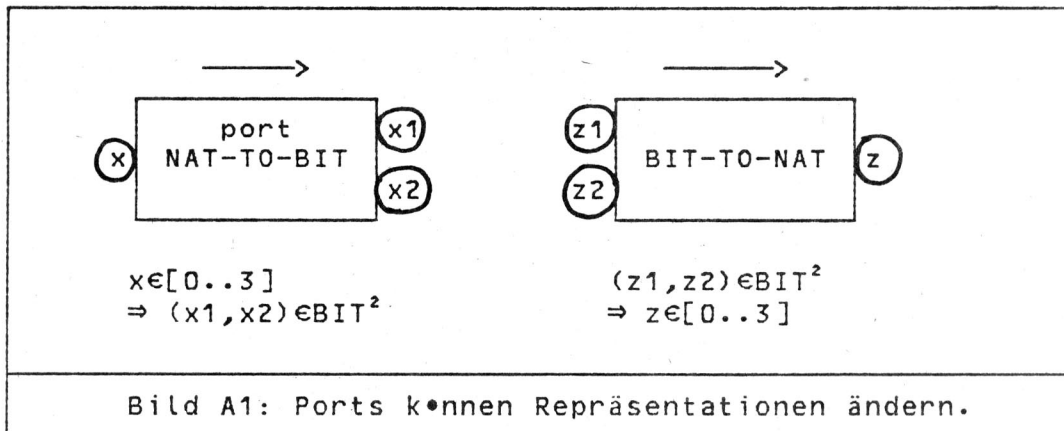
Das Verhalten von Komponenten wird durch einfache Constraints spezifiziert. Die entscheidende Technik zur Diagnostik besteht darin, den Constraint einer verdächtigen Komponente temporär aufzuheben, um so zu erfahren, ob ohne die Wirkung der Komponente alle aufgetretenen Symptome erklärt werden können.

A2 DARSTELLUNG DER STRUKTUR

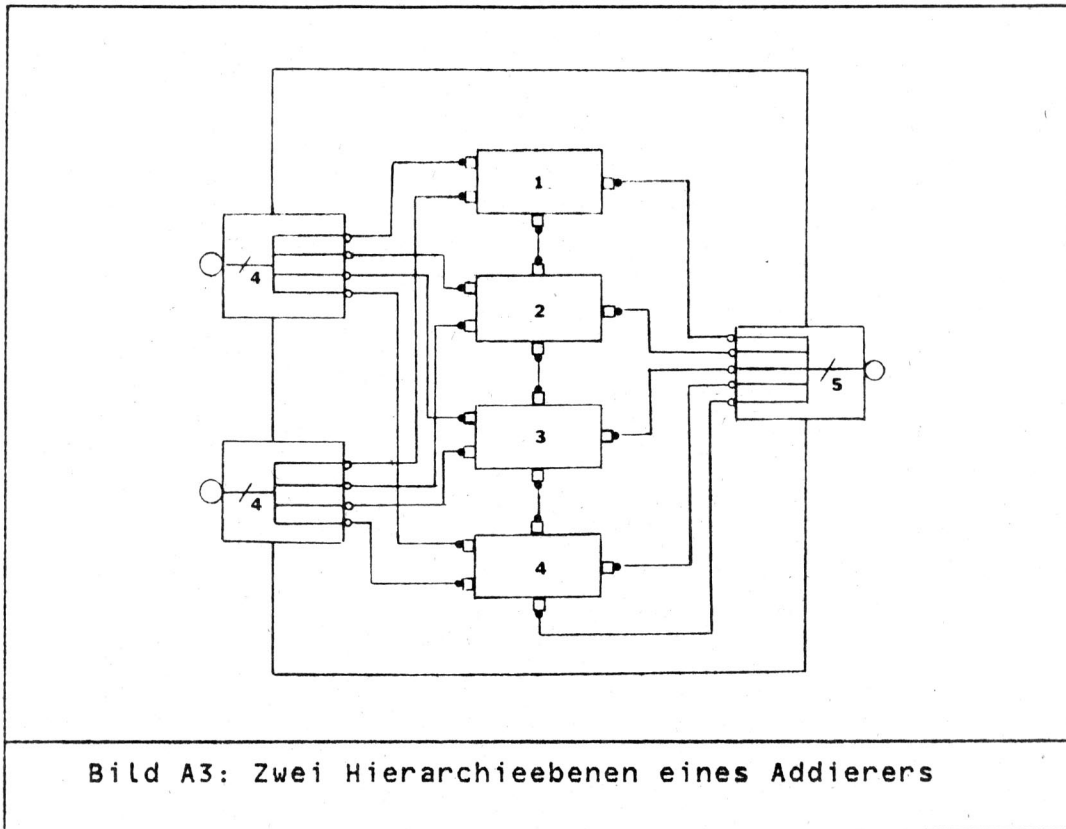
Die Struktur eines Geräts wird auf der logischen und auf der physikalischen Ebene repräsentiert. In beiden Repräsentationen treten die folgenden Einheiten auf:

- **Module:** Module sind Prototypen von Objekten, die instanziiert und über Ports miteinander verbunden werden können.
- **Ports:** Ports haben ein äußeres und mindestens ein inneres Terminal. Ports erlauben einen Wechsel der Datenrepräsentation (Bild A1).
- **Terminals:** Module werden miteinander verknüpft, indem zwei äußere Terminals miteinander verschmolzen werden (Bild A2).

Ein Modul kann Komponenten besitzen, die selbst wieder als Module realisiert sind und miteinander oder mit den inneren Terminals des äußeren Moduls verbunden sind (Bild A3).



Die Hierarchie der physikalischen Module ist dreistufig und besteht aus Integrierten Schaltungen (Chips), Platinen (Boards) und Gehäusen (Cabinets). Die Blattknoten in der Hierarchie der logischen Module (Gatter) und der physikalischen Module (Chips) werden einander zugeordnet. Dadurch kann jeder logischen Einheit der Ort ihres physikalischen Auftretens als diejenige physikalische Einheit zugewiesen werden, die alle physikalischen Orte von Gattern der logischen Einheit als Nachfolger besitzt.



A3 DARSTELLUNG DES VERHALTENS

Das black-box Verhalten einer Komponente wird durch einen einfachen Constraint dargestellt. Dabei wird jede Auswertungsrichtung des Constraints bei einer Unbekannten durch eine eigene Regel repräsentiert. Diese Regeln heißen

- **Simulationsregeln** (S-Regeln), wenn die zu bestimmende Größe der Wert eines Ausgabe-Ports ist.
- **Ableitungsregeln** (A-Regeln), wenn der Wert eines Eingabe-Ports bestimmt wird.

Beispiel: Ein ADDIERER-Constraint $x + y = z$ wird dargestellt durch

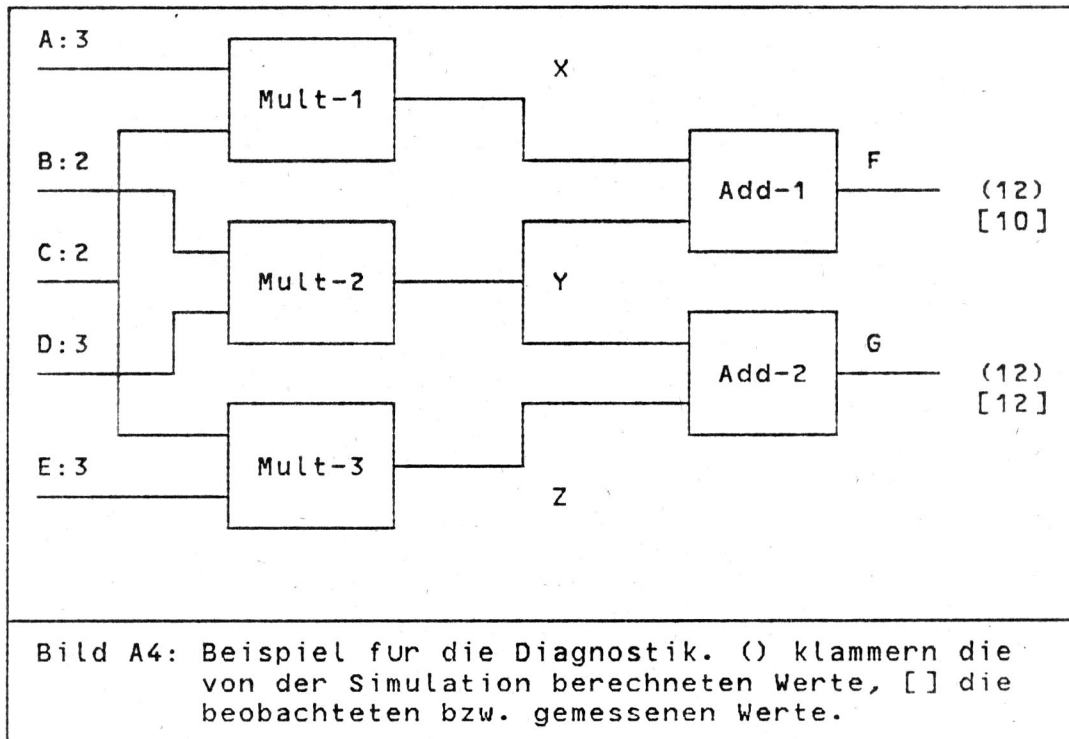
berechne z aus (x,y) durch $(+ x y)$ {S-Regel}
 berechne x aus (z,y) durch $(- z y)$ {A-Regel}
 berechne y aus (z,x) durch $(- z x)$ {A-Regel}

Im Gegensatz zu A-Regeln modellieren S-Regeln den tatsächlich möglichen Fluß von Elektrizität. Die beiden Typen von Regeln werden durch zwei voneinander unabhängige Netzwerke implementiert. Jedes Terminal besitzt dazu zwei Plätze, von denen einer einen Simulationswert speichern kann, der andere möglicherweise einen abgeleiteten Wert enthält. Zusätzlich enthält jeder Platz Informationen darüber, welche Regeln prinzipiell auf den Platz zugreifen können, und wenn ein Wert vorhanden ist, von welcher Regel dieser Wert berechnet wurde. Diese Informationen sind grundlegend für den Diagnostikprozeß.

A4 ANWENDUNG

Die Diagnostik beruht auf einer Wechselwirkung zwischen S-Regeln und A-Regeln. Ausgehend von vorgegebenen Inputs generieren die S-Regeln die vom Modell erwarteten Outputs. Die tatsächlich am realen Gerät vorgefundenen Outputs können von den A-Regeln rückwärts propagiert werden und ermöglichen so Schlußfolgerungen über das aktuelle Verhalten. Vergleiche der von S-Regeln und von A-Regeln ermittelten Werte bilden die Grundlage für die Diagnostik.

Beispiel: Für den Schaltkreis in Bild A4 wird für F der Wert 10 beobachtet, die Simulationsregeln berechnen aber mit den eingezeichneten Eingaben den (gewünschten) Wert 12. Wir schildern nun die weiteren prinzipiellen diagnostischen Überlegungen, die dann von Davis durch eine geschickte Technik realisiert werden.



Die Analyse des Abhängigkeitsnetzwerks für die Simulation ergibt, daß die einzigen an dem Ergebnis beteiligten Komponenten Mult-1, Mult-2 und Add-1 sind. Add-1 kommt als unmittelbar für die Ausgabe verantwortliche Komponente direkt als Kandidat in Frage. Gilt das auch für Mult-1 und Mult-2? Zur Überprüfung von Mult-1 geht Davis nun davon aus, daß Add-1 und Mult-2 korrekt sind. Die Ableitungsregel für X in Add-1 liefert den Wert 4 für die Ausgabe von Mult-1, wenn sie für $F = 10$ und $Y = 6$ { wie im Ausgabeplatz von Mult-2 noch gespeichert } ausgeführt wird. Da Mult-1 keine Vorgänger-Komponenten besitzt, kommt auf diesem Weg nur Mult-1 als weiterer Kandidat in Frage. Außerdem wissen wir nun, welchen Wert Mult-1 im Fehlerfall für seine gegebenen Eingaben liefert. Diese Information könnte z.B. später in weiteren Tests bzw. bei weiterer Detaillierung der Komponenten Mult-1 als Kandidaten ausschließen.

Wir untersuchen nun, ob Mult-2 auch ein möglicher Kandidat sein kann. Die Ableitungsregel für Y von Add-1 ergibt, daß Mult-2 den Wert 4 geliefert haben muß. Eine Vorwärts-Simulation dieser Ausgabe ergibt jedoch, daß in diesem Fall G ebenfalls den Wert 10 gehabt haben müßte. Gemessen wurde aber der Wert 12. Also kann die Annahme, daß (nur) Mult-2 fehlerhaft ist, die beobachteten Symptome nicht erklären. Damit scheidet Mult-2 als Kandidat aus.

Davis hat zur Überprüfung von Kandidaten eine besondere Technik entwickelt, die als **Aufhebung von Constraints** (constraint suspension) bezeichnet wird und mit der das oben beschriebene sequentielle Verfahren parallel ablaufen kann. Während in

Constraint-Netzen bei Auftreten von Konflikten eine der gemachten Annahmen (Wert einer Variablen) rückgängig gemacht werden muß, wird hier ein Constraint - und damit das Verhalten einer Komponente - für eine Inkonsistenz verantwortlich gemacht. Die Idee besteht nun darin, den Constraint einer verdächtigten Komponente zeitweise aufzuheben, um ihn in einer anschließenden Propagierung nicht mehr zu berücksichtigen.

Die verdächtige Komponente kommt nun genau dann als tatsächlicher Kandidat in Frage, wenn eine Propagierung

- unter Verwendung von S-Regeln und A-Regeln
- bei Eingabe aller Symptome und der ursprünglichen Inputs

keine Inkonsistenz ergibt.

Beispiel: Wird der Constraint von Add-1 aufgehoben und wird anschließend propagiert, so erscheint kein Wert in dem Simulations-Platz der Ausgabe von Add-1. Es tritt also auch keine Inkonsistenz mehr zwischen dem beobachteten Wert 10 in dem Ableitungs-Platz der Ausgabe von Add-1 und dem (ohne Aufhebung von Add-1) von Add-1 durch Simulation gelieferten Wert auf. Da der Wert 10 von Add-1 auch nicht rückwärts propagiert wird, entstehen auch an anderen Plätzen keine Inkonsistenzen. Add-1 ist also ein Kandidat.

Bei Aufhebung der Constraints für Mult-2 wird von Add-1 der Wert 4 für Y berechnet. Der eigentlich von Mult-2 zu berechnende Wert 6 für Y wird nicht erzeugt, da Mult-2 aufgehoben ist. So entsteht also auch hier keine Inkonsistenz. Zu einer solchen kommt es dann aber doch, wenn $Y = 4$ und $Z = 6$ von Add-2 zu dem Ergebnis $G = 10$ führt, was mit dem beobachteten Wert nicht übereinstimmt.

Allgemein kann das gesamte Verfahren zur Generierung von Kandidaten in drei Schritten beschrieben werden:

1. Sammle alle Differenzen zwischen beobachtetem und simuliertem Wert.
2. Bestimme potentielle Kandidaten aufgrund der Abhängigkeitsnetze.

Für jede in 1. erkannte Differenz erzeuge die Menge der an dem Simulationsergebnis beteiligten Komponenten.

Bilde die Schnittmenge aller in 2.1 erzeugten Mengen.

3. Teste die Inkonsistenz der Kandidaten durch Aufhebung ihrer Constraints.

Enthält die Kandidatenmenge nach Schritt 3 noch mehrere gleich plausible Kandidaten, so müssen geeignete Testeingaben entwickelt werden, um einige dieser Kandidaten auszuschneiden. Einen Beitrag zu diesem wohl noch nicht abgeschlossenen Problem lie-

fert [Shirley/Davis 83].

A5 FEHLERKATEGORIEN

Es gibt schwierigere Probleme, die allein aufgrund des bisher vorhandenen Wissens nicht zu diagnostizieren sind. Für diese Probleme muß die "Closed-World-Assumption", die nur die in den Modulen definierten Strukturen und Wechselwirkungspfade zuläßt, zugunsten von erweiterten Modellen aufgehoben werden. Davis schlägt für das betrachtete Gebiet die folgende weitere Reihenfolge in der Berücksichtigung von Fehlern vor:

1. Bestimmte Fehler im Verhalten (z.B. stuck-at Fehler)
2. Brücken (Verbindungen, wo in der Strukturbeschreibung keine vorgesehen sind). Zur Einschränkung der kombinatorischen Anzahl möglicher Brücken kann die Eigenschaft der 'physikalischen Nachbarschaft' herangezogen werden. Zur Bestimmung dieser Eigenschaft sind Referenzen zur physikalischen Hierarchie notwendig.
3. Nicht vorhergesehene Richtung der Elektrizität (Inputs arbeiten plötzlich als Outputs)
4. Mehrere Fehler sind gleichzeitig verantwortlich
5. Intermittierende Fehler (intermittent error)
6. Fertigungsfehler
7. Entwurfsfehler

In der weiteren Entwicklung der Diagnostik-Systeme sollen diese Fehlerkategorien explizit modelliert werden, indem die jeweils zugrundeliegenden Annahmen explizit formuliert und in Abhängigkeitsnetzen festgehalten werden.

Anhang B Das System von De Kleer

Die Ziele der Envisioning Theorie als einer qualitativen Theorie der Physik werden am besten von den Autoren selbst beschrieben:

"A qualitative physics predicts and explains the behavior of mechanisms in qualitative terms. The goals for the qualitative physics are (1) to be far simpler than the classical physics and yet retain all the important distinctions (e.g. state, oscillation, gain, momentum) without invoking the mathematics of continuously varying quantities and differential equations, (2) to produce causal accounts of physical mechanisms that are easy to understand, and (3) to provide the foundations for common sense models for the next generation of expert systems. ..."

[De Kleer/Brown 84, p.7]

Die in dem ENVISION System implementierte Envisioning Theorie (ET) ist also einerseits eine Theorie der Physik mit dem Ziel einer qualitativen Beschreibung und Verhaltensanalyse von technisch-physikalischen Geräten. Andererseits wird mit ET eine Theorie der Kausalität angestrebt, die als Ziel dem Menschen verständliche Erklärungen der Funktionsweise von solchen Geräten liefern möchte.

Abschnitt B1 gibt einen Überblick. Nach der Strukturbeschreibung in B2 wird die Repräsentation des Verhaltens einzelner Komponenten in B3 dargestellt. Der letzte Abschnitt B4 erklärt die Analyse des Gesamtverhaltens eines komplexen Gerätes an einem Beispiel.

B1 Überblick

ET ist objektorientiert. Aus einer Beschreibung der Gerätetopologie mit den gewünschten Komponenten und ihren Verbindungen, sowie den einer Modellbibliothek entnommenen Verhaltensbeschreibungen der geforderten Komponenten wird versucht, das Gesamtverhalten des Gerätes zu analysieren. Objekte besitzen Zustände, denen spezifische Verhaltensweisen in Form von qualitativen Differentialgleichungen zugeordnet werden. Die qualitativen Differentialgleichungen sind linear und werden als Konfluenzen bezeichnet. Ausgangspunkt für die Verhaltensanalyse des gesamten Gerätes ist eine Menge von Konfluenzen, die zustandspezifisch aus den Verhaltensbeschreibungen der Einzelkomponen-

ten entnommen werden.

ET unterscheidet zwischen einer zustandsinternen Verhaltensanalyse (intrastate analysis) und einer globalen Verhaltensanalyse (interstate analysis). Die globale Analyse ergibt sich als eine Iteration der zustandsinternen Analyse, wobei von einer Iteration zur nächsten die zustandsinternen Veränderungen tendenzmäßig erfaßt werden, und die sich hieraus möglicherweise ergebenden neuen Objektzustände durch die neuen zustandsspezifischen Konfluenzen berücksichtigt werden. Das Ergebnis einer globalen Analyse ist i.a. mehrdeutig und läßt sich in einem nichtdeterministischen Zustandsübergangsgraphen darstellen.

Eine ausführliche Beschreibung von ET (auch von Forbus' QPT u.a.) findet man in [Dilger 85]. Aus diesem Vorlesungsmanuskript werden auch einige Korrekturen an dem Originaltext in [De Kleer/Brown 84], insbesondere für das Beispiel in Abschnitt B4, übernommen.

B2 Darstellung der Struktur

ET kennt die Einheiten

- **Materialien:** das sind Informationen, Kräfte oder Stoffe
- **Verbindungen:** transportieren Materialien, ohne sie zu verändern
- **Komponenten:** können Materialien verändern.

Auf eine exakte Darstellung der Sprachmittel zur Strukturbeschreibung möchten wir verzichten. Die prinzipielle Vorgehensweise sollte aus dem Beispiel (Bild B1, B2) hervorgehen.

B3 Darstellung des Verhaltens einer Komponente

Eine einzelne Komponente wird beschrieben durch eine Menge von Konfluenzen. Eine Konfluenz enthält Konstanten aus $VZ = \{+, -, 0\}$ und Variablen über VZ . Eine Variable ist entweder eine Grundgröße V oder eine abgeleitete Variable (notiert als ΔV). Zur Beschreibung nichtlinearen Verhaltens kann die Menge der Konfluenzen auf eine endliche Menge von Komponentenzuständen aufgeteilt werden. Ein Zustand wird definiert durch einen Namen und eine Menge von Prädikaten über nicht-abgeleiteten Variablen (Grundgrößen des Systems).

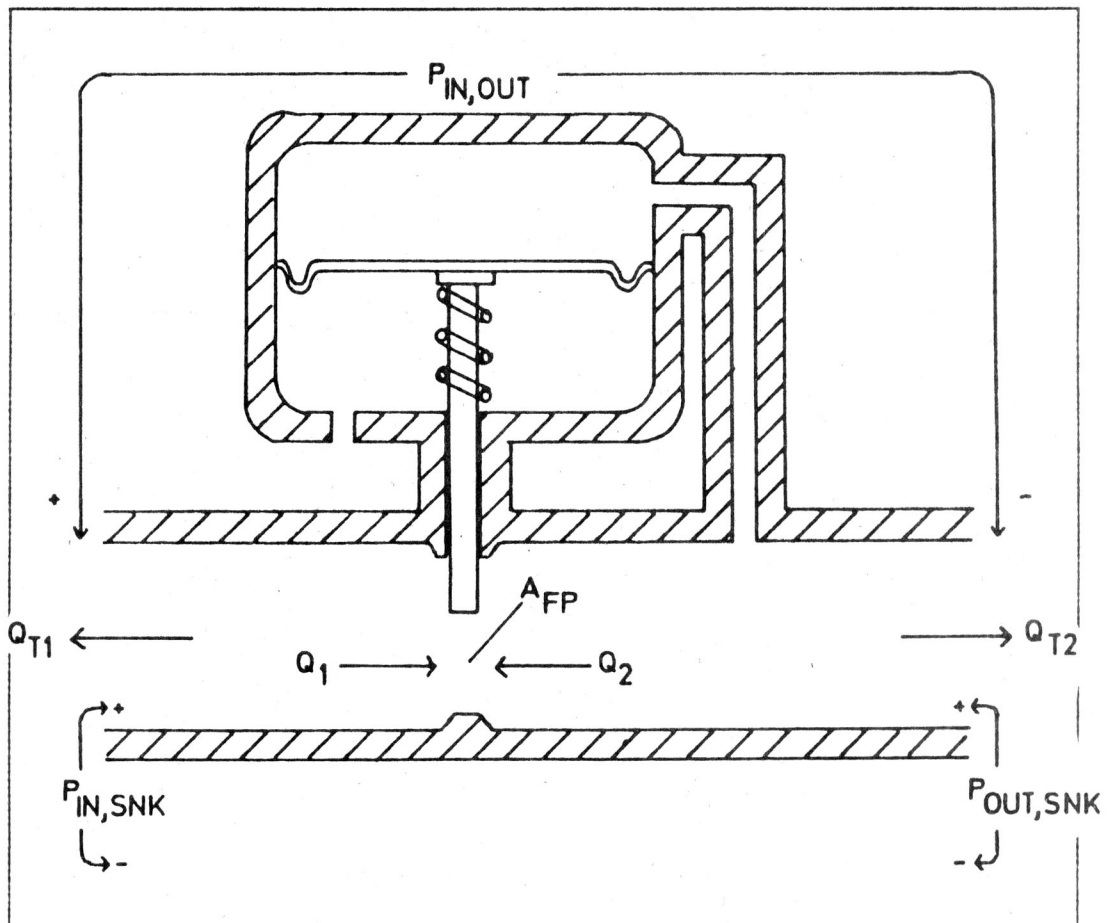
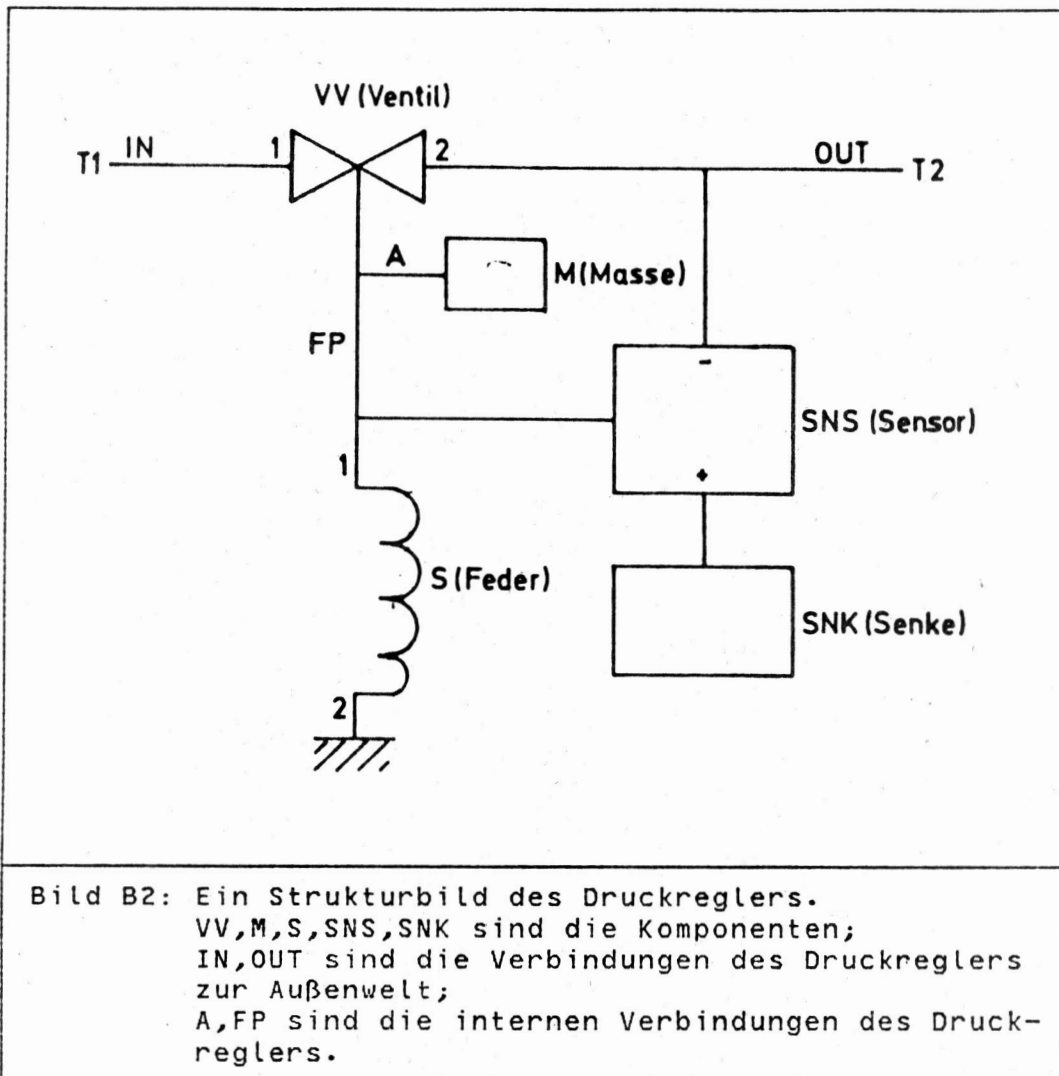
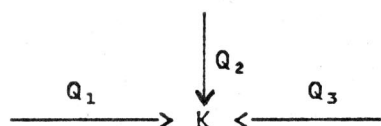


Bild B1: Der Druckregler.

Eine Druckerhöhung am Eingang des Reglers führt zu einem erhöhten Fluß durch die freie Fläche des Ventils (A.FP). Damit wird zwar der Druck und der Fluß ($Q.T2$) am Ausgang erhöht, doch wird dieser Anstieg auch auf die Membran übertragen, die mit dem Ventil verbunden ist. Die Membran drückt also das Ventil gegen die Federkraft nach unten, wodurch die Durchlaßfläche A.FP und damit der Fluß durch das Ventil kleiner wird. Auf diese Weise erhöht sich der Druck am Ausgang des Reglers wesentlich weniger als es ohne die Reglerfunktion der Fall wäre.



Auch Verbindungen sind mit einer Menge von Konfluenzen assoziiert. Im Unterschied zu Komponenten werden die Verbindungskonfluenzen jedoch auf der Basis der gebietsspezifischen Annahmen über die Modellierung (class wide assumptions) vom System automatisch generiert. Eine gebräuchliche Annahme für ein flüssigkeitstransportierendes System besteht z.B. darin, daß alle Verbindungen mit der Flüssigkeit vollkommen ausgefüllt sind, und sich so ein Druck unverzögert fortpflanzen kann. Mit dieser Annahme läßt sich z.B. für die in einem Knotenpunkt K einwirkenden Flüsse Q_1 , Q_2 und Q_3 eines Systems eine Konfluenz erzeugen, die ihre Analogie in Kirchhoff's Gesetz für die in einen Knoten hineinlaufenden Ströme besitzt:



$$Q_1 + Q_2 + Q_3 = 0$$

Die für einen Komponentenzustand charakteristischen Gleichungen und die Wahl der Komponentenzustände selbst kann beliebig erfolgen, ergibt sich aber im allgemeinen aus den physikalischen Gleichungen für das betreffende (Sub-)System. ET spezifiziert eine allgemeine Methode zur Herleitung der qualitativen Gleichungen aus den quantitativen (auch nicht-linearen) Gleichungen. Wir können hierauf nicht detailliert eingehen und beschränken uns deshalb auf einige Beispiele für den Druckregler:

1. Der Massen aspekt des Ventils wird physikalisch beschrieben durch die Gleichung $f = m * b$. Da die Masse stets positiv und konstant ist, ergibt sich in der Vorzeichenalgebra VZ die Konfluenz

$$F = \Delta V$$

wobei V für die Geschwindigkeit der Masse steht und ΔV für ihre Beschleunigung. Die in ET in einer Konfluenz auftretenden Terme sollten möglichst nicht gemischt sein, d.h. ein Term sollte nicht sowohl Grundgrößen als auch abgeleitete Größen enthalten. Gemischte Terme können durch Einführung von Zuständen zu "reinen" Termen reduziert werden:

$$\begin{aligned} F\text{-POS} &: [F > 0], \Delta V = + \\ F\text{-ZERO} &: [F = 0], \Delta V = 0 \\ F\text{-NEG} &: [F < 0], \Delta V = - \end{aligned}$$

2. Das Hooke'sche Gesetz für die Feder läßt sich darstellen als $dF/dt = k * v$. Die positive Konstante k hat qualitativ keinen Einfluß auf die Beziehung, wodurch sich die Konfluenz

$$\Delta F = V$$

und wie in 1. die drei Zustände

$$\begin{aligned} V\text{-POS} &: [V > 0], \Delta F = + \\ V\text{-ZERO} &: [V = 0], \Delta F = 0 \\ V\text{-NEG} &: [V < 0], \Delta F = - \end{aligned}$$

für die Feder ergeben.

3. Der Regelaspekt des Ventils läßt sich durch die gemischte Konfluenz

$$\Delta P - V - \Delta Q = 0$$

ausdrücken. Hierbei ist P der Druck über dem Ventil, Q der Fluß durch das Ventil und V die Geschwindigkeit. Auflösen der gemischten Konfluenz für V und Hinzufügen einer Richtung für den Druck P führt zu neun verschiedenen Verfeinerungen des Zustands WORKING ($0 < A < A-MAX$), wobei A die Durchlaßfläche des Ventils ist). Drei dieser Zustände sind:

WORKING- $V > 0$: [$0 < A < A-MAX$, $P < 0$, $V > 0$], $\Delta P - \Delta Q = +$
 WORKING- $V = 0$: [$0 < A < A-MAX$, $P < 0$, $V = 0$], $\Delta P - \Delta Q = 0$
 WORKING- $V < 0$: [$0 < A < A-MAX$, $P < 0$, $V < 0$], $\Delta P - \Delta Q = -$

B4 Darstellung des Verhaltens eines Systems

Ein globaler Zustand eines Systems ergibt sich durch die Auswahl eines Zustands für jede Komponente des Systems. Sind für ein Modell keine expliziten Zustände definiert, so wird die Existenz eines einzigen Dummy-Zustands angenommen. Die Auswahl der Einzelzustände muß natürlich konsistent sein, z.B. darf nicht für das Ventil ein Zustand mit $V > 0$ und für die Feder der Zustand $V-NEG$ zur gleichen Zeit ausgewählt werden.

B4.1 Das zustandsinterne Verhalten

Ein globaler Zustand definiert implizit eine Menge von Konfluenzen, die für die einzelnen Komponenten und die Verbindungen in dem betrachteten Zustand gültig sind. Ein zustandsinternes Verhalten ist damit eine Variablenbelegung, unter der die Konfluenzen des betrachteten Zustands nicht inkonsistent sind.

Im allgemeinen gibt es aufgrund der qualitativen Beschreibung mehrere verschiedene zustandsinterne Verhaltensweisen (Interpretationen). Die Berechnung aller Interpretationen wird als zustandsinterne Analyse (intrastate analysis) bezeichnet. Das Verfahren zur Berechnung aller Interpretationen (= Lösungen des Gleichungssystems) ist eine konstruktive Constraint-Propagierung mit der Einführung von Annahmen für die Fälle, wo in einer n -stelligen Relation das Ergebnis nicht eindeutig ist oder weniger als $n-1$ Variablen bekannt sind. De Kleer und Brown bezeichnen ihr Verfahren als eine Mischung aus Constraint-Propagierung und Generate-and-Test.

Bild B3 enthält alle Konfluenzen des Druckreglers für abgeleitete Größen für den Zustand charakterisiert durch $P < 0$, $V > 0$, und $F > 0$. Zusätzlich wird in Gleichung (11) als Störfall angegeben, daß sich der Druck am Eingang des Druckreglers erhöht (Störfallanalyse). In Bild B4 werden für jeden globalen Zustand des Druckreglers die möglichen Interpretationen gelistet. Für

$P < 0$, $V > 0$ und $F > 0$ sind beispielsweise zwei Interpretationen möglich, d.h. die Gleichungen in Bild B3 haben zwei Lösungen unter der Vorzeichenalgebra VZ.

Zustandsunabhängige Konfluenzen:

- (1) $\Delta P. OUT, SNK + \Delta F. 1(S) + \Delta F. A(M) = 0$
- (2) $\Delta F. 1(S) + \Delta F. 2(S) = 0$
- (3) $\Delta P. IN, OUT + \Delta P. OUT, SNK - \Delta P. IN, SNK = 0$
- (4) $\Delta Q. T1 + \Delta Q. 1(VV) = 0$
- (5) $\Delta Q. T2 + \Delta Q. 2(VV) = 0$
- (6) $\Delta Q. 1(VV) + \Delta Q. 2(VV) = 0$
- (7) $\Delta Q. T2 - \Delta P. OUT, SNK = 0$

Zustandsabhängige Konfluenzen:

- (8) $\Delta P. IN, OUT - \Delta Q. 1(VV) = +$
- (9) $\Delta F. 1(S) = +$
- (10) $\Delta V. FP = +$

Störfall:

- (11) $\Delta P. IN, OUT = +$

Bild B3: Konfluenzen des Druckreglers für den Zustand $P < 0$, $V > 0$, $F > 0$.

Lösungen für den Druckregler mit $dP_{IN,SNK} = +$

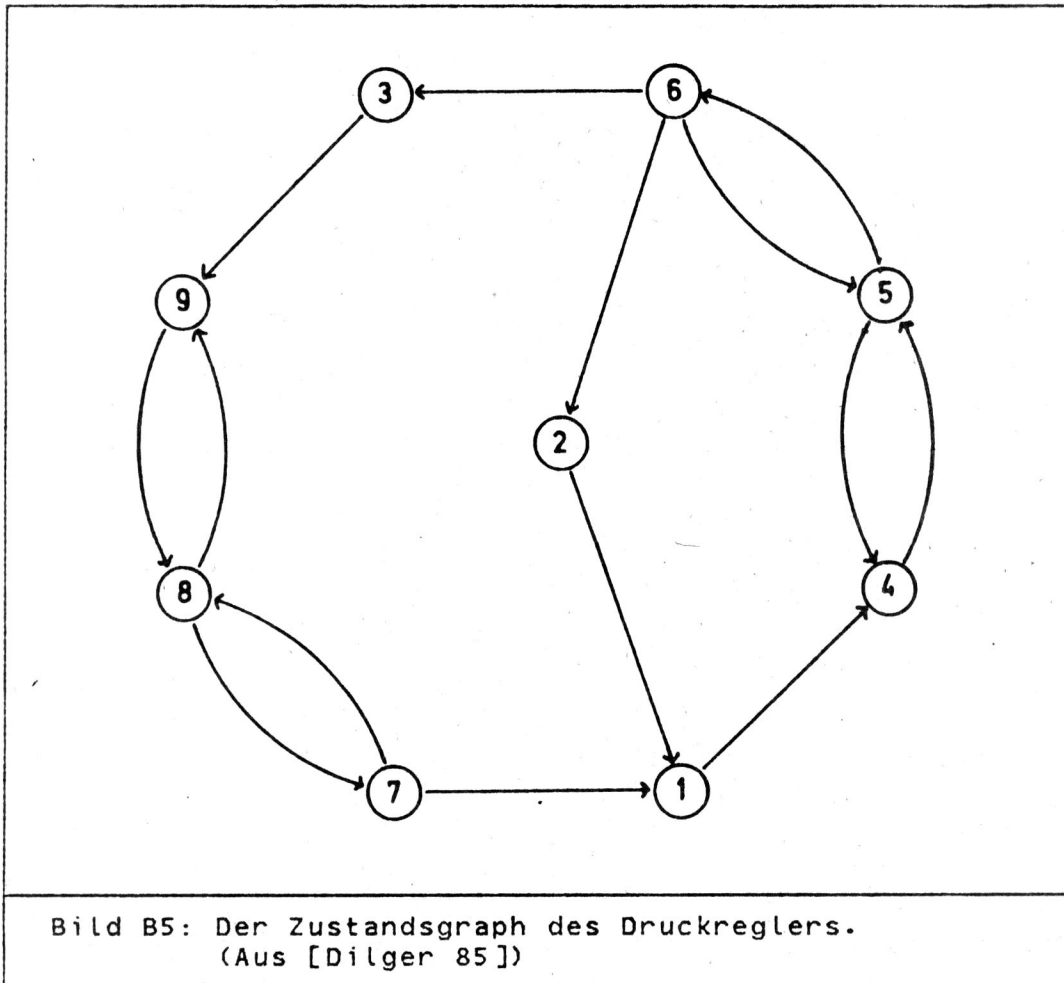
Zustand	1		2		3		4		5		6		7		8		9	
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Zustands- spezifikationen	V=0	F<0	V=0	F=0	V=0	F=0	V<0	F<0	V<0	F=0	V<0	F>0	V>0	F<0	V>0	F=0	V>0	F>0
zulässige Werte- belegungen	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
dQ_{T1}	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
dQ_{T2}	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
$dQ_1(W)$	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
$dQ_2(W)$	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
$dP_{IN,OUT}$	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
$dP_{OUT,SNK}$	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
$dF_1(S)$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$dF_2(S)$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$dF_A(M)$	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
dV_{FP}	-	0	+	-	-	-	-	-	00000000	-	+	+	+	+	+	+	+	+

Bild B4: Die zustandsinternen Verhalten des Druckreglers für seine globalen Zustände mit $P<0$.
(Aus [Dilger 85])

B4.2 Das globale Verhalten

Das globale Verhalten eines Systems ergibt sich aus der Gesamtmenge der zustandsinternen Verhalten durch eine Analyse der möglichen Übergänge von einem zustandsinternen Verhalten des einen Zustands zu einem zustandsinternen Verhalten eines anderen Zustands. Das Ergebnis einer solchen Zustandsänderungsanalyse (interstate analysis) ist ein i.a. nichtdeterministischer Zustandsgraph, dessen Knoten die globalen Zustände des Systems sind. Bild B5 zeigt den Zustandsgraph für die Interpretationen aus Bild B4. Man beachte beim Nachrechnen, daß für verschiedene Zustände auch teilweise unterschiedliche Konfluenzen anzuwenden sind.

Die Analyse des globalen Verhaltens stützt sich auf eine Anzahl von Übergangsregeln, die sämtlich in [De Kleer/Brown 84] dargestellt werden. Viele dieser Regeln beruhen auf der Stetigkeitsannahme für die Variablen. Sowohl Grundgrößen als auch abgeleitete Größen können ihre Werte nur stetig verändern, d.h. eine Variable V kann von $+$ ($-$) nach $-$ ($+$) nur gelangen, wenn sie zwischendurch den Wert 0 annimmt. Zusätzlich kann eine Grundgröße von $+$ ($-$) nach 0 nur gelangen, wenn für ihre Ableitung $\Delta V = -$ ($\Delta V = +$) gilt. In [De Kleer/Brown 84] werden für denselben Zweck auch höhere Ableitungen in Betracht gezogen. Ein Beispiel für eine Regel, die auch Zustände involviert, wurde bereits in Kapitel 11 gegeben.



16. LITERATUR

- [AI-Journal 84] Artificial Intelligence (1984), Vol.24, Numbers 1-3, Special Volume on Qualitative Reasoning about Physical Systems.
- [Allen 83] Allen, James F. : Maintaining Knowledge about Temporal Intervals, CACM, Vol.26, No.11, 832-843.
- [Blum 85] Blum, Norbert : Fehlererkennung in kombinatorischen Schaltkreisen, SFB 124-B2, Bericht 10/1985, Fachbereich 10 der Universität des Saarlandes, Saarbrücken.
- [Charniak/McDermott 85] Charniak, Eugene; McDermott, Drew : Introduction to Artificial Intelligence, Addison Wesley.
- [Davis 82] Davis, Randall: Expert Systems: Where are we? and where do we go from here?, AI-Magazine, Spring 1982, pp. 3-22.
- [Davis 84] Davis, Randall: Diagnostic Reasoning Based on Structure and Behavior, in [AI-Journal 84], pp.347-410.
- [de Kleer 84] de Kleer, Johan : How Circuits Work, in [AI-Journal 84], pp. 205-280.
- [de Kleer 86] de Kleer, Johan : An Assumption-Based TMS, draft version, to appear in Artificial Intelligence, 1986.
- [de Kleer/Bobrow 85] de Kleer, Johan; Bobrow, Daniel G. : Qualitative Reasoning with Higher-Order Derivatives, in Proc. AAAI-84, pp. 86-91.
- [de Kleer/Brown 83] de Kleer, Johan; Brown, John S. : Assumptions and Ambiguities in Mechanistic Mental Models, in D. Gentner/ A. L. Stevens (Eds.): Mental Models, Lawrence Erlbaum Associates.
- [de Kleer/Brown 84] de Kleer, Johan; Brown, John S. : A Qualitative

- Physics Based on Confluences, in [AI-Journal 84], pp 7-83.
- [de Kleer/Williams 85]
de Kleer, Johan; Williams, Brian C.: Diagnosing Multiple Faults, draft of September 85, Intelligent Systems Lab., XEROX Palo Alto Research Center.
- [Descotte/Latombe 85]
Descotte, Y; Latombe, J. C.: Making Compromises among Antagonist Constraints in a Planner, Artificial Intelligence, Vol.27, No.2, pp. 183-217.
- [Dilger 85] Dilger, Werner : Expertensysteme für technische Anwendungen, Manuskript zur gleichnamigen Vorlesung im Wintersemester 1985/86 an der Universität Kaiserslautern.
- [Dilger/Kippe 85]
Dilger, Werner; Kippe, Jörg: COMODEL, a Language for the Representation of Technical Knowledge, Proc. 9th International Conference on Artificial Intelligence, Los Angeles, pp. 353-358.
- [Doyle 79] Doyle, Jon : A Truth Maintenance System, Artificial Intelligence Vol 12 (1979), pp. 231-272.
- [Forbus 84] Forbus, Kenneth D.: Qualitative Process Theory, in [AI-Journal 84], pp. 85-168.
- [Genesereth 84]
Genesereth, Michael R. : The Use of Design Descriptions in Automated Diagnosis, in [AI-Journal 84], pp. 411-436.
- [Gentner/Stevens 83]
Gentner, Dedre; Stevens, A. L. (eds) : Mental Models, Lawrence Erlbaum Associates.
- [Hayes 79] Hayes, Patrick J. : The Naive Physics Manifesto, in Michie, D. (ed), Expert Systems in the Micro-electronic Age, Edinburgh University Press, pp. 242-270.
- [Hayes 85] Hayes, Patrick J. : The Second Naive Physics Manifesto, in Hobbs, Jerry R.; Moore, Robert C. (eds), Formal Theories of the Commonsense World, Ablex Series in Artificial Intelligence, pp. 1-36.
- [Hayes 85] Hayes, Patrick J. : Naive Physics I : Ontology for Liquids, ebenda, pp.71-107.
- [Iwasaki/Simon 85]
Iwasaki, Yumi; Simon, Herbert A. : Causality in Device Behavior, Carnegie-Mellon University, Computer Science Department, CMU-CS-85-118.

- [Klaeren 83] Klaeren, Herbert A. : Algebraische Spezifikationen, eine Einführung, Springer Verlag.
- [Kuipers 84] Kuipers, Benjamin : Commonsense Reasoning about Causality: Deriving Behavior from Structure, in [AI-Journal 84], pp. 169-203.
- [Long 86] Long, W. J.; Naimi, S.; Criscitiello, M. G.; Kurzrok, S.: Reasoning about Therapy from a Physiological Model, MIT Laboratory for Computer Science, submitted to MEDINFO 86.
- [Mason 56] Mason, Samuel J.: Feedback Theory - Further Properties of Signal Graphs, Proc. of the IRE, 1956, 44: 920-926.
- [McCarthy 68] McCarthy, John : Programs with Common Sense, in Minsky, Marvin (ed), Semantic Information Processing, MIT press, pp.403-418. Diese Arbeit enthält Material, das bereits in den Jahren 1958 und 1963 erstmalig veröffentlicht wurde.
- [McCarthy/Hayes 69] McCarthy, John; Hayes, Patrick J.: Some Philosophical Problems from the Standpoint of Artificial Intelligence, in Meltzer, B.; Michie, D. (eds), Machine Intelligence 4, Edinburgh University Press.
- [McDermott 82] McDermott, Drew : A Temporal Logic for Reasoning about Processes and Plans, Cognitive Science 6, pp. 101-155.
- [Patil 81] Patil, Ramesh S.: Causal Representation of Patient Illness for Electrolyte and Acid-Base Diagnosis, MIT/LCS/TR-267, Massachusetts Institute of Technology.
- [Pople 82] Pople, H.: Heuristic Methods for Imposing Structure on Ill-structured Problems, in Szolovits, P. (ed.), Artificial Intelligence in Medicine, AAAS Selected Symposium 51.
- [Raulefs 84] Raulefs, Peter : Foundations of Expert Systems for Conceptual Design in Mechanical Engineering, Memo SEKI-84-08, Fachbereich Informatik, Universität Kaiserslautern.
- [Reinfrank 85a] Reinfrank, Michael Th. : An Introduction to Non-Monotonic Reasoning, Memo SEKI-85-02, Fachbereich Informatik, Universität Kaiserslautern.

- [Reinfrank 85b] Reinfrank, Michael Th. : SCENELAB, Scene Labelling by a Society of Agents. A Distributed Constraint Propagation System., Memo SEKI-85-06, Fachbereich Informatik, Universität Kaiserslautern.
- [Rieger/Grinberg 77] Rieger, Chuck; Grinberg, Milt : The Declarative Representation and Procedural Simulation of Causality in Physical Mechanisms, Proc. 5th International Joint Conference on Artificial Intelligence, pp.250-256.
- [Roth 66] Roth, J. P.: Diagnosis of Automata Failures: A Calculus and a Method, IBM Research Development 10, pp.278-291.
- [Schwind 85] Schwind, Camilla B.: Temporal Logic in Artificial Intelligence, in Proc. GWAI-84, 8th German Workshop on Artificial Intelligence, Wingst/Stade.
- [Shirley/Davis 83] Shirley, M.; Davis, R.: Digital Test Generation from Hierarchical Models and Symptom Information, Proc. IEEE International Conference on Computer Design.
- [Silverberg 81] Silverberg, Brad A., An Overview of the SRI Hierarchical Development Methodology, Horst Hünke (ed), Software Engineering Environments, North Holland.
- [Stallman/Sussman 77] Stallman, R. M.; Sussman, G. J.: Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis, Artificial Intelligence Vol.9, pp.135-196.
- [Sussman/Steele 80] Sussman, Gerald; Steele, Guy Lewis Jr.: CONSTRAINTS - a Language for Expressing Almost-Hierarchical Descriptions, Artificial Intelligence 14, pp. 1-40.
- [Steele 80] Steele, Guy Lewis Jr.: The Definition and Implementation of a Computer Programming Language Based on Constraints, MIT AI-Lab., AI-TR-595.
- [Voss 86a] Voss, Hans: Representing and Analyzing Time and Causality in HIQUAL Models, Memo SEKI-85-07, Fachbereich Informatik, Universität Kaiserslautern, erscheint 1986 in Proc. 9th German Workshop on Artificial Intelligence, Springer Verlag.
- [Voss 86b] Voss, Hans: Constraints - Probleme, Methoden, Anwendungen., in Vorbereitung.

[Waltz 75] Waltz, David: Understanding Line Drawings of Scenes with Shadows, in 'The Psychology of Computer Vision', P. H. Winston (ed.), McGraw-Hill Book Company 1975, pp. 19-91.

[Williams 84] Williams, Brian C.: Qualitative Analysis of MOS Circuits, in [AI-Journal 84], pp. 281-346.