

Handwritten notes at the top left of the page, including the number '1' and some illegible text.

Handwritten notes at the top right of the page, including the number '1', 'Luzerne', 'Kanton', and other illegible text.

First column of handwritten notes, containing several lines of text.

Second column of handwritten notes, continuing the text from the first column.

Third column of handwritten notes, continuing the text from the first column.

Fourth column of handwritten notes, continuing the text from the first column.

Fifth column of handwritten notes, continuing the text from the first column.

Sixth column of handwritten notes, continuing the text from the first column.

First column of printed text in the right-hand section, containing several lines of a letter or document.

Second column of printed text in the right-hand section, starting with 'Die werbe...'.

Third column of printed text in the right-hand section, starting with 'Die werbe...'.

Fourth column of printed text in the right-hand section, starting with 'Die werbe...'.

Fifth column of printed text in the right-hand section, starting with 'Die werbe...'.

Sixth column of printed text in the right-hand section, starting with 'Die werbe...'.

Signature and address block of the sender, including 'Prof. Dr. J. L. ...' and '3000 ...'.

Text 'angefordert werden.' located below the signature block.

Text 'Luzerne' located at the bottom right of the page.

5.2 BERICHT VOM "LOGIC PROGRAMMING MEETING"

Auf Initiative von R. ("Bob") Kowalski fand vom 24. bis 28.5.1976 am Imperial College, London, ein Arbeitstreffen statt, das (fast) ausschließlich der "Prädikatenlogik als Programmiersprache" gewidmet war. Insbesondere stand die Programmiersprache PROLOG im Mittelpunkt des Interesses; jedoch wurden auch vergleichbare Ansätze wie GOLUX, LUCID u. a. diskutiert.

Der Entwicklungsstand dieser Systeme und die Erfahrungen aus den praktischen Anwendungen begründeten den ungetrübten Optimismus der mehr als 40 Teilnehmer aus 11 Ländern und sollten vielleicht manchem Skeptiker zu denken geben.

Die ca. 50 Vorträge verteilten sich im wesentlichen auf die folgenden thematischen Schwerpunkte:

Einführung in PROLOG (Roussel) und ihre praktische Anwendung zum

- Lösen von Gleichungen (Welham, Bundy)
- Problemlösen in der Geometrie (Welham, Pereira)
- Austesten der Komponenten einer Telefonschaltstelle des GPO (Hill).

Es wurde ausführlich über das Problem expliziter Kontrollelemente in prädikativen Programmen diskutiert: wenn sich der automatische Beweiser zum Prädikatenkalkül verhält wie LISP-Eval zu LISP, dann entspricht (nicht: ist gleich!) der Prädikatenkalkül dem 'pure' LISP ohne expliziten Einfluß auf den Kontrollfluß - und niemand würde, obwohl theoretisch möglich, komplizierte Programme so schreiben wollen (Hayes). Die vorgeschlagenen Mittel: 'slash', 'ajou' etc. stehen jedoch im Gegensatz zu einem der Hauptargumente für das prädikative Programmieren: sie verletzen teilweise die klare Semantik solcher Programme (Colmerauer, Roussel, Warren).

Implementierungen

- Hier wurden viele Details der verschiedenen Implementierungen in großer Ausführlichkeit diskutiert, insbesondere auch solche der zugrundeliegenden Beweisverfahren (Warren, Roussel, Bruynooghe).

Weiterentwicklungen

Für die Darstellung vieler Probleme ist die in PROLOG verwendete Clausenform eine zu starke Einschränkung. Die Einbeziehung von Clausen, die nicht vom Horn'schen Typ sind, ist ein erster Schritt (Hogger) in die Richtung der vollen Ausnutzung der Ausdruckskraft der Prädikatenlogik. In diesem Rahmen wurde gezeigt, wie das Kontroll- oder Steuerproblem für viele Probleme durch automatische Transformation in effizienten Code gelöst werden kann und allgemeinere Problemstellungen auf Darstellungen dieser Art zurückgeführt werden können (Dibel).

Die Möglichkeit, für den Beweiser nachteilige Axiome aus der Datenbasis herauszunehmen und in den Unifikationsalgorithmus hinein-zubauen ist bekannt: gewisse solche Axiome definieren jedoch übliche Datentypen und das Unifikationsproblem kann als Matchingproblem, wie es in verschiedenen AI-Sprachen auftritt, angesehen werden und so die Methodologie des automatischen Beweisens auf dieses übertragen werden (Siekmann).

'Alle erfolgreichen automatischen Beweiser benutzen Ersetzungsregeln (rewrite rules) in der einen oder anderen Form' war die von F. Brown vorgebrachte These: Kann ein Beweiser allein auf rewrite rules aufgebaut werden (er kann, wie F. Brown's Programm zeigt) oder wie weit lassen sie sich mit den bisherigen Deduktionsmechanismen vereinbaren (Vollständigkeit und Entscheidbarkeit)?

Im Hinblick auf den Benutzer ist das entscheidende Ziel die Einbeziehung natürlicher Sprechformulierungen. Im Hinblick darauf wurde eine interessante Formalisierung der natürlich-sprachlichen Grammatik mittels λ -Termen in der Logik 2. Stufe gegeben (Colmerauer). Ein anderer Ansatz verwendete semantische Netze (Delyanni).

Eine grundsätzliche Gegenüberstellung der verschiedenen "Schulen" verhalf zum besseren gegenseitigen Verständnis (Hayes).

Datenbanken

Die durch "relational data bases" charakterisierte Richtung auf diesem Gebiet steht dem prädikativen Programmieren sehr nahe. So kam es zu einem interessanten Meinungsaustausch zwischen den jeweiligen Spezialisten (eine Gruppe von ICL, Hitchcock/IBM bzw. Kowalski, Nicolas/Gallaire).

Weitere Vorträge befaßten sich mit

- Datenstrukturen (Clark)
 - speziellen Beweisverfahren (Hayes, Eder, Amble)
 - Unifizierung von Substitutionen (v. Vaalen)
 - Termlogik (Tärnlund)
 - der Sprache LUCID (Wadga)
- u. a. m.

Leider kann ein so kurzer Bericht nur wenig von dem allgemeinen Eindruck wiedergeben, den dieses Treffen hinterlassen hat: nämlich an einer erfolgreichen Tagung über etwas, das 'im Kommen ist' teilgenommen zu haben.

Eines muß abschließend noch erwähnt sein: nämlich die außergewöhnliche Gastfreundschaft unserer englischen Freunde; many thanks to Bob, Jaqueline and all others.

Wolfgang Eitel und Jörg Siekmann

5.2. Rückblick

- 1. Pattern-Matching-Verfahren für die Verarbeitung komplexer Datenstrukturen (zur Vorbildstellung meiner selbst ca. 1/2 Jahr laufendes Experiment)

In meiner Promotion versuchte ich, drei aufeinander aufbauende Problemgebiete, die hier aus meiner A-Forschung ergeben, mit Methoden und Formalismen der Informatik zu bearbeiten. Allgemeines Ziel ist, die hier zu ergebenden prozeduralen Theorien auch eine präzise, elegante Theorie zu schaffen, so das theoretische Ergebnisse (d.h. nicht eine Implementation) direkt in der Praxis verwendet werden können.

Die genannten Problemkreise lassen sich in der Form von Zielen darstellen, welche am besten lehrweise (d.h. bis ich in der Mitte des zweiten Durchlaufs, der die Form eines Zwischenberichts annehmen wird, den ich stetigen Interessierten auf Anfrage zuschicken könnte):

- 1) Einheitliche Formalismen für die wichtigsten komplexen Datenstrukturen (KDS), die bisher zugunsten von Programmstrukturen vernachlässigt wurden, aber zusammen mit der Repräsentation von Wissen an Bedeutung gewinnen.
- 2) Formalisierung von (KDS-)Algorithmen und (KDS-)Pattern-Matching, die in der KI schon seit längerem informell benutzt werden und heute v.a. in den PLANKartigen Sprachen eine fundamentale Rolle spielen.
- 3) Formale(s) System(e) zur Pattern-Matching-Verarbeitung von KDS, welche(s) die wesentlichen Charakteristika PLANKartiger Sprachen formalisieren und als mitwirts very-high-level Programmiersprache(n) verwendbar sein sollte(n), die den Teilziel einer gemeinsamen Sprache für Theoretische und Praktische Informatik.

Harold Healy, Institut für Informatik Universität Hamburg

- 2. Als Institutsbericht Nr. 27 ist erschienen:

"Formation of an Object Concept by Analysis of Systematic Time Variations in the Optically Perceptible Environment"

An algorithm to isolate a representation for a moving object of unknown form and size from a sequence of TV frames is described, together with results of its application to real world scenes. It is based on the assumption that a group of connected regions being jointly displaced in a systematic way from frame to frame without changing their relative position represents the image of a moving object. These assumptions relate to properties which are to a large extent independent from specific scenes and thus may appear as components in a general object concept. Implications of this approach for the conception of scene analyzing systems that "learn from observation" are discussed.

Hans-Hellmut Nagel
Institut für Informatik
Universität Hamburg

6.4. Zuschriften

- 1. Prof. Schöne/Lomförde weist unter Bezugnahme auf die Zuschrift von R. Baker (Rundbrief Nr. 5) auf folgenden in der Deutschen Schachzeitung erschienenen Artikel hin:

Dr. Kurt Fischer/Stuttgart: Wie ein Computer Schach spielt, Deutsche Schachzeitung, 21. Jahrgang, Nr. 3, März 1972, S. 99 ff. und Nr. 5, Mai 1972, S. 165 ff.

6.2. Adressänderungen

Dr. Peter Eisenberg, TU Hannover, Germanisches Seminar
Jörg Siekmann, Universität Karlsruhe, Institut für Informatik 1
Das Institut für Informatik in Stuttgart hat die Adresse geändert, und zwar sind zu erreichen

Agenbergstraße 12:
Rokhara Falckenberg
Dr. A. Hanakata
Dr. J. Leusch

Herdweg 51:
Dr. W. Glatthair
R. Kurz

6.6 "Dialoge in natürlicher Sprache und Darstellung von Wissen"

Eine beschränkte Anzahl von Exemplaren der Beiträge zu dieser Tagung in Freudenstadt vom 17. - 19. März 1976 können für DM 10,-- pro Exemplar angefordert werden bei

Dr. J. Laubsch
Universität Stuttgart
Institut für Informatik
Azenbergstr. 12
7000 Stuttgart 1

6.7 AISB Summer Conference 1976

Eine vorläufige Liste der Beitragsthemen ist im 5. Rundbrief veröffentlicht worden.

Proceedings of AISB Summer Conference are available from Mrs. J. Lee, Department of Artificial Intelligence, Forrest Hill, Edinburgh EH1 2QL. £7.50 (post paid) - limited stocks, soft covers.

Herrn Dr. W. Bibel
Institut für Informatik