DEDUCTION SYSTEMS IN ARTIFICIAL INTELLIGENCE

Editors:

KARL HANS BLÄSIUS

HANS-JÜRGEN BÜRCKERT

ELLIS HORWOOD SERIES IN ARTIFICIAL INTELLIGENCE

Joint Series Editors: Professor JOHN CAMPBELL, Department of Computer Science, University College London, and
Dr JEAN HAYES MICHIE, Knowledgelink, The Turing Institute, Glasgow

Andrew, A.M.	CONTINUOUS HEURISTICS*
Attar, A.	KNOWLEDGE ENGINEERING*
Bläsius, K.H. and Bürckert, H.-J	DEDUCTION SYSTEMS IN AI*
Bramer, M.A. (editor)	COMPUTER GAME PLAYING: Theory and Practice
Campbell, J.A. (editor)	IMPLEMENTATIONS OF PROLOG
Campbell, J.A. and Cuena, J. (editors)	PERSPECTIVES IN ARTIFICIAL INTELLIGENCE, Vols. 1 & 2
Campbell, J. A. & Cox, P.	IMPLEMENTATIONS OF PROLOG, Vol. 2*
Carlini, U.	TRANSPUTERS AND PARALLEL ARCHITECTURES: Message-Passing Distributed Systems*
Carter, D.	INTERPRETING ANAPHORS IN NATURAL LANGUAGE TEXTS
Colomb, R.	IMPLEMENTING PERSISTENT PROLOG
Davies, R. (editor)	INTELLIGENT INFORMATION SYSTEMS: Progress and Prospects
Evans, J.B.	STRUCTURES OF DISCRETE EVENT SIMULATION
Farreny, H.	AI AND EXPERTISE
Forsyth, R. & Rada, R.	MACHINE LEARNING: Applications in Expert Systems and Information Retrieval
Frixione, S.G.	REPRESENTING CONCEPTS IN SEMANTIC NETS
Futó, I. & Gergely, T.	ARTIFICIAL INTELLIGENCE IN SIMULATION
Gabbay, D.M.	PROGRAMMING IN PURE LOGIC*
Hawley, R. (editor)	ARTIFICIAL INTELLIGENCE PROGRAMMING ENVIRONMENTS
Hayes, J.E. & Michie, D. (editors)	INTELLIGENT SYSTEMS: The Unprecedented Opportunity
Koskimies, K. *et al.*	AUTOMATING LANGUAGE IMPLEMENTATION*
Lopez de Mantaras, R.	APPROXIMATE REASONING AND KNOWLEDGE ACQUISITION*
Łukaszewicz, W.	NONMONOTONIC REASONING*
McGraw, K. & Westphal, C.	READINGS IN KNOWLEDGE ACQUISITION: Current Practices and Trends*
Mellish, C.	COMPUTER INTERPRETATION OF NATURAL LANGUAGE DESCRIPTIONS
Michie, D.	ON MACHINE INTELLIGENCE, Second Edition
Mortimer, H.	THE LOGIC OF INDUCTION
Mozetic, I.	MACHINE LEARNING OF QUALITATIVE MODELS*
Obermeier, K.K.	NATURAL LANGUAGE PROCESSING TECHNOLOGIES IN ARTIFICIAL INTELLIGENCE
Partridge, D.	ARTIFICIAL INTELLIGENCE: Applications in the Future of Software Engineering
Ramsay, A. & Barrett, R.	AI IN PRACTICE: Examples in POP-11
Savory, S.E.	ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS
Shanahan, M.	EXPLOITING DEPENDENCIES IN SEARCH AND INFERENCE MECHANISMS*
Spacek, L.	ADVANCED PROGRAMMING IN PROLOG*
Sparck Jones, K. & Wilks, Y. (editors)	AUTOMATIC NATURAL LANGUAGE PARSING
Steels, L. & Campbell, J.A. (editors)	PROGRESS IN ARTIFICIAL INTELLIGENCE
Smith, B. & Kelleher, G. (editors)	REASON MAINTENANCE SYSTEMS AND THEIR APPLICATIONS
Szpakovic, S.	LOGIC AND LOGIC GRAMMARS FOR LANGUAGE PROCESSING*
Torrance, S. (editor)	THE MIND AND THE MACHINE
Turner, R.	LOGICS FOR ARTIFICIAL INTELLIGENCE
Wallace, M.	COMMUNICATING WITH DATABASES IN NATURAL LANGUAGE
Wertz, H.	AUTOMATIC CORRECTION AND IMPROVEMENT OF PROGRAMS
Yazdani, M. (editor)	NEW HORIZONS IN EDUCATIONAL COMPUTING
Yazdani, M. & Narayanan, A. (editors)	ARTIFICIAL INTELLIGENCE: Human Effects
Zeindenberg, M.	NEURAL NETWORK MODELS FOR AI AND COGNITION*

* *In preparation*

DEDUCTION SYSTEMS IN ARTIFICIAL INTELLIGENCE

Editors;

KARL HANS BLÄSIUS

IBM Deutschland GmbH, Stuttgart, FRG

HANS-JÜRGEN BÜRCKERT

Fachbereich Informatik, Universitat Kaiserslautern, FRG

ELLIS HORWOOD LIMITED

Publishers · Chichester

Halsted Press: a division of

JOHN WILEY & SONS

New York · Chichester · Brisbane · Toronto

This English edition first published in 1989 by
ELLIS HORWOOD LIMITED
Market Cross House, Cooper Street,
Chichester, West Sussex, PO19 1EB, England
The publisher's colophon is reproduced from James Gillison's drawing of the ancient Market Cross, Chichester.

Distributors:
Australia and New Zealand:
JACARANDA WILEY LIMITED
GPO Box 859, Brisbane, Queensland 4001, Australia
Canada:
JOHN WILEY & SONS CANADA LIMITED
22 Worcester Road, Rexdale, Ontario, Canada
Europe and Africa:
JOHN WILEY & SONS LIMITED
Baffins Lane, Chichester, West Sussex, England
North and South America and the rest of the world:
Halsted Press: a division of
JOHN WILEY & SONS
605 Third Avenue, New York, NY 10158, USA

This English edition is translated from the original German edition *Deduktionssysteme — Automatisierung des logischen Kenkens,* in 1987 by R. Oldenbourgh Verlag, GmbH, © the copyright holders.

Translated by Patricia Sarach

British Library Cataloguing in Publication Data
Bläsius, K.H. and Bürckert, H-J.
Deduction systems in artificial intelligence. —
(Ellis Horwood Series in Artificial Intelligence).

CIP catalogue record for this book is available from the British Library

Library of Congress data available

ISBN 0-7458-0409-8 (Ellis Horwood Limited)
ISBN 0-470-21550-X (Halsted Press)

Printed in Great Britain by Hartnolls, Bodmin

Contents

PREFACE

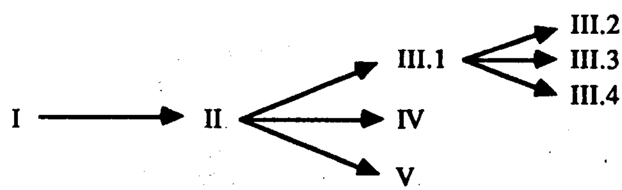
The ability to draw logical conclusions is of fundamental importance to intelligent behavior. For this reason, deduction components are an integral part of numerous artificial intelligence systems. Even though the original motivation for developing these systems was to automatically prove mathematical theorems, their applications now go far beyond. Logic programming languages such as PROLOG have been developed from deduction systems, and these systems are used within natural language systems and expert systems as well as in intelligent robot control. In addition, this field's logic-oriented methods have influenced the basic research of almost all areas of artificial intelligence.

In the past, scientists as well as users from different areas of AI have repeatedly noted the need for introductory material on deduction systems. Such suggestions, voiced most frequently at AI conferences, led us to work out a tutorial on deduction systems for the 1987 GWAI (German Workshop on Artificial Intelligence) and to publish it as a book containing introductory papers on this subject.

This book was written for students, scientists, and users interested in artificial intelligence. It was intended to give the reader an easily understandable yet comprehensive and up-to-date view of deduction systems. Fundamental terms and methods used in these systems will be introduced and explained using many examples and diagrams. The reader will find a certain familiarity with predicate logic useful. In spite of its introductory nature, in many cases this book does represent the current state of research and thus points out new and interesting aspects even to the expert in this field.

The book has been divided into five chapters. The first chapter gives a review of the history of automated deduction and points out several applications of deduction systems. Chapter two makes up the core of this book: here the fundamental methods of this field are explained with the aid of numerous examples. The authors base this chapter on a four level model of an automated deduction system and discuss relevant logics, important calculi, forms of representation, and control mechanisms. Chapter three deals with equality reasoning. Contributions on general equality reasoning, unification theory, and rewriting systems point out the importance of equality and resulting problems. The fourth chapter deals with foundations of logic programming, and chapter five provides an introduction to automated theorem proving using complete induction.

Because of the relative independence of the contents of chapter III, IV, and V, these can be read in any order. The same is true for sections 2, 3, and 4 of chapter III. The following diagram can be used for orientation:



In Germany, automated deduction has had its own tradition. Even before the publication of the resolution principle in 1965, a team based in Bonn, led by G. Veenker, worked on automated theorem proving methods. W. Bibel has been leading an internationally known team in Munich since the early 1970's. Roughly ten years ago, the "Markgraf Karl Group" was founded in Karlsruhe and later, along with J. H. Siekmann, moved to Kaiserslautern. The authors of the papers collected in this book are or were members of this research team and have worked on developing the Markgraf Karl System, which is known as one of the most powerful deduction systems existing today.

We would like to thank the authors for making this book possible. With their joint contribution, Norbert Eisinger and Hans Jürgen Ohlbach established the basis for the remaining papers and thus for the book as a whole. In addition, their numerous suggestions made the comparatively uniform representation of the different papers possible. Jörg H. Siekmann was willing to provide the necessary historical framework and a review of different applications of deduction systems in the introductory chapter; for this and his helpful criticisms of the individual papers we sincerely thank him. The authors and editors would also like to thank Susanne Biundo, Peter Borst, Richard Göbel, Birgit Hummel, Manfred Kerber, Achim Posegga, Robert Rehbold, Wolfgang Reif, Michael M. Richter, Ingrid Walter, Christoph Walther, and Martin Weigele, who all read parts of the manuscripts and supplied important suggestions.

Konstanz and Kaiserslautern, July 1987

Karl-Hans Bläsius and Hans-Jürgen Bürckert

CHAPTER ONE

THE HISTORY OF DEDUCTION SYSTEMS AND SOME APPLICATIONS

(Jörg H. Siekmann)

1. INTRODUCTION

The development of the first mechanical calculating machines by W. Schickard (1623), B. Pascal (1642), and G. W. Leibniz (1671) in the 17th century is generally seen as a milestone of technology. A reconstruction of the very first machine built by W. Schickard in 1623 in Tübingen is exhibited at our computer science department and draws new admirers from professional as well as students every year. What makes these machines so attractive and what is their historical significance? Neither the mathematical principles they are based on, nor the craftsmanship necessary to build such a machine are exceptional compared to other accomplishments of that century. Yet we know more about the history and development of these machines than about other, academically possibly more impressive contributions of the time. What is it, then, that makes them so outstanding?

These early pieces of evidence from the long history of mechanizing human thought [McC 79] had two important prerequisites for their realization: For one thing, the craftmanship to manufacture mechanical instruments and clocks were already advanced enough in Tübingen Paris, Nürnberg and other places such that the technical realization of such a computing machine did not present a problem too large to overcome. Secondly, computation with numbers, which among the Romans for example had been an art mastered only by a few select specialists, had been systemized to a degree that it had become a comparatively easy and mechanical task even for a layman.

The development of today's deduction systems had two similar prerequisites: the mchanical realization of fast electronic computers on the one hand, and the invention of a logical calculus, i.e. the systemization of the important human ability to draw deductive conclusions form given premises, on the other. The idea to join these two seemingly unrelated developments was at least as historically significant as the early calculating machines. Its consequences can be seen in the field of artificial intelligence (AI), for which automated deduction became a foundational discipline, as well as within traditional computer science: programming and computing not only rely on logic as their foundational science – as anticipated by J. McCarthy in the 1960's – they **are** logical deduction. Using R. Kowalski's famous equation, this can be expressed as:

COMPUTATION = DEDUCTION + CONTROL

This insight has radically changed our view of the computer and of the nature of computation [Kow 79, HS 85, FGCS 84], and is considered by many – because of its scientific and technical consequences – as one of the great discoveries of our century.

While the development of electronic computers from Zuse's Z1 in 1936 to today's LISP machines (lambda calculus) or the fifth generation machines (Horn logic) is probably familiar to most of the readers of this book, the second historical prerequisite, namely the development of a mechanizable logical calculus, has not received the amount of attention it deserves.

2. THE EARLY HISTORY

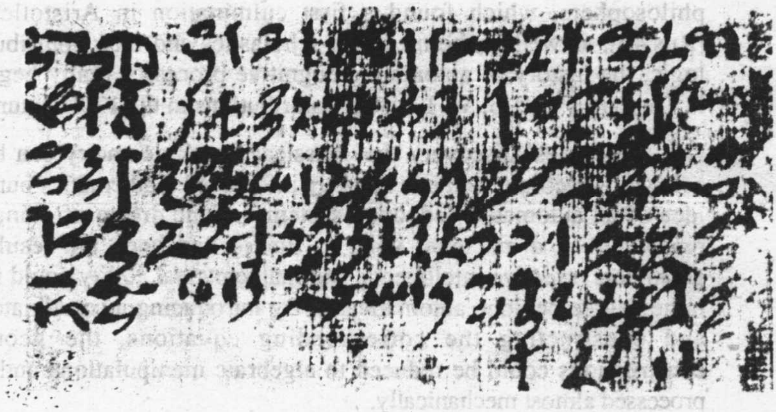
If we ignore for the moment the early logic-oriented studies of Greek philosophers, which found a first culmination in Aristotle's work [Rus 46], as well as the medieval scholastics and their contributions to logic, the history of automating cognitive processes really begins with R. Descartes' and G. W. Leibniz' contributions in the 17th century.

Descartes' discovery that classical Greek geometry can be based solely on algebraic methods not only influenced mathematics but also the idea of an automated deduction system, i.e. the dream of doing logical reasoning on a machine: what to Euclid had been the result of the geometers' creative intelligence and mathematical ability could now to a large degree be done automatically. By introducing a coordinate system and constructing the corresponding equations, the geometrical constructions could be reduced to algebraic manipulations and then be processed almost mechanically.

Descartes was aware of this aspect of his work and put much emphasis on it: "... it is possible to construct all the problems of ordinary geometry by doing no more than the little [namely the four operations suggested by him] covered in the four figures that I have explained. This is one thing which I believe the ancients did not notice, for otherwise they would not have put so much labor into writing so many books in which the very sequence of the propositions showed that they did not have a sure method of finding all..." [Des 37]. His belief, strongly rejected by many mathematicians of his time, was that thus geometry had lost its appeal to the creative mathematician, since it had become mechanizable.

What R. Descartes had done for Euclidian geometry, G. W. Leibniz hoped to extend to the entire area of human thought. Influençed by ideas of the "Ars Magna" of Ramonus Lullus in the 14th century, he suggested a long-range research project (as a note to our funding agency: it took more than 300 years till its first completion): to develop a universal formal language, the "lingua characteristica", in which any proposition could be formulated, as well as a calculus to go with it, the "calculus ratiocinator".

Just as it is possible to translate the following Hau-problem (1800 BC) from one natural language into another, e.g. from demotic characters:



into possibly easier to read hieroglyphs:

or into medieval German:

> *Form der Berechnung eines Haufens, gerechnet anderthalb mal zusammen mit vier. Er ist gekommen bis zehn. Der Haufe nun nennt sich?*

or, for the convenience of our anglo-saxon readers into:

> *Computation of a heap, taken half as much again together with four. It came unto ten. Knowest thou ye heap?*

one can translate it into a **formal language** more common today:

$$1.5 \cdot x + 4 = 10$$

In this formal language we can compute using only the syntactical operations of a calculus, in this case e.g.

$$y + a = b \Rightarrow y = b - a \quad \text{and} \quad c \cdot z = d \Rightarrow z = \frac{d}{c}$$

In the same manner, Leibniz wanted to translate natural language descriptions also of nonmathematical facts into a formal language ("lingua characteristica") and an appropriate calculus ("calculus ratiocinator").

Even though Leibniz' technical contributions to this research project are considered minor from today's point of view, his conception of the long-range objective and the significance of the project turned out to be most influential. He argued that, in comparison to the Greek mathematicians, who obtained immortality by studying mathematical laws inherent in geometrical bodies even though these do not occur in nature and are of little use in practical applications, how much more important would a mathematical genius be, who was able to study the logical laws inherent in human thought? Leibniz' hope was that such a calculus, once established, would be mechanizable just as the calculating machines for arithmetic developed at his time, and that thus humans could be spared all boring intellectual tasks. "The intellect is freed of all conception of the objects involved, and yet the computation yields the correct result." Leibniz' idea of such a universal language and a corresponding calculus climaxes with the touching description of two people of good will engaged in a philosophical dispute who, in search for the truth, translate their arguments into the "lingua characteristica" and, instead of arguing like two philosophers would, act like two computer scientists and say: "Calculemus – let's just compute it" [Ger 90].[1]

A true fragment of the calculus in the sense of Leibniz was not developed until 200 years later by A. de Morgan and G. Boole and is familiar to most of us as **propositional logic** since our school days. This Boolean algebra, originally interpreted as referring to sets or properties, has, as Boole well recognized, the propositional logic interpretation common today. Boole viewed his algebra as a continuation of Leibniz' programme. The extent to which Boole's works tried to mechanize logical calculations is well demonstrated by a machine developed by the economist and logician Stanley Javins in 1869 to evaluate

[1] "Quo facto, quando orientur controversiae, non magis disputatione opus erit inter duos philosophos, quam inter duos computistas. Sufficiet enim calamos in manus sumere sedereque ad abacos, et sibi mutuo (accito si placet amico) dicere: calculemus."

Boolean expressions – a machine not unlike the cash registers found in offices during the last century.

Perhaps the most important contribution to the development of a logical calculus, upon which deduction systems of today are based, was made by Gottlob Frege in his book "Begriffsschrift", which he saw as the fulfillment and elaboration of Leibniz' research program (restricted to the area of mathematics). To E. Schröder's [Sch 82] accusation of not sufficiently taking Boole's work into consideration, Frege answered: "Ich wollte nicht (wie Boole) eine abstracte Logik in Formeln darstellen, sondern einen Inhalt durch geschriebene Zeichen in genauerer und übersichtlicherer Weise zum Ausdruck bringen, als es durch Worte möglich ist. Ich wollte in der That nicht einen blossen "calculus rationator", sondern eine "lingua characterica" [sic] im Leibnizschen Sinne schaffen, wobei ich jene schlussfolgernde Rechnung immerhin als einen nothwendigen Bestandteil einer Begriffsschrift anerkenne." [Fre 82].[2]

Frege's "Begriffsschrift" contains the first complete development of that part of mathematical logic we today call **first-order predicate calculus**. Its enormous significance becomes apparent when we compare it with the degree of confusion about the different styles and interpretations given to a mathematical logic by Frege's fellow and predecessor mathematicians – in particular, the discussion of whether the set theoretical interpretation or a propositional interpretation of the Boolean algebra was to be seen as primary. In Frege's "Begriffsschrift", the propositional logic interpretation is assumed to be fundamental and is the basis for all further constructions, including the usage of quantifiers. In particular, it established the essential **functional** structure of logic, as it is now common to all of us and developed the basic principles by which today's logic textbooks introduce the logical connectives, quantifiers, and relations. Thus, the original programme of Leibniz, to reduce logical human thought to a purely syntactical operation had, to an important extent, been realized and completed.

The development and formal construction of a logical calculus is not the only reason for the relevance of the "Begriffsschrift": it is moreover the methodological contribution by which the syntax and semantics of a formal

---

[2] "Unlike Boole, I did not want to represent an abstract logic using formulas; instead I wanted to express a content in formal symbols in a more clearly and precise manner than it is possible using words. In fact, I did not want to create just a "calculus ratiocinator" but a "lingua characterica" [sic] in the sense Leibniz anticipated, indeed recognizing that deductive computation is a necessary component of a "Begriffsschrift".