Ellis Horwood Series in **ARTIFICIAL INTELLIGENCE**

REASON MAINTENANCE SYSTEMS AND THEIR APPLICATIONS

Editors
Barbara Smith and Gerald Kelleher

REASON MAINTENANCE SYSTEMS AND THEIR APPLICATIONS

Editors:
BARBARA SMITH
School of Computer Studies, University of Leeds
and
GERALD KELLEHER
Computer Based Learning Unit, University of Leeds

ELLIS HORWOOD LIMITED
Publishers · Chichester

Halsted Press: a division of
JOHN WILEY & SONS
New York · Chichester · Brisbane · Toronto

First published in 1988 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
South-East Asia
JOHN WILEY & SONS (SEA) PTE LIMITED
37 Jalan Pemimpin # 05–04
Block B, Union Industrial Building, Singapore 2057
Indian Subcontinent
WILEY EASTERN LIMITED
4835/24 Ansari Road
Daryaganj, New Delhi 110002, India

British Library Cataloguing in Publication Data
Reason maintenance systems and their applications.
1. Artificial intelligence
I. Smith, Barbara, *1948–* II. Kelleher, Gerald
006.3

Library of Congress CIP data available

ISBN 0–7458–0482–9 (Ellis Horwood Limited)
ISBN 0–470–21367–1 (Halsted Press)

Printed in Great Britain by Hartnolls, Bodmin

Ellis Horwood Series in
Artificial Intelligence

Joint Series Editors: Professor JOHN CAMPBELL, Department of Computer Science, University College London, and JEAN HAYES-MICHIE, Knowledgelink Limited, Edinburgh

REASON MAINTENANCE SYSTEMS AND THEIR APPLICATIONS

Editors: BARBARA SMITH, School of Computer Studies, and GERALD KELLEHER, Computer Based Learning Unit, University of Leeds

This book brings together researchers and workers active in the field of Reason Maintenance Systems (RMS), and is designed to foster the exchange of ideas and new developments in this fast-growing area of artificial intelligence. The chapters are based on papers presented at a workshop sponsored by the SSAISB. They have been carefully selected and edited to form a cohesive and uniformly readable contribution to the literature. The authors review areas of increasing interest to the AI community and to computer scientists, providing concise and up-to-the-minute information of increasing relevance.

Each chapter is written in a self-contained way, making the book suitable for the reader only interested in one or a few of the topics discussed, as well as the reader who requires a general overview of RMS at the present date.

Readership: Artificial intelligence, computer science, computer architecture, cognitive psychology, cognitive science, computer learning, knowledge engineering, expert systems.

REASON MAINTENANCE SYSTEMS AND THEIR APPLICATIONS

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 Limited, Edinburgh

Andrew, A.M.	**Continuous Heuristics***
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., Forsyth, R., Narayanan, A. & Teague, M.	**Dictionary of Artificial Intelligence***
Campbell, J. A. & Cox, P.	**Implementations of PROLOG, Vol. 2***
Carter, D.	**Interpreting Anaphors in Natural Language Texts**
Davies, R. (editor)	**Intelligent Information Systems: Progress and Prospects**
Evans, J.B.	**Structures of Discrete Event Simulation**
Farreny, H.	**Artificial Intelligence Exercise Programs***
Forsyth, R. & Rada, R.	**Machine Learning: Applications in Expert Systems and Information Retrieval**
Gabbay, D.M.	**Programming in Pure Logic***
Gergely, T. & Futó, I.	**Artificial Intelligence in Simulation***
Gottinger, H.W.	**Artificial Intelligence: The Commercialisation of Intelligent Systems***
Hawley, R. (editor)	**Artificial Intelligence Programming Environments***
Hayes, J.E. & Michie, D. (editors)	**Intelligent Systems: The Unprecedented Opportunity**
Lopez de Mantaras, R.	**Approximate Reasoning and Knowledge Acquisition***
Łukaszewicz, W.	**Nonmonotonic Reasoning***
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***
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**
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**
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**

* *In preparation*

LIST OF CONTRIBUTORS

Rob Bodington, Department of Computer Science, University of Reading, Whiteknights Park, Reading, RG6 2AX.

Michael Clarke, Department of Computer Science & Statistics, Queen Mary College, Mile End Road, London E1 4NS.

Nikos Drakos, School of Computer Studies, University of Leeds, Leeds LS2 9JT.

Peter Elleby, Department of Computer Science, University of Reading, Whiteknights Park, Reading, RG6 2AX.

Michael Hopkins, Department of Computer Science & Statistics, Queen Mary College, Mile End Road, London E1 4NS.

John Jones, Department of Computer Science, University of Hull, Hull HU6 7RX.

Gerald Kelleher, Computer Based Learning Unit, University of Leeds, Leeds LS2 9JT.

Mark Millington, Computer Science Subject Group, MAPS, University of Sussex, Brighton BN1 9QH.

Gregory Provan, Department of Computer Science, University of British Columbia, 6356 Agricultural Rd., Vancouver BC, Canada V6T 1W5.

Han Reichgelt, AI Group, Department of Psychology, University of Nottingham, University Park, Nottingham, NG7 2RD.

Table of Contents

LIST OF CONTRIBUTORS

Rob Bodington, Department of Computer Science, University of Reading, Whiteknights Park, Reading, RG6 2AX.

Michael Clarke, Department of Computer Science & Statistics, Queen Mary College, Mile End Road, London E1 4NS.

Nikos Drakos, School of Computer Studies, University of Leeds, Leeds LS2 9JT.

Peter Elleby, Department of Computer Science, University of Reading, Whiteknights Park, Reading, RG6 2AX.

Michael Hopkins, Department of Computer Science & Statistics, Queen Mary College, Mile End Road, London E1 4NS.

John Jones, Department of Computer Science, University of Hull, Hull HU6 7RX.

Gerald Kelleher, Computer Based Learning Unit, University of Leeds, Leeds LS2 9JT.

Mark Millington, Computer Science Subject Group, MAPS, University of Sussex, Brighton BN1 9QH.

Gregory Provan, Department of Computer Science, University of British Columbia, 6356 Agricultural Rd., Vancouver BC, Canada V6T 1W5.

Han Reichgelt, AI Group, Department of Psychology, University of Nottingham, University Park, Nottingham, NG7 2RD.

Murray Shanahan, Department of Computing, Imperial College of Science and Technology, 180 Queen's Gate, London SW7 2BZ.

Barbara Smith, School of Computer Studies, University of Leeds, Leeds LS2 9JT.

Preface

This book is based on the papers presented at the 'Workshop on Reason Maintenance Systems and Their Applications' held at the University of Leeds, England, on the 14th and 15th April 1988. The Workshop was sponsored by the Society for the Study of Artificial Intelligence and the Simulation of Behaviour and by the Computer Based Learning Unit of the University of Leeds.

The main theme of the book is the theory and application of Reason Maintenance Systems, although several papers reflect the overlap of this field with the more general problem of non-monotonic reasoning.

We should like to take this opportunity to thank all the people who contributed, in one way or another, to the smooth running of the Workshop and the production of this book. In particular, we should like to thank all the authors of papers; Tony Cohn and Masoud Yazdani of the SSAISB; Richard Thomas, who provided the initial impetus for the Workshop; and Roger Hartley, Director of the CBL Unit, for their support and help. The administrative assistance of Pat Greenwood, Barbara Lewis, Nikos Drakos, Jackie Bailey and Farath Arshad made a difficult task a great deal easier, as did the patience of the members of the CBL Unit on the days they were invaded by the Workshop hordes.

Gerry Kelleher and Barbara Smith

University of Leeds,
September 1988.

1

A BRIEF INTRODUCTION TO REASON MAINTENANCE SYSTEMS

Gerald Kelleher
Computer Based Learning Unit and School of Computer Studies,
University of Leeds.
Barbara M. Smith
School of Computer Studies, University of Leeds.

Interest in the use of Reason Maintenance Systems (RMSs) has been increasing since the pioneering work of Doyle in the mid seventies, and as the contents of this book demonstrate, the range of current applications of the various RMSs is very wide. This chapter will outline the basic concepts underlying a few of the RMSs available. It is by no means an exhaustive survey of the field but gives a flavour of the major trends and ideas within the rapidly expanding field of RMS application. The bibliography at the end of the chapter should provide a useful entry point into the literature on RMSs.

1.1 BACKGROUND

A Reason Maintenance System (RMS) acts a house-keeping subsystem of an overall reasoning system. The problem solver passes to the RMS the inferences it makes, and the RMS uses the structure of these inferences to organise the beliefs of the problem solver. The problem solver receives information back about what it believes by interrogating the RMS. The form of the interaction between the two sub-systems varies, but typically the RMS would provide information about the beliefs on which inconsistencies rest and sets of mutually consistent beliefs with which to work. The relationship is shown schematically in Figure 1 (adapted from de Kleer 1986a).

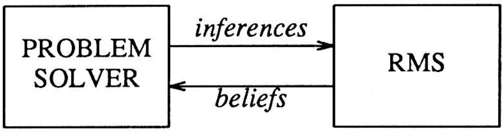

Fig. 1 A Reasoning System.

An important characteristic, shared by all RMSs, is that the RMS has no access to the semantics of the problem solver behaviour. The RMS treats the expressions passed to it by the problem solver purely syntactically. As a consequence it is the responsibility of the problem solver to ensure the correctness of the information passed to the RMS.

The basic motivation for the introduction of RMSs has been the need to perform belief revision, in the widest sense. Many problems involve the need to make choices about how to proceed with less than perfect information. Because of this some choices may be wrong and alternatives need to be considered. Many difficulties arise in such situations, ranging from the need to determine the 'knock on' effects of revising choices, to deciding which choices need remaking and the avoidance of repeating work.

In a typical classically based reasoning system the discovery that the premises on which an argument is based are inconsistent leads to the wholesale rejection of the premises. In many situations, such as those often encountered in AI, the premises from which an argument is constructed are not inviolable. A common example is the use of plausible assumptions when reasoning about an incompletely specified problem. This may lead to difficulties when it is discovered that some initially plausible choice is, in fact, inconsistent with what we later find out. Reason Maintenance Systems are intended to address this problem by providing machinery that allows the consequences of assumptions to be determined and the set of assumptions revised, if needed.

Until the introduction of RMSs much of this work, such as the cacheing of expensive inferences, was implemented anew for each problem encountered. Clearly this is wasteful, and, because these support mechanisms were not always separated from the problem solving, it could lead to unnecessary confusion in the system design.

Hence, the advantages of using an RMS may be summarised as increased design clarity, improved search efficiency, decreased redundancy of computation and access to the consequences of choices. These advantages are most clearly seen through an example and will be enlarged upon in the context of a particular RMS, de Kleer's assumption-based truth maintenance system (ATMS).

Historically there has been some confusion over the name that should be given to Reason Maintenance Systems. The initiator of the field, Doyle, called his system a Truth Maintenance System (TMS). Doyle himself, however, acknowledged that this was a misnomer (Doyle 1979). Consistency or belief maintenance have also been used in describing systems of this sort, but throughout this paper we will use the term reason maintenance, as this now seems generally accepted.

1.2 HISTORICAL OVERVIEW

It is difficult to say exactly where the fundamental notions of reason maintenance originally arose. Doyle (1978, 1979) is normally credited with the founding of the field, and he certainly provided the first comprehensive and implemented version of an RMS. However, many of the ideas had previously been used, in a somewhat *ad hoc* way, in different searching and reasoning programs. Hayes (1975) seems to be the earliest reference to what might be regarded as an RMS. His program, for the construction of robot plans, implemented a control structure that utilised the dependency relations of the choices in the plan creation process to minimise the replanning necessary on failure in either the generation or execution of the plan. This control apparatus was not generalised outside the domain of the particular program.

Rescher (1964, 1974) provides a perspective on some of the issues addressed by RMSs from the point of view of a philosophical logician. Many of the ideas presented in these works prefigure later work in RMSs. The notion of a maximal mutually consistent hypothesis set is, for example, very similar to de Kleer's idea of an interpretation. Anderson and Belnap (1975) provide a framework for logical reasoning which, in its use of logical dependency, involves many of the insights needed for the development of an RMS. Anderson and Belnap's system FR, was in fact the starting point for the development of the logic which underpins Martins and Shapiro's belief revision proposals, embodied in their Multiple Belief Reasoner (MBR) system (1986, 1987).

If Doyle's system is taken as the beginning of the work on RMSs, later systems can be seen to have developed in two related, but separate, directions. The direct line from Doyle is the work on what are known as *justification* based RMSs, the best known of which are the systems of McAllester (1980), Goodwin (1982, 1984), Thompson (1979) and McDermott (1983). The inclusion of McDermott's system in this list is somewhat arbitrary as his work can be seen as overlapping the boundary between justification based systems and the *assumption* based RMSs of de Kleer (1984, 1986a,b,c) and Martins and Shapiro (1983, 1984, 1986, 1988). This second strand of work, that based on assumptions, appears to have started with the work of Martins and Shapiro (see their 1988 paper, pp. 25 & 27), but the most widely used example of the assumption based approach to reason maintenance is probably de Kleer's.

1.3 A SIMPLE TAXONOMY OF RMSs

Current RMSs are normally distinguished by the ways in which they maintain dependency relations and the scope of the information to which they allow access. Systems are typically referred to as being justification based or assumption based (the type of dependency storage) and single context or multiple context (the type of access to information).

1.3.1 Justification and Assumption Based Systems

The distinction between justification and assumption based RMSs is important and is repeatedly referred to in the papers presented in this volume and in the literature generally. Discussions of the differences may be found in Martins and Shapiro (1988) and de Kleer (1984, 1986a).

The basic difference between an assumption based and a justification based system is the way in which the dependencies between the data are recorded. In a justification based system only the immediate relationship between a datum and its supports are recorded, whereas in an assumption based system the set of hypotheses on which a datum ultimately depends are identified and recorded with each datum.

In a justification based system the fact that two (or more) elements of a database storing inferences are dependent on one another can only be determined by examining the justification relationships, and tracing through those relationships to connect the elements in question.

For example, if the RMS is told that A $\rightarrow$ B and B $\rightarrow$ C, the fact that C depends on A is determined by following the justifications. The relationship is established via the link through B. Typically, of course, the problem of establishing such a dependency will involve a more tortuous route than the one involved in this trivial example.

In an assumption based system the hypotheses on which an inference depends are stored with the derived information. The supports for the data are incrementally updated as each new inference is passed to the RMS. If A is a hypothesis and the RMS is given the inference A $\rightarrow$ B the system associates with B its dependence on A (in ATMS terms, it *labels* the derived datum). When the new inference B $\rightarrow$ C is passed to the system the fact that C ultimately depends on A is also recorded. A natural consequence of this is that establishing a relationship is runtime efficient.

A common use of the term 'justification based system' is as a synonym for a Doyle or McAllester type of TMS. This is a more specific usage of the term and implies not only a particular type of representation but also that particular mechanisms, such as automatic consistency maintenance, are present.

1.3.2 Single context and multiple context systems

A context, in the sense of single or multiple context systems, is a set of data that is made available to the problem solver by the RMS. In a single context system only one set of data at a time is made available, in a multiple context system several sets of data may be available simultaneously. A context will be expected to be consistent and is normally determined by some set of hypotheses or assumptions. The context is the set of data that follow from the set of assumptions: the assumptions *characterise* the context. (Note that this is a very general view of the term 'context', its precise definition varying from system to system).

The differences between single and multiple context RMSs are bound up with the distinction between justification and assumption based systems. The essential idea is that in a single context system the RMS insists on maintaining for the problem solver one consistent subset (a context) of the data that has been passed to the RMS, whereas the multiple context systems provide a facility for determining contexts dynamically, without enforcing the usage of any particular one.

Doyle's system, for example, is a single context system as it maintains for the user one consistent subset of the data known to the RMS. De Kleer's ATMS is a multiple context system.

Presenting the issue of justification or assumption based RMS and single or multiple context systems separately is slightly controversial. Typically, justification based systems operate with a single context and assumption based systems operate with multiple contexts. There is no intrinsic reason why this should be the case, since the dependencies stored in a justification based system are capable of determining consistent contexts without insisting on any particular one, and on the other hand the assumption based systems could insist on a particular context.

It is true, however, that finding such consistent contexts or reasoning about them would be a computationally expensive and conceptually difficult task for a justification based system. The reason for this is that each datum is only implicitly associated with any context in which it may be derived (de Kleer, 1986a, p. 141). In practice this means that the single context systems have been justification based and the multiple context systems assumption based.

1.3.3 Justification v. Assumption Based systems

Which of these systems should be used in practice depends to a great extent on the nature of the support required from the RMS.

The problem solver may require the ability to examine several (perhaps mutually contradictory) possibilities at a time, for example if it is carrying out a breadth first search. In this situation, a multiple context system is probably to be preferred because switching contexts is an expensive and hard to control process in existing single context systems.

Single context RMSs are probably the better choice when the problem solver is operating in a search space which has many solutions and where only a few are required. This is because the justification based systems can make some claim to ensuring the work they do is relevant. Multiple context systems, because they are assumption based, may do work which is of no use to the problem solver. Potentially the assumption based systems store information about dependency relationships that are not important to the problem in hand. This difficulty is analogous to the problems of some algorithms using constraints which may spend time restricting search in areas of the search