Artificial Intelligence

in

Chemical Engineering

Thomas E. Quantrille & Y. A. Liu

Artificial Intelligence in Chemical Engineering

Thomas E. Quantrille

and

Y. A. Liu

Department of Chemical Engineering
Virginia Polytechnic Institute and State University
Blacksburg, VA

ACADEMIC PRESS
Harcourt Brace Jovanovich, Publishers
San Diego New York Boston London Sydney Tokyo Toronto

Academic Press Rapid Manuscript Reproduction

This book is printed on acid-free paper. ∞

Academic Press, Inc.
San Diego, California 92101

United Kingdom Edition published by
Academic Press Limited
24–28 Oval Road, London NW1 7DX

Library of Congress Cataloging-in-Publication Data

Quantrille, Thomas E.
Artificial intelligence in chemical engineering / by Thomas E. Quantrille and Y. A. Liu.
p. cm.
Includes bibliographical references and index.
ISBN 0-12-569550-0
1. Chemical engineering--Data processing. 2. Artificial intelligence--Industrial applications. I. Liu, Y. A. (Yih-An) II. Title.
TP184.Q36 1991
660'.0285'63--dc20 91-34293
CIP

PRINTED IN THE UNITED STATES OF AMERICA
92 93 94 95 9 8 7 6 5 4 3 2 1

PREFACE

This book is intended as an introduction to artificial intelligence (AI) for chemical engineers. In the classroom, the book can serve as the textbook for a one-semester course in artificial intelligence in chemical engineering for seniors or graduate students. The book is also appropriate as a self-study guide for practicing engineers who wish to understand more about the field of AI in engineering. Finally, computer scientists interested in engineering applications of AI should find this book informative, and rich in references.

Computer applications in chemical engineering have increased dramatically over the last decade. AI techniques in particular have received much attention. This book is a response to that growth, and therefore, has two major purposes:

- To introduce chemical engineers to the principles and applications of AI.
- To introduce chemical engineers to computer programming techniques for AI.

Before we can accomplish these goals, however, we must first address a more fundamental question: *what is artificial intelligence*? There is no universally accepted definition of AI, so we cannot provide a single, clear-cut answer. But we can suggest several definitions relevant to this text. The goal of AI has always been to make computers "think," to solve problems requiring human intelligence. Based on this idea, Elaine Rich and Kevin Knight, in their book *Artificial Intelligence* (McGraw-Hill, 1991) define AI as:

The study of how to make computers do things which, at the moment, people do better.

Avron Barr and Edward A. Feigenbaum in *The Handbook of Artificial Intelligence*, Volume I (Addison-Wesley, 1981) propose the following definition:

Artificial Intelligence is the part of computer science concerned with designing intelligent computer systems, that is, systems that exhibit characteristics we associate with intelligence in human behavior.

We prefer this latter definition; the focus of this book is to introduce the computational means and methodologies to enable computers to perform intelligent engineering tasks.

Thus, one of the primary goals of this book is to move beyond the basic principles of AI solidly into its applications in chemical engineering. After reading this book, a chemical engineer will have a firm grounding in AI, know what chemical engineering applications of AI exist today, and understand the current challenges facing AI in engineering. Importantly, after reading this book, the chemical engineer will also be able to implement AI systems using the language Prolog.

To achieve our objectives, we have divided this book into four parts:

- Introduction to AI programming, including Prolog (chapters 1 to 8) and LISP (chapter 9),

and comparison of Prolog with LISP (chapter 9).
- Introduction to AI principles (chapters 10 and 11).
- Chemical engineering case study: the development of EXSEP, an EXpert System for SEParation Synthesis (chapters 12 to 15).
- Overview of knowledge-based systems and artificial neural networks in chemical engineering (chapters 16 and 17).

Prolog is one of the two principal languages of artificial intelligence, is used extensively in Europe and Japan, and with increasing frequency in the United States. Prolog originated in the early 1970's, where researchers attempted to use logic as a programming language. The name itself stands for PROgramming in LOGic. Robert Kowalski and Maarten van Emden established the theoretical foundations for Prolog at Edinburgh. Alain Colmerauer, at the University of Aix-Marseille in Marseille, France, completed the first implementation of Prolog. Prolog owes its current popularity to an efficient implementation credited to David Warren at Edinburgh in the mid-1970's.

Prolog is a young, "up-and-coming" language. Consequently, it has no "standard" syntax. For this book, we have chosen the most popular Prolog syntax in use today, known as the Edinburgh syntax (also called DEC-10 syntax). For the most part, we have used Arity/Prolog to develop the programs presented in this text. Arity/Prolog's syntax closely resembles Edinburgh syntax. However, we have tried to avoid using features unique to Arity/Prolog to keep the programs as broadly applicable as possible. We also provide some programs in Turbo Prolog (presently being sold under the name PDC Prolog). Turbo Prolog is less powerful than Arity/Prolog, but is also much less costly. As with Arity/Prolog, Turbo Prolog's syntax is closely related to Edinburgh syntax. Thus, applications described in this book can be applied to virtually any version of Prolog as long as differences in syntax are recognized.

Part One (chapters 1–9) introduces and discusses AI programming in detail. We discuss aspects of syntax, program control, built-in functions, and programming techniques of Prolog using chemical engineering examples. At the end of chapter 8, the reader will be able to write complex AI programs in Prolog.

Note, however, that this book is *not* strictly a Prolog book. In chapter 9, we introduce LISP (List Processing), the other major AI language, and contrast and compare Prolog and LISP. In the United States, LISP is actually used more widely than Prolog, and many chemical engineering applications of AI use LISP. LISP owes its origins to John McCarthy, who developed the language at MIT in 1959 and formally presented it in 1960. LISP is the second oldest computer language around—the oldest is FORTRAN. Because of its age, LISP has been modified and optimized more than Prolog. LISP systems generally have many more built-in functions than Prolog systems do, and in that sense, they are more powerful.

Why, then, do we emphasize Prolog? We feel that *economically*, Prolog has an advantage over LISP, when it comes to *learning and teaching AI*. We strongly feel that engineers should "get their feet wet" and understand AI *with as little time and initial financial investment as possible*. Once an engineer has a firm grasp of the field, then he or she can justify more expensive software and hardware systems if necessary. Prolog allows us to adopt this approach.

For teaching purposes:

- Prolog is simpler and easier to learn than LISP.
- Prolog provides for faster learning—an engineer can develop and implement prototype programs more quickly in Prolog than in LISP.
- Prolog programs can be developed and implemented on inexpensive personal computers. Most LISP development packages need specialized hardware, which raises the cost to both the developer and the user.
- Prolog is a relatively inexpensive investment—a user can purchase personal-computer-based Prolog software for as little as $100.

Consequently, for learning and teaching AI systems quickly and inexpensively, we feel that Prolog is superior to LISP.

Each language, then, has its own advantages *and* disadvantages. While Prolog is faster and easier to learn, LISP is more powerful. Before choosing which language to use, the wise developer will understand these advantages and disadvantages. Indeed, Patrick Winston, Director of the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology, writes in his Foreword to the Prolog text by Ivan Bratko (Addison-Wesley, 1990, p. vii):

> *In the Middle Ages, knowledge of Latin and Greek was essential for all scholars. The one-language scholar was necessarily a handicapped scholar who lacked the perception that comes from seeing the world from two points of view. Similarly, today's practitioner of Artificial Intelligence is handicapped unless thoroughly familiar with both LISP and Prolog, for knowledge of the two principal languages of Artificial Intelligence is essential for a broad point of view.*

Chapter 9 will help in this regard by introducing LISP and providing a comparison of both LISP and Prolog.

In Part Two (chapters 10 and 11), we discuss AI principles. Chapter 10 presents a broad overview, and discusses problem-solving techniques that have been used successfully in AI programs. We also include strategies of *knowledge representation*, i.e., systematic ways to represent knowledge (i.e., facts, rules and heuristics) such that an AI program can solve problems accurately and efficiently.

Chapter 11 is a more specific discussion of Prolog programming for AI. We introduce AI programming techniques in Prolog, including the *search*. In AI, the *search* is the methodology for *processing* the information in the program to draw conclusions and solve problems. We discuss various problem-solving approaches in AI, and discuss how to implement these approaches using Prolog. After the completion of chapter 11, the reader should have a firm grasp of Prolog and AI, and, more importantly, know how to implement Prolog programs for AI applications.

Part Three (chapters 12–15) is a chemical engineering case study. The goal of Part Three is to demonstrate, by example, how to develop, build, and implement an expert system. Chapter 12 introduces EXSEP (EXpert System for SEParation Synthesis), a process design tool for multicomponent separations in chemical engineering. EXSEP uses heuristics with shortcut feasibility analysis to develop technically feasible and economically attractive chemical process flowsheets.

Chapter 13 discusses EXSEP from the chemical engineering perspective. Through this chapter, we hope to describe how to develop a knowledge representation that opportunistically takes advantage of the chemical engineering problem characteristics. Expert-system success hinges on good knowledge representation.

Chapter 14 describes EXSEP from the user's perspective. We provide a copy of the EXSEP program listing in the Appendix, and a diskette for use on IBM PCs (or MS DOS compatible computers). We discuss EXSEP's user-interface, how to access the program, and how to interact with and use the program to develop chemical process flowsheets.

Part Three concludes with Chapter 15, where we discuss both the knowledge representation and the problem-solving strategy in EXSEP from the AI standpoint. AI researchers have developed various techniques and problem-solving strategies for expert systems. Certain techniques are more useful than others, depending on the problem. Knowing the available AI tools and techniques, as well as when to apply these tools, is critical to the success of an expert system. At the close of Part Three, the reader should know how to implement expert systems for problems of moderate to high complexity.

In Part Four (chapters 16 and 17), we "spread our wings" and take a look at recent trends in AI in science and engineering, and in chemical engineering in particular. Chapter 16 is a survey of knowledge-based applications in chemical engineering implemented in Prolog, LISP, and other AI programming languages. We discuss how to manage expert-system projects. We then move on to specific application areas, such as fault diagnosis, process control, process design, planning and operations, modeling and simulation, and product design, development, and selection.

Finally, chapter 17 introduces a more recent AI application in chemical engineering, *Artificial Neural Networks* (ANNs). The ANN is a numerical, empirical modeling tool that has received much attention recently. ANN's origins lie in the computer modeling of the human brain, with associated attempts to emulate neuron interaction. Chapter 17 describes the unique properties that separate ANNs from other empirical modeling techniques.

Importantly, chapter 17 assumes no prior knowledge of ANNs, and discusses their purpose and structure starting with the most basic principles. We then discuss current and potential ANN applications in chemical engineering. We close the chapter with a thorough summary of the advantages *and* limitations of ANNs. Again, the wise developer will always know the limitations of any technique being used.

This text contains numerous example problems to clarify the discussion. We have included a large number of simple exercises and more complex practice problems for use as classroom assignments and self-study materials. Solutions to most of the exercises and practice problems appear in the Appendix. In addition, we include a glossary, which explains the terminology used in AI applications in science and engineering.

After working through this book, a chemical engineer will have the breadth and depth to implement AI projects using Prolog. Depending on the problem, however, LISP may be the better language choice. LISP is a more powerful language, with more built-in procedures, and thus, may be more convenient. The reader may wish to refer to *LISP*, by Patrick H. Winston and Berthold K.P. Horn (Addison-Wesley, 1984), for a discussion of the language.

Overall, this text serves as an introduction to AI techniques in engineering (and chemical engineering, in particular). It includes both a ***broad coverage*** and an ***in-depth review*** of AI techniques. We hope that this book will benefit engineering students (seniors or graduate students), practicing chemical engineers, and computer scientists interested in applications of AI to engineering.

ACKNOWLEDGEMENTS

It is a pleasure to thank a number of very special persons who contributed to the preparation of this book.

First, we would like to thank the enduring patience of our wives, Sharon Quantrille and Hing-Har Liu. Their support through the laborious process of book preparation and editing was exemplary and most appreciated.

The idea for this book originated from the doctoral work of the junior author, Tom Quantrille. The junior author would like to thank the members of his advisory committee, in particular: Professor Y. A. Liu, who developed the original idea and the details of the book and was the major advisor; Professor Peter Rony, whose thorough review and comments proved invaluable; Professor John Roach, who introduced the subject of artificial intelligence to both authors; Professor William Conger, the Department Head who provided both technical and logistic support; and finally, Professor Henry McGee, who served on the committee even while on leave at the National Science Foundation as Director of the Division of Chemical and Thermal Systems.

In addition, we wish to express our sincere gratitude to Professor Michael Mavrovouniotis of the University of Maryland and to Professor Lyle Ungar of the University of Pennsylvania for their prompt and thorough reviews of the manuscript for the whole book. Professor Mavrovouniotis was particularly helpful to our revision of chapter 17.

To Ms. Marie Paretti, we owe a deep gratitude for her tireless and skillful copy-editing of the entire manuscript for this book. We would also like to thank Ms. Lydia Morgan, editorial assistant at Academic Press, for her help in making the final pages and artwork come together and Ms. Lisa Herider of Office Arts for her electronic production of final camera ready pages.

We would also like to express our appreciation to the Dow Chemical Foundation, whose financial support made this work possible.

We wish to thank Mr. Jean-Christophe N. Brunet for his help with many aspects of this book project.

The senior author should like to thank Professor Gregory Botsaris of Tufts University for his continued support and encouragement.

Lastly and most importantly, we wish to thank Mrs. Diane Cannaday, who patiently and graciously helped all aspects of the preparation of this writing project.

RECOMMENDATIONS

This section gives some recommendations for student learning, course topics and software selection.

Student Learning and Course Topics

For a first reading through this text or for a one-semester course, we recommend that the following portions be emphasized.

- Chapters 1 to 5. These chapters give the foundation for Prolog programming for AI applications.
- In Chapters 6 to 9, focus on sections 6.1 to 6.3, 7.3, 7.5, 8.1, 8.3, 9.1 and 9.2. These sections give an understanding of some essential built-in Prolog tools, explain techniques for improving programming efficiency, and contrast Prolog with LISP. We skip over some advanced and less-used tools and techniques.
- Chapter 10. This key chapter gives the foundations of AI and expert systems.
- In Chapter 11, focus on sections 11.1 and 11.2. These sections represent the "meat" of Prolog programming for AI applications.
- In Chapters 12 to 15, focus on sections 12.2, 13.1 to 13.3, 13.6, 14.1 to 14.3, 15.1 and 15.2. These sections are the key to understanding the case study, EXSEP (EXpert system for SEParation synthesis). In addition, we recommend using the diskette for EXSEP included inside the backcover of this book to develop separation process flowsheets for examples and practice problems in section 14.4C before progressing to Chapter 15.
- Chapter 16 is a stand-alone chapter that focuses on the knowledge-based applications in chemical engineering and can be used to enhance course effectiveness if time permits. The topics discussed in sections 16.3 to 16.8 can be assigned to students to write term papers or develop course projects. In addition, students may be able to update the literature on the knowledge-based applications to specific areas by reviewing the latest issues of journals cited in these sections.
- Chapter 17 is also a stand-alone chapter. Sections 17.1 and 17.2 cover the fundamentals of artificial neural networks (ANNs), an increasingly important area of AI applications to chemical engineering. Topics discussed in section 17.3 can be assigned to students for term papers or course projects on applications of ANNs to chemical engineering.

Software Selection

We strongly recommend that the reader practice Prolog programming. There are two Prolog packages that we recommend for IBM personal computers or compatibles under MS DOS.

1. Turbo Prolog (presently being sold under the name PDC Prolog). Prolog Development Corporation, 568 14th Street, N.W., Atlanta, GA 30318, phone (404) 873-1366. PDC Prolog is identical to Turbo Prolog, previously sold by the Borland International, Inc., Scotts Valley, CA. This Prolog software is the least expensive and the reader will need to use it to develop and compile Prolog codes, such as exercises in Part One.
2. Arity/Prolog. Arity Corporation, 29 Domino Drive, Concord, MA 01742, phone (508) 371-1243. Arity/Prolog is a powerful, well-organized, PC-based Prolog.

Both companies offer student prices and group discounts. For a class of ten or more students, Arity/Prolog is an excellent choice.

ABOUT THE AUTHORS

Thomas E. Quantrille is a Ph.D. student in the Department of Chemical Engineering, Virginia Polytechnic Institute and State University ("Virginia Tech"). He is completing his doctoral research on artificial intelligence in chemical engineering design.

Mr. Quantrille received his B.S. degree with honors from Virginia Tech in 1984. He worked for the Procter and Gamble Company as a research engineer from 1984 to 1986, and then joined James River Corporation as a technical research leader. He was granted a number of U.S. and European patents during his industrial employment. He returned to Virginia Tech in 1988 for graduate studies. For his outstanding academic progress at Virginia Tech, he has received the Pratt Presidential Award and the Dow Chemical Ph.D. Fellowship. He expects to receive his Ph.D. degree in 1991.

Mr. Quantrille's professional interests include artificial intelligence, process synthesis, and polymer science and technology. He is a member of AIChE, AAAI, and TAPPI.

Y. A. Liu is the Frank C. Vilbrandt Professor of Chemical Engineering at Virginia Tech. He received his B.S. (1967), M.S. (1970), and Ph.D. (1974) degrees from National Taiwan University, Tufts University and Princeton University, respectively.

Professor Liu taught at Auburn University from 1974 to 1981, where his last position was Alumni Associate Professor endowed by the Auburn Alumni Association. He joined Virginia Tech as a Professor of Chemical Engineering in 1982. In 1983, he was appointed the Vilbrandt Professor.

He has two patents and over 100 publications in the fields of process design, magnetochemical engineering, coal desulfurization and fluidization processes. He has also edited four books which have become standard references in these fields.

Professor Liu's contributions to chemical engineering teaching and research have been recognized by university and national awards. In 1984, he received the Western Electric Award from the American Society of Engineering Education (ASEE). The award recognizes his excellence as an engineering teacher and his contributions to the engineering profession. In 1986, Professor Liu received the National Catalyst Award from the Chemical Manufacturers Association for outstanding teaching and contributions to chemical education. In 1990, he received the George Westinghouse Award, ASEE's highest honor for an engineering educator under age 45. This award recognizes the outstanding early achievements of a young engineering educator in both teaching and scholarship. In 1990, Professor Liu also received the Distinguished Alumni Award from Tufts University.

CONTENTS