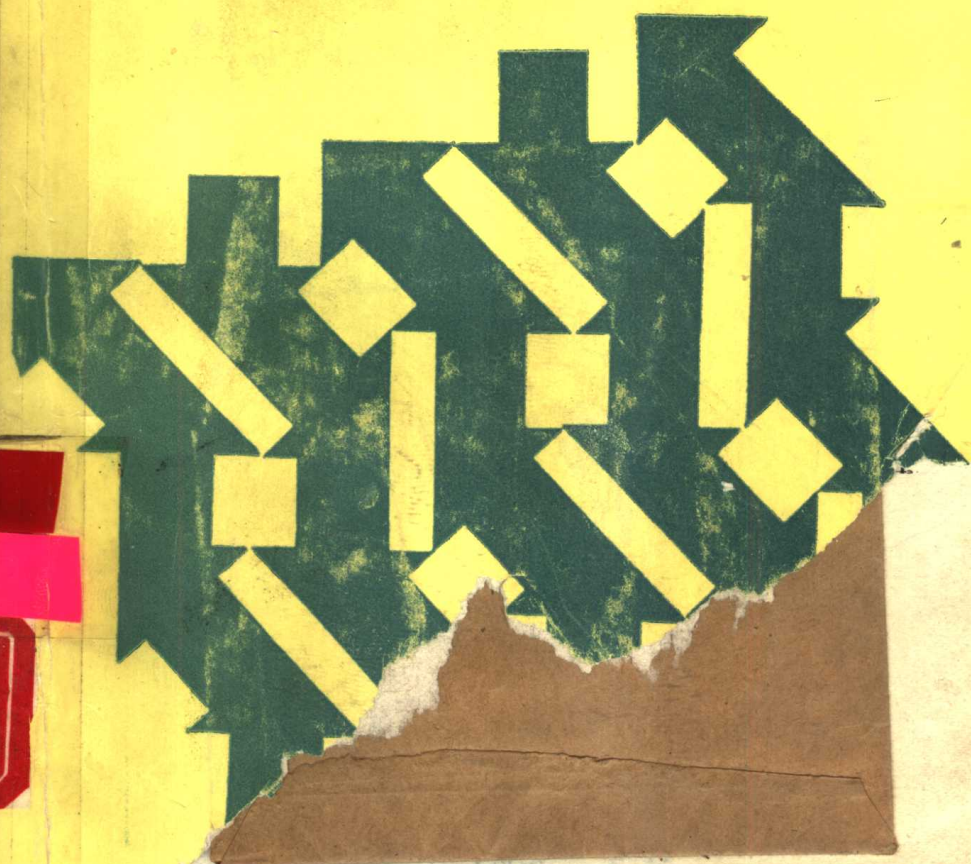


逻辑程序设计 原理和方法

刘凤歧 陈耀新 编著



○ 刘凤歧 陈耀新 编著

逻辑程序设计 原理和方法

LUOJI CHENGXU SHEJI YUANLI HE FANGFA

○ 国防科技大学出版社

内 容 简 介

逻辑程序是由人工智能的研究发展起来的适于人工智能应用的新型程序设计语言。它的诞生和发展反映了人工智能技术已取得的成就，且对计算机科学和技术会产生深刻影响。本书阐述了人工智能的基本概念，逻辑程序设计的原理，并且给出了一种具体的逻辑程序设计语言——Micro-PROLOG。本书还介绍了专家系统，专家系统开发工具及其使用 Micro-PROLOG 的实现技术。最后介绍了并行逻辑程序设计语言PARLOG，综述了逻辑程序设计对计算机科学的贡献。

本书适用于计算机专业和自动化专业的本科生和研究生以及从事人工智能，专家系统和新一代计算机研究的科技工作者。

逻辑程序设计原理和方法

刘凤岐 陈跃新 编著

责任编辑 张俊科

封面设计 侯 云

国防科技大学出版社 出版

湖南省新华书店发行

国防科技大学印刷厂 印装

开本 787×109 21/32 印张 12 字数 278千字

1987年10月第1版 1987年10月第1次印刷 印数 5 000册

统一书号 15415·028

ISBN 7-81024-012-9

TP·4 定价：2.00元

前 言

我国在四化建设的进程中，对人工智能这一学科的研究，特别是对专家系统的开发日益急迫。许多有关科技工作者渴望有我国自己编写的较系统地介绍人工智能及其实现和应用技术并能反映当前发展情况的书，本书试图为这些读者服务。本书有以下特点：

一、以逻辑程序设计为主题，具体地阐述了人工智能的基础知识和基本原理，详细介绍了PROLOG语言及其理论基础，并且论述了按逻辑程序设计风格开发和构造专家系统及专家系统建造工具的方法和途径。对于目前国内外正在潜心研究的并行逻辑程序设计语言的机制和实现以及这些研究在第五代计算机研究中的重要地位也作了介绍。读者能通过阅读本书获得人工智能原理、技术、应用及实现的较全面和系统的知识，可望在掌握本书内容的基础上可开始从事人工智能、专家系统、逻辑程序设计以及第五代计算机等几个层次的研究工作。

二、本书是作者从事本科生及研究生教学多年，对教材做了多次修改、充实、更新的基础上写成的。内容结构和阐述层次吸收了多年教学经验，所以本书也是一本适合大学教学的教科书或参考书。为使不熟悉离散数学的读者能直接阅读本书，书中设置了预先知识部分。

全书共有十二章。第一章概述了人工智能，专家系统，逻辑程序设计和第五代计算机的基本思想和概念。第二章简明介绍了符号逻辑，熟悉符号逻辑的读者可将本章略去。第三章介

绍了机械定理证明的基础——Herbrand定理，并且给出了直接应用 Herbrand 定理实现机械定理证明的方法。第四章介绍 J. A. Robinson 在 1965 年发现的“归结原理”，并且说明了“归结原理”是一种强有力的机械推理规则。第五章给出了“语义归结”，对归结原理的实现效率做了改进。本章还介绍了适于计算机实现的几种机械定理证明方法。第六章引入了 Horn 短句概念，按“Horn 短句归结定理证明器”描述了逻辑程序的语法和执行机制。第七章介绍逻辑程序设计语言 Micro-PROLOG，细致的讨论了用 PROLOG 语言进行程序设计的主要特点和技巧。第八章介绍基于规则的专家系统，以及用 Micro-PROLOG 实现专家系统时的一些技巧问题。本章还给出了一个专家系统开发工具 APES 以及用 APES 实现的一个专家系统。第九章讨论逻辑程序设计中的不确定性，并行性，元级语言和目标级语言，可证明性关系等问题。第十章介绍逻辑程序的过程语义，模型论语义，和不动点语义，以及它们之间的关系。第十一章介绍并行逻辑程序设计语言 PARLOG，引入了“急性计算”和“惰性计算”等概念。最后一章综述逻辑程序设计的更广泛内容及其对计算机科学的贡献。

俞咸宜 1987.5

目 录

第一章 导 言	1
第二章 符号逻辑导引	
2.1 命题逻辑	12
2.2 一阶逻辑	16
2.3 逻辑推论	25
2.4 符号逻辑中的标准型	27
2.5 一阶逻辑的应用	33
第三章 Herbrand定理	
3.1 Skolem 标准型	36
3.2 Herbrand解释	43
3.3 语义树	49
3.4 Herbrand 定理	52
3.5 Herbrand 定理的实现	55
第四章 归结原理	
4.1 命题逻辑的归结原理	61
4.2 取代和一致化	64
4.3 一致化算法	67
4.4 一阶逻辑的归结原理	71
4.5 归结原理的完全性	76
4.6 归结原理应用示例	80
4.7 删除策略	84
第五章 语义归结	
5.1 语义归结导引	91
5.2 语义归结的定义	94
5.3 语义归结的完全性	97
5.4 超归结和支撑集合策略——语义归结的特殊情况	99

第六章 Horn短句与逻辑程序

6.1 Horn 短句的归结原理	104
6.2 输入反驳	110
6.3 单位反驳	113
6.4 逻辑程序	116

第七章 Micro-PROLOG

7.1 基本语法	134
7.2 扩充 Horn 短句	141
7.3 表格处理与语法分析	154
7.4 某些语用考虑	165
7.5 元逻辑程序设计	174
7.6 Micro-PROLOG 的标准语法	184

第八章 专家系统的实现

8.1 专家系统导引	196
8.2 MYCIN 的不精确推理	204
8.3 知识表示和人-机交互	217
8.4 实现解释功能	225
8.5 专家系统外壳 APES	235
8.6 在法律中的应用	254

第九章 逻辑程序的若干问题

9.1 不确定性	265
9.2 “当”型定义	272
9.3 可证明性关系	282
9.4 信息吸收	292

第十章 逻辑程序的语义

10.1 过程语义	299
10.2 模型论语义	301
10.3 不动点语义	304
10.4 模型论语义和不动点语义	306
10.5 过程语义和不动点语义	307

第十一章 PARLOG: 并行逻辑程序设计

11.1 与-并行——单解关系	312
11.2 惰性计算和受限缓冲器进程	328
11.3 或-并行——全解关系	337
11.4 元级程序设计	344
11.5 结语	346
第十二章 综述	
12.1 计算理论	347
12.2 计算实践	353
12.3 计算技术	385
参考文献	

第一章 导 言

本书的主题是使用符号逻辑做为程序设计语言。在谈到逻辑程序设计时不能不涉及人工智能，专家系统（或含义更广的智能知识系统），以及日本人首先提出的第五代计算机概念。逻辑程序设计和专家系统是人工智能发展中几乎同时诞生的产物，而日本的第五代计算机计划将逻辑程序设计做为核心体系使逻辑程序设计赢得了国内外计算界的广泛兴趣和重视。

人工智能是计算机科学的一个分支，它研究的是表现出人类智力行为的计算机系统的设计问题。人工智能的研究工作在电子数字计算机诞生不久就开始了，迄今已有30余年历史，在此期间它并没有很快得到象试管婴儿，人类登月，集成度达上百万件的超大规模集成电路那样的令人兴奋的成就。究其根本，人类虽然能做出惊天动地的创造性工作，而且心理学家和哲学家们已经做了数百年工作，但“身在庐山不知庐山真面貌”，我们还不知道“智能”的真实含义和范围。此外，机器是否能具有人类那样的学习，推理，求解问题，理解语言等智力行为是长期以来争论不休的问题。按传统观念，智力是人类独有的，后来发现某些动物也具有类似人类的某些智力行为，但人是高级动物所以对此人们还能忍受，至于纯属无机物的机器能具有人类的智力行为往往令人费解。按传统观念看问题的人往往认为他们知道什么是智能，什么是创造性，实际上他们

知道的只是观念，对智力行为没有精确的了解，如果真的有精确了解的话对人工智能的争论也就不存在了。争论还会继续下去，但对人工智能的发展并不无裨益。从计算机科学角度出发我们宁愿取实用主义态度，不去过多追求智能的定义以及是否机器可以具有智力行为。这里应该提到的是逻辑学家 Turing, 1950年在他的论文“计算机与智能”中提出的测试机器是否有智能的方法：由某人提出问题分别由人和计算机来回答，如果提问者不能区分回答者是人还是计算机则认为该计算机有智力行为。要强调的是，Turing并没有要求计算机与人在理解问题和思考答案时一定采用完全相同的方法，关键是答案。当然，如果对大量问题机器都能得到与人相近似的答案，那么分析和归纳机器的工作过程会得到人脑产生智力行为的某些机理。机器毕竟是机械的，有条不紊的，细心的进行工作，如果过分追求让它按人脑的工作方式进行工作既不现实更会束缚我们的手脚。但是，可以象控制论的发展由神经系统的研究获得营养一样，当前可以将人脑看成一个问题求解系统，探讨它的过程模型推进人工智能的发展。人工智能研究的目标是使计算机更加有益于人类，更多的代替人的脑力劳动，这与机械，电气和自动控制设备代替人的体力劳动是相似的。如果把机械、电气和自动控制设备视为人类体力的延伸，那么计算机将成为人类脑力的延伸。现有的计算机已经为认识论的研究提供了试验场所，不管是否能得出对智力的良好理解，人工智能的发展总会产生一些智能技术，它会对我们的社会产生深刻的影响，并且推进计算机的发展。

人工智能的早期工作集中在用数字计算机下棋，做符号积分演算，以及证明数学定理等问题上，人们设计了许多算法和新型语言去完成这些工作。当然开始时计算机做这类工作是愚

笨的，蹩脚的，迟缓的。但经过十几年的不懈努力计算机已能下一手好棋，已达到国际象棋大师的水平，我们当中的绝大部分人都会败在它的名下，其中甚至包括写下棋程序的人。符号积分程序也达到了极高的水平，并且已为各国科学家和工程师所使用，人们不必在纸上做复杂的积分演算了。机械定理证明工作的成果更多，早在1963年人们就在计算机上证明了B. Russell等人所著“Principia Mathematica”一书第一章的全部定理。这些工作的成功使人工智能的研究发生了根本转变，人们不再去追求思考问题的普遍规律，而转向研究以知识为中心的系统。这种转变不是在理论指引下完成的，而是大量事实说明必须进行转变。产生智力行为的基础是知识，只有拥有丰富知识的人才能表现出他们的才华。当然知识包括一般性知识和某领域的专门知识。这里又遇到了一个什么是“知识”的问题。象“智能”一样，我们仍然采用实用主义态度，不去追究“知识”的严格定义，把精力集中在表现智力行为上。有人把人工智能归为实验科学，这是有道理的，因为设计的任何人工智能系统都必须经过试验看它是否表现出预期的智力行为，没有经过实验的系统是不可信的。人工智能系统转向以知识为中心便诞生了专家系统（或称为知识系统）。专家系统是个高性能的计算机程序，它拥有某专业领域（可能很狭小）的高质量知识以及使用这些知识求解问题的方法（可能通用一些）。它的工作方式从外观来看很类似人类专家。当一般医生使用医疗专家系统为患者看病时，专家系统要询问患者的病痛情况，如果是细菌感染病患者，那么专家系统还要询问对患者所做培养的情况，推断感染患者的可能病菌，判断抵抗这些病菌的可能药物，还可以询问进一步的情况，最后选出适于患者情况的药物。专家系统要根据患者的具体信息使用自己拥有的适于患者

情况的知识做出决断。构成专家系统的基本成分是知识库和推理机制。知识库中包含专业领域中的知识，它们可以是专家在学校里学到的，也可以是在工作实践中积累的，当然还可以是从文献中查阅的。知识库中采用适当的符号数据结构表达专业领域中的知识。在符号积分专家系统的知识库中要包含下述知识：

$$\begin{aligned}
 \int cf(x)dx &= c \int f(x)dx \\
 \int (f(x) + g(x))dx &= \int f(x)dx + \int g(x)dx \\
 \int u dv &= uv - \int v du \\
 &\vdots \\
 \int c dx &= cx \\
 \int x dx &= x^2/2 \\
 \int \sin x dx &= -\cos x \\
 \int a^x dx &= a^x \log_e a \\
 &\vdots
 \end{aligned}$$

当然不能直接将上述公式纳入知识库。我们用 $\text{int}(\text{exp1}, x, \text{exp2})$ 表示代数式 exp1 关于变量 x 的不定积分为 exp2 。尚需规定 exp1 和 exp2 的语法结构，这里不赘述。上述公式可表示为

$$\begin{aligned}
 \text{int}(c * f(x), x, c * z) &\leftarrow \text{int}(f(x), x, z) \\
 \text{int}(f(x) + g(x), x, z1 + z2) &\leftarrow \text{int}(f(x), x, z1), \\
 &\quad \text{int}(g(x), x, z2) \\
 \text{int}(u, v, u * v - z) &\leftarrow \text{int}(v, u, z) \\
 &\vdots
 \end{aligned}$$

```

int(c, x, c * x)
int(x, x, x * x/2)
int(sinx, x, -cosx)

```

⋮

上述后面的各公式都称为事实，如 x 对变量 x 的不定积分是 $x * x/2$ ，这是公认的。前面的各公式都称为规则，第二个规则可读为：若 $f(x)$ 对 x 的不定积分是 $z1$ ，并且 $g(x)$ 对 x 的不定积分为 $z2$ ，则 $f(x) + g(x)$ 对 x 的不定积分为 $z1 + z2$ 。规则的形式是“如果…，则…”。早期的专家系统都是按事实和规则表示知识的。知识表示是人工智能的重要研究领域之一，事实和规则只是表示知识的一种方式。使用知识库中的知识计算不定积分

$$\int (ax + b \sin x) dx$$

时采用的方法是通常的“套公式”法。问题可表示为：

```

←int(a * x + b * sinx, x, y)

```

这里 y 是需求取的变量。“←”表示提问，与规则中的“←”含义不同。在求 y 时只考虑“←”右部的

```

int(a * x + b * sinx, x, y)

```

我们在知识库中进行检索，找出“←”左端与它相匹配的规则或与它相匹配的事实。人很快就能找出第二个规则满足所述条件，对应关系为

```

f(x) = a * x   g(x) = b * sinx   y = z1 + z2

```

按此规则可将原提问转化为

```

←int(a * x, x, z1), int(b * sinx, x, z2)

```

该问题表示需要分别求取 $a * x$ 和 $b * \sin x$ 关于 x 的积分 $z1$ 和 $z2$ 。那么可以先求第一个积分。同样在知识库中进行检索，找出“←”左端与

$$\text{int}(a * x, x, z1)$$

相匹配的规则。显然第一个规则就满足这种条件，有

$$c = a \quad f(x) = x \quad z1 = a * x$$

由第一个规则，求 $z1$ 的问题化为

$$\leftarrow \text{int}(x, x, z)$$

知识库中的事实 $\text{int}(x, x, x * x/2)$ 与 $\text{int}(x, x, z)$ 是匹配的，于是得出

$$z = x * x/2 \quad \text{并且} \quad z1 = a * z = a * x * x/2$$

对 $\text{int}(b * \sin x, x, z2)$ 做同样处理可以得出

$$z2 = b * z = b * (-\cos x)$$

最后得出所需的积分结果：

$$y = z1 + z2 = a * x * x/2 + b * (-\cos x)$$

以上我们看到了如何使用事实和规则表达专业领域中的知识并且由它们构成知识库。使用知识库中的知识求解专业领域的问题时涉及的操作是匹配和检索（顺序搜索）。可以把规则解释为求解问题的方法，如

$$\begin{aligned} \text{int}(f(x) + g(x), x, z1 + z2) &\leftarrow \text{int}(f(x), x, z1), \\ &\quad \text{int}(g(x), x, z2) \end{aligned}$$

将求

$$\int (f(x) + g(x)) dx$$

的问题转化为分别求 $\int f(x) dx$ 和 $\int g(x) dx$ 的问题，即将复杂问题分解为若干简单问题。而事实，如

$$\text{int}(x, x, x * x/2)$$

是直接解决了问题的，不再进行问题转化。这里要强调的是，对一个给定的问题在庞大的知识库中可能有若干条规则的左端

与它存在匹配，由其中每条规则转化出的问题仍然会出现这种情况，即，对一个给定的问题可能存在若干条求解它的途径，有的途径最后能得到解，有的不能得到解（上述例子是理想情况）。这正象我们做积分演算时一样，在求积过程的任一步上都可能有一些公式可用，但按某个公式做下去到某一步上可能发现没有公式可“套用”了，不能得出结果，此时再返回去使用其它公式继续求积一直到得出结果为止。所以有时将规则称为试探式（或启发式）规则。有丰富求积经验的人在求积过程的任一步上能很快找出中肯的求积公式，用很少的求积步骤得出答案。而专家系统通常只是机械的穷举出求积的所有可能途径（一般是顺序的）得出积分结果，也可能由于知识库的知识不完善得不出结果。如何把人求积的丰富经验更多的纳入知识库是专家系统研究的重点问题之一。但是，像求积专家一样，对复杂的被积函数求积时总会难免走些弯路做一些“无用功”。

上面已经看到了如何使用知识库中的知识求解问题。求积过程的每一步都套用一个公式，对问题做一次转化，这恰好与做一步逻辑推断相对应。在形为

$$A \leftarrow B, C$$

的规则中， A, B, C 代表三个命题，规则可读为：若 B 和 C 都成立则 A 成立。提问“ $\leftarrow A$ ”中的“ $\leftarrow$ ”理解为“否定”，它可读为： A 不成立。如果 B 和 C 都成立则 A 成立，现在 A 不成立显然有 B 不成立或 C 不成立，这可表示为

$$\leftarrow B, C$$

其中“ $\leftarrow$ ”仍理解为“否定”。“ $\leftarrow B, C$ ”表示命题“ B 与 C ”不成立。所以由“ $\leftarrow A$ ”和“ $A \leftarrow B, C$ ”可推断出“ $\leftarrow B, C$ ”。当然专家系统的推理机制还可采用其它的推断方法，不论采用什么方法都要力求简单使用户易于理解。

象下棋，符号积分，数学定理证明这样的领域，知识形式化（公式化或符号化）的程度是极高的，显然这是理想情况。医学等其它领域远非如此，恐怕很难有象积分表那样的工具可用。此外，一个医生可能很难说出对患者所做的决断究竟使用了哪些细微知识，采用了什么规则，如何去推断的。经验知识更难以把握。七十年代中期E. A. Feigenbaum提出了“知识工程”概念，他认为专家系统知识库的建立需要领域中的专家与“知识工程师”密切合作。知识工程师是专家知识的发掘者，他研究人工智能懂得如何在计算机上表达知识和使用知识求解问题。知识工程师必须与专家一起冥思苦想，逐步诱导出专家求解问题时使用的知识。这种“知识获取”问题在专家系统的研制过程中是最耗时费力的。七十年代以来已经诞生了一批专家系统，其中许多都达到或超过了专家的水平，如：

- DENDRAL系统，Stanford大学研制。该系统拥有丰富的质量化学知识，能根据质谱仪数据帮助化学家推断分子结构和原子成分。

- PROSPECTOR，Stanford国际研究所（SRI）研制。该系统具有20多名优秀经济地质专家的知识，能根据地质数据推断矿藏的可能位置且估价开采价值。它曾发现华盛顿州的一个钼矿，据说开采价值达一亿美元。

- MYCIN系统，Stanford大学研制。是诊断和治疗细菌感染和脑膜炎的专家系统。该系统对专家系统的发展有较大影响。

- 数学专家系统 MACSYMA，麻省理工学院研制。该系统能求解600多种数学问题，包括微积分，解方程式和方程组，泰劳级数展开，矩阵和向量运算等。

- XCON 计算机配制专家系统，卡内基—梅隆大学和DEC

公司研制。该系统每年为DEC公司节省1 500万美元。

专家系统和知识工程的出现使人工智能的研究达到了应用阶段。显然专家系统表现是人类高层次的智力行为——专家的智力行为。另外要强调的是，用户请教专家系统时不一定能象符号积分那样对问题能说得很清楚，这就要求专家系统通过人一机接口诱导用户给出问题的更多信息。

人类的智力行为以知识为中心的思想不只产生了专家系统而且使机器翻译，自然语言理解，问题求解，机器人学和机器学习的研究发生了根本转变。的确，人的所有智力行为如理解，求解问题，乃至学习能力都是以知识为基础的。人必须先有知识才能去理解，才能知道更多的东西。

支持人工智能发展的两大支柱是数理逻辑和计算。早在计算机出现之前数理逻辑就建立了逻辑推理的数学形式化描述，使人们看到了智力行为与计算的关系和发展前景。逻辑推理应该转化为符号处理，不是数值运算。描述数字计算机内部运行状态的是数字，它是一种特殊的符号。逻辑推理是任何人工智能系统都必须具备的功能，可以将它定义为计算。人工智能中对机械定理证明的研究将逻辑与计算概念结合在一起，几乎与专家系统同时产生了逻辑程序设计体系。逻辑程序设计语言PROLOG(PROgramming in LOGic) 于1972年诞生于法国马赛大学，后来在英国得到了完善和发展。真正懂得它的原理的人都认为这是计算机科学的重大发展。学习用PROLOG语言写程序时会感到象第一次学写程序那样兴奋。逻辑程序设计显示了一种新的思维方式和一种描述问题的新方法，这种方法极其有效和优雅。逻辑程序——更具体的讲，是PROLOG程序——与传统程序差别甚大，它由事实和规则构成，执行逻辑程序的是“定理证明器”。显然，这与专家系统是极其相似的；知识库