**7.1基于知识的智能体**

**7.3逻辑**

知识库：由语句组成

语句：根据表示语言的语法（syntax）表达

语义：语句的含义，语义定义每条语句在每个可能世界中的**真值**

模型：形式上结构化的世界

对于每个相关的语句，每个模型都有固定的真值（真或假）

如果语句α在模型m中为真，我们说m满足α，有时也可以说m是的一个模型。我们使用记号M(α)来代表 的所有模型的集合

蕴含（entailment）：语句α蕴含语句β，则是α比β更强的断言

α|=β 当且仅当 M(α)包含于M(β)

推导：KB |—i a ，i为一个推断算法

推断算法的

可靠性/保真性：每当KB |—i a时，KB|=a也为真

完备性：每当KB|=a时，必有KB |—i a

**如果KB在真实世界中为真，则用可靠的推断过程从KB中推出的所有语句 在真实世界中也为真。**

如何知道KB在真实世界中为真？

知识库真值表枚举检验

**7.4命题逻辑：一种非常简单的逻辑**

**7.4.1语法**

原子语句（atomic sentence）：由单个命题符号（proposition symbol）构成，每个这样的符号代表一个为真或假的命题。

逻辑联结词（logical connective）：与、或、非、蕴含、当且仅当

复合语句：原子语句+逻辑联结词

正文字：原子语句 负文字：原子语句的否定

**7.4.2语义**

语义定义了用于判定特定模型中**语句真值的规则。**

命题逻辑中，**模型**就是**对每个命题符号设定真值**，即真（true）或假（false）。

命题逻辑的**语义**必须**指定**在给定模型下**如何计算任一语句的真值**。（计算规则）

这些规则（**语义**）也可以用**真值表**表示

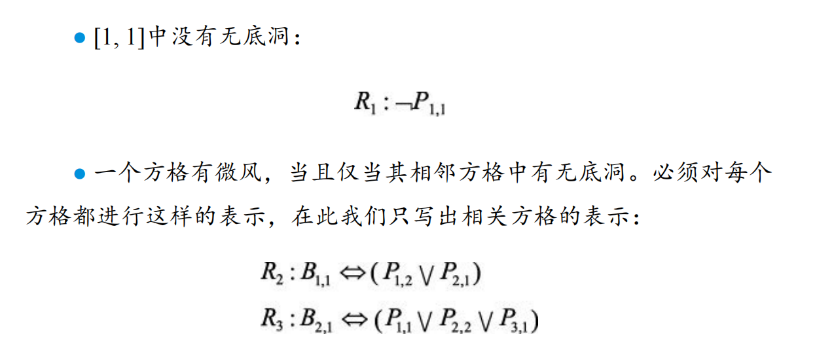
关于蕴含：P蕴涵Q

命题逻辑并不要求P和Q之间有任何因果关系或相关性

前件为假的所有蕴涵式都为真

唯一为假的情形是当P为真而Q为假时

**7.4.3一个简单的知识库**

KB中：用Ri代表每个语句

**7.4.4一个简单的推断过程**

目标：确定对于某些语句α，KB|=alpha是否成立

命题蕴含是余NP完全的（即很可能不比NP完全简单）

**推理算法1：模型检验方法**

枚举所有模型（深度优先），检验在**KB为真的每个模型**中是否为真。

（模型是对每个命题符号进行真或假的赋值 ）

**该方法是可靠的、完备的**

时间复杂度：O(2n)

**7.5命题定理证明**

**逻辑等价**α≡β：如果两个语句α和β在相同的模型集合中都为真，则这两个语句逻辑等价

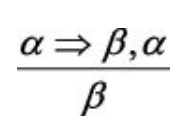
**有效性/重言式/永真式**：如果一条语句在所有模型中都为真，则这条语句是有效的

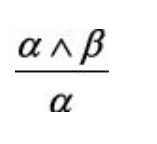
**可满足性**：如果一条语句在某些模型中为真或能够被满足，则这条语句是可满足的。

在命题逻辑中**确定语句的可满足性**的问题——**SAT问题**——是第一个被证明为**NP完全**的问题。

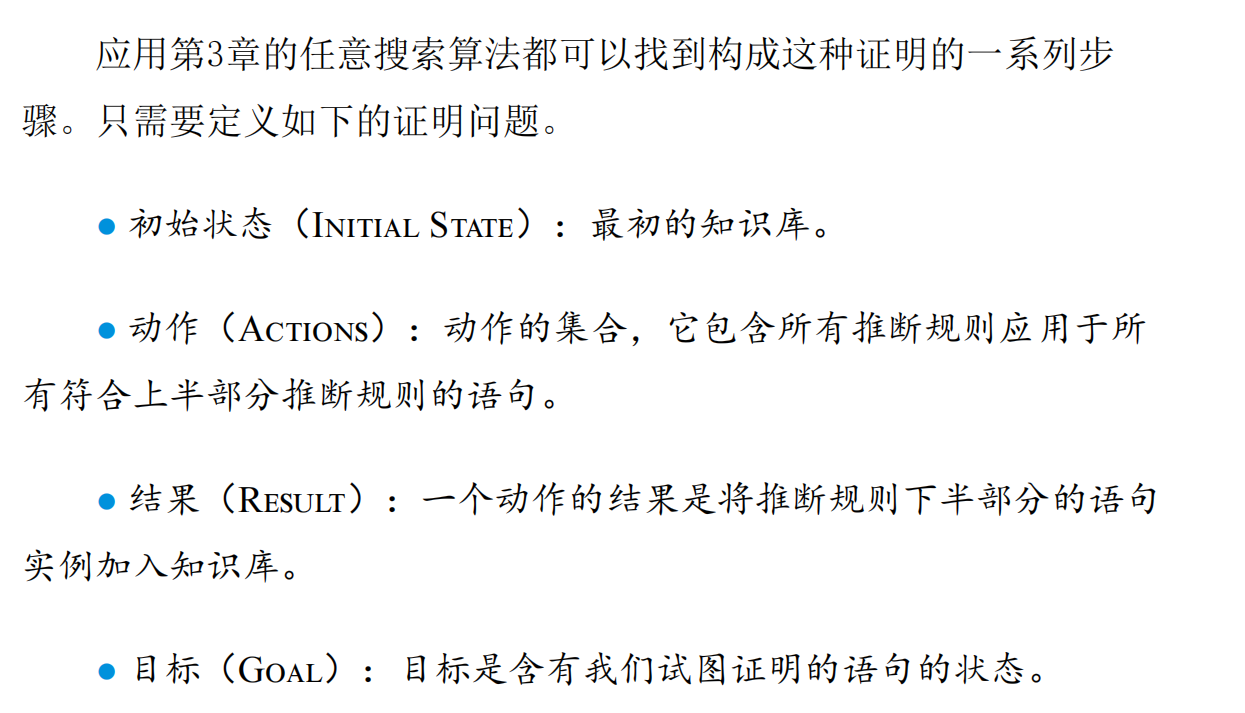
**7.5.1推断与证明**

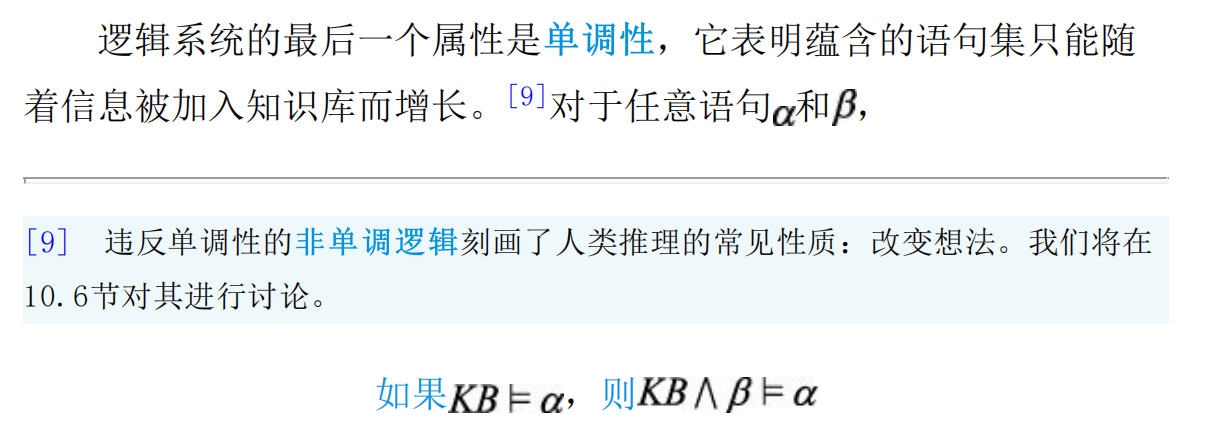
推导证明的推断规则

肯定前件:

合取消去：

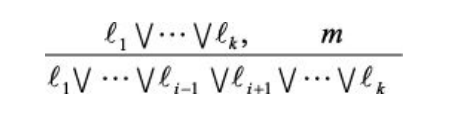
等……





**7.5.2通过归结证明**

归结（resolution）

AvB，~A 其中，m与*li*互补

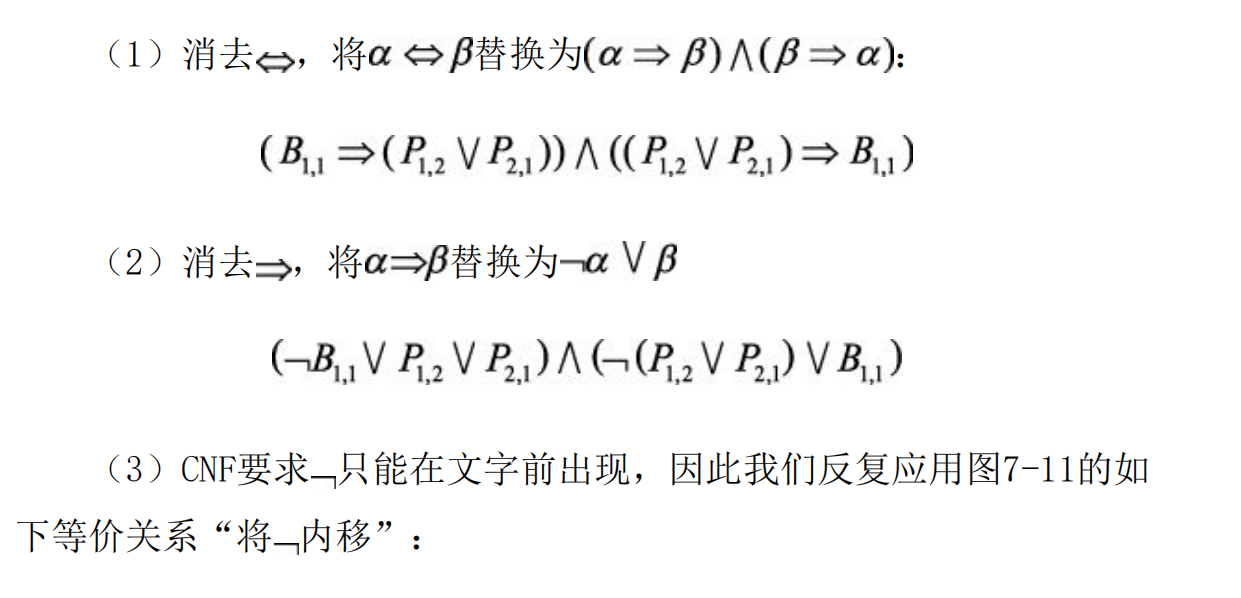
B

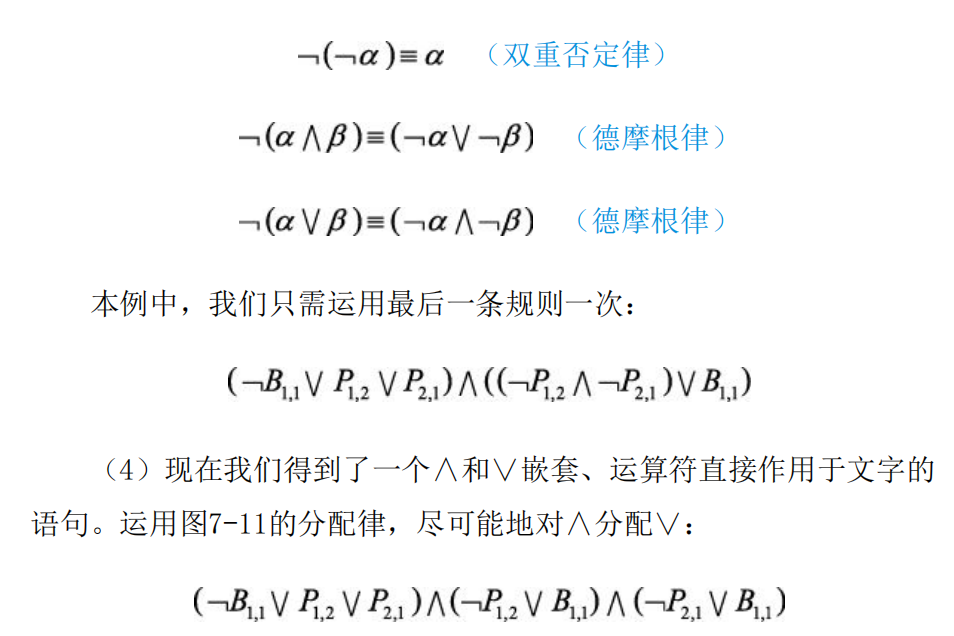
当它与任意完备的搜索算法结合后，可以产生一个**完备的**推断算法

基于归结的定理证明器可以对命题逻辑中的任意语句α和β确定α|=β是否成立

1.合取范式CNF

转换步骤：





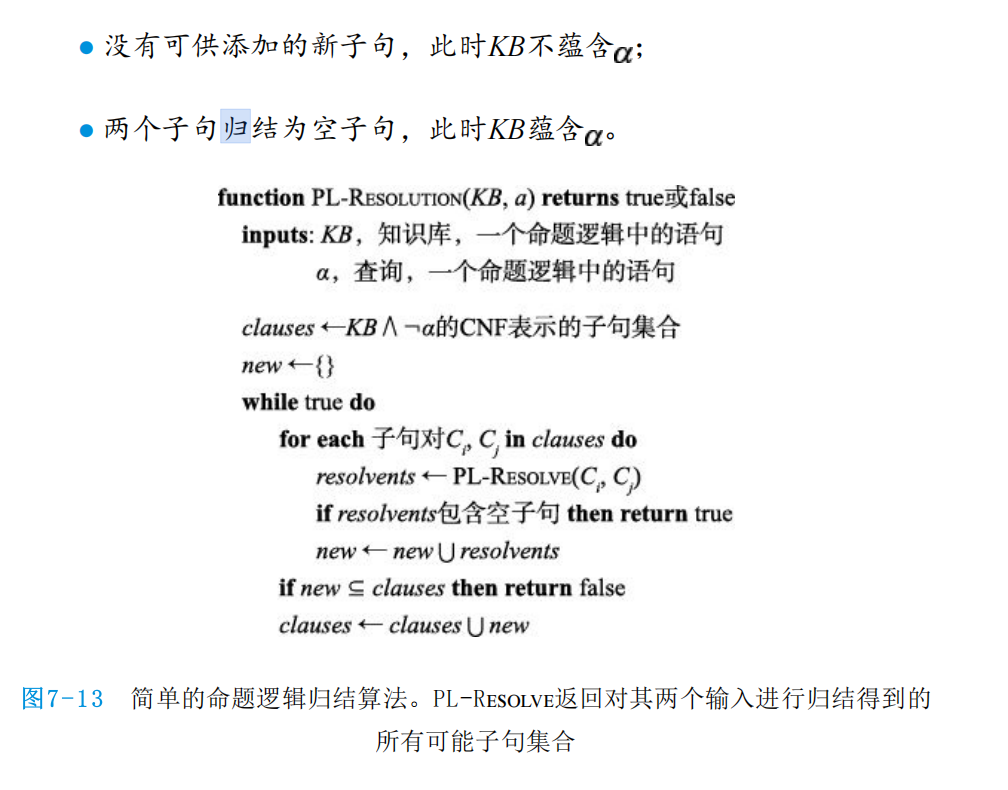
2.归结算法

证明KB |= α，需证明：KB & ~α不可满足

对CNF的子句不断进行两两归结（只配对能归结的子句，配对过的子句从clauses中删除）

若出现空句(=False)合取后整体为False，说明KB|=α成立

若归结操作不能再进行下去，说明整体表达式并非恒错，说明KB|=α不成立



3.归结的完备性

子句集合S的归结闭包（resolution closure）：RC(S)

即对S中子句及其生成子句反复使用归结规则可推得的所有子句的集合。

易知**RC(S)必然是有限的**：得益于因子提取，由S中出现的符号P1,…, Pk得出的子句数量是有限的。因此，PL-Resolution总是能够终止

**归结基本定理：**

**如果一个子句集是不可满足的，则这些子句的归结闭包含有空子句**。

**7.5.3霍恩子句与确定子句**

确定子句：文字的析取式，且文字中**只有一个**是正文字

霍恩子句：文字的析取式，且文字中**最多只有一个**是正文字

目标子句：**没有正文字**的霍恩子句

**霍恩子句关于归结封闭：**如果归结两个霍恩子句，仍然会得到霍恩子句。

仅含有确定子句的知识库：

（1）每个确定子句都可以写成一个蕴涵式：（~Av~BvC）可写为(A&B)=>C

在霍恩形式中，前提被称为体（body）而结论被称为头（head）

由单个正文字构成的语句，例如A，被称为事实（fact）

（2）用霍恩子句进行推断可以通过前向链接（forward-chaining）算法和反向链接（backward-chaining）算法完成

（3）用霍恩子句确定蕴含关系**所需的时间与知识库大小呈线性关系**

**7.5.4前向链接与反向链接**

**前向链接算法（一种数据驱动推理）**

**PL-FC-ENTAILS?(KB,q)** ：

确定单个命题符号q（即查询）是否被确定子句的知识库所蕴含。它从知识库中的已知事实（正文字）开始。如果一个蕴涵式的所有前提都已知，则将其结论添加到已知事实的集合中。这一过程持续进行，**直到查询q被添加**，或**直到无法进一步进行推断**。运行时间是线性的

与或图：用**曲线连接**的多个**边**表示一个合取式，每个边都要证明；

而没有曲线连接的多个边表示一个析取式，证明任一边即可。

前向链接算法是**可靠的、完备的**

**反向链接算法（一种目标导向推理）**

从查询开始反向运作。

如果查询q已知为真，则不需要做任何操作。否则，算法将在知识库中找寻**结论为q**的**蕴涵式**。

如果这些蕴涵式的**所有前提**都可以（用反向链接）证明为真，则q为真。

时间复杂度是**线性的**

通常，**反向链接的代价远小于知识库规模的线性变化**，因为这个过程仅涉及相关的事实

**7.6高效命题模型检验**