命题逻辑（二）

命题的语义

- 语法：符号表达式的形式结构

- 语义：符号和符号表达式的涵义（给符号以某种解释）

命题的语义

- 什么是命题逻辑的语义？

- 对于任意的**赋值** $v: PS \to \{T, F\}$，定义一个**解释**

$$\hat{v}: PROP \to \{T, F\}$$

联结词定义的布尔函数

定义1.21. 令真值集$\boldsymbol{B}=\{T,F\}$,

- 联结词 ¬ 被解释为一元函数 $H_{\neg}: \boldsymbol{B} \rightarrow \boldsymbol{B}$;
- 联结词 * 被解释为二元函数 $H_{*}: \boldsymbol{B}^2 \rightarrow \boldsymbol{B}$,

  其中$* \in \{\wedge, \vee, \rightarrow\}$;
- $H_{\neg}, H_{\wedge}, H_{\vee}, H_{\rightarrow}$定义如下:

$\boldsymbol{p}$	$\boldsymbol{q}$	$H_{\neg}(\boldsymbol{p})$	$H_{\wedge}(\boldsymbol{p},\boldsymbol{q})$	$H_{\vee}(\boldsymbol{p},\boldsymbol{q})$	$H_{\rightarrow}(\boldsymbol{p},\boldsymbol{q})$
T	T	F	T	T	T
T	F	F	F	T	F
F	T	T	F	T	T
F	F	T	F	F	T

命题的语义

定义1.22（命题的语义）.

- v 为一个**赋值**指它是函数 $v\colon PS \to \boldsymbol{B}$，

  从而对任何命题符P_i，$v(P_i)$为T或F；

命题的语义

定义1.22（命题的语义）.

- v 为一个**赋值**指它是函数 $v: PS \to \boldsymbol{B}$，

  从而对任何命题符P_i，$v(P_i)$为T或F；

- 对于任何赋值 v，定义 $\hat{v}: PROP \to \boldsymbol{B}$ 如下：

  $\hat{v}(P_n) = v(P_n), n \in N$；

  $\hat{v}(\neg A) = H_{\neg}(\hat{v}(A))$；

  $\hat{v}(A * B) = H_{*}(\hat{v}(A), \hat{v}(B))$，其中$* \in \{\wedge, \vee, \to\}$。

对于命题A，它在赋值v下的**解释** $\hat{v}(A)$ 为T或F。

命题的语义

例，$A=(p\wedge q)\rightarrow(\neg q\vee r)$，设 v 是一个赋值，使得

$$v(p)=v(q)=v(r)=1.$$

命题的语义

例，$A = (p \wedge q) \rightarrow (\neg q \vee r)$，设 v 是一个赋值，使得

$$v(p) = v(q) = v(r) = 1.$$

那么，我们有

$$\hat{v}(p \wedge q) = H_{\wedge}(p, q) = 1,$$

$$\hat{v}(\neg q \vee r) = H_{\vee}(H_{\neg}(q), r) = 1,$$

$$\hat{v}(A) = H_{\rightarrow}(H_{\wedge}(p, r), H_{\vee}(H_{\neg}(q), r)) = 1.$$

命题的语义

例，$A=(p\wedge q)\rightarrow(\neg q\vee r)$，设 v 是一个赋值，使得

$$v(p)=v(q)=v(r)=0.$$

命题的语义

例，$A = (p \wedge q) \to (\neg q \vee r)$，设 v 是一个赋值，使得

$$v(p) = v(q) = v(r) = 0.$$

我们有

$$\hat{v}(p \wedge q) = H_{\wedge}(p, q) = 0,$$

$$\hat{v}(\neg q \vee r) = H_{\vee}(H_{\neg}(q), r) = 1,$$

$$\hat{v}(A) = H_{\to}(H_{\wedge}(p, q), H_{\vee}(H_{\neg}(q), r)) = 1.$$

解释与赋值的关系

引理1.23. 设A为命题，令$FV(A) = \{P \in PS | P出现在A中\}$，设$v_1$和$v_2$为赋值。若$v_1|FV(A) = v_2|FV(A)$，则$\hat{v}_1(A) = \hat{v}_2(A)$。

$v_1: PS \to \boldsymbol{B}$，

$v_1|FV(A): FV(A) \to \boldsymbol{B}$，

即，对于$p \in FV(A)$，则$v_1(p) = v_2(p)$。

解释与赋值的关系

例，$A=(p\wedge q)\to(\neg q\vee r)$，设 v_1和 v_2 是两个赋值，使得

$$v_1(p)=v_1(q)=v_1(r)=v_1(s)=1,$$

$$v_2(p)=v_2(q)=v_2(r)=1, v_2(s)=0.$$

解释与赋值的关系

例，$A = (p \wedge q) \rightarrow (\neg q \vee r)$，设 v_1和 v_2 是两个赋值，使得

$$v_1(p) = v_1(q) = v_1(r) = v_1(s) = 1,$$

$$v_2(p) = v_2(q) = v_2(r) = 1, v_2(s) = 0.$$

令

$$FV(A) = \{p, q, r\},$$

解释与赋值的关系

例，$A = (p \wedge q) \rightarrow (\neg q \vee r)$，设 v_1和 v_2 是两个赋值，使得

$$v_1(p) = v_1(q) = v_1(r) = v_1(s) = 1,$$
$$v_2(p) = v_2(q) = v_2(r) = 1, v_2(s) = 0.$$

令

$$FV(A) = \{p, q, r\},$$

那么

$$v_1|FV(A) = v_2|FV(A),$$
$$\hat{v}_1(A) = \hat{v}_2(A).$$

解释与赋值的关系

引理1.23. 设A为命题，令$FV(A) = \{P \in PS | P出现在A中\}$，设$v_1$和$v_2$为赋值。若$v_1|FV(A) = v_2|FV(A)$，则$\hat{v}_1(A) = \hat{v}_2(A)$。

证明：对A的结构作归纳。

A为五种形式之一：原子公式，$\neg B$，$B \wedge C$，$B \vee C$，$B \to C$。

归纳基础：当$A \in PS$时，显然有$\hat{v}_1(A) = \hat{v}_2(A)$。

归纳假设：对于B和C，有 $\hat{v}_1(B) = \hat{v}_2(B)$和 $\hat{v}_1(C) = \hat{v}_2(C)$。

归纳步骤：

情况¬：$A = \neg B$，

$$\hat{v}_1(A) = \hat{v}_1(\neg B) = H_\neg(\hat{v}_1(B))$$

$$= H_\neg(\hat{v}_2(B)) = \hat{v}_2(\neg B) = \hat{v}_2(A)。$$

归纳步骤：

情况¬：$A = \neg B$，

$$\hat{v}_1(A) = \hat{v}_1(\neg B) = H_{\neg}(\hat{v}_1(B))$$

$$= H_{\neg}(\hat{v}_2(B)) = \hat{v}_2(\neg B) = \hat{v}_2(A)。$$

情况*：$A = (B * C)$，

$$\hat{v}_1(A) = \hat{v}_1(B * C) = H_*(\hat{v}_1(B), \hat{v}_1(C))$$

$$= H_*(\hat{v}_2(B), \hat{v}_2(C)) = \hat{v}_2(B * C) = \hat{v}_2(A)。$$

□

可满足性

定义1.24. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

1. v **满足** A，记为 $v \vDash A$，指 $\hat{v}(A) = T$；

   A 是**可满足的**，指 $\exists v$ 使得 $v \vDash A$；

2. v **满足** Γ，记为 $v \vDash \Gamma$，指对于 $\forall B \in \Gamma$，$v \vDash B$；

   Γ 是**可满足的**，指 $\exists v$ 使得 $v \vDash \Gamma$。

注：若$v \nvDash A$，则$v \vDash \neg A$。

Γ的可满足性蕴含Γ中所有公式的可满足性。

但反之不一定成立。

元语言

- 注意$\vDash$不是命题语言中的符号，而是元语言（也称上层语言）中的符号。

- 除此之外，在元语言中我们也需要使用一些联结词。
  - 如，iff（当且仅当）、not（非）、 and（与）、or（或）、imply（蕴含）等；
  - $v \vDash \neg A$ iff not $v \vDash A$；
  - $v \vDash (A \wedge B)$ iff $v \vDash A$ and $v \vDash B$；
  - $v \vDash (A \vee B)$ iff $v \vDash A$ or $v \vDash B$；
  - $v \vDash (A \to B)$ iff $v \vDash A$ implies $v \vDash B$。

另一种等价的语义定义

给定一个模型（赋值）$v : PS \to \mathbf{B}$，对于任意 $\varphi \in PROP$

$v \vDash \varphi$ 定义如下：

- $v \vDash P$ iff $v(P) = T$
- $v \vDash \neg\varphi$ iff $v \nvDash \varphi$
- $v \vDash \varphi_1 \wedge \varphi_2$ iff $v \vDash \varphi_1$ and $v \vDash \varphi_2$
- $v \vDash \varphi_1 \vee \varphi_2$ iff $v \vDash \varphi_1$ or $v \vDash \varphi_2$
- $v \vDash \varphi_1 \to \varphi_2$ iff not （$v \vDash \varphi_1$ and $v \nvDash \varphi_2$)

  iff $v \nvDash \varphi_1$ or $v \vDash \varphi_2$

都满足

满足一个

此外，还可以定义：

$\vDash \varphi$ iff $\forall v : v \vDash \varphi$

可满足性问题

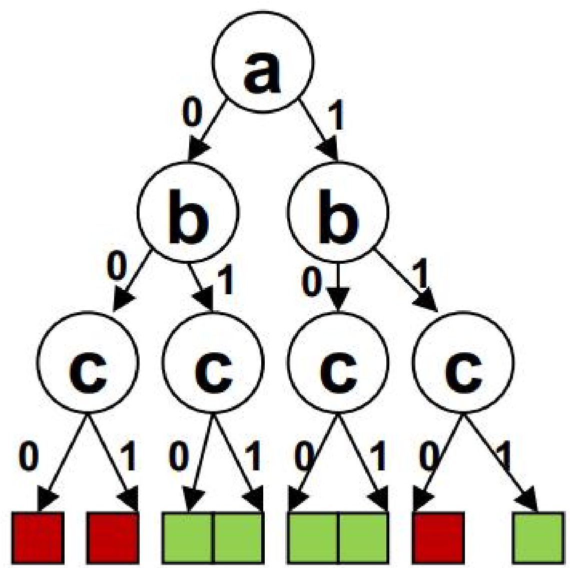
- 给定一个命题公式A，问是否存在一个赋值v，使得$v \vDash A$?
  - 此赋值v也被称为问题的一个解。

可满足性问题

- 给定一个命题公式A，问是否存在一个赋值v，使得$v \vDash A$?
  - 此赋值v也被称为问题的一个解。

$$F = (a \vee b) \wedge (\neg a \vee \neg b \vee c)$$

对n个变量的问题，一共有2^n组可能的赋值。

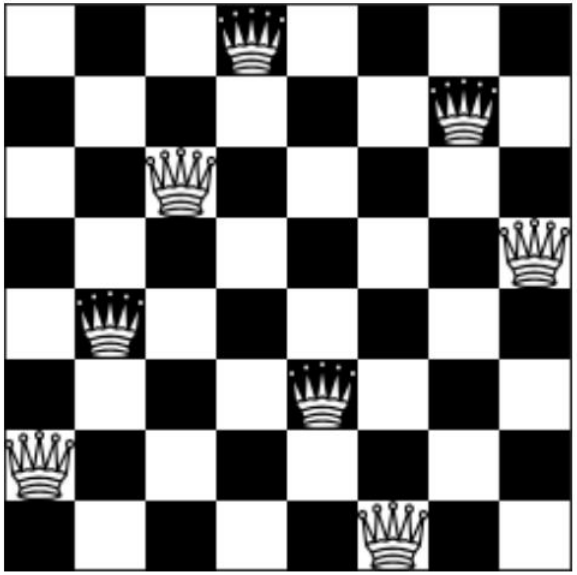


可满足性问题

- 给定一个命题公式A，问是否存在一个赋值v，使得$v \vDash A$?
  - 此赋值v也被称为问题的一个解。

- 命题逻辑公式的可满足性问题（也称**布尔可满足性问题，或SAT问题**）
  - 第一个被证明的**NP完全问题**（**NP-Complete, NPC**）
    （它是NP问题且所有NP问题可以多项式时间归约到它）；
  - 非确定性算法：将问题分解为**猜测**和**验证**两个部分；
  - 验证一个赋值是公式的一个解很容易（多项式时间，即NP）；
  - 找到一个解很困难；
  - P ⊆ NP✓ P=NP？（七个千禧年难题）

n-皇后问题



n-皇后问题

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

不同行:

第 i 行只有一个皇后

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

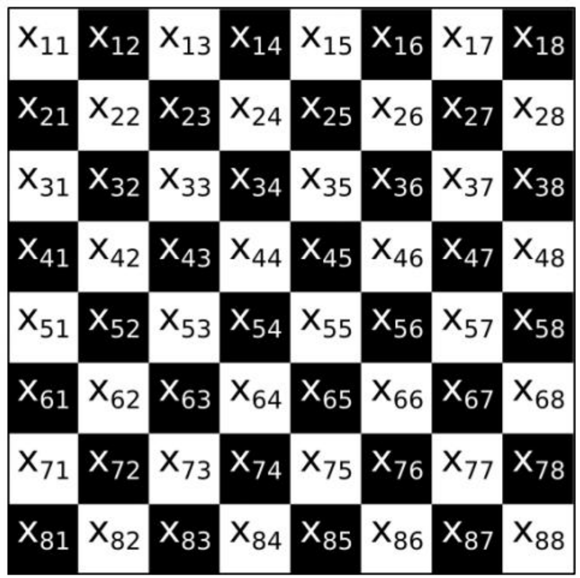
n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

不同行:

第 i 行只有一个皇后

$\Rightarrow x_{i1}, .., x_{i8}$中只有一个为真



n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

不同行:

第 i 行只有一个皇后

$\Rightarrow x_{i1}, .., x_{i8}$中只有一个为真

$\Rightarrow (x_{i1} \vee x_{i2} \vee ... \vee x_{i8})$

$\wedge (\neg x_{i1} \vee \neg x_{i2}) \wedge ... \wedge (\neg x_{i1} \vee \neg x_{i8})$

$\wedge (\neg x_{i2} \vee \neg x_{i3}) \wedge ... \wedge (\neg x_{i2} \vee \neg x_{i8})$

$\wedge ...$

$\wedge (\neg x_{i7} \vee \neg x_{i8})$

n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

不同行:

第 i 行只有一个皇后

$\Rightarrow x_{i1},..,x_{i8}$中只有一个为真

$\Rightarrow (x_{i1} \vee x_{i2} \vee ... \vee x_{i8})$

$\wedge (\neg x_{i1} \vee \neg x_{i2}) \wedge ... \wedge (\neg x_{i1} \vee \neg x_{i8})$

$\wedge (\neg x_{i2} \vee \neg x_{i3}) \wedge ... \wedge (\neg x_{i2} \vee \neg x_{i8})$

$\wedge ...$

$\wedge (\neg x_{i7} \vee \neg x_{i8})$

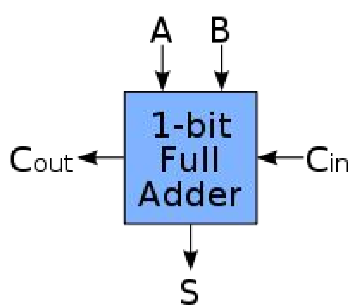
不同列

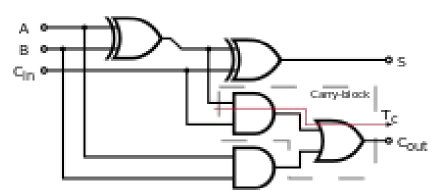
不同对角线

Circuit to SAT

- 加法电路的形式化（1-bit）
  - $A + B + C_{in} = C_{out}S \Leftrightarrow$
  - C_{out} = (A and B) or (C_{in} and (A or B))
  - S = A xor B xor C_{in}

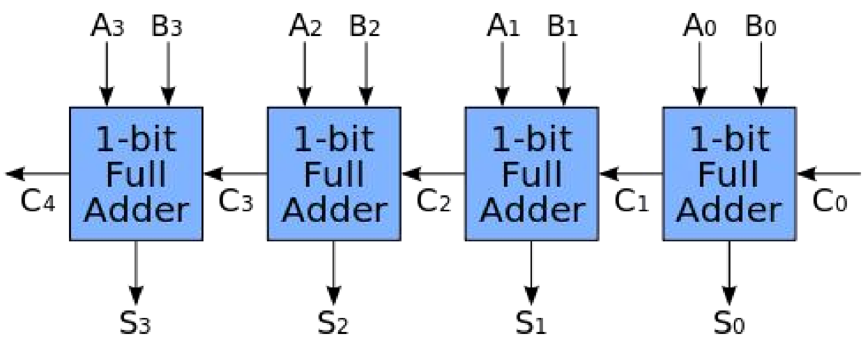
Inputs			Outputs	
A	*B*	C_{in}	C_{out}	*S*
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
1	1	0	1	0
0	0	1	0	1
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1



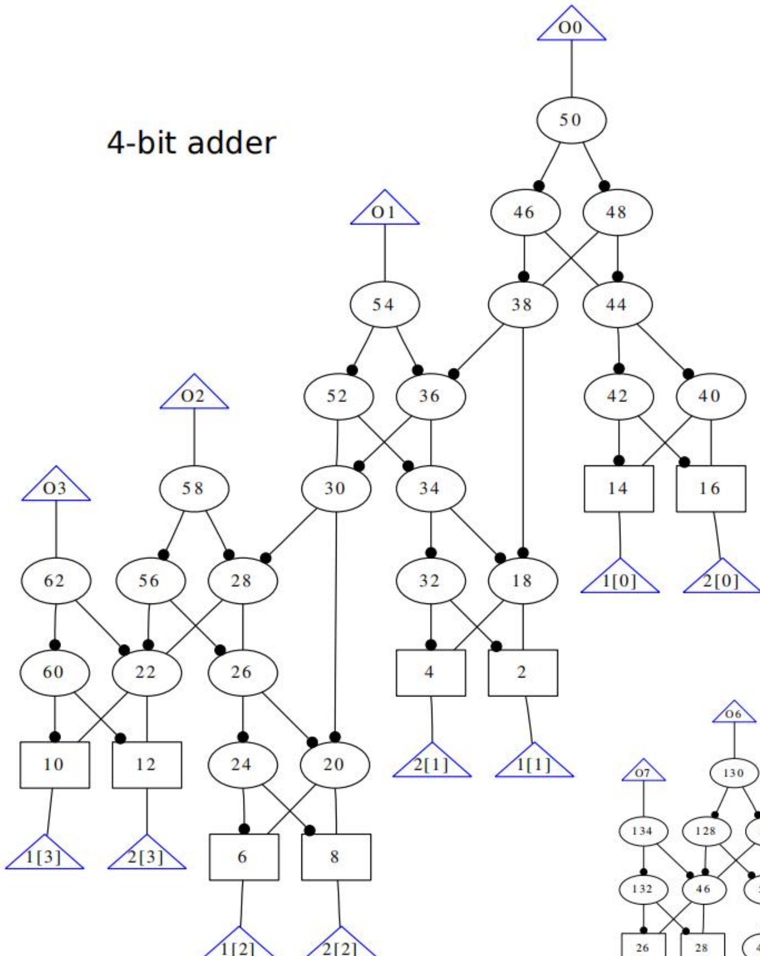


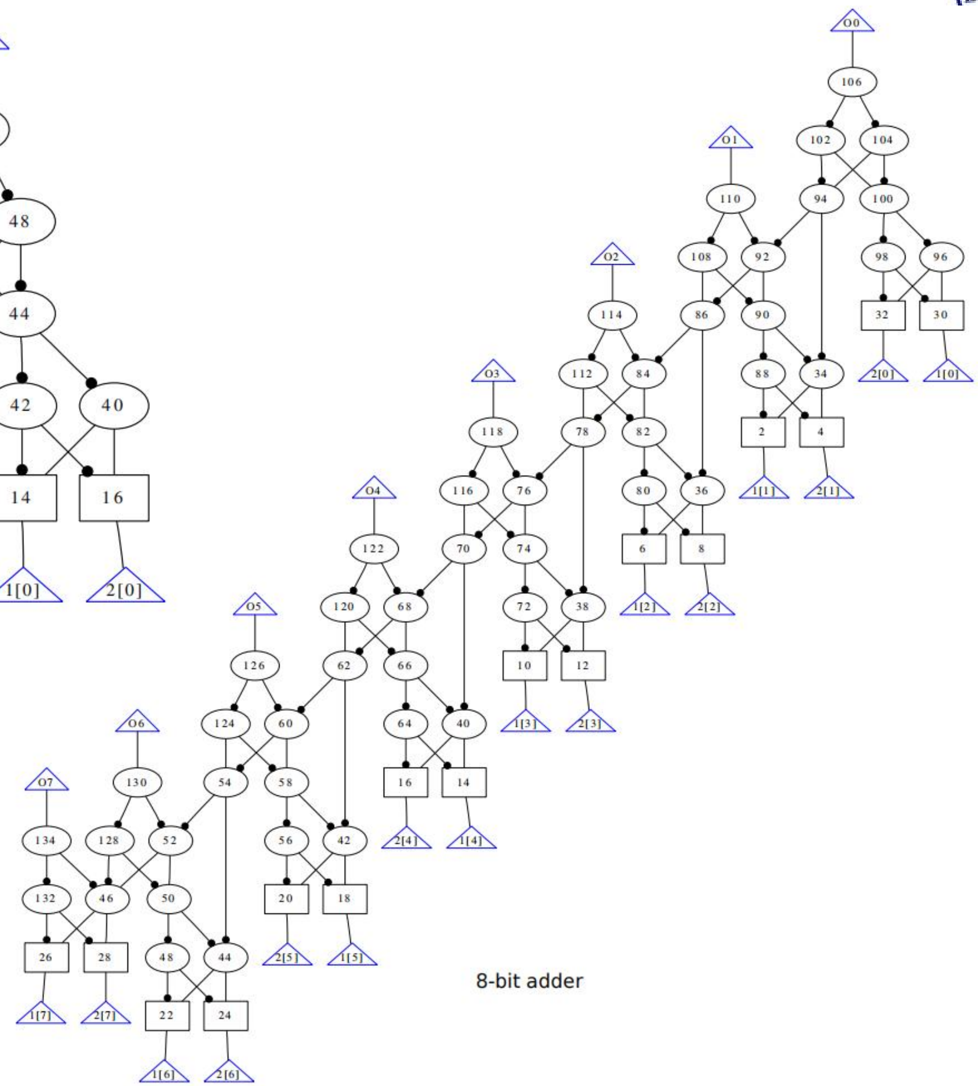
Circuit to SAT

- 加法电路的形式化（n-bit）



4-bit adder





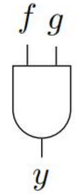
8-bit adder

Circuit to SAT

- 乘法 ⇔ 移位+加法

```
     1011   (this is 11 in decimal)
   x 1110   (this is 14 in decimal)
   ======
     0000   (this is 1011 x 0)
    1011    (this is 1011 x 1, shifted one position to the left)
   1011     (this is 1011 x 1, shifted two positions to the left)
+ 1011      (this is 1011 x 1, shifted three positions to the left)
=========
 10011010   (this is 154 in decimal)
```

Circuit to SAT



AND-Gate

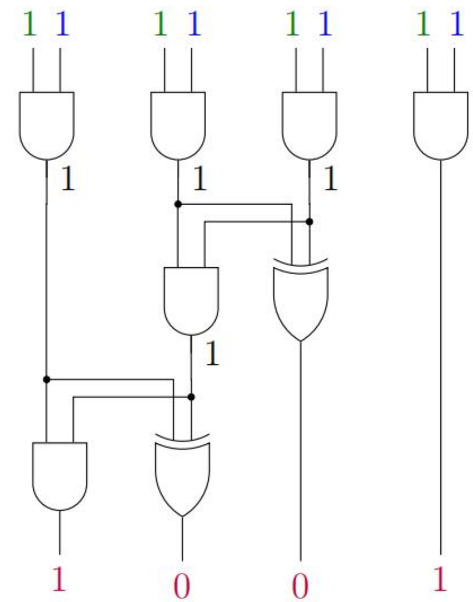
f	g	y
0	0	0
0	1	0
1	0	0
1	1	1

XOR-Gate

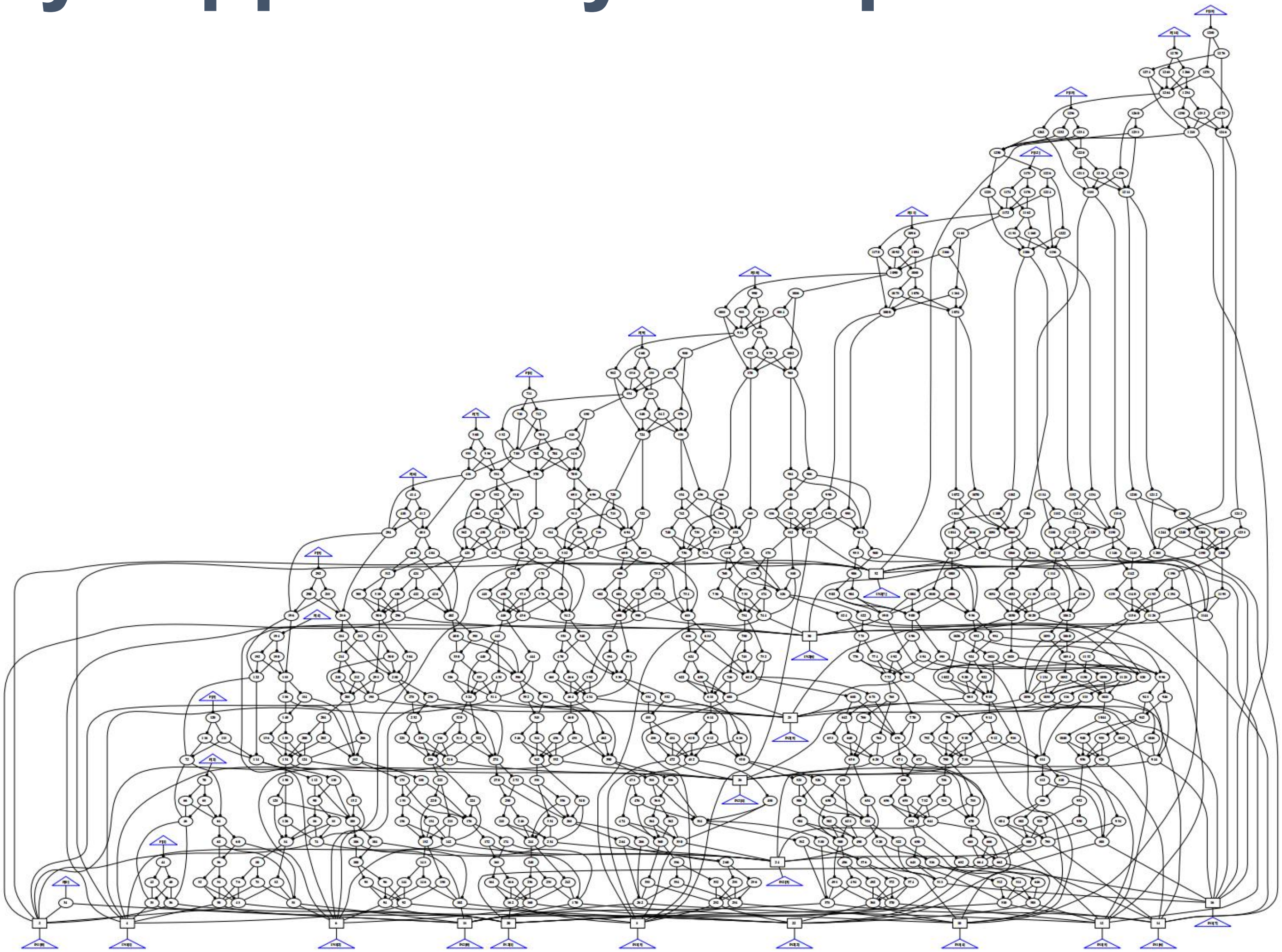
f	g	y
0	0	0
0	1	1
1	0	1
1	1	0

$$
\begin{array}{cccccccc}
 & & 1 & 1 & \cdot & 1 & 1 \\
\hline
 & & & & & 1 & 1 \\
 & & & {}_{1}\,1 & {}_{1}\,1 & {}_{0} & \\
\hline
 & & & 1 & 0 & 0 & 1
\end{array}
$$

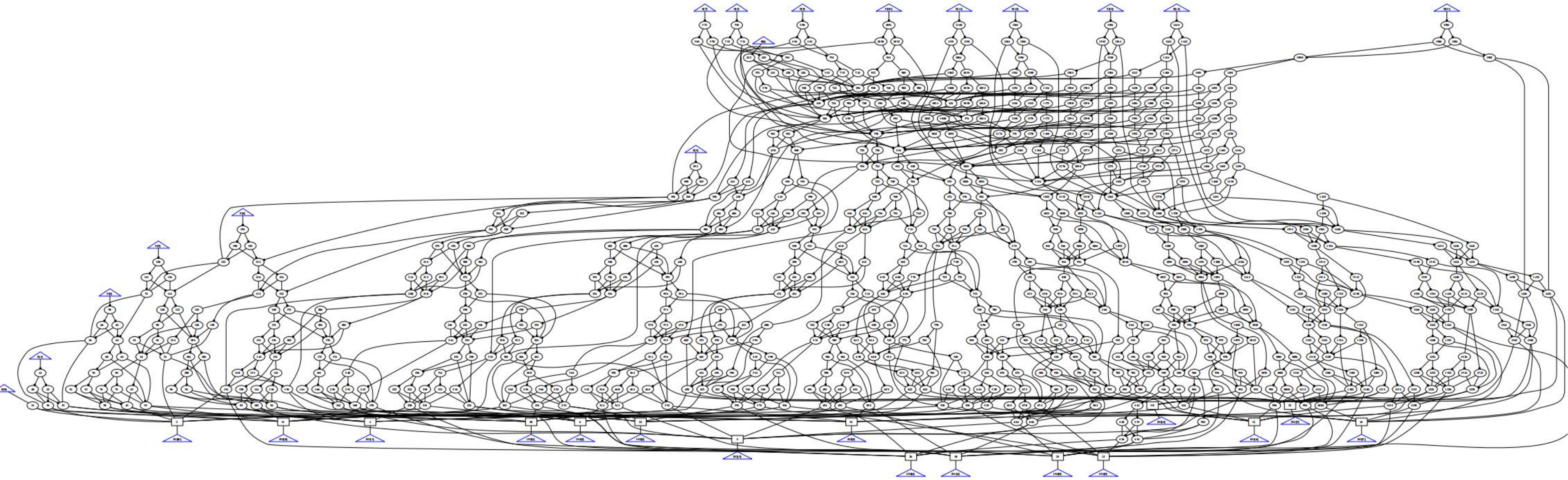
$3 \cdot 3 = 9$



Array Ripple Carry Multiplier



Wallace-Tree Carry-Lookahead Multiplier



Circuit to SAT

- 整数除法
  - 有余数，引入辅助变量表示余数

```
      11   R=10
11 )1011
    -11
     101
     -11
      10  <-- remainder, R
```

```
       0011
x      abcd
==========
         ef
         11  x d
        11   x c
       11    x b
      11     x a
=========
   00001011
```

SAT问题求解的应用

- Bounded Model Checking (Clarke, Emerson and Sifakis. 2007 Turing Award)
- Electronic Design Automation (EDA)
  - Widely used in many aspects of chip design: equivalence checking, assertion verification, synthesis, debugging, post-silicon validation
- Software Verification
- Automated Theorem Proving
  - Pythagorean Triples (200TB), Schur Number Five (2PB), Certification: Coq, ACL2, Isabelle
- AI and Planning problems

永真式

定义1.25. 设A为命题，v为赋值。

1. A 为**永真式**（也称**重言式**），记为 $\vDash A$，指对于 $\forall v$ 都有 $\hat{v}(A) = T$；
2. A 为**矛盾式**，指对于 $\forall v$ 都有 $\hat{v}(A) = F$；

例，$A \to A$，

$\neg\neg A \to A$，

$(A \wedge B) \to (B \wedge A)$.

永真式与矛盾式

- 一个公式是永真式或矛盾式或两者都不是。
- A 不是永真式当且仅当 $\neg A$ 是可满足的。
- A 不是矛盾式当且仅当 A 是可满足的。

真值表

例，$(A \vee B) \rightarrow (\neg B \wedge C)$

A	B	C	$(A \vee B)$	$\neg B$	$(\neg B \wedge C)$	$(A \vee B) \rightarrow (\neg B \wedge C)$
1	1	1	1	0	0	0
1	1	0	1	0	0	0
1	0	1	1	1	1	1
1	0	0	1	1	0	1
0	1	1	1	0	0	0
0	1	0	1	0	0	0
0	0	1	0	1	1	1
0	0	0	0	1	0	1

真值表

例，$(A \vee B) \rightarrow (\neg B \wedge C)$

A	B	C	(A	∨	B)	→	(¬	B	∧	C)
1	1	1	1	1	1		0	1		1
1	1	0	1	1	1		0	1		0
1	0	1	1	1	0		1	0		1
1	0	0	1	1	0		1	0		0
0	1	1	0	1	1		0	1		1
0	1	0	0	1	1		0	1		0
0	0	1	0	0	0		1	0		1
0	0	0	0	0	0		1	0		0

真值表

例，$(A \vee B) \rightarrow (\neg B \wedge C)$

A	B	C	(A	∨	B)	→	(¬	B	∧	C)
1	1	1	1	1	1		0	1	0	1
1	1	0	1	1	1		0	1	0	0
1	0	1	1	1	0		1	0	1	1
1	0	0	1	1	0		1	0	0	0
0	1	1	0	1	1		0	1	0	1
0	1	0	0	1	1		0	1	0	0
0	0	1	0	0	0		1	0	1	1
0	0	0	0	0	0		1	0	0	0

真值表

例，$(A \vee B) \rightarrow (\neg B \wedge C)$

A	B	C	(A	∨	B)	→	(¬	B	∧	C)
1	1	1	1	1	1	0	0	1	0	1
1	1	0	1	1	1	0	0	1	0	0
1	0	1	1	1	0	1	1	0	1	1
1	0	0	1	1	0	0	1	0	0	0
0	1	1	0	1	1	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0	0
0	0	1	0	0	0	1	1	0	1	1
0	0	0	0	0	0	1	1	0	0	0

真值表证明

证明 $(A \to B) \to (\neg B \to \neg A)$ 为永真式。

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

注：此处 $\vDash$ 也是元语言中的符号，

$\Gamma \vDash A$ 也可以读作“Γ逻辑地蕴含A”，

$\Gamma \vDash A$ 不是形式语言中的公式，是元语言中的命题。

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

$\Gamma \nvDash A$ 表示 $\Gamma \vDash A$ 不成立，

即存在赋值 v，使得 $v \vDash \Gamma$，但 $\hat{v}(A) = F$。

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

当 $\Gamma = \emptyset$ 时，$\Gamma \vDash A$ 变成 $\emptyset \vDash A$。

$\emptyset \vDash A$ 是什么涵义？

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

由定义知，$\emptyset \vDash A$ 是

对于 $\forall v$，$v \vDash \emptyset$ 蕴含 $v \vDash A$。 (1)

$v \vDash \emptyset$ 是

对于 $\forall B$，$B \in \emptyset$ 蕴含 $v \vDash B$。 (2)

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

由定义知，$\emptyset \vDash A$ 是

对于 $\forall v$，$v \vDash \emptyset$ 蕴含 $v \vDash A$。 (1)

$v \vDash \emptyset$ 是

对于 $\forall B$，$B \in \emptyset$ 蕴含 $v \vDash B$。 (2)

$B \in \emptyset$ 是假命题，(2)是真命题

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

由定义知，$\emptyset \vDash A$ 是

对于 $\forall v$，$v \vDash \emptyset$ 蕴含 $v \vDash A$。 (1)

即，对于 $\forall v$，$v \vDash A$。

也即，A是永真式。

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$,

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

直观上，$\Gamma \vDash A$ 表示 Γ 中的公式的真是 A 为真的充分条件。

由于 $\emptyset$ 中没有公式，所以 $\emptyset \vDash A$ 表示 A 是无条件为真。

即，A是永真式。

语义结论

定义1.26. 设$A \in PROP$，v为赋值，$\Gamma \subseteq PROP$。

A 是 Γ 的**语义结论**（也称**逻辑推论**），记为 $\Gamma \vDash A$，

指对所有 v，若$v \vDash \Gamma$，则 $v \vDash A$。

注：若$\Gamma = \{B\}$，$\Gamma \vDash A$ 也可写成 $B \vDash A$。

例，$A \to B, B \to C \vDash A \to C$。

证明：真值表

A	B	C	$A \to B$	$B \to C$	$A \to C$
1	1	1	1	1	1
1	1	0	1	0	0
1	0	1	0	1	1
1	0	0	0	1	0
0	1	1	1	1	1
0	1	0	1	0	1
0	0	1	1	1	1
0	0	0	1	1	1

例，$A \to B, B \to C \vDash A \to C$。

证明：反证法。

假设 $A \to B, B \to C \nvDash A \to C$，即存在赋值 v 使得

$$\hat{v}(A \to B) = 1, \tag{1}$$

$$\hat{v}(B \to C) = 1, \tag{2}$$

$$\hat{v}(A \to C) = 0. \tag{3}$$

例，$A \to B, B \to C \vDash A \to C$。

证明：反证法。

假设 $A \to B, B \to C \nvDash A \to C$，即存在赋值 v 使得

$$\hat{v}(A \to B) = 1, \tag{1}$$

$$\hat{v}(B \to C) = 1, \tag{2}$$

$$\hat{v}(A \to C) = 0. \tag{3}$$

由 (3) 可得

$$\hat{v}(A) = 1, \tag{4}$$

$$\hat{v}(C) = 0. \tag{5}$$

由 (1)(4) 得 $\hat{v}(B) = 1$，结合 (2) 得 $\hat{v}(C) = 1$，与 (5) 矛盾。

$\square$

例，$A \vee B, B \wedge \neg C \nvDash \neg A \wedge (B \rightarrow C)$.

证明：可以构造赋值 v，使得

$$\hat{v}(A) = 0, \hat{v}(B) = 1, \hat{v}(C) = 0.$$

那么

$$\hat{v}(A \vee B) = 1,$$

$$\hat{v}(B \wedge \neg C) = 1,$$

$$\hat{v}(\neg A \wedge (B \rightarrow C)) = 0.$$

$\square$

逻辑等价

定理1.27.

（1）$A_1, ..., A_n \vDash A$ 当且仅当 $\emptyset \vDash A_1 \wedge ... \wedge A_n \to A$；

（2）$A_1, ..., A_n \vDash A$ 当且仅当 $\emptyset \vDash A_1 \to (...(A_n \to A)...)$。

$\Rightarrow$：若$A_1, ..., A_n \vDash A$，则$A_1 \to (...(A_n \to A)...)$是永真式

反证法。

假设$A_1, ..., A_n \vDash A$时，$\exists v$ 使 $\hat{v}(A_1 \to (...(A_n \to A)...)) = 0$。

蕴含式为假当且仅当前件为真后件为假，可得$\hat{v}(A_1) = 1$...

逻辑等价

定理1.27.

（1）$A_1, \ldots, A_n \vDash A$ 当且仅当 $\emptyset \vDash A_1 \wedge \ldots \wedge A_n \rightarrow A$；

（2）$A_1, \ldots, A_n \vDash A$ 当且仅当 $\emptyset \vDash A_1 \rightarrow (\ldots(A_n \rightarrow A)\ldots)$。

⇐：若$A_1 \rightarrow (\ldots(A_n \rightarrow A)\ldots)$是永真式，则$A_1, \ldots, A_n \vDash A$

反证法。

逻辑等价

定义1.28. 设 A, B 为命题，A 与 B **逻辑等价**（也称**逻辑等值**），记为$A \simeq B$，指对于任意赋值 v，$v \vDash A$ 当且仅当 $v \vDash B$。

注：有如下等价的定义：

$A \simeq B$，当且仅当 $A \vDash B$ 且 $B \vDash A$。

任何赋值 v，$\hat{v}(A) = \hat{v}(B)$。

逻辑等价

命题1.29.

1. （自反性）$A \simeq A$；
2. （对称性）若 $A \simeq B$，则 $B \simeq A$；
3. （传递性）若 $A \simeq B$ 且 $B \simeq C$，则 $A \simeq C$；
4. 若 $A \simeq B$，则 $\neg A \simeq \neg B$；
5. 若 $A_1 \simeq B_1$ 且 $A_2 \simeq B_2$，则 $(A_1 * A_2) \simeq (B_1 * B_2)$。

逻辑等价

交换律与结合律

- $A \wedge B \simeq B \wedge A$
- $(A \wedge B) \wedge C \simeq A \wedge (B \wedge C)$
- $A \vee B \simeq B \vee A$
- $(A \vee B) \vee C \simeq A \vee (B \vee C)$

等值替换

定理1.30（等值替换）. 若 $B \simeq C$ 且在 A 中把 B 的某些出现替换为 C 而得到 A'，则 $A \simeq A'$。

例，$A = \neg B \wedge (B \to D)$，

$A' = \neg C \wedge (B \to D)$。

证明：对 A 的结构做归纳。

若 $B = A$，则 $C = A'$。

A 为以下形式之一（定理1.16）：

原子公式，$\neg A_1$，$A_1 \wedge A_2$，$A_1 \vee A_2$，$A_1 \to A_2$。

归纳基础：$A \in PS$，这时 $B = A$，故成立。

归纳假设：令A_1’和A_2’分别为A_1和A_2经过替换后的公式，

那么，$A_1 \simeq A_1$’且$A_2 \simeq A_2$’。

归纳步骤：

设 $A = \neg A_1$。

若 $B = A$，则如上述可知成立。

若 $B \neq A$，即B是A的真段，则B是A_1的段（定理1.20）。

此时 $A' = \neg A_1'$。

由归纳假设可知 $A_1 \simeq A_1'$，

根据命题1.29 (4)，得$\neg A_1 \simeq \neg A_1'$。

设 $A = A_1 * A_2$。

若 $B = A$，则如上述可知成立。

若 $B \neq A$，则 B 是 A_1的段或是 A_2 的段（定理1.20）。

此时 $A' = A_1' * A_2'$。

由归纳假设可知 $A_1 \simeq A_1'$，$A_2 \simeq A_2'$，

根据命题1.29 (5)，得$(A_1 * A_2) \simeq (A_1' * A_2')$。 □

决策树

例，$(A \vee B) \rightarrow (\neg B \wedge C)$

$$(A \vee B) \rightarrow (\neg B \wedge C)$$

A≃1　　　　A≃0

$$(1 \vee B) \rightarrow (\neg B \wedge C) \qquad (0 \vee B) \rightarrow (\neg B \wedge C)$$

$$(\neg B \wedge C) \qquad B \rightarrow (\neg B \wedge C)$$

决策树

- $\neg 1 \simeq 0$

  $\neg 0 \simeq 1$

- $A \wedge 1 \simeq A$

  $1 \wedge A \simeq A$

  $A \wedge 0 \simeq 0$

  $0 \wedge A \simeq 0$

- $A \vee 1 \simeq 1$

  $1 \vee A \simeq 1$

  $A \vee 0 \simeq A$

  $0 \vee A \simeq A$

- $A \rightarrow 1 \simeq 1$

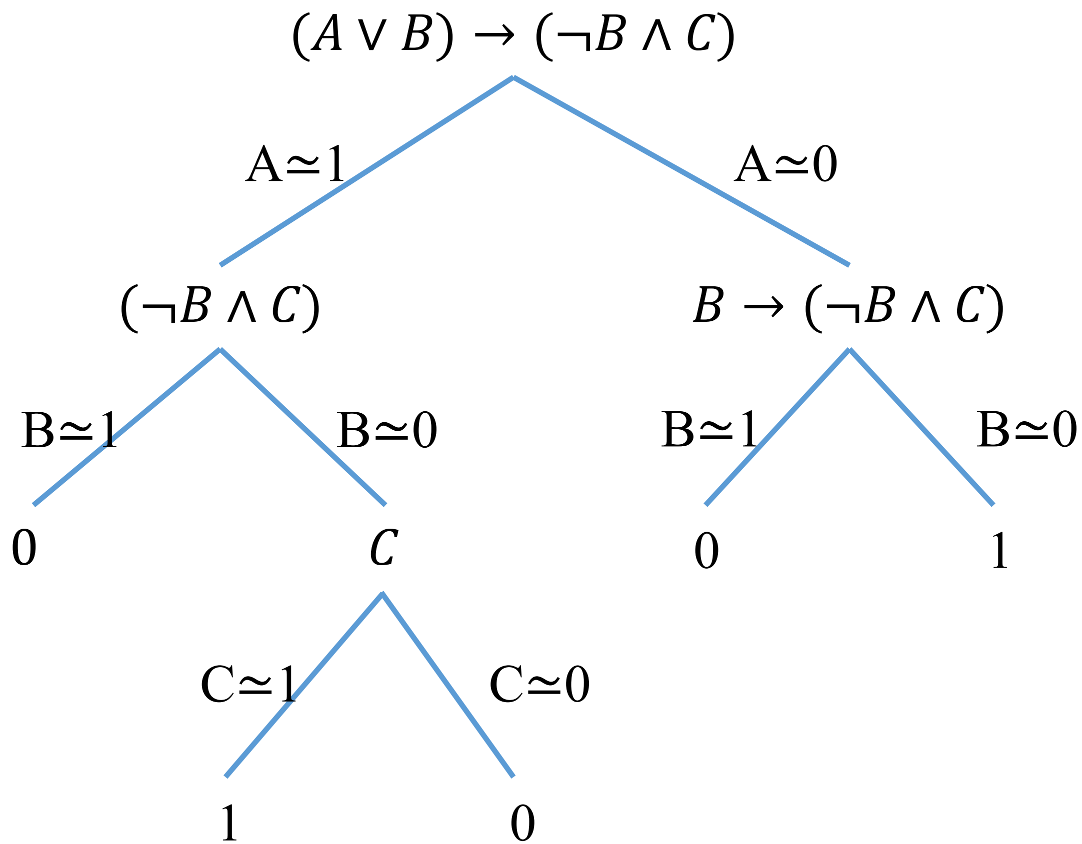
  $1 \rightarrow A \simeq A$

  $A \rightarrow 0 \simeq \neg A$

  $0 \rightarrow A \simeq 1$

决策树

例，$(A \vee B) \to (\neg B \wedge C)$



决策树

例，n取何值，公式

$$\underbrace{(\ldots((A \to A) \to A)\ldots) \to A}_{n\text{个}A}$$

是永真式？

$A \to A$

决策树

例，n取何值，公式

$$\underbrace{(\ldots((A \to A) \to A)\ldots) \to A}_{n\text{个}A}$$

是永真式？

$A \to A \simeq 1$

$(A \to A) \to A$

决策树

例，n取何值，公式

$$\underbrace{(\ldots((A \to A) \to A)\ldots) \to A}_{n\text{个}A}$$

是永真式？

$A \to A = 1$

$(A \to A) \to A \simeq 1 \to A \simeq A$

$((A \to A) \to A) \to A$

$\simeq A \to A \simeq 1$

命题与真值函数

定义1.31. 设 A 为命题，$FV(A) = \{Q_1, \ldots, Q_n\}$。

n元函数H_A: $\boldsymbol{B}^n \to \boldsymbol{B}$定义如下：

对于 $\forall(a_1, \ldots, a_n) \in \boldsymbol{B}^n$，$H_A(a_1, \ldots, a_n) = \hat{v}(A)$，

这里赋值 v 满足 $v(Q_i) = a_i \, (1 \leq i \leq n)$。

称 f 为**n元真值函数**，称H_A为**由A定义的真值函数**。

命题与真值函数

例，设 A 为$(P \land \neg Q) \lor (\neg P \land Q)$，那么 $H_A\colon \boldsymbol{B}^2 \to \boldsymbol{B}$为不可兼或运算（也称异或）。

P	Q	A	$H_A(\boldsymbol{p},\boldsymbol{q})$
1	1	0	0
1	0	1	1
0	1	1	1
0	0	0	0

$$P \oplus Q \simeq (P \land \neg Q) \lor (\neg P \land Q)$$

命题与真值函数

命题1.32. 设$FV(A \wedge B) = \{Q_1, \dots, Q_n\}$，且$H_A$: $\boldsymbol{B}^n \to \boldsymbol{B}$，$H_B$: $\boldsymbol{B}^n \to \boldsymbol{B}$。我们有 $A \simeq B$ 当且仅当 $H_A = H_B$。

任意两个具有相同命题符集的命题，它们逻辑等价 当且仅当 它们定义的真值函数相等。

析取范式-合取范式

定义1.33（文字，子句）.

（1）命题符和命题符的否定式称为**文字**（Literal）；

（2）以文字为析（合）取项的析（合）取式称为**析（合）取子式**，简称**子式**，也称**子句**（Clause）。

析取范式-合取范式

定义1.34（范式 Normal Form）.

（1）命题A为**析取范式**（∨∧-nf，DNF），指A为 m 个合取子式的析取式，呈形$\bigvee_{i=1}^{m}(\bigwedge_{k=1}^{n_i}P_{i,k})$。

（2）命题A为**合取范式**（∧∨-nf，CNF），指A为 l 个析取子式的合取式，呈形$\bigwedge_{j=1}^{l}(\bigvee_{k=1}^{n_j}Q_{j,k})$。

以上

- $\bigwedge_{k=1}^{n}B_k$ 为 $(\ldots(((B_1\wedge B_2)\wedge B_3)\ldots\wedge B_n)\ldots)$ 的简写；
- $\bigvee_{k=1}^{n}B_k$ 为 $(\ldots(((B_1\vee B_2)\vee B_3)\ldots\vee B_n)\ldots)$ 的简写。

析取范式-合取范式

析取范式$\bigvee_{i=1}^{m}(\bigwedge_{k=1}^{n} P_{i,k})$为如下形式：

$$(P_{11} \wedge \ldots \wedge P_{1n_1}) \vee \ldots \vee (P_{m1} \wedge \ldots \wedge P_{mn_m}),$$

文字

析取范式-合取范式

析取范式$\vee_{i=1}^{m}(\wedge_{k=1}^{n} P_{i,k})$为如下形式：

$$\underbrace{(P_{11} \wedge \ldots \wedge P_{1n_1})}_{\text{子句}} \vee \ldots \vee \underbrace{(P_{m1} \wedge \ldots \wedge P_{mn_m})}_{\text{子句}},$$

析取范式-合取范式

析取范式$\bigvee_{i=1}^{m}(\bigwedge_{k=1}^{n} P_{i,k})$为如下形式：

$$(P_{11} \wedge \ldots \wedge P_{1n_1}) \vee \ldots \vee (P_{m1} \wedge \ldots \wedge P_{mn_m}),$$

合取范式$\bigwedge_{j=1}^{l}(\bigvee_{k=1}^{n} Q_{j,k})$为如下形式：

$$(Q_{11} \wedge \ldots \wedge Q_{1n_1}) \vee \ldots \vee (Q_{l1} \wedge \ldots \wedge Q_{ln_l}).$$

析取范式-合取范式

析取范式$\bigvee_{i=1}^{m}(\bigwedge_{k=1}^{n} P_{i,k})$为如下形式：

$$(P_{11} \wedge \ldots \wedge P_{1n_1}) \vee \ldots \vee (P_{m1} \wedge \ldots \wedge P_{mn_m}),$$

合取范式$\bigwedge_{j=1}^{l}(\bigvee_{k=1}^{n} Q_{j,k})$为如下形式：

$$(Q_{11} \wedge \ldots \wedge Q_{1n_1}) \vee \ldots \vee (Q_{l1} \wedge \ldots \wedge Q_{ln_l}).$$

一个析（合）取范式的每个子句都包含这个公式的所有原子公式，且每个子句都不相同，称为**完全析（合）取范式**。

析取范式-合取范式

例，

（1） p

（2） $\neg p \vee q$

（3） $\neg p \wedge q \wedge \neg r$

（4） $\neg p \vee (q \wedge \neg r)$

（5） $\neg p \wedge (q \vee \neg r) \wedge (\neg q \vee r)$

析取范式-合取范式

∨∧-nf：$A = \bigvee_{i=1}^{m} (\bigwedge_{k=1}^{n_i} P_{i,k})$

令$n = |FV(A)|$，也可写成

$$\bigvee_{i=1}^{m} (\bigwedge_{k=1}^{n} P_{i,k})$$

任意真值函数均可表示为范式

定理1.35. 设$f: \boldsymbol{B}^n \to \boldsymbol{B}$，

（1）存在命题A，其为 ∨∧-nf 使 $f = H_A$；

（2）存在命题A'，其为 ∧∨-nf 使 $f = H_{A'}$。

证明：

设$f: \boldsymbol{B}^n \to \boldsymbol{B}$，$P_1, ..., P_n$为n个命题符。令

- $T_f = \{(x_1, ..., x_n) \in \boldsymbol{B}^n | f(x_1, ..., x_n) = T\}$，
- $F_f = \{(x_1, ..., x_n) \in \boldsymbol{B}^n | f(x_1, ..., x_n) = F\}$。

由于T_f和F_f是有穷集合，可设

- $T_f = \{(a_{i1}, ..., a_{in}) \in \boldsymbol{B}^n | 1 \leq i \leq m\}$，
- $F_f = \{(b_{j1}, ..., b_{jn}) \in \boldsymbol{B}^n | 1 \leq j \leq l\}$，

其中$m + l = 2^n$。

令

$$P_{i,k}^{*}=\begin{cases}P_k, & 若 a_{ik}=T,\\ \neg P_k, & 若 a_{ik}=F.\end{cases}$$

$$A=\bigvee_{i=1}^{m}\left(\bigwedge_{k=1}^{n}P_{i,k}^{*}\right).$$

令

$$P_{i,k}^{*}=\begin{cases}P_k, & 若 a_{ik}=T,\\ \neg P_k, & 若 a_{ik}=F.\end{cases}$$

$$A=\bigvee_{i=1}^{m}\left(\bigwedge_{k=1}^{n}P_{i,k}^{*}\right).$$

又令

$$Q_{j,k}^{*}=\begin{cases}\neg P_k, & 若 b_{jk}=T,\\ P_k, & 若 b_{jk}=F.\end{cases}$$

$$A'=\bigvee_{j=1}^{l}\left(\bigwedge_{k=1}^{n}Q_{j,k}^{*}\right).$$

显然$FV(A)=\{P_1,\ldots,P_n\}$。

定义1.31

下证$H_A = f$，

只需证：令$v(P_i) = x_i$，有$f(x_1, \ldots, x_n) = \hat{v}(A) = H_A$。

只需证：$\hat{v}(A) = T \text{ iff } (x_1, \ldots, x_n) \in T_f$，

即，$v \vDash A \text{ iff } (x_1, \ldots, x_n) \in T_f$。

$$v \vDash A \text{ iff } v \vDash \bigvee_{i=1}^{m} \left(\bigwedge_{k=1}^{n} P_{i,k}^{*} \right)$$

$$\text{iff } \exists i \leq m \text{ 使 } v \vDash \bigwedge_{k=1}^{n} P_{i,k}^{*}$$

$$\text{iff } \exists i \leq m \text{ 使 } \forall k \leq n \text{ 有 } v \vDash P_{i,k}^{*}$$

$$\text{iff } \exists i \leq m \text{ 使 } \forall k \leq n \text{ 有 } \hat{v}(P_{i,k}^{*}) = T$$

$$P^*_{i,k} = \begin{cases} P_k, & 若 a_{ik} = T, \\ \neg P_k, & 若 a_{ik} = F. \end{cases}$$

$v \vDash A$ iff $v \vDash \bigvee_{i=1}^{m} \left(\bigwedge_{k=1}^{n} P^*_{i,k} \right)$

iff $\exists i \le m$ 使 $\forall k \le n$ 有 $\hat{v}(P^*_{i,k}) = T$

iff $\exists i \le m$ 使 $\forall k \le n$ 有 $v(P_k) = a_{ik}$

iff $\exists i \le m$ 使 $\forall k \le n$ 有 $x_k = a_{ik}$

iff $\exists i \le m$ 使 $(x_1, \ldots, x_n) = (a_{i1}, \ldots, a_{in})$

iff $(x_1, \ldots, x_n) \in T_f$

所以 $H_A = f$，同理可证 $H_{A'} = f$。 □

任意命题均有逻辑等价的范式

命题1.36. 若 A 为命题，则存在合取范式 B 和析取范式 B' 使$A \simeq B$ 且 $A \simeq B'$，也称 B 和 B' 分别为 A 的合取范式和析取范式。

由命题1.32和定理1.35（任意真值函数均可表示为范式）可证。

> **命题1.32.** 设$FV(A \wedge B) = \{Q_1, \ldots, Q_n\}$，且$H_A: \boldsymbol{B}^n \to \boldsymbol{B}$，$H_B: \boldsymbol{B}^n \to \boldsymbol{B}$。我们有 $A \simeq B$ 当且仅当 $H_A = H_B$。

例，求$((P \wedge Q) \to R) \wedge P$的析取范式和合取范式。

例，求$((P\wedge Q)\to R)\wedge P$的析取范式和合取范式。

解：不妨设$P,Q,R\in PS.$

真值表如下，

P	Q	R	$((P\wedge Q)\to R)\wedge P$	∨∧-nf	∧∨-nf
1	1	1	1		
1	1	0	0		
1	0	1	1		
1	0	0	1		
0	1	1	0		
0	1	0	0		
0	0	1	0		
0	0	0	0		

例，求$((P\wedge Q)\to R)\wedge P$的析取范式和合取范式。

解：不妨设$P,Q,R\in PS.$

真值表如下，

$$P_{i,k}^{*}=\begin{cases}P_k, & 若a_{ik}=T,\\ \neg P_k, & 若a_{ik}=F.\end{cases}$$
$$(x_1,\ldots,x_n)\in T_f$$

P	Q	*R*	$((P\wedge Q)\to R)\wedge P$	∨∧-nf	∧∨-nf
1	1	1	1	$P\wedge Q\wedge R$	
1	1	0	0		
1	0	1	1	$P\wedge\neg Q\wedge R$	
1	0	0	1	$P\wedge Q\wedge\neg R$	
0	1	1	0		
0	1	0	0		
0	0	1	0		
0	0	0	0		

例，求$((P \wedge Q) \to R) \wedge P$的析取范式和合取范式。

解：不妨设$P, Q, R \in PS$.

真值表如下，

$$Q_{j,k}^{*} = \begin{cases} \neg P_k, & 若 b_{jk} = T, \\ P_k, & 若 b_{jk} = F. \end{cases}$$
$$(x_1, \ldots, x_n) \in F_f$$

P	Q	R	$((P \wedge Q) \to R) \wedge P$	∨∧-nf	∧∨-nf
1	1	1	1	$P \wedge Q \wedge R$	
1	1	0	0		$\neg P \vee \neg Q \vee R$
1	0	1	1	$P \wedge \neg Q \wedge R$	
1	0	0	1	$P \wedge Q \wedge \neg R$	
0	1	1	0		$P \vee \neg Q \vee \neg R$
0	1	0	0		$P \vee \neg Q \vee R$
0	0	1	0		$P \vee Q \vee \neg R$
0	0	0	0		$P \vee Q \vee R$

它的析取范式：

$$(P \land Q \land R) \lor (P \land \neg Q \land R) \lor (P \land Q \land \neg R),$$

它的合取范式：

$$(\neg P \lor \neg Q \lor R) \land (P \lor \neg Q \lor \neg R) \land (P \lor \neg Q \lor R) \land (P \lor Q \lor \neg R) \land (P \lor Q \lor R).$$

□

n-皇后问题

$x_{ij} = T$ 表示 (i,j) 处有皇后

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}

不同行:

第 i 行只有一个皇后

$\Rightarrow x_{i1}, .., x_{i8}$中只有一个为真

$\Rightarrow (x_{i1} \vee x_{i2} \vee ... \vee x_{i8})$

$\wedge (\neg x_{i1} \vee \neg x_{i2}) \wedge ... \wedge (\neg x_{i1} \vee \neg x_{i8})$

$\wedge (\neg x_{i2} \vee \neg x_{i3}) \wedge ... \wedge (\neg x_{i2} \vee \neg x_{i8})$

$\wedge ...$

$\wedge (\neg x_{i7} \vee \neg x_{i8})$

析取范式-合取范式

∨∧-nf：$A = \bigvee_{i=1}^{m} C_i = \bigvee_{i=1}^{m} (\bigwedge_{k=1}^{n} P_{i,k})$

- 给定一个赋值v，若存在i，使得$v \vDash C_i$，则$v \vDash A$。
- 判断A的可满足性？
- 对于任意公式 F，找到析取范式的公式 H，使得 F 可满足当且仅当 H 可满足，这一过程不易于求解公式的可满足性问题。

析取范式-合取范式

∧∨-nf：$A = \bigwedge_{j=1}^{l} C_j = \bigwedge_{j=1}^{l}(\bigvee_{k=1}^{n} Q_{j,k})$

- 给定一个赋值v，若$v \vDash A$，则 v 同时满足所有子句C_j。
- 对于任意公式 F，存在多项式时间的算法找到合取范式的公式 H，使得 F 可满足当且仅当 H 可满足。

析取范式-合取范式

设A为命题，$\neg A$为A的**补式**，A和$\neg A$为**互补公式**。

定理1.37.

（1）一个析取范式是矛盾式，当且仅当它的每个（合取）子式是矛盾式，即每个子式含互补的文字。

（2）一个合取范式是永真式，当且仅当它的每个（析取）子式是永真式，即每个子式含互补的文字。

析取范式-合取范式

推论1.38.

（1）一个公式是矛盾式，当且仅当它的析取范式的每个（合取）子式含互补的文字。

（2）一个公式是永真式，当且仅当它的合取范式的每个（析取）子式含互补的文字。

等值替换

（1）$A \to B \simeq \neg A \vee B$；

（2）$A \leftrightarrow B \simeq (\neg A \vee B) \wedge (A \vee \neg B)$；

（3）$A \leftrightarrow B \simeq (A \wedge B) \vee (\neg A \wedge \neg B)$；

（4）$A \oplus B \simeq (A \wedge \neg B) \vee (\neg A \wedge B) \simeq \neg(A \leftrightarrow B)$；

(1)~(4)：消去→，↔，⊕

可以使用等值替换（定理1.30），

在原公式中把(1)~(4)左边的公式替换成右边。

等值替换

（1）$A \to B \simeq \neg A \lor B$；

（2）$A \leftrightarrow B \simeq (\neg A \lor B) \land (A \lor \neg B)$；

（3）$A \leftrightarrow B \simeq (A \land B) \lor (\neg A \land \neg B)$；

（4）$A \oplus B \simeq (A \land \neg B) \lor (\neg A \land B) \simeq \neg(A \leftrightarrow B)$；

（5）$\neg\neg A \simeq A$；

（6）$\neg(A_1 \land \ldots \land A_n) \simeq \neg A_1 \lor \ldots \lor \neg A_n$；

（7）$\neg(A_1 \lor \ldots \lor A_n) \simeq \neg A_1 \land \ldots \land \neg A_n$；

(1)~(4)：消去$\to$，$\leftrightarrow$，$\oplus$

(5)~(7)：消去$\neg$的辖域中的$\neg$，$\land$，$\lor$

等值替换

（1）$A \to B \simeq \neg A \vee B$；

（2）$A \leftrightarrow B \simeq (\neg A \vee B) \wedge (A \vee \neg B)$；

（3）$A \leftrightarrow B \simeq (A \wedge B) \vee (\neg A \wedge \neg B)$；

（4）$A \oplus B \simeq (A \wedge \neg B) \vee (\neg A \wedge B) \simeq \neg(A \leftrightarrow B)$；

（5）$\neg\neg A \simeq A$；

（6）$\neg(A_1 \wedge \ldots \wedge A_n) \simeq \neg A_1 \vee \ldots \vee \neg A_n$；

（7）$\neg(A_1 \vee \ldots \vee A_n) \simeq \neg A_1 \wedge \ldots \wedge \neg A_n$；

（8）$A \wedge (B_1 \vee \ldots \vee B_n) \simeq (A \wedge B_1) \vee \ldots \vee (A \wedge B_n)$；

（9）$A \vee (B_1 \wedge \ldots \wedge B_n) \simeq (A \vee B_1) \wedge \ldots \wedge (A \vee B_n)$.

(1)~(4)：消去→，↔，⊕

(5)~(7)：消去¬的辖域中的¬，∧，∨

(8)：消去∧的辖域中的∨
(9)：消去∨的辖域中的∧

例，求$\neg((P \wedge Q) \to R)$的析取范式与合取范式。

解：$\neg((P \wedge Q) \to R)$

$\simeq \neg(\neg(P \wedge Q) \vee R)$

$\simeq \neg\neg(P \wedge Q) \wedge \neg R$

$\simeq P \wedge Q \wedge \neg R.$

等值替换

（10）$A \vee A \simeq A$

（11）$A \wedge A \simeq A$

(10)(11)：重复项

（12）$A \vee (A \wedge B) \simeq A$

（13）$A \wedge (A \vee B) \simeq A$

(12)(13)：一个子句的所有文字出现在另一个子句中

（14）$A \vee (B \wedge \neg B \wedge C) \simeq A$

（15）$A \wedge (B \vee \neg B \vee C) \simeq A$

(14)(15)：删去含互补文字的子句

联结词的完全组

（1）$A \to B \simeq \neg A \vee B$；

（2）$A \leftrightarrow B \simeq (\neg A \vee B) \wedge (A \vee \neg B)$；

（3）$A \leftrightarrow B \simeq (A \wedge B) \vee (\neg A \wedge \neg B)$；

（4）$A \oplus B \simeq (A \wedge \neg B) \vee (\neg A \wedge B) \simeq \neg(A \leftrightarrow B)$。

可以说，$\to$, $\leftrightarrow$, $\oplus$可以由$\neg$, $\wedge$, $\vee$定义。

联结词的完全组

一元联结词（4种）

f_3为$\neg$

A	$f_1(A)$	$f_2(A)$	$f_3(A)$	$f_4(A)$
1	1	1	0	0
0	1	0	1	0

联结词的完全组

二元联结词（16种）

g_2为∨，g_4为→，g_{12}为∧。

A	B	$g_1(A,B)$	g_2	g_3	g_4	g_5	g_6	g_7	g_8
1	1	1	1	1	1	0	1	1	1
1	0	1	1	1	0	1	1	0	0
0	1	1	1	0	1	1	0	1	0
0	0	1	0	1	1	1	0	0	1

A	B	g_9	g_{10}	g_{11}	g_{12}	g_{13}	g_{14}	g_{15}	g_{16}
1	1	0	0	0	1	0	0	0	0
1	0	1	1	0	0	1	0	0	0
0	1	1	0	1	0	0	1	0	0
0	0	0	1	1	0	0	0	1	0

联结词的完全组

n元联结词（2^{2^n}种）。

“如果…那么…否则”是一个三元联结词。

一个命题公式A，它的真值函数H_A是一个n元真值函数，对应一个n元联结词。

联结词的完全组

称联结词的集合是完备的，当且仅当任意n元的联结词都能由集合中的联结词定义。

由定理1.35可知，对于任何n元真值函数f，存在命题A，其中仅使用联结词$\neg, \wedge, \vee$使$f = H_A$。因此有如下定理：

定理1.39. $\{\neg, \wedge, \vee\}$是联结词的完全组。

联结词的完全组

又由于

- $A \wedge B \simeq \neg(\neg A \vee \neg B)$
- $A \vee B \simeq \neg(\neg A \wedge \neg B)$

故，有如下结论：

推论1.40. $\{\neg, \wedge\}$，$\{\neg, \vee\}$，$\{\neg, \rightarrow\}$是联结词的完全组。

- 联结符号g_5与g_{15}也构成联结词的完全组。

小结

- 命题逻辑的语义
  - 赋值、解释、可满足、永真、语义结论、逻辑等价
- 证明方法
  - 真值表、决策树、等值替换
- 析取范式与合取范式
- 联结词的完全组

$A \simeq B$ 语义层面相等
$A = B$ 表达式层面相等