命题的逻辑

对应《问题求解》
问题的提出: 为什么要研究简单的数理逻辑; 开始行动: 符号化逻辑;命题推演的真象: 命题的推演;
变得更加严谨: 形式化推理系统

为什么要学习命题的逻辑?

1. 对于数学的学习 — 分析长概念的层次

- 高等数学 — 极限的定义

请你否定一下这个命题

- 线性代数 — 线性相关的定义

请你否定一下这个命题

为什么要学习命题的逻辑?

2. 对于程序设计

- if(a && !a){…}

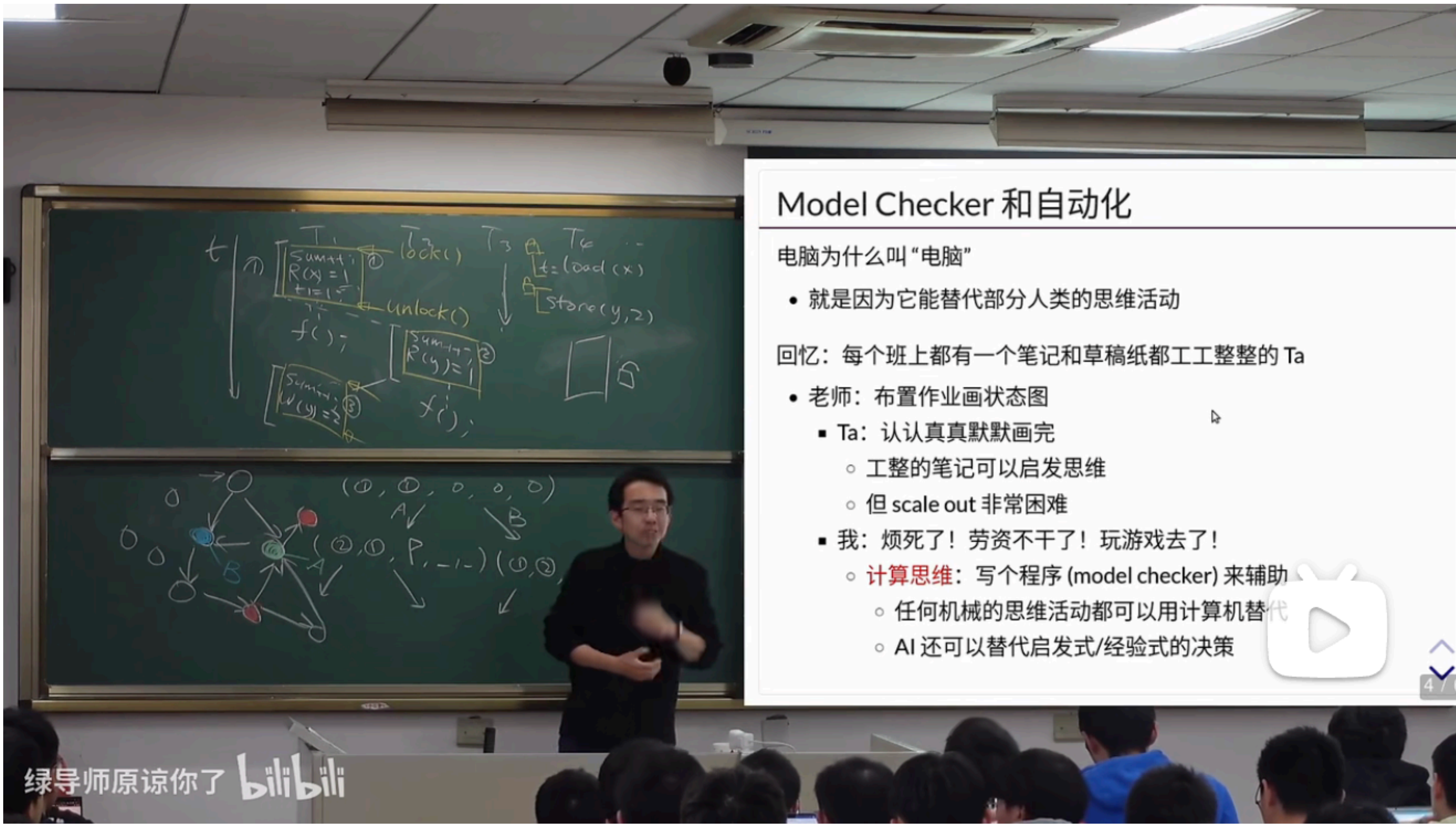
你这写了个啥?

3. 对于求解更加有趣的问题

比如使用一些逻辑约束条件求解数独问题.
用更加powerful的内容来解答实际问题!

为什么要学习命题的逻辑?

为了获得更多的力量!



蒋炎岩 操作系统 *P6*

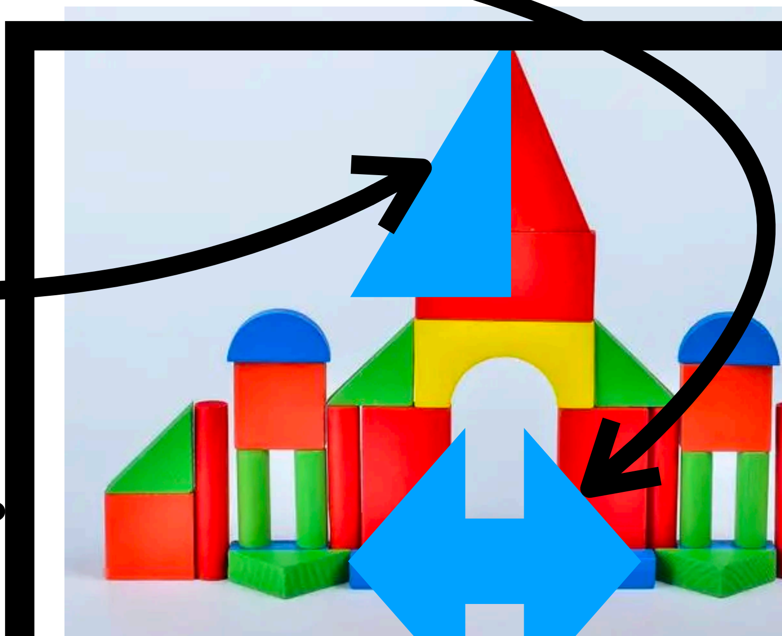
一. 逻辑的引擎

1. 命题: 能够判断真假的陈述句
    - 能够判断真假 — 不能是悖论(这句话是假的)

      — 不能含有变量(X+2>3)
    - 陈述句
2. 真值: 对命题的判断结果
    - 取值: 只有真(T)/ 假(F)
3. 命题的构成

(一) 形式上来看

- 原子命题: 不含有 “与, 或, 非” 这样的连接词语. 其真假性一般由人们的常识确定.
- 复合命题: 可以分解为更加简单的命题



一. 逻辑的引擎

3. 命题的构成

(二) 连接词语

$\land$ (\land) $\lor$ (\lor) $\lnot$(\lnot)

和 或 非

可以用它们表示所有的命题的逻辑!

合取（与）

P	Q	P∧Q
0	0	0
0	1	0
1	0	0
1	1	1

析取（或）

P	Q	P∨Q
0	0	0
0	1	1
1	0	1
1	1	1

否定（非）

P	¬P
0	1
1	0

一. 逻辑的引擎

3. 命题的构成

(二) 连接词语

数学文本里面的若p则q是什么意思?

当且仅当(iff, 等价)是什么意思?

- 若p则q

p满足	q满足	✓	没有说谎
	q不满足	✗	说谎了! 若p但是推断不出q
p不满足	q满足	✓	没有说谎
	q不满足	✓	没有说谎

一. 逻辑的引擎

3. 命题的构成

(二) 连接词语

数学文本里面的若p则q是什么意思?

当且仅当(iff, 等价)是什么意思?

- 当且仅当: 若p则q, 并且若q则p

一. 逻辑的引擎

3. 命题的构成

(二) 连接词语

数学文本里面的若p则q是什么意思?

当且仅当(iff, 等价)是什么意思?

若p则q

P	Q	P→Q
0	0	1
0	1	1
1	0	0
1	1	1

等价

P	Q	P↔Q
0	0	1
0	1	0
1	0	0
1	1	1

二. 命题的公式

$$((A \wedge B) \to C) \leftrightarrow (D \wedge \neg E \wedge \text{True})$$

Syntax	Schematics

1. 命题公式的构成
   - 字母表:
     - 任意多的命题符号
     - 5个逻辑连接词
     - 左括号, 右括号
   - 单词(句子) — 归纳的定义
     - 命题符号是公式
     - 如果P, Q是命题公式, $\neg(P), (P * Q)$, 是命题公式
     - 仅仅用(1)(2)执行有限次的都是公式.

"a,b,c"

I love apple

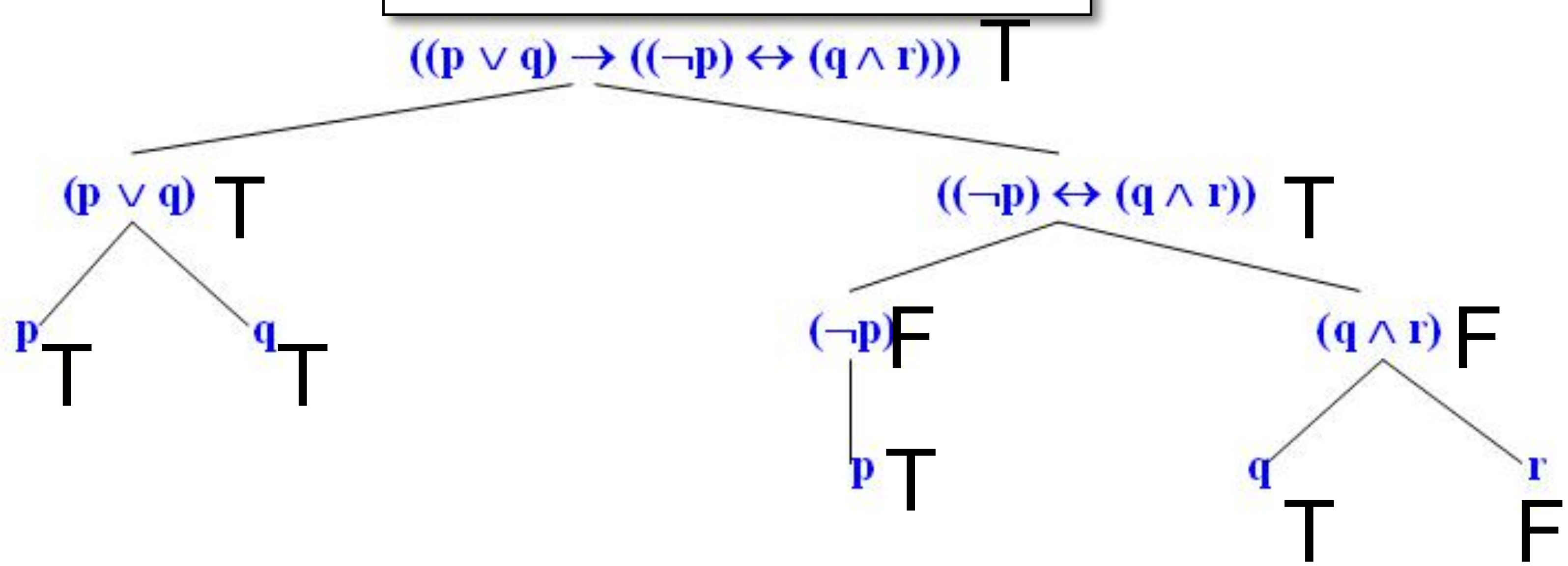
二. 命题的公式

$$((A \land B) \rightarrow C) \leftrightarrow (D \land \neg E \land \text{True})$$

Syntax | Schematics

2. 命题公式的语义
- 运算律: 优先级(非, 和, 或, →, ↔), 从右往左结合
  忘记这些! 可以用加括号的方法化解.
- 运算: 通过逻辑连接词的语义 “替换”

假设知道了一些真假...



二. 命题的公式

$$((A \wedge B) \to C) \leftrightarrow (D \wedge \neg E \wedge \text{True})$$

Syntax	Schematics

3. 考察所有的内容: 指派和指派的扩展
   - 指派: 把这些命题的变元每个安排一个真假的值
   - 指派扩张: 用一组真假值分别考察这一些变元.

把所有的情况都枚举出来了, 我们就知道了这个命题到底是什么德行.

4. 命题公式的类型

我们可以画真值表解答这个问题...

   - 重言式, 永真式: 任意的解释, 公式均为真. 如$p \to p$
   - 矛盾式, 永假式: 任意的解释, 公式均为假. 如$p \wedge \neg p$
   - 可满足式: 存在解释使得公式是真的, 也存在是假的.

二. 命题的公式

$$((A \wedge B) \to C) \leftrightarrow (D \wedge \neg E \wedge \text{True})$$

Syntax	Schematics

4. 命题公式的类型

判定方法(1). 画真值表

判定方法(2). Quine方法

记号$W[P/x]$表示用x替换W中的所有的P得到的结果

- W是永真的 iff W[P/真]和 W[P/假] 都是永真的
- W是永真的 iff W[P/真]和 W[P/假] 都是永假的

二. 命题的运算 — 命题与命题之间的关系

1. 命题的等价

定义 2.2.11 (命题的等价). 设 G, H 是两个命题公式, $P_1, \cdots, P_n$ 是出现在命题 G, H 中的所有变元, 如果对于 $P_1, P_2, \cdots, P_n$ 的 2^n 组不同的解释, G, H 的真值结果都相同, 那么称 G 和 H 是等价的, 记作 $G \Leftrightarrow H$.

* G, H等价就是看一看$G \leftrightarrow H$是不是恒真

指数增长: 事情很快会失控

```
In [1]: 2**64 // 10**9
Out[1]: 18446744073
```

1s计算机可以运行1e9次运算

```
In [2]: secs = 2**64 // 10**9

In [3]: hours = secs // 3600

In [4]: hours
Out[4]: 5124095

In [5]: days = hours // 24

In [6]: days
Out[6]: 213503

In [7]: years = days // 365

In [8]: years
Out[8]: 584
```

对于一个含有64个命题符号的公式, 暴力检验需要584年!

二. 命题的运算 — 命题与命题之间的关系

2. 一些恒真式

- 交换律:

$$(A \wedge B) \leftrightarrow (B \wedge A)$$

$$(A \vee B) \leftrightarrow (B \vee A)$$

- 结合律:

$$((A \wedge B) \wedge C) \leftrightarrow (A \wedge (B \wedge C))$$

$$((A \vee B) \vee C) \leftrightarrow (A \vee (B \vee C))$$

- 分配律:

$$(A \wedge (B \vee C)) \leftrightarrow ((A \wedge B) \vee (A \wedge C))$$

$$(A \vee (B \wedge C)) \leftrightarrow ((A \vee B) \wedge (A \vee C))$$

- De Morgan 律:

$$\neg(A \wedge B) \leftrightarrow (\neg A \vee \neg B)$$

$$\neg(A \vee B) \leftrightarrow (\neg A \wedge \neg B)$$

二. 命题的运算 — 命题与命题之间的关系

2. 一些恒真式

- 双重否定律:

$$\neg\neg A \leftrightarrow A$$

- 排中律:

$$A \vee (\neg A)$$

- 矛盾律:

$$\neg(A \wedge \neg A)$$

- 逆否命题:

$$(A \to B) \leftrightarrow (\neg B \to \neg A)$$

二. 命题的运算 — 命题与命题之间的关系

例子:

问题 2.2.3. 我们已经知道 Bill, Jim 和 Sam 分别来自 Boston, Chicago 和 Detroit. 以下每句话半句对, 半句错:

- Bill 来自 Boston(p_1), Jim 来自 Chicago(p_2).
- Sam 来自 Boston(p_3), Bill 来自 Chicago(p_4).
- Jim 来自 Boston(p_5), Bill 来自 Detroit(p_6).

能确定每个人究竟谁来自何处吗?

二. 命题的运算 – 命题与命题之间的关系

例子:

解答: 我们可以将上述条件用以下逻辑表达式来表示:

$$((p_1 \wedge \neg p_2) \vee (\neg p_1 \wedge p_2)) \wedge ((p_3 \wedge \neg p_4) \vee (\neg p_3 \wedge p_4)) \wedge ((p_5 \wedge \neg p_6) \vee (\neg p_5 \wedge p_6))$$

先看前两个括号 (上述式子红色的部分), 以连接两个式子中间的 $\wedge$ 展开 (下式红色符号), 我们有

$$\begin{aligned} &((p_1 \wedge \neg p_2) \vee (\neg p_1 \wedge p_2)) \wedge ((p_3 \wedge \neg p_4) \vee (\neg p_3 \wedge p_4)) \\ =&(p_1 \wedge \neg p_2 \wedge p_3 \wedge \neg p_4) \vee (p_1 \wedge \neg p_2 \wedge \neg p_3 \wedge p_4) \vee (\neg p_1 \wedge p_2) \wedge \neg p_3 \wedge p_4) \vee (\neg p_1 \wedge p_2 \wedge \neg p_3 \wedge p_4) \end{aligned}$$

根据已知条件, $p_1 \wedge p_4, p_2 \wedge p_4, p_1 \wedge p_3$ 均为假的, 所以上述式子是

$$(\neg p_1 \wedge p_2 \wedge p_3 \wedge \neg p_4)$$

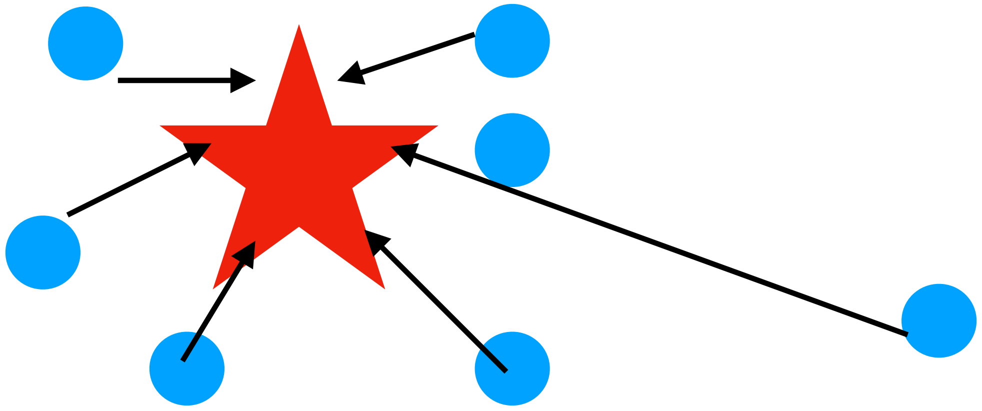
与后面的 $((p_5 \wedge \neg p_6) \vee (\neg p_5 \wedge p_6))$ 进行 $\wedge$ 操作, 也就是有

$$\begin{aligned} &(\neg p_1 \wedge p_2 \wedge p_3 \wedge \neg p_4)((p_5 \wedge \neg p_6) \vee (\neg p_5 \wedge p_6)) \\ =&(\neg p_1 \wedge p_2 \wedge p_3 \wedge \neg p_4 \wedge p_5 \wedge \cancel{p_6}) \vee (\neg p_1 \wedge p_2 \wedge p_3 \wedge \neg p_4 \wedge \neg p_5 \wedge p_6) \\ =&(\neg p_1 \wedge p_2 \wedge p_3 \wedge \neg p_4 \wedge \neg p_5 \wedge p_6) \end{aligned}$$

所以我们知道: p_2, p_3, p_6 是对的. #

二. 命题的运算 – 命题与命题之间的关

3. 化简的目标–范式



二. 命题的运算 — 命题与命题之间的关系

3. 化简的目标—范式

(1) 合取范式, 析取范式和得到的方法

定义 2.2.12 (合取范式 (Conjunctive Normal Form)). 我们称公式 α 是**合取范式**, 如果它形如

$$\alpha = \beta_1 \wedge \beta_2 \wedge \cdots \wedge \beta_k,$$

其中, 每个 β_i 都形如

$$\beta_i = \beta_{i1} \vee \beta_{i2} \vee \cdots \vee \beta_{in},$$

并且 β_{ij} 或是一个命题符号, 或者命题符号的否定.

定义 2.2.13 (析取范式 (Disjunctive Normal Form)). 我们称公式 α 是**析取范式**, 如果它形如

$$\alpha = \beta_1 \vee \beta_2 \vee \cdots \vee \beta_k,$$

其中, 每个 β_i 都形如

$$\beta_i = \beta_{i1} \wedge \beta_{i2} \wedge \cdots \wedge \beta_{in},$$

并且 β_{ij} 或是一个命题符号, 或者命题符号的否定.

(1) 用 $\neg, \wedge, \vee$ 代替 $\rightarrow, \leftrightarrow$;
(2) 用双重否定律, 消去律去掉多余的否定连接词, 运用 De Morgan 律将否定连接词内移.
(3) 利用分配率, 结合律, 幂等律整理得到.
实际上, 在上面的三个人从哪里来的例子中, 我们就用到了这样的想法.

二. 命题的运算 – 命题与命题之间的关

为什么范式如此重要?

逻辑公式中的变量（输入）可以以复杂的方式混在一起. 如果公式采用 CNF 或 DNF，变量就更加分离，从而更容易看出表达式何时成立. 比如, 要检查 CNF 是否成立，只需逐个检查每个子句有一个是假的整个都是假的. DNF 也类似: 逐个检查子句，并在找到一个为真的子句时停止, 整个句子都是真的.

很多时候真值表会带来很多的麻烦: 意味着穷举和非常麻烦的事情. 有了 CNF 与 DNF 以后, 我们不必从真值表开始枚举, 可以通过操作表达式来形式地构造标准形式. 在某些情况下可能可以更方便一些.

二. 命题的运算 — 命题与命题之间的关系

3. 化简的目标—范式

(2) 简单的性质

P Q	$\neg P \wedge \neg Q$	$\neg P \wedge Q$	$P \wedge \neg Q$	$P \wedge Q$
0 0	1	0	0	0
0 1	0	1	0	0
1 0	0	0	1	0
1 1	0	0	0	1

表 2.8: 合取范式枚举命题符号是否取反以及指派为真假的情形 (极小项)

P Q	$\neg P \wedge \neg Q$	$\neg P \wedge Q$	$P \wedge \neg Q$	$P \wedge Q$
0 0	0	1	1	1
0 1	1	0	1	1
1 0	1	1	0	1
1 1	1	1	1	0

表 2.9: 合取范式枚举命题符号是否取反以及指派为真假的情形 (极大项)

个出现过并且只出现了一次, 那么我们说这个合取范式为极小项 (析取范式为极大项). 比如我们有两个变量, 那么就可以有如下的四个逻辑表达式:

$$P \wedge Q \qquad \neg P \wedge Q \qquad P \wedge \neg Q \qquad \neg P \wedge \neg Q$$

三. 简化的命题推演系统

1. 推理: 前提与结论之间类似于公理一样的规则

$$\frac{\alpha \qquad \beta \qquad \ldots \quad (\text{前提})}{\gamma \quad (\text{结论})} \quad (\text{规则名称})$$

2. 推理的规则

(1) 引入假设(CP):

问: xxx是否存在? 答: 假设它存在(引入假设), 然后一通求, 如果存在了(假设的释放), 就写上答案

$$\frac{}{[x : P]} \quad (\text{assum})$$

$$\frac{P \qquad Q}{P \wedge Q} \quad (\wedge\text{-intro})$$

$$\frac{P \wedge Q}{P} \quad (\wedge\text{-elim-left})$$

$$\frac{P \wedge Q}{Q} \quad (\wedge\text{-elim-right})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

∧的推理规则

$$\frac{P \qquad Q}{P \wedge Q} \quad (\wedge\text{-intro})$$

$$\frac{P \wedge Q}{P} \quad (\wedge\text{-elim-left})$$

$$\frac{P \wedge Q}{Q} \quad (\wedge\text{-elim-right})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

$\wedge$的推理规则

$$\{p \wedge q, r\} \vdash q \wedge r$$

$p \wedge q$	前提	(1)
r	前提	(2)
q	$\wedge$-elim-right (1)	(3)
$q \wedge r$	$\wedge$-intro $(3), (2)$	(4)

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

¬¬的推理规则

$$\frac{\alpha}{\neg\neg\alpha} \quad (\neg\neg\text{-intro})$$

$$\frac{\neg\neg\alpha}{\alpha} \quad (\neg\neg\text{-elem})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

→的推理规则

$$\frac{\alpha \to \beta \qquad \alpha}{\beta} \quad (\to\text{-elim (modus ponens)})$$

$$\frac{\alpha \to \beta \qquad \neg\beta}{\neg\alpha} \quad (\text{modus tollens})$$

$$\frac{\begin{array}{c}[x : \alpha] \\ \vdots \\ \beta\end{array}}{\alpha \to \beta} \quad (\to\text{-intro}/x)$$

Assumption x is discharged

问: xxx是否存在? 答: 假设它存在(引入假设), 然后一通求, 如果存在了(假设的释放), 就写上答案

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

∨的推理规则

$$\frac{\alpha}{\alpha \vee \beta} \quad (\vee\text{-intro-left})$$

$$\frac{\beta}{\alpha \vee \beta} \quad (\vee\text{-intro-right})$$

$$\frac{\alpha \vee \beta \qquad \alpha \to \gamma \qquad \beta \to \gamma}{\gamma} \quad (\vee\text{-elim; (分情况分析)})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

False(假), ⊥ 的推理规则

$$\frac{\alpha \qquad \neg\alpha}{\perp} \quad (\perp\text{-intro})$$

$$\frac{\perp}{\alpha} \quad (\perp\text{-elim; EFQ, ex falso quodlibet (Principle of Explosion)})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

¬的推理规则

$$\frac{\alpha \to \bot}{\neg\alpha} \quad (\neg\text{-intro})$$

$$\frac{\neg\alpha}{\alpha \to \bot} \quad (\neg\text{-elim})$$

三. 简化的命题推演系统

2. 推理规则

(2) 进行变形(I)

例子: 证明排中律 $\vdash p \vee \neg p$ (排中律; Law of the Excluded Middle (LEM))

$\neg(p \vee \neg p)$	(引入假设)	(1)
$[p]$	(引入假设)	(2)
$p \vee \neg p$	($\vee$-intro-left)	(3)
$\perp$	($\perp$-intro)	(4)
$\neg p$	($\neg$-intro)	(5)
$p \vee \neg p$	($\vee$-intro-right)	(6)
$\perp$	($\perp$-intro)	(7)
$\neg\neg(p \vee \neg p)$	($\neg$-intro)	(8)
$p \vee \neg p$	($\neg\neg$-elim)	(9)

比较具有技巧!

三. 简化的命题推演系统

把这些推理规则一并忘却吧!

因为你本来就会了

三. 简化的命题推演系统

2. 推理规则

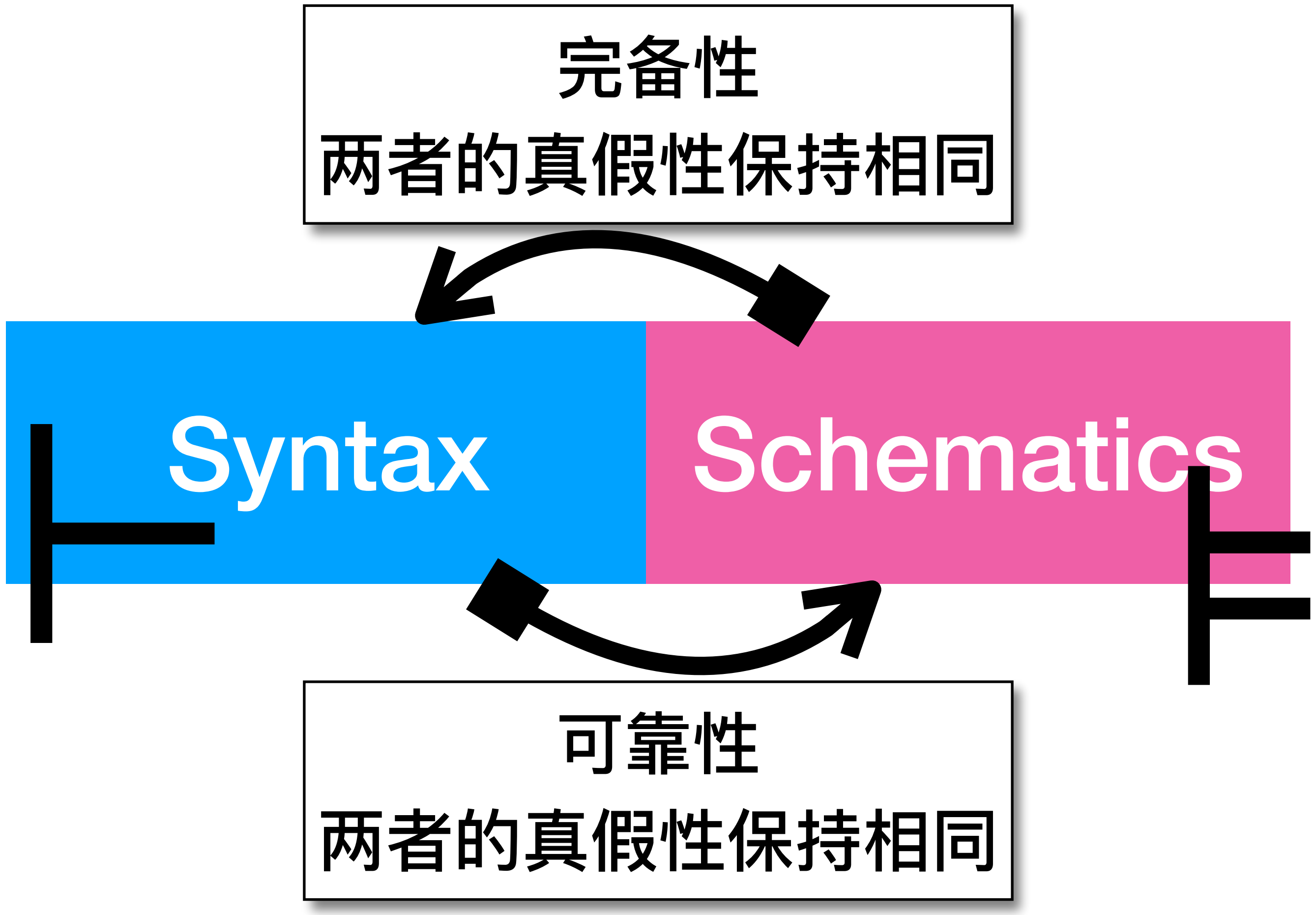
某公司要从赵、钱、孙、李、吴 5 名员工中选派某些人出国考察。由于某些不可描述的原因，选派要求如下:

(1) 若赵去，钱也去;
(2) 李、吴两人中必有一人去;
(3) 钱、孙两人中去且仅去一人;
(4) 孙、李两人同去或同不去;
(5) 若吴去，则赵、钱也去;
(6) 只有孙去，赵才会去。

(1) $Z \to Q$;
(2) $L \vee W$;
(3) $(Q \wedge \neg S) \vee (S \wedge \neg Q)$;
(4) $(S \wedge L) \vee (\neg S \wedge \neg L)$;
(5) $W \to Z \wedge Q$;
(6) $Z \to S$。

四. 命题推演系统的完备性

如果 $\Sigma \vdash \alpha$, 则 $\Sigma \models \alpha$。

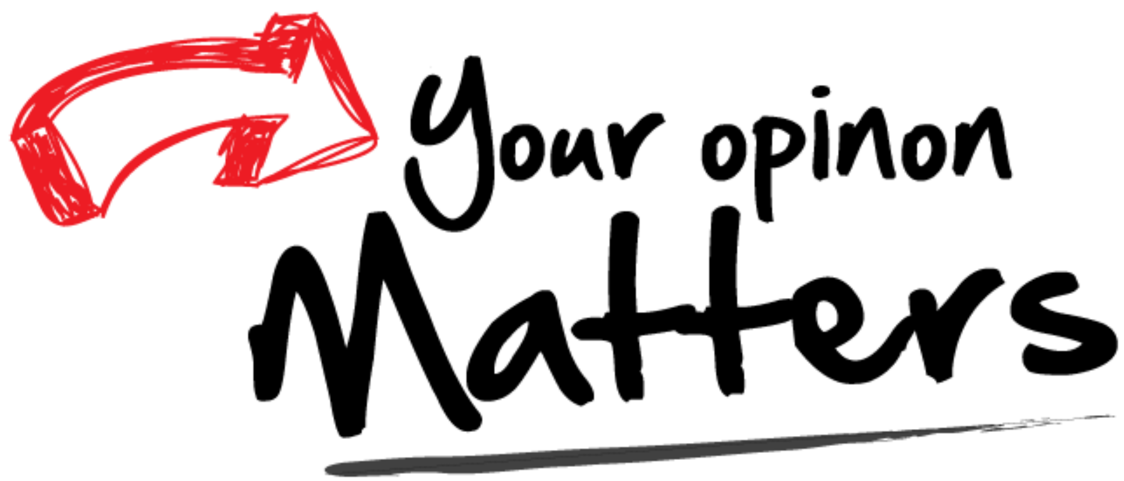


如果 $\Sigma \models \alpha$, 则 $\Sigma \vdash \alpha$。

参考的课件

1. 魏恒峰《离散数学2020》命题逻辑

Thank You!



QQ: 2095728218
Email: micoael@qq.com
(学校) gwzhang@cug.edu.cn