

一阶逻辑

命题不仅可能对单个具体事物的性质或几个具体事物之间的关系做出判断,也可能对一类事物的性质或几类事物之间的关系做出判断。例如,命题“张三是计算机专业学生”是对单个人“张三”的性质的判断,而命题“计算机专业学生要学习离散数学”是对“计算机专业学生”这一类人的性质的判断。

命题逻辑只研究复合命题与其子命题之间的真值关系,即逻辑运算符的性质,不能分析一类事物与单个具体事物之间的联系。例如,从“计算机专业学生要学习离散数学”和“张三是计算机专业学生”这两个命题,人们很容易得到“张三要学习离散数学”这个结论,但在命题逻辑中,这三个命题都是独立的原子命题,只能分别使用命题变量 p, q, r 表示,而 $(p \wedge q) \rightarrow r$ 不是命题逻辑的永真式,也即由于无法表达像“张三”这个具体个体与一类个体“计算机专业学生”之间的联系,在命题逻辑中无法分析一些人们广为接受的推理的有效性。

一阶逻辑 (first-order logic) 对原子命题的结构进行细分,以便表达一类事物与单个具体事物之间的联系。本章首先介绍一阶逻辑的一些基本概念,然后定义一阶逻辑公式及其真值的确定,并讨论一阶逻辑的等值演算和推理理论,最后介绍一阶逻辑公式的一些应用。

3.1 一阶逻辑的基本概念

命题是具有真值的陈述句,是对事物是否具有某种性质或事物之间是否具有某种关系的判断。一阶逻辑对原子命题的结构进行细分,将原子命题要判断的事物称为**个体** (individual),而原子命题中给出的性质或关系称为**谓词** (predicate)。例如,命题“张三是计算机专业学生”中的“张三”是个体,而“……是计算机专业学生”是谓词。

谓词总是要作用于个体,例如,“……是计算机专业学生”中的省略号就是代表谓词要作用的个体。我们使用“ x 是计算机专业学生”这种更数学化的句式给出谓词,其中 x 代表某个不确定的个体,在一阶逻辑中称为**个体变量** (individual variable)。我们使用大写字母 P, Q, R 等作用在一个或多个个体变量的形式符号化谓词,例如用 $P(x)$ 表示“ x 是



PredLogic-Basic(1)



PredLogic-Basic(2)



PredLogic-Basic(3)



PredLogic-Basic(4)



PredLogic-Basic(5)

计算机专业学生”，用 $Q(x, y)$ 表示“ x 和 y 是同学”等。这里 $P(x)$ 是一元谓词，只作用于一个个体，而 $Q(x, y)$ 是二元谓词，作用于两个个体，表示这两个个体之间的关系。能作用的个体数目称为谓词的**元数** (arity)。一元谓词表达个体性质，而二元或更多元谓词表达二个或多个个体之间的关系。

我们使用 a, b, c 等小写字母表示具体的个体，称为**个体常量** (individual constant)。当原子命题是对单个具体事物的性质或几个具体事物间的关系进行判断时，用个体常量表示其中的具体事物，用谓词表示其中的性质和关系，从而将这类命题进行符号化。例如，对原子命题“张三是计算机专业学生”，使用个体常量 a 表示“张三”，谓词 $P(x)$ 表示“ x 是计算机专业学生”，从而这个命题符号化为 $P(a)$ 。对原子命题“张三和李四是同学”，分别使用个体常量 a 和 b 表示“张三”和“李四”，谓词 $Q(x, y)$ 表示“ x 和 y 是同学”，从而这个命题符号化为 $Q(a, b)$ 。

当原子命题是对一类事物的性质或几类事物间的关系进行判断时，人们通常是对一类事物的所有个体或某些个体的性质或参与的关系进行判断。例如命题“计算机专业学生要学习离散数学”是“所有计算机专业学生要学习离散数学”的省略形式，是对“计算机专业学生”这一类人员中所有个体进行判断，而命题“有的计算机专业学生选修数理逻辑”是对“计算机专业学生”这一类人员中某些个体进行判断。

人们将这种命题称为**量化命题** (quantification proposition)，命题中像“所有”“有的”这种修饰事物类的词语称为**量词** (quantifier)。表达事物类的所有个体都具有某种性质或参与某种关系的量词称为**全称量词** (universal quantifier)，在符号化命题时用 \forall 表示；而表达事物类的某些个体具有某种性质或参与某种关系的量词称为**存在量词** (existential quantifier)，用 \exists 表示。用到全称量词的量化命题称为**全称命题**，而用到存在量词的量化命题称为**存在命题**。

量化命题的符号化必须用到个体变量，用于表示某一类事物中某个不确定的个体。个体变量所表示的不确定个体所属的事物类称为该个体变量的**论域** (domain)。例如，对于命题“计算机专业学生要学习离散数学”，可用谓词 $H(x)$ 表示“ x 要学习离散数学”，使用个体变量 x 表示“计算机专业学生”这一类人员中某个不确定的个体，即 x 的论域是所有“计算机专业学生”构成的集合，整个命题符号化为 $\forall xH(x)$ ，读作“对所有 x ， $H(x)$ ”，或更准确地读作“对所有计算机专业学生 x ， $H(x)$ ”，因为 x 的论域是“计算机专业学生”。

量化命题的符号化需要先确定其中出现的个体变量的论域，不同的论域可能有不同的符号化形式。例如，对于命题“计算机专业学生要学习离散数学”，个体变量 x 的论域也可以是任意包含所有“计算机专业学生”的集合，例如，所有“学生”的集合。这时需要为命题中的事物类，即“计算机专业学生”也引入谓词，这种谓词称为**特征谓词** (characteristic predicate)。仍用 $P(x)$ 表示“ x 是计算机专业学生”，当 x 的论域是所有“学生”的集合时，整个命题应细化理解为“对所有学生 x ，若 x 是计算机专业学生，则 x 要学习离散数学”，从而符号化为 $\forall x(P(x) \rightarrow H(x))$ 。

这里给出了一阶逻辑的基本概念,包括个体、谓词、个体常量、个体变量、全称量词、存在量词、论域和特征谓词。后面介绍一阶逻辑应用时,将在定义一阶逻辑公式语法和语义的基础上更深入地讨论自然语言命题在一阶逻辑中的符号化。

最后要指出的是,一阶逻辑是一阶谓词逻辑 (first-order predicate logic) 的简称。一阶谓词是指只作用于个体变量的谓词。一阶逻辑不引入可以作用于谓词、函数或者公式的高阶谓词,也不使用取值范围为某类谓词、函数或者公式的谓词变量、函数变量或公式变量等,一阶逻辑的量词也只能作用于个体变量,而不能作用于谓词、函数或公式。例如,“说 x 和 y 是相同的事物,如果对任意性质 P , x 具有性质 P 当且仅当 y 具有性质 P ”就是典型的非一阶逻辑的命题,这里“对任意性质 P ”就是量词作用于谓词,而非作用于个体变量。

3.2 一阶逻辑公式的语法

一阶逻辑公式的语法比命题逻辑公式的语法复杂,首先要将一阶逻辑公式中可以出现的符号分为逻辑符号和非逻辑符号,非逻辑符号由应用一阶逻辑公式的应用领域决定,而逻辑符号与应用领域无关,一阶逻辑本质上是研究逻辑符号,特别是量词的性质。给定非逻辑符号集,先定义一阶逻辑公式中的项,然后再定义一阶逻辑公式。项没有真值,是对个体的符号化,谓词作用于项才是一阶逻辑的原子公式,原子公式通过逻辑运算符和量词按照一定的规则构成一阶逻辑公式。

本节先讨论一阶逻辑公式的符号集,然后定义一阶逻辑公式中的项和一阶逻辑公式,最后讨论一些与一阶逻辑公式语法相关的概念,主要包括量词的辖域、个体变量的约束出现和自由出现、一阶逻辑公式的约束变量和自由变量等。

3.2.1 一阶逻辑公式的符号

命题逻辑公式中允许出现的符号包括命题变量符号、逻辑运算符以及圆括号。一阶逻辑公式中允许出现的符号更为复杂,分为逻辑符号和非逻辑符号。

一阶逻辑公式中出现的逻辑符号包括:① 个体变量符号,使用小写字母 x, y, z 等表示个体变量,记所有个体变量符号构成的集合为 V ;② 逻辑运算符,包括否定 \neg 、逻辑与 \wedge 、逻辑或 \vee 、逻辑蕴涵 \rightarrow 和逻辑双蕴涵 \leftrightarrow ;③ 量词符号,包括全称量词 \forall 和存在量词 \exists ;④ 辅助符号,包括圆括号 (和),以及逗号。

一阶逻辑公式中出现的非逻辑符号包括:① 个体常量符号,使用小写字母 a, b, c 等表示个体常量;② 函数符号,使用小写字母 f, g, h 等表示函数,每个函数都有一个元数信息,表明该函数是几元函数;③ 谓词符号,使用大写字母 F, G, H 等表示谓词,每个谓词都有一个元数信息,表明该谓词是几元谓词。

将一阶逻辑公式中可以出现的符号分为逻辑符号和非逻辑符号是因为非逻辑符号与一阶逻辑的具体应用领域相关,例如应用一阶逻辑公式研究整数的性质,可以使用 $0, 1$ 等作为个体常量符号, $+, -, \times, \div$ 等作为函数符号, $\geq, \leq, =$ 等作为谓词符号,而研

究集合的性质,可以使用空集作为常量符号,集合并、交等运算作为函数符号,而子集关系、集合相等作为谓词符号。非逻辑符号由应用领域抽象而得到,在确定一阶逻辑公式的真值时也必须基于应用领域先给出非逻辑符号的解释。因此,区分一阶逻辑公式中的逻辑符号和非逻辑符号有助于读者更好地应用一阶逻辑。

一阶逻辑本身只研究与逻辑符号,特别是与量词符号有关的性质,因此本章总是假定有一个非逻辑符号集,记为 \mathcal{L} ,其中的个体常量使用小写字母 a, b, c 等表示,函数符号使用 f, g, h 等表示,谓词符号使用 F, G, H 等表示,读者从上下文很容易区别这些符号。一阶逻辑公式在给定的非逻辑符号集 \mathcal{L} ,以及个体变量集 V 的基础上定义。非逻辑符号集 \mathcal{L} 可以没有个体常量符号和函数符号,但至少要有个谓词符号。

3.2.2 一阶逻辑公式的定义

定义 3.1 给定非逻辑符号集 \mathcal{L} 和个体变量集 V ,一阶逻辑公式中的项(term)归纳定义如下。

- (1) 归纳基: \mathcal{L} 的任意个体常量 c 都是项; V 的任意个体变量 x 也是项。
- (2) 归纳步: 对 \mathcal{L} 的任意 n 元函数 f ,如果 t_1, t_2, \dots, t_n 是项,则 $f(t_1, t_2, \dots, t_n)$ 也是项。

直观地说,一阶逻辑公式中的项就是通过函数构造的复杂个体。例如,设 $0, 1$ 是非逻辑符号集 \mathcal{L} 中的个体常量, $+$ 是二元函数, x, y 是个体变量,则 $+(x, y), ++(x, 0), 1$ 等是项,当函数是运算符时我们用更熟悉的中缀方式表示,即写成 $(x + y), ((x + 0) + 1)$ 这种形式的表达式。由于在应用领域中广泛存在函数与运算,因此一阶逻辑公式引入函数符号,用基于个体常量和个体变量构造项作为对应用领域中函数或运算表达式的符号化。

定义 3.2 给定非逻辑符号集 \mathcal{L} 和个体变量集 V ,一阶逻辑公式(first-order logic formula)归纳定义如下。

- (1) 归纳基: 对 \mathcal{L} 的任意 n 元谓词 F ,如果 t_1, t_2, \dots, t_n 是项,则 $F(t_1, t_2, \dots, t_n)$ 是公式,并称为一阶逻辑的原子公式。
- (2) 归纳步: ①如果 A 是一阶逻辑公式,则 $(\neg A)$ 是一阶逻辑公式; ②如果 A, B 是一阶逻辑公式,则 $(A \wedge B), (A \vee B), (A \rightarrow B)$ 和 $(A \leftrightarrow B)$ 是一阶逻辑公式; ③如果 A 是一阶逻辑公式, x 是个体变量,则 $\forall xA$ 和 $\exists xA$ 是一阶逻辑公式。

将 $(\neg A), (A \wedge B), (A \vee B), (A \rightarrow B)$ 和 $(A \leftrightarrow B)$ 分别称为否定式、合取式、析取式、蕴涵式和双蕴涵式,而将 $\forall xA$ 称为全称量词公式, $\exists xA$ 称为存在量词公式。

后面有时将一阶逻辑公式简称为一阶公式,或者直接称公式,因为在没有特别说明的情况下,本章说的公式都是指一阶逻辑公式。注意,上述定义要求非逻辑符号集 \mathcal{L} 至少有一个谓词符号(否则不能构造出原子公式,更不能构造出其他一阶公式)。

例子 3.1 设非逻辑符号集 \mathcal{L} 有一元谓词 F 和二元谓词符号 G, H ,个体变量集 V 有个体变量 x, y, z ,则下面是一阶逻辑公式的例子。

$$(1) \forall x(F(x) \rightarrow \exists yG(x, y))$$

$$(2) \forall x F(x) \wedge \exists y (F(y) \wedge \forall z H(y, z))$$

$$(3) (\forall x (F(x) \wedge H(x, y)) \rightarrow \exists x (G(x, y) \rightarrow (\neg \exists y H(x, y))))$$

$$(4) \exists x ((G(x, y) \vee (\neg (\exists y F(y) \wedge H(x, y)))) \wedge \forall z \exists y (F(z) \leftrightarrow H(z, y)))$$

与命题逻辑公式类似，对于复杂的一阶逻辑公式，根据定义 3.2 画出公式的抽象语法树更容易看清楚它的语法结构。图 3.1 是上面 (3) 给出的一阶公式的抽象语法树，可以看到它是一个逻辑蕴涵式，前件是由全称量词 $\forall x$ 构造的子公式 $\forall x A$ ，其中 A 是一个合取式，而后件是由存在量词 $\exists x$ 构造的子公式 $\exists x B$ ，其中 B 是一个逻辑蕴涵式。

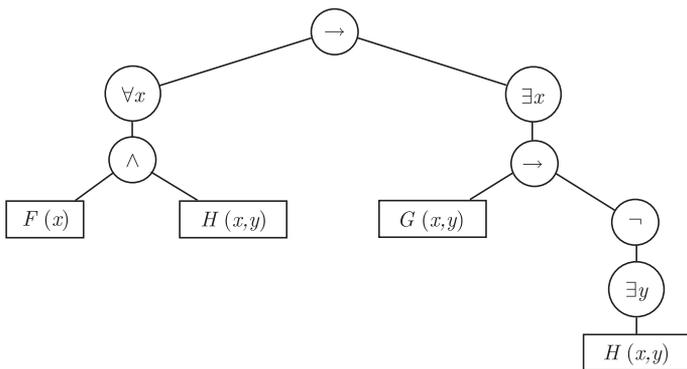


图 3.1 公式 $(\forall x(F(x) \wedge H(x, y)) \rightarrow \exists x(G(x, y) \rightarrow (\neg \exists y H(x, y))))$ 的抽象语法树

注意，在给出一阶公式的抽象语法时，叶子节点是原子公式，不再给出原子公式的语法构造（即不再给出原子公式是怎样由谓词作用于项而得到）。我们将量词及其约束的个体变量一起作为一阶公式的一种构造方式，标记抽象语法树的内部节点，例如上面 (3) 给出的一阶公式的逻辑蕴涵前件是由 $\forall x$ 构造的子公式，而后件是由 $\exists x$ 构造的子公式。

图 3.2 是上面 (4) 给出的一阶公式的抽象语法树，它最后由存在量词 $\exists x$ 构造，具有 $\exists x A$ 的形式，其中 A 是一个合取式 $(B \wedge C)$ ，其中 B 是一个析取式，而 C 是一个全称量词公式 $\forall z D$ ，这里 D 又是一个存在量词公式等。

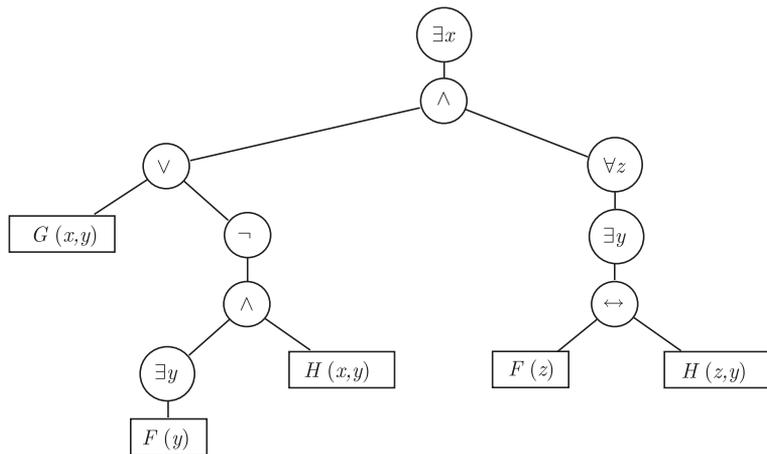


图 3.2 公式 $\exists x((G(x, y) \vee (\neg (\exists y F(y) \wedge H(x, y)))) \wedge \forall z \exists y (F(z) \leftrightarrow H(z, y)))$ 的抽象语法树

我们选了例子 3.1 的 (3) 和 (4) 给出的两个比较复杂的公式说明一阶公式的语法结构和抽象语法树, (1) 和 (2) 给出的公式语法结构比较简单, 读者可自己练习画出它们的抽象语法树。在实际应用中, 下面例子的一阶公式可能更为常见。

例子 3.2 设非逻辑符号集 \mathcal{L} 有个体常量符号 $0, 1$, 有二元函数符号 $+$ 和 $*$, 以及二元谓词符号 $\leq, =$, 个体变量集 V 有个体变量 x, y , 则下面是一阶逻辑公式的例子。

$$(1) \forall x((0 \leq x) \rightarrow \exists y(x = y * y)) \quad (2) \forall x \exists y(x \leq y)$$

$$(3) (\exists x \forall y(x + y = y) \wedge \forall x \exists y(x * y = 1)) \quad (4) \exists x \forall y(x \leq y)$$

这里将项和原子公式都写成了更为熟悉的中缀表示形式, 也即原子公式 $0 \leq x$ 严格地说应该是 $\leq(0, x)$, 而项 $y * y$ 严格地说应该是 $*(y, y)$, 等等。

基于一阶逻辑公式的抽象语法树, 我们可清楚地看到每个公式包含哪些子公式, 因为这个公式的抽象语法树的每棵子树都对应它的一个子公式。具体来说, 可如下定义一阶逻辑公式的子公式。

定义 3.3 一阶逻辑公式 A 的子公式 (sub-formula) 按如下方式归纳定义。

(1) 若公式 A 是原子公式 $F(t_1, t_2, \dots, t_n)$, 则它的子公式只有它自己, 即只有 $F(t_1, t_2, \dots, t_n)$ 。

(2) 若公式 A 是公式 $(\neg B)$, 则它的子公式包含 A , 以及 B 的所有子公式。

(3) 若公式 A 是公式 $(B \oplus C)$, 这里 \oplus 代表 $\wedge, \vee, \rightarrow$ 或 \leftrightarrow , 则它的子公式包括 A , 以及 B 和 C 的所有子公式。

(4) 若公式 A 是公式 $\forall x B$ 或 $\exists x B$, 则它的子公式包括 A , 以及 B 的所有子公式。

例子 3.3 (1) 公式 $\forall x \exists y(x \leq y)$ 的子公式除它自己外还有: $\exists y(x \leq y)$ 和 $x \leq y$ 。

(2) 公式 $(\exists x \forall y(x + y = y) \wedge \forall x \exists y(x * y = 1))$ 的子公式除它自己外还有 $\exists x \forall y(x + y = y), \forall x \exists y(x * y = 1), \forall y(x + y = y), \exists y(x * y = 1), x + y = y, x * y = 1$ 。

通过规定优先级和组合性可省略一阶公式中的一些圆括号。通常规定量词的优先级最高, 然后是否定、合取、析取、蕴涵和双蕴涵运算符, 因此公式 $\forall x F(x) \wedge G(x, y)$ 不是 $\forall x(F(x) \wedge G(x, y))$ 。蕴涵从右至左结合, 而合取、析取和双蕴涵都是从左至右结合。注意, 使用量词构造公式时, 我们采用简单的形式 $\forall x A$ 和 $\exists x A$, 而不是 $(\forall x A)$ 和 $(\exists x A)$, 也不是 $\forall x(A)$ 和 $\exists x(A)$, 这意味着: ① 当最后一步是使用量词构造公式时, 最外层总是不再加圆括号; ② 对于公式 $\forall x A$ 或 $\exists x A$, 如果 A 是原子公式或量词公式时, 量词与 A 之间也不再使用圆括号。由于一阶公式的语法结构可能比较复杂, 因此需要仔细使用圆括号以清楚展示公式结构, 并保证左右圆括号配对使用。

3.2.3 自由变量和约束变量

个体变量在一阶逻辑公式中起着非常重要的作用, 并且同样的个体变量符号在一个一阶逻辑公式中可能以不同的身份出现, 从而起不同的作用, 所以有必要对个体变量在一阶逻辑公式中的出现方式做更深入的讨论。

定义 3.4 设公式 A 是 $\forall x B$ 或 $\exists x B$, 称用于构造公式 A 的量词 $\forall x$ 或 $\exists x$ 中的个体变量 x 为这个量词 (即 $\forall x$ 或 $\exists x$) 的指示变量, 称子公式 B 为这个量词 (即 $\forall x$ 或

$\exists x$) 的辖域 (scope), 或作用域。

若个体变量 x 在公式 A 的一处出现是在 A 的一个以 x 为指示变量的量词子公式 $\forall xB$ 或 $\exists xB$ 的辖域 B 中, 则称 x 在 A 的这处出现是约束出现。若个体变量 x 在公式 A 的一处出现不在 A 的任意以 x 为指示变量的量词子公式 $\forall xB$ 或 $\exists xB$ 的辖域 B 中, 则称 x 在 A 的这处出现是自由出现。

设个体变量 x 在公式 A 中出现, 如果 x 在公式 A 中有一处出现是自由出现, 则称 x 是公式 A 的自由变量 (free variable); 如果 x 在公式 A 的所有出现都是约束出现, 则称 x 是公式 A 的约束变量 (bound variable)。没有自由变量的公式称为闭公式 (closed formula), 或句子 (sentence)。

从抽象语法树的角度看, 一个量词的辖域就是以它的儿子节点为根的子树对应的公式。对一个个体变量在公式的一处出现, 若从抽象语法树的根节点到该处的唯一路径上存在以该变量为指示变量的量词节点, 则是约束出现, 否则是自由出现。一个个体变量可以既在一个公式中自由出现, 又在这个公式中约束出现, 但它要么是这个公式的自由变量, 要么是这个公式的约束变量, 二者必居其一。

问题 3.4 给出下面公式中每个量词的辖域, 确定个体变量的每处出现是约束出现还是自由出现, 并说明出现的每个个体变量是公式的自由变量还是约束变量。

$$(1) \forall x(F(x) \rightarrow \exists yG(x, y))$$

$$(2) \forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z))$$

$$(3) (\forall x(F(x) \wedge H(x, y)) \rightarrow \exists x(G(x, y) \rightarrow (\neg \exists yH(x, y))))$$

$$(4) \exists x((G(x, y) \vee (\neg(\exists yF(y) \wedge H(x, y)))) \wedge \forall z\exists y(F(z) \leftrightarrow H(z, y)))$$

【解答】我们在这里只给出 (2) 和 (4) 的解答, (1) 和 (3) 留作读者自行练习, 对于 (3) 给出的公式, 读者可参考图 3.1 给出的抽象语法树。

(2) 公式 $\forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z))$ 的量词 $\forall x$ 的辖域是 $F(x)$, 量词 $\exists y$ 的辖域是 $(F(y) \wedge \forall zH(y, z))$, 而量词 $\forall z$ 的辖域是 $H(y, z)$ 。每个个体变量的出现方式如下:

指示变量	约束出现	指示变量	约束出现	指示变量	约束出现	约束出现
↓	↓	↓	↓	↓	↓	↓
\forall	x	F	$($	x	\wedge	\exists
	$)$	\wedge	\forall	z	H	$($
	$)$	$)$	\wedge	z	$($	$)$

所有个体变量的出现都是约束出现, 因此个体变量 x 、 y 和 z 都是这个公式的约束变量, 即这个公式没有自由变量, 是闭公式。

(4) 对于公式 $\exists x((G(x, y) \vee (\neg(\exists yF(y) \wedge H(x, y)))) \wedge \forall z\exists y(F(z) \leftrightarrow H(z, y)))$ 中的量词辖域情况根据图 3.2 给出的抽象语法树容易确定以下判断。

① 量词 $\exists x$ 的辖域是 $((G(x, y) \vee (\neg(\exists yF(y) \wedge H(x, y)))) \wedge \forall z\exists y(F(z) \leftrightarrow H(z, y)))$ 。

② 前一个量词 $\exists y$ 的辖域是 $F(y)$, 而后一个量词 $\exists y$ 的辖域是 $(F(z) \leftrightarrow H(z, y))$ 。

③ 量词 $\forall z$ 的辖域是 $\exists y(F(z) \leftrightarrow H(z, y))$ 。

每个个体变量的出现方式如下:

$$\begin{array}{ccccccc}
 \text{指示变量} & \text{约束出现} & \text{自由出现} & \text{指示变量} & \text{约束出现} & \text{约束出现} & \\
 \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 \exists & x((G(x, & y) \vee (\neg(\exists yF(y) \wedge H(x, & \\
 & \text{自由出现} & \text{指示变量} & \text{指示变量} & \text{约束出现} & \text{约束出现} & \text{约束出现} \\
 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 & y))) \wedge \forall z \exists & y(F(z) \leftrightarrow H(z, & y)))
 \end{array}$$

可以看到, 个体变量 x 和 z 在这个公式中的出现都是约束出现, 因此是这个公式的约束变量, 但个体变量 y 在这个公式中既有约束出现又有自由出现, 因此是这个公式的自由变量。

【讨论】(1) 在一阶逻辑中, 量词只能作用于个体变量, 因此在说明量词的辖域时, 可通过量词及其指示变量一起区分不同的量词。当量词与指示变量都相同时, 例如上面(4)给出的公式有两个 $\exists y$, 可进一步根据量词的位置进行区分。

(2) 量词的辖域与计算机程序中变量的作用域类似。一个一阶公式中的量词子公式相当于一个子模块, 量词的指示变量相当于声明一个局部变量, 它只是在其辖域内起作用, 辖域之外同名的个体变量与辖域内的个体变量本质上是两个不同的个体变量。

例如, 公式 $\exists x((G(x, y) \vee (\neg(\exists yF(y) \wedge H(x, y)))) \wedge \forall z \exists y(F(z) \leftrightarrow H(z, y)))$ 中前一个 $\exists yF(y)$ 的个体变量 y 与后面的 $\exists y(F(z) \leftrightarrow H(z, y))$ 中的 y 是本质上不同的个体变量, 这类似于计算机程序中有两个子模块恰巧有相同名字的局部变量, 但实际上它们是不同的程序变量。

$\exists yF(y)$ 中的 y 与合取运算后面的 $H(x, y)$ 中的 y 也是不同的个体变量, 这两者类似计算机程序中局部变量和全局变量的关系。一个一阶公式中自由出现的个体变量在整个公式中起作用, 相当于计算机程序的全局变量。在这个公式中, $G(x, y)$ 和 $H(x, y)$ 中的 y 都是自由出现, 它们是同一个 y 。

可看到, 当个体变量既在公式中自由出现又在公式中约束出现时容易带来混淆。实际上, 人们在使用一阶逻辑公式符号化自然语言命题时, 个体变量的选择有一定的随意性。例如, 对命题“所有计算机专业学生都要学离散数学”, 用谓词 $P(x)$ 表示“ x 是计算机专业学生”, $H(x)$ 表示“ x 要学离散数学”, 可将命题符号化为 $\forall x(P(x) \rightarrow H(x))$, 也可符号化为 $\forall y(P(y) \rightarrow H(y))$ 。这正如编写同样的计算机程序, 不同程序员也会用不同的变量名。直观地看, 这两个一阶公式没有实质差别。因此, 在符号化时最好能小心选择个体变量符号以尽量避免混淆。

在一阶逻辑中, 可利用约束变量改名避免一个个体变量在公式中既自由出现又约束出现。约束变量改名是指将公式 $\forall xA$ 或 $\exists xA$ 的指示变量 x 及它的辖域 A 中所有自由出现的 x 都改为一个不在 A 中自由出现的个体变量 y 。例如, 公式 $\forall x(F(x) \wedge \exists xH(x, y))$ 使用约束变量改名可得到公式 $\forall z(F(z) \wedge \exists xH(x, y))$ 。这里有两点需要注意。

(1) 这个公式的量词 $\forall x$ 的辖域是 $A = (F(x) \wedge \exists xH(x, y))$, 其中 $F(x)$ 中的 x 对于 A 这个子公式是自由出现, 但 $H(x, y)$ 中的 x 即使对于 A 也是约束出现, 因此在对于量词 $\forall x$ 进行约束变量改名时, 不能改变 $H(x, y)$ 中的 x 。这个公式的量词 $\exists x$ 的辖域

嵌套在量词 $\forall x$ 的辖域中, 从一阶公式的语法上说没有问题, 但辖域嵌套更容易带来混淆, 所以后面我们将完全避免出现辖域嵌套的情况。

(2) 约束变量改名时要选择不在辖域中自由出现的个体变量, 所以这里不能选择 y 而将上述公式错误地约束变量改名为 $\forall y(F(y) \wedge \exists xH(x, y))$ 。原本公式 A 的 $F(x)$ 中的个体变量 x 与 $H(x, y)$ 中的 y 就不是同一个个体变量, 约束变量改名后成为相同的个体变量, 显然是错误的。进一步, 为避免辖域嵌套, 应该选择在辖域中不出现的个体变量。

通过约束变量改名总可以使得一个一阶逻辑公式中没有个体变量符号既自由出现又约束出现, 也可完全避免辖域嵌套, 甚至可使得每个量词都有不同的指示变量。后面给出的一阶公式会完全避免出现辖域嵌套, 也会尽量避免同一个个体变量符号在公式中既自由出现又约束出现。不过, 在容易理解的情况下允许不同的量词有相同的指示变量。

在一阶逻辑中, 将公式 A 中所有自由出现的个体变量 x 都替换为一个不在 A 中自由出现的个体变量 y 称为自由变量替换。只要替换时选择不在 A 中出现的个体变量, 通过自由变量替换也可使得在一个一阶公式中没有个体变量既自由出现又约束出现。后面主要使用约束变量改名, 因为这种操作是相对局部的操作, 更不容易出错。

3.3 一阶逻辑公式的语义

一阶逻辑公式的语义讨论如何确定一阶逻辑公式的真值。一阶逻辑公式在给定非逻辑符号集 \mathcal{L} 以及个体变量集 V 的基础上构造, 因此要确定一阶逻辑公式的真值也需要先给出非逻辑符号的解释, 以及对个体变量的指派。本节先定义解释和指派, 然后给出一阶逻辑公式的真值定义, 最后讨论一阶逻辑公式的语义分类, 即永真式、矛盾式和可满足式。

3.3.1 一阶逻辑公式的解释

一阶逻辑公式的非逻辑符号来自对应用领域事物的抽象, 个体常量是应用领域特定事物的抽象, 谓词是应用领域事物的性质或事物之间关系的抽象, 而函数符号是应用领域事物之间的运算或变换的抽象。考察一阶逻辑公式的真值首先要将其中的非逻辑符号与具体的应用领域进行对应, 这种对应称为一阶逻辑公式的解释, 或说模型。也就是说, 一阶逻辑公式是应用领域中事物性质、关系或约束的抽象, 而应用领域是一阶逻辑公式的解释或说模型。通过这种抽象与对应, 人们可更深入地理解应用领域以及不同应用领域之间的联系。

定义 3.5 设一阶逻辑公式的非逻辑符号集是 \mathcal{L} , 它的解释 (interpretation) 或说模型 (model), 记为 \mathcal{M} , 包括:

- (1) 一个非空集合 D , 称为解释 \mathcal{M} 的论域 (domain)。
- (2) 对 \mathcal{L} 的每个个体常量符号 c , 有论域 D 的一个元素与之对应, 记这个元素为

$\llbracket c \rrbracket$ 。

(3) 对 \mathcal{L} 的每个 n 元函数符号 f , 有论域 D 上的一个 n 元函数与之对应, 记这个 n 元函数为 $\llbracket f \rrbracket: D^n \rightarrow D$, 注意, 这里 D^n 是 n 个集合 D 的笛卡儿积, 即 $D^n = D \times D \times \cdots \times D$ 。

(4) 对 \mathcal{L} 的每个 n 元谓词符号 F , 有论域 D 上的一个 n 元关系与之对应, 记这个 n 元关系为 $\llbracket F \rrbracket \subseteq D^n$, 注意, 第 1 章的基础知识已经提到关系是笛卡儿积的子集。

直观地说, 论域是应用领域中所有个体 (或说事物) 构成的集合, 它不能是空集, 否则不存在与非逻辑符号集中谓词对应的关系。一阶逻辑公式的非逻辑符号可以没有个体常量符号和函数符号, 但必须至少有一个谓词符号。给定解释的论域之后, 个体常量符号解释为论域中的元素, 函数符号解释为论域上的函数, 谓词符号解释为论域上的关系。

例子 3.5 假定一阶逻辑公式非逻辑符号集包括个体常量符号 $0, 1$, 二元函数符号 $+$ 和 $*$, 以及二元谓词符号 $\leq, =$ 。这里可给出这些非逻辑符号的两个解释。

(1) 解释 \mathcal{M}_1 的论域是自然数集 \mathbb{N} , 常量符号 $0, 1$ 分别解释为自然数 $0, 1$, 二元函数符号 $+$ 和 $*$ 分别解释为 \mathbb{N} 上的加法和乘法运算, 二元谓词符号 $\leq, =$ 分别解释为 \mathbb{N} 上的小于等于和相等关系。

(2) 解释 \mathcal{M}_2 的论域是实数集 \mathbb{R} , 常量符号 $0, 1$ 分别解释为实数 $0, 1$, 二元函数符号 $+$ 和 $*$ 分别解释为 \mathbb{R} 上的加法和乘法运算, 二元谓词符号 $\leq, =$ 分别解释为 \mathbb{R} 上的小于等于和相等关系。

从例子 3.5 可以看到:

(1) 一阶逻辑公式的非逻辑符号集可以有不同的解释, 这是人们从具体应用领域中抽象出逻辑公式的意义之一。

(2) 由于一阶逻辑公式的非逻辑符号集是应用领域事物的抽象, 因此常出现模型中的符号与非逻辑符号集的符号相同的情况, 这是因为这个模型是非逻辑符号集最自然的解释, 人们为方便起见采用了相同的符号。但严格地说, 非逻辑符号集中的符号仅仅是符号, 而模型中的符号已经预先赋予了含义。例如, 作为一阶逻辑公式中的个体常量符号 0 , 它仅仅是个符号, 没有含义, 但作为以自然数集为论域的解释 \mathcal{M}_1 , 0 不仅仅是符号, 而且是人们熟知的那个自然数 0 。

在逻辑学研究中, 将所研究的逻辑公式, 如命题逻辑公式、一阶逻辑公式称为对象语言 (object language), 而研究逻辑时使用的语言, 如自然语言称为元语言 (meta-language)。人们对命题逻辑公式、一阶逻辑公式的符号集及公式语法的归纳定义给出了对象语言的严格定义, 但采用的元语言是自然语言, 即汉语, 无法严格定义。

逻辑学研究的一个特点就是元语言也不得不采用一些逻辑语言, 因此容易混淆元语言和对象语言。例如, 在等值演算中, 使用 \equiv 表示两个公式逻辑等值, 它是元语言的符号, 而双蕴涵运算符 \leftrightarrow 是对象语言的符号。在推理理论中人们使用 \implies 分隔推理的前提和结论, 也带有“蕴涵”“推出”的意味, 但它是元语言符号, 而蕴涵运算符 \rightarrow 是对象语言的符号。有些读者可能会混淆 \equiv 和 \leftrightarrow , 以及 \implies 和 \rightarrow 。同样地, 一阶逻辑公

式的非逻辑符号集中的符号是对象语言符号，而它的解释中的符号则是元语言符号。

定义 3.6 给定一阶逻辑公式的非逻辑符号集 \mathcal{L} 及它的一个解释 \mathcal{M} 。解释 \mathcal{M} 的一个个体变量指派函数 σ 是从个体变量集 V 到解释 \mathcal{M} 的论域 D 的函数，即是一个函数 $\sigma: V \rightarrow D$ 。

为确定一阶逻辑公式中量词的真值，需要一点小技巧，使得人们可以对个体变量指派函数进行变换，从而对某个特定的个体变量指派论域中特定的元素。

定义 3.7 给定一阶逻辑公式的非逻辑符号集 \mathcal{L} 及它的一个解释 \mathcal{M} ，其论域是 D 。对于 \mathcal{M} 的一个个体变量指派函数 σ ，以及一个个体变量 $x \in V$ ，和论域 D 的一个元素 $d \in D$ ，定义一个新的 \mathcal{M} 的个体变量指派函数，记为 $\sigma[x \mapsto d]$ ：对任意个体变量 $y \in V$ ，

$$\sigma[x \mapsto d](y) = \begin{cases} d & \text{若 } y = x \\ \sigma(y) & \text{若 } y \neq x \end{cases}$$

例子 3.6 设一阶逻辑公式的非逻辑符号集 \mathcal{L} 只包含一个一元谓词符号 F ，以及两个二元谓词符号 G, H ，个体变量集 $V = \{x, y, z\}$ 。

定义 \mathcal{L} 的一个解释 \mathcal{M} ：论域 $D = \{a, b, c\}$ ，谓词符号 F 的解释是 D 上的一元关系，即 D 的子集 $\llbracket F \rrbracket = \{a, b\} \subseteq D$ ，谓词符号 G 的解释是 D 上的二元关系 $\llbracket G \rrbracket = \{\langle a, b \rangle, \langle b, c \rangle\}$ ，谓词符号 H 的解释是 D 上的二元关系 $\llbracket H \rrbracket = \{\langle a, a \rangle, \langle c, c \rangle\}$ 。这里解释的论域集、谓词符号的解释，都使用元素枚举法给出了这些集合的元素。

定义 \mathcal{M} 的一个个体变量指派函数 $\sigma: V \rightarrow D$ ， $\sigma(x) = a, \sigma(y) = b, \sigma(z) = c$ 。这样，对于个体变量 x ，以及论域 D 的元素 c ，则有 $\sigma[x \mapsto c](x) = c, \sigma[x \mapsto c](y) = \sigma(y) = b, \sigma[x \mapsto c](z) = \sigma(z) = c$ 。同样，这里定义函数也是使用元素枚举法给出定义域（即个体变量集 V ）的每个元素的函数值。

3.3.2 一阶逻辑公式的真值

一阶逻辑公式在给定的非逻辑符号集和个体变量集的基础上归纳构造，相应地，一阶逻辑公式的真值则在给定非逻辑符号集的解释以及个体变量指派函数的基础上确定。在给出一阶逻辑公式的语法时，先定义了一阶逻辑公式中的项，同样这里也需要先定义项的语义。

定义 3.8 设一阶逻辑公式的非逻辑符号集是 \mathcal{L} ，个体变量集是 V 。给定 \mathcal{L} 的解释 \mathcal{M} ，以及一个个体变量指派函数 $\sigma: V \rightarrow D$ ，这里 D 是解释 \mathcal{M} 的论域。

在解释 \mathcal{M} 及个体变量指派函数 σ 下，一阶逻辑公式的项 t 的语义解释是论域 D 的元素，记为 $\llbracket t \rrbracket_\sigma$ ，根据项 t 的结构归纳定义如下。

(1) 如果项 t 是个体常量 c ，则 $\llbracket t \rrbracket_\sigma = \llbracket c \rrbracket$ ，这里 $\llbracket c \rrbracket \in D$ 是个体常量符号 c 在 \mathcal{M} 的解释。

(2) 如果项 t 是个体变量 x ，则 $\llbracket x \rrbracket_\sigma = \sigma(x)$ 。

(3) 如果项 t 是 $f(t_1, t_2, \dots, t_n)$, 则

$$[[t]]_{\sigma} = [[f(t_1, t_2, \dots, t_n)]]_{\sigma} = [[f]]([t_1]_{\sigma}, [t_2]_{\sigma}, \dots, [t_n]_{\sigma})$$

这里 $[[f]] : D^n \rightarrow D$ 是 n 元函数符号 f 在 \mathcal{M} 的解释。

例子 3.7 设一阶逻辑公式非逻辑符号集包括个体常量符号 $0, 1$, 二元函数符号 $+$ 和 $*$, 以及二元谓词符号 $\leq, =$ 。给定解释 \mathcal{M}_1 , 论域是自然数集 \mathbb{N} , 个体常量符号 $0, 1$ 分别解释为自然数 $0, 1$, 二元函数符号 $+$ 和 $*$ 分别解释为自然数集上的加法和乘法运算, 二元谓词符号 $\leq, =$ 分别解释为自然数集上的小于或等于以及相等关系。

设个体变量集 $V = \{x, y\}$, 个体变量指派函数 $\sigma : V \rightarrow \mathbb{N}$ 定义为: $\sigma(x) = 0, \sigma(y) = 1$ 。我们知道, $0, 1, x + y, x * y$ 都是基于这里定义的非逻辑符号集和个体变量集构造的一阶公式中的项, 它们的语义解释如下:

$$\begin{aligned} [[0]]_{\sigma} &= 0 & [[1]]_{\sigma} &= 1 \\ [[x + y]]_{\sigma} &= [[x]]_{\sigma} + [[y]]_{\sigma} = \sigma(x) + \sigma(y) = 1 & [[x * y]]_{\sigma} &= [[x]]_{\sigma} * [[y]]_{\sigma} = \sigma(x) * \sigma(y) = 0 \end{aligned}$$

注意, 这里在对象语言 (即一阶逻辑公式) 和元语言 (公式的解释) 中使用了相同的符号, 读者注意区别。例如在 $[[0]]_{\sigma}$ 中的 0 是指一阶逻辑公式中的个体常量, 而等号后面的 0 是自然数 0 , 在 $[[x + y]]_{\sigma}$ 中的 $+$ 是一阶逻辑公式中的函数符号, 而在 $[[x]]_{\sigma} + [[y]]_{\sigma}$ 中的 $+$ 是自然数加法运算。

定义 3.9 设一阶逻辑公式的非逻辑符号集是 \mathcal{L} , 个体变量集是 V 。给定 \mathcal{L} 的解释 \mathcal{M} , 以及一个个体变量指派函数 $\sigma : V \rightarrow D$, 这里 D 是解释 \mathcal{M} 的论域。

在解释 \mathcal{M} 及个体变量指派函数 σ 下, 一阶逻辑公式 A 的语义解释是 $\mathbf{2} = \{1, 0\}$ 的元素, 即公式 A 的真值, 记为 $\sigma(A)$, 根据公式 A 的结构归纳定义如下。

(1) 如果公式 A 是原子公式 $F(t_1, t_2, \dots, t_n)$, 则

$$\sigma(A) = \mathbf{1} \text{ 当且仅当 } \langle [[t_1]]_{\sigma}, [[t_2]]_{\sigma}, \dots, [[t_n]]_{\sigma} \rangle \in [[F]]$$

这里 $[[F]] \in D^n$ 是 n 元谓词符号 F 在 \mathcal{M} 的解释。

(2) 如果公式 A 是 $(\neg B)$, 则 $\sigma(A) = \mathbf{1}$ 当且仅当 $\sigma(B) = \mathbf{0}$ 。

(3) 如果公式 A 是 $(B \wedge C)$, 则 $\sigma(A) = \mathbf{1}$ 当且仅当 $\sigma(B) = \sigma(C) = \mathbf{1}$ 。

(4) 如果公式 A 是 $(B \vee C)$, 则 $\sigma(A) = \mathbf{0}$ 当且仅当 $\sigma(B) = \sigma(C) = \mathbf{0}$ 。

(5) 如果公式 A 是 $(B \rightarrow C)$, 则 $\sigma(A) = \mathbf{0}$ 当且仅当 $\sigma(B) = \mathbf{1}$ 且 $\sigma(C) = \mathbf{0}$ 。

(6) 如果公式 A 是 $(B \leftrightarrow C)$, 则 $\sigma(A) = \mathbf{1}$ 当且仅当 $\sigma(B) = \sigma(C)$ 。

(7) 如果公式 A 是 $\forall x B$, 则 $\sigma(A) = \mathbf{1}$ 当且仅当对论域 D 的任意元素 d 有 $\sigma[x \mapsto d](B) = \mathbf{1}$ 。

(8) 如果公式 A 是 $\exists x B$, 则 $\sigma(A) = \mathbf{1}$ 当且仅当存在论域 D 的元素 d 使得 $\sigma[x \mapsto d](B) = \mathbf{1}$ 。

当公式的真值是 **1** 时人们通俗地称这个公式的真值为真, 否则称这个公式的真值为假。简单地说, 原子公式的真值取决于谓词作用的项的解释 (即 D 的元素) 是否具有谓词对应的关系 (即谓词的解释), 逻辑运算符构造的公式的真值确定方法与命题逻辑完全相同, 量词公式的真值使用定义 3.7 给出的个体变量指派函数的变换将全称量词公式 $\forall xB$ 定义为对论域的每个元素都使得公式 B 的真值为真, 而存在量词公式 $\exists xB$ 定义为存在论域的元素使得公式 B 的真值为真。

问题 3.8 设一阶逻辑公式的非逻辑符号集 \mathcal{L} 只包含一个一元谓词符号 F , 以及两个二元谓词符号 G, H , 个体变量集 $V = \{x, y, z\}$ 。

定义 \mathcal{L} 的一个解释 \mathcal{M} : 论域 $D = \{a, b, c\}$, 谓词符号 F 的解释 $\llbracket F \rrbracket = \{a, b\} \subseteq D$, 谓词符号 G 的解释是 $\llbracket G \rrbracket = \{\langle a, b \rangle, \langle b, c \rangle\}$, 谓词符号 H 的解释是 $\llbracket H \rrbracket = \{\langle a, a \rangle, \langle c, c \rangle\}$ 。

定义 \mathcal{M} 的一个个体变量指派函数 $\sigma: V \rightarrow D$, $\sigma(x) = a, \sigma(y) = b, \sigma(z) = c$ 。确定下面公式在 \mathcal{M} 和 σ 下的真值。

- (1) $\forall x(F(x) \rightarrow \exists yG(x, y))$
- (2) $\forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z))$
- (3) $F(x) \wedge G(x, y) \wedge \forall zH(y, z)$

【解答】(1) 记 $A = \forall x(F(x) \rightarrow \exists yG(x, y))$, 它具有形式 $\forall xB$, 其中 $B = F(x) \rightarrow \exists yG(x, y)$, 因此 $\sigma(A)$ 为真当且仅当对论域 D 的元素 a, b, c 有

$$\sigma[x \mapsto a](B) = \mathbf{1} \text{ 且 } \sigma[x \mapsto b](B) = \mathbf{1} \text{ 且 } \sigma[x \mapsto c](B) = \mathbf{1}$$

B 是蕴涵式, 因此 $\sigma[x \mapsto a](B) = \mathbf{0}$ 当且仅当 $\sigma[x \mapsto a](F(x)) = \mathbf{1}$ 且 $\sigma[x \mapsto a](\exists yG(x, y)) = \mathbf{0}$ 。而

$$\begin{aligned} \sigma[x \mapsto a](F(x)) = \mathbf{1} \quad \text{当且仅当} \quad \llbracket x \rrbracket_{\sigma[x \mapsto a]} \in \llbracket F \rrbracket \\ \text{当且仅当} \quad \sigma[x \mapsto a](x) \in \llbracket F \rrbracket = \{a, b\} \end{aligned}$$

注意到 $\sigma[x \mapsto a](x) = a$, 因此 $\sigma[x \mapsto a](F(x)) = \mathbf{1}$ 。由于 F 是谓词, 为简便起见, 当 F 解释为 D 的子集 $\{a, b\}$ 时, 可理解为 $F(a), F(b)$ 为真 (因为 a, b 属于这个子集), 而 $F(c)$ 为假 (因为 c 不属于这个子集), 这样 $\sigma[x \mapsto a](F(x))$ 的真值与 $F(a)$ 相同。

对于 $\sigma[x \mapsto a](\exists yG(x, y))$, 根据存在量词的语义, $\sigma[x \mapsto a](\exists yG(x, y)) = \mathbf{1}$ 当且仅当对论域 D 的元素 a, b, c 有

$$\begin{aligned} \sigma[x \mapsto a][y \mapsto a](G(x, y)) = \mathbf{1} \text{ 或 } \sigma[x \mapsto a][y \mapsto b](G(x, y)) = \mathbf{1} \\ \text{或 } \sigma[x \mapsto a][y \mapsto c](G(x, y)) = \mathbf{1} \end{aligned}$$

类似地, 当 G 解释为 $D \times D$ 的子集 $\{\langle a, b \rangle, \langle b, c \rangle\}$, 可理解为 $G(a, b), G(b, c)$ 为真, 而对于 $D \times D$ 的其他有序对都为假。注意到 $\sigma[x \mapsto a][y \mapsto a]$ 总是将 x 指派为 a , 将 y 也指派为 a , 因此可认为 $\sigma[x \mapsto a][y \mapsto a](G(x, y))$ 的真值与 $G(a, a)$ 相同, $\sigma[x \mapsto a][y \mapsto b](G(x, y))$ 的真值与 $G(a, b)$ 相同, $\sigma[x \mapsto a][y \mapsto c](G(x, y))$ 的真值与 $G(a, c)$ 相同。

更进一步,可认为 $\sigma[x \mapsto a](\exists yG(x, y))$ 的真值与 $\exists yG(a, y)$ 的真值相同,而 $\exists yG(a, y)$ 的真值与 $G(a, a) \vee G(a, b) \vee G(a, c)$ 的真值相同,而 $\sigma[x \mapsto a](B) = \sigma[x \mapsto a](F(x) \rightarrow \exists yG(x, y))$ 的真值与 $F(a) \rightarrow \exists yG(a, y)$ 的真值相同,进一步与 $F(a) \rightarrow (G(a, a) \vee G(a, b) \vee G(a, c))$ 的真值相同。对 $\sigma[x \mapsto b](B), \sigma[x \mapsto c](B)$ 也做类似地处理,这样在给定的解释和个体变量指派函数下,可使用类似等值演算的方式确定公式 A 的真值:

$$\begin{aligned} & \forall x(F(x) \rightarrow \exists yG(x, y)) \\ \equiv & (F(a) \rightarrow \exists yG(a, y)) \wedge (F(b) \rightarrow \exists yG(b, y)) \wedge (F(c) \rightarrow \exists yG(c, y)) \\ \equiv & (F(a) \rightarrow (G(a, a) \vee G(a, b) \vee G(a, c))) \wedge (F(b) \rightarrow (G(b, a) \vee G(b, b) \vee G(b, c))) \wedge \\ & (F(c) \rightarrow (G(c, a) \vee G(c, b) \vee G(c, c))) \\ \equiv & (\mathbf{1} \rightarrow (\mathbf{0} \vee \mathbf{1} \vee \mathbf{0})) \wedge (\mathbf{1} \rightarrow (\mathbf{0} \vee \mathbf{0} \vee \mathbf{1})) \wedge (\mathbf{0} \rightarrow (\mathbf{0} \vee \mathbf{0} \vee \mathbf{0})) \equiv \mathbf{1} \wedge \mathbf{1} \wedge \mathbf{1} \equiv \mathbf{1} \end{aligned}$$

综上,有 $\sigma(\forall x(F(x) \rightarrow \exists yG(x, y))) = \mathbf{1}$ 。

(2) 直接用类似等值演算的方式确定公式 $\forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z))$ 的真值:

$$\begin{aligned} & \forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z)) \\ \equiv & (F(a) \wedge F(b) \wedge F(c)) \wedge \exists y(F(y) \wedge \forall zH(y, z)) \\ \equiv & (\mathbf{1} \wedge \mathbf{1} \wedge \mathbf{0}) \wedge \exists y(F(y) \wedge \forall zH(y, z)) \end{aligned}$$

根据合取运算符的特点,到这一步已经能确定这个公式的真值为假。不过为了让大家进一步熟悉这种演算方式,继续计算 $\exists y(F(y) \wedge \forall zH(y, z))$ 的真值:

$$\begin{aligned} & \exists y(F(y) \wedge \forall zH(y, z)) \\ \equiv & (F(a) \wedge \forall zH(a, z)) \vee (F(b) \wedge \forall zH(b, z)) \vee (F(c) \wedge \forall zH(c, z)) \\ \equiv & (F(a) \wedge (H(a, a) \wedge H(a, b) \wedge H(a, c))) \vee (F(b) \wedge (H(b, a) \wedge H(b, b) \wedge H(b, c))) \vee \\ & (F(c) \wedge (H(c, a) \wedge H(c, b) \wedge H(c, c))) \\ \equiv & (\mathbf{1} \wedge (\mathbf{1} \wedge \mathbf{0} \wedge \mathbf{0})) \vee (\mathbf{1} \wedge (\mathbf{0} \wedge \mathbf{0} \wedge \mathbf{0})) \vee (\mathbf{0} \wedge (\mathbf{0} \wedge \mathbf{0} \wedge \mathbf{1})) \equiv \mathbf{0} \vee \mathbf{0} \vee \mathbf{0} \equiv \mathbf{0} \end{aligned}$$

综上,有 $\sigma(\forall xF(x) \wedge \exists y(F(y) \wedge \forall zH(y, z))) = \mathbf{0}$ 。

(3) 公式 $F(x) \wedge G(x, y) \wedge \forall zH(y, z)$ 在 \mathcal{M} 和 σ 下的真值为真,当且仅当 $\sigma(F(x))$, $\sigma(G(x, y))$ 以及 $\sigma(\forall zH(y, z))$ 的值都为 $\mathbf{1}$,注意到 $\sigma(x) = a, \sigma(y) = b$,因此 $\sigma(F(x))$ 的值等于 $F(a)$ 的真值, $\sigma(G(x, y))$ 的值等于 $G(a, b)$ 的真值,它们都为真,因此这个公式的真值等于 $\sigma(\forall zH(y, z))$,即等于 $H(b, a) \wedge H(b, b) \wedge H(b, c)$ 的真值,即为假。总之,有 $\sigma(F(x) \wedge G(x, y) \wedge \forall zH(y, z)) = \mathbf{0}$ 。

【讨论】(1) 这里给出的解释的论域是有限集 $D = (a, b, c)$,因此对于全称量词公式 $\forall xF(x)$,它的真值等于 $F(a) \wedge F(b) \wedge F(c)$ 的真值,而对于存在量词公式 $\exists xF(x)$,它的真值等于 $F(a) \vee F(b) \vee F(c)$ 的真值。也就是说,全称量词公式在有限论域可展开为合取式,存在量词公式在有限论域可展开为析取式。

注意, 严格来说, $F(a), F(b), F(c)$ 都不是由给定的非逻辑符号集 \mathcal{L} 和个体变量集 V 能构造的公式, 因为 a, b, c 不是 \mathcal{L} 中的符号。所以上面的演算严格来说不是逻辑公式的等值演算, 而是当解释的论域是有限集时, 我们确定一阶公式真值的一种简便计算方式。将谓词符号作用于论域的元素或有序对, 它的真值为真当且仅当所作用的元素或有序对属于谓词符号的解释所对应的子集或关系, 例如, $F(a)$ 为真当且仅当 $a \in \llbracket F \rrbracket$ 。

(2) 上面 (1) 和 (2) 给出的公式都是闭公式, 不含有自由变量, 可以看到它们的真值与个体变量指派函数 σ 没有关系。简单地说, 对于量词公式 $\forall xF(x)$ 或 $\exists xF(x)$, 它们的真值与个体变量指派函数 σ 对个体变量 x 的指派无关。上面 (3) 给出的公式 $F(x) \wedge G(x, y) \wedge \forall zH(y, z)$ 含有自由变量 x 和 y , 这时它的真值与 $\sigma(x)$ 和 $\sigma(y)$ 的值有关, 但 z 是它的约束变量, 所以它的真值与 $\sigma(z)$ 无关。

一阶公式的真值与个体变量指派函数对它的约束变量所指派的值无关, 这样可得到如下定理。

定理 3.1 设公式 $\forall xA$ 通过约束变量改名得到公式 $\forall yA'$, 即 A' 是将 A 中所有自由出现的 x 改为 y 得到的公式, 这里设 y 不在 A 中出现, 则在任意的解释 \mathcal{M} 和任意的个体变量指派函数 σ 下, 公式 $\forall xA$ 和 $\forall yA'$ 都有相同的真值。类似地, 若公式 $\exists xA$ 通过约束变量改名得到公式 $\exists yA'$, 则 $\exists xA$ 与 $\exists yA'$ 在任意解释和个体变量指派函数下也有相同的真值。 \square

下面是使用类似等值演算的方法确定闭公式真值的例子, 其中的闭公式都是嵌套量词公式。所谓的嵌套量词公式, 是指一个量词的辖域中还包括另外的量词。我们也用这个例子说明一阶公式中含有多个量词时, 量词的顺序会影响公式的真值。

例子 3.9 设非逻辑符号集 \mathcal{L} 只包含二元谓词 F , 个体变量集 $V = \{x, y\}$ 。定义解释 \mathcal{M} , 论域 $D = \{a, b\}$, 二元谓词 F 的解释是 $\llbracket F \rrbracket = \{\langle a, b \rangle, \langle b, b \rangle\}$, 即 $F(a, b)$ 和 $F(b, b)$ 为真, 而 $F(a, a)$ 和 $F(b, a)$ 为假。

(1) 对于公式 $\forall x\forall yF(x, y)$, 有:

$$\begin{aligned}\forall x\forall yF(x, y) &\equiv \forall yF(a, y) \wedge \forall yF(b, y) \\ &\equiv (F(a, a) \wedge F(a, b)) \wedge (F(b, a) \wedge F(b, b)) \equiv \mathbf{0}\end{aligned}$$

(2) 对于公式 $\forall y\forall xF(x, y)$, 有:

$$\begin{aligned}\forall y\forall xF(x, y) &\equiv \forall xF(x, a) \wedge \forall xF(x, b) \\ &\equiv (F(a, a) \wedge F(b, a)) \wedge (F(a, b) \wedge F(b, b)) \equiv \mathbf{0}\end{aligned}$$

由 (1) 和 (2) 容易看到, $\forall x\forall yF(x, y)$ 和 $\forall y\forall xF(x, y)$ 具有相同的真值。类似地, 读者可自行验证公式 $\exists x\exists yF(x, y)$ 和 $\exists y\exists xF(x, y)$ 也具有相同的真值, 即如果都是存在量词, 或者都是全称量词 (指示变量不同), 则交换量词的顺序不影响公式的真值。

(3) 对于公式 $\forall x\exists yF(x, y)$, 有:

$$\begin{aligned}\forall x\exists yF(x, y) &\equiv \exists yF(a, y) \wedge \exists yF(b, y) \\ &\equiv (F(a, a) \vee F(a, b)) \wedge (F(b, a) \vee F(b, b)) \equiv \mathbf{1}\end{aligned}$$

(4) 对于公式 $\exists y\forall xF(x, y)$, 有:

$$\begin{aligned}\exists y\forall xF(x, y) &\equiv \forall xF(x, a) \vee \forall xF(x, b) \\ &\equiv (F(a, a) \wedge F(b, a)) \vee (F(a, b) \wedge F(b, b)) \equiv \mathbf{0}\end{aligned}$$

由 (3) 和 (4) 容易看到, $\forall x\exists yF(x, y)$ 和 $\exists y\forall xF(x, y)$ 具有不同的真值。类似地, 读者可自行验证公式 $\exists x\forall yF(x, y)$ 和 $\forall y\exists xF(x, y)$ 也具有不同的真值。实际上, 这四个公式中的任意两个都可能具有不同的真值。

(5) 对于公式 $\exists x\forall yF(x, y)$, 有:

$$\begin{aligned}\exists x\forall yF(x, y) &\equiv \forall yF(a, y) \vee \forall yF(b, y) \\ &\equiv (F(a, a) \wedge F(a, b)) \vee (F(b, a) \wedge F(b, b)) \equiv \mathbf{0}\end{aligned}$$

可以看到, 当 F 的解释是关系 $\{\langle a, b \rangle, \langle b, b \rangle\}$ 时, $\exists y\forall xF(x, y)$ 和 $\exists x\forall yF(x, y)$ 的真值都是 $\mathbf{0}$, 但不难给出 F 的另一个解释, 例如它的解释是关系 $\{\langle a, a \rangle, \langle a, b \rangle\}$ 时, 这两个公式就具有不同的真值。同样, 容易给出 F 的解释使得 $\forall x\exists yF(x, y)$ 与 $\forall y\exists xF(x, y)$ 也具有不同的真值。

总之, 如果公式中的两个量词类型不同 (即一个是全称量词一个是存在量词), 则交换量词的顺序会影响公式的真值。

上面讨论了非逻辑符号的解释的论域是有限集的情况, 下面的问题讨论当非逻辑符号解释的论域不是有限集时如何确定一阶公式的真值。

问题 3.10 设一阶逻辑公式非逻辑符号集包括个体常量符号 $0, 1$, 二元函数符号 $+$ 和 $*$, 以及二元谓词符号 $\leq, =$ 。个体变量集 $V = \{x, y\}$ 。

给定解释 \mathcal{M}_1 , 论域是自然数集 \mathbb{N} , 个体常量符号 $0, 1$ 分别解释为自然数 $0, 1$, 二元函数符号 $+$ 和 $*$ 分别解释为自然数集上的加法和乘法运算, 二元谓词符号 $\leq, =$ 分别解释为自然数集上的小于或等于以及相等关系。确定下面闭公式的真值。

$$\begin{aligned}(1) \forall x((0 \leq x) \rightarrow \exists y(x = y * y)) & \quad (2) \forall x\exists y(x \leq y) \\ (3) (\exists x\forall y(x + y = y) \wedge \forall x\exists y(x * y = 1)) & \quad (4) \exists x\forall y(x \leq y)\end{aligned}$$

【解答】(1) 公式 $\forall x((0 \leq x) \rightarrow \exists y(x = y * y))$ 的真值为真当且仅当对任意的自然数 n , $n \geq 0$ 蕴涵存在自然数 m 使得 $n = m * m$ 成立, 但显然不是所有的自然数 n 都存在自然数 m 使得 $n = m * m$ 成立, 因此这个公式的真值为假。

(2) 公式 $\forall x\exists y(x \leq y)$ 的真值为真当且仅当对任意的自然数 n , 存在自然数 m 使得 $n \leq m$ 成立, 也即, 任意自然数都有大于或等于它的自然数, 这显然成立, 因此这个公式的真值为真。

(3) 公式 $(\exists x\forall y(x + y = y) \wedge \forall x\exists y(x * y = 1))$ 为真当且仅当公式 $\exists x\forall y(x + y = y)$ 为真且公式 $\forall x\exists y(x * y = 1)$ 为真。而公式 $\exists x\forall y(x + y = y)$ 为真, 当且仅当存在自然数 n , 使得对任意自然数 m , 都有 $n + m = m$ 成立, 这显然成立, 因为存在自然数 0 , 对任意的自然数都有 $0 + m = m$, 所以公式 $\exists x\forall y(x + y = y)$ 的真值为真; 公式 $\forall x\exists y(x * y = 1)$ 为真当且仅当对任意的自然数 n , 都存在自然数 m 使得 $n * m = 1$, 这显然不成立, 因此公式 $\forall x\exists y(x * y = 1)$ 的真值为假, 因此整个公式的真值为假。

(4) 公式 $\exists x \forall y (x \leq y)$ 的真值为真当且仅当存在自然数 n , 对任意的自然数 m , 都有 $n \leq m$, 也即存在小于或等于任意自然数的自然数, 也即存在最小的自然数, 这显然成立, 因为 0 是最小的自然数, 因此这个公式的真值为真。

【讨论】 (1) 当解释的论域不是有限集时, 对于全称量词公式 $\forall x A(x)$ 只能直观地说, 它的真值为真, 当且仅当对论域的每个元素 d 有 $A(d)$ 成立, 这里 $A(d)$ 是用论域元素 d 替换 A 中自由出现的 x 后得到的命题, 而对于存在量词公式 $\exists x A(x)$ 只能直观地说, 它的真值为真, 当且仅当存在论域的元素 d 使得 $A(d)$ 成立。

当存在论域的元素 d 使得命题 $A(d)$ 不为真时, $\forall x A(x)$ 的真值为假, 而当存在论域的元素 d 使得命题 $A(d)$ 为真时, $\exists x A(x)$ 为真, 对于这两种情况, 我们都应该具体地指出 d 到底是论域的哪个元素。

注意, 这里说的 $A(d)$ 严格来说不是一阶逻辑公式, 因为 d 是论域元素而不是非逻辑符号集中的个体常量符号, 但可将 $A(d)$ 看作是使用 d 替换 A 中自由出现的 x 后的命题, 这个命题是元语言的命题, 而非对象语言 (即一阶逻辑) 中的公式, 然后根据与论域相关的知识判断该命题是否成立, 以说明量词公式的真值。

(2) 当一阶逻辑公式的解释的论域是有穷集时, 可以像例子 3.8 一样使用类似等值演算的方法将全称量词公式展开为合取式, 存在量词公式展开为析取式而确定公式真值。但一般来说, 由于一阶逻辑公式的解释可以有无穷多, 解释的论域也可能是无穷集, 因此不存在通用算法能确定任意一阶逻辑公式的真值。这也是本章没有介绍一阶逻辑公式语法和语义相关算法的主要原因。

3.3.3 一阶逻辑公式的分类

根据构造一阶逻辑公式的最后一步所使用的逻辑运算或量词, 一阶逻辑公式从语法上可分为否定式、合取式、析取式、蕴涵式、双蕴涵式, 以及全称量词公式和存在量词公式。从语义角度, 也即从公式的真值角度, 一阶逻辑公式和命题逻辑公式一样也可分为永真式、矛盾式和可满足式。

定义 3.10 设 A 是基于非逻辑符号集 \mathcal{L} 和个体变量集 V 构造的一个一阶逻辑公式。

(1) 如果 A 在 \mathcal{L} 的任意解释, 以及任意个体变量指派函数下的真值都为真, 则称 A 为**永真式** (tautology), 也称为**普遍有效式**。

(2) 如果 A 在 \mathcal{L} 的任意解释, 以及任意个体变量指派函数下的真值都为假, 则称 A 为**矛盾式** (contradiction)。

(3) 如果 A 不是矛盾式, 即存在一个 \mathcal{L} 的解释, 及一个个体变量指派函数使得 A 的真值为真, 则称 A 是**可满足的** (satisfiable)。

显然, 公式 A 是永真式当且仅当 $(\neg A)$ 是矛盾式。可根据定义, 证明一些一阶逻辑公式是永真式。

例子 3.11 设 F 是构造一阶逻辑公式的非逻辑符号集中的一元谓词, x 是个体变量。一阶逻辑公式 $\neg(\forall x F(x)) \leftrightarrow \exists x(\neg F(x))$ 是永真式。

因为, 对非逻辑符号集的任意解释, $\neg(\forall xF(x))$ 为真当且仅当 $\forall xF(x)$ 为假。 $\forall xF(x)$ 为真当且仅当对解释论域的任意元素 d 有 $F(d)$ 为真, 因此 $\forall xF(x)$ 为假当且仅当存在论域的元素 d 使得 $F(d)$ 为假, 也即存在论域的元素 d 使得 $\neg F(d)$ 为真, 也即 $\exists x(\neg F(x))$ 为真。因此, 在任意解释下, 公式 $\neg(\forall xF(x))$ 和 $\exists x(\neg F(x))$ 都有相同的真值, 也即在任意解释下 $\neg(\forall xF(x)) \leftrightarrow \exists x(\neg F(x))$ 的真值都为真。

类似地, 一阶逻辑公式 $\neg(\exists xF(x)) \leftrightarrow \forall x(\neg F(x))$ 也是永真式。

例子 3.12 设 F 是构造一阶逻辑公式的非逻辑符号集中的二元谓词, x, y 是个体变量。一阶逻辑公式 $\exists x\forall yF(x, y) \rightarrow \forall y\exists xF(x, y)$ 是永真式。

因为对非逻辑符号集的任意解释, 若 $\exists x\forall yF(x, y)$ 为假, 则整个蕴涵式为真, 所以只需要考虑 $\exists x\forall yF(x, y)$ 为真的情况。而 $\exists x\forall yF(x, y)$ 为真当且仅当存在解释论域的元素 d_0 , 使得对论域的任意元素 d , 都有 $F(d_0, d)$ 为真, 从而对论域的任意元素 d , 都存在论域元素 d_0 使得 $F(d_0, d)$ 为真, 也即 $\forall y\exists xF(x, y)$ 为真, 因此在任意解释下蕴涵式 $\exists x\forall yF(x, y) \rightarrow \forall y\exists xF(x, y)$ 总为真。

不存在通用算法确定一个一阶逻辑公式的真值, 因此也不存在通用算法判定一个公式是否永真式、矛盾式或可满足式。判断一个逻辑公式是否可满足的, 称为逻辑公式的可判定问题。命题逻辑公式的可判定问题可以由计算机自动求解, 例如, 可以构造命题逻辑公式的真值表。但邱奇 (Alonzo Church, 1903—1995) 和图灵 (Alan Turing, 1912—1954) 早在 1936 年就分别独立证明了一阶逻辑公式的判定问题不是计算机可解的。实际上, 一阶逻辑公式是半可判定的, 即存在通用的算法, 对任意的一阶逻辑公式, 如果它是可满足的, 则该算法会终止并回答说它是可满足的, 但如果它不是可满足的, 则算法不会终止。

有一类称为命题逻辑公式替换实例的一阶逻辑公式, 可利用命题逻辑的永真式判断它们是否一阶逻辑的永真式。

定义 3.11 设 A 是命题逻辑公式, 其中出现的命题变量是 p_1, p_2, \dots, p_n , 用任意的一阶逻辑公式 A_1, A_2, \dots, A_n 分别替换 A 中 p_1, p_2, \dots, p_n 的所有出现得到的一阶逻辑公式称为命题逻辑公式 A 的替换实例。

例子 3.13 (1) 一阶逻辑公式 $\forall xF(x) \rightarrow (\exists xG(x, y) \rightarrow \forall xF(x))$ 是命题逻辑公式 $p \rightarrow (q \rightarrow p)$ 的替换实例, 是用 $\forall xF(x)$ 替换其中的 p , 而用 $\exists xG(x, y)$ 替换其中的 q 。

(2) 一阶逻辑公式 $\forall xF(x) \rightarrow \forall xF(x)$ 是命题逻辑公式 $p \rightarrow p$ 的替换实例, 但是 $\forall x(F(x) \rightarrow F(x))$ 不是命题逻辑公式 $p \rightarrow p$ 的替换实例。

定理 3.2 若一阶逻辑公式 A 是命题逻辑某个永真式的替换实例, 则 A 也是一阶逻辑的永真式。类似地, 若一阶逻辑公式 A 是命题逻辑某个矛盾式的替换实例, 则 A 也是一阶逻辑公式的矛盾式。

证明 设公式 A 是命题逻辑永真式 B 的替换实例, 即是使用某些一阶逻辑公式 B_1, B_2, \dots, B_n 分别替换 B 中出现的命题变量 p_1, p_2, \dots, p_n 的所有出现而得到的公式。

对构造公式 A 的非逻辑符号集的任意解释和任意个体变量指派函数, 在这个解释和个体变量指派函数下无论 B_1, B_2, \dots, B_n 的真值如何都相当于对命题变量 p_1, p_2, \dots, p_n 的一个真值赋值, 但永真式 B 的真值与 p_1, p_2, \dots, p_n 的真值赋值无关, 因此 A 在这个解释和个体变量指派函数下的真值也与 B_1, B_2, \dots, B_n 的真值无关, 也总是为真, 因此 A 是一阶逻辑公式的永真式。由于一个公式是永真式当且仅当它的否定是矛盾式, 因此命题逻辑矛盾式的替换实例是一阶逻辑的矛盾式。□

例子 3.14 (1) 不难验证命题逻辑公式 $p \rightarrow (q \rightarrow p)$ 是永真式, 因此 $\forall xF(x) \rightarrow (\exists xG(x, y) \rightarrow \forall xF(x))$ 是一阶逻辑的永真式。

(2) 不难验证命题逻辑公式 $p \rightarrow p$ 是永真式, 因此 $\forall xF(x) \rightarrow \forall xF(x)$ 是一阶逻辑的永真式, $F(x) \rightarrow F(x)$ 也是一阶逻辑的永真式。可以证明, 如果 x 是 A 的自由变量, 则 A 是永真式当且仅当 $\forall xA$ 是永真式, 因此由 $F(x) \rightarrow F(x)$ 是永真式可得到 $\forall x(F(x) \rightarrow F(x))$ 也是永真式, 虽然它不是命题逻辑永真式 $p \rightarrow p$ 的替换实例。

问题 3.15 判断下面一阶逻辑公式的类型 (即是永真式、矛盾式, 还是非永真的可满足式)。

$$(1) \forall xF(x) \rightarrow \exists xF(x)$$

$$(2) \exists xF(x) \rightarrow \forall xF(x)$$

$$(3) (\forall xF(x) \wedge \exists xF(x)) \rightarrow (\forall xF(x) \vee \exists xF(x))$$

【分析】 判断一阶逻辑公式的类型首先观察它是否某个合适的命题逻辑公式的替换实例, 如果是, 则利用命题逻辑的真值表构造或等值演算判断这个命题逻辑公式是否永真式或矛盾式。虽然没有固定方法给出一个一阶逻辑公式是否某个命题逻辑公式的替换实例, 但对许多简单的一阶逻辑公式都容易确定它是哪个命题逻辑公式的替换实例。

如果不能通过命题逻辑公式的替换实例判断一阶逻辑公式的类型, 则只能根据一阶逻辑公式永真式、矛盾式和可满足式的定义进行判断。对于永真式要证明它在任意解释、任意个体变量指派函数下为真; 对于矛盾式要证明它在任意解释、任意个体变量指派函数下为假; 而对于非永真的可满足式, 则可以给出一个具体的解释和个体变量指派函数使得这个公式为真, 并且再给出一个具体的解释和个体变量指派函数使得这个公式为假。当待判断的一阶公式是闭公式时, 则可以不考虑个体变量指派函数, 因为闭公式的真值与个体变量指派函数无关。

【解答】 (1) 公式 $\forall xF(x) \rightarrow \exists xF(x)$ 虽然可看作命题逻辑公式 $p \rightarrow q$ 的替换实例, 但 $p \rightarrow q$ 不是永真式。因此, 只能通过定义判断它的类型。对于任意的解释, 如果 $\forall xF(x)$ 为假, 则整个蕴涵式为真。如果 $\forall xF(x)$ 为真, 则对解释的论域的任意元素 d 有 $F(d)$ 为真, 注意到论域不能为空, 因此至少存在一个元素 d_0 使得 $F(d_0)$ 为真, 因此 $\exists xF(x)$ 也为真, 因此蕴涵式 $\forall xF(x) \rightarrow \exists xF(x)$ 也为真。因此, 在任意的解释下这个公式的真值都为真, 即它是永真式。

(2) 同样只能通过定义判断它的类型, 直观地看, 对任意解释, 存在论域一个元素 d_0 使得 $F(d_0)$ 为真不能得到对论域的任意元素 d 都有 $F(d)$ 为真, 除非这个论域只有一个元素。另一方面, 当论域不存在元素 d 使得 $F(d)$ 为真时, $\exists xF(x)$ 为假, 从而整

个蕴涵式为真。因此公式 $\exists xF(x) \rightarrow \forall xF(x)$ 应该是非永真的可满足式。

为了使回答更为严谨,要给出一个具体的解释使得公式为真,以及一个具体的解释使得公式为假。设解释的论域 $D = \{a, b\}$, F 的解释是 D 的子集 $\{a\}$, 即 $F(a)$ 为真, 而 $F(b)$ 为假。这时 $\exists xF(x)$ 为真, 而 $\forall xF(x)$ 为假, 整个蕴涵式的真值在这个解释下为假。又设解释的论域 $D = \{a, b\}$, 但 F 的解释是空集, 即 $F(a)$ 和 $F(b)$ 都为假, 这时 $\exists xF(x)$ 为假, 从而整个蕴涵式为真。综上, 公式 $\exists xF(x) \rightarrow \forall xF(x)$ 是非永真的可满足式。

(3) 公式 $(\forall xF(x) \wedge \exists xF(x)) \rightarrow (\forall xF(x) \vee \exists xF(x))$ 是命题逻辑公式 $(p \wedge q) \rightarrow (p \vee q)$ 的替换实例, 容易验证这个命题逻辑公式是永真式, 因此这个一阶逻辑公式也是永真式。

综上, (1), (3) 给出的公式是永真式, 而 (2) 给出的公式是非永真的可满足式。

3.4 一阶逻辑的等值演算

与命题逻辑公式的逻辑等值一样, 两个一阶公式是否逻辑等值也可通过永真式判断, 而且命题逻辑公式的逻辑等值式的替换实例也是一阶逻辑公式的逻辑等值式, 这使得一阶逻辑公式中与逻辑运算符相关的等值演算方法与命题逻辑相同, 因此本节主要讨论与量词相关的一阶逻辑等值式。

下面先给出两个一阶逻辑公式逻辑等值的定义, 然后重点介绍与量词相关的一阶逻辑等值式, 最后介绍前束范式, 这是具有某种固定形式的一阶逻辑公式。后面会看到, 一阶逻辑的自然推理系统中与量词有关的推理规则只能应用于前束范式。

3.4.1 一阶逻辑公式的逻辑等值

定义 3.12 如果一阶逻辑公式 A 和 B 在非逻辑符号集的任意解释及任意个体变量指派函数下都有相同的真值, 则称 A 和 B **逻辑等值** (logically equivalent), 简称**等值**, 记为 $A \equiv B$ 。通常也称 $A \equiv B$ 为**逻辑等值式**。

当在研究两个或一组一阶逻辑公式是否逻辑等值时, 总假定所有公式是基于相同的非逻辑符号集和个体变量集构造。因此, 后面通常不再给出构造一阶公式的非逻辑符号集和个体变量集, 读者可从所研究公式的语法得到构造它们的非逻辑符号集应包括哪些个体常量、函数和谓词符号, 以及个体变量集应包括哪些个体变量符号。特别地, 在对公式进行约束变量改名或自由变量替换时, 假定个体变量集总包含所需的新的个体变量符号。

与命题逻辑相同, 两个一阶公式是否逻辑等值也可基于永真式判断, 因为根据一阶公式逻辑等值的定义以及逻辑双蕴涵运算符的真值计算方法, 可得到下面的定理。

定理 3.3 一阶逻辑公式 A 与 B 逻辑等值当且仅当一阶逻辑公式 $A \leftrightarrow B$ 是永真式。 \square

例子 3.16 不难根据逻辑等值的定义证明有下面的量词交换逻辑等值式: 设 F 是



PredLogic-
Equivalence(1)



PredLogic-
Equivalence(2)