第三章 一阶理论

Peano公设的形式化:

引入特殊符号: 0(个体常元), 函数符号$'$(后继), 一元谓词符号N

给出一个公式集Γ_N, 使得在它的每一个模型M_N中, 0, $'$, N必须/必然分别解释为自然数0, 后继函数和自然数(集合)

(P1) $N(0)$

(P2) $\forall x(N(x)\rightarrow \exists! y(y=x' \wedge N(y)))$ ∃!: 存在唯一

(P3) $\forall x \neg(0=x')$

(P4) $\forall x \forall y(x'=y' \rightarrow x=y)$

(P5) $P(0)\wedge \forall x(P(x)\rightarrow P(x'))\rightarrow \forall x P(x)$, P为任意谓词

$\exists! x P(x) =_{def} \exists x(P(x)\wedge \forall y(P(y)\leftrightarrow y=x))$

问题: = 没有定义

K^+的语言$K^+(Y)$比$K(Y)$多一个符号"≈", 称为等词, 在K上增加3条"等词公设"

(E1) $u \approx u$

(E2) $u_k \approx u \rightarrow (f_i^n(u_1,\cdots,u_k,\cdots,u_n)\approx f_i^n(u_1,\cdots,u,\cdots,u_n))$

(E3) $u_k \approx u \rightarrow (P_i^n(u_1,\cdots,u_k,\cdots,u_n)\approx P_i^n(u_1,\cdots,u,\cdots,u_n))$

任给一阶结构$M=(\mathbb{D},\mathbb{F},\mathbb{P})$, 若$\approx^M$为$\mathbb{D}$上的相等, 则所有等词公设都是M有效的

在K^+的任何模型M中, $\approx^M$是否一定是$\mathbb{D}$上的相等? 未必

等词公设不强迫"任何K^+模型一定把≈解释为相等 同余偶…

1101C-08 201412·2500

由 扫描全能王 扫描创建

若 $M=(\mathbb{D},\mathbb{F},\mathbb{P})$ 是一个 K^+ 模型，则 $\approx^M$ 是 $\mathbb{D}$ 上等价关系

等项可替换性：

1° $\vdash_{K^+} u\approx v\to t(u)\approx t(v)$，其中项 u 是项 t 的子项，项 $t(v)$ 是项 $t(u)$ 中用 v 替换 u 的结果

2° $\vdash_{K^+} u\approx v\to(p(u)\to p(v))$，其中 $p(x)$ 是任意 K^+ 公式，u,v 对 $p(x)$ 中 x 自由

正规模型：设 $\Gamma\subseteq K^+(Y)$，$M=(\mathbb{D},\mathbb{F},\mathbb{P})$ 是 Γ 的一个 K^+ 模型，若 $\approx^M$ 为 $\mathbb{D}$ 上的相等，则称 M 为 Γ 的一个正规模型

若 Γ 有 K^+ 模型，则 Γ 一定有正规 K^+ 模型：

$M^{\approx}=(\mathbb{D}^{\approx},\mathbb{F}^{\approx},\mathbb{P}^{\approx})$ 是 Γ 的 K^+ 正规模型.

$\mathbb{D}^{\approx}$ 是由 $\mathbb{D}$ 中关于 $\approx^M$ 的等价类为个体组成的集合 $\{[x]\mid x\in\mathbb{D}\}$

$[x]=_{def}\{y\in\mathbb{D}\mid y\approx^M x\}$，$\mathbb{F}$ 中所有函数的定义域、值域，$\mathbb{P}$ 中所有关系的定义域，都变换为 $\mathbb{D}^{\approx}$

非正规模型的必然性：设 Σ^* 为 Σ 的任何相容扩张，则 Σ^* 有非正规模型.

形式语言 $K_N(Y)$：

逻辑符号：同 K

非逻辑符号：

个体常元：$\bar{0}$

一元函数符号：$'$

二元函数符号：$\bar{+}$，$\bar{\times}$

谓词符号：$\approx$

1101C-08 201412·2500

由 扫描全能王 扫描创建

形成规则：同K

公理：逻辑公理(模式)：同K

非逻辑公理/公设：等词公设：同K^+

(N1) $\neg(u' \approx \overline{0})$

(N2) $v' \approx u' \to v \approx u$

$$\left.\begin{array}{l}\text{(N3)}\ u \# \overline{0} \approx u \\ \text{(N4)}\ v \# u' \approx (v \# u)'\end{array}\right\}\text{加法的递归定义}$$

$$\left.\begin{array}{l}\text{(N5)}\ v \divideontimes \overline{0} \approx \overline{0} \\ \text{(N6)}\ v \divideontimes u' = (v \divideontimes u) \# v\end{array}\right\}\text{乘法的递归定义.}$$

(N7) $P(\overline{0}) \to ((\forall x P(x) \to P(x')) \to \forall x P(x))$ 归纳公设

推理规则：同K.

构造K_N的主要目的是形式化数论的一个片断，即用K_N刻画该片断中的概念、命题、推理等。这个片断是一个正规K^+模型 $N=(\mathbb{N}, \mathbb{F}, \mathbb{P})$，$\mathbb{N}$为自然数集，$\mathbb{F}$包含自然数的后继函数、加法函数和乘法函数. $\mathbb{P}$包含自然数的相等关系，并且$\overline{0}^N$为$0 \in \mathbb{N}$，$\#^N$为$\mathbb{N}$上的$+$，$\divideontimes^N$为$\mathbb{N}$上的$\times$，$'^N$为$\mathbb{N}$上的后继(可理解为$+1$)

上述N是K_N的模型，N称为K_N的标准模型

简写：$\#$，$\divideontimes$分别简写为$+$，$\times$并约定$\overline{0}'$，$\overline{0}''$，…，$\overline{0}^{\overbrace{''}^{n个'}}$分别简写为$\overline{1}$，$\overline{2}$，…，$\overline{n}$，称为$K_N$的数字，公式$\neg(v \approx u)$简写为$v \not\approx u$

$\overline{n+m}$中加法是自然数里的，语义的加法. ∴有$\overline{4} = \overline{3+1} = \overline{2+2}$

$\overline{n}+\overline{m}$中加法是语法的，代表$\#$，$\overline{n}+\overline{m} \approx \overline{n+m}$

1101C-08 201412·2500

由 扫描全能王 扫描创建

中国科学技术大学

定理1 1° $\vdash_{K_N} \overline{m}+\overline{n}\approx\overline{m+n}$

2° $\vdash_{K_N} \overline{m}\times\overline{n}\approx\overline{m\times n}$

3° $\vdash_{K_N} \overline{0}+u\approx u$

4° $\vdash_{K_N} v'+u\approx(v+u)'$

5° $\vdash_{K_N} v+u\approx u+v$

6° $\vdash_{K_N} (w+u)+v\approx w+(u+v)$

定理2 1° $\vdash_{K_N} u_1+u_2\approx u_2\to u_1\approx\overline{0}$

2° $\vdash_{K_N} u_1+u_2\approx u_2\to u_1\approx\overline{0}$

3° $\vdash_{K_N} u\not\approx\overline{0}\to\overline{1}\leq u$ $u_1\leq u_2$ 定义为 $\exists x(x+u_1\approx u_2)$

4° $\vdash_{K_N} (u\not\approx\overline{0}\wedge\cdots\wedge u\not\approx\overline{n-1})\to\overline{n}\leq u$

5° $\vdash_{K_N} \neg(x\leq\overline{n})\to\overline{n}\leq x$, $n>0$

$m=n\Rightarrow\vdash_{K_N}\overline{m}\approx\overline{n}$; $m\neq n\Rightarrow\vdash_{K_N}\overline{m}\not\approx\overline{n}$

$$K\xrightarrow{\approx(E)}K^{+}\xrightarrow{\overline{0},{}',+,\times,N}K_N$$

k元函数: $f:\mathbb{N}^k\to\mathbb{N}$; k元关系: $R\subseteq\mathbb{N}^k$

可表示函数: k元函数f在K_N中可表示,如果存在含k+1个自由变元的公式 $p(x_1\cdots x_k x_{k+1})$ 使得: 对任意对 $p(x_1\cdots x_{k+1})$ 中的 x_{k+1} 自由的项u及 $n_1 n_2\cdots n_{k+1}\in\mathbb{N}$ 有

(i) $f(n_1,\cdots,n_k)=n_{k+1}\Rightarrow\vdash_{K_N}p(\overline{n_1},\overline{n_2},\cdots,\overline{n_k},\overline{n_{k+1}})$

(ii) $f(n_1,\cdots,n_k)\neq n_{k+1}\Rightarrow\vdash_{K_N}\neg p(\overline{n_1},\cdots,\overline{n_k},\overline{n_{k+1}})$

(iii) $\vdash_{K_N}p(\overline{n_1},\cdots,\overline{n_k},u)\to u\approx\overline{f(n_1,\cdots,n_k)}$

1101C-08 201412·2500

由 扫描全能王 扫描创建

中国科学技术大学

可表示关系：k元关系R在K_N中可表示，如果存在含k个自由变元的公式 $p(x_1,\cdots,x_k)$ 使对任意 $n_1,\cdots,n_k\in\mathbb{N}$ 有

(i) $(n_1,\cdots,n_k)\in R \Rightarrow \vdash_{K_N} p(\overline{n_1},\cdots,\overline{n_k})$

(ii) $(n_1,\cdots,n_k)\notin R \Rightarrow \vdash_{K_N} \neg p(\overline{n_1},\cdots,\overline{n_k})$

是否每个K_N公式都可用来表示一个数论函数？否. $x \approx x \wedge y \approx y$

同一个K_N公式是否可用于表示两个不同的k元函数？否

是否每个数论函数是可用K_N表示的？否

全体递归函数是K_N可表示的

可计算函数就是递归函数 ⇐ K_N可表示 ⇐ 图灵机

基本函数：以下三种函数称为基本函数

1° 一元零函数Z. $Z(n)=0$;

2° 一元后继函数S, $S(n)=n+1$;

3° k元投影函数P_i^k, $P_i^k(n_1,\cdots,n_k)=n_i$ $(i=1,\cdots,k)$

复合规则：一个i元函数g和i个k元函数$f_1,\cdots,f_i$的复合是一个k元函数.

$$l(n_1,\cdots,n_k)=_{def} g(f_1(n_1,\cdots,n_k),\cdots,f_i(n_1,\cdots,n_k))$$

递归规则：由k元函数g和k+2元函数f使用递归规则生成的k+1元函数l定义如下：

$$\begin{cases} l(n_1,\cdots,n_k,0)=_{def} g(n_1,\cdots,n_k) \\ l(n_1,\cdots,n_k,n+1)=_{def} f(n_1,\cdots,n_k,n,l(n_1,\cdots,n_k,n)) \end{cases}$$

有限时间机械步骤可算出

1101C-08 201412·2500

由 扫描全能王 扫描创建

μ算子：设k+1元函数满足根存在条件，对任意$n_1,\cdots,n_k$存在x使$g(n_1,\cdots,n_k,x)=0$。应用μ算子于g生成的函数f定义为：
$f(n_1,\cdots,n_k)=\min\{x \mid g(n_1,\cdots,n_k,x)=0\}$

递归函数：三个基本函数以及由它们经有限次应用三条规则生成的函数称为(一般)递归函数，不用μ算子生成的称为原始递归函数，不要求根存在条件的应用μ算子生成的称为部分递归函数

若k元关系R的特征函数$C_R(n_1,\cdots,n_k)=\begin{cases}1, & (n_1,\cdots,n_k)\in R\\ 0, & \text{otherwise}\end{cases}$是递归函数，则R为递归关系，一元递归关系称为递归集

所有递归函数及递归关系是K_N可表示的

所有K_N可表示的函数(关系)是递归函数(关系)

Godel编码：将所有公式和公式序列映射为自然数(唯一)

由 扫描全能王 扫描创建