

# 目次

## 第 1 章 論理学概説

	物理科学科 1 回生 吉田 伊吹	1
1.1	はじめに . . . . .	1
1.1.1	目標 . . . . .	1
1.1.2	論理学とは . . . . .	1
1.1.3	正しい論理とは . . . . .	1
1.2	命題論理 . . . . .	2
1.2.1	命題 . . . . .	2
1.2.2	記号化 . . . . .	2
1.2.3	複合命題の真理値 . . . . .	3
1.2.4	論理的真理 . . . . .	3
1.2.5	推論 . . . . .	4
1.3	述語論理 . . . . .	5
1.3.1	命題論理の限界 . . . . .	5
1.3.2	述語論理における記号化 . . . . .	6
1.3.3	命題の真理値 . . . . .	7
1.3.4	恒真命題 . . . . .	7
1.3.5	意味論と構文論 . . . . .	8
1.4	構文論 . . . . .	8
1.4.1	これからの議論展開 . . . . .	8
1.4.2	形式的体系 $\mathcal{K}$ . . . . .	9
1.4.3	補足説明 . . . . .	14
1.5	ここまでのまとめと補足的説明 . . . . .	15
1.6	理論の整理 . . . . .	16
1.7	形式的体系 $\mathcal{K}$ の健全性と完全性 . . . . .	20
1.7.1	証明すべきこと . . . . .	20
1.7.2	メタ論理 . . . . .	20

---

1.7.3	形式的体系 $\mathcal{K}$ の健全性 . . . . .	21
1.8	おわりに～数学の形式化は可能か～ . . . . .	22
	参考文献 . . . . .	23

# 第 1 章

## 論理学概説

物理科学科 1 回生

吉田 伊吹

### 1.1 はじめに

#### 1.1.1 目標

この著作は論理学を前提知識なしで、すなわち一から説明することを目標にしている。しかし易しすぎず難しすぎずかつ少ないページ数で論理学の大事な内容を伝えることを目標に書いた。よってかなり内容を絞ったため、本格的に論理学を学びたい方は他の著作物も読むことを勧める。このでは著者が論理学を学ぶ上でつまづいたことを丁寧に解説したので同じく論理学を学ぶ方がつまづくことなく理解できたなら嬉しい限りである。

#### 1.1.2 論理学とは

我々が何かを議論するとき、必ずと言っていいほど論理を使う。しかしはたしてその論理は正しいだろうか。そもそも正しい論理とはなにか、そしてそれは何で決まるのであろうか。それを考え、我々が正しく論理を扱えるように整理するのが論理学である。

#### 1.1.3 正しい論理とは

さっそく前節での問いに回答を与えるが、正しい論理は推論の正しきで決まる。推論とは前提から結論を導くことである。つまり論理学は推論についての学問である。では正しい推論はなにで決まるのだろうか。例えばある事件が起こったとして、太郎は犯人であり、太郎が犯人であるならば次郎も犯人であることがわかっているとする（前提）。これから次郎も犯人であることを結論づけたとする。これは正しい推論である（なぜ正しいか

は後述。)が、そこで推論の正しさを決定するのは太郎や次郎といったものではなく、 $\circ\circ$  と、 $\circ\circ$  ならば  $\times\times$  という前提から  $\times\times$  を結論づけるという推論の形式が重要であることがわかるだろう。様々な推論の例を考えて確かめるとよい。つまり、我々はこれから正しい推論の形式が何なのか、何によって決まるのかを議論するのである。

## 1.2 命題論理

この章では命題単位で議論する命題論理を解説する。

### 1.2.1 命題

命題とは、真か偽のいずれか一方がはっきり定まる文のことである。命題の真偽を真理値と言い、真を  $T$ 、偽を  $F$  で表す。以下にいくつかの命題を挙げる。

- ① 太郎は 100 点である。
- ② 太郎は 100 点ではない。(太郎は 100 点である、ではない。)
- ③ 太郎と花子が 100 点である。(太郎が 100 点である、かつ花子が 100 点である。)
- ④ 太郎か花子が 100 点である。(太郎が 100 点である、または花子が 100 点である。)
- ⑤ 太郎が 100 点ならば花子も 100 点である。(太郎が 100 点である、ならば花子が 100 点である。)
- ⑥ 太郎と花子が 100 点ならば次郎は 100 点ではない。

これらの命題を観察すると①のようにこれ以上分割できない命題と②～⑥のようにより小さい命題と否定詞(ない)、接続詞(かつ、または、ならば)によって構成される命題があることがわかるだろう。前者を原子命題、後者を複合命題という。

### 1.2.2 記号化

これからの議論において、我々は個々の命題の内容については興味がない。というのも我々の目的は正しい推論の形式を見つけ出すことにあるからだ。よって、命題を記号化する。こうして議論を進めることで命題一般について正しい推論の形式をみつけることができる。そして記号として得られた正しい推論に、どんな具体的な命題を代入しようとも成り立つというわけである。

次のように命題を記号化する。

- 原子命題  $\dots A, B, C, \dots$
- 否定詞  $\dots \neg(\text{ない})$
- 接続詞  $\dots \wedge(\text{かつ}), \vee(\text{または}), \rightarrow(\text{ならば})$

誤解を生まないよう複雑な命題には括弧を用いるが、命題と否定詞・接続詞の結合力を  $\neg > \wedge = \vee > \rightarrow$  と定めることで、煩雑な括弧を省略する。

例)  $((\neg A) \wedge (\neg B)) \rightarrow (A \rightarrow B)$  は  $\neg A \wedge \neg B \rightarrow (A \rightarrow B)$  と括弧を省略できる。

命題①～⑥を記号化すると次のようになる。

A… 太郎は 100 点である., B… 花子は 100 点である., C… 次郎は 100 点である.  
とすると,

- ① A
- ②  $\neg A$
- ③  $A \wedge B$
- ④  $A \vee B$
- ⑤  $A \rightarrow B$
- ⑥  $(A \wedge B) \rightarrow (\neg C)$  結合力の定義から、括弧はなくてもよい。

### 1.2.3 複合命題の真理値

命題  $\varphi, \psi$  の真理値に対して命題  $\neg\varphi, \varphi \wedge \psi, \varphi \vee \psi, \varphi \rightarrow \psi$  の真理値を次のように定義する。

$\varphi$	$\neg\varphi$
T	F
F	T

$\varphi$	$\psi$	$\varphi \wedge \psi$	$\varphi \vee \psi$	$\varphi \rightarrow \psi$
T	T	T	T	T
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

どんなに複雑な複合命題でもそれを構成する原子命題の真理値からスタートして、上の定義から真理値を定めることができる。

これらの命題は意味を持った我々の世界の命題を記号で置き換えたものに過ぎない。したがって真理値の定義もくそもかつやまたはの意味からこうなるのは当然じゃないかと思うかもしれない。しかし、かつやまたはとはどういう意味だろうか、その意味を知らない人に説明できるだろうか。なにが言いたいのかというと、この真理値の定義によって否定詞と接続しの意味を改めて規定したということである。

### 1.2.4 論理的真理

複合命題を構成する原子命題に真理値を割り当てることを真理値割り当てと言う。各割り当てに対して複合命題の真理値がどうなるかを示した表を真理表と言う。複合命題を構

成する原子命題が  $n$  個のときこれらへの真理値のわりあて方は  $2^n$  通りあり、真理表も  $2^n$  行となる。複合命題の中にはいかなる真理値わりあてに対しても真となる命題があり、これをトートロジーと言う。

例えば  $A \rightarrow (B \rightarrow A)$  という命題の真理表をかくと下のようになる。

表 1.1 真理表

A	B	$B \rightarrow A$	$A \rightarrow (B \rightarrow A)$
T	T	T	T
T	F	T	T
F	T	F	T
F	F	T	T

この命題は真理表のすべての行で真となっておりトートロジーである。

### 1.2.5 推論

いよいよ推論について議論しよう。繰り返しになるが推論とは前提から結論を導くことであり、論理学はこの推論を扱う学問に他ならない。

次の二つの推論において、㉑は正しい推論の形式であり、㉒は誤った推論の形式である。

㉑	前提	$A \rightarrow B$		㉒	前提	$A \rightarrow B$
		$A$				$\neg A$
	結論	$B$			結論	$\neg B$

では、正しい推論とはこういったものを言うのだろうか。推論とは、前提が全て正しいならばこれこれと言えるということである。つまり正しい推論とは前提が全て真ならば結論も必ず真となるような推論である。逆に言えば、前提が全て真であるにもかかわらず、結論が偽になることはないということであり、これは前提を  $\varphi_1, \varphi_2, \dots, \varphi_n$  結論を  $\psi$  としたとき、命題  $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \psi$  がトートロジーになることに他ならない。

実際に推論㉑、㉒が正しいかどうかを真理表をかいて確かめると以下のようになる。

表 1.2 ㊦

A	B	$A \rightarrow B$	$(A \rightarrow B) \wedge A$	$((A \rightarrow B) \wedge A) \rightarrow B$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

表 1.3 ㊧

A	B	$A \rightarrow B$	$\neg A$	$(A \rightarrow B) \wedge \neg A$	$\neg B$	$((A \rightarrow B) \wedge \neg A) \rightarrow \neg B$
T	T	T	F	F	F	T
T	F	F	F	F	T	T
F	T	T	T	T	F	F
F	F	T	T	T	T	T

したがってやはり㊦は正しい推論であり，㊧は誤った推論であることがわかる。

## 1.3 述語論理

### 1.3.1 命題論理の限界

前章で正しい推論がいかなるものであるか，またその判定方法がわかった．そこで数学に応用してみよう．

前章の推論㊦は正しい推論であった．これの A, B を

A ... 三角形の 2 角が等しい．

B ... 二等辺三角形である．

とすると，正しい二つの前提から，正しい結論を導くことに成功している．なかなかいい感じである．では，次の推論についてその正しさを判定してみよう．

前提 任意の  $x, y$  について， $x \cdot y > 0$

結論  $2 \cdot 3 > 0$

前提  $1 = 1$

結論  $n$  が存在して,  $n = n$

命題論理では命題単位でしか記号化できないため, これらの推論を記号化すると前提 A, 結論 B となる. はたしてこの推論の形式は正しいだろうか. これを判定するには  $A \rightarrow B$  の真理値表をかいいてこれがトートロジーになるかをみればよかった. これは明らかにトートロジーにならない. この推論は正しいはずなのだが, 命題論理ではうまく扱えない. そこでこれから命題の細部にまで立ち入って記号化することを試みる. これが述語論理である. 述語論理は命題論理の中に含むより表現が豊かな体系であり, この論理によって数学における推論を十分に行うことができる.

### 1.3.2 述語論理における記号化

述語論理においては命題を次のように記号化する.

項 ...

ある決まった対象を  $a, b$  などと記号化し, これを定項という. また, 不定の対象を  $x, y$  などと記号化し, これを変項という.

関数 ...

足し算や累乗などの関数 (数学的なものでなくとも一般に一つまたは複数の対象から一つの対象を定めるもの) を  $f, g$  などと記号化し,  $x+y, 3^x$ などを  $f(x, y), g(x)$  と表す.

述語 ...

$\circ\circ$  は整数である,  $\circ\circ$  は  $\times\times$  より大きい, など対象の性質や対象同士の関係を述語といい  $F, G$  などと記号化する. さらに  $x$  は整数である,  $x$  は  $y$  より大きい, を  $F(x), G(x, y)$  と記号化する. これらは  $x, y$  に具体的な対象 (定項) を代入することで命題となり真偽が確定する.

量化 ...

任意の  $x$  に対して  $F(x)$  を  $\forall xF(x)$ ,  $x$  が存在して  $F(x)$  を  $\exists xF(x)$  と記号化する. ここで  $\forall xF(x), \exists xF(x)$  は真偽が確定するので, 命題である. また, 任意の  $x$  に対して  $y$  が存在して  $G(x, y)$  は  $\forall x\exists yG(x, y)$  と記号化できる. これを多重量化という. これも真偽が決まるので命題であるが,  $\exists yG(x, y)$  は命題ではない. このとき,  $y$  を束縛変項といい,  $x$  を自由変項という. 束縛変項は実質的には変項ではない.  $\exists x G(x, x)$  の  $x$  にどんな変項記号 ( $x$  を除く) が入っても意味は変わらないからである.



それでは前述の推論に現れた命題，任意の  $x, y$  について， $x \cdot y > 0$  を記号化してみよう．まず定数は 0 であり，これを定項記号  $a$  で記号化する．このなかで関数は  $\cdot$  であり，これを  $f$  と記号化する．また述語は  $>$  であり，これを  $G$  と記号化すると，以下ようになる．

$$\forall x \forall y G(f(x, y), a)$$

### 1.3.3 命題の真理値

前章で一つの命題を記号化したわけであるが，我々は記号化によって命題一般について成り立つ形式を探りたいのだ．したがってこのように記号化される命題一般について議論を進めることになる．そこでまず，前章で記号化した命題の真理値がどのようなになるか見ていこう．

$$\forall x \forall y G(f(x, y), a)$$

この命題の真理値を決定するには次の二つを定めなければならない．

- 対象領域
- 定項，関数，述語の解釈

この二つを命題の解釈という．例えば 2 以上の自然数を対象領域とし，定項  $a$  を 4，関数  $f$  を  $+$ ，述語  $G$  を  $>$  と解釈すると，この命題は真になる．また，整数を対象領域とし，定項  $a$  を 0，関数  $f$  を  $\cdot$ ，述語  $G$  を  $>$  と解釈すると，この命題は偽になる．

命題論理における真理値割り当てについてもう一度考えてみよう．真理値割り当てとは複合命題を構成する原子命題に真理値を割り当てることであった．これは次のように考えることができる．各原子命題記号に具体的な命題（解釈）を与えた結果，原子命題の真偽が決まり（これが真理値割り当て），よって複合命題の真偽が決まった．このように考えると真理値割り当ては，命題論理における解釈といえるのである．

### 1.3.4 恒真命題

命題論理において，いかなる真理値割り当て（解釈）においても真となる命題をトートロジーといった．述語論理においても任意の解釈のもとで真となる命題が存在し，これを恒真命題という．ではこれをどう判定すればよいだろう．命題論理のときは真理値表をかくて，全ての行で真となるかを見ればよかった．しかし，述語論理においてはこのように機械的に判定する方法（決定手続きという）が存在しないことが証明されている．しかし一般的な機械的方法が存在しないというだけで個別に判定することはできる．

### 1.3.5 意味論と構文論

命題論理では推論の正しさをトートロジーによって判定した。しかし述語論理においては恒真命題の決定手続きが存在しない。よって今までの方法では、述語論理の推論の正しさをすぐに判定できない。そこでこれから先に議論を進めるには命題の真偽によって推論の正しさを判定する意味論から脱却し、前提となる論理式（命題をただの記号列とみたもの、後述）から正しい論理式を導く規則について考える構文論の方へと進まなければならない。構文論では論理式を全く意味を考えない体系によって扱うことで正しい推論を得ることを試みる。つまり推論の形式化をめざすのである。

- ※注意 1. 説明の都合上正しいという言葉を使ったが、構文論では意味抜きされた体系を扱うため、そこに真偽は考えない。
2. 構文論は意味論あつてのものであり、構文論だけで論理学が完結するわけではない。（後にわかるだろう。）

## 1.4 構文論

### 1.4.1 これからの議論展開

これからどのように議論を展開するか、あらかじめざっと説明しておこう。まず、目的をはっきりさせておこう。我々の目的は推論を形式化することである。そこで形式的体系というものを作る。形式的体系は（記号、論理式、公理、推論規則）からなり、この体系を用いて、推論を形式化することを考える。まず、命題から意味を抜き去り、ただの記号列（これを論理式という）として扱う。そして前提となる論理式に推論規則と呼ばれるルールを適用することで新しい論理式を導き、さらにこれまでに導かれた論理式に対して推論規則を適用することで新しい論理式を導く、といったように前提となる論理式から推論規則によって次々に論理式を導く。そうして最終的に得られた論理式（結論）と前提の論理式を解釈（論理式の解釈は前章で行った記号化された命題の解釈と同様である。）によって意味づけをしたとき、前提が全て真ならば結論も真となっていればこの推論の形式化は成功したことになる。このように意味を考えずとも正しい推論を得られるような形式的体系を作るのである。すなわちこれからの議論は以下のように進める。

- ① まず、形式的体系をつくる。（ここでは一つの体系を提示するのみ）。
- ② その形式的体系がうまくいっているか確かめる。

### 1.4.2 形式的体系 $\mathcal{K}$

これから形式的体系  $\mathcal{K}$  を定義する。形式的体系は（記号，論理式，公理，推論規則）からなる。このうち記号と論理式についてはすでに前章での命題の記号化を用いればよいのだが，ここでは意味を全く考えない形式を扱うのである。したがって使用する記号，また記号列のうち何が論理式で何が論理式でないかを厳格に定めなければならない。

#### $\mathcal{K}$ の記号，項，論理式

使用する記号を定め，論理式を次のように定義する。（論理式の定義に先立って項を定義しておく。）

- 記号
  - 定項記号  $a, b, c, a_1, a_2, \dots$
  - 変項記号  $x, y, z, x_1, x_2, \dots$
  - 関数記号  $f, g, \dots$ （各関数記号には引数の数が決まっている。）
  - 述語記号  $F, G, \dots$ （各述語記号には引数の数が決まっている。）
  - 論理記号  $\neg, \wedge, \vee, \rightarrow, \forall, \exists, \perp$
- 項
  - 定項記号は項である。
  - 変項記号は項である。
  - $t_1, t_2, \dots, t_n$  が項で， $f$  が  $n$  項関数記号ならば， $f(t_1, t_2, \dots, t_n)$  は項である。
  - このようにして項だとわかったものだけが項である。
- 論理式
  - $t_1, t_2, \dots, t_n$  が項で， $F$  が  $n$  項述語記号ならば， $F(t_1, t_2, \dots, t_n)$  は論理式である。
  - $\varphi$  が論理式ならば  $\neg(\varphi)$  は論理式である。
  - $\varphi, \psi$  が論理式ならば  $(\varphi) \wedge (\psi)$ ,  $(\varphi) \vee (\psi)$ ,  $(\varphi) \rightarrow (\psi)$  は論理式である。
  - $x$  が変項記号， $\varphi$  が論理式ならば  $\forall x(\varphi)$ ,  $\exists x(\varphi)$  は論理式である。
  - $\perp$  は論理式である。
  - このようにして論理式だとわかったものだけが論理式である。
  - 1 章の定義のように煩雑な括弧は省略する。

ここで新しく  $\perp$  という記号を導入したが，これは矛盾を意図した記号であり，いかなる解釈のもとでも偽となる。項と論理式について一つ例を挙げておこう。定項記号  $a$ ，変数記号  $x$  は項であり， $f$  を 2 項関数記号とすると  $f(a, x)$  は項であり，次に変項記号  $y$ ，2 項述語記号  $G$  を用いると， $G(y, f(a, x))$  は論理式である。さらに論理記号  $\forall, \perp$ ，

$\rightarrow$  を用いると,  $\forall xG(y, f(a, x))$  は論理式であり,  $\perp$  はそれ自体が論理式であるから,  $\perp \rightarrow \forall xG(y, f(a, x))$  は論理式である. また, このとき前章での定義と同様に  $x$  を束縛変項といい,  $y$  を自由変項という. この論理式の  $y$  を束縛して,  $\forall y(\perp \rightarrow \forall xG(y, f(a, x)))$  とすると, この論理式には自由変項がない. このような論理式を閉論理式という.

### K の推論規則, 証明

次に形式的体系  $K$  の推論規則を定義し, 続いて  $K$  における証明を定義したいのだが説明の都合上, 先に形式的体系  $K$  における証明の一例を挙げることにする. ここで証明と推論規則について軽く説明し, 感覚的に理解してもらった後で厳密な定義を与える.

表 1.4 形式的体系  $K$  による証明のサンプル

(1)	$\forall x(F(x) \rightarrow H(x)) \wedge \forall x(G(x) \rightarrow H(x))$	前提
(2)	$\forall x(F(x) \rightarrow H(x))$	1, $\wedge_-$
(3)	$\forall x(G(x) \rightarrow H(x))$	1, $\wedge_-$
(4)	$F(a) \rightarrow H(a)$	1, $\forall_-$
(5)	$G(a) \rightarrow H(a)$	1, $\forall_-$
(6)	$F(a) \vee G(a)$	[6(a)] 仮定
(7)	$F(a)$	[7(a)] 仮定
(8)	$H(a)$	[7(a)] 4, 7, $\rightarrow_-$
(9)	$G(a)$	[9(a)] 仮定
(10)	$H(a)$	[9(a)] 5, 9, $\rightarrow_-$
(11)	$H(a)$	[6(a)] 6, 8, 10, $\vee_-$
(12)	$(F(a) \vee G(a)) \rightarrow H(a)$	6, 11, $\rightarrow_+$
(13)	$\forall x((F(x) \vee G(x)) \rightarrow H(x))$	12, $\forall_+$

これは  $\forall x(F(x) \rightarrow H(x)) \wedge \forall x(G(x) \rightarrow H(x))$  を前提とする  $\forall x((F(x) \vee G(x)) \rightarrow H(x))$  の証明である. この証明の見かたについて説明する. まず左の列から行番号, 導かれた論理式, その論理式が依存する仮定 (が書かれた行番号) を示しており, 最後にどの行とどの行にどんな推論規則を用いたかを書いている. 例えば, 8 行目は 4 行目と 7 行目に  $\rightarrow_-$  という推論規則を用いて導かれ, 依存する仮定は 7 行目といった具合である. ここで依存する論理式の行番号 7 のとなりに (a) と書いてあるが, これは  $F$  が成り立つ限りの  $a$  ということであり  $a$  を限定つきの項という.

さて, 形式的体系  $K$  における推論規則を定義しよう. ここで  $\Gamma$  や  $\Delta$  は論理式の集合である. (証明のサンプルでは [ ] 内に依存する仮定が書かれた行番号を書いているが, ここではその論理式を直接示している.)

- $\mathcal{K}$  の推論規則

–  $\wedge_+$

$$\begin{array}{lll} (l) & \varphi & [\Gamma] \\ (m) & \psi & [\Delta] \\ (n) & \varphi \wedge \psi & [\Gamma, \Delta] \quad l, m, \wedge_+ \end{array}$$

ただし  $l, m < n$

–  $\wedge_-$

$$\begin{array}{lll} (l) & \varphi \wedge \psi & [\Gamma] \\ (n) & \varphi(\psi) & [\Gamma] \quad l, \wedge_- \end{array}$$

ただし  $l < n$

–  $\vee_+$

$$\begin{array}{lll} (l) & \varphi & [\Gamma] \\ (n) & \varphi \vee \psi(\psi \vee \varphi) & [\Gamma] \quad l, \vee_+ \end{array}$$

ただし  $l < n$

–  $\vee_-$

$$\begin{array}{lll} (l) & \varphi \vee \psi & [\Gamma] \\ (m_1) & \rho & [\Delta_1, \varphi] \\ (m_2) & \rho & [\Delta_2, \psi] \\ (n) & \rho & [\Gamma, \Delta_1, \Delta_2] \quad l, m_1, m_2, \vee_- \end{array}$$

ただし  $l, m_1, m_2 < n$

–  $\rightarrow_+$

$$\begin{array}{lll} (l) & \varphi & [\Gamma, \psi] \\ (n) & \psi \rightarrow \varphi & [\Gamma] \quad l, \rightarrow_+ \end{array}$$

ただし  $l < n$

–  $\rightarrow_-$

$$\begin{array}{lll}
 (l) & \varphi & [\Gamma] \\
 (m) & \varphi \rightarrow \psi & [\Delta] \\
 (n) & \psi & [\Gamma, \Delta] \quad l, m, \rightarrow -
 \end{array}$$

ただし  $l, m < n$

–  $\neg_+$

$$\begin{array}{lll}
 (l) & \perp & [\Gamma, \varphi] \\
 (n) & \neg\varphi & [\Gamma] \quad l, \neg_+
 \end{array}$$

ただし  $l < n$

–  $\neg_-$

$$\begin{array}{lll}
 (l) & \varphi & [\Gamma] \\
 (m) & \neg\varphi & [\Delta] \\
 (n) & \perp & [\Gamma, \Delta] \quad l, m, \neg_-
 \end{array}$$

ただし  $l, m < n$

– 背理法

$$\begin{array}{lll}
 (l) & \perp & [\Gamma, \neg\varphi] \\
 (n) & \varphi & [\Gamma] \quad l, \text{背理法}
 \end{array}$$

ただし  $l < n$

– 矛盾

$$\begin{array}{lll}
 (l) & \perp & [\Gamma] \\
 (n) & \varphi & [\Gamma] \quad l, \text{矛盾}
 \end{array}$$

ただし  $l < n$

–  $\forall_+$

$$\begin{array}{lll}
 (l) & \varphi(\dots t \dots) & [\Gamma] \\
 (n) & \forall x \varphi(\dots x \dots) & [\Gamma] \quad l, \forall_+
 \end{array}$$

ただし  $t$  は限定なしの項（すなわち  $\Gamma$  中に自由変項として現れない）とし、 $x$  は  $\varphi(\dots t \dots)$  中に含まれていない変項とする。

ただし  $l < n$

–  $\forall_-$

$$\begin{array}{ll} (l) & \forall x\varphi(\dots x \dots) \quad [\Gamma] \\ (n) & \varphi(\dots t \dots) \quad [\Gamma] \quad l, \forall_- \end{array}$$

ただし  $t$  中の自由変項は束縛されないとする.

ただし  $l < n$

–  $\exists_+$

$$\begin{array}{ll} (l) & \varphi(\dots t \dots) \quad [\Gamma] \\ (n) & \exists x\varphi(\dots x \dots) \quad [\Gamma] \quad l, \exists_+ \end{array}$$

ただし  $l < n$

–  $\exists_-$

$$\begin{array}{lll} (l) & \exists x\varphi(\dots x \dots) & [\Gamma] \\ (m) & \rho & [\Delta, \varphi(\dots t \dots)] \\ (n) & \rho & [\Gamma, \Delta] \quad l, m, \exists_- \end{array}$$

ただし  $t$  は  $\Delta, \rho, \exists x\varphi(\dots x \dots)$  に自由変項として含まれないとする.

ただし  $l, m < n$

以上が形式的体系  $\mathcal{K}$  の推論規則である. 仮定は基本的に導かれる論理式に引き継がれるが規則によっては解消される仮定もある. また, 推論規則によって解消可能な仮定は別に解消しなくてもよい. これらの規則のうち  $\wedge_+, \wedge_-, \vee_+, \rightarrow_-, \neg_-, \forall_-, \exists_+$  は感覚的に明らかだろう. よって他の推論規則について説明する.

- $\vee_-$

$\varphi \vee \psi$  がわかっていて,  $\varphi$  を仮定しても  $\psi$  を仮定しても  $\rho$  が導かれるならば結局はそれらの仮定なしに  $\rho$  が導けるということである.

- $\rightarrow_+$

$\psi$  を仮定して  $\varphi$  が導かれるならば  $\psi$  の仮定なしに  $\psi \rightarrow \varphi$  を導いてよいということである.

- $\neg_+$ , 背理法

$\varphi$  を仮定して矛盾が導かれるならば  $\neg\varphi$  を結論してよく,  $\neg\varphi$  を仮定して矛盾が導かれるならば  $\varphi$  を結論してよいということである.

- 矛盾

$\perp \rightarrow \varphi$  は任意の解釈のもとで真であるから、仮定  $\Gamma$  のもとで矛盾がでるなら、この仮定のもとではいかなる論理式でも導いてよいということである。

- $\forall_+$

一見ひとつの項で成り立つことがらを勝手に拡大しているように見えるが、 $t$  は限定なしの項であり、すなわち  $t$  は全くの任意の項であるからこのようなことが許されるのである。証明のサンプルにおいてこの推論規則を用いている。この証明では  $F(a) \vee G(a)$  を仮定し、この時点では  $a$  は制限ありの項であるが、後に推論規則  $\rightarrow_+$  によって仮定が解消され、 $a$  の制限がなくなり、よってこの推論規則  $\forall_+$  が適用できたのである。

- $\exists_-$

この推論規則は次のように考えるとよい。まず、 $\varphi(\dots x \dots)$  を満たすような項の存在が保証されている。そしてその項を  $t$  とし推論を進めた結果、 $t$  を含まない結論  $\rho$  が導けたとする。よって  $t$  が何だったかに関係なく  $\rho$  が導けるということである。

ここまでを踏まえて、もう一度証明のサンプルを見るとよく理解できるだろう。最後に流れる的には逆になるが形式的体系一般についてその証明を定義する。前に挙げた  $\mathcal{K}$  における証明がこの定義に沿っていることを確認してほしい。

定義 形式的体系（記号、論理式、公理、推論規則）における  $\Gamma$  を仮定とする  $\varphi$  の証明とは次を満たす論理式の列である。

- 論理式列の最後は  $\varphi$  である。
- 各論理式は公理であるか、仮定であるか、それ以前の論理式から推論規則によって導かれたものである。
- 解消されずに残った仮定の集合を  $\Gamma'$  とすると  $\Gamma' \subset \Gamma$  である。

$\Gamma$  が空集合であるとき、 $\varphi$  をその形式的体系の定理という。形式的体系  $\mathcal{K}$  には公理はないが、公理を設ける形式的体系も存在する。 $\mathcal{K}$  には公理がないため、 $\mathcal{K}$  における証明の第1行目は必ず仮定になる。

### 1.4.3 補足説明

#### 前提, 仮定

形式的体系  $\mathcal{K}$  における証明の第1行目は必ず仮定になると述べた。しかし、証明のサンプルでは第1行目は前提となっている。これはどういうことか。言ってしまうと前提とは仮定である。実は前提と仮定には本質的な違いはないのである。証明のサンプルで1行目を前提として、仮定 [1] と書かなかったのは、この仮定は初めから解消するつもりがな



く、[1] とかくと証明が見つらなくなるからである。この辺りは好みの問題であって、もちろん全てを仮定としてかいてもよい。いずれにせよ証明の仮定  $\Gamma$  は前提としておいたものと最後まで仮定として残ったものの両方となる（普通はどちらかに統一するが）。

### 公理

まだ公理の説明をしていなかった。形式的体系における公理とは論理式のうちいかなる解釈のもとで真となるものをいくつか集めてきたものである。公理を設ける形式的体系では推論規則が少なく ( $\rightarrow$  のみとか)、仮定を途中で解消できない体系が多い。それに対し形式的体系  $K$  では公理を設けず、多くの推論規則によって証明を行う。ここで述べたような公理は論理的公理である。公理と呼ばれるものにはほかに数学的公理がある。群の公理はその一例である。数学における公理は前提（つまり仮定）のことをさす。ということかという数学では、ある性質を満たすような数学的対象について議論をする。その際議論において、(例えば) 群の公理（結合則、単位元の存在、逆元の存在）を満たすならば  $\infty$ 、群の公理を満たすならば  $\times \times$  と一々言うのはめんどろだし、今、群について議論しているのだから群の公理を満たしているのは言うまでもないのだ。そこで今議論している数学的対象で前提としている性質を公理として形式的体系のなかに含めてしまうと便利である。または逆に仮定  $\Gamma$  で  $\varphi$  が証明できたなら、公理系  $\Gamma$  のもとで  $\varphi$  は証明できるということもできる。すなわち前節で述べた前提、仮定と数学的公理には本質的な違いはない。これら呼び分けるのはそれぞれの人の気持ちの違いでしかない。

### 注意

これまで、形式的体系の説明をし、形式的体系  $K$  を作った。形式的体系とは形式のみを考え、その真偽は考えないのだった。しかし形式的体系  $K$  の推論規則の説明や公理の説明において真や偽という言葉を使った。これは次に議論する内容、これからの議論展開の②番、"作った形式的体系がうまくいっているか確かめる"の内容に立ち入ったものである。我々の目的は推論を形式化することで、そのために形式的体系を作った。そしてその形式的体系がうまくいっているか、すなわち形式的体系によって得られた、仮定（前提） $\Gamma$  における  $\varphi$  の証明をどのように解釈しても、前提が全て真ならば結論も真であるかを確認しようというわけである。そこで推論規則や公理の説明に際して、その妥当性を感じてもらうために先にこの内容に立ち入り、真偽を用いた次第である。混乱を招いたなら申し訳ない。

## 1.5 ここまでのまとめと補足的説明

これまでの議論を振り返ってみよう。我々は正しい論理とは何かを考え、それは推論の正しさによることがわかった。そして正しい推論とはなにかを考えるため、まず命題単位

で議論する命題論理をみた。正しい推論はその形式で決まるため、その形式のみを抽出するために（日常言語の）命題の記号化を行った。この時点で記号化された命題は特定の意味を失い、同様の形で記号化される命題一般について議論できるようになった。しかしこれを用いて推論を議論するときにはそれらを解釈（真理値割り当て）することで正しい推論を規定した。解釈をするということは記号に意味づけをするということにほかならない。しかしここで意味を考えるのは至極当然である。命題に関してはそれが具体的に何であるかはどうでもよいので記号化できた。しかし、我々が意味の世界に生きている以上推論の正しさまでは意味抜きで規定できないのである。次に、命題単位の議論では数学で行うような複雑な推論を扱えないということで命題論理を内に含むより表現の豊かな体系として述語論理を展開した。そこでは命題をより細部にわたって記号化した。しかし述語論理では正しい推論の形式の判定において機械的な手続き（決定手続き）が存在しなかった。そこでこれに対する有効な手段である構文論へと進んだ。構文論では意味を考えずとも正しい推論が得られるような体系によって、推論を形式化することを試みた。そこで①形式的体系を作り、②その体系がうまくいっているか確かめるという二つのステップを踏むことにした。しかし少し前、推論の正しさは意味抜きでは規定できないといった。これに関してはステップ②が担っており、これが構文論と意味論を結びつけている。これは前章のこれからの議論展開や注意で述べたことを思い出せばわかるだろう。

## 1.6 理論の整理

前章でこれまでの議論を振り返ったが少々議論が込み入っている。これはいちから理解できるように話をするためには仕方がないことである。しかし今、我々はすでにこれだけのことを理解している。そこで議論をもう一度いちから今度はきれいにまとめた理論を展開しよう。この章はこれまでの議論を整理しきれいな形にまとめることを目標にしており、内容的に進むわけではないので読み飛ばして次の章に進んでもらってかまわない。

まず形式的体系とその証明を定義する。

定義 形式的体系とは以下の要素からなるものである。

- － 記号
- － 論理式（記号列のうち論理式であると定めたもの）
- － 公理（論理式のうち公理として定めたもの）
- － 推論規則（論理式から他の論理式を導く規則）

定義 形式的体系における  $\Gamma$  を仮定とする  $\varphi$  の証明とは以下を満たす論理式列である。（ただし  $\Gamma$  は論理式の集合  $\varphi$  は論理式である。）

- － 論理式列の最後は  $\varphi$  である。
- － 各論理式は公理であるか、仮定であるか、それ以前の論理式から推論規則に

よって導かれたものである.

- 解消されずに残った仮定の集合を  $\Gamma'$  とすると  $\Gamma' \subset \Gamma$  である.

次に形式的体系  $\mathcal{K}$  を導入する.

- 形式的体系  $\mathcal{K}$  (詳しくは前章参照)
  - 記号 定項記号, 変項記号, 関数記号, 述語記号, 論理記号, 補助記号
  - 論理式 定義は前章参照
  - 公理 なし
  - 推論規則  $\wedge_+, \wedge_-, \vee_+, \vee_-, \rightarrow_+, \rightarrow_-, \neg_+, \neg_-$ , 背理法, 矛盾,  $\forall_+, \forall_-$ ,  $\exists_+, \exists_-$

ここで形式的体系  $\mathcal{K}$  において, 仮定  $\Gamma$  から  $\varphi$  が証明できることを  $\Gamma \vdash \varphi$  とかく. 次に  $\mathcal{K}$  の論理式の解釈を定義する. 解釈  $I$  はストラクチャー  $M$  と割り当て  $g$  からなる.

- 解釈  $I$ 
  - ストラクチャー  $M = [D; \rho; \sigma; \tau]$ 
    - \* 対象領域 (存在物の集合)  $D$
    - \* 定項記号の集合から  $D$  への写像  $\rho$
    - \*  $n$  項関数記号の集合から  $D^{(D^n)}$  への写像  $\sigma$
    - \*  $n$  項述語記号の集合から  $B^{(D^n)}$  への写像  $\tau$
  - ここで  $D^n$  は直積集合 ( $D$  の元  $n$  組の集合),  $D^{(D^n)}$  は  $D^n$  から  $D$  への写像全体の集合,  $B^{(D^n)}$  は  $D^n$  から真理値集合  $B = \{T, F\}$  への写像全体の集合である.
  - 割り当て  $g$ 
    - \* 変数記号の集合から  $D$  への写像  $g$

この定義はこれからの議論であまり重要ではないので, わからなくてもよい. 前章で行った述語論理の命題の解釈と同じものだと考えてよい. ただし前章では閉論理式 (に相当する命題) のみを考えたがここでは閉論理式でない論理式も扱うので変項記号への  $D$  の要素の割り当て  $g$  を加えた. 次に論理式の解釈の仕方を定義する.

- 基本解釈
  - 定項記号  $I(a) = \rho(a)$
  - 変項記号  $I(x) = g(a)$
  - $n$  項関数記号  $I(f) = \sigma(f)$
  - $n$  項述語記号  $I(F) = \tau(F)$
  - 論理記号
    - \*  $I(\neg) = [T \mapsto F, F \mapsto T] \in B^B$

- \*  $I(\wedge) = [(T, T) \mapsto T, (T, F) \mapsto F, (F, T) \mapsto F, (F, F) \mapsto F] \in B^{(B^2)}$
- \*  $I(\vee) = [(T, T) \mapsto T, (T, F) \mapsto T, (F, T) \mapsto T, (F, F) \mapsto F] \in B^{(B^2)}$
- \*  $I(\rightarrow) = [(T, T) \mapsto T, (T, F) \mapsto F, (F, T) \mapsto T, (F, F) \mapsto T] \in B^{(B^2)}$
- \*  $I(\perp) = F \in B$
- \*  $I(\forall x(\varphi)) = T \Leftrightarrow \text{すべての } \alpha \in D \text{ に対して } I'(\varphi) = T$
- \*  $I(\exists x(\varphi)) = T \Leftrightarrow I'(\varphi) = T \text{ となる } \alpha \in D \text{ が存在する.}$

ただし  $I'$  は解釈  $I$  の割り当て  $g$  の変項  $x$  への割り当てを  $\alpha$  としたもの.

● 項, 論理式の解釈

- $I(f(t_1, t_2, \dots, t_n)) = I(f)(I(t_1), I(t_2), \dots, I(t_n))$
- $I(F(t_1, t_2, \dots, t_n)) = I(F)(I(t_1), I(t_2), \dots, I(t_n))$
- $I(\neg\varphi) = I(\neg)(I(\varphi))$
- $I(\varphi \wedge (\vee, \rightarrow)\psi) = I(\wedge(\vee, \rightarrow))(I(\varphi), I(\psi))$

ただし,  $t_1, t_2, \dots, t_n$  は項,  $f$  は  $n$  項関数記号,  $F$  は  $n$  項述語記号,  $\varphi, \psi$  は論理式である.

ここで一例を挙げる. 証明のサンプルの一行目を解釈してみよう. 次のような解釈  $I$  を考える.

● 解釈  $I$

– ストラクチャー  $M$

- \*  $D = \{1, 2, 3, 4, 5\}$
- \*  $\rho = [a \mapsto 1, b \mapsto 2, c \mapsto 3, d \mapsto 4, e \mapsto 5, \dots]$
- \*  $\sigma = [$   
 $f \mapsto [1 \mapsto 2, 2 \mapsto 3, 3 \mapsto 4, 4 \mapsto 5, 5 \mapsto 1],$   
 $g \mapsto [(1, 1) \mapsto 0, (1, 2) \mapsto 1, \dots, (5, 5) \mapsto 0],$   
 $\dots]$
- \*  $\tau = [$   
 $F \mapsto [1 \mapsto T, 2 \mapsto T, 3 \mapsto F, 4 \mapsto F, 5 \mapsto F],$   
 $G \mapsto [1 \mapsto T, 2 \mapsto T, 3 \mapsto T, 4 \mapsto F, 5 \mapsto F],$   
 $H \mapsto [1 \mapsto T, 2 \mapsto T, 3 \mapsto T, 4 \mapsto T, 5 \mapsto F]$   
 $]$

– 割り当て  $g$

- \*  $g = [x \mapsto 1, y \mapsto 2, z \mapsto 3, \dots]$

この解釈  $I$  において証明のサンプル一行目の論理式を解釈すると,

$$\begin{aligned} & I(\forall x(F(x) \rightarrow H(x)) \wedge \forall x(G(x) \rightarrow H(x))) \\ &= I(\wedge)(I(\forall x(F(x) \rightarrow H(x))), I(\forall x(G(x) \rightarrow H(x)))) \end{aligned}$$

であり,

$$\begin{aligned} I(\forall x(F(x) \rightarrow H(x))) &= T \\ \Leftrightarrow \text{全ての } \alpha \in D \text{ に対して } I'(F(x) \rightarrow H(x)) &= T \end{aligned}$$

であるから,  $I'(F(x) \rightarrow H(x))$  をすべての  $\alpha \in D$  に対して計算する.

$$\begin{aligned} I'(F(x) \rightarrow H(x)) &= I'(\rightarrow)(I'(F(x)), I'(H(x))) \\ &= I'(\rightarrow)(\tau(F)(\alpha), \tau(H)(\alpha)) \end{aligned}$$

であり,  $\alpha = 1$  のとき,

$$\begin{aligned} I'(F(x) \rightarrow H(x)) &= [(T, T) \mapsto T, (T, F) \mapsto F, (F, T) \mapsto T, (F, F) \mapsto T] \\ &\quad ([1 \mapsto T, 2 \mapsto T, 3 \mapsto F, 4 \mapsto F, 5 \mapsto F](1), \\ &\quad [1 \mapsto T, 2 \mapsto T, 3 \mapsto T, 4 \mapsto T, 5 \mapsto F](1)) \\ &= [(T, T) \mapsto T, (T, F) \mapsto F, (F, T) \mapsto T, (F, F) \mapsto T](T, T) \\ &= T \end{aligned}$$

同様に計算すると,

$$I'(F(x) \rightarrow H(x)) = \begin{cases} T & (\alpha = 2) \\ T & (\alpha = 3) \\ T & (\alpha = 4) \\ T & (\alpha = 5) \end{cases}$$

よって,

$$\text{全ての } \alpha \in D \text{ に対して } I'(F(x) \rightarrow H(x)) = T$$

したがって,

$$I(\forall x(F(x) \rightarrow H(x))) = T$$

同様のことを  $I(\forall x(G(x) \rightarrow H(x)))$  に対して行くと.

$$I(\forall x(G(x) \rightarrow H(x))) = T$$

がわかる. したがって目的の論理式の解釈  $I$  による真理値は,

$$\begin{aligned} &I(\forall x(F(x) \rightarrow H(x)) \wedge \forall x(G(x) \rightarrow H(x))) \\ &= I(\wedge)(I(\forall x(F(x) \rightarrow H(x))), I(\forall x(G(x) \rightarrow H(x)))) \\ &= [(T, T) \mapsto T, (T, F) \mapsto F, (F, T) \mapsto F, (F, F) \mapsto F](T, T) \\ &= T \end{aligned}$$

この解釈は, 全体集合を  $X = \{1, 2, 3, 4, 5\}$  とし, 集合  $A, B, C$  を  $A = \{1, 2\}$ ,  $B = \{1, 2, 3\}$ ,  $C = \{1, 2, 3, 4\}$  としたとき,  $F(x)$ ,  $G(x)$ ,  $H(x)$  をそれぞれ  $x \in A$ ,  $x \in B$ ,

$x \in C$  と解釈したのに相当する。この解釈において証明のサンプルは  $A$  が  $C$  の部分集合でかつ  $B$  が  $C$  の部分集合であることを前提として、 $A$  と  $B$  の合併が  $C$  の部分集合であることを結論する推論を表す。

論理式の集合  $\Gamma$  と論理式  $\varphi$  において、いかなる解釈のもとでも  $\Gamma$  に含まれる論理式がすべて真ならば、 $\varphi$  も真となる（すなわち任意の解釈  $I$  に対して、 $I(\gamma) = T(\gamma \in \Gamma) \Rightarrow I(\varphi) = T$ ）とき、 $\varphi$  は  $\Gamma$  の意味論的帰結といって、 $\Gamma \models \varphi$  とかく。

## 1.7 形式的体系 $\mathcal{K}$ の健全性と完全性

前章をスキップした人のためにこの章で必要になる知識を再掲する。

- 形式的体系  $\mathcal{K}$  において、仮定  $\Gamma$  から  $\varphi$  が証明できることを  $\Gamma \vdash \varphi$  とかく。
- 解釈  $I$  による論理式  $\varphi$  の解釈を  $I(\varphi)$  とかく。
- 論理式の集合  $\Gamma$  と論理式  $\varphi$  において、いかなる解釈のもとでも  $\Gamma$  に含まれる論理式がすべて真ならば、 $\varphi$  も真となる（すなわち任意の解釈  $I$  に対して、 $I(\gamma) = T(\gamma \in \Gamma) \Rightarrow I(\varphi) = T$ ）とき、 $\varphi$  は  $\Gamma$  の意味論的帰結といって、 $\Gamma \models \varphi$  とかく。

### 1.7.1 証明すべきこと

前置きが非常に長くなってしまったがいよいよステップ②に進もう。もうおわかりだと思うが証明すべきことは次である。

$$\Gamma \vdash \varphi \Leftrightarrow \Gamma \models \varphi \quad (1.1)$$

$\Rightarrow$  の方を形式的体系の健全性と言い、 $\Leftarrow$  の方を形式的体系の完全性と言う。（ $\Leftrightarrow$  で完全性と言ったりもする。）残念ながら著者の知識不足と時間不足により健全性については証明の一部を挙げるにとどめ、完全性については証明をしない。ここまで引っ張っておいて申し訳ない。

### 1.7.2 メタ論理

ここで重要な注意をする。それはこれから行うのがメタ論理による証明であることである。これまでもそうであったが我々は論理についての議論においてやはり論理を用いている。正しい論理（推論）とは何かを明らかにするために議論したはずが、その議論に当の論理を用いているのである。結局、我々はゼロからは何も生み出せず、議論のスタート地点でいくつかの道具立てを認めなければならないのだ。その道具のひとつが論理であり、これをメタ論理という。そう、つまり我々の論理が正しいと仮定したうえで論理の体系を

組み立てているのである。これは一見、結局はなにも成果がないように見える。しかしメタ論理として使う論理は我々が当然認められるようなものである。それが正しいとしたとき論理の形式的体系が得られ、それによって非常に複雑な推論が形式的に行えるようになったのなら十分な成果ではないだろうか。

これから行うのはメタ論理における証明であり、形式的体系による証明ではない。実は前章でもこっそり使っていたが、 $\Rightarrow$  や  $\Leftrightarrow$  はメタレベルでのならばと同値を表す。すでにメタ論理はオブジェクトの論理における推論や真理値といった概念を定義するのに多く使っている。たとえば真理値表なんかもそうで例えば  $\varphi \wedge \psi$  の真理値は  $\varphi$  が真でかつ  $\psi$  が真ならば真でそれ以外は偽であると定義した。また、これまでには集合の概念を何の断りもなく使い、前章での解釈の定義には写像の概念を用いている。数学も論理によって議論されるはずなのに前段階ですでに数学の概念を用いている。これらの数学もメタレベルの道具立てとして認めたのである。

しかし我々がメタレベルの道具立てとして認められるものは何であろうか。そのことに関してはいろいろな立場があり、 $\circ\circ$  主義の論理と呼ばれる。

### 1.7.3 形式的体系 $\mathcal{K}$ の健全性

$$\Gamma \vdash \varphi \Rightarrow \Gamma \models \varphi \quad (1.2)$$

が成り立つことを示す。そのためには以下を証明すればよい。形式的体系  $\mathcal{K}$  における証明のすべての行について、その行の依存する仮定を  $\Gamma$ 、論理式を  $\varphi$  とすると、

$$\Gamma \models \varphi \quad (1.3)$$

であることを行に関する数学的帰納法を用いて証明する。

1 行目は必ず仮定であるから論理式  $\varphi$ 、依存する仮定  $\Gamma = \varphi$  となる。これは明らかに  $\Gamma \models \varphi$

次に  $n$  行目までのすべての行について (1.3) が成り立つと仮定して、 $n+1$  行目に (1.3) が成り立つことを示す。 $n+1$  行目が仮定であれば明らかに成り立つ。 $n+1$  行目が前の行から推論規則を用いて導かれた場合について (1.3) が成り立つことを示す。すなわちすべての推論規則についてこれを示す。ここでは  $\wedge_+$ 、 $\rightarrow_+$  についてのみ示す。

$\wedge_+$  について、

$$\begin{array}{lll} (l) & \varphi & [\Gamma] \\ (m) & \psi & [\Delta] \\ (n+1) & \varphi \wedge \psi & [\Gamma, \Delta] \quad l, m, \wedge_+ \end{array}$$

$I(\gamma) = T(\gamma \in \Gamma)$  かつ  $I(\delta) = T(\delta \in \Delta)$  を仮定して,  $I(\varphi \wedge \psi) = T$  を示せばよい. 帰納法の仮定から  $I(\gamma) = T(\gamma \in \Gamma) \Rightarrow I(\varphi) = T$  かつ  $I(\delta) = T(\delta \in \Delta) \Rightarrow I(\psi) = T$ . よって仮定  $I(\gamma) = T(\gamma \in \Gamma)$  かつ  $I(\delta) = T(\delta \in \Delta)$  と帰納法の仮定から  $I(\varphi) = T$  かつ  $I(\psi) = T$  これより真理値の定義から  $I(\varphi \wedge \psi) = T$ . よって  $\Gamma, \Delta \models \varphi \wedge \psi$  が示せた.

$\rightarrow_+$  について,

$$\begin{array}{ll} (l) & \varphi \quad [\Gamma, \psi] \\ (n+1) & \psi \rightarrow \varphi \quad [\Gamma] \quad l, \rightarrow_+ \end{array}$$

$I(\gamma) = T(\gamma \in \Gamma)$  であることを仮定して  $I(\psi \rightarrow \varphi) = T$  を示せばよい. 帰納法の仮定は,  $(I(\gamma) = T(\gamma \in \Gamma) \text{ かつ } I(\psi) = T) \Rightarrow I(\varphi) = T$  である.  $I(\psi) = T$  または  $I(\psi) = F$  であり  $I(\psi) = F$  のときは, 真理値の定義から  $I(\psi \rightarrow \varphi) = T$  であり,  $I(\psi) = T$  のときは仮定  $I(\gamma) = T(\gamma \in \Gamma)$  と帰納法の仮定より  $I(\varphi) = T$  である. よってどちらにしろ  $I(\psi \rightarrow \varphi) = T$  である. よって  $\Gamma \models \psi \rightarrow \varphi$  が示せた.

以下省略し, 形式的体系  $\mathcal{K}$  の健全性の証明とする. これを証明することで晴れて我々は形式的体系  $\mathcal{K}$  を安心して使えるようになったのである. あとは完全性であるが, これは証明しない. 完全性を証明できると, その体系が十分な証明能力を有していることがわかり, この二つの証明をもって, 証明可能性と意味論的帰結が等価なものであることがいえるのである.

## 1.8 おわりに～数学の形式化は可能か～

我々はこれまでの議論において, 簡単な論理と数学的帰納法や集合の概念をメタレベルの道具立てとして, 論理を形式化することに成功した. これによって得られる恩恵は第一に厳密な論理を安心して扱えるようになったことである. そして第二に形式化された推論を自由な解釈によってさまざまに応用できることである. この恩恵を最も受けるのは数学であるが, はたして数学は完全に形式化できるのだろうか. 実はそう簡単な話ではない. 述語論理は述語  $F(x)$  を  $F$  を満たす  $x$  を集めた集合と考えることで集合の概念と同一視できる. 数学では集合の集合を扱うことが多いがこれは述語の述語を扱うのと同じである. しかしそれを無制限に許すとラッセルのパラドックスと呼ばれる矛盾を導いてしまう. これを回避するために, 矛盾が生じないように公理系を設けた公理的集合論が考案された. しかし算術を含む体系においては有限回の操作と数学的帰納法というメタレベルの道具立てにおいてその体系が無矛盾ならその無矛盾性をその体系内では証明できないことが示された (これをゲーデルの不完全性定理という). よって数学を完全に形式化し論理学に基礎づけることは不可能である.



## 参考文献

- [1] 飯田賢一・中才敏郎・中谷隆雄, 『論理学の基礎』, 昭和堂, 1994.
- [2] 鹿島亮, 『数理論理学』, 朝倉書店, 2009.
- [3] 斎藤毅, 『集合と位相』, 東京大学出版会, 2009.
- [4] 齋藤正彦, 『数学の基礎 集合・数・位相』, 東京大学出版会, 2002.
- [5] 野矢茂樹, 『論理学』, 東京大学出版会, 1994.
- [6] 戸次大介, 『数理論理学』, 東京大学出版会, 2012.
- [7] 松本和夫, 『復刊 数理論理学』, 共立出版, 1970(初版), 2001(復刊).