# 2019 年度 計算可能性理論特論・講義ノート[*1] [*2]

木原 貴行
名古屋大学 情報学部・情報学研究科

最終更新日: 2020 年 3 月 10 日

[*1] 本講義ノートは，2019 年度秋 1 期および秋 2 期開講の名古屋大学大学院情報学研究科における講義「計算可能性理論特論 1」および「計算可能性理論特論 2」の内容をまとめる予定のものである．

[*2] このノートは講義期間中にリアルタイムで更新しており，現時点では未完成なので，あまり拡散しないでください．講義のページ： http://www.math.mi.i.nagoya-u.ac.jp/~kihara/teach.html

# 目次

# 第 1 章

# 部分結合子代数

## §1. チャーチ・チューリングの提唱

計算という概念自体は，はるか昔の時代から人びとの頭の中にあった．アルゴリズムという言葉の由来は 9 世紀の数学者アル＝フワーリズミーに遡る．実際に計算を実行する機械の起源としては，1642 年にパスカルは足し算を実行する機械を発明したらしい．より高度な計算として，1671 年にライプニッツの発明した機械は，乗算，除算や平方根の計算などもできたという．ライプニッツが，記号計算の研究にかなり尽力していたことは有名である．その後も，様々な計算を行う機械が考案されていった．たとえば，1822 年にバベッジは多項式の値を計算する能力をもつ機械である階差機関 (difference engine) を設計したが，実際に完成させることはなかった．とにかく，このように 19 世紀以前においても，様々な計算機械が考案されてきたのである．しかし，これらのような，計算論前史における機械の多くは，特定の関数だけを計算することに特化している．

その例外となるものとして，1830 年代からはバベッジは解析機関 (analytic engine) と呼ばれる機械の設計に没頭する．彼が亡くなる 1870 年代まで設計の改良が続けられたが，これもまた現実に完成することはなかった．解析機関は，計算論前史における典型的な機械と違い，パンチカードを用いて様々な計算を実現可能な，いわゆる汎用コンピュータの一種であった．ただし，解析機関でそれなりの種類の計算が実行できるからといって，ありとあらゆる種類の計算が実行できるかどうかは，少なくとも当時の感覚では，まったく分かったものではない．そもそも「ありとあらゆる計算」とは何であるか，バベッジの時代には，まったく想像の付くものではなかったと思われる．

もちろん，現代の計算論的知識を用いることによって，解析機関で実行できるいくつかの種類の計算をとても非自明な方法で複雑に組み合わせると，（計算時間を気にしなければ）理論上は現代のコンピュータと同じ計算ができると示せる，と言われている．実際，これから本稿で見るように，ほんのわずかな種類の基本的な計算さえ実行できれば，それが汎用計算モデルとなることが示せるため，そうであってもまったく不思議なことではない．

しかし，計算理論の誕生において最も重要であったブレイクスルーは，「計算とは何か」が定式化されたこと，そして，「わずかな種類の計算から膨大な種類の計算を実行可能である」と証明さ

れたことである．これらの偉大な発見は，バベッジよりもかなり後の時代のこととなる．「計算とは何か」に至る研究は，20 世紀の数学基礎論の発展以降に活発となる．その大きなイベントは 1936 年に起きた．1936 年に，チャーチはラムダ計算を導入し，チューリングはチューリングマシンを導入し，彼らは独立にヒルベルトの決定問題の否定的解決を宣言した．同年に，クリーネは，エルブランとゲーデルのアイデアに基づき $\mu$-再帰関数を導入し，またポストは彼が Formulation I と呼ぶ計算モデルを導入した．クリーネは $\mu$-再帰関数と $\lambda$-計算が，計算モデルとして等価であることを示し，翌 1937 年に，チューリングは，チューリングマシンとそれらの計算モデルとの等価性を示した．その後，現代に至るまで，幾百もの計算モデルと幾千ものプログラミング言語が世に溢れることとなる．そして，「これらの計算モデルによって計算可能なものは（計算速度などを気にしなければ）チューリングマシンによって計算可能なものと正確に一致する」ということが数学的に証明され，ならばこれが計算可能性の妥当な定義であろう，ということで決着が付いた．これが**チャーチ・チューリングの提唱** (*Church-Turing thesis*) である．

念のため，この計算史の中で，どこまでが《数学》でどこからが《提唱》なのかを明示しておこう．まず，

> 「1930 年代以降に人類が考案した幾千幾万の計算モデルの計算能力が（計算速度などを気にしなければ）等価である」

という点については，**数学的な証明が与えられている**部分であるから，ここは《**数学**》である．ちなみに 1930 年代以降と注記したのは，現代的な計算モデルの計算能力よりも真に弱い計算モデルも，過去には考案されてきたためである．1930 年代以降でも，弱い計算モデルが意図的に扱われることがあるが，とにかく，弱い計算モデルは無視することにする．さておき，その一方で，

> 「これまでに人類が考えついた計算モデルの計算能力はすべて等価なのだから，これが計算可能性の定義なのだろう」

という点については，《**提唱**》である．なぜなら，現時点までに人類が考えついた幾千幾万の計算モデルの計算能力がたまたま等価だった，というだけのことに過ぎないかもしれず，実は，人類はまだ完璧な計算モデルを考案できていないのかもしれない．そして，千年後あるいは十億年後の未来に，何らかの生命体が，既知の計算モデルを超越する計算能力を持つ機械を発明するかもしれない．まず有り得ないだろうが，全く有り得ないと断言できるわけではない，かもしれない．

とにかく，このように，世には幾千幾万の等価な計算モデルが存在している．すると，これらの等価な計算モデルに共通する何か，計算可能性の本質のようなものがあってよいはずである．そのような本質は，人類史において既に現れた計算モデルだけでなく，未だ見ぬ遠い未来の超越的な計算モデルにも（そのようなものが存在するかどうかはさておいて）共有されているものであろう．つまるところ，われわれは，ありとあらゆる計算モデルの共有する，計算可能性の根源となる普遍的原理を，何らかの単純な数学的構造として抽出したい．未来の超越的な計算モデルという夢想はさておくにせよ，現存する計算モデルの数学的本質を追求することに価値があるという点については疑いの余地はないであろう．

本稿では，このために，まず，計算という概念の代数構造としての側面を分析することから始

めよう.

# §2. 代数構造としての計算

## 2.1. 計算の部分モノイド作用

まず，コンピュータ・プログラム全体のなす集合がどのような代数的性質を持つかについて考察したい．本稿ではプログラミング言語あるいは計算モデルに依存しない普遍的な性質のみを扱うので，好きなプログラミング言語あるいは計算モデル（ただしチューリング完全なものに限る）を思い浮かべて欲しい．さて，入出力を持つプログラムは関数と考えることができそうだから，少なくとも関数の集合と同等以上には豊富な代数的構造を持つはずである.

ただし，コンピュータの動作には，「計算が停止しない（計算が無限ループに陥って出力を返さない)」という状態が発生し得る．このため，プログラムによって表されるものは，厳密に言えば，関数ではなく，部分関数や部分演算と呼ばれるものである.

**定義 2.1. 部分関数** (*partial function*) とは，与えられた入力に対して出力を返すとは限らない関数である．可能な入力の集合が $X$ であり，出力がすべて $Y$ に属すならば，$X$ から $Y$ への部分関数といい，$f :\subseteq X \to Y$ と書く．入力 $x \in X$ に対して，$f(x)$ が定義されるとき，$f(x)\downarrow$ と書き，さもなくば $f(x)\uparrow$ と書く．このとき，$\mathrm{dom}(f) := X$ を $f$ の**始域** (*domain*) と呼び，$\mathrm{def}(f) := \{x \in X : f(x)\downarrow\}$ を**定義域** (*domain of definition*) と呼ぶことがある．始域と定義域が等しい関数は**全域関数** (*total function*) と呼ばれる.

ところで，$\mathrm{def}(f)$ を始域とする関数として見れば $f$ はふつうの関数なのに，なぜわざわざ $\mathrm{dom}(f)$ 上の部分関数と考えるのか，と思う人もいるかもしれない．しかし，計算理論においては，

始域 $\mathrm{dom}(f)$ が何であるかは分かるが，定義域 $\mathrm{def}(f)$ が何であるかは絶対に分からない

という状況が起こり得る．つまり，コンピュータ・プログラムが与えられたとき，どの入力に対して出力を返すかを一般的に判定するような計算可能な方法は存在しない．それが，計算可能性理論の「はじまりの定理」である 1936 年のチューリングの定理であった.

そういうわけで，部分関数や部分演算を考えることは計算理論においては本質である．集合 $X$ 上の関数全体の集合 $X^X$ には，関数合成 $g \circ f$ を 2 項演算と考えることによって，（非可換）モノイドの構造が入る．部分関数に対する関数合成は，以下によって定義され，これもモノイド演算をなす.

$$g \circ f(x)\downarrow = y \iff f(x)\downarrow \text{ かつ } g(f(x))\downarrow = y.$$

ここで，群とは，結合律を満たし，単位元と逆元が存在する 2 項演算をもつ集合であったが，そのうち結合律のみを満たすものを半群，それに加えて単位元が存在するものをモノイドと呼ぶ.

**定義 2.2.** 集合 $S$ 上の 2 項演算 $* : S^2 \to S$ に関する以下の性質を考える．

$$\text{結合律:}\quad \forall a, b, c \in S\ ((a * b) * c = a * (b * c)).$$
$$\text{単位元の存在:}\quad \exists \varepsilon \in S \forall a \in S\ (a * \varepsilon = \varepsilon * a = a).$$

集合 $S$ とその上の 2 項演算 $*$ の組で，結合律を満たすものを**半群** (*semigroup*) と呼び，結合律と単位元の存在を満たすものを**モノイド** (*monoid*) と呼ぶ．

**例 2.3.** 自然数上の加法 $+$ と零 0 を考えると，$(\mathbb{N}; +, 0)$ はモノイドをなす．$n \times n$ 整数行列全体の集合を $M_n(\mathbb{Z})$ と書き，$M_n(\mathbb{Z})$ 上の積 $\cdot$ と単位行列 $E$ を考えると，$(M_n(\mathbb{Z}); \cdot, E)$ はモノイドをなす．集合 $X$ 上の関数 $f : X \to X$ の合成 $\circ$ と恒等写像 id を考えると，$(X^X; \circ, \mathrm{id})$ はモノイドをなす．

**例 2.4.** 集合 $A$ 上の語全体 $A^*$ について，積 $*$ を語の結合，つまり $u * v = uv$ によって定義し，$\varepsilon$ を空語とすると，$(A; *, \varepsilon)$ はモノイドをなす．これを $A$ 上の**自由モノイド** (*free monoid*) と呼ぶ．

**例 2.5.** プログラム $p, q$ を 2 つ繋ぎ合わせることによって，新たなプログラム $p * q$ を得ることができる．また，「受け取った入力をそのまま出力するプログラム」という単位元があるが，一般には計算は「不可逆」であるから，逆元が存在するとは限らない．このようにして，コンピュータ・プログラム全体の集合はモノイドと考えることができる．以後，このモノイドを計算モノイドと呼び，$\mathbf{P}$ と書くことにする．

さて，計算概念を数学的に扱うひとつの方法としては，計算を「作用」として理解することである．つまり，計算 $p$ が作用することによって，ある対象 $x$ が別の対象 $y$ に変化する，という仕組み $p \cdot x = y$ であると考える．数学において最もよく扱われる作用概念は以下である．

**定義 2.6.** 群 $(G; *)$ の集合 $X$ への**左群作用** (*left group action*) とは，次の性質を満たす二項演算 $\alpha \colon G \times X \to X$ である．ただし，以下 $\alpha(g, x)$ を $g \cdot x$ と略記する．

1. $G$ の単位元 $e$ と任意の $x \in X$ について，$e \cdot x = x$ が成り立つ．
2. 任意の $g, h \in G$ と $x \in X$ について，$g \cdot (h \cdot x) = (g * h) \cdot x$ が成り立つ．

**例 2.7.** 群 $(G, *)$ は $G$ 自身に次のように自明に作用する：任意の $g, x \in G$ について，$g \cdot x = g * x$ とすればよい．

**例 2.8.** 集合 $X$ 上の対称群 $\mathrm{Sym}(X)$ とは，$X$ 上の全単射全体のなす群であった．ここで，少し特殊であるが $g * h = h \circ g$ と定義する．対称群 $\mathrm{Sym}(X)$ は，冪集合 $\mathcal{P}(X)$ に次のように作用する．任意の $g \in \mathrm{Sym}(X)$ と $A \subseteq X$ に対して，$g \cdot A = g^{-1}[A]$ とする．

**例 2.9.** $\mathbb{N}$ 上の計算可能全単射全体のなす群を $\mathrm{Sym}^{\mathbf{P}}(\mathbb{N})$ 書く．ここで群演算は $g * h = h \circ g$ で定義されているとする．このとき，群 $\mathrm{Sym}^{\mathbf{P}}(\mathbb{N})$ は冪集合 $\mathcal{P}(\mathbb{N})$ に次のように作用する．任意の $g \in \mathrm{Sym}^{\mathbf{P}}(\mathbb{N})$ と $A \subseteq X$ に対して，$g \cdot A = g^{-1}[A]$ とする．このとき，以下のようにして $\mathcal{P}(\mathbb{N})$

上の同値関係 $\simeq_{\mathbf{P}}$ が定義される：

$$A \simeq_{\mathbf{P}} B \iff (\exists g \in \mathrm{Sym}^{\mathbf{P}}(\mathbb{N}))\ g \cdot B = A.$$

このとき，集合 $A$ と $B$ は**計算可能同型** (*computably isomorphic*) であるという．

以後，左群作用のことを単に群作用と呼ぶことにする．群作用の類似物として，半群作用やモノイド作用などがある．定義 2.6 における $G$ をモノイドに置き換えたものが，モノイド作用の定義である．例 2.8 の対称群の代わりに，$X$ 上の単射のなすモノイドや関数のなすモノイドを考えても，モノイド作用を得ることができる．

**例 2.10.** $\mathbb{N}$ 上の計算可能単射全体のなすモノイドを $\mathcal{F}_1^{\mathbf{P}}(\mathbb{N})$ と書き，$\mathbb{N}$ 上の計算可能関数全体のなすモノイドを $\mathcal{F}_m^{\mathbf{P}}(\mathbb{N})$ と書くとする．ここでモノイド演算は $g * h = h \circ g$ で定義されているとする．このとき，モノイド $\mathcal{F}_1^{\mathbf{P}}(\mathbb{N})$ と $\mathcal{F}_m^{\mathbf{P}}(\mathbb{N})$ は冪集合 $\mathcal{P}(\mathbb{N})$ に例 2.8 および例 2.9 と同様の作用をする．このとき，以下のようにして $\mathcal{P}(\mathbb{N})$ 上の擬順序関係 $\leqslant_1$ と $\leqslant_m$ が定義される：

$$A \leqslant_1 B \iff (\exists g \in \mathcal{F}_1^{\mathbf{P}}(\mathbb{N}))\ g \cdot B = A.$$
$$A \leqslant_m B \iff (\exists g \in \mathcal{F}_m^{\mathbf{P}}(\mathbb{N}))\ g \cdot B = A.$$

集合 $A, B \subseteq \mathbb{N}$ について，$A \leqslant_1 B$ であるとき $A$ は $B$ へ**一対一還元可能** (*one-one reducible*) であるといい，$A \leqslant_m B$ であるとき $A$ は $B$ へ**多対一還元可能** (*many-one reducible*) であるという．この概念はエミール・ポスト (Emil Post; 1897–1954) によって 1944 年に導入された．

上で述べたように，計算の基礎は部分関数と部分演算であったから，作用についても同様に，部分モノイド作用と呼ばれるものを考えた方が都合が良い．

**定義 2.11.** モノイド $(M; *)$ の集合 $X$ への**部分モノイド作用** (*partial monoid action*) とは，次の性質を満たす二項演算 $\alpha :\subseteq M \times X \to X$ である．ただし，以下 $\alpha(g, x)$ を $g \cdot x$ と略記し，これが定義されるとき $g \cdot x \downarrow$ と書く．

1. $M$ の単位元 $e$ と任意の $x \in X$ について，$e \cdot x = x$ が成り立つ．
2. 任意の $g, h \in M$ と $x \in X$ について，$g \cdot (h \cdot x) \simeq (g * h) \cdot x$ が成り立つ．

次が計算理論における部分モノイド作用の最も基本的な例のひとつである．

**例 2.12.** プログラム $p \in \mathbf{P}$ に自然数 $n \in \mathbb{N}$ を入力した結果を $p \cdot n$ と書けば，これは $\mathbf{P}$ の $\mathbb{N}$ への部分モノイド作用を与える．

ところで，コンピュータにおける計算において，プログラム $p$ もその入出力も有限バイナリ列によってコードされている．つまり，コンピュータの世界においては，プログラムの集合と入出力の集合は本質的には同じものと考えられる．もう少し数学的に述べれば，モノイド $M = \mathbf{P}$ の台集合と $M$ が作用する集合 $X$ が等しい，という特徴を持つ．つまるところ，モノイドが台集合

自身に「自己作用」するのだが，一応，注意しておくと，作用 $*$ とモノイド演算 $\cdot$ が異なるので，これは例 2.7 のような自明な作用とは異なる．このように「$(X, *)$ が $X$ に（非自明に）作用する」という状況は計算理論においてはしばしば現れる．この状況が，次の節において導入する部分結合子代数のアイデアに繋がっていく．

**例 2.13.** もうひとつの重要な部分モノイド作用としては，計算モノイド $\mathbf{P}$ の自然数上の関数全体の集合 $\mathbb{N}^{\mathbb{N}}$ への作用がある．コンピュータ・プログラム $p$ が作用することによって，自然数上の関数 $f$ が別の関数 $g$ に変化するという状況を考える．自然数上の関数は完結した有限的情報として表されるとは限らないが，コンピュータとその利用者のやり取りが延々続く，あるいはネットワークなどを介したコンピュータ間のやり取りが延々続く計算としてモデル化できる．これは**オンライン計算** (*on-line computation*) と呼ばれることもある．このようにして，$\mathbf{P}$ の $\mathbb{N}^{\mathbb{N}}$ への部分モノイド作用を得られる．

次によって定義される $\mathbb{N}^{\mathbb{N}}$ 上の擬順序構造は，計算可能性理論において深く研究されている．

$$f \leqslant_T g \iff (\exists p \in \mathbf{P})\ p \cdot g = f.$$

関係 $f \leqslant_T g$ が成立するとき，$f$ は $g$ に**チューリング還元可能** (*Turing reducible*) であると言われる．この概念は，アラン・チューリング (Alan Turing; 1912–1954) によって 1939 年に導入された．

## 2.2. 部分結合子代数

我々の目的は，計算のために本質的な代数的構造を抽出することである．代数構造といえば，群，環，体などがよく知られるが，我々の扱う代数構造は，それらとは方向性が異なる．群より弱い代数構造としては，モノイド，半群などがあり，マグマが最弱の代数構造として扱われることが多い．

$$\text{(強)} \qquad \text{群} \implies \text{モノイド} \implies \text{半群} \implies \text{マグマ} \qquad \text{(弱)}$$

しかし，本稿では，マグマより弱い，以下のような最もプレーンな代数構造を導入する．

**定義 2.14.** 集合 $A$ と部分 2 項演算 $\cdot :\subseteq A \times A \to A$ の対は，**部分マグマ** (*partial magma*) と呼ばれる．部分マグマ $A$ の元 $x, y \in A$ について，$x \cdot y$ の値が定義されているとき，$x \cdot y \downarrow$ と書く．そうでないときは，$x \cdot y \uparrow$ と書く．

**注意.** マグマは古くは亜群 (groupoid) と呼ばれることもあったらしいが，亜群 (groupoid) の名はより重要な別概念（全ての射が可逆である圏，つまり “partial group” とも呼べる代数構造）に用いられて紛らわしいので，ここでは用いない．ちなみに partial を「部分」と訳すと，sub- と紛らわしいので，この種の partial のことも「偏」と訳す流儀があるようである．たとえば，部分マグマでなく偏マグマと訳すような感じである．

**例 2.15.** 任意の半群は部分マグマである．実際，半群とは，結合律 $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ を満たす（全域）マグマのことである．

しかし，これから取り扱う具体的な構造は，いずれも結合律を満たさない．実際，計算論的にある種の良い性質を持つ部分マグマは，決して結合律を満たし得ない，ということが後に分かる．それでは，結合律を満たさない部分マグマには，たとえばどのようなものがあるだろうか．

**例 2.16.** $(\mathbb{Z}, -)$ や $(\mathbb{R}, \div)$ は非結合的な部分マグマである．たとえば，$(4 \div 2) \div 2 = 2 \div 2 = 1$ であるが，$4 \div (2 \div 2) = 4 \div 1 = 4$ である．

とはいえ，今後扱う部分マグマは，例 2.16 のような非結合的マグマとは大きく異なる．今後の話で念頭に置いておくとよい部分マグマは，関数適用のなす部分マグマである．

**例 2.17.** Sets を集合全体のクラスとする．集合 $f \in \mathsf{Sets}$ の定義域 $\mathrm{dom}(f)$ とは，$(x, y) \in f$ となる $y \in \mathsf{Sets}$ が存在するような $x \in \mathsf{Sets}$ 全体のことである．集合 $f \in \mathsf{Sets}$ が関数であるとは，任意の $x \in \mathrm{dom}(f)$ について $(x, y) \in f$ となる $y \in \mathsf{Sets}$ が唯一であることであり，このとき $f(x) = y$ と書く．このとき，$f, x \in \mathsf{Sets}$ について，

$$f \cdot x \downarrow \iff f \text{ は関数である } \& \ x \in \mathrm{dom}(f)$$

とし，また，このときの値は $f \cdot x = f(x)$ により定義する．

部分マグマは，このような関数適用のなす代数系 $(\mathsf{Sets}, \cdot)$ を念頭に置いておくとイメージしやすい．ただし，厳密には Sets 自体は集合ではないので，気になる人は，$\mathsf{Sets} = V$ の代わりに，適当な基数 $\kappa$ について $V_\kappa$ を考えるとよい．

もう一つ，今後の部分マグマの利用のアイデアを思い浮かべやすいものは，群作用あるいは部分モノイド作用のなす部分マグマである．

**例 2.18.** 群 $G$ の集合 $X$ への群作用は，$G \cup X$ 上の部分 2 項演算とみなすこともできる．つまり，$g, x \in G \cup X$ について，

$$g \cdot x \downarrow \iff g \in G \ \& \ x \in X$$

かつ，$g \cdot x$ は群作用によって定義される．よって，群 $G$ が集合 $X$ に作用しているとき，$(G \cup X, \cdot)$ は部分マグマをなす．同様に，モノイド $M$ の集合 $X$ への部分モノイド作用が与えられているとき，$(M \cup X, \cdot)$ は部分マグマをなす．

さて，計算概念の代数的抽象化の議論に戻ろう．まず，「計算モデル」というからには，基本的な関数は実装できてほしい．たとえば，各 $a \in A$ に対して，定数関数 $\mathtt{const}_a : B \to A$ を実装できるべきである．さらに言えば，$a \mapsto \mathtt{const}_a$ を実装できてほしい．つまり，$\mathtt{k}(a) = \mathtt{const}_a$ となる関数 $\mathtt{k} : A \to A^B$ を実装できる．これは以下のように表される．

$$\mathtt{k}(a)(b) = a.$$

続いて，関数適用も実装できるのがよいだろう．関数のリスト $f = (f_a : B \to C)_{a \in A}$ が与えられたとき，入力リスト $x = (x_a)_{a \in A}$, $x_a \in B$, に出力リスト $(f_a(x_a))_{a \in A}$ を対応させる関数を実装したい．これは，次の関数 $\mathtt{s} : [B \to C]^A \to [B^A \to C^A]$ が実装できるということである．

$$\mathtt{s}(f)(x)(a) = (f(a))(x(a)).$$

とりあえず，我々の「計算モデル」に求めるものは以上であるとする．例 2.17 のように関数適用を二項演算として表せば，これは以下の要件として書き直せる．

**定義 2.19. 部分結合子代数** (*partial combinatory algebra*) とは，部分マグマ $(M, \cdot)$ に**結合子** (*combinator*) と呼ばれる以下の特殊な元 $\mathtt{k}, \mathtt{s} \in M$ が備わったものである．

$$(\forall a, b \in M) \quad \mathtt{k} \cdot a \downarrow \text{ and } (\mathtt{k} \cdot a) \cdot b = a$$
$$(\forall f, x, a \in M) \quad (\mathtt{s} \cdot f) \cdot x \downarrow \text{ and } ((\mathtt{s} \cdot f) \cdot x) \cdot a \equiv (f \cdot a) \cdot (x \cdot a).$$

ここで，$\equiv$ は強い意味での同値性，すなわち $M \cup \{\uparrow\}$-値として一致するということである．これは，上の説明で言うところの $f_a$ が部分関数であり，$f_a(x_a)$ が定義されない場合を想定している．さらに言えば，そもそも $a \mapsto f_a$ や $a \mapsto x_a$ が部分関数である状況も考慮に入れている．

**豆知識.** 部分結合子代数は**シェーンフィンケリ代数** (*Shönfinkel algebra*) とも呼ばれる．これは，ロシアの数学者モイセイ・シェーンフィンケリ (Moses Shönfinkel; 1889–1942) によって導入された**結合子論理** (*combinatory logic*) を代数的に表現したものであるためである．シェーンフィンケリは 1914 年から 1924 年までゲッティンゲンのダフィット・ヒルベルトの研究グループに属しており，結合子論理のアイデアはそこで 1920 年 12 月に発表されたものであるようだ．結合子論理に関するシェーンフィンケリの研究は，「数理論理学の構成単位について」という題で 1924 年に出版された．

部分結合子代数の定義 2.19 を人為的だと感じる人もいるかもしれないので，あらかじめ述べておくと，部分結合子代数を「結合子完全な部分マグマ」として抽象的に定義することもできる (定理 3.3)．さて，部分結合子代数の代数的性質を見ると，単なる部分マグマよりは幾分か秩序を持つ．たとえば，部分結合子代数は，左単位的部分マグマである．

**命題 2.20.** 部分結合子代数 $M$ は必ず左単位元 $\mathtt{i} \in M$ を持つ．つまり，ある $\mathtt{i} \in M$ が存在して，任意の $a \in M$ に対して，$\mathtt{i} \cdot a \downarrow = a$ となる．

*Proof.* $\mathtt{i} = (\mathtt{s} \cdot \mathtt{k}) \cdot \mathtt{k}$ と定義する．このとき，

$$\mathtt{i} \cdot a = ((\mathtt{s} \cdot \mathtt{k}) \cdot \mathtt{k}) \cdot a \equiv (\mathtt{k} \cdot a) \cdot (\mathtt{k} \cdot a) = a$$

となるから，$\mathtt{i}$ は左単位元である． □

以後，部分組合せ代数が与えられたときに，積 $\cdot$ の記号は省略し，また積は左結合的であると仮定する．つまり，$(a \cdot b) \cdot c$ は $abc$ のように積と括弧は省略する．たとえば，

$$\mathtt{k}ab = a, \quad \mathtt{s}abc \equiv ac(bc), \quad \mathtt{i} = \mathtt{skk}$$

のように書かれる．記法の慣れのために，簡単な命題をいくつか証明してみることにしよう．そ

の前に，1 つの元からなる単位的部分マグマ $\{\mathtt{i}\}$ を自明な部分結合子代数と呼ぶことにする．次の主張は重要ではないが，非自明な部分結合子代数のもつ代数的性質に関する少しの情報を与える．

**命題 2.21.** 非自明な部分結合子代数は非結合的かつ非可換である．

*Proof.* 部分結合子代数 $M$ が与えられているとする．一方，$\mathtt{k}$ の定義より $\mathtt{kkk} = \mathtt{k}$ であり，また，任意の $a \in M$ について $\mathtt{k}a\mathtt{k} = a$ かつ $\mathtt{k}(\mathtt{kk})a = \mathtt{kk}$ である．よって，$\mathtt{k}(\mathtt{kk})a\mathtt{k} = \mathtt{kkk} = \mathtt{k}$ である．もし $M$ が結合的ならば，特に $\mathtt{kkk} = \mathtt{k}(\mathtt{kk})$ なので，$\mathtt{k}(\mathtt{kk})a\mathtt{k} = \mathtt{kkk}a\mathtt{k} = \mathtt{k}a\mathtt{k} = a$ を得る．よって，$M = \{\mathtt{k}\}$ を得るから $M$ は自明である．

つづいて，$M$ が可換であったとする．特に，命題 2.20 の左単位元 $\mathtt{i} \in M$ について，$\mathtt{ki}(\mathtt{kii}) = \mathtt{kii}(\mathtt{ki})$ を得る．いま，$\mathtt{k}$ の定義より $\mathtt{ki}(\mathtt{kii}) = \mathtt{kii} = \mathtt{i}$ である．また，$\mathtt{k}$ と $\mathtt{i}$ の定義より $\mathtt{kii}(\mathtt{ki}) = \mathtt{i}(\mathtt{ki}) = \mathtt{ki}$ となる．よって，可換性より $\mathtt{i} = \mathtt{ki}$ を得る．このとき，任意の $a \in M$ について，$\mathtt{a} = \mathtt{ia} = \mathtt{kia} = \mathtt{i}$ となり，つまり $M = \{\mathtt{i}\}$ であるから $M$ は自明である． □

もう一つ重要な例として，部分結合子代数において，関数合成 $(f, g) \mapsto f \circ g$ を表せることを見ておこう．つまり，$\mathtt{b}fgx = f(gx)$ となる $\mathtt{b} \in M$ が存在する．

**命題 2.22.** 部分結合子代数 $M$ は必ず次の性質をもつ元 $\mathtt{b} \in M$ を持つ．任意の $f, g \in M$ について $\mathtt{b}fg\downarrow$ であり，$\mathtt{b}fgx \equiv f(gx)$ を満たす．

*Proof.* $\mathtt{b} = \mathtt{s}(\mathtt{ks})\mathtt{k}$ と定義する．このとき，$\mathtt{s}$ と $\mathtt{k}$ の定義に沿って書き換えを行えば，

$$\mathtt{b}fg = \mathtt{s}(\mathtt{ks})\mathtt{k}fg = \mathtt{ks}f(\mathtt{k}f)g = \mathtt{s}(\mathtt{k}f)g\downarrow$$

を得る．したがって，

$$\mathtt{b}fgx = \mathtt{s}(\mathtt{k}f)gx = \mathtt{k}fx(gx) = f(gx)$$

となるから，目的の性質が導かれた． □

命題 2.22 より，部分結合子代数 $M$ が与えられたとき，$M$ 上の 2 項演算 $*$ を $x * y = \mathtt{b}xy$ によって定義できる．ここで，$M$ 上の外延的同値関係 $\equiv_\eta$ を次によって定義しよう：

$$x \equiv_\eta y \iff (\forall a \in M)\ \ x \cdot a \simeq y \cdot a.$$

このとき，$\mathtt{i} * x \equiv_\eta x * \mathtt{i} \equiv_\eta x$ であることと $x * (y * z) \equiv_\eta (x * y) * z$ であることが導かれる．したがって，2 項演算 $*$ は商代数 $M_\eta := M/\equiv_\eta$ 上のモノイド構造を誘導する．さらに，$*$ の定義により，$(g * h) \cdot x \simeq g \cdot (h \cdot x)$ が成立しているから，モノイド $(M_\eta, *)$ の $M_\eta$ への自己部分モノイド作用が自動的に得られる．

さて，そろそろ部分結合子代数の具体例をあげておく必要があるだろう．部分結合子代数には，関数適用 (例 2.16) に類似した形のものが多いため，そのような例を下に 4 つ述べる．最初の例は，最も基本的な部分結合子代数の例であり，部分結合子代数のイメージを掴むために重要である．

**例 2.23** (クリーネの第 1 代数)**.** 部分結合子代数の重要な具体例は，計算モデルによる自己作用から得られる．例 2.12 で，プログラム全体の集合 $\mathbf{P}$ の $\mathbb{N}$ への自然な部分モノイド作用があったことを思い出そう．したがって，例 2.18 のようにして $(\mathbf{P} \cup \mathbb{N}, \cdot)$ は部分マグマをなす．具体的には，プログラム $p \in \mathbf{P}$ と自然数 $n \in \mathbb{N}$ について，

$$p \cdot n \downarrow = m \iff \text{プログラム } p \text{ に } n \text{ を入力すると } m \text{ を出力する}.$$

しかし，例 2.12 の直後に述べたように，プログラムはバイナリ列（あるいは自然数）としてコードできるから，$\mathbf{P} = \mathbb{N}$ と考えてよい．つまり，計算モデルが与えられれば，自然数上の自己作用から得られる部分マグマを作ることができる．この部分マグマ $(\mathbb{N}, \cdot)$ が部分結合子代数をなすことは，実際に `s` と `k` の機能を持つプログラムを書くことで確認できる．チューリングマシンの文脈では，これは万能チューリングマシンの存在とパラメータ定理 (smn 定理) から保証できる（第 4.1 節を参照せよ）．部分結合子代数 $K_1 := (\mathbb{N}, \cdot)$ は**クリーネの第 1 代数** (*Kleene's first algebra*) としてよく知られている．

以下に述べる 3 つの例は，部分結合子代数の例が豊富にあるということを示すものであるが，現時点で理解しておく必要はない．

**例 2.24** (クリーネの第 2 代数)**.** $\mathbb{N}$ には離散位相が入っているとし，$\mathbb{N}^{\mathbb{N}}$ をその可算直積空間とする．この位相空間は，たとえば無理数全体 $\mathbb{R}\backslash\mathbb{Q}$ に $\mathbb{R}$ 上のユークリッド位相の相対位相が入ったものと同相であることが連分数展開によって示される．いま，$\mathcal{C}$ を $\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数全体の集合とすると，$\mathcal{C} \cup \mathbb{N}^{\mathbb{N}}$ は以下の部分 2 項演算によって，部分マグマをなす．

$$f \cdot x \downarrow = y \iff f \in \mathcal{C} \text{ かつ } x \in \mathrm{dom}(f) \text{ かつ } f(x) = y.$$

しかし，これでは部分結合子代数をなさないので，もう一工夫が必要である．いま，$\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数全体 $\mathcal{C}$ の濃度は $\mathbb{N}^{\mathbb{N}}$ の濃度と等しい（ともに連続体濃度である）ので，$\mathcal{C} = \{\psi_p\}_{p \in \mathbb{N}^{\mathbb{N}}}$ と添字付けることができる．そのような添字付けの下で，$\mathbb{N}^{\mathbb{N}}$ 上に以下のような部分 2 項演算を定義することができる．

$$p \cdot x \downarrow = y \iff x \in \mathrm{dom}(\psi_p) \ \&\ \psi_p(x) = y.$$

もちろん，添字付けの方法によって，この部分マグマ $(\mathbb{N}^{\mathbb{N}}, \cdot)$ の性質は異なる．しかし，極めて自然で構成的な $\mathcal{C}$ の標準的添字付けが存在することが知られており，それを利用すると，$(\mathbb{N}^{\mathbb{N}}, \cdot)$ が部分結合子代数をなすことを示せる．この部分結合子代数 $(\mathbb{N}^{\mathbb{N}}, \cdot)$ は**クリーネの第 2 代数** (*Kleene's second algebra*) と呼ばれる．

**例 2.25** (スコットのグラフモデル)**.** 2 点集合 $\{0, 1\}$ に $\varnothing, \{1\}, \{0, 1\}$ を開集合とする位相を入れた空間を $\mathbb{S}$ と書く．空間 $\mathbb{S}$ は連結かつ距離化不可能な 2 点空間であり，**シエルピンスキ空間** (*Sierpiński space*) の名で知られ，計算可能性理論においては非常に重要な空間のひとつである．この可算直積空間 $\mathbb{S}^{\mathbb{N}}$ は普遍第二可算 $T_0$ 空間の一例である．つまり，任意の第二可算 $T_0$ 空間は $\mathbb{S}^{\mathbb{N}}$ に位相的に埋め込める．この空間 $\mathbb{S}^{\mathbb{N}}$ の特性として，$\mathbb{S}^{\mathbb{N}}$ 上の部分連続関数は全域連続関数に拡張可能である，というものがある．例 2.24 のように，$\mathbb{S}^{\mathbb{N}}$ 上の連続関数全体の添字付け $\{\psi_p\}_{p \in \mathbb{S}^{\mathbb{N}}}$ を与えて，$\mathbb{S}^{\mathbb{N}}$ 上に以下のような全域 2 項演算を定義できる．

$$p \cdot x \downarrow = y \iff \psi_p(x) = y.$$

これは（全域）マグマをなすが，さらに適切な添字付けによっては，$(\mathbb{S}^{\mathbb{N}}, \cdot)$ は部分結合子代数をなす．ここで，2 項演算 $\cdot$ が全域であるような部分結合子代数は**全域結合子代数** (*total combinatory algebra*) と呼ばれる．この全域結合子代数 $(\mathbb{S}^{\mathbb{N}}, \cdot)$ は，**スコットのグラフモデル** (*Scott's graph model*) として知られる．

**例 2.26** (可測関数の代数)．部分結合子代数は，不連続関数を含む関数族，たとえば可測関数の記述的階層から得ることもできる．こう聞くと，ルベーグ可測関数や**ボレル可測関数** (*Borel measurable function*) の族を思い浮かべるかもしれないが，これらは圏論的振る舞いが悪く，部分結合子代数とはならない．圏論的に良い性質を持つのは，たとえばボレル可測関数の「部分関数」版である．ただし，それは部分ボレル可測関数ではなく，**部分 $\underset{\sim}{\mathbf{\Pi}}^1_1$-可測関数** (*partial $\underset{\sim}{\mathbf{\Pi}}^1_1$-measurable function*) である．ここで，記述集合論におけるススリンの定理より（完備可分距離空間上の）全域 $\underset{\sim}{\mathbf{\Pi}}^1_1$-可測関数はボレル可測関数と一致する．部分 $\underset{\sim}{\mathbf{\Pi}}^1_1$-可測関数全体の族には標準的な添字付け $\{\psi_p\}_{p\in\mathbb{N}^{\mathbb{N}}}$ が存在し，以下の演算によって部分結合子代数をなす．

$$p\cdot x\downarrow = y \iff x\in \mathrm{dom}(\psi_p)\ \&\ \psi_p(x)=y.$$

この概念は，計算可能性理論においては**超算術的還元** (*hyperarithmetical reduction*) の名で 1950 年代後半から深く研究され始め，特に 1960 年代以降は長らく計算可能性理論の中心的な研究対象であった．1970 年代には，この概念を抽象化した**スペクター点類** (*Spector pointclass*) という概念が考案され，任意のスペクター点類に対応する関数族から部分結合子代数を構成することができる．

## §3. ラムダ計算，不動点，再帰定理

これから，いかなる部分結合子代数の中でも “計算” を展開できることを示す．つまり，あらゆる計算を，積の適用という代数的演算の組合せで表せる，ということを見る．より正確には，次の概念を考える．

**定義 3.1.** 部分マグマ $M$ 上の部分関数 $f\colon\subseteq M^n\to M$ が $M$ で**実現可能** (*realizable*) であるとは，ある $a\in M$ が存在して，任意の $x_1,\dots,x_n\in M$ について，$f(x_1,\dots,x_n)\downarrow$ ならば $ax_1\dots x_n\downarrow = f(x_1,\dots,x_n)$ が成り立つことを意味する．

環のような代数構造が与えられると，我々は多変数多項式を考えることができる．多変数多項式とは $6x^5y^2+3x^2y^4+x^3+9$ のようなものである．しかし多項式を扱うには，和と積という 2 つの演算が必要であるが，我々の扱う代数には 1 つの演算しかないから，考えるものは単項式である．多変数単項式とは $4x^3y^2z^4$ のようなものであり，つまり $4xxxyyzzzz$ のことである．このように多変数単項式を整理された形で書けるのは，積が結合的かつ可換であるときのみである．たとえば，文字列 $xy(yxz)yay(zx)x$ は結合律と可換性によって $axxxxyyyyzz$ あるいは $ax^4y^4z^2$ と整理される．しかし，命題 2.21 で見たように，我々の代数は結合律も可換性も満たさない．つまり，我々の代数における多変数単項式（単に項と呼ぶ）は，$axy(yxz)yy(zx)x$ のように整理されていない文字列であり，$M$ が半群の場合には，これはつまり変数を含み得る有限語である．結合律を満たさない場合には，単なる有限語ではなく括弧をつけて演算の適用順序を明示する必要がある．たとえば，元 $a,b,c\in M$ と変数 $x,y,z$ について $a(yz)zbx$ であるとか $a(bz)(yc)$ のようなものが項 (3 変数単項式) である．それでは，部分結合子代数で一体どのような項が実現可能であるだろうか．

**定義 3.2.** $(M, *)$ を部分マグマとする．$M$ 上の**項** (*term*) を以下のように帰納的に定義する．

1. 変数 $x, y, z, \ldots$ は項である．
2. 各元 $a \in M$ は項である．
3. $P$ と $Q$ が項ならば $PQ$ も項である．

部分マグマ $M$ が**結合子完全** (*combinatory complete*) とは，$M$ 上の任意の項 $t(x_1, \ldots, x_{n+1})$ に対して，ある元 $a_t \in M$ が存在し，任意の $v_1, \ldots, v_{n+1} \in M$ について以下が成立する．

$$a_t v_1 \ldots v_n \downarrow \text{ かつ } a v_1 \ldots v_{n+1} \simeq t(v_1, \ldots, v_{n+1}).$$

まず，明らかに結合子完全な部分マグマは部分結合子代数をなす．なぜなら，項 $k(x, y) = x$ と $s(x, y, z) = xz(yz)$ を考えれば，結合子完全性より $a_k, a_s \in M$ を得るが，$\mathtt{k} := a_k$ および $\mathtt{s} := a_s$ とすれば，これは部分結合子代数の条件を満たす．つまり，部分結合子代数は特定の 2 つの項 $k, s$ に対してのみ結合子完全性の条件を満たす部分マグマとして定義されていた．しかし，実は，この 2 つの項だけ考えれば十分である，ということを見ていこう．つまり，部分結合子代数ならば常に結合子完全であることを示せる．

**定理 3.3.** 部分マグマ $M$ について，$M$ が部分結合子代数であることと $M$ が結合子完全であることは同値である．

この証明のために，部分結合子代数の中で**ラムダ計算** (*lambda calculus*) に相当するものを展開できることを示そう．ラムダ計算において，記号 $\lambda x.f(x)$ によって $x$ を入力すると $f(x)$ を出力する関数を表す．これだけを聞くと，単なる自明な記法の話に過ぎないと思うかもしれない．実際，余談であるが，筆者は学生時代に初めてラムダ計算という概念を聞いて，てっきり数学的には特に中身のない単なる記法の話だと思い込み，長年ラムダ計算を勉強することはおろか，そもそもラムダ計算の理論などというものがあり得るということを想像すらできなかった．が，実はこれだけのことから想像以上に深いラムダの数学的理論が展開される．

ラムダ計算の理論はさておき，ここではラムダ記法を用いて後の議論のアイデアを説明する．まず左単位元 $\mathtt{i}$ は恒等関数 $\lambda x.x$ を表し，$\mathtt{k}y$ は常に $y$ を出力する定数関数 $\lambda x.y$ を意図していたことを思い出そう．また，記号的には少し不正確であるが，$\mathtt{s}$ が意図する関数適用をラムダ記法を用いて説明すると，$\mathtt{s}(\lambda x.f(x))(\lambda x.z(x))$ とは $a$ を入力すると $f(a)$ に $z(a)$ を適用したものを出力する $\lambda x.f(x)(z(x))$ のことを意図していた．このアイデアを用いて，部分結合子代数の中でいわゆる**ラムダ抽象** (*lambda abstraction*) に相当する概念が取り扱えることを示す．

**定義 3.4.** 部分結合子代数 $M$ 上の項 $P$ と変数 $x$ が与えられたとき，項 $\Lambda x.P$ を以下によって帰

納的に定義する．

$$\begin{aligned} \lambda x.x &\equiv \mathtt{i} \\ \lambda x.y &\equiv \mathtt{k}y && (y \neq x \text{ が変数であるか } y \in M \text{ のとき}) \\ \lambda x.(PQ) &\equiv \mathtt{s}(\lambda x.P)(\lambda x.Q) \end{aligned}$$

ここで $\mathtt{i}$ は命題 2.20 のような $M$ の左単位元とする．

以後，$\lambda x.(\lambda y.(\lambda z.P))$ などは $\lambda xyz.P$ のように省略する．たとえば，

$$\lambda xy.x = \lambda x.(\lambda y.x) = \lambda x.(\mathtt{k}x) = \mathtt{s}(\lambda x.\mathtt{k})(\lambda x.x) = \mathtt{s}(\mathtt{kk})\mathtt{i}$$

となる．項 $\lambda x.P$ に含まれる変数は，項 $P$ に含まれる変数から $x$ を除いたものである．よって，$P$ が $x$ のみを変数として含むならば，$\lambda x.P$ は変数を含まない．定義 2.19 より，$\mathtt{k}a\downarrow$ および $\mathtt{s}ab\downarrow$ が成立しているから，このとき $\lambda x.P$ は $M$ の元を定義することが示される．

項 $P$ が与えられたとき，$P[Q/x]$ によって，$P$ の変数 $x$ に項 $Q$ を代入した結果を表す．上の議論をより一般化すると，$y_0, \ldots, y_n$ を $\lambda x.P$ に含まれる自由変数のリストとすれば，任意の $a_0, \ldots, a_n \in M$ について，$(\lambda x.P)[a_0, \ldots, a_n/y_0, \ldots, y_n]$ は $M$ の元を表す．

以下，重要な性質として，$\lambda x.P$ は本物のラムダ計算のような働きをするということを示す．つまり，部分結合子代数が与えられれば，内部でラムダ計算っぽいものを展開できてしまうということである．

**補題 3.5.** $P$ と $Q$ を $M$ 上の項とする．このとき，$(\lambda x.P)Q \equiv P[Q/x]$ が成り立つ．

*Proof.* $P$ の構造に関する帰納法による．$P = x$ のとき，

$$(\lambda x.P)Q = \mathtt{i}Q = Q = P[Q/x].$$

続いて，$P = y$ が $y \neq x$ なる変数であるか，$y \in M$ であるとき，

$$(\lambda x.P)Q = \mathtt{k}yQ = y = P[Q/x].$$

最後に，$P = RS$ である場合，帰納的に $(\lambda x.R)Q \equiv R[Q/x]$ かつ $(\lambda x.S)Q \equiv S[Q/x]$ が成り立っていると仮定する．このとき，

$$(\lambda x.P)Q = \mathtt{s}(\lambda x.R)(\lambda x.S)Q \equiv (\lambda x.R)Q((\lambda x.S)Q) \equiv R[Q/x]S[Q/x] = (RS)[Q/x] = P[Q/x].$$

以上より，帰納法によって，求める性質が得られる． □

**注意.** ラムダ記法の略記に対する補題 3.5 の適用について注意すると，$i \leqslant n$ のとき

$$(\lambda x_1 x_2 \ldots x_n.P)Q_1Q_2 \ldots Q_i \simeq \lambda x_{i+1} \ldots x_n.P[Q_1/x_1] \ldots [Q_i/x_i]$$

となる．なぜなら，$\lambda x_1 x_2 \ldots x_n.P = \lambda x_1.(\lambda x_2 \ldots x_n.P)$ であるから，これに $Q_1$ を右から掛けると，補題 3.5 より $(\lambda x_1 x_2 \ldots x_n.P)Q_1 = (\lambda x_1.(\lambda x_2 \ldots x_n.P))Q_1 \simeq \lambda x_2 \ldots x_n.P[Q_1/x_1]$ となる．次に $Q_2$ を右から掛

け，同様の変形を繰り返せばよい．具体例として，たとえば $(\Lambda xyz.vyy(ux)zx)(aa)(cbc)$ は $vyy(ux)zx$ に現れる $x$ の部分に $aa$ を代入し $y$ の部分に $cbc$ を代入した結果となる．つまり，$(\Lambda xyz.vyy(ux)zx)(aa)(cbc) = vyy(ux)zx[aa/x][cbc/y] = v(cbc)(cbc)(u(aa))z(aa)$ を得る．

*Proof (*定理 *3.3).* 簡単のために 1 変数の項 $t(x)$ を考える．定義 3.4 の直後に注意したように，$a_t \equiv \Lambda x.t(x) \downarrow$ である．補題 3.5 より，任意の $v \in M$ について $a_t v \simeq t(v)$ を得る．よって，定理 3.3 は示された． □

**命題 3.6.** 任意の部分結合子代数 $M$ について，次を満たす $\mathtt{pair}, \pi_0, \pi_1 \in M$ が存在する．

$$\mathtt{pair}\, ab \downarrow, \quad \pi_0(\mathtt{pair}\, ab) = a, \quad \pi_1(\mathtt{pair}\, ab) = b$$

*Proof.* $\mathtt{pair} = \Lambda xyz.zxy$, $\pi_0 = \Lambda z.z(\Lambda xy.x)$, $\pi_1 = \Lambda z.z(\Lambda xy.y)$ によって定義する．このとき，補題 3.5 を利用すると，$\Lambda$ は代入と解釈できるから，$\mathtt{pair}\, ab = \Lambda z.zab$ となる．したがって，補題 3.5 より，

$$\pi_0(\mathtt{pair}\, ab) = (\Lambda z.z(\Lambda xy.x))(\Lambda z.zab) = (\Lambda z.zab)(\Lambda xy.x) = (\Lambda xy.x)ab = a,$$
$$\pi_1(\mathtt{pair}\, ab) = (\Lambda z.z(\Lambda xy.y))(\Lambda z.zab) = (\Lambda z.zab)(\Lambda xy.y) = (\Lambda xy.y)ab = b.$$

よって，主張は示された． □

以後，$\mathtt{pair}\, ab$ のことを $\langle a, b\rangle$ と書くことにする．このとき，$M$ の任意の有限列 $(a_i)_{i\leqslant n}$ は，次のように 1 つの要素としてコードできる．

$$\langle a_0, a_1, a_2, \ldots, a_n\rangle := \langle\langle\langle a_0, a_1\rangle, a_2\rangle, \ldots\rangle, a_n\rangle.$$

それでは，部分結合子代数が与えられれば，自然数上の計算論が展開できることを見ていこう．このためには，部分結合子代数 $M$ の中で自然数をコードする必要がある．これはたとえば次のようにコードできる．

**定義 3.7.** $M$ を部分結合子代数とする．$\mathtt{true}, \mathtt{false} \in M$ および各自然数 $n \in \mathbb{N}$ について $\underline{n} \in M$ を以下のように定義する．

$$\mathtt{true} = \Lambda xy.x \qquad \mathtt{false} = \Lambda xy.y$$
$$\underline{0} = \langle \mathtt{true}, \mathtt{i}\rangle \qquad \underline{n+1} = \langle \mathtt{false}, \underline{n}\rangle$$

この自然数のコード方法を理解しかねるという人もいるかもしれないが，取り扱いの容易さから，このコーディングを利用する．しかし，「自然数を実装できる」という点のみが重要なのであって，自然数の具体的な実装方法は何でもよいし，その意味を問うことはあまり生産的ではない．文字列によって自然数をコードする方法がいくらでもあるように，部分結合子代数 $M$ の中で自然数をコードする方法は唯一ではない．

**命題 3.8.** 次のような元 $\mathtt{succ}, \mathtt{pred}, \mathtt{iszero} \in M$ が存在する．

$$\mathtt{succ}\,\underline{n} = \underline{n+1} \qquad \mathtt{pred}\,\underline{n} = \underline{n \dot{-} 1}$$
$$\mathtt{iszero}\,\underline{n} = \begin{cases} \mathtt{true} & \text{if } n = 0 \\ \mathtt{false} & \text{if } n \neq 0 \end{cases}$$

ここで $\dot{-}$ は $x \dot{-} y = \max\{0, x - y\}$ によって定義される部分的減法である．

*Proof.* まず，$\mathtt{succ} = \Lambda x.\langle \mathtt{false}, x\rangle$ によって定義する．このとき，補題 3.5 より，$\mathtt{succ}\,\underline{n} = \langle \mathtt{false}, \underline{n}\rangle = \underline{n+1}$ である．$\mathtt{iszero} = \pi_0$ が条件を満たすことは明らかである．最後に，$\mathtt{pred}$ を定義するために，補題 3.5 から以下の式を得られることに注意する．

$$\langle a, b\rangle \mathtt{true} = (\Lambda xyz.zxy)ab(\Lambda xy.x) = (\Lambda z.zab)(\Lambda xy.x) = (\Lambda xy.x)ab = a$$
$$\langle a, b\rangle \mathtt{false} = (\Lambda xyz.zxy)ab(\Lambda xy.y) = (\Lambda z.zab)(\Lambda xy.y) = (\Lambda xy.y)ab = b.$$

$\mathtt{pred} = \Lambda x.\langle \underline{0}, \pi_1 x\rangle(\mathtt{iszero}\,x)$ と定義する．このとき，上の式と補題 3.5 より，

$$\mathtt{pred}(\underline{0}) = \langle \underline{0}, \pi_1\,\underline{0}\rangle(\mathtt{iszero}\,\underline{0}) = \langle \underline{0}, \mathtt{i}\rangle \mathtt{true} = \underline{0}$$
$$\mathtt{pred}(\underline{n+1}) = \langle \underline{0}, \pi_1\,\underline{n+1}\rangle(\mathtt{iszero}\,\underline{n+1}) = \langle \underline{0}, \underline{n}\rangle \mathtt{false} = \underline{n}.$$

よって，主張は示された． □

次に，部分結合子代数におけるある種の不動点定理を示そう．関数 $f : X \to X$ の**不動点** (*fixed point*) とは，$f(x) = x$ なる $x \in X$ のことである．しかし，ここでは $X$ は関数空間 $[A \to B]$ であると考えると都合がよい．つまり，関数 $f : [A \to B] \to [A \to B]$ の不動点とは $g = f(g)$ を満たす関数 $g : A \to B$ のことである．$f$ の不動点のことを $\mathtt{fix}\,f$ と書くとすると，$\mathtt{fix}\,f = f(\mathtt{fix}\,f)$ を満たす．つまり

$$(\forall a \in A)\ f(\mathtt{fix}\,f)(a) = (\mathtt{fix}\,f)(a)$$

を満たすということである．このような不動点 $\mathtt{fix}\,f$ のようなものが常に存在する，というのが次の定理である．

**定理 3.9.** 次のような元 $\mathtt{fix} \in M$ が存在する．任意の $f, a \in M$ に対して，

$$\mathtt{fix}\,f \downarrow \qquad \mathtt{fix}\,fa = f(\mathtt{fix}\,f)a.$$

*Proof.* $\mathtt{r} = \Lambda xyz.x(yy)z$ とし，$\mathtt{fix} = \Lambda g.\mathtt{r}g(\mathtt{r}g)$ と定義する．このとき，

$$\mathtt{fix}\,f = \mathtt{r}f(\mathtt{r}f) = (\Lambda xyz.x(yy)z)f(\mathtt{r}f) = (\Lambda yz.f(yy)z)(\mathtt{r}f) = \Lambda z.f(\mathtt{r}f(\mathtt{r}f))z$$

である．いま，$\mathtt{r}$ は自由変数を持たず，よって，$f(\mathtt{r}f(\mathtt{r}f))z$ は $z$ のみを自由変数に含む．一方，定義 3.4 の直後に述べたように，もし $P$ が $z$ のみを変数に含むならば，$\Lambda z.P \downarrow$ である．よって，

$\mathtt{fix}\, f \downarrow$ を得る．また，

$$\mathtt{fix}\, fa = (\mathtt{r}f(\mathtt{r}f))a = (\Lambda z.f(\mathtt{r}f(\mathtt{r}f))z)a = f(\mathtt{r}f(\mathtt{r}f))a = f(\mathtt{fix}\, f)a$$

となるから定理は示された． □

不動点定理 3.9 の重要な帰結の 1 つが，**クリーネの再帰定理** (*Kleene's recursion theorem*) である．まず，定理 3.9 から次の系が得られる．

**系 3.10.** $M$ を部分結合子代数とし，$Q$ を $e$ のみを自由変数とする $M$ 上の項とする．このとき，次を満たす項 $r \in M$ が存在する．

$$(\forall a \in M)\ ra \equiv Q[r/e]a.$$

*Proof.* $f = \Lambda e.Q$ とする．ここで，$Q$ に含まれる自由変数は $e$ のみであるから，定義 3.4 の直後の議論より，$f \downarrow \in M$ と考えてよい．よって，不動点定理 3.9 の証明の $\mathtt{fix}$ について，$\mathtt{fix}\, f \downarrow = r$ となる $r \in M$ を得る．いま，任意の $a \in M$ について，

$$ra = fra = (\lambda e.Q)ra \equiv Q[r/e]a$$

であるから，主張は示された． □

ここで，$r = Q[r/e]$ となるとは限らないことに注意する．系 3.10 を例 2.23 のクリーネの第一代数 $K_1$ で解釈した結果は，クリーネの再帰定理として知られる．クリーネの第 1 代数を用いているときは，それを明示するために $p \cdot x$ ではなく $\{\!\{p\}\!\}(x)$[*1]と書くことにする．つまり，$\{\!\{p\}\!\}$ はコンピュータ・プログラム $p$ が表す関数であり，$\{\!\{p\}\!\}(x)$ はプログラム $p$ に $x$ を入力した結果の値である．

**定理 3.11** (クリーネの再帰定理)**.** 任意の計算可能関数 $q\colon \mathbb{N} \to \mathbb{N}$ について，次を満たす $r \in \mathbb{N}$ が存在する．

$$\{\!\{r\}\!\} \equiv \{\!\{q(r)\}\!\}$$

*Proof.* 部分結合子代数として，例 2.23 のクリーネの第一代数 $K_1$ を取る．このとき，$q$ は計算可能であるから，ある $d \in \mathbb{N}$ について，$q = \{\!\{d\}\!\}$ となる．$Q = de$ とすると，系 3.10 より，任意の $a \in K_1$ について，$ra \equiv Q[r/e] \equiv dra$ なる $r \in K_1$ を得る．$K_1$ における積演算の定義より，

$$(\forall a \in \mathbb{N})\ \{\!\{r\}\!\}(a) \equiv \{\!\{\{\!\{d\}\!\}(r)\}\!\}(a) \equiv \{\!\{q(r)\}\!\}(a)$$

となるから，定理は示された． □

[*1] 計算可能性理論の初期に多用されたクリーネ記法では $\{p\}(x)$ と書かれるが，$\{p\}$ と書くと単元集合と紛らわしいので本稿では用いない．

**解説.** クリーネの再帰定理の直感的な説明を与えよう．固定した未知変数 $I$ を使いながら，コンピュータ・プログラム $Q(I)$ を書いている，というシチュエーションを想定しよう．我々はその段階では $I$ が何であるかは知らないが，とにかく $I$ は自由に使えるので，プログラム内部に「プログラム $I$ に $n$ を入力した計算を実行せよ」などの命令を書き込むことができる．

さて，クリーネの再帰定理 3.11 における $r$ を取ってきて，$Q = Q(r)$ としよう．つまり，先ほど我々の書いたプログラムの中で $I$ と書かれている部分を $r$ で上書きしたものが新しいプログラム $Q$ である．すると，再帰定理より，コード $r$ のプログラムを実行したものと $Q$ を実行したものの計算結果は等しい．したがって，$Q$ のプログラム内部に書かれている「プログラム $I$ に $n$ を入力した計算を実行せよ」という命令は「プログラム $Q$ に $n$ を入力した計算を実行せよ」という命令に等しい．

これが意味していることは何だろうか．我々はプログラム $Q$ を書いている途中段階では，最終的な $Q$ がどうなるかは知らないし，無限の自由度がある．それにも関わらず，我々が途中で書いた「プログラム $I$ に $n$ を入力した計算を実行せよ」と言う命令の意味は，常に「最終的な $Q$ に $n$ を入力した計算を実行せよ」を表す．つまり，我々はあたかも「最終的に書き上げる予定のプログラムが何であるか既に知っているかの如く」プログラムを記述することができるのである．

そういうわけで，以後，プログラム $I$ すなわち『私』すなわち自己に言及したプログラムは自由に記述してよい．特に自身のソースコードを出力するプログラムは，コンピュータ・プログラミングの文脈では**クワイン** (*Quine*) としてよく知られており，様々なプログラミング言語での実装例を見つけることができるだろう．

さて，クリーネの再帰定理は数学的には一見単純であるが，それ故に強力である．基礎的なレベルから研究の最先端に至るまで，極めて広範な応用を持つ．計算可能性理論の入門的内容における最も重要な定理と言っても過言ではないだろう．

ここでは，クリーネの再帰定理の簡単な応用として，原始再帰法の実装を行う．原始再帰法とは，関数 $f, g$ から次のような関数 $h$ を作る操作である．

$$\begin{cases} h(0, x) = g(x) \\ h(n+1, x) = f(n, x, h(n, x)) \end{cases}$$

この関数 $h$ を $\mathtt{rec}\, gf$ として表そう．この原始再帰作用素 $\mathtt{rec}$ は，不動点として実装できる．

**命題 3.12.** 任意の部分結合子代数 $M$ について，次のような元 $\mathtt{rec} \in M$ が存在する．

$$\mathtt{rec}\, gf\underline{0} = g$$
$$\mathtt{rec}\, gf\underline{n+1} = f\underline{n}(\mathtt{rec}\, gf\underline{n})$$

*Proof.* まず，命題 3.8 の証明で見たように，

$$\langle a, b\rangle \mathtt{iszero}\, n = \begin{cases} a & \text{if } n = \underline{0} \\ b & \text{if } n \neq \underline{0} \end{cases}$$

が成り立つことに注意する．このため，$\langle a, b\rangle \mathtt{iszero}\, n$ を $\mathtt{if}\ n\ \mathtt{iszero\ then}\ a\ \mathtt{else}\ b$ と表す．

証明のアイデアとしては，$n = 0$ ならば $hn = g$ であり，さもなくば $hn = f(\mathtt{pred}\, n)(h(\mathtt{pred}\, n))$ であるような $h$ を作ればよい．ただし，$h$ は $f$ と $g$ に依存するので，$h = egf$ という形である．具体的には，次の項 $Q$ を考える．

$$\Lambda gfn.\mathtt{if}\ n\ \mathtt{iszero\ then}\ g\ \mathtt{else}\ f(\mathtt{pred}\, n)(egf(\mathtt{pred}\, n)).$$

項 $Q$ は $e$ のみを自由変数に持つので, 系 3.10 より, $e$ を自己への言及と解釈するような $\mathtt{rec} \in M$ が存在する．つまり，任意の $a \in M$ について $\mathtt{rec}\, a = Q[\mathtt{rec}/e]a$ が成立する．このとき，

$$\mathtt{rec}\, g f \underline{n} = \mathtt{if}\ \underline{n}\ \mathtt{iszero\ then}\ g\ \mathtt{else}\ f(\mathtt{pred}\, \underline{n})(\mathtt{rec}\, g f (\mathtt{pred}\, \underline{n}))$$

であるから，明らかに

$$\mathtt{rec}\, g f \underline{0} = g, \qquad \mathtt{rec}\, g f \underline{n+1} = f \underline{n} (\mathtt{rec}\, g f \underline{n})$$

が導かれる．よって求める性質が示された． □

最後に，任意の部分結合子代数で **$\mu$-最小化** ($\mu$-*minimization*) を実装できることを示そう．$\mu$-最小化とは，$x$ を変数とする式 $P(x)$ が与えられたとき，$P(x)$ を満たす $x$ が存在するならば，そのような最小の $x$ を返す演算 $\mu x.P(x)$ である．

**定理 3.13.** 関数 $f\colon \mathbb{N}^{n+1} \to \mathbb{N}$ が実現可能ならば，$h(\bar{x}) = \mu y.[f(\bar{x}, y) = 0]$ によって定義される関数 $h\colon \mathbb{N}^n \to \mathbb{N}$ も実現可能である．

*Proof.* $e$ を変数とする項 $Q$ を次のように定義する．

$$Q := \Lambda \bar{x} y.\mathtt{if}\ f\bar{x}y\ \mathtt{iszero\ then}\ y\ \mathtt{else}\ e\bar{x}\underline{y+1}.$$

再帰定理 (系 3.10) より，$r\bar{x}y \simeq Q[r/e]\bar{x}y$ となる $r$ が存在する．つまり，

$$r\bar{x}y \simeq \mathtt{if}\ f\bar{x}y\ \mathtt{iszero\ then}\ y\ \mathtt{else}\ r\bar{x}\underline{y+1}.$$

このとき，$h = \Lambda \bar{x}.r\bar{x}0$ とすれば，帰納法によって，$h$ が求めるものであることを容易に示せる． □

## § 4. 部分結合子代数の具体例

本節では，部分結合子代数の具体例について見ていこう．ここでポイントとなるのが，例 2.24 や例 2.26 のように，位相的概念や可測性概念から，計算的な成分を抽出することができ，それが部分結合子代数の構造を持つという点である．このように，一見，計算概念から程遠い，たとえば無限集合論的概念からさえ，計算的成分を抽出して計算論的な分析を行うことができる．

### 4.1. 万能マシンから部分結合子代数へ

最も基本的な部分結合子代数は，チューリングマシンなどの計算モデルから得ることができる．本節では，その証明のために必要な本質を抽出し，一般に，計算モデルから無関係な関数族から部分結合子代数を得る方法を探る．まず，計算モデルから部分結合子代数を得るために重要な概念は，万能チューリングマシンとして知られるものである．

チューリングマシンという計算モデルは，物理的に見れば，計算する関数毎にマシンを作っている．したがって，100 個のプログラムを書くということは，100 個のチューリングマシンを用意するということである．しかし，我々の世代のコンピュータは，「たった 1 つのマシン」の中で，プログラムを書くことによって，全ての計算可能関数を実装できる[*2]．このようなコンピュータは，数学的には万能チューリングマシンと呼ばれるものである．

1930 年代，そのようなマシンがまだ存在しなかった時代，チューリングは万能マシンの概念を定式化し，理論的にその存在を示した．このようにして，我々の世代が用いているようなコンピュータの到来をチューリングは予言したのである．これを数学的に抽象化して表すと，以下のようなものである．

**定義 4.1. 万能チューリングマシン** (*universal Turing machine*) とは，次のような計算可能部分関数 $M :\subseteq \mathbf{P} \times \mathbb{N}^n \to \mathbb{N}$ である：任意の計算可能部分関数 $f :\subseteq \mathbb{N}^n \to \mathbb{N}$ に対して，あるプログラム $p \in \mathbf{P}$ が存在して，$M(p, x) \equiv f(x)$ が成立する，というものである．

プログラムと自然数を同一視，つまり $\mathbf{P} \simeq \mathbb{N}$ と考え，また対関数によって複数の自然数をまとめてしまえば $\mathbb{N}^n \simeq \mathbb{N}$ であるから，$M$ と $f$ は共に自然数上の部分関数と思ってよい．

現代的に言えば，$M$ が我々の眼前にあるコンピュータであり，$p$ というものは，マシン $f$ の計算をシミュレートするプログラムである．我々のコンピュータ $M$ の中でプログラム $p$ に文字列 $x$ を入力すれば，$M(p, x)$ という計算結果が得られる．現代文明の恩恵を享けている全ての人は，次の定理を体で知っている．

**定理 4.2** (チューリング)**.** 万能チューリングマシンが存在する．

この概念を一般化しよう．いま，集合 $P$ と $N$ が与えられているとする．上述のように，$P$ をプログラムの集合，$N$ を入出力の集合と考え，さらに $P^m \simeq P$ および $P^n \times N^m \simeq N$ が成立すると仮定する．以後，標準的な対関数 $\langle \cdot, \ldots, \cdot \rangle : P^n \times N^m \simeq N$ と射影関数 $\pi_i$ は固定されているとする．たとえば，$P$ と $N$ が $\mathbb{N}$ または $\mathbb{N}^{\mathbb{N}}$ と同一視できるなら，このような条件は満たす．

**定義 4.3.** 集合 $N$ 上の部分関数の族 $\mathcal{F}$ および集合 $P$ が与えられているとする．部分関数 $M :\subseteq P \times N \to N$ が $P$ **を介して** $\mathcal{F}$**-万能**とは，次の条件を満たすことを意味する：$M \in \mathcal{F}$ であり，任意の部分関数 $f \in \mathcal{F}$ に対して，ある $p \in P$ が存在して，任意の $x \in N$ について $M(p, x) \equiv f(x)$ を満たす．

チューリングの定理 4.2 より，$\mathbb{N}$ 上の部分関数全体の族は $\mathbb{N}$ を介した万能関数を持ち，つまり，それが万能チューリングマシンのことである．上の定義は，一般に関数族 $\mathcal{F}$ に対する万能マシンに相当する概念を定義するものである．このような一般化は，記述集合論などの分野で頻繁に用いられる．たとえば，記述集合論において，$P$ を介して $\mathcal{F}$-万能な部分関数が存在するとき，部分

[*2] あるいは，「たった 1 つのマシン」にソフトウェアをインストールすることによって，無数のアプリケーションを実行できる．

関数の族 $\mathcal{F}$ は **$P$-媒介化** ($P$-*parametrized*) されているという．

以後，恒等関数，$P \subseteq N$ に値を取る定数関数，対関数 $\langle \cdot, \cdot \rangle$ と射影関数 $\pi_0, \pi_1$ の組合せで定義可能な $N$ 上の関数を**自明な関数**と呼ぶことにする．いま，集合 $X$ 上の部分関数の族 $\mathcal{F}$ が次のような良い性質を持つと仮定する．

1. $P$ を介して $\mathcal{F}$-万能な部分関数 $M$ が存在する．
2. 任意の $f \in \mathcal{F}$ と自明な関数 $g :\subseteq N \to N$ に対して，$f \circ g \in \mathcal{F}$ である．

このような関数族 $\mathcal{F}$ に対しては，万能マシン $M$ に少し修正を施せば，さらに良い性質を持った万能マシンに改良できる，というのが次の補題である．

**補題 4.4.** $\mathcal{F}$ を上述の前提を満たす関数族とする．このとき，次のような $\mathcal{F}$-万能関数 $U\colon P \times N \to N$ が存在する．ある自明な関数 $s\colon P \times N \to P$ が存在して，任意の $p \in P$ と $z, x_1, \ldots, x_n \in N$ について以下が成立する．

$$U(p, \langle z, x_1, \ldots, x_n \rangle) \equiv U(s(p, z), \langle x_1, \ldots, x_n \rangle)$$

*Proof.* 前提 (1) より，$\mathcal{F}$ は万能関数 $M :\subseteq P \times N \to N$ を持つ．仮定より $N$ には対関数と対応する射影が存在するので，このとき，$U(p, x) \equiv M(\pi_0(p), \langle \pi_1(p), x \rangle)$ と定義する．明らかに $U(\langle a, b \rangle, x) \equiv M(a, \langle b, x \rangle)$ が成立する．前提 (2) より $\mathcal{F}$ は自明な関数との合成で閉じているので，$U \in \mathcal{F}$ である．

まず，$U$ が万能であることを確認しよう．任意の $f \in \mathcal{F}$ について，前提 (2) より $f \circ \pi_1 \in \mathcal{F}$ なので，ある $p \in P$ が存在して $M(p, x) \equiv f \circ \pi_1(x)$ となる．特に $M(p, \langle b, x \rangle) \equiv f(x)$ である．よって，$U(\langle p, b \rangle, x) \equiv M(p, \langle b, x \rangle) \equiv f(x)$ を得る．適当に $b \in P$ を選べば，仮定より $\langle p, b \rangle \in P$ であるから，$U$ の万能性が示された．

また，$u(\langle p, z \rangle, x) \equiv U(p, \langle z, x \rangle)$ と定義すると，同様にして $u \in \mathcal{F}$ であるから，$M$ の万能性より，ある $q \in P$ が存在して $M(q, \langle x, y \rangle) \equiv u(x, y)$ が成立する．以上より，

$$U(p, \langle z, x \rangle) \equiv u(\langle p, z \rangle, x) \equiv M(q, \langle \langle p, z \rangle, x \rangle) \equiv U(\langle q, \langle p, z \rangle \rangle, x)$$

を得る．いま $s(p, z) \equiv \langle q, \langle p, z \rangle \rangle$ と定義すれば，これは自明な関数であり，補題の条件を満たす． □

この補題は，計算理論においては smn 定理あるいは**パラメータ定理** (*parameter theorem*) と呼ばれる．以後，補題のような $U$ に対して，$U(p, x)$ をしばしば $\{\!\{p\}\!\}(x)$ と略記する．この補題は，$\{\!\{p\}\!\} :\subseteq N^{n+1} \to N$ と $\lambda z.\{\!\{s(p, z)\}\!\}\colon N \to [\subseteq N^n \to N]$ を同一視できると述べている．つまり，パラメータ定理 (補題 4.4) は計算機科学における**カリー化** (*currying*) と呼ばれる操作に対応する．

さて，関数族 $\mathcal{F}$ から部分結合子を構成するためには，自明な関数との合成で閉じているだけでは不足で，自身との合成についても閉じている必要がある．つまり，以後は，

2a. 任意の自明な関数は $\mathcal{F}$ に属す．

2b. $f, g \in \mathcal{F}$ ならば $g \circ f \in \mathcal{F}$ である．

という性質を要求する．この前提 (1),(2a),(2b) の下で，パラメータ定理 (補題 4.4) から得られる万能関数を利用すると，以下の定理が成立する．

**定理 4.5.** $N$ 上の 2 項演算を $p \cdot n \equiv \{\!\{p\}\!\}(n)$ によって定義する．このとき，$(N, \cdot)$ は部分結合子代数をなす．

*Proof.* まず，前提 (2a) と (2b) より，$(N, \cdot)$ 上の項 $t(\bar{x})$ によって定義される $N$ 上の部分関数は $\mathcal{F}$ に属すことを示せる．たとえば $p \cdot q \cdot n \equiv \{\!\{\{\!\{p\}\!\}(q)\}\!\}(n) \equiv U(U(p,q),n)$ であるから，$\mathcal{F}$ の属す関数の合成として書けるため，合成で閉じているという前提より，これは $\mathcal{F}$ に属す．一般の項についても同様である．証明のアイデアは以下である．いま $t :\subseteq N^n \to N$ は $\mathcal{F}$ に属すので，補題 4.4 のカリー化を繰り返した結果 $t^* :\subseteq N \to N \to \cdots \to N \to N$ に対応する関数も $\mathcal{F}$ に属す．よって，万能性より $t^*$ に対応するコードを持ってくれば，これは結合子完全性を導く．ここでは，もう少し具体的に $\mathtt{s}$ と $\mathtt{k}$ を作ることにしよう．

まず $\mathtt{k}$-コンビネータを構成しよう．まず，射影 $\pi_0 : \langle u, v\rangle \mapsto u$ は自明なので，万能性より $\{\!\{i\}\!\}(\langle u, v\rangle) = u$ となるコード $i \in P$ が存在する．このとき，$s$ をパラメータ定理 4.4 の条件を満たす自明な関数とすると，$\{\!\{s(i,a)\}\!\}(v) \equiv \{\!\{i\}\!\}(\langle a, v\rangle) = a$ である．いま $a \mapsto s(i,a)\downarrow$ は自明であるから，万能性より，$\{\!\{\mathtt{k}\}\!\}(a) = s(i,a)$ となる $\mathtt{k} \in P$ が存在する．よって，

$$\{\!\{\{\!\{\mathbf{k}\}\!\}(a)\}\!\}(b) = \{\!\{s(i,a)\}\!\}(b) \equiv \{\!\{i\}\!\}(a,b) = a.$$

つづいて，$\mathtt{s}$-コンビネータを構成しよう．まず，項 $(f,x,a) \mapsto \{\!\{\{\!\{f\}\!\}(a)\}\!\}(\{\!\{x\}\!\}(a))$ は $\mathcal{F}$ に属すので，万能性より，そのコード $e \in P$ が存在する．パラメータ定理 4.4 を適用して，$\{\!\{s(e,f,x)\}\!\}(a) \equiv \{\!\{e\}\!\}(f,x,a)$ となる自明な関数 $s$ を得る．$(f,x) \mapsto s(e,f,x)$ は自明であるから，万能性よりそのコード $d \in P$ が存在する．再びパラメータ定理 4.4 より，$\{\!\{s(d,f)\}\!\}(x) \simeq \{\!\{d\}\!\}(f,x)$ を得る．このとき $f \mapsto s(d,f)$ は自明であるから，万能性よりそのコード $\mathtt{s} \in P$ が存在する．つまり，

$$\{\!\{\{\!\{\mathbf{s}\}\!\}(f)\}\!\}(x) \equiv \{\!\{s(d,f)\}\!\}(x) \equiv \{\!\{d\}\!\}(f,x) \equiv s(e,f,x)\downarrow$$

を得る．したがって，

$$\{\!\{\{\!\{\{\!\{\mathbf{s}\}\!\}(f)\}\!\}(x)\}\!\}(a) \equiv \{\!\{s(e,f,x)\}\!\}(a) \equiv \{\!\{e\}\!\}(f,x,a) \equiv \{\!\{\{\!\{f\}\!\}(a)\}\!\}(\{\!\{x\}\!\}(a)).$$

以上より，$(N, \cdot)$ が部分結合子代数であることが示された． □

特に，例 2.23 で挙げたクリーネの第一代数 $K_1$ が部分結合子代数であることが導かれる．

## 4.2. クリーネの第 2 代数

まずは，例 2.24 で述べた，クリーネの第 2 代数と呼ばれる，部分連続関数を 2 項演算とする部分結合子代数の詳細を述べていこう．余談であるが，このクリーネの第 2 代数に関しては，一般

再帰理論の誕生以前の古典的な構成的解析学に起源をもつものであるため，一般再帰理論の枠には含まないことが多い．

**■連続関数の計算モデル:** クリーネの第 2 代数を語るにあたって便利な概念は，**ストリーム** (*stream*) というデータ型に関する計算である．ストリームとは，次々に与えられるデータの奔流である．数学的には，

$$\mathtt{a}_0\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3\ldots\mathtt{a}_n\mathtt{a}_{n+1}\ldots$$

という無限に続く文字列と同一視できるが，実際には，時間経過に従って徐々に $\mathtt{a}_0, \mathtt{a}_1, \ldots$ というデータを流し込んでくるものと思う方が都合が良い．つまり，各時刻では，ストリームは，まだ $\mathtt{a}_0\mathtt{a}_1\ldots\mathtt{a}_k$ というような有限データしか配信していない．

ストリームは基本的には外部情報であるが，我々はストリーム型のデータを利用して計算を行うことが可能である．もう少し正確に議論するために，チューリングマシンのような計算モデルを考えよう．この計算モデルは，外部からのストリームを流し込むための専用テープを持つ．このテープは読込専用で，我々は書込不可能である．これが，ストリーム型のデータの入力テープである．

そうすると，我々もストリーム型のデータの出力を行いたい．この場合，我々はチューリングマシンの計算を停止させずに，延々と稼働させつづけ，文字列を

$$\mathtt{b}_0\mathtt{b}_1\mathtt{b}_2\mathtt{b}_3\ldots\mathtt{b}_n\mathtt{b}_{n+1}\ldots$$

というように，時間経過に沿って，次々に出力させていくこととなる．このためには，ストリーム型のデータの出力テープがあると便利である．ただし，一度配信したデータは，別の人物が既に受信しているかもしれないから，もう無かったことにはできない．この状況は，ストリーム出力テープは書込専用だが上書き不可能である，という特性によって定式化できる．

このストリーム入出力テープに加え，読込，書込，上書きが可能な作業用テープを合わせた計算モデルを考えよう．その他の部分は，通常のチューリングマシンと同一である．さて，この計算モデルにおいて，どのようなストリーム上の関数が計算可能であるだろうか．この計算モデルを直接取り扱ってもよいが，ストリーム計算のモデルは，非常に単純明快な数学的記述を持つ．

部分マグマ $(M,*)$ の元 $a,b\in M$ に対して，$b$ が $a$ を割り切るとは，$a=b*c$ となる $c\in M$ が存在することである．代数学の文脈では，このとき $b\mid a$ と書かれることが多いが，ここでは $b\sqsubseteq a$ と書くことにする．この関係 $\sqsubseteq$ を**整除関係**と呼ぶ．

ここでは集合 $A$ が生成する自由モノイド $(A^*,*)$ を考えよう (例 2.4)．つまり，アルファベット $A$ 上の有限文字列全体の集合 $A^*$ に，文字列結合 $\mathtt{a}_0\mathtt{a}_1\ldots\mathtt{a}_i * \mathtt{b}_0\mathtt{b}_1\ldots\mathtt{b}_j = \mathtt{a}_0\mathtt{a}_1\ldots\mathtt{a}_i\mathtt{b}_0\mathtt{b}_1\ldots\mathtt{b}_j$ を表す 2 項演算 $*$ が定義されたものである．自由モノイドの場合，整除関係 $\sigma\sqsubseteq\tau$ は，$\sigma\in A^*$ が $\tau\in A^*$ の**始切片** (*initial segment*) であることを表す．自由モノイドを無限文字列も含むように $A^{\leqslant\omega} := A^*\cup A^{\mathbb{N}}$ と拡張する．ここで，$\sigma\in A^*$ と $x\in A^{\leqslant\omega}$ に対する文字列結合 $\sigma * x$ は通常のように定義され，整除関係つまり始切片関係 $\sigma\sqsubseteq x$ も上と同様に定義される．また，$\sigma\sqsubset x$ と書いたとき，$\sigma\sqsubseteq x$ かつ $\sigma\neq x$ であることを意味する．

**定義 4.6.** 部分関数 $\varphi :\subseteq A^* \to A^*$ が**単調** (*monotone*) とは，以下の条件を満たすことである．

$$(\forall \sigma, \tau \in \mathrm{def}(\varphi))\ \sigma \sqsubseteq \tau \implies \varphi(\sigma) \sqsubseteq \varphi(\tau).$$

ここで，定義 2.1 で述べたように $\mathrm{def}(\varphi)$ は $\varphi$ の定義域を意味する．部分単調関数 $\varphi :\subseteq A^* \to A^*$ が与えられたとき，部分関数 $\widehat{\varphi} :\subseteq A^{\mathbb{N}} \to A^{\mathbb{N}}$ を次によって定義する．

$$\widehat{\varphi}(x)(n) = m \iff (\exists \sigma \sqsubset x)\ \varphi(\sigma)(n) = m.$$

つまり，集合論的記法を用いれば，$\widehat{\varphi}(x) = \bigcup_{\sigma \sqsubset x} \varphi(\sigma)$ によって定義するということである．部分関数 $f :\subseteq A^{\mathbb{N}} \to A^{\mathbb{N}}$ が**連続** (*continuous*) とは，ある部分単調関数 $\varphi$ が存在して，部分連続関数 $\widehat{\varphi}$ の $f$ の定義域への制限が $f$ と等しいことを意味する．つまり，任意の $x \in \mathrm{def}(f)$ について $f(x) = \widehat{\varphi}(x)$ が成り立つことである．

**演習問題 4.7.** 集合 $A$ に離散位相を入れ，$A^{\mathbb{N}}$ にはその積位相を入れる．このとき，任意の $f :\subseteq A^{\mathbb{N}} \to A^{\mathbb{N}}$ について，

$$f \text{ は定義 } 4.6 \text{ の意味で連続である} \iff f \text{ は位相空間論の意味で連続である}$$

ということを示せ．

**注意.** 定義 4.6 の単調関数は常に全域関数であると仮定してよい．なぜなら，部分単調関数 $\varphi :\subseteq A^* \to A^*$ が与えられたとき，$\psi(\sigma) = \bigcup\{\varphi(\tau) : \tau \sqsubseteq \sigma \text{ and } \varphi(\tau)\downarrow\}$ で定義すれば $\psi$ は全域になるが，$\varphi$ と $\psi$ が定義する連続関数は同一である．また，空語 $\varepsilon$ に対して $\varphi(\varepsilon) = \varepsilon$ としても一般性を失わない．

つまり，連続関数とは，有限文字列上の単調関数 $\varphi :\subseteq A^* \to A^*$ によって制御されるストリーム上の関数のことである．ストリームの計算処理を行うマシンは，この有限文字列上の単調関数の部分の計算を行うのみであり，動作としては通常のチューリングマシンと全く等しい．ストリーム上の関数が計算可能とは，このようなマシンによって計算されることである．数学的には，以下のように定式化される．

**定義 4.8.** 部分関数 $\Phi :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ が**計算可能** (*computable*) または**計算可能連続** (*computably continuous*) とは，ある部分計算可能単調関数 $\widehat{\varphi}$ の $f$ の定義域への制限が $\Phi$ と等しいことを意味する．つまり，任意の $x \in \mathrm{def}(\Phi)$ について $\Phi(x) = \widehat{\varphi}(x)$ が成り立つことである．

**注意.** 定義 4.8 の単調関数も常に全域関数であると仮定できる．ただし，定義 4.6 の後の注意のような全域関数 $\psi$ は計算可能とは限らないので，以下の修正が必要である．部分計算可能単調関数 $\varphi :\subseteq A^* \to A^*$ が与えられたとき，$\eta(\sigma) = \bigcup\{\varphi(\tau) : \tau \sqsubseteq \sigma \text{ and } \varphi(\tau)[|\sigma|]\downarrow\}$ で定義する．ここで $|\sigma|$ は文字列 $\sigma$ の長さを意味し，$\varphi(\tau)[|\sigma|]\downarrow$ は，$\varphi(\tau)$ を計算するチューリングマシンの動作が高々 $|\sigma|$ ステップで停止することを意味する．このように定義すれば，$\eta$ は全域計算可能単調関数であり，$\widehat{\eta} = \widehat{\varphi}$ となる．

計算可能連続関数の定義 4.8 は，クリーネ第 2 代数を定義するにあたっては必要ないが，関数の連続性をストリーム計算と結びつけることで，連続関数の秘める計算的成分を理解する助けに

なると思う．

さて，定義 4.6 のように，$\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数とは $\mathbb{N}^*$ 上の（全域）単調関数に他ならない．ところで，$\mathbb{N}^*$ は可算集合であるから $\mathbb{N}$ と一対一対応があるので，$\mathbb{N}^*$ 上の関数全体と $\mathbb{N}^{\mathbb{N}}$ との間にも一対一対応がある．多少の調整を加えて，$\mathbb{N}^*$ 上の単調関数全体と $\mathbb{N}^{\mathbb{N}}$ との一対一対応付けを与えることができ，よって $\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数全体を $\mathbb{N}^{\mathbb{N}}$ によって添字付けすることができる，というのがクリーネ第 2 代数の定義の基本方針である．この方針を念の為，丁寧に解説しよう．

いま，$p \in \mathbb{N}^{\mathbb{N}} \simeq [\mathbb{N}^* \to \mathbb{N}]$ が与えられたとき，$p$ がコードする単調関数 $\eta_p$ は，以下のように帰納的に定義される．空語 $\varepsilon$ について $\eta_p(\varepsilon) = p(\varepsilon) \in \mathbb{N}^*$ とし，与えられた $\tau \in \mathbb{N}^*$ と $n \in \mathbb{N}$ について，$\eta_p(\tau * n) = \eta_p(\tau) * p(\tau * n)$ によって定義する．明らかに $\eta_p$ は $\mathbb{N}^*$ 上の単調関数であるから，$\psi_p := \widehat{\eta}_p$ は $\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数となる．さらに，$\mathbb{N}^{\mathbb{N}}$ 上の任意の部分連続関数 $f$ はある $\psi_p$ の制限である．この添字付け $\{\psi_p\}$ に対して，定義 2.24 の 2 項演算

$$p \cdot x \downarrow = y \iff x \in \mathrm{def}(\psi_p) \mathbin{\&} \psi_p(x) = y$$

を用いて得られた $\underset{\sim}{\mathbf{K}}_2 := (\mathbb{N}^{\mathbb{N}}, \cdot)$ がクリーネ第 2 代数である．

このアイデアを元に，もうひとつ部分結合子代数を作ることもできる．いま，$(\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}$ を $\mathbb{N}$ 上の全域計算可能関数全体の集合とする．定義より，$\{\psi_p : p \in (\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}\}$ は $\mathbb{N}^{\mathbb{N}}$ 上の部分計算可能連続関数の族をなす．このとき上と同じ部分 2 項演算を考えると，これはクリーネの第 2 代数の部分代数をなし，$\underline{\mathrm{K}}_2 := ((\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}, \cdot)$ もまた部分結合子代数である．

## 4.3. スコットのグラフモデル

自然数の部分集合 $A \subseteq \mathbb{N}$ の枚挙をストリームの一種として考えることができる．つまり，枚挙とは，以下のように自然数を並べていくストリームである．

$$2, 3, 7, \bullet, 11, 5, 7, 13, \bullet, 19, 17, \bullet, \bullet, 23, 3, 29, \ldots$$

ここで，記号 $\bullet$ は何も並べないことを意味する．このストリームは素数の集合の枚挙のつもりである．ここで注意することは，枚挙を考える際，並べる順番は気にしないし，同じものを何度も並べてもよいとする．したがって，たとえば素数の集合を枚挙するストリームは無限パターン存在する．

数学的な定式化を与えれば，**枚挙** (*enumeration*) というものは，関数 $p\colon \mathbb{N} \to \mathbf{1} + \mathbb{N}$ である．ここで $\mathbf{1} = \{\bullet\}$ であり，$\mathbf{1} + \mathbb{N}$ は $\mathbf{1}$ と $\mathbb{N}$ の和集合を表しているが，和集合の記号 $\cup$ を使わなかったのには理由があり，それは後の節で明らかになる．しかし，今回のケースでは単に

$$\mathbf{1} + \mathbb{N} = \mathbf{1} \cup \mathbb{N} = \{\bullet, 0, 1, 2, \ldots\}$$

だと思っても差し支えはない．集合 $A \subseteq \mathbb{N}$ について，$\mathbf{1} + A$ も同様に，$\mathbf{1}$ と $A$ の和集合を表すものとする．

**定義 4.9.** 集合 $A \subseteq \mathbb{N}$ が**計算的可算**あるいは**計算可枚挙** (*computably enumerable*) とは，計算可能な全射 $p : \mathbb{N} \to \mathbf{1} + A$ が存在することを意味する．

**注意.** 計算可枚挙集合は，長らく**再帰的可算** (*recursively enumerable*) 集合と呼ばれてきたものである．頭文字を取って，計算量クラス RE と書かれることもあった．しかし，計算論の基礎概念について，英語圏での大規模な名称変更が前世紀末から始まり，少なくとも計算可能性理論の周辺分野では，名称変更がかなり浸透している．計算論の文脈ではなぜか “enumerable” を「可算」と伝統的に訳しているようであるから，今回は試しに “computably enumerable” を「計算的可算」と訳してみることにする．英名を無視すれば，これは通常の可算性を計算の世界で翻訳した定義であるから，そもそも計算的可算と呼ぶほうが妥当かもしれない．

関数 $p : \mathbb{N} \to \mathbf{1} + \mathbb{N}$ が枚挙する集合を $\mathrm{Enum}(p) = \{p(n) : n \in \mathbb{N}\} \cap \mathbb{N}$ によって定義する．以後，$\mathcal{P}\mathbb{N}$ によって，$\mathbb{N}$ の冪集合，つまり自然数の部分集合全体の集合を表す．

**定義 4.10.** 関数 $F\colon \mathcal{P}\mathbb{N} \to \mathcal{P}\mathbb{N}$ が**連続** (*continuous*) であるとは，次のような連続関数 $f : (\mathbf{1} + \mathbb{N})^{\mathbb{N}} \to (\mathbf{1} + \mathbb{N})^{\mathbb{N}}$ が存在することである．

$p \in (\mathbf{1} + \mathbb{N})^{\mathbb{N}}$ が $P \subseteq \mathbb{N}$ の枚挙ならば，$f(p) \in (\mathbf{1} + \mathbb{N})^{\mathbb{N}}$ は $F(P) \subseteq \mathbb{N}$ の枚挙である．

もし，この $f$ が計算可能関数であれば，$F$ は**枚挙作用素** (*enumeration operator*) と呼ばれる．

言い換えると，以下の図式が可換になっている．

$$\begin{array}{ccc} \mathcal{P}\mathbb{N} & \xrightarrow{\ F\ } & \mathcal{P}\mathbb{N} \\ {\scriptstyle \mathrm{Enum}}\uparrow & & \uparrow{\scriptstyle \mathrm{Enum}} \\ (\mathbf{1}+\mathbb{N})^{\mathbb{N}} & \xrightarrow[\ f\ ]{} & (\mathbf{1}+\mathbb{N})^{\mathbb{N}} \end{array}$$

**注意.** 位相空間論を既に学んでいる人のために述べておくと，上の意味での $\mathcal{P}\mathbb{N}$ 上の連続性は，$\mathcal{P}\mathbb{N}$ 上のカントール位相での連続性でなく，$\mathcal{P}\mathbb{N}$ 上のスコット位相での連続性であることに注意する．これは，$\mathcal{P}\mathbb{N}$ を離散空間 $\mathbf{2}$ の可算積 $\mathbf{2}^{\mathbb{N}}$ ではなくシエルピンスキ空間 $\mathbb{S}$ の可算積 $\mathbb{S}^{\mathbb{N}}$ と考えることと同値である．

$\mathbb{N}^{\mathbb{N}}$ 上の連続関数と同様に，$\mathcal{P}\mathbb{N}$ 上の枚挙作用素もまた有限的に取り扱うことができる．まず，$\mathcal{P}_{\mathrm{fin}}(\mathbb{N})$ を自然数の有限部分集合全体を表すものとすると，$\mathbb{N}$ と $\mathcal{P}_{\mathrm{fin}}(\mathbb{N})$ を一対一に対応付けられる．たとえば，有限集合 $A \subset \mathbb{N}$ と自然数 $\sum_{n\in A} 2^n$ を同一視すればよい．この自然数を有限集合 $A$ の**標準コード** (*canonical code*) と呼ぶ．標準コード $e \in \mathbb{N}$ の有限集合は $\mathsf{D}_e$ と書かれる．

連続関数 $F\colon \mathcal{P}\mathbb{N} \to \mathcal{P}\mathbb{N}$ の**グラフコード**とは，次によって与えられる．

$$\Gamma(F) = \{\langle n, e\rangle : n \in F(\mathsf{D}_e)\}$$

逆に集合 $G \subseteq \mathbb{N}$ が与えられれば，次のように連続関数 $E_G\colon \mathcal{P}\mathbb{N} \to \mathcal{P}\mathbb{N}$ を定義できる．

$$n \in E_G(A) \iff (\exists e \in \mathbb{N})\,[\langle n, e\rangle \in G \text{ and } \mathsf{D}_e \subseteq A].$$

このとき，$\{E_G : G \in \mathcal{P}\mathbb{N}\}$ は $\mathcal{P}\mathbb{N}$ 上の連続関数全体の集合をなす．また，この $\mathcal{P}\mathbb{N}$ 上の連続関数の添字付けは，次によって $\mathcal{P}\mathbb{N}$ 上の全域 2 項演算を与える．

$$G \cdot A = E_G(A).$$

この 2 項演算の下で $\mathbf{G} = (\mathcal{P}\mathbb{N}, \cdot)$ は全域結合子代数 (例 2.25) をなし，これはスコットのグラフモデルと呼ばれる．クリーネの第 2 代数のように，スコットのグラフモデル $\mathbf{G}$ の計算可能版のようなものを考えることもできる．これを確認するために，まずは次の命題を証明しよう．

**命題 4.11.** 枚挙作用素 $F$ のグラフコード $\Gamma(F)$ は計算的可算である．逆に，$\Psi \subseteq \mathbb{N}$ が計算的可算ならば，$F_\Psi$ は枚挙作用素である．

*Proof.* $F$ を枚挙作用素とすると，定義 4.10 のような計算可能関数 $f : (\mathbf{1} + \mathbb{N})^{\mathbb{N}} \to (\mathbf{1} + \mathbb{N})^{\mathbb{N}}$ が存在する．このとき，定義 4.8 とその直後の注意より，ある全域計算可能単調関数 $\varphi$ について $f = \widehat{\varphi}$ となる．このとき，与えられた $e \in \mathbb{N}$ に対して，有限集合 $\mathsf{D}_e$ の枚挙 $p_e$ を作るプログラムは容易に構成できる．たとえば，$\mathsf{D}_e$ を小さい順に並べ上げ，全て並べ終えたら後は $\bullet$ を出力し続ければよい．つまり，ある計算可能関数 $p : \mathbb{N} \times \mathbb{N} \to \mathbf{1} + \mathbb{N}$ で，$p_e = p(e, \cdot)$ は $\mathsf{D}_e$ の枚挙であるようなものが存在する．$f$ の定義より，$f(p_e)$ は $F(\mathsf{D}_e)$ の枚挙である．よって，$n \in F(\mathsf{D}_e)$ ならば，ある $s$ について，$f(p_e)(s) = n$ となる．以上より，

$$\langle n, e\rangle \in \Gamma(F) \iff (\exists s \in \mathbb{N})\ f(p_e)(s) = n \iff (\exists s, t \in \mathbb{N})\ \varphi(p(e,0)p(e,1)\dots p(e,t))(s) = n$$

となるから，$\Gamma(F)$ は $\Sigma_1$ である．

逆に $\Psi$ を半計算可能であるとする．このとき，$\Psi_s$ を $\Psi$ の時刻 $s$ 近似とする．つまり，命題 **??**の証明のように，$n \in \Psi$ と $M(n)\downarrow$ が同値であるような $M$ を取り，$\Psi_s = \{n : M(n)[s]\downarrow\}$ と定義する．与えられた有限列 $\sigma \in (\mathbf{1} + \mathbb{N})^*$ について，$D_\sigma$ を $\sigma$ が枚挙する有限集合，つまり $D_\sigma = \{n \in \mathbb{N} : (\exists s \in \mathbb{N})\ \sigma(s) = n \neq \bullet\}$ とする．このとき，$|\sigma|$ によって $\sigma$ の長さを表すとし，

$$n \in E_\sigma \iff (\exists e)\ [\langle n, e\rangle \in \Psi_{|\sigma|} \text{ and } \mathsf{D}_e \subseteq D_\sigma]$$

と定義すると，$E_\sigma$ は有限集合であり，$E = \{(n, \sigma) : n \in E_\sigma\}$ は計算可能集合である．

次によって単調関数 $\varphi$ を定義する．与えられた $\sigma$ に対し，$\varphi(\sigma)$ は $E_{\sigma\restriction 0}$ の要素を小さい順に枚挙し，次に $E_{\sigma\restriction 1}$ の要素を小さい順に枚挙し，$E_{\sigma\restriction 1}$ の要素を小さい順に枚挙し，... という動作を繰り返すものとする．ここで $\sigma \restriction n$ は $\sigma$ の長さ $n$ までの制限を表す．つまり，$E_\tau = \{a^\tau_0 < a^\tau_1 < \cdots < a^\tau_{i(\tau)}\}$ のように $E_\tau$ を下から順に並べたとき，

$$\varphi(\sigma) = \langle a^{\sigma\restriction 0}_0, a^{\sigma\restriction 0}_1, \dots, a^{\sigma\restriction 0}_{i(\sigma\restriction 0)}, a^{\sigma\restriction 1}_0, a^{\sigma\restriction 1}_1, \dots, a^{\sigma\restriction 1}_{i(\sigma\restriction 1)}, \dots, a^\sigma_0, a^\sigma_1, \dots, a^\sigma_{i(\sigma)}\rangle$$

と定義する．$\varphi$ が単調関数であることは明らかである．$\varphi$ の計算可能性は，$E$ の計算可能性から従う．また，任意の $A$ の枚挙 $p$ に対して，$\widehat{\varphi}(p)$ が $F_\Psi(A)$ の枚挙となっていることは容易に確認できる．したがって，$F_\Psi$ は枚挙作用素である．□

いま，$(\mathcal{P}\mathbb{N})_{\mathbf{P}}$ を $\mathbb{N}$ の計算的可算部分集合全体の集合とする．定義より，$\{E_G : G \in (\mathcal{P}\mathbb{N})_{\mathbf{P}}\}$ は枚挙作用素全体の族をなす．このとき，上と同じように $G \cdot A = E_G(A)$ によって全域 2 項演算を考えると，これはスコットのグラフモデルの部分代数をなし，$\underline{\mathrm{G}} := ((\mathcal{P}\mathbb{N})_{\mathbf{P}}, \cdot)$ もまた全域結合子代数である．

## 4.4. 相対部分結合子代数

ところで，ここで例 2.13 で触れた作用の詳細を説明することができる．プログラム $p \in \mathbf{P}$ によって計算される部分単調関数を $\varphi_p :\subseteq \mathbb{N}^* \to \mathbb{N}^*$ と書くことにしよう．これから定義 4.8 のように $\mathbb{N}^{\mathbb{N}}$ 上の部分計算可能連続関数 $\Phi_p = \widehat{\varphi_p}$ を得ることができる．計算モノイド $\mathbf{P}$ の $\mathbb{N}^{\mathbb{N}}$ への部分モノイド作用を以下のように定義できる．

$$p \cdot x \downarrow = y \iff x \in \mathrm{def}(\Phi_p) \,\&\, \Phi_p(x) = y$$

この部分モノイド作用に対して，例 2.13 のように定義した前順序 $\leqslant_T$ がチューリング還元と呼ばれるものである．

**定義 4.12.** 関数 $f, g : \mathbb{N} \to \mathbb{N}$ が与えられたとき，$f$ が $g$ に**チューリング還元可能** (*Turing reducible*) であるとは，$f = \Phi(g)$ なる部分計算可能関数 $\Phi :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ が存在することである．このとき，$f \leqslant_T g$ と書く．

**演習問題 4.13.** チューリング還元可能性関係 $\leqslant_T$ が $\mathbb{N}^{\mathbb{N}}$ 上の**前順序** (*preorder*) であることを示せ．つまり，$\leqslant_T$ が反射的 (reflexive) かつ推移的 (transitive) であることを証明せよ．

同様にして，$\mathbf{P}$ は $\mathcal{P}\mathbb{N}$ に作用する．プログラム $p \in \mathbf{P}$ が枚挙する $\mathbb{N}$ の部分集合を $W_p$ と書く．すると，$\mathbf{P}$ の $\mathcal{P}\mathbb{N}$ へのモノイド作用を $p \cdot G = E_{W_p}(G)$ によって定義できる．このとき，$G, H \subseteq \mathbb{N}$ について，$H \leqslant_e G$ を $p \cdot G = H$ となる $p \in \mathbf{P}$ が存在することとして定義できる．

**定義 4.14.** 集合 $A, B \subseteq \mathbb{N}$ について，$A$ が $B$ に**枚挙還元可能** (*enumeration reducible*) であるとは，ある枚挙作用素 $\Psi : \mathcal{P}\mathbb{N} \to \mathcal{P}\mathbb{N}$ が存在して，$\Psi(B) = A$ となることである．このとき，$A \leqslant_e B$ と書く．

枚挙還元は，自然数の部分集合の持つ正の情報の複雑さを測る概念である．つまり，$A \leqslant_e B$ とは，$B$ の正例（$B$ を満たす例）が全て漏れずにストリームとして次々に与えられれば，$A$ の正例を全て並べることができる，というものである．

さて，上記では $\mathbf{P}$ の部分モノイド作用として，チューリング還元と枚挙還元を扱ったが，これらについて別の見方もある．まず，第 4.2 節で導入した $\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数の添字付け $\{\psi_p : p \in \mathbb{N}^{\mathbb{N}}\}$ に対して，

$$\{\psi_p : p \in (\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}\} = \{\Phi_p : p \in \mathbf{P}\}$$

が成立する．また，第 4.3 節で説明したように，$\mathcal{P}\mathbb{N}$ 上の連続関数全体の添字付け $\{E_G : G \in \mathcal{P}\mathbb{N}\}$ に対して，

$$\{E_G : G \in (\mathcal{P}\mathbb{N})_{\mathbf{P}}\} = \text{“枚挙作用素全体”}$$

が成立する．このように，チューリング還元や枚挙還元は，$\mathbb{N}^{\mathbb{N}}$ や $\mathcal{P}\mathbb{N}$ に対して $(\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}$ や $(\mathcal{P}\mathbb{N})_{\mathbf{P}}$ が何らかの意味で作用していると考えることができる．

さて，これから考察するものは「計算**不**可能なものに満ち溢れた世界 $\mathbb{N}^{\mathbb{N}}, \mathcal{P}\mathbb{N}$ の事象を，計算可能な道具 $(\mathbb{N}^{\mathbb{N}})_{\mathbf{P}}, (\mathcal{P}\mathbb{N})_{\mathbf{P}}$ を用いて分析する」ことを目的とする理論である．チューリング次数や枚挙次数の理論がその代表例であるが，この観点は，計算可能性理論，計算可能性解析学，記述集合論など様々な分野に遍在する考え方である．部分結合子代数の文脈では，これは相対部分結合子代数として定式化される．

**定義 4.15. 相対部分結合子代数** (*relative partial combinatory algebra*) とは，2 項演算とコンビネータ $\mathtt{k}, \mathtt{s}$ を共有する部分組合せ代数の対 $\mathbb{M} = (\underset{\sim}{\mathbf{M}}; \underline{\mathrm{M}})$ で，$\mathtt{k}, \mathtt{s} \in \underline{\mathrm{M}} \subseteq \underset{\sim}{\mathbf{M}}$ かつ，以下を満たすものである．

$$a, b \in \underline{\mathrm{M}} \text{ and } a \cdot b \downarrow \in \underset{\sim}{\mathbf{M}} \implies a \cdot b \in \underline{\mathrm{M}}.$$

このとき，$\underset{\sim}{\mathbf{M}}$ 上の部分関数 $f :\subseteq \underset{\sim}{\mathbf{M}} \to \underset{\sim}{\mathbf{M}}$ が $\mathbb{M}$**-実現可能** ($\mathbb{M}$-*realizable*) であるとは，以下を満たすことである．

$$(\exists e \in \underset{\sim}{\mathbf{M}})(\forall a \in \mathrm{def}(f))\ [e \cdot a \downarrow, \text{ and } e \cdot a = f(a)].$$

$e \in \underline{\mathrm{M}}$ として取れる場合，$f$ は $\mathbb{M}$**-計算可能** ($\mathbb{M}$-*computable*) であるという．

相対部分結合子代数の概念を説明すると，我々は計算可能なものと計算不可能なものが混在する世界に住んでいるものと考えよう．$\underset{\sim}{\mathbf{M}}$ がそのような世界を表し，$\underline{\mathrm{M}}$ はそのうちの計算可能なものだけを取り出したものである．これはたとえば，記述集合論やその周辺分野における基本概念である，**太字** (*boldface*) と**細字** (*lightface*) の点類と類似の発想である．つまり，細字点類 $\Gamma$ とその相対化である太字点類 $\underset{\sim}{\mathbf{\Gamma}}$ との対 $(\underset{\sim}{\mathbf{\Gamma}}, \Gamma)$ のようなものであり，実現可能関数と計算可能関数は，$\underset{\sim}{\mathbf{\Gamma}}$-可測関数と $\Gamma$-再帰関数に対応する．

**例 4.16.** $K_1$ をクリーネ第 1 代数 (例 2.23) とすると，$\mathbb{K}_1 = (K_1, K_1)$ は相対部分結合子代数である．このとき，$\mathbb{K}_1$-実現可能関数および $\mathbb{K}_1$-計算可能関数とは，$\mathbb{N}$ 上の部分計算可能関数である．

**例 4.17.** 第 4.2 節のクリーネ第 2 代数 $\underset{\sim}{\mathbf{K}}_2$ とその計算可能化 $\underline{\mathrm{K}}_2$ について，$\mathbb{K}_2 = (\underset{\sim}{\mathbf{K}}_2, \underline{\mathrm{K}}_2)$ は相対部分結合子代数をなす．このとき，$\mathbb{K}_2$-実現可能関数および $\mathbb{K}_2$-計算可能関数とは，それぞれ $\mathbb{N}^{\mathbb{N}}$ 上の部分連続関数および部分計算可能関数である．

**例 4.18.** 第 4.3 節のスコットのグラフモデル $\underset{\sim}{\mathbf{G}}$ とその計算可能化 $\underline{\mathrm{G}}$ について，$\mathbb{G} = (\underset{\sim}{\mathbf{G}}, \underline{\mathrm{G}})$ は相対部分結合子代数をなす．このとき，$\mathbb{G}$-実現可能関数および $\mathbb{G}$-計算可能関数とは，それぞれ $\mathcal{P}\mathbb{N}$ 上の連続関数および枚挙作用素である．

任意の部分結合子代数 $M$ は，例 4.16 のように $\mathbb{M} = (M, M)$ と同一視することによって，相対部分結合子代数であるとみなせる．そのような相対部分結合子代数，つまり $\underset{\sim}{\mathbf{M}} = \underline{\mathrm{M}}$ であるような相対部分結合子代数 $(\underset{\sim}{\mathbf{M}}, \underline{\mathrm{M}})$ は**フル** (*full*) であるという．たとえば，$\mathbb{K}_1$ はフルだが，$\mathbb{K}_2$ と $\mathbb{G}$ はフルではない．

さて，相対部分結合子代数が与えられたとき，その元の間のチューリング還元あるいはチューリング次数の類似物を常に導入できる．

**定義 4.19.** 相対部分結合子代数 $\mathbb{M} = (\underset{\sim}{\mathbf{M}}, \underline{\mathrm{M}})$ が与えられているとする．$a, b \in \underset{\sim}{\mathbf{M}}$ について，ある部分 $\mathbb{M}$-計算可能関数 $f :\subseteq \underset{\sim}{\mathbf{M}} \to \underset{\sim}{\mathbf{M}}$ が存在して，$f(b) = a$ となるとき，$a$ は $b$ から**相対的 $\mathbb{M}$-計算可能** (*relatively* $\mathbb{M}$*-computable*) であると言い，$a \leqslant_{\mathbb{M}} b$ と書く．つまり，

$$a \leqslant_{\mathbb{M}} b \iff (\exists e \in \underline{\mathrm{M}})\ e \cdot b \downarrow = a$$

によって定義する．

**演習問題 4.20.** 例 *4.17* の相対第 2 代数 $\mathbb{K}_2$ における相対的計算可能性 $\leqslant_{\mathbb{K}_2}$ は定義 *4.12* のチューリング還元 $\leqslant_T$ と同値であり，例 *4.18* の相対グラフモデル $\mathbb{G}$ における相対的計算可能性 $\leqslant_{\mathbb{G}}$ は定義 *4.14* の枚挙還元 $\leqslant_e$ と同値であることを示せ．

# 第 2 章

# 実現可能性と高階関数空間

## §1. クリーネ実現可能性

本節では論理式の正しさを計算可能性によって保証することを試みる**クリーネ実現可能性** (*Kleene realizability*) について議論する．ここで正しさの保証というものは，たとえば $A \vee B$ が真ならば，どちらが正しいかの証拠を持ってくることを要求するものである．たとえば，RH を現代数学における最大の未解決問題のひとつであるリーマン予想を表すものとする．われわれは普段，古典論理を用いているから，特に排中律より，リーマン予想は正しいか正しくないかのいずれかである，つまり $\mathsf{RH} \vee \neg\mathsf{RH}$ である．しかし，RH と $\neg$RH のどちらが正しいかを断言することは，リーマン予想を解決することと同等である．また，もしリーマン予想が ZFC 集合論から独立であったとしたら，状況はますますややこしくなりそうである．つまるところ，$A \vee B$ の正しさの証拠を見つけてくるというのは，単に $A \vee B$ が正しいと主張することより難しそうである．

この正しさの保証のアイデアの源流は，直観主義論理の**ブラウワー-ハイティング-コルモゴロフ解釈** (*Brower-Heyting-Kolmogorov interpretation*) あるいは略して **BHK 解釈**と呼ばれるものである．BHK 解釈のアイデアは，以下のように帰納的に説明できる．

1. $A$ が原始論理式ならば，$A$ の証拠とは $A$ の証明である．
2. $A \wedge B$ の証拠とは，$A$ の証拠と $B$ の証拠の対である．
3. $A \vee B$ の証拠とは，どちらの式が正しいかの言及 $i$ と，正しい側の式の証拠 $p$ の対 $\langle i, p\rangle$ である．より正確には，$i = 0$ ならば $p$ は $A$ の証拠であり，$i = 1$ ならば $p$ は $B$ の証拠である．
4. $A \to B$ の証拠とは，次を満たす関数 $f$ である．もし $x$ が $A$ の証拠ならば，$f(x)$ は $B$ の証拠である．
5. $\exists x A(x)$ の証拠とは，対 $\langle a, p\rangle$ である．ここで，$p$ は $A(a)$ の証拠である．
6. $\forall x A(x)$ の証拠とは，次を満たす関数 $f$ である．量化領域内の任意の $x$ について，$f(x)$ は $A(x)$ の証拠である．

実際には，上記項目内の関数のクラスをある程度制限するのが標準的である．クリーネ実現可能性においては，部分計算可能関数 $f$ のみを考え，より一般の実現可能性では，固定した部分結

合子代数による実現可能関数を考える．

しかし，古典論理における真理をすべて計算可能性によって保証できるわけではない．そこで，どのような論理式についての真理であれば計算可能性によって保証できるかを調査したい．クリーネ実現可能性のアイデアは，部分結合子代数 $M$ が与えられたとき，各文について $\mathcal{P}(M)$-値の真理値を割り当てることと考えることもできる．まず示すことは，ハイティング算術によって証明可能な式は，部分結合子代数によって正しさを保証できる，ということである．

形式的には，まずは命題論理式に対しては，実現可能性は次のように定義される．

**定義 1.1.** 部分結合子代数 $M$ が与えられているとする．各論理式 $A$ について，$||A||$ を以下のように帰納的に定義する．

1. $||\top|| = M$ かつ $||\bot|| = \varnothing$ と定義する．
2. $||A_0 \wedge A_1|| = ||A_0|| \times ||A_1|| := \{\langle p, q\rangle : p \in ||A_0|| \text{ and } q \in ||A_1||\}$.
3. $||A_0 \vee A_1|| = ||A_0|| \sqcup ||A_1|| := \{\langle \underline{i}, p\rangle : p \in ||A_i||\}$.
4. $||A \to B|| := \{p \in M : (\forall x \in M)\ [x \in ||A|| \implies px \downarrow \in ||B||]\}$.

以後，$a \in ||A||$ をしばしば “$a$ $\mathbf{r}$ $A$” と略記し，**元 $a$ は式 $A$ を実現する** ($a$ *realizes* $A$) と読む．

ここから扱う論理は，古典論理の一般化である非古典論理である．ひとびとが，線形な現象だけでなく非線形な現象も研究するように，あるいは可換から非可換なものへ数学の興味が移ろっていくように，論理もまた古典から非古典へと拡張された．特にコンピュータの時代が訪れると，非古典論理的な現象が随所に現れることが発見される．

## 1.1. 直観主義命題論理の実現

ハイティング算術を説明するために，まずは直観主義命題論理の体系を導入しよう．本稿では，ゲンツェンによる**自然演繹** (*natural deduction*) の体系と実質的に同値な体系を採用する．

論理式の有限列 $\Gamma$ に対して，「$\Gamma$ に属す式が全て成立すると仮定すれば，式 $\varphi$ を証明できる」あるいは「公理系 $\Gamma$ の下で $\varphi$ が証明可能である」を意味する記号 $\Gamma \vdash \varphi$ を帰納的に定義しよう．

**定義 1.2.** 各 $A \in \Gamma$ について，$\Gamma \vdash A$ が成立する．また，次の推論規則が成立する．

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash A}$$

つづいて，命題論理記号 $\to, \wedge, \vee$ に対する推論規則を定義する．自然演繹では，各論理記号に対して，それぞれ**導入規則** (I) と**除去規則** (E) という 2 つの推論規則が以下のように割り当てられる．

**定義 1.3.** 各命題論理記号に対して，以下のように 2 つずつ推論規則を割り当てる．

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ (\to I) \qquad \frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}\ (\to E)$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\wedge I) \qquad \frac{\Gamma \vdash A_0 \wedge A_1}{\Gamma \vdash A_i}\ (\wedge E)$$

$$\frac{\Gamma \vdash A_i}{\Gamma \vdash A_0 \vee A_1}\ (\vee I) \qquad \frac{\Gamma \vdash A \vee B \qquad \Gamma, A \vdash C \qquad \Gamma, B \vdash C}{\Gamma \vdash C}\ (\vee E)$$

ここで，たとえば $\to$ の除去規則 $(\to E)$ は三段論法 (modus ponens) として知られるものである．まずは命題論理式に関する実現可能性について証明してみよう．

さて，先に述べたように $\Gamma \vdash A$ の「意図」とは，「$\Gamma$ に属す式をすべて仮定すれば，$A$ が成立する」というものであった．つまり，$\Gamma = D_0, D_1, \ldots D_n$ とすれば，$D_0, D_1, \ldots, D_n \vdash A$ の「意図」とは $\bigwedge_{i \leqslant n} D_i \to A$ である．この「意図」の下で，実現の定義 1.1 を拡張しよう．以後，部分結合子代数 $M$ の元 $p$ が $\Gamma \vdash A$ を実現するとは，$p$ が $\bigwedge_{i \leqslant n} D_i \to A$ を実現することを意味する．つまり，もし $x$ が $\bigwedge_{i \leqslant n} D_i$ を実現するならば $px$ が $A$ を実現することである：

$$p \ \mathbf{r}\ D_0, D_1, \ldots, D_n \vdash A \iff \forall x \left( x \ \mathbf{r} \ \bigwedge_{i \leqslant n} D_i \implies px \ \mathbf{r} \ A \right).$$

ただし，一般に，$\Gamma$ と $A$ は命題変数を含みうる．出現する命題変数 $\bar{v} = v_0, v_1, \ldots, v_j$ を明示的に書くと，$\Gamma(\bar{v}) \vdash A(\bar{v})$ のように表される．この命題変数列 $\bar{v}$ に対する付値 (evaluation) とは，$\top$ と $\bot$ からなる列 $\bar{b} = b_0, b_1, \ldots, b_j$ を意味する．ここで，記号 $\top, \bot$ は，部分結合子代数の元 $\mathtt{true}, \mathtt{false} \in \underline{\mathrm{M}}$ とも同一視される．

いま，$\mathbb{M}$ の項 $t$ が命題論理式 $\Gamma(\bar{v}) \vdash A(\bar{v})$ を実現するということを次によって定義する：命題変数列 $\bar{v}$ に対する任意の付値 $\bar{b}$ に対して，$t(\bar{b})$ が $\Gamma(\bar{b}) \vdash A(\bar{b})$ を実現する，つまり，

$$t(\bar{v}) \ \mathbf{r}\ \Gamma(\bar{v}) \vdash A(\bar{v}) \iff (\forall \bar{b})(\forall x) \left( x \ \mathbf{r} \ \bigwedge \Gamma(\bar{b}) \implies t(\bar{b})x \ \mathbf{r} \ A(\bar{b}) \right).$$

このとき，$t(\bar{v}) \ \mathbf{r}\ \Gamma(\bar{v}) \vdash A(\bar{v})$ と書く．この定義の下で，各推論規則が実現可能であることを示そう．これが意味することは，各推論規則の上式が $M$ 上の項 $s(\bar{v})$ によって実現されるならば，下式もある項 $t(\bar{v})$ によって実現されるということである．ただし，各推論は付値とは無関係なので，命題変数を含まない（あるいは既に付値が与えられた）命題論理式に対してのみ議論しても一般性を失わない．この仮定の下で，上で導入した直観主義命題論理の公理と推論規則はいずれも計算論的に妥当である，ということが以下によって示される．

**定理 1.4.** $M$ を部分結合子代数とする．定義 *1.2* と *1.3* の推論規則は $M$ で実現可能である．

*Proof.* まず，矛盾に関する規則について，矛盾を実現する元は存在しないから，上式は決して実現されない．

次に，$\to$ の導入規則は $\lambda$ 抽象に，除去規則は適用に対応する．議論の単純化のために $\Gamma$ はひとつの論理式 $D$ からなると仮定する．まず，$\to$ の除去規則 $(\to E)$ については，以下を得る．

$$\dfrac{\dfrac{\dfrac{f\ \mathbf{r}\ D \vdash A \to B}{a\ \mathbf{r}\ D \Longrightarrow fa\ \mathbf{r}\ A \to B}}{a\ \mathbf{r}\ D \text{ and } z\ \mathbf{r}\ A \Longrightarrow faz\ \mathbf{r}\ B} \qquad \dfrac{x\ \mathbf{r}\ D \vdash A}{a\ \mathbf{r}\ D \Longrightarrow xa\ \mathbf{r}\ A}}{\dfrac{a\ \mathbf{r}\ D \Longrightarrow fa(xa)\ \mathbf{r}\ B}{\mathbf{s}fx \equiv \Lambda a.fa(xa)\ \mathbf{r}\ D \vdash B}}$$

次に $\to$ の導入規則 $(\to I)$ の意味を考えると，カリー化 $\mathrm{curry}\colon [(D \times A) \to B] \simeq [D \to (A \to B)]$ に対応するが，これは以下のように $\lambda$-抽象によって実現できる．

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p\ \mathbf{r}\ D, A \vdash B}{p\ \mathbf{r}\ D \wedge A \to B}}{x\ \mathbf{r}\ D \wedge A \Longrightarrow px\ \mathbf{r}\ B}}{y\ \mathbf{r}\ D \text{ and } z\ \mathbf{r}\ A \Longrightarrow p\langle y,z\rangle\ \mathbf{r}\ B}}{y\ \mathbf{r}\ D \Longrightarrow \Lambda z.p\langle y,z\rangle\ \mathbf{r}\ A \to B}}{\Lambda y.\Lambda z.p\langle y,z\rangle\ \mathbf{r}\ D \to (A \to B)}}{\mathsf{curry}(p) := \Lambda y.\Lambda z.p\langle y,z\rangle\ \mathbf{r}\ D \vdash A \to B}$$

つづいて，$\wedge$ の推論規則は直積に対応する．

$$\dfrac{\dfrac{a\ \mathbf{r}\ D \vdash A}{x\ \mathbf{r}\ D \Longrightarrow ax\ \mathbf{r}\ A} \qquad \dfrac{b\ \mathbf{r}\ D \vdash B}{x\ \mathbf{r}\ D \Longrightarrow bx\ \mathbf{r}\ B}}{\dfrac{x\ \mathbf{r}\ D \Longrightarrow \langle ax, bx\rangle\ \mathbf{r}\ A \wedge B}{\Lambda x.\langle ax, bx\rangle\ \mathbf{r}\ D \vdash A \wedge B}} \qquad \dfrac{\dfrac{c\ \mathbf{r}\ D \vdash A_0 \wedge A_1}{x\ \mathbf{r}\ D \Longrightarrow cx\ \mathbf{r}\ A_0 \wedge A_1}}{\dfrac{x\ \mathbf{r}\ D \Longrightarrow \pi_i(cx)\ \mathbf{r}\ D \vdash A_i}{\Lambda x.\pi_i(cx)\ \mathbf{r}\ D \vdash A_i}}$$

最後に，$\vee$ の推論規則は余積に対応する．

$$\dfrac{\dfrac{\dfrac{c\ \mathbf{r}\ D \vdash A_i}{x\ \mathbf{r}\ D \Longrightarrow cx\ \mathbf{r}\ A_i}}{x\ \mathbf{r}\ D \Longrightarrow \langle \underline{i}, cx\rangle\ \mathbf{r}\ A_0 \vee A_1}}{\Lambda x.\langle \underline{i}, cx\rangle\ \mathbf{r}\ D \vdash A_0 \vee A_1}\ (\vee I)$$

$\vee$ の除去規則について，まずは議論の本質を明確にするために $\Gamma$ は省略する．

$$\dfrac{p\ \mathbf{r}\ A \vee B \qquad \dfrac{\dfrac{s\ \mathbf{r}\ A \vdash C}{x\ \mathbf{r}\ A \Longrightarrow sx\ \mathbf{r}\ C} \qquad \dfrac{t\ \mathbf{r}\ B \vdash C}{x\ \mathbf{r}\ B \Longrightarrow tx\ \mathbf{r}\ C}}{[s,t]p := \mathtt{if}\ \pi_0 p\ \mathtt{iszero\ then}\ s(\pi_1 p)\ \mathtt{else}\ t(\pi_1 p)\ \mathbf{r}\ C}\ (\vee E)$$

一応，$\vee$ の除去規則 $(\vee E)$ の実現について説明すると，$p$ は $A \vee B$ を実現しているので，$\pi_0 p = \underline{0}$ の場合は，$\pi_1 p\ \mathbf{r}\ A$ である．よって，$x = \pi_1 p$ を考えれば，$sx = s(\pi_1 p)\ \mathbf{r}\ C$ を得る．同様に，

$\pi_0 p \neq \underline{0}$ の場合は，$\pi_1 p \ \mathbf{r}\ B$ であるから，$x = \pi_1 p$ を考えれば，$tx = t(\pi_1 p) \ \mathbf{r}\ C$ を得る．また，最後の条件分岐による式が長く煩雑なので，以下のような略記をしばしば用いる．

$$[s,t] := \lambda p.\texttt{if } \pi_0 p \texttt{ iszero then } s(\pi_1 p) \texttt{ else } t(\pi_1 p).$$

$\Gamma$ が空でない場合，$\vee$ の除去規則 $(\vee E)$ の実現は，一旦 $\to$ の導入規則 $(\to I)$ を経由すると多少，楽になる．つまり，$p, s, t$ が $(\vee E)$ の上式を実現している場合，$(\to I)$ の実現の議論から，$s$ と $t$ を $\tilde{s} = \lambda yz.s\langle y, z\rangle$ と $\tilde{t} = \lambda yz.t\langle y, z\rangle$ に置き換えると，以下が成立する．

$$\frac{p \ \mathbf{r}\ D \vdash A \vee B}{a \ \mathbf{r}\ D \Longrightarrow pa \ \mathbf{r}\ A \vee B} \qquad \frac{\dfrac{s \ \mathbf{r}\ D, A \vdash C}{\tilde{s} \ \mathbf{r}\ D \vdash A \to C}}{a \ \mathbf{r}\ D \Longrightarrow \tilde{s}a \ \mathbf{r}\ A \to C} \qquad \frac{\dfrac{t \ \mathbf{r}\ D, B \vdash C}{\tilde{t} \ \mathbf{r}\ D \vdash B \to C}}{a \ \mathbf{r}\ D \Longrightarrow \tilde{t}a \ \mathbf{r}\ B \to C}$$

したがって，$a \ \mathbf{r}\ D$ が与えられているとき，以下が成立する．

$$\frac{pa \ \mathbf{r}\ A \vee B \qquad \dfrac{\tilde{s}a \ \mathbf{r}\ A \to C}{x \ \mathbf{r}\ A \Longrightarrow \tilde{s}ax \ \mathbf{r}\ C} \qquad \dfrac{\tilde{t}a \ \mathbf{r}\ B \to C}{x \ \mathbf{r}\ B \Longrightarrow \tilde{t}ax \ \mathbf{r}\ C}}{[\tilde{s}, \tilde{t}]pa = \texttt{if } \pi_0(pa) \texttt{ iszero then } \tilde{s}a(\pi_1(pa)) \texttt{ else } \tilde{t}a(\pi_1(pa)) \ \mathbf{r}\ C}$$

以上より，$\lambda a.[s,t]pa$ が $D \vdash C$ を実現することが示された． □

ちなみに自然演繹とラムダ計算の対応（カリー-ハワード同型対応）のようなものを経由して，実現可能性を示したが，ヒルベルト式の論理体系の方が結合子論理と明示的な対応がある．

## 1.2. 直観主義述語論理の実現

述語論理の式の実現可能性を定義する前に，述語論理の意味論について復習する必要がある．命題論理では命題変数の解釈，つまり各変数への真偽の割り当てに基づいて意味論を与えた．一方で，述語論理において変数は量化記号 $\forall$ および $\exists$ によって束縛される．通常，述語論理の意味論においては，まず量化範囲を指定する領域 $U$ が与えられる．つまり，$\forall$ は「$U$ に属す任意の…」と解釈され，$\exists$ は「$U$ の中に存在する」と解釈される．しかし，現時点では，とりあえずは細かいことは気にせず，部分結合子代数 $M$ が与えられており，領域は $U = M$ を考えることにしよう．

**定義 1.5.** 部分結合子代数 $M$ が与えられているとする．各論理式 $A$ について，$||A||$ を以下のように帰納的に定義する．

- $||\forall x A(x)|| = \prod_x ||A(x)|| := \{p \in M : (\forall x \in M)\ px \downarrow \in ||A(x)||\}$.
- $||\exists x A(x)|| = \coprod_x ||A(x)|| := \{\langle t, p\rangle : p \in ||A(t)||\}$.

以前と同様に，$p \in ||A||$ であるとき，$p$ は $A$ を実現すると言い，$p \ \mathbf{r}\ A$ と書く．

つづいて，量化記号 $\forall, \exists$ に対する導入規則を導入しよう．命題論理に対する推論規則と同様に，自然演繹では，各論理記号に対して，それぞれ導入規則 (I) と除去規則 (E) という 2 つの推論規則が以下のように割り当てられる．

**定義 1.6.** 各量化記号に対して，以下のように 2 つずつ推論規則を割り当てる．

$$\frac{\Gamma(\bar{v}) \vdash A(z,\bar{v})}{\Gamma(\bar{v}) \vdash \forall x A(x,\bar{v})}\ (\forall I) \qquad \frac{\Gamma(u,\bar{v}) \vdash \forall x A(x,\bar{v})}{\Gamma(u,\bar{v}) \vdash A(u,\bar{v})}\ (\forall E)$$

$$\frac{\Gamma(u,\bar{v}) \vdash A(u,\bar{v})}{\Gamma(u,\bar{v}) \vdash \exists x A(x,\bar{v})}\ (\exists I) \qquad \frac{\Gamma(\bar{v}) \vdash \exists x A(x,\bar{v}) \qquad \Gamma(\bar{v}), A(z,\bar{v}) \vdash B(\bar{v})}{\Gamma(\bar{v}) \vdash B(\bar{v})}\ (\exists E)$$

ここで，規則 $(\forall I)$ と $(\exists E)$ の中に現れる変数 $z$ は**変数条件**を満たす，つまり，この推論規則内では，$A$ 以外の論理式には現れない変数である．

命題論理の場合と同様に，述語論理においても，$\Gamma \vdash A$ の「意図」とは，「$\Gamma$ に属す式をすべて仮定すれば，$A$ が成立する」というものである．一般に，$\Gamma$ と $A$ は自由変数を持つかもしれない．出現する変数記号を明示的に書くと，$\Gamma(\bar{v}) \vdash A(\bar{v})$ と表される．その意味を考えるとき，各変数 $v$ は領域 $M$ 内の元として解釈される．そして，$\Gamma(\bar{v}) \vdash A(\bar{v})$ が意味論的に正しいということは，各変数 $v_i$ をいかなる元 $a_i \in M$ として解釈しても，$\Gamma(\bar{a}) \vdash A(\bar{a})$ が正しいということである．つまり，$\Gamma = D_0, D_1, \ldots D_n$ とすれば，$D_0, D_1, \ldots, D_n \vdash A$ の「意図」とは $(\forall \bar{a}) \bigwedge_{i \leqslant n} D_i(\bar{a}) \to A(\bar{a})$ である．この「意図」の下で，実現の定義 1.1 と 1.5 を拡張しよう．以後，部分結合子代数 $M$ の元 $p$ が $\Gamma(\bar{v}) \vdash A(\bar{v})$ を実現するとは，$p$ が $(\forall \bar{a}) \bigwedge \Gamma(\bar{a}) \to A(\bar{a})$ を実現することを意味する．

結合子完全性より，$M$ の元 $p$ を用いる代わりに，$M$ 上の項 $t$ を用いても変わりはない．したがって，以後は簡便性のため，しばしば，項 $t(\bar{v})$ が $\Gamma(\bar{v}) \vdash A(\bar{v})$ を実現する，とも言うことにする．これはつまり，以下を意味する．

$$t(\bar{v})\ \mathbf{r}\ \Gamma(\bar{v}) \vdash A(\bar{v}) \iff \forall \bar{a} \forall x \left( x\ \mathbf{r}\ \bigwedge \Gamma(\bar{a}) \implies t(\bar{a})x\ \mathbf{r}\ A(\bar{a}) \right).$$

この定義の下で，各推論規則が実現可能であることを示そう．これが意味することは，各推論規則の上式が $M$ 上の項 $s(u,\vec{v})$ によって実現されるならば，下式もある項 $t(u,\vec{v})$ によって実現されるということである．

さて，上で導入した直観主義命題論理の公理と推論規則はいずれも計算論的に妥当である，ということは以下によって示される．

**定理 1.7.** $M$ を部分結合子代数とする．定義 *1.6* の推論規則は $M$ で実現可能である．

*Proof.* まず，$(\exists I)$ と $(\forall E)$ の推論規則の実現について，$\Gamma$ が空でない場合を考えても，記号が煩雑になるだけで証明は本質的に変わらないので，$\Gamma = \varnothing$ であると仮定する．$\Gamma$ が空でない場合については，読者の演習問題とする．特に変数条件のない $(\exists I)$ と $(\forall E)$ については，以下のように

容易に実現できる．

$$\frac{t(u,\bar{v})\ \mathbf{r}\ A(u,\bar{v})}{\langle u,t(u,\bar{v})\rangle\ \mathbf{r}\ \exists x A(x,\bar{v})}\ (\exists I) \qquad \frac{t(\bar{v})\ \mathbf{r}\ \forall x A(x,\bar{v})}{t(\bar{v})u\ \mathbf{r}\ A(u,\bar{v})}\ (\forall E)$$

次に $(\forall I)$ について，議論を明確にするために $\Gamma = B$ の場合を考えよう．いま，$t(z,\bar{v})$ が $B(\bar{v}) \vdash A(z,\bar{v})$ を実現すると仮定する．このとき，変数 $z$ に $y$ を代入してみよう．変数条件から，$B$ は $z$ を変数として含まない，つまり $\bar{v}$ の中に $z$ は含まれない．よって，文字通り $z$ を $y$ に置き換えると，$t(y,\bar{v})$ は $B(\bar{v}) \vdash A(y,\bar{v})$ を実現する，と言い換えられる．以上をまとめると，

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{t(z,\bar{v})\ \mathbf{r}\ B(\bar{v}) \vdash A(z,\bar{v})}{t(y,\bar{v})\ \mathbf{r}\ B(\bar{v}) \vdash A(y,\bar{v})}\ (\text{変数条件より})}{a\ \mathbf{r}\ B(\bar{v}) \implies t(y,\bar{v})a\ \mathbf{r}\ A(y,\bar{v})}}{(\forall y)\ [a\ \mathbf{r}\ B(\bar{v}) \implies t(y,\bar{v})a\ \mathbf{r}\ A(y,\bar{v})]}\ (y \text{ は任意なので})}{a\ \mathbf{r}\ B(\bar{v}) \implies (\forall y)\ t(y,\bar{v})a\ \mathbf{r}\ A(y,\bar{v})}}{a\ \mathbf{r}\ B(\bar{v}) \implies \Lambda x.(t(x,\bar{v})a)\ \mathbf{r}\ \forall x A(x,\bar{v})}}{\Lambda a.\Lambda x.(t(x,\bar{v})a)\ \mathbf{r}\ B(\bar{v}) \vdash \forall x A(x,\bar{v})}$$

最後に $(\exists E)$ について議論する．簡単のために $\Gamma$ は省略する．先程の議論のように，もし $t(z,\bar{v})$ が $A(z,\bar{v}) \vdash B(\bar{v})$ を実現すると仮定する．変数条件から $\bar{v}$ の中には $z$ は含まれないので，$z$ に $y$ を代入すると，$t(y,\bar{v})$ は式 $A(y,\bar{v}) \vdash B(\bar{v})$ を実現する．これはどんな $y$ についても成り立つから，任意の $y$ について $t(y,\bar{v})$ は $A(y,\bar{v}) \vdash B(\bar{v})$ を実現する．以上より，

$$\dfrac{\dfrac{s(\bar{v})\ \mathbf{r}\ \exists x A(x,\bar{v})}{\pi_1(s(\bar{v}))\ \mathbf{r}\ A(\pi_0(s(\bar{v})),\bar{v})} \qquad \dfrac{\dfrac{t(z,\bar{v})\ \mathbf{r}\ A(z,\bar{v}) \vdash B(\bar{v})}{t(y,\bar{v})\ \mathbf{r}\ A(y,\bar{v}) \vdash B(\bar{v})}\ (\text{変数条件より})}{a\ \mathbf{r}\ A(y,\bar{v}) \implies t(y,\bar{v})a\ \mathbf{r}\ B(\bar{v})}}{t(\pi_0(s(\bar{v})),\bar{v})\pi_1(s(\bar{v}))\ \mathbf{r}\ B(\bar{v})}\ (y \text{ は任意なので})$$

□

以上より，直観主義一階述語論理の推論規則は任意の部分結合子代数によって実現可能であることが示された．

実際，直観主義論理上の一階算術の体系である**ハイティング算術** (*Heyting arithmetic*) を実現可能である．ハイティング算術とは，離散全順序環の非負部の公理（四則演算および順序に関する公理）に数学的帰納法の公理図式を加えたものである．自然数の四則演算や順序に関する基本的な公理が実現可能であることは自明なので，ここでは数学的帰納法が実現可能であることだけ確認しよう．ここで，算術的論理式 $A$ に対する**数学的帰納法** (*mathematical induction*) とは，以下の式である．

$$A(0)\ \wedge\ \forall x\ (A(x) \to A(x+1))\ \to\ \forall x A(x).$$

**定理 1.8.** 任意の部分結合子代数において，数学的帰納法は実現可能である．

*Proof.* 数学的帰納法は，以下のように原始再帰法によって実現されることを示そう．

$$\frac{p\ \mathbf{r}\ A(0) \qquad q\ \mathbf{r}\ \forall n\,(A(n)\to A(n+1))}{\mathtt{rec}\,pq\ \mathbf{r}\ \forall nA(n)}$$

ここで，$\mathtt{rec}$ は命題 3.12 で与えられた原始再帰作用素である．これを示すためには，任意の $n\in\mathbb{N}$ について $\mathtt{rec}\,pq\underline{n}\ \mathbf{r}\ A(n)$ を示せばよい．定義より，まず $\mathtt{rec}\,pq\underline{0}=p\ \mathbf{r}\ A(0)$ である．帰納的に，$\mathtt{rec}\,pq\underline{n}\ \mathbf{r}\ A(n)$ を仮定する．定義より，$\mathtt{rec}\,pq\underline{n+1}=q\underline{n}(\mathtt{rec}\,pq\underline{n})$ であるが，以下に注目する．

$$\frac{\dfrac{q\ \mathbf{r}\ \forall n\,(A(n)\to A(n+1))}{q\underline{n}\ \mathbf{r}\ A(n)\to A(n+1)}}{x\ \mathbf{r}\ A(n)\implies q\underline{n}x\ \mathbf{r}\ A(n+1)}$$

帰納的仮定より $x=\mathtt{rec}\,pq\underline{n}\ \mathbf{r}\ A(n)$ であるから，$q\underline{n}x=q\underline{n}(\mathtt{rec}\,pq\underline{n})\ \mathbf{r}\ A(n+1)$ が成立する．よって，帰納法によって，任意の $n\in\mathbb{N}$ について $\mathtt{rec}\,pq\underline{n}\ \mathbf{r}\ A(n)$ であることが示された．つまり，$\mathtt{rec}\,pq\ \mathbf{r}\ \forall nA(n)$ である． □

つづいて，**選択公理** (*axiom of choice*) の実現可能性について議論しよう．選択公理は超越的な原理の代表格として取り沙汰されることが多い．しかし，ここでは，選択公理が実は構成的な側面を持つ，ということを示そう．さて，選択公理とは，次の式で与えられる主張である．

$$\forall x\exists yA(x,y)\ \to\ \exists f\forall xA(x,f(x)).$$

このような関数 $f$ は $A$ の選択関数と呼ばれる．選択公理は様々な同値な言い換えがある．たとえば，$A_x=\{y:A(x,y)\}$ とおけば，選択公理の前提は $\{A_x\}_x$ が要素を持つ集合の族であることを述べ，選択公理の結論は $f\in\prod_x A_x$ となる $f$ の存在を意味する．つまり，選択公理は「空でない集合の族の直積は空でない」という主張であると言い表されることもある．

それでは，ある意味で，選択公理は計算的に正しい原理である，ということを証明しよう．

**定理 1.9.** 任意の部分結合子代数において，選択公理は実現可能である．

*Proof.* 実現可能性の文脈では，$\forall x\exists yA(x,y)$ の正当性を証明するためには，その証拠を持ってくる必要がある．その証拠とは，入力 $x$ に対して，ある $y$ と $A(x,y)$ の証拠を出力するアルゴリズムである．このアルゴリズムは，明らかに $A(x,y)$ を満たす関数 $x\mapsto y$ をひとつ選択する．より厳密には，選択公理は以下のようにして実現される．

$$\frac{\dfrac{\dfrac{\dfrac{p\ \mathbf{r}\ \forall x\exists yA(x,y)}{px\ \mathbf{r}\ \exists yA(x,y)}}{\pi_1px\ \mathbf{r}\ A(x,\pi_0px)}}{\pi_1p\ \mathbf{r}\ \forall xA(x,(\boldsymbol{\lambda}v.\pi_0pv)x)}}{\langle\boldsymbol{\lambda}v.\pi_0pv,\pi_1p\rangle\ \mathbf{r}\ \exists f\forall xA(x,f(x))}$$

□

構成的解析学の大家であるエレット・ビショップ (Errett Bishop, 1928–1983) は，超越的な原理を徹底的に拒絶したことで知られるが，選択公理についてはこう述べたという．

> 構成的数学において選択関数は存在する．選択とは，まさに存在というものの意味そのものから導かれるためである．

これはそのまま定理 1.9 の証明を述べている．つまり $\forall x \exists y A(x,y)$ が構成的に真ならば，そのような存在の保証 $f\colon x \mapsto y$ は必ず構成されているはずであるが，この $f$ は明らかに $A$ の選択関数である，つまり，選択公理は構成的に真である．そういうわけで，部分結合子代数という記号的世界上では，選択公理というものは常に成立する真理である．

しかし，たとえば構成的数学のある種の形式化においては，一般の数学的対象は，記号的世界の商として得られる．記号上の選択関数を数学的対象（つまり商集合）上の選択関数に変えるためには，同値類の代表元を取る必要があり，これは構成的には成し遂げられない．この意味で，外延的 (extensional) な選択公理は構成的には偽であるが，内包的 (intensional) な選択公理は構成的に真であると言われることがある．この詳細については次節以降で述べよう．

## § 2. 実現不可能な式

ここまでは様々な論理式の妥当性を計算によって保証できることを示してきた．それでは，どのような論理式は，計算によってその妥当性を保証できないだろうか．計算によって妥当性を保証できない原理の代表例となるものが**排中律** (*law of excluded middle*) である．これは，「$A$ であるか $A$ でないかのどちらかである」ということを述べる原理である．

**定理 2.1.** どんな部分結合子代数においても，排中律 $A \vee \neg A$ は実現可能ではない．

*Proof.* 適当に $a \in \underline{\mathrm{M}}$ を取り，$A(v)$ を $vv = a$ という式としよう．このとき，$(vv = a) \vee (vv \neq a)$ がクリーネ実現可能でない，つまり $||(vv = a) \vee (vv \neq a)|| = \varnothing$ ということを示せば十分である．与えられた $p \in \underline{\mathrm{M}}$ と $x \in \underset{\sim}{\mathbf{M}}$ に対して，もし $p \in ||(vv = a) \vee (vv \neq a)||$ であったとしたら，実現の定義より $\pi_0(px) = \underline{0}$ ならば $xx = a$ であり，$\pi_0(px) = \underline{1}$ ならば $xx \neq a$ であることに注意する．あらかじめ，$b \neq a$ となる $b \in \underline{\mathrm{M}}$ を適当に取っておく．このとき，

$$q := \lambda x.\mathtt{if}\ \pi_0(px)\ \mathtt{iszero}\ \mathtt{then}\ b\ \mathtt{else}\ a$$

と定義すると，$\pi_0(pq) = \underline{0}$ のときは $qq = b \neq a$ となり，$\pi_0(px) = \underline{1}$ のときは $qq = a$ となる．したがって，$pq \notin ||(qq = a) \vee (qq \neq a)||$ となる．つまり，任意の $p \in \underline{\mathrm{M}}$ について $p \notin ||(vv = a) \vee (vv \neq a)||$ が従う．よって $||(vv = a) \vee (vv \neq a)|| = \varnothing$ が導かれる． □

もし，$\mathbb{M}$ が全域でない部分結合子代数であったならば，上の証明を適当に修正すれば，**停止問題** (*halting problem*) の決定不可能性を導くことができる．さらに，われわれが用いているものは抽象的な部分結合子代数であるから，計算可能関数の代数ではなくとも，たとえば部分連続関数の代数や部分 $\Pi^1_1$-可測関数の代数においても，停止問題の決定不可能性の類似が成立する．

**命題 2.2** (一般停止問題の決定不可能性)**.** $\mathbb{M}$ を全域でない部分結合子代数とする．このとき，$(vx\downarrow)\vee(vx\uparrow)$ はクリーネ実現可能ではない．

*Proof.* 与えられた $p\in\underline{\mathrm{M}}$ と $x\in\underline{\mathbf{M}}$ に対して，もし $p\in||(vv\downarrow)\vee(vv\uparrow)||$ であったとしたら，実現の定義より $\pi_0(px)=\underline{0}$ ならば $xx\downarrow$ であり，$\pi_0(px)=\underline{1}$ ならば $xx\uparrow$ であることに注意する．あらかじめ，$s\downarrow$ かつ $t\uparrow$ となる項 $s,t\in\underline{\mathrm{M}}$ を適当に取っておく．このとき，

$$q:=\Lambda x.\texttt{if }\pi_0(px)\texttt{ iszero then }t\texttt{ else }s$$

と定義すると，$\pi_0(pq)=\underline{0}$ のときは $qq=t\uparrow$ となり，$\pi_0(pq)=\underline{1}$ のときは $qq=s\downarrow$ となる．したがって，$pq\notin||(qq\downarrow)\vee(qq\uparrow)||$ となる．つまり，任意の $p\in\underline{\mathrm{M}}$ について $p\notin||(vv\downarrow)\vee(vv\uparrow)||$ が従う．よって $||(vv\downarrow)\vee(vv\uparrow)||=\varnothing$ が導かれる． □

**系 2.3** (チューリングの定理)**.** プログラム $v$ に $x$ を入力した結果の計算が有限時間で出力を返すか否かを判定する計算可能なアルゴリズムは存在しない．

*Proof.* クリーネ第一代数 $\mathbb{M}=K_1$ に対して，命題 2.2 を適用せよ． □

さて，定理 1.9 で証明したものをスローガン的に言えば，「選択公理は構成的に正しい」というものである．そして，定理 2.1 で証明したものをスローガン的に言えば，「排中律は構成的に偽である」という感じであろうか．この 2 つが，構成的数学あるいは計算的数学における 2 つの代表的な基本原理である．もうひとつ，構成的数学における有名な定理をこれから証明しよう．その定理とは，スローガン的に言えば，「選択公理は排中律を導く」というものである．というわけで，これらの構成的数学の基本定理を表す 3 つのわかりやすいスローガンを並べてみよう．

「選択公理は構成的に正しい」「排中律は構成的に偽である」「選択公理は排中律を導く」

おや，どうやら，これらの 3 つのスローガンを組み合わせると矛盾しているようである．一体なぜだろうか……というと，簡単な話で，スローガン的な主張というものは，同じ言葉をほんのちょっと違う意味で用いていたり，前提条件が必要なのにそれを省略して書かれていたりするものである．とくに，数理論理学や数学基礎論では，矛盾が起きないギリギリの瀬戸際を攻めている議論が多いので，少し前提を間違えたりすると簡単に矛盾が発生する．そういうわけで，文脈の異なる複数のスローガンを組み合わせると，大体このように矛盾が起きるものである．つまるところ，正確な主張を知らないスローガンを丸暗記しても仕方がないので，正確な主張を知るつもりがなければ完全に忘却してしまった方がよい．そういうわけで，

「わかりやすいスローガンは基本的には正しくない」

という，わかりやすいスローガンをここに掲げておくことにする．

さて，それではトリックを説明しよう．定理 1.9 で述べた形式化では，選択公理とは，要素を持

つ集合の族 $\{A_x\}_{x\in\Lambda}$ の直積は要素を持つ，という主張であった：

$$\left(\exists f\colon \Lambda \to \bigcup_{x\in\Lambda} A_x\right)(\forall x\in\Lambda)\ f(x)\in A_x. \tag{2.1}$$

定理 1.9 で示したように，もし $\Lambda$ が記号の集合ならば，この形式 (2.1) の選択公理は構成的に正しいと考えてよさそうであった．ところで，選択公理には同値な形式化が多数ある．そのうち，最も有名な形式のひとつは次のようなものである．要素を持つ集合の族 $\{A_x\}_{x\in\Lambda}$ に対して，次が成立する．

$$\left(\exists f\colon \{A_x\}_{x\in\Lambda} \to \bigcup_{x\in\Lambda} A_x\right)(\forall x\in\Lambda)\ f(A_x)\in A_x. \tag{2.2}$$

後者の形式 (2.2) の選択公理の方が論点が明確にわかりやすいので，これについて考察しよう．前者では，選択関数 $f$ はインデックス $x$ 毎に $f(x)\in A_x$ を選んでいたが，後者においては集合 $A_x$ 毎に $f(A_x)\in A_x$ を選んでいる．当然のことながら，$f$ は関数なので，$a=b$ ならば $f(a)=f(b)$ でなければならない．ところで，たとえ $x\neq y$ であったとしても，たまたま $A_x$ と $A_y$ に属す要素が等しいということは有り得る．すると，**外延性公理** (*axiom of extensionality*) より，つまりそれは $A_x=A_y$ ということであるから，$f(A_x)=f(A_y)$ でなければならない．このような選択公理と外延性公理の組合せを**外延的な選択公理**と呼ぶことにしよう．

この外延的な選択公理が超越的である理由は以下である．われわれは何か $A_x$ と $A_y$ を定義し，それらが空でないことを保証したとしよう．つまり，我々は $A_x$ と $A_y$ から要素を取り出すことは可能である．ただし，$A_x$ と $A_y$ に入っている要素が同じであるか否かはよく分からない．外延的な選択公理が要求することは，もし**偶然** $A_x$ と $A_y$ の要素が一致しているとしたら，それらから取り出す要素は同じでなければならない．

こう説明すると，計算論的には少し妙な気持ちを覚える選択公理であるが，これが最も標準的な選択公理の形式化であるから仕方がない．たとえば，同値関係が与えられたとき，各同値類から代表元を取り出す際などには，この形式の選択公理が用いられる．1975 年，ディアコネスク (Radu Diaconescu) は，外延性公理を含む構成的集合論の体系 CZF においては，選択公理が排中律を導くことを証明した．CZF については本講義のスコープを逸脱するので，少し仮定を曖昧にして，外延的な選択公理から排中律を導く方法を述べることにする．

**定理 2.4** (ディアコネスクの定理)**.** 外延性公理と選択公理を組み合わせると排中律を導く．

*Proof.* 閉論理式 $A$ に対して，$A\vee\neg A$ を証明しよう．いま，$i\in\{0,1\}$ について，

$$E_i=\{x\in\{0,1\}:A\vee(x=i)\}$$

と定義する．明らかに $i\in E_i$ なので，各 $E_i$ から要素を取り出せる．よって，選択公理 (2.2) から選択関数 $f(E_i)\in E_i$ を得る．ここで，$E_0=E_1$ ならば $f(E_0)=f(E_1)$ である．

まず，$f(E_i) \in E_i \subseteq \{0,1\}$ であるから，自明に $f(E_i) = 0$ または $f(E_i) = 1$ が成立している．特に $f(E_0) = f(E_1)$ または $f(E_0) \neq f(E_1)$ である．次の主張を示す．

$$f(E_0) = f(E_1) \to A,$$
$$f(E_0) \neq f(E_1) \to \neg A.$$

どちらかの $i \in \{0,1\}$ について $f(E_i) \neq i$ の場合は，明らかに $A$ が成立していなければならない．特に，上の主張の一行目が成立する．残るパターンは，$f(E_0) \neq f(E_1)$ の場合である．このとき，$\neg A$, つまり $A \to \bot$ が成立していることを示す．もし前提の $A$ が成立しているならば，外延性公理より $E_0 = E_1 = \{0,1\}$ である．すると $f(E_0) = f(E_1)$ であるから，仮定より矛盾 $\bot$ が導かれた．つまり，$\neg A \equiv A \to \bot$ が成立する．

以上より，いずれのパターンでも $A$ または $\neg A$ が成り立つことが示された．したがって，排中律 $A \vee \neg A$ が成立する． □

念のため注意しておくと，形式 (2.1) の選択公理についても同様の議論が通じる．数学的対象は記号的世界の商（あるいは記号的世界 $M$ 上の同値関係 $E$）として表されるので，インデックスの集合が商集合 $\Lambda = M/E$ の形（たとえば，$\Lambda = \{A_x\}_{x \in I}$ のようなもの）ということもあり得る．このような $\Lambda$ に対する選択公理 (2.1) は，外延性公理と組み合わせると，同値関係 $E$ に従う外延的な選択関数を生み出すが，これは一般的には構成的には成し遂げられない．ただし，インデックスの集合 $\Lambda$ が記号的世界であれば，外延性公理の有無に関わらず，定理 1.9 のように選択公理 (2.1) は構成的に真と考えてもよかった．これはインデックスの集合 $\Lambda$ が記号的世界であっても，外延性を要求すると非構成的な，選択公理 (2.2) とは対照的である．

余談であるが，外延性の例としてよく取り上げられるものとして，「明けの明星」と「宵の明星」がある．「明けの明星」と「宵の明星」は，結果的には「金星」という同じ対象を指すことが分かったが，人類は初めからそう考えていたわけではない．歴史のある段階で，「明けの明星」と「宵の明星」が同一の惑星を指していることが認識されたのである．このように「明けの明星」と「宵の明星」は別の名が与えられたものの同値性が既に証明された例であるが，その他に，違う名を与えているが我々は未だ同じものか否か知らないものが無数にある．たとえば，P という名の与えられたものと NP という名の与えられたものが等しいか否か，我々は未だ知らず，これは $\mathsf{P} \stackrel{?}{=} \mathsf{NP}$ 予想と呼ばれる．このように「違う呼び名」で「同じ対象」を指しているかもしれないという可能性を考慮しなければならないとき，構成的には様々な困難が起こり得るのである．

## 2.1. マッカーティ実現可能性 ⋆

論理式の正しさの保証するという試みにおいて，あくまで各記号をどう理解するかには色々な方法があり，その考え方毎に実現可能性がある．前節までではクリーネ実現可能性を紹介したが，他にもクライゼルによる修正実現可能性 (modified realizability) など幾多の実現可能性がある．集合記号 $\in$ と等号記号 $=$ を外延的に解釈する実現可能性，つまり外延性公理を正しいものとして解釈する実現概念のうちひとつは，マッカーティ実現可能性 (McCarty realizability) として知られる．

まず，部分結合子代数 $M$ において，$M$-**集合**という概念を再帰的に定義する．$M$-集合とは，以下の条件を満たす任意の集合 $\dot{x}$ である．

$$\dot{x} \subseteq \{\langle \dot{u}, p\rangle : \dot{u} \text{ は } M\text{-集合 かつ } p \in M\}.$$

ここで $M$-集合は一般的には $M$ の元ではないので，区別するために $\dot{x}$ のような記法を用いている．文脈から明らかな場合は，しばしばドット記号は省略する．

さて，$M$-集合全体のなすクラスを $V^{(M)}$ と書くことにしよう．集合論の**強制法** (*forcing*) を知っている人には，これは $\mathbb{P}$-**名** ($\mathbb{P}$-*name*) のようなものだと思ってもよい．強制半順序 $\mathbb{P}$ からブール値モデルを得られるように，部分結合子代数 $M$ による実現可能性からはハイティング値モデルを得られる．たとえば，クリーネ実現可能性では，各 $||A|| \in \mathcal{P}(M)$ を論理式 $A$ の “真理値” と考えていたが，実際に $\mathcal{P}(M)$ 上にハイティング代数の構造が入っていることを確認できる．以後，$\langle \dot{u}, p\rangle \in \dot{x}$ であることを，しばしば $p \in ||\dot{u} \mathrel{\dot{\in}} \dot{x}||$ であるとか $p \ \mathbf{r}\ \dot{u} \mathrel{\dot{\in}} \dot{x}$ と書く．厳密には $\dot{\in}$ は本物の要素関係 $\in$ とは異なるものであるから，以後も $\dot{\in}$ と $\in$ を区別していることについては注意する．

**豆知識．**マッカーティ自身は $M$-集合をポアンカレ的集合 (Poincaré set) としても説明している．フランスの数学者ポアンカレは，数学から超越性を排除しようと試み，その思想はフランス経験主義あるいは前直観主義 (pre-intuitionism) として知られる．ポアンカレが述べたことは，集合 $X$ というものは構成的に記述されているべきで，それはつまり，領域内の各 $u$ について $u \in X$ を証明する**証拠** $p$ を必ず伴っているということである．このポアンカレのアイデアは，マッカーティの $M$-集合の定義と全く等しい．

さて，クリーネ実現可能性においては，$\forall x A(x)$ の証拠として，任意の $x$ について $px$ が $A(x)$ の証拠となるような $p$ を考えていた．しかし，マッカーティ実現可能性は集合を対象とし，$x \in M$ ではなく $\dot{x} \in V^{(M)}$ を考える必要がある．ところが，$M$ 上の部分マグマ演算は $M$-集合に対しては定義されていないため，このとき $p\dot{x}$ という記法は意味をなさない．実際，集合という概念を考えるにあたって重要なものは外延性であるから，$A(x)$ の証拠が $x$ の名前 $\dot{x}$ に依存するのは好ましくない．つまり，$A(x)$ の証拠は $x$ の外延のみに依存すべきである．

いま，$M$-集合 $\dot{x}, \dot{y} \in V^{(M)}$ が与えられているとき，マッカーティ実現可能性における $\dot{x} \in \dot{y}$, $\dot{x} \subseteq \dot{y}$, $\dot{x} = \dot{y}$ の “真理値” は次によって与えられる．

$$\begin{aligned}
||\dot{x} \in \dot{y}|| &= \{\langle p, q\rangle \in M : (\exists \dot{v})\ [p \in ||\dot{v} \mathrel{\dot{\in}} \dot{y}|| \ \wedge\ q \in ||\dot{x} = \dot{v}||]\}. \\
||\dot{x} \subseteq \dot{y}|| &= \{p \in M : (\forall \dot{u}, q)\ [q \in ||\dot{u} \mathrel{\dot{\in}} \dot{x}|| \ \rightarrow\ pq \in ||\dot{u} \in \dot{y}||]\}. \\
||\dot{x} = \dot{y}|| &= ||(\dot{x} \subseteq \dot{y}) \wedge (\dot{y} \subseteq \dot{x})|| = ||\dot{x} \subseteq \dot{y}|| \times ||\dot{y} \subseteq \dot{x}||.
\end{aligned}$$

つまり，記号 $\dot{x} \in \dot{y}$ は「$\dot{x}$ は $\dot{y}$ のある元 $\dot{v}$ と外延的に等しい」と解釈し，記号 $\dot{x} \subseteq \dot{y}$ は「$\dot{x}$ の任意の元 $\dot{u}$ は $\dot{y}$ の元と外延的に等しい」と解釈する．この解釈は，たとえば強制法のブール値モデルにおける要素関係および等号の解釈の方法とほとんど同じである．

それ以外の論理式について，集合量化以外については，クリーネ実現可能性と全く同じように定義する．集合量化については，上で述べたように $M$ 上の部分マグマ演算は $M$-集合に対しては

定義されていないから，以下のように修正を加える．

$$\begin{aligned}
||\exists x \in a\ A(x)|| &= \{\langle p, q\rangle : (\exists x)\ [p \in ||x \mathrel{\dot\in} a|| \ \wedge\ q \in ||A(x)||]\} \\
||\forall x \in a\ A(x)|| &= \{p : (\forall x, q)\ [q \in ||x \mathrel{\dot\in} a|| \ \to\ pq \in ||A(x)||]\} \\
||\exists x A(x)|| &= \{p : (\exists x)\ p \in ||A(x)||\} \\
||\forall x A(x)|| &= \{p : (\forall x)\ p \in ||A(x)||\}
\end{aligned}$$

これがマッカーティ実現可能性の定義である．クリーネ実現可能性と同じように，$p \in ||A||$ のことを $p\ \mathbf{r}\ A$ と書く．これを強制法の類似物として，$p \Vdash A$ と書く流儀もある．

マッカーティ実現によって，直観主義集合論の公理系 IZF を実現できると知られている．いくつか集合の構成の例を挙げると，たとえば集合論の公理系では，集合族 $\mathcal{S}$ が与えられたとき和集合 $\bigcup\mathcal{S} = \{y : (\exists X \in \mathcal{S})\ y \in X\}$ を構成される操作が認められている．実現の文脈では，あくまで $M$-集合 $\mathcal{S}$ の和 $M$-集合 $\dot\bigcup\mathcal{S}$ を構成することによって，和集合公理を実現することとなる．具体的には，次のように定義される．

$$\langle a, b\rangle\ \mathbf{r}\ y \mathrel{\dot\in} \dot\bigcup\mathcal{S} \iff \exists X\ [a\ \mathbf{r}\ X \mathrel{\dot\in} \mathcal{S} \quad \& \quad b\ \mathbf{r}\ y \mathrel{\dot\in} X].$$

これによって和集合公理を実現可能であることを示すのは，難しくはないが，かなり面倒な単純作業である．さて，和集合公理などは元々それなりには構成的な公理であるから，それを計算的に実現できることは驚くに値しないかもしれない．しかし，べき集合公理 (powerset axiom) も計算的に実現できるとなると，それはかなり非自明なことである．べき集合公理とは，与えられた集合のべき集合（部分集合全体の集合）の存在を認める公理である．

まず，通常のフォン・ノイマン宇宙 $V$ の累積的階層 $(V_\alpha)_{\alpha\in\mathrm{Ord}}$ を思い出そう．これは空集合 $V_0 = \varnothing$ から始まり，べき集合操作を積み重ねて集合論的宇宙を得る構成である．つまり，$V_{\alpha+1} = \mathcal{P}(V_\alpha)$ として定義し，さらに極限順序数 $\alpha$ については $\bigcup_{\beta<\alpha} V_\beta$ とする．$M$-集合の宇宙も同様の方法によって累積的階層 $(V_\alpha^{(M)})_{\alpha\in\mathrm{Ord}}$ をなす．階数 $\alpha$ 以下の $M$-集合 $A \in V_\alpha^{(M)}$ が与えられたとき，階数 $\alpha$ における $A$ のべき $M$-集合 $\dot{\mathcal{P}}_\alpha(A)$ の概念を用いることによって，べき集合公理を実現する．

$$q\ \mathbf{r}\ B \mathrel{\dot\in} \dot{\mathcal{P}}_\alpha(A) \iff [B \in V_\alpha^{(M)} \quad \& \quad q\ \mathbf{r}\ \forall x\ (x \in B\ \to\ x \in A)].$$

つまり，階数 $\alpha$ 以下の $M$-集合 $B$ 毎に，$B$ が $A$ の部分集合であることの証拠 $q$ を付随させた対 $\langle B, q\rangle$ 全体である．もうひとつ，分出公理 (separation axiom) についても触れておこう．これは，集合 $a$ と論理式 $A$ が与えられたら，集合 $\{x \in a : A(x)\}$ を構成できるという公理である．これを実現するためには，$M$-集合 $a$ と論理式 $A$ に対して，次の集合 $a|_A$ を構成する．

$$\langle p, q\rangle\ \mathbf{r}\ x \mathrel{\dot\in} a|_A \iff [p\ \mathbf{r}\ x \mathrel{\dot\in} a \quad \& \quad q\ \mathbf{r}\ A(x)].$$

つまり，$x$ が集合 $\{z \in a : A(z)\}$ に属すと主張するためには，$x \mathrel{\dot\in} a$ の証拠 $p$ と $A(x)$ の証拠 $q$ を用意する必要がある，ということである．

**注意．**通常の集合論においては，たとえば分出公理を用いれば，$\{\langle p, n\rangle \in \mathbb{N} : \{\!\{p\}\!\}(n)\downarrow\}$ のような $\mathbb{N}$ の部分集合を得ることができ，停止問題の決定不可能性より，その特性関数は計算不可能である．一方，分出公理

を実現するという文脈では，論理式の正しさの証拠を持ってくる必要があるが，上記の集合の場合には，それは計算の停止性の証拠である．つまり，少なくともプログラム $p$, 入力 $n$, 計算 $\{\!\{p\}\!\}(n)$ が停止するまでのステップ数 $s$ の情報を持つ．ところで，そのような 3 つ組 $\langle p, n, s\rangle$ 全体の集合の特性関数は計算可能である．したがって，計算の停止性に分出公理を適用して停止性集合を作り上げたとしても，実現の文脈では，それは計算不可能な集合の存在を導かない．

さて，他にも集合論には多数の公理があるが，おおよそのアイデアは掴めたであろうから，残りの公理の実現のアイデアについては省略することにする．

さて，マッカーティ実現を利用することで，集合に言及する様々な式の実現を厳密に議論することができるようになる．たとえば，上では曖昧な仮定の下でディアコネスクの定理の証明のスケッチを書いたが，マッカーティ実現可能性を用いれば，ディアコネスクの定理を実現可能である，ということを数学的に厳密な意味で証明することができる．より正確には，「外延性公理」がマッカーティ実現可能であり，さらに「選択公理は排中律を導く」もマッカーティ実現可能となる．しかし，マッカーティ実現可能性では，$\in$ や $=$ などの解釈が複雑であるため，実現を具体的に書き下すためにはかなりの労力が必要である．つまり，原理的にはディアコネスクの定理の実現を作れるとはいえど，実際に実現を書き下すのは容易ではない．

## §3. 論理式から型構造そして空間へ

部分結合子代数 $M$ が与えられたとき，任意の $p \in M$ は，$\{\!\{p\}\!\}(x) = px$ によって定義される $M$ 上の部分関数 $\{\!\{p\}\!\}\colon\subseteq M \to M$ と同一視できる．すると，$p$ が $A \vdash B$ を実現しているということは，これが $||A||$ から $||B||$ への関数 $\{\!\{p\}\!\}\colon ||A|| \to ||B||$ を与えている，ということである：

$$p \ \mathbf{r} \ A \vdash B \quad \sim \quad \{\!\{p\}\!\}\colon ||A|| \to ||B||$$

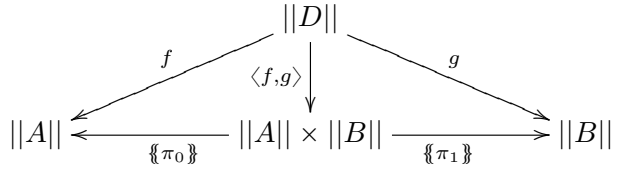
いま，$||A||$ と $||B||$ はあくまで関数の始域と終域なのであるから，論理式というよりも**空間**あるいは**型**であると考えることにする．空間として見ると，たとえば論理結合子 $\wedge$ は直積に対応していた，つまり $||A \wedge B|| = ||A|| \times ||B||$ であった．論理結合子 $\wedge$ に対する導入規則の実現を空間的に見ると，以下のようになる．

$$\frac{a \ \mathbf{r} \ D \vdash A \qquad b \ \mathbf{r} \ D \vdash B}{\lambda x.\langle ax, bx\rangle \ \mathbf{r} \ D \vdash A \wedge B} \ (\wedge I) \qquad \frac{f\colon ||D|| \to ||A|| \qquad g\colon ||D|| \to ||B||}{\langle f, g\rangle\colon ||D|| \to ||A \times B||} \ (\wedge I)$$

ここで，関数 $f, g$ に対して，$\langle f, g\rangle$ を $\langle f, g\rangle(x) = \langle f(x), g(x)\rangle$ によって定義している．同様に，論理結合子 $\wedge$ に対する除去規則の実現を空間的に見ると，以下のようになる．

$$\frac{c \ \mathbf{r} \ D \vdash A_0 \wedge A_1}{\lambda x.\pi_i(cx) \ \mathbf{r} \ D \vdash A_i} \ (\wedge E) \qquad \frac{h\colon ||D|| \to ||A_0|| \times ||A_1||}{\{\!\{\pi_i\}\!\} \circ h\colon ||D|| \to ||A_i||} \ (\wedge E)$$

また，具体的に作った実現子の性質から，図式的に表すと以下のようになっている．



つづいて，論理結合子 $\vee$ は余積に対応していた，つまり $||A \wedge B|| = ||A|| \sqcup ||B||$ であった．論理結合子 $\vee$ に対する導入規則の実現を空間的に見ると，以下のようになる．

$$\frac{c \ \mathbf{r} \ D \vdash A_j}{\lambda x.\langle \underline{j}, cx\rangle \ \mathbf{r} \ D \vdash A_0 \vee A_1} \ (\vee I) \qquad \frac{h\colon ||D|| \to ||A_j||}{\iota_j \circ h\colon ||D|| \to ||A_0|| \sqcup ||A_1||} \ (\vee I)$$

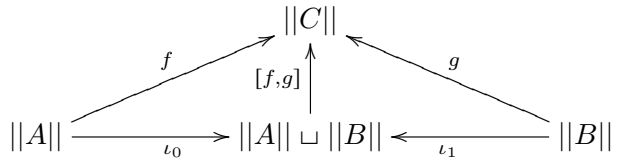
ここで，$\iota_j(x) = \langle j, x\rangle$ として定義している．同様に，論理結合子 $\vee$ に対する除去規則の実現を思い出そう．

$$\frac{p \ \mathbf{r} \ A \vee B \qquad s \ \mathbf{r} \ A \vdash C \qquad t \ \mathbf{r} \ B \vdash C}{[s,t]p \ \mathbf{r} \ C} \ (\vee E)$$

上式の $p \ \mathbf{r} \ A \vee B$ を下式に移動すると，下式は $[s,t] = \lambda p.[s,t]p \ \mathbf{r} \ A \vee B \to C$ となる．この実現と，それを空間的に見たものを並べてみよう．

$$\frac{s \ \mathbf{r} \ A \vdash C \qquad t \ \mathbf{r} \ B \vdash C}{[s,t] \ \mathbf{r} \ A \vee B \to C} \qquad \frac{f\colon ||A|| \to ||C|| \qquad g\colon ||B|| \to ||C||}{[f,g]\colon ||A|| \sqcup ||B|| \to ||C||}$$

先程と同様に，具体的に作った実現子の性質から，図式的に表すと以下のようになっている．




つづいて，論理式 $\to$ は関数空間に対応していた．つまり，

$$||A \to B|| = ||B||^{||A||} := \{p \in M : (\forall x \in ||A||)\ px \downarrow \in ||B||\}.$$

ところで，$\to$ の導入規則について，$\Gamma$ が空の場合には次を得る．

$$\frac{f \ \mathbf{r} \ A \to B \qquad x \ \mathbf{r} \ A}{fx \ \mathbf{r} \ B} \ (\to I)$$

しかし，これを利用すると，明らかに次の式のような実現を得られる：

$$\frac{\dfrac{\langle f, x\rangle \ \mathbf{r} \ (A \to B) \wedge A \implies fx \ \mathbf{r} \ B}{p \ \mathbf{r} \ (A \to B) \wedge A \implies \pi_0 p(\pi_1 p) \ \mathbf{r} \ B}}{\varepsilon := \lambda p.\pi_0 p(\pi_1 p) \ \mathbf{r} \ A \to B, A \vdash B}$$

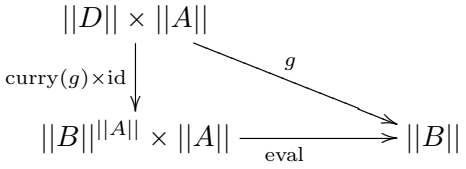
最下式に注目すると，これは以下のような空間上の関数を表す：

$$\mathrm{eval}\colon ||B||^{||A||} \times ||A|| \to ||B||.$$

ここで，$\mathrm{eval}(f,x) = \{\!\{f\}\!\}(x)$ となることに注意する．最後に，$\to$ の除去規則の実現を空間的に見れば，以下を得る．

$$\frac{p \ \mathbf{r} \ D, A \vdash B}{\mathsf{curry}(p) \ \mathbf{r} \ D \vdash A \to B} \qquad \frac{g\colon ||D|| \times ||A|| \to ||B||}{\mathrm{curry}(g)\colon ||D|| \to ||B||^{||A||}}$$

これもまた具体的に作った実現子の性質から，図式的に表すと以下のようになっている．




ところで，空間 $||A||$ と $||B||$ から関数空間 $||B||^{||A||}$ を作れたが，この関数空間構成は幾度でも積み重ねられるので，高階の関数空間を作ることができる．関数空間構成を多重に行うとき，関数空間を $Y^X$ と記述すると，縦に積み上がっていき，少々見づらくなってしまうので，以後は部分結合子代数 $M$ で関数空間を構成しているということも明示し，以下の記法を用いる．

$$M(X \to Y) := \{p \in M : (\forall x \in X)\ px \downarrow \in Y\}.$$

たとえば，部分結合子代数 $M$ の中では，自然数 $\mathbf{N} = \{\underline{n} \in M : n \in \mathbb{N}\}$ を作ることができた．これを初期空間 $\mathrm{HRO}_{\mathbf{0}}^{M}$ として，たとえば，次のような高階関数空間を考えることができる．

$$\begin{aligned}
\mathrm{HRO}_{\mathbf{1}} &:= M(\mathbf{N} \to \mathbf{N}) = \{p \in M : (\forall n \in \mathbf{N})\ pn \downarrow \in \mathbf{N}\} \\
\mathrm{HRO}_{\mathbf{2}} &:= M(\mathrm{HRO}_{\mathbf{1}} \to \mathbf{N}) = \{p \in M : (\forall q \in \mathrm{HRO}_{\mathbf{1}})\ pq \downarrow \in \mathbf{N}\} \\
\mathrm{HRO}_{\mathbf{3}} &:= M(\mathrm{HRO}_{\mathbf{2}} \to \mathbf{N}) = \{p \in M : (\forall q \in \mathrm{HRO}_{\mathbf{2}})\ pq \downarrow \in \mathbf{N}\}
\end{aligned}$$

つまり，$M$ がクリーネの第一代数 $\underline{\mathrm{K}}_1$ の場合には，まず $\mathrm{HRO}_{\mathbf{1}}$ は自然数上の計算可能関数（のコード）全体のなす空間である．ここで $\mathrm{HRO}_{\mathbf{1}} = M(\mathbf{N} \to \mathbf{N})$ と書かれているが，$M$ の元によって実現されるものは関数の極一部なので，関数空間 $M(\mathbf{N} \to \mathbf{N})$ は関数全体の集合 $\mathbb{N}^{\mathbb{N}}$ より遥かに小さいことに注意する．次に，$\mathrm{HRO}_{\mathbf{2}}$ は自然数上の計算可能関数（のコード）を入力として自然数を出力する計算可能型 2 汎関数（のコード）全体のなす空間である．そして，$\mathrm{HRO}_{\mathbf{3}}$ はそのような計算可能型 2 汎関数（のコード）を入力して自然数を出力する計算可能型 3 汎関数（のコード）全体のなす空間である．一般に，次のような高階関数空間を定義できる．

**定義 3.1.** 高階関数空間 $\mathrm{HRO}_\sigma$ を以下のように帰納的に定義する．

$$\begin{aligned}
\mathrm{HRO}_{\sigma\to\tau} &:= M(\mathrm{HRO}_\sigma \to \mathrm{HRO}_\tau) = \{p \in M : (\forall q \in \mathrm{HRO}_\sigma)\ pq \downarrow \in \mathrm{HRO}_\tau\} \\
\mathrm{HRO}_{\sigma\times\tau} &:= \mathrm{HRO}_\sigma \times \mathrm{HRO}_\tau = \{\langle p, q\rangle \in M : p \in \mathrm{HRO}_\sigma \ \wedge\ q \in \mathrm{HRO}_\tau\}.
\end{aligned}$$

部分結合子代数 $M$ がクリーネの第一代数 $\underline{\mathrm{K}}_1$ の場合には，ある高階関数空間 $\mathrm{HRO}_\sigma$ の元を**遺伝的計算可能汎関数**あるいは遺伝的再帰作用素 (*hereditarily recursive operator*) と呼ぶ．

たとえば，$\mathrm{HRO}_{\mathbf{3}} = \mathrm{HRO}_{\mathbf{2}\to\mathbf{0}} = \mathrm{HRO}_{(\mathbf{1}\to\mathbf{0})\to\mathbf{0}} = \mathrm{HRO}_{((\mathbf{0}\to\mathbf{0})\to\mathbf{0})\to\mathbf{0}}$ が成立している．遺伝的計算可能型 $n+1$ 汎関数は，すべての型 $n$ 汎関数を入力として受け付けることはできず，遺伝的計算可能なもののみを入力とすることに注意する．これが遺伝的と言われる所以である．前節では一階ハイティング算術の実現について議論したが，この高階関数空間の型構造は高階ハイティング算術のモデルを与える．

HRO の元はあくまで高階汎関数のコードであるが，高階汎関数の実体も取り扱えるようにしておくと便利である．つまり，自然数のコード全体の集合 $\mathbf{N}$ を $\mathbb{N}$ と同一視してしまい，計算可能関数のコードの空間 $\mathrm{HRO}_{\mathbf{1}}$ の代わりに，その実体である計算可能関数の空間

$$[\![\mathrm{HRO}]\!]_1 = \{\{\!\{p\}\!\} \colon \mathbb{N} \to \mathbb{N} \mid p \in \mathrm{HRO}_1\} \subseteq \mathbb{N}^{\mathbb{N}}$$

を考えたい．さて，ここまでは良いのであるが，たとえば $\mathrm{HRO}_2$ を考えたときに少し困ったことが起きる．遺伝的計算可能型 2 汎関数の定義域は $[\![\mathrm{HRO}]\!]_1 \subseteq \mathbb{N}^{\mathbb{N}}$ であり値域は $\mathbb{N}$ であると考えられる．したがって，$p \in \mathrm{HRO}_2$ が意味するものは部分汎関数 $[\![p]\!] \colon \subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, より正確には $[\![p]\!] \colon [\![\mathrm{HRO}]\!]_1 \to \mathbb{N}$ であると想像するかもしれない．言い換えれば，各 $p \in \mathrm{HRO}_2$ が表す型 2 汎関数 $[\![p]\!]$ とは，型 1 汎関数 $\{\!\{q\}\!\}$ を入力として，出力となる自然数は次によって与えられると考えられるだろう．

$$[\![p]\!](\{\!\{q\}\!\}) = n \iff p \cdot q = \underline{n}$$

そうすると，次のような図式が成立していることが期待できる．

$$\begin{array}{lccc} \text{数学的実体:} & [\![\mathrm{HRO}]\!]_1 & \xrightarrow{[\![p]\!]?} & \mathbb{N} \\ & {\scriptstyle q \mapsto \{\!\{q\}\!\}}\uparrow & & \uparrow{\scriptstyle \underline{n} \mapsto n} \\ \text{記号的世界:} & \mathrm{HRO}_1 & \xrightarrow[q \mapsto pq]{} & \mathbf{N} \end{array}$$

ところが，入力 $q_0, q_1 \in \mathrm{HRO}_1$ が同じ関数 $\{\!\{q_0\}\!\} = \{\!\{q_1\}\!\}$ を計算するとしても，$pq_0 = pq_1$ であるという保証はどこにもない．つまり，$[\![p]\!]$ は $\mathbf{N}^{\mathbf{N}}$ 上の関数としては well-defined ではない．

問題を明示化しよう．関数のコードの空間の元 $p, q \in \mathrm{HRO}_1 = M(\mathbf{N} \to \mathbf{N})$ から，関数 $\{\!\{p\}\!\}, \{\!\{q\}\!\} \colon \mathbf{N} \to \mathbf{N}$ が与えられる．しかし，任意の $x \in \mathbf{N}$ について $\{\!\{p\}\!\}(x) \equiv \{\!\{q\}\!\}(x)$ であるならば，$\{\!\{f\}\!\} = \{\!\{g\}\!\}$ である．このように，同じ関数を表すコード $p, q$ は同一視してしまうのが自然だろう．つまり，関数空間 $\mathrm{HRO}_1 = M(\mathbf{N} \to \mathbf{N})$ には，「同じ関数を表す」ということを意味する同値関係が自然に入る．これを**外延的同値関係** (*extensional equivalence relation*) と呼ぶ．

明らかな問題点として，遺伝的計算可能汎関数は外延的同値関係に従わない．つまり，たとえば $p \in \mathrm{HRO}_2$ であり，$a, b \in \mathrm{HRO}_1$ について $\{\!\{a\}\!\} = \{\!\{b\}\!\}$ であったとしても，$pa \neq pb$ であり得る．つまり，遺伝的計算可能汎関数は，外延的に等しい関数を入力しているにも関わらず出力が異なる，ということが有り得る．この問題を解消するものが，次の HEO の階層である．

**定義 3.2.** 高階関数空間 $\mathrm{HEO}_\sigma$ とその上の外延的同値関係 $\sim_\sigma$ を以下のように帰納的に定義する．

$$\begin{gathered} x \sim_{\mathbf{0}} y \iff x = y \\ \langle u, x\rangle \sim_{\sigma\times\tau} \langle v, y\rangle \iff (u \sim_\sigma v) \wedge (x \sim_\tau y). \\ \mathrm{HEO}_{\sigma\to\tau} = \{x : \forall u, v\, [u \sim_\sigma v \ \to\ xu \sim_\tau xv]\}. \\ x \sim_{\sigma\to\tau} y \iff \forall u, v\, [u \sim_\sigma v \ \to\ xu \sim_\tau yv] \end{gathered}$$

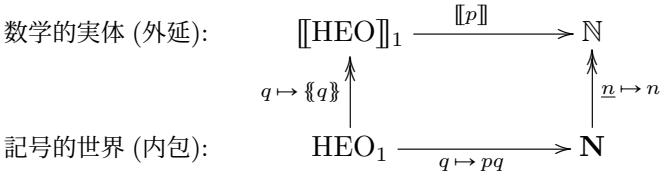
部分結合子代数 $M$ がクリーネの第一代数 $\underline{\mathrm{K}}_1$ の場合には，ある高階関数空間 $\mathrm{HEO}_\sigma$ の元を**遺伝的実効汎関数**あるいは遺伝的実効作用素 (*hereditarily effective operator*) と呼ぶ.

つまり, $(\mathrm{HEO}_\sigma, \sim_\sigma)$ から $(\mathrm{HEO}_\tau, \sim_\tau)$ への関数として矛盾なく定義されていると考えられるものが $\mathrm{HEO}_{\sigma\to\tau}$ の要素足り得るのである．たとえば, $\sim_{\mathbf{0}}$ は等号であるから, $\mathrm{HEO}_{\mathbf{1}} := \mathrm{HEO}_{\mathbf{0}\to\mathbf{0}} = \mathrm{HRO}_{\mathbf{1}}$ である．この $\mathrm{HEO}_{\mathbf{1}}$ 上の同値関係 $\sim_{\mathbf{1}} := \sim_{\mathbf{0}\to\mathbf{0}}$ は，自然数上の関数としての外延的同値性を意味する．つまり，次が成立していることは定義より明らかである.

$$p \sim_{\mathbf{1}} q \iff \{\!\{p\}\!\} = \{\!\{q\}\!\}.$$

次に $\mathrm{HEO}_{\mathbf{2}} := \mathrm{HEO}_{\mathbf{1}\to\mathbf{0}}$ を考えると，各 $p \in \mathrm{HEO}_{\mathbf{2}}$ の表すものは，遺伝的計算可能関数のコードを入力として自然数を返す関数のうち，外延的同値関係に従うものである．つまり，$q_0 \sim_{\mathbf{1}} q_1$ ならば $pq_0 = pq_1$ を満たすものである．したがって，$p \in \mathrm{HEO}_{\mathbf{2}} \subseteq \mathrm{HRO}_{\mathbf{2}}$ は，遺伝的計算可能関数を入力として自然数を返す型 2 汎関数 $[\![p]\!] \colon [\![\mathrm{HEO}]\!]_{\mathbf{1}} = [\![\mathrm{HRO}]\!]_{\mathbf{1}} \to \mathbb{N}$ を表す.

このように，記号的世界の内包的記述が，その外延となる数学的実体の上で（一価）関数を定義していると主張するためには，記号的世界で外延的同値関係を記述し，その外延的同値関係に従う必要がある.




さて，HEO の定義においては外延的同値関係を考えたが，他にも様々な同値関係を考えることができる．より一般に，自明な同値関係も考えれば，いま，我々の扱う空間はいずれも何らかの同値関係の入った集合と思うことができる．つまり，部分結合子代数において表現される空間とは，集合 $S \subseteq M$ と同値関係 $E$ の組 $(A, E)$ であるとする．これは，部分結合子代数 $M$ 上の**部分同値関係** (*partial equivalence relation*) を空間と思うことと同等である．ここで，$M$ 上の部分同値関係とは，対称律と推移律を満たす 2 項関係 $E \subseteq M \times M$ である：

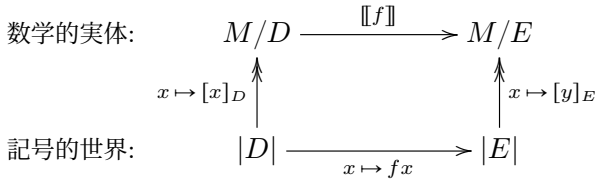
1. (対称律) $xEy \implies yEx$.
2. (推移律) $xEy \mathbin{\&} yEz \implies xEz$.

つまり，同値関係から反射律を除いたものである．$E$ が $M$ 上の部分同値関係ならば，これは台集合 $|E| := \{a \in M : aEa\}$ 上の同値関係と思うことができる．たとえば商集合 $M/E$ は，つまり $|E|/E$ を意味するものとする．一般に，部分同値関係 $D, E$ に対して，外延的部分同値関係 $[D \to E]$ を次によって定義する.

$$f[D \to E]g \iff (\forall x, y)\,[xDy \implies fx\,E\,gy].$$

つまり，部分同値関係 $[D \to E]$ の台集合 $|[D \to E]|$ に属す $f$ とは，部分同値関係 $D, E$ の意

図する数学的実体の上の関数 $[\![f]\!]$ を与える，ということである．




ここで，$[x]_D$ は $x \in |D|$ の $D$-同値類を意味する．つまり，$[x]_D = \{z \in M : zDx\}$ である．同様に，$[y]_E$ も $y \in |E|$ の $E$-同値類を表す．

かくして，数学的対象は，同値関係（およびその商）として定義されていく．このように部分結合子代数の理論では，数学的対象を記号世界にコード（実現）していくが，念のために注意しておけば，いかなる数学的理論においてもそれは同様であり，これは部分結合子代数の理論に特有のものではない．たとえば，公理的集合論においては，自然数は技巧的な方法で集合としてコードされ，実数もまた同様に，いわゆる実数の構成と呼ばれる方法によって，集合論の内部にコードされる．

部分結合子代数の理論によって行われることもまた同様である．たとえば数の構成を考えよう．われわれは既に，部分結合子代数において自然数を構成している．ここから整数を構成することは容易である．有理数は，整数の対に適切な同値関係を入れることによって構成される．そして，実数は，たとえば有理コーシー列上の同値関係として構成される．

このアイデアを拡張すると，部分結合子代数上の表現空間の理論が得られる．この理論においては，同値関係こそが最も基本的な空間概念だと考えることにより，記号世界をベースに数学を実現していくのである．

# 第 3 章

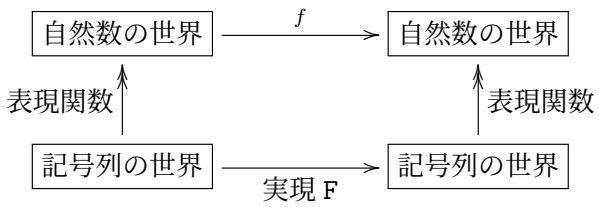
# 表現空間の理論

## §1. 表現空間と実現可能性

計算理論は，記号列によってコードされた数学的オブジェクトに対する計算を取り扱う分野である．この節では，文字列以外の数学的対象に関する様々な計算論を統一的に導入することを試みる．まず，自然数を記号列によって**表現**（コーディング）する，ということに立ち返って，表現（コーディング）とは何であるかの再理解を目指そう．

たとえば，チューリングマシン (Turing machine) は，基本的には，利用可能な記号の集合 $\Sigma$ を固定して，その上の語，つまり有限記号列 $\sigma \in \Sigma^*$ を入出力とする関数を議論するものである．このとき，自然数を 2 進表記というバイナリ列として表現することによって，自然数上の関数の計算理論を展開できる．これは，記号列がどんな自然数を表しているのか，という**意味**を与えることにより，ただの記号列上の関数に自然数上の関数としての意味を与えていた，ということである．もう少し正確には，$\mathtt{bin}(n)$ によって自然数 $n$ を自然数の 2 進表記を表すものとすれば，部分全射 $\mathtt{bin}(n) \mapsto n$ が「記号列に自然数としての意味を与える」関数となる．この関数を**表現関数**と呼ぶことにしよう．

たとえば，自然数上の関数 $f : \mathbb{N}^k \to \mathbb{N}$ がチューリングマシンによって計算可能であるとは，（表現関数 $\mathtt{bin}(n) \mapsto n$ を介することにより）それを記号列上の機械的操作 $\mathtt{F} :\subseteq (\Sigma^*)^k \to \Sigma^*$ として**実現** (realize) できるということである．

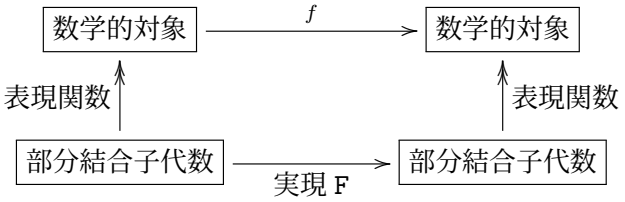



このような「**表現と実現を経由した計算**」が計算理論の基本である．有理数や代数的数は容易に自然数でコードできるから，そのような数に関する計算論も展開できる．また，その他にも幾らかの単純な可算構造ならばコーディングできそうである．それでは，実際，どのような数学的対象ならば，記号的に表現でき，そして良い計算論を展開できるだろうか．

歴史的には，1950 年代頃に，語あるいは自然数によるコーディングという概念自体を研究対象と

する**ナンバリングの理論** (*Theory of numbering*) が誕生する．また，それとは並行に，20 世紀中頃から，実数などの連続的概念を対象とする計算論である**計算可能解析学** (*computable analysis*) という分野も徐々に発展を見せた．計算理論とはデジタル（離散的）な概念を対象とするものであり，実数や複素数などの連続的概念は計算理論の対象外である……と思われがちだが，実際はそうではない．解析学と物理学における計算可能性理論の教科書として，1989 年に Pour-El と Richards によって執筆された “Computability in Analysis and Physics” は非常に有名である

それでは，このような連続的オブジェクトを含む様々な数学的対象は如何にして計算論的に取り扱われているだろうか．その 1 つの解答は，表現と実現に関する上の図式に少々の修正を加えるだけである．我々の記号の世界は，任意の**部分結合子代数**であり，これによって様々な数学的対象の上の計算は，部分結合子代数の単なる演算適用として実現される．




たとえば，部分結合子代数の内部で自然数という数学的対象を表現でき，初等的な自然数論を展開できることは，第 3 節で見たとおりである．ここからは，より一般的な数学的対象を考察しよう．まず，上の図式の縦方向，すなわち，部分結合子代数による数学的対象の表現は，以下のように定式化される．

**定義 1.1.** $M$ を部分結合子代数とする．このとき，集合 $X$ と部分全射 $\nu_X :\subseteq M \to X$ の対 $(X, \nu_X)$ を **$M$-表現空間** ($M$-*represented space*) と呼び，$\nu_X$ を $X$ の **$M$-表現** ($M$-*representation*) と呼ぶ．

これは，集合 $X$ の各要素が，部分結合子代数 $M$ の元によって名付けられたことを意味する．つまり，$\nu_X(a) = x$ であるとき，$a \in M$ は $x$ の **$\nu_X$-コード** (*code*) または **$\nu_X$-名** (*name*) と呼ばれる．表現関数 $\nu_X$ が文脈から明らかな場合は，単にコードまたは名と呼ぶ．1 つの元 $x \in X$ が複数のコードを持ち得る場合もあることに注意する．

ところで，前節では部分同値関係として空間を表現するというアイデアを解説した．定義 1.1 は部分同値関係と本質的に等しいということを説明しよう．まず，表現 $\nu_X : M \to X$ が与えられたら，部分結合子代数 $M$ 上に以下のような部分同値関係を定義できる．

$$a \sim_X b \iff a, b \in \mathrm{dom}(\nu_X) \ \ \& \ \ \nu_X(a) = \nu_X(b).$$

このとき，$a$ の $\sim_X$-同値類を $[a]_X$ と書けば，$[a]_X \mapsto \nu_X(a)$ によって商集合 $M/\sim_X$ と空間 $X$ を同一視できる．このため，$\nu_X(a)$ の代わりにしばしば $[a]_X$ と書くことがある．あるいは，型 $X$ と宣言された値 $a$ の評価と考え，$[\![a]\!]_X$ と書くこともある．

逆に，部分結合子代数 $M$ 上の部分同値関係 $E$ が与えられたとき，台集合 $|E| = \{a \in M : aEa\}$

を定義域とする商集合への全射 $a \mapsto [a]_E \colon |E| \to M/E$ が自動的に与えられる．したがって，商集合 $M/E$ は，$\nu_E \colon a \mapsto [a]_E$ を表現とする表現空間である．このようにして，表現空間と部分同値関係は等価な概念と考えることができる．

表現空間は，部分結合子に対して型あるいは空間という数学的実体を対応させるものを考えるが，この逆操作を考えよう．つまり，空間（数学的実体）から部分結合子（記号的世界）へ向かう方向，言い換えれば，数学的実体を記号的世界にコードするという観点から表現空間を理解することを試みる．まず，各 $x \in X$ は $M$ に属すコードによって実現される．このため，実現可能性の記法を借用して，$||x||_X$ によって $x$ のコード全体の集合を表すことにする．つまり，以下のように定義する．

$$||x||_X := \{a \in M : x = [a]_X\} = \{a \in M : x = \nu_X(a)\}.$$

あるいは実現可能性の言葉を使えば，$a$ が $x$ のコードである，という関係を「$a$ というコードによって $x$ が実現される」とみなし，$a \ \mathbf{r} \ x$ と書いてもよい．以上の定義についてまとめると，

- $a \mapsto [a]_X$ は，どの記号 $a \in M$ がどの数学的対象を表すかを指し示す表現写像であり，
- $x \mapsto ||x||_X$ は，各数学的対象 $x \in X$ に名を与える命名写像である．

記号によって数学的対象を表現することとは，すなわち，数学的対象に記号によって名を与えることであり，その逆もまたしかり，である．これらは同じものを別の側面から見ているにすぎない，表裏一体の関係にある．

**注意．実現可能性理論** (*realizability theory*) では，$(X, ||\cdot||_X)$ は**モデスト集合** (*modest set*) と呼ばれる．本稿では，しばしば表現空間 $(X, [\cdot]_X)$ とモデスト集合 $(X, ||\cdot||_X)$ を同一視し，後者のことも表現空間と呼ぶ．

**例 1.2.** 部分結合子代数 $M$ の中で常に自然数をコードできることは定義 3.7 において見た通りである．これを表現空間の言葉で言い直そう．定義 3.7 では，自然数 $n \in \mathbb{N}$ に対して，対応する $\underline{n} \in M$ を具体的に与えた．部分関数 $\nu_{\mathbb{N}} :\subseteq M \to \mathbb{N}$ を $\nu_{\mathbb{N}}(\underline{n}) \to n$ によって定義すれば，$(\mathbb{N}, \nu_{\mathbb{N}})$ は $M$-表現空間をなす．各 $n \in \mathbb{N}$ の $\nu_{\mathbb{N}}$-コードは $\underline{n} \in M$ のみであり，よって $||n||_{\mathbb{N}} = \{\underline{n}\}$ である．

**豆知識．** 表現が多価である場合もしばしば重要となる．集合 $X$ と部分多価全射 $\delta_X :\subseteq \mathbb{M} \rightrightarrows X$ の対 $(X, \delta_X)$ を **$M$-多価表現空間** ($M$-*multi-represented space*) と呼ぶ．実現可能性理論においては，これと同一な概念は**アセンブリ** (*assembly*) と呼ばれる．

つづいて，上の図式における，数学的対象上の写像の部分結合子代数上の関数による実現の部分である．クリーネ実現可能解釈において，論理結合子 $\to$ の実現が関数空間の構成に対応していたことを思い出そう．そして，関数の実現とは論理結合子 $\to$ の実現に他ならない，と考えることにする．具体的には，以下のように関数の実現を定義する．

**定義 1.3.** $\mathcal{X} = (X, \nu_X)$ と $\mathcal{Y} = (Y, \nu_Y)$ を $M$-表現空間とする．このとき，$\mathtt{f} \in M$ が $f \colon \mathcal{X} \to \mathcal{Y}$ の **$M$-実現子** ($M$-*realizer*) であるとは，任意の $x \in X$ に対して，$x$ のどんな $\nu_X$-コード $\mathtt{x} \in M$

が与えられても，$\mathtt{fx}$ が $f(x)$ の $\nu_Y$-コードを与えていることを意味する．つまり，

$$\mathtt{x} \in ||x||_X \implies \mathtt{fx} \in ||f(x)||_Y.$$

関数 $f\colon \mathcal{X} \to \mathcal{Y}$ が $M$-**実現可能** ($M$-*realizable*) とは，$f$ の $M$-実現子 $\mathtt{f} \in M$ が存在することを意味する．

上では表現空間の定義を実現可能解釈を用いて導入しているが，表現空間あるいは部分同値関係から直接的に定義することもできる．この場合，$\mathtt{f} \in M$ が $f$ の $M$-実現子であるとは，次を満たすことである．

$$\mathtt{x} \in \mathrm{dom}(\nu_X) \implies f([\mathtt{x}]_X) = [\mathtt{fx}]_Y$$

図式的に表せば，$f$ が $M$-実現可能であるとは，以下の図式を可換にする $\mathtt{f} \in M$ が存在することである：

$$\begin{array}{ccc} X & \xrightarrow{\quad f \quad} & Y \\ {\scriptstyle \nu_X}\uparrow & & \uparrow{\scriptstyle \nu_Y} \\ \subseteq M & \xrightarrow[\mathtt{x} \mapsto \mathtt{fx}]{} & \subseteq M \end{array}$$

この図式では $\mathtt{f}$ が $M$-実現子に相当するが，関数 $\mathtt{x} \mapsto \mathtt{fx}$ のことを $M$-実現子と呼ぶこともある．また，部分結合子代数 $M$ が文脈から明らかな場合には，$M$ を省略して単に実現子や実現可能などという．

**例 1.4.** 例 1.2 で定義した $M$-表現空間 $(\mathbb{N}, \nu_{\mathbb{N}})$ において，$n \mapsto n+1$ の具現化は $\underline{n} \mapsto \underline{n+1}$ であり，これは命題 3.8 より $\mathtt{succ} \in M$ によって実現可能である．つまり，$f : (\mathbb{N}, \nu_{\mathbb{N}}) \to (\mathbb{N}, \nu_{\mathbb{N}})$ を $f(n) = n+1$ で定義すれば，この $f$ は実現可能関数である．

部分結合子代数 $M$ に対し，しばしば $\mathsf{Mod}(M)$ によって $M$-表現空間と実現可能関数のなす圏を表すことがある．部分結合子代数による表現を考えるメリットは，部分結合子代数が関数適用の抽象化であり，任意の $M$-表現空間 $X, Y$ に対して，関数空間 $[X \to Y]$ もまた $M$-表現空間になるということである．専門用語を用いれば，$\mathsf{Mod}(M)$ は**デカルト閉圏** (*cartesian closed category*) をなすことが分かる．具体的には，以下のようにして関数空間は表現される．

**定義 1.5.** $M$-表現空間 $X, Y$ に対し，$X$ から $Y$ への $M$-実現可能関数全体の空間は次の関数 $\mathtt{f} \mapsto [\![\mathtt{f}]\!]_{X \to Y}$ によって表現できる．

$$[\![\mathtt{f}]\!]_{X \to Y} = f \iff (\forall \mathtt{x} \in \mathrm{dom}(\nu_X))\ [\mathtt{fx}]_Y = f([\mathtt{x}]_X).$$

つまり，関数 $f\colon X \to Y$ の名とは，$f$ の実現子 $\mathtt{f} \in M$ である．この表現空間を $M[X \to Y]$ と書く．$M$ が文脈から明らかな場合は，単に $[X \to Y]$ と書く．

事実，1950 年代に，クリーネは高階関数の計算理論を作り上げた．現代的な視点からは，クリーネの高階関数論の基本的な部分は，クリーネの第一または相対第二代数による表現空間の理論の一部として展開することができる．部分組合せ代数によって空間を表現することの有り難みは，その空間上の計算論を導入できるというだけでなく，様々な圏論的構成を受容できる，ということでもある．しかし，本稿では圏論についてはこれ以上深入りしない．

定義 1.5 では全域実現可能関数の空間を考えているが，部分実現可能関数も考えると都合がよい．表現空間 $X, Y$ が与えられたとき，コード $\mathtt{f}$ が部分関数 $f :\subseteq X \to Y$ を実現するのは，次の条件を満たすときである．

$$f(x)\downarrow \quad \Longrightarrow \quad (\forall \mathtt{x} \in ||x||_X)\ \ \mathtt{fx}\downarrow \in ||f(x)||_Y.$$

厳密に言えば，この定義だと 1 つのコード $\mathtt{f}$ が複数の部分関数 $f$ を表し得るので，これは多価表現と呼ばれるものになる．多価表現を考えても理論的には特に問題ないのだが，一応，これを一価表現に修正したものも考えよう．これは，コード $\mathtt{f}$ によって実現される部分関数 $f$ のうち極大なものだけに制限する，というものである．つまり，コード $\mathtt{f}$ が部分関数 $f :\subseteq X \to Y$ を実現するのは，次の条件を満たすときである．

$$f(x)\downarrow \quad \Longleftrightarrow \quad (\forall \mathtt{x} \in ||x||_X)\ \ \mathtt{fx}\downarrow \in ||f(x)||_Y.$$

この表現 $\mathtt{f} \mapsto f$ が与える表現空間を $M[\subseteq X \to Y]$ と書くことにする．ところで，関数空間の定義において，関数 $f$ という数学的実体からその実現であるコード $\mathtt{f}$ を構成するという方向で定義してきた．一方，記号的世界から数学的実体を与えるという方向で，この関数空間の定義を理解することもできる．この場合，関数空間 $M[X \to Y]$ は遺伝的実効作用素 $\mathrm{HEO}_\sigma$ の構成と全く等しい．より正確には，$\mathrm{HEO}_\sigma$ 上の同値関係 $\sim_\sigma$ から得られる表現空間である．つまり，

$$M[[\![\mathrm{HEO}]\!]_\sigma \to [\![\mathrm{HEO}]\!]_\tau] = [\![\mathrm{HEO}]\!]_{\sigma\to\tau} = \mathrm{HEO}_{\sigma\to\tau}/{\sim_{\sigma\to\tau}}$$

すると，遺伝的再帰作用素 HRO がいかなる表現空間を表すかというのは気になるところである．遺伝的再帰作用素 HRO と遺伝的実効作用素 HEO の違いが何であったかというと，遺伝的再帰作用素 HRO は，入力となる 2 つの名が同じモノを表すとしても，別のモノを表す名を出力してもよい．つまり，たとえば「明けの明星」を入力すると 0 を出力し，「宵の明星」を入力すると 1 を出力するということが許される．一方，遺伝的実効作用素 HEO の場合には，「明けの明星」と「宵の明星」は共に同じ「金星」の名であることから，出力値は等しくなければならない．

これを解釈すると，遺伝的再帰作用素 HRO とは，「明けの明星」と「宵の明星」の外延である「金星」に対して，0 と 1 の両方の出力値があり得る**多価関数** (*multi-valued function*) であるということである．言い換えれば，コード $\mathtt{f} \in M$ によって実現される数学的実体とは，以下のような多価関数である．

$$f(x) = \{[\mathtt{fx}]_Y : \mathtt{x} \in ||x||_X\}$$

この表現 $\mathtt{f} \mapsto f$ が与える表現空間を $M[X \rightrightarrows Y]$ と書く．二重矢印は，多価関数からなる空間であるというイメージを表している．また，前と同様にして，全域実現可能多価関数の表現空間だけでなく，部分実現可能多価関数の表現空間も考えると便利であり，これを $M[\subseteq X \rightrightarrows Y]$ と書くことにしよう．

## §2. ライスの定理と空間の連結性

本節では，1953 年にヘンリー・ライス (Henry G. Rice) によって証明された**ライスの定理** (*Rice's theorem*) を取り扱う．これは，次のような驚くべき主張をする定理である．

> 部分計算可能関数に関する非自明な性質 $P$ が任意に与えられている．このとき，与えられたプログラム $p$ の計算する関数が $P$ を満たすか否かを判定するアルゴリズムは存在しない．

チューリングの定理 (系 2.3) はあくまで「停止問題」というひとつの決定問題が計算不可能であることを示すものであるが，ライスの定理は一度で大量の決定問題の計算不可能性を導く．究極の計算不可能性定理の 1 つである．

プログラム $p$ によって計算される $\mathbb{N}$ 上の部分関数を $\{\!\{p\}\!\}$ と書いていたことを思い出そう．ライスの定理によれば，たとえば，以下のような集合への所属判定はすべて計算不可能である．

$$\{p \in \mathbb{N} : \{\!\{p\}\!\} \text{ は全域関数である.}\}$$
$$\{p \in \mathbb{N} : \{\!\{p\}\!\} \text{ は 2 値部分関数である.}\}$$
$$\{p \in \mathbb{N} : \{\!\{p\}\!\} \text{ の定義域は有限である.}\}$$
$$\{p \in \mathbb{N} : \{\!\{p\}\!\} \text{ は空関数である.}\}$$

ライスの定理は，一見すると非常に興味深いが，しかし，クリーネの再帰定理などとは異なり，定理自体の有用性は低く，理論的にも深みがあるわけではない．それにも関わらず，ライスの定理は何故か伝統的に計算可能性理論入門における必須トピックとして取り扱われる．これは，その定理の衝撃的な内容に比較すると，証明が極めて簡単であるからであろう．

しかし，それだけでは，必須トピックとして扱うには少し説得力が足りない．ここでは，ライスの定理の空間的側面，つまり，これが計算可能性理論の概念を空間的に理解するためには良いトピックであることを強調する．「計算と空間」というコンテキストに置くことで，ライスの定理は特筆すべき価値を持つ．本節における我々の目標は，ライスの定理に対する以下のような幾何学的イメージを持つことである．

> 自然数や文字列の世界は離散空間（デジタル）であるが，自然数や文字列上の部分計算可能関数全体を空間として見ると，そこは連結空間のようになっている．

この意味を理解するために，まず，数学において，離散と連結という言葉がどのように定義されていたかを思い出そう．

- 位相空間 $X$ が**離散** (*discrete*) とは，$X$ の任意の部分集合 $P$ が開集合であることを意味する．これは，$X$ から離散空間 $\mathbf{2} = \{0, 1\}$ への**任意の関数が連続である**ことと同値である．
- 位相空間 $X$ が**連結** (*connected*) とは，$X$ の非自明な開集合分割が存在しないことを意味する．これは，$X$ から離散空間 $\mathbf{2} = \{0, 1\}$ への**連続関数は定数関数しか存在しない**ことと同値である．

クリーネの第 2 代数で見たように，連続関数は計算概念の一種であるから，上の定義の連続を計算と読み替えてみよう．いま，$\chi_P\colon X \to \mathbf{2}$ を $P$ の特性関数とする．つまり，$x \in P$ ならば $\chi_P(x) = 1$ であり，$x \notin P$ ならば $\chi_P(x) = 0$ であるから，与えられた元 $x \in X$ が $P$ に属すか否かを判定することである．空間が離散ならば，特性関数 $\chi_P\colon X \to \mathbf{2}$ もまた連続である．つまり，離散性とは，与えられた $x \in X$ が $P$ に所属するか否かを決定できることを意味する．一方で，空間が連結である場合，$\chi_P$ が定数関数であるのは $P = \varnothing$ または $P = X$ のときのみであるから，$\varnothing \subsetneq P \subsetneq X$ ならば $\chi_P$ は連続ではない．つまり，与えられた $x \in X$ が $P$ に所属するか否かを決定することはできない．

そして，ライスの定理とは，クリーネの第 1 代数において，$\varnothing \subsetneq P \subsetneq [\subseteq \mathbb{N} \to \mathbb{N}]$ ならば $\chi_P\colon [\subseteq \mathbb{N} \to \mathbb{N}] \to \mathbf{2}$ は計算可能ではない，ということを述べるものである．つまり，与えられた部分計算可能関数が $P$ に属すか否かを判定する計算可能なアルゴリズムは存在しない．この発想を元に，連結性の定義を一般化しよう．

**定義 2.1.** $M$ を部分結合子代数とする．$M$-空間 $X$ が **$M$-連結** ($M$-*connected*) であるとは，$X$ から $\{\underline{0}, \underline{1}\}$ への実現可能関数が定数関数しか存在しないことを意味する．

特に，クリーネの第 1 代数 $\underline{\mathrm{K}}_1$ において連結であることを**計算可能連結** (*computably connected*) と呼ぶことにする．これは，$X$ から $\{0, 1\}$ への計算可能関数が定数関数しか存在しないことを意味する．言い換えれば，$P$ が $X$ の非自明な部分集合ならば，与えられた $x \in X$ が $P$ に属すか否かを判定する計算可能なアルゴリズムは存在しない，ということである．

**豆知識.** ちなみに，位相空間論的な代数，つまりクリーネの第 2 代数 $\underset{\sim}{\mathbf{K}}_2$ における連結性は，位相空間論の意味での連結性と同じ定義となる．一方，たとえば部分 $\Pi^1_1$-可測関数の代数では，実現可能な全域関数とはボレル可測関数のことであった．しかし，位相空間論的には，空間が連結（開集合で分割できない）であったとしても，ボレル集合でならば容易に分割できることは多々ある．実際，密着でない位相空間ならば，非自明な開集合とその補集合によって分割される．すると，一見，部分 $\Pi^1_1$-可測関数の代数における連結性は自明な概念に感じるかもしれない．しかし，あくまで位相空間論的な代数における空間，つまり $\underset{\sim}{\mathbf{K}}_2$-空間は位相空間にかなり近いものであると解釈できるだけであって，部分 $\Pi^1_1$-可測関数の代数における空間はもはや位相空間とは程遠い．このため，部分 $\Pi^1_1$-可測関数の代数における連結性もまたそんなに自明な概念ではない．

**定理 2.2** (一般化ライスの定理)**.** $M$ を全域でない部分結合子代数とする．このとき，$M$-空間 $M[\subseteq \mathbb{N} \to \mathbb{N}]$ は $M$-連結である．

*Proof.* まず，証明の方針を説明する．$M$-実現可能関数 $\varphi\colon M[\subseteq \mathbb{N} \to \mathbb{N}] \to \mathbf{2}$ が与えられているとする．このとき，関数 $f \in M[\subseteq \mathbb{N} \to \mathbb{N}]$ について，$\psi_f\colon M[\subseteq \mathbb{N} \to \mathbb{N}] \to M[\subseteq \mathbb{N} \to \mathbb{N}]$ を次によって定義する．

$$\psi_f(g) = \begin{cases} f & \text{if } g(0)\downarrow \\ \varnothing & \text{otherwise.} \end{cases}$$

ここで $\varnothing$ は空関数，つまり，どんな入力に対しても出力を返さない関数を意味する．このとき，合成関数 $\varphi\circ\psi_f\colon M[\subseteq\mathbb{N}\to\mathbb{N}]\to\mathbf{2}$ を考えると，

$$\varphi\circ\psi_f(g)=\begin{cases}\varphi(f) & \text{if } g(0)\downarrow\\ \varphi(\varnothing) & \text{otherwise.}\end{cases}$$

となる．この場合分けは停止問題のようであるから，決定不可能であろう．つまり，実現可能な方法では区別できないはずであるから，もし $\varphi\circ\psi_f$ が $M$-実現可能ならば，$\varphi(f)=\varphi(\varnothing)$ を得る．ここで $f$ は任意であったから，これは $\varphi$ が定数関数であることを導く．こうして，$M[\subseteq\mathbb{N}\to\mathbb{N}]$ が連結であることが示されるであろう．

以上が証明の方針である．厳密な証明を与えるために，まず関数 $\psi_f$ が部分結合子代数 $M$ の元によって実現できることを示そう．このために，次が成立することを確認する．

$$\mathtt{k}c(q\underline{0})=\begin{cases}c & \text{if } q\underline{0}\downarrow\\ \uparrow & \text{otherwise.}\end{cases}$$

なぜなら，まず $\mathtt{k}\in M$ の性質より，$ab\downarrow$ ならば $\mathtt{k}c(ab)=c$ となり，$ab\uparrow$ ならば $\mathtt{k}c(ab)\uparrow$ となる．よって，$q\underline{0}\downarrow$ となることと $kc(q\underline{0})=c$ となることは同値である．このとき，$u=\Lambda pqm.\mathtt{k}(pm)(q\underline{0})$ と定義しよう．いま $p\in M$ が $f$ の名として与えられているとすると，$up$ が $\psi_f$ の名であることを確認しよう．このためには，$g$ の任意の名 $q\in M$ について，$upq$ が $\psi_f(g)$ の名であることを証明すればよい．言い換えれば，$\{\!\{q\}\!\}\equiv g$ ならば $\{\!\{upq\}\!\}\equiv\psi_f(g)$ であることを示す．定義より，

$$upqm=\mathtt{k}(pm)(q\underline{0})\equiv\begin{cases}pm & \text{if } q\underline{0}\downarrow\\ \uparrow & \text{otherwise.}\end{cases}$$

であるから，もし $q\underline{0}\downarrow$ ならば，任意の $m$ について $upqm\equiv pm$ となるので，$upq$ と $p$ は外延的に同じ関数 $f$ を定義する，つまり $\{\!\{upq\}\!\}\equiv\{\!\{p\}\!\}\equiv f$ である．もし $(\exists\underline{n})\ q\underline{n}\downarrow$ が成り立たないならば，任意の $m$ について $upqm\uparrow$ となるので，$upq$ は空関数を定義している，つまり $\{\!\{upq\}\!\}\equiv\varnothing$ である．よって，$\{\!\{upq\}\!\}\equiv\psi_f(g)$ であることが示された．

また，$\varphi$ が $M$-実現可能ならば $\varphi\circ\psi_f$ もまた $M$-実現可能である．実際，$e$ が $\varphi$ の名ならば，$\Lambda q.e(upq)$ は $\varphi\circ\psi_f$ の名である．このとき，一般停止問題の決定不可能性 2.2 のアイデアを用いて，$\varphi\circ\psi_f$ が定数関数であることを示そう．$M$ の項 $s,t$ で $s\downarrow$ かつ $t\uparrow$ となるものを固定する．$\varphi$ は 2 値関数なので，$\varphi(f)$ が 0 または 1 の場合を考えればよい．もし $\varphi(f)=0$ ならば，項 $Q$ を

$$Q(r)=\Lambda n.\mathtt{if}\ e(upr)\ \mathtt{iszero}\ \mathtt{then}\ t\ \mathtt{else}\ s$$

によって定義する．クリーネの再帰定理より，$Q(r)\equiv r$ となる $r\in M$ が存在するが，このとき，

$$e(upr)=\underline{0}\iff(\forall\underline{n})\ r\underline{n}\equiv t\uparrow\iff r\underline{0}\uparrow\iff\varphi\circ\psi_f(\{\!\{r\}\!\})\equiv\varphi(\varnothing)$$

となることは明らかである．さらに，$\Lambda q.e(upq)$ は $\varphi\circ\psi_f$ の名であったことから，

$$\varphi(\varnothing)\equiv\varphi\circ\psi_f(\{\!\{r\}\!\})\equiv[\![\Lambda q.e(upq)]\!](\{\!\{r\}\!\})\equiv[\![e(upr)]\!]=0=\varphi(f)$$

となり，$\varphi(\varnothing) = \varphi(f)$ が導かれた．ここで $f$ は任意だったので，これは $\varphi$ が定数関数であることを導く．よって，どんな $M$-実現可能関数 $\varphi\colon [\subseteq \mathbb{N} \to \mathbb{N}] \to \mathbf{2}$ も定数関数であることが導かれた．以上より，部分関数の空間 $M[\subseteq \mathbb{N} \to \mathbb{N}]$ が $M$-連結であることが示された． □

**豆知識．** 上の証明を分析すると，部分関数の空間上の特化順序 (specialization order) において空関数が最小元となる，という性質が重要な役割を担っている．ここで，位相空間 $X$ の元の間の特化順序 $x \leqslant y$ は，もし $X$ の開集合 $U$ が $x$ を含むならば $y$ を含むこととして定義される．これは非 $T_1$ 位相を用いる分野では頻繁に利用される概念である．したがって，ライスの定理は「特化順序の下で最小元を持つ空間は連結である」という主張を位相空間以外の概念（たとえば計算可能性やボレル可測性）にも一般化したものと考えられる．

外延性の観点からは，これは $M[\subseteq \mathbb{N} \to \mathbb{N}]$ から $\mathbf{2}$ への（外延的でない）計算可能関数はたくさん存在するにも関わらず，外延的な計算可能関数は自明なものしか存在しない，と述べることもできる．

## § 3. 実数の計算論

計算理論の誕生以来，実数上の計算論は計算可能性理論の中心的テーマであり続けた．しかし，もちろん，あらゆる実数を有限文字列として取り扱うということは不可能である．それでは，研究者たちは如何にして実数上の計算論を取り扱ってきただろうか．1 つの解法は，クリーネの第一代数 $\mathbf{N}$ における計算可能実数論であり，もう 1 つの解法は，クリーネの相対第二代数 $\mathbf{K} := (\mathbb{K}_2, \mathbb{K}_{2^\circ})$ における実数上の計算可能関数論である．実数の計算論の初期は前者のアプローチが主流だったように思うが，時代を経るにつれ，徐々に後者の理論の方が優れていると分かってきた．

この正確な意味を説明する前に，まず，実数の計算論の具体的なアイデアを述べよう．まず，有理数上の計算論は，例**??**のような有理数の表現を用いて自明に展開できる．実数の計算可能性を導入する最も簡単だが最も優れた方法は，任意精度計算（精度保証計算）である．

**定義 3.1.** 実数 $x \in \mathbb{R}$ が**任意精度計算可能**とは，任意の正有理数 $\varepsilon$ 精度で $x$ を有理近似するアルゴリズムが存在することである．つまり，ある計算可能関数 $\Phi : \mathbb{Q}_{>0} \to \mathbb{Q}$ が存在して，次を満たすことである．

$$(\forall \varepsilon \in \mathbb{Q}_{>0}) \quad |x - \Phi(\varepsilon)| < \varepsilon.$$

とりあえず，これが実数の計算可能性の妥当な定義のように思うが，少しだけ歴史的経緯を振り返ろう．チューリングが 1936 年にチューリング機械を導入した記念碑的論文「計算可能数とその決定問題への応用」では，実数の計算可能性は 2 進小数展開の計算可能性として導入されている．ここでは，その計算可能性を 2 進計算可能性と呼ぶことにする．

**定義 3.2.** 実数 $x \in \mathbb{R}$ が 2 **進計算可能** (*binary computable*) とは，$x$ の任意の 2 進小数展開

$$a_0 a_1 \dots a_n . a_{n+1} a_{n+2} \dots a_{n+k} a_{n+k+1} \dots$$

に対して，関数 $i \mapsto a_i$ が計算可能であることを意味する．

しかし，すぐにチューリングは，実数の計算論を 2 進展開で導入するのは誤りだと気づいたようである．翌年にチューリングは「計算可能数とその決定問題への応用 – 訂正」を出版し，その後半部では，自身の実数の 2 進計算論を撤回した．実数の計算論の正しい与え方として，チューリングは，たとえばブラウワーの直観主義数学における縮小区間による実数の表現を挙げている．これは区間の縮小列によって実数を表現するものである．

**定義 3.3.** 実数 $x \in \mathbb{R}$ が**区間計算可能**とは，有理数の対の計算可能な列 $(p_n, q_n)_{n\in\mathbb{N}}$ で次のようなものが存在することである．

$$(\forall n \in \mathbb{N})\ [p_n < p_{n+1} < q_{n+1} < q_n],\ \text{and}\ x = \lim_{n\to\infty} p_n = \lim_{n\to\infty} q_n.$$

任意精度計算との関連性に触れておくと，区間計算において，ストリーム $(p_n, q_n)_{n\in\mathbb{N}}$ の各段階では実数を両側から挟み込んでいるため，常に近似精度の保証が可能となっている．実際，以下のように，チャーチ・チューリングの提唱の実数の計算論版のようなものも成立する．

**命題 3.4.** 実数について，任意精度計算可能性，2 進計算可能性，区間計算可能性はいずれも同値である．

*Proof.* 任意精度計算可能性と区間計算可能性の同値性は読者の演習問題とする．任意の 2 進計算可能実数が任意精度計算可能であることを確認しよう．実数の与えられた無限 2 進小数表記 $\alpha = a_0a_1 \dots a_k.a_{k+1}a_{k+2} \dots$ に対して，$q_n$ を有限小数 $a_0a_1 \dots a_k.a_{k+1}a_{k+2} \dots a_{k+n}a_{k+n+1}$ が表す有理数とする．このとき，$q = (q_n)_{n\in\omega}$ が $[\alpha]_{\text{bin}}$ の任意精度近似，つまり $[q]_A = [\alpha]_{\text{bin}}$ であることは容易に分かる．

最後に，任意精度計算可能実数が 2 進計算可能実数であることを示す．有理数がどちらの意味でも計算可能であることは明らかなので，無理数 $x$ について任意精度計算可能性が 2 進計算可能性を導くことを示せばよい．また，$x \in [0, 1]$ であると仮定しても一般性を失わない．$x$ を任意精度計算可能な無理数とし，$\Phi$ をその任意精度近似とする．$x$ は無理数であるから，2 進小数展開した際に，必ず 0 と 1 の両方を無限に含む．有理数 $\Phi(s)$ を 2 進小数展開した結果を $\alpha_s = 0.a_0^s a_1^s \dots a_{\ell(s)}^s$ と書く．任意の $n \in \mathbb{N}$ について，$\alpha_s$ の小数点以下 $n+1$ 桁目以降 $s$ 桁目以前に 01 または 10 が出現するような $s$ を探す．このとき，$\Phi(s)$ から最大 $2^{-s}$ の誤差が発生したとしても，この 01 または 10 の出現以前の部分は変動しない．よって，$a_n = a_n^s$ と定義すれば，$[0.a_1a_2 \dots]_{\text{bin}} = \lim_s \Phi(s) = x$ を得る． □

ところで，チューリングの最初の定義である「2 進計算可能性」とチューリングの第 2 の定義である「区間計算可能性」が同値であるならば，チューリングは訂正論文を出す必要は無かったのでは，と思うかもしれない．しかし，実は，チューリングが訂正論文を出した判断は正しかった．実関数の計算可能性を考える段階になると「2 進計算可能性」と「区間計算可能性」は大きく異なる．そして，そのとき「2 進計算可能性」は破綻する．なんと，実数を 3 倍するだけのごく単純な関数

$x \mapsto 3x$ ですら，2 進計算可能ではないのである．これについては，後の定理 3.7 で確認する．

このような問題を説明する前に，実数の表現について考えよう．そもそも実数の定義とは何であっただろうか．実数の集合 $\mathbb{R}$ とは，有理数の集合 $\mathbb{Q}$ の完備化，すなわちコーシー列の同値類であると学んだかもしれない．あるいは，実数とは，デデキント切断であると学んだかもしれない．

**■コーシー実数:** 有理**コーシー列** (*Cauchy sequence*) とは有理数列 $(q_n)_{n\in\omega} \in \mathbb{Q}^{\mathbb{N}}$ で，

$$(\forall n > 0)(\exists k \in \mathbb{N})(\forall i, j \geqslant k)\ |q_i - q_j| < 2^{-n}$$

を満たすものである．有理コーシー列 $(p_n)_{n\in\omega}$ と $(q_n)_{n\in\omega}$ が等しいとは，任意の $\varepsilon > 0$ について，十分大きな任意の $i, j \in \mathbb{N}$ について $|p_i - q_j| < \varepsilon$ が成立することである．実数の集合 $\mathbb{R}$ の**素朴コーシー表現** (*naive Cauchy representation*) とは，有理コーシー列の同値類として実数を表現する関数 $[\cdot]_{\mathrm{nC}} :\subseteq \mathbb{Q}^{\mathbb{N}} \to \mathbb{R}$ である．より正確には，任意の有理コーシー列 $(q_n)_{n\in\omega}$ について，

$$[(q_n)_{n\in\omega}]_{\mathrm{nC}} = \{(p_n)_{n\in\omega} : (p_n) \text{ は有理コーシー列であり，} (p_n) \text{ と } (q_n) \text{ は等しい.}\}$$

同値類 $[(q_n)_{n\in\omega}]_{\mathrm{nC}}$ のことを $\lim_{n\to\infty} q_n$ と表す．素朴コーシー表現は非常に使い勝手が悪い．本節で言及する実数の表現のうちでは最悪な表現である．コーシー列の収束速度が分からないので，コーシー列というストリームを読み込む過程で，極限の値にどれくらい近づいたか判断できないからである．素朴コーシー表現は極限概念を伴う表現であり，第**??**節で詳述するが，極限は容易に計算不可能性を生み出すのである．

そういうわけで，コーシー列と言った場合には，収束速度の情報が常に備わっていて欲しい．関数 $k : \mathbb{N} \to \mathbb{N}$ が列 $(q_n)_{n\in\omega}$ の**収束係数**とは，

$$(\forall n > 0)(\forall i, j \geqslant k(n))\ |q_i - q_j| < 2^{-n}$$

を満たすことを意味する．実数の集合 $\mathbb{R}$ の**係数付きコーシー表現**とは，有理コーシー列の正しい収束係数が与えられたとき，その有理コーシー列の同値類として実数を表現する関数 $[\cdot]_{\mathrm{mC}} :\subseteq \mathbb{Q}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{R}$ である．つまり，任意の有理コーシー列 $(q_n)_{n\in\omega}$ について，もし $k$ が $(q_n)_{n\in\omega}$ の収束係数であるときのみ，$[(q_n), k]_{\mathrm{mC}}$ を定義し，$[(q_n), k]_{\mathrm{mC}} = \lim_{n\to\infty} q_n$ とする．

係数付きコーシー表現は非常に良い表現なのだが，定義が若干ごちゃごちゃするという難点がある．このため，係数付きコーシー表現の代わりに，id が収束係数となるようなコーシー列（急収束コーシー列）だけを考えて単純化することが多い．つまり，**急収束コーシー列** (*rapidly converging Cauchy sequence*) とは有理数列 $(q_n)_{n\in\omega} \in \mathbb{Q}^{\mathbb{N}}$ で，

$$(\forall n > 0)(\forall i, j \geqslant n)\ |q_i - q_j| < 2^{-n}$$

を満たすものである．このとき，実数の集合 $\mathbb{R}$ の**コーシー表現** (*Cauchy representation*) とは，急収束コーシー列の同値類として実数を表現する関数 $[\cdot]_{\mathrm{C}} :\subseteq \mathbb{Q}^{\mathbb{N}} \to \mathbb{R}$ である．つまり，$(q_n)_{n\in\omega}$ が急収束コーシー列であるときのみ，$[(q_n)]_{\mathrm{C}}$ を定義し，$[(q_n)]_{\mathrm{C}} = \lim_{n\to\infty} q_n$ とする．

**■デデキント実数:**

有理数の集合 $A \subseteq \mathbb{Q}$ が下方閉とは，任意の $a \in A$ と有理数 $q \leqslant a$ について $q \in A$ となることである．同様に，$A$ が上方閉とは，任意の $a \in A$ と有理数 $q \geqslant a$ について $q \in A$ となることで

ある．**デデキント切断** (*Dedekind cut*) とは，有理数の空でない集合の対 $L, R \subseteq \mathbb{Q}$ で，$L$ は上方閉，$R$ は下方閉，$L \cap R = \varnothing$ かつ $|\mathbb{Q}\backslash(L \cup R)| \leqslant 1$ となることである．この $L$ の開部分として，$L^\circ = \{r \in \mathbb{Q} : (\exists s \in L)\ r < s\}$ と定義する．デデキント切断 $(L_0, R_0)$ と $(L_1, R_1)$ が等しいとは，$L_0^\circ = L_1^\circ$ となることである．

実数の集合 $\mathbb{R}$ の**デデキント表現** (*Dedekind representation*) とは，デデキント切断の枚挙の同値類として実数を表現する関数 $[\cdot]_D :\subseteq (\mathbb{Q} \times \mathbb{Q})^{\mathbb{N}} \to \mathbb{R}$ である．より正確にこの表現を定義するため，$p = (p_n)_{n\in\mathbb{N}}$ と $q = (q_n)_{n\in\mathbb{N}}$ について，$p$ と $q$ がそれぞれ集合 $A$ と $B$ の枚挙であるとき，$(p, q)$ を $(A, B)$ の枚挙と呼び，$\mathrm{Rng}(p, q) = (A, B)$ と書く．このとき，デデキント切断の枚挙 $(p, q)$ に対して，同値類 $[(p, q)]_D$ を以下によって定義する．

$$[(p, q)]_D = \{(r, s) : \mathrm{Rng}(r, s) \text{ は } \mathrm{Rng}(p, q) \text{ と等しいデデキント切断である.}\}$$

$(p, q)$ がデデキント切断 $(L, R)$ の枚挙であるとき，$[(p, q)]_D$ のことを $\sup L$ または $\inf R$ と書く．デテキント切断を用いた表現として，他にも開デデキント表現や決定可能デデキント表現などが知られているが，いずれも計算論を展開する際にはあまり有用でないので，ここでは触れない．

**豆知識.** ところで，デデキント表現の等しさの定義では，$L$ についての情報しか用いていない．このため，切断の定義は単に $L$ だけ考えればよいのではないか，と思う人もいるかもしれない．そのような表現は**左デデキント表現**などと呼ばれるが，ユークリッド直線 $\mathbb{R}$ の表現としては正しくない．左右からしっかり挟み込まないと，実数の近似精度が評価できないため，位相的に全く異なった概念になってしまう．しかし，$\mathbb{R}$ 上のユークリッド位相の表現として正しくないだけで，$\mathbb{R}$ 上の**上半位相** (*upper topology*) と呼ばれる非 $T_1$ 位相の表現としては正しい．これは $\{0, 1\}$ を離散的なオブジェクトとして見るときはシエルピンスキ表現は誤りだが，$\{0, 1\}$ を連結空間として見るときはシエルピンスキ表現が正しい，という状況と同様である．

コーシー表現では，実数は有理数のストリームとして表される．デデキント表現では，実数は有理数の対のストリームとして表される．したがって，実数の計算論を展開するための適切な舞台は，ストリーム上の計算論，つまりクリーネの相対第二代数 $\mathbf{K}$ 上の計算論 $\mathsf{Mod}(\mathbf{K})$ である．

このアイデアを用いて，任意精度表現，2 進小数表現，縮小区間表現を，実数のストリーム表現として再定義しよう．

- 2 進小数表現 $[\cdot]_{\mathrm{bin}}$ は，$\{0, 1, .\}$ の記号のストリーム $\alpha$ で小数点記号 . が高々 1 つしか含まないものについて，$[\alpha]_{\mathrm{bin}}$ を実数の 2 進表記であると理解する表現である．
- 縮小区間表現 $[\cdot]_I$ は，$\lim_{n\to\infty}(q_n - p_n) = 0$ となるような有理数の対のストリーム $(p, q) = (p_n, q_n)_{n\in\mathbb{N}}$ が与えられたとき，$[(p, q)]_I = \lim_{n\to\infty} p_n$ を与える関数である．
- 任意精度表現 $[\cdot]_A$ は，与えられた有理数のストリーム $(\Phi(n))_{n\in\mathbb{N}}$ について，$|x - \Phi(n)| < 2^{-n}$ なるような実数 $x$ が存在するとき，そのような $x$ を返す関数である．

ここで，「実数の計算可能性」を定義することと「実数の表現」を与えることの理論的な違い，そして前者よりも後者の方が重要である理屈を説明しよう．まず，「実数の計算可能性」という概念を定義しても，「実関数の計算可能性」はまた個別に定義する必要がある．しかし，「実数の表現」を与えると，そこから「実数の計算可能性」と「実関数の計算可能性」が自動的に定義される．これについて説明しよう．実数の表現を与えた，という事実は，いま $\mathbb{R}$ に $\mathbf{K}$-表現空間として

の構造が与えられた，ということである．表現空間上の関数の計算可能性は定義??で与えた通りである．表現空間の点の計算可能性は以下によって与えられる．

**定義 3.5.** $(\mathbf{A}, \mathbf{A}_\circ)$ を相対部分組合せ代数とする．$\mathbf{A}$-表現空間 $X$ の点 $x \in X$ が**計算可能** (*computable*) であるとは，$x = [a]_X$ となる $a \in \mathbf{A}_\circ$ が存在することを意味する．

**例 3.6.** クリーネの相対第二代数 $\mathbf{K}$ において，$\mathbf{K}$ はストリーム全体の集合 $\mathbb{N}^\mathbb{N}$ であり，$\mathbf{K}_\circ$ は計算可能ストリーム全体の集合，つまり $\mathbb{N}$ 上の計算可能関数全体の集合であった．このとき，$\mathbf{K}$-表現空間 $X$ の点 $x \in X$ が計算可能であるとは，$x$ が計算可能な名を持つことを意味する．

また，実数や実関数の計算可能性だけでなく，たとえば「$\mathbb{R}$ の開部分集合の計算可能性」「$\mathbb{R}$ のコンパクト部分集合の計算可能性」などを含む無数の計算可能性概念も自動的に定義される．このように「表現」を与えるだけで，豊穣な計算可能性理論の世界が自動的に広がっていくのである．

実数の計算論は実数の表現に依存する．したがって，実数のどの表現が同値であり，どの表現が異なる計算論を与えるか，ということは計算可能解析学の初期における問題の 1 つであった．つまり，与えられた実数の計算論が異なる実数の計算論に翻訳可能かどうか，というナンバリングの理論と同様のシチュエーションに辿り着く．ナンバリングの理論のときと同様に，この問題は定義??で用いた表現の還元可能性の概念を用いて定式化できる．

**定理 3.7.** 任意精度表現，区間表現，コーシー表現，デデキント表現は同値である．一方，2 進小数表現および素朴コーシー表現は異なる表現であり，以下が成立する．

$$[\cdot]_{\mathrm{bin}} < [\cdot]_A \equiv [\cdot]_I \equiv [\cdot]_C \equiv [\cdot]_D < [\cdot]_{nC}.$$

*Proof.* $[\cdot]_{\mathrm{bin}} \leqslant [\cdot]_A$ であることは，命題 3.4 の証明において，$[\alpha]_{\mathrm{bin}} = [q]_A$ となることを示したが，$\alpha \mapsto q$ が計算可能であるから，これが $[\cdot]_{\mathrm{bin}} \leqslant [\cdot]_A$ を保証する．$[\cdot]_C \leqslant [\cdot]_{nC}$ は自明である．$[\cdot]_A \equiv [\cdot]_C$ は明らかであろう．$[\cdot]_D \leqslant [\cdot]_I$ について，与えられたデデキント切断の枚挙 $(\ell_n, r_n)_{n\in\mathbb{N}}$ に対して，各 $n$ について $p_n = \max_{s<n} \ell_s$ かつ $q_n = \max_{s<n} r_s$ とすれば，$(p_n, q_n)_{n\in\omega}$ は区間の縮小列であり，自明に $\sup_n \ell_n = \lim_n p_n$ かつ $\inf_n r_n = \lim_n q_n$ であるから，同じ実数を与える．よって，計算可能関数 $(\ell_n, r_n)_{n\in\mathbb{N}} \mapsto (p_n, q_n)_{n\in\omega}$ によって $[\cdot]_D \leqslant [\cdot]_I$ は保証される．$[\cdot]_D \leqslant [\cdot]_I$ も容易に示せる．$[\cdot]_A \leqslant [\cdot]_I$ について，実数の任意精度近似 $\Phi$ に対して，区間 $(\Phi(n) - 2^{-n+1}, \Phi(n) + 2^{-n+1})$ を考えればよい．$[\cdot]_I \leqslant [\cdot]_A$ について，区間縮小列 $(p_n, q_n)_{n\in\mathbb{N}}$ が与えられたとき，各 $n$ に対して $q_s - p_s < 2^{-n}$ なる $s \in \mathbb{N}$ を探し，$\Phi(n)$ をその区間 $(p_s, q_s)$ の中点，つまり $\Phi(n) = (q_s - p_s)/2$ と定義すればよい．

次に $[\cdot]_{nC} \not\leqslant [\cdot]_A$ を示そう．$H_s$ を停止問題の時刻 $s$ 近似，つまり $H_s = \{e < s : \{\!\{e\}\!\}(e)[s] \downarrow\}$ とする．このとき $q_s = \sum_{e\in H_s} 2^{-e}$ と定義すると，$(q_s)_{s\in\mathbb{N}}$ は計算可能なコーシー列である．特に，その極限 $x = \lim_{s\to\infty} q_s$ は，素朴コーシー表現において計算可能である．しかし，$x$ の 2 進表記を

見ると，明らかに停止問題 Halt の情報を記している．停止問題の計算不可能性より，これは $x$ が $(\mathbb{R}, [\cdot]_{\mathrm{bin}})$ で計算不可能であることを意味する．一方，命題 3.4 より，$x$ は任意精度表現でも計算不可能である．よって，$[\cdot]_{nC} \not\equiv [\cdot]_A$ を得る．既に $[\cdot]_A \leqslant [\cdot]_{nC}$ は示してあるから，$[\cdot]_{nC} \not\leqslant [\cdot]_A$ である．

最後に，$[\cdot]_A \not\leqslant [\cdot]_{\mathrm{bin}}$ を示す．$\mathbb{R}_{\mathrm{bin}}$ を表現空間 $(\mathbb{R}, [\cdot]_{\mathrm{bin}})$ とすると，$f(x) = 3x$ で定義された関数 $f : \mathbb{R}_{\mathrm{bin}} \to \mathbb{R}_{\mathrm{bin}}$ が計算不可能であることを示す．他の表現の場合は，$x \mapsto 3x$ は容易に計算可能であることが分かるので，これが 2 進小数表現と他の表現の違いを導く．もし $f(x) = 3x$ が 2 進小数表現の元で計算可能だったとする．$1/3$ を 2 進小数展開すると $0.01010101\ldots$ という循環小数になることに注意する．$f$ を実現するチューリング機械 $M$ が，入力ストリーム $0.01010101\ldots$ を読み込んでいるとしよう．このとき，$M$ は必ずある時点でストリームを出力し始めなければならない．入力ストリームの $n$ 桁目を読み込んだ段階で，$M$ がある文字 $k$ を出力したとしよう．$M$ は $x \mapsto 3x$ を計算しているはずなので，$k = 1$ であるか $k = 0$ であり，その後に小数点とそれ以下の値が続くはずである．もし $k = 1$ だったとしたら，入力ストリームの $n+1$ 桁以降が何であろうと，$M$ は 1 以上の実数を記述する．しかし，$n+1$ 桁目以降ずっと 0 を返すストリーム $\alpha$ を考えると，$[\alpha]_{\mathrm{bin}} < 1/3$ であるから，$f([\alpha]_{\mathrm{bin}}) < 1 \leqslant [\{\!\{M\}\!\}(\alpha)]_{\mathrm{bin}}$ となり，$M$ は $f$ を正しく計算できていない．もし $k = 0$ だったとしたら，入力ストリームの $n+1$ 桁以降が何であろうと，$M$ は 1 以下の実数を記述する．しかし，$n+1$ 桁目以降ずっと 1 を返すストリーム $\alpha$ を考えると，$[\alpha]_{\mathrm{bin}} > 1/3$ であるから，$f([\alpha]_{\mathrm{bin}}) > 1 \geqslant [\{\!\{M\}\!\}(\alpha)]_{\mathrm{bin}}$ となり，$M$ は $f$ を正しく計算できていない．よって，$x \mapsto 3x$ が 2 進小数表現の下では計算不可能であることが示された．以上より，$[\cdot]_{\mathrm{bin}}$ が他の表現と同値でないことが示されるが，既に $\leqslant [\cdot]_{\mathrm{bin}} \leqslant [\cdot]_A$ は示してあるから，$[\cdot]_A \not\leqslant [\cdot]_{\mathrm{bin}}$ を得る． □

この定理の興味深いところは，悪い表現，というものにも複数の方向性があり，表現の還元可能性概念がそれを明示してくれる点にある．つまり 2 進小数表現は，要求が厳しすぎる表現であり，素朴コーシー表現は，要求が甘すぎる表現であると言える．

**豆知識．**本節で述べた実数の計算モデルは，**厳格実数計算** (*exact real computation*), **誤差なし実数計算** (*error-free real computation*), あるいは**任意精度計算**などとも呼ばれる．最近は，幾つかのプログラミング言語で厳格実数計算用のライブラリが少しずつ充実しつつあるようだ．

# 第 4 章

# 計算可能解析学

## §1. 量化記号とゲーム

第??節では，いくつかの論理式の実現可能性について議論した．ここまでに，実数の表現を導入してきたので，これによって実数論や実解析学に関する数学的定理の実現可能性について議論することができる．しかし，実数論や実解析学の定理の実現を議論するにあって少し困難な点がある．実数論や実解析学の定理となると，厳密に論理式で記述しようとすると，量化の入れ子が多く，分析がかなり面倒くさい．たとえば，実数論を代表するものとして，「$\varepsilon$-$\delta$ 論法」がある．「$\varepsilon$-$\delta$ 論法」が具体的に使われる基本的な例として，たとえば連続性の定義であろう．関数 $f : \mathbb{R} \to \mathbb{R}$ が連続である，という文は次によって表される．

$$(\forall x_0)(\forall \varepsilon > 0)(\exists \delta > 0)(\forall x)\ [|x - x_0| < \delta \ \to \ |f(x) - f(x_0)| < \varepsilon].$$

なんと，連続である，ということを書くだけで，$\forall$ と $\exists$ が合計 4 つも出現する．単に関数が連続である，と述べるだけでこれなのだから，連続関数の性質を議論する際は更に多くの量化記号が出現するであろう．すると，実現可能性の定義に従って，実解析の定理が実現可能かどうか分析するのは骨が折れそうである．

**■ゲーム意味論:** 量化記号の分析のために便利な道具は，**ゲーム** $(game)$ である．たとえば，簡単な例として，1 手詰みの詰将棋を考えよう．「1 手詰み」とは，上手く駒を打てば，こちらの勝利が確定する，ということである．もう少し細かく言えば，「勝利が確定する」というのは，相手が次の駒をどう打とうとも，こちらが上手く駒を打てば相手の王将または玉将を奪える，ということである．これを抽象的に記述すれば，

$$\exists x \forall y \exists z\ A(x, y, z)$$

と書ける．ここで $x, y, z, A$ は，次を意味する．

- $x$ は我々がどの駒を盤面のどの位置に打つかの指定である．
- $y$ はそれに対応して，相手がどの駒をどの位置に打つかの指定である．
- $z$ はそれに応じて，我々がどの駒を盤面のどの位置に打つかの指定である．
- $A$ は各手が将棋のルールに従っており，王将または玉将が取られた状態を表す文である．

このように，1 手詰めの詰将棋は 3 つの量化記号を持つ文として記述できる．普通の詰将棋は最低でも 3 手詰めであり，通常はそれ以上であろう．その場合の，量化記号 $\forall$ と $\exists$ の数はとんでもない量となる．詰将棋を嗜む人は，$\varepsilon$-$\delta$ 論法よりも果てしなく難しい問題を普段から取り扱っているということである．

逆に，量化記号を使って書かれた文は，$\forall$ さんと $\exists$ くんの 2 人のプレイヤーによる完全情報ゲームとして解釈されることが，論理学や計算機科学などでは古くから標準的である．たとえば，量化記号が複雑に入り混じった論理式として，

$$\forall a\,\exists b\,\forall c\,\forall d\,\exists e\ B(a,b,c,d,e)$$

を考えよう．これは次のようなゲームとして図示できる．

| | 1 手目 | 2 手目 | 3 手目 | 4 手目 | 勝利条件 |
|---|---|---|---|---|---|
| $\forall$ | $a$ | | $c,d$ | | $B(a,b,c,d,e)$ は偽 |
| $\exists$ | | $b$ | | $e$ | $B(a,b,c,d,e)$ は真 |

このゲームは，次の意味で，上に記述した論理式を特徴づける．

$$\forall a\,\exists b\,\forall c\,\forall d\,\exists e\ B(a,b,c,d,e) \text{ は真である} \iff \text{後手の } \exists \text{ くんが必勝戦略を持つ.}$$

このような $\forall$ さんと $\exists$ くんのゲームは，**ヒンティッカ・ゲーム** (*Hintikka game*) と呼ばれたりする．複雑に入れ子になった量化記号をゲームとして理解しようという発想は歴史が長く，1950 年頃に遡る．といっても，ヒンティッカによるゲーム意味論は，非線形な量化（分岐量化子）といった極めて複雑な量化子の分析のために誕生したものであり，本稿で語るような線形かつ有限な量化子をゲームと考える試みであれば，1950 年よりも遥かに古いかもしれない．

## 1.1. 数列の極限

**■極限ゲーム:**

数列 $(x_n)$ が $x$ へ収束する，という文は以下のように定義されていた．

$$(\forall \varepsilon > 0)(\exists k)(\forall n \geqslant k)\ |x - x_n| < \varepsilon.$$

数列 $(x_n)$ の $x$ への収束性に対応するヒンティッカ・ゲームを図示しよう．

| | 1 手目 | 2 手目 | 3 手目 | 勝利条件 |
|---|---|---|---|---|
| $A$ | $\varepsilon > 0$ | | $n \geqslant k$ | $\lvert x - x_n\rvert \geqslant \varepsilon$ |
| $B$ | | $k$ | | $\lvert x - x_n\rvert < \varepsilon$ |

上の図が意味していることは，先手の $A$ さんが初手で実数 $\varepsilon > 0$ を打つ．続いて，後手の $B$ さんは自然数 $k$ を選び，最後に $A$ さんが $n \geqslant k$ を選ぶ．もし $|x - x_n| < \varepsilon$ ならば $B$ さんの勝利，さもなくば $A$ さんの勝利，ということである．すると，次が成立する．

$$(x_n) \text{ が } x \text{ に収束する} \iff \text{後手の } B \text{ さんが必勝戦略を持つ.}$$

そうすると，数列 $(x_n)$ が $x$ へ収束することを証明するために何をすればいいだろうか．答えは次である．

$$(x_n) \text{ の } x \text{ への収束性を証明する} \iff \text{後手の } B \text{ さんの必勝戦略を作る．}$$

実際には，我々は収束先を具体的に求めさせられることも多い．この場合，$B$ さんは先手となり，収束先 $x$ を初手で答える必要がある．これをヒンティッカ・ゲームとして表すと次のようになる．

| | 1 手目 | 2 手目 | 3 手目 | 4 手目 | 勝利条件 |
|---|---|---|---|---|---|
| $A$ | | $\varepsilon > 0$ | | $n \geqslant k$ | $\lvert x - x_n \rvert \geqslant \varepsilon$ |
| $B$ | $x$ | | $k$ | | $\lvert x - x_n \rvert < \varepsilon$ |

**■収束性証明の例:** それでは，$(\frac{1}{n})$ が 0 に収束することを証明しよう．このヒンティッカ・ゲームは，次のように表される．

| | 1 手目 | 2 手目 | 3 手目 | 勝利条件 |
|---|---|---|---|---|
| $A$ | $\varepsilon > 0$ | | $n \geqslant k$ | $\lvert 0 - \frac{1}{n} \rvert \geqslant \varepsilon$ |
| $B$ | | $k$ | | $\lvert 0 - \frac{1}{n} \rvert < \varepsilon$ |

我々は後手 $B$ さんの戦略を記述すればよい．まず，先手の $A$ さんが初手に $\varepsilon > 0$ を打ってきたとする．我々は「$B$ さんが次にどんな手を打てば勝てるか」を考えなければならない．ここは一般には，たとえば将棋やチェスの対局のときのように，頭を捻る必要がある．今回の場合は，好きな自然数 $k > \frac{1}{\varepsilon}$ を取ってくればよい．すると，$A$ の次の手 $n \geqslant k$ が何であれ，以下のように $B$ の勝利が確定している．

$$\dfrac{\dfrac{\boxed{B \text{ の第 2 手}} \quad k > \frac{1}{\varepsilon} \qquad \boxed{A \text{ の第 3 手}} \quad n \geqslant k}{\frac{1}{n} < \varepsilon}}{\boxed{B \text{ の勝利}} \quad \lvert 0 - \frac{1}{n} \rvert < \varepsilon}$$

**■コーシー列:**

数列の極限の存在というだけでなく収束先まで求めるためのゲームは，4 手詰めである．一方，数列の極限の存在を示したいだけで極限値を求める必要がない場合には，数列がコーシー列であることを示すだけで十分である．数列 $(x_n)$ がコーシー列である，という文は以下のように定義されていた．

$$(\forall \varepsilon > 0)(\exists k)(\forall n, m \geqslant k)\ \lvert x_n - x_m \rvert < \varepsilon.$$

この場合のゲームは，以下のように 3 手詰めである．

| | 1 手目 | 2 手目 | 3 手目 | 勝利条件 |
|---|---|---|---|---|
| $A$ | $\varepsilon > 0$ | | $n, m \geqslant k$ | $\lvert x_n - x_m \rvert \geqslant \varepsilon$ |
| $B$ | | $k$ | | $\lvert x_n - x_m \rvert < \varepsilon$ |

ここで，3 手目において，$A$ さんは 2 つの数の対 $(n, m)$ を打つこととなる．コーシー列と収束列の同値性から，上記のゲームについて，次が成立する．

$$(x_n) \text{ は収束列である} \iff \text{後手の } B \text{ さんが必勝戦略を持つ．}$$

## 1.2. 関数の連続性

**■連続性ゲーム:**

関数 $f$ が点 $x_0$ で連続である，という文は次によって表される．

$$(\forall\varepsilon>0)(\exists\delta>0)(\forall x)\ [|x-x_0|<\delta\ \rightarrow\ |f(x)-f(x_0)|<\varepsilon].$$

これに対応するヒンティッカ・ゲームは以下のように図示できる．

| | 1 手目 | 2 手目 | 3 手目 | ルール | 勝利条件 |
|---|---|---|---|---|---|
| $A$ | $\varepsilon>0$ | | $x$ | $\lvert x-x_0\rvert<\delta$ | $\lvert f(x)-f(x_0)\rvert\geqslant\varepsilon$ |
| $B$ | | $\delta>0$ | | | $\lvert f(x)-f(x_0)\rvert<\varepsilon$ |

これまでと同様に，以下が成立する．

$$f \text{ は } x_0 \text{ で連続である} \iff \text{後手の } B \text{ さんが必勝戦略を持つ.}$$

単純に $f$ が連続であることを証明したい場合は，全ての $x_0$ に対して，上のゲームで $B$ さんが必勝戦略を持つことを示せば良い．ここで注意点としては，$x_0$ は $A$ さんが選ぶという点である．

| | 1 手目 | 2 手目 | 3 手目 | ルール | 勝利条件 |
|---|---|---|---|---|---|
| $A$ | $x_0,\varepsilon$ | | $x$ | $\lvert x-x_0\rvert<\delta$ | $\lvert f(x)-f(x_0)\rvert\geqslant\varepsilon$ |
| $B$ | | $\delta$ | | | $\lvert f(x)-f(x_0)\rvert<\varepsilon$ |

**■連続性の証明例:**

それでは $f(x)=|x|$ という関数が連続であることを証明しよう．我々は $B$ さんが $A$ さんに勝つ方法を教えてやればよい．1 手目に $A$ さんは $x_0,\varepsilon$ を選ぶ．第 2 手で $B$ さんはどの手を出すべきだと教えてやればいいだろうか．今回の場合は単純で，「$\delta=\varepsilon$ で大丈夫だよ」と教えてあげよう．第 3 手で $A$ はルール $|x-x_0|<\delta=\varepsilon$ を遵守するような $x$ を選んでくるはずである．いま，ゲームは以下のように進行している．

| | 1 手目 | 2 手目 | 3 手目 | ルール | 勝利条件 |
|---|---|---|---|---|---|
| $A$ | $x_0,\varepsilon$ | | $x$ | $\lvert x-x_0\rvert<\varepsilon$ | $\lvert\lvert x\rvert-\lvert x_0\rvert\rvert\geqslant\varepsilon$ |
| $B$ | | $\varepsilon$ | | | $\lvert\lvert x\rvert-\lvert x_0\rvert\rvert<\varepsilon$ |

$A$ がルール $|x-x_0|<\varepsilon$ を遵守している限り，$B$ の勝利条件 $||x|-|x_0||<\varepsilon$ を満たすことは容易に確かめられるであろう．よって，$B$ が勝利するから，$f(x)=|x|$ は連続である．

**■連続性の計算論的意味:**

この連続性ゲームの意味を計算理論の目線から再考察しよう．まず，たとえば次の状況設定を考えよう．

- コンピュータでは実数を直接的には扱えないので，適当に近似する必要がある．
- 有理数入力 $x\in\mathbb{Q}$ に対して，値 $f(x)$ を任意に近似する方法を知っている．
- $A$ さんは，関数 $f(x_0)$ の値を求めたい（ここで $x_0$ は有理数とは限らない）．

そういうわけで，$A$ さんは $f(x_0)$ の値を求めたいが，正確な数値というよりは，$f(x_0)\pm\varepsilon$ の範囲での値を求めれば十分であるとしよう．すると，連続性ゲームにおける $A$ さんと $B$ さんの役割を以下のように要約される．

$A$ さん: $f(x_0)$ の値を任意の精度で近似したい．
$B$ さん: どうやれば $f(x_0)$ を近似できるかを $A$ さんに伝える．

$B$ さんが必勝戦略を持つ，つまり $A$ さんがどんな精度を要求しても，$\varepsilon$-近似方法を答えられるならば，関数 $f$ は点 $x_0$ で連続であるということである．この観点では，ゲームの流れは以下のように説明できる．

$A$ さんの初手 $\varepsilon$ は，「わたしは $f(x_0)$ の $\varepsilon$ 精度での近似値が知りたい」という宣言である．$A$ さんは有理数値 $x$ に対しては $f(x)$ を求める方法を知っているが，$x_0$ は無理数かもしれないので，$f(x_0)$ を求める方法は分からない．しかし，「$x\in\mathbb{Q}$ が十分 $x_0$ に近ければ，$f(x)$ は $f(x_0)$ に十分近い」ことを期待できる．それでは，$x$ が $x_0$ にどれくらい近ければ，$f(x)$ が $f(x_0)\pm\varepsilon$ 範囲内に入ってくれるだろうか．$A$ さんには分からないので，$B$ さんに助言を仰ぐことにする．すると，$B$ さんが第 2 手で $\delta$ を打ってくるであろう．これは $B$ さんが「$x_0\pm\delta$ の範囲内の有理数 $x$ を入力してみなよ」と $A$ に助言しているということを意味する．助言に従って，第 3 手で $A$ さんは $x_0\pm\delta$ の範囲内の有理数 $x\in\mathbb{Q}$ を打つ．これはもちろんルール $|x-x_0|<\delta$ を遵守しているから問題ない．もし $B$ が必勝戦略を持っていて，それに従って正しく $\delta$ を選んでいれば，$B$ は必勝なのであるから，$A$ が 1 手目の $\varepsilon$ と 3 手目の $x$ として何を選んでいたとしても，$B$ が勝利しているはずである．つまり，$B$ の勝利条件 $|f(x_0)-f(x)|<\varepsilon$ が満たされている．以上の議論をまとめよう．

| | |
|---|---|
| 第 1 手 | $A$ は「出力 $f(x_0)$ を精度 $\varepsilon$ で求めたい」と宣言する． |
| 第 2 手 | $B$ は「入力 $x_0$ を精度 $\delta$ で入力すればいけるよ」と教える． |
| 第 3 手 | $A$ は $x_0$ と誤差 $\delta$ の範囲内の有理数 $x\in\mathbb{Q}$ を選び，$f(x)$ を求める． |
| 勝利条件 | $B$ が勝利しているので，$B$ の勝利条件から，$f(x)$ と $f(x_0)$ の誤差は $\varepsilon$ 程度しかない． |
| 結論 | そういうわけで，$A$ は無事に出力 $f(x_0)$ を精度 $\varepsilon$ で求めることができた． |

つまり，関数の連続性とは，関数の**任意精度近似可能性**というものにおおよそ対応する．このように，連続性とは「関数のグラフが連続的に繋がっている」というよりも，図形的な意味は忘れて「関数の任意精度近似が可能である」と考える方が，連続性の定義の理解として有効であることがしばしばある．

**豆知識．**この事実をもってして，「**計算可能関数は必ず連続である**」という主張が解析学における計算可能性理論の基本定理となっている．しかし，初学者の中には，この「計算可能性解析学の基本定理」に違和感を抱く人もいるようである．そのひとつの理由としては，「計算論というものは離散的な数学である」といった不正確なイメージが一般的には植え付けられており，さらに「『離散』と『連続』が対極にある」という漠然とした印象を持っている人が多いからであろう．しかし，「離散」と「連続」が相反するものであるというのは，ある意味では誤りである．なぜかといえば，位相空間論の基本的な事実であるが，「**離散空間上の全ての関数は連続**」だからである．この観点に基づくと，離散的な計算理論が容易な理由は，離散空間こそが最も多くの連続関数を持つ空間であるからである，と解釈できる．一方，離散空間から離れれば離れるほど連続関数が減り，それ故に計算論の展開が難しくなっていく．

離散空間 $\Longrightarrow$ 連続関数が多い $\Longrightarrow$ 計算論の展開が容易．

離散から程遠い空間 $\Longrightarrow$ 連続関数が少ない $\Longrightarrow$ 計算論の展開に工夫が必要.

## §2. 解析学における計算可能性と不可能性

**■構成的とは何か:** さて，初等解析学の定理群の興味深い点は，「微妙に非構成的」であるという部分である．つまり，真の選択公理が必要なほど超越的というわけでもないが，ある程度の無限的な論法を要する．たとえば，本稿で挙げる初等解析学の定理については，選択公理なしの集合論 ZF と比べてさえミジンコみたいな超弱い体系である $\mathsf{ACA}_0$ で全て証明できる．一方で，そんなミジンコ理論である $\mathsf{ACA}_0$ もそれほど極端に弱いわけではない．$\mathsf{ACA}_0$ ですら停止問題のような理論的に絶対に計算不可能な集合や関数の取り扱いができるという点を考慮に入れれば，地にしっかりの足に付いた日常的な観点から言えば，$\mathsf{ACA}_0$ の時点で既に非構成的で超越的すぎるとも言えるだろう．しかし，あまり $\mathsf{ACA}_0$ やら $\mathsf{RCA}_0$ やら $\mathsf{EL}_0$ だのといった摩訶不思議な謎の記号を出しすぎても，初学者の理解の妨げになるだけであろうから，ここからはもう少し具体的な話に移りたい.

**■初等解析学の計算可能性と不可能性:** 「構成的」という概念を厳密に記述するのは難しいし，あまり一般的すぎてもイメージを掴みづらいので，ここでは「構成的」の一種である「計算可能」に焦点を絞って議論しよう．ただし，あくまで計算可能性は構成的概念の特殊例であり，計算可能性以外にも様々な構成的概念がある.

計算可能性には厳密な数学的定義があるが，計算論の講義ではないので，ここでは説明しない．しかし，ほとんどの現代人が感覚的に持っている「コンピュータ・プログラムを用いて現実に具体的に計算できるもの」程度の漠然とした理解でも，ここから先の理解にさほど影響は及ぼさないであろう．ただし，ここからの議論には全てに厳密な数学的定義があり，曖昧さなど欠片もないという点だけは，頭に留めておいて欲しい.

さて，実数はコーシー列の同値類として構成する方法があった．実際には，有理数のコーシー列のみを考えれば十分である．さて，有理数は分子と分母を表す整数の対として表現できるので，有理数列，有理数上の関数の計算可能性の定義などは明白であろう．すると，計算可能実数とは，有理数の計算可能なコーシー列（の同値類）として導入するのが妥当かと思われる．それでは，計算可能なコーシー列とは何か，という議論に移ろう．まず，$(x_n)_{n\in\mathbb{N}}$ がコーシー列であるとは，以下のゲームにおいて E が必勝戦略を持つことであった.

| | 1 手目 | 2 手目 | 3 手目 | 勝利条件 |
|---|---|---|---|---|
| A | $\varepsilon > 0$ | | $n, m \geqslant k$ | $\lvert x_n - x_m\rvert \geqslant \varepsilon$ |
| E | | $k$ | | $\lvert x_n - x_m\rvert < \varepsilon$ |

実数 $\varepsilon$ は有理数だけを考えれば十分であるし，更に言えば，自然数 $s \in \mathbb{N}$ に対して $\varepsilon = 2^{-s}$ の形のものだけ考えれば十分である．つまり，有理数列 $(q_n)_{n\in\mathbb{N}}$ がコーシー列であるとは，以下の

ゲームに E が必勝戦略を持つことである．

| | 1 手目 | 2 手目 | 3 手目 | 勝利条件 |
|---|---|---|---|---|
| A | $s$ | | $n, m \geqslant k$ | $\lvert q_n - q_m \rvert \geqslant 2^{-s}$ |
| E | | $k$ | | $\lvert q_n - q_m \rvert < 2^{-s}$ |

そうすると，有理数列 $(q_n)_{n\in\mathbb{N}}$ と上記のコーシー性ゲームにおける E の必勝戦略 $\mathcal{S}$ の対 $((q_n)_{n\in\mathbb{N}}, \mathcal{S})$ が実数を表現している．この考えを発展させて，**計算可能**な有理数列 $(q_n)_{n\in\mathbb{N}}$ と，この列に対するコーシー性ゲームにおける E の**計算可能**な必勝戦略 $\mathcal{S}$ の対 $((q_n)_{n\in\mathbb{N}}, \mathcal{S})$ が**計算可能実数**を表現していると考えるのである．

他にも計算可能実数の同値な定義はたくさんある．最もよく見かける定義としては，急収束 (rapidly convergent) する有理コーシー列を使うものがある．別の定義としては，計算可能デデキント切断を使うものもあるし，他にも色々な方法がある．いずれも基本的には同値になるが，ここでは詳細は省略する．

さて，計算可能実数を有理数列とコーシー性ゲームの計算可能必勝戦略の対として取り扱うという発想は，他の概念にも適用可能である．つまり，ここまでで取り扱った様々なゲームに対して，「E の**計算可能**な必勝戦略の存在」として，それぞれの概念の計算可能版を定義できる．

たとえば，関数 $f : \mathbb{R} \to \mathbb{R}$ が点 $x_0$ で**計算可能連続**であるということを，E が以下の連続性ゲームに対して計算可能な必勝戦略を持つこととして定義することは妥当かと思われる．

| | 1 手目 | 2 手目 | 3 手目 | ルール | 勝利条件 |
|---|---|---|---|---|---|
| A | $s \in \mathbb{N}$ | | $q \in \mathbb{Q}$ | $\lvert q - x_0 \rvert < 2^{-t}$ | $\lvert f(q) - f(x_0) \rvert \geqslant 2^{-s}$ |
| E | | $t \in \mathbb{N}$ | | | $\lvert f(q) - f(x_0) \rvert < 2^{-s}$ |

計算可能連続関数は，伝統的に，単に計算可能関数と呼ぶことが多い．

## 2.1. 単調収束定理

単調収束定理とは，実数の有界単調列には極限が存在する，という定理であった．計算可能な実数の有界単調列が与えられたとき，その極限の任意精度近似を計算可能な方法で求めることができるだろうか．実は，それは不可能であるということが 1940 年代には既に知られていた．

**定理 2.1.** 計算可能な極限を持たないような計算可能な有理数の有界単調列が存在する．

このような列は，**スペッカー列** (*Specker sequence*) としてよく知られている．証明は計算論を知っていれば極めて簡単で，計算可能性理論における最も初等的な定理の 1 つである．

*Proof (*定理 *2.1).* 集合 $A \subseteq \mathbb{N}$ に対して，$\rho_A$ を次によって定義される実数とする．

$$\rho_A = \sum_{n\in A} 2^{-n}.$$

集合 $A$ が空であるか，ある計算可能関数 $f : \mathbb{N} \to \mathbb{N}$ の像であるとき，$A$ は計算可枚挙 (computably enumerable) あるいは再帰的可算 (recursively enumerable) と呼ばれる．集合 $A$ が

計算可枚挙ならば実数 $\rho_A$ は計算可能な有理数の有界単調増大列の上限であることは自明であろう．一方，計算不可能な計算可枚挙集合が存在することはよく知られている．たとえば，$A$ を停止問題とすればよい．このとき $\rho_A$ が計算不可能であることは明らかである． □

**■位相空間論的解釈:** 単調収束定理の非構成性を計算論を知らない人にも説明する方法がある．$B \subseteq \mathbb{R}^{\mathbb{N}}$ を実数の有界単調列全体の集合とする．$\mathbb{R}^{\mathbb{N}}$ にはユークリッド空間の積位相が入っているものとし，$B \subseteq \mathbb{R}^{\mathbb{N}}$ にはその相対位相を入れる．すると，単調収束定理から，次の関数を考えることができる．

$$L : B \to \mathbb{R}; \quad L((x_n)_{n\in\mathbb{N}}) = \lim_{n\to\infty} x_n.$$

スペッカー列の存在証明は，この**関数 $L : B \to \mathbb{R}$ が不連続関数である**ことを示しているのに極めて近い．連続性と任意精度近似可能性とは同値であったから，つまり，関数 $L : B \to \mathbb{R}$ を任意精度で近似することは不可能である．特に，$L$ は計算不可能である，つまり，実数の有界単調列の極限値をコンピュータによって任意精度で近似する一般的な方法は存在しない．

**豆知識.** 補足であるが，連続関数の各点極限として書ける関数は**ベール 1 級** (*Baire class one*) であると言われる．つまり，関数 $L$ は連続でないベール 1 級関数の具体例である．また，$L$ の定義域を有界単調増大列に制限すれば，**下半連続** (*lower semicontinuous*) となり，有界単調下降列に制限すれば，**上半連続** (*upper semicontinuous*) となる．実際，単調収束定理の構成的証明の非存在は，連続でない半連続関数の存在という有名な事実とほぼ同等である．

ちなみに計算可能な単調増大実数列の極限として書ける実数は，**下半計算可能** (*lower semicomputable*) と呼ばれ，計算可能な単調減少実数列の極限として書ける実数は，**上半計算可能** (*upper semicomputable*) と呼ばれる．スペッカー列の存在は，計算不可能な下半（上半）計算可能実数の存在を意味する．

## 2.2. 有界性定理

$f$ に対する一様連続ゲームに対して $B$ が計算可能な必勝戦略を持つとき，$f$ は計算可能一様連続であるという．次はコンパクト性を用いて容易に示せる．

**定理 2.2.** *$f$ が有理端点を持つ有界閉区間上の計算可能関数ならば，計算可能一様連続である．*

上では，定義域を有理端点を持つ有界閉区間上としているが，現代的な用語を用いれば，計算可能コンパクト集合ならば何でもよい．さて，有界性定理の計算可能版を示そう．有界性定理のよく知られた証明では，ボルツァノ・ワイエルシュトラスの定理を用いるものが多い．しかし，ボルツァノ・ワイエルシュトラスの定理はかなり非構成的な定理であることが知られており，これを用いると，計算可能性などの構成的成分は失われてしまう．一方，有界性定理については，非構成的な手法を用いずとも，以下のように容易に示せることがよく知られている．

**定理 2.3** (計算可能有界性定理)**.** *$f$ を計算可能一様連続関数とする．もし $f$ の定義域が計算可能な上限と下限を持つ有界区間ならば，$f$ の値域は計算可能な上限と下限を持つ．*

*Proof.* $f$ を有界集合 $S$ 上の計算可能一様連続関数とする．E は $f$ に対する一様連続ゲームの計算可能な必勝戦略を用いて，以下のようにゲームを進める．

| | 1 手目 | 2 手目 | 3 手目 | ルール | 勝利条件 |
|---|---|---|---|---|---|
| A | $2^{-k}$ | | $x, y$ | $\lvert x-y\rvert < \delta_k$ | |
| E | | $\delta_k$ | | | $\lvert f(x)-f(y)\rvert < 2^{-k}$ |

したがって，定義域の有界集合 $S$ を長さ $\delta_k$ 未満の部分区間 $(I_j)_{j<n_k}$ の和として書き表せば，各 $I_j$ 上で $f$ の値は高々 $2^{-k}$ の内部に収まる．$x_{k,j} = \min I_j$ とする．明らかに，

$$(\forall x \in S)(\exists j < n_k)\ |f(x) - f(x_{k,j})| < 2^{-k}$$

が成立するから，$M_k = \max_{j<n_k} f(x_{k,j}) + 2^{-k}$ および $m_k = \min_{j<n_k} f(x_{k,j}) - 2^{-k}$ は $f$ の値域のそれぞれ上界と下界をなす．さらに，

$$M := \lim_{k\to\infty} M_k = \sup\{f(x) : x \in S\}, \qquad m := \lim_{k\to\infty} M_k = \inf\{f(x) : x \in S\}$$

であることは容易に確かめられるであろう． □

上の定理において，もし $f$ の定義域が有界閉集合であれば，上の $M$ と $m$ は実際に $f$ の値域の最大値および最小値となるから，つまり $f$ の値域の最大値と最小値は計算可能である，ということである．しかし，これはあくまで出力先の計算可能性を述べるのみであり，最大値および最小値を達成する入力が具体的に何であるかを見つけるのは一般には計算不可能である，というものが次に述べる最大値・最小値定理の非構成性（定理 2.4）である．

## 2.3. 最大値・最小値定理

最大値・最小値原理または極値定理 (extreme value theorem) とは，有界閉区間を定義域とする実連続関数は最大値または最小値を取る，というものである．

**定理 2.4.** どんな計算可能実数上でも最大値・最小値を取らないような計算可能関数 $f : [0,1] \to \mathbb{R}$ が存在する．つまり，どんな計算可能実数 $x \in [0,1]$ に対しても，$f(a) < f(x) < f(b)$ となる $a, b \in [0,1]$ が存在する．

**■数学基礎論的証明:** 最大値・最小値原理が非構成的であることの証明については様々なものが知られているが，ここではゲーデルの不完全性定理を利用した証明を紹介しよう．以下の過程は，再帰的公理化可能な理論 $T$ が計算可能関数 $F$ に変貌する様を描写する．

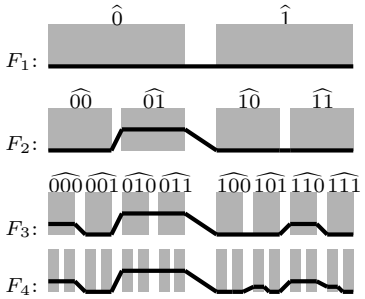
. . . . . .

0 と 1 の無限列の集合 $\{0,1\}^{\mathbb{N}}$ と 3 進カントール集合 $C \subseteq [0,1]$ が同相であることはよく知られている[*1]ので，同相写像 $\alpha \mapsto \hat{\alpha} : \{0,1\}^{\mathbb{N}} \to C$ を固定しよう．実際には，カントール集合の構成を模倣すれば，有限列 $\sigma \in \{0,1\}^*$ に対して区間 $\hat{\sigma} \subseteq [0,1]$ が対応する．

さて，0 と 1 の無限列 $\alpha \in \{0,1\}^{\mathbb{N}}$ が与えられたとき，$\alpha(n) = 1$ によって『$n$ 番目の文は真である』を表し，$\alpha(n) = 0$ によって『$n$ 番目の文は偽である』を表す．各無限列 $\alpha$ に対応する実数 $\hat{\alpha} \in [0,1]$ に対して，

$$F(\hat{\alpha}) = 0 \iff \text{理論 } T \text{ は } \alpha \text{ と矛盾しない}$$

を成立させるような計算可能関数 $F$ を構成しよう．$F$ を近似する有理折れ線の列 $\{F_n\}_{n\in\mathbb{N}}$ は以下の手続きによって描かれる．




【$F_n$ の構成】

1. 長さ $n$ の各列 $\sigma \in \{0,1\}^n$ について，$F_{n-1}$ の構成までに $T + \sigma$ の矛盾を証明していたか？
   YES: 区間 $\hat{\sigma}$ 上の $F_n$ の値は $F_{n-1}$ と等しい．
   NO: ステップ 2 に進む．
2. 長さ $n$ 以下の証明で，$T + \sigma$ から矛盾を導くものは存在するか？
   YES: 区間 $\hat{\sigma}$ 上 $F_n$ の値を $2^{-n}$ と指定する．
   NO: 区間 $\hat{\sigma}$ 上 $F_n$ の値を 0 と指定する．
3. $F_n$ は以上より生成される有理折れ線である．

**主張．** $T$ が無矛盾ならば，各 $F_n$ は零点を持つ．

証明．理論 $S$ が無矛盾ならば，任意の文 $\varphi$ に対して，$S + \varphi$ または $S + \neg\varphi$ の一方は無矛盾である．よって，帰納的に，長さ $n$ の列 $\sigma \in \{0,1\}^n$ で，$T + \sigma$ が無矛盾なものを得ているならば，$T + \sigma 0$ または $T + \sigma 1$ の一方は無矛盾である．したがって，対応する区間 $\widehat{\sigma i}$ 上，$F_{n+1}$ は値 0 を返す．　□

**主張．** $T$ が無矛盾ならば，$F$ は零点を持つ．

証明．$F$ が零点を持たないならば，$\{F_n^{-1}(0,1]\}_{n\in\mathbb{N}}$ は $[0,1]$ の開被覆である．閉区間 $[0,1]$ のコンパクト性より，有限部分被覆 $\{F_n^{-1}(0,1]\}_{n\leqslant k}$ が存在する．$F_n^{-1}(0,1] \subseteq F_{n+1}^{-1}(0,1]$ であるから，$F_k^{-1}(0,1]$ は $[0,1]$ を被覆する．言い換えれば，$F_k$ は零点は持たない．これは前主張に矛盾する．　□

$F$ の各零点 $\hat{\alpha}$ は，$T$ を含む無矛盾かつ完全な理論 $T_\alpha = T + \alpha$ に相当する．$T$ がロビンソン算

[*1] 歴史的にはカントール集合は当初は反例として用いられたものである．しかし，ここで述べたように，無限語の空間と同相であるなど，様々な文脈でカントール集合が最も基本的なオブジェクトとして自然に現れることが次第に判明した．このため，いくつかの分野では，今日ではカントール集合はむしろ「最も自然な空間」のひとつとして扱われている．ユークリッド空間などよりもカントール集合の方が自然で基本的な数学的対象だと言う人も多いであろう．

術を含むならば，$T_\alpha$ はロビンソン算術を含む無矛盾かつ完全な理論であるから，ゲーデルの不完全性定理より，$T_\alpha$ は再帰的公理化可能では有り得ない．言い換えれば，$\widehat{\alpha}$ は計算不可能点である．

さて，$F$ の最小値は 0 であるから，$F$ の最小値を見つけるのは $F$ の零点を見つけることに他ならないが，これは上記の議論より計算不可能である．最大値については，区間 $[1,2]$ で $F$ を上下反転させたものを貼り合わせた後，定義域を半分に縮めれば，目的の関数が得られることは容易に分かるだろう．

**■位相空間論的解釈:** 最大値・最小値原理の非構成性は, 位相空間論的には次のように解釈できる．関数空間 $C([0,1],\mathbb{R})$ にはコンパクト開位相が入っているものとする．関数 $f:[0,1]\to\mathbb{R}$ が与えられたとき，$\max f=\{z : f(z)=\max_{x\in[0,1]} f(x)\}$ および $\min f=\{z : f(z)=\min_{x\in[0,1]} f(x)\}$ と定義する．極値定理によって，$\max f$ も $\min f$ も空でないことが保証される．さて，極値定理を実現する次の関数を考えよう．

$$E : C([0,1],\mathbb{R})\to[0,1];\quad E(f)\in \max f\cup\min f.$$

もちろん，最大値・最小値は複数存在し得るので，上の条件を満たす $E$ は一意ではない．しかし，上の定理の証明が実際に導いていることは，上の条件を満たす連続関数 $E$ は存在しない，ということである．つまり，

**最大値・最小値定理を実現するどんな関数 $E : C([0,1],\mathbb{R})\to[0,1]$ も必ず不連続である**

ということである．

今回の場合のような解が一意でない問題に関して，厳密に言えば，このような位相的解釈するのはあまり適切ではない．（非構成性証明はこの位相的解釈よりも遥かに強いことを言っている！）しかし，計算論や構成的数学に馴染みがない人が，構成不可能性のイメージをぼんやりと掴むには，このような位相空間的解釈が理解しやすいと思われる．

特に，有界閉区間上の与えられた実連続関数の最大値・最小値をコンピュータによって任意精度で近似する一般的な方法は存在しない．

**豆知識.** しかし，実を言うと，極値定理を実現する $E$ について，単調収束定理のときの不連続関数 $L$ よりも「不連続度」は低い．この不連続度の低さは，証明に必要な公理，という観点でも，単調収束定理よりは容易である，という点に反映される．与えられた数学の定理を証明するのに必要可能な公理を調べる**逆数学** (*reverse mathematics*) と呼ばれる分野がある．単調収束定理は $\mathsf{ACA}_0$ と同値であるのに対し，極値定理はそれより真に弱い $\mathsf{WKL}_0$ と同値であることが容易に確かめられる．単調収束定理も極値定理もいずれも非構成的な定理であるが，構成不可能性というものにもレベルがあるのである．

## 2.4. 近似的な最大値・最小値定理

最大値・最小値定理は非構成的であると述べた．しかし，ある意味で，最大値・最小値定理には，構成版と呼ばれるものが存在する．つまり，最大値・最小値を達成する入力を計算するのは諦めたとしても，最大値・最小値 $\pm\varepsilon$ を達成する入力を計算することを試みることはできる．

**定理 2.5** (近似的最大値・最小値定理)．$f : [0,1] \to \mathbb{R}$ を計算可能関数とする．与えられた $k \in \mathbb{N}$ に対して，次の条件を満たす有理数 $x_0$ と $x_1$ を返す計算可能関数が存在する．

$$|f(x_0) - \min_{x\in[0,1]} f(x)| < 2^{-k}, \quad |f(x_1) - \max_{x\in[0,1]} f(x)| < 2^{-k}.$$

通常の最大値・最小値定理との違いを明確にするために，この定理をゲームとして解説しよう．まず，関数 $f$ に対する通常の最大値・最小値定理をゲームとして表すと以下のように書ける．

| | 1 手目　2 手目 | 勝利条件 |
|---|---|---|
| A | $a$ | $f(a) < f(x_0)$ or $f(x_1) < f(a)$ |
| E | $x_0, x_1$ | $f(x_0) \leqslant f(a) \leqslant f(x_1)$ |

つまり，E さんは最大値を達成する $x_1$ と最小値を達成する $x_0$ の候補を持ってきて，それが本当に正しければ E さんの勝利ということである．E さんは一般にはこれに対する計算可能な必勝戦略を持たない，というのが先に述べたことであった．

一方で，定理 2.5 のような近似的最大値・最小値定理ゲームを考えよう．近似的最大値・最小値定理ゲームでは，E さんは $|f(x_1) - \max_a f(a)| < \varepsilon$ および $|f(x_0) - \min_a f(a)| < \varepsilon$ となる $x_0, x_1$ を見つけてくれば勝ちである．この式は，任意の $a \in [0,1]$ について $f(x_0) - \varepsilon < f(a) < f(x_1) + \varepsilon$ となることと同値であるから，次のゲームを考えればよい．

| | 1 手目　2 手目　3 手目 | 勝利条件 |
|---|---|---|
| A | $\varepsilon > 0$　　$a$ | $f(a) \leqslant f(x_0) - \varepsilon$ or $f(x_1) + \varepsilon \leqslant f(a)$ |
| E | $x_0, x_1$ | $f(x_0) - \varepsilon < f(a) < f(x_1) + \varepsilon$ |

*Proof (*定理 2.5*).* 上記ゲームに対する計算可能な必勝戦略が存在するということを示すために，$2^{-k} < \varepsilon$ なる $k$ を取ってくる．計算可能有界性定理 2.3 の証明における $M_k$ と $m_k$ を実現する $x_{k,j}$ をそれぞれ $x_1, x_0$ とすれば，これが必ず E の勝利条件を満たすことは容易に分かる．　□

さて，最大値・最小値定理と近似的最大値・最小値定理の違いは何であろうか．近似的定理では，我々は $\varepsilon$ に対して，$f(x_\varepsilon)$ が極値を $\varepsilon$-近似するような $x_\varepsilon$ を見つけてくることができた．しかし，それは $x_\varepsilon$ の近くに本当の極値が存在することを意味しない．つまり，もし精度を少し上げると，$x_\varepsilon$ の近くには極値が一切ないということが判明するかもしれない．つまり，精度 $\varepsilon$ を修正する度に，全く別の場所にある実数 $x_\varepsilon$ を取ってくる必要性に迫られる．実際，最大値・最小値定理の非構成性が述べていることは，このような極値の任意精度近似不可能性であり，つまりは，精度を変えると，絶対にこのような実数の完全な選び直しを行わなければならない．

最大値・最小値定理のように，計算可能性あるいは任意精度近似可能性の意味では成立しないが，定理 2.5 のような，いわゆる「その場しのぎ近似」型の構成的定理は成立する，というものは多々ある．応用上は「その場しのぎ近似」型の構成的定理で十分であることも多い．

近似的最大値・最小値定理のような「その場しのぎ近似」型の主張は，論理的には元の主張とは異なることに注意しよう．たとえば，ブラウワーの不動点定理の構成的証明は絶対に存在し得な

いということが数学的に証明されている一方で，ブラウワーの不動点定理の構成的証明と呼称される何かが色々と知られている．

## 2.5. 中間値の定理

中間値の定理は，初等解析学の定理の中ではそこまで非構成的ではない部類の定理であり，実際，以下のような計算可能版が存在する．

**定理 2.6** (計算可能中間値の定理)．$f : [0,1] \to \mathbb{R}$ を $f(0) < 0 < f(1)$ となる計算可能関数とする．このとき，$f(x) = 0$ となる計算可能実数 $x \in [0,1]$ が存在する．

*Proof.* $f^{-1}\{0\}$ が有理数を含むならば，有理数は計算可能実数であるから，主張は成立する．$f^{-1}\{0\}$ が有理数を含まない場合は，通常の中間値の定理のように，一点に収束する部分区間の無限下降列を得るが，これは収束先の実数を任意精度近似しているので，主張を得る． □

上の証明の難しい点は，$f^{-1}\{0\}$ が有理数か否かという非構成的場合分けを含むところにある．与えられた実数が有理数か否かで場合分けをするという操作とは，すなわちディリクレの関数

$$\chi_{\mathbb{Q}} = \lim_k \lim_j \cos(k!\pi x)^{2j} = \begin{cases} 1 & \text{if } x \in \mathbb{Q}, \\ 0 & \text{if } x \in \mathbb{R}\backslash\mathbb{Q} \end{cases}$$

を証明に合成することであり，そしてディリクレの関数が連続関数から程遠いことは皆もよく知っている．場合分けが連続ではないから，定理の解 $x$ を求める流れを任意精度近似ではシミュレートすることができない．この事実が導くことを明らかにするために，以下の中間値の定理ゲームを考えよう．

| | 1 手目　2 手目 | ルール | 勝利条件 |
|---|---|---|---|
| A | $f$ | $f(0) < 0 < f(1)$ | $f(x) \neq 0$ |
| E | $x$ | | $f(x) = 0$ |

計算可能中間値の定理 2.6 が述べることは，計算可能な $f$ 毎に，E さんが勝利するような計算可能な $x$ が**存在する**ということを述べる存在証明であり，具体的な $x$ の計算方法が分かるとは限らない．そして，証明中の非構成的場合分けは，実際に除去不可能であることが分かり，E さんは中間値の定理ゲームの計算可能な必勝戦略を持たないことが証明できる．

**定理 2.7** (中間値の定理の一様計算不可能性)．$f(0) < 0 < f(1)$ となる計算可能関数 $f : [0,1] \to \mathbb{R}$ に対して，$f(x) = 0$ となる $x \in [0,1]$ を見つける計算可能なアルゴリズムは存在しない．

この証明は，定理 2.4 と同様の発想に基づくが，定理 2.7 の証明の方が少し易しい．証明の方針としては，まず，計算可能関数の列 $(f_e : [0,1] \to \mathbb{R})_{e \in \mathbb{N}}$ を作る．そして，与えられた $e \in \mathbb{N}$ から

$f_e(x_e) = 0$ なる $x_e$ を見つけるアルゴリズムが存在しないことを示す．定理 2.4 のように，数学基礎論的証明を与えよう．

**■準備:** $g_s^{\pm}, g_s^{-}, g_s^{+} : [0,1] \to \mathbb{R}$ を以下のそれぞれ 4 点を結ぶ区分的線形写像とする．

$$(0,-1), \left(\frac{1}{4}, -\frac{1}{2^s}\right), \left(\frac{3}{4}, \frac{1}{2^s}\right), (1,1)$$
$$(0,-1), \left(\frac{1}{4}, \frac{1}{2^s}\right), \left(\frac{3}{4}, \frac{1}{2^s}\right), (1,1)$$
$$(0,-1), \left(\frac{1}{4}, -\frac{1}{2^s}\right), \left(\frac{3}{4}, -\frac{1}{2^s}\right), (1,1)$$

このとき，$g_s^{-}(x) = 0$ ならば $x < \frac{1}{4}$ であり，$g_s^{+}(x) = 0$ ならば $x > \frac{3}{4}$ であることは明らかであろう．与えられた $e \in \mathbb{N}$ に対して，計算可能関数 $f_e : [0,1] \to \mathbb{R}$ を構成しよう．$f_e$ は計算可能関数列 $(f_{e,s})_{s\in\mathbb{N}}$ の計算可能一様極限として与えられ，$g_s^{+}$, $g_s^{-}$, または $\lim_s g_s^{\pm}$ のいずれかに収束する．まず，$f_{e,0} = g_0^{\pm}$ から開始する．

**■ $(f_e)$ の構成:** いま，無矛盾な理論 $T$ を固定し，$\varphi_e$ を $e$ 番目の論理式とする．$f_{e,s+1}$ を定義するために，理論 $T$ における $\varphi_e$ または $\neg\varphi_e$ のサイズ $s$ 以下の証明図を探索する．$s$ 以前に既にそのような証明図が見つかっていたならば，$f_{e,s+1} = f_{e,s}$ とする．もしサイズ $s$ の証明図の探索の結果，$\varphi_e$ の証明が見つかったならば，$f_{e,s+1} = g_{s+1}^{+}$ とする．同様に，$\neg\varphi_e$ の証明が見つかったならば，$f_{e,s+1} = g_{s+1}^{-}$ とする．$\varphi_e$ と $\neg\varphi_e$ のどちらについてもサイズ $s$ 以下の証明図が存在しないならば，$f_{e,s+1} = g_{s+1}^{\pm}$ と定義する．このとき，$f_e = \lim_s f_{e,s}$ と定義する．すると，以下が成立していることは容易に分かる．

$$\begin{cases} f_e = \lim_{s\to\infty} g_s^{\pm} & \text{if } T \nvdash \varphi_e \text{ and } T \nvdash \neg\varphi_e, \\ f_e \in \{g_s^{+} \mid s \in \mathbb{N}\} & \text{if } T \vdash \varphi_e, \\ f_e \in \{g_s^{-} \mid s \in \mathbb{N}\} & \text{if } T \vdash \neg\varphi_e \end{cases}$$

特に，以下が成立している．

$$\left[\left(\exists x \geqslant \frac{1}{4}\right)\ f_e(x) = 0\right] \implies T \nvdash \neg\varphi_e,$$
$$\left[\left(\exists x \leqslant \frac{3}{4}\right)\ f_e(x) = 0\right] \implies T \nvdash \varphi_e.$$

**■零点の計算不可能性:** さて，$e \mapsto f_e$ を構成する計算可能なアルゴリズムは具体的に与えている．もし $f_e \mapsto x_e$ で $f_e(x_e) = 0$ なるものを見つける計算可能なアルゴリズムが存在したと仮定しよう．このアルゴリズムは $x_e$ の任意精度近似を与える．よって，このアルゴリズムに $x_e \pm \frac{1}{8}$ の間の有理数を尋ねて，$q$ が返ってきたとしよう．このとき，$q \geqslant \frac{1}{2}$ ならば $h(e) = 1$ とし，$q < \frac{1}{2}$ ならば $h(e) = 0$ とする．この $h$ は計算可能である．とくに，

$$h(e) = 1 \iff q \geqslant \frac{1}{2} \implies x_e \geqslant \frac{1}{4} \implies T \nvdash \neg\varphi_e$$
$$h(e) = 0 \iff q < \frac{1}{2} \implies x_e \leqslant \frac{3}{4} \implies T \nvdash \varphi_e$$

であるから，$\hat{T} = \{\varphi_e \mid h(e) = 1\}$ は $T$ を含む無矛盾かつ完全な理論である．しかし，$h$ が計算可能であるならば，$\hat{T}$ が再帰的公理化可能であるということである．もし，$T$ をロビンソン算術を含む無矛盾な再帰的公理化可能理論として取ってきていたならば，これはゲーデルの不完全性定理に矛盾する．よって，そのような $T$ に対して，$f_e \mapsto x_e$ は計算可能では有り得ない．特に，定理が導かれる．

**豆知識.** 計算可能中間値の定理 2.6 をもう少し細かく分析すると，古典逆数学の文脈では，中間値の定理が $\mathsf{RCA}_0$ で証明できることを示せる．一方，直観主義逆数学の文脈では，中間値の定理はある種の非構成的原理を要求することが分かる．具体的には，中間値の定理は LLPO あるいは $\Sigma^0_1$-ド・モルガンの法則と呼ばれる非構成的原理を導く．

## 2.6. ロシア学派の構成的解析学

これまで，計算可能関数の定義域として，全ての実数 $\mathbb{R}$ あるいは有界閉区間 $[0,1]$ などを取っていた．しかし，我々は計算可能実数値しか入力できないのであるから，計算不可能実数が定義域に入っているかどうかを気に留める必要はあるだろうか．

同様に，連続性を任意精度近似可能性と捉えた場合，計算可能実数上（あるいは代数的実数）で任意精度近似できればいいと考えると考えるのは自然だろう．この場合，計算不可能実数は定義域に入らないかもしれないし，定義域に入ったとしても不連続点になるかもしれない．つまり，計算不可能な入力 $x$ の近似から $f(x)$ の近似を求めることはできないかもしれない．

しかし，そもそも我々が計算不可能な値を入力することはないから，計算不可能実数が定義域に入っていようがいまいが，連続点になろうが不連続点になろうが，全く気にする必要はないのである．これがロシア学派の構成的数学と呼ばれるものの考えである．

■* 計算可能有界性定理・再考: 定理 2.2 と 2.3 を合わせると，任意の計算可能関数 $f : [0,1] \to \mathbb{R}$ には最大値 $\max\{f(x) \mid x \in [0,1]\}$ と最小値 $\min\{f(x) \mid x \in [0,1]\}$ が存在し，それらは共に計算可能であった．一方，ロシア学派の構成的数学のように，計算可能実数入力のみしか気にする必要がないという考えでは，実は関数 $f$ の有界性すら保証できないのである．

以下，$X \subseteq \mathbb{R}$ について，$X_{\mathrm{com}}$ を $X$ に含まれる計算可能実数全体の集合としよう．たとえば，$[0,1]_{\mathrm{com}}$ は単位閉区間 $[0,1]$ に含まれる計算可能実数全体の集合である．

**定理 2.8** (有界性定理の不成立)**.** 計算可能関数 $f : [0,1]_{\mathrm{com}} \to \mathbb{R}$ で，値域が非有界であるものが存在する．

*Proof.* 関数 $F$ を定理 2.4 の証明で用いた計算可能関数とする．このとき，$H(x) = \frac{1}{F(x)}$ を考えよう．任意の計算可能実数 $x$ について，$F(x) > 0$ であるから，$H$ は $[0,1]_{\mathrm{com}}$ 上で定義されている．また，$F$ は計算可能なので，$F(x) > 0$ となる実数 $x$ に対して $H(x) = \frac{1}{F(x)}$ を計算できる．つまり，$H : [0,1]_{\mathrm{com}} \to \mathbb{R}$ は計算可能である．

一方で，$H : [0,1]_{\mathrm{com}} \to \mathbb{R}$ が有界だったと仮定しよう．$F$ の定義より，ある $n$ について，どんな文 $\sigma$ についても，$T + \sigma$ の矛盾を導く長さ $n$ 以上の証明は存在しないということである．つまり，長さ $n$ までの証明を機械的にチェックすれば，$T + \sigma$ と $T + \neg\sigma$ が矛盾を導くか否かを確認できる．よって，計算可能な方法で，矛盾を導かない文を次々に公理として $T$ に加えていくことにより，$T$ を含む無矛盾かつ完全な再帰的公理化可能理論を作ることができるが，これはゲーデルの不完全性定理に矛盾する． □

実数区間の有界稠密集合上で定義された一様連続関数の値域は明らかに有界であるから，上の定理の系として，計算可能関数 $f : [0,1]_{\mathrm{com}} \to \mathbb{R}$ で，一様連続でないものが存在することが分かる．

**■位相空間論的解釈:** 位相的には，このトリックは以下のように理解できる．$[0,1]_{\mathrm{com}}$ はあくまで $F_\sigma$ 集合であってコンパクトではない．このため，連続関数 $f : [0,1]_{\mathrm{com}} \to \mathbb{R}$ で非有界なものを構成するだけであれば，位相空間論の簡単な演習問題である．とはいえ，この $f$ として，単に連続というだけでなく計算可能性を保証するのは，定理 2.8 の証明のようにそう簡単ではなかった．というのも，たとえ超越的観点からは $[0,1]_{\mathrm{com}}$ がコンパクトではないということを知っていたとしても，ロシア学派の世界にいる人類からは計算可能実数しか見えていない．したがって，計算論的視点からは $[0,1]$ と $[0,1]_{\mathrm{com}}$ の区別が付かないはずである．しかし，それにも関わらず，$[0,1]$ は計算可能コンパクトであるが，$[0,1]_{\mathrm{com}}$ は計算可能コンパクトでない，ということが成り立ってしまう．

つまり，現実的には計算可能実数値しか入力できないのだから，計算可能実数値上での連続性（任意精度計算可能性）のみを要求しておけば十分であろう，と考えていると，このように思わぬ所で足をすくわれることもあるのである．

## §3. 実数の非可算性証明

ここまでで見てきたように，不幸にも，実数論や初等実解析の多くの基本定理たちは，計算によっては実現できないということが示されてしまう．このように，超越的なものが溢れる実数論や初等実解析の主張の中で，飛び抜けて構成的なものがある．そのひとつは，実数の非可算性証明，つまり**対角線論法** (*diagonal argument*) である．つまり，実数論や実解析の主張の中では珍しく，実数の非可算性証明は有限的な観点，アルゴリズム的な観点から見ても価値のあるものである．これは，計算論的数学や構成的解析学などの分野（つまり排中律を仮定しない数学）ではよく知られている話である．

実数の非可算性の計算論的解釈としては，「計算可能実数全体の計算可能な番号付けは存在しない」あるいは「計算可能実数全体は計算的に非可算である」などと言い表されるであろう．ただ，実数の非可算性証明はそれ以上の情報量を持つ，ということを最初に少しだけ述べておく．

**■構成的アルゴリズムとしての対角線論法:**

まず，「**対角線論法は構成的アルゴリズムである**」という観点から一つの例を見ていこう．実数の非可算性の最も古い応用例としては，1874 年のカントールによる超越数の存在証明がある．大雑把に言えば，これは以下のステップによってなされる．

1. $\mathbb{Q}$-係数多項式は可算種類しか存在せず，零でない $\mathbb{Q}$-係数多項式は高々有限個しか根を持たないので，代数的数は可算種類しか存在しない．
2. 一方で，対角線論法によって，実数は非可算に存在する．
3. よって，ほとんどすべての実数は代数的でない，つまり超越数である．

このカントールの証明は非構成的証明の例として述べられることも多いが，そこは若干微妙なところがある．つまり，超越数の存在証明という部分だけを取り出せば，上の各ステップのアルゴリズムをすべて理解している人にとっては，このカントールの証明は，**超越数を具体的に作る構成的なアルゴリズム**である．

その前に，数学的には，実数とは（収束係数付きの）有理コーシー列の同値類である．つまるところ，実数とは，有理数による近似列である，というのが実数の定義であると思ってもよい．さて，カントールの証明を計算論的に分析しよう．

1. まず，与えられた $\mathbb{Q}$-係数多項式の根の値を任意精度で近似するアルゴリズム $A$ を作ることができる．また，代数的数の番号付けをするアルゴリズム $B$ を作ることができ，アルゴリズム $A$ を利用すれば，入力 $n$ に対して，$B$ による番号付けの $n$ 番目の代数的数の小数点以下 $n$ 桁目の値を出力するアルゴリズム $C$ を得る．
2. このアルゴリズム $C$ に対角線論法アルゴリズム $D$ を適用すると，つまり，入力 $n$ に対して，$n$ 番目の代数的数の小数点以下 $n$ 桁目の値とは異なる値を出力するアルゴリズム $E$ となる．
3. よって，このアルゴリズム $E$ の入力 $n$ に対する出力値を小数点以下 $n$ 桁目の値とする実数を考えれば，つまり $E$ は計算可能な超越数（つまり，ある超越数の任意精度近似アルゴリズム）を具体的に与える．

そういうわけで，カントールの対角線論法による証明は，具体的な超越数を与えるアルゴリズムという側面も持つことが分かった．このように，たとえば計算可能性理論においては，ある性質を持つ何かを具体的に構成する計算可能なアルゴリズムを作る際に，対角線論法あるいはその一般化をしばしば利用する．

以上をまとめると，実数の非可算性証明とは，計算論的には，「可算種類の実数のリスト（を生成するアルゴリズム）を入力として，そのリストには含まれていない実数を出力するアルゴリズム」と考えられる．対角線論法のような「入力を求める構成的論法」は，証明の一部をブラックボックスにしてしまうと，何を入力したらよいか分からない，という状況になり，非構成的な証明に見えてしまうことがある．しかし，証明のすべてのステップを明確にすると，「入力を求める構成的論法」への入力を具体的に構成することができ，構成的な証明の姿が明らかになる．

カントールの超越数の存在証明と似た立場にある証明の例として，ボレルによる絶対正規数の存在証明がある．これはいわゆる測度論を用いて，「ほとんどすべての実数は絶対正規数である」と主張する．これもまたカントールの証明と並んで非構成的証明の例として述べられることがある．しかし，ボレルの証明もまた，絶対正規数の存在証明という部分だけ取り出せば，構成的な部分があると言えなくもない．ボレルはその辺りを明確にしていないが，後のルベーグによる証明は，絶対正規数の構成的存在証明を与えているように見える，とされる．ただし，ボレルやルベーグがその証明を行った時代には，計算可能性の厳密な定義が与えられていなかったので，具体的な構成アルゴリズムを与えていると言えるかどうかは議論の余地があるようである．確実な意味で，絶対正規数の具体的な構成アルゴリズムを実際に書き下したのは，チューリングである．

チューリングによる絶対正規数の構成的存在証明（絶対正規数の構成アルゴリズム）は，後の時代に詳細な分析がなされることとなった．チューリングのアルゴリズムは改良に改良を重ねられ，その後，絶対正規数の構成アルゴリズムの具体的な計算量などについても詳しく議論されている．

**■実現可能解釈:** 実現可能性の文脈から，数学的に厳密に，実数の非可算性証明について議論しよう．ところで，実現可能性のコンピュータ科学へのひとつの応用として，「証明からのプログラム抽出」というものが挙げられることがあるようである．筆者はそちら方面についてあまり詳しくないが，おそらく上で説明したようなものが，「証明からのプログラム抽出」の一例なのだろうと思われる．

とはいえ，実数の非可算性証明がどれくらいすばらしく構成的であるかは，「非可算」という言葉をどう定義するかに依存する．代表的な可算性の定義は，以下の 2 つである．

1. 集合 $X$ が**可算**であるとは，ある全射 $f : \mathbb{N} \to X$ が存在することである．
2. 集合 $X$ が**可算**であるとは，ある単射 $f : X \to \mathbb{N}$ が存在することである．

前者は，「$X$ の要素を自然数を用いて数え尽くすことができる」という可算性のアイデアをダイレクトに表したもので，最も自然な可算性の定義だと思われる．後者は，「与えられた $X$ の要素に対して，自然数で重複しないように名前を与えられる」という，これも可算性のアイデアを書き表したものである．前者は「$\mathbb{N}$ **による枚挙可能性**」であり，後者は「$\mathbb{N}$ **による命名可能性**」であると表現すると分かりやすいかもしれない．他にも色々な可算性の定義が知られているが，ここではこの 2 つだけを考えよう．

もちろん，集合論など強い公理を許す体系であればこの 2 種類の可算性が同値であることはよく知られている．しかし，そこに構成的観点や計算可能性の観点などを入れると，そうそう簡単な問題ではなくなる．たとえば，自然数の部分集合の「枚挙可能性」について，計算的に枚挙可能な集合 $A \subseteq \mathbb{N}$ は，いわゆる計算量クラス RE に相当すると考えられる．一方，自然数の部分集合の「命名可能性」について，任意の集合 $A \subseteq \mathbb{N}$ に対して，$A$ の要素は自明に（恒等関数で）命名されている．よって，計算論的観点に立てば，枚挙可能性と命名可能性は**明らかに**別物である．

**■全射による非可算性:** まず，全射による非可算性について考察しよう．この意味で，実数が非可算であることは，直観主義論理上の解析学である構成的解析学などでもよく知られている．しかも，単に実数が「可算でない」というだけでなく，「構成的に非可算」である，というより強い性質を証明できる．つまり，

$$X \text{ が可算でない} \iff \neg[(\exists f : \mathbb{N} \to X)(\forall x \in X)(\exists n \in \mathbb{N})\ f(n) = x].$$
$$X \text{ が構成的に非可算} \iff (\forall f : \mathbb{N} \to X)(\exists x \in X)(\forall n \in \mathbb{N})\ f(n) \neq x.$$

もちろん，この 2 つは古典論理上では同値であるが，直観主義論理の上ではそうでもない．「可算でない」というのは，あくまで「$X$ を数え尽くせない」ことを述べているだけであるが，「構成的に非可算」というのは，「どんな方法を持ってきても，それが $X$ を数え尽くしていない証拠を与えることができる」という強い性質である．実数がこの強い非可算性を満たすことを（$\mathsf{EL}_0$ などの）標準的な直観主義解析学の体系で証明できる．

また，この計算論的な対応物としては，「可算でない」というのは，「実数を自然数で数え尽くす計算可能なアルゴリズムは存在しない」というものであり，「構成的可算である」というのは，「実数を自然数で数え尽くそうとするどんな方法を持ってきても，それが実数を数え尽くしていない証拠を具体的に計算するアルゴリズムを与えることができる」というものである．実際，通常の対角線論法などはそのようなアルゴリズムを与えている．

**定理 3.1.** 与えられた関数 $f : \mathbb{N} \to \mathbb{R}$ に対して，$x \notin \{f(n) \mid n \in \mathbb{N}\}$ となるような実数 $x \in \mathbb{R}$ を計算するアルゴリズムが存在する．

**■単射による非可算性:** つづいて，単射による非可算性である．実はこちらの方が少し難しい．こちらも（全射的）非可算性と同様に，単に「可算でない」というものと「構成的に非可算」という 2 種類が考えられる．つまり，

$$X \text{ が（単射的）可算でない} \iff \neg[(\exists f : X \to \mathbb{N})(\forall a, b \in X)\ [f(a) = f(b) \ \to\ a = b]].$$
$$X \text{ が構成的に非（単射的）可算} \iff (\forall f : X \to \mathbb{N})(\exists a, b \in X)\ [f(a) = f(b) \ \wedge\ a \neq b].$$

つまり，「（単射的）可算でない」というのは，あくまで「$X$ を重複なく命名することはできない」ことを述べているだけであるが，「構成的に非（単射的）可算」というのは，「どんな命名方法を持ってきても，同名の別人が 2 人いることを具体的に指摘できる」という強い性質である．$\mathbb{R}$ の非可算性が構成的に成り立つ，ということは，単に単射 $f : \mathbb{R} \to \mathbb{N}$ が存在しない，というだけでなく，任意の命名 $f : \mathbb{R} \to \mathbb{N}$ に対して，名前の重複があると指摘してやればよい．

実数の単射的非可算性証明の難しい部分としては，初等解析学における通常の実数の非可算性証明はいずれも実数の（構成的な）全射的非可算性を証明しており，単射的非可算性を直接的に証明しているものがほとんどないという点にあるかもしれない．しかし，計算論的な実数の構成的な非（単射的）可算性ならば，次のようにして示すことができる．

**定理 3.2.** 与えられた計算可能関数 $f : \mathbb{R} \to \mathbb{N}$ に対して，$f(a) = f(b)$ かつ $a \neq b$ なる実数 $a, b \in \mathbb{R}$ を計算するアルゴリズムが存在する．

*Proof.* 計算可能な関数 $f : \mathbb{R} \to \mathbb{N}$ があったとしよう．つまり，次の任意精度近似ゲームに対して，E は計算可能な必勝戦略を持つ．

| | 1 手目　2 手目　3 手目 | ルール | 勝利条件 |
|---|---|---|---|
| A | $x_0, \varepsilon$　　$x$ | $\lvert x - x_0\rvert < \delta$ | $\lvert f(x) - f(x_0)\rvert \geqslant \varepsilon$ |
| E | $\delta$ | | $\lvert f(x) - f(x_0)\rvert < \varepsilon$ |

たとえば $x_0 = 0$ とし，$\varepsilon = 1$ としよう．E は必勝戦略に従って，2 手目で $\delta > 0$ を出してくる．これに対して，我々は $x = \frac{\delta}{2}$ を選ぶのみである．このとき，$x_0 \neq x$ かつ $f(x_0) = f(x)$ である．なぜなら，E が勝つので，$|f(x) - f(x_0)| < \varepsilon = 1$ であるが，$f(x), f(x_0) \in \mathbb{N}$ であるということから，$|f(x) - f(x_0)| < 1$ は $f(x) = f(x_0)$ を導く． □

**■位相空間論的解釈:** 計算論的な実数の単射的非可算の証明は，まさに連続単射 $f : \mathbb{R} \to \mathbb{N}$ が存在しないことを示していることに近い．もう少し細かくみれば，関数 $f \in \mathcal{C}(\mathbb{R}, \mathbb{N})$ に対して $a \neq b$ かつ $f(a) = f(b)$ なる実数の対 $(a, b) \in \mathbb{R}^2$ を返す多価関数が連続であることを示していると思ってもよい．

**豆知識．**ところで，定理 3.2 のような計算論的単射的非可算性証明の微妙な点は，**連続**な単射 $f : \mathbb{R} \to \mathbb{N}$ のみを議論する点であり，通常の単射的非可算性を証明するためには，もちろん連続でない関数も考慮に入れる必要がある．つまり，計算論的な単射的非可算性証明は，本当の単射的非可算的証明とは質が異なる．

これを構成的立場から見直そう．もちろん普通の古典的数学では実数は単射的非可算であるし，ロシア学派の構成的数学のような数学でも実数は単射的非可算である．しかし，古典的でもなく計算論的でもない中途半端な直観主義的世界では，もしかしたら実数は単射的可算であるかもしれない．

実際，2010 年代になって Andrej Bauer が示したことは「無限時間チューリング機械と呼ばれる奇妙な計算モデルによる実現可能性トポス (realizability topos) では，実数の単射的可算性が成り立つ」という衝撃的な結果であった．ただし，Bauer の証明で無限時間チューリング機械を用いる必然性はなく，1960 年〜80 年代頃に一般再帰理論において活発に研究されていた無限的計算モデルのほとんど（たとえば任意の Spector pointclass）で証明を代替できる．その手の一般再帰理論的計算モデルが部分組合せ代数となることは自明に分かるから，自動的に実現可能性トポスを得られるので，同様の証明が通用する．