

プログラムの自動合成と定義可能性について

京産大 理 謝 章文

1. Theory における Definition の考察

ここでは object language を function symbols と equality symbol を含む first-order language とし、対象とする formal system のクラスは first-order theory (first-order calculus, 単に theory とよぶ) とする。すなわち、theory はその conservative subtheory として applied predicate calculus with equality を含む。

theory の rules は主として formal deduction の手順、すなわち proofs や derivations の構成を決める。theory は definitions を含む。definition の目的は primitive symbols と既に定義済みの symbols のもとで、新しい symbol を導入することである。

definition は (I) defining formulas とよばれる formula の形か、(II) 単に definition とよばれる defining rule の形をとる。theory T の defining formula は T の additional axiom とみなされ、 T に対する defining rule は T の additional rule of inference とみなされる。

defining formula は $\mathcal{U}_1 \leftrightarrow \mathcal{U}_2$ または $\mathcal{U}_1 = \mathcal{U}_2$ の形をとり、
 defining rule は ' $\mathcal{U}_1$ for ' $\mathcal{U}_2$ ' または $\mathcal{U}_1 =_{\text{df}} \mathcal{U}_2$ の形をとる。 $\mathcal{U}_1$
 は definiendum とよばれ、定義される symbol を含む。 $\mathcal{U}_2$ は definien
 とよばれ、primitive symbols, 既に定義済みの symbols およびここ
 で定義される symbol だけを含みうる。

definition はつぎの3つの性質のうち direct C-interchangeability
 を保証するために適当な制約をみたさなければならない。

(1) 新しい symbol の introducing と eliminating に対する両方向への
 translatability. (2) original theory が C-consistent ならば、
 その definition を含む theory の C-consistency. (3) primitive symbols
 が interpret されたならば、defined symbol の unique interpretation.

2. Function Symbol の Definability

theory T の n -place function symbol を f とする ($n \geq 0$)。 f 以外
 の T の nonlogical symbols の集合を Q 、 f および Q に属さない残
 りの nonlogical symbols の集合を R とする。また、 T の axioms の
 closures のある finite conjunction を $E[f, Q, R]$ と表わす。
 brackets 内はその formula に出現する nonlogical symbols を示す。

$E[f, Q, R]$ に対して

$$E[f, Q, R], E[f', Q, R'] \vdash f_{x_1 \dots x_n} = f'_{x_1 \dots x_n}.$$

ならば、 f は T において (R を auxiliary symbols として) Q が

ら implicitly definable であるという。

$E[f, Q, R]$ とある formula $A[Q]$ に対して,

$$E[f, Q, R] \vdash y = f_{x_1, \dots, x_n} \leftrightarrow A(x_1, \dots, x_n, y)$$

ならば, f は T において Q から explicitly definable であるという。また, $E[f, Q, R]$ は Q から f を explicitly に $A[Q]$ として定義するという。

f が T において Q から explicitly definable であるならば, implicitly definable であることは明らかである (Padoa's method の正当性)。さらに Beth の定理より, f が T において Q から implicitly definable であるならば, f はまた T において Q から explicitly definable である。

3. Extensions by Definitions

theory T , 相異なる変数 $x_1, \dots, x_n, y, y'$, そして $x_1, \dots, x_n, y$ 以外のいかなる変数も free でない T の formula を D とする。

$$\vdash_T \exists y D(x_1, \dots, x_n, y) \quad (1)$$

$$\vdash_T D(x_1, \dots, x_n, y) \wedge D(x_1, \dots, x_n, y') \supset y = y' \quad (2)$$

であるとき, T に新しい n -place function symbol f とつぎの新しい nonlogical axiom:

$$y = f_{x_1, \dots, x_n} \leftrightarrow D(x_1, \dots, x_n, y)$$

$$\text{または} \quad D(x_1, \dots, x_n, f_{x_1, \dots, x_n}) \quad (3)$$

をつけ加えて theory T' を作る。(1) 式を f に対する existence condition, (2) 式を f に対する uniqueness condition, (3) 式を f の defining axiom とよぶ。

ここで導入された f とその axiom は eliminable である。明らかに、 T の nonlogical symbols の集合 Q が存在して、 f は T' において Q から implicitly definable である。 T' が T からこの型の有限回の extensions によって得られるならば、 T' を T の 1 つの extension by definitions という。 T' は T の conservative extension であり、 T が consistent であるとき、およびそのときのみ T' は consistent である。

4. Program Synthesis と Definability

m 個の相異なる function symbols の並び $(f_1, f_2, \dots, f_m)$ を program という ($m \geq 1$)。各 function symbol f_i を n_i -place, $n = \max \{n_i | i=1, \dots, m\}$ とするとき、program は n 入力 m 出力であるという。

formulas の集合 Γ , 相異なる変数 $x_1, \dots, x_n, y_1, \dots, y_m$, そしてこれら以外のいかなる変数も free でない formula を D , Γ および D の nonlogical symbols のある集合を Q とするとき、つぎの条件がみたされるならば、 $(\Gamma, \exists y_1 \dots \exists y_m D, Q)$ を $\text{program}(f_1, \dots, f_m)$ の specification であるという。

(1) Γ を nonlogical axioms とする theory T は consistent である。

(2) D および Q は $f_1, \dots, f_m$ を含まない。

$$(3) \quad \vdash_T \exists X_1 \dots \exists X_n \exists Y_1 \dots \exists Y_m D$$

$$(4) \quad X_i = f_i X_1 \dots X_{n_i} \wedge \dots \wedge Y_m = f_m X_1 \dots X_{n_m} \supset D(X_1, \dots, X_n, Y_1, \dots, Y_m) \quad \text{または} \\ D(X_1, \dots, X_n, f_1 X_1 \dots X_{n_1}, \dots, f_m X_1 \dots X_{n_m}) \quad (4)$$

を新しい axiom として T につけ加えて consistent な theory T' が構成できる。

(4) 式を program $(f_1, \dots, f_m)$ の specifying axiom とよぶ。

theory T' のある formula $A[Q]$ に対して

$$\vdash_{T'} X_i = f_i X_1 \dots X_{n_i} \wedge \dots \wedge Y_m = f_m X_1 \dots X_{n_m} \leftrightarrow A(X_1, \dots, X_n, Y_1, \dots, Y_m) \quad (5)$$

ならば、defining formula (5) 式 またはこれと等価な defining rule による definition: $(f_1 X_1 \dots X_{n_1}, \dots, f_m X_1 \dots X_{n_m}) =_{df} \lambda$ を $(f_1, \dots, f_m)$ の T' における description とよぶ。

program の defining formula を $F[f_1, \dots, f_m, Q]$ とするとき、 F を新しい axiom として T につけ加えて consistent な theory が構成でき、かつ

$$F[f_1, \dots, f_m, Q] \vdash_T D(X_1, \dots, X_n, f_1 X_1 \dots X_{n_1}, \dots, f_m X_1 \dots X_{n_m})$$

ならば、program は $(\Gamma, \exists X_1 \dots \exists X_n D, Q)$ に対して正当であるという。

program synthesis とは specification から それに対して正当である program の defining formula または defining rule による definition を構成することである。

specification が つぎの total existence condition および uniqueness

condition をみたすとき、

$$\vdash_T \exists y_1 \dots \exists y_m D(x_1, \dots, x_n, y_1, \dots, y_m)$$

$$\vdash_T D(x_1, \dots, x_n, y_1, \dots, y_m) \wedge D(x_1, \dots, x_n, y'_1, \dots, y'_m) \supset y_1 = y'_1 \wedge \dots \wedge y_m = y'_m$$

$(f_1, \dots, f_m)$ は T' において T の nonlogical symbols のある集合 S から implicitly definable である。 $S \subseteq Q$ なる S が存在するとき、 T' は S から $(f_1, \dots, f_m)$ を explicitly に T' における description として定義する。 そうでないときは T' における description は存在しない。

total existence condition が成立しない場合は、 T' は S のもとで $(f_1, \dots, f_m)$ の partial implicit definition を与える。 したがって、 T' をみたす $(f_1, \dots, f_m)$ の解すなわち正当な program は存在すれば T' において限定されない部分では任意となる。 このとき T' における description は、 存在すれば、 T' から derivable であることおよび A が $x_1, \dots, x_n, y_1, \dots, y_m$ 以外の変数を free に含まないことから、 T' で限定されない部分を含まない partial explicit definition となり正当である。

uniqueness condition が成立しない場合は、 T' は $(f_1, \dots, f_m)$ として許される範囲を implicit に define している。 したがって、 正当な program は implicit definition で許される範囲から任意に選ぶ unique にすることができる。 T' における description は、 存在すれば、 specifying axiom が defining axiom と異なること (existence condition および uniqueness condition がともに成立するときのみ、 2つの

defining axiomは等価である) から異なる description が複数個存在し、そのいづれもが正当である。

〔定理〕 与えられた specification に対して、 T' における description が存在するとき、それによって explicitly に define される program は正当である。 (証明略)

〔1, 2〕において著者が提案した resolution-refutation method に基づく Mechanical Program Synthesis の方法は、ここで定義した specification から T' における description が存在するときはそれを与える能力を有する。さらに $S \subseteq Q$ なる S が存在しないときは、 $Q \subset S$ なる適当な S を求め、 $(\Gamma, \exists y_1 \dots \exists y_m D, S)$ のもとで正当である T' における description を与える。また total existence condition が成立しないときは、 T' が限定する部分だけの description を求め、uniqueness condition が成立しないときはその定義されている範囲を多価関数の形で与えることができる。

また〔2〕において合成プログラムの正当性に関する定理のプログラムの不動点理論に基づく証明を与えている。

参考文献

- [1] 謝 (1975): Resolution-Refutation法による M.P.S., 信学会AL 74-40
- [2] 謝 (1975): プログラム・シセシスのための情報抽出系, 信学会AL 75-13
- [3] 謝 (1975): Mechanical Flowchart Synthesis について, 信学会AL 75-2
- [4] Shoenfield, J.R. (1967): Mathematical Logic, Addison-Wesley
- [5] Kleene, S.C. (1967): Introduction to Metamathematics, N.H.
- [6] Kleene, S.C. (1967): Mathematical Logic, JOHN WILEY & SONS
- [7] Tarski, A. (1969): Logic, Semantics, Meta-Mathematics, Oxford
- [8] Carnap, R. (1942): Introduction to semantics, Cambridge, Mass.,
- [9] Carnap, R. (1943): Formalization of Logic, Cambridge, Mass.,
- [10] Manna, Z. (1974): Mathematical Theory of Computation, McGraw-Hill
- [11] 伊藤貴康 (1975): プログラム理論, コロナ社