SBORNÍK PRACÍ FILOSOFICKÉ FAKULTY BRNĚNSKÉ UNIVERSITY 1967, B 14

PAVEL MATERNA

SET OF METHODS AND SET OF PROBLEMS

The term „method" may be explicated in two ways: the explication is either dependent on the term "problem" or it is not dependent on it.

We are going to give here independent explication, which will enable us to find some relationships between the set of methods and the set of problems. No proofs are given in this sketch.

Definition 1: *Method M* is a set of instructions determining operations that correlate to each element a_i of a class A_M ("input data") a certain element b_i of a class B_M ("output data").

Definition 2: A. *Problem P* is given by the pair $\{A_P, B_P\}$ and by the task to correlate to each element a_i of the class A_P a certain element b_i of the class B_P.

B. a) *Problem P* is *singular* just when A_P contains a single element.

b) *Problem P* is *general* just when A_P contains more than one element.

c) *Problem P* is *decision problem* just when B_P is the set $\{1, 0\}$ the elements of which we correlate to the elements a_i according to whether $F(a_i)$ is valid or not, where F stands for a predicate constant applicable to objects a_i.

Definition 3: *Method M solves* problem P just when A_P is identical with a subclass A'_M of the class A_M, B_P is identical with a subclass B'_M of the class B_M, and the classes A'_M and B'_M determine some method M' in the sense of Definition 1. (consequently, we have $A'_M = A_{M'}$ and $B'_M = B_{M'}$),and each element $f(a_i)$ generated by the application of M' to a_i^M, $a_i^M \in A'_M$, is identical with the element b_i, $b_i \in B'_P$, which is to be correlated to the element a_i^P, $a_i^P \in A_P$, $a_i^P = a_i^M$.

Note: Method M', of course, also solves problem P, since it contains itself as a subclass.

Example for Definition 3: Method M, which solves problem P_1 of recognizing well formed formulae of the predicate calculus [Church § 30] also solves problem P_2 of recognizing well formed formulae of the propositional calculus: A'_M will be the class of combinations of propositional variables and signs $\sim$, $\supset$, [,], (which is subclass of A_M), B'_M will be the set $\{1,0\}$ (which is improper subclass of B_M), and there exists a set of instructions for operations transforming A'_M into B'_M according to Def. 1. i. e. method M' (derivable from [Church § 20]). Problem P_2 is thus solved both by method M and method M'.

Definition 4: Problem P is *solvable* just when there exists a method that solves problem P.

Agreement: We exclude from our considerations cases in which transfinite sets would be elements of B_M and B_P.

Statement 1 a) A decision problem is solvable just when the respective predicate F is general recursive.

Argumentation: If F is general recursive, then the characteristic function f of this predicate defined on the class A_P is general recursive too. Consequently there exists such a method M that $A_M = A_P$ and M satisfies Def. 1.

Conversely: If there exists a method M satisfying Def. 3, then the application of this method to objects a_i is equivalent (vide Definition 2 B c) to the computation of the characteristic function f of predicate F. On the basis of our Agreement, the existence of such a method warrants the intuitive computability of function f, and consequently — provided that Church's thesis is valid — its general recursiveness.

Statement 1 b) The problem is solvable just when for $a_i \in A_P$, $x_i \in B_P$ the identity $f(a_i) = x_i$ is valid, where f is a general recursive function.

It is obvious that Statement 1 a) follows from Statement 1 b) as a special case.

Statement 2: A problem may be correlated to each function.

To each n-argument function f_j^n, j, $n = 1,2\ldots$ we correlate a class $A_{P_j} = \{\langle z_1, \ldots, z_n \rangle_i\}$ of all n-tuples on the basis of which f_j^n has been defined, and the class $B_{P_j} = \{x_i\}$, where x_i is the value of function f_j^n for the i-th n- tuple of arguments.

Considering the set of all functions. Let $\mathscr{F}_{1R}$ be the set of definable general recursive functions, $\mathscr{F}_{1\overline{R}}$ the set of the other definable functions, $\mathscr{F}_1$ the set of definable functions, and $\mathscr{F}_2$ the set of undefinable functions. Consequently the set of all functions $\mathscr{F} = \mathscr{F}_1 \cup \mathscr{F}_2 = \mathscr{F}_{1R} \cup \mathscr{F}_{1\overline{R}} \cup \mathscr{F}_2$ and not a single on of the mentioned sets is empty.

(The nonemptiness of set $\mathscr{F}_2$ is the consequence of the existence of undefinable relations — vide e. g. [Grzegorczyk] —, but thinkable is also such a conception of existence in which this set may be considered to be empty. This circumstance will not influence our further interpretation — vide *Considering the set of all problems*). The nondenumerability of set $\mathscr{F}$ and $\mathscr{F}_2$, as well as the denumerability of sets $\mathscr{F}_1$, $\mathscr{F}_{1R}$ and $\mathscr{F}_{1\overline{R}}$ is evident.

Considering the set of all problems. This consideration is the consequence of the preceding *Consideration* and of Statement 2. Let $\mathscr{P}_{1S}$ be the set of definable solvable problems (of element 0 of Medvedev's set Ω—[Medvedev]), $\mathscr{P}_{1\overline{S}}$ the set of unsolvable definable problems, $\mathscr{P}_1$ the set of definable problems, and $\mathscr{P}_2$ the set of undefinable problems. Consequently the set of all problems $\mathscr{P} = \mathscr{P}_1 \cup \mathscr{P}_2 = \mathscr{P}_{1S} \cup \mathscr{P}_{1\overline{S}} \cup \mathscr{P}_2$ and none of the mentioned sets is empty.

(As far as the nonemptiness of set $\mathscr{P}_2$ is concerned, we shall not take this set into consideration any longer having obtained information from [Wittgenstein §§ 5,6 and 7].

Theoretically we might further consider some set of "pseudoproblems" for which $\sim (\exists f) . f(a_i) = x_i$ is valid. We shall not consider this set either (just as Medvedev).

Relationship between the set of problems $\mathscr{P}_1$ and the set of methods $\mathscr{M}$

Note: By agreement, we have excluded from the concept of the method such sets of instructions as "add a stroke to the given term," the application of which generates (potentionally) an infinite series of strokes.

Statement 3:. Each method solves at least one problem of set P_{1S}.

Statement 3 follows from the given definitions.

Definition 5: Method M_i is *equivalent* to method M_j ($M_i \equiv {}_M M_j$), just when $A_{M_i} = A_{M_j}$ and $f_{M_i}(a_i) = f_{M_j}(a_i)$.

Note: The existence of pairs of equivalent methods can be empirically verified.

Statement 4. The relation $\equiv_M$ is reflexive, symmetrical and transitive in set M.

Statement 4 is evident.

From Statement 4 follows the possibility of defining equivalence classes of methods: let $|M_i|$ be the class of methods equivalent to method M_i.

Statement 5. $M_i \equiv {}_M M_j$ just in that case when the class of problems that are solved by method M_i is identical with the class of problems that are solved by method M_j.

Statement 5 follows from Definition 3 and Definition 5.

We are going to write now some results concerning the relationship of set M and set $\mathscr{P}_1$ in one of the standard ways of writing expressions of predicate logic (for the sake of brevity). $S(v, w)$ will stand for the relation "v solves w" in the sense of Definition 3.

Statement 6. The following relationships are valid between set $\mathscr{M}$ and set $\mathscr{P}_1$:

a) $(\exists x)(y)\,(x \in \mathscr{P}_1 \;\&\; y \in \mathscr{M} \supset \sim S(y, x))$;

b) $(x)\,(x \in \mathscr{M} \supset (\exists y)(y \in \mathscr{P}_{1S} \;\&\; S(x, y)))$; (Statement 3)

c) $(x)\,(x \in \mathscr{P}_{1S} \supset (\exists y)(y \in \mathscr{M} \;\&\; S(y, x)))$;

d) $(x)\,(x \in \mathscr{P}_{1\overline{S}} \supset \sim (\exists y)(y \in \mathscr{M} \;\&\; S(y, x)))$;

e) $\sim (x)\,\{x \in \mathscr{P}_{1S} \supset (\exists y)\,(y \in \mathscr{M} \;\&\; S(y, x) \;\&\; (z)\,[z \in \mathscr{M} \;\&\; S(z, x) \supset z = y)]\}$; (i. e. we cannot speak about mapping set $\mathscr{P}_{1S}$ into set $\mathscr{M}$).

f) $\sim (x)\,(x \in \mathscr{M} \supset (\exists y)\,\{y \in \mathscr{P}_{1S} \;\&\; S(x, y) \;\&\; (z)\,[z \in \mathscr{P}_{1S} \;\&\; S(x, z) \supset z = y)]\})$; (i. e. we cannot speak about mapping set $\mathscr{M}$ into set $\mathscr{P}_{1S}$).

g) $\sim (x)\,(x \in \mathscr{P}_{1S} \supset (\exists i)\,\{(y)\,(y \in |M_i| \supset S(y, x)) \;\&\; (j)\,(j \neq i \supset (z)\,(z \in |M_j| \supset \sim S(z, x)]\})$,

REFERENCES

A. Church: *Introduction to Mathematical Logic I.* Princeton University Press 1956.

A. Grzegorczyk: Konsekwencje teoriopoznawcze dwóch twierdzeń metamatematiky. *Studia filozoficzne 5* (42), 1965, 115—118.

U. T. Medvedev: Stépéni trudnosti massovyh problém. *Papers of the Ac. Sc. of the USSR* 104 (1955), 501—504 (russian).

L. Wittgenstein: *Tractatus Logico-philosophicus.* London 1955

Translated by Mikoláš Adler

MNOŽINA METOD A MNOŽINA PROBLÉMŮ

Článek informuje o některých jednoduchých vztazích mezi množinou metod a množinou problémů. Činí tak na základě definice pojmů: problém, metoda, řešit, a některých tvrzení vyplývajících z teorie rekurzívních funkcí. Za důležité, byť i velice elementární tvrzení pokládá autor zejména větu, že nelze zobrazit množinu řešitelných problémů do množiny metod, ani množinu metod do množiny řešitelných problémů.