Докл. Акад. Наук СССР
Том 285 (1985), № 2

Soviet Math. Dokl.
Vol. 32 (1985), No. 3

ON THE BASIC STRUCTURES OF THE DESCRIPTIVE THEORY OF ALGORITHMS

UDC 517.11

AN. A. MUCHNIK

The question has previously been raised (see for example [1] and [2]) as to what constitutes the object of study in the theory of algorithms, and what the basic methods of proof are in this theory. In this connection the phenomenon of relativization has often been discussed. In this note we try to give a partial answer to the above questions, and to furnish mathematical assertions which support the proposed answers. Here we will only deal with the descriptive, qualitative, theory of algorithms, leaving aside the metric, quantitative, theory of algorithms (complexity theory).

A number of authors have expressed the point of view which considers the theory of algorithms to be the theory of a single universal algorithm (see [2]). Theorem 1 below provides precise mathematical confirmation of this thesis.

As the consideration of partial functions is a central feature of the theory of algorithms, we will also consider structures (algebraic systems) with partial operations, without any special proviso. We will denote by f_x the function $\lambda y f(x,y)$, i.e. the unary function obtained by fixing the first variable in $f(x,y)$.

Let $F(x,y)$ be some primary (Gödelian) universal function (see [3]). Then the pair $\langle \mathbf{N}; F\rangle$ will be called a *computation structure.*

THEOREM 1. *All computation structures are isomorphic.*

We remark that in Theorem 1 it is of course unnecessary to require that the underlying set of the structure be the natural number sequence $\mathbf{N}$. Its role may be taken over by any "ensemble" of constructive objects (see [2]).

PROOF OF THEOREM 1. Let $\langle \mathbf{N}; F\rangle$ and $\langle \mathbf{N}; G\rangle$ be two computation structures. The theorem asserts the existence of a bijection $h\colon \mathbf{N} \to \mathbf{N}$ such that

$$F(x,y) = z \Leftrightarrow G(h(x), h(y)) = h(z).$$

In other words, for every x in $\mathbf{N}$, necessarily

$$F_x = h^{-1} \circ G_{h(x)} \circ h.$$

Suppose that we have an arbitrary computable bijection h. Then the function H for which $H_x = h^{-1} \circ G_x \circ h$ is a primary universal function. According to a theorem of Rogers on isomorphism of Gödel (primary) *numberings* (see [4]), there is a bijection k such that $F_x = H_{k(x)}$ for all x. In other words $F_x = h^{-1} \circ G_{k(x)} \circ h$. We will use the fixed point theorem to get k coinciding with h.

In order to make it actually possible to apply the fixed point theorem we will refine the construction occurring in Rogers' theorem in such a way that the bijection k is constructed for every computable function h (not necessarily bijective). If h is indeed a bijection then k will have the required properties.

Construction of k from h. It is easy to see that from h we can effectively construct a certain function i which turns out to be the inverse of h if h is bijective. As F and $G$are primary numberings, it is possible to find completely defined computable functions

1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 68Q15; Secondary 03D99.

u and v for which

$$F_{u(x)} = i \circ G_x \circ h, \qquad G_{v(x)} = h \circ F_x \circ i.$$

We construct a computable bijection k for which $k(m) = n$ implies that either $F_n = i \circ G_m \circ h$ or $G_m = h \circ F_n \circ i$ (if h is a bijection then i and h are mutual inverses, and the equalities are equivalent). As in Rogers' theorem the bijection k is constructed in stages. At even stages we define $k(m)$ for some m not yet contained in the domain of k. At odd stages we define $k^{-1}(n)$ for some n not yet contained in the range of k. Specifically, we take $k(m) = u(m)$ if $u(m)$ is not found among the values of the function k which have already been defined, and otherwise we find a value of n which does not occur among these values, and for which $F_n = F_{u(m)}$. This can be done, since for any index of a function in a primary numbering it is possible to effectively find arbitrarily many indices of the same function. In a similar fashion $k^{-1}(n)$ is defined as $v(n)$ or any other m for which $G_m = G_{v(m)}$ and for which k is not yet defined. Thus the bijection k is constructed.

If the initial function h is bijective, then $i = h^{-1}$ and $F_{k(m)} = h^{-1} \circ G_m \circ h$, as was required. Now an application of the fixed point theorem completes the proof of Theorem 1.

The question naturally arises: what will happen if in Theorem 1 we consider relative computability in place of computability? It turns out that all structures corresponding to a single oracle (or equivalent oracles) are isomorphic. In other words, Theorem 1 relativizes.

The definition of the notion of *A-computation structure* is obtained from the definition of a computation structure by replacing computability by computability with the oracle A.

It can be shown that for noncomputable A, the A-computation structures are neither isomorphic nor even elementarily equivalent to computation structures without oracles.

Let Φ be a property of structures of the signature under consideration (with a single binary partial function; we consider only properties which are invariant under structure isomorphism). Then in view of Theorem 1, for each oracle A, Φ is either true for all A-computation structures, or false for all such. Thus three possibilities arise:

1. Φ is true for A-computation structures for all A;
2. Φ is false for A-computation structures for all A;
3. Φ is true for some A-computation structures and false for others, with different A.

In cases (1) and (2) Φ is naturally called a *relativizably true* or *relativizably false* property, and in case (3) a *nonrelativizable* property. It turns out that the majority of natural assertions in the theory of algorithms refer to relativizable properties. We will now see that it is possible to distinguish the relativizably true propositions from the relativizably false ones without appealing to the notion of computability. For this we need a game, which we will now define:

The Game. Two players take part in the game: Nature (player N) and Mathematician (player M). Each player gradually constructs a structure of the form $\langle \mathbf{N}; F \rangle$, where F is a partial function from $\mathbf{N} \times \mathbf{N}$ to $\mathbf{N}$. (Here any countable set may be used in place of $\mathbf{N}$. The structure of $\mathbf{N}$ is not used.) At the start of the game $F = \varnothing$. At each stage the definition of the function is extended to finitely many points. Previously defined values of the function may not be altered. The players make their moves alternately. Thus the rules of a move are always the same, and the possible moves do not depend on the opponent's moves. The difference between concrete plays is determined by a payoff function (which, as we will see, corresponds to the property whose truth Mathematician wishes to establish). Let Φ be a fixed property of structures of the class under consideration. Then in the Φ-game, Nature wins if she has constructed a function ν, and Mathematician has constructed a function μ such that:

(WN1) Φ is false in $\langle \mathbf{N}, \nu \rangle$, and

(WN2) There is an integer e such that ν_e is completely defined, and, for all x, $\mu_x = \nu_{\nu_e(x)}$.

As we see, only condition (WN1) depends on the property Φ, but it does not depend on μ. Condition (WN2) means that the numbering of unary functions constructed by Mathematician is reduced to the numbering of unary functions constructed by Nature, where the reduction function is itself also constructed by Nature.

The following assertion makes it possible to distinguish the relativizably true propositions from the relativizably false ones in terms of the game described: *If Φ is relativizably true then there is a winning strategy for player M; if Φ is relativizably false,, then there is a winning strategy for player N.*

This assertion follows from the following theorem.

THEOREM 2. *The following equivalences hold, where B is an arbitrary subset of the natural numbers, and $B \leq_T A$ denotes decidability of B relative to A.*

(a) *Property Φ is true for all A-computation structures for which $B \leq_T A$ if and only if there is a winning strategy for M in the Φ-game which is computable relative to B.*

(b) *Property Φ is false in all A-computation structures for which $B \leq_T A$ if and only if there is a winning strategy for N in the Φ-game which is computable relative to B.*

The proof of Theorem 2 will be given for decidable B; its relativization to arbitrary B proceeds in the usual fashion. We consider the following computable strategy (which may be used by either M or N): "Watch your opponent's play. Construct for yourself a primary universal function for functions which are computable relative to your opponent's play (more precisely, relative to the function $n \mapsto$ the nth move by your opponent."

Following this strategy, N ensures the satisfaction of condition (WN2), since the function constructed by M will of course be computable realtive to M's actions. Thus the forward implication is valid in (b). If M uses a similar strategy then N, not wishing to violate condition (WN2), is obliged to construct a primary universal function for X-computable functions, where X is the play of N, viewed as an oracle (the X-computability of this function is evident, and condition (WN2) ensures that the numbering is primary, since M constructs a primary X-computable function). Hence the forward implication holds in (a).

The reverse implications are proved as follows.

(b) If Φ fails for a primary universal A-computable function, then M can beat any computable strategy for N, by constructing a primary universal A-computable function; since the strategy for N is computable, the function M constructs will be A-computable and primary for A-computable functions by condition (WN2) (more precisely, M must act so that his or her actions are also A-computable, which is evidently possible).

(a) Similarly, if for some A property Φ is satisfied then N beats any computable strategy for M by constructing a primary universal A-computable function.

Theorem 2 is proved.

In the corollary presented below the words "... for all sufficiently large A" signify "there is a subset B of $\mathbf{N}$ such that ... for all $A \subset \mathbf{N}$ for which $B \leq_T A$."

COROLLARY. *The following equivalences hold:*

1. *Property Φ holds for A-computation structures for all sufficiently large A if and only if M has a winning strategy in the Φ-game.*

2. *Property Φ is false for A-computation structures for all sufficiently large A if and only if N has a winning strategy in the Φ-game.*

We remark that from the above together with the theorem on determinacy of Borel games we may deduce the following result (Martin): if Φ is a Borel property then either Φ or its negation holds for all A-computation structures, for A sufficiently large.

Theorem 2 gives a scheme for the proof of the relativizable truth of a given property: it is necessary to construct a computable winning strategy for M in the Φ-game. An analysis of the usual proofs in the theory of algorithms shows that the majority of them are easily cast into this scheme, and the main portion of the proof is the description of a strategy and the verification that it wins; its computability is easily verified.

Scientific Council on the Complex Problem "Cybernetics"
Academy of Sciences of the USSR
Moscow

Received 17/DEC/84

BIBLIOGRAPHY

1. Hartley Rogers, Jr., *Theory of Recursive Functions and Effective Computability*, McGraw-Hill, 1966.

2. V. A. Uspenskiĭ and A. L. Semenov, *Algorithms in Modern Mathematics and Their Applications* (Proc. Internat. Sympos., Urgench, 1979; A. P. Ershov and D. E. Knuth, editors), Part I, Vychisl. Tsentr, Sibirsk. Otdel. Akad. Nauk SSSR, Novosibirsk, 1982, pp. 99–342; English version, Algorithms in Modern Mathematics and Computer Science (Proc. Internat. Sympos., Urgench, 1979; A. P. Ershov and D. E. Knuth, editors), Lecture Notes in Computer Sci., vol. 122, Springer-Verlag, 1981, pp. 100–234.

3. V. A. Uspenskiĭ, *Lectures on computable functions*, Fizmatgiz, Moscow, 1960; French transl., Actualités Sci. Indust., no. 1317, Hermann, Paris, 1966.

4. Hartley Rogers, Jr., J. Symbolic Logic **23** (1958), 331–341.

Translated by G. L. CHERLIN