CONSTRUCTIVE GENERALIZED FUNCTIONS

MATHEMATICS

1967

SovietRxiv

Abstract

Full Text

UDC 517.11

MATHEMATICS

FAN DINH ZIEU

CONSTRUCTIVE GENERALIZED FUNCTIONS

(Presented by Academician P. S. Novikov on 22 X 1966)

1. The present note is devoted to defining the concept of a generalized function in constructive mathematics. First of all we introduce some concepts and notation*.

By a **word of type** ^{k} we shall mean the complete code of a uniformly continuous function on the interval $-k\Delta k$, where k is an arbitrary natural number greater than 0. By C_k we shall denote the normed space of words of type ^{k}. Let us note that if $k \leq l$, then every word of type ^{l} is a word of type ^{k}. The equality relation, the addition algorithm, and the algorithm of multiplication by real duplexes in the space C_k will be denoted respectively by

$$\underset{,k}{=}, \quad \underset{,k}{+}, \quad \underset{,k}{\cdot}.$$

Let f be an everywhere defined constructive function. We shall say that f is an **almost uniformly continuous function** if there is an algorithm g of type $(\to)$ such that, for every $k \geq 1$, g_k is a regulator of uniform continuity of the function f on the interval $-k\Delta k$.

The algorithm g is called a **regulator of almost uniform continuity** of the function f. The word $\xi f \overline{1} \xi g$ will be called the **complete code of the almost uniformly continuous function**, and also a word of type . One can define an equality relation , an addition algorithm $+$, and an algorithm of multiplication by real duplexes $\cdot$ in the usual way, so that the set of words of type forms a constructive linear space. We denote this space by $\mathcal{C}$. It is not difficult to construct an algorithm ξ having the following property: for every $k \geq 1$, $\tilde{\xi}_k$ transforms any word of the form $\xi f \overline{1} \xi g$ into the word $\xi f \overline{1} \xi \tilde{g}_k$. Obviously, ξ_k transforms every word of type into a word of type ^{k}. Let $_1$ and $_2$ be words of type . Instead of the formula

$$\xi(k \square_1) \underset{,k}{=} \xi(k \square_2)$$

we shall simply write

$$_1 \underset{,k}{=} {_2}.$$

We can construct an algorithm J_1, which, when applied to any word of the form $k \square x \square y \square {}_1^k$, where k is a natural number ≥ 1, x and y are real duplexes, $-k \leq x \leq y \leq k$, and ${}_1^k$ is a word of type ^{k}, transforms every such word into the integral of the complete code ${}_1^k$ in the space $\mathfrak{L}_1(x, y)$ (see (^{2}), § 15.5). Then we construct an algorithm J such that, for every $k \geq 1$, J_k is an algorithm of type $({}^k \to {}^k)$ and such that, for any number $k \geq 1$, any word ${}_1^k$ of type ^{k}, and any duplex x, $-k \leq x \leq k$, the following holds:

$$\underbrace{J(k \square {}_1^k)}_{,k}(x) \simeq J_1(k \square \min(x \square 0) \square x \square {}_1^k) - J_1(k \square \min(x \square 0) \square 0 \square {}_1^k).$$

* All specifically unexplained terms and notation are understood in the same way as in ($^{1-3}$).

The number $J(k\square\mathfrak{sh}_1^k)(x)$ is called the **integral** of the complete cipher $\mathfrak{sh}_1^k$ from 0 to x, and $J(k\square\mathfrak{sh}_1^k)$ is called the **primitive** of the complete cipher $\mathfrak{sh}^k$ on the segment $-k\Delta k$. Now we can construct an algorithm I such that, for every $k \geq 1$, I_k is an algorithm of type $(\mathfrak{nsh}^k \to \mathfrak{sh}^k)$ and the following scheme is satisfied:

$$I(k\square 0\square\mathfrak{sh}_1^k) \simeq \mathfrak{sh}_1^k; \qquad I(k\square n+1\square\mathfrak{sh}_1^k) \simeq J(k\square I(k\square n\square\mathfrak{sh}_1^k)).$$

$I(k\square n\square\mathfrak{sh}_1^k)$ is called the **primitive of order** n of the complete cipher $\mathfrak{sh}_1^k$ on the segment $-k\Delta k$.

After this one can construct an algorithm $\mathfrak{J}$ of type $(\mathfrak{nsh} \to \mathfrak{sh})$, satisfying the following condition for any numbers k, n $(k \geq 1)$ and any word $\mathfrak{sh}_1$ of type $\mathfrak{sh}$:

$$\xi(k\square\mathfrak{J}(n\square\mathfrak{sh}_1)) = I(k\square n\square\xi(k\square\mathfrak{sh}_1)).$$

$\mathfrak{J}(n\square\mathfrak{sh}_1)$ is called the **primitive of order** n of the complete cipher $\mathfrak{sh}_1$.

We shall call every σ-system of real duplexes a **polynomial germ**, and also a word of type $\mathfrak{yu}$ (^{2}). Let L denote an algorithm computing the number of σ-terms in a word of type $\mathfrak{yu}$, and let G denote an algorithm of type $(\mathfrak{yu}\text{—}\mathfrak{sh})$ having the following property: whatever the word $\mathfrak{yu}_1$,

$$\forall_x (G(\mathfrak{yu}_1)(x) \simeq a_1 + a_2 \cdot x + \cdots + a_n \cdot x^{n-1}),$$

where $n \simeq L(\mathfrak{yu}_1)$, and a_i is the i-th σ-term of the word $\mathfrak{yu}_1$ $(i = 1, \dots, n)$. We note that all judgments constructed with the aid of the indicated concepts and notation can be formulated rigorously within the framework of the logico-mathematical languages of N. A. Shanin (^{1}).

2. We now define the concept of a constructive generalized function. In order to define this concept, we shall rely on the classical definition of Mikusinski–Sikorski–Korevaar (^{7,8}).

Of an algorithm λ of type $(\mathfrak{n}\text{—}\mathfrak{sh})$ we shall say that it is a **fundamental sequence of words of type $\mathfrak{sh}$** if potentially realizable algorithms N of type $(\mathfrak{n} \to \mathfrak{n})$, P of type $(\mathfrak{nn} \to \mathfrak{yu})$, and Q of type $(\mathfrak{nn} \to \mathfrak{n})$ exist, satisfying the following conditions:

$$1) \quad \forall kn(L(P(k\Box n)) < N(k))^{*};$$

$$2) \quad \forall kijmx(|x| \leq k \ \& \ i, j \geq Q(k\Box m) \supset |F(k\Box i)(x) - F(k\Box j)(x)| < \bar{2}^{m}).$$

where F is an algorithm of type $(\mathfrak{nn} \to \mathfrak{sh})$, defined as follows:

$$\forall kn(F(k\Box n) \simeq J(N(k)\Box\lambda(n)) + G(P(k\Box n))).$$

Of a word X in the alphabet $\mathfrak{Ch}_0$ we shall say that it is an **F-generalized function** if the algorithm $\langle X \rangle$ is a fundamental sequence of words of type $\mathfrak{sh}$.

Of a word A in the alphabet $\mathfrak{Ch}_0 \cup \{t\}$ we shall say that it is an **FR-generalized function** if it has the form $XtYtZtT$, where X is a record of some algorithm λ of type $(\mathfrak{n} \to \mathfrak{sh})$, Y is a record of some algorithm N of type $(\mathfrak{n} \to \mathfrak{n})$, Z is a record of some algorithm P of type $(\mathfrak{nn} \to \mathfrak{yu})$, T is a record of some algorithm Q of type $(\mathfrak{nn} \to \mathfrak{n})$, and conditions 1)–2) are satisfied.

An FR-generalized function we shall also call a **constructive generalized function** or a word of type Φ. It is easy to see that the set of constructive generalized functions can be defined by a normal formula.

[*] The notation $<$ is understood in the following way:

$$(L(\mathfrak{yu}_1) < n) \Leftrightarrow (n \geq 1 \supset L(\mathfrak{yu}_1 \leq n) \ \& \ (n = 0 \supset \mathfrak{yu}_1 = 0)),$$

where $\mathfrak{yu}_1$ is a word of type $\mathfrak{yu}$ and n is a natural number.

Let $\Phi_1 \rightleftarrows \xi\lambda\mathcal{E}N\mathcal{E}P\mathcal{E}Q\mathcal{E}$ be a constructive generalized function. We agree to denote:

$$|\Phi_1|_k \rightleftarrows N(k); \qquad [\Phi_1]_{k,n} \rightleftarrows P(k\Box n).$$

According to condition 2), the algorithm $(\xi_k \circ \bar{F}_{k\Box})$ determines a sequence of words of type ^{k} converging in itself, with convergence regulator $Q_{k\Box}$. One can construct an algorithm that transforms each word of the form $k\Box\Phi_1$ into the limit of the sequence $(\xi_{k\Box} \circ \bar{F}_{k\Box})$ in the space C_k. We denote this limit by $\{\Phi_1\}_k$. This is a word of type ^{k}.

Let two constructive generalized functions Φ_1 and Φ_2 be given. Denote

$$p_k \rightleftarrows |\Phi_1|_k, \qquad q_k \rightleftarrows |\Phi_2|_k, \qquad l_k \rightleftarrows \max(p_k\Box q_k).$$

We shall say that Φ_1 is equal to Φ_2 and shall write $\Phi_1 = \Phi_2$, if there is an algorithm $\mathcal{P}$ of type ($\to$) satisfying the conditions:

1) $\forall k(\bar{L}(\mathcal{P}(k)) < l_k)$;
2) $\forall k(I(k\Box l_k - p_k\Box\{\Phi_1\}) = I(k\Box l_k - q_k\Box\{\Phi_2\}_k) + \xi(k\Box G(\mathcal{P}(k))))$.

Theorem 1. *There is an algorithm that transforms each pair of equal constructive generalized functions Φ_1 and Φ_2 into a notation of some algorithm $\mathcal{P}$ of type ($\to$) satisfying conditions 1)–2).*

It follows from Theorem 1 that the relation of equality between constructive generalized functions can be defined by a normal formula.

Let $\Phi_1 \rightleftarrows X\tau Y\tau Z\tau T$ be a constructive generalized function. The word X will be called the **basis** of the generalized function Φ_1. We shall say that the constructive generalized function Φ_1 is equal to the F-generalized function X, if it is equal to some constructive generalized function whose basis is X.

Theorem 2. *a) For every F-generalized function one can construct an equal constructive generalized function; b) there is no algorithm that transforms every F-generalized function into an equal constructive generalized function.*

It is not difficult to define addition of constructive generalized functions, multiplication of constructive generalized functions by real duplexes, and the zero constructive generalized function in such a way that the set of constructive generalized functions forms a linear space (see the definition in (6)).

3. We now define the derivative of a constructive generalized function.

Let $\Phi_1 \rightleftarrows X\tau Y\tau Z\tau T$ be a constructive generalized function. Construct algorithms γ and δ of type ($\to$), such that for any natural numbers k, n:

$$\gamma(n) \simeq \langle X\rangle(n)(n\Box n); \qquad \delta(k) \simeq \mu j\left(2^j > \frac{4k^{|\Phi_1|_k}}{|\Phi_1|_{k-1}!}\right).$$

After this, construct algorithms λ of type ($\to$), N of type ($\to$), P of type ($\to$), and Q of type ($\to$), satisfying the following conditions for any natural numbers k, n, m and any duplex x:

$$\lambda(n)(x) \simeq 2^{\gamma(n)}(\langle X\rangle(n)(x + 2^{-\gamma(n)}) - \langle X\rangle(n)(x)); \qquad N(k) \simeq |\Phi_1|_k + 1;$$

$$P(k\Box n) \simeq [\Phi_1]_{k,n} + \underbrace{0\sigma\ldots\sigma 0\sigma}_{|\Phi_1|_k \text{ times}}\frac{\langle X\rangle(n)(0)}{|\Phi_1|_k!};$$

$$Q(k\Box m) \simeq \max(m + 1 + \delta(k)\Box k + 1\Box\langle T\rangle(k\Box m + 1)).$$

Put

$$X_1 \rightleftarrows \mathcal{E}\lambda\mathcal{E}, \qquad Y_1 \rightleftarrows \mathcal{E}N\mathcal{E}, \qquad Z_1 \rightleftarrows \mathcal{E}P\mathcal{E}, \qquad T_1 \rightleftarrows \mathcal{E}Q\mathcal{E}.$$

It is easy to prove that the word $X_1 \tau Y_1 \tau Z_1 \tau T_1$ is a constructive generalized function. We shall call this generalized function the **derivative** of the constructive generalized function Φ_1. One can construct an algorithm D,

constructing, for each constructive generalized function, its derivative. After this one can construct an algorithm $\mathfrak{D}$ of type $(\mathfrak{nf} \to \Phi)$ satisfying the scheme:

$$\mathfrak{D}(n \square \Phi_1) \simeq \Phi_1; \qquad \mathfrak{D}(n+1 \square \Phi_1) \simeq D(\mathfrak{D}(n \square \Phi_1)).$$

$\mathfrak{D}(n \square \Phi_1)$ is called the derivative of n-th order of Φ_1. Thus every constructive generalized function has derivatives of arbitrary order.

4. Denote by R an algorithm of type $(\mathfrak{m} \to \Phi)$ which transforms every word $\mathfrak{m}_1$ of type $\mathfrak{m}$ into a word $XtYtZtT$ of type Φ such that

$$\forall_n (\langle X \rangle (n) \doteq \mathfrak{m}_1); \qquad \forall_k (\langle Y \rangle (k) \doteq 0);$$

$$\forall_{kn} (\langle Z \rangle (k \square n) \doteq 0); \qquad \forall_{km} (\langle T \rangle (k \square m) \doteq 0).$$

We shall say that $R(\mathfrak{m}_1)$ is a representative of the complete cipher of an almost uniformly continuous function $\mathfrak{m}_1$ in the space of constructive generalized functions.

Let Φ_1 be a constructive generalized function. We shall say that Φ_1 is a generalized function of order $\leqslant n$, if

$$\exists \Phi_2 (\Phi_1 = \Phi_2 \& \forall k (|\Phi_2|_k \geq n));$$

that Φ_1 is a generalized function of finite order if there exists a number n such that Φ_1 is a generalized function of order $\leqslant n$; that Φ_1 is a generalized function of n-th order if it is a generalized function of order $\leqslant n$ and, when $n \geq 1$, it is not a generalized function of order $\leqslant n-1$; that Φ_1 is a regular generalized function if it is a generalized function of order 0.

Theorem 3. It is not true that for every constructive generalized function Φ_1 of finite order there exists a number n such that Φ_1 is a constructive generalized function of n-th order.

Theorem 4. There is no algorithm which constructs, for every constructive generalized function Φ_1 of finite order, a number n such that Φ_1 is a constructive generalized function of order $\leqslant n$.

Theorem 5. There exists an algorithm which transforms every pair of the form $n \square \Phi_1$, satisfying the condition $\forall k (|\Phi_1|_k \leqslant n)$, into a word $\mathfrak{m}_1$ of type $\mathfrak{m}$ such that

$$\Phi_1 = \mathfrak{D}(n \square R(\mathfrak{m}_1)).$$

Roughly speaking, this means that every constructive generalized function of order $\leqslant n$ is the derivative of n-th order of some locally uniformly continuous function.

Corollary. For every regular constructive generalized function one can construct a word $\mathfrak{m}_1$ of type $\mathfrak{m}$ such that

$$\Phi_1 = R(\mathfrak{m}_1).$$

Theorem 6. Whatever the natural number n, there is no algorithm which transforms every constructive generalized function Φ_1 of order $\leqslant n$ into a constructive generalized function Φ_2 such that

$$\Phi_1 = \Phi_2 \& \forall k(|\Phi_2|_k \leqslant n).$$

Corollary. There is no algorithm which transforms every regular constructive generalized function Φ_1 into a word $\mathfrak{m}_1$ of type $\mathfrak{m}$ such that

$$\Phi_1 = R(\mathfrak{m}_1).$$

Theorem 3 is proved with the aid of Lemma 2, § 2 from (4). Theorem 6 is proved with the aid of Theorems 10.2.2 and 10.3.1 from (5).

The author expresses deep gratitude to A. A. Markov and N. A. Shanin for valuable advice and attention to the work.

Moscow State University
named after M. V. Lomonosov

Received
15 X 1966

CITED LITERATURE

1. N. A. Shanin, *Tr. Matem. inst. im. V. A. Steklova AN SSSR*, **52**, 266 (1958).
2. N. A. Shanin, ibid., **67**, 15 (1962).
3. G. S. Tseitin, ibid., **67**, 295 (1962).
4. G. S. Tseitin, ibid., **67**, 362 (1962).
5. G. E. Mints, ibid., **72**, 383 (1964).
6. Fan Dingzhu, *DAN*, **162**, No. 4, 766 (1965).
7. J. Mikusinski, R. Sikorski, *The Elementary Theory of Distributions*, Warszawa, 1 (1957); 2 (1959).
8. J. Korevaar, *Indagation Math.*, **17**, 3, 368; **17**, 4, 483; **17**, 5, 663 (1955).