PROCEEDINGS OF THE YEREVAN STATE UNIVERSITY

Physical and Mathematical Sciences 2018, **52**(3), p. 191–199

Informatics

ON MAIN CANONICAL NOTION OF δ-REDUCTION

D. A. GRIGORYAN *

Chair of Programming and Information Technologies YSU, Armenia

In this paper the main canonical notion of δ-reduction is considered. Typed λ-terms use variables of any order and constants of order ≤ 1, where constants of order 1 are strongly computable, monotonic functions with indeterminate values of arguments. The canonical notion of δ-reduction is the notion of δ-reduction that is used in the implementation of functional programming languages. For main canonical notion of δ-reduction the uniqueness of $\beta\delta$-normal form of typed λ-terms is shown.

MSC2010: 68N18.

***Keywords*:** main canonical notion, δ-reduction, $\beta\delta$-reduction, normal form.

Typed λ-Terms, Canonical Notion of δ-Reduction. The definitions of this section can be found in [1–3]. Let M be a partially ordered set, which has a least element $\perp$ which corresponds to the indeterminate value, and each element of M is comparable only with $\perp$ and itself. Let us define the set of types (denoted by Types).

1. $M \in Types$.

2. If $\beta, \alpha_1, \ldots, \alpha_k \in Types$ $(k > 0)$, then the set of all monotonic mappings from $\alpha_1 \times \cdots \times \alpha_k$ into β (denoted by $[\alpha_1 \times \cdots \times \alpha_k \to \beta]$) belongs to Types.

Let $\alpha \in Types$, then the order of type α (denoted by $ord(\alpha)$) will be a natural number, which is defined in the following way: if $\alpha = M$ then $ord(\alpha) = 0$, if $\alpha = [\alpha_1 \times \cdots \times \alpha_k \to \beta]$, where $\beta, \alpha_1, \ldots, \alpha_k \in Types$, $k > 0$, then $ord(\alpha) = 1 + \max(ord(\alpha_1), \ldots, ord(\alpha_k), ord(\beta))$. If x is a variable of type α and constant $c \in \alpha$, then $ord(x) = ord(c) = ord(\alpha)$.

Let $\alpha \in Types$ and V_α be a countable set of variables of type α, then $V = \bigcup\limits_{\alpha \in Types} V_\alpha$ is the set of all variables. The set of all terms, denoted by $\Lambda = \bigcup\limits_{\alpha \in Types} \Lambda_\alpha$, where Λ_α is the set of terms of type α, is defined in the following way:

1. if $c \in \alpha, \alpha \in Types$, then $c \in \Lambda_\alpha$;

2. if $x \in V_\alpha$, $\alpha \in Types$, then $x \in \Lambda_\alpha$;

* E-mail: david.grigoryan.a@gmail.com

3. if $\tau \in \Lambda_{[\alpha_1\times\dots\times\alpha_k\to\beta]}, t_i \in \Lambda_{\alpha_i}$, where $\beta, \alpha_i \in Types, i = 1,\dots,k,\ k \geq 1$, then $\tau(t_1,\dots,t_k) \in \Lambda_\beta$ (the operation of application $(t_1,\dots,t_k)$ is the scope of the applicator τ);

4. if $\tau \in \Lambda_\beta,\ x_i \in V_{\alpha_i}$ where $\beta, \alpha_i \in Types,\ i \neq j \implies x_i \neq x_j,\ i,j = 1,\dots,k$, $k \geq 1$, then $\lambda x_1 \dots x_k[\tau] \in \Lambda_{[\alpha_1\times\dots\times\alpha_k\to\beta]}$ (the operation of abstraction τ is the scope of the abstractor $\lambda x_1 \dots x_k$).

The notion of free and bound occurrences of variables as well as free and bound variable are introduced in the conventional way. The set of all free variables in the term t is denoted by *FV(t)*. Terms t_1 and t_2 are said to be congruent (which is denoted by $t_1 \equiv t_2$), if one the term can be obtained from the other by renaming the bound variables. The free occurrence of a variable in the term is called internal, if it does not enter in the applicator, which scope contains a free occurrence of some variable. The free occurrence of a variable in the term is called external, if it does not enter in the scope of the applicator that contains a free occurrence of some variable.

Let $t \in \Lambda_\alpha, \alpha \in Types$ and $FV(t) \subset \{y_1,\dots,y_n\}, \bar{y}_0 = (y_1^0,\dots,y_n^0)$, where $y_i \in V_{\beta_i},\ y_i^0 \in \beta_i,\ \beta_i \in Types,\ i = 1,\dots,n,\ n \geq 0$. The value of the term t for the values of the variables $y_1,\dots,y_n$, equal to $\bar{y}_0 = (y_1^0,\dots,y_n^0)$, is denoted by $Val_{\bar{y}_0}(t)$ and is defined in the conventional way.

Let terms $t_1, t_2 \in \Lambda_\alpha, \alpha \in Types,\ FV(t_1) \cup FV(t_2) = \{y_1,\dots,y_n\},\ y_i \in V_{\beta_i}$, $\beta_i \in Types,\ i = 1,\dots,n,\ n \geq 0$, then terms t_1 and t_2 are called equivalent (denoted by $t_1 \sim t_2$), if for any $\bar{y}_0 = (y_1^0,\dots,y_n^0)$, where $y_i^0 \in V_{\beta_i},\ i = 1,\dots,n$, we have the following: $Val_{\bar{y}_0}(t_1) = Val_{\bar{y}_0}(t_2)$. A term $t \in \Lambda_\alpha, \alpha \in Types$, is called a constant term with value $a \in \alpha$, if $t \sim a$.

Further, we assume that M is a recursive set and considered terms use variables of any order and constants of order ≤ 1, where constants of order 1 are strongly computable, monotonic functions with indeterminate values of arguments. A function $f : M^k \to M,\ k \geq 1$, with indeterminate values of arguments, is said to be strongly computable if there exists an algorithm, which stops with value $f(m_1,\dots,m_k) \in M$ for all $m_1,\dots,m_k \in M$ (see [1]).

To show mutually different variables of interest $x_1,\dots,x_k,\ k \geq 1$, of a term t, the notation $t[x_1,\dots,x_k]$, is used. The notation $t[t_1,\dots,t_k]$ denotes the term obtained by the simultaneous substitution of the terms $t_1,\dots,t_k$ for all free occurrences of the variables $x_1,\dots,x_k$, respectively, where $x_i \in V_{\alpha_i},\ i \neq j \Rightarrow x_i \not\equiv x_j,\ t_i \in \Lambda_{\alpha_i}$, $\alpha_i \in Types,\ i,j = 1,\dots,k,\ k \geq 1$. A substitution is said to be admissible if all free variables of the term being substituted remain free after the substitution. We will consider only admissible substitutions.

A term t with a different fixed occurrences of subterms τ_1, τ_2, where τ_1 is not a subterm of τ_2 and τ_2 is not a subterm of τ_1 and $\tau_i \in \Lambda_{\alpha_i},\ \alpha_i \in Types,\ i = 1,2$, is denoted by t_{τ_1,τ_2}. A term with the fixed occurrences of the terms τ_1, τ_2 replaced by the terms τ_1', τ_2' respectively is denoted by $t_{\tau_1',\tau_2'}$, where $\tau_i' \in \Lambda_{\alpha_i},\ i = 1,2$.

A term of the form $\lambda x_1,\dots,x_k[\tau[x_1,\dots,x_k]](t_1,\dots,t_k)$, where $x_i \in V_\alpha, i \neq j \Rightarrow x_i \not\equiv x_j, \tau \in \Lambda, t_i \in \Lambda_{\alpha_i}, \alpha_i \in Types,\ i,j = 1,\dots,k,\ k \geq 1$, is called a β-redex, and its convolution is the term $\tau[t_1,\dots,t_k]$. The set of all pairs (τ_0, τ_1), where τ_0 is a β-redex and τ_1 is its convolution, is called a notion of β-reduction and it is denoted by β. A one-step β-reduction ($\rightarrow_\beta$) and β-reduction ($\rightarrow\!\!\rightarrow_\beta$) are defined in the conventional way. A term containing no β-redexes is called a β-normal form. The set of all β-normal forms is denoted by β-NF.

δ-redex has a form $f(t_1,\dots,t_k)$, where $f \in [M^k \to M]$, $t_i \in \Lambda_M$, $i = 1,\dots,k$, $k \geq 1$, its convolution is either $m \in M$ and in this case $f(t_1,\dots,t_k) \sim m$ or a subterm t_i and in this case $f(t_1,\dots,t_k) \sim t_i$, $i = 1,\dots,k$. A fixed set of term pairs (τ_0,τ_1), where τ_0 is a δ-redex and τ_1 is its convolution, is called a notion of δ-reduction and is denoted by δ. A one-step δ-reduction ($\rightarrow_\delta$) and δ-reduction ($\rightarrow\!\!\rightarrow_\delta$) are defined in the conventional way.

A one-step $\beta\delta$-reduction ($\rightarrow$) and $\beta\delta$-reduction ($\rightarrow\!\!\rightarrow$) defined in the conventional way. A term containing no $\beta\delta$-redexes is called normal form. The set of all normal forms is denoted by NF.

A notion of δ-reduction is called a single-valued notion of δ-reduction, if δ is a single-valued relation, i.e. if $(\tau_0,\tau_1) \in \delta$ and $(\tau_0,\tau_2) \in \delta$, then $\tau_1 \equiv \tau_2$, where $\tau_0,\tau_1,\tau_2 \in \Lambda_M$. A notion of δ-reduction is called an effective notion of δ-reduction, if there exists an algorithm, which for any term $f(t_1,\dots,t_k)$, where $f \in [M^k \to M]$, $t_i \in \Lambda_M, i = 1,\dots,k$, $k \geq 1$, gives its convolution if $f(t_1,\dots,t_k)$ is a δ-redex and stops with a negative answer otherwise.

Definition 1. [2]. An effective, single-valued notion of δ-reduction is called a canonical notion of δ-reduction if:

1. $t \in \beta$-NF, $t \sim m$, $m \in M \setminus \{\perp\} \Rightarrow t \rightarrow\!\!\rightarrow_\delta m$;
2. $t \in \beta$-NF, $FV(t) = \varnothing$, $t \sim \perp \Rightarrow t \rightarrow\!\!\rightarrow_\delta \perp$.

Main Canonical Notion of δ-Reduction, Church–Rosser Property, the Uniqueness of the $\beta\delta$-Normal Form.

Definition 2. Let C be a recursive set of strongly computable, monotonic functions with indeterminate values of arguments. The following notion of δ-reduction is called main canonical notion of δ-reduction if for every $f \in C$, $f : M^k \to M$, $k \geq 1$, we have:

1. If $f(m_1,\dots,m_k) = m$, where $m, m_1,\dots,m_k \in M$, $m \neq \perp$, then $(f(\mu_1,\dots,\mu_k), m) \in \delta$, where $\mu_i = m_i$ if $m_i \neq \perp$, and $\mu_i \equiv t_i, t_i \in \Lambda_M$ if $m_i = \perp$, $i = 1,\dots,k$, $k \geq 1$.

2. If $f(m_1,\dots,m_k) = \perp$, where $m_1,\dots,m_k \in M$, $m \neq \perp$, then $(f(m_1,\dots,m_k),\perp) \in \delta$.

It is shown in the [2] that the δ is a canonical notion of δ-reduction.

Definition 3. The term $t \in \Lambda$ is said to be strongly normalizable, if the length of each $\beta\delta$-reduction chain from the term t is finite.

Theorem 1. [3]. Every term is strongly normalizable.

Theorem 2. [3]. For every term $t \in \Lambda$, if $t \rightarrow\!\!\rightarrow_\beta t', t \rightarrow\!\!\rightarrow_\beta t''$ and $t', t'' \in \beta$-NF, then $t' \equiv t''$.

Definition 4. Let $t \in \Lambda_\alpha, \alpha \in Types$ and $t \equiv t_1 \to \cdots \to t_n$, $n \geq 1$, where $t_i \in \Lambda_\alpha$, $i = 1,\dots,n$, then the sequence $t_1,\dots,t_n$ is called the inference of the term t_n from the term t and n is called the length of that inference.

D e f i n i t i o n 5. The inference tree of the term t is an oriented tree with the root t, and if a term τ is some node of the tree and $\tau_1,\ldots,\tau_k,\ k\geq 0$, are all $\beta\delta$-redexes of τ, then $\tau_{\tau_1'},\ldots,\tau_{\tau_k'}$ are all descendants of the node τ, where τ_i' is the convolution of $\tau_i,\ i=1,\ldots,k$.

It is easy to see that each node in the inference tree of the term t has finite number of descendants, and if τ is a leaf of that tree, then $\tau \in NF$.

L e m m a 1. Let δ be the main canonical notion of δ-reduction, t_τ be a term with a fixed occurrence of the term τ. If t is a δ-redex, τ is a $\beta\delta$-redex, then there exists $m\in M, m\neq\perp$ such that $t\rightarrow_\delta m$ and $t_{\tau'}\rightarrow_\delta m$, where τ' is the convolution of the $\beta\delta$-redex τ.

P r o o f. Let $t_\tau \equiv f(t_1,\ldots,t_{j_\tau},\ldots,t_k), f\in[M^k\rightarrow M],\ t_i\in\Lambda_M,\ i=1,\ldots,k,$ $1\leq j\leq k$. Since τ is a $\beta\delta$-redex, then $t_{j_\tau}\notin M$ and since t is a δ-redex, then from Definition 2 it follows that there exists $m\in M,\ m\neq\perp$, such that $(t,m)\in\delta$. Therefore, $t\rightarrow_\delta m$. Since $t_{j_\tau}\notin M$ and $(t,m)\in\delta$, where $m\neq\perp$, then from Definition 2 it follows that $f(t_1,\ldots,\mu,\ldots,t_k)\in\delta$ for every term $\mu\in\Lambda_M$. Therefore, $(f(t_1,\ldots,t_{j_{\tau'}},\ldots,t_k),m)\in\delta$ and $t_{\tau'}\rightarrow_\delta m$.

L e m m a 2. For the main canonical notion of δ-reduction δ and for every term $t\in\Lambda_\alpha, \alpha\in Types$, if $t\rightarrow t_1, t\rightarrow t_2,\ t_1,t_2\in\Lambda_\alpha$, then there exists a term $t'\in\Lambda_\alpha$ such that $t_1\twoheadrightarrow t'$ and $t_2\twoheadrightarrow t'$.

$$\begin{tikzcd} & t \arrow[dl] \arrow[dr] & \\ t_1 \arrow[dr, two heads] & & t_2 \arrow[dl, two heads] \\ & t' & \end{tikzcd} \tag{1}$$

P r o o f. If $t_1\equiv t_2$, then $t'\equiv t_1\equiv t_2$. If $t_1\not\equiv t_2$, then there exist $\beta\delta$-redexes $\tau_1,\tau_2\in\Lambda$ such that $t\equiv t_{\tau_1}\equiv t_{\tau_2},\ t_1\equiv t_{\tau_1'}$ and $t_2\equiv t_{\tau_2'}$, where terms τ_1',τ_2' are the convolutions of τ_1 and τ_2 accordingly. If τ_1 is not a subterm of τ_2 and τ_2 is not a subterm of τ_1, then from (2) it follows that $t'\equiv t_{\tau_1',\tau_2'}$.

$$\begin{tikzcd} & t\equiv t_{\tau_1,\tau_2} \arrow[dl] \arrow[dr] & \\ t_1\equiv t_{\tau_1',\tau_2} \arrow[dr] & & t_{\tau_1,\tau_2'}\equiv t_2 \arrow[dl] \\ & t_{\tau_1',\tau_2'}\equiv t' & \end{tikzcd} \tag{2}$$

If τ_2 is a subterm of τ_1 or τ_1 is a subterm of τ_2, then the following cases are possible: τ_1 and τ_2 are both δ-redexes. Without loss of generality we suppose that τ_2 is a subterm of τ_1 $(\tau_1 \equiv \tau_{1_{\tau_2}})$. From Lemma 1 it follows (3):

$$\begin{tikzcd}
 & \theta \equiv \tau_{1_{\tau_2}} \arrow[dl, "\delta"'] \arrow[dr, "\delta"] & \\
\theta_1 \equiv \tau_1' \equiv m \arrow[dr, "\delta"'] & & \tau_{1_{\tau_2'}} \equiv \theta_2 \arrow[dl, "\delta"] \\
 & m \equiv \theta' &
\end{tikzcd} \tag{3}$$

where $m \in M$, $m \neq \perp$ and m is the convolution of the term τ_1. Therefore, from (4) it follows that $t' \equiv t_m$.

$$\begin{tikzcd}
 & t \equiv t_\theta \arrow[dl, "\delta"'] \arrow[dr, "\delta"] & \\
t_1 \equiv t_{\theta_1} \arrow[dr, "\delta"'] & & t_{\theta_2} \equiv t_2 \arrow[dl, "\delta"] \\
 & t_{\theta'} \equiv t' &
\end{tikzcd} \tag{4}$$

τ_1 and τ_2 are both β-redexes. From Theorem 2 and (5) it follows that $t' \equiv t_1' \equiv t_2'$.

$$\begin{tikzcd}
 & t \arrow[dl, "\beta"'] \arrow[dr, "\beta"] \arrow[dd, "\beta"] & & \\
t_1 \arrow[dr, "\beta"'] & & t_2 \arrow[dl, "\beta"] & t', t_1', t_2' \in \beta\text{-}NF \\
 & t_1' \equiv t' \equiv t_2' & &
\end{tikzcd} \tag{5}$$

τ_1 is δ-redex and τ_2 is β-redex or τ_1 is β-redex and τ_2 is δ-redex. Without lose of generality we suppose that τ_1 is δ-redex and τ_2 is β-redex. Let $\tau_2 \equiv \lambda x_1, \ldots, x_n[\tau[x_1, \ldots, x_n]](\mu_1, \ldots, \mu_n)$, $\tau \in \Lambda, x_i \in V_{\alpha_i}, \alpha_i \in Types$, $i = 1, \ldots, n$.

If $\tau_1 \equiv \tau_{1_{\tau_2}}$, then Lemma 1 implies (6):

$$\begin{tikzcd}
 & \theta \equiv \tau_{1_{\tau_2}} \arrow[dl] \arrow[dr] & \\
\theta_1 \equiv \tau_1' \equiv m \arrow[dr, two heads] & & \tau_{1_{\tau_2'}} \equiv \theta_2 \arrow[dl] \\
 & m \equiv \theta' &
\end{tikzcd} \tag{6}$$

where $m \in M, m \neq \perp$ and m is the convolution of the term τ_1. Therefore from (7) it follows that $t' \equiv t_m$.

$$\begin{tikzcd}
 & t \equiv t_\theta \arrow[dl, "\delta"'] \arrow[dr, "\beta"] & \\
t_1 \equiv t_{\theta_1} \arrow[dr, two heads] & & t_{\theta_2} \equiv t_2 \arrow[dl, "\delta"] \\
 & t_{\theta'} \equiv t' &
\end{tikzcd} \tag{7}$$

If $\tau_2 \equiv \lambda x_1 \ldots x_n[\tau_{\tau_1}[x_1,\ldots,x_n]](\mu_1,\ldots,\mu_n)$, then it is easy to see that if $\tau_1[x_1,\ldots,x_k] \to_\delta \tau_1'$, then $\tau_1[\mu_1,\ldots,\mu_k] \to_\delta \tau_1'$ and we have:

$$\begin{tikzcd}
 & \theta \equiv \lambda x_1 \ldots x_n \left[\tau_{\tau_1}[x_1,\ldots,x_n]\right](\mu_1,\ldots,\mu_n) \arrow[dl, "\delta"'] \arrow[dr, "\beta"] & \\
\theta_1 \equiv \lambda x_1 \ldots x_n \left[\tau_{\tau_1'}[x_1,\ldots,x_n]\right](\mu_1,\ldots,\mu_n) \arrow[dr, "\beta"'] & & \tau_{\tau_1}[\mu_1,\ldots,\mu_n] \equiv \theta_2 \arrow[dl, "\delta"] \\
 & \tau_{\tau_1'}[\mu_1,\ldots,\mu_n] \equiv \theta' &
\end{tikzcd} \tag{8}$$

Therefore from (9) it follows that $t' \equiv t_{\tau_{\tau_1'}[\mu_1,\ldots,\mu_n]}$.

$$
\begin{tikzcd}
& t \equiv t_\theta \arrow[dl, "\delta"'] \arrow[dr, "\beta"] & \\
t_1 \equiv t_{\theta_1} \arrow[dr, "\beta"'] & & t_{\theta_2} \equiv t_2 \arrow[dl, "\delta"] \\
& t_{\theta'} \equiv t' &
\end{tikzcd}
\qquad (9)
$$

If $\tau_2 \equiv \lambda x_1 \ldots x_n[\tau[x_1,\ldots,x_n]](\mu_1,\ldots,\mu_{i_{\tau_1}},\ldots,\mu_n)$, then without loss of generality we suppose that $i = 1$ and from (10) and (11) we get $t' \equiv t_{\tau[\mu_{1_{\tau_1'}},\ldots,\mu_n]}$.

$$
\begin{tikzcd}
& \theta \equiv \lambda x_1 \ldots x_n[\tau[x_1,\ldots,x_n]]\left(\mu_{1_{\tau_1}},\ldots,\mu_n\right) \arrow[dl, "\delta"'] \arrow[dr, "\beta"] & \\
\theta_1 \equiv \lambda x_1 \ldots x_n[\tau[x_1,\ldots,x_n]]\left(\mu_{1_{\tau_1'}},\ldots,\mu_n\right) \arrow[dr, "\beta"'] & & \tau\left[\mu_{1_{\tau_1}},\ldots,\mu_n\right] \equiv \theta_2 \arrow[dl, two heads, "\delta"] \\
& \tau\left[\mu_{1_{\tau_1'}},\ldots,\mu_n\right] \equiv \theta' &
\end{tikzcd}
\qquad (10)
$$

$$
\begin{tikzcd}
& t \equiv t_\theta \arrow[dl, "\delta"'] \arrow[dr, "\beta"] & \\
t_1 \equiv t_{\theta_1} \arrow[dr, "\beta"'] & & t_{\theta_2} \equiv t_2 \arrow[dl, two heads, "\delta"] \\
& t_{\theta'} \equiv t' &
\end{tikzcd}
\qquad (11)
$$

In conclusion, we showed that in all cases there exists a term t' such that $t_1 \to\!\to t'$ and $t_2 \to\!\to t'$.

L e m m a 3. For every term t the number of inferences to normal forms from the term t is finite.

P r o o f . We consider the inference tree of the term t. Let us suppose that the number of inferences to normal forms from the term t is infinite, which means that the number of paths from root t to leafs is also infinite. Since every node in the inference tree has finite number of descendants from the König's lemma it follows that there exists an infinite path that starts from the root t, which contradicts Theorem 1. Therefore, the number of paths from the root t to leafs is finite, which means that the number of inferences to normal forms from the term t is also finite.

It follows from Lemma 3 that for every term t the inference tree of the term t is a finite tree. The height of an inference tree of the term t is the length of the longest path from the root t to a leaf.

Definition 6. The set of all terms the height of the inference tree of which is equial to $n-1$ is denoted by $\Lambda^{(n)}$, $n \geq 1$.

Definition 7. The notion of $\beta\delta$-reduction has the Church–Rosser property (CR-property), if for every term $t \in \Lambda_\alpha, \alpha \in Types$, if $t \twoheadrightarrow t_1$ and $t \twoheadrightarrow t_2$, $t_1, t_2 \in \Lambda_\alpha$, then there exists a term $t' \in \Lambda_\alpha$ such that $t_1 \twoheadrightarrow t'$ and $t_2 \twoheadrightarrow t'$.

$$\begin{tikzcd}
 & t \arrow[dl, two heads] \arrow[dr, two heads] & \\
t_1 \arrow[dr, two heads] & & t_2 \arrow[dl, two heads] \\
 & t' &
\end{tikzcd} \tag{12}$$

Theorem 3. For the main canonical notion of δ-reduction δ the notion of $\beta\delta$-reduction has the CR-property.

Proof. Let $t \in \Lambda^{(1)}$, then $t \in NF$ and $t \equiv t_1 \equiv t_2 \equiv t'$. Now, let us suppose that CR-property holds for every term $\tau \in \Lambda^{(k)}, k \leq n-1, n \geq 2$ and show that it holds for every term $t \in \Lambda^{(n)}$. If $t \equiv t_1$ then $t_1 \twoheadrightarrow t_2$ and $t' \equiv t_2$. If $t \equiv t_2$, then $t_2 \twoheadrightarrow t_1$ and $t' \equiv t_1$. If $t_1 \not\equiv t$ and $t_2 \not\equiv t$, then there exist terms $t_1', t_2' \in \Lambda$, such that $t \to t_1' \twoheadrightarrow t_1$ and $t \to t_2' \twoheadrightarrow t_2$. Therefore from Lemma 2 it follows that there exists a term t' such that $t_1' \twoheadrightarrow t'$ and $t_2' \twoheadrightarrow t'$.

Since $t_1' \twoheadrightarrow t_1$, $t_1' \twoheadrightarrow t'$ and $t_1' \in \Lambda^{(k_1)}, 1 \leq k_1 \leq n-1$, from the induction hypothesis it follows that there exists a term t_1'' such that $t_1 \twoheadrightarrow t_1''$ and $t' \twoheadrightarrow t_1''$. Since $t_2' \twoheadrightarrow t_2$, $t_2' \twoheadrightarrow t'$ and $t_2' \in \Lambda^{(k_2)}, 1 \leq k_2 \leq n-1$, from the induction hypothesis it follows that there exists a term t_2'' such that $t_2 \twoheadrightarrow t_2''$ and $t' \twoheadrightarrow t_2''$. Since $t' \twoheadrightarrow t_1''$, $t' \twoheadrightarrow t_2''$ and $t' \in \Lambda^{(k_3)}, 1 \leq k_3 \leq n-1$, from the induction hypothesis it follows that there exists a term t'' such that $t_1'' \twoheadrightarrow t''$ and $t_2'' \twoheadrightarrow t''$. Therefore $t_1 \twoheadrightarrow t''$ and $t_2 \twoheadrightarrow t''$.

$$\begin{tikzcd}
 & & t \arrow[dl] \arrow[dr] & & \\
 & t_1' \arrow[dl, two heads] \arrow[dr, two heads] & & t_2' \arrow[dl, two heads] \arrow[dr, two heads] & \\
t_1 \arrow[dr, two heads] & & t' \arrow[dl, two heads] \arrow[dr, two heads] & & t_2 \arrow[dl, two heads] \\
 & t_1'' \arrow[dr, two heads] & & t_2'' \arrow[dl, two heads] & \\
 & & t'' & &
\end{tikzcd} \tag{13}$$

Theorem 4. For the main canonical notion of δ-reduction δ and for every term $t \in \Lambda$, if $t \twoheadrightarrow t', t \twoheadrightarrow t''$ and $t', t'' \in NF$, then $t' \equiv t''$.

Proof. Let us suppose that the original statement is false and $t' \not\equiv t''$. Since for the main canonical notion of δ-reduction δ the $\beta\delta$-reduction has the CR-property, there exists a term $t''' \in \Lambda$ such that $t' \twoheadrightarrow t'''$ and $t'' \twoheadrightarrow t'''$. Since $t', t'' \in NF$, $t' \equiv t'' \equiv t'''$. Therefore, we have a contradiction and the original statement is true.

Received 04.07.2018

REFERENCES

1. **Nigiyan S.A.** On Non-classical Theory of Computability. // Proceedings of the YSU. Physical and Mathematical Sciences, 2015, № 1, p. 52–60.
2. **Nigiyan S.A., Khondkaryan T.V.** On Canonical Notion of δ-Reduction and on Translation of Typed λ-Terms Into Untyped λ-Terms. // Proceedings of the YSU. Physical and Mathematical Sciences, 2017, v. 51, № 1, p. 46–52.
3. **Budaghyan L.E.** Formalizing the Notion of δ-Reduction in Monotonic Models of Typed λ-Calculus. // Algebra, Geometry and Their Applications, 2002, v. 1, p. 48–57.