ON A FORMALISM WHICH MAKES ANY SEQUENCE OF SYMBOLS WELL-FORMED

SHIGEO ŌHAMA

Any finite sequence of primitive symbols is not always well-formed in the usual formalisms. But in a certain formal system, we can normalize any sequence of symbols uniquely so that it becomes well-formed. An example of this kind has been introduced by Ono [2]. While we were drawing up a practical programming along Ono's line, we attained another system, a modification of his system. The purpose of the present paper is to introduce this modified system and its application. In **1**, we will describe a method of normalizing sentences in ***LO***[1] having only two logical constants, implication and universal quantifier, so that any finite sequence of symbols becomes well-formed. In **2**, we will show an application of **1** to proof. I wish to express my appreciation to Prof. K. Ono for his significant suggestions and advices.

1. Normalizing sentences. In our formalism, similarly in Ono [2], we use only one category of variables and a pair of brackets "[" and "]" called HEAD- and TAIL-BRACKET, respectively. So a sentence $\mathscr{A}$ in usual notation is transformed to A as follows;

(1) If $\mathscr{A}$ is an n-ary relation $R(x, ---, z)$, then A is $[rx---z]$, where r is denoted as a predicate variable corresponding to R,

(2) If $\mathscr{A}$ is of the form $(x)---(z)\mathscr{B}$, then A is of the form $x---zB$,

(3) If $\mathscr{A}$ is of the form $\mathscr{B}\rightarrow(---(\mathscr{C}\rightarrow\mathscr{D})---)$ and B, C, D are translated forms of $\mathscr{B}$, $\mathscr{C}$, $\mathscr{D}$, respectively, then A is of the form $B^*---C^*D^*$, where B^* and C^* denote B and C, respectively, in the case of the left most symbols of B and C being corresponding head-brackets of the right most tail-brackets of them and otherwise $[B]$ and $[C]$, respectively, and D^* denote $[D]$ in the case of the left most symbol of D being a variable and otherwise D.

Received October 21, 1966.

1) See Ono [1].

For example, a sentence

$$(x)\,(R(x,u)\rightarrow(y)\,(z)\,(S(y,z,u)\rightarrow R(y,z)))\rightarrow(R(w,u)\rightarrow S(w,v,u))\,,$$

where R and S are binary and ternary relation, respectively, is translated as follows,

$$[\,x[\,rxu\,]\,[\,yz\,[\,syzu\,]\,[ryz\,]\,]\,]\,[\,rwu\,]\,[\,swvu\,]\,.$$

Now, let us define SENTENCE and NORMAL SENTENCE.

Definition 1. *Any sequence of symbols is called SENTENCE.*

Definition 2. *Any sentence A is called NORMAL SENTENCE if and only if*

(1) *A includes at least one barcket,*

(2) *any tail-bracket in A is not immediately followed by a variable(s),*

(3) *in any segment* A_i *(i.e. subsequence of A, from the first symbol to the i-th symbol), the number of tail-brackets does not exceed the number of head-brackets and the whole number of tail-brackets is equal to that of head-brackets.*

We can uniquely normalize any sentence, if not normal, by the following operation.

Operation 1. *If a given sentence does not satisfy the condition* (2) *in Definition* 2, *insert a head-bracket between the tail-bracket and the variable(s), and repeat this operation until a resulting sentence satisfies the condition* (2) *in Definition* 2.

Operation 2. *If the sentence resulting from Operation* 1 *does not satisfy the condition* (1) *or* (3) *in Definition* 2, *then add head-bracket(s) and tail-bracket(s) at the beginning and at the end of the sentence, respectively, so that it satisfies the condition* (1) *and* (3) *in Definition* 2.

For example, a sentence

$$fuvw\,]\,]\,xyz\,[\,gxyzu\,]\,fxyz\,]\,]\,[\,xyz\,[\,guvwz\,]\,fxyw$$

becomes by Operation 1

$$fuvw\,]\,]\overset{\downarrow}{[}\,xyz\,[\,gxyzu\,]\overset{\downarrow}{[}\,fxyz\,]\,]\,[xyz\,[guvwz\,]\overset{\downarrow}{[}\,fxyw$$

and this becomes normal by applying Operation 2

$$\overset{\downarrow}{[}\;\overset{\downarrow}{[}\,fuvw\,]\,]\,[\,xyz\,[\,gxyzu\,]\,[\,fxyz\,]\,]\,[\,xyz\,[\,guvwz\,]\,[fxyw\overset{\downarrow}{]}\;\overset{\downarrow}{]}\,,$$

where $\overset{\downarrow}{[}$ and $\overset{\downarrow}{]}$ mean brackets added by each operation.

2. APPLICATION TO PROOF. Our normalization excludes sentences of the form $x---zAu---w$,[2] which is regarded as normal sentence in Ono's system and is useful. But if we modify description of proof-notes a little, we can describe any proof-note without making use of sentences of the above form. Now let us rewrite the example proof in Ono [2] by our modified way.

$[xy[rxy][ryx]][xyz[rxy][ryx]rxz]][xy[rxy][rxx]],\ a,\ b,$

$a,,\ xy[rxy][ryx],\ xyz[rxy][ryz][rxz],$

$b,\ xy[rxy][rxx],\ ba,\ be,$

$ba,,\ [ruv],\ uv,$

$bb,\ [ruv][rvu],,\ uv,\ a,$

$bc,\ [rvu],\ ba,\ bb,$

$bd,\ [ruv][rvu][ruu],,\ uvu,\ a,$

$be,\ [ruu],\ ba,\ bc,\ bd.$

In assumption step φa, we allow only one reference index which is a series of the same number of mutually distinct variables as the outest series of quantifiers in the step φ. And any variable of the reference index does not occur as free in any step beginning with φ or beginning with an index in the ground of φ. The φa step

$$\varphi a,,\ \alpha_1,\ ---,\ \alpha_k,\ \sigma,$$

means "Take any series of variables of fixed length, say σ, satisfying the condition α_1, ---, α_k". where α_1, ---, α_k are sentences. (*cf.* the step *ba*).

In assertion step φ, we allow the reference index following immediately after a sentence over two successive commas. The φ step

$$\varphi,\ \alpha,,\ \sigma,\ ---,$$

means "Any series of variables of fixed length satisfies condition α, so we take any one series of them and call it σ", where α is a sentance, σ is an index, and "---" represents a sequence of indices (*cf.* steps *bb*, *bd*).

[2] For two variable series, $x---z$ and $u---w$ being the same length and a sentence A, this means that free variables x, ---, z in A are substituted by u, ---, w, respectively.

References

[1] Ono. K: A certain kind of formal theories, Nagoya Math. J., Vol. **25** (1965), pp. 59–86.
[2] ———: A formalism for primitive logic and mechanical proof-checking, Nagoya Math. J., Vol. **26** (1966), pp. 195–203.

Nagoya University