# Foundations of Semantic Coding: Experience of Knowledge Representation and Transformation

Prof. Victor V. Martynov National Academy of Science of Belarus

#### Introduction

The theoretical basis of USC has been developed during the last twenty seven years from version to version (Victor V. Martynov, 1974 – the first version, 1977 – the second one, 1983 – the third, 1988 the fourth, 1996 – the fifth, and the current 2001 - the sixth one).

As far as the content of USC-5 is concerned we specify it in the following way:

1. Every string of symbols corresponds to only one meaning and every string transformation corresponds to only one meaning transformation.

2. Declarative knowledge is represented as procedural, in other words, no other criteria are considered for any object of the system but the function the object performs.

3. All the key verbs chosen for USC vocabulary have symbolic and canonized natural language representation, the rest of the verbs refer to the key verbs.

4. Nouns are defined on the basis of the corresponding verbs and modal characteristics being added to each noun: that which is intended for ... (or, can be used for).

5. According to formal characteristics the rules of reading USC strings in any natural language have been set up.

It seems expedient to determine the place USC occupies within the scope of formal knowledge representing means designed for artificial intelligence (AI) systems. Formal grammars generated on the basis of N. Chomsky (1981) conception can be used for creating computer-programming languages. Absence of semantic interpretation does not allow them to become knowledge representing languages with one-to-one correlation between syntactic and semantic elements.

As per S. Amarel (1968) and R. Schank (1975) priority in intellectual problems solving is given to knowledge representing, thus the necessity of developing language systems has been predetermined. Natural language in its noncanonized form cannot meet this requirement. Formal system with full semantic interpretation is embodied in R. Montague (1974) grammar. However, Montague semantics is defined at the natural language word level without exposing its inner semantic structure. We could say on well-known analogy that in this case the investigation covers only molecular level. It's easy to imagine what level would have reached our technology if physics had shared the same fate.

Besides USC there are actually only two projects to make up a language with formalized semantics: the model of conceptual dependence proposed by R.C. Schank (1975) and the "meaning-text" model of I.A. Melchuk (1974). The basis of both is some primitives (semantic elements): primitive actions in Schank's model and lexical functions in Melchuk's, which form semantic notation of utterances. The primitives of the given models do not claim to be

complete, independent, and consistent in the strict sense of the word because of their empirical elaboration. The deductive theory of knowledge representing language has been embodied in USC for the first time. Nowadays it becomes clear that none of the variants of artificial intelligence can be effective without formal representation and transformation of sense since only under these conditions the computer modeling of mind processes is ensured.

A third version of USC (USC-3) is entirely in agreement with the given principles. That is why it was realized as computer assisted. A fourth version of USC (USC-4) differs from USC-3 in two fundamental characteristics:

- 1. In exclusion of special means of representation of information, and modality (these categories are represented by traditional ternary strings of elementary symbols).
- 2. In the explication of USC-4 as a certain algebra.

Each axiom represents a regular transformation of sense in explicit form. It is a pity that a majority of researchers does not understand so far that no kind of artificial intelligence system can exist without semantic explication in the sense of substitution of the strict concept for the intuitional one. The addition of the fact compels us to return to the main question of knowledge representing languages, bases of knowledge, knowledge as such.

We shall demonstrate our understanding of the problem by displaying some examples. It is natural enough for a human to come to the following conclusion: *The engineer has seen the device before that is why he would recognize it* or in the general form: X has seen  $Y \rightarrow X$  would recognize Y. If our system is intellectual enough it would know how to draw this immediate conclusion. In other words, the creator of artificial intelligence systems has to know the way of teaching a computer to draw such a kind of conclusion. Regretfully he does not know how to do it. Moreover, he cannot perceive how the human does it.

Let us try to assist the system giving for comparison another instance of deduction: *He* has already played Rossini's "Tarantella" that is why he would play it or in a more general form: X has played  $Y \rightarrow X$  would play (X can possibly play Y). A human identifies the verb to play in spite of grammatical differences. It is evident that the former deduction can be reduced to the postulate of modal logic rule  $P \rightarrow \Diamond P$ . Though we guess the first deduction is reducible to the same postulate, we do not know how to explain it if only for a human. The human easily uses such deduction but by intuition purely.

Let us begin from the right part of the first utterance: X would recognize Y. This sentence signifies a possible result of action represented in the left part: X can possibly recognize Y. It should be emphasized that after transformation the right part has coincided with the same part of the former utterance and the above-mentioned rule. Concerning the left part, the verb to see may be interpreted as to receive information, to get to know. Then we note the whole as follows: X has known  $Y \rightarrow X$  can possibly know (recognize) Y. In this case we are concerned with the same postulate of modal logic:  $P \rightarrow \Diamond P$ .

In order to identify P in left and right parts of our conclusion we must put semantics into formalisms of representation.

It is easy to understand that the formal representation and the canonized language as well demonstrate the identity of P in the left and right parts of the conclusion and distinguish the modal operator in the right one. The number of such instances can be multiplied, but it is quite enough to make clear that the only means of human conclusion formalization is a powerful semantic code. Its potentialities can be implemented in the working system of artificial intelligence if it would be supplied with specialized dictionary for translating natural language phrases into semantic notation (USC notation) or if a human would use USC type of language in the process of this intercourse with computer. We see no other solution.

Formalization of lexical semantics cannot solve AI problems because of the natural language vagueness that follows from the discrepancy between the complexity of the syntactic and semantic structures. Such discrepancy arises due to ellipticity of natural language phrases. Thus the following phrase "Your child eats with his hands" is reconstructed in full as "Your child eats with his (mouth, holding food with his) hands".

Making comparative analysis of the following phrases "John beats Jim" and "John expects Jim", we figure out that in spite of their full syntactic coincidence they have important semantic distinction. Asking the question "What does John do to Jim?" we get a regular answer "He beats him" and meaningless, in the case "He expects him". Actually, the phrase "John beats Jim" has an "atomic" semantic structure while the phrase "John expects Jim" has a "molecular" one. Semantic reconstruction of the second phrase is: "John is where he expects Jim to be soon".

Let's compare semantic-syntactic representation in USC of two phrases: "John protects Jim from something" and "John guides Jim through something". The symbolic representation of the first phrase is: ((XY)Z)(Z(WY)), which reads in natural canonized language as "John by means of Y protects Jim (i.e., by means of Y preserves him from something, John does so that Jim still exists)". The symbolic representation of the second phrase would be: ((XY)Z)((ZZ)Y), which reads in natural canonized language as "X by means of Y guides Z (i.e., by means of Y lets Jim goes through Y)". It is easy to see that Y in these phrases performs different function. "Protects, preserves" by means of Y makes us think that Y is some medicine. "Guides through" by means of Y shows that Y is some entrance or exit (like a door or a hatch). The difference between syntactic structures of the phrases is isomorphic to the difference in semantic structures.

Creating on USC basis semantic intellectual system shows its potentialities first of all as an automatic solver of intellectual (inventive) problems. A user just has to fill out a form of the following type: who "X" by means of what "Y" acting on what "Z" gets what "W".

Having assigned names of actors of the action and devices that are being used to the variables a user determines initial situation (actually, does not realize that this way he is

developing a knowledge base) and goal situation. Then a computer shows possible ways of changing from the initial to the final state.

We have been trying to demonstrate the main ways of generating and transforming the strings of USC and to show how this knowledge representing language in its program variant can become a solver of intellectual problems.

The knowledge base (KB) in our system is based on the axioms of the USC algebra and is formed as an oriented graph (presentation in the form of a matrix or a list is also possible). The nodes of the graph are represented in the form of USC strings. The arcs – in the form of USC axioms or theorems of the given algebra. It is obvious that the solution of the intellectual problem can be presented in the form of a route set by the succession of arcs. The algorithm of the problem solution is based on the successive drawing of the route from the goal situation to the initial one.

Apart from the axioms, the KB of our system contains the semantic vocabulary of most commonly used verbs of scientific and technology literature. Each of the verbs is either defined in the USC vocabulary or has a reference to the synonym having a similar definition. The user forms the KB in respect to the given domain. His utterances are limited by only one verb, which is picked up by the user himself and introduced into the computer. There it is referred to the verb vocabulary and by means of definition of a USC string a necessary set of positions is given. The positions are filled in by the user with relevant names and again introduced into the computer. If the verb is not in the vocabulary the system suggests to introduce a synonym or to simplify the whole utterance.

#### **USC** axiomatic

At present, formal part of the USC-6 version has the following status: type of algebra (A) of USC strings representation and transformation has been determined as:

#### $\mathsf{A} = <\mathsf{M}, \ \not \rightarrow \ , \ \thicksim >$

Where:

[M] is a set of elements

 $[\rightarrow]$  is the binary-non-commutative and non-associative operation on the given set (the operation of implication)

[~] is the unary operation on the given set (the operation of negation).

This kind of universal algebra is widely spread and strictly corresponds to Lukaszewicz variant of Lindenbaum algebra:

Lindenbaum: A = < M, U,  $\cap$ ,  $\rightarrow$ , ~ > disjunction, conjunction, implication, negation.

Lukasiewicz J.(1958): A = < M,  $\rightarrow$ , ~ > implication, negation. Lukaszewicz has proved selfsufficiency of implication and negation. Using those two operations all other operations can be determined.

USC algebra includes only strict implication and special case of negation ranked in ascending order of magnitude (three grades of rank):

 $[ \rightarrow ] = [ \rightarrow 1 \rightarrow 2 \rightarrow 3 ]$  - if ... then = start of influence  $\rightarrow$  influence  $\rightarrow$  end of influence

 $[\sim] = [\sim 1 \sim 2 \sim 3]$  - not = inside ~ superficially ~ outside

We accept a set of axioms within the scope of this "semantic" algebra.

## I. Axioms of generation

## Axiom of application

If <X> and <Y> are set elements <X  $\rightarrow$  Y> is a set element too and vice versa. The set elements are:

 $X \rightarrow Y$  - kernel string

 $(X \rightarrow Y) \rightarrow Z$  - extended string

 $((X \rightarrow Y) \rightarrow Z) \rightarrow ((... \rightarrow ...) \rightarrow ...)$  - complex string

### Axiom of canonization 1

The left part of the complex string takes the following canonized forms:

 $(X \rightarrow Y) \rightarrow Z$  (complex physical string)

 $(X \rightarrow Y) \rightarrow X$  (complex informational string)

# Axiom of canonization 2

The starting element of the right part of the complex string is identical to the final element of the left part

$$((X \rightarrow Y) \rightarrow Z) \rightarrow ((Z \rightarrow ...) \rightarrow ...)$$

$$((X \rightarrow Y) \rightarrow X) \rightarrow ((X \rightarrow ...) \rightarrow ...)$$

In this case only the right part of the complex string is recorded. We can also eliminate the sign of implication:

 $(X \rightarrow Y) \rightarrow Z$  is equal to (XY)Z etc.

### Axiom of fixation

The unary operation can be executed only on the element in the final position of the complex string.

Axioms of generation make it possible to construct words and ideas classified on the basis of semantic modeling.

# **II.** Axioms of transformation

### Axiom of transposition

The right part of the complex string can be transformed by means of changing the sequence of operations (brackets shift). The string before transformation we name as 'active' and after as 'passive'.

 $(ZY)W \rightarrow Z(YW)$  etc.

## Axiom of diffusion

The right part of the complex string can be transformed by spreading the element in the first or second position to the second or third.



### Axiom of divergence

The left part of the complex string can be transformed by spreading the element in the first position to the third one.

 $(XY)Z \rightarrow (XY)X$ 

# Axiom of correlation

Each X of the complex information string, except for X in the first position, correlates with Z in the complex physical string and each Y of the complex information string, except Y in the second position, correlates with W in the complex physical string.

 $((XY)X)((XY)Y) \rightarrow ((XY)Z)((ZW)W)$  etc.

Note: The strokes ( ' , " ) mean grade rank [  $\sim$  ], the synonyms compensate their semantic inaccuracy.

Complex physical strings								
Active			Passive					
1	create	(ZW)Y	exist	Z(WY)				
	start	(ZW)Y'	originate	Z(WY')				
	preserve	(ZW)Y"	be preserved	Z(WY")				
	destroy	(ZY)W	be absent	Z(YW)				
2	accomplish	(ZY)W'	degenerate	Z(YW')				
	prevent	(ZY)W"	lose	Z(YW")				
3	compress	(ZW)W	squeeze	Z(WW)				
	join	(ZW)W'	contract	Z(WW')				
	concentrate	(ZW)W"	combine	Z(WW")				
	separate	(ZY)Y	become dissociated	Z(YY)				
4	part	(ZY)Y'	dissolve	Z(YY')				
	disperse	(ZY)Y"	scatter	Z(YY")				
	lead in	(ZZ)W	go in	Z(ZW)				
5	bring	(ZZ)W'	come	Z(ZW')				
	bring closer	(ZZ)W"	come up	Z(ZW")				
	bring out	(ZZ)Y	go out	Z(ZY)				
6	move away	(ZZ)Y'	move aside	Z(ZY')				
	go away	(ZZ)Y"	depart	Z(ZY")				
	rise	(ZW)Z	rise oneself	Z(WZ)				
7	lift	(ZW)Z'	raise	Z(WZ')				
	hold	(ZW)Z"	hang	Z(WZ")				
	lower	(ZY)Z	go down	Z(YZ)				
8	bring down	(ZY)Z'	fall	Z(YZ')				
	put	(ZY)Z"	lie	Z(YZ")				
	group	(ZZ)Z	be found	Z(ZZ)				
9	sway	(ZZ)Z'	swing	Z(ZZ')				
	shake	(ZZ)Z"	vibrate	Z(ZZ")				
	Con	nplex infor	mational strings					
	Active		Passive					
	order	(XW)Y	submit	X(WY)				
1	convince	(XW)Y'	agree	X(WY')				
	allow	(XW)Y"	act	X(WY")				
2	abolish	(XY)W	resist	X(YW)				

	dissuade	(XY)W'	insist	X(YW')
	forbid	(XY)W"	do nothing	X(YW")
3	instruct	(XW)W	perceive	X(WW)
	teach	(XW)W'	fix	X(WW')
	enlighten	(XW)W"	remember	X(WW")
4	break off	(XY)Y	lose	X(YY)
	confuse	(XY)Y'	forget	X(YY')
	deceive	(XY)Y"	trust	X(YY")
5	inform	(XX)W	know	X(XW)
	declare	(XX)W'	understand	X(XW')
	tell	(XX)W"	apprehend	X(XW")
6	hide	(XX)Y	don't know	X(XY)
	conceal	(XX)Y'	guess	X(XY')
	keep silent	(XX)Y"	err	X(XY")
	raise	(XW)X	be proud	X(WX)
7	inspirate	(XW)X'	triumph	X(WX')
	love	(XW)X"	enjoy	X(WX")
	humiliate	(XY)X	be distressed	X(YX)
8	insult	(XY)X'	take offense	X(YX')
	detest	(XY)X"	suffer	X(YX")
9	think	(XX)X	live	X(XX)
	want	(XX)X'	plan out	X(XX')
	be able	(XX)X"	suppose	X(XX")

#### **USC** algorithm

The USC algorithm of semantic conversions is based on the axioms of USC algebra and is formed as an oriented graph. Nodes of the graph are represented in the form of the USC strings. The arcs are represented as USC algebra axioms. The solution of the intellectual problem can be viewed in the form of the route set succession of arcs. The algorithm of the problem solution is based on successive route from the initial situation to the goal one.

USC algorithm of semantic conversion as a classificatory oriented graph (block diagram)



Each module of the algorithm corresponds to six strings:

 $[\mathsf{ZWY} = (\mathsf{ZW})\mathsf{Y} \rightarrow \mathsf{Z}(\mathsf{WY})] ==> [\mathsf{ZWY'} = (\mathsf{ZW})\mathsf{Y'} \rightarrow \mathsf{Z}(\mathsf{WY'})] ==> [\mathsf{ZWY"} = (\mathsf{ZW})\mathsf{Y"} \rightarrow \mathsf{Z}(\mathsf{WY"})]$ 

**o** - ellipse indicates the conversion of the left part (string divergence):  $(XY)Z \rightarrow (XY)X$ 

 $\diamond$  - diamond indicates the conversion of the right part (string diffusion): ZWW  $\rightarrow$  ZZW or ZWW  $\rightarrow$  ZWZ

As a result of search we select three groups of strings:

Unary string	ZZW	lead in			
Double string	ZZW + ZZW	lead in + lead in = pump up			
Confluence of two strings	ZZW + ZWZ	lead in + hold = put in			

Here is an example of a questionnaire and the relevant vocabulary item. The vocabulary item "to create" is described by complex string ((XY)Z)((ZW)Y) - X by means of Y affects Z so that W exists. The questionnaire: Who "X" by means of what "Y" affects what "Z" creates what "W".

The strings introduced this way and denoting the initial and goal situations are placed on the relevant nodes of the graph drawing a route between them. The transfer algorithm reduced to correlation of the initial and the goal situations can be defined as a one-step inference, which has a form of a complex string.

Here is an example within our verb vocabulary. Suppose, in the user's text there is a sentence "The master restores a picture". The item "to restore" is a reference item addressing "to preserve". The following structure is suggested in the item:

((XY)Z)((ZW)Y") - X by means of Y affects Z so that W is preserved.

The user is asked to name the subjects denoted by symbols of the variables: "The master by means of some set of tools affects colors so that the picture is preserved". According to the axioms ZWY -> ZWW is equal to: preserve -> concentrate, join, compress; combine, contract, squeeze.

What follows is determined by addressing the list of typical effects in the knowledge base for tools for preserving and restoring pictures.

The typical effects are complex strings that form the base of classification of physical, chemical, geometrical, informational, and other effects. The difference between axioms and typical effects is that axioms are true irrespective of the value of the variables composing them, typical effects are not. The physical effect of magnetic separation is completely dependent on a value of variables such as a magnetic field and a metal when the USC string for the action "separate" is able to be filled with variety of domain independent values including values for the presented effect.

#### References

- Amarel S. On the Representation of Problems of Reasoning about Action. // Machine Intelligence, 1968, N3.
- Chomsky N. Lectures on Government and Binding, Dodrecht, 1981.

Lukasiewicz J. Elementy Logiki Matematycznej. Warszawa, 1958.

- Martynov V.V. USC-3: New Variant of a Language for Representing Knowledge and Effecting Calculations. In: Artificial Intelligence. Proceedings of the IFAC Symposium, Oxford-New-York-Toronto-Sydney-Francfurt, 1983.
- Melchuk I.A. Experience of theory of linguistic models "meaning-text". Moscow, 1974. (Russian).
- Montague R. Formal Philosophy. New Haven, London, 1974. Schank R.C. Conceptual Information Processing. Amsterdam, North-Holland, 1975.