To appear in *Dependency and Valency. An International Handbook of Contemporary Research*, Berlin: De Gruyter.

# THE MEANING-TEXT THEORY

*Sylvain KAHANE*

The goal of Meaning-Text theory (MTT) is to write systems of explicit rules that express the correspondence between meaning and text (or sound) in various languages. Apart from the use of dependency rather constituency, MTT can be characterized by the massive relocation of syntactic information into the lexicon—anticipating on that characteristic, most of the contemporary linguistic theories—and a transductive, rather than a generative, presentation, which favors the direction of (speech) synthesis, rather than analysis (see Section 3.3).

The Meaning-Text approach to language was put forward in Moscow, thirty-five years ago, by Žolkovskij and Mel'čuk (1965, 1967) in the framework of research in machine translation. Some of the central ideas of the Meaning-Text approach may be traced back to (Žolkovskij/Leont'eva/Martem'janov 1961) and (Žolkovskij 1964). During the following ten-year period, some twenty linguists contributed to the work on a Meaning-Text model of Russian around a nucleus composed of Mel'čuk, Žolkovskij and Apresjan (cf. Mel'čuk 1981 or Hartenstein/Schmidt 1983 for a fairly complete bibliography). A new group is now formed around Mel'čuk at the University of Montreal, where a dictionary and a grammar of French are being elaborated (Mel'čuk *et al.* 1984, 1988, 1992, 1999).

Presentations of MTT can be found in (Mel'čuk 1974, 1988a, 1997, Polguère 1998a, Weiss 1999, Milićević 2001, Kahane 2001c). This presentation is widely inspired by (Mel'čuk 1988a = DS), but integrates most of the recent developments of the theory; it contains also a tentative comparison with other frameworks. It is particularly oriented towards the formal aspects of the theory.

Contents

## 1. Meaning-Text theory postulates

MTT *postulates at first* that a natural language $\mathcal{L}$ is a logical device that establishes the *correspondence* between the set of possible *meanings* of $\mathcal{L}$ and the set of possible *texts* of $\mathcal{L}$. Meanings, as well as texts, are taken to be distinguishable entities forming an infinite countable set. They are formalized by symbolic representations: *semantic representations* for meaning and *phonetic representations* for text. The first postulate of MTT can be compared to the Chomskyan (1957) viewpoint on the language based on the characterization of acceptable sentences. It is formally equivalent to describe the correspondence between semantic and phonetic representations than to describe the set of all acceptable sentences of the language, provided that an acceptable sentence is described as the correspondence of a semantic and a phonetic representation of the sentence.

The *second postulate* of MTT is that each language $\mathcal{L}$ is described by a *Meaning-Text model* (MTM), that is, a symbolic model, including a finite set of rules, which defines the correspondence between the set of possible semantic representations of $\mathcal{L}$ and the set of possible phonetic representations of $\mathcal{L}$. "For a given meaning, this logical device must ideally produce all texts that, in the judgment of native speakers, correctly express this meaning, thus simulating *speaking*; from a given text, the device must extract all the meanings that, according to native speakers, can be correctly expressed by the text, thus simulating *speech understanding*" (DS, 44). As Nakhimovsky (1990, 3) notices, an MTM is nothing else than a formal grammar. Nevertheless, this term was not retained probably for at least two reasons: The dominating role of the lexicon (which is opposed to the grammar proper) and the transductive presentation of the system—an MTM is not formalized by a generative grammar, but by a kind of transducer (cf. Aho/Ullman 1972, 212ff, for a definition of sequence-to-sequence transducers).

Although the correspondence between meanings and texts is bidirectional, "MTT is developed and presented strictly in the *synthesis direction*: from meanings to texts" (DS, 46). The reason for this is that speaking is a more linguistic task than speech understanding. In particular, the problem of disambiguation, specific to understanding, cannot be solved without using extralinguistic knowledge and reasoning capabilities. Otherwise, the synthesis viewpoint gives a prominent position to the problem of lexical choices (which is one of the major preoccupations of MTT): for example, we say *to make* ⟨**do*⟩ *a mistake* ⟨*decision*⟩, but *to do* ⟨**make*⟩ *a favor* ⟨*one's job*⟩ (while, from the analysis viewpoint, *to make a mistake* presents neither a problem, nor a particular interest).

The correspondence between meanings and texts is many-to-many. All natural languages know *synonymy* (a meaning corresponds to many texts) and *homonymy/polysemy* (a text corresponds to many meanings). Synonymy is especially rich: a fairly complex sentence of 10 words can have thousands of paraphrases; the number of paraphrases is exponential as function of the number of words, and therefore a 20 ( $= 10 + 10$) words sentence can have millions (= thousands of thousands) of paraphrases. "One can even say that a natural language is essentially a system designed to produce a great many synonymous texts for a given meaning" (DS, 48).

"The words *meaning* and *text* are used here as purely technical terms whose content must be specified. 'Meaning' stands for 'invariant of synonymic transformations' and refers only to information conveyed by language; in other words, it is whatever can be extracted from or put into an utterance solely on the basis of linguistic skills, without recourse to encyclopedic knowledge, logic, pragmatics or other extralinguistic abilities” (DS, 44). In other words, ‘meanings’ are purely linguistic and language-specific objects. “‘Text’ stands for ‘the physical form of any utterance’ and refers to all linguistic signals (words, phrases, sentences, etc.)” (*ibid.*). To this day, all studies in MTT have been limited to sentences.

An MTM is just a component of a global model of human linguistic behaviors. The correspondence between phonetic representations (which are discrete representation) and real sounds and the correspondence between semantic representations and cognitive representations (= discrete human representations of the continuous reality) goes beyond the scope of MTT. The former is the subject of acoustics and articulatory phonetics. The latter “is the subject of a science that does not yet exists as a unified discipline and is distributed among philosophy, psychology, cognitive science, logic, documentation, artificial intelligence, etc. [...] This component must ensure the interaction between the cognitive representation, the internal thesaurus of the Speaker, the pragmatics of a given situation and the like, in order to produce the ‘meaning’ of a future utterance” (DS, 46ff).

As most other linguistic theories, MTT postulates two intermediate-level representations between the semantic representation (= meaning) and the surface-phonological representation (= text): a syntactic representation and a morphological representation (it is the *third postulate* of the theory). All levels, except the semantic one, are further split into deep- and surface levels, the former oriented towards the meaning, and the latter, towards the physical form (the text). This gives us a total of seven levels of representation.

Semantic representation (SemR), or the meaning
⇕ } semantics
Deep-syntactic representation (DSyntR)
⇕ } deep syntax
Surface-syntactic representation (SSyntR)
⇕ } surface syntax
Deep-morphological representation (DMorphR)
⇕ } deep morphology
Surface-morphological representation (SMorphR)
⇕ } surface morphology
Deep-phonological representation (DPhonR)
⇕ } phonology
Surface-phonological representation (SPhonR), or the text

**Fig. 1.** Levels of utterance representation
and modules of the Meaning-Text theory

[In Fig. 1, acronyms used by Mel’čuk in all his publications have been introduced. They will not be used here.]

Many contemporary theories assume syntactic and morphological levels. The particularity of MTT is to consider them as *intermediate levels* between the semantic level (the meaning) and the phonetic level (the text). Thus the correspondence between meanings and texts is completely modular: a correspondence between the semantic and deep-syntactic levels, a correspondence between the deep-syntactic and surface-syntactic levels, a correspondence between the surface-syntactic and deep-morphological levels, etc. An MTM is divided into six modules, each taking care of the correspondence between two adjacent levels, giving us a stratificational system, like Lamb's (1966) system (Fig. 1).

As a result, each module ensures the correspondence between representations of adjacent levels, but is not dedicated, contrary to a generative grammar, to caracterizing the representations it handles. Consequently, a syntactic or a morphological representation is not caracterized by a particular modular, but simply by the fact that it can be a possible intermediate representation between a well-formed semantic representation and a corresponding phonetic representation. In other words, well-formedness rules of the syntactic or morphological representations can be deduced from an MTM, but it is not the goal of an MTM to enounce them. As a consequence, MTT does not give any primacy to syntax.

## 2. Utterance representation at different levels

I will now present the representations at the four deeper levels (semantic, deep-syntactic, surface-syntactic, deep-morphological), which are the levels concerned by dependency and valency. The lexicon and the correspondence rules of the three considered modules will be presented afterwards.

### 2.1. Semantic representation

**2.1.1.** In the Meaning-Text approach, a semantic representation specifies the meaning of a set of synonymous utterances, i.e. utterances having the *same meaning*. Meaning is taken to be an invariant of synonymic transformations between such utterances. Thus the concept of "meaning" is based on the concept of "same meaning". The precision with which the synonymy of two texts is established must be allowed to vary as a function of the nature of the task (a journalistic description vs. a legal formulation) and can be captured by the degree of specification of the semantic representation. Mel'čuk (DS, 52) adds that "a certain degree of approximation in the semantic is necessary, if we want to obtain linguistically interesting results." Mel'čuk (2001, 15ff) even says: "During the process of sentence construction (= synthesis), lexical and syntactic choices carried out by the Speaker very often lead to the modification of the starting meaning, i.e. of the initial semantic representation, making it more precise and specific: the lexical units bring with them additional nuances of meaning that have not been present in the initial semantic representation. MTT tries to model this phenomenon; as a result, quite often the following situation obtains: Suppose that the synthesis starts with the representation ‹$\sigma$› and produces sentences $S_1, S_2, \ldots, S_a$; the sentences having as their common source the semantic representation ‹$\sigma$› are considered to be synonymous. Now if we analyze these sentences semantically, the semantic ‹$S_1$›, ‹$S_2$›, ...,‹$S_a$› obtained from this process may well be different from each other and from the initial semantic representation ‹$\sigma$› ! [...] The initial semantic representation is taken to be rather approximate—it need not necessarily fully specify the meaning of the sentences that can be obtained from it. The meaning can become more precise—or less precise—in the course of its lexicalization and syntacticization."

By the primordial role it gives to synonymy, MTT is not so far from the generative-transformational approach (Chomsky 1965), even if the latter does not clearly identify an object as meaning: a deep-structure is rather a representative of a class of equivalent surface-structures than an object of a different level. But in sharp contrast to the generative-transformational approach, which only deals with the syntactic synonymy, MTT emphasizes lexical synonymy. Moreover, semantic representations are not only of use to represent the meaning of a sentence, but they are also used to represent the meaning of a word as it is done in a dictionary.

Unlike most semantic models (particularly truth-conditional ones), "the analysis of meaning itself goes beyond the scope of MTT: it does not distinguish "normal" meanings from absurdities, contradictions or trivialities. Discovering that something is stupid or absurd or detecting contradictions is by no means a linguistic task" (DS, 47).

The semantic representation represents the meaning of an utterance regardless of its surface form: distribution among words or phrases is ignored in a semantic representation as is its expression through different devices (lexemes, suffixes, syntactic constructions or prosody).

**2.1.2.** The main structure of the semantic representation, the *semantic structure* (or *graph* or *network*), reflects the meanings of the words of the sentence and their organization. Formally, it is a connected directed graph. Each *node*, or vertex, of a semantic graph is labeled by a *semanteme*, i.e. a language-specific semantic unit which corresponds to one particular word-sense or, more precisely, to the *signifié* of a (desambiguatized) lexical unit. (Consequently, semantemes, as well as lexemes, should be accompanied by a numerical index as it is done in a dictionary.) Semantemes are written between single quotes: ‹semanteme›. From a mathematical point of view, a semanteme is a functor, whose arguments are called the *semantic actants* of the semanteme ("functors" without arguments are called *semantic names*). The link between a semanteme and one of its actants is represented by an *arc*, or arrow, and is called a *semantic dependency*. The arc pointing on the $i$th argument, or *semantic actant*, of a semanteme is labeled by $i$. Arc labels are thus strictly distinctive and asemantic (even if the order in which actants are numbered is not arbitrary and follows, roughly speaking, the syntactic oblicity). A semantic graph (with some additional communicative features concerning the rheme-theme partition; see section 2.1.7) is given in Fig. 2.

This semantic representation can be expressed by a lot of (quasi-)synonymous sentences:

(1) **a.** *John feels no revulsion at the sight of a dead animal.*
  **b.** *John does not feel revulsion in seeing a dead animal.*
  **c.** *John experiences no revulsion at the sight of some dead animal.*
  **d.** *John experiences no revulsion when he sees some dead animal.*
  **e.** *John is not revolted by the sight of a dead animal.*

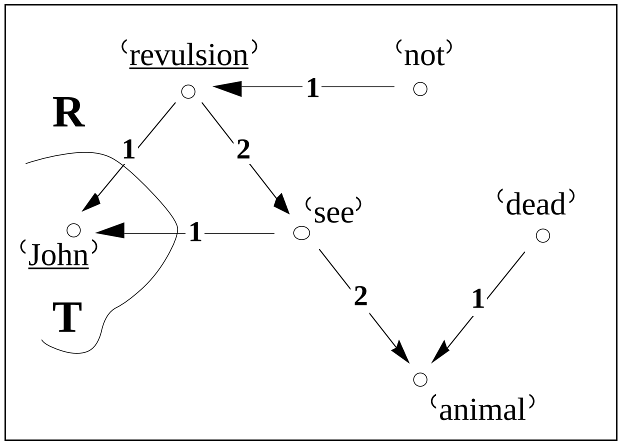



**Fig. 2. A semantic representation**

**2.1.3.** Žolkovski & Mel'čuk (1965) are probably the first to formalize the distinction between actants and modifiers. Except for some cases such as raising or *tough*-movement, as noted by Tesnière (1959, 42), when a word B syntactically depends on a word A, there is a semantic dependency between ‹A› and ‹B›. But, what Tesnière did not see is that this dependency can be directed from ‹A› to ‹B› as well as from ‹B› to ‹A›. For instance, in the phrase *a small river*, *small* syntactically depends on *river* and, because the smallness is a property of the river, ‹river› acts as an argument of the predicate ‹small›; conversely, in the clause *the river swelled*, *river* syntactically depends on *swelled* and, because the swelling is a property of the river, ‹river› acts as an argument of ‹swell›. When the semantic dependency has the same direction as the syntactic dependency, B is an *actant* of A (*river* is an actant of *swelled* in the *the river swelled*), while, when the syntactic and the semantic dependencies have opposite direction, B is a *modifier* of A (*small* is a modifier of *river* in *a small river*).

**2.1.4.** Semantemes can be decomposed in terms of simpler semantemes (Mel'čuk 1988b), except for a small set of semantemes—the *semantic primitives*. In principle, semantic decomposition can be carried on until semantic primitives, which are indecomposable semantemes, are reached. This ensures the non-circularity of the dictionary (Mel'čuk 1989). In (Wierzbicka 1972, 1980 and 1985), a set of about fifty semantic primitives is proposed and argued for.

The decomposition of a semanteme is its *lexicographic definition*. "The meaning of language signs reflects *naive* conceptions of objects, properties, actions, progresses, events, etc. [...] The task of a lexicographer [...] is to uncover the naive worldview in lexical meanings and reflect it in his system of definitions" (Apresjan 1974: 56ff, from Nakhimovsky 1990). For instance, the semanteme ‹revulsion› can be decomposed as follows (Mel'čuk 1998, 52):

‹X's revulsion for Y› ≡ ‹X's (strong) negative emotion about Y similar to what people normally experience when they are in contact with something that makes them sick and such that it causes X to want to avoid any contact with Y›

This decomposition can be represented by a semantic graph (Fig. 3). Some communicative specifications are needed: in particular, following Polguère (1990), the semanteme which "summarizes" the meaning of the decomposed semanteme, the generic meaning, called the (communicatively) *dominant node*, is underlined.

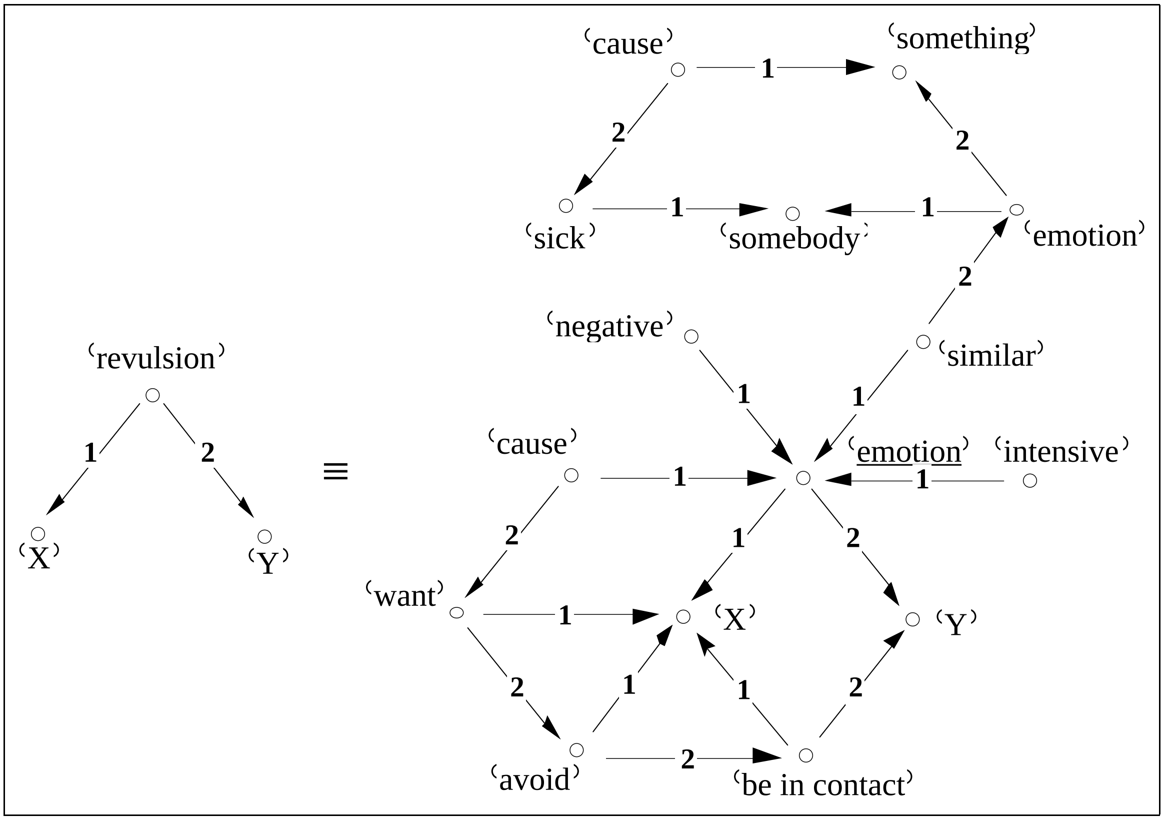



**Fig. 3. Semantic decomposition of** ‹revulsion›

The number and the roles of the actants of a semanteme are determined by its decomposition. For instance, in accordance with its decomposition, the semanteme ‹revulsion› has two semantic actants. The fact that the meaning ‹X kills Y› can be roughly decomposed into ‹X causes Y to die› indicates that the first actant ‹X› of ‹kill›, which is the first actant of the primitive meaning ‹cause› in the decomposition, is a causer. The determination of the "semantic role" of the second actant ‹Y› needs a deeper decomposition of ‹kill› (and consequently of ‹die›).

The correspondence between a semanteme and its decomposition is a rule of the MTM encoded in the dictionary. Rules of semantic decomposition define an equivalence relation, noted $\equiv$, on semantic graphs (and more generally on semantic representations). (The relation $\equiv$ defines correspondences between structures of the same level; it must not be confounded with a correspondence between structures of two adjacent levels, noted $\Leftrightarrow$.) Therefore the representation of a sentence meaning (viewed as an invariant of synonymy) is not exactly a semantic representation but rather an equivalence class of semantic representations (for the equivalence relation $\equiv$).

Semantic decompositions are particularly useful to capture synonymy of sentences whose underlying semantemes do not correspond one to one. For example, the synonymy of sentences (2a-c) can be established in the following way: ‹hurry› ≡ ‹walk quickly› and ‹cross› ≡ ‹walk across› (Milićević 1999).

(2) **a.** *Peter walked quickly across the road.*

**b.** *Peter hurried across the road.*

**c.** *Peter crossed the road quickly.*

It seems that semantic decomposition rules are not sufficient to define ≡ and that some other types of rules are needed. Consider for instance:

(3) **a.** *John's pushing of Mary caused her to fall.*

**b.** *John caused Mary's fall by pushing her.*

Sentences (3a) and (3b) are synonymous (barring communicative differences) and consequently their semantic graphs should be equivalent. But in the semantic structure of (3a), the first actant of ‹cause› is ‹push›, while, in the semantic structure of (3b), ‹John› and ‹push› are related to ‹cause› by way of a biactancial semanteme ‹means› (expressed by *by + -ing*). The equivalence of these two configurations cannot be obtained without the use of a rule of equivalence such as the rule of Fig. 4 (with ‹P› = ‹push›, ‹X› = ‹John› and ‹Y› = ‹fall›). (Another solution, defended by Mel'čuk, consist in assuming two different semanteme ‹cause1› and ‹cause2› and considering the rules of Fig. 4 as the semantic decomposition of ‹cause2› according to ‹cause1›.)

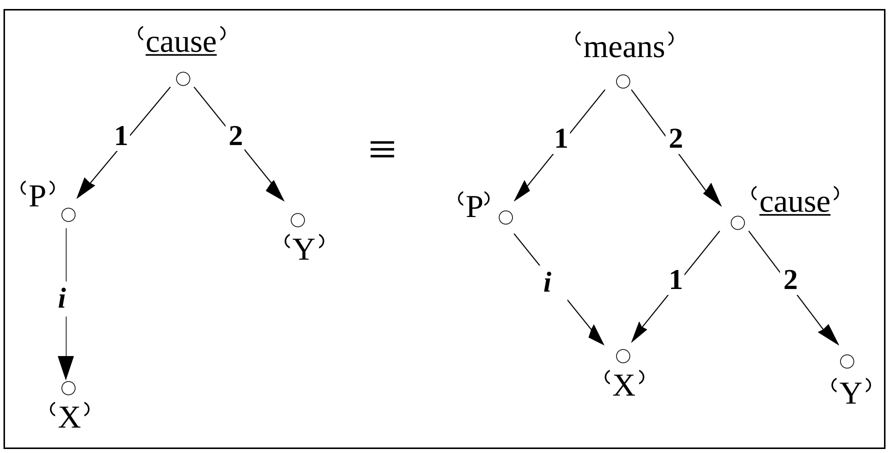



**Fig. 4. A rule of equivalence which is not a rule of decomposition**

**2.1.5.** In the semantic graph of Fig 2, *grammemes* (= inflectional meanings) have been ignored. Mel'čuk proposes to represent them by the same type of device as lexical meanings. Rather than to introduce nodes labeled by grammemes, Mel'čuk represents them with the help of lexical semantemes: for example, one of the meanings of the present tense in (1) could be represented roughly as ‹at any time›.

It is important to notice that not all grammemes are part of what the Speaker wants to say, but are rather part of what the language requires him to say. For instance, compare the two synonymous sentences (4a-b) where the choice of a completive clause (in (4a)) forces the Speaker to introduce the tense on ‹see›, while the choice of an infinitive (in (4b)) does not.

(4) **a.** *I hope I'll see you.*

**b.** *I hope to see you.*

**c.** *I hope he'll see you.*

And if the subjects of *hope* and *see* are not co-referential (as in (4c)), the Speaker must use a completive clause and thus has no other choice but to express the tense on ‹see›.

In the spirit of the Meaning-Text approach, a semantic representation would contain what the Speaker *wants* to say. Therefore a semantic graph needs not necessarily contain configurations of semantemes for each grammeme required. And when it contains one it would be an approximate expression, e.g, ‹before now› or ‹accomplished› for the past tense. A real synthesis system might be able to compute the necessary grammemes (at subsequent stages of treatment) when the expression of an inflectional category is needed.

**2.1.6.** A semantic graph can be encoded in a more logical style. The translation of a semantic graph into a formula of predicate calculus needs to introduce a variable for each node in the graph (except for the nodes labeled by a grammatical semanteme). This variable represents the node and serves as an argument for any dependency pointing to the node. Adding the variables $x$, $y$, $p$, $q$, $e$ and $w$ for the lexical semantemes ‹John›, ‹animal›, ‹dead›, ‹not›, ‹see› and ‹revulsion›, the semantic graph of Fig. 2 is thus translated into the formula (5):

(5) $x$ : ‹I› $\wedge$ $y$ : ‹animal› $\wedge$ $p$ : ‹dead›$(y)$ $\wedge$ $e$ : ‹see›$(x, y)$ $\wedge$ $w$ : ‹revulsion›$(x, e)$ $\wedge$ $q$ : ‹not›$(w)$

The variable attributed to the node can be incorporated in the valency of the semanteme (the semanteme is reified), giving us the equivalent formula (6):

(6) ‹I›$(x)$ $\wedge$ ‹animal›$(y)$ $\wedge$ ‹dead›$(p, y)$ $\wedge$ ‹see›$(e, x, y)$ $\wedge$ ‹revulsion›$(w, x, e)$ $\wedge$ ‹not›$(q, w)$

Although they seems similar, the semantic representations of MTT must be distinguished from the semantic representations of the semantics descended from the Fregean logic, such as the Discourse Representation Theory (Kamp/Reyle 1993). In MTT, the semantic representations does not represent the state of the world denoted by the meaning, but the meaning itself. In particular, the variables that has been introduced in the reification do not refer to objects of the world (entities or events) as in fregean logic. The variables refers here to the semantemes, that is to the meanings of the words. Let us consider an example: In the meaning of *a big ant*, the semanteme ‹big› is a unary predicate whose argument is the semanteme ‹ant›, and not the referent of ‹ant›. When we speak about *a big ant*, we do not want to say that the referent of *ant* is big in itself (nothing is big in itself), but that it is big as an ant. Perhaps it is even clearer for *a big boss*. In this case also, one do not want say that the referent of *boss* is big, but that something in the meaning ‹boss› is big: If a ‹boss› is ‹someone who makes decisions›, a ‹big boss› is ‹someone who makes big decisions›. Moreover, the semanteme ‹big› can be the argument of another semanteme, like in *a very big ant* or *a bigger ant than my finger*, which needs to introduce a variable for ‹big› which serves as argument for ‹very› or ‹more (than my finger is big)› in the MTT semantic graph, without having to assume that *big* introduces a discourse referent.

There is another difference between MTT semantic representations and logical calculus formulas adopted as representation by truth-conditional semantic theories. Indeed, contrary to the MTT's practice, in logicians' representations, some semantemes are not interpreted by logic predicates but rather by quantifiers, connectors or more complex operators. If this differentiation of quantifiers is certainly necessary for logical deduction, it seems not to be needed for paraphrasing and translation (especially as only a small part of quantifiers are described by logicians). Nevertheless, it must be noted that the scope of a quantifier is not directly encoded in standard MTT semantic representations, where quantifiers are monoactancial semantemes whose argument is the quantified semanteme . Therefore nothing in the semantic graph of *All the men are looking for a cat* indicates whether or not *a cat* is in the scope of *all*. Mel'čuk (2001) considers that this must be indicated in the semantic communicative structure, but no serious study supports this idea. Polguère (1992; 1997) proposes to encode quantifiers as biactantial semantemes whose second argument is the scope, but having semantic dependencies pointing to areas of the graph (rather than a single node) is a serious complexification of the formalism of semantic representations. Dymetman/Copperman 1996 proposes a representation intermediate between a semantic graph and a logical calculus formula.

**2.1.7.** I will now introduce the other components of a semantic representation: the semantic communicative structure, the rhetoric structure and the referential structure. The *semantic communicative structure* specifies the manner in which the Speaker wants his message to be organized: what is said and what is spoken about; what should be explicitly asserted and what can be only presupposed; what should be emphasized and what can be left in the background; etc.

Formally, the communicative structure is encoded by spotting out some areas of the semantic graph and labeling each of them with a *communicative marker*. As Polguère 1990 demonstrates, in each communicative area, it is necessary to indicate (e.g. by underlining) the communicatively *dominant node*, that is the node which summarizes the semantic content of the area.

Mel'čuk (2001, 77ff) proposes the following set of communicative markers, corresponding to eight *communicative categories* (communicative categories are considered by Mel'čuk as *oppositions*, and therefore a Neutral or Irrelevant value is added in certain categories, although this value never appears in semantic representations):

1. *Thematicity*: Rheme (i.e. Comment) vs. Theme (i.e. Topic) vs. Specifier;
2. *Giveness*: Given vs. New;
3. *Focalization*: Focalized;
4. *Perspective*: Foregrounded vs. Backgrounded;
5. *Emphasis*: Emphasized;
6. *Presupposedness*: Presupposed vs. Asserted;
7. *Unitariness*: Unitary vs. Articulated;
8. *Locutionality*: Signaled vs. Performed vs. Communicated.

The semantic communicative structure will not be defined but just exemplified and commented upon. Consider:

(7) **a.** *John feels no revulsion at the sight of a dead animal.*

**b.** *The sight of a dead animal does not revolt John.*

**c.** *As for John, he feels no revulsion at the sight of a dead animal.*

**d.** *It's John who feels no revulsion at the sight of a dead animal.*

**e.** *What does not revolt John is the sight of a dead animal.*

Every message is necessarily intended to say something (≈ Rheme) about something (≈ Theme) (only rhematic sentences such as *It rains* are also possible). The most neutral thematic division of (7a) is ‹John› as Theme and ‹feel no revulsion at the sight of some dead animal› (with ‹revulsion› as dominant node) as Rheme (see Fig. 2, where the Theme area is marked by a **T** and the Rheme area by an **R**), because in English the Theme tends to be expressed as subject. With another distribution of thematicity, a different expression can be obtained: The same semantic graph with ‹John sees a dead animal› as Theme and ‹John feels no revulsion› as Rheme should be rather expressed by (7b).

Giveness concerns what the Speaker believes is or is not in the Addressee's active consciousness. "Roughly speaking, in languages having articles (i.e. grammatical expression of definiteness), the Given is regularly (but by no means always) characterized as definite and the New, as indefinite. In *The book was on the table*, both ‹book› and ‹table› are Given; but *There was a book on the table*, only ‹table› is Given, while ‹book› is New" (Mel'čuk 2001, 75). Giveness needs a further study and the links between it and the referential structure (see below) must be elucidated.

Thematicity and Giveness are universally obligatory, in the sense that in any language, a semantic representation in which values of these categories are not explicitly specified cannot be uniquely turned into a sentence (although in a real process of synthesis the semantic representation can be underspecified and the Speaker might have to choose between different possible communicative structures or parts thereof when trying to synthesize the sentence he wants).

A Focalized element is "a part of a meaning which the Speaker presents as being *logically* prominent for him—or, in other words, as being in the focus of his attention" (Mel'čuk 2001, 175). Typical devices expressing focalization in English are left dislocation, clefting and pseudo-clefting: in (7c), ‹John› is a Focalized Theme, while in (7d), it is a Focalized Rheme; in (7e), ‹John feels no revulsion› is a Focalized Rheme.

Other communicative categories correspond to their standard definition and do not need particular comments. Let us just note that unitariness concerns the way in which the Speaker presents a complex events (for instance, KILL vs. CAUSE TO DIE).

Melčuk (2001, 58) points out that "there is no logical necessity to formally separate genuinely semantic and communicative information: they *need not* be distinguished. It is technically possible to represent all communicative choices within the semantic graph by the same means as are used to represent propositional meanings. In other words, communicative choices can be represented by some special semantemes introduced just for this purpose—such as ‹be the theme of›, ‹be the rheme of›, ‹be given›, ‹be emphasized›, etc. [...] In other words, we can introduce into the semantic graph the following obligatory functor, which constitutes, so to speak, an illocutionary frame: ‹About X I tell you Y› (Sgall 1974, 71). Here, ‹about X› represents the Theme and ‹I tell you that Y› the Rheme.

In much the same vein, Wierzbicka (1975; 1978, 124) has proposed that the Theme (= Topics, in her terms) be specified by a genuine semantic component: ‹thinking of X, I say: ...›.” (It must nevertheless be noted that semantemes pointing to the dominant nodes of the rheme and the theme do not suffice to indicate the limits of the rheme and theme, particularly if the semantic graph has cycles, and consequently does not exempt us from spotting out areas.) Even if there is no necessity, it is still preferable to separate what can be separated: “the semantemes reflect the world—or rather the conceptualization of the world by the Speaker, i.e. what he *wants* to express; the communicative elements reflect the Speaker’s wishes concerning *how* to express it, that is, how to structure the semantemes” (*ibid.*).

**2.1.8.** The *rhetoric structure* “encodes the ‘artistic,’ or ‘aesthetic,’ intentions of the Speaker (whether he wants his utterance to be neutral, ironic, pathetic, humorous, etc.) and his stylistic choices” (Mel’čuk 2001, 13). Although theoretically provided by MTT, the rhetoric structure has unfortunately never been developed and the relevant information simply appears in the lexical entries of the MTT dictionaries, as it is done in traditional dictionaries.

**2.1.9.** The *referential structure* has been mentioned only recently (Mel’čuk 1997, 21; no mention in DS) and has never been developed, whereas it is the pillar of any truth-conditional semantic theory. According to Mel’čuk (2001, 12), the MTT semantic graph “is aimed at representing exclusively propositional, or situational, meaning of a family of more or less synonymous utterances. In other words a semantic graph mirrors a situation—a state of affairs in the real or an imaginary world [...], and it does so by mirroring situations as they are reflected in speech. This type of meaning is easily expressed by logical propositions; hence the name *propositional meaning*.” But the semantic graph is not an invariant of the “propositional meaning” without a referential structure. For instance, consider:

(8) **a.** *The linguists who were tired stopped.*

**b.** *The linguists who stopped were tired.*

The sentences (9a) and (9b) have the same semantic graph (inflectional meanings included) (‹stop›$\xrightarrow{1}$‹linguist›$\xleftarrow{1}$‹tired›), but they are not synonymous: they differ in communicative structure (the semantic theme of (8a) is ‹ <u>linguist</u> › $\xleftarrow{1}$ ‹tired›, while (8b)’s one is ‹ <u>linguist</u> › $\xleftarrow{1}$ ‹stop›), but especially in referential structure. In (8a), the referent of ‹linguist› is determined by the area ‹ <u>linguist</u> › $\xleftarrow{1}$ ‹tired›, while in (8b) it is determined by the area ‹ <u>linguist</u> › $\xleftarrow{1}$ ‹stop›. (Note that Fregean logic used a significantly different device to encode the difference of “propositional meaning” between (8a) and (8b): (8a) corresponds to the formula $\forall x$ [(‹linguist›($x$) $\wedge$ ‹tired›($x$)) $\rightarrow$ ‹stop›($x$)] and (8b) to the formula $\forall x$ [(‹linguist›($x$) $\wedge$ ‹stop›($x$)) $\rightarrow$ ‹tired›($x$)].)

## 2.2. Deep-syntactic representation

**2.2.1.** The main structure of the deep-syntactic representation, the *deep-syntactic structure* or *tree*, is an unordered dependency tree, whose nodes are labeled with *semantically full lexemes*. Semantically empty lexemes, such as governed prepositions and conjunctions, as well as auxiliary verbs, are introduced only in the surface-syntactic structure. Each lexeme comes with appropriate semantically full grammemes, such as number and definiteness for nouns or voice and mode-tense-aspect for verbs. Syntactic grammemes imposed by government and agreement, such as case for nouns, person and number for verbs or gender, number and case for adjectives, are excluded. Lexemes are put in upper case and grammemes, in subscript: $\mathrm{LEXEME}_{\mathrm{grammemes}}$. The arcs, or *branches*, of a deep-syntactic structure are labeled with a very small set of deep-syntactic relations. Actantial relations are just numbered by increasing oblicity: I for the most salient actant, II for the next, etc. Three other relations are considered: an *attributive* relation, noted ATTR, for all kinds of modifiers (circumstantials and attributes); a *coordinative* relation, noted *COORD*, for all conjoined constructions (*Mary, John and Peter*: MARY $\xrightarrow{COORD}$ JOHN $\xrightarrow{COORD}$ AND $\xrightarrow{II}$ PETER); and an *appenditive* relation, noted APPEND, for parentheticals, interjections, direct addresses, and the like (*Naturally,* $\xleftarrow{APPEND}$ *[he] neglects nothing*; *[Where] are you going,* $\xrightarrow{APPEND}$ *Kathleen ?*). Kahane/Mel'čuk 1999 introduces another relation: $\mathrm{ATTR}_{\mathrm{QUAL}}$, in order to distinguish qualificative from restrictive modification, while Kahane 1998 argues for a specific relation for demoted actant (such as the agent complement).

Deep-syntactic structure of (1a) is presented Fig. 5. The dashed bidirectional arrow belongs to the *deep-syntactic anaphoric structure*, which indicates the coreference between nodes corresponding to a same semantic node.

The deep-syntactic representation contains also a *deep-syntactic communicative structure*, similar to the semantic one, and a *deep-syntactic prosodic structure*, which encodes such semantic prosodies as "question vs. affirmation, menace vs. indignation, irony, doubt, tenderness, secretiveness and the like" (DS, 67).

**2.2.2.** Some remarks are necessary concerning the lexemes appearing in the deep-syntactic structure, that is *deep lexemes*. On the one hand, *phrasemes* (= idioms = multilexical phraseological units, e.g. PULL SOMEONE'S LEG), which are semantically a whole, label a single node. On the other hand, due to the fact that only a small set of deep-syntactic relations are considered, some *fictitious lexemes* are needed to encode semantically non empty syntactic constructions (such as postposing the modifying numeral to the noun in Russian, which roughly means ‹approximately›: *sto metrov* ‹100m› vs. *metrov sto* ‹approximately 100m›).

Lastly, an important specificity of the Meaning-Text approach is the concept of *lexical function*, first introduced by Žolkovskij/Mel'čuk 1965 and 1967. Lexical functions are used to describe paradigmatic and syntagmatic relations between lexemes, that is derivation and collocation ("derivation" is used in a broad sense, not limited to morphological derivation). A lexical function, as its name indicates, is encoded by a mathematical function whose arguments (= keywords) and values are lexical units. MTT uses about sixty standard simple lexical functions; simple lexical functions can be combined into complex expressions which operate as simple ones. Here are some examples.

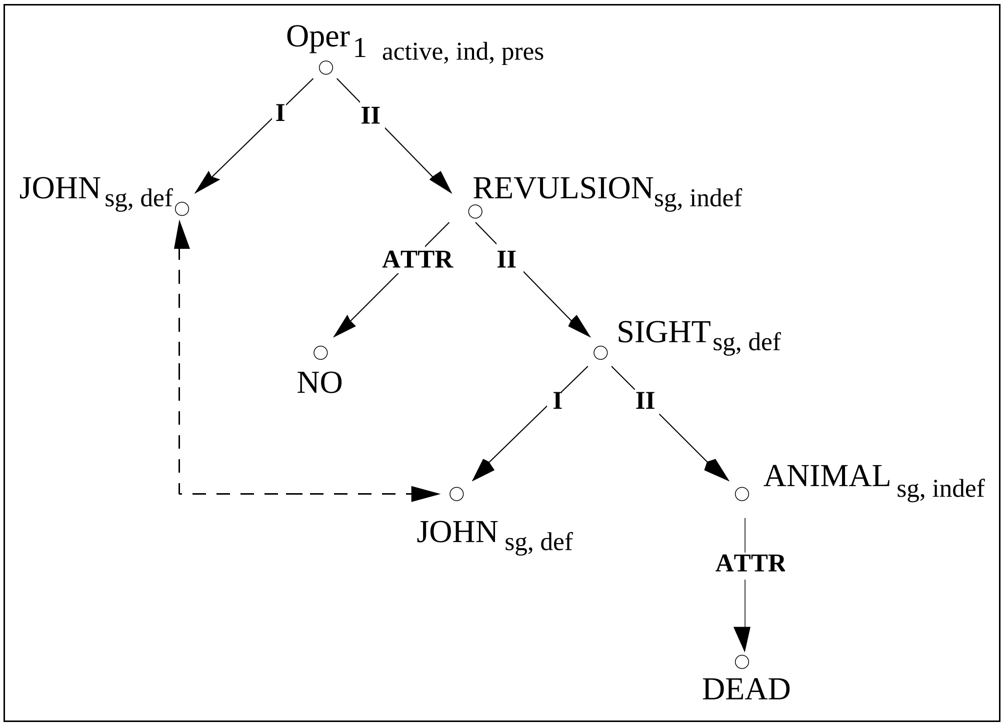



**Fig. 5. The deep-syntactic structure of (1a)**

**Magn** stands for a collocational modifier that means ‹very›, ‹to a high degree› (= intensifier):
**Magn**(*belief*) = *staunch*
**Magn**(*bore*$_N$) = *crashing*
**Magn**(*deserve*) = *richly*
**Magn**(*work*$_V$) = *as a Trojan, one's guts out*

**Oper**$_i$ stands for a light verb such that its 1st deep-syntactic actant is the $i$th semantic actant of the keyword and the keyword itself is its 2nd actant:
**Oper**$_1$(*blow*$_N$) = [*to*] *deal* [*ART* ∼ *to N*]
**Oper**$_1$(*order*$_N$) = [*to*] *give* [*ART* ∼ *to N*]
**Oper**$_1$(*support*$_N$) = [*to*] *lend* [∼ *to N*]
**Oper**$_1$(*resistance*$_N$) = [*to*] *put up* [*ART* ∼], [*to*] *offer* [*ART*/∅ ∼]
**Oper**$_2$(*blow*$_N$) = [*to*] *receive* [*ART* ∼ *from N*]
**Oper**$_2$(*control*$_N$) = [*to*] *be* [*under N's* ∼]
**Oper**$_2$(*resistance*$_N$) = [*to*] *meet* [*ART* ∼], [*to*] *run* [*into ART* ∼]

**Real**$_i$ stands for a collocational verb that means, roughly, ‹fulfill the requirement of› (and such that its 1st deep-syntactic actant is the $i$th semantic actant of the keyword and the keyword itself is its 2nd actant).
**Real**$_1$(*accusation*) = [*to*] *prove* [*ART* ∼]
**Real**$_1$(*bus*) = [*to*] *drive* [*ART* ∼]
**Real**$_1$(*illness*) = [*to*] *succumb* [*to ART* ∼]

**Real**$_2$(*bus*) = [*to*] *ride* [*on ART* ~]
**Real**$_2$(*law*) = [*to*] *abide* [*by ART* ~]
**Real**$_2$(*hint*) = [*to*] *take* [*ART* ~]

Values of lexical functions, which are stored in the dictionary entries of their keywords, are introduced only in the surface-syntactic tree. More information on lexical functions can be found in Mel'čuk/Clas/Polguère 1995, Mel'čuk 1998, Steele 1990, Wanner 1994, 1996 and most of the publications concerning the ECD (see below).

**2.2.3.** The deep-syntactic representation is certainly the least defined level of representation of MTT.

First, while semantic and surface syntactic dependencies have been relatively well defined (DS, 105ff; [article 19 of the Handbook]), the deep-syntactic dependencies have not been sufficiently characterized (see nevertheless Žolkovskij/Mel'čuk 1967). Consequently in case of discontinuities between the semantic and the surface syntactic level, no criteria are available to decide what dependencies to choose. For instance, in *Peter is drinking a glass of beer*, ‹beer› is the semantic argument of ‹drink› (‹Peter is drinking beer, which is in a glass›), while GLASS is the surface syntactic dependent of DRINK. What happens on the deep-syntactic structure ? Mel'čuk (1974) proposes that GLASS depends in BEER by an ATTR relation, because GLASS is here a quantifying expression, but there is no general principle for justifying the decision.

Second, the deep-syntactic relations have never been argued for. Why are actancial relations only classified according to their oblicity degree ? For instance, the actancial relation II covers just as well direct objects, indirect objects, prepositional complements or agent complements; conversely, an indirect object can be II or III expecting on the presence/absence of a direct object. In view of this, it seems strange to say, as Mel'čuk (1988, 63) does, that "each deep-syntactic relation stands for a family of specific syntactic constructions of particular languages, representing them in a generalized way."

Third, which grammemes must be present in the deep-syntactic tree ? For instance, must finite and non-finite verbs be distinguished ? (In many cases, the choice between finite and non-finite form of a verb depends on the government pattern of the governor, which is considered in the transition from deep-syntactic to surface-syntactic level.) And if finite and non-finite verbs are not distinguished, must grammemes of tense be always present ?

**2.2.4.** A last point: the deep-syntactic structure is not exactly a tree, but rather a rooted *dag* (= direct acyclic graph); see, e.g., (Aho & Ullman 1972, 39) for a definition. The reason is that pronominalization is performed in the transition from the deep- to the surface-syntactic tree and a tree with its co-refential structure is formally equivalent to a dag (identifying all co-referential nodes). The representation by a dag is justified for at least two reasons. First, if a set of co-referential nodes of the tree have dependents, these dependents cannot be attached to one node rather to another one and they consequently "float" between all their potential governors. The question does not arise in a dag where all these potential governors are represented by a single node (cf. Fig. 6).

Second, the rules of pronominalization cannot treat independently the different co-referential nodes of a deep-syntactic tree. Therefore, rather than to separate the different

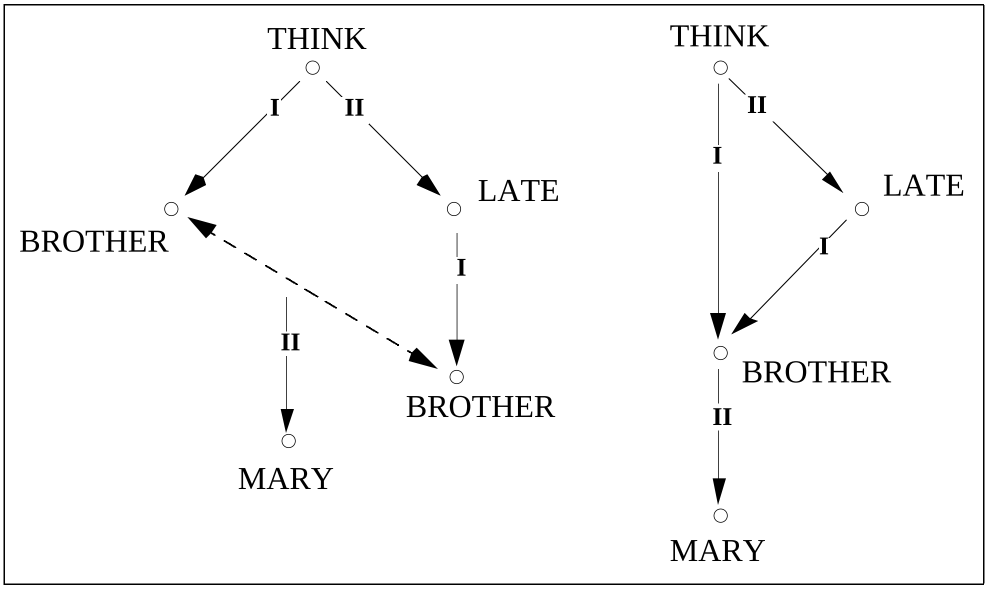



**Fig. 6. Deep-syntactic tree vs. dag**
**for** *Mary's brother thinks he is late*

nodes of a given relation of coreference, it is simpler to apply to each node of the deep-syntactic dag that have more than one governor a single pronominalization rule, which separates the deep-syntactic node into as many surface syntactic nodes as necessary and decides which of these nodes will not be pronominalized.

Kahane (2000) argues for a representation intermediate between the semantic graph and the deep-syntactic tree based on a dag.

## 2.3. Surface-syntactic representation

A *surface-syntactic representation* specifies the organization of a particular sentence, much as a deep-syntactic representation does; but unlike the latter, a surface-syntactic representation is geared to the surface form: to morphologization and linear ordering of the nodes (= of wordforms).

The surface-syntactic representation consists of four structures corresponding to the four representations of the deep-syntactic representation. The main structure of the surface-syntactic representation, the *surface-syntactic structure* or *tree* is an unordered dependency tree, whose nodes are labeled with an actual lexemes of the sentence. "Five differences between the nodes of surface-syntactic structure and those of deep-syntactic structure should be indicated. First, in a surface-syntactic structure all the lexemes of the sentence are present, including semantically empty (i.e. structural) words. Second, all the idioms are expanded into actual surface-syntactic trees. Third, the values of all the lexical functions are computed (using the dictionary) and spelled out as actual lexemes, replacing the lexical function symbols. Fourth, all the fictitious lexemes of the deep-syntactic structure are expressed in the surface-syntactic one by the corresponding surface-syntactic relations and thus disappear. And fifth, all the pronominalizations [...] are carried out, so that a surface-syntactic node can be a pronoun. However, generally speaking, there is no one-to-one mapping between the nodes of a surface-syntactic structure and the actual wordforms of the sentences: either a surface-syntactic node may correspond to a zero wordform, or two surface-syntactic nodes may correspond to one amalgamed word-form" (DS, 68). Each lexeme is accompanied by grammemes as in the deep-syntactic structure; some deep syntactic grammemes are translated by analytic constructions (such as voices and some tenses in English) and do not appear as grammeme at the surface-syntactic level. Conversely, agreement and government grammemes (such as cases) are only surface grammeme (Mel'čuk 1993; [article 19]).

An *arc*, or *branch*, of surface-syntactic tree is labeled with a surface-syntactic relation. These relations are language-specific and describe particular syntactic constructions of particular languages. In the framework of MTT, lists of surface-syntactic relations have been proposed for Russian (Mel'čuk 1974, 221-235), English (Mel'čuk/Pertsov 1987, 85-156; Apresjan *et al.* 1989, 71-121) and French (Apresjan *et al.* 1984-1985; Iordanskaja/Mel'čuk, *to appear*). The last paper proposes criteria to establish such an inventory (for the actants of a verb).

Information about surface-syntactic representation of many linguistic phenomena can be found in (Mel'čuk/Pertsov 1987; DS and [article 19]).

## 2.4. Deep- and surface-morphological representations

A *deep-morphological representation* specifies the form of a particular sentence in terms of its wordforms and their linear order. The main structure of the deep-morphological representation, the *deep-morphological structure* or *string*, is an ordered string (in the speech order) of lexemes accompanied by a full set of corresponding grammemes (including agreement grammemes which appear at this level); these are in fact the deep-morphological representations of the wordforms of the sentence (Fig. 8).

$\mathrm{JOHN}_{sg}$ $\mathrm{FEEL}_{ind,\,pres,\,3,\,sg}$ NO $\mathrm{REVULSION}_{sg}$
AT THE $\mathrm{SIGHT}_{sg}$ OF A DEAD $\mathrm{ANIMAL}_{sg}$

**Fig. 8. The deep-morphological string of (1a)**

The deep-morphological string is complemented by the *deep-morphological prosodic structure* which indicates the grouping of the words into prosodic units labeled by prosodic markers calculated from the communicative markers of the corresponding grouping in the surface-syntactic representation. The real prosodic structure will be introduced at the phonological level and calculated from the morphological prosodic structure and the phonological properties of the words (which are not taken into account at the morphological level where the phonemes are not considered). (Gerdes/Kahane 2001) proposes, following ideas of Mel'čuk, to built at the morphological level a phrase structure that, contrary to phrase structure grammars based on the X-bar Syntax, does not encode the syntactic structure of the sentence, but its morphological prosodic structure.

The *surface-morphological representation* is similar to the deep-morphological one, but it stresses the internal organization of wordforms. The surface-morphological representation of a wordform is the set of morphemes making it up (with no formal distinctions

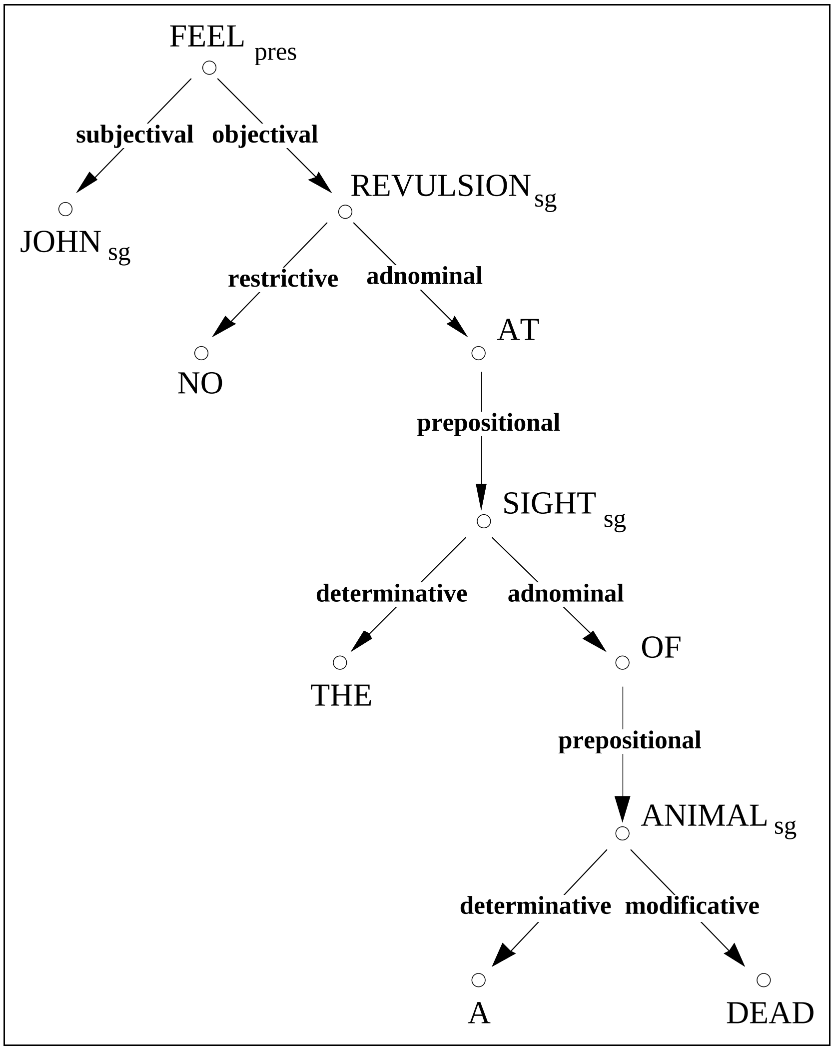



**Fig. 7. The surface-syntactic tree of (1a)**

between lexical and grammatical morphemes). About all questions of morphology, see the extensive book of morphology of Melčuk 1993-2000.

## 3. Meaning-Text models

A Meaning-Text model [MTM] is a model (= formal grammar) of a particular natural language. It is made up of a lexicon and six correspondence modules, which define the correspondence between semantic representations (= meanings) and the surface-phonological representations (= texts) of this language.

The presentation of the MTT lexicon will be followed by a presentation of the three first correspondence modules. A last part concerns the definition of the correspondence described by an MTM.

### 3.1. Meaning-Text lexicon: The Explanatory Combinatorial Dictionary

The MTT lexicon is called the Explanatory Combinatorial Dictionary [ECD]. It describes the behavior of the deep lexical units of a language. A first ECD for Russian was proposed by Mel'čuk/Zholkovsky 1965 and developed in Mel'čuk/Zholkovsky 1984. An ECD of French is being developed at the University of Montreal (Mel'čuk *et al.* 1984, 1988, 1992, 1999; see also Mel'čuk 1992a and Iordanskaja/Mel'čuk 1995). ECD entries for English can be found in Mel'čuk/Polguère 1987, Ilson/Mel'čuk 1989 or Steele 1990. Presentations of an ECD can be found in Mel'čuk/Polguère 1987, Mel'čuk/Zholkovsky 1988 and Mel'čuk/Class/Polguère 1995.

Further studies about the specific surface lexical units, such as pronouns, articles, auxiliaries, lexical function's values, empty words or parts of idioms, are needed to decide where information about them must be encoded: in the correspondence rules, in the general ECD, or in a specific surface-ECD.

A sample lexical article, ECD-style, of the headword REVULSION will follow (it is a revised version of the article in Mel'čuk 1998). As every article of an ECD, it is divided into three major zones :

- a semantic zone: the lexicographic definition or semantic decomposition of the headword.

- a syntactic zone: the *government pattern* (= subcategorization frame), which specifies, for each semantic actant (X, Y, ...) the corresponding deep-syntactic relation (I, II, ...) and lists of all surface means of expressing them in the text (N*'s*, *against* N, ...); some particular conditions follow, such as: if Y is expressed by *toward* N, N must denote people ($C_{II.4}$ means Column II, line 4); examples show what is possible and what is not.

- a lexical cooccurence zone: lexical functions, which describe the restricted lexical cooccurence of the headword; let us recall that the numbers in the names of lexical functions refer to the government pattern.

**REVULSION**

**Semantic definition**

*X's revulsion for Y* ≡ X's (strong) negative emotion about Y similar to what people normally experience when they are in contact with something that makes them sick and such that it causes that X wants to avoid any contact with Y. (See Fig. 3.)

**Government Pattern**

| X = I | Y = II |
|---|---|
| 1. N*'s* | 1. *against* N |
| 2. $A_{poss}$ | 2. *at* N |
| | 3. *for* N |
| | 4. *toward* N |

(1) $C_{II.2}$ : N denotes something that happens and can be seen or felt
(2) $C_{II.4}$ : N denotes people

**Lexical Functions**

| | |
|---|---|
| $\text{Syn}_{\cap}$ | : repugnance; repulsion; disgust; loathing; distaste |
| $\text{Anti}_{\cap}$ | : attraction |
| $\text{Conv}_{21}\text{Anti}_{\cap}$ | : appeal |
| $\text{A}_1$ | : revulsed |
| $\text{Able}_2$ | : revulsive |
| $\text{Magn+Able}_2$ | : of utmost ∼ \| Y = SCENE, SIGHT |
| Magn | : deep < extreme < utmost |
| AntiMagn | : slight |
| Propt | : in, out of [∼] |
| $\text{Adv}_1\text{Manif}$ | : with [∼] |
| $\text{Oper}_1$ | : experience, feel [∼] |
| $\text{Magn+Labor}_{12}$ | : fill [N=X with ∼] |
| $\text{IncepLabor}_{12}$ | : drive [N=X to ∼] |

**Examples**

He did it out of deep revulsion against the bitterness of the sectarian strife. Any revulsion they might feel from fat-ass bastards they ran up against professionally was *ad hominem* and not *ad genus* [Alison Lurie]. Kathleen turned her head away in revulsion. I felt no revulsion for her maternal phantasies, only a practical concern. She met his advances with revulsion. It was a scene of utmost revulsion. Pam was driven to revulsion (by the sight of the dead animal). Revulsion at slaughter cut war short [newspaper heading].

Note that the government pattern should be more precise than it is in present ECDs; in particular, the explicit indication of surface-syntactic relations is necessary.

## 3.2. Correspondence modules

I will now present the first three correspondence modules of an MTM (which ensure the correspondence from the semantic level to the deep-morphological level). Each module of an MTM contains a set of *correspondence rules* having the form:

$$X \Leftrightarrow Y|C$$

where $X$ and $Y$ stand for fragments of utterance representation at two adjacent levels (e.g., semantic and deep-syntactic levels) and $C$ is a set of *conditions* under which this correspondence holds. The rule must be read “if conditions $C$ are verified, $X$ can be translated into $Y$” in the synthesis direction and “if conditions $C$ are verified, $Y$ can be translated into $X$” in the analysis direction. In fact, it is not the whole configurations $X$ and $Y$ which are translated into each other by the rule, but only a part that is printed in bold (Kahane/Mel’čuk 1999). The rest indicates the context and allows us to put together the configurations produced by the rules.

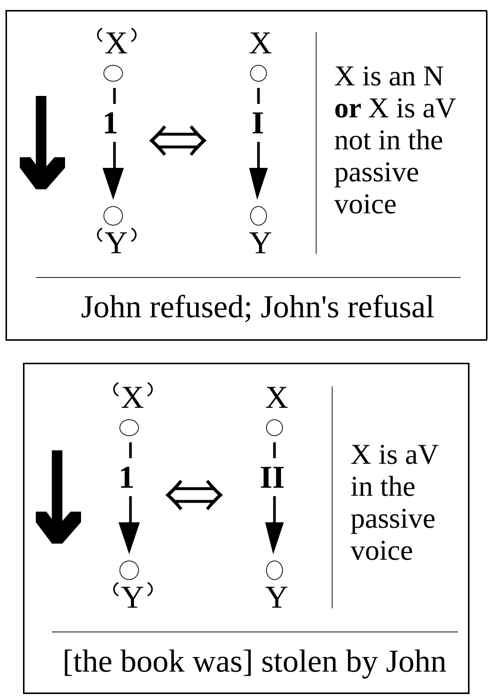
Following the classification proposed in Kahane/Mel’čuk 1999, the correspondence rules are separated into *nodal* and *sagittal rules*, that is, rules where the part of $X$ which is actually handled by the rule is a node, respectively an arrow (Lat. *sagitta* = arrow = arc of a graph or branch of a tree).

I will now see some examples of rules before giving additional comments on the formalism.

### 3.2.1. The semantic module

The semantic module of an MTM establishes the correspondence between semantic and deep-syntactic representations of sentences. It ensures two basic operations: the lexicalization and the hierarchization or arborization of the semantic graph.

The hierarchization is ensured by the *sagittal semantic correspondence rules*. Among the sagittal rules, positive and negative rules are distinguished: In a *positive rule*, the semantic arc is translated into a deep-syntactic branch going in the same direction, that is, an actancial dependency, while, in a *negative rule*, the semantic arc is translated into a branch going in the opposite direction, that is, an ATTR, COORD or APPEND dependency. The hierarchization consists in choosing an *entry node* in the graph, which yields the root node of the tree, and running through the whole graph from this entry node. Sagittal semantic rules for the translation of semantic dependency 1 are shown in Fig. 9. Each semantic dependency is attached to two semantic nodes ‹X› and ‹Y› whose deep syntactic correspondents are X and Y; these labels allows us to joint the sagittal rule with the nodal rules that translate ‹X› into X and ‹Y› into Y. The big arrow at the left of the rule indicates the communicative hierarchy between the nodes ‹X› and ‹Y›, which corresponds to the direction of the running and is related to the communicative structure (notably by the fact that the entry node must be the communicatively dominant node of the rheme or the theme): A positive rule is triggered when the semantic dependency and the communicative hierarchy go in the same direction and a negative rule when they go in opposite directions.



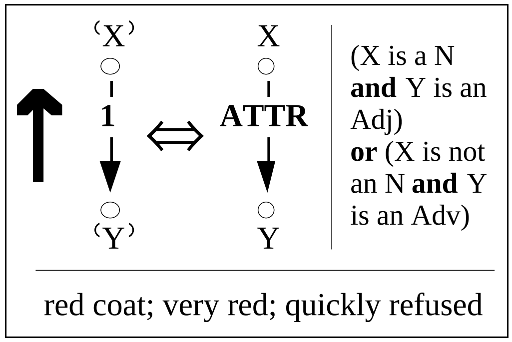



**Fig. 9. Three sagittal semantic rules**

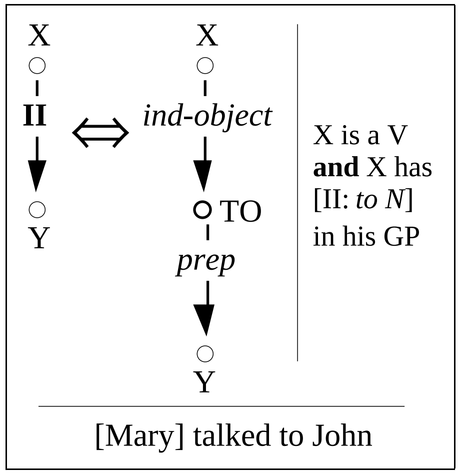
Note that all the rules presented here are *local*, meaning that they translate a limited part of the semantic graph into a limited part of the deep-syntactic tree. Nevertheless, there can be serious mismatches between the two structures, which need non-local rules to be dealt with (see Kahane/Mel'čuk 1999 where rules for extractions are presented).

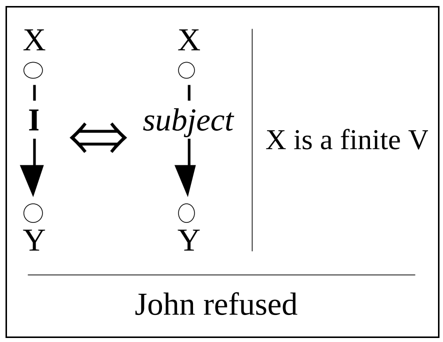
A nodal semantic rules associates a semantic node labeled by a semanteme ‹*s*› to a deep-syntactic node labeled by a deep lexeme whose signified is ‹*s*› (= a lexicalization of ‹*s*›) or a more complex configuration, such as the copular verb BE with an adjective or a light verb with a predicative noun. Inflectional rules, which produce deep grammemes, will not be considered here. I will also leave out the communicative rules, which produce the deep-syntactic communicative structure. Note that an important part of the information contained in the semantic communicative structure is already used by the structural rules, notably in the arborization process—by specifying whether a positive or a negative sagittal rule must be triggered off—, as well as in the choice of the diathesis—by controlling the lexicalization and the inflectional rules that introduce the voice grammemes (see Polguère 1990; Kahane/Mel'čuk 1999).

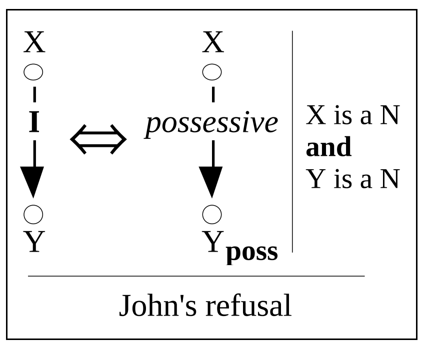
### 3.2.2. The deep-syntactic module

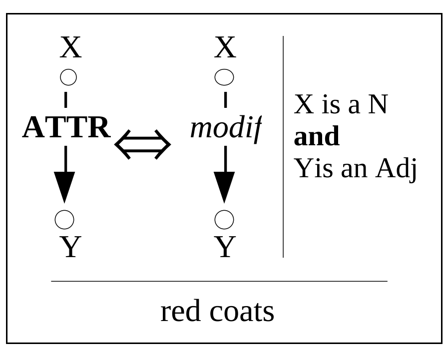
The deep-syntactic module of an MTM establishes the correspondence between deep- and surface-syntactic representations. It ensures the introduction of all the surface lexical units of the sentence (which corresponds one-to-one to the words of the sentence).

Sagittal deep-syntactic rules generally translate a deep-syntactic dependency into a surface-syntactic dependency; these rules also introduce prepositions controlled by the government pattern (Fig. 10).


X
II
Y
X
ind-object
TO
prep
Y
X is a V
and X has
[II: to N]
in his GP
[Mary] talked to John


X
I
Y
X
subject
Y
X is a finite V
John refused


X
I
Y
X
possessive
Y poss
X is a N
and
Y is a N
John's refusal


X
ATTR
Y
X
modif
Y
X is a N
and
Yis an Adj
red coats

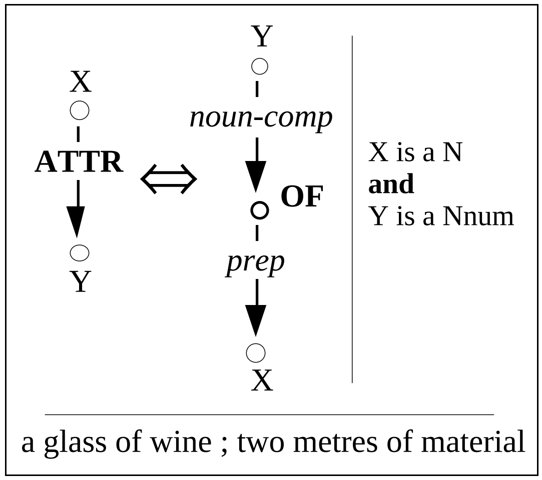
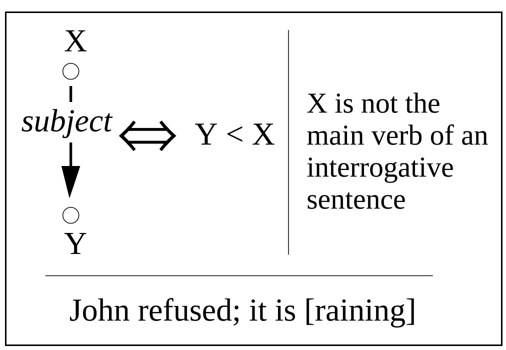



**Fig. 10. Five English sagittal deep-syntactic rules**

Nodal deep-syntactic rules translate deep lexical units into surface lexical units. If a deep lexical unit is a phraseme, it corresponds to a complex surface-syntactic configuration. If a deep lexical unit is a lexical function, it corresponds to a value of this function. Nodal deep-syntactic rules also include pronominalization rules: if several deep lexical units are in a co-referential relation, some of them must be pronominalized.

### 3.2.3. The surface-syntactic module

The surface-syntactic module of an MTM establishes the correspondence between surface-syntactic and deep-morphological representations. It ensures the linearization, the agreement and the government (Fig. 11). See Mel'čuk/Pertsov 1987 for a detailed presentation of sagittal surface-syntactic rules of English.



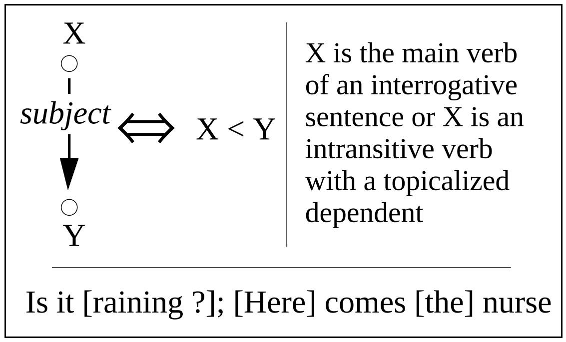



**Fig. 11. Two English sagittal surface-syntactic rules**

Sagittal surface-syntactic rules such as the rules presented here do not suffice to control linearization, for two reasons. First, they do not say anything about the linearization of co-dependents: Different solutions are possible, such as to extend the rules by indicating which co-dependents can be found between X and Y (Mel'čuk/Pertsov 1987; Nasr 1996), by specifying the distance between X and Y (Mel'čuk 1967; Kahane 2000; 2001c) or by introducing morphological constituents with different fields for the different co-dependent (Gerdes/Kahane 2001). Second, in non projective constructions, such as extractions, non-local rules are needed, because some nodes are not positioned towards their governor but towards an ancestor of their governor.

Nodal surface-syntactic rules are trivial. Agreement grammemes are introduced on deep-morphological nodes by specific agreement rules, which are generally part of sagittal surface-syntactic rules (because agreement between two lexical units is triggered by the type of the dependency holding between them; for example, in English, a finite verb agrees with its *subject*).

## 3.3. Definition of the correspondence described by an MTM

We will now see how the correspondence between meanings and texts that models a natural language is defined with an MTM.

### 3.3.1. The nature of the correspondence

An MTM module defines more than a correspondence between two sets of structures: For each structures $S$ and $S'$ that are put in correspondence by an MTM module, partitions of $S$ and $S'$ are defined (which are the fragments considered by the correspondence rules), as well as a one-to-one mapping $\phi_{(S,S')}$ between the components of these two partitions. We call this a *supercorrespondence* between two sets of structures. The supercorrespondence defined between sets of structures $\mathcal{S}$ and $\mathcal{S}'$ is mathematically equivalent to a family of product structures $(S, S', \phi_{(S,S')})$, with $S \in \mathcal{S}$, $S' \in \mathcal{S}'$ and $\phi_{(S,S')}$ a correspondence between the components of partitions of $S$ and $S'$. For instance, the surface-syntactic module of an MTM defines not only a correspondence between the set $\mathcal{S}$ of surface-syntactic dependency trees and the set $\mathcal{S}'$ of deep-morphological strings, but also, for each couple $(S, S')$ where $S$ and $S$' are in correspondence, $S$ is a dependency tree of $\mathcal{S}$ and $S'$ is a morphological string of $\mathcal{S}'$, a one-to-one correspondence $\phi_{(S,S')}$ between the nodes of $S$ and $S'$ (due to the surface-syntactic nodal rules, which translate a node into a node). Each triple $(S, S', \phi_{(S,S')})$—the product of a dependency tree and a linear order—is equivalent to a linearly ordered dependency tree (Fig. 12). In other words, the surface-syntactic supercorrespondence is equivalent to a family of linearly ordered dependency trees. From this point of view, an MTM surface-syntactic module is not so far from a generative grammar, whose production is a family of ordered trees—the derivation trees (see Kahane 2001a). But, if the two approaches—MTT and the generative frameworks—are mathematically equivalent, they are different from the theoretical point of view: Only MTT explicitly models a natural language as a (super)correspondence.

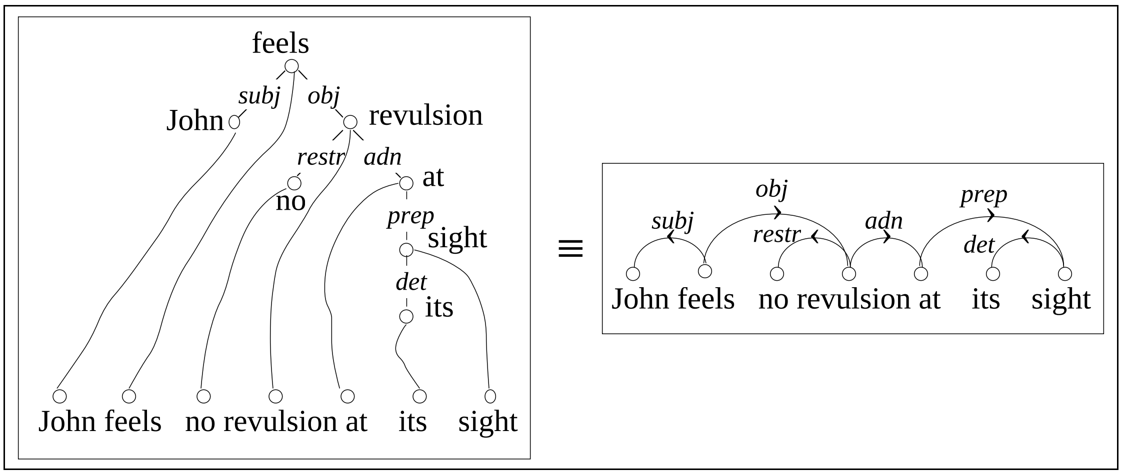



**Fig. 12. Equivalence between a tree and a linear order in correspondence and a linearly ordered tree**

### 3.3.2. Global rules

The correspondence rules, which simply establish correspondences between limited fragments of structures, are generally not sufficient to encode all properties of the global correspondence, so that some global rules must be stated. First of them are properties which ensure the well-formedness of the structures in correspondence, notably the rules which control that the structures are trees, strings, ... But there are also properties which involve simultaneously two structures in correspondence, that is, properties on the product structures. Consider again the surface-syntactic module, where the correspondence between a dependency tree and a string obeys some global properties. One of the most known properties of compatibility between a dependency tree and a linear order is *projectivity*, which state that if $a$, $b$, $c$ and $d$ are four nodes such that $a < b < c < d$, then the nodes $a$ and $c$ and the nodes $b$ and $d$ cannot be both linked by a dependency (which means that two ordered dependencies cannot cross each other). Although most of the linguistic constructions are projective, there are some phenomena in natural languages, such as cliticization or extraction, which create non projective constructions. Consequently only a weaker property than projectivity can be stated, but it is not possible not to state any property. (Mel'čuk/Pertsov 1987), following (Iordanskaja 1963), propose to give a list of couples of syntactic relations that can cross each other, that is, that can violate the projectivity. Another solution, proposed in (Kahane 1997; 2000; 2001b), is to state projectivity not directly on the linearly ordered dependency tree, but on a "bubble tree" obtained by grouping together some strings of nodes into bubble, which can occupy a node of the tree.

### 3.3.3. From the correspondence rules to the correspondence itself

To give a set of correspondence rules does not suffice to define a correspondence. Anyone who has been nurtured on the generative grammar knows that a formal grammar cannot be reduced to its set of rules. Even if we want to avoid the generative framework, something more or less equivalent to a derivation process is needed.

Before proposing different ways to define an MTM correspondence, it must be recalled that a complete definition of the correspondence is generally missing in the presentations of MTT. For example, (DS, 45) reduces a module of an MTM to a set of rules and justifies this saying: "The transition mechanism, i.e., the dynamic device, or procedure, for moving from actual complex semantic representations to actual complex phonological representations and vice-versa is not considered [by MTT]. I believe that such a dynamic device, while necessary to put the above static mapping to work, lies outside the field of linguistics, at least as yet. The MTM can be compared to a bilingual dictionary, which presupposes, but does not include, rules looking up the words it contains; then the dynamic device driving the MTM corresponds to the psychological ability of a human to use these rules in order to actually look up any given word. It stands to reason that such an ability is not part of the dictionary and should not concern the lexicographer too much.".

I think there is a confusion between two notions in Mel'čuk's explanations: the "derivation" (or something equivalent), which only states how to theoretically prove that two structures corresponds to each other, and a real procedure of synthesis or analysis (eventually psycholinguistically or computationally motivated). If we want to present a real procedure of analysis or synthesis, it is much more complicated because we have to take into account the question of multiple choices between rules (and, consequently, problems of memorization, choices, backtracking and parallelism). Moreover, a psycholinguistically motivated algorithm cannot do the processing level by level (i.e. it cannot construct the whole representation of level $a$ before beginning the construction of the representation of level $a+1$), but must manage all the levels of representation simultaneously (Kahane, 2001c).

I will now define the correspondence proper, based on the correspondence rules, and I will propose three ways to do that: the transductive, the generative and the equative presentations of an MTM (Kahane, 2001a).

### 3.3.4. Transductive presentation

The *transductive* presentation is the most natural and useful presentation of an MTM, because the goal of an MTM is to synthesize or analyze a sentence, that is, to transduce a semantic representation into a sentence (synthesis) or to do the converse (analysis). In a transductive process, a source structure $S$ is given and an MTM module is used to produce a corresponding target structure $S'$ of an adjacent level of representation. A transductive model can be compared to the well known string-to-string transducer (e.g. Aho/Ullman 1972), although, in our case, we want to transform a graph into a tree or a tree into a string (or vice versa). Roughly speaking, the process consists in reading the whole source structure and partitioning it, then triggering a correspondence rule for each component of the partition and finally putting together pieces of the target structure obtained by applying the correspondence rule. A rule $R : X \Leftrightarrow Y|C$ can be applied to a part $A$ of the source structure $S$ if $A$ is an $X$ and if the condition $C$ is verified by the product structure $(A, A')$, where $A'$ is the piece corresponding to $A$ by $R$.

For example, the last sagittal semantic rule $R$ of Fig. 9 can apply to the part $A =$ ‹dead› $\xrightarrow{1}$ ‹animal› of the semantic structure $S$ of Fig. 2 in order to produce the part $A' =$ ANIMAL $\xrightarrow{ATTR}$ DEAD (of the deep-syntactic structure of Fig. 5). Note that the lexicalization of ‹animal› into ANIMAL and ‹dead› into DEAD is made possible by separate nodal rules, whose application is controlled by the condition $C$ of $R$ which imposes that ‹dead› be lexicalized by an adjective if ‹animal› has been lexicalized by a noun.

Each module needs a particular procedure, because the reading and the assembling of a graph, a tree or a string are different operations. More exactly, each module needs two procedures: one for the synthesis and one for the analysis. Although the rules are bidirectional, the process of transduction cannot be easily inverted. The first formal presentation of an MTM, a transductive presentation for the synthesis, was proposed by (Boyer/Lapalme 1984). Another transductive presentation (again for the synthesis) can be found in (Iordanskaja/Polguère 1988; Iordanskaja 1990 and Polguère 1990). (Kahane/Mel'čuk 1999) contains a detailed transductive presentation of the semantic module for the synthesis. For a comparison between *global procedures* (the rules are triggered in one shoot) and *structure-driven procedures* (the rules are triggered running through the source structure), see (Kahane, 2001a; Bohnet/Wanner 2001).

### 3.3.5. Generative presentation

It is also possible to use the correspondence rules of an MTM in a *generative* way, in order to generate the couple of structures in correspondence. In this case, a correspondence rule is viewed as a couple of fragments of structure. A derivation process consists in putting together correspondence rules and building a couple of structures. Such a generative presentation, inspired from tree rewriting systems such as TAG, is proposed by (Nasr 1995; 1996; Kahane 2000; 2001b; 2001c). In my opinion, the differences between transductive and generative presentations are very tenuous and, generally, a presentation of one type gives naturally a presentation of the other type. In particular, the derivation generally favors one structure over the other in the product structure, that is, the derivation will be guided by one of the two structure. For example, in the generative presentation of the surface-syntactic module by (Nasr 1995; 1996), the derivation of a linearly ordered tree is guided by the tree structure and not by the linear order, exactly as it is done in a context-free grammar. Consequently, this presentation is very near to a transductive presentation for the synthesis (= from the tree to the linear order). Conversely, a generative presentation of the surface-syntactic module that favors the linear order will look like a push-down automaton.

### 3.3.6. Equative presentation

As we had seen, the main difference between a transductive and a generative presentation is that the former presupposes that one of the two sets of structures in correspondence is available (and produces the other one), while the latter produces directly both ones. The third way of defining the correspondence, the *equative* one, presupposes that both sets of structures are available: the process consists in filtering all the couples of structures which correspond to each other. The correspondence rules are used as a set of equations or constraints to be satisfied. For a purely mathematical point of view, it is the simplest definition of the correspondence and also the least procedural one: A product structure is given and each piece of the structure must be validated by a correspondence rule. In other words, the equative presentation is the declarative presentation of an MTM. Nevertheless, such a presentation is not very useful from computational or cognitive points of view, because, in such cases, meaning and text are never available together. And, in fact, another device is needed which generates couples of structures, the equative grammar being only used to filter the couples of structures that really correspond to each other.

## 4. Paraphrasing and translation

I will finish this presentation of MTT by reminding the reader that it was initially designed for translation and paraphrasing. Paraphrases are sentences which have nearly the same meaning. Given a sentence $S$, it is possible to obtain all its paraphrases by extracting its meaning ‹$S$› (the sentence $S$ is considered in one of its possible interpretations) and then synthesizing all the sentences that have the meaning ‹$S$›. Nevertheless, it is not necessary to obtain the meaning of $S$ and the paraphrasing can be performed at an intermediate level. Žolkovski/Mel'čuk 1967, Mel'čuk 1974, 146-176 and Mel'čuk 1992b proposed paraphrasing rules at the deep-syntactic level (Fig. 13).

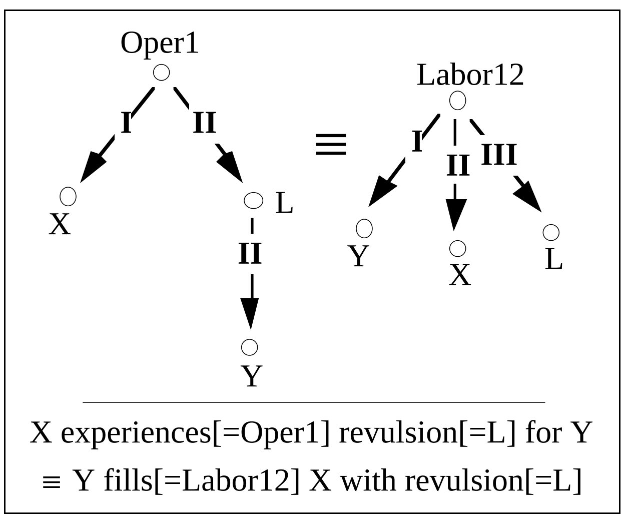



**Fig. 13. A deep-syntactic paraphrasing rule**

Note that correspondence rules and paraphrasing rules are in competition: if a semantic element $A$ corresponds to a deep-syntactic element $B$ by a rule $R$ and to a deep-syntactic element $B'$ by a rule $R'$, then there is a rule of paraphrasing $R''$ between $B$ and $B'$. Therefore, $B'$ can be synthesized directly from $A$ by $R'$ or by composition of $R$ and $R''$ (Fig. 14).

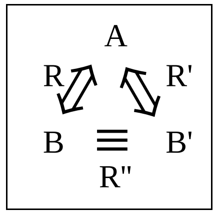



**Fig. 14. Correspondence and paraphrasing**

With a deep-syntactic module of paraphrasing, only a subset of semantic rules is necessary to synthesize all the sentences corresponding to a given meaning, because, if at least one deep-syntactic representation can be synthesized, all the other ones can be synthesized from it by paraphrasing. Moreover, with a deep-syntactic module of paraphrasing, the semantic correspondence module is almost unnecessary for a system of reformulation: In order to reformulate a given sentence, it suffices to get its deep-syntactic representation, to activate the module of paraphrasing and to synthesize a new sentence (see Nasr 1996 for such a system).

Paraphrasing is also used in translation systems. One strategy for translation is as follows: for a given sentence $S$ of the source language, we get its semantic representation ‹$S$›, translate ‹$S$› into a semantic representation of the target language and synthesize it. But, this strategy has never been implemented because of the lack of study of semantic representations (in particular of communicative structure) and semantic modules. Another strategy consists in building only the deep-syntactic representation $R$ of $S$ and translating it into a deep-syntactic representation $R'$ of the target language. But, this $R'$ is not necessarily synthesizable, for example, if some nodes are labeled by a lexical function that has no value for the keyword concerned. For instance, suppose you want to translate the sentence *It was a scene of utmost revulsion* in French; REVULSION will be translated by DÉGOÛT and OF UTMOST $\sim$ is the value of the lexical function Magn+Able$_2$. But DÉGOÛT has no value for Magn+Able2, although it has several values for Able$_2$, e.g. *dégoûtant* or *répugnant*. Thus the rule of paraphrasing of Fig. 15 can be triggered, yielding the translation *C'était une scène absolument répugnante* (where *absolument* is a value of Magn for RÉPUGNANT). See (DS, 99) for other examples.

Therefore a translation system that carries out the translation at an intermediate level is possible, provided it contains a module of paraphrasing. Schubert (1987; 2001) has developed a similar translation system, where the translation and paraphrasing rules function at the surface-syntactic level.

From a cognitive point of view, paraphrasing modules seem equally justified, because speakers always resort to reformulations, as studies of oral corpora show (e.g. Blanche-Benveniste *et al.* 1990).

## Conclusion

Mel'čuk and Žolkovski are probably the first, with the Meaning-Text theory, to say explicitly that a natural language must be modeled as a correspondence between

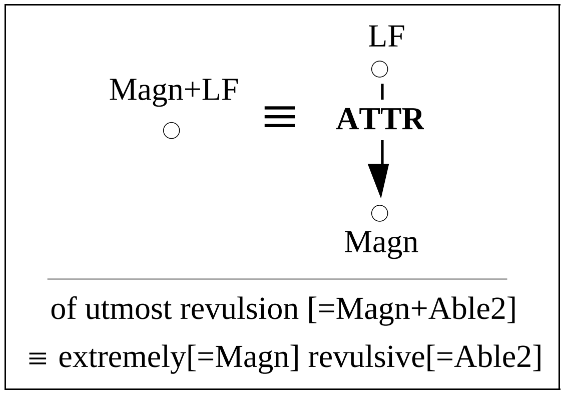



**Fig. 15. Another deep-syntactic paraphrasing rule**

meanings and sounds, thirty years before the generativists adopted this view with the Minimalist Program. As an example, Brody 1997's monograph on MP begins with the following sentences: "It is a truism that grammar relates sound and meaning. Theories that account for this relationship with reasonable success postulate representational levels corresponding to sound and meaning and assume that the relationship is mediated through complex representations that are composed of smaller units". And, as it is done in MTT, MP considers that sound and meaning representations correspond to the interface of the grammar (= the linguistic model, in our terms). But, as opposed to MTT, which considers a lot of intermediate levels of representation, MP puts forward the "minimalist" hypothesis that the grammar contains no non-interface levels (Chomsky 1993). And even for previous generativist models which use intermediate levels, such as the deep- and surface- structures, these intermediate levels are not laid out as in an MTM: The surface-structure, the logical form (= the meaning representation) and the phonological form (= the sound representation) are derived in parallel from the deep-structure, which suggests the primacy of syntax in these approaches. Although there is now a consensus about the fact that grammars describing natural languages must relate meanings and texts, few formalisms models languages as correspondences between meanings and texts explicitly. Another particularity of the Meaning-Text approach is to favor the synthesis direction in the presentation of a language, while most of the linguistic studies in other frameworks favor the analysis direction. The real knowledge of a natural language is to be able to speak it and not to understand it. Some important characteristics of languages, such as lexical functions, actually emerges when they are studied in the synthesis direction.

One mastery of MTT is that several engineering text-generation systems based on MTT has been successfully developed: FoG (Kittredge/Polguère 1991), the first system to produce actual weather forecasts (in English and French), LFS (Iordanskaja *et al.* 1992, 1996), a system generating stockmarket reports, RealPro, a generic system based on the previous one and developed by CoGenTex (Lavoie/Rambow 1997), as well as LexiGen (previously AlethGen), developed by LexiQuest (Coch 1996a, 1998b) and used in a reclamation letters answer system (Coch 1996b) and MultiMeteo, a system which produces multilingual and multistyle weather forecasts for the biggest European meteorology offices (Coch 1998a). Last, but not least, I could mention two machine-translation systems

(French-to-Russian: ETAP-1; English-to-Russian: ETAP-2; Apresjan *et al.* 1984-1985, 1989, 1992), which were fully developed within the MTT framework.

## Bibliography

AHO Alfred & ULLMAN Jeffrey, 1972, *The Theory of Parsing, Translation and Compiling, Vol. I: Parsing*, London: Prentice-Hall.

APRESJAN Jurij, 1974, *Leksičeskaja semantika* [Lexical Semantics], Moscow: Nauka.

APRESJAN Jurij, BOGUSLAVSKIJ I., IOMDIN L., LAZURSKIJ A., PERCOV N., SANNIKOV V. & CINMAN L., 1984-1985, *Lingvističeskoe obespečenie sistemy francuzsko-russkogo avtomatičeskogo perevoda ÈTAP-1*, Preprinty Problemnoj gruppy po èksperimental'noj i prikladnoj lingvistike, vyp. 154, 155, 159, 160, 166, 167, 174, Moscow: Institut Russkogo jazyka AN SSSR.

APRESJAN Jurij, BOGUSLAVSKIJ I., IOMDIN L., LAZURSKIJ A., PERCOV N., SANNIKOV V. &TSINMAN L., 1989, *Lingvističeskoe obespečenie sistemy ÈTAP-2*, Moscow: Nauka.

APRESJAN Jurij, BOGUSLAVSKIJ I., IOMDIN L., LAZURSKIJ A., SANNIKOV V. & TSINMAN L., 1992, "ETAP-2: The Linguistics of a Machine-Translation System", *Meta*, *37:1*, 97-112.

BLANCHE-BENVENISTE C., DELOFEU J., STEFANINI J. & EYNDE K. van den, 1990, *Le français parlé. Etudes grammaticales*, Coll. Sciences du langage; Paris: CNRS.

BOHNET, Bernd & WANNER, Leo, 2001, "On using a parallel graph rewriting formalism in generation", *Workshop on Natural Language Generation*, *ACL 2001*, Toulouse.

BOYER Michel & LAPALME Guy, 1985, "Generating paraphrases from Meaning-Text semantic networks", *Comput. Intell.*, *1*, 103-117.

BRODY Michael, 1997, *Lexico-Logical Form: A Radically Minimalist Theory*, Cambridge: MIT Press.

CANDITO Marie-Hélène & KAHANE Sylvain, 1998, "Une grammaire TAG vue comme une grammaire Sens-Texte précompilée", *Proc. TALN'98*, Marseille, 102-111.

CHOMSKY Noam, 1957, *Syntactic Structure*, Mouton: De Gruyter.

CHOMSKY Noam, 1965, *Aspects of the Theory of Syntax*, Cambridge: MIT Press.

CHOMSKY Noam, 1993, "A Minimalist Program for Linguistic Theory", in K. Hale & S. J. Keyser (eds), *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger*, Cambridge: MIT Press.

COCH José, 1996a, "Overview of AlethGen", *Proc. 8th Int. Workshop on Natural Language Generation (INLG'96)*, Herstmonceux, Vol.2, 25-28.

COCH José, 1996b, "Evaluating and comparing three text production techniques", *Proc. COLING'96*, Copenhagen, 249-54.

COCH José, 1998a, "Interactive generation and knowledge administration in MultiMeteo", *Proc. 9th Int. Workshop on Natural Language Generation (INLG'98)*, Niagara-on-the-Lake, 300-303.

COCH José, 1998b, "Application industrielle de la génération: pourquoi et comment", *T.A.L.*, *39:2*, 89-105.

DYMETMAN Marc & COPPERMAN Max, 1996, "Extended dependency structures and their formal interpretation", *COLING'96*, 255-61.

FONTENELLE Thierry, 1997, *Turning a bilingual dictionary into a lexical-semantic database*, Tübingen: Niemeyer.

GAZDAR Gerald, KLEIN Ewan, PULLUM Geoffrey K. & SAG Ivan A., 1985, *Generalised Phrase Structure Grammar*, Cambridge, MA: Harvard Univ. Press.

GERDES Kim & KAHANE Sylvain, 2001, "Word order in German: A formal dependency grammar using a topological hierarchy", *ACL 2001*, Toulouse.

HARTENSTEIN Klaus & SCHMIDT Peter, 1983, "Kommentierte Bibliographie zum "Smysl ⇔ Tekst"-Modell", *Wiener Slawitischer Almanach*, 11, 355-409.

HUDSON Richard A., 1990, *English Word Grammar*, Oxford: Blackwell.

ILSON, R. & MEL'ČUK Igor, 1989, "English BAKE revisited (BAKE-ing an ECD)", *International Journal of Lexicography*, *2*, 325-45.

IORDANSKAJA Lidija, 1963, "O nekotoryx svojstvax pravil'noj sintaksičeskoj struktury (na materiale russkogo jazyka)" [On some properties of correct synt. structure (on the basis of Russian)], *Voprosy jazykoznanija*, *4*, 102-12.

IORDANSKAJA Lidija, 1990, "Ot semantičeskoj seti k glubinno-sintaksičeskomu derevu: pravila naxoždenija veršiny dereva [Du réseau sémantique à l'arbre syntaxique profond: les règles pour la détermination du sommet de l'arbre]", in Z. Saloni (ed.), *Metody formalne w opisie języków słowiańskich*, Białystok: Wydawnictwo Filii UW w Białymstoku, 33-46.

IORDANSKAJA L., KIM M., KITTREDGE R. I., LAVOIE B. & POLGUÈRE, 1992, "Generation of Extended Bilingual Statistical Reports", *Proceedings of COLING'92*, Nantes, 1019-1023.

IORDANSKAJA L., KIM M. & POLGUÈRE, 1996, "Some Procedural Problems in the Implementation of Lexical Functions for Text Generation", in (Wanner 1996), 279-297.

IORDANSKAJA Lidija & MEL'ČUK Igor, 1995, "Traitement lexicographique de deux connecteurs textuels du français contemporain: EN FAIT vs EN RÉALITÉ", in H. Shydlkrot & L. Kupferman (eds), *Tendances récentes en linguistique française et générale*, Amsterdam: Benjamins, 211-36.

IORDANSKAJA Lidija & MEL'ČUK Igor, *to appear*, "Towards establishing an inventory of surface-syntactic relations: Valency-controlled surface-syntactic dependents of the verb in French", in Polguère A. & Wanner L., *Selected Topics in Dependency Grammar*, Amsterdam : Benjamins.

IORDANSKAJA Lidija & POLGUÈRE Alain, 1988, "Semantic processing for text generation", in *Proc. First International Computer Science Conf. - 88*, Hong Kong, 310-18.

KAHANE Sylvain, 1997, "Bubble trees and syntactic representations", in Becker & Krieger (eds), *Proc. 5th Meeting of the Mathematics of Language (MOL5)*, Saarbrücken: DFKI.

KAHANE Sylvain, 1998, "Le calcul des voix grammaticales", *Bull. Soc. Ling. Paris., 93:1*, Paris, 325-48.

KAHANE Sylvain, 2000, “Extractions dans une grammaire de dépendance lexicalisée à bulles”, *T.A.L.*, *41:1*, 211-243.

KAHANE Sylvain, 2001a, “What is a natural language and how to describe it? Meaning-Text approaches in contrast with generative approaches”, in A. Gelbukh (ed), *Computational Linguistics and Intelligent Text Processing*, Springer, 1-17.

KAHANE Sylvain, 2001b, “A fully lexicalized grammar for French based on Meaning-Text theory”, in A. Gelbukh (ed), *Computational Linguistics and Intelligent Text Processing*, Springer, 18-31.

KAHANE Sylvain, 2001c, “Grammaires de dépendance formelles et théorie Sens-Texte”, Tutoriel, *TALN 2001*, Tours, 60p.

KAHANE Sylvain & MEL’ČUK Igor, 1999, “Synthèse des phrases à extraction en français contemporain (Du graphe sémantique à l’arbre de dépendance)”, *T.A.L.*, *40:2*, 25-85.

KAHANE Sylvain & POLGUÈRE Alain, 2001, “Formal foundation of lexical functions”, in B. Daille, G. Williams (eds), *Workshop on Collocation*, *ACL 2001*, Toulouse.

KAMP Hans & REYLE Uwe, 1993, *From Discourse to Logic*, Dordrecht: Kluwer.

KITTREDGE Richard I. & POLGUÈRE Alain, 1991, “Dependency grammars for bilingual text generation: inside FoG’s stratificational models”, *Proc. Int. Conf. on Current Issues in Computational Linguistics*, Penang, 318-330.

LAVOIE Benoit & RAMBOW Owen, 1997, “RealPro - A fast, portable sentence realizer”, *Proc. 5th Conf. on Applied Natural Language Processing (ANLP’97)*, Washington, 265-68.

LAMB Sydney, 1966, *Outline of Stratificational Grammar*, Washington: Georgetown Univ. Press.

MEL’ČUK Igor, 1967, “Ordre des mots en synthèse automatique des textes russes”, *T.A. Informations*, *8:2*, 65-84.

MEL’ČUK Igor, 1974, *Opyt teorii lingvističeskix modelej ≪Smysl ⇔ Tekst≫ [Toward a Theory of Meaning-Text Linguistics Models]*, Moscow: Nauka.

MEL’ČUK Igor, 1981, “Meaning-Text Models: A recent Trend in Soviet Linguistics”, *Annual Review of Anthropology*, 10, 27-62.

MEL’ČUK Igor, 1988a, *Dependency syntax: theory and practice*, Albany, NY: State Univ. of New York Press.

MEL’ČUK Igor, 1988b, “Semantic description of lexical units in an Explanatory Combinatorial Dictionary: basic principles and heuristic criteria”, *International Journal of Lexicography*, *1*, 165-88.

MEL’ČUK Igor, 1989, “Semantic primitives from the viewpoint of the Meaning-Text linguistic theory”, *Quaderni di semantica*, *10*, 65-102.

MEL’ČUK Igor, 1992a, “CHANGER et CHANGEMENT en français contemporain (étude sémantico-lexicographique)”, *Bulletin de la Société de Linguistique de Paris*, *87*, 161-223

MEL’ČUK Igor, 1992b, “Paraphrase et lexique: La Théorie Sens-Texte et le *Dictionnaire explicatif et combinatoire*”, in Mel’čuk *et al.* 1992, 9-58.

MEL’ČUK Igor, 1993, “Agreement, government, congruence”, *Lingvisticae Investigationes*, *17:2*, 307-373.

MEL'ČUK Igor, 1993-2000, *Cours de morphologie générale*, *Vol. 1-5*, Montréal: Presses de l'Univ. Montréal / Paris: CNRS.

MEL'ČUK Igor, 1997, *Vers une Linguistique Sens-Texte*, Leçon inaugurale au Collège de France, Paris: Collège de France.

MEL'ČUK Igor, 1998, "Collocations and lexical functions", in A.P. Cowie (ed.), *Phraseology. Theory, Analysis, and Applications*, Oxford: Clarendon Press, 23-53.

MEL'ČUK Igor, 2001, *Communicative Organization in Natural Language: The Semantic-Communicative Structure of Sentences*, Amsterdam: Benjamins.

MEL'ČUK Igor & PERTSOV Nikolaj, 1987, *Surface Syntax of English. A Formal Model within the Meaning-Text Framework*, Amsterdam: Benjamins.

MEL'ČUK Igor & POLGUÈRE Alain, 1987, "A formal lexicon in the Meaning-Text Theory (or how to do lexica with words)", *Computational Linguistics*, *13*, 261-75.

MEL'ČUK Igor, CLAS André & POLGUÈRE Alain, 1995, *Introduction à la lexicologie explicative et combinatoire*, Paris: Duculot.

MEL'ČUK Igor & ŽOLKOVSKIJ Aleksandr, 1984, *Explanatory Combinatorial Dictionary of Modern Russian*, Vienna: Wiener Slawistischer Almanach.

MEL'ČUK Igor & ŽOLKOVSKIJ Aleksandr, 1988, "The Explanatory Combinatorial Dictionary", in M. Evens (ed.), *Relational Models of the Lexicon*, Cambridge: Cambridge Univ. Press, 41-74.

MEL'ČUK Igor *et al.*, 1984, 1988, 1992, 1999, *Dictionnnaire explicatif et combinatoire du français contemporain*, *Vol. 1, 2, 3, 4*, Montréal: Presses de l'Univ. Montréal.

MILIĆEViĆ Jasmina, 2001, "A short guide to the Meaning-Text linguistic theory", to appear in A. Gelbukh (ed), *Intelligent Text Processing and Computational Linguistics*.

NAKHIMOVSKY Alexander, 1990, "Word meaning and syntactic structure: some comparative notes", in J. Steele (ed.), *Meaning-Text Theory*, Ottawa: Univ of Ottawa Press, 3-17.

NASR Alexis, 1995, "A formalism and a parser for lexicalised dependency grammars", *4th Int. Workshop on Parsing Tecnologies*, State Univ. of NY Press.

NASR Alexis, 1996, *Un modèle de reformulation automatique fondé sur la Théorie Sens-Texte – Application aux langues contrôlées*, Thèse de Doctorat, Univ. Paris 7.

POLLARD Carl & SAG Ivan A., 1994, *Head-Driven Phrase Structure Grammar*, CSLI series, Chicago: Univ. Chicago Press.

POLGUÈRE Alain, 1990, *Structuration et mise en jeu procédurale d'un modèle linguistique déclaratif dans un cadre de génération de texte*, Thèse de l'Univ. de Montréal.

POLGUÈRE Alain, 1992, "Remarques sur les réseaux sémantiques Sens-Texte", in A. Clas (ed.), *Le mot, les mots, les bons mots*, Montréal: Presses de l'Univ. de Montréal.

POLGUÈRE Alain, 1997, "Meaning-Text Semantic Networks as a Formal Language", in L. Wanner (ed.), *Recent Trends in Meaning-Text Theory*, Amsterdam/Philadelphia: Benjamins.

POLGUÈRE Alain, 1998a, "La théorie Sens-Texte", *Dialangue*, *8-9*, Univ. du Québec à Chicoutimi, 9-30.

POLGUÈRE Alain, 1998b, "Pour un modèle stratifié de la lexicalisation en génération de texte", *T.A.L.*, *39:2*, 57-76.

SCHUBERT Klaus, 1987, *Metataxis: Contrastive Dependency Syntax for Machine Translation*, Dordrecht: Foris.

SCHUBERT Klaus, 2000, “Metataxis: Translation at the Core of Grammar Design”, in Polguère A. & Wanner L. (eds), ??

STEELE John (ed.), 1990, *Meaning-Text Theory: Linguistics, lexicography and Implications*, Ottawa: Univ. of Ottawa Press.

TESNIÈRE Lucien, 1959, *Éléments de syntaxe structurale*, Paris: Klincksieck.

WANNER Leo (ed.), 1994, *Lexical Functions*, Amsterdam: Benjamins.

WANNER Leo (ed.), 1996, *Lexical Functions in Lexicography and Natural Language Processing*, Amsterdam: Benjamins.

WANNER Leo, BOHNET Bernd & GIERETH Mark, 2000, “Arborization in a MTM-based Generation-Grammar”, ??

WEISS Daniel, 1999, “Sowjetische Sprachmodelle und ihre Weiterführung”, *Handbuch der sprachwissenschaftlichen Russistik und ihrer Grenzdisziplinen*, Wiesbaden: Harrassowitz Verlag, 873-909.

WIERZBICKA Anna, 1972, *Semantic Primitives*, Frankfurt am Main: Athenäum.

WIERZBICKA Anna, 1975, “Topic, Focus and Deep Structure”, *Papers in Linguistics*, *8:1-2*, 59-87.

WIERZBICKA Anna, 1978, “Syntax vs. Semantics”, *Theorical Linguistics*, *5:1*, 115-133.

WIERZBICKA Anna, 1980, *Lingua Mentalis: The Semantics of Natural Language*, Sydney: Academic Press.

WIERZBICKA Anna, 1985, *Lexicography and Conceptual Analysis*, Ann Arbor, MI: Karoma.

ŽOLKOVSKIJ, Aleksandr, 1964, “O pravilax semantičeskogo analiza” [On Rules for Semantic Analysis], *Mašinnyj perevod i prokladnaja lingvistika*, 8, 17-32.

ŽOLKOVSKIJ Aleksandr, LEONT’EVA Nina & MARTEM’JANOV Jurij, 1961, “O principial’nom ispol’zovanii smysla pri mašinnom perevode” [On Fundamental Use of Meaning in Machine Translation], in *Trudy Instituta Točnoj Maxaniki i Vyčisl. Texniki AN SSSR*, 2, 17-46.

ŽOLKOVSKIJ Aleksandr & MEL’ČUK Igor, 1965, “O vozmožnom metode i instrumentax semantičeskogo sinteza” [On a possible method an instruments for semantic synthesis (of texts)], *Naučno-texničeskaja informacija* [Scientific and Technological Information], 6, 23-28.

ŽOLKOVSKIJ Aleksandr & MEL’ČUK Igor, 1967, “O semantičeskom sinteze” [On semantic synthesis (of texts)], *Problemy kybernetiki* [Problems of Cybernetics], 19, 177-238. [Fr. transl. : 1970, *T.A. Information, 2*, 1-85.]