Functional Generative Description, Restarting Automata and Analysis by Reduction

Markéta Lopatková,[1] Martin Plátek,[2] and Petr Sgall[1]

[1] ÚFAL MFF UK, Praha, {lopatkova,sgall}@ufal.mff.cuni.cz
[2] KTIML MFF UK, Praha, martin.platek@mff.cuni.cz

Functional Generative Description (FGD) is a dependency based system for Czech, which has been developed since the 1960s (see esp. [5]). FGD can be interpreted as a generative system and/or as an analytical system (analyzer) as well (see [4]).

Here we propose a new formal frame for FGD based on *restarting automata*, see e.g. [2]. This new approach mirrors straightforwardly the so-called *analysis by reduction*, an implicit method used for linguistic research – analysis by reduction allows to obtain dependencies from the correct reductions of Czech sentences as well as to describe properly the complex word-order variants of free a word order language (see [1]).

FGD as a formal system for natural language L should (at least) determine:

- The set of (all momentarily determined) correct sentences of the (natural) language L, denoted by LC.
- The formal language LM representing all possible underlying (disambiguated) structures of sentences in L.
- The relation SH between LC and LM which describes the ambiguity and the synonymy of L.
- The set of the correct structural descriptions SD representing all possible underlying structures of sentences in L as dependency-based structures.

In the paper we discuss formal and 'practical' linguistic advantages of the proposed formal frame and we present other relevant properties.

Now let us briefly describe the type of restarting automaton we use for modelling FGD. An 4-LRL-*automaton* (*4-levelled* RL-*automaton*) M is (in general) a nondeterministic machine with a finite-state control Q, a finite characteristic vocabulary Σ (see below), and a head (window of size 1) that works on a flexible tape. Automaton M performs *move-right steps* and *move-left steps*, which change the state of M and shift the window one position to the right or to the left, respectively, *delete steps*, which delete the content of the window, change the state and shift the window to the right neighbor of the symbol deleted, and *rewrite steps*, which rewrite the content of the tape and change the state. At the right end of the tape, M either halts and accepts the input sentence, or it halts and rejects it, or it *restarts*, that is, it places its window over the left end of the tape and reenters the initial state. It is required that before the first restart step and also between any two restart steps, M executes at least one delete operation.

In order to model the step by step (FGD-like) translation of a sentence from LC onto its structural description from SD, the 4-LRL-automaton works with a complex characteristic vocabulary. The vocabulary Σ is partitioned into (sub)vocabularies $\Sigma_0, \cdots, \Sigma_3$; the particular vocabularies Σ_i represent the particular layers of the modelled FGD. E.g. Σ_0 is the set of Czech written word-forms, and the Σ_3 is the vocabulary of the *tectogrammatical (underlying) layer* of FGD.

The automaton M works with the vocabulary Σ representing both word forms (Σ_0) and metalanguage categories ($\Sigma_1, \Sigma_2, \Sigma_3$). The language accepted by M (that consists

Σ_0:		*Přišel*	*domů*	*pozdě*	.
Σ_1:		*přijít*.VpYS-	*domů*.Db- - -	*pozdě*.Dg- - -	..Z- - - -
Σ_2:		*přijít*.Pred	*domů*.Adv	*pozdě*.Adv	..AuxK
Σ_3:	*[On]*.ACT	*přijít*.PRED.Frame1	*domů*.DIR3	*pozdě*.TWHEN	

of all sentences from LC enriched with a metalanguage information from $\Sigma_1, \Sigma_2, \Sigma_3$) is called *characteristic language* $L_\Sigma(M)$. It embraces information from all the layers of FGD, particulary morphological lemma and tag (Σ_1), surface syntactic functions (Σ_2), and tectogrammatical information (esp. valency frame for frame evoking words and 'deep' roles, Σ_3), see the example (it is simplified in favor of lucidity). That means that the automaton has an access to all the information encoded in the processed sentence (as well as a human reader/linguist has all the information for his/her analysis).

Now we formally introduce an *analysis by reduction system* involved by M, $\mathsf{RS}(M) := (\Sigma^*, \vdash_M^c, S_\Sigma(M))$, where

- Σ is the characteristic vocabulary of M
- $u \vdash_M^c v$ denotes the fact that M can reduce u to v between two (re)starts; such a sequence of steps of M is called a *reduction*
- $S_\Sigma(M)$ consists of all sentences that M accepts without restarting; $S_\Sigma(M)$ is called the *simple characteristic language* accepted by M

We can introduce the corresponding notions also for particular levels of M. E.g., *characteristic language for level i*, denoted as $L_{\Sigma_i}(M)$, is a set of all sentences (strings) that are obtained from $L_\Sigma(M)$ by removing all symbols which do not belong to Σ_i.

Obviously, M satisfies the so called *error preserving property* – i.e., for each $w, v \in \Sigma^*$ such that $w \vdash_M^{c^*} v$, $w \notin L_\Sigma(M)$ it holds that $v \notin L_\Sigma(M)$. Informally, a string not belonging to the characteristic language $L_\Sigma(M)$ cannot be reduced to a sentence from this language.

A dual property often required for restarting automata is the so called *correctness preserving property*. Informally, a nondeterministic 4-LRL-automaton M is correctness preserving if for any $u \in L_\Sigma(M)$ each reduction that M may apply to u produces an element of $L_\Sigma(M)$.

The correctness preserving property allows us to formulate in a formal way a basic requirement on the modelled FGD. Thus, if M is not correctness preserving then its characteristic (or perhaps tectogrammatical) language must be improved (refined) in order to become a correct (completed) FGD (or its tectogrammatical layer).

It remains to say that the proposed formal frame for FGD based on restarting automata can be simply enriched so as to be able to construct *dependency structures* during their computations that fulfil the requirements on the correspondence between reductions and dependencies formulated in [1]. The added structures allow us to study the complexity issues of FGD in more detail – especially non-projectivity is in the center of our interest.

References

1. Lopatková M., Plátek M., Kuboň V.: Modeling Syntax of Free Word-Order Languages: Dependency Analysis by Reduction, In: Lecture Notes in Computer Science, Volume 3658, pp. 140-147, 2005
2. Otto, F.: Restarting Automata and their Relations to the Chomsky Hierarchy. In: Developments in Language Theory, Proceedings of DLT'2003 (eds. Esik, Z., Fülöp, Z.), LNCS 2710, Springer, Berlin, 2003
3. Plátek, M., Sgall, P.: A Scale of Context-Sensitive Languages: Aplication to Natural Language, Information and Control, Vol. 38., No 1., 1978
4. Plátek, M.: Composition of Translation with D - trees, COLING' 82, pp. 313-318, 1982
5. Sgall P., Nebeský L., Goralčíková A., and Hajičová E.: *A Functional Approach to Syntax in Generative Description of Language*, New York, 1969