THE LINGUISTIC RELEVANCE OF LINDENMAYER SYSTEMS

Leonor Becerra-Bonache
GRLMC, Rovira i Virgili University, Av. Catalunya, 35, 43002 Tarragona, Spain

Suna Bensch
Institutionen för Datenvetenskap, Umeå Universitet, 90187 Umeå, Sweden

M. Dolores Jiménez-López
GRLMC, Rovira i Virgili University, Av. Catalunya 35, 43002 Tarragona, Spain

Keywords: Lindenmayer systems, Natural language, Parallelism, Non-context freeness, Bio-inspired models.

Abstract: In this paper, we investigate the linguistic relevance of Lindenmayer Systems (L Systems). L systems were introduced in the late sixties by Aristid Lindemayer as a mathematical theory of biological development. Thus they can be considered as one of the first bio-inspired models in the theory of formal languages. Two main properties in L systems are 1) the idea of parallelism in the rewriting process and 2) their expressiveness to describe non-context free structures that can be found in natural languages. Therefore, the linguistic relevance of this formalism is clearly based on three main features: *bio-inspiration*, *parallelism* and *generation of non-context free languages*. Despite these interesting properties, L systems have not been investigated from a linguistic point of view. With this paper we point out the interest of applying these bio-inspired systems to the description and processing of natural language.

1 INTRODUCTION

Most current natural language approaches show several facts that invite to search for new formalisms to account in a simpler and more natural way for natural languages. In the last decades, biology has become a rich source of models for other sciences. The knowledge of the behavior of nature has influenced a number of areas such as artificial intelligence, mathematics or theoretical computer science, giving rise to new perspectives in research. Natural computing. neural networks, genetic algorithms and L systems are just some examples of this eclosion of new bio-inspired computational paradigms. The correspondences between several structures of natural language and biology allow us, in the field of linguistics, to think that maybe we can take advantage of the bio-inspired formal models in theoretical computer science. In fact, one of the goals in this research area is to offer simple bio-inspired theories for describing natural languages in order to make easier their manipulation and their implementation in Natural Language Processing (NLP) systems.

The question whether grammatical sentences of natural languages form regular, context-free, context-sensitive or recursively enumerable sets has been subject to many discussions since it was posed by Chomsky in 1957. There seems to be little agreement among linguists concerning the position of natural languages in the Chomsky hierarchy. It seems that neither the family of regular (REG) or context-free (CF) languages have enough expressiveness to describe the basic context-sensitive syntactic constructions found in natural languages. Several attempts have been made to prove the non-context-freeness of natural languages (Bresnan et al., 1985; Culy and Reidel, 1987; Shieber, 1987). Despite the fact that the non-context-freeness of natural language has become the standardly accepted theory, there are linguists such as Pullum and Gazdar who, after reviewing the various attempts to establish that natural languages are not context-free, come to the conclusion that every published argument purporting to demonstrate the non-context-freeness of some natural language is invalid, either formally or empirically or both (Pullum and Gazdar, 1987). Despite these arguments,

Becerra-Bonache L., Bensch S. and Dolores Jiménez-López M. (2010).
THE LINGUISTIC RELEVANCE OF LINDENMAYER SYSTEMS.
In *Proceedings of the 2nd International Conference on Agents and Artificial Intelligence*, pages 395-402
DOI: 10.5220/0002788003950402

it seems to be an untenable position that all syntactical aspects in natural languages can be captured by context-free grammars. However, the overwhelming bulk of natural language syntax is context-free.

Traditional linguistics presents a *hierarchical* view of grammar in the sense of taking the organizational dimensions of language to be 'levels' obtainable one from another in a certain fixed order. In this hierarchical view, the output of one component serves as the input to the next. This conception of grammar deprives components from any autonomy, since each of them has to wait for information of the previous one in order to start its task. The initial grammar models investigated in computational linguistics for natural language processing used to be *sequential* grammar formalisms. However, hierachicality and sequentiality have revealed as not appropriate features to account for natural languages. Therefore in natural language processing, researchers have turned from the initial serial models to *parallel* ones. And in theoretical linguistics, the problems with the hierarchical view of grammar have led to linguistic concepts/theories based on *parallel* and autonomous components (Jackendoff, 1997), (Sadock, 1991).

It follows from what we have said up to now that models in linguistics demand *bio-inspired devices* that avoid the *sequential* rewriting and that have *enough expressiveness* to describe natural languages. We propose, in this paper, to investigate the linguistic relevance of Lindenmayer systems and languages, in particular of ET0L systems, which have such desired properties. In fact, L systems are *parallel* and non sequential grammatical formalism, they are *biologically* inspired and they can generate the *non-context free structures* present in natural language.

The paper is organized as follows. Section 2 presents the basic idea and definition of Lindenmayer systems and offers examples. Section 3 discusses some of the advantages and the linguistic relevance of this framework and section 4 presents some conclusions and directions for future work.

2 L SYSTEMS

2.1 Preliminaries

We assume the reader to be familiar with basic notions in the theory of formal languages. With our notation we mainly follow (Dassow and Păun, 1989). In general, we have the following conventions: $\subseteq$ denotes inclusion, while $\subset$ denotes strict inclusion. By V^+ we denote the set of nonempty words over the alphabet V; if the empty word λ is included, then we use the notation V^*. Let Σ and Ω be two alphabets (which are not necessarily different). Let σ be a mapping from Σ into 2^{Ω^*}. This definition is extended to a mapping from Σ^* by:

$$\sigma(\lambda) = \{\lambda\} \text{ and } \sigma(w_1w_2) = \sigma(w_1)\sigma(w_2),$$

where w_1, w_2 in Σ^*.

Moreover, for a language $L \subseteq \Sigma^*$, let

$$\sigma(L) = \{\, u \mid u \in \sigma(w) \text{ for some } w \in L\,\}.$$

Such a mapping σ is called a substitution from Σ into Ω^*.

2.2 L Systems

Aristid Lindenmayer introduced, in 1968 (Lindenmayer, 1968), specific rewriting systems as models of developmental biology, which today are called Lindenmayer systems or L systems. L systems model biological growth and because growth happens in multiple areas of an organism, growth is parallel. This parallelism is the main difference from *sequential* rewriting systems of the Chomsky hierarchy. The investigations of L systems are an important and wide area in the theory of formal languages. The modelling of different environmental influences, for example, growth during day versus night, lead to different L systems and thus to different L languages. The study of L languages has resulted in a language hierarchy, namely the L system hierarchy. Lindenmayer systems are well investigated parallel rewriting systems. For an overview see (Kari et al., 1997) and for the mathematical theory of L systems see (Rozenberg and Salomaa, 1980).

Definition 2.1. An extended tabled Lindenmayer system without interaction (ET0L system, for short) is a quadruple $G = (\Sigma, H, \omega, \Delta)$, where Σ is the alphabet, Δ is the terminal alphabet, $\Delta \subseteq \Sigma$, H is a finite set of finite substitutions from Σ into Σ^*, and $\omega \in \Sigma^*$ is the axiom.

Definition 2.2. For x and y in Σ^* and $h \in H$, we write $x \underset{h}{\Longrightarrow} y$ if and only if $y \in h(x)$. A substitution h in H is called a *table*.

Definition 2.3. The language generated by G is defined as:

$$L(G) = \{w \in \Delta^* \mid \omega \underset{h_{i_1}}{\Longrightarrow} w_1 \underset{h_{i_2}}{\Longrightarrow} \dots \underset{h_{i_m}}{\Longrightarrow} w_m = w$$
for some $m \geq 0$ and $h_{i_j} \in H$ with $1 \leq j \leq m\}$.

By $\mathcal{L}(\text{ET0L})$ we denote the family of ET0L languages.

2.3 Examples

Example 2.1. Let

$$G_1 = (\{A, B, C, a, b, c\}, \{h_1, h_2\}, ABC, \{a, b, c\})$$

be an ET0L system, where h_1 and h_2 are given as follows:

$h_1 = \{A \to aA, B \to bB, C \to cC, a \to a, b \to b, c \to c\}$,

$h_2 = \{A \to a, B \to b, C \to c, a \to a, b \to b, c \to c\}$.

The axiom ABC can be rewritten using the first table h_1 or the second table h_2. Using the first three productions $A \to aA$, $B \to bB$, $C \to cC$ in h_1 adds a symbol a, b, and c, respectively in every derivation step. Using table h_2 terminates the derivation process. Consider the derivation of the word $a^2b^2c^2$: $ABC \underset{h_1}{\Longrightarrow} aAbBcC \underset{h_2}{\Longrightarrow} aabbcc$.

The language generated is $L(G_1) = \{a^nb^nc^n \mid n, m \geq 1\}$.

Example 2.2. Let

$$G_2 = (\{A, B, a, b\}, \{h_1, h_2, h_3, h_4\}, AA, \{a, b\})$$

be an ET0L system, where the tables are given as follows:

$h_1 = \{A \to aA, B \to B, a \to a, b \to b\}$,

$h_2 = \{B \to A, A \to B, a \to a, b \to b\}$,

$h_3 = \{B \to bB, A \to A, a \to a, b \to b\}$,

$h_4 = \{A \to a, B \to b, a \to a, b \to b\}$.

Table h_1 introduces a symbol a in every derivation step and table h_3 introduces a symbol b in every derivation step. Table h_2 serves as switch between the symbols A and B and table h_4 terminates the derivation process.

Consider the derivation for the word $baba$: $AA \underset{h_2}{\Longrightarrow} BB \underset{h_3}{\Longrightarrow} bBbB \underset{h_2}{\Longrightarrow} bAbA \underset{h_4}{\Longrightarrow} baba$.

The language generated is $L(G_2) = \{ww \mid w \in \{a, b\}^+\}$.

Example 2.3. Let $G_3 = (\{A, B, C, D, a, b, c, d\}, \{h_1, h_2, h_3, h_4\}, ABCD, \{a, b, c, d\})$ be an ET0L system, where the tables are given as follows:

$h_1 = \{A \to aA, C \to cC, B \to B, D \to D, a \to a, b \to b, c \to c, d \to d\}$,

$h_2 = \{A \to a, C \to c, B \to B, D \to D, a \to a, b \to b, c \to c, d \to d\}$,

$h_3 = \{B \to bB, D \to dD, A \to A, C \to C, a \to a, b \to b, c \to c, d \to d\}$,

$h_4 = \{B \to b, D \to d, A \to A, C \to C, a \to a, b \to b, c \to c, d \to d\}$.

Using tables h_1 or h_3 introduce in every derivation step the symbols a and c or the symbols b and d, respectively. The tables h_2 and h_4 are used in order to terminate the derivation process.

The language generated is $L(G_3) = \{a^nb^mc^nd^m \mid n, m \geq 1\}$.

3 LINGUISTIC RELEVANCE OF L SYSTEMS

The linguistic relevance of the above formalism is based on three main features: *parallelism*, *generation of non-context free structures in natural language* and *bio-inspiration*.

3.1 Parallelism

Linguistics has always been studied from a linear and sequential point of view. Indeed, the sound is produced in a sequential way, but due to many studies one comes to the conclusion that the production could not be sequential. Moreover, the multimodal approach to communication, where not just production, but also gestures, vision and supra-segmental features of sounds have to be tackled, refers to a parallel way of processing.

In general, formal and computational approaches to natural language demand non-hierarchical, parallel, distributed models in order to explain the complexity of linguistic structures as the result of the interaction of a number of independent but cooperative modules. According to (Smith, 1991), if we want to realize about how important is *parallelism* in our world, we just need to look at the *'most successful computing device'*, namely the *human mind*. Many cognitive processes exhibit degrees of parallelism. In computer science many procedures –in particular those that try to emulate human cognitive processes– call for parallel processing, either by requiring concurrently executable subtasks or relying on collective decision making. Computer networks, distributed data bases, highly parallel computers, parallel logic programming languages present a philosophy of computing very different from the traditional Turing-von Neumann sequential one. Parallel computational models have been applied to several domains. Language, vision, motor control and knowledge representation are some examples.

In natural language processing, researchers have turned from the initial serial models to highly parallel ones. Serial models used to adopt a 'syntax- first' approach, where syntactic processing of the sentence had to be done before semantic processing began, which in turn preceded pragmatic processing. In such models, information of lower levels cannot be used to correct decisions at higher levels with the consequent explosion of syntactic possibilities. That situation led to the preference of *parallel and interactive* models, where a system is capable of using any type of knowledge at any moment, without being constrained by a serial and hierarchical structure. Marslen and Wilson reported about some experiments that gave psychological evidence that processing at each level of natural language description can constrain and guide simultaneous processing at other levels, defending in this way the parallelism in language processing.

In theoretical linguistics, the hierarchical view of grammar has revealed as problematic and there has been a search for systems with parallel and autonomous components that cooperate in order to generate natural language. The advantages of parallel models of grammar have been pointed out in the last twenty five years by the arising of theories like Autolexical Grammar (Sadock, 1991), Jackendoff's Parallel Architecture (Jackendoff, 1997) or Head-Driven Phrase Structure Grammars (Pollard and Sag, 1994). All of them can be seen as parallel-architecture models. Moreover, recently parallel formalisms from the theory of formal languages have been applied to the description of natural languages. Examples of this can be, for instance, the application of grammar systems (Csuhaj-Varjú et al., 1994) to natural language description by defining Linguistic Grammar Systems (LGS) (Jiménez-López, 2006). LGS is a framework where the various dimensions of linguistic representation are arranged in a parallel distributed framework and where the language of the system is the result of the interaction of those independent cooperative modules that form the LGS. Another example of a parallel model from formal languages applied to the description of natural languages is the use of Networks of Evolutionary Processors (NEPs) (Castellanos et al., 1985) –a new computing mechanism directly inspired from the behavior of cell populations– suggested in (Bel-Enguix et al., 2009) where an implementation of NEPs for parsing of simple structures is suggested.

As mentioned earlier, Lindenmayer systems were introduced in connection with biological development. The new feature of Lindenmayer systems - in contrast to the sequential rewriting systems of the Chomsky hierarchy - was the parallel rewriting process. In Lindenmayer systems in every derivation step all available symbols are manipulated simultanously, that is, in *parallel*, whereas in the sequential rewriting systems of the Chomsky hierarchy in every derivation step only one symbol is manipulated. Therefore, L systems satisfied the parallelism demanded in the research fields of natural language processing and theoretical linguistics.

3.2 Generation of Non-context Free Structures

A grammatical formalism that attempts to model natural language syntax should have the same expressive power as natural languages. But, how much power is necessary to describe natural languages? This question has been a subject of discussion since it was posed by Chomsky in 1957. This debate was first focused on whether natural languages are CF or not.

CF grammars are well investigated in formal language theory due to their wide applicability and their mathematical properties. CF are simple devices and offer some advantages: they are powerful enough to describe most of the structures in natural languages and, at the same time, restricted enough so that efficient parsers can be built. However, context-free grammars are not always enough to account for natural languages. Many authors have brought as a demonstration of the non-context-freeness of natural languages examples of structures that are present in some natural languages and that cannot be described using a context-free grammar. Among those arguments we can refer to:

- 'Respectively' constructions (Bar-Hillel and Shamir, 1960);
- English comparative clauses;
- Mohawk noun-stem incorporation;
- Morphology of words in Bambara (Culy and Reidel, 1987);
- Dutch infinitival verb phrases (Bresnan et al., 1985);
- Assertions involving numerical expressions;
- English 'such that' (Higginbotham, 1987);
- English 'sluicing' clauses;
- Subordinate infinitival clauses in Swiss-German (Shieber, 1987).

Those structures are examples of the three following non-context-free languages:

1. $\{xx \mid x \in V*\}$, *reduplication*;
2. $\{a^n b^n c^n \mid n \geq 1\}$, $\{a^n b^n c^n d^n \mid n \geq 1\}$, *multiple-agreements*;
3. $\{a^n b^m c^n d^m \mid n, m \geq 1\}$, *cross-serial dependencies*.

Those constructions have been found in different languages, as follows from the above-mentioned list. Bambara, Mohawk, Walpiri, Dutch, Swiss-German, English or Romance languages are just some examples. These syntactic structures require more generative capacity than CF grammar provides to describe natural languages. Therefore, it is of interest to study grammatical formalisms with more generative power than CF. However, context-sensitive grammars seems not to be the right solution: they are too powerful, many problems are undecidable, etc. Therefore, it is desirable to find intermediate generative devices able of conjoining the simplicity of context-free grammars with the power of context-sensitive ones.

Within the field of formal languages, the above idea has led to the branch of *Regulated Rewriting* (Dassow and Păun, 1989). *Matrix grammars, programmed and controlled grammars*, *random context grammars*, *conditional grammars*, etc. are examples of devices that use context-free grammars while applying some restrictions to the rewriting process in order to obtain context-free structures as well as the non-context-free constructions present in natural language. But, those devices present, in general, an excesive big generative power that leads to the generation of structures non-significative for natural languages. The idea of keeping under control the generative power, while generating context-free structures and non-context-free constructions, has led to the so-called *mildly context-sensitive grammars* (Joshi, 1985). *Tree adjoining-grammars*, *head grammars*, *indexed grammars*, *categorial grammars*, *simple matrix grammars*, etc. are well-known mechanisms that generate mildly context-sensitive languages.

Taking into account the problems that context-free grammars seem to pose when applied to the syntax of natural language, but having in mind the difficulty of working with context-sensitive grammars, L systems offer an alternative way to generate the non-context-free structures present in natural languages while using context-free rules as it is shown in the following example. The parallel rewriting mechanism and the use of tables in ET0L provide the tools in order to achieve more generative capacity than that of context-free grammars.

In the example of Figure 1, from (Shieber, 1987), the verb *hälfe* requires a dative object, namely *em Hans*, while the verb *aastriiche* requires an accusative object, here *es huus*. As indicated by the lines, there is a crossed agreement within the sentence between the verbs and the corresponding objects concerning their case. Theoretically an unbounded number of verbs can occur in a Swiss-German sentence of this kind and each of these would require its corresponding object with the correct case marking. Thus, one could extend all sentences, such that after a number of occurrences of dative objects and after a number of occurrences of accusative objects the corresponding number of occurences of the verbs which require a dative object and the corresponding number of occurences of those verbs which require an accusative object follows.

In the following we give an ET0L system G_{SG} that generates the Swiss-German sentence above. We will only list the productions in every table in G_{SG} and for every table we only give the productions relevant for our example. Let the axiom of G_{GS} be:

$$\omega = Pronoun\ A\ B\ NP\ C\ D\ Verb$$

and the tables given by:

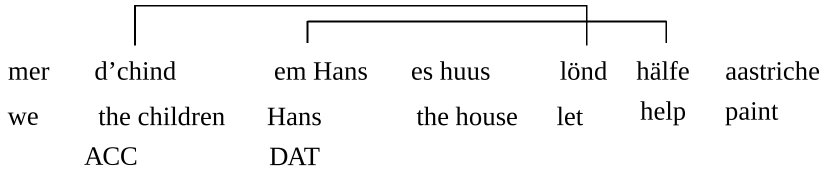
- $h_1 = \{Pronoun \to Pronoun, A \to AccObj\ A, B \to B, NP \to NP, C \to AccVerb\ C, D \to D, Verb \to Verb\}$,
- $h_2 = \{Pronoun \to Pronoun, B \to DatObj\ B, A \to A, NP \to NP, D \to DatVerb\ D, C \to C, Verb \to Verb\}$,
- $h_3 = \{Pronoun \to Pronoun, A \to AccObj, B \to B, NP \to NP, C \to AccVerb, D \to D, Verb \to Verb\}$,
- $h_4 = \{Pronoun \to Pronoun, B \to DatObj, A \to A, NP \to NP, D \to DatVerb, C \to C, Verb \to Verb\}$,
- $h_5 = \{Pronoun \to mer, DatObj \to emHans, AccObj \to d'chind, Np \to esHuus, DatVerb \to haelfe, AccVerb \to loend, Verb \to aastriche\}$.

Using tables h_1 or h_2 generates in every derivation step an auxiliary symbol for an accusative object or a dative object, respectively. Tables h_3 and h_4 terminate the introduction of auxiliary symbols for accusative objects and dative objects, respectively. Table h_5 rewrites the nonterminal symbols into terminal words of Swiss-German.

For the case in which, in a Swiss-German sentence as above, accusative and dative objects occur mixed one can construct in a similar way an ET0L system generating the language $\{ww \mid w \in \{a,b\}^+\}$.

3.3 Bio-inspiration

Most current natural language approaches show several facts that somehow invite to the search of new formalisms to account in a simpler and more natural way for natural languages. The fact that natural language sentences cannot be placed in any of the families of the Chomsky hierarchy in which current computational models are basically based or the idea that rewriting methods used in a large number of natural language approaches seem to be not very adequate,



'we let the children help Hans paint the house'

Figure 1: A Swiss-German Sentence.

from a cognitive perspective, to account for the processing of language lead us to look for a more natural computational system to give a formal account of natural languages.

During the 20th century, biology has become a pilot science, so that many disciplines have formulated their theories under models taken from biology. Computer science has become almost a bio-inspired field thanks to the great development of natural computing and DNA computing. From linguistics, several attempts of establishing structural parallelism between DNA sequences and verbal language have been performed. In general, it can be stated that the processing of natural language can take great advantage of the structural and ”semantic” similarities between those codes. Natural language processing could become another ”bio-inspired” science, by means of theoretical computer science, that provides the theoretical tools and formalizations which are necessary for approaching such exchange of methodology. In fact, during the last years, different bio-inspired methods have been succesfuly applied to several natural language issues, from syntax to pragmatics. Those methods are taken mainly from computer science and are basically the following: *DNA computing, cell computing, membrane computing and networks of evolutionary processors*. The main advantage of those bio-models is to account for natural language in a more natural way than classical models (based on rewriting).

L systems are the first bio-inspired model in the field of formal language theory. Aristid Lindenmayer introduced L systems in 1968 as a theoretical framework in order to model the development of filamentous organisms, which are composed of cells. These cells receive inputs from their neighbours and change their states and produce outputs based on their states and the input received. Cell division is modelled by inserting two new cells in the filament in order to replace one cell. With these theoretical framework of Lindenmayer an organism as a whole was provided that models individual acts of division, unequal divisions, interaction of two or more cells and cell enlargment. The possible combinations of interactions among more than a handful of cells becomes rapidly unmanagabale without a mathematical theory and computer application, which are provided both in the framework of Lindenmayer systems.

The biological inspiration of L systems is an interesting property from a linguistic point of view since, as we have said, in the last decades there seem to be a tendency to use bio-inspired models for the description and processing of natural languages. Linguistics has not been able to solve the problem of generation/understandig natural langauge, partly because of the fail in the models adopted. Bio-inspired models could be a possible solution since one of the advantages of such kind of models is to offer more 'natural' tools than the ones used so far.

4 CONCLUSIONS

Formal language theory was born in the middle of the 20th century as a tool for modeling and investigating syntax of natural languages. Lindenmayer systems are defined as a bio-inspired model in the area of formal languages. As we have shown in this paper, L Systems present important features from a linguistic perspective. However, although L Systems have relevant linguistic properties, up to now they have not been applied to describe natural languages.

Formal models in linguistics demand *parallel* devices that are able to generate the structures (CF and non-CF) present in natural languages with simple mechanisms that describe/explain those structures in a more natural way than the usual rewriting systems. Taking into account that current biology has become a pilot science, so that many disciplines have formulated their theories under models taken from biology, it seems natural that linguistics searches the improvement of its models in this field, mainly if we take into account the structural parallelism between biological sequences and verbal language. Morever since languages, either natural or artificial, are particular cases of symbol systems and the manipulation of symbols is the stem of formal language theory, it seems adequate to look for bio-inspired models that have been defined in that research area. If we do so, linguistics could become bio-inspired science, by means of theoretical computer science, that provides the theoretical tools and formalizations which are necessary for approaching such exchange of methodology. It is clear that interdisciplinarity must be an essential trait of the research on language. Linguistics, biology and computer science collaborate through the framework of formal language theory to give rise to the emergence of new scientific models that provide new ideas, tools and formalisms that can improve the description, analysis and processing of natural or artificial languages.

Now, Lindenmayer systems are the first bio-inspired model proposed in the field of formal languages and it is also the first one that replaces the sequential rewriting by the parallel one. Moreover, as we have shown, by using a L system you can easily generate the so-called non-context free structures in natural language. Therefore, it seems that Lindenmayer systems offer a great deal of the properties that seem to be necesssary in order to approach linguistic structures. Many parallel bio-inspired methods from the field of theoretical computer science have been successfully applied to several NLP issues, from syntax to pragmatics (DNA computing, membrane computing and networks of evolutionary processors). The main advantage of all those bio-models is to account for natural language in a more natural way than classical models. L systems is another example of bio-model, with many of the same properties as the ones that have been already applied to linguistics, so we consider that it could be interesting to apply L systems to the description/processing of natural language in order to see if this first parallel bio-inspired model may improve current linguistic approaches.

ACKNOWLEDGEMENTS

The research of Leonor Becerra-Bonache was supported by a Marie Curie International Fellowship within the 6th European Community Framework Programme.

REFERENCES

Bar-Hillel, Y. and Shamir, E. (1960). Finite-state languages: Formal representations and adequacy problems. *Bulletin of Research Council of Israel*, 8F:155–166.

Bel-Enguix, G., Jiménez-López, M., Mercas, R., and Perekrestenko, A. (2009). Networks of evolutionary processors as natural language parsers. In *Proceedings of the First International Conference on Agents and Artificial Intelligence - ICAART 2009*, pages 619–625. INSTICC Press.

Bresnan, J., Kaplan, R., Peters, S., and Zaenen, A. (1985). Cross-serial dependencies in Dutch. In Savitch, W., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 286–319. Kluwer, Dordrecht.

Castellanos, J., Martín-Vide, C., Mitrana, V., and Sempere, J. (1985). Solving np-complet problems with networks of evolutionary processors. In Mira, J. and Prieto, A., editors, *IWANN 2001*, pages 621–628. Springer, Berlin.

Csuhaj-Varjú, E., Dassow, J., Kelemen, J., and Păun, G. (1994). *Grammar Systems: A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach, London.

Culy, C. and Reidel, D. (1987). The complexity of the vocabulary of Bambara. In Savitch, W., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 349–357. Kluwer, Dordrecht.

Dassow, J. and Păun, G. (1989). *Regulated Rewriting in Formal Language Theory*. Akademie-Verlag, Berlin.

Higginbotham, J. (1987). English is not a context-free language. In Savitch, W., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 335–348. Kluwer, Dordrecht.

Jackendoff, R. (1997). *The Architecture of the Language Faculty*. MIT Press, Cambridge.

Jiménez-López, M. (2006). A grammar systems approach to natural language grammar. *Linguistics and Philosophy*, 29:419–454.

Joshi, A. (1985). How much context-sensitivity is required to provide reasonable structural descriptions: Tree adjoining grammars. In Dowty, D., Karttunen, L., and Zwicky, A., editors, *Natural Language Parsing: Psychological, Computational and Theoretical Perspectives*, pages 206–250. Cambridge University Press, New York, NY.

Kari, L., Rozenberg, G., and Salomaa, A. (1997). L systems. In Rozenberg, G. and Salomaa, A., editors, *Handbook of Formal Languages*, volume 1. Springer.

Lindenmayer, A. (1968). Mathematical models for cellular interaction in development, i and ii. *Journal of Theoretical Biology*, 18:280–315.

Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. Chicago University Press.

Pullum, G. and Gazdar, G. (1987). Natural languages and context-free languages. In Savitch, W., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 138–182. Kluwer, Dordrecht.

Rozenberg, G. and Salomaa, A. (1980). *The Mathematical Theory of L-Systems*. Academic Press, New York.

Sadock, J. (1991). *Autolexical Syntax - A Theory of Parallel Grammatical Representations*. The University of Chicago Press.

Shieber, S. (1987). Evidence against the context-freeness of natural languages. In Savitch, W., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 320–334. Kluwer, Dordrecht.

Smith, G. (1991). *Computers and Human Languages*. Oxford University Press, New York.