

# Book Reviews

## Parsing with Principles and Classes of Information

Paola Merlo

(University of Geneva)

Dordrecht: Kluwer Academic Publishers (Studies in linguistics and philosophy, edited by Gennaro Chierchia, Pauline Jacobson, and Francis J. Pelletier, number 63), 1996, x+245 pp; hardbound, ISBN 0-7923-4103-1, Dfl 140.00, \$90.00, £62.00

*Reviewed by*

Giorgio Satta

*University of Padua*

After Chomsky's (1981) introduction of the Government and Binding (GB) theory of grammar, a research area called GB parsing developed in the mid-eighties to explore parsing architectures based on that framework. In this area, parsing is viewed as the characterization of a mental process rather than a crude mapping from strings to syntactic structures. Therefore in GB parsing there is a need to develop a motivated mapping between the postulated model of humans' knowledge of language (the grammar) and the parsing architecture, an enterprise in which psychological as well as computational issues are at stake.

Both practical experimentation and theoretical investigation revealed early on that imposing a direct correspondence between the assumed grammar and the parsing architecture causes serious problems of efficiency (and in some cases the parsing process might not even terminate). This is in strong contrast with the empirical fact that humans make use of their knowledge of language in a very effective way. For these reasons, research on GB parsing has to face the strong tension between the desiderata of keeping the parser architecture close enough to the abstract linguistic formulation of the grammar, in order to inherit its explanatory power and its generalizations, and the clear need for grammar covering and compiling techniques to reduce the inefficiencies of the abstract linguistic formulation of the theory. This research monograph, which is a revised version of the author's doctoral dissertation, is an original contribution to this line of research. (A shorter version of this research was published as Merlo [1995].) Below, I will provide a general presentation of the book and of the author's ideas. Following that, I will discuss in greater detail the content of each chapter, adding specific comments and observations.

The *leitmotif* in this book is the distinction between linguistic modularity and computational modularity. As is well known, GB theory conceives of grammar as a set of modules, each consisting of a set of parameterized principles concerning some specific linguistic abstraction, as for instance phrase structure, abstract case assignment, binding, movement, and so on. In other terms, linguistic modularity accounts for the linguistic data by means of the interaction of abstract devices, in a way that achieves a high degree of explanation and generalization. From the computational perspective,

the author observes that in some cases GB modules express a precompiled combination of several computational processes that are “independent” of one another and should therefore be factored out in order to achieve a higher degree of computational efficiency and succinctness. The factorization of these processes is induced by five feature sets, called information content classes, that are homogeneous in their linguistic content, as explained below. On this basis, the author introduces the **information content modularity hypothesis** (ICMH), stating that precompilation of the grammar within the parser should be allowed only within each single information content class, in order to achieve computational efficiency. The author defends the ICMH by providing some experimental results showing the increased succinctness of the parser that derives from its application, at no extra computational cost at run-time. In addition, several psycholinguistic arguments from the existing literature are provided in support of the hypothesis. The ICMH is then used throughout the book to guide the design of the parser, as discussed below. The factorization of GB modules according to the ICMH is an original idea that has provided interesting results. Moreover, the book presents a complete investigation of the proposal, ranging from theoretical linguistics and psycholinguistics to parsing theory and computer algorithms. Although there are some shortcomings in the presentation of some of the arguments, as I will discuss below, I think this book is definitely worth reading for people working in the area of computational models of modern linguistic theories.

The first two chapters of the book state the problem and introduce the approach. In the first chapter, the author introduces the grammar/parser relation problem, discussing the issue of precompilation of the grammar and reporting the debate in the literature. The distinction between linguistic modularity and computational modularity is next introduced through discussion of a model proposed by Berwick (1985), wherein modules are conceived as filters and are implemented by means of deterministic finite-state automata. The basic idea is that interaction between modules corresponds to intersection of finite-state automata. In the design of the system, then, one is faced with the trade-off between the precompilation of the automaton recognizing the intersection of the source languages and the computation at run-time of the intersection language. In both cases, the resulting system will have a running time linear in the length of the input sentence. Nevertheless, in the second case we have a linear dependence on the number of modules that is not observed in the precompilation case, which is thus more time efficient. On the other hand, space requirements are worse in the first case, because precompilation might result in an automaton with a number of states equal to the product of the number of states in the source automata.<sup>1</sup> The first problem that arises is then that of determining in exactly which cases the intersection of, say, two automata gives rise to the mentioned worst case. This could then be taken as a measure of the “independence” between the two devices (the modules). But there is a second important problem, which is at the basis of the development of the work in this book, concerning when and how we could factor a module into “independent” devices, with the gain of a more succinct representation and at the expense of a limited overhead in running time. In the specific case of GB modules, the author proposes a factorization method based on the partition of the relevant linguistic features into five sets, or information classes. Each class encapsulates a specific linguistics

---

<sup>1</sup> This cross-product argument is based on a well-known upper bound for the computation of the intersection of regular languages (see for instance Hopcroft and Ullman [1979]). It should be observed that in order to make the point, a lower bound is instead needed. I am not aware of such a result, although it seems that such a lower bound could be easily provided by means of the Myhill-Nerode theorem (again, see Hopcroft and Ullman [1979]).

tic abstraction, such as structural configuration, lexical features, syntactic information, locality information, and referential information. The ICMH is then introduced and discussed.

In the second chapter, the author presents an overview of the parser architecture and the related data structures, along with a running example that shows a step-by-step analysis of a *wh*-sentence with raising. This is followed by a careful review of five related proposals in GB parsing, namely those of Abney (1989), Fong (1991), Dorr (1993), Frank (1992), and Crocker (1992), accompanied by some interesting criticism.

The presentation of the core of the author's work starts in the third chapter with the development of the phrase-structure component of the parser. Testing the ICMH on  $\bar{X}$ -theory, the author unfolds the standard  $\bar{X}$  schemata into a structural component and a categorial component. The parser then uses an LR table, encoding a category-neutral context-free backbone, and a so-called co-occurrence table with the categorial information. It is shown how this results in increased succinctness of the grammar representation, at no extra cost in terms of the overall amount of nondeterminism. In order to get further confirmation of the ICMH, the separation between structural and categorial information is carried out also for different parsing techniques, such as LL and LC, with less clear-cut but nonetheless interesting experimental results.

The author then considers the further step of unfolding the immediate-dominance information and the linear-precedence information within the structural component, but discards this idea on the basis of NP-completeness results presented by Barton, Berwick, and Ristad (1987). Unfortunately, this is not correct. A careful inspection of the proof in Barton, Berwick, and Ristad (1987) shows that the NP-hardness result makes crucial use of the unboundedness of the length of the productions in the grammar. It is not difficult to show that in the case of productions with bounded right-hand sides, the ID/LP universal recognition problem can be solved in polynomial deterministic time. And this is exactly the case for the grammar that the author is using, since  $\bar{X}$ -theory and thematic theory impose a bound on the length of the right-hand side.

As already mentioned, the adopted LR variant uses two tables in place of the standard LR table. Also, three stacks are used in a synchronous manner by the parsing algorithm. Unfortunately, this chapter does not provide a formal specification of the resulting method, which would be needed in order to have a full computational understanding of its advantages. For instance, the reduced size of the two tables as compared with a single LR table should also be checked against the computational overhead of using two tables in place of a single one. Although from the discussion of an example it seems that such an overhead is only constant time per step, a mathematical specification of the parsing cycle would have been in order here. Even more important, the reader is told several times that, although the categorial co-occurrence table considerably reduces the number of conflicts in the entries of the LR parse table compiled from the bare structural grammar, the resulting parser is still a nondeterministic device. Several solutions have been proposed in the literature for deterministic simulation of nondeterministic LR parsing—see for instance Lang (1974) or Tomita (1986)—but the author does not give any algorithmic specification in this book on how nondeterminism is dealt with.<sup>2</sup>

The author concludes the third chapter with interesting discussion of experimental findings regarding the resolution of categorial ambiguity, taken from the psycholin-

---

<sup>2</sup> From some of the examples in the book, I understand that nondeterministic choices are explored by means of features of the programming language that is used in the implementation, perhaps Prolog. If this is the adopted solution, then some efficiency problems that are relevant to the evaluation of the parsing architecture might be hidden behind this choice.

guistics literature, that could naturally be explained if the separation between structural and categorial information is assumed within the parsing architecture.

According to the ICMH, syntactic information exploited by different modules of the theory should be separated from structural, categorial, and lexical information, and should be computed by the parser using precompiled information. This is the subject of the fourth chapter. Following previous approaches in GB parsing, processing of syntactic information is performed through the application of precompiled constraints that act on so-called syntactic features. Crucial to the efficiency of the parser, such a computation is interleaved with the computation of the phrase structure, using standard techniques from the theory of compiler design.

Two kinds of computations are distinguished here, namely local and nonlocal computation. The proposed definition of locality consists in the maximal projection of a head, with the inclusion, in the case of the category *V*, of the associated functional complex (that is the projections of the *I* and *C* nodes that select the *VP*). Five syntactic features are involved in the local computation, encoding the assignment of abstract case and  $\theta$ -role, barrierhood, referential information, and licensing of empty categories. Assignment of these features and the checking of the precompiled constraints are performed with strict observance of the given definition of locality. To improve efficiency, the author precomputes for each local substructure typology the set of features and constraints that should be processed. It should be pointed out here that, under the given definitions, the parser would be unable to process cases of exceptional case marking, which would fall out of the local computation case. But this contradicts the assumptions on case assignment. It is not clear to me how the given definitions could be extended in order to treat these cases.

Nonlocal computation is used by the parser to deal with so-called *A*-movement and  $\bar{A}$ -movement. In this domain, the parser is able to analyze a remarkable range of constructions, including the intersection of *wh*-movement with *NP*-movement, and parasitic gaps. This is the result of several algorithms that are specified in the second part of Chapter 4. I will present the main idea underlying the proposal. In the parsing architecture at hand, empty categories are inserted by the phrase structure component at positions in which they are licensed, using structural and categorial information precompiled within the LR tables. The problems that must then be solved are the determination of the type of an empty category within the phrase structure and its insertion in the appropriate chain. Type determination is solved deterministically using underspecification. The idea is that if the parser is unable to solve type ambiguity on the basis of the local context, it delays the decision until enough structure has been constructed. The finitely many possible cases are precompiled within the grammar, and the resulting algorithm is completely deterministic and does not use backtracking. After type determination has been performed, empty categories are immediately inserted into the chain they belong to, and chains are carried over using an appropriate data structure. In this way, chain construction is interleaved with the computation of the phrase structure. Insertion of empty categories into the proper chain is again done deterministically, and no backtracking is needed. The author also provides a computational analysis of the above chain construction algorithms.

Long-distance movement is a very important problem in parsing, and its efficient analysis usually represents a major computational problem. In the presentation of the proposed algorithms, I would therefore have spent more words than the author actually does. More specifically, the author points out the case in which an empty category is underspecified as both an intermediate *wh*-trace and an empty operator. The way the ambiguity is eventually solved by the chain construction algorithm is not adequately discussed; I think that a specific example would have been in order here.

As a second point, in the composition of  $\bar{A}$ -chains (i.e., parasitic gap constructions) the author invokes an anti-c-command constraint on the heads of the chains (the operator positions) without adding any discussion or reference to the literature. This is incomprehensible, since in parasitic gap analyses the anti-c-command condition is usually imposed on the traces (see for instance Kayne [1983], where the condition is derived from Principle C of the binding theory). There is also an important case that the author does not deal with, which arises in so-called *pro*-drop languages, where a fully pronominal empty category called "small *pro*" can be found in the specifier position of an IP. Small *pro* receives case, and in the local contexts considered by the author is always compatible with a *wh*-trace. This is then another case in which the issue of nondeterminism arises in the determination of the type of an empty category, and it is not clear how to extend the author's algorithms in a way that does not give up determinism.<sup>3</sup>

This chapter, too, provides interesting discussion of psycholinguistic findings in support of the proposed approach in the determination of empty categories. The chapter then ends with the discussion of the issue of incrementality, which is an important requirement in computational models of the human sentence-processing capability. Drawing on an analysis of LR parsing by Shieber and Johnson (1993), the author is able to show that the proposed parser satisfies the standard definition of incrementality in a way that is independent of the type of the language which is parsed (SVO versus SOV, in the case at hand), while other incremental parsing architectures presented in the literature do not share this uniformity property.

The fifth and last chapter in the book completes the treatment of  $\bar{A}$ -movement by addressing the implementation of the notion of locality within the parser. This is an important issue, since within the GB framework long-distance dependencies are usually accounted for by means of (composition of) conditions that can be expressed on some local domains, where *local* essentially means nonrecursive (finite). The author starts by presenting some data and by critically reviewing some proposals from the GB parsing literature. An algorithm is then developed that is an extension of a proposal presented in Berwick and Weinberg (1984). The main feature of this algorithm is the parameterization of the notion of locality, which allows the author to account for variations in *wh*-extraction observed between English and Italian. Some examples are carefully discussed here. The book then ends with appendices in which the basics of GB theory are reviewed for the reader, and parse traces are provided for several standard English constructions.

As a final note, I should point out that a treatment of binding theory, a standard module of GB theory, has been left out of this book. This is, to some extent a weakness of the investigation, since it would have been quite significant to see the result of the application of the ICMH to the computation of the constraints imposed by the syntax on the referential relations of the anaphoric elements in a sentence.

<sup>3</sup> In fact it might well be that determinism must be given up in some of these cases. Consider for instance the Italian sentence *Chi ha detto che ha chiamato?* (lit. 'who has said that has called'). If the verb *chiamato* is read as an intransitive verb, then the operator *chi* binds a *wh*-trace in the specifier position of the IP in the subordinate clause. Alternatively, *chiamato* might be read as a transitive verb with the operator binding a *wh*-trace in the complement position of its projection, in which case the specifier position of the IP in the subordinate clause must be filled by a small *pro*.

## References

- Abney, Steven. 1989. A computational model of human parsing. *Journal of Psycholinguistic Research*, 18(1):129–144.
- Barton, G. Edward, Robert C. Berwick, and Eric Sven Ristad. 1987. *Computational Complexity and Natural Language*. MIT Press, Cambridge, MA.
- Berwick, Robert C. and Amy Weinberg. 1984. *The Grammatical Basis of Linguistic Performance*. MIT Press, Cambridge, MA.
- Berwick, Robert C. 1985. *The Acquisition of Syntactic Knowledge*. MIT Press, Cambridge, MA.
- Chomsky, Noam. 1981. *Lectures on Government and Binding*. Foris, Dordrecht.
- Crocker, Matthew. 1992. *A Logical Model of Competence and Performance in the Human Sentence Processor*. Ph.D. thesis, University of Edinburgh, Edinburgh, UK.
- Dorr, Bonnie Jean. 1993. *Machine Translation: A View from the Lexicon*. MIT Press, Cambridge, MA.
- Fong, Sandiway. 1991. *Computational Properties of Principle-Based Grammatical Theories*. Ph.D. thesis, Massachusetts Institute of Technology.
- Frank, Robert E. 1992. *Syntactic Locality and Tree Adjoining Grammar: Grammatical, Acquisition and Processing Perspectives*. Ph.D. thesis, Department of Computer and Information Science, University of Pennsylvania.
- Hopcroft, John E. and Jeffrey D. Ullman. 1979. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, Reading, MA.
- Kayne, Richard. 1983. Connectedness. *Linguistic Inquiry*, 14:223–249.
- Lang, Bernard. 1974. Deterministic techniques for efficient non-deterministic parsers. In J. Loeckx, editor, *Proceedings of the 2nd Colloquium on Automata, Languages and Programming*, pages 255–269, Saarbrücken, Germany. Lecture Notes in Computer Science, Springer-Verlag.
- Merlo, Paola. 1995. Modularity and information content classes in principle-based parsing. *Computational Linguistics*, 21(4):515–542.
- Shieber, Stuart and Mark Johnson. 1993. Variations on incremental interpretation. *Journal of Psycholinguistic Research*, 22(2):287–318.
- Tomita, Masaru. 1986. *Efficient Parsing for Natural Language*. Kluwer, Boston, MA.

Giorgio Satta is an assistant professor at the Department of Electronic and Computer Engineering, University of Padua, Italy. His main research interests are in the design of natural language parsing algorithms and in mathematics of language. Satta's address is: Università di Padova, Dipartimento di Elettronica ed Informatica, via Gradenigo 6/A, 35131 Padova, Italy; e-mail: satta@dei.unipd.it