Constraint-driven Concurrent Parsing Applied to Romanian VP

Liviu Ciortuz *
Dept. of Computer Science
University of Iasi, ROMANIA
Email: ciortuz@infoiasi.ro

Abstract

We show that LP constraints (together with language specific constraints) could be interpreted as meta-rules in (an extended) head-corner parsing algorithm using weakened ID rule schemata from the theory of HPSG [Pollard and Sag, 1994].

We present a refinement of the head-corner algorithm [van Noord, 1996], suitable for grammars containing ID/LP rules, as naturally used for high free-order languages.[1] We show that the ID schemata in the standard framework of the HPSG theory can be "weakened"/decomposed into subcomponents to allow for a fine-tuning of the ID rule application, according to the language specificity, on one hand, and – if desired – for concurrent parsing, on the other hand.

The concurrent-constraint programming (CCP) paradigm [Saraswat, 1993] lead to a powerful model [Smolka, 1995], which allows for the integration of logic (constraint), functional, and object-oriented programming, and also for smoothly passing from sequential to distributed programming, as this model adapts nicely to both these types of implementations [Mehl et al., 1995] [Haridi et al., 1997].

We use the CCP "metaphor" — 'fair interleaving of constraint-based computation processes' — and the logic characterization of attribute–value matrices [Smolka, 1992] to decompose the HPSG ID schemata into quasi-independent components. For instance, the ID schema 3 — eventually combined with the ID schema 5 — decomposes into the following "weakened" ID schemata:[2]

0.	Head – Argument Composition	(H–AC)
1.	Head – ACC Specifier	(H–S1)
1′.	Head – Direct Complement	(H–C1)
2.	Head – DAT Specifier	(H–S2)
2′.	Head – Indirect Complement	(H–C2)
3.	Head – Subject	(H–Subj)
4.	Head – Modifier	(H–Mod)

Such a decomposition is perfectly adapted for (LP) constraint-driven (concurrent) parsing of partially free-ordered phrases. For instance, the Romanian VP is characterized by

— *free* order up to the SUBJECT + COMPLEMENTS component list

— *strict* order inside the SPECIFIERS bloc (finite verb form + clitics) as given by LP constraints like:

$$(\text{Nu}) < \text{CL}_{\text{DAT}} < \text{CL}_{\text{ACC},\sim 3sf} < \text{Avea}_{\text{AUX}+} < \text{Participle} < \text{CL}_{\text{ACC, 3sf}}$$

Decomposing/weakening the HPSG ID schemata would not be needed in case *i.* we adopt for Romanian the fully lexicalized account of clitics introduced for Italian [Monachesi, 1995] [Monachesi, 1997] and French [Sag and Miller, 1997], or *ii.* one is not interested in exploring concurrent/distributed constraint parsing. Concerning the first point, we argued [Ciortuz, 1997] that in our opinion this is a hardly tenable position in case

*The author is engaged now with the LT Lab at DFKI – Saarbrücken, Germany. Email: ciortuz@dfki.de.

[1] This is the case of many human languages like Italian, Romanian, Slavic languages, etc.

[2] We restrict ourselves to what's needed to parse the Romanian transitive verbal complex. Recovering the general case is a straightforward task.

of highly inflected languages (like Romanian), because it leads to exploding with 1–2 orders of magnitude the size of the lexicon, and — independent of this argument — because clitic combination is not so arbitrary (at least for Romanian) as supposed by the papers previously mentioned, so it lacks encoding a certain generality.

Otherwise, i.e. assuming that clitics encode their (conditional) combination capacity as verb's specifiers (cf. [Ciortuz, 1997]), and exploring (the possible benefits of) concurrent parsing, one could see the application of weakened ID schemata as driven by the LP constraints and, possibly, some constraints specific to the language like (CP.C) — Clitic Pronouns Constraint, and (PA.C) — *Pe*-Accusative Constraint in Romanian.

Example for Romanian VP decomposition using the above weakened ID schemata:
the sentence

Cheia, i-am dat-o ieri lui Ion. = *The key, I gave it to John yesterday.*

is parsed as:

	Cheia,		i-		am dat		-o		ieri	lui Ion.
	•				H+AC					
(LP) →	•	<	HS2	<		<	HS1	<		
(CP.C) →	• HC1									HC2
	•								HMod	

Conclusion

We exploit the LP constraints and the possibility of (logic) "atomization" of HPSG ID schemata in order to
— account for the (partial) free-orderness inside the verbal complex in Romanian and
— explore the benefit of (as much as possible) concurrency in the VP parsing.

This approach aims to smoothly adapt HPSG head-driven parsing design primarily for a strict-order language like English to a partially free-order language like Romanian. It was elaborated while trying to provide for a clean explanation of clitics' behavior in Romanian.

References

[Ciortuz, 1997] Ciortuz, L. (1997). Mastering clitics' (dis)order. In *Informal Proceedings of The GL&GE Workshop*, Tuşnad, Romania.

[Haridi et al., 1997] Haridi, S., Roy, P. V., and Smolka, G. (July 1997). An overview of the design of Distributed Oz. In *Proceedings of PASCO '97*, Maui, Hawaii.

[Mehl et al., 1995] Mehl, M., Scheidhauer, R., and Schulte, C. (1995). An Abstract Machine for Oz. In Hermenegildo, M. and Swierstra, S. D., editors, *Programming Languages: Implementations, Logics and Programs, 7th International Symposium, PLILP'95*, Lecture Notes in Computer Science, vol. 982, pages 151–168, Utrecht, The Netherlands. Springer-Verlag.

[Monachesi, 1995] Monachesi, P. (1995). *A grammar of Italian clitics.* PhD thesis, Tilburg University. ITK Dissertation Series 1995-3 and TILDIL Dissertation Series 1995-3.

[Monachesi, 1997] Monachesi, P. (1997). Decomposing Italian clitics. In *HPSG for Romance*, Stanford. Center for the Study of Language and Information.

[Pollard and Sag, 1994] Pollard, C. and Sag, I. (1994). *Head-driven Phrase Structure Grammar.* Center for the Study of Language and Information, Stanford.

[Sag and Miller, 1997] Sag, I. and Miller, P. (1997). French clitic movement without clitics or movement. *To appear.*

[Saraswat, 1993] Saraswat, V. (1993). *Concurrent constraint programming.* The MIT Press, Cambridge, MA.

[Smolka, 1992] Smolka, G. (1992). Feature-constraint logics for unification grammars. *Journal of Logic Programming*, 12:51–87.

[Smolka, 1995] Smolka, G. (1995). The Oz programming model. In van Leeuwen, J., editor, *Computer Science Today*, Lecture Notes in Computer Science, vol. 1000, pages 324–343. Springer-Verlag, Berlin.

[van Noord, 1996] van Noord, G. (1996). An efficient Prolog implementation of the head-corner parser.