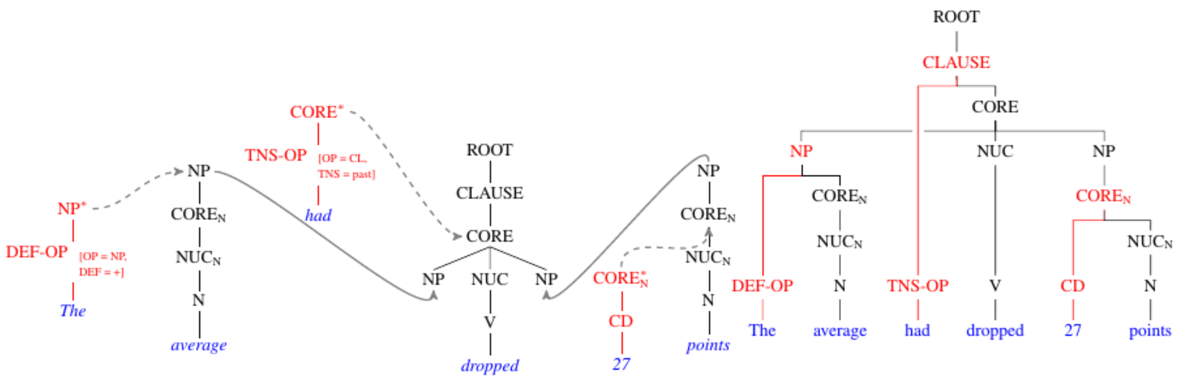
Chart-based RRG parsing for automatically extracted and hand-crafted RRG grammars

David Arps, Tatiana Bladier & Laura Kallmeyer
Heinrich Heine University of Düsseldorf

Introduction. During the last decade, several parsing algorithms for RRG have been proposed. (Guest, 2008) implemented a chart-based parser which uses RRG templates (van Valin, 2005) instead of tree-rewriting rules in order to better capture semantic information. (Diedrichsen, 2008) develops an RRG parser for German using a set of descriptions for syntactic constructions in German along with a richly annotated lexicon. (Cortes-Rodriguez, 2016) proposed an approach to incorporate parsing for RRG into a natural language understanding application "Artemis" based on a lexical construction model. The described approaches are, however, not able to deal with long-distance dependencies and two of them are either language- or application-specific (Cortes-Rodriguez, 2016; Diedrichsen, 2008). In this paper we propose a general parsing algorithm for RRG based on Tree-Wrapping Grammar (Kallmeyer, Osswald, & van Valin, 2013; Osswald & Kallmeyer, 2018),which shares with (Guest, 2008) the idea of a template-based chart parsing. Our parser can be applied to hand-crafted or to computationally induced RRG grammar fragments of various languages. We discuss first parsing results, and we also show an approach to automatically extract a grammar for parsing, that could then be extended with probabilities in order to be used in a data-oriented approach with a statistical chart-based parser.

RRG elementary trees and their syntactic composition. Following (Kallmeyer et al., 2013; Osswald & Kallmeyer, 2018), we adopt a formalization of RRG as a tree-rewriting grammar involving the tree composition operations *substitution* (for argument slot filling), *wrapping substitution* (for argument slot filling combined with "extraction") and *sister adjunction* (for adding operators and periphery elements among others). In other words, starting from a set of elementary trees, larger trees are generated by these operations. Fig. 1 provides an example that involves only substitution (the filling of the two NP argument slots by the elementary trees of "*average*" and "*points*" respectively, indicated by solid arrows) and sister adjunction (adding the operators "*the*" and "*had*" and the cardinal "*27*", indicated by dashed arrows). We choose to avoid crossing branches in cases of mismatches between operator projection and constituency structure (and, similarly, for the periphery). Instead, if needed, operators attach lower. We provide a feature OP on the operator tree that signals its attachment layer in the operator projection, thereby, the intended RRG structure can be retrieved. An example is "*had*" in Fig. 1a, which contributes a clausal operator (OP=cl) but comes with a CORE adjunct tree. In the last step, this operator is re-attached to its intended position as a daughter of CLAUSE. Fig. 1b gives the resulting RRG tree (with the operator projection integrated into the constituency tree).



(a) (b)

Figure 1: Composition of RRG elementary trees (a) and a resulting RRG structure (b).

Elementary trees can be enriched with *feature structures on the nodes* (see the feature [OP=NP] in Fig. 1a) and *features on the edges* that restrict adjunction possibilities during parsing. These can be used for instance to guarantee that operators attach in the correct order [see (Kallmeyer & Osswald, 2017), for more details]. The elementary trees of a grammar can be systematically described within a hand-written grammar or they can be induced from syntactically annotated data. In this paper we pursue the latter approach but the parser can also be used for hand-written grammar fragments.

Chart-based RRG parsing with elementary trees with features. In order to perform parsing with implemented RRG grammars, we developed a CYK-style bottom-up chart parser. Input to the parser are a set of elementary trees and a sentence to parse. The parser returns all derivations for the sentence that can be derived by combining the elementary trees. Node and edge features are stored during parsing. The algorithm first scans the lexical items. From these, it generates the derived RRG constituency tree bottom-up and aims at reaching one or more root nodes that dominate the whole input. Tree com-position operations trigger unification of the feature structures at the nodes at which elementary trees combine. Unification of edge feature structures is the last step of parsing. If unification fails at least once, the derivation is discarded. The parsing algorithm is implemented as an extension of TuLiPA [1], a parsing environment for Tree Adjoining Grammar (TAG) and Frame Semantics (Kallmeyer & Osswald, 2017; Osswald & Kallmeyer, 2018).

Automatic induction of RRG grammar and parsing experiments. We perform a rule-based automatic induction of elementary trees on the RRGbank (Bladier et al., 2018a), an English treebank, in order to extract an English RRG. The trees in this resource follow a special notational variant of RRG, which includes the operator projection in the constituent projection (an example of this notational variant is shown in the RRG structure in Fig. 1b; [see (Bladier et al., 2018a), for more details]). The tree-extraction algorithm uses heuristic percolation tables to distinguish arguments from modifiers along with rules for adding feature information and follows the top-down extraction approach proposed by (Xia, 1999) for TAG.

When the induced elementary trees without any feature structures are used for parsing, the parser over-generates, i.e., besides the correct analysis, it also yields ungrammatical constituency trees. In order to avoid this, we equip the elementary trees with feature structures that can be obtained automatically during elementary tree induction. These feature structures contain linguistic information, such as attachment levels of operators and peripheral elements. We evaluate the usefulness of these features by counting how many ungrammatical parsing results they eliminate. Our special interest will be (i) modelling the operator projection in complex sentences and (ii) raising and control constructions that are modelled by wrapping substitution.

Outlook. In future work, we plan to use the parser both for data-driven parsing with probabilistic large coverage grammars as well as for symbolic parsing of manually developed precision grammar fragments. Concerning the former, we will automatically induce probabilistic grammars for various languages and apply a combination of supertagging, dependency parsing and A∗ chart parsing along the lines of (Bladier, van Cranenburgh, Samih, & Kallmeyer, 2018b; Waszczuk, 2017).We also plan to abstract away from the lexical items in the induced elementary trees. By extracting elementary tree templates based on the POS-tags provided by the treebank, the coverage of the extracted grammar can be increased. Our parsing approach can be extended to cover a variety of typologically different languages, provided the existence of sufficiently large suitable resources (such as, for example, RRGbank (Bladier et al., 2018a)). Concerning precision grammars, we will use the parser as a means to test RRG-analyses of specific phenomena via hand-written grammar implementations.

References

Bladier, T., van Cranenburgh, A., Evang, K., Kallmeyer, L., Möllemann, R., & Osswald, R. (2018a). RRGbank: a Role and Reference Grammar Corpus of Syntactic Structures Extracted from the Penn Treebank. In D. Haug, S. Oepen, L. Øvrelid, M. Candito, & J. Hajič (Eds.), *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018).* Linköping University Electronic Press, Sweden.

Bladier, T., van Cranenburgh, A., Samih, Y., & Kallmeyer, L. (2018b). German and French Neural Supertagging Experiments for LTAG Parsing. In *Proceedings of ACL 2018, SRW* (pp. 59–66).

Cortes-Rodriguez, F. J. (2016). Towards the computational implementation of Role and Reference Grammar: Rules for the syntactic parsing of RRG Phrasal constituents. *Cırculo de lingüıstica aplicada a la comunicación, 65*, 75–108.

Diedrichsen, E. (2008). A Role and Reference Grammar parser for German. In R. D. van Valin (Ed.), *Studies in Language Companion Series. Investigations of the Syntax–Semantics–Pragmatics Interface* (pp. 105–142). Amsterdam: John Benjamins Publishing Company.

Guest, E. (2008). Parsing for Role and Reference Grammar. In R. D. van Valin (Ed.), *Studies in Language Companion Series. Investigations of the Syntax–Semantics–Pragmatics Interface* (pp. 435–453). Amsterdam: John Benjamins Publishing Company.

Kallmeyer, L., & Osswald, R. (2017). Combining Predicate-Argument Structure and Operator Projection: Clause Structure in Role and Reference Grammar. *Proceedings of the 13th International Workshop on Tree Adjoining Grammars and Related Formalisms*, 61–70.

Kallmeyer, L., Osswald, R., & van Valin, R. D. (2013). Tree Wrapping for Role and Reference Grammar. In G. Morrill & M.-J. Nederhof (Eds.), *Lecture Notes in Computer Science / Theoretical Computer Science and General Issues: v.8036. Formal Grammar. 17th and 18th International Conferences, FG 2012 Opole, Poland, August 2012, Revised Selected PapersFG 2013 Dsseldorf, Germany, August 2013, Proceedings* (pp. 175–190). Berlin/Heidelberg: Springer Berlin Heidelberg.

Osswald, R., & Kallmeyer, L. (2018). Towards a formalization of Role and Reference Grammar. In R. Kailuweit, L. Künkel, & E. Staudinger (Eds.), *Applying and Expanding Role and Reference Grammar* (pp. 355–378). Freiburg: Albert-Ludwigs-Universität, Universitätsbibliothek. [NIHIN studies].

van Valin, R. D. (2005). *Exploring the syntax-semantics interface.* Cambridge: Cambridge University Press.

Waszczuk, J. (2017). *Leveraging MWEs in practical TAG parsing : towards the best of the two worlds*. PhD Thesis. Unpublished manuscript.

Xia, F. (1999). Extracting Tree Adjoining Grammars from Bracketed Corpora. *Proceedings of the 5th Natural Language Processing Pacific Rim Symposium (NLPRS-99)*, 398–403.