

Extrapolated relative clauses in Role and Reference Grammar. An analysis using Tree Wrapping Grammars

Laura Kallmeyer

Heinrich-Heine-Universität Düsseldorf

ABSTRACT

This paper proposes an analysis of extrapolated relative clauses in the framework of Role and Reference Grammar (RRG), adopting its formalization as a tree rewriting grammar, specifically as a Tree Wrapping Grammar (TWG). Extrapolated relative clauses are a puzzle since the link to the antecedent noun can be rather non-local but it seems nevertheless appropriate to model it as a syntactic dependency and not a purely anaphoric relation. Moreover, certain types of determiners require their NP to be modified by a (possibly extrapolated) relative clause, and any comprehensive framework should account for this. We show that the tree wrapping operation of TWG, which is conventionally used to fill argument slots out of which some elements have been extracted, can be used to model extrapolated relative clauses. The analysis accounts for the non-locality of the phenomenon while capturing the link to the antecedent NP in a local way (i.e., within a single elementary tree).

Keywords: Role and Reference Grammar, tree rewriting grammars, extrapolated relative clauses, extended domain of locality

INTRODUCTION

1

This paper makes two contributions: first, it proposes a precise and well-defined analysis of extrapolated relative clauses within the

grammar theory of *Role and Reference Grammar* (RRG; Van Valin and LaPolla 1997; Van Valin 2005); and, second, by doing so it develops an analysis of this phenomenon within a tree rewriting grammar formalism in the spirit of *Lexicalized Tree Adjoining Grammar* (LTAG Joshi and Schabes 1997; Abeillé and Rambow 2000) while overcoming the limitations of LTAG when dealing with extra-position.

Extraposited relative clauses are a challenge for any grammar theory due to the possible non-locality of the link between the relative clause and the antecedent (see Walker 2017, for an extensive description of the phenomenon). Some German examples of extraposed restrictive relative clauses are given in (1) (our own examples).¹

- (1) a. Es fängt der *Spieler* an, [*der zuletzt in Portugal*
EXPL starts the player PTCL, who most.recently in Portugal
war].
was
'The player is starting who was in Portugal most recently.'
- b. Ich fahre mit dem *Freund* nach Portugal, [*der gestern*
I go with the friend to Portugal, who yesterday
das Spiel gewonnen hat].
the game won has
'I go to Portugal with the friend who won the game yesterday.'
- c. Es fängt das Team des *Spielers* an, [*der zuletzt*
EXPL starts the team the player PTCL, who most.recently
in Portugal war].
in Portugal was
'The team of the player is starting who was in Portugal most recently.'

¹ Throughout the paper, antecedent noun and relative clause are both in italics, and the relative clause is in additional brackets. In sentences with more than one relative clause, additional indices indicate the respective antecedent-modifier relations.

In some places, abbreviations are used that follow the Leipzig Glossing Rules (Lehmann 1982), for instance EXPL for 'expletive'. Less standard, PTCL in (1) stands for 'particle'.

- d. Es fängt die Figur aus dem Team desjenigen *Spielers*
EXPL starts the figure from the team the.one player
an, [*der zuletzt in Portugal war*].
PTCL, who most.recently in Portugal was
'The figure from the team of the player is starting who was
in Portugal most recently.'

The antecedent in (1a) is an argument of the main verb, in (1b) it is part of an adjunct PP, and in (1c) and (1d) it is embedded in an argument. These examples illustrate that the antecedent of the extraposed relative clause is not necessarily an argument or modifier of the verbal head of the clause to which the relative clause attaches. It can be further embedded, and in principle there is no limit to the level of embedding (see (1d)). Consequently, one needs to find some "non-local" way for the antecedent NP and the relative clause to communicate with each other.

A further example of an embedded antecedent and an extraposed (non-restrictive) relative clause is the following, from Müller (2004), who also points out the non-local character of such dependencies.

- (2) Karl hat mir [eine Kopie [einer Fälschung [des
Karl has me [a copy.ACC [a forgery.GEN [the
Bildes [*einer Frau*]]] gegeben, [*die schon*
painting.GEN [a woman.GEN]]]] given, [who already
lange tot ist].
long.time dead is]
'Karl gave the copy of a forgery of a painting of a woman to me,
who has been dead for a long time.' (Müller 2004)

Grammar theories that are able to establish non-local syntactic dependencies by percolating an arbitrary number of objects (for example a list of identifiers of antecedent NPs that might be modified by an extraposed relative clause) through the constituent tree can deal with such data. The main task is then to constrain the mechanisms for these non-local dependencies in appropriate ways (see Kiss 2005, Crysmann 2013 and Walker 2017 for an HPSG analysis along these lines). In contrast to this, grammar theories that assume an extended domain of syntactic locality, i.e., that have a set of elementary syntactic building blocks that each comprises a predicate together with

its argument slots and adjunction sites for possible modifiers, would preferably choose a local analysis. In other words, they would group the antecedent NP and the extraposed relative clause (or its attachment site) into the same elementary unit. Such approaches, however, usually come with a formalization that assumes rather constrained composition operations for elementary structures, which results in restrictions concerning the non-locality of these dependencies. They therefore often have difficulties with the largely unrestricted character of extraposition. An example of such a formalism is LTAG (Joshi and Schabes 1997; Abeillé and Rambow 2000). To our knowledge, an analysis of extraposed relative clauses in LTAG has not yet been proposed. We discuss different options in Section 5.1 and show that, due to the restricted nature of LTAG's adjunction operation, the formalism is not able to account for extraposed relative clauses with an analysis that models the dependency between antecedent and relative clause as part of an elementary tree and that is in line with standard LTAG assumptions concerning grammar theory, i.e., concerning the form of elementary trees.

In this paper, we start from RRG (Van Valin and LaPolla 1997; Van Valin 2005), a grammar theory that has been shown to be adequate for describing a large range of typologically different languages. We adopt its formalization as a *Tree Wrapping Grammar (TWG)* (Kallmeyer *et al.* 2013; Kallmeyer 2016; Kallmeyer and Osswald 2017; Osswald and Kallmeyer 2018), a tree rewriting grammar along the lines of LTAG but with a larger generative capacity. We will show that this grammar formalism can model the relation between antecedent NP and relative clause as a local dependency, due to the expressive power of the tree wrapping operation.

Note that, in this paper, we only model the syntax of extraposed relative clauses; semantics is left aside. The main goal of the paper is, starting from RRG's assumptions about the form a constituent tree should have, to explain how this tree comes about. In other words, to develop a decomposition of the constituent tree into its elementary building blocks that captures all dependencies and constraints we want to model.

The remainder of this paper is organized as follows. The next section introduces RRG, then gives a more detailed overview of the data we are concerned with, and also introduces TWG and explains the

way RRG is formalized. The analysis we propose for extraposed relative clauses is developed in Section 3, and Section 4 discusses different possibilities to model obligatory extraposed relative clauses. Section 5 compares our approach to others; and Section 6 concludes the paper.

PRELIMINARIES

2

Role and Reference Grammar

2.1

RRG is a non-transformational linguistic theory whose development has been strongly inspired by typological concerns and in which semantics and pragmatics play significant roles. The assumptions RRG makes concerning syntactic structure are guided by the question of what a linguistic theory would “look like if it were based on the analysis of languages with diverse structures such as Lakhota, Tagalog and Dyirbal [...]” (Van Valin 2005, page 1). That is, the syntactic structures underlying RRG cover among others free word order languages such as Dyirbal where a verb and its arguments and adjuncts can apparently be in any order (see Van Valin 2005, page 5 for an example) and where, therefore, a distinction between sentence or clause on the one hand and VP on the other hand does not seem appropriate. In general, RRG’s syntactic structures are rather flat due to the aim to develop something applicable to all varieties of languages.

RRG’s syntactic theory reflects semantic distinctions: One of the basic assumptions of RRG is that clauses have a *layered structure* which reflects the distinction between predicates, arguments, and non-arguments. The *core* layer (category CORE) consists of the *nucleus* (category NUC), which specifies the verb or rather the predicate, and its arguments. The *clause* layer (category CLAUSE) contains the core as well as extracted arguments. Each of the layers can have a *periphery* for attaching adjuncts. Furthermore, operators (e.g., temporal operators, definiteness operators, modals, etc.) are taken to be part of a separate operator projection which is nonetheless linked to the constituent structure. Each operator scopes over a specific layer. Other projections of predicative elements (NPs, APs, etc.) also come with

nucleus and core layers. For such a category XP, the different layers are called NUC_X , CORE_X and XP while for the entire clause, they are NUC, CORE, CLAUSE, and SENTENCE. The latter layer is added to clauses that have illocutionary force.

There are two treebanks of RRG structures currently under construction, which we use as sources for sample RRG trees: RRGbank (Bladier *et al.* 2018),² which constitutes an RRG-based annotation of parts of the Penn Treebank (PTB, Marcus *et al.* 1994), and RRGparbank (Bladier *et al.* 2020a),³ a parallel treebank of Orwell’s 1984 novel, based on the Multext-East 1984 corpus (MULTEXT-East “1984” annotated corpus 4.0, Erjavec *et al.* 2010), and extended with German and French. In the latter, besides English, there are also German, Russian, French and (to a lesser degree) Hungarian and Farsi RRG annotations. In these treebanks, operators and periphery elements are marked as such (category OP or category extension *-peri*) and they attach to the element they scope over/to whose periphery they belong. An example, taken from RRGbank,⁴ is given in Figure 1 with two operators, a tense operator that attaches at the CLAUSE node, and a definiteness operator that attaches at the NP level, and two periphery elements, namely an adjectival modifier attaching at the corresponding CORE_N and a modifier NP attaching at the CORE node.⁵ Punctuation is omitted in the figure. This example is special in that it is a copula construction, therefore the nucleus (“be payable”) is not a verbal predicate but a predication consisting of an auxiliary and an adjectival phrase.

The two treebanks mark extraposed relative clauses by a coreferential index $\text{REF} = 1$, $\text{REF} = 2$, etc., that is shared by the antecedent NP and the relative pronoun, which facilitates the search for these constructions. Many of the examples used in this paper are taken from these treebanks.

Concerning relative clauses, which are modifiers and, consequently, peripheral elements in RRG, RRG makes the following assumptions with respect to their categories and attachment sites: Depending on whether a relative clause is restrictive or not, it modi-

²<https://rrgbank.phil.hhu.de>

³<https://rrgparbank.phil.hhu.de>

⁴RRGbank sentence no. 3921, 12 Feb 2021.

⁵NP = nominal phrase, AP = adjectival phrase, QP = quantifier phrase.

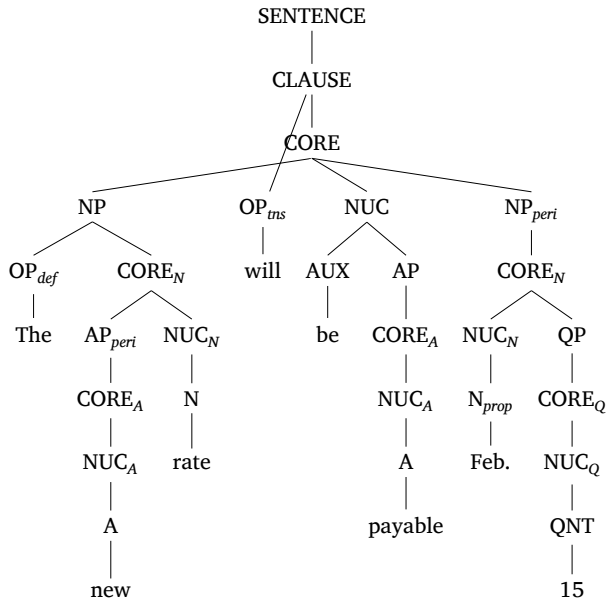


Figure 1:
Layered structure
of the clause, operators
and periphery elements

ties different parts of the NP. A restrictive relative clause provides an additional restriction on the predicate expressed by the NP’s noun. Therefore, in RRG, restrictive relative clauses are considered to be part of the periphery of the nucleus of the NP (Van Valin 2005, Table 7.8, page 267). In contrast to this, non-restrictive relative clauses provide additional information about the NP’s referent, therefore RRG considers them as being part of the periphery of the NP node (Van Valin 2005, Figure 6.29, page 222). Furthermore, a non-restrictive relative clause can have its own illocutionary force and is therefore treated as a SENTENCE constituent in RRG, while restrictive relative clauses are of category CLAUSE. Example (3) gives examples for both such types from the RRG treebanks. The corresponding RRG trees can be found in Figures 2 and 3 (punctuation is omitted).⁶

- (3) a. “That’s the *detail* [*that appeals to me*].”
(restrictive relative clause from RRGparbank)⁷

⁶PrCS = pre-core slot, a position mainly for extracted arguments.

⁷RRGparbank sentence no. 853, en, 12 Feb 2021.

Figure 2:
A restrictive
relative clause
from
RRGparbank

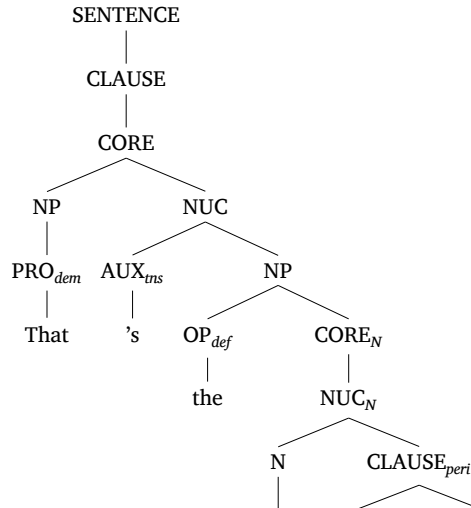
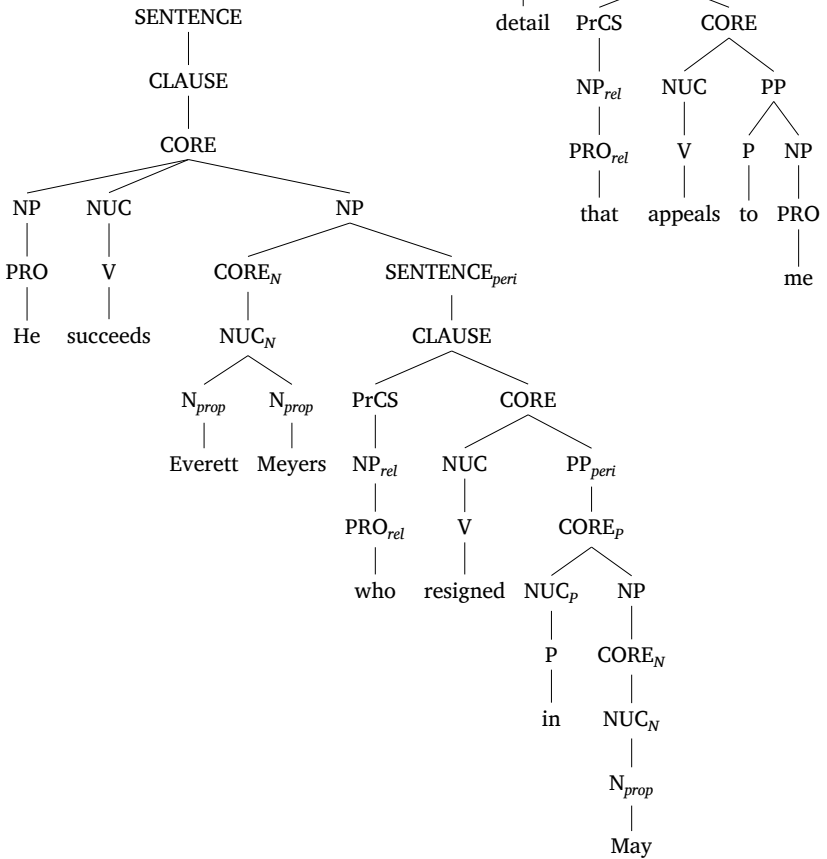


Figure 3:
A non-restrictive
relative clause
from the
RRGbank



- b. He succeeds *Everett Meyers*, [*who resigned in May*].
(non-restrictive rel. clause, RRGbank)⁸

The focus of this paper is on restrictive relative clauses that have an overt antecedent NP, i.e., an NP that they modify.

Extrapolated relative clauses: data

2.2

As mentioned above, restrictive relative clauses can not only appear inside the NP whose nucleus they modify but they can also be extrapolated. Examples are (4b) (from Walker 2017) and (5) (from the RRG treebanks).

- (4) a. *A girl* [*who was singing a song*] came in.
b. *A girl* came in [*who was singing a song*].
(Walker 2017, example (1), page 1)
- (5) a. “You ’ve got *some minds* here [*that wo n’t think progressively*],”
he says.⁹
b. Stratus Computer, which reported *earnings* late Friday [*that were in line with a disappointing forecast*], eased 3/4 to 24 on 816,000 shares.¹⁰
c. “*Nothing* has happened [*that you did not foresee*].”¹¹

In the RRG trees for the sentences in (5), the extrapolated relative clause always attaches to the CLAUSE node that dominates the antecedent NP. The RRG tree for (5c) is given in Figure 4.

In (4b), we have the nucleus (*came in*) in between the NP and its relative clause, and the same holds for (5c). In (5a) and (5b) the constituents that separate the antecedent NP from its relative clause are modifiers of the predication (i.e., the CORE), namely the adverb *here* and the NP *late Friday* respectively.

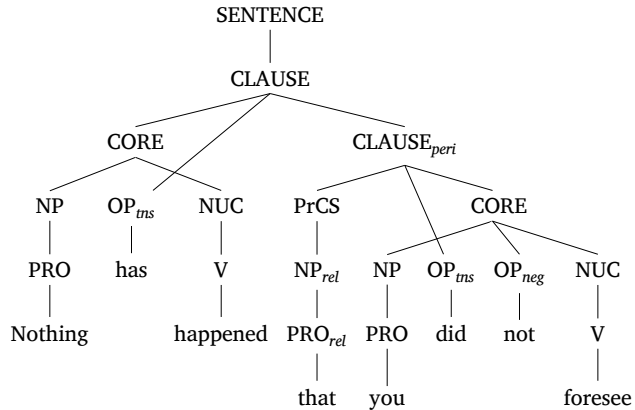
⁸ RRGbank sentence no. 40620, 12 Feb 2021.

⁹ RRGbank sentence no. 31153, 12 Feb 2021. The separations of *You’ve* into two tokens *You* and *’ve* and of *won’t* into *wo* and *n’t* has been a choice of the tokenization of the treebank.

¹⁰ RRGbank sentence no. 24269, 12 Feb 2021.

¹¹ RRGparbank sentence no. 5922, en, 12 Feb 2021.

Figure 4:
Tree from RRGparbank
for an extraposed
restrictive relative clause
(punctuation omitted)



In (4) and (5), the antecedent NP is always an argument of the main verb, i.e., the link between antecedent NP and extraposed relative clause is still local in the sense that the attachment sites for antecedent NP and relative clause are part of the same layered structure, which means that there is a single NUC–CORE–CLAUSE spine such that the antecedent NP is an argument node immediately below CORE and the relative clause attaches at the CLAUSE node. However, as pointed out, among others, by Kiss (2005), Crysmann (2013), Holler (2013) and Walker (2017), this is not always the case: we can also have extraposition in cases where the antecedent NP is embedded within a PP below CORE while the relative clause attaches at the CLAUSE node, both with peripheral (i.e., modifying) PPs as well as with argument PPs, as in (6).

- (6) a. I saw it [in a *magazine*]_{PP_peri} yesterday [*which was lying on the table*].
(Baltin 1978, example (138), page 115)
- b. I arrived [at a *solution*]_{PP_arg} yesterday [*which I found totally unsatisfying*].
(Baltin 1978, example (140), page 115)

One might however argue that in (6a), the PP is not a clear modifier but may be an argument. The two examples in (7) from German (from the RRGparbank), where extraposed relative clauses are more frequent, are two cases where the antecedent NP is part of a PP that is clearly a modifier, i.e., a periphery PP. The same holds for (1b) above.

- (7) a. [...] über die sie [mit einem unumwunden höhnischen
 [...] about which she with an outright mocking
Hass]_{PP_{peri}} sprach, [*der Winston ganz unsicher machte*] [...] _{peri}
 hatred talked that Winston quite uneasy made [...] _{peri}
 ‘about which she talked with an outright mocking hatred that
 made Winston quite uneasy’¹²
- b. [...] dass der Tod seiner Mutter [...] [auf eine *Weise*]_{PP_{peri}}
 [...] that the death of his mother [...] in a way
 traurig und tragisch gewesen war, [*die es heutzutage*
 sad and tragic been had that EXPL these days
nicht mehr gab].
 not more existed
 ‘ [...] that the death of his mother had been sad and tragic in
 a way that did not exist any longer these days.’¹³

Depending on the way PPs are decomposed into elementary building blocks, the link is still relatively close. But the antecedent noun can also be further embedded. Examples were already given in (1c) and (1d). In addition, (8), cited after Walker (2017), gives naturally occurring examples which Strunk and Snider (2013) have found in English corpora, and (9) gives further German examples (a constructed example from Kiss 2005, and three corpus examples, one from TüBa-D/Z, mentioned by Strunk and Snider 2008, and two from the RRGparbank) of extraposed restrictive relative clauses with an embedded antecedent NP. Note that in the case of (9d), there are two extraposed relative clauses, both with a genitive antecedent NP embedded in another NP. Indices indicate which relative clause modifies which antecedent NP. The second, embedded relative clause is definitely restrictive while the first one is rather non-restrictive.

- (8) a. A wreath was placed in [the doorway of [the brick *row-house*]_{NP}]_{NP} yesterday, [*which is at the end of a block with other vacant dwellings*]. (Walker 2017, example (18c), p.16, originally from Strunk and Snider 2013)

¹² RRGparbank sentence no. 2441, de, 02 April 2021.

¹³ RRGparbank sentence no. 517, de, 02 April 2021.

- b. For example, we understand that Ariva buses have won [a number of [contracts for [routes in [London]_{NP}]_{NP}]_{NP}] recently, [which will not be run by low floor accessible buses]. (Walker 2017, example (18d), p.16, originally from Strunk and Snider 2013)
- (9) a. Man hat [die Frau [des Boten]_{NP}]_{NP} beschimpft, [der den one has the wife of.the messenger insulted who the Befehl überbrachte].
command delivered
'The wife of the messenger who delivered the command was insulted.'
(Kiss 2005, example (12), page 4)
- b. Und dann sollte ich [Augenzeuge [der Zerstörung [einer and then should I eye.witness of.the destruction of.a Stadt]_{NP}]_{NP}]_{NP} werden, [die mir am Herzen lag] –
city become that me to.the heart laid –
Sarajevo
Sarajevo
'And then I was about to become an eye witness of the destruction of a city that was dear to my heart – Sarajevo'
(Strunk and Snider 2008, slide 15)¹⁴
- c. Wenn Schauprozesse stattfanden, hatte sie [ihren when public.trials were.happening had she her Platz [unter [der Abordnung der Jugendliga]_{NP}]_{PP}]_{NP} place among the detachments of.the Youth.League eingenommen, [die [...] vor dem Gerichtsgebäude taken who [...] in.front.of the courthouse Stellung bezog [...]].
positions took.up [...]
'When public trials were happening she had taken her place among the detachments from the Youth League who took up positions in front of the courthouse.'¹⁵

¹⁴Tübinger Baumbank des Deutschen / Schriftsprache (TüBa-D/Z), sentence 16294.

¹⁵RRGparbank sentence no. 3125, de, 02 April 2021.

- d. Er begann [gehäuften Löffel [des Eintopfgerichtes₁]_{NP}]_{NP}
 he started heaped spoons of.the stew
 herunterzuschlingen, [in dessen schlüpfriger Masse auch
 swallow in whose slimy mass also
 [Würfel [eines schwammigen, rosafarbenen Zeugs₂]_{NP}]_{NP}
 cubes of.some spongy pink stuff
 auftauchten, [das vermutlich ein Kunstfleischprodukt
 appeared which presumably a artificial.meat.product
 war]₂]₁.

was

‘He started swallowing spoonfuls of the stew, in whose slimy mass appeared cubes of a spongy pinkish stuff which was presumably an artificial meat product.’¹⁶

Given the examples in (1c), (1d) and (8) and (9), an antecedent noun for an extraposited relative clause that is deeper embedded should in principle be possible and an analysis has to account for that.

So far, all examples we considered contained only a single extraposited relative clause. This is, however, no strict limitation. Examples with more than one extraposited relative clause within the same clause sometimes are acceptable. Example (10), cited after Walker 2017, shows two sentences where we have two extraposited relative clauses with different antecedent NPs. An example in German (our own example) is given in (11).

- (10) a. *Someone*₁ picked some *books*₂ up [*which were lying on the table*]₂ [*who really didn’t want to*]₁. (Baltin 2006, page 241–242)
- b. *No one*₁ puts *things*₂ in the sink [*that would block it*]₂ [*who wants to go on being a friend of mine*]₁. (Fodor 1978, page 452)
- (11) *Keiner*₁ wird *die*₂ verraten, [*die nicht jubeln*]₂, [*der selber*
 nobody will those betray who not cheer who himself
am Regime zweifelt]₁.
 the regime doubts
 ‘Nobody who doubts the regime himself will betray those who don’t cheer.’

¹⁶RRGparbank sentence no. 869, de, 05 April 2021.

However, only in certain cases are such multiple extraposed relative clauses acceptable. It might be that the mirroring property mentioned for instance in de Vries (2002) plays an important role, which states that the antecedent–relative clause pairs must have a nested order. That is, if NP₂ follows NP₁, and both are modified by extraposed relative clauses, then the one modifying NP₂ must precede the one modifying NP₁. But this is probably not the only factor responsible for the unacceptability of certain examples. In the examples in (10) and (11), the second, outermost relative clause has pronominal antecedents such as *no one* or *someone*; it seems to be more restricted with respect to the possible antecedent NPs, and the focus structure might play a role.

In this paper, we aim at allowing in principle for multiple extraposed relative clauses with different antecedent NPs but we do not model restrictions on their order in a general way. Within the analysis we propose, the mirroring property could however be modelled as a restriction on derivation order (see Section 3.4.2 for a brief discussion).

Another possibility is to have several extraposed relative clauses modifying the same noun. Example (12a) gives an example of such stacked relative clauses that are not extraposed, and (12b) (our own example, judged acceptable by several native speakers) gives an example with extraposition.

- (12) a. The *theory of light* [*that Newton proposed*] [*that everyone laughed at*] was more accurate than the one that met with instant acceptance. (McCawley 1998, example 3c, page 382)
- b. He explained the *theory of light* to her [*that Newton proposed*] [*that everyone laughed at at the time*].

Note, however, that only the first relative clause clearly is a restrictive relative clause. The second is rather non-restrictive. In contrast, in the following examples (13) and (14), we have several restrictive relative clauses.

Example (13a) is an example of stacked relative clauses from RRGparbank. A variant of this with extraposed relative clauses (our own example) is in (13b), again judged acceptable by several native speakers.

- (13) a. After confessing to these things they had been pardoned, reinstated in the Party, and given *posts* [*which were in fact sinecures*] [*but which sounded important*].¹⁷
- b. After confessing to these things, *posts* were given to them [*which were in fact sinecures*] [*but which sounded important*].

Concerning German, where (due to the verb-final word order) extrapolated relative clauses are more frequent, we found such examples in RRGparbank, see (14).¹⁸

- (14) a. Unzählige Male hatte sie [...] [die Hinrichtung [von numerous times had she [...] the execution of [Menschen]_{NP}]_{PP}]_{NP} gefordert, [deren Namen sie nie people demanded whose names she never zu.vor gehört hatte] [und an deren angebliche Verbrechen before heard had and in whose alleged crimes sie nicht im entferntesten glaubte]. she not in the.least believed
‘On numerous occasions, she had [...] demanded the execution of people whose names she had never heard before and in whose alleged crimes she did not even remotely believe.’¹⁹
- b. [...] wie sie [auf [das Vorbeikommen [von [...] how they for the passing of [Lastautos]_{NP}]_{PP}]_{NP}]_{PP} gewartet hatten, [die gewisse trucks waited had which certain Fernfahrten machten] [und von denen man long.distance.journeys made and of which one wusste, dass sie Viehfutter geladen hatten]; [...] knew that they cattle.feed loaded had [...] ‘[...]] how they had waited for trucks to pass, which made certain long distance journeys and which were known to be carrying cattle feed; [...]’²⁰

¹⁷ RRGparbank sentence no. 1376, en, 12 Feb 2021.

¹⁸ The fact that “zu vor” appears in the first sentence as two tokens (instead of one, which would have been correct) is a tokenization error in the electronic version. In the original text, it is one word (Orwell 2000, page 141, l. 6–7).

¹⁹ RRGparbank sentence no. 3124, de, 12 Feb 2021.

²⁰ RRGparbank sentence no. 3301, de, 12 Feb 2021.

Note that the two examples in (14) are not only examples of multiple extraposed relative clauses but, in addition, display cases of embedded antecedent NPs since in both cases, the antecedent is embedded within an argument NP (resp. PP) of the matrix verb.

One might argue that in these examples of multiple extraposed relative clauses, only a single clause has been extraposed consisting of a coordination of two relative clauses. In RRG, however, two clauses that are coordinated and that (can) have different tense values, form a SENTENCE; since in a CLAUSE cosubordination, i.e., a CLAUSE with two CLAUSE daughters, the two clauses share certain features, such as tense. But, on the other hand, restrictive relative clauses are assumed to be of category CLAUSE. Therefore, the standard RRG analysis would tend to assume multiple relative clauses in these cases, as well as in the extraposed case as in (14). This is also in line with the annotations we find in the RRGparbank.

Besides this rather theory-internal argument, a further point in favour of assuming two different relative clauses instead of a complex one is that we can also have cases where only one of the two relative clauses is extraposed, as in (15). Of course, in this case, neither needs a conjunction but that can be modelled via appropriate features.

- (15) [...] wie sie [auf [das Vorbeikommen [von [Lastautos [die
 [...] how they for the passing of trucks which
gewisse Fernfahrten machten]]_{NP}]_{PP}]_{NP}]_{PP} gewartet
 certain long.distance.journeys made waited
 hatten, [von denen man wusste, dass sie Viehfutter geladen
 had of which one knew that they cattle.feed loaded
hatten] ; [...]
 had [...]
 ‘[...] how they had waited for trucks to pass, which made certain
 long distance journeys and which were known to be carrying
 cattle feed; [...]’

We therefore assume that the sentences in (14) are cases of multiple extraposed relative clauses that do not form a single complex extraposed relative clause. As already mentioned, the fact that the second one needs a conjunction can be captured via some appropriate feature that enforces the adjunction of the clause linkage marker.

An additional complication arises from the fact that some determiners, such as *derjenige* ('the one') in German, require a relative clause (Alexiadou *et al.* 2000; Sternefeld 2008). Examples (16a) and (16b) are grammatical while (16c) is not. In German, the relative clause in this case can be adjacent to its antecedent or extraposed. *Derjenige* used as a pronoun, i.e., without a noun, behaves exactly the same way.

- (16) a. *Derjenige (Läufer)*, [*der zuerst ins Ziel läuft*], gewinnt.
the.one (runner) who at.first into.the goal runs wins
'The runner who finishes first wins.'
- b. *Derjenige (Läufer)* gewinnt, [*der zuerst ins Ziel läuft*].
- c. **Derjenige (Läufer)* gewinnt.

The following examples (17) are actual corpus examples with an antecedent NP *diejenigen* ('those') and an extraposed relative clause, taken from the German part of RRGparbank. In both cases, the relative clause is obligatory.

- (17) a. In gewisser Weise ließen sich *diejenigen* am
in certain way let themselves those most
leichtesten von der Parteidoktrin überzeugen, [*die ganz*
easily of the Party.doctrine convince who totally
außerstande waren, sie zu verstehen].
incapable were it to understand
'In a way, those who were totally incapable of understanding
it, could most easily be convinced of the Party doctrine.'²¹
- b. [...] *diejenigen* zu notieren und verschwinden zu lassen,
[...] those to mark.down and disappear to let
[*die vielleicht gefährlich werden konnten*]
who perhaps dangerous become might
'to mark down and eliminate those who might potentially
become dangerous'²²

Note that the requirement for a restrictive relative clause is actually rather a requirement for some additional specification that could also

²¹ RRGparbank sentence no. 3203, de, 12 Feb 2021.

²² RRGparbank sentence no. 1317, de, 12 Feb 2021.

be met by something other than a relative clause, for instance a genitive NP:

- (18) In einer von D. verfaßten Denkschrift sind alle seine
in a by D. written memorandum are all his
Bauten und diejenigen seiner zahlreichen Schüler
buildings and the.ones of.his numerous pupils
verzeichnet.
listed
'In a memorandum written by D. all his buildings and those of
his numerous pupils are listed.'²³

In such a case, the request for additional information would already be satisfied at the NP node, due to the adjunction of the NP *seiner zahlreichen Schüler* ('of his numerous pupils').

So far, we have concerned ourselves with data showing how non-local the phenomenon of extraposed relative clauses is. There are, however, also limitations on how far apart from each other the relative clause and its antecedent can be. One is the *Right Roof Constraint* (Ross 1967), stating that no maximal projection can be in between the antecedent NP and the clause that the relative clause attaches to (see for instance Crysmann 2013). Examples in (19) (our own examples) illustrate this; further examples can be found in Ross (1967).

- (19) a. [*diejenigen zu notieren*]_{CORE} hat er versprochen, [*die*
those to mark.down has he promised, who
vielleicht gefährlich werden konnten]
perhaps dangerous become might
'He has promised to mark down those who might potentially
become dangerous'
- b. er hat versprochen, [*dass er diejenigen notiert, [die*
he has promised that he those marks.down who
vielleicht gefährlich werden konnten]]_{CLAUSE}
perhaps dangerous become might

²³In: Olbrich, Harald (ed.), *Lexikon der Kunst*, Berlin: Directmedia Publ. 2001 [1989], page 6533. From the Kernkorpus of the Digitales Wörterbuchs der deutschen Sprache, <https://www.dwds.de/d/korpora/kern>, 09 April 2021.

‘He has promised that he marks down those who might potentially become dangerous’

- c. *[dass er *diejenigen* notiert]_{CLAUSE} hat er versprochen, [*die vielleicht gefährlich werden konnten*]

Concerning extraposed relative clauses in German, it has been proposed that the antecedent NPs cannot be in the vorfeld of the sentence (see the discussion in Holler 2013, page 271), i.e., preceding the finite verb when the finite verb is in second and not in final position. Examples are in (20).

- (20) a. *Der Mann hat die Frau getroffen, der im Kino
the man has the woman met who at.the cinema
war.
was
‘The man who was at the cinema has met the woman.’²⁴
b. ?Dem Mann hat sie etwas zugeflüstert, der dort
the man has she something to-whispered who there
steht.
stands
‘She whispered something to the man who is standing there.’
(example (47), p.24, Buring and Hartmann 1997)

Note, however, that (20b), though so marked, is, according to Buring and Hartmann (1997), not ungrammatical. As observed also by Holler (2013), it seems that a contrastive focus on the vorfeld constituent makes such examples much better. Concerning (20a), imagine for instance a situation where there are three men, one went to the cinema, one to the theater and the third one to a concert. And we know that one of them met the woman we are interested in. In that case the following dialogue is perfectly fine:

- (21) Welcher der drei Männer hat nochmal die Frau
Which of.the three men has again the woman
getroffen? – DER *Mann* hat die Frau getroffen, [*der im*
met? – the man has the woman met who at.the
Kino war].
cinema was

²⁴ Example provided by an anonymous reviewer.

‘Which of the three men met the woman again? – The man who was at the cinema met the woman.’

Other examples where the NP in the *vorfeld* is a perfect antecedent for the extraposed restrictive relative clause are the ones in (22).

- (22) a. *Jeder* wird dieses Lied sofort wiedererkennen,
everybody will this song immediately recognize
[*der es schon einmal gesungen hat*].
who it already once sung has
‘Everybody who has already sung this song once will recognize it immediately.’ (our own example)
- b. Nur die *Wanderer* waren erschöpft, [*die den Gipfel erklommen hatten*].
only those hikers were exhausted who the summit climbed-to had
‘Only those hikers were exhausted who had climbed to the summit.’ (Holler 2013, example (30), page 276)
- c. Der fette *Musiker* von Achselroths Tisch kam herein
the fat musician from Achselroth’s table came in
[*der schon einmal bis Kuba gekommen war*].
who already once as.far.as Cuba come had
‘The fat musician from Achselroth’s table came in who had already come as far as Cuba once.’²⁵

This paper is not concerned with modelling focus, which would be necessary in order to capture the (in)acceptability of sentences with a *vorfeld* antecedent for an extraposed restrictive relative clause. Given the preceding examples, we choose to allow any NP, whether in the *vorfeld* or *mittelfeld*, to serve in principle as antecedent to extraposed relative clauses.

Concerning the structure of the relative phrase, it can also be the case that the relative pronoun is not an argument of the verbal head of the relative clause. Two examples from RRGbank (i.e.,

²⁵In Seghers, Anna: *Transit*, Gütersloh: Bertelsmann 1995 [1943], page 175. From the Kernkorpus of the Digitales Wörterbuch der deutschen Sprache, <https://www.dwds.de/d/korpora/kern>, 09 April 2021.

from the PTB) are given in (23). In (23b), the relative pronoun is embedded in an argument and, furthermore, parts of the argument NP (“49% of which”) is stranded, i.e., is positioned inside the CORE.

- (23) a. He assumed the missing piece contained a *gene or genes* [whose loss had a critical role in setting off the cancer].²⁶
b. It said the programs , largely game shows , will be provided by its E.C. Television unit along with Fremantle International , a *producer and distributor of game shows* [of which it recently bought 49 %].²⁷

Such cases, where the relative pronoun is embedded in an argument of the head of the relative clause, can occur in combination with extraposition. A German example from Wikipedia is given in (24), and an example from the German part of RRGparbank is given in (25).

- (24) Räuberschach ist eine Schachvariante, bei der
robber.chess is a chess.variant in which
Schlagzwang besteht und derjenige *Spieler* gewinnt,
capturing.obligation holds and the.one player wins
dessen Spielsteine alle geschlagen wurden.
whose pieces all captured have.been
'Robber Chess is a chess variant in which capturing is obliga-
tory and the player whose pieces have all been captured is the
winner.'
(Wikipedia)²⁸

- (25) Die ungewöhnliche Anlage des Zimmers war zum Teil für
the unusual setting of.the room was partly for
den *Gedanken* verantwortlich, [zu *dessen Verwirklichung* er
the thought responsible to whose realization he
jetzt schritt].
now went

²⁶ RRGbank sentence no. 9028, 13 Feb 2021.

²⁷ RRGbank sentence no. 1153, 13 Feb 2021.

²⁸ <https://de.wikipedia.org/wiki/Räuberschach>, 05 Nov 2019.

‘The unusual geography of the room was partly responsible for the idea that he was now about to realize.’²⁹

Besides these cases of complex relative phrases, we can also have a long-distance dependency within the relative clauses such that the relative pronoun is an argument of an embedded verb. An example from the English part of RRGparbank is given in (26) where the relative pronoun is an argument of the embedded predication *to take a look at*.

(26) “There’s another *room* upstairs [*that you might care to take a look at*],” he said.³⁰

In this paper, we will concentrate on establishing the relation between relative clause and antecedent NP, and we will leave the cases exemplified in (23)–(26) aside, given that the phenomena in these sentences are to a large extent independent from the difficulty of linking extra-posed relative clauses to their antecedents.

2.3

Formalizing Role and Reference Grammar: Tree Wrapping Grammar

In the following, we adopt the formalization of RRG as a tree rewriting grammar, more precisely a TWG (Kallmeyer *et al.* 2013; Kallmeyer 2016; Kallmeyer and Osswald 2017; Osswald and Kallmeyer 2018). A TWG consists of a finite set of *elementary trees* that can be combined into larger trees via *substitution*, *sister adjunction* and *wrapping substitution*. Substitution simply replaces a non-terminal leaf (called a *substitution node*) with a new tree, provided the category of the substitution node and the root category of the new tree are the same and the new tree is not an adjunct tree. Sister adjunction adds a new adjunct tree to a node, provided that the category of the root of the newly added tree and the category of the adjunction site are the same. Adjunct trees are such that the root is marked with an asterisk and below the root, there is only a single daughter tree. This new daughter tree can be inserted at any

²⁹RRGparbank sentence no. 81, de, 12 Feb 2021.

³⁰RRGparbank sentence no. 1853, en, 12 Feb 2021.

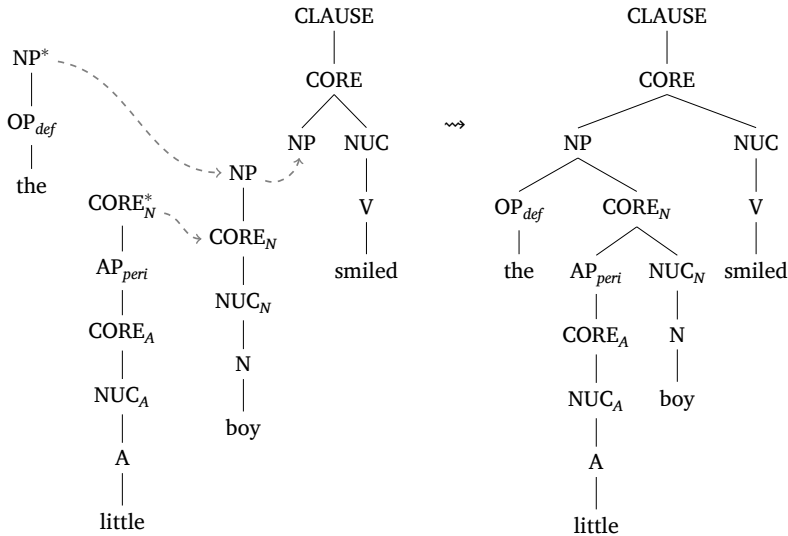


Figure 5: Example involving substitution (argument filling) and operator/periphery adjunction

position among the other daughter subtrees below the adjunction site.

Roughly, substitution is used to add arguments while sister adjunction is used to add operators and periphery elements. A sample derivation involving one argument insertion (substitution), one operator adjunction and one modifier (i.e., periphery element) adjunction is given in Figure 5.

The third operation, wrapping substitution, is the one that adds expressive power to the formalism. It adds a tree with a *d-edge* (= dominance edge) between a node v_1 and its *d-daughter* v_d to a derived tree that has a substitution node with the same category as v_d and an internal node v (which can be the root) with the same category as v_1 . The substitution node is replaced with the subtree below v_d while the node v_1 merges with the node v of the target tree, thereby adding new daughter trees to v (to the left or to the right of the already existing daughters) or new nodes dominating v (the latter is only allowed if v is the root).³¹ Wrapping substitution is used for extraction; the filling of the substitution node adds an argument while the upper part adds

³¹ Note that this is the slightly relaxed definition of wrapping from Bladier *et al.* (2020a).

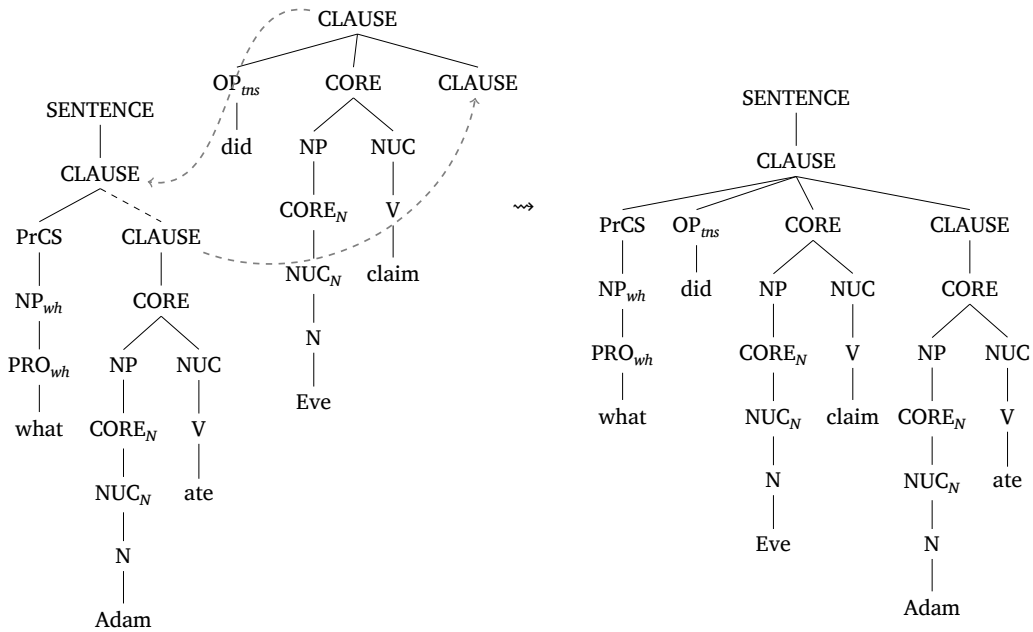


Figure 6: Sample wrapping substitution

material that is extracted out of that argument. A sample wrapping substitution is shown in Figure 6.

As in the case of TAG, nodes can have features, though not bottom and top feature structures but just a single feature structure. As in TAG, feature structures are untyped and restricted in depth such that only a finite set of feature structures is possible. Besides nodes, edges can have left and right features, expressing what is expected to the left/right of a node respectively. We will introduce these features and the way they unify more in detail in Section 4.1.

Note that TWG does not allow for crossing branches, i.e., cannot yield exactly the trees we find in the RRGbank. See for instance Figure 1, where the tense operator *will* attaches at the CLAUSE node, which leads to a crossing branch. Put differently, the yield of the CORE node has a gap. The TWG formalization would attach the tense operator lower while capturing the fact that it scopes at CLAUSE level in the features (see Kallmeyer and Osswald 2017 for more details).

TWG is more powerful than TAG (Kallmeyer 2016). There are two main reasons: a) TWG allows for more than one wrapping sub-

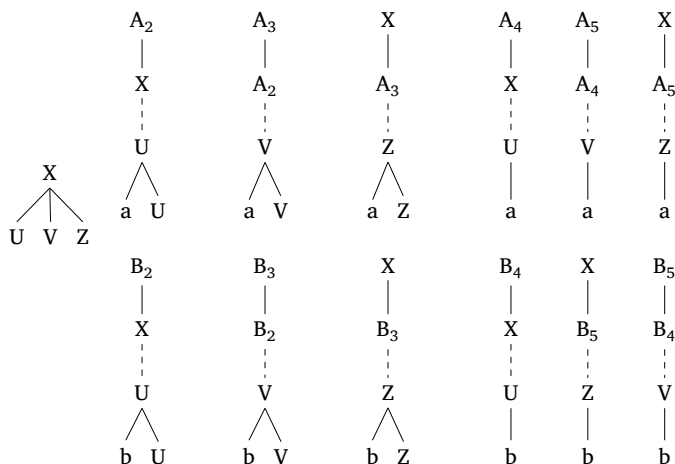


Figure 7: TWG for the double copy language $\{w^3 \mid w \in \{a, b\}^+\}$

stitution stretching across specific nodes in the derived tree; and b) the two target nodes of a wrapping substitution (the substitution node and the higher internal node) need not come from the same elementary tree, which makes wrapping non-local compared to adjunction in TAG. To see why this property matters, consider the sample TWG in Figure 7, which generates the double copy language, a language that is not a tree adjoining language. The idea of this TWG simply is that the new *a*'s (respectively *b*'s) for the three copies are added one after the other from left to right, and the root label always determines which substitution slot has to be filled next. Root label *X* means that *U* has to be filled next, root label A_2 (respectively B_2) means that *V* has to be filled next, and so on. Figure 8 shows a sample derivation with this grammar.

If the number of d-edges that stretch across a certain node and that are not nested within each other is limited to some *k* (this type of TWG is called *k*-TWG), one can show that for every *k*-TWG, a simple Context-Free Tree Grammar (CFTG, Kanazawa 2016) of rank *k* can be constructed (Kallmeyer 2016). Simple CFTGs of rank *k* are, in turn, equivalent to well-nested Linear Context-Free Rewriting Systems (LCFRS) of fan-out *k* + 1. Consequently, 1-TWGs are weakly equivalent to TAG while *k*-TWGs in general are more powerful. The TWG in Figure 7 is a 3-TWG.

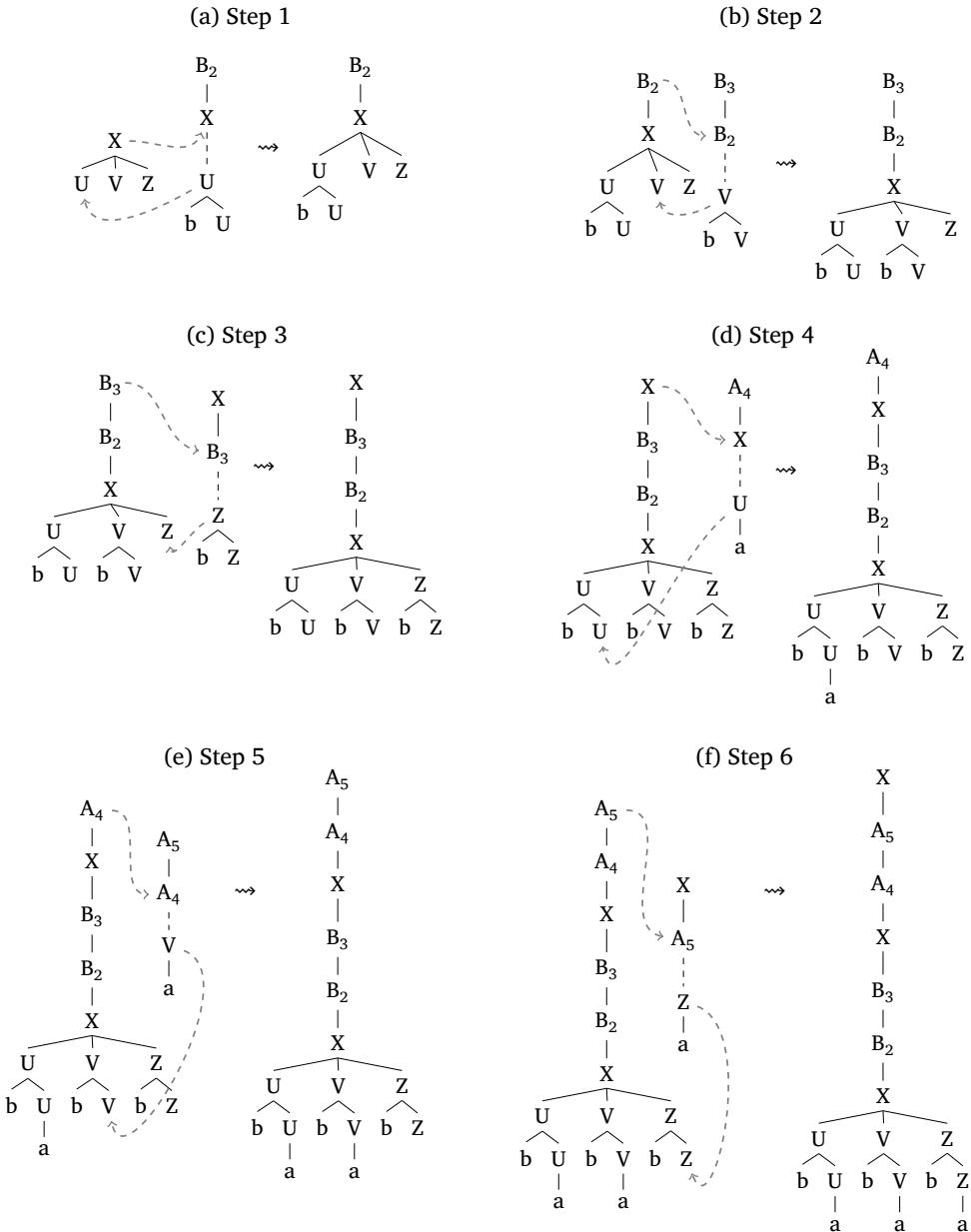


Figure 8: Sample TWG derivation for *bababa*, for the TWG from Figure 7

Given the trees RRG assumes for relative clauses and given the TWG formalization, we will now address the question of how the underlying elementary trees could look and how they might combine. Before coming to extraposed relative clauses, let us start by giving an analysis of (4a) (repeated here as (27a)), where the relative clause is adjacent to its antecedent noun. In this case, we can simply add it as a further daughter to the NUC_N node using sister adjunction. This is our standard way of adding peripheral elements, i.e., modifiers. Figure 9 gives the corresponding adjunction step.

- (27) a. A girl [*who was singing a song*] came in.
 b. A girl came in [*who was singing a song*].

We assume the following features in order to capture the type and scope of the relative clause: The `CLAUSE` node of the relative clause has a feature `PERI` that characterizes the type of category that this relative clause modifies (i.e., of which it is a periphery element). Here, we have a restrictive relative clause, which means that it modifies the nucleus of an NP (which is of category NUC_N). Furthermore, a second feature `PERI-SCOPE` has as its value the identifier of the relevant NUC_N node. Node identifiers are captured within a feature `N(ODE)-ID`. This feature is used to pass the id of the antecedent NUC_N node into the `PERI-SCOPE` value of the relative clause via unification at the sister adjunction site (here unification of ① and ②).

This analysis yields the structures standardly assumed for restrictive relative clauses in RRG (see Van Valin 2005, and see Section 2.1 above) and it is in line with the proposal to use sister adjunction for adding periphery elements (see Kallmeyer *et al.* 2013).

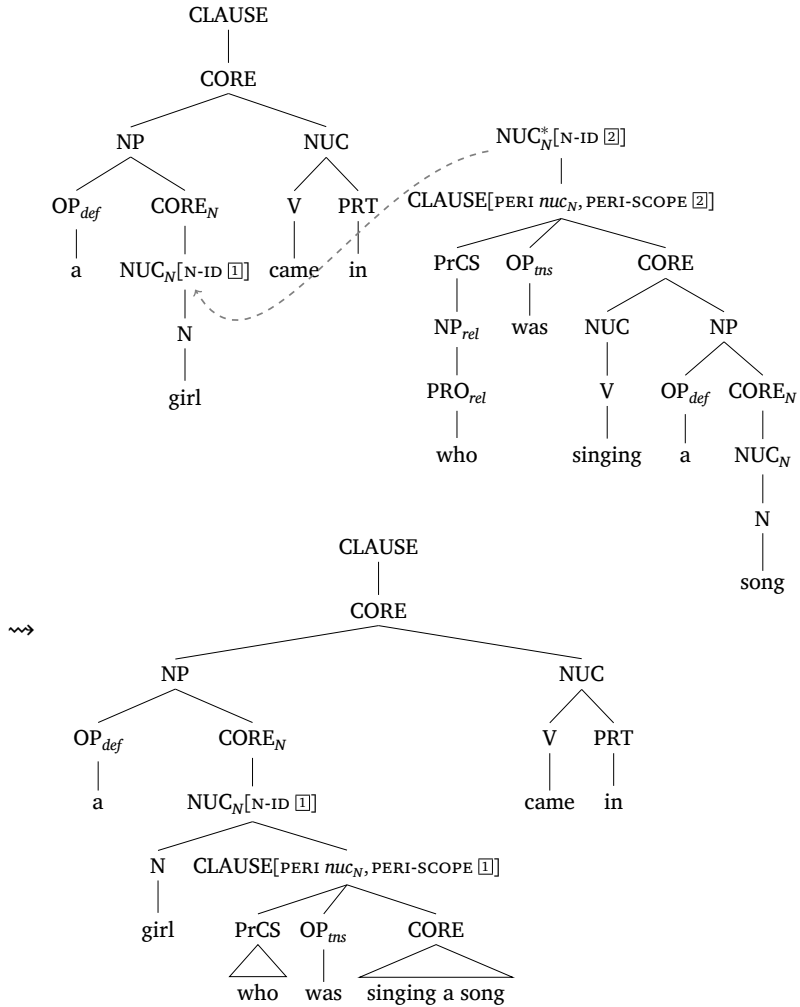
To our knowledge, extraposed relative clauses have not been analyzed in RRG and there is no proposal for extraposed relative clauses in any tree rewriting grammar (such as TAG or variants of TAG). In the following, we will develop and discuss three options.

Approach 1: Anaphoric approach

3.1

Now let us go through several possibilities for analysing (4b) (repeated above as (27b)), i.e., the variation of (4a) with the relative clause be-

Figure 9:
RRG analysis
of (4a) with
relative clause
as periphery



ing extraposed. One possibility is to add the relative clause by sister adjunction to the verbal CLAUSE node and to mark the fact that this is a nominal NUC_N periphery element within the features, as in Figure 10. Within the derived constituency tree, the features of the relative clause tell us that this is a periphery element of some nominal nucleus but they do not specify which NUC_N is the antecedent. In this respect, it contains less information than the structure derived for a non-extraposed relative clause as in Figure 9. In order to find the antecedent, one would have to find an appropriate NUC_N node in one of

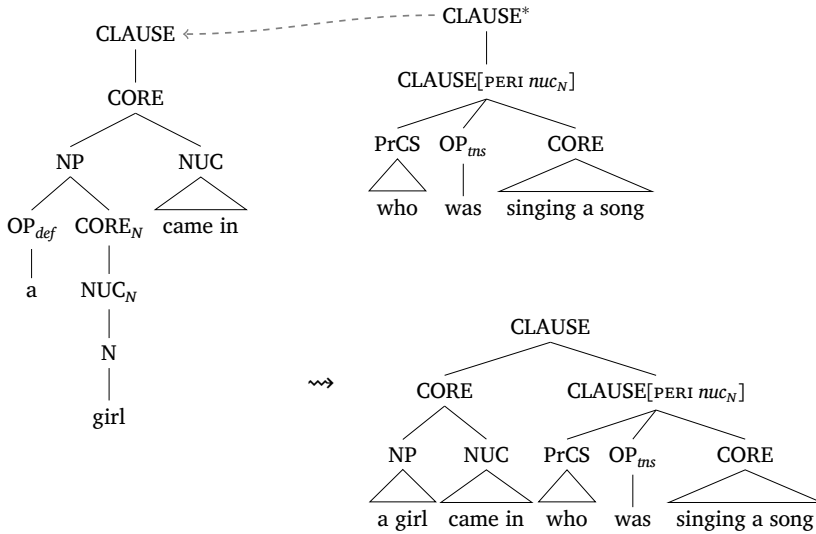


Figure 10:
First possibility
for (4b) (no link
to antecedent
NUC_N)

the sisters of the relative clause tree. This resolution step would be a separate post-processing step.

This first approach is in a sense an anaphoric approach since the linking of relative clause to antecedent is considered an anaphoric link that is established by some non-local process that operates on the derived tree. It is close to what Kiss (2005) proposes as a ‘semantic’ approach where, syntactically, relative clause extraposition is considered as ordinary adjunction, and the link to the antecedent is established via a condition on interpretation (his condition (16), page 7), which states that a suitable antecedent has to be found in the phrase to which the relative clause adjoins. This search for a suitable antecedent can be realized in HPSG via appropriate principles (see the related work in Section 5.2). An anaphoric approach within a tree rewriting grammar such as TAG or TWG would, however, have difficulties capturing syntactic constraints that are due to the syntactic dependency between antecedent NP and relative clause. For instance, agreement between antecedent noun and relative pronoun is not accounted for or, rather, has to be accounted for when resolving the anaphoric link to an antecedent noun, along the lines of pronoun antecedent resolution, which is a choice one can make. More problematic is, however, that there are NPs for which the adjunction of a (possibly extraposed) relative clause is obligatory. Examples are the above-mentioned classes

of determiners, such as *derjenige* in German (see Section 2.2). This is something that one might want to capture within syntax and not in a separate module of anaphora resolution. But an approach that does not establish a syntactic link between antecedent noun and relative clause cannot do so.

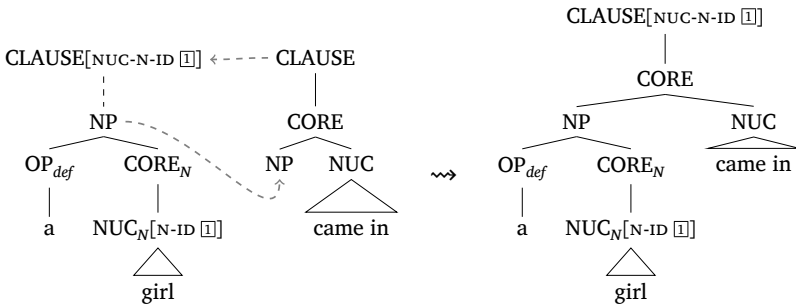
3.2

Approach 2: NPs provide landing sites for relative clauses

The following two options both assume that the antecedent NP and the CLAUSE to which the relative clause attaches are part of the same elementary tree, linked by a d-edge. One such possibility is to have this d-edge in the tree of the antecedent, i.e., add the NP that is modified by the relative clause via wrapping substitution. The upper part of its elementary structure could be a slightly degenerate single node that adds only an identifier. That is, the upper part is a CLAUSE node that identifies with the root of the verbal tree. It adds an identifier of the embedded nominal NUC node in order to provide access to it when adding a modifier. To this end, we use a feature NUC-N-ID (for NUC_N node identifier) on the upper CLAUSE node. The step of adding the NP *a girl* is shown in Figure 11a while Figure 11b gives the subsequent step where the extraposed relative clause is added. It is adjoined to the CLAUSE node but retrieves its antecedent (feature PERI-SCOPE) via the NUC-N-ID on the CLAUSE node, which is the N-ID feature from the antecedent NUC_N node. These two steps of wrapping substitution and adjunction could also be performed in reverse order, i.e., first adjoining the relative clause to the root of the NP tree and then wrapping the NP tree around its predicate. The result would be the same, and, furthermore, the derivation would of course also be the same since the way the elementary trees combine are identical.

Note that this analysis allows also for more embedded antecedent NPs (as in the examples in (8)), as long as they are added by filling a substitution slot. This is due to the non-locality of the wrapping operation: When wrapping a tree γ_1 around some tree γ_2 , the upper part of γ_1 targets some internal node of γ_2 , no matter whether this internal node and the substitution node in γ_2 that gets filled come from the same elementary tree or not.

(a) Step 1



(b) Step 2

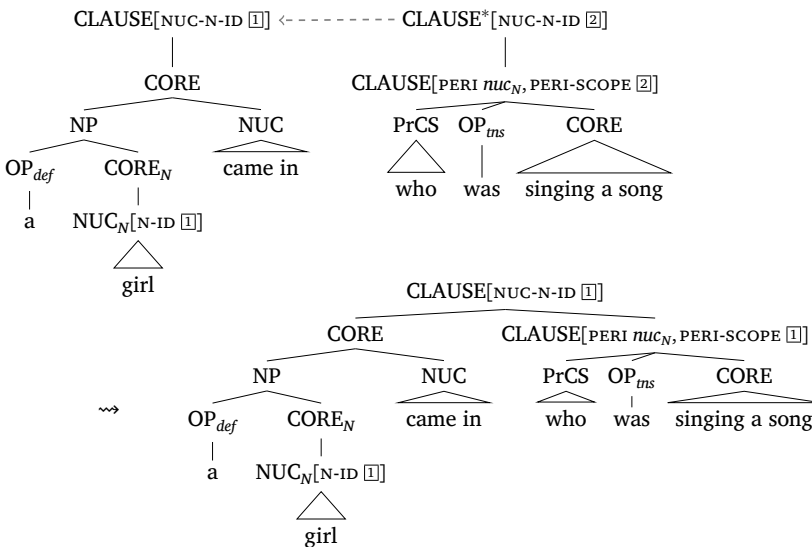


Figure 11:
Second
possibility
for (4b):
the antecedent
provides
a “landing site”
for extraposed
relative clauses

When adopting such an analysis, we need to make sure that at most one NP below a CLAUSE node provides such a node identifier for attaching an extraposed relative clause. Take for instance (28). With trees along the lines of Figure 11a for both NPs (*a girl* with N-ID = 1 and *the room* with N-ID = 2), we would end up unifying NUC-N-ID = 1 and NUC-N-ID = 2 at the CLAUSE node, which would be perfectly possible. The extraposed relative clause would then identify both NUC_N nodes as its scope.

(28) *A girl entered the room [who was singing a song].*

This can be avoided. Instead of using variables (which can unify with each other) as in Figure 11a, we use actual labels drawn from a set of node identifiers as values of the features N-ID and NUC-N-ID in the NP trees, for instance *node_1*, *node_2*, In our feature structure signature, these values would be part of the set of possible attribute values while the variables we use here are of course not part of the feature structure signature. Each nominal nucleus has then its own unique identifier as value of its N-ID attribute that cannot unify with the (different) identifier of the nucleus of a different NP, being different values of the same attribute. In the case of Figure 11a, we might replace the variable $\boxed{1}$ with the attribute value *node_1*. In a sentence with more than one NP, such as (28), we might assume that the first NP (*a girl*) has N-ID = *node_1*, the second (*the room*) has N-ID = *node_2*. If we use trees that provide landing sites for extraposed relative clauses for both NPs, we would have a unification failure at the NUC-N-ID attribute at the respective CLAUSE nodes. Therefore, at most one of them could provide such a landing site.

The second approach comes with the inconvenience that, for each NP, we need an extra elementary structure that is used only for modification of the nucleus with some extraposed relative clause. This is possible but slightly unsatisfying given that the relative clause is a true modifier and should therefore not be anticipated in the elementary structure of the noun. Furthermore, it would lead to spurious ambiguities since such a specialized elementary structure can also be used in cases where no extraposed relative clause is adjoined. An advantage of the second approach might be that it is able to express the fact that for certain NPs such as German *derjenige (N)*,³² the NP comes with the desire to be modified by a restrictive relative clause. In this case, one would provide only the elementary structure with the single CLAUSE node for attaching a relative clause. Note however, that this does not yet require the adjunction of a relative clause. We will discuss ways to impose obligatory adjunction in these cases below.

A more serious problem with this second approach is that multiple extraposed relative clauses with different antecedent NPs (see (10) above) are not possible since only one NP can provide its NUC-ID as

³²*Derjenige (N)* stands for either a pronoun *derjenige* or an NP of the form *derjenige N* as for instance *derjenige Läufer* in example (16), page 241.

a feature at the clause node. The restricted form of feature structures used in TWG does not allow list-valued features, as in HPSG.

*Approach 3: Relative clauses incorporate
their antecedent NPs*

3.3

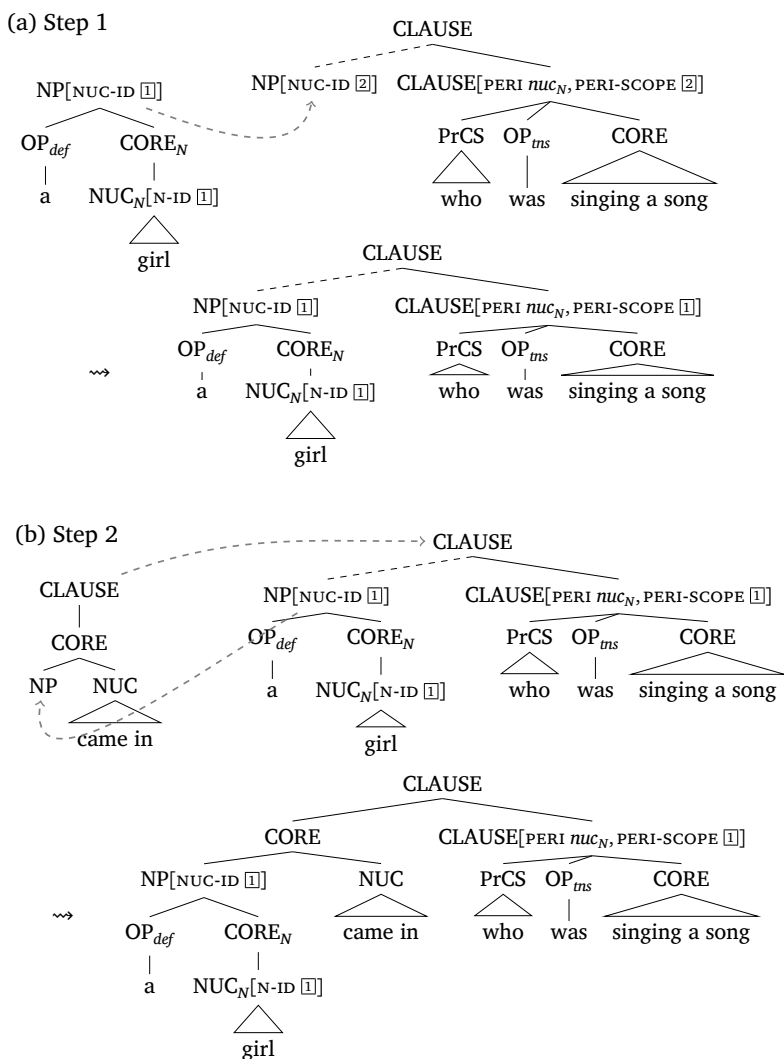
A third and, as we will see, better possibility is to include the d-edge between the CLAUSE node and the NP node in the elementary tree of the relative clause. The NP node can be a leaf node, i.e., a substitution site, that can be filled by the antecedent NP. The combination of antecedent NP and relative clause is a substitution step while the resulting structure is added to the matrix sentence by wrapping substitution. The first step for our example, i.e., combining the NP *a girl* and the relative clause into a complex NP, is given in Figure 12a, and the subsequent step of filling the argument slot of *came in* via wrapping substitution is depicted in Figure 12b. Note that this order is not obligatory; one can also first wrap the relative clause tree around the matrix clause tree (in this case the lower NP leaf merges with the NP substitution slot) and then add the antecedent NP by substitution.

This solution, in contrast to the preceding one, has the advantage that we do not need a special NP tree with a single CLAUSE node, just for the possibility to be modified by an extraposited relative clause. Instead, the NP trees look the same, whether we add a relative clause or not. Furthermore, the problem of accidentally unifying the NUC-ID features of different NPs does not arise since these features do not appear on the CLAUSE node of the matrix sentence, only on the CLAUSE node of the relative clause (feature PERI-SCOPE).

The fact that in this third approach, the combination of extraposited relative clause and antecedent NP is achieved via substitution reflects nicely the semantic argument status of this antecedent NP with respect to the relative clause: In the underlying semantic logical structure, it is either an argument of the predicate denoted by the head of the relative clause or an argument of something embedded within the relative clause.

The third option, in contrast to the second, allows easily for more than one extraposited relative clause, both with different antecedent NPs or with the same. This is because structure sharing concerning the

Figure 12:
Third possibility
for (4b):
computing first
the complex NP
including the
extracted
relative clause



features NUC-ID and PERI-SCOPE only occurs between the antecedent NP node and the corresponding $CLAUSE_{peri}$ node, and it does not involve the upper CLAUSE node, which might serve as attachment site for multiple relative clauses.

In the following, we adopt the third analysis because it easily covers cases of multiple extraposed relative clauses and it does not require special NP trees that anticipate modification by an extraposed relative clause.

Further constraints on extraposition of relative clauses 3.4

Island constraints 3.4.1

As mentioned above, in between the CLAUSE that the relative clause attaches to and its antecedent NP, no further CLAUSE nodes may appear (see Ross 1967). This could be modelled by excluding certain non-terminal categories on the path spanned by a d-edge, in the spirit of V-TAGs *integrity constraints* (Rambow 1994). In other words, for every d-edge in an elementary tree, we allow the specification of “islands”, i.e., of categories that are excluded on the corresponding path. For extraposition of relative clauses, the category CLAUSE would be disallowed. Something similar was proposed in Kallmeyer *et al.* (2013) as a general way to model island constraints in TWG.

Mirroring property 3.4.2

In cases of multiple extraposed relative clauses attaching to the same CLAUSE node, the order of the relative clauses depends on the order in which they are added because each wrapping substitution that fills an NP slot and adds at the same time a corresponding extraposed relative clause, adds this as a new rightmost daughter of the CLAUSE node. If we wanted to restrict the order, for instance according to the mirroring property (de Vries 2002), we could impose a specific derivation order, for instance a filling of argument slots from the right to the left or from the NUC node outwards.

For example in the case of (11), repeated here as (29), we could impose that first the pronoun *die* (‘those’) is added, which would add the corresponding relative clause as new rightmost element below the CLAUSE. Then, in a subsequent step, one moves to the left and adds *keiner* (‘nobody’), which adds the next extraposed relative clause further to the right.

- (29) *Keiner*₁ wird *die*₂ verraten, [*die nicht jubeln*]₂, [*der selber*
 nobody will those betray who not cheer who himself
am Regime zweifelt]₁.
 the regime doubts
 ‘Nobody who doubts the regime himself will betray those who
 don’t cheer.’

3.4.3

Agreement between antecedent and relative pronoun

So far, the antecedent NP node and the relative clause share the value of the respective features NUC-ID on the NP and PERI-SCOPE on the CLAUSE node, in order to establish something like a coreference link between the two or, more precisely, to characterize the scope of the relative clause. In addition, we can of course also share other features between the different nodes of the relative clause tree, in particular agreement features. Take for example (30), where the relative pronoun must have agreement features GEN = *n*, NUM = *sg*, which is the reason why *das* is possible while *die* (features either GEN = *f*, NUM = *sg* or NUM = *pl*) yields an ungrammatical sentence (see the second option in (30)).

- (30) Das Team gewinnt, [*das*/**die* zuerst ankommt].
 the.N team wins that.N/*that.F first arrives
 ‘The team that arrives first wins.’

The relevant derivation is shown in Figure 13. The agreement features (AGR) of the antecedent NP unify with the agreement features of the relative pronoun. This can be achieved via a feature REL-AGR, which is identical with the AGR feature of the relative pronoun. In a case like (30), the latter is also the AGR feature of the whole relative phrase. This is however not always the case. If the relative pronoun is embedded into the NP under PrCS (e.g., *the picture of whom*, *whose daughters*, etc.), the antecedent NP must share its agreement features with the embedded pronoun (transported to the root of the relative phrase via the REL-AGR feature) and not with the entire NP (AGR feature of the relative phrase, variable [2] in our example). The latter plays a role inside the relative clause. In a case of subject relativization like (30) for instance there will be a shared AGR feature between relative NP and the verb of the relative clause (variable [5] in our example). This way of dealing with agreement is very much in line with what we find in TAG, for instance in the XTAG grammar (XTAG Research Group 2001).

4

OBLIGATORY RELATIVE CLAUSES

Now let us turn to the phenomenon that some determiners, such as *derjenige* in German, require a relative clause, see (16), repeated here

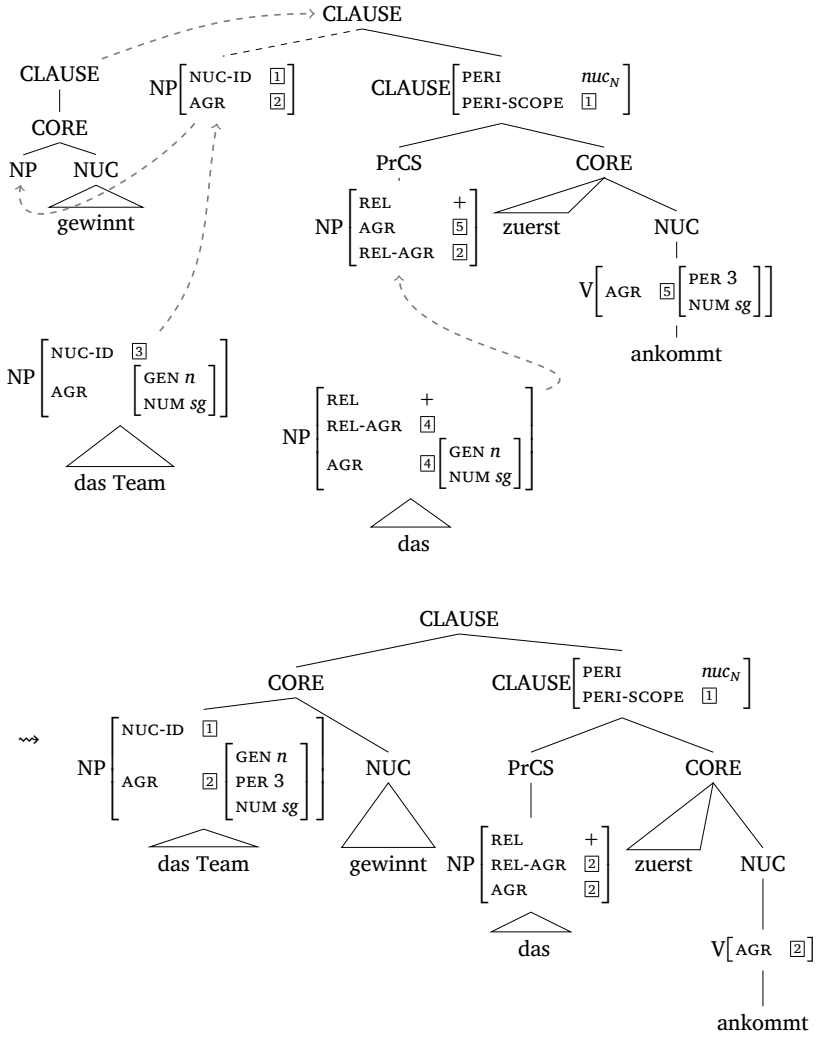


Figure 13: Sharing agreement features between antecedent NP and relative pronoun

as (31). Sentences (31a) and (31b) are grammatical while (31c) is not. The relative clause can be adjacent to its antecedent or extraposed. *Derjenige* used as a pronoun, i.e., without a noun, behaves exactly the same way.

- (31) a. Derjenige (Läufer), der zuerst ins Ziel läuft, gewinnt.
the.one (runner) who at.first into.the goal runs wins.
'The runner who finishes first wins.'
- b. Derjenige (Läufer) gewinnt, der zuerst ins Ziel läuft.
- c. *Derjenige (Läufer) gewinnt.

The difficulty is that we want to express an obligatory adjunction constraint. Within the RRG formalization used in this paper, this is usually done via edge features (Kallmeyer and Osswald 2017) that are shared between neighbouring edges and between edges dominating each other via some automatic feature unification mechanism on the final derived tree. But edge features, as defined in Kallmeyer and Osswald (2017), cannot be shared across substitution nodes.³³ We therefore have to provide some additional way of explicitly enforcing feature unification in these cases, if needed. To this end, in the following, Kallmeyer and Osswald's (2017) analysis will be slightly extended.

4.1 *Edge feature unification on final derived tree*

Edge feature unification is performed only on the final derived tree; during derivation, only node feature structures unify whenever two nodes merge because of substitution, sister adjunction or wrapping substitution. The idea is the same as that of top and bottom feature structures in LTAG (Vijay-Shanker and Joshi 1988). In LTAG, each node has a top and a bottom feature structure. If something adjoins, the two get separated. On the final derived tree, for each node, the top and bottom feature structure have to unify. This creates a means to express obligatory adjunction constraints via a mismatch between top and bottom at the respective node. In TWG, structures are flatter and we use sister adjunction. Therefore, instead of top and bottom on

³³ Edge features are used for instance to keep track of operators and periphery elements and, since substitution nodes are (usually) full projections, they should act as islands concerning these aspects.

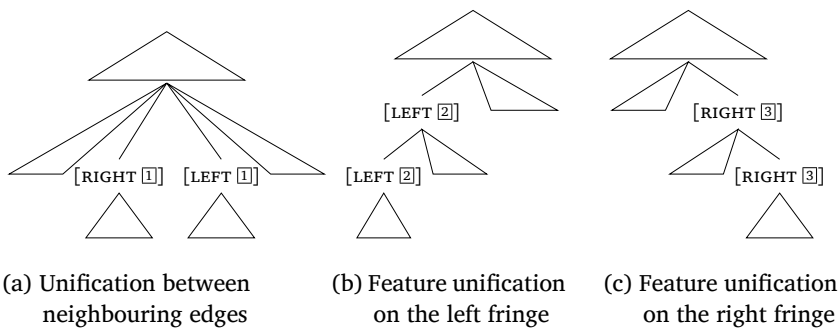


Figure 14:
Final feature unifications

nodes, Kallmeyer and Osswald (2017) use features on the left and the right of edges in order to express constraints on sister adjunction. On the final derived tree, for two neighbouring edges, the right feature structure of the left edge and the left feature structure of the right edge have to unify. Consequently, a mismatch between features on two neighbouring edges acts as an obligatory adjunction constraint for sister adjunction.

We will use the left and right features on edges in order to express and pass the requirement to be modified by a relative clause. In the following we will notate edge features on the daughter node of the corresponding edge, embedded under features `LEFT` and `RIGHT`. This means that the final edge feature unification amounts to unifications between specific `LEFT` and `RIGHT` features on the nodes. Substitution nodes block unification of edge features (Kallmeyer and Osswald 2017) but, if the root node of the tree that is added by substitution has explicit `LEFT` and `RIGHT` features, one can nevertheless have specific features shared between lower and higher edges. (This last option is not used by Kallmeyer and Osswald 2017.) We will use this, in combination with an additional mechanism that allows nodes to look into the left/right edge features on their leftmost/rightmost daughters respectively, as a means to model the obligatory adjunction of an extrapolated relative clause in (31).

Let us briefly explain how edge features work (Kallmeyer and Osswald 2017), in particular how they unify on the final derived tree (see Figure 14). As mentioned, nodes can have, as part of their feature structure, special features `LEFT` and `RIGHT`. In the final derived tree, the `LEFT` feature of a node v unifies with the `RIGHT` feature of its immediate sister to the left (see Figure 14a). Furthermore, the `LEFT`

feature of a node v that does not have a sister to the left unifies with the LEFT feature of the mother of v , provided this mother is not the root node of an elementary tree or the lower node of a d-edge (see Figure 14b). Similarly, the RIGHT feature of a node v that does not have a sister to the right unifies with the RIGHT feature of the mother of v , again provided this node is not the root node of an elementary tree or the lower node of a d-edge (see Figure 14c). (These feature unifications along the left (resp. right) fringe are independent from whether the lower node has a sister to the right (resp. left), i.e., they are also performed for unary edges.) Finally, whenever we substitute a tree with root v into a substitution node v' , the complete feature structures of the two unify, including the features LEFT and RIGHT. This gives us the means to share features even across substitution sites by stating this feature sharing explicitly.

We assume, slightly extending the approach of Kallmeyer and Oswald (2017), that not only do substitution nodes (which are often full projections) block automatic edge feature unification, except if stated otherwise, but so too do the daughters of root nodes in adjunct trees (for instance the $CLAUSE_{peri}$ node below a NUC^* node in a restrictive relative clause tree adjacent to its noun antecedent), and, furthermore, so too do the daughters of root nodes in trees where the only other daughter is linked to the root by a d-edge (as is the case for extraposed relative clauses). This makes sense given that these nodes are often also full projections, for instance in the case of relative clauses, where a clause coming with its own operator projection, i.e., aspect, tense, etc. information, is added to a NUC_N node or, in the case of extraposition, to a $CLAUSE$ node that has its own separate operator projection.

Edge features are mainly used to express obligatory or selective adjunction constraints for sister adjunction. Figure 15 shows for instance how to enforce the adjunction of a tense operator using a boolean edge feature TNS that signals the presence/absence of tense depending on whether it has a value + or -. L and R are short for LEFT and RIGHT. In the example in Figure 15, the value - of the TNS feature on the left of the NUC node unifies with the one on the right of the NP *the girl* (variable \exists), which also occurs on the left of this NP.³⁴ Since

³⁴Note that TNS is not a feature of the NP node but a left/right feature of the

this is the leftmost daughter of the CORE node, the TNS value also unifies with the one to the left of CORE, i.e., embedded in the L feature of the CORE. With the adjunction of *does*, unification of the feature on the left of the CORE with the one on the right of the next sister to the left is possible, as both feature structures are [TNS –]. To its left, the tense operator *does* signals the presence of tense (TNS = +), which can unify with the feature on the right of the PrCS node (which also signals the presence of tense and thereby expresses the requirement of a tense operator). With these features, a tense operator has to adjoin somewhere between the NUC and the PrCS and there cannot be more than one tense operator. Of course, this assumes that tense operators always come with features [L [TNS +], R [TNS –]].³⁵

Features LEFT and RIGHT are supposed to represent features on edges, even though they are notated on the nodes. Their unification does not interact with the proper node features, at least not automatically. But sometimes a node should be able to look into the LEFT feature of its leftmost daughter or the RIGHT feature of its rightmost daughter (in the final derived tree). To this end, in addition to the edge features from Kallmeyer and Osswald (2017), we introduce further node features LEFT-DAUGHTER-EDGE and RIGHT-DAUGHTER-EDGE (LD-EDGE and RD-EDGE for short), which are processed as other features in the context of unifications triggered by substitution or sister adjunction and for which the following holds (see Figure 16): On the final derived tree, the LD-EDGE feature of a node that has daughters unifies with the feature LEFT on the leftmost daughter and the feature RD-EDGE unifies with the feature RIGHT on the rightmost daughter.

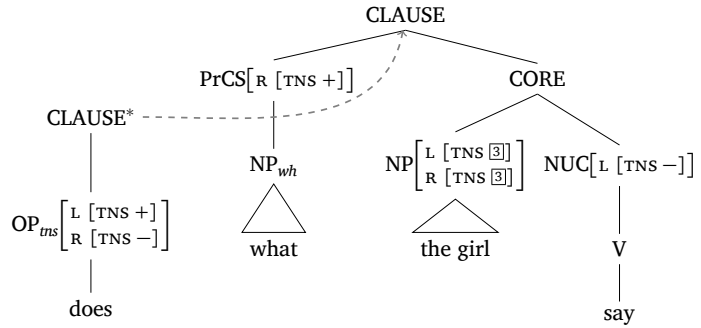
edge between the CORE node and the NP node. The identity between the left and right TNS values signifies that the lower node (i.e., the NP) does not change tense in any way, which is why the information about presence/absence of tense is passed across the edge.

³⁵ As pointed out by a reviewer, in some cases, in particular in English, tense is contributed by more than one element. Such an example is (i).

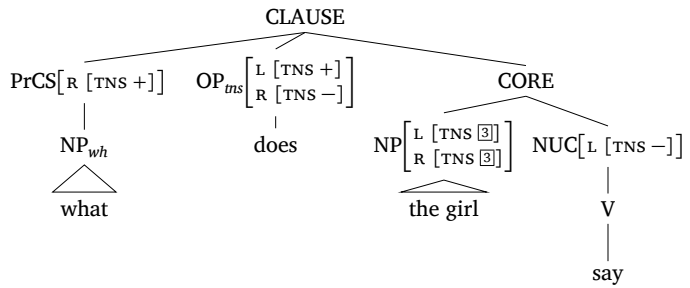
- (i) He should have enjoyed the trip.

In (i), both *should* and *have* would contribute the the overall tense of the sentence (represented by some node feature on the CORE and CLAUSE node, called for instance TENSE), while *should* would be treated as the element that satisfies the requirement of a tense operator encoded in the TNS edge feature.

(a) Derivation



(b) Result before final unification



(c) Result after final unification

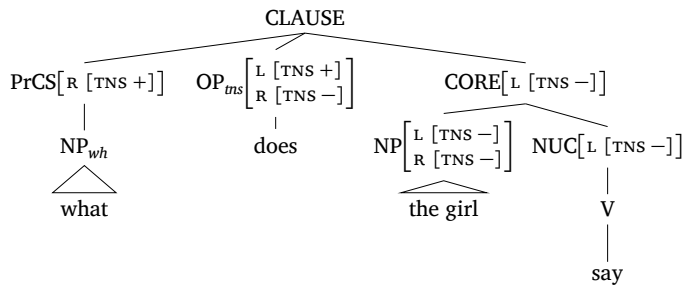


Figure 15: Obligatory adjunction of a tense operator

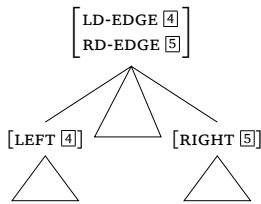


Figure 16:
Final unifications between edge and node features

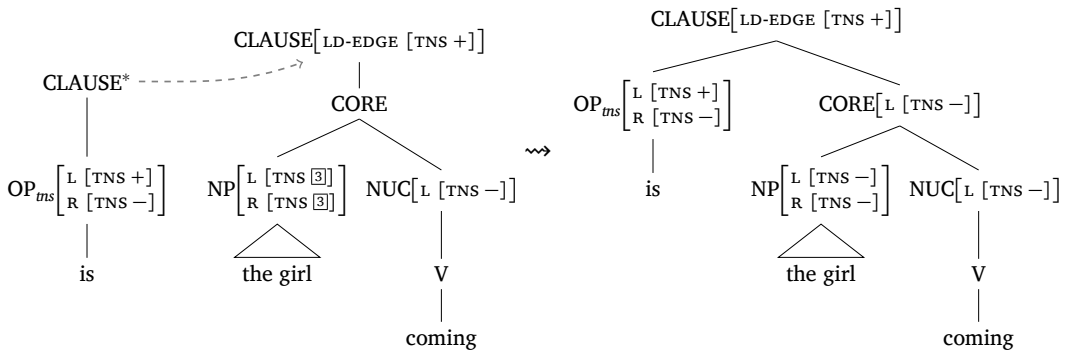


Figure 17: Obligatory adjunction of a tense operator with LD-EDGE features (“Is the girl coming?”)

Note that these features are not only needed in our special case of obligatory relative clauses but also in other cases, for instance when checking for the obligatory adjunction of a tense operator in a case where the tense operator does not have a PrCS sister to the left. Such a use of LD-EDGE in order to express the requirement to have a tense operator somewhere in the tree below is given in Figure 17. Here, the tense operator adjoins to the left of the leftmost daughter of the root node. Consequently, its requirement cannot be expressed using only edge features as in Figure 15. The additional feature LD-EDGE, however, allows us to formulate constraints for the left feature structure on the leftmost edge below the root node (after derivation).³⁶

³⁶ Note that, within LTAG, such obligatory adjunctions are handled via the top and bottom features on nodes and via the distinction between root and foot node in adjoining trees. This is why LTAG does not need features such as LD-EDGE and RD-EDGE with special unification treatments. But this is also why LTAG necessarily generates binary structures when using adjunction. See for instance XTAG Research Group (2001) for a range of analyses that model obligatory adjunction via a mismatch between top and bottom feature structures.

Other cases where a single element has to be adjoined exactly once or at most once below the root node are for instance clause linkage markers (CLM) such as *to* in (32a) and *that* in (32b).

- (32) a. He promised to come.
b. He promised that he would come.

4.2 *An analysis of extraposed obligatory relative clauses using edge features*

In order to capture the requirement for obligatory relative clauses, we introduce a binary feature that expresses that a relative clause has been found or has to be found, REL-CL-EXISTS or REL-EX for short (values + or –). Using the above-mentioned features L(EFT) and R(IGHT) and their unifications on the final derived tree, the relevant constraints can be captured as follows: An NP that requires an extraposed relative clause carries features L(EFT) [REL-EX +] and R(IGHT) [REL-EX –], thereby indirectly expressing that somewhere to the right a relative clause has to be found. The requirement LEFT [REL-EX +], stating that there is (or, rather, has to be) an extraposed relative clause in the final derived tree, is passed upwards to the left while the lack of a relative clause so far, RIGHT [REL-EX –], is passed to the right and upwards. We put the latter on the edge between NUC_N and N (notated, as mentioned, on the N node), which means that it gets passed upwards only if no relative clause adjoins to NUC_N (as in (31a), where the NP node would have a R [REL-EX +] feature, i.e., there would not be any requirement for an extraposed relative clause). An example is the left tree in Figure 18, where the final edge feature unification leads to $\boxed{3} = -$. For other roots of NP trees, these features are not specified, leaving it open whether a relative clause is added.

The tree into which the NP substitutes makes sure any REL-EX features on edges get percolated via edge feature unification across non-leaves towards the outermost nodes and then upwards (in our example, because the NP is the leftmost daughter of the CORE node, this is given anyway for the LEFT feature on the NP, and for the

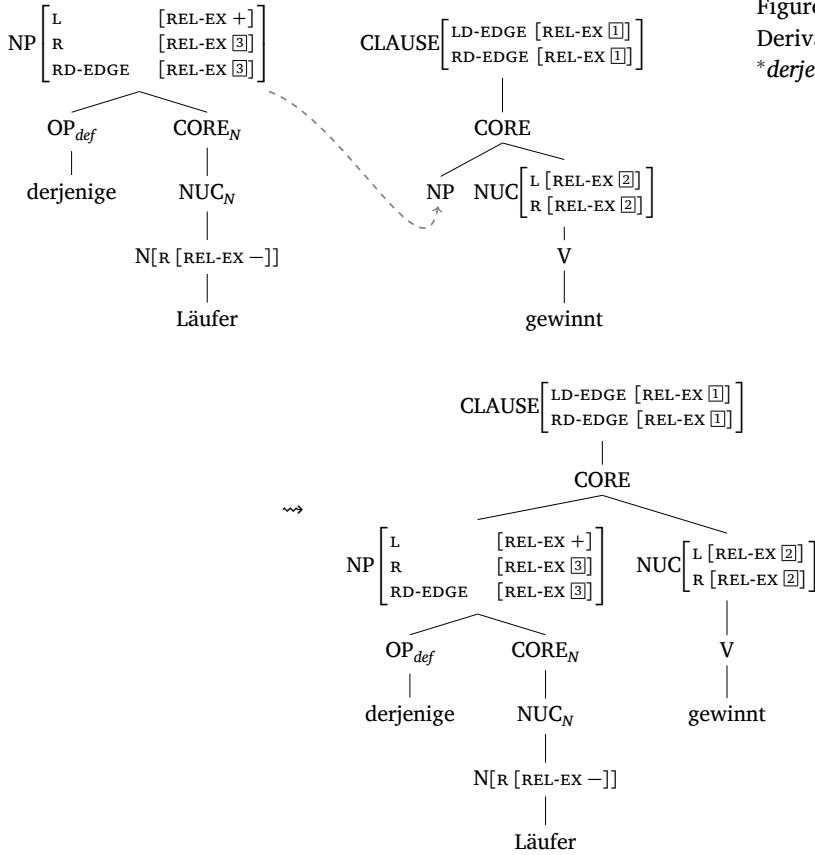


Figure 18:
Derivation of (31c)
**derjenige Läufer gewinnt*

RIGHT feature, we have to make sure it can percolate³⁷ via the NUC node, which is done in the LEFT and RIGHT features on that node). At the root, unification of the information coming from the left (the requirement) and coming from the right (the information on existing relative clauses) is then unified. This last unification, which matches the requirement with what has been found, is done by stating on the CLAUSE node that the REL-EX value on the left of the edge to the leftmost daughter (feature LD-EDGE) has to unify the REL-EX value on the right of the edge to the rightmost daughter (feature RD-EDGE).

³⁷ Strictly speaking, there is no percolation here but only feature unification. The term “percolate” is used to indicate that a specific value is specified in one place and, due to unification, gets passed to other places.

The trees for NP and matrix verb in our example are given in Figure 18, which sketches the derivation for the ungrammatical (31c). If we perform the substitution and then end the derivation, the final unification on the derived tree will fail for two reasons: firstly, the LEFT feature on the NP node unifies with the LEFT feature on its mother and the LD-EDGE feature of the CLAUSE node (as a result, we obtain $\boxed{1} = +$); and, secondly, the RD-EDGE feature on the CLAUSE node (now REL-EX +) has to unify with the right edge feature on the rightmost daughter, which would be the RIGHT feature on the CORE and the NUC nodes (REL-EX –). Consequently, adding a further daughter of the CLAUSE node to the right of the CORE is obligatory, in order to change the REL-EX value on the rightmost daughter of the CLAUSE node to +.

The derivation of (31b) with the extraposed relative clause is given in Figure 19 (the previously introduced features NUC-ID, N-ID and PERI-SCOPE are left aside here for the sake of readability). Instead of combining the NP *derjenige Läufer* directly with the NP substitution slot in the *gewinnt* tree, we have to substitute it into the NP leaf in the relative clause tree, which is then in turn substituted into the subject slot of *gewinnt* via wrapping substitution, adding at the same time the relative clause to the root. The $\text{CLAUSE}_{\text{peri}}$ node has a feature R [REL-EX +], which signals that a relative clause has been attached.

Figure 20 shows the final derived tree: Figure 20a gives the features before final unification, while Figure 20b specifies them after the final edge feature unification.

Let us briefly inspect the structure of the NPs of type *derjenige* (a pronoun) and *derjenige N* (a full NP) more closely. The two cases are given in Figure 21 and Figure 22. In the pronoun case (Figure 21), we can put the + and – values on the left and right of the NP root directly on that node. In the case of *derjenige N*, in the tree for *derjenige*, which adjoins at the NP root of the nominal tree (see Figure 22a), we can put the value + for REL-EX on the left of the OP_{def} node and – on the right. In the final feature unification, once the derivation is finished, this latter gets passed down on the left of the $\text{CORE}_N - \text{NUC}_N - \text{N}$ spine. On the N node we pass it explicitly to the right (see variable $\boxed{8}$ in Figures 22a and 22b). From there, if nothing intervenes (for example a relative clause attaching at NUC_N), the value – gets passed upwards and ends up in the right of the root NP. The NP root

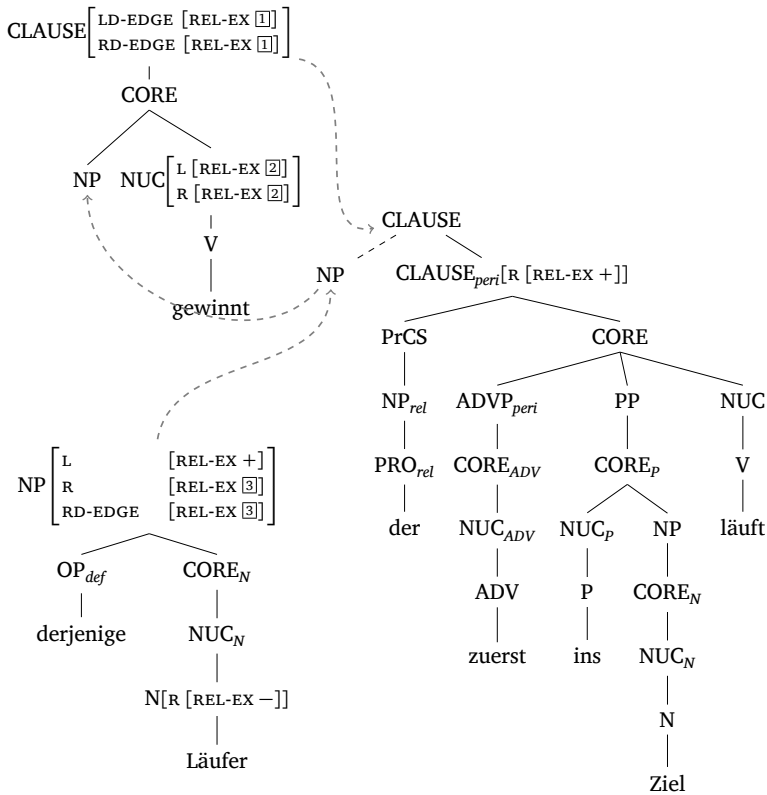
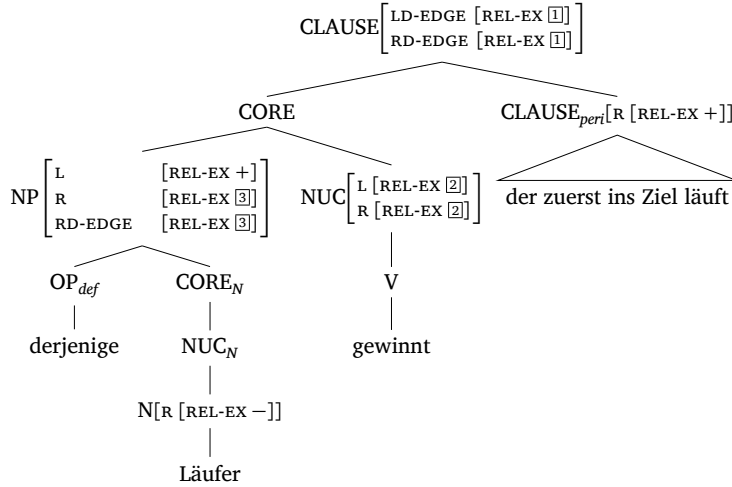


Figure 19:
Derivation
for (31b)

node in the nominal tree does not fix the left and right values; it just states that the one coming from the leftmost daughter has to unify with the one under LEFT on that node and, respectively, the one coming from the rightmost daughter has to unify with the one under RIGHT on the NP node (see variables [6] and [7] in Figures 22a and 22b). As a consequence, if there is no requirement, the two will be equal while in the case of a *derjenige* operator adjoining, the left will be + and the right – (see the derived tree after feature unification in Figure 22c).

Now let us go back to the overall way to model requirements for extrapolated relative clauses with edge features. There is still something missing with the analysis proposed so far: It only guarantees that whenever we have an NP of the form *derjenige* (*N*), we will also have a relative clause. But if this relative clause is extrapolated, it does however not guarantee that it is a modifier of the NP in question, it can also

(a) Before final edge feature unifications:



(b) After final edge feature unification:

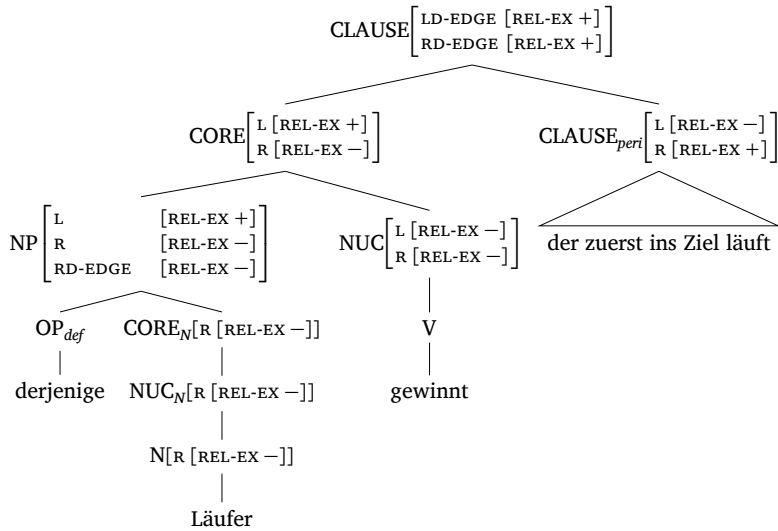


Figure 20: Resulting derived tree for (31b)

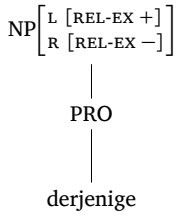
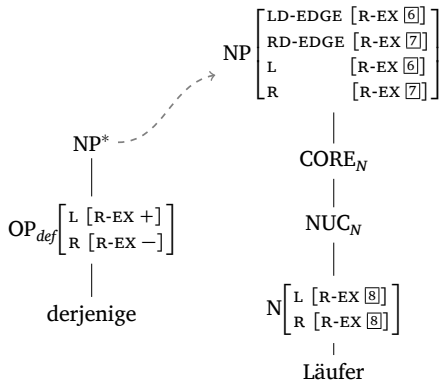
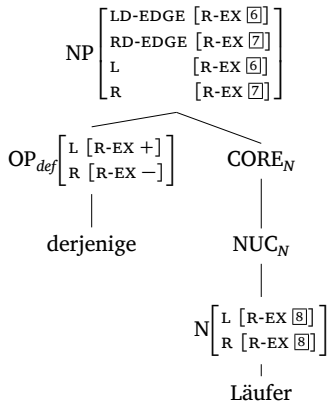


Figure 21: Internal structure of NP of the form *derjenige*

(a) Derivation:



(b) Before final unification:



(c) After final unification:

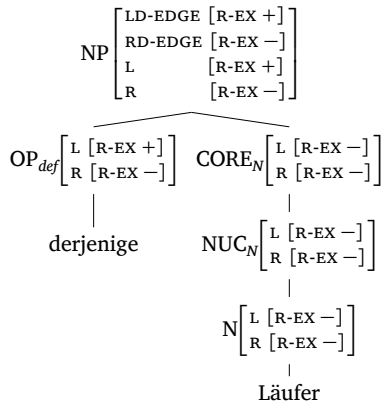


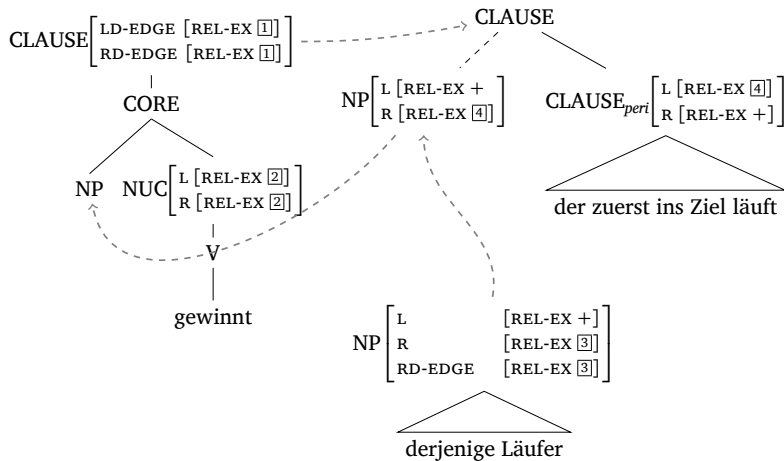
Figure 22: Internal structure of NPs *derjenige* *N* (R-EX is short for REL-EX): derivation and results before and after final unification

modify another NP. In other words, the NP leaf coming with the elementary tree of the relative clause does not necessarily merge with the one of the corresponding antecedent NP. We would for instance also be able to derive the ungrammatical (33) (our own example), where the agreement features of the relative clause do not match those of the NP *demjenigen Mädchen* ('the one girl').

(33) *Der Junge gibt demjenigen Mädchen ein Buch, [der
the boy.M gives the.one.N girl.N a book.N who.M
zuerst den Raum betritt].
at.first the room enters.

In order to enforce a substitution of the correct antecedent NP into the NP node of the relative clause, we add an identity requirement for the REL-EX feature on the left of the edge to the $CLAUSE_{peri}$ node and the REL-EX feature on the right of the antecedent NP. In the case of an extraposed obligatory relative clause, the value on the left of the edge to the $CLAUSE_{peri}$ node is $-$, consequently, the NP also has to have the right REL-EX value $-$. In addition, we impose that the left REL-EX value on the NP node is $+$. Figure 23 shows this extension for our previous example (31b). If we have an NP of the type *derjenige N* in the sentence, only this NP will have different features REL-EX to its left (value $+$) and its right (value $-$), whereas all other NPs have equal values and thereby just pass along what they see to their right/left

Figure 23:
Enforcing
substitution
of correct
antecedent NP
for extraposed
obligatory
relative clauses



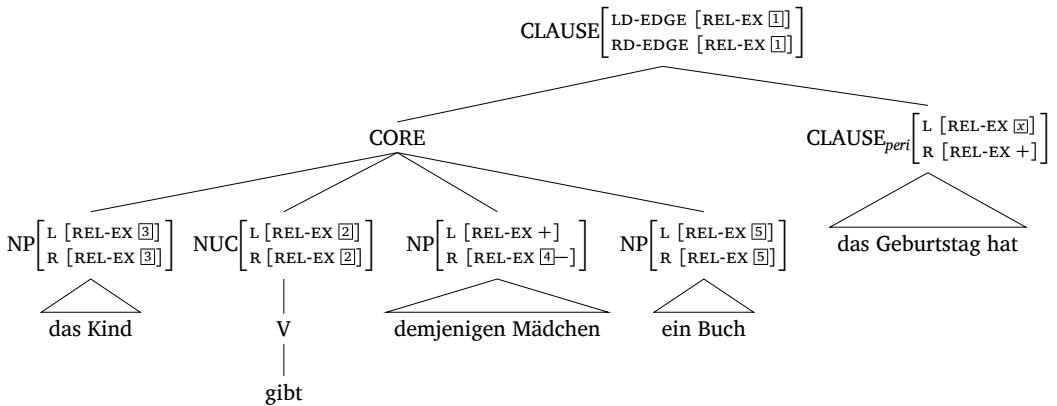


Figure 24: Derived tree for (34) before final feature unifications

respectively. If there is no such NP in the sentence, all REL-EX feature values will be the same, namely either undefined (if no extraposed relative clause is added) or + (if one is added).

As a further example, let us have a look at the tree we would derive for (34), where we have three NPs, all of them with the same agreement features ($GEN = n$, $NUM = sg$) and therefore in principle all of them possible antecedents for the extraposed relative clause (we left aside agreement features for reasons of readability but of course they are taken into account when choosing the antecedent NP, see Section 3.4.3). But since one of the NPs is an NP of the type *derjenige N*, this one necessarily has to become the antecedent.

- (34) Das Kind gibt demjenigen Mädchen ein Buch, [das
the child.N gives the.one.N girl.N a book.N who.N
Geburtstag hat].
birthday has
'The child gives a book to the girl whose birthday it is.'

Figure 24 shows the derived tree before final feature unifications. More precisely, it represents the three different derived trees we would obtain depending on which of the argument NPs was substituted into the NP slot of the extraposed relative clause, i.e., which of the three NPs was chosen as antecedent: The variable \boxed{x} in this tree is a placeholder for either $\boxed{3}$ or $\boxed{4}$ or $\boxed{5}$ depending on whether the *Kind* NP, the *Mädchen* NP or the *Buch* NP was substituted into the relative clause

NP antecedent slot. We will see that the final feature unifications will exclude the first and the last possibility.

Concerning final feature unifications, no matter which NP has been targeted, we always obtain $\boxed{4} = \boxed{5} = \boxed{x} = -$ and $\boxed{2} = \boxed{3} = \boxed{1} = +$ because of the outwards and upwards percolation starting from the NP node of *demjenigen Mädchen*. Consequently, since the NP node in the relative clause elementary tree states that its REL-EX value under R has to unify with the REL-EX value under L at the $\text{CLAUSE}_{\text{peri}}$ node, and since the latter necessarily is $-$, the *demjenigen Mädchen* NP is the only possible antecedent. All other NPs have identical REL-EX values on their left and their right due to the internal structure of the NP. They could be antecedents of non-obligatory relative clauses, with the two REL-EX features in question having a value $+$.

In clauses where an extraposed relative clause is present but is not required because none of the NPs is a *derjenige N* NP, all the REL-EX values would become $+$ since, starting from the right feature on the $\text{CLAUSE}_{\text{peri}}$ they would be passed around. On the other hand, in clauses with no extraposed relative clause (and no requirement for adding one), the REL-EX values would all unify but remain unspecified.

A potential problem of this approach might however be that the feature REL-EX only expresses the requirement for an extraposed relative clause and whether the requirement has been met so far. It does not specify which NP has triggered the requirement. Therefore, this approach hypothesizes that we have at most one NP in a CLAUSE that requires an extraposed relative clause.

The examples in (35) (our own constructed examples) suggest that we can have more than one NP of the form *derjenige (N)* with corresponding extraposed relative clauses in a single sentence but not in the same clause. In the examples in (35), the agreement features of the pronouns *derjenige* and *derjenigen* and of the two relative pronouns leave only one option for the choice of antecedent NPs for the two relative clauses, namely the one expressed by the coindexations. The two examples (35a) and (35b) with the antecedent NPs being arguments of the same verbs are both ungrammatical. It seems that we can have more than one such NP in a sentence only if these NPs (and their corresponding extraposed relative clauses) occur in different CLAUSE subtrees, as in (35c) and (35d), which is possible with our analysis.

- (35) a. **Derjenige*₁ schenkt *derjenigen*₂ ein Buch, [*die als erstes den Raum betritt*]₂, [*der die Wette verloren hat*]₁.
 the.one.M offers the.one.F a book who.F as first the
Raum betritt]₂, [*der die Wette verloren hat*]₁.
 room enters who.M the bet lost has
 ‘The one who lost the bet offers a book to the one who enters the room first.’
- b. **Derjenige*₁ schenkt *derjenigen*₂ ein Buch, [*der die Wette verloren hat*]₁, [*die als erstes den Raum betritt*]₂.
- c. Hans, der heute *denjenigen* abholt, [*den er gestern angerufen hat*], schenkt nächste Woche *derjenigen* ein Buch, called has offers next week the.one.F a book [*die die Wette gewonnen hat*].
 who.F the bet won has
 ‘Hans who fetched today the one whom he called yesterday will next week offer a book to the one who won the bet.’
- d. Hans schenkt *demjenigen*₁ ein Buch, [*der der Bruder von derjenigen*₂ ist, [*die die Wette gewonnen hat*]₂]₁.
 the.one.F is who.F the bet won has
 ‘Hans offers a book to the one who is the brother of the one who has won the bet.’

Note, however, that it is hard to tell whether more than one extrapolated obligatory relative clause attaching to the same clause should be possible, based only on the examples in (35). It might be that this restriction, that comes with our analysis, is a problem. Consider for instance (36) (our own example), where we have one *derjenige* (N) NP embedded in another *derjenige* (N) NP, both with extrapolated relative clauses that attach to the same CLAUSE node. Example (36) seems more acceptable than (35a) and (35b).

- (36) Winston hat dasjenige *Buch*₁ von demjenigen *Autor*₂
 Winston has the.one.N book.N of the.one.M author.M
 ausgeliehen, [*der eigentlich verboten ist*]₂, [*das er aber überraschenderweise in der Bibliothek entdeckt hatte*]₁.
 borrowed who.M actually forbidden is that.N he but
 surprisingly in the library discovered had

‘Winston has borrowed the one book of that author who is actually forbidden, but which he had surprisingly discovered in the library.’

The fact that we needed some non-local feature sharing (via edge feature unification) here in order to capture the obligatoriness of certain relative clauses might seem contradictory to our initial claim that with RRG/TWG, the relation between a relative clause and its antecedent NP can be captured locally, within one elementary tree. Our analysis, however, still captures this relation locally; the only aspect that the shared features capture is the request for some extraposed relative clause. It does not indicate the exact antecedent that requires that relative clause. In this respect, the RRG analysis proposed here still differs fundamentally from HPSG analyses, as in Walker (2017), where information about the actual antecedent NPs is percolated (see Section 5.2 below). In our case, the percolated feature is only a single binary feature, while in the HPSG analyses, it is a list-valued feature that can, in principle, have arbitrarily many different values.

4.3

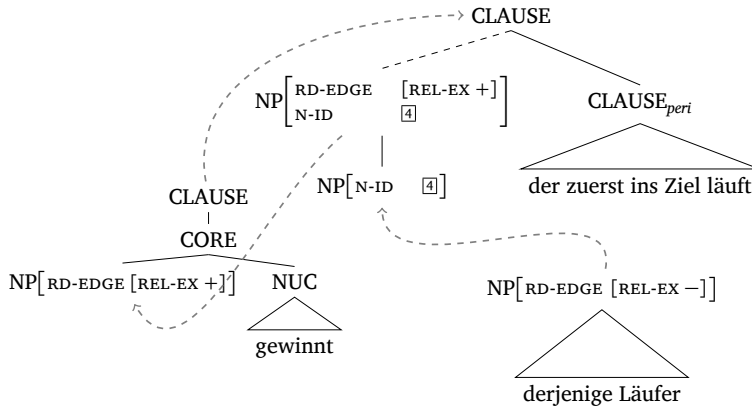
An alternative local feature-based analysis

The reason why we introduced the feature percolation mechanism for REL-EX in the preceding section was that the lower NP node of the relative clause cannot directly change the value of that feature from – to + at the NP root node of the *derjenige*-NP. This is not possible simply because the feature structures of the NP-node of the *derjenige*-NP, those of the relative clause antecedent NP slot and those of the NP argument slot in the tree of the matrix verb all unify. Unification is monotonic, i.e., it can only add information. Changing features is only possible from a node to a different node or between edge features of for instance sister nodes.

Given that the relative clause and its antecedent NP are, however, linked at the NP antecedent node via substitution, it would be more in line with the overall ideas of the grammar theory to take care of the relative clause requirement in some local way at that node.

Furthermore, we have seen that the feature percolation approach to extraposed obligatory relative clauses comes with the constraint to

(a) Derivation:



(b) Result:

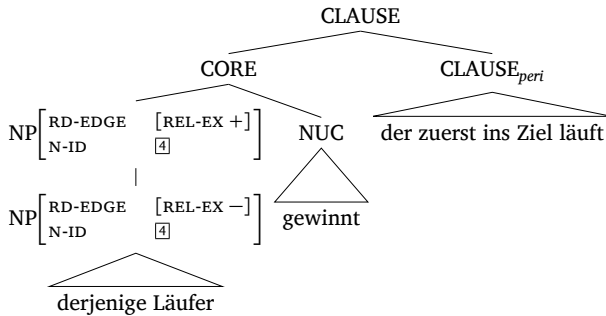


Figure 25:
Local feature use
for enforcing
extraposed
relative clause

have at most one such obligatory extraposed relative clause attaching at the same CLAUSE node, which might be too restrictive.

This observation leads to a different, more local, way of using the REL-EX feature in order to enforce adding a relative clause, exemplified in Figure 25. We replace the single NP antecedent slot in the tree of the extraposed relative clause with two NP nodes, one a daughter of the other, and the higher one carries a feature RD-EDGE [REL-EX +]. That is, no matter whether the NP tree we insert below has a request or not for a relative clause, the root NP node of the tree that fills the argument slot will have RD-EDGE [REL-EX +]. This is also what we require for every NP argument slot, which means that a tree such as the one for *derjenige Läufer* in Figure 25 cannot be substituted into such a slot. For all other features, the two NP-nodes in

the relative clause tree require identity, except edge features, i.e., features `LEFT` and `RIGHT`. In order to signal that they are thought of as two copies of the same node, different only in terms of edge features, we give them the same node identifier (see the shared feature `N-ID`).

Note that the semantics of the feature `REL-EX` is slightly different with this approach, `REL-EX = -` now signifies that there is a so far unsatisfied request for a relative clause, while `REL-EX = +` means that all requests for relative clauses below an NP node have been satisfied. The latter holds also in cases where there are no requests. That is, we can interpret `REL-EX` as meaning something like “any request for an extraposed obligatory relative clause satisfied?”.

This is a simple local way to enforce adding an extraposed relative clause in the case of a *derjenige* NP that does not yet contain a relative clause. The inconvenience is that we have added an extra NP node and a unary branch to the tree. However, if we assume that features `LEFT`, `RIGHT`, `LD-EDGE` and `RD-EDGE` can actually be ignored and therefore deleted once the derivation including the final edge feature unifications is finished, we could perform a merging of identical nodes linked by a unary immediate dominance edge in the derived tree.³⁸

This analysis still allows for antecedent NPs with multiple extraposed relative clauses as exemplified in (14). In these cases, we would obtain three NP nodes in a unary spine in the derived tree that would collapse into one node after final feature unifications.

So far, this analysis does not restrict the number of obligatory relative clauses that can attach below the same clause to one. For such a constraint, we could use a simple boolean edge feature on the edge from `CLAUSE` to `CLAUSEperi`. However, it is not clear whether this constraint really holds (see the discussion of example (36) above). It might

³⁸Note that this local analysis is close to what is performed in the RRG parser implementation described in Bladier et al. (2020b), where such unary branches with copies of nodes are created solely for a technical, parser-internal reason, namely because the parser does not allow d-daughters (i.e., the lower nodes of d-edges) to be at the same time substitution nodes. That is, the parser in Bladier et al. (2020b) introduces a temporary daughter (with an identical label) in these cases that gets deleted after parsing.

actually be an advantage of this approach, compared to the previous, feature percolation based one, that several extraposed obligatory relative clauses attaching to the same CLAUSE node are possible.

It is hard to tell which solution is better: the feature percolation solution in Figure 23 or the one with the extra NP node in the antecedent part of the relative clause trees (see Figure 25). So far, the TWG formalization of RRG is inspired by the idea that long-distance dependencies should arise from tree wrapping (and not from unbounded feature percolation). This points towards the latter option, even though it comes with a slightly unusual unary branch. Concerning predictions that the two approaches make, it might be an advantage of the second that it does not exclude more than one obligatory relative clause at the same CLAUSE node. This, together with a preference for local solutions leads us to opting for this latter solution, keeping in mind that in the final derived tree, we can merge identical nodes linked by a unary branch.

COMPARISON TO OTHER APPROACHES

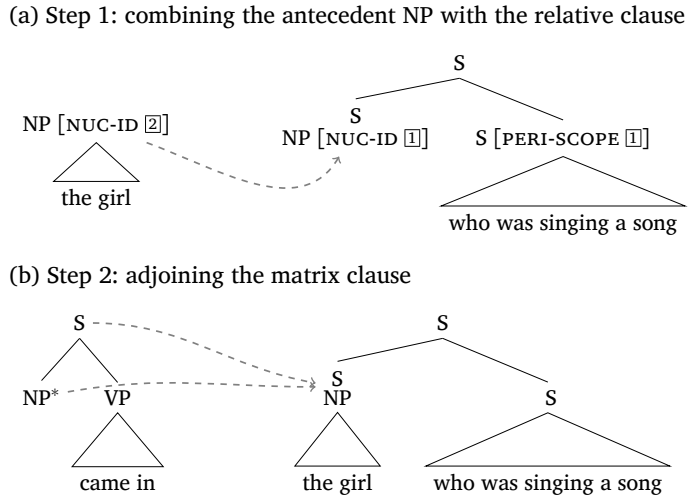
5

Lexicalized Tree Adjoining Grammar (LTAG)

5.1

An LTAG (Joshi *et al.* 1975; Joshi and Schabes 1997) is also a tree rewriting grammar, as TWG, but with different composition operations. Trees can be combined either via *substitution* or via *adjunction*. The latter consists of replacing an internal node with an *auxiliary tree*, which is a tree with a non-terminal leaf node marked as *foot node*. When adjoining, the subtree below the adjunction site ends up below the foot node. Adjunction is more powerful than sister adjunction. It serves, roughly, two purposes: on the one hand, it is used to add modifiers and functional operators; on the other hand, it realizes long-distance dependencies by adding material in between two nodes that come from the same elementary tree. In the case of RRG-TWG, the former is modelled with sister adjunction and the latter with wrapping substitution. Note that the tree added in an adjunction is a (possibly derived) auxiliary tree, i.e., a tree with a single foot node. Its root and foot node always originate from the same elementary tree.

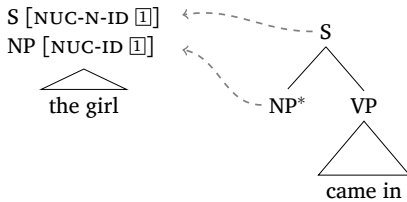
Figure 26:
 Sketch
 of an LTAG
 adjunction
 analysis
 for extraposed
 relative clauses
 along the third
 TWG analysis
 above



To our knowledge, there is no LTAG analysis of extraposed relative clauses. Papers that deal with relative clauses in the context of LTAG are concerned with cases where the relative clause has a complex internal structure, a long-distance dependency for example or a relative pronoun that is embedded into a complex relative phrase (Kahane 2000; Han 2002; Kallmeyer 2003).

LTAG does not easily provide an analysis for extraposed relative clauses that combines antecedent NP or antecedent noun and relative clause in one elementary tree. It is too restricted to provide such a solution, at least with standard LTAG trees: An analysis along the lines of our TWG analysis above (the third analysis, see Sec. 3.3) would amount to adjoining the matrix clause into the relative clause tree, thereby separating the antecedent NP slot from the relative clause. This possibility is sketched in Figure 26. The relative clause tree has a substitution node for the antecedent NP. Deviating slightly from standard TAG, one could allow different top and bottom categories (here: S and NP), which can be seen as different CAT features in the two feature structures. When substituting the antecedent NP into the relative clause tree (Figure 26a), the root feature structure (CAT = *np*) of the incoming *the girl* tree unifies with the lower feature structure, which leads then to the tree on the lower right in the second derivation step. In the second step (Figure 26b), instead of wrapping this around the *came in*

(a) Step 1: combining the antecedent NP with the matrix clause



(b) Step 2: adjoining the relative clause

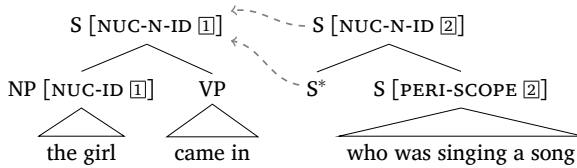


Figure 27:
Sketch of an LTAG
adjunction analysis
for extraposed relative
clauses along the second
TWG analysis above

tree, as in TWG, one could adjoin the latter to the relative clause tree, following LTAG's general strategy of doing extraction by adjunction.

Such a solution, however, would exclude cases where the antecedent NP (the foot node in the adjoining tree) is not part of the argument structure of the matrix clause. Put differently, the root and the foot node of the adjoining tree have to come from the same elementary tree. This is a crucial difference to TWG where, in a wrapping step, the two target nodes can come from different elementary trees.

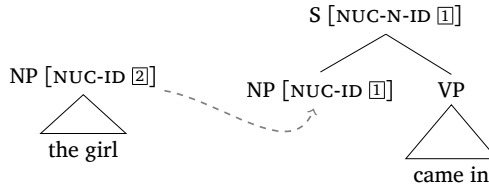
An LTAG analysis along the lines of our second analysis option (Section 3.2) would, roughly, look as exemplified in Figure 27. Here, again, the S that is the adjunction site for the relative clause and the NP antecedent node have to be part of the same elementary tree.

Yet another possibility would be to let the matrix verb anticipate the adjunction of an extraposed relative clause for one of its argument NPs, see Figure 28.

In all three cases, it would be possible to further embed the antecedent NP but only by adjoining material in between the S and the NP node, not by substitution. A second problem is that the matrix clause tree has an NP leaf for the antecedent NP, which, according to standard LTAG principles, means that this NP is an argument. But

Figure 28:
 Sketch of an
 LTAG adjunction
 analysis
 for extraposed
 relative clauses
 where the matrix
 verb anticipates
 the extraposed
 relative clause

(a) Step 1: combining the antecedent NP with the matrix clause



(b) Step 2 (adjoining the relative clause): see step 2 in Figure 27

the antecedent NP can also be part of an adjunct, in which case it is unclear how to model that.

A more severe problem of all three options would be that they exclude multiple extraposed relative clauses with different antecedent NPs since LTAG elementary trees can have at most one foot node, i.e., we cannot have more than one NP* node in a tree (this limits the options in Figures 26 and 27), and we can provide at most one value for the NUC-N-ID feature at the S node of the matrix verb (this limits the options in Figures 27 and 28). This concerns the second crucial difference to TWG where we can have more than one d-edge stretching across a node.

The only solution TAG can easily offer is the anaphoric approach, where a subsequent process on the derived tree determines the antecedent of the relative clause. This would look like step 2 in Figure 27 but without the features that link the antecedent NP to the scope of the relative clause. It is, however, not clear how such a subsequent process of relating relative clauses with their antecedents could look, given the limited possibilities coming with LTAG’s use of feature structures.

5.2

Head-Driven Phrase Structure Grammar (HPSG)

Following Kiss’ 2005 theory of Generalized Modifiers in HPSG, Walker (2017) proposes an HPSG analysis along the following lines: The “extraposed relative clause is base-generated”, and “an anchor that percolates throughout the tree is used to establish the relationship between the relative pronoun and its antecedent” (page 159). A set-valued attribute ANCHORS is used to collect referential phrases that are antecedents of relative clauses. Like HPSG’s SLASH feature, it

is part of the values of the attributes INHERITED and TO-BIND under NONLOCAL. The anchors are passed upwards as elements of INHERITED ANCHORS, and when encountering a relative clause, an appropriate element on the anchors list is identified with an index value on the ANCHORS set under the feature MOD of the relative clause.

In order to account for obligatory relative clauses with *derjenige* (*N*) NPs, Walker (2017) imposes that at the root of the entire tree, the ANCHORS set must be empty. NPs can introduce anchors but need not do so, except for *derjenige* (*N*) NPs where the introduction of an anchor is obligatory.

A crucial difference to the RRG-TWG approach proposed in this paper is that TWG makes use of its extended domain of locality, connected to the operation of wrapping substitution, in order to group the antecedent NP node and the relative clause into one elementary tree. This would not be possible for HPSG, which is lacking an extended domain of locality. On the other hand, a SLASH or ANCHORS feature percolation analysis along the lines of HPSG is not possible for RRG-TWG because of the more restricted types of feature structures used on nodes and edges. TWG uses only a finite set of feature structures, which is crucial for not extending its generative capacity beyond mildly context-sensitive languages. This, however, excludes set- or list-valued features. Even the percolation techniques proposed above in our first approach for dealing with extraposed obligatory relative clauses assume that there is at most one such request or pronoun that is dealt with in a specific node. (There might be of course more than one in an entire tree but in different parts of the tree.)

This illustrates the fundamental difference between, on the one hand, tree rewriting formalisms that come with an extended domain of locality (TAG, TWG) but with restricted tree composition operations and therefore a restricted generative capacity; and, on the other hand, formalisms such as HPSG without a notion of extended domain of locality but with an increased generative capacity due to a highly expressive logic. The former frequently enable a local analysis of non-local dependencies but are sometimes too restricted. We claim that the expressive power of TWG is sufficient to deal with a large range of phenomena in an appropriate way.

CONCLUSION

In this paper, we have developed an analysis of extraposed relative clauses that establishes the link between a relative clause and its antecedent NP in a local way in the sense of placing them in the same elementary building block. The analysis is formulated in the theory of Role and Reference Grammar, assuming its formalization as a Tree Wrapping Grammar. It can account for embedded antecedent NPs, multiple extraposed relative clauses, and extraposed obligatory relative clauses. We have shown that tree wrapping allows us to deal with this phenomenon in a local way, i.e., by comprising the relative clause and the slot for its antecedent NP in the same elementary tree. There is no need for unlimited feature percolation across the derived tree, even for obligatory relative clauses (if a slightly unusual form for the slot of the antecedent NP is used).

The paper contributes a detailed and formally precise analysis of extraposed relative clauses within RRG, a topic that has not been considered so far within this grammar theory. Furthermore, and even more importantly, it proposes an analysis of this phenomenon in a tree rewriting grammar formalism inspired by LTAG but extending it. It addresses the fact that a restricted tree rewriting operation such as LTAG's adjunction allows for an elegant analysis of certain long-distance dependencies (Kroch and Joshi 1987) while being in some cases too restricted. The use of tree wrapping instead of adjunction gives us a less restricted operation for long-distance dependencies that can also model rather non-local phenomena such as extraposed relative clauses in a local way, i.e., with the long-distance dependency originating from a single elementary tree.

ACKNOWLEDGEMENTS

This paper has benefitted immensely from numerous discussions with Rainer Osswald and Robert Van Valin. Their valuable comments helped considerably towards improving the analyses and the paper. Furthermore, special thanks go to three anonymous reviewers, whose

detailed comments contributed a lot to better organizing the paper, including more data and being clearer and more precise in the analyses and the comparisons with other approaches. This work was carried out as a part of the research project TreeGraSP funded by a Consolidator Grant of the European Research Council (ERC).

REFERENCES

- Anne ABEILLÉ and Owen RAMBOW (2000), Tree Adjoining Grammar: An overview, in Anne ABEILLÉ and Owen RAMBOW, editors, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*, pp. 1–68, CSLI.
- Artemis ALEXIADOU, Paul LAW, André MEINUNGER, and Chris WILDER (2000), Introduction, in Artemis ALEXIADOU, Paul LAW, André MEINUNGER, and Chris WILDER, editors, *The syntax of relative clauses*, volume 32 of *Linguistics Today*, pp. 1–51, John Benjamins, Amsterdam and Philadelphia.
- Mark R. BALTIN (2006), Extraposition, in Martin EVERAERT and Henk VAN RIEMSDIJK, editors, *The Blackwell Companion to syntax, volume 2*, pp. 237–271, Blackwell Publishing, Malden, MA.
- Mark Reuben BALTIN (1978), *Toward a theory of movement rules*, Ph.D. thesis, Massachusetts Institute of Technology.
- Tatiana BLADIER, Laura KALLMEYER, Rainer OSSWALD, and Jakub WASZCZUK (2020a), Automatic extraction of Tree-Wrapping Grammars for multiple languages, in *Proceedings of the 19th International Workshop on Treebanks and Linguistic Theories*, pp. 55–61, Association for Computational Linguistics, Düsseldorf, Germany, doi:10.18653/v1/2020.tlt-1.5, <https://www.aclweb.org/anthology/2020.tlt-1.5>.
- Tatiana BLADIER, Andreas VAN CRANENBURGH, Kilian EVANG, Laura KALLMEYER, Robin MÖLLEMANN, and Rainer OSSWALD (2018), RRGbank: A Role and Reference Grammar corpus of syntactic structures extracted from the Penn Treebank, in *Proceedings of Treebanks and Linguistic Theories*, pp. 5–16, <http://www.ep.liu.se/ecp/155/003/ecp18155003.pdf>.
- Tatiana BLADIER, Jakub WASZCZUK, and Laura KALLMEYER (2020b), Statistical parsing of Tree Wrapping Grammars, in *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6759–6766, International Committee on Computational Linguistics, Barcelona, Spain (Online), doi:10.18653/v1/2020.coling-main.595, <https://www.aclweb.org/anthology/2020.coling-main.595>.

- Daniel BÜRING and Katharina HARTMANN (1997), Doing the right thing, *The Linguistic Review*, 14(1):1–42.
- Berthold CRYSMANN (2013), On the locality of complement clause and relative clause extraposition, in Gert WEBELHUTH, Manfred SAILER, and Heike WALKER, editors, *Rightward movement in a comparative perspective*, *Linguistik Aktuell/Linguistics Today*, pp. 369–396, John Benjamins.
- Mark DE VRIES (2002), *The syntax of relativization*, Ph.D. thesis, Universiteit van Amsterdam.
- Tomaž ERJAVEC, Ana-Maria BARBU, Ivan DERZHANSKI, Ludmila DIMITROVA, Radovan GARABÍK, Nancy IDE, Heiki-Jaan KAALEP, Natalia KOTSYBA, Cvetana KRSTEV, Csaba ORAVECZ, Vladimír PETKEVIČ, Greg PRIEST-DORMAN, Behrang QASEMIZADEH, Adam RADZISZEWSKI, Kiril SIMOV, Dan TUFIŞ, and Katerina ZDRAVKOVA (2010), MULTEXT-East “1984” annotated corpus 4.0, Slovenian language resource repository CLARIN.SI.
- Janet D. FODOR (1978), Parsing strategies and constraints on transformations, *Linguistic Inquiry*, 9(3):427–473.
- Chung-hye HAN (2002), Compositional semantics for relative clauses in Lexicalized Tree Adjoining Grammars, in *Proceedings of the Sixth International Workshop on Tree Adjoining Grammar and Related Frameworks (TAG + 6)*, pp. 1–10, Association for Computational Linguistics, Universitá di Venezia, <https://www.aclweb.org/anthology/W02-2201>.
- Anke HOLLER (2013), D- und w-Relativsätze, in Jörg MEIBAUER, Markus STEINBACH, and Hans ALTMANN, editors, *Satztypen des Deutschen*, De Gruyter Lexikon, pp. 266–300, De Gruyter, Berlin.
- Aravind K. JOSHI, Leon S. LEVY, and Masako TAKAHASHI (1975), Tree Adjunct Grammars, *Journal of Computer and System Science*, 10:136–163.
- Aravind K. JOSHI and Yves SCHABES (1997), Tree-Adjoining Grammars, in Grzegorz ROZENBERG and Arto SALOMAA, editors, *Handbook of formal languages*, pp. 69–123, Springer, Berlin.
- Sylvain KAHANE (2000), How to solve some failures of LTAG, in *Proceedings of the Fifth International Workshop on Tree Adjoining Grammar and Related Frameworks (TAG + 5)*, pp. 123–128, Université Paris 7, <https://aclanthology.org/W00-2017>.
- Laura KALLMEYER (2003), LTAG semantics for relative clauses, in Harry BUNT, Ielka VAN DER SLUIS, and Roser MORANTE, editors, *Proceedings of the Fifth International Workshop on Computational Semantics IWCS-5*, pp. 195–210, Tilburg.
- Laura KALLMEYER (2016), On the mild context-sensitivity of *k*-Tree Wrapping Grammar, in Annie FORET, Glyn MORRILL, Reinhard MUSKENS, Rainer OSSWALD, and Sylvain POGODALLA, editors, *Formal Grammar. 20th and 21st*

International Conferences, FG 2015, Barcelona, Spain, August 2015, Revised Selected Papers. FG 2016, Bozen, Italy, August 2016, Proceedings, volume 9804 of *Lecture Notes in Computer Science*, pp. 77–93, Springer.

Laura KALLMEYER and Rainer OSSWALD (2017), Combining predicate-argument structure and operator projection: Clause structure in Role and Reference Grammar, in *Proceedings of the 13th International Workshop on Tree Adjoining Grammars and Related Formalisms*, pp. 61–70, Association for Computational Linguistics, Umeå, Sweden.

Laura KALLMEYER, Rainer OSSWALD, and Robert D. VAN VALIN, JR. (2013), Tree wrapping for Role and Reference Grammar, in Glyn MORRILL and Mark-Jan NEDERHOF, editors, *Formal Grammar 2012/2013*, volume 8036 of *LNCS*, pp. 175–190, Springer.

Makoto KANAZAWA (2016), Multidimensional trees and a Chomsky-Schützenberger-Weir representation theorem for simple context-free tree grammars, *Journal of Logic and Computation*, 26(5):1469–1516.

Tibor KISS (2005), Semantic constraints on relative clause extraposition, *Natural Language & Linguistic Theory*, 23(2):281–334, ISSN 1573-0859, doi:10.1007/s11049-003-1838-7, <https://doi.org/10.1007/s11049-003-1838-7>.

Anthony S. KROCH and Aravind K. JOSHI (1987), Analyzing extraposition in a Tree Adjoining Grammar, in Geoffrey J. HUCK and Almerido E. OJEDA, editors, *Syntax and semantics: Discontinuous constituency*, pp. 107–149, Academic Press, Inc.

Christian LEHMANN (1982), Directions for interlinear morphemic translations, *Folia Linguistica*, 16:199–224.

Mitchell MARCUS, Grace KIM, Mary Ann MARCINKIEWICZ, Robert MACINTYRE, Ann BIES, Mark FERGUSON, Karen KATZ, and Britta SCHASBERGER (1994), The Penn Treebank: Annotating predicate argument structure, in *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*, pp. 114–119, Association for Computational Linguistics, <https://www.aclweb.org/anthology/H94-1020>.

James D. McCAWLEY (1998), *The syntactic phenomena of English*, University of Chicago Press, 2d edition.

Stefan MÜLLER (2004), Complex NPs, subjacency, and extraposition, *Snippets*, 8:10–11, <https://hpsg.hu-berlin.de/~stefan/Pub/subjacency.html>.

George ORWELL (2000), *1984*, Ullstein, German translation by Kurt Wagenseil.

Rainer OSSWALD and Laura KALLMEYER (2018), Towards a formalization of Role and Reference Grammar, in Rolf KAILUWEIT, Eva STAUDINGER, and Lisann KÜNKEL, editors, *Applying and expanding Role and Reference Grammar*, pp. 355–378, Albert-Ludwigs-Universität, Universitätsbibliothek, Freiburg.

Owen RAMBOW (1994), *Formal and computational aspects of natural language syntax*, Ph.D. thesis, University of Pennsylvania.

John Robert ROSS (1967), *Constraints on variables in syntax*, Ph.D. thesis, Massachusetts Institute of Technology MIT.

Wolfgang STERNEFELD (2008), *Syntax. Eine morphologisch motivierte generative Beschreibung des Deutschen, volume 1*, Stauffenburg, Tübingen, 3rd revised edition.

Jan STRUNK and Neal SNIDER (2008), Extraposition without subadjacency, slides, DGfS 2008, Bamberg. <https://www.hlp.rochester.edu/~nsnider/pubs/ExtrapositionWithoutSubadjacencyDGfS2008.pdf>.

Jan STRUNK and Neal SNIDER (2013), Subclausal locality constraints on relative clause extraposition, in Gert WEBELHUTH, Manfred SAILER, and Heike WALKER, editors, *Rightward movement in a comparative perspective*, volume 200 of *Linguistics Today*, pp. 99–143, John Benjamins, Amsterdam.

Robert D. VAN VALIN, Jr. (2005), *Exploring the syntax-semantics interface*, Cambridge University Press.

Robert D. VAN VALIN, Jr. and Randy LAPOLLA (1997), *Syntax: Structure, meaning and function*, Cambridge University Press.

K. VIJAY-SHANKER and Aravind K. JOSHI (1988), Feature Structures Based Tree Adjoining Grammar, in *Proceedings of COLING*, pp. 714–719, Budapest.

Heike WALKER (2017), *The syntax and semantics of relative clause attachment*, Ph.D. thesis, Johann Wolfgang Goethe-Universität zu Frankfurt am Main.

XTAG RESEARCH GROUP (2001), A Lexicalized Tree Adjoining Grammar for English, Technical report, Institute for Research in Cognitive Science, Philadelphia.

Laura Kallmeyer

© 0000-0001-9691-5990


kallmeyer@phil.hhu.de

Institut für Sprache und Information,
Heinrich-Heine Universität Düsseldorf

Laura Kallmeyer (2021), *Extrapolated relative clauses in Role and Reference Grammar. An analysis using Tree Wrapping Grammars*, *Journal of Language Modelling*, 9(2):225–290

doi <https://dx.doi.org/10.15398/jlm.v9i2.255>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

cc  <http://creativecommons.org/licenses/by/4.0/>