

Branching constraints*

Erwin R. Komen

SIL-International, Radboud University

Rejecting approaches with a directionality parameter, mainstream minimalism has adopted the notion of strict (or unidirectional) branching. Within optimality theory however, constraints have recently been proposed that presuppose that the branching direction scheme is language specific. I show that a syntactic analysis of Chechen word order and relative clauses using strict branching and movement triggered by feature checking seems very unlikely, whereas a directionality approach works well. I argue in favor of a mixed directionality approach for Chechen, where the branching direction scheme depends on the phrase type. This observation leads to the introduction of context variants of existing markedness constraints, in order to describe the branching processes in terms of optimality theory. The paper discusses how and where the optimality theory selection of the branching directions can be implemented within a minimalist derivation.

Keywords: minimalist program, optimality theory, focus, branching, extraposition, Chechen

1 Introduction

One of the fundamental operations adopted in the Minimalist Program is the *merge* operation, which combines syntactic elements from the numeration as well as previously produced structures in order to form a hierarchical structure (Chomsky 1995). The merge operation has generally not been regarded as being directional in nature.¹ One widely accepted conversion from the two-dimensional hierarchy into a one-dimensional output fed into the phonology

* Many thanks to the participants of the DEAL-II workshop for their valuable comments.

¹ There has been a proposal to make the merge operation directional, but I have not seen other researchers taking on that lead (Saitu and Fukui 1998).

interface builds upon the linear correspondence axiom, which says that asymmetric c-command implies precedence (Kayne 1994).² This implies that all languages adhere to one specific branching scheme, i.e. that they have the same underlying order. Differences in word orders between languages and within any particular language must, by this concept, be due to a different hierarchical syntactic structure. Differences in structure are arrived at by movement of syntactical objects. The trigger for movement is feature checking.

The idea of directionality, which in Government and Binding theory was implemented with a directionality parameter, has within Optimality Theory been implemented using universal violable constraints that determine the relative positions of specifier, head and complement for a language (Grimshaw 1993, 1997, 2001, 2002, 2006, Zepter 2003). The hierarchical ranking of such constraints is what determines the general word order differences between languages. Within languages directionality can differ between functional and lexical projections, which, for optimality theory, is explained by introducing more constraints (Zepter 2003).³

² Persuasive arguments have been provided against the LCA (e.g. Abels and Neeleman 2007). However, the mapping to the output is not as simple as stated here. A major problem is the question how two lexical sister items dominated by one syntactic node should be linearized.

(i) a. *kygalxuochuo* *Muusa* *vyyr* *vu*
 leader-ERG Musa v-kill-FUT v-PRS
 ‘the leader will kill Musa’.

b. $[_{IP} \textit{kygalxuochuo}_i [_{I'} [_{VP} t_i [_{V'} \textit{Muusa vyyr}]] \textit{vu}]]$

For example sentence (ia) is analyzed as the structure (ib), whereby *Muusa* and *vyyr* are two lexical nodes c-commanding one another. One solution might be to postulate an empty D head for the object *Muusa*, but this seems unlikely, since DP’s have not been attested for Chechen. A few other solutions have been offered for the problem in general, but, as far as I know, none take the head-complement difference into account (Hornstein 2005:228-232).

³ Outside of Optimality Theory others have also suggested making differentiations between the branching direction within functional and lexical projections (Haider 1997, Broekhuis 2006).

In section 2 of this paper I show data from the Northeast Caucasian Chechen language, which, when attempting to describe the syntax of that data, make a strict branching approach less attractive than one where the branching direction of specifiers and heads can vary. I show that Chechen as a whole favors left branching specifiers and right branching heads, except for focus phrases – there left branching heads are favored. How a directionality approach could be implemented is the topic of section 3, where it is shown how an optimality theory selection scheme that determines the branching direction can be combined with a minimalist derivation. A ranking scheme is proposed for Chechen and verified against the data. The paper concludes in section 4 by summarizing the results and drawing some conclusions.

2 Chechen branching

In this chapter I show why strict branching is less capable of describing the Chechen data than a directionality approach, leaving the question how such an approach might be implemented for the next chapter. In section 2.1 I show that the auxiliary can be regarded as an overt realization of the head of the inflectional phrase. Under that assumption a strict branching approach gives the wrong results for the unmarked word order with compound verb tenses, as shown in section 2.2. In that same section I provide an alternative: assume left branching specifiers and right branching heads. The necessity to switch from strict branching to a directionality approach is confirmed by attempts to describe the syntax of relative clauses in section 2.3.

The directionality approach advocated shows that at least VP, vP, IP and CP need to have a right branching head instead of a left branching one. But further data shows that not all projections need to have the head branch in the same direction. So Chechen is analyzed as not having a uniform directionality

scheme, but a mixed one. Specifically in section 2.4 I show that Focus phrases can best be described as having a left branching head. This is confirmed by the syntax of extraposed relative and possessive clauses in section 2.5.

2.1 The auxiliary as an IP head

This subsection briefly explains why the auxiliary can be regarded as an overt realization of the head of the IP, the inflectional phrase (see also Komen 2007a). The Chechen auxiliary marks agreement and tense, but lacks an overt verb root component. This is reason enough to regard the auxiliary as overt realization of the inflectional phrase's head. Auxiliaries are built up by a class-marking prefix *v-*, *j-*, *b-* or *d-*, followed by a tense marker. This tense marker is *u* for the present and *-ra* for the past. Negated forms of the present auxiliary like *daac* lack a tense marker, but have a negating suffix *-ac*. Negated forms of the past auxiliary have both tense as well as negation.

The auxiliary occurs in sentences with a compound tense (present continuous, past continuous, etc.). Such tenses consist of a verb in a particular form followed by the auxiliary, as in (1). In compound tenses the verb can be expressed as a past participle (with adverbial meaning), a present participle (also with adverbial meaning) or a future form. Simple tenses don't use a form of the auxiliary, as illustrated in (2).

- (1) Muusa dika buolx biesh vu.
 Musa-OBL good work-ABS B-do-PTC V-PRS⁴
 'Musa is doing a good work.'

⁴ The following abbreviations are used: ABS=absolute, DAT=dative, ERG=ergative, GEN=genitive, IMPF=imperfective, NML=nominalizer, OBL=oblique (non-absolute case), PL=plural, PRS=present, PSTN=past using *-ina*, PSTR=past using *-ira*, PTC=present participle, REL=relativizer.

-
- (2) SielxanaMuusas buolx bira.
 yesterday Musa-ERG work-ABS B-do-PSTR
 ‘Musa worked yesterday.’

2.2 The unmarked word orders

Let us now take a closer look at the syntactic description of the unmarked word order in Chechen, assuming the following:

- The theoretical framework is minimalism (Chomsky 1995).
- The auxiliary is an overt realization of the IP head (see 2.1).
- Applying the linear correspondence axiom gives the correct spell-out order (Kayne 1994).
- Chain reduction applies: only the highest items in a chain are spelled out.

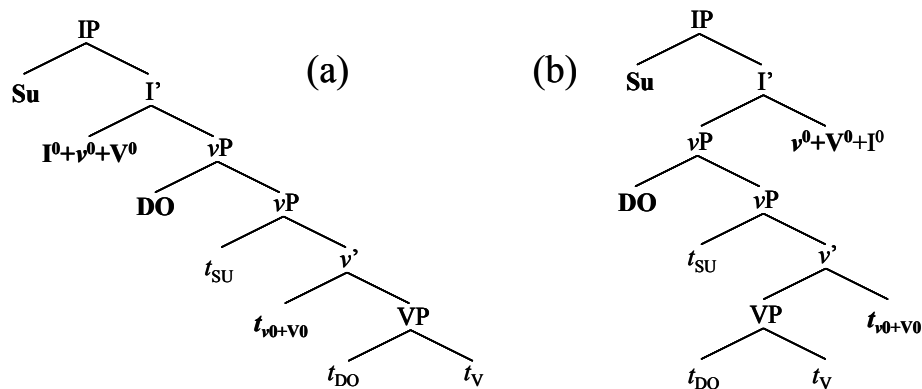
The unmarked word order for clauses with a transitive verb is SOV (Komen 2007a, 2007c). An example of such a clause is given in (3).

- (3) C’aruo ysh baaguosh bu.
 fire-ERG 3P-ABS B-burn-PTC B-PRS
 ‘The fire is burning them.’

The syntactic derivation of this word order, assuming strict branching, is illustrated in Figure 1a. The main verb and the direct object are taken from the numeration and the operation merge forms them into the lower VP. Then the light verb is taken from the numeration, merging with the lower VP. Next the subject is taken from the numeration, and it merges with the existing structure, becoming the first specifier of vP . Since the light verb can check object case, a copy is made of the direct object and it merges with the syntactic structure so far to become a second specifier of the vP and check case from there. Next the auxiliary, being the overt realization of the IP head, is taken from the numeration to merge with the structure. The IP head can check the subject case feature, so a copy of the subject is made and merged as the specifier of the IP. Next, either in overt or in covert syntax, the combined head v^0+V^0 adjoins to the

IP head to check tense (in Chechen tense is present both on the auxiliary as well as on the verb). At spell-out the linear correspondence axiom is applied, so that the order of words fed into the phonological component is strictly determined by the asymmetric c-command relationships.⁵

Figure 1 Unmarked SOV order



Now the problem with the unmarked word order becomes apparent: the word order arrived at with the strict branching approach is S-Aux-V-O, whereas the language data has shown that the unmarked word order is S-O-V-Aux.⁶ If one insists on strict branching (which implies that copying (*gebruik gewoon*: movement) can only occur leftward), the only way to derive the correct word order would be by the following two operations: (a) copy the remnant vP (consisting of O) to the specifier of an XP above the subject S, (b) head movement of $I^0+v^0+V^0$ to adjoin to the head of this XP, and (c) copy the remnant IP (consisting of subject S) to a clause initial position—the specifier of a projection above XP. But movement is only supposed to take place under the pressure of feature checking and there are no features to be checked by this movement.

⁵ In other terms: walk the tree from left to right.

⁶ Instead of S-Aux-V-O the word order alternatively is S-Aux-O-V, depending on whether the verb adjoins to the head of IP in overt or covert syntax.

The alternative to strict branching in the description of the Chechen data is to allow for heads to branch right while specifiers branch left, as illustrated in Figure 1b.⁷ Except for the branching directionality the derivation of the unmarked word order runs along the lines as given above.

If the selection of the branching direction takes place at the level of the basic minimalist operation merge, then it merge needs to “know” whether a head or a specifier is merged. But merge needs to be aware what it is merging anyway, since only heads project, only heads provide room for one or more specifiers and for complements (Hornstein et al. 2005:202).

The word order of the unmarked clause now correctly becomes S-O-V-Aux, if at spell-out the syntactical tree is walked from left to right.

2.3 Relative clauses

Let us take a look at the syntactic description of Chechen relative clauses, and see what this tells us about branching. Besides the minimalist assumptions introduced in section 2.2, I will adopt the adjunct analysis of relative clauses, which is a unification of the matching and the raising analysis (Henderson 2007). Furthermore I assume that the suffix *-l* is a complementizer (specifically a relativizer), and that forms like *dolu* ‘that is/are’ should be regarded as a combination of the auxiliary *du* (the overt realization of the inflectional head) and this complementizing affix; cf. Komen 2007b for more discussion.

I will show that, given these assumptions, the strict branching analysis runs into problems. This is illustrated in section 2.3.1. The only analysis which seems to reflect reality is one where heads branch to the right and specifiers to the left. This is illustrated in section 2.3.2.

⁷ If an element *x* “branches right” then it is positioned at the right branch of its parent.

2.3.1 Analysis with strict branching

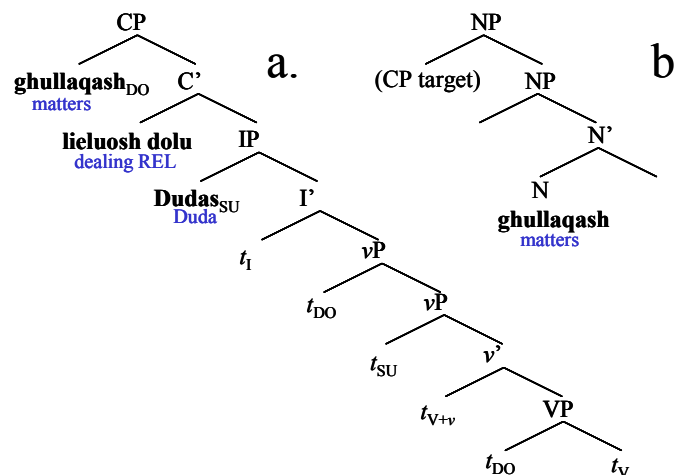
Adopting for the sake of the argument strict branching again, the analysis of the unmarked SOV clause runs along the lines given by Komen (2007a). The clause is built up as shown and described in section 2.2.

Let me illustrate the derivation of the relative clause using part of (4).

- (4) [Dudas t_i lieluosh **dolu**] ghullaqash_i
 DUDA-ERG deal-PRS-PTC D-REL matter-PL-ABS
 ‘the things Duda was dealing with.’ (Baduev 1991:25)

As shown in Figure 2, the relativizer projects a CP, and the direct object is copied to its specifier attracted by a relativizing feature. The relativizer has joined up with the head of IP (which is overtly realized as an auxiliary) to form a compound head *dolu*.⁸ According to the adjunct analysis of relative clauses a copy of the NP *ghullaqash* is made, which then is used as a separate syntactic object for building the matrix clause (Henderson 2007).⁹ The relative clause CP adjoins to this copy of the NP, as shown in Figure 2b).

Figure 2 Formation of relative clause using strict branching



⁸ The fact that the IP head has formed a phonological compound with the relativizer, the head of the CP, shows that head movement has taken place from I⁰ to C⁰.

⁹ According to Henderson adjunction takes place to the NP, but within the DP. For Chechen no DP has been established so far.

With this analysis the whole relative clause would, at the point of spell-out, be as shown in (5), which is not in line with the observed form in (4).

(5) * $[_{NP} [_{CP} t_i \text{ lieluosh } \mathbf{dolu} \text{ Dudas}] \text{ ghullaqash}_i]$

One might be tempted to argue that the compound IP head *lieluosh du* does not move to adjoin to the head of CP until after spell-out. But that conflicts with the observed phonological compound $I^0+C^0 \text{ dolu}$, and so would be in conflict with the phasing theory, which predicts the whole CP to be formed correctly before spell-out.

Another point that might be raised is that the participle *lieluosh* ‘dealing’ has moved upwards too far – it should not have left the ν P. This would be valid point if the participle is tenseless, but it is not: it carries a present tense morpheme.’ The participle can be marked for as many as three different tenses: *lieluosh* ‘dealing-PRS’, *lieliina* ‘dealing-PST’, and *lieluor* ‘dealing-FUT’. Since it contains a tense feature to be checked, it must move to the head of an IP and adjoin to it. That this argument holds is confirmed by the fact that no material may intervene (as far as I have been able to ascertain) between a participial form and the relativizing form of the auxiliary.

2.3.2 Analysis with directionality

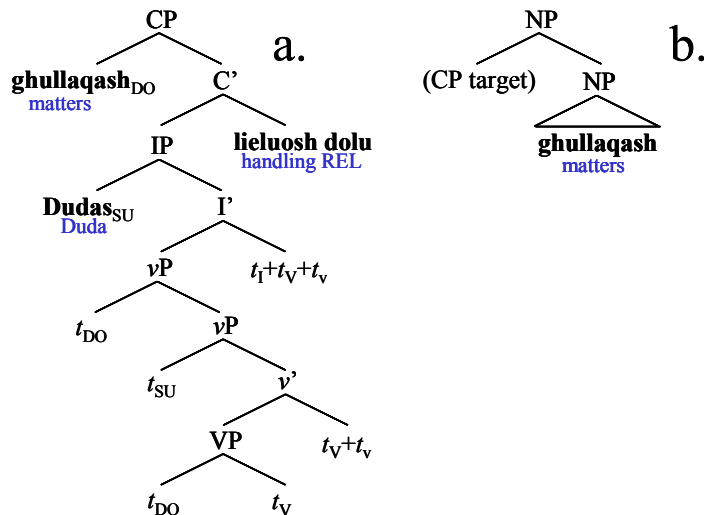
Consider an approach which is based on the following assumptions :

- Heads of the VP, ν P, IP and CP branch right, and specifiers branch left.
- Spell-out order is arrived by walking the tree from left to right.
- Chain reduction applies: only the highest items in a chain are spelled out.

The derivation of the relative clause given in (4) runs as follows (see Figure 3). A relativizing head C^0 is taken from the numeration and projects a CP. The head has a strong feature, attracting the direct object being relativized into its specifier. The compound head $V^0+\nu^0+I^0$ moves up and combines with the head

of CP. This yields the object shown in Figure 3a). Then a separate copy of the NP *ghullaqash* ‘matters’ is made for the matrix clause as in Figure 3b, and the relative clause CP adjoins to its left.

Figure 3 Formation of relative clause using the directionality approach



At spell-out chain reduction is applied, resulting in the correct surface order.

Assuming a left branching specifier and a right branching head, then, give realistic and straightforward results while retaining other “standard” minimalist assumptions (features, copying etc).

In the next two sections I show that the branching directionality may be different, depending on the kind of phrase being projected.

2.4 Focus and Topic phrases

As was noted in section 2.2, the unmarked word order of a Chechen clause is SOV. When the direct object is focused, word orders SO_FV and O_FVS are observed. When the subject is focused, word orders S_FVO and OS_FV are observed (Komen 2007a). The syntactic analysis of focus in Chechen assumes that there is a phonologically empty head Foc^0 of a focus phrase $FocP$, that there is a language specific requirement that this head be overtly filled, and that for Chechen the verb is the prime candidate for filling this head.

If Chechen heads *uniformly* are assumed to occupy the right branch of their parents, the heads of vP , VP and IP, possibly combining into a compound, will then always be found at the right edge of any IP. Focus phrases would build on top of such an IP, and the head of the focus phrase would appear at the rightmost edge of a clause containing a focused constituent. But this runs foul of the data observed. Whenever the object is focused, the combination of O_FV seems to form one entity. This entity, which I assume to be a constituent, could be labeled as the focus phrase.¹⁰ This focus phrase does not appear at the right edge of the clause, however, witness the occurrence of the O_FVS word order in Chechen.

The same is true for focused subjects, which result in a combination S_FV . But this combination, the focus phrase, cannot be broken up and does not necessarily occur clause finally, as for instance in the S_FVO word order.

Therefore I suggest that at least focus and possibly topic phrases have a left branching head in Chechen.¹¹ This provides a straightforward account for all the data.

2.5 Extraposed clauses

Assuming that CP, IP, vP and VP have right branching heads turns out to be crucial for an effective description of relative clauses in Chechen. Focus phrases and their branching are also relevant to the discussion of extraposed relative

¹⁰ The idea that there are separate functional projections called "focus phrase" and "topic phrase" is not accepted by everyone (Szendrői 2001, Neeleman and van der Koot 2007). But for Chechen no alternative descriptions have yet been posited.

¹¹ The reason for my hesitation about topic phrases stems from the fact that the head of the topic phrase is, as far as I have been able to determine, phonologically empty, and does not need to have an overt head to adjoin to it (whereas the focus phrase does). Therefore it is impossible to know for sure where the head of the topic phrase branches to – left or right. But for the moment I am assuming it branches left, like the focus phrase does.

clauses¹² as we will see in section 2.5.1, and extraposed possessive clauses, as we will see in section 2.5.2.

2.5.1 Extraposed relative clauses

Under the adjunct analysis, relative clauses are regarded as adjuncts, which is why I assume them to be base-generated in the position in which they occur. So when such a relative clause appears to be extraposed (and occurs in sentence-final position), then it must have been “left behind” while the noun phrase to which it belongs was copied leftward, for instance due to attraction by a focus or topic feature.

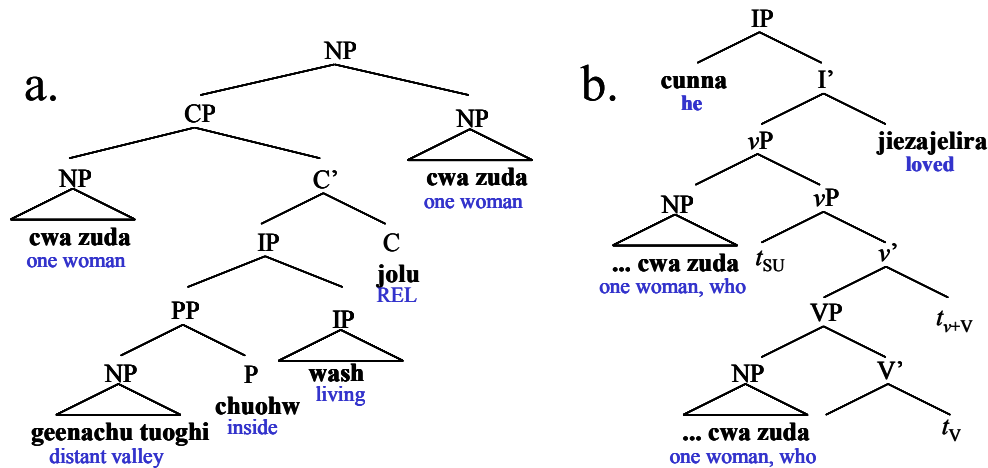
Let me illustrate this process using the relative clause given in (6).

- (6) Cunna cwa zuda jiezajelira, [geenachu tuoghi chuohw wash jolu]
 3S-DAT one woman-ABS J-love-PSTR distant-OBL valley-DAT inside live-PTC J-REL
 ‘He fell in love with a woman that lived in a distant valley.’

The syntax of the relative clause *geenachu tuoghi chuohw wash jolu* ‘that was living in a distant valley’ can be described as shown in Figure 4a). The CP is adjoined above the NP proper. The matrix clause would look as in Figure 4b). There are two copies of the object NP, and the relative clause CP may either adjoin to the copy in the specifier of VP, or to the copy in the upper specifier of vP —but *not* to both. This construction would not result in the relative clause being extraposed, however, since the last constituent of the IP continues to be the verb (i.e.: I⁰+v⁰+V⁰).

¹² Extraposed possessive clauses work the same way.

Figure 4 Extraposition of relative clause

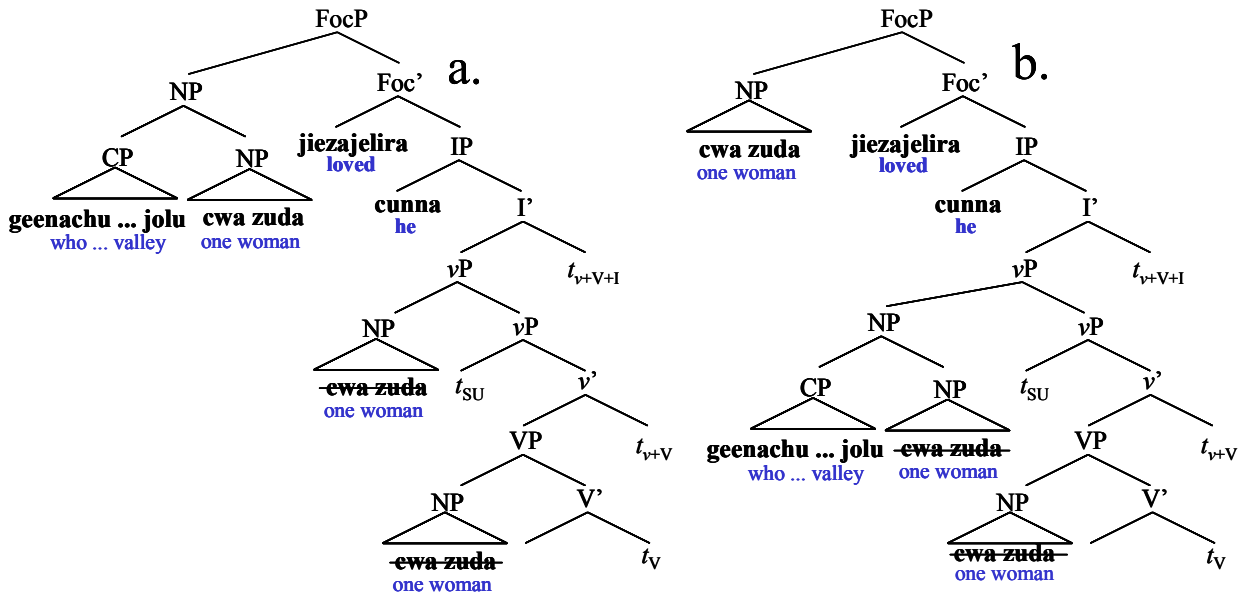


The idea of partial deletion giving two plausible configurations will only work if the object NP proper, i.e. the part without the relative clause, is moved out of the IP further upwards—for instance into a focus phrase. Let me illustrate this with the OVS variants of (6), which are shown here in (7) and (8).

- (7) Cwa zuda jiezajelira cunna, [geenachu tuoghi chuohw wash jolu]
 one woman-ABS J-love-RFPS 3S-DAT distant-OBL valley-DAT inside live-PTC J-REL
 ‘He fell in love with a woman that lived in a distant valley.’
- (8) [Geenachu tuoghi chuohw wash jolu]cwa zuda jiezajelira cunna.
 distant-OBL valley-DAT inside live-PTC J-REL one woman-ABS J-love-RFPS 3S-DAT
 ‘He fell in love with a woman that lived in a distant valley.’

Figure 5 shows that the relative clause part of a noun phrase can occur in two positions. If adjoined to the focus phrase (je bedoelt: the phrase in focus?), as in Figure 5a), there is no extraposition, resulting in the word order of example (8). But notice that the correct word order only follows if it is accepted that the head of the focus phrase branches *left* and not right. If adjoined to the copy of the noun phrase that resides in the vP, as in Figure 5b), the relative clause is extraposed, and the word order of example (7) follows. Again, the correct word order crucially depends on the left-branching head of the focus phrase.

Figure 5 Extraposition and focus



Let us look at data confirming the relationship between focus and extraposition discussed above. Native speakers were asked to evaluate sentences like (6)-(8) with and without extraposition, where the main part of the object consisted of a noun or of a question word.¹³ The results are shown in *Table 1*.¹⁴ This data confirms the analysis: question words (which inherently have a focus feature in Chechen) can *only* occur within the relative clause part of a noun phrase when they are *not* extraposed (as in line *g* of the table). As soon as a relative clause is extraposed, it can no longer contain a *wh*-word (as indicated by the * in the “Eval” column of lines *c* and *d*).

¹³ A sentence like (6) has S-O-V order, where O is in the specifier of the **FocP**, and S is in the specifier of a higher **TopP** (Komen 2007a).

¹⁴ When the object has a question word, it is labeled as O_q . The relative clause part of the object is identified as O_{RC} . Where the relative clause part contains a question word, it is labeled as $O_{RC,q}$.

Table 1 Acceptability of question words in relative clause

#		Order		Eval
a	O	V S	O _{RC}	ok
b	O _q	V S	O _{RC}	ok
c	O _q	V S	O _{RC,q}	*
d	O	V S	O _{RC,q}	*
e	O _{RC} +O	V S		ok
f	O _{RC} +O _q	V S		ok
g	O _{RC,q} +O _q	V S		ok
h	O _{RC,q} +O	V S		ok

2.5.2 Extraposed possessive clauses

The same features emerge when we compare questioned extraposed possessive phrases and “normal” possessive phrases. Take as a starting point the examples (9)-(12).

(9) [San] nastarsh ca lielara.

1s-GEN knee-PL NEG move-IMPF

‘My knees did not move.’

(10) [Hweenan]nastarsh ca lielara?

whose? knee-PL NEG move-IMPF

‘Whose knees did not move?’

(11) Nastarsh ca lielara [cigahw wash volchu stegan].

knee-PL NEG move-IMPF there live-PTC V-REL person-GEN

‘The knees of the person living over there did not move.’

(12) *Nastarsh ca lielara [michahw wash volchu stegan]?

knee-PL NEG move-IMPF where? live-PTC V-REL person-GEN

‘The knees of the person living where did not move?’

Native speakers were asked to evaluate different modifications of these sentences, where the position of the possessor varied and where part of the possessor was replaced by the question word *michahw* “where”. The results are shown in Table 2. The “normal” position for the Possessive Phrase (e.g. “my” or “whose?”) would be the specifier position, i.e. immediately preceding the noun that is possessed, as shown in lines (a) and (b) of the table. But the possessor can also be extraposed, as shown in (c) and (d), in which case only (c) is acceptable.

Sentence (d) is not acceptable, since the Possessive Phrase contains a question word (indicated by PossP_q), implying that it has a focus feature, and therefore needs to be part of a focus phrase.

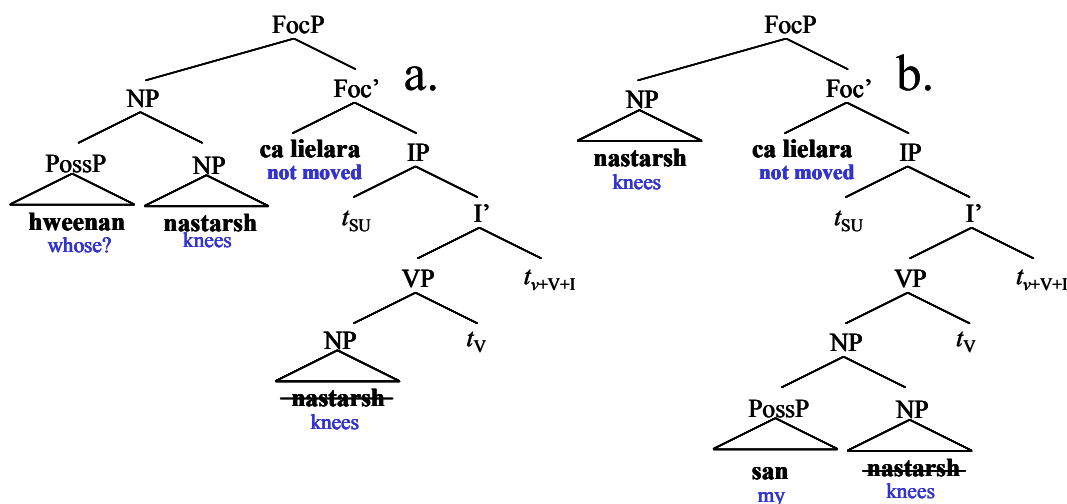
Table 2 Acceptability of question words in possessive clause

#	Order	Eval	Ref
a	[PossP N] Neg V	ok	(9)
b	[PossP _q N] Neg V	ok	(10)
c	N Neg V PossP	ok	(11)
d	N Neg V PossP _q	*	(12)

The analysis of possessive phrases runs along the following lines, as illustrated by Figure 6. The whole noun phrase *hweenan nastarsh* is in a focus phrase, where the focus feature of the question word *hweenan* ‘whose’ can be checked. Pronunciation of the possessee can take place either within the focus phrase position (as in part a) or at the lower copy within the verb phrase (not shown).

In part Figure 6b) the possessor *san* ‘my’ has no focus, but the possessee *nastarsh* ‘knees’ does. For that reason *nastarsh* must appear in the focus phrase, while the possessor is adjoined to the copy of the possessee in the verb phrase.

Figure 6 Possessives



In both situations sketched in Figure 6 it is essential that the focus phrase is to the left of the IP– in other words, that the head of the focus phrase branches left.

3 Combining optimality theory and minimalism

It remains to be shown how heads can branch in different directions within a minimalist framework.¹⁵ Within minimalism the linear correspondence axiom is widely used as a mapping procedure between the hierarchical structure derived by a syntactic derivation and the linear output required for the phonological component. With a directionality approach this linear correspondence axiom cannot be used anymore. Instead, an optimality theoretical branching direction selection mechanism will be used. The question is *where* such a selection mechanism should be included in the otherwise minimalist derivation.

Section 3.1 touches upon some fundamental questions concerning the combination of OT and minimalism. I then turn back to the Chechen data, but this time from the perspective of optimality theory. Section 3.2 shows how the branching constraints can be used to describe the general preference of the Chechen language to have left branching specifiers but right branching heads (Grimshaw 2001, 2006). Continuing the discussion in section 3.3, I introduce context sensitive variants of some existing branching constraints, which enable us to describe the different behavior of Chechen Focus phrases and the extraposition data. The introduction of these constraints give rise to speculations about what could be found in other languages, were the orderings of constraints different.

¹⁵ I realise that branching directions have been used *before* minimalism using a parametric approach. Good arguments have been provided in favour of accepting language specific directionality instead of a universal strict branching scheme (Ackema and Neeleman 2002, Abels and Neeleman 2007). But I have not found attempts to show how directionality should be implemented *within* the minimalist program, although some come close (Haider 1997, Broekhuis 2006).

3.1 Some fundamental questions

There are some fundamental questions that need to be answered in order to see how an optimality theory branching selection scheme could be combined with minimalism. The question at which point within the derivation a branching selection mechanism should apply is addressed in section 3.1.1, while I consider implications for the syntax-phonology interface in section 3.1.2.

3.1.1 *The level at which OT should be applied*

The optimality theory branching selection mechanism could be applied at (a) the level of the merge operation, (b) after the completion of an XP, (c) at the end of the complete derivation.

Option (a) does not allow alignment to the edge of an XP.¹⁶ Option (c), though possible, is surplus to requirements since the branching selection scheme does not really need to have the global information supplied at the end of a derivation—it only needs the information contained in a complete XP. Therefore let us consider how option (b) works, where the optimality theoretic branching direction scheme is applied right after the formation of every XP.¹⁷

The process can be illustrated by the noun phrase *ocu beq'achu qaachanax* ‘that dry food’ in example (13).

- (13) I shi' sutara tasavella [ocu beq'achu qaachanax].
 these two-ABS greedily entangle-V-PSTN that-OBL dry-OBL food-MAT
 ‘The two of them greedily got onto that dry food.’ (*Ajdamirov 2007*)

¹⁶ The edge of the XP needs to be taken into account with a constraint such as LEXHEAEDGE, saying that a lexical head should be at the edge of an XP (Zepter 2002).

¹⁷ The point at which the formation of an XP is finished does *not* necessarily coincide with the phases. The formation of an PP, for instance, finishes as soon as it merges with a projection realized by a different head (e.g. an element of an NP). But a PP is not traditionally seen as a phase.

Two merge steps are needed to complete the whole NP, leading to a hierarchical structure of $\{ocu, \{qaachanax, beq'achu\}\}$. This structure consists of a specifier, a head and a complement.¹⁸ At this point a branching selection mechanism is needed to choose the candidate that is most optimal for Chechen, as illustrated in Table 3.

Table 3 Branching selection at the level of XP

Input: Spec={ocu} _{DemP} , Comp={beq'achu} _{AdjP} , Head={qaachanax} _N	
a. \curvearrowright	{spec _{DemP} (ocu), {comp _{AdjP} (beq'achu), head _N (qaachanax)}}
b.	{spec _{DemP} (ocu), {head _N (qaachanax), comp _{AdjP} (beq'achu)}}
c.	{{ comp _{AdjP} (beq'achu), head _N (qaachanax)}, spec _{DemP} (ocu)}
d.	{{ head _N (qaachanax), comp _{AdjP} (beq'achu)}, spec _{DemP} (ocu)}

The question is which kind of constraints provide enough flexibility to allow for all the variation in different languages, yet restrictive enough to not overgenerate. Zepter has a mixed scheme. He uses a few branching constraints that only look at the local relationship (i.e. HEADLEFT, HEADRIGHT and BRANCHINGRIGHT), while others look at the structure of the whole XP (e.g. LEXHEADEDGE). For our purposes it is enough to use three gradient branching constraints, which count the number of syntactic objects to the left or right boundary of an XP (see Grimshaw 1993, 1997, 2001, 2006 and also section 3.2).

If we choose to apply a branching selection mechanism on completion of an XP, we need to know when such completion occurs. I argue that the completion of an XP is marked by either of the following two situations:

¹⁸ I am treating the Adjective *beq'achu* as a complement here for the sake of the argument. More evidence would be needed to substantiate this.

-
- (14) a. Whenever a head Y^0 (functional or lexical) is merged into an existing syntactic object of type XP. It is then that you know that XP is complete.
- b. Whenever a derivation finishes. It is then that you know that the last XP is complete.

The OT branching selection mechanism can be applied at each of these two points, and the branching direction of specifiers and complements in the top XP projection of the hierarchy created thus far can be adjusted according to the winning candidate.

Linearization (in the sense of transforming the 2-dimensional hierarchy into a 1-dimensional string) of the completed XP at this point is not yet possible, in my view, because the process of chain reduction still has to take place at the end of the derivation. That process still requires the hierarchical structure.

3.1.2 *The interface*

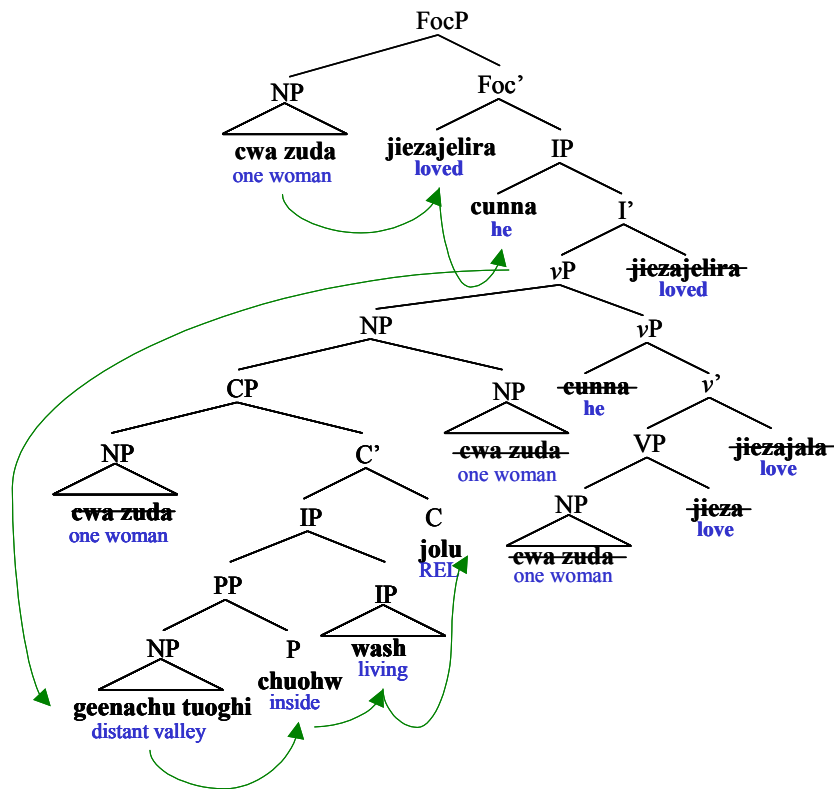
Under strict branching the interface between the syntactical structure and the phonology component provided by optimality theory was very simple: first chain reduction applies to “cross off” syntactic objects that are not supposed to be pronounced, followed by a one-to-one mapping from the structure to the phonological output, due to the strict branching. Let us now see how the hierarchical structures that result as output of the syntax component by employing a branching selection scheme can be fed into the phonological component in a similarly straightforward way.

Suppose a sentence with an extraposed relative clause as in (7), repeated here as (15), has been built up by the syntax component. At that point branching direction constraints have been applied, resulting in an ordered hierarchical structure. The tree representation of this sentence is as shown in Figure 7.

(15) Cwa zuda jiezajelira cunna, [geenachu tuoghi chuohw wash jolu]
 one woman-ABS J-love-RFPS 3S-DAT distant-OBL valley-DAT inside live-PTC J-REL
 ‘He fell in love with a woman that lived in a distant valley.’

In order to feed this to the phonological component, first chain reduction applies, so that several copies of *cwa zuda* ‘one woman’ and the subject *cunna* ‘she’ in the vP are crossed-off. Then the tree is walked from left to right, from top to bottom and up again (as indicated by the arrows). That results in the correctly ordered output for the phonology component.

Figure 7 Hierarchical structure of example (15)



3.2 Context free branching constraints

In section 3.1 several important fundamental questions were raised pertaining to when and how an optimality theory branching direction selection scheme could be combined with an otherwise minimalist derivation. As explained in 3.1.1, I will regard the branching direction selection scheme to apply after the finishing

of every XP, and I will mainly use the branching direction constraints introduced by Grimshaw (2006) in (17) and (18).

- (16) HDLEFT Align(Head, Left, XP, Left).
Align the left edge of the head with the left edge of the XP containing the head.
- (17) SPECLEFTAlign(Specifier, Left, XP, Left).
The specifier of a projection should be as close to the left edge of the projection as possible.
- (18) COMPLEFT Align(Complement, Left, XP, Left).
The complement of a projection should be as close to the left of the XP containing it as possible.

These constraints are gradient ones. The counting is in the number of syntactic objects (although counting in words would give the same results). For the SVO language English Grimshaw showed that the following branching scheme is valid:

- (19) SPECLEFT >> HDLEFT >> COMPLEFT


For the SOV language Chechen I argue that the following general branching scheme is valid:

- (20) SPECLEFT >> COMPLEFT >> HDLEFT

That COMPLEFT >> HDLEFT can be seen from a postpositional phrase such as the one shown in (21). The complement *geenachu tuoghi* should be left of the head *chuohw*. For this reason the constraint COMPLEFT outranks HDLEFT, as shown in the tableau in Table 4.

- (21) [PP [NP geenachu tuoghi] chuohw]
 distant-OBL valley-DAT inside
 ‘In a distant valley’

Table 4 Ranking of a postpositional phrase


Input: Hd(chuohw) Comp(geenachu tuoghi)	SPECLEFT	COMPLEFT	HDLEFT
a. [PP chuohw [NP geenachu tuoghi]]		*!	
b.  [PP [NP geenachu tuoghi] chuohw]			*

That SPECLEFT >> HDLEFT can be seen from a noun phrase such as the one shown in (22). The specifier *vajn q'ooman* ‘of our nation’ should be left of the nominal head *stag* ‘person’. The same clause also illustrates that SPECLEFT >> COMPLEFT, since the specifier should be left of the complement *dika* ‘good’.

- (22) [NP [_{POSSP} *vajn q'ooman*] *dika stag*]
IP-INC nation-GEN good person-ABS
 ‘A good person of our nation’

The constraint ranking for this noun phrase is exemplified in Table 5. The winning candidate is (c), which coincides with (22), so confirms that the correct ranking scheme has been made.

Table 5 Ranking of a noun phrase

Input: Hd(<i>stag</i>), Spec(<i>i</i>), Comp(<i>dika</i>)	SPECLEFT	COMPLEFT	HDLEFT
a. [NP [<i>vajn q'ooman</i>] [_{N'} <i>stag dika</i>]]		*!*	*
b. [NP [_{N'} <i>dika stag</i>] [<i>vajn q'ooman</i>]]	*!*		*
c.  [NP [<i>vajn q'ooman</i>] [_{N'} <i>dika stag</i>]]		*	**
d. [NP [_{N'} <i>stag dika</i>] [<i>vajn q'ooman</i>]]	*!*	*	

In the next section I will show that the three constraints need to be extended by one or two more for the focus phrase (and the topic phrase) to end up into the right position.

3.3 Context sensitive branching constraints

Since the focus phrase, and possibly the topic phrase, have a left branching head in Chechen, I introduce constraints capturing this in section 3.3.1, and I discuss typological consequences of these constraints in section 3.3.2.

3.3.1 Introducing context sensitive branching constraints

Let us consider context sensitive variants of the generic branching constraints that were introduced in section 3.2, like for example HDLEFT(FocP) as defined in (23).

(23) HDLEFT(FocP)Align(Head, Left, FocP, Left).

The head of a focus phrase should be as close to the left edge of the focus phrase as possible.

No independent motivation for this constraint is needed, since it is a more specific variant of the already existing generic HDLEFT.

With context sensitive constraints it is possible to capture the non-uniform branching scheme from a language like Chechen. While the normal HDLEFT constraint is completely at the bottom in the general branching scheme for Chechen, this HDLEFT(FocP) and the HDLEFT(TopP) constraints need to be somewhat higher. I suggest a ranking scheme for Chechen as sketched in (24).

(24) SPECLEFT >> HDLEFT(FocP), HDLEFT(TopP) >> COMPLEFT >> HDLEFT.


The ranking of HDLEFT(FocP) can be illustrated by looking at example (25). In this example the subject *nastarsh* ‘knees’ of the intransitive verb *liela* ‘move’ is focused. The possessor of the subject, *san* ‘my’, does not have a focus feature, and therefore has adjoined to the copy of the subject that is inside the vP. This situation is graphically shown in Figure 6b).

(25) *Nastarsh ca lielara san*
knee-PL NEG move-IMPF 1S-GEN
 ‘My knees did not move.’

What needs to be ascertained here is whether the ranking scheme I proposed in (24) delivers the correct result in this situation. The input to the selection scheme is the unordered, but labeled set of three syntactic objects: Head = *ca lielara*, Complement = *san* and Specifier = *nastarsh*.¹⁹ The four output candidates (a)-(d) in Table 6 are the only possible candidates faithful to the input labeling (provided that FocP is the highest projection, and no CP is above it).

¹⁹ The head is a compound consisting of the phonologically empty Foc⁰ and the I⁰ + v⁰ + V⁰ that have adjoined to it. The possessive *san* is complement of this Foc⁰, and the noun *nastarsh* is specifier of the FocP.

Table 6 Ranking a clause with an extraposed possessor

Input:	Hd(ca lielara), Cp(san), Sp(nastarsh)	SPECLEFT	HDLEFT(FocP)	COMPLEFT	HDLEFT
a.	[_{FocP} [_{Foc'} ca lielara [_{IP} san]] nastarsh]	*!*		*	
b.	[_{FocP} [_{Foc'} [_{IP} san] ca lielara] nastarsh]	*!*	*		*
c. 	[_{FocP} nastarsh [_{Foc'} ca lielara [_{IP} san]]]		*	**	*
d.	[_{FocP} nastarsh [_{Foc'} [_{IP} san] ca lielara]]]		*!*	*	**

The winning candidate coincides with the actually observed Chechen clause in (25). This confirms that the correct ranking has been chosen.

The non-uniform branching-scheme of a language like Tzotzil can also be derived with a context-sensitive variant of the generic branching constraints. In Tzotzil the word order normally is head-complement-specifier (VOS), so that in general the ranking scheme in (26) is used.

(26) HDLEFT >> SPECLEFT, COMPLEFT

When the subject is focused, the order SVO results. This difference in branching can be captured by introducing a context-sensitive variant of the SPECLEFT constraint, leading to a ranking scheme such as (27).

(27) SPECLEFT(FocP) >> HDLEFT >> SPECLEFT, COMPLEFT

3.3.2 Typology

The three general branching constraints SPECLEFT, COMPLEFT and HDLEFT can be put into ranking schemes together with the newly introduced constraint HDLEFT(FocP), resulting in 24 possible ranking schemes. Although most of these identify languages with uniform branching, there are 5 ranking schemes that identify languages with mixed directionality, as shown in Table 7. Each line in this table represents one ranking scheme. The word order within the winning

candidate for the focus phrase is given in the column labeled “FocP Order”. As a representative for other phrases the winning candidate for the ν P is given in the columns labeled “ ν P Order”.²⁰

Row (e) shows the ranking scheme of Chechen, which is normally spec-comp-head, but for the Focus Phrase it is spec-head-comp. Note that in row (b) the winning candidate is *not* the order OSV, since this is not possible within the hierarchical structure of the ν P.

Table 7 Ranking schemes using the HDLEFT(FocP) constraint

	Ranking	FocP Order	ν P Order
a.	HDLEFT(FocP) >> COMPLEFT >> HDLEFT >> SPECLEFT	[_{FocP} [_{Foc'} FocHd Comp] FocSp]	[[OV]S]
b.	HDLEFT(FocP) >> COMPLEFT >> SPECLEFT >> HDLEFT	[_{FocP} [_{Foc'} FocHd Comp] FocSp]	[[OV]S]
c.	HDLEFT(FocP) >> SPECLEFT >> COMPLEFT >> HDLEFT	[_{FocP} [_{Foc'} FocHd Comp] FocSp]	[S[OV]]
d.	HDLEFT(FocP) >> SPECLEFT >> HDLEFT >> COMPLEFT	[_{FocP} [_{Foc'} FocHd Comp] FocSp]	[S[VO]]
e.	SPECLEFT >> HDLEFT(FocP) >> COMPLEFT >> HDLEFT	[_{FocP} FocSp [_{Foc'} FocHd Comp]]	[S[OV]]

In like fashion the general branching constraints can be extended with the context sensitive SPECLEFT(FocP), which was argued to be operative in Tzotzil. From the 24 resulting ranking schemes most, again, identify languages with uniform branching. Those identifying languages with mixed directionality are shown in Table 8. Rows (c) and (d) in this table illustrate Tzotzil, where the normal head-comp-spec order becomes spec-head-comp.

²⁰ Note that the column ν P Order does *not* necessarily give the unmarked order observed in the language. There may still be movement to the inflectional phrase IP. For instance VSO word order in the unmarked clause can be the result of ν P order of SVO, followed by raising of the verb to the head of IP.

Table 8 Ranking schemes using the SPECLEFT(FocP) constraint

	Ranking	FocP Order	vP Order
a.	SPECLEFT(FocP) >> COMPLEFT >> HDLEFT >> SPECLEFT	[_{FocP} FocSp [_{Foc'} Comp FocHd]]	[[OV]S]
b.	SPECLEFT(FocP) >> COMPLEFT >> SPECLEFT >> HDLEFT	[_{FocP} FocSp [_{Foc'} Comp FocHd]]	[[OV]S]
c.	SPECLEFT(FocP) >> HDLEFT >> COMPLEFT >> SPECLEFT	[_{FocP} FocSp [_{Foc'} FocHd Comp]]	[[VO]S]
d.	SPECLEFT(FocP) >> HDLEFT >> SPECLEFT >> COMPLEFT	[_{FocP} FocSp [_{Foc'} FocHd Comp]]	[[VO]S]

4 Conclusions and discussion

Looking closely at otherwise minimalist syntactic descriptions of the unmarked word order and the relative clauses in Chechen the conclusion is that right branching heads for NP, vP, VP, IP and CP are more likely than left branching ones. Left branching heads are not only required to account for the syntactic behaviour of focus phrases, but also for the behaviour of extraposed relative clause and possessive phrases. The branching direction of topic phrase heads is unclear, since topic phrases do not have overt heads in Chechen.

Generalizing the results for Chechen, I suggest that the branching direction of specifiers and complements should be treated as being language specific, and that minimalism should allow for right branching heads. One way to implement choosing branching directions in the minimalist framework is to adopt an optimality-theoretic selection mechanism using branching constraints. I have shown how the three branching constraints introduced by Grimshaw are able to provide a satisfactory account for most, but not all, of the branching in Chechen; the different behaviour of focus phrases shows that Chechen has a non-uniform branching scheme.

I have shown that context sensitive variants HDLEFT(FocP) and HDLEFT(TopP) of the generic HDLEFT constraint can be used to account for the Chechen focus data. These context sensitive constraints are a natural

continuation of the context free constraints already introduced by Grimshaw. Adopting projection specific branching constraints implies that other context sensitive constraints might be operative in other languages. I have illustrated this with the context sensitive constraint SPECLEFT(FocP), which is operative in a language like Tzotzil.

The point at which the branching direction selection mechanism applies has not been determined in this paper – it could apply at such a early point as at the level of the merge operation or it could apply just before the point of spell-out. More research will be needed to determine the pros and cons of any particular mechanism.

The implication of accepting language specific and phrase specific branching schemes is that strict branching – more specifically the application of the linear correspondence axiom – should be abandoned as a principle within minimalism. Mainstream theories within the minimalist program have made use of strict branching to provide a straightforward and simple feeding of the phonological component. Since the linear correspondence axiom is no longer available, we need another mechanism, possibly, as I have suggested, one which lines up the correct word order to be fed into the phonological component when the usual chain reduction applies first, and then the tree is walked from left to right.

5 References

- Abels Klaus, Ad Neeleman. 2007. *Linear asymmetries and the LCA*. Ms. University of Tromsø / UCL. Reference: draft 2.3, lingBuzz/000279.
- Broekhuis Hans. 2006. *The universal base hypothesis: VO or OV?* To appear in Bettelou Los and Jeroen van de Weijer (Eds), *Linguistics in the Netherlands 2006*. Amsterdam, Philadelphia: John Benjamins.
- Chomsky Noam. 1995. *The minimalist program*. Cambridge, MA: MIT Press.

-
- Comrie Bernard. 1989. *Language universals and linguistic typology*. Second edition. Blackwell.
- Grimshaw Jane. 1993. *Minimal projection, heads, and optimality*. Ms., Rutgers University.
- Grimshaw Jane. 1997. *Projection, heads, and optimality*. *Linguistic Inquiry* 28:373-422.
- Grimshaw Jane. 2001. *Economy of structure in OT*. Rutgers Optimality Archive 434-0601.
- Grimshaw Jane. 2002. *Economy of structure in OT*. In: *Papers in Optimality Theory II*. University of Massachusetts Occasional Papers 26. 81-120.
- Grimshaw Jane. 2006. *Last resorts and grammaticality*. In: Hans Broekhuis and Ralf Vogels (eds). *LiP 25. Optimality Theory and Minimalism: A Possible Convergence?* University of Potsdam. 33-41
- Haider Hubert. 1997. *Typological considerations of a directionality constraint on projections*. In: Artemis Alexiadou and T. Alan Hall (eds.). *Studies on universal grammar and typological variation*. Amsterdam, Philadelphia: John Benjamins. 17-33.
- Hornstein Norbert, Nunes J., Grohmann K.K. 2005. *Understanding Minimalism*, Cambridge University Press.
- Kayne Richard. 1994. *The antisymmetry of syntax*. Cambridge, MA: MIT Press.
- Komen Erwin R. 2007a. *Focus in Chechen*. MA-thesis. Leiden University.
- Komen Erwin R. 2007b. *Chechen relative clauses*. Manuscript.
- Komen Erwin R. 2007c. *Clause structure in Chechen*. Term-paper for 2007 Hungarian syntax class. Leiden University.
- Neeleman Ad, Peter Ackema. 2002. *Effects of short-term storage in processing rightward movement*. In S. Nooteboom et al. (eds.) *Storage and Computation in the Language Faculty*. Dordrecht. Kluwer. 219-256.
- Neeleman Ad, Hans van de Koot. 2007. *The nature of discourse templates*. Ms. To appear in JGCL.
- Saito Mamoru, Fukui Naoki. 1998. *Order in phrase structures and movement*. *Linguistic Inquiry* 29(3):439-474.
- Szendrői Kriszta. 2001. *Focus and the syntax-phonology interface*. PhD Thesis. University College London.
- Zepter Alex. 2003. *Phrase Structure Directionality: Having a few Choices*. PhD Thesis. Rutgers University. (See also: ROA 611-0803).

Erwin R. Komen
Radboud University
Box 9103
6500 HD Nijmegen
E.Komen@let.ru.nl