Tier-Based Strict Locality and the Typology of Agreement

Kenneth Hanson
Stony Brook University
mail@kennethhanson.net

May 11, 2025
To appear in *Journal of Language Modelling*

ABSTRACT

This paper presents a subregular analysis of syntactic agreement patterns modeled using *command strings* over Minimalist Grammar (MG) dependency trees (Graf and Shafiei 2019), incorporating a novel MG treatment of agreement. Phenomena of interest include relativized minimality and its exceptions, direction of feature transmission, and configurations involving chains of agreeing elements. Such patterns are shown to fall within the class of *tier-based strictly 2-local* (TSL-2) languages, which has previously been argued to subsume the majority of long-distance syntactic phenomena, as well as those in phonology and morphology (Graf 2022a). This characterization places a tight upper bound on the range of configurations that are predicted to occur while providing parameters for variation which closely match the observed typology.

Keywords: syntax, agreement, locality, Minimalist Grammars, tier-based strictly local languages

INTRODUCTION 1

Linguistic patterns display tremendous variation, yet are also subject to strong structural constraints. For example, syntactic dependencies are generally understood to follow the *c-command* relation (Reinhart 1981) and also to obey *relativized minimality* (Rizzi 1990). But this is not the full picture. In recent years, it has become increasingly apparent that whatever mechanism underlies agreement must be parameterized so as to make fine-grained distinctions regarding which elements agree, for what features, and under what configurations (Bobaljik 2008; Deal 2015; Keine 2019, a.o.). From a computational perspective, what is striking about the range of attested patterns is that, under the appropriate representation, they largely fall within the class of *tier-based strictly local* (TSL) languages, mirroring previous results on long-distance phonotactics (McMullin 2016), movement (Graf 2022b) and case (Vu *et al.* 2019; Hanson 2023). The primary aim of this paper is to demonstrate this.

The basic intuition behind a TSL pattern is that when the irrelevant elements are ignored, the pattern can be described using local constraints on those that remain visible. To illustrate, consider the standard Minimalist treatment of subject-verb agreement. Finite T bears unvalued ϕ-features which serve as a probe for agreement, and the subject DP bears valued ϕ-features which serve as a goal. The probe must c-command the goal; additionally, the probe and goal may occur at some distance from each other as long as no other ϕ-bearing element intervenes (i.e. it must obey relativized minimality). If we take the chain of elements along the clausal spine below T and ignore everything except for these elements, we obtain a string which is called a *tier*. The relevant local constraint on the tier is that T must agree with the immediately following D. This is schematized below:

(1) $\mathrm{T}_{[\phi]} \ldots \mathrm{D}_{[\phi]} \ldots \mathrm{D}_{[\phi]}$ $\qquad$ $\mathrm{T}_{[\phi]} \ldots \mathrm{D}_{[\phi]} \ldots \mathrm{D}_{[\phi]}$

Crucially, TSL is a restrictive class of formal languages. Aside from the fact that it can relate elements at a distance, the space of patterns that it can express is severely limited. This helps to explain why we see the patterns that we do and no others. In fact, the range of linguistic patterns which have been described as being TSL are overwhelmingly TSL-2, meaning that all constraints can be stated within a window of two elements on the tier. Here, I show that the parameters provided by the formalism – the set of elements which appear on the tier and the local constraints on the tier – closely correspond to several key dimensions of the formal typology of agreement, echoing previous results on movement (Graf 2022b) and long-distance harmony (Mc-

Mullin and Hansson 2016). The parallel between syntactic agreement and phonological harmony is particularly striking; this observation is not entirely new (cf. Nevins 2010), but the present perspective brings it into unusually sharp focus. Overall, these results lend further support to the idea that linguistic phenomena across domains are united in the kinds of computations they are built upon (Graf 2022a).

In order to show that agreement patterns are TSL-2, I adopt a formalization based on *Minimalist Grammars* (MGs, Stabler 1997, 2011) and *command-strings* (c-strings, Graf and Shafiei 2019), which are paths through an MG dependency tree whose ordering corresponds approximately to asymmetric c-command. In doing this, a novel approach to agreement is proposed, which utilizes "probe" and "goal" features analogous to standard MG licensor and licensee features. In addition to highlighting the parallel between syntax and phonology, this model also allows us to cleanly separate phenomena which are explained well by computational restrictions from those which derive from other sources such as the tree geometry; the latter include the c-command restriction as well as certain island constraints.

Similarly, some typological generalizations most likely derive from extragrammatical factors, such as constraints on language acquisition. While the formalism does not say anything about which sets of elements may form a tier, it is natural to suppose that the learner only considers elements which are obviously related. There is evidence from phonotactic learning that this is indeed the case, as participants in an artificial language experiment succeeded at learning long-distance dependencies involving only consonants or vowels, but failed to learn those involving both (Newport and Aslin 2004). Exceptions to this rule could occur when agreeing DPs differ from non-agreeing DPs in a highly salient manner, such as bearing a specific case or undergoing movement, as discussed in detail here.

The remainder of this paper is structured as follows. In Section 2, I provide an overview of the classification of linguistic patterns according to their computational complexity, and introduce the formal class TSL along with its subclass SL (strictly local). Section 3 develops a formal model which allows agreement patterns to be analyzed using TSL-2 constraints over c-strings, and presents an analysis of subject-verb agreement. In Section 4, I show that a wide variety of agreement patterns across languages are just slight variations of the basic TSL-2 pattern. Section 5 concludes, with a discussion of some open questions.

2 COMPUTATIONAL COMPLEXITY

In this section, we review some concepts from formal language theory which are crucial to the TSL analysis of linguistic patterns. I start with the motivation for modeling syntactic dependencies with subregular constraints over trees (Section 2.1). From there, I provide definitions and examples of SL (Section 2.2) and TSL (and Section 2.3), the two classes most relevant to local and long-distance linguistic syntactic patterns, followed by a brief discussion of multi-TSL grammars, which represent the intersection of multiple (T)SL constraints (Section 2.4).

2.1 *String languages and tree languages*

Formal languages are sets of objects, traditionally strings, which can be used to model linguistic patterns. We can categorize the complexity of formal languages in terms of the kinds of patterns they are able to represent – more complex classes of languages can encode a wider range of patterns. Generally speaking, more complex classes also require more powerful machinery both to learn and process. *String languages* (or stringsets) are commonly used to model the computational complexity of phonological and morphological patterns, but for syntax *tree languages* (or treesets) are more insightful. While the reasons for this are intuitive to syntacticians, we can also motivate this representational choice from a purely formal perspective.

The Chomsky Hierarchy (Chomsky 1959), shown in Figure 1, outlines the major classes of string languages.[1] Phonological and morphological patterns almost exclusively lie within the class of *regular*

[1] Many details are omitted from this figure for simplicity. For example, syntax does not seem to require the full power of the context-sensitive languages, but rather some *mildly context-sensitive* subclass (Joshi 1985). Furthermore, only the Swiss German data in Shieber (1985) is not context-free on the surface, unlike the earlier Dutch data in Huybregts (1976, 1984). However, Bresnan *et al.* (1982) argue that when considering the structures assigned to sentences by the grammar, Dutch is also not context-free. Thus, the importance of structures emphasized in this section has a long history.

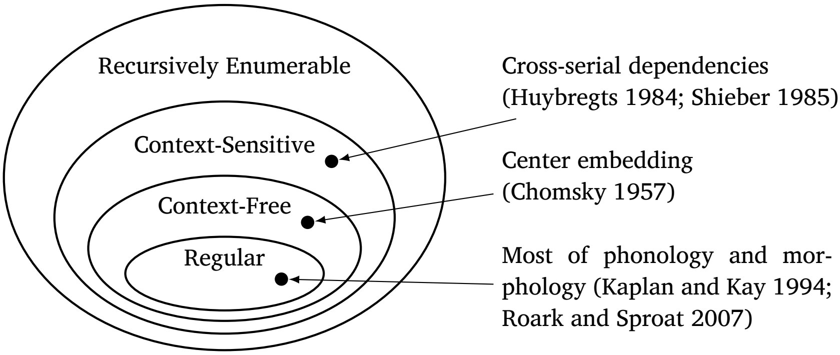


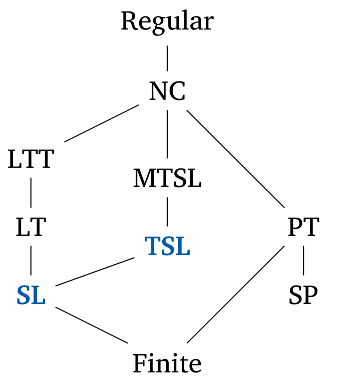
Figure 1: The original Chomsky Hierarchy (Chomsky 1959), showing the categorization of important linguistic patterns when modeled as surface strings

languages, while many syntactic patterns (analyzed as surface strings) are *context-free*, and some are *context-sensitive*. While useful in many respects, this characterization also obscures the formal similarities between phonology and morphology on the one hand, and syntax on the other. The classification of syntax is particularly problematic, for there are many types of *regular* patterns which are not attested in any module of grammar, including syntax.

In recent years a more fine-grained view has emerged. We can decompose the regular languages of the Chomsky Hierarchy into many smaller classes of *subregular languages*. A relevant subset of the resulting Subregular Hierarchy is shown in Figure 2. Many of these classes have been known for some time (Schützenberger 1965; McNaughton and Papert 1971; Simon 1975), but their significance for language cognition has only been recognized more recently (Rogers *et al.* 2013; Heinz 2018; Graf 2022a). It is now conjectured that all linguistic patterns lie within the subregular region: phonological and morphological patterns are subregular string languages, while syntactic patterns fall in the tree-based equivalents of these classes. This has been termed the *cognitive parallelism hypothesis* by Graf (2022a).

Following recent work, I pursue the specific hypothesis that local dependencies fall within the *strictly local* (SL) languages, while long-distance dependencies are *tier-based strictly local* (TSL), which is a proper superclass of SL. Within syntax, the former includes selection (Graf 2018) and functional hierarchies (Hanson 2024a), while the latter includes movement (Graf 2018, 2022b) and case (Vu *et al.* 2019; Hanson 2023). Adding to this, I argue that agreement is also TSL.

Figure 2: The (simplified) Subregular Hierarchy, adapted from Heinz (2018). SL and TSL, which subsume most phonological and morphological phenomena, are highlighted. By hypothesis, syntactic dependencies fall under the the tree-based equivalents of SL and TSL



If this is correct, then we have an explanation for why linguistically preposterous constraints along the lines of “a sentence may not contain both a verb and an adjective unless it also contains at least one quantifier” and “a word must not include both a consonant cluster and vowel hiatus” do not exist: such constraints are LT (locally testable), but not TSL, and therefore not of the variety handled by the computational machinery underlying language. Similarly non-existent patterns include “a sentence must contain between two and four adverbs”, which is LTT (locally threshold testable), “a word must obey consonant harmony or vowel harmony, but need not obey both”, which is NC (non-counting, a.k.a. star-free), and “the number of prepositions in a sentence must be a multiple of three”, which is properly regular. In addition, TSL languages can be learned in polynomial time and data (Jardine and McMullin 2017; Lambert 2021), and in stark contrast to the classes just mentioned, with low memory requirements (Lambert *et al.* 2021). As a consequence, the typological facts that we attribute to computational complexity might ultimately be grounded in considerations of efficient learnability (ibid.).

Note that while some linguistic patterns do in fact go beyond TSL, only a few have been noticed in the literature so far, and crucially, none occur in the data examined here.[2] Furthermore, as discussed in

[2] Known examples include Samala sibilant harmony, Uyghur backness harmony, and Sanskrit n-retroflexion. All of these can be modeled with a set of extensions to TSL known collectively as *structure-sensitive* TSL (cf. De Santo and Graf 2019; Mayer and Major 2018; Graf and Mayer 2018).

detail in Section 4, the typology of agreement patterns fits very closely to what we predict based on the expressive capabilities of TSL with a size two window. Because of this, I describe only SL and TSL in detail.

Strictly local languages 2.2

Each class of subregular languages has several equivalent characterizations. Here, I present definitions using *forbidden factors* – which for our purposes may be either substrings (as in SL) or substrings on a tier (as in TSL) – adapted from Mayer (2021). For illustration, I draw on examples from phonotactics.

In what follows, S^n and S^* denote all strings over set S of length n and of any finite length, respectively. Also, s^k denotes the string consisting of k repetitions of s. Σ denotes a finite set of symbols called the *alphabet*, and a *string language* is a subset of Σ^*.

Intuitively, a SL grammar is just a finite set of forbidden substrings of some fixed length, and the corresponding SL language is the set of all strings that do not contain any forbidden substrings.[3] We formalize this intuition as follows. $\rtimes$ and $\ltimes$ are the *edge markers*, which are added to a string so that the beginning, middle, and end can be modeled uniformly. Next, the *k-factors* of a string w, denoted $f_k(w)$, are the set of length-k substrings of $\rtimes^{k-1}w\ltimes^{k-1}$. For example, the 2-factors of the string *abcabc* are $\{\rtimes a, ab, bc, ca, c\ltimes\}$; the string *abcabcabc* also contains the same 2-factors. A grammar containing the 2-factor *ab* would rule out both of these strings, along with many others.

Definition 1 A *strictly k-local* (SL-k) grammar is a finite set $G \subseteq (\Sigma \cup \{\rtimes, \ltimes\})^k$. A language $L \subseteq \Sigma^*$ is SL-k iff there exists a SL-k grammar G such that

$$L = \{w \in \Sigma^* : f_k(w) \cap G = \emptyset\}$$

A language is SL iff it is SL-k for some $k \geq 1$.

SL grammars correspond to categorical k-gram models, and express what a linguist would identify as local constraints. As a simple example, consider a natural language which exhibits CV syllable

[3] Equivalently, we can define a SL grammar as a set of *permissible* substrings. Converting between these two formulations is trivial, assuming that all substrings in the grammar have the same length and that wellformedness is categorical.

structure with an optional word-final consonant. We can model this pattern as a SL language consisting of strings of the symbols C and V, as summarized in (2). Licit words in this language include CV, CVC, and CVCV, but not VC, CVV, or CVCCV. The forbidden substrings for this language are {⋊V, VV, CC}, making it SL-2. The licit word ⋊CVC⋉ contains the substrings {⋊C, VC, VC, C⋉}, none of which are forbidden. On the other hand, the illicit word ⋊VC⋉ contains {⋊V, VC, C⋉}, of which ⋊V is forbidden.

(2) Example SL-2 pattern: CV syllable structure, optional final C

- $\Sigma = \{\mathrm{C}, \mathrm{V}\}$
- $k = 2$
- $G = \{\rtimes\mathrm{V}, \mathrm{VV}, \mathrm{CC}\}$
- Licit words: ⋊CV⋉, ⋊CVC⋉, ⋊CVCV⋉, ⋊CVCVC⋉, …
- Illicit words: ⋊V⋉, ⋊VC⋉, ⋊CVV⋉, ⋊CVCCV⋉, …

Note that SL grammars cannot relate two symbols that do not occur within the same k-factor, nor can they count occurrences of k-factors; if two strings contain the same set of k-factors, they are indistinguishable. These restrictions distinguish SL (and TSL) from more powerful classes in the subregular hierarchy. Note also that a grammar which enforces this abstract constraint using separate symbols for each distinct consonant and vowel, while considerably *larger*, is no more *complex* in the relevant sense.

2.3 *Tier-based strictly local languages*

A TSL language is much like a SL language except that the forbidden factors are substrings on a tier, allowing a limited type of long-distance dependency to be expressed (Heinz *et al.* 2011; Lambert and Rogers 2020). Any symbol not appearing on the tier is ignored completely and the remainder are treated as adjacent; a SL language is the special case of TSL where every symbol appears on the tier. Note that while this notion of a tier was inspired by autosegmental phonology (Goldsmith 1976), it is conceptually distinct, as the tier elements are just a special subset of the elements of the full structure. Lambert (2023) uses the term *relativized adjacency* to describe the type of relativized locality encapsulated by a tier in this sense.

Formally, in addition to the alphabet Σ, there is also a *tier alphabet* T, and every string w is associated with a *tier projection*, denoted $\textsc{proj}_T(w)$, in which all symbols not in T are removed. A string is in the language iff its tier projection contains no forbidden k-factors.

Definition 2 A *tier-based strictly k-local* (TSL-k) grammar is a tuple (T, G) where $T \subseteq \Sigma$ and $G \subseteq (T \cup \{\rtimes, \ltimes\})^k$. A language L is TSL-k iff there exists a TSL-k grammar such that

$$L = \{w \in \Sigma^* : f_k(\textsc{proj}_T(w)) \cap G = \emptyset\}$$

A language is TSL if it is TSL-k for some $k \geq 1$.

For our next example, consider a language with (symmetric) sibilant harmony, a TSL-2 pattern, as shown in (3). Licit words in such a language include 'saksa' and 'ʃakʃa', but not 'sakʃa' or 'ʃaksa'. In this case, the tier alphabet contains only the sibilant consonants {s, ʃ}. Mismatched sibilants are forbidden on the tier, ruling out any strings that do not obey harmony. For example, the illicit word sakʃa has the tier projection ⋊sʃ⋉, which contains the forbidden substring sʃ.

(3) Example TSL-2 pattern: Sibilant harmony
- Σ = {a,k,s,ʃ}
- k = 2
- T = {s,ʃ}
- G = {sʃ,ʃs}
- Licit words: asa, aʃa, saksa, ʃakʃa, …
- Illicit words: saʃa, ʃasa, sakʃa, ʃaksa, …

While the pattern just described is also in the class SP (strictly piecewise, Rogers *et al.* 2010), and some harmony patterns are SL with a suitably large k-value, only TSL-2 subsumes both types as well as long-distance harmony with blocking (cf. McMullin and Hansson 2016). The latter type is particularly pervasive in syntax, making TSL-2 the prime candidate for the maximally restrictive classification of long-distance syntactic patterns.

2.4 *Multi-tier grammars*

The reader may have noticed that in (3) there is nothing preventing the generation of absurd words such as *skskaaakkk*. To obtain a full model, we must intersect the subregular languages representing isolated patterns like those above to produce one that obeys all of them.

The intersection of several TSL languages is known as a multi-TSL, or MTSL language (De Santo and Graf 2019). In general, a realistic description of any natural language is necessarily (at least) MTSL due to the existence of both local and long-distance dependencies. Furthermore, it is empirically well-established that long-distance dependencies such as EPP movement, *wh*-movement, ϕ-agreement are subject to different locality constraints (cf. Keine 2019). Thus, when we say that long-distance dependencies are in general TSL, this should be interpreted to mean that each individual dependency is TSL.

This raises the question of what exactly constitutes an independent linguistic dependency. For example, the analysis of case in Japanese by Hanson (2023) includes three tiers, and is therefore technically MTSL. The same situation is likely to arise in agreement patterns in which a single predicate agrees with multiple noun phrases simultaneously, such as those analyzed by Béjar and Rezac (2009) and Nevins (2011). It seems plausible that such patterns can likewise be decomposed into a set of intersecting constraints, each of which is TSL. However, it could be the case that interactions other than intersection are needed, in which case the full pattern is not MTSL. Due to the complexity of the data, a proper investigation of this issue is beyond the scope of the present article, which focuses exclusively on "individual" dependencies.

3 A TSL MODEL OF AGREEMENT

In this section, we extend TSL languages to trees in order to model agreement. Following recent work (Graf and Shafiei 2019; Graf 2022b; Hanson 2023), I use Minimalist Grammar dependency trees (Sections 3.1 and 3.2) for the tree language. To date, there are two ways in which TSL languages have been generalized to trees. Here, I develop a model based on command strings (Section 3.3), and show how the model can be applied to agreement (Section 3.4).

Minimalist Grammars 3.1

Minimalist Grammars (MGs, Stabler 1997, 2011) are a formalization of ideas from Chomsky's (1995) Minimalist Program. Standard MGs contain just two operations: Merge and Move (we will add agreement later). The grammar of a language is just a lexicon of syntactic heads annotated with features to guide these operations.

The features for each operation come in two polarities, which I notate +F and −F. For Merge, these are *selector* and *category* features, whose meanings are intuitive. For example, transitive *v* has selector features +V and +D, since it selects a VP complement and a DP specifier, and a category feature −*v*. Since these features play no direct role in this paper and can be inferred from context, I omit them in all derivations. Additionally, I will continue to refer to categories as V/D/C, etc., even though they are technically −V/−D/−C. For Move, we have *licensor* features, which mark the landing site of movement, and *licensee* features, which mark the mover. For example, finite T carries +EPP, and the DP which moves to its specifier carries −EPP.[4]

It is important to note that MG features are just diacritics which describe what happens in the derivation. In a language with single *wh*-movement, for example, only the highest *wh*-element bears −Wh since it is the one that must move. Placing −Wh on every *wh*-element is tantamount to saying that all of them move; indeed, this is what Graf and Kostyszyn (2021) do in their model of multiple *wh*-movement.

MGs can be used to generate syntactic structures in several ways. The standard approach is to generate a language of *derivation trees*, which show the order of Merge and Move steps. The derivation tree is then mapped to a phrase structure tree by executing all movements and inserting X′-style labels. It is the derivation tree language which is our focus, as this is where syntactic dependencies are formed. The constraints on the mapping to the derived tree are also a topic of cur-

[4] In Stabler's original notation selector/category/licensor/licensee features are notated =f/f/+f/−f. In Graf (2018) and related work they are $F^+/F^-/f^+/f^-$. I selected the present notation in part because the addition of agreement features in Section 3.4 produces a six-way distinction. For our purposes it is unlikely that Merge/Move/Agree features will be misinterpreted, so a binary split is sufficient.

rent research; see Graf (2023) for a subregular model which handles placement of moved elements in the correct position.

3.2 *Dependency trees*

The specific kind of derivation tree we will use is a *dependency tree*. This representation is especially compact while providing all necessary information about the derivation, namely what elements merged with what, and what their features are.[5]

An example phrase structure tree and the corresponding dependency tree for the sentence "The cat chases the rats" are shown in Figure 3. Every node in the dependency tree is a feature-annotated lexical item, but as mentioned above I omit all selector and category features for brevity. The daughters of a node are the heads of its arguments, ordered from right to left in order of first merge. Thus, the rightmost daughter is the complement and any others are specifiers. For example, the left daughter of *v* is the determiner heading its specifier, while the right daughter is the verb *chases*, which heads the complement. Additionally, all nodes in the dependency tree appear in base position only. In the present example, this applies to the subject, which undergoes EPP movement to Spec-TP.

Our goal is to show that the set of licit feature configurations conforms to a TSL grammar over dependency trees. There are several ways in which this can be done. Graf (2018) defines a direct analog of string TSL: the tiers are trees, and the constraints restrict the string of daughters of each node on the tier. Graf and Shafiei (2019) propose an alternative in which we extract paths through the derivation tree along which syntactic dependencies occur and enforce constraints on the resulting string language. I adopt a modified version of the latter approach, as described below.

As a final note, although elements appear only in their base position in the dependency tree, it is often nonetheless possible to handle interactions with movement just by inspecting the features of the

[5] These MG dependency trees first appear in Graf and Shafiei (2019) and are formally defined in Graf and Kostyszyn (2021). However, the use of dependency structures in MG has extensive precedent. The earliest use seems to be Kobele (2002), and the system in Kobele (2012) is essentially identical to that used here. Also see Boston *et al.* (2010), who use MGs to derive surface dependency trees.

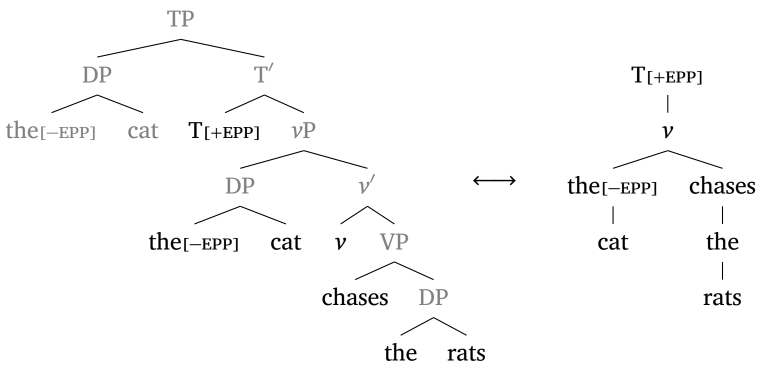


Figure 3: Phrase structure tree and dependency tree for “The cat chases the rats”. Nodes appearing only in the phrase structure tree are grayed out

moving elements. For example, differential object marking in many languages can be analyzed as being fed by movement out of VP, as in the analysis of Sakha by Baker and Vinokurova (2010). In this case, presence or absence of a particular licensee feature on the D head of the object is enough to determine if it should be marked. This method will not work when it is crucial to know the exact landing site, but it will work whenever we just need to know whether or not a phrase has moved at all and perhaps also the type of movement, as is true of the patterns examined in this paper.

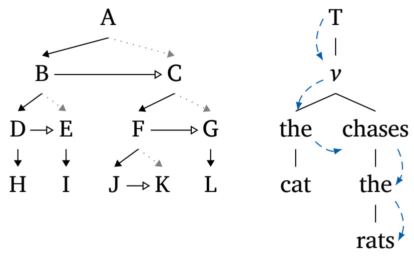
Command strings and spines 3.3

The specific model I utilize in this paper, building upon ideas in Graf and De Santo (2019), splits each tree into a set of strings, which represent the *complement spine* of the tree and each complex left branch. But first, let us overview the basics of *command strings* (c-strings) as introduced by Graf and Shafiei (2019).

The c-string of a node is the path to that node from the root which includes its left siblings, its ancestors, and the left siblings of its ancestors, with both sibling and ancestor order preserved. Put another way, every path from the root made by tracing the first-daughter and left-sister relations is a c-string. This is schematized on the left side of Figure 4. As a concrete example, the c-string for *rats* in our running example sentence is $\mathrm{T}_{[+\mathrm{EPP}]} \cdot v \cdot \mathrm{the}_{[-\mathrm{EPP}]} \cdot \mathrm{chases} \cdot \mathrm{the} \cdot \mathrm{rats}$, as illustrated on the right side of the same figure.

A formal definition is given below. This definition is identical to that given by Graf and Shafiei except that the ordering is from root to

Figure 4:
Left: a c-string follows the mother-of relation to the first daughter (filled arrow) and the left-sister relation (open arrow). The mother-of relation to non-first daughters (dotted lines) is not used. Right: c-string for *rats* in the sentence 'The cat chases the rats' (blue dashed lines)



target node rather than the reverse. This allows c-strings to be read more easily, but is otherwise inconsequential since TSL string languages are closed under reversal, as are many other subregular classes.

Definition 3 Let T be a tree such that node m has the daughters $d_1, \ldots, d_i, \ldots, d_n$ with $n \geq 0$. The *immediate c-string* $ics(d_i)$ of d_i is the string $d_1 \cdots d_i$. For every node n of T, its c-string $cs(n)$ is recursively defined as follows, where $\cdot$ indicates string concatenation:

$$cs(n) := \begin{cases} n & \text{if } n \text{ is the root of } T \\ cs(m) \cdot ics(n) & \text{if } m \text{ is } n\text{'s mother} \end{cases}$$

The ordering relation encoded by a c-string, which Graf and Shafiei call *d(erivational)-command*, reflects the hierarchical order of maximal projections in the phrase structure tree, or alternatively, the order in which category features are checked. It can be thought of as a hybrid of asymmetric c-command, since the complement is commanded by both its head and the specifier, and m-command (Aoun and Sportiche 1982), since the head commands the specifier rather than the other way around. For example, T d-commands its complement *v*, which in turn d-commands V; in addition, *v* d-commands the D which heads its specifier. Also notice that, since d-command is defined in terms of the dependency tree, it avoids unnecessary complications related to X′ projections and their labels. For example, it rarely matters whether the head of XP commands its specifier or the other way around, but the maximal projection should certainly be considered superior to the specifier. D-command provides the latter, as desired. In the end, this allows us to enforce both traditional c-command-based constraints, such as reflexive licensing, as well as containment-based constraints, such as islands, in a simple and unified manner.

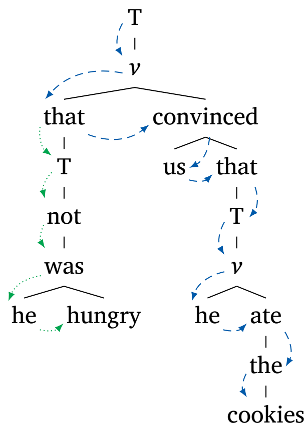


Figure 5:
Tree and two spines for the sentence "[That he was not hungry] convinced us [that he ate the cookies]". The subject clause begins a new spine (green dotted), while the complement clause shares a spine with the main clause (blue dashed)

Now, Graf and Shafiei are concerned primarily with licensing of individual nodes; for them, wellformedness of a dependency tree requires (among other things) that all c-strings are well-formed.[6] In this paper, I utilize just the c-string which traces the *complement spine* (henceforth *spine*) of the tree, plus the partial c-strings which trace the spine of some complex left branch. Figure 5 gives an example with two spines: *[That he was not hungry] convinced us [that he ate the cookies]*. The complement clause is part of the main spine (T · *v* · that · convinced · us · that · T · *v* · he · ate · the · cookies), while the subject clause constitutes its own spine (that · T · not · was · he · hungry). Notice that each node appears in at most two spines: those which head a complex left branch appear in both the spine of that branch and in the containing spine. In the present example, this applies to *that* in *that he was not hungry*.

From now on, I will refer to the (partial) c-strings tracing spines as *spinal* c-strings, and all of our TSL grammars will apply to these strings. This will allow us to model pairwise dependencies such as agreement in manner which is highly intuitive and which closely parallels the treatment of phonological harmony discussed previously. However, there is an extra benefit to making this shift. As discussed by Graf and De Santo (2019), syntactic dependencies tend not to occur between

[6] As will be discussed momentarily, not all syntactic constraints can be modeled with c-strings. For example, as a reviewer remarks, they cannot enforce the SMC of standard MGs; see Graf and Kostyszyn (2021) for a TSL treatment.

a head and an element deeply embedded in some adjunct or specifier. For example, movement out of adjuncts and specifiers is often degraded (these are the well-known *adjunct island* and *specifier island* constraints), while movement out of complements is unremarkable. Similarly, finite T usually agrees with a DP in the same spine, perhaps embedded in a complement clause, but not one which is embedded in a subject clause. As a consequence, by applying our grammars to spinal c-strings, we effectively enforce these constraints as well.

There are, of course, numerous exceptions to this generalization, such as reflexive licensing (in which information must be passed down all paths) and parasitic gaps (which cannot be handled with c-strings at all). Additionally, since c-strings conflate ancestors and c-commanders, there are situations where constraints which should only affect one type of blocker are incorrectly applied to both; only containing phrases should induce island effects, for instance. Enriching the c-string representation to indicate whether the path has just entered a specifier/complement/adjunct, as in Graf and De Santo (2019), would allow some of these complications to be handled directly. However, tackling such issues here would take us too far afield, so I leave the development of a more complete model for future research.

3.4 *Constraining agreement*

Having established how to obtain a spinal c-string from a dependency tree, we are almost ready to show how TSL constraints on these strings can be used to model syntactic agreement. But first, recall that standard MGs have no agreement operation; we now add one.

As mentioned in the introduction, I assume agreement to involve two types of features: *unvalued* features, which receive their value during the derivation and *valued* features, which enter the derivation with their value; I will refer to a node with unvalued features as a *probe*, and the node which provides its value as a *goal*. Accordingly, I define a notation for agreement probes and goals parallel to the movement features of standard MGs. For each agreement feature F, +F denotes a probe and −F denotes a goal. In the case of subject-verb agreement, T

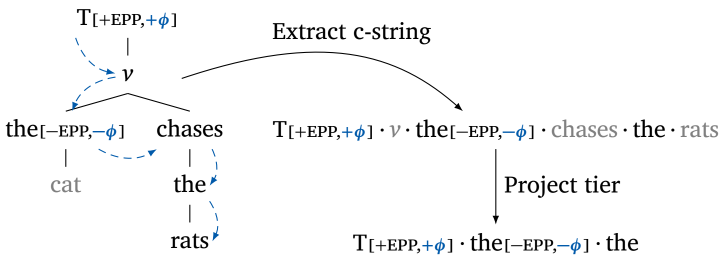


Figure 6: TSL analysis of English subject-verb agreement. Elements ignored at each step are grayed out

bears $+\phi$ and the D head that it agrees with bears $-\phi$.[7] To be clear, this does *not* mean that other heads do not bear ϕ-features in the theoretical sense, only that they do not serve as the goal of agreement in the current derivation. This is analogous to the MG treatment of movement: a *potential* EPP mover (or *wh*-mover, etc.) does not necessarily bear −EPP (or −Wh), only *actual* movers do.[8]

Now, we restrict the set of English dependency trees to just those with well-formed agreement feature configurations. We continue with our running example, illustrated in Figure 6. Our goal is to ensure that the $+\phi$ feature on T is paired with a $-\phi$ feature on the closest visible DP in its c-command domain, which is normally the verbal subject. To do this, we extract the (main) spinal c-string and project a tier which includes all elements which are potential bearers of $\pm\phi$, that is, all D heads and finite T heads. In addition, we project C since agreement cannot cross a finite or non-finite CP boundary, nor can it skip a CP subject and agree with a DP object (this will be discussed in detail momentarily). On the tier, we require that every element bearing $+\phi$ be immediately followed by one bearing $-\phi$, and that every element bearing $-\phi$ be immediately preceded by one bearing $+\phi$. In the current structure, this constraint is satisfied: our tier consists of T followed by two D heads, and the only probe and goal are adjacent.

[7] Following common practice, I abbreviate the bundle of ϕ-features as a single feature when they act together, as is true of the examples in this paper.

[8] An exploration of alternative feature systems, such as the four-way split in Pesetsky and Torrego (2007), is beyond the scope of this paper. Upon initial consideration, it seems unlikely that there will be any major formal differences. In fact, as a reviewer notes, just a single undifferentiated ϕ diacritic would be sufficient for the patterns studied here.

Next, we consider some ways in which this constraint could be violated, which correspond to the banned substrings in the TSL grammar. For comparison, the licit example from Figure 6 is repeated in (4a). (For simplicity, I swap out most lexical items for their categories when writing out c-strings and their tiers, and also omit movement features when not relevant.) First, it is not possible for T to agree with any DP other than the closest. For example, it cannot bypass the subject in favor of agreement with the object, as in (4b). In such a structure, the subject D head intervenes on the tier, violating both clauses of our constraint. Formally, we say that the tier substrings $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}$ and $\mathrm{D} \cdot \mathrm{D}_{[-\phi]}$ are illicit. Similarly, agreement with the object in (4c) is impossible because the subject C head intervenes, even though by hypothesis it cannot agree. Finally, agreement into CP is impossible (4d), though agreement into non-finite TP is possible; see (6) below.[9]

(4) Licit and illicit subject-verb agreement configurations

a. Subject agreement (✓)
   - Sentence: The cat **chases** the rats.
   - C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{V} \cdot \mathrm{D} \cdot \mathrm{N}$
   - ϕ-agreement tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{D}$
   - Constraints violated: n/a

b. Object agreement across DP (✗)
   - Sentence: The cat **chase** the rats.
   - C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D} \cdot \mathrm{V} \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{N}$
   - ϕ-agreement tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D} \cdot \mathrm{D}_{[-\phi]}$
   - Constraints violated: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}$, $\mathrm{D} \cdot \mathrm{D}_{[-\phi]}$

c. Object agreement across CP (✗)
   - Sentence: [$_{\mathrm{CP}}$ That he plays the bassoon] **impress** us.
   - C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{C} \cdot \mathrm{V} \cdot \mathrm{D}_{[-\phi]}$
   - ϕ-agreement tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{C} \cdot \mathrm{D}_{[-\phi]} \cdot$
   - Constraints violated: $\mathrm{T}_{[+\phi]} \cdot \mathrm{C}$, $\mathrm{C} \cdot \mathrm{D}_{[-\phi]}$

d. Agreement into non-finite CP (✗)
   - Sentence: It **are** possible [$_{\mathrm{CP}}$ for rats to have fleas.]
   - C-string: $\mathrm{T}_{[+\phi]} \cdot \mathrm{be} \cdot \mathrm{A} \cdot \mathrm{C} \cdot \mathrm{to} \cdot v \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{V} \cdot \mathrm{D} \cdot \mathrm{N}$

[9] I use a non-finite embedded clause in this example to avoid a confound with finite clauses, which is that finite embedded T intervenes even if C is invisible.

- ϕ-agreement tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{C} \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{D}$
- Constraints violated: $\mathrm{T}_{[+\phi]} \cdot \mathrm{C}$, $\mathrm{C} \cdot \mathrm{D}_{[-\phi]}$

Some readers may be wondering about the treatment of C as an arbitrary blocker. Admittedly, English is not an ideal example of this since there are alternative analyses available in most cases. For example, we could posit that CPs do agree, but that this agreement is always singular. At the same time, it is descriptively true that ϕ-agreement is blocked whenever a CP intervenes. This aligns with the behavior of EPP movement, but contrasts sharply with *wh*-movement, in which declarative C does not interfere. As discussed by Keine (2019), opaqueness of a given type of phrase must be relativized to individual dependencies, even in theories which include successively cyclic movement and/or phases. The blockers for specific dependencies such as ϕ-agreement also vary across languages. This issue will be explored further in Section 4.1.

Two remaining ways our constraint could be violated include tiers which contain two probes or two goals in sequence, that is, those that contain substrings such as $\mathrm{T}_{[+\phi]} \cdot \mathrm{T}_{[+\phi]}$ or $\mathrm{D}_{[-\phi]} \cdot \mathrm{D}_{[-\phi]}$.[10] Putting all of this together, we arrive at the (informal) TSL-2 grammar shown in (5) below. Here, I introduce several additional notational shortcuts. We are already using T/D/C as a stand-in for any item of the relevant category; in addition, X will be used as a placeholder for an element of any category. Next, when a category is followed by a list of features in square brackets, this denotes an element bearing *exactly* those features, while X with no brackets denotes an element with no relevant features (in this case, $\pm\phi$). It is understood that the tier alphabet should be compiled out to all of the matching symbols in the MG lexicon, and likewise for the banned substrings on the tier. Note that the alphabet Σ contains all elements in the MG lexicon; this never varies for a given language, so I omit it from all grammar definitions.

(5) ϕ-agreement tier for English

- Project: all finite T, all D, all C

[10] It is difficult to think of a context in English where we could find two probes not separated by a CP boundary, but since any DP can bear $-\phi$, a sequence of two goals could potentially occur in any transitive clause.

- Constraints: $\left\{\begin{array}{cc} X_{[+\phi]} \cdot X_{[+\phi]}, & X_{[-\phi]} \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot X, & X \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot \ltimes, & \rtimes \cdot X_{[-\phi]} \end{array}\right\}$

The overall analysis is extremely similar to the analysis of movement in Graf (2022b). The primary difference is that because we are using c-strings, we are able to handle relativized minimality in full generality; with tree tiers, only blocking by containing elements can be handled correctly. Another notable characteristic of the model is that domain-based and intervention-based blocking are treated uniformly, as exemplified by our treatment of intervention by C.

These, of course, are the simple cases; even in English, there are situations when the correlation between subject-hood and agreement comes apart. This happens in existential sentences like (6). For whatever reason, existential *there* seems to be invisible for agreement. Assuming it to be absent from the ϕ-agreement tier, long-distance agreement with the embedded subject follows, as the latter is adjacent on the tier just like a canonical subject.[11] In addition, long-distance agreement across *there* is optional for many, if not most English speakers, an issue which we will revisit in Section 4.4.

(6) Long-distance agreement in existential sentences

- Sentence: There **seem** [$_{\text{TP}}$ to be [$_{\text{PredP}}$ some squirrels in the attic]].
- C-string: $\text{T}_{[+\text{EPP},+\phi]} \cdot v \cdot \text{seem} \cdot \text{to} \cdot \text{be} \cdot \text{there}_{[-\text{EPP}]} \cdot \text{Pred} \cdot \text{D}_{[-\phi]} \cdot \text{P} \cdot \text{D} \cdot \text{N}$
- ϕ-agreement tier: $\text{T}_{[+\phi]} \cdot \text{D}_{[-\phi]} \cdot \text{D}$
- Constraints violated: n/a

At this point, I should mention an alternative model, which is to posit that all lexical items enter the derivation with concrete feature values (1SG/1PL/etc.), and the TSL grammar checks that they match in configurations where agreement applies. In this case, the tier constraints for subject-verb agreement would require finite T to bear the

[11] I assume that *there* originates in Spec-*v*P of specific verbs including *be*. See Deal (2009) for arguments in favor of this analysis. I also assume that the complement of *be* is a PredP, though nothing crucial hinges on this.

same set of valued ϕ-features as the following D node. This is analogous to the treatment of phonological harmony in Section 2.3, and is also similar to the checking model of agreement in early Minimalism. I believe there is value in such an approach, but the present system more clearly highlights the structural configurations of the agreeing heads, which are the primary focus of this paper.

There is also an existing version of MG which handles agreement (Ermolaeva 2018; Ermolaeva and Kobele 2022, 2023). In this system, agreement occurs via dependencies created by Merge and Move, and is restricted through subdiacritics on the relevant MG features on nodes along dependency paths. This is analogous to using c-strings obtained from multidominance trees, an intriguing possibility which merits future exploration. One disadvantage of the model is that long-distance agreement requires either covert movement or passing of features along unbounded selectional chains in the absence of any morphological realization. The former, assuming covert movement to affect scope, contradicts recent empirical findings, including the famous Tsez data (Polinsky and Potsdam 2001). The latter is problematic from a subregular perspective, since arbitrarily complex selectional features can simulate any regular tree constraint (cf. Rogers 1997; Graf 2013).

There are yet other reasons to assume agreement to be independent from movement. First and foremost, it would be be methodologically backwards to do otherwise, given that we are trying to establish their formal properties in the first place. Also, even if the claim that both phenomena are TSL-2 is upheld, this does not imply that they must be unified in the grammar; instead, each can be seen as an independent manifestation of the same underlying computational resources (cf. Graf 2022a). Likewise, I assume case to be assigned/licensed independently; see Hanson (2023) for a subregular approach to case that uses MG dependency trees. In summary, I treat agreement, movement, and case dependencies as being essentially autonomous, though they may interact when one tier grammar makes reference to features that are themselves regulated by another tier. Multiple examples of this sort appear in the following section, in which I survey a wide variety of agreement patterns from the syntactic literature and show that they are all TSL-2.

Table 1: Variants of a TSL-2 grammar and corresponding agreement phenomena

	Tier projection	Tier constraints	Phenomenon
a.	All $\pm\phi$ elements	Strict matching of $+\phi/-\phi$	Minimality
b.	Some D heads do not project	(as in (a))	Invisibility
c.	Some non-agreeing items also project	(as in (a))	Blocking
d.	(as in (a))	Swap order of $+\phi/-\phi$	Upward agreement
e.	(as in (a))	Allow sequential $+\phi$	Chain agreement
f.	(as in (a))	Allow sequential $-\phi$	Multiple agreement

4 THE TYPOLOGY OF AGREEMENT

Graf (2022b) showed how the space of parameters made available by a TSL-2 grammar closely matches the attested variation in movement patterns across languages. Here, I do the same for ϕ-agreement. We begin with examples of variation in the set of elements which are projected on the tier, which together with their features controls the set of agreeing, invisible, and blocking elements (Section 4.1). In addition, TSL-2 also permits variation in directionality (Section 4.2), as well as seemingly complex configurations in which multiple probes share a single goal (Section 4.3) or a single probe interacts with multiple goals (Section 4.4). The section closes by revisiting the power of TSL-2 and its alignment with the observed typology (Section 4.5).

An overview of these parameters of the grammar and the corresponding agreement patterns is given in Table 1; the full set of patterns treated in this paper is summarized in Table 2 at the end of this section. The existence of such patterns is hardly a mystery, but rather to be expected if agreement is TSL-2. We do not expect every logically possible pattern to be attested since, as discussed briefly in the introduction, there are other factors influencing typology, including but not limited to constraints on acquisition and diachronic development. But we do expect to find a reasonably diverse subset of the patterns made possible by the computational system, and this is certainly the case for agreement.

It is not possible to conduct an exhaustive survey here, so I have chosen to focus on two major themes – case-sensitive agreement and complementizer agreement – in order to show both that the same formal patterns occur across agreement phenomena and that various patterns are attested within a single phenomenon. In particular, I do not treat any patterns in which subfeatures of ϕ act independently; while clearly important, in principle these involve multiple tiers and are therefore beyond the scope of this paper. Similarly, I do not provide a detailed analysis of patterns in which a single predicate overtly agrees with multiple DPs, for reasons discussed in Section 2.4. However, I will sketch how this could be done in terms of the interaction-satisfaction theory (Deal 2015), applying the approach to the optionality problem mentioned above and discussing some of the caveats.

Invisibility and blocking 4.1

We have already seen both invisibility and blocking in action, even in the simple examples of English subject-verb agreement: elements such as V and *there* are invisible while D heads block agreement with more distant DPs, as do C heads. The visibility conditions for EPP-movement are similar, but declarative C does not block *wh*-movement, and so does not appear on the corresponding tier. Furthermore, exactly which elements agree, block agreement, or are invisible varies across languages. In this section, we examine several examples of case-sensitive agreement, in which the behavior of a DP depends on its case.

Our first example comes from Hindi, which features split ergative case marking conditioned by aspect (Mahajan 1990). Imperfective clauses have a nominative-accusative pattern while perfective clauses are ergative-absolutive, as shown in (7).[12] I assume that nominative

[12] Abbreviations used in this paper: 1 = first person, 2 = second person, 3 = third person, C1 = class 1, C2 = class 2, CS = construct state, DAT = dative, DEM = demonstrative, ERG = ergative, F = feminine, FUT = future, GEN = genitive, HAB = habitual, IPFV = imperfective, LOC = locative, M = masculine, NF = non-finite, NOM = nominative, OV = object voice, PASS = passive, PFV = perfective, PL = plural, PRF = perfect, PST = past, SG = singular, SV = subject voice.

and absolutive case (both unmarked) are the same case, call it nominative. Also, Hindi displays differential object marking, so not all objects are overtly case-marked; I gloss these as nominative as well. We see that in the imperfective clause the verb agrees with the subject, while in the perfective clause it agrees with the object. Thus, descriptively, the verb agrees with the highest nominative DP, and ergative DPs are invisible.

(7) Case-sensitive agreement in Hindi (Mahajan 1990)

a. Raam roTii khaataa thaa.
Raam.**M.NOM** bread.F.NOM eat.IPFV.M be.PST.M
'Raam ate bread (habitually).'

b. Raam-ne roTii khaayii.
Raam.**M-ERG** bread.F.NOM eat.PFV.F
'Raam ate bread.'

As discussed in the previous section, I assume that case information is available in the form of syntactic features like those for movement and agreement; see Preminger (2014) for a syntactic argument in favor of this idea. I also assume that agreement is conditioned on case-marking rather than the reverse (Bobaljik 2008). Thus, all we need to do to capture the Hindi agreement pattern is to modify the tier alphabet for the ϕ-agreement tier: instead of projecting all DPs, we project only nominative DPs, since only these are ever eligible for agreement. The tier constraints remain unchanged.

Let us confirm that this analysis derives the correct results. I assume the same basic T/*v*/V clause structure as in English unless there are relevant differences. In the imperfective Hindi clause, we have an additional auxiliary verb, which can be assumed to occupy an Asp(ect) projection. The resulting structures for the sentences in (7) are as shown in Figure 7. For simplicity, I will ignore the agreement on the non-finite verb, focusing just on the finite verb (we will return to the issue of multiple agreeing elements in Section 4.3).

C-strings and their ϕ-agreement tiers for these examples are shown in (8) along with the illicit opposite agreement configurations. In the imperfective clause both subject and object are projected, so only the subject can agree, as in (8a); if $-\phi$ is placed on the object (8b), the subject intervenes, resulting in a minimality violation. In the

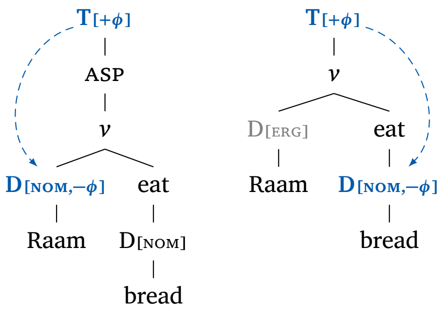


Figure 7: Case-sensitive agreement in Hindi. Ergative DPs are invisible, causing the object to agree instead of the subject

imperfective clause, this is reversed. If $-\phi$ is placed on the subject, this will not work as it is ergative and therefore does not appear on the tier, leaving a lone probe (8c). Instead, it is the object that must agree, since it is adjacent to T on the tier (8d). This is also what happened in the example of agreement across existential *there* in Section 3.4.

(8) Example c-strings and tier projections for Hindi

a. Imperfective clause, subject agreement (✓)

- C-string: $\mathrm{T}_{[+\phi]} \cdot \mathrm{ASP} \cdot v \cdot \mathrm{D}_{[\textsc{nom},-\phi]} \cdot \mathrm{V} \cdot \mathrm{D}_{[\textsc{nom}]} \cdot \mathrm{N}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\textsc{nom},-\phi]} \cdot \mathrm{D}_{[\textsc{nom}]}$
- Constraints violated: n/a

b. Imperfective clause, object agreement (✗)

- C-string: $\mathrm{T}_{[+\phi]} \cdot \mathrm{ASP} \cdot v \cdot \mathrm{D}_{[\textsc{nom}]} \cdot \mathrm{V} \cdot \mathrm{D}_{[\textsc{nom},-\phi]} \cdot \mathrm{N}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\textsc{nom}]} \cdot \mathrm{D}_{[\textsc{nom},-\phi]}$
- Constraints violated: $\mathrm{X}_{[+\phi]} \cdot \mathrm{X}$, $\mathrm{X} \cdot \mathrm{X}_{[-\phi]}$

c. Perfective clause, subject agreement (✗)

- C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[\textsc{erg},-\phi]} \cdot \mathrm{V} \cdot \mathrm{D}_{[\textsc{nom}]} \cdot \mathrm{N}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\textsc{nom}]}$
- Constraints violated: $\mathrm{X}_{[+\phi]} \cdot \mathrm{X}$

d. Perfective clause, object agreement (✓)

- C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[\textsc{erg}]} \cdot \mathrm{V} \cdot \mathrm{D}_{[\textsc{nom},-\phi]} \cdot \mathrm{N}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\textsc{nom},-\phi]}$
- Constraints violated: n/a

It should be noted that some Hindi verbs take dative subjects, and that these are also invisible.[13] In other words, it really is only nominatives that can agree. To summarize, the TSL grammar for Hindi is shown in (9). From this point forward, I will highlight what has changed in comparison to the English grammar from Section 3.4. In this case, only the tier projection rules have changed.[14]

(9) ϕ-agreement tier for Hindi

- Project: all T, **D if [NOM]**, all C
- Constraints: $\left\{\begin{array}{cc} X_{[+\phi]} \cdot X_{[+\phi]}, & X_{[-\phi]} \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot X, & X \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot \ltimes, & \rtimes \cdot X_{[-\phi]} \end{array}\right\}$

It should also be emphasized that there is nothing inherent about the (in)visibility of certain cases. In Nepali, which is closely related to Hindi, the verb agrees with the subject whether it is ergative or nominative (Coon and Parker 2019), as shown in (10).

(10) Case-insensitive agreement in Nepali (Coon and Parker 2019)

a. Maile yas pasal-mā patrikaā kin-ē.
1SG.ERG DEM store-LOC newspaper.NOM buy-**1SG**
'I bought the newspaper in this store.'

b. Ma thag-ī-ē.
1SG.NOM cheat-PASS-**1SG**
'I was cheated.'

Nepali also allows dative subjects, and these do not agree, just as they do not in Hindi. Broadly speaking, there appears to be a hierarchy

[13] The same is true of marked objects, which can be considered to bear accusative case. Thus, when the subject is either ergative or dative and the object is accusative, there is no DP which is eligible for agreement. In this case, default agreement arises. We will see an example of this momentarily in our discussion of Icelandic, so I omit treatment of this phenomenon here.

[14] A reviewer expressed concern about the potential power of these conditional tier projection rules. Because the lexicon is finite, so too is the set of possible tier projections. Therefore, we gain no power compared to exhaustively listing every item. The current notation serves only as a convenient shorthand.

for case visibility in which unmarked case (nominative) ranks above dependent cases (accusative and ergative), followed by oblique cases (dative); each language chooses a point along the hierarchy below which DPs are invisible for agreement (Bobaljik 2008). As such, Nepali is best characterized as a language in which DPs bearing unmarked or dependent case are visible, but those with an oblique case are not. We can easily encode such information in our TSL grammar. In the case of Nepali agreement, we simply project DPs if they are nominative or ergative, though not if they are dative, as in (11).

(11) ϕ-agreement tier for Nepali

- Project: all T, **D if [NOM/ERG]**, all C
- Constraints: $\left\{ \begin{array}{cc} X_{[+\phi]} \cdot X_{[+\phi]}, & X_{[-\phi]} \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot X, & X \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot \ltimes, & \rtimes \cdot X_{[-\phi]} \end{array} \right\}$

Of course, we could do this just as easily for a combination of cases which violates the case visibility hierarchy. However, there is good reason to think that implicational hierarchies such as this one should be attributed to extragrammatical factors, since they are just one member of a much larger class of monotonicity effects which are prevalent throughout language; see Graf (2020) for discussion.[15]

Next, let us turn our attention to blocking effects. What is especially interesting about dative DPs is that while they are often invisible for agreement, they are also known to block it, a phenomenon known as *dative intervention*. A famous example comes from Icelandic, where both possibilities occur depending on the structure. First, (12) is an example of agreement across a dative subject in a simple transitive clause, demonstrating invisibility; next, (13) shows the transitive expletive construction, one of the contexts where datives instead block agreement. It is important to note that the singular verb form in (13a) is not agreement with the dative DP, but a default ending.

(12) Dative invisibility in Icelandic
(Holmberg and Hróarsdóttir 2003)

[15] Note that even if the grammar formalism is formulated in such a way so as to enforce such hierarchies, this still does not explain their existence, but rather raises the question as to why the grammar should be this way.

Henni likuðu hestarnir.
her.SG.DAT liked.PL the.horse.PL.NOM

'She liked the horses.'

(13) Dative intervention in transitive expletives (Holmberg and Hróarsdóttir 2003)

a. Það finnst einhverjum stúdent
EXPL find.SG some student.SG.DAT
[tölvurnar ljótar].
the.computer.PL.NOM ugly

b. * Það finnast einhverjum stúdent
EXPL find.PL some student.SG.DAT
[tölvurnar ljótar].
the.computer.PL.NOM ugly

'Some student finds the computers ugly.'

It is well known that non-nominative subjects do not trigger subject-verb agreement in Icelandic (Andrews 1982; Thrainsson 2007, a.o.). This applies not only to dative subjects, but also to genitive and accusative subjects when they occur. In most cases, the result is long-distance agreement with a lower nominative, as in Hindi. But, in the transitive expletive construction, dative subjects intervene, at least as a first approximation. The full data is quite complex, as the visibility of a dative DP is determined in part by whether it undergoes a specific type of movement which is not always available; on top of this, long-distance agreement is subject to dialectal differences and is optional for certain speakers under certain conditions (Sigurðsson and Holmberg 2008; Kučerová 2016). For now, let us focus just on the above data, in which datives are blockers. Later, we will deal with the interaction with movement. I do not attempt to treat optionality here, though we will consider several approaches to optionality in Section 4.4 which could potentially be applied to the Icelandic data.

I assume the transitive expletive construction to involve a small clause structure, modeled here as a PredP, as shown in Figure 8. Since dative DPs block agreement with a more distant DP, they must be projected on the ϕ-tier just like nominatives. As for the fact that we get default agreement in cases of dative intervention, there are several plausible ways in which this could be analyzed, each of which

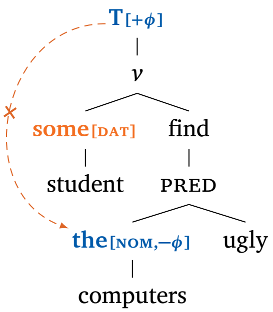


Figure 8:
Dative intervention in Icelandic transitive expletive construction. Dative DPs do not agree, but also block agreement with a more distant DP

is TSL. For concreteness, let us assume the default agreement means that T has not agreed with anything. In other words, we can have a lone probe without a goal, at least in this specific circumstance.[16] We modify the constraints accordingly, banning a probe which is immediately followed by a non-agreeing DP only when it is nominative and therefore eligible for agreement.

The provisional grammar for Icelandic is given in (14), with the relevant constraint modification highlighted. Next, c-strings and their tiers for the licit and illicit agreement configurations in (13) are shown in (15). This time, the items that are projected are the same in either case; what differs is whether the lower nominative bears $-\phi$. When it does, the derivation is illicit since it is preceded by the higher dative. When it does not, no constraints are violated, since T need not agree in this context.

(14) ϕ-agreement tier for Icelandic (provisional)

- Project: all T, all D, all C
- Constraints: $\left\{ \begin{array}{cc} \mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi]}, & \mathrm{X}_{[-\phi]} \cdot \mathrm{X}_{[-\phi]}, \\ \mathrm{X}_{[+\phi]} \cdot \mathbf{D}_{\textbf{[NOM]}}, & \mathrm{X} \cdot \mathrm{X}_{[-\phi]}, \\ \mathrm{X}_{[+\phi]} \cdot \ltimes, & \rtimes \cdot \mathrm{X}_{[-\phi]} \end{array} \right\}$

(15) C-strings and tier projections for Icelandic transitive expletives

a. Default agreement (✓)
- C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[\mathrm{DAT}]} \cdot \mathrm{V} \cdot \mathrm{Pred} \cdot \mathrm{D}_{[\mathrm{NOM}]} \cdot \mathrm{A}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\mathrm{DAT}]} \cdot \mathrm{D}_{[\mathrm{NOM}]}$
- Constraints violated: n/a

[16] In Preminger's (2014) terms, agreement is an obligatory operation in the sense that it must occur when applicable, not that it must occur no matter what.

b. Agreement across dative DP (✗)

- C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[\text{DAT}]} \cdot \mathrm{V} \cdot \mathrm{Pred} \cdot \mathrm{D}_{[\text{NOM},-\phi]} \cdot \mathrm{A}$
- ϕ-agr. tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[\text{DAT}]} \cdot \mathrm{D}_{[\text{NOM},-\phi]}$
- Constraints violated: $\mathrm{X} \cdot \mathrm{X}_{[-\phi]}$

In the above analysis, we are assuming that the dative DP cannot itself bear $-\phi$; in contrast, finite T is lexically specified to always carry $+\phi$ since it must agree when possible. A common alternative is to say that datives do agree, but that default features are transmitted. In this analysis, the dative subject does bear $-\phi$ in the licit agreement configuration, and agreement with the lower nominative is just an ordinary minimality violation. This situation, where the fine details of the analysis have no bearing on whether the phenomenon in question is TSL, seems to be quite common, and points to the robustness of the computational characterization of the empirical facts; we will encounter several more examples like this later in this paper.

At this point, we have seen how our grammar can be adjusted to account for DPs which are invisible to or block agreement according to their case. We can also handle variable visibility within a single language, which as mentioned above is a core aspect of the Icelandic pattern. Again, the full data is notoriously complex, so to keep the discussion simple while still addressing the relevant computational issue, we will add just one additional data point. Recall that dative subjects in simple transitive clauses are invisible. Long-distance agreement is also possible in sentences analogous to the transitive expletive construction, but in which no expletive is inserted and the logical subject raises to Spec-TP, as is assumed in simplex sentences like (12). This is shown in (16).

(16) Long-distance agreement when dative DP moves to Spec-TP (Holmberg and Hróarsdóttir 2003)

Einhverjum	stúdent	finnast	[tölvurnar
some	student.**SG**.**DAT**	find.**PL**	the.computer.**PL**.NOM
ljótar].			
ugly			

‘Some student finds the computers ugly.’

Based on this data, we would say that datives are invisible precisely when they undergo EPP movement.[17] Accordingly, we project all nominatives, plus datives which do *not* bear −EPP; other non-nominative DPs are always invisible for agreement. Since default agreement also occurs in intransitive sentences with a dative subject, such as (12), whose tier would contain $\mathrm{T}_{[-\phi]}$ at the right edge of the tier, we also remove the constraint $\mathrm{X}_{[+\phi]} \cdot \ltimes$. The revised grammar is presented in (17) below. The tier projection rule for D heads is fairly complex; I have taken what I believe to be the least confusing option, which is to list nominative and dative DPs separately. Similarly, I indicate that a constraint has been removed by striking it out.

(17) ϕ-agreement tier for Icelandic (revised)

- Project: all T, **all $\mathrm{D}_{[\textsc{nom}]}$**, **$\mathrm{D}_{[\textsc{dat}]}$ if not [−EPP]**, all C
- Constraints: $\left\{ \begin{array}{cc} \mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi]}, & \mathrm{X}_{[-\phi]} \cdot \mathrm{X}_{[-\phi]}, \\ \mathrm{X}_{[+\phi]} \cdot \mathbf{D}_{[\textsc{nom}]}, & \mathrm{X} \cdot \mathrm{X}_{[-\phi]}, \\ \cancel{\mathrm{X}_{[+\phi]} \cdot \ltimes}, & \rtimes \cdot \mathrm{X}_{[-\phi]} \end{array} \right\}$

The situation with visibility of dative DPs in Icelandic is essentially the opposite of the differential object marking pattern mentioned in Section 3.4, in which the DP becomes visible to its case assigner if and only if it *does* move. Either way, we must refer to a movement feature on the ϕ-tier just as we did with case in Hindi. As I will argue in the following sections, there are yet other movement features which interact with agreement in a similar way.

4.2 *Directionality*

The issue of directionality has received considerable attention in the theoretical literature on agreement. Theories differ as to whether it is always downward (Chomsky 2000), always upward (Zeijlstra 2012), or varies parametrically (Baker 2008). (To be clear, when we say here that agreement is downward, this means that the goal appears below

[17] According to Kučerová (2016), the correct generalization is that dative DPs in Icelandic are invisible if they undergo object shift (movement to Spec-*v*P), and intervene if they do not. This movement tends to be unavailable in the transitive expletive construction for semantic reasons.

the probe, and vice versa for upward agreement.) Despite this theoretical disagreement, from the present perspective agreement is clearly predicted to be able to proceed in either direction. This is because TSL patterns do not have any fixed notion of directionality: for any two elements X and Y, the grammar may allow XY, YX, both, or neither. Thus, it is not surprising that in phonology we find both progressive and regressive harmony. Likewise, in syntax subject-verb agreement is usually downward looking (since the subject is below T in the derivation tree), while negative concord is upward looking, as is case concord within the DP (if case is inherent on D, as we have been assuming).[18]

Ideally, we would like to see evidence of variation within a single agreement phenomenon. This appears at first glance to be true of complementizer agreement (Diercks 2013). In West Flemish, we find cases where the complementizer heading an embedded clause agrees downward for number with the embedded subject (18). In contrast, in Lubukusu (a Bantu language spoken in Kenya), we find upward agreement for noun class with the next higher subject (19). Note that not all complementizers agree in Lubukusu.

(18) Downward agreement in West Flemish (Diercks 2013)

a. Kpeinzen da-j [(gie) morgen goat].
I.think that-**you (you)** tomorrow go
'I think that you'll go tomorrow.'

b. Kvinden dan [die boeken te diere zyn].
I.find that.**PL** the book.**PL** too expensive be.**PL**
'I find those books too expensive.'

(19) Upward agreement in Lubukusu (Diercks 2013)

a. Ba-ba-ndu ba-bolela Alfredi [ba-li a-kha-khile].
C2-**C2**-people **C2**-said C1.Alfred **C2**-that C1-FUT-conquer
'The people told Alfred that he will win.'

[18] Even if a harmony pattern is symmetric on the surface, as in the example in Section 2.3, the process that generates it may be clearly directional. TSL string languages have been generalized to functions to model such processes; see Burness *et al.* (2021) for an overview.

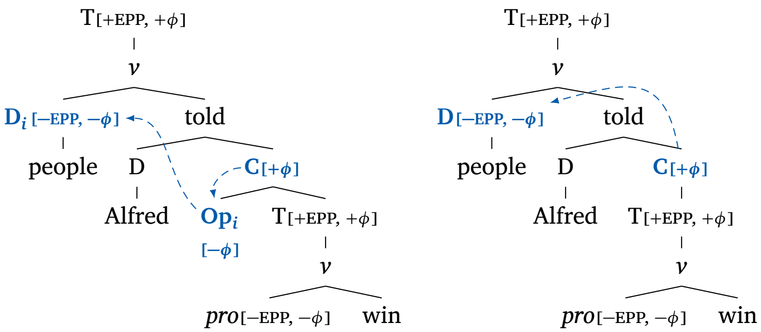


Figure 9: Complementizer agreement in Lubukusu. Left: indirect analysis. Right: direct agreement. In either analysis, one upward dependency is involved

b. Alfredi ka-bolela ba-ba-ndu [a-li
c1.Alfred **c1**-said c2-c2-people **c1**-that
ba-kha-khile].
c2-FUT-conquer
'Alfred told the people that they will win.'

This data is particularly informative in that in neither case does the agreeing DP move out of the embedded CP, avoiding ambiguity in directionality of agreement depending on whether the DP agrees from the lower or higher position. However, there is a complication. In Diercks' analysis of Lubukusu, the matrix clause subject actually binds an operator in embedded Spec-CP, which in turn agrees locally with embedded C, a process which he calls *indirect agreement*. In our terms, this means that the (local) agreement between C and the operator is downward, though the binding relation between the operator and the subject is still upward in the sense that the bound operator must be licensed from above. For comparison, Figure 9 shows Diercks' analysis for (19a) alongside the direct agreement analysis.

Diercks' arguments against upward agreement in Lubukusu can be summarized as follows: 1) agreement is strictly subject-oriented, and other intervening DPs are ignored; 2) in subject questions the verb follows a reduced agreement paradigm while complementizer agreement is as usual; and 3) Lubukusu also features hyperraising (i.e. raising out of a finite clause), and complementizer agreement is absent when this occurs across a C head which is otherwise expected to agree. None of these arguments hold up. First, regarding subject-orientation, we have already seen multiple examples where agreement targets only a subset of DPs (e.g. depending on case) and the literature contains many besides these. We can easily pick out the subject in the TSL analysis as the DP which undergoes EPP movement. Second, the fact that subject-verb agreement may sometimes follow a different paradigm is irrelevant, as there is no reason to think that complementizer agreement here is parasitic on verbal agreement. Finally, the lack of complementizer agreement in hyperraising constructions is in fact predicted by the present model, since the subject appears only below C in the dependency tree and is thus invisible to upward looking dependencies.[19]

While Diercks presents additional arguments in favor of his indirect agreement analysis, they are circumstantial at best. For example, he draws a parallel between certain blockers of complementizer agreement and well-known binding phenomena, which is suggestive of the presence of a bound variable. But as we have already discussed, blocking conditions for agreement can also be quite complex and varied, and there is no particular reason to think that this data can only be explained in terms of binding. This being the case, it is simpler to dispense with the bound operator and assume direct agreement.

Before continuing, I wish to stress that whether we choose to analyze the dependency in question as binding or agreement, the formal shape of the pattern is identical; the only difference is whether the lower element is the C head itself or an operator in its specifier. Furthermore, even if Diercks' analysis is correct for Lubukusu, this would not be enough to discount the existence of upward agreement as a whole; indeed, he does not discuss any other possible instances.

That said, there are in principle several ways to implement variable directionality, and I see no strong reason to prefer one over the others, so I will take the obvious route and simply specify that agreeing C must follow its goal on the tier rather than precede it. (An alternative will be discussed at the end of this section.) Also, since the agreement on C is strictly subject-oriented, only DPs bearing −EPP are projected on the ϕ-agreement tier, as in our treatment of case-sensitive agree-

[19] This last fact is somewhat mysterious in a Minimalist analysis, since without additional assumptions there is no reason why a raised subject should not value the lower complementizer, or alternatively bind the operator in Spec-CP. Diercks (2013) proposes one possible solution. Under the present model, no special treatment is required. In contrast, the present model may struggle when movement feeds subsequent operations, as mentioned in Section 3.2.

ment. Finally, for simplicity I will ignore agreement with T, assuming it to be regulated on a separate tier as discussed in Section 2.4.

As usual, a selection of licit and illicit c-strings are given in (20), representing variants of example (19a). First, we have agreement with the upper subject, shown in (20a). The intervening object is invisible, so the subject immediately precedes the agreeing C head, as required. Next are two illicit configurations, in which agreement is attempted with the upper object (20b) and the lower subject (20c), respectively. The former does not work because the object is missing from the tier, leaving the probe on C without a goal. In the latter case, the goal follows the probe rather than preceding it, violating multiple constraints.

(20) C-strings and tier projections for Lubukusu

a. Complementizer agrees with upper subject (✓)
   - C-string: $\mathrm{T}_{[+\textsc{epp}]} \cdot v \cdot \mathbf{D}_{[-\textsc{epp},\,-\phi]} \cdot \mathrm{V} \cdot \mathrm{D} \cdot \mathbf{C}_{[+\phi]} \cdot \mathrm{T}_{[+\textsc{epp}]} \cdot v \cdot \mathrm{D}_{[-\textsc{epp}]} \cdot \mathrm{V}$
   - Tier: $\mathbf{D}_{[-\textsc{epp},\,-\phi]} \cdot \mathbf{C}_{[+\phi]} \cdot \mathrm{D}_{[-\textsc{epp}]}$
   - Constraints violated: n/a

b. Complementizer agrees with upper object (✗)
   - C-string: $\mathrm{T}_{[+\textsc{epp},\,+\phi]} \cdot v \cdot \mathrm{D}_{[-\textsc{epp}]} \cdot \mathrm{V} \cdot \mathbf{D}_{[-\phi]} \cdot \mathbf{C}_{[+\phi]} \cdot \mathrm{T}_{[+\textsc{epp},\,+\phi]} \cdot v \cdot \mathrm{D}_{[-\textsc{epp}]} \cdot \mathrm{V}$
   - Tier: $\mathrm{D}_{[-\textsc{epp}]} \cdot \mathbf{C}_{[+\phi]} \cdot \mathrm{D}_{[-\textsc{epp}]}$
   - Constraints violated: $\mathrm{X} \cdot \mathrm{C}_{[+\phi]}$

c. Complementizer agrees with lower subject (✗)
   - C-string: $\mathrm{T}_{[+\textsc{epp}]} \cdot v \cdot \mathrm{D}_{[-\textsc{epp}]} \cdot \mathrm{V} \cdot \mathrm{D} \cdot \mathbf{C}_{[+\phi]} \cdot \mathrm{T}_{[+\textsc{epp}]} \cdot v \cdot \mathbf{D}_{[-\textsc{epp},\,-\phi]} \cdot \mathrm{V}$
   - Tier: $\mathrm{D}_{[-\textsc{epp}]} \cdot \mathbf{C}_{[+\phi]} \cdot \mathbf{D}_{[-\textsc{epp},\,-\phi]}$
   - Constraints violations: $\mathrm{X} \cdot \mathrm{C}_{[+\phi]}$, $\mathrm{X}_{[-\phi]} \cdot \ltimes$

There is a potential problem with this analysis: if $-\phi$ is placed on both the subject and the object, then we get the same result as if it appears only on the subject, but only the latter configuration should be licit. This can be avoided by specifying in the lexicon that only D heads with −EPP may also bear $-\phi$. Similarly, though we have not discussed any examples of non-agreeing complementizers, these are distinguished from agreeing complementizers in that the former never bear $-\phi$, while the latter always do.[20] Thus, our the grammar for Lubukusu is given in (21). Again, the primary change compared to the baseline English grammar is that the constraints have been mirrored.

(21) Complementizer ϕ-agreement tier for Lubukusu

- Project: **D if [—EPP]**, all C
- Constraints:

$$\left\{\begin{array}{cc} X_{[+\phi]}\cdot X_{[+\phi]}, & X_{[-\phi]}\cdot X_{[-\phi]}, \\ \mathbf{X}\cdot \mathbf{X}_{[+\phi]}, & \mathbf{X}_{[-\phi]}\cdot \mathbf{X} \\ \rtimes\cdot \mathbf{X}_{[+\phi]}, & \mathbf{X}_{[-\phi]}\cdot \ltimes \end{array}\right\}$$

Note that Ermolaeva and Kobele (2022) arrive at a similar analysis, in which agreement is upward and targets the highest base-generated argument of the containing clause. However, their analysis requires successive overwriting of the morphology on the C head (the subject's value being the last to be written); as such, the present analysis could be considered simpler. An alternative TSL analysis would be to preserve the direction of probing, allowing D heads to search for and value agreeing C heads below them, as in Adger's (2003) treatment of affix hopping. Compared to the above analysis, we restore the relative positions of $+\phi$ and $-\phi$ in the TSL grammar and instead modify the featural content of the D and C heads in the lexicon. Ultimately, the direction of feature copying is the same under either analysis, suggesting that these two types of analysis are essentially notational variants.

4.3 *Multiple probes, one goal*

Until now, I have omitted treatment of several cases where multiple functional elements agree with the same DP. This included agreement on both the non-finite verb and finite auxiliary in Hindi, and agreement on both the verb and complementizer in West Flemish. Unlike in the example from Lubukusu, in these languages both agreeing elements are above the DP they agree with. In order to model such patterns with a single tier, we can adjust the tier constraints to allow

[20] Failure of agreement in hyperraising across an agreeing complementizer can be treated by selectively relaxing the constraints against lone probes while retaining those against lone goals, similar to our treatment of Icelandic.

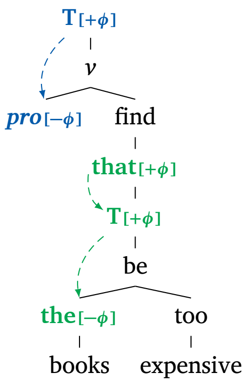


Figure 10: Embedded C and T share a goal in West Flemish. In such cases, the tier grammar permits a sequence of probes followed by a goal

multiple probes to appear in sequence, followed by a single shared goal. In essence, each probe obtains its value not from the source DP, but from the next closest agreeing functional head. This is analogous to the standard treatment of phonological harmony, in which feature spreading proceeds incrementally. We could also utilize separate tiers for each probe as we did for Lubukusu. However, we would lose the parallel with phonology; additionally, given that chain agreement is one of the basic predictions of a TSL-2 model, it is important to demonstrate that it can be handled with a single tier.

Keeping with the theme of complementizer treatment, let us return to the example from West Flemish in (18b), repeated in (22).[21] The structure assumed for this sentence is shown in Figure 10. While Flemish presumably has EPP movement, as well as V2 in main clauses, this is omitted for simplicity. As usual, the categories projected on the ϕ-agreement tier are D, T, and C. Upon extracting the spinal c-string for this example and projecting the tier, shown in (23), we obtain two adjacent chains of agreeing elements: a typical T/D pair in the main clause and a triple C/T/D in the embedded clause.

(22)	Kvinden	[$_{\text{CP}}$ dan	die boeken	[$_{\text{DegP}}$ te	diere]	zyn].
	I.find	that.PL	the book.PL	too	expensive	be.PL

‘I find those books too expensive.’

(23) C-string and tier projection for West Flemish chain agreement

[21] A very similar analysis can be used for concord phenomena of the sort mentioned in Section 4.2.

- C-string: $\mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{V} \cdot \mathrm{C}_{[+\phi]} \cdot \mathrm{T}_{[+\phi]} \cdot v \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{DEG} \cdot \mathrm{A}$
- ϕ tier: $\mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[-\phi]} \cdot \mathrm{C}_{[+\phi]} \cdot \mathrm{T}_{[+\phi]} \cdot \mathrm{D}_{[-\phi]}$

Our TSL grammars have always allowed multiple pairs of agreeing elements. What is new here is that a single agreement chain can contain more than two probes as long as they are ultimately followed by a goal. This is accomplished by removing the constraint $\mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi]}$.[22] Thus, the ϕ-agreement tier grammar for West Flemish is given in (24), with the removed constraint struck out.

(24) ϕ-agreement tier for West Flemish

- Project: all T, all D, all C
- Constraints: $\left\{ \begin{array}{cc} \cancel{\mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi]}}, & \mathrm{X}_{[-\phi]} \cdot \mathrm{X}_{[-\phi]}, \\ \mathrm{X}_{[+\phi]} \cdot \mathrm{X}, & \mathrm{X} \cdot \mathrm{X}_{[-\phi]}, \\ \mathrm{X}_{[+\phi]} \cdot \ltimes, & \rtimes \cdot \mathrm{X}_{[-\phi]} \end{array} \right\}$

As an alternative to the above analysis, we could also mark intermediate elements in the chain with both $+\phi$ and $-\phi$, explicitly signifying that they serve as both a probe and goal for agreement. In this case, the TSL grammar would be set up to allow substrings of the form $\mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi,-\phi]}$ and $\mathrm{X}_{[+\phi,-\phi]} \cdot \mathrm{X}_{[-\phi]}$, but not $\mathrm{X}_{[+\phi]} \cdot \mathrm{X}_{[+\phi]}$. The trouble with this approach is that some elements may need to be lexically specified as being available either as $\mathrm{X}_{[+\phi]}$ or $\mathrm{X}_{[+\phi,-\phi]}$ if they can occur both initially and chain-internally depending on the structural context. I am unaware of any cases in syntax where it is crucial to distinguish between probes which allow agreement to continue and those that do not, though it should be noted that the latter would correspond to so-called *icy targets* in phonology, which both harmonize and prevent harmony from spreading further. For now, it is simpler to treat all probes as equivalent.

At this point, the reader may be wondering about the opposite configuration, in which one probe agrees with several goals. We will allow exactly this in order to model a type of syntactic optionality, to be discussed in Section 4.4. Before that, I present an example which

[22] Note that the chains must be non-overlapping for this to work. In cases where chains of agreeing elements are interleaved with one another, multiple tiers are required.

summarizes several of the patterns that we have examined so far: invisibility, interaction with the other features, and shared goals. This is A′ agreement.

In previous examples, we projected DPs only if they were nominative, or only if they were EPP movers. These are both features normally associated with A-positions, but if we only project DPs bearing a certain A′ feature on the tier controlling ϕ-agreement, then we can derive agreement that targets A′ positions. A clear example comes from Dinka, a Nilotic language spoken in South Sudan. This language has a V2 clause structure in which agreement targets the initial DP, regardless of whether it is a subject, object, or oblique (Van Urk 2015, ch. 3). Examples with an initial subject (25a) and object (25b) are shown below. Additionally, agreement with Spec-CP occurs in relative clauses (of which *wh*-questions are one type) and intermediate movement sites; an example is given in (26).

(25) Dinka verb agreement with Spec-CP

a. Mòc à-cé yîin tîiŋ.
man 3SG-PRF.SV you see.NF
'The man has seen you.' (Van Urk 2015, ch. 4, 19b)

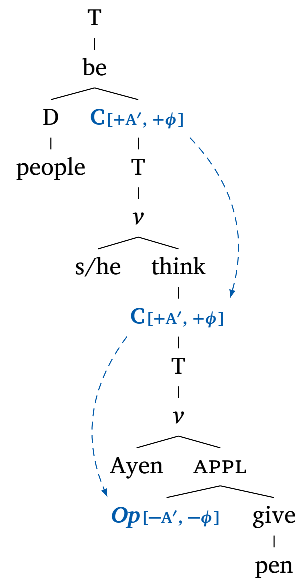
b. Yîin ∅-cíi môc tîiŋ.
you 2-PRF.OV man.GEN see.NF
'You, the man has seen.' (Van Urk 2015, ch. 4, 20a)

(26) Agreement in both matrix and embedded clause

Yè kɔ̂ɔc-kó [$_{CP}$ *Op* é-kè-yá ké tàak
be **people**.CS-which PST-**PL**-HAB.2SG 3PL think.NF
[$_{CP}$ è ___ é-kè-cíi Áyèn ké gâam gàlàm]]?
C PST-**PL**-PRF.OV Ayen.GEN 3PL give.NF pen
'Which people did (s)he think that Ayen had given a pen to?'
(Van Urk 2015, ch. 5, 14a)

Van Urk argues that there is a single generalized A′ feature that encompasses topicalization, relativization, and *wh*-movement. Additionally, there is some variation in whether embedded clauses allow, require, or disallow V2 (Van Urk 2015, p. 130). He proposes that the CP should be split into at least two levels, the lower of which, Fin, hosts V2. Following his lead, I will continue to refer to this head as

Figure 11: A′-agreement in Dinka. Each C head along the movement path agrees with the moving operator



C. It is also this head that is the locus of agreement. Accordingly, I assume that it bears the features $+\mathrm{A}'$ and $+\phi$. Note that I treat intermediate and final landing sites uniformly, mirroring our treatment of chain agreement, though this is not crucial to the analysis; an alternative would be to treat the complementizer morphology as the spell-out of a C head along a movement path, as proposed by Graf (2022b). A slightly simplified structure for example (26) which has been annotated accordingly is shown in Figure 11.[23]

The TSL analysis for this pattern is as follows. Since it is always the moving DP that agrees with C, we project only DPs bearing $-\mathrm{A}'$. C also appears on the tier since it agrees, but T neither agrees not blocks agreement, so it is omitted. The tier constraints are identical to our previous example of West Flemish. The full grammar is shown in (27), and example c-strings corresponding to sentences (25a) and (26) are shown in (28).

(27) ϕ-agreement tier for Dinka

[23] The *ké* morpheme glossed as 3PL is omitted for simplicity. According to Van Urk, this morpheme occurs both as an independent pronoun and as a copy of movement at the edge of *v*P; in this case it is the latter.

- Project: **D if [−A′]**, all C
- Constraints: $\left\{\begin{array}{ll} \cancel{X_{[+\phi]} \cdot X_{[+\phi]}}, & X_{[-\phi]} \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot X, & X \cdot X_{[-\phi]}, \\ X_{[+\phi]} \cdot \ltimes, & \rtimes \cdot X_{[-\phi]} \end{array}\right\}$

(28) C-string ϕ-agreement tiers for Dinka

a. Within-clause A′-movement
- C-string: $C_{[+A', +\phi]} \cdot T \cdot v \cdot D_{[-A', -\phi]} \cdot V \cdot D$
- Tier: $C_{[+A', +\phi]} \cdot D_{[-A', -\phi]}$

b. Long-distance A′-movement
- C-string: $T \cdot be \cdot D \cdot C_{[+A', +\phi]} \cdot T \cdot v \cdot D \cdot V \cdot C_{[+A', +\phi]} \cdot T \cdot v \cdot D \cdot \text{APPL} \cdot D_{[-A', -\phi]} \cdot V \cdot D$
- Tier: $C_{[+A', +\phi]} \cdot C_{[+A', +\phi]} \cdot D_{[-A', -\phi]}$

The present perspective cannot explain why ϕ-agreement occurs on C in Dinka, or why it is sensitive to A′-movement. But given that the distribution of these features is what it is, we correctly predict that their interaction results in an A′ locality profile for ϕ-agreement.

Optionality and multiple goals 4.4

In most of the preceding examples, there has been only a single agreement configuration available for each construction. But it is not rare to find instances of agreement which appear to be optional. As was briefly mentioned in Section 3.4, this happens even in English with agreement across existential *there*: for many speakers both singular and plural agreement are possible, especially when the logical subject is in an embedded clause, as in (6), repeated below as (29a).[24] In contrast, agreement with the logical subject is obligatory if *there* is not inserted (29b).

(29) Optional agreement across *there* in English existential clauses

a. There **seem(s)** to be some squirrels in the attic.

b. Some squirrels **seem(*s)** to be in the attic.

[24] This pattern should not be confused with reduced *'s*, as in *There's some squirrels in the attic*, which is acceptable even for speakers who do not accept %*There is some squirrels in the attic*.

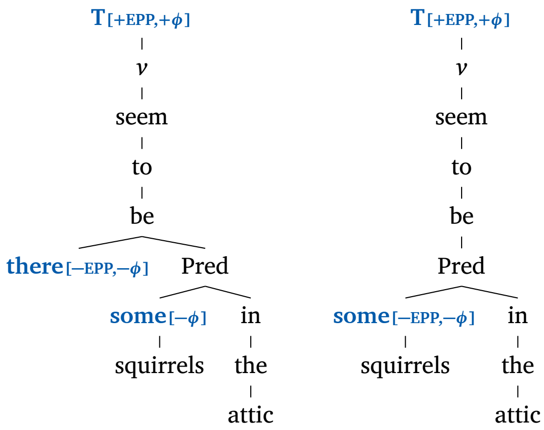


Figure 12: Interaction-satisfaction analysis of optional agreement in English existential clauses. Left: when *there* is inserted, T agrees both with *there* and the logical subject. Right: structure without *there*, where the logical subject is the only target for agreement

Such optionality presents a puzzle which has several plausible solutions, e.g. by positing multiple competing grammars (Kroch 1989). Even if we posit only a single grammar, there are several analytical options. Among these, the *interaction and satisfaction* theory of agreement (Deal 2015) fits well with the analyses presented so far. In the interaction-satisfaction theory, a probe can agree with multiple goals in a manner that is relativized to the individual probe. For each probe (EPP, ϕ, etc.), we specify its *interaction set*, which are the features that the probe agrees with, and its *satisfaction set*, which are the features that cause probing to stop. Upon spell-out, the probe may realize the features of any or all of the elements that it has agreed with in accordance with the morphology of the language. If there is more than one possible output, optionality results.

In the present case, we analyze optional agreement as resulting from agreement with both *there* and the logical subject, as illustrated in Figure 12. We further posit that *there* lacks some feature which is in the satisfaction set for the ϕ-probe on T: perhaps it has number features but lacks person features, for example. This allows T to agree with *there* but continue probing until it finds the logical subject. For present purposes, it does not matter exactly what is deficient about *there*, so for simplicity I will continue to treat all ϕ-features as a unit.

This brings us to the TSL analysis. This time, we project *there* on the tier just like any other DP, and we relax the tier constraints so that a ϕ-goal can be immediately followed by another ϕ-goal if the first goal is *there*, as shown in (30). As usual, c-strings for structures both with and without *there* are provided in (31). Only when *there* is selected is it possible for T to agree with two elements, so this is the only structure in which optional agreement occurs. Note that although both *there* and *some* are marked $-\phi$, this does not imply that they have the same value of ϕ; both are goals and therefore enter the derivation with separate values.

(30) ϕ-agreement tier for English (revised)

- Project: all T, all D, all C
- Constraints: as in (5), but allow there$_{[-\phi]}\cdot$D$_{[-\phi]}$

(31) C-strings and tier projections for optional agreement

a. With *there*
  - C-string: T$_{[+\text{EPP},+\phi]}\cdot v \cdot$ seem $\cdot$ to $\cdot$ be $\cdot$ there$_{[-\text{EPP},-\phi]}$ $\cdot$ Pred $\cdot$ some$_{[-\phi]}\cdot$ in $\cdot$ the $\cdot$ attic
  - ϕ-agreement tier: T$_{[+\text{EPP},+\phi]}\cdot$ there$_{[-\text{EPP},-\phi]}\cdot$ some$_{[-\phi]}$

b. Without *there*
  - C-string: T$_{[+\text{EPP},+\phi]}\cdot v \cdot$ seem $\cdot$ to $\cdot$ be $\cdot$ Pred $\cdot$ some$_{[-\text{EPP},-\phi]}\cdot$ in $\cdot$ the $\cdot$ attic
  - Tier: T$_{[+\text{EPP},+\phi]}\cdot$ some$_{[-\text{EPP},-\phi]}$

In general, an analysis based on the interaction-satisfaction theory can be described as a TSL-2 pattern in which the probe is immediately followed by zero or more agreeing items with features in the interaction set but *not* the satisfaction set, possibly followed by one with features in the satisfaction set (regardless of whether it has any in the interaction set).[25] Thus, the class of agreement patterns which are TSL-2 potentially extends to many others which fall under the general schema of interaction and satisfaction, such as omnivorous agreement (Nevins 2011), also discussed by Deal (2015). For example, in the case of omnivorous number agreement where [PL] outranks [SG], the probe may be valued as [PL] if *any* DP in its search domain is [PL].

[25] Space prohibits me from providing a full analysis, but the basic idea is as follows. Let P denote a probe, I an interacting element, S a satisfying element, and G a normal goal, which both interacts with and satisfies the probe. We allow substrings such as {P $\cdot$ I, P $\cdot$ S, P $\cdot$ G, I $\cdot$ I, I $\cdot$ S, I $\cdot$ G} but not {S $\cdot$ I, S $\cdot$ G, G $\cdot$ I, G $\cdot$ G}. We must also distinguish *actual* interactors from *potential* interactors, perhaps with the same $-\phi$ diacritic used in this paper.

The TSL analysis of this pattern is essentially identical to the example of optional agreement across existential *there*. We place [SG] in the interaction set and [PL] in the satisfaction set. As before, $D_{[-\phi]}$ may therefore be followed by another $D_{[-\phi]}$ iff the first D is singular.

For completeness, I briefly mention an alternative approach to optional agreement, which is to allow certain items to project depending on whether or not they bear $-\phi$. For the present example, we would posit variants of *there* both with and without $-\phi$, and project only the variant bearing $-\phi$. Then, long-distance agreement would occur only when non-agreeing *there* is merged into the derivation. The disadvantage to this approach is that it violates the principle that potential agreeing elements should always project, which we have maintained in all preceding examples due to the pervasiveness of relativized minimality. However, formally there is nothing to prevent us from constructing a tier in this manner, and it may even be necessary for optional extraction morphology (Thomas Graf, p.c.).[26]

There is also a weakness to the interaction-satisfaction approach, which is that the tier constraints distinguish sets of lexical items in a more intricate manner than in previous examples. Unlike our treatment of multiple probes, the behavior of intermediate and final goals is different, and not controlled solely by the $-\phi$ feature. This potentially subverts the typology predicted by the present model, where the presence of $\pm\phi$ (or lack thereof) is the primary factor in the tier constraints. Again, it may the case that optionality should not even be handled within the syntactic grammar, but if we do so, there are several options which fit within the current framework; I leave a more thorough investigation of these and other options to future work.

4.5 *Summary*

To conclude this section, the agreement patterns analyzed in this paper are summarized again in Table 2, now including the specific tier projection functions for each case study. As before, the individual patterns are described in comparison to the baseline pattern of relativized

[26] The idea of treating extraction morphology as constraints on a movement path goes back at least to Bouma *et al.* (2001).

	Example	**Tier Projection**	**Tier Constraints**
a.	Subject-verb agr.	All T/D/C	Strict matching of $+\phi$ and $-\phi$
b.	Case-sensitive agr.	All T/C, D if [NOM]	(as in (a))
c.	Subject-orientation	All C, D if [−EPP]	(as in (a))
d.	A′ agreement	All C, D if [−A′]	(as in (a))
e.	Dative intervention	All T/C/$\text{D}_{[\text{NOM}]}$, $\text{D}_{[\text{DAT}]}$ if not [−EPP]	Non-agreeing dative may follow $+\phi$
f.	Upward agreement	(as in (a))	Swap order of $+\phi/-\phi$
g.	Chain agreement	(as in (a))	Allow sequential $+\phi$
h.	Multiple agreement	(as in (a))	Allow sequential $-\phi$

Table 2: Summary of agreement patterns and their TSL-2 analysis

minimality. For conciseness, only the Hindi variant of case-sensitive agreement is included; additionally, the two components of the Dinka complementizer agreement pattern from Section 4.3 (A′ agreement and multiple probes) have been factored out and listed separately.

Having seen how the proposed model works in a variety of languages, we can now better assess the match between its formal capabilities and the observed typology. Recall from Section 2.1 that by restricting ourselves to TSL, many conceptually simple yet linguistically unnatural constraints become impossible to implement, at least in full generality. Sometimes, it is possible to construct limited counterexamples. For example, threshold counting can be simulated by choosing a tier which contains just the elements of interest and a window size large enough to contain the maximum number we wish to count to. This could be used to construct a language in which a verb is plural iff any of the first *four* DPs in its c-command domain is plural. The restriction to TSL-2 helps to further rule out such tricks.

Indeed, as argued by McMullin and Hansson (2016) and Graf (2022b), TSL-2 gives us exactly the kind of locality restrictions characteristic of natural language: the presence of even a single blocker breaks any long-distance dependency. Some dependencies are strictly local, while others lack blockers altogether, but what we do not find are patterns in which at most one blocker, or two, or three, may be tolerated but no more.[27] Other obvious manipulations of a TSL-2 grammar, such as mirroring the constraints, and allowing adjacent pairs of like elements, likewise correspond to real agreement phenomena.

Even so, as several reviewers of this paper remarked, the freedom of the tier projection function to include or exclude any symbol according to its label seems to overgenerate. For example, we could define a function that projects a random assortment of D heads, rather than all of them. This seems unavoidable since the computational system has no knowledge of the substantive interpretation of the element labels. Furthermore, the existence of lexical exceptions alongside productive generalizations suggests that only the acquisition theory can correctly restrict the set of possible tier projections.[28] It is for these reasons, among others, that I have stressed that the TSL-2 hypothesis is only one component of a complete theory.

5 CONCLUSION

We have seen that a wide variety of agreement phenomena are in fact variations on a simple theme: a TSL-2 pattern which involves the pairing of probes and goals for agreement. This simple model predicts the prevalence of relativized minimality as well as variation in the sets of invisible and blocking elements. Variation across languages can be accounted for using slight adjustments in the tier projection and the constraints in a way that closely tracks the logical possibilities afforded by the formalism.

It is worth reiterating that we do not expect every possible formal pattern to be attested due to the limited number of existing languages

[27] The literature contains theories in which at most one blocker may be crossed, including subjacency (Chomsky 1973) and some versions of phase theory (Chomsky 2001, 2008). Such theories are enmeshed in many auxiliary assumptions, such as successive cyclic movement, making them difficult to evaluate.

[28] The Tolerance Principle (Yang 2016) seems to me to be eminently compatible with the subregular perspective. There already exists some work which attempts to integrate this idea with the learning of TSL-2 grammars. See Belth (2023) for an example from phonology and Hanson (2024b) regarding syntax.

and the other factors influencing typology. As such, it is informative that so many predicted patterns represent actual agreement phenomena. Also, as I have pointed out several times, the formal parallel between syntactic agreement and phonological harmony is particularly close. This can be explained as follows: both are feature-copying phenomena, and both involve the same TSL-2 computations, so both admit the same basic range of formal variation.

Since I have only treated a handful of illustrative examples, additional work is needed to strengthen the claim that individual agreement patterns are in fact TSL-2 over c-strings. In particular, agreement patterns with constraints on multiple tiers merit a detailed examination, in order to confirm that the full grammar is MTSL. It would also be prudent to formalize larger fragments of the grammars of individual languages. And yet, as we have seen several times, the question of whether a given agreement pattern is TSL-2 tends to be robust to differences in the precise details of the analysis. Because of this, it is mainly the empirical facts that need to be scrutinized carefully.

Several additional questions remain open. As mentioned previously, some instances of feeding/bleeding of agreement by movement may require knowledge of the exact position of movers at different points in the derivation. At the same time, it would appear that not all instances of movement feed agreement (similar to how some moved elements undergo semantic reconstruction), as we saw with hyper-raising and complementizer agreement in Lubukusu. This suggests the need for a model which tracks both the base and subsequent positions of movers. Coordination also introduces difficulties such as first/last conjunct agreement, which appear at first glance to be beyond the scope of the c-string model. A more complete model may require the ability to look a short distance into complex left branches, as discussed by Graf and De Santo (2019). Alternatively, we might use feature percolation to bring the correct information up to the top of the structure so that it becomes visible to the containing c-string.

Finally, it is unclear what the exact relation is between agreement in the strict sense and similar long-distance dependencies such as NPI-licensing. The Minimalist literature contains many claims of the form "phenomenon X should be reduced to operation Y", where Y is typically Merge/Move/Agree. I have suggested that this might be the wrong level of granularity, and instead, we should consider movement, agreement, case, and so on to each be instances of TSL computations, and likewise for other conceptually distinct phenomena. Now that we have evidence that all of these patterns are related by their computational complexity, it should be possible to factor out this property in order to tell what, if any, differences remain.

ACKNOWLEDGMENTS

This work was partly supported by NSF Grant BCS-1845344 and by the Institute for Advanced Computational Science at Stony Brook University. Many thanks to Thomas Graf and three anonymous reviewers for their extremely detailed feedback on several drafts of this paper, which resulted in a much improved presentation. I also thank Sandhya Sundaresan, Tom McFadden, John Bailyn, Gary Thoms, and audiences at the NYU Linguistics Brown Bag, CLS 60, and the Workshop on Myopia in Grammar at Leipzig University.

REFERENCES

David ADGER (2003), *Core syntax: A Minimalist approach*, Oxford University Press, doi:10.1093/oso/9780199243709.001.0001.

Avery ANDREWS (1982), The representation of case in Modern Icelandic, in Joan BRESNAN, editor, *The Mental Representation of Grammatical Relations*, pp. 427–503, MIT Press.

Joseph AOUN and Dominique SPORTICHE (1982), On the formal theory of government, *The Linguistic Review*, 2(3):211–236, doi:10.1515/tlir-1983-020303.

Mark BAKER (2008), *The syntax of agreement and concord*, Cambridge University Press, doi:10.1017/CBO9780511619830.

Mark BAKER and Nadya VINOKUROVA (2010), Two modalities of case assignment: Case in Sakha, *Natural Language & Linguistic Theory*, 28(3):593–642, doi:10.1007/s11049-010-9105-1.

Susana BÉJAR and Milan REZAC (2009), Cyclic agree, *Linguistic Inquiry*, 40(1):35–73, doi:10.1162/ling.2009.40.1.35.

Caleb BELTH (2023), Towards a learning-based account of underlying forms: A case study in turkish, in *Proceedings of the Society for Computation in Linguistics 2023*, pp. 332–342, University of Massachusetts Amherst, doi:10.7275/BC8Q-VJ22.

Jonathan BOBALJIK (2008), Where's phi? Agreement as a postsyntactic operation, in Daniel HARBOUR, David ADGER, and Susana BÉJAR, editors, *Phi Theory*, pp. 295–328, Oxford University Press, doi:10.1093/oso/9780199213764.003.0010.

Marisa Ferrara BOSTON, John T. HALE, and Marco KUHLMANN (2010), Dependency structures derived from Minimalist Grammars, in *Proceedings of the 10th and 11th Biennial Conference on The Mathematics of Language*, pp. 1–12, Springer-Verlag, Berlin, Heidelberg, doi:10.1007/978-3-642-14322-9_1.

Gosse BOUMA, Robert MALOUF, and Ivan A. SAG (2001), Satisfying constraints on extraction and adjunction, *Natural Language & Linguistic Theory*, 19(1):1–65, doi:10.1023/a:1006473306778.

Joan BRESNAN, Ronald M. KAPLAN, Stanley PETERS, and Annie ZAENEN (1982), Cross-serial dependencies in Dutch, *Linguistic Inquiry*, 13(4):613–635, `https://www.jstor.org/stable/4178298`.

Phillip Alexander BURNESS, Kevin James MCMULLIN, Jane CHANDLEE, Phillip BURNESS, and Kevin MCMULLIN (2021), Long-distance phonological processes as tier-based strictly local functions, *Glossa: a journal of general linguistics*, 6(1), doi:10.16995/glossa.5780.

Noam CHOMSKY (1957), *Syntactic structures*, De Gruyter Mouton, doi:10.1515/9783112316009.

Noam CHOMSKY (1959), On certain formal properties of grammars, *Information and control*, 2(2):137–167, doi:10.1016/S0019-9958(59)90362-6.

Noam CHOMSKY (1973), Conditions on transformations, in Stephen R. ANDERSON and Paul KIPARSKY, editors, *A Festschrift for Morris Halle*, pp. 232–286, Holt, Rinehart and Winston.

Noam CHOMSKY (1995), *The Minimalist Program*, MIT Press, doi:10.7551/mitpress/9780262527347.001.0001.

Noam CHOMSKY (2000), Minimalist inquiries: The framework, in R. MARTIN, D. MICHAELS, and J. URIAGEREKA, editors, *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, pp. 89–155, MIT Press.

Noam CHOMSKY (2001), Derivation by phase, in Michael KENSTOWICZ, editor, *Ken Hale: A Life in Language*, pp. 1–52, MIT Press, doi:10.7551/mitpress/4056.003.0004.

Noam CHOMSKY (2008), On Phases, in *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*, pp. 133–166, MIT Press, doi:10.7551/mitpress/7713.003.0009.

Jessica COON and Clint PARKER (2019), Case interactions in syntax, in *Oxford Research Encyclopedia of Linguistics*, Oxford University Press, doi:10.1093/acrefore/9780199384655.013.315.

Aniello DE SANTO and Thomas GRAF (2019), Structure sensitive tier projection: Applications and formal properties, in Raffaella BERNARDI, Gregory KOBELE, and Sylvain POGODALLA, editors, *Formal Grammar: 24th International Conference*, pp. 35–50, Springer, doi:10.1007/978-3-662-59648-7_3.

Amy Rose DEAL (2009), The origin and content of expletives: Evidence from "selection", *Syntax*, 12(4):285–323, doi:10.1111/j.1467-9612.2009.00127.x.

Amy Rose DEAL (2015), Interaction and satisfaction in ϕ-agreement, in *Proceedings of NELS 45*, pp. 179–192, GLSA Publications.

Michael DIERCKS (2013), Indirect agree in Lubukusu complementizer agreement, *Natural Language & Linguistic Theory*, 31(2):357–407, doi:10.1007/s11049-013-9187-7.

Marina ERMOLAEVA (2018), Morphological agreement in Minimalist Grammars, in Annie FORET, Reinhard MUSKENS, and Sylvain POGODALLA, editors, *Formal Grammar 2018: 23rd International Conference*, pp. 20–36, Springer, doi:10.1007/978-3-662-56343-4_2.

Marina ERMOLAEVA and Gregory KOBELE (2023), Formal properties of agreeing Minimalist Grammars, in *Proceedings of the Society for Computation in Linguistics 2023*, doi:10.7275/KQY6-0737.

Marina ERMOLAEVA and Gregory M. KOBELE (2022), Agree as information transmission over dependencies, *Syntax*, 25(4):466–507, doi:10.1111/synt.12240.

John GOLDSMITH (1976), *Autosegmental phonology*, Ph.D. thesis, Massachusetts Institute of Technology, `https://dspace.mit.edu/handle/1721.1/16388`.

Thomas GRAF (2013), *Local and transderivational constraints in syntax and semantics*, Ph.D. thesis, University of California, Los Angeles, `https://thomasgraf.net/output/graf13thesis.html`.

Thomas GRAF (2018), Why movement comes for free once you have adjunction, in *Proceedings of CLS 53*, pp. 117–136, The Chicago Linguistic Society.

Thomas GRAF (2020), Monotonicity in syntax, in Dun DENG, Fenrong LIU, Mingming LIU, and Dag WESTERSTÅHL, editors, *Monotonicity in Logic and Language*, pp. 35–53, Springer, doi:10.1007/978-3-662-62843-0_3.

Thomas GRAF (2022a), Subregular linguistics: bridging theoretical linguistics and formal grammar, *Theoretical Linguistics*, 48(3–4):145–184, doi:10.1515/tl-2022-2037.

Thomas GRAF (2022b), Typological implications of tier-based strictly local movement, in *Proceedings of the Society for Computation in Linguistics 2022*, pp. 184–193, University of Massachusetts Amherst Libraries, doi:10.7275/gb65-ht31.

Thomas GRAF (2023), Subregular tree transductions, movement, copies, traces, and the ban on improper movement, in *Proceedings of the Society for Computation in Linguistics 2023*, pp. 289–299, University of Massachusetts Amherst Libraries, doi:10.7275/TK1N-Q855.

Thomas GRAF and Aniello DE SANTO (2019), Sensing tree automata as a model of syntactic dependencies, in *Proceedings of the 16th Meeting on the Mathematics of Language*, pp. 12–26, University of Massachusetts Amherst Libraries, doi:10.18653/v1/W19-5702.

Thomas GRAF and Kalina KOSTYSZYN (2021), Multiple wh-movement is not special: The subregular complexity of persistent features in Minimalist Grammars, in *Proceedings of the Society for Computation in Linguistics 2021*, pp. 275–285, University of Massachusetts Amherst Libraries, doi:10.7275/2dfv-v219.

Thomas GRAF and Connor MAYER (2018), Sanskrit n-retroflexion is input-output tier-based strictly local, in *Proceedings of SIGMORPHON 2018*, pp. 151–160, Association for Computational Linguistics, doi:10.18653/v1/W18-5817.

Thomas GRAF and Nazila SHAFIEI (2019), C-command dependencies as TSL string constraints, in *Proceedings of the Society for Computation in Linguistics 2019*, pp. 205–215, University of Massachusetts Amherst Libraries, doi:10.7275/4rrx-x488.

Kenneth HANSON (2023), A TSL analysis of Japanese case, in *Proceedings of the Society for Computation in Linguistics 2023*, pp. 15–24, University of Massachusetts Amherst Libraries, doi:10.7275/xqhr-r404.

Kenneth HANSON (2024a), Strict locality in syntax, in *Proceedings of CLS 59*, pp. 131–145, The Chicago Linguistic Society.

Kenneth HANSON (2024b), Tiers, paths, and syntactic locality: The view from learning, in *Proceedings of the Society for Computation in Linguistics 2024*, pp. 107–116, University of Massachusetts Amherst Libraries, doi:10.7275/scil.2135.

Jeffrey HEINZ (2018), The computational nature of phonological generalizations, in Larry M. HYMAN and Frans PLANK, editors, *Phonological Typology*, number 23 in Phonetics and Phonology, pp. 126–195, De Gruyter Mouton, doi:10.1515/9783110451931-005.

Jeffrey HEINZ, Chetan RAWAL, and Herbert G. TANNER (2011), Tier-based strictly local constraints for phonology, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human language technologies*, pp. 58–64.

Anders Holmberg and Thorbjörg Hróarsdóttir (2003), Agreement and movement in Icelandic raising constructions, *Lingua*, 113(10):997–1019, doi:10.1016/S0024-3841(02)00162-6.

Riny Huybregts (1976), Overlapping dependencies in Dutch, *Utrecht Working Papers in Linguistics*, 1:24–65.

Riny Huybregts (1984), The weak inadequacy of context-free phrase structure grammars, in G. J. de Haan, M. Trommelen, and W. Zonneveld, editors, *Van Periferie naar Kern*, pp. 81–99, Foris.

Adam Jardine and Kevin McMullin (2017), Efficient learning of tier-based strictly k-local languages, in Frank Drewes, Carlos Martín-Vide, and Bianca Truthe, editors, *Language and Automata Theory and Applications*, pp. 64–76, Springer, Cham, doi:10.1007/978-3-319-53733-7_4.

Aravind K. Joshi (1985), Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?, in David R. Dowty, Lauri Karttunen, and Arnold M. Zwicky, editors, *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, pp. 206–250, Cambridge University Press, doi:10.1017/CBO9780511597855.007.

Ronald M. Kaplan and Martin Kay (1994), Regular models of phonological rule systems, *Computational Linguistics*, 20(3):331–378, `http://www.aclweb.org/anthology/J94-3001.pdf`.

Stefan Keine (2019), Selective opacity, *Linguistic Inquiry*, 50(1):13–62, doi:10.1162/ling_a_00299.

Gregory M. Kobele (2002), Formalizing mirror theory, *Grammars*, 5(3):177–221, doi:10.1023/A:1022104104992.

Gregory M. Kobele (2012), Eliding the derivation: A minimalist formalization of ellipsis, in *Proceedings of the International Conference on Head-Driven Phrase Structure Grammar*, pp. 409–426, CSLI Publications, doi:10.21248/hpsg.2012.23.

Anthony S. Kroch (1989), Reflexes of grammar in patterns of language change, *Language Variation and Change*, 1(3):199–244, doi:10.1017/S0954394500000168.

Ivona Kučerová (2016), Long-distance agreement in Icelandic: locality restored, *The Journal of Comparative Germanic Linguistics*, 19(1):49–74, doi:10.1007/s10828-016-9077-6.

Dakotah Lambert (2021), Grammar interpretations and learning TSL online, in Jane Chandlee, Rémi Eyraud, Jeff Heinz, Adam Jardine, and Menno van Zaanen, editors, *Proceedings of the Fifteenth International Conference on Grammatical Inference*, volume 153 of *Proceedings of Machine Learning Research*, pp. 81–91, PMLR, `https://proceedings.mlr.press/v153/lambert21a.html`.

Dakotah LAMBERT (2023), Relativized adjacency, *Journal of Logic, Language and Information*, 32(4):707–731, doi:10.1007/s10849-023-09398-x.

Dakotah LAMBERT, Jonathan RAWSKI, and Jeffrey HEINZ (2021), Typology emerges from simplicity in representations and learning, *Journal of Language Modelling*, 9(1):151–194, doi:10.15398/jlm.v9i1.262.

Dakotah LAMBERT and James ROGERS (2020), Tier-based strictly local stringsets: Perspectives from model and automata theory, in *Proceedings of the Society for Computation in Linguistics 2020*, pp. 159–166, University of Massachusetts Amherst, Amherst, MA, `https://aclanthology.org/2020.scil-1.21`.

Anoop Kumar MAHAJAN (1990), *The A/A-bar distinction and movement theory*, Ph.D. thesis, Massachusetts Institute of Technology, `https://dspace.mit.edu/handle/1721.1/13650`.

Connor MAYER (2021), Capturing gradience in long-distance phonology using probabilistic tier-based strictly local grammars, in *Proceedings of the Society for Computation in Linguistics 2021*, pp. 39–50, University of Massachusetts Amherst Libraries, doi:10.7275/231q-p480.

Connor MAYER and Travis MAJOR (2018), A challenge for tier-based strict locality from Uyghur backness harmony, in *Formal Grammar 2018: 23rd International Conference*, pp. 62–83, Springer, doi:10.1007/978-3-662-57784-4_4.

Kevin MCMULLIN (2016), *Tier-based locality in long-distance phonotactics: learnability and typology*, Ph.D. thesis, University of British Columbia.

Kevin MCMULLIN and Gunnar Ólafur HANSSON (2016), Long-distance phonotactics as tier-based strictly 2-local languages, in *Proceedings of the 2014 Annual Meeting on Phonology*, Linguistic Society of America, doi:10.3765/amp.v2i0.3750.

Robert MCNAUGHTON and Seymour A. PAPERT (1971), *Counter-free automata*, MIT Press.

Andrew NEVINS (2010), *Locality in vowel harmony*, MIT Press, doi:10.7551/mitpress/9780262140973.001.0001.

Andrew NEVINS (2011), Multiple agree with clitics: Person complementarity vs. omnivorous number, *Natural Language & Linguistic Theory*, 29:939–971, doi:10.1007/s11049-011-9150-4.

Elissa L. NEWPORT and Richard N. ASLIN (2004), Learning at a distance I. Statistical learning of non-adjacent dependencies, *Cognitive Psychology*, 48(2):127–162, doi:10.1016/S0010-0285(03)00128-2.

David PESETSKY and Esther TORREGO (2007), The syntax of valuation and the interpretability of features, in Simin KARIMI, Vida SAMIIAN, and Wendy K. WILKINS, editors, *Phrasal and Clausal Architecture: Syntactic Derivation and Interpretation*, Benjamins, doi:10.1075/la.101.14pes.

Maria POLINSKY and Eric POTSDAM (2001), Long-distance agreement and topic in Tsez, *Natural Language & Linguistic Theory*, 19(3):583–646, doi:10.1023/A:1010757806504.

Omer PREMINGER (2014), *Agreement and its failures*, MIT Press, doi:10.7551/mitpress/9780262027403.001.0001.

Tanya REINHART (1981), Definite NP anaphora and c-command domains, *Linguistic Inquiry*, 12(4):605–635, `https://www.jstor.org/stable/4178245`.

Luigi RIZZI (1990), *Relativized minimality*, MIT Press.

Brian ROARK and Richard SPROAT (2007), *Computational approaches to morphology and syntax*, Oxford University Press, doi:10.1093/oso/9780199274772.001.0001.

James ROGERS (1997), Strict LT_2 : Regular :: Local : Recognizable, in Christian RETORÉ, editor, *Logical Aspects of Computational Linguistics: First International Conference, LACL '96 (Selected Papers)*, volume 1328 of *Lectures Notes in Computer Science/Lectures Notes in Artificial Intelligence*, pp. 366–385, Springer, doi:10.1007/BFb0052167.

James ROGERS, Jeffrey HEINZ, Gil BAILEY, Matt EDLEFSEN, Molly VISSCHER, David WELLCOME, and Sean WIBEL (2010), On languages piecewise testable in the strict sense, in Christian EBERT, Gerhard JÄGER, and Jens MICHAELIS, editors, *The Mathematics of Language*, volume 6149 of *Lecture Notes in Artifical Intelligence*, pp. 255–265, Springer, doi:10.1007/978-3-642-14322-9_19.

James ROGERS, Jeffrey HEINZ, Margaret FERO, Jeremy HURST, Dakotah LAMBERT, and Sean WIBEL (2013), Cognitive and sub-regular complexity, in Glyn MORRILL and Mark-Jan NEDERHOF, editors, *Formal Grammar: 17th and 18th International Conferences*, volume 8036 of *Lecture Notes in Computer Science*, pp. 90–108, Springer, doi:10.1007/978-3-642-39998-5_6.

Marcel Paul SCHÜTZENBERGER (1965), On finite monoids having only trivial subgroups, *Information and Control*, 8(2):190–194, doi:10.1016/S0019-9958(65)90108-7.

Stuart M. SHIEBER (1985), Evidence against the context-freeness of natural language, in Walter J. SAVITCH, Emmon BACH, William MARSH, and Gila SAFRAN-NAVEH, editors, *The formal complexity of natural language*, pp. 320–334, Springer, doi:10.1007/BF00630917.

Halldór Ármann SIGURÐSSON and Anders HOLMBERG (2008), Icelandic dative intervention: Person and number are separate probes, in Roberta D'ALESSANDRO, Susann FISCHER, and Gunna Hrafn HRAFNBJARGARSON, editors, *Agreement restrictions*, pp. 176–194, De Gruyter Mouton, doi:10.1515/9783110207835.251.

Imre SIMON (1975), Piecewise testable events, in *Proceedings of the 2nd GI Conference on Automata Theory and Formal Languages*, pp. 214–222, Springer-Verlag, doi:10.1007/3-540-07407-4_23.

Edward P. STABLER (1997), Derivational Minimalism, in Christian RETORE, editor, *Logical Aspects of Computational Linguistics*, Springer, doi:10.1007/BFb0052152.

Edward P. STABLER (2011), Computational perspectives on Minimalism, in Cedric BOECKX, editor, *Oxford Handbook of Linguistic Minimalism*, pp. 617–643, Oxford University Press, doi:10.1093/oxfordhb/9780199549368.013.0027.

Hoskuldur THRAINSSON (2007), *The syntax of Icelandic*, Cambridge University Press, doi:10.1017/CBO9780511619441.

Coppe VAN URK (2015), *A uniform syntax for phrasal movement: A Dinka Bor case study*, Ph.D. thesis, Massachusetts Institute of Technology, https://dspace.mit.edu/handle/1721.1/101595.

Mai Ha VU, Nazila SHAFIEI, and Thomas GRAF (2019), Case assignment in TSL syntax: A case study, in *Proceedings of the Society for Computation in Linguistics 2019*, pp. 267–276, University of Massachusetts Amherst Libraries, doi:10.7275/sywz-xw23.

Charles YANG (2016), *The price of linguistic productivity: How children learn to break the rules of language*, MIT Press.

Hedde ZEIJLSTRA (2012), There is only one way to agree, *The Linguistic Review*, 29(3):491–539, doi:10.1515/tlr-2012-0017.