Monotonicity in Syntax

Thomas Graf(✉)

Stony Brook University, Stony Brook, NY 11794, USA
mail@thomasgraf.net

Abstract. Extending previous work on monotonicity in morphology and morphosyntax, I argue that some of the most important constraints in syntax can be analyzed in terms of monotonic functions that map specific kinds of syntactic representations to fixed, universal hierarchies. I cover the Ban Against Improper Movement, the Williams Cycle, the Ban Against Improper Case, and omnivorous number. The general method of analysis is remarkably similar across all phenomena, which suggests that monotonicity provides a unified perspective on a wide range of phenomena in syntax as well as morphology and morphosyntax. I also argue that syntax, thanks to extensive work in computational syntax, provides a unique opportunity to probe whether the prevalence of monotonicity principles in natural language is due to computational complexity considerations. Not only, then, is it possible to extend the purview of monotonicity from semantics to syntax, doing so might yield new insights into monotonicity that would not be obtainable otherwise.

Keywords: Monotonicity · Syntax · Typology · Ban against improper movement · Dependent case · Omnivorous number

There has been plenty of research on monotonicity in semantics, but much less on its role in phonology, morphology, and syntax. One could construe this as strong evidence that monotonicity is mostly a semantic phenomenon, but in this paper I will argue for the very opposite position: not only are there syntactic phenomena that can be insightfully analyzed in terms of monotonicity, syntax may be the key to understanding why monotonicity should have any role to play in language, be it in semantics or any other subdomain.

I will investigate a number of phenomena that have been discussed in the generative literature: the Ban Against Improper Movement, the Williams Cycle [36,37], the Ban Against Improper Case [28], and omnivorous number [25]. While these phenomena are widely regarded as unrelated, I show that they can all be unified under the umbrella of a single monotonicity requirement. I do so building on an approach first presented in [11] for morphology and morphosyntax. In this approach, universal grammar is assumed to furnish specific linguistic hierarchies, e.g. for person or number. Linguistic phenomena are analyzed as mappings operating on these linguistic hierarchies, and the typologically attested patterns turn out to be exactly those that can be represented as monotonically increasing

© Springer-Verlag GmbH Germany, part of Springer Nature 2020
D. Deng et al. (Eds.): TLLM 2020, LNCS 12564, pp. 35–53, 2020.
https://doi.org/10.1007/978-3-662-62843-0_3

mappings between two structures. The very same idea can be applied to syntax, given suitable partial orders and linguistic hierarchies.

The paper thus makes several contributions. First, it unifies a number of seemingly unrelated syntactic phenomena. Second, it connects these phenomena to others in morphology and morphosyntax that have been previously analyzed in terms of monotonicity. Finally, the paper shows that there is merit to pushing the study of monotonicity beyond semantics. Moreover, the fact that the computational properties of syntax are better understood than those of semantics means that syntax is a better choice for exploring the connections between monotonicity and computation.

I will proceed as follows: I start out with a general description of monotonicity and how it is applied to morphology and morphosyntax in [11]. Section 2 then discusses one of the most robust constraints on syntactic movement, namely the *Ban Against Improper Movement*. This section also derives a *Ban Against Improper Selection*, another constraint that is widely attested but to the best of my knowledge does not have a standardized name. It also discusses the Williams Cycle, a generalized version of the Ban Against Improper Movement, and the recently proposed Ban Against Improper Case [28]. After that, in Sect. 3, I turn to a very different phenomenon known as *omnivorous number*, and I show that it, too, is an instance of monotonicity in syntax. Finally, Sect. 4 addresses the question why syntax should be sensitive to monotonicity. While I cannot offer a conclusive answer at this point, I argue that this is just a special case of a more general issue: why should any aspect of language care about monotonicity? This is a fundamental question that all research on monotonicity has to tackle, and I conjecture that there might be a link between monotonicity and computation. If this is the case, then syntax is better suited to exploring this connection because the computational properties of semantics are not as well-understood as those of syntax.

1 Background and Prior Work

In [11], a specific approach is presented for explaining typological gaps in morphology and morphosyntax in terms of mappings from underlying algebras to surface forms. It is this approach that will form the conceptual backbone of this paper.

Let us look at adjectival gradation as a concrete example. Each adjective has three forms: positive, comparative, and superlative. In many cases all three forms share the same stem, e.g. *hard-harder-hardest*. But there is also *good-better-best*, and its Latin counterpart *bonus-melior-optimus*. In the former, only the comparative and the superlative have similar stems, while in the latter each form uses a distinct stem. Abstracting away from these specific adjectives, we may refer to these three patterns as AAA, ABB, and ABC. Curiously absent is the pattern ABA, which would correspond to something like *good-better-goodest*. This gap exists across a variety of paradigms beyond adjectival gradation, suggesting a general ban against ABA patterns [3].

As shown in [11], this ban against ABA patterns can be construed as an instance of monotonicity. Consider once more the case of adjectival gradation. The three adjectival forms can be arranged according to their denotational semantics, yielding the adjectival gradation hierarchy

$$\text{positive} < \text{comparative} < \text{superlative}$$

Now assume that we take A, B, and C as arbitrary placeholders for surface forms and put them in an arbitrary order. For the sake of exposition, let's say that this order is

$$\text{A} < \text{B} < \text{C}$$

Patterns AAA and ABC can be viewed as mappings from the adjectival gradation hierarchy into this hierarchy of output forms. For instance, AAA arises when $f(\text{positive}) = f(\text{comparative}) = f(\text{superlative}) = \text{A}$ (note that AAA, BBB, and CCC all describe the same pattern as the important issue is which forms share stems, not whether we denote this stem as A, B, or C). The mappings corresponding to AAA, ABB, ABC, and ABA are depicted in Fig. 1. Since we are dealing with two linear orders, we may also view them as axes of a diagram in which we plot each pattern (Fig. 2).

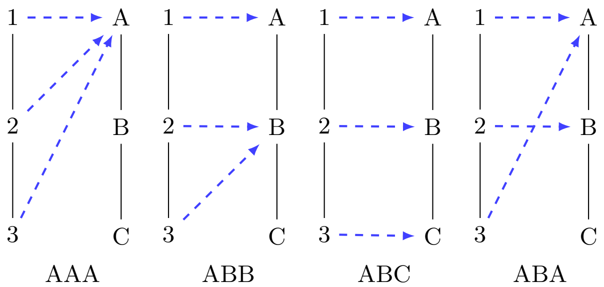


Fig. 1. Pictorial representation of mappings yielding AAA, ABB, ABC, and ABA

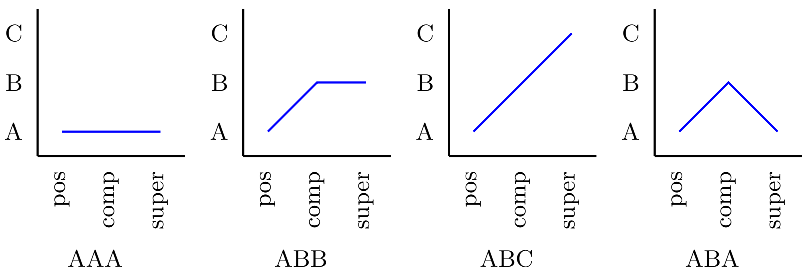


Fig. 2. Diagrammatic representation of the mappings for AAA, ABB, ABC, and ABA

Notice how the unattested ABA pattern differs from the attested ones in that I) it involves two crossing branches in Fig. 1, and II) it is the only pattern to change direction in Fig. 2. Hence we can explain the absence of ABA patterns in terms of some principle that does not allow functions to behave this way. That is exactly what one gets from the familiar notion of monotonicity.

Definition 1. *Let* $\mathcal{A} := \langle A, \leq_A \rangle$ *and* $\mathcal{B} := \langle B, \leq_B \rangle$ *be two partially ordered sets. Then a mapping* f *from* $\mathcal{A}$ *to* $\mathcal{B}$ *is*

- monotonically increasing *iff* $x \leq_A y$ *implies* $f(x) \leq_B f(y)$*,*
- monotonically decreasing *iff* $x \leq_A y$ *implies* $f(y) \leq_B f(x)$*.*

Throughout the paper, I will use the terms *monotonic* and *monotonically increasing* interchangeably. According to the definition above, the ABA pattern for adjectival gradation is not monotonic because we have $f(\text{positive}) = f(\text{superlative}) = \text{A} < \text{B} = f(\text{comparative})$, yet comparative < superlative. Hence the ban against ABA patterns follows from the assumption that mappings must be monotonic and the adjectival gradation forms are ordered such that positive < comparative < superlative.

In isolation, this is not particularly remarkable. But as shown in [11], the idea can be extended to a large number of phenomena in morphology and morphosyntax: personal pronoun syncretism, case allomorphy, noun stem allomorphy, the Person Case Constraint, and the Gender Case Constraint. In some cases, the linguistic hierarchy is not a linear order but a partial one, so that some elements are unordered with respect to each other. Monotonicity generalizes immediately to these partial orders, too, and thus it provides a uniform explanation for a large number of seemingly unrelated typological gaps.

As I will show in the next two sections, the same is true for syntax. I start with a discussion of the Ban Against Improper Movement, which involves hierarchies that are linear orders. In Sect. 3, I then show how the typology of omnivorous number can be explained via monotonicity over a partial order.

Before moving on, though, I have to remark on the general methodology of this approach. The line of research pursued in this paper differs from typical work on monotonicity in that the functions under discussion have fairly small domains and co-domains. Whereas work on monotonicity in semantics often assumes infinite (co-)domains, the most complex function in [11] has a domain of size 16 and a co-domain of size 2. With such small numbers, it is to be expected that most phenomena allow us to order the elements they involve in such a manner that the mapping turns out to be monotonic. This is why it is important that the posited orders be linguistically plausible. Sometimes, multiple orders could be motivated on linguistic grounds—for instance, one may posit a number hierarchy singular < dual < plural on semantic grounds, or instead go with singular < plural < dual due to the typological implication that if a language has a dual, it most likely also has a plural. In this case, there is no *a priori* reason to prefer one order over the other, and the decision is made based on whichever order offers a better fit for the available data. But once the decision has been made, the same hierarchy must be used uniformly across all relevant phenomena;

one cannot use one number hierarchy for phenomenon X and a different number hierarchy for phenomenon Y, as this would only lead to circular reasoning. This paper marks the first foray into syntax for the monotonicity approach, and thus the posited hierarchies are still limited to a few phenomena. Nonetheless, they already succeed at unifying distinct phenomena (for example, Sect. 3 ties the existence of omnivorous number directly to the existence of resolved agreement). While the findings are still limited in scope, they provide a fertile starting point.

2 Restrictions on Movement Types

Generative syntacticians make a distinction between at least three types of syntactic dependencies: selection, A-movement, and A′-movement. These dependencies are subject to a fundamental syntactic law, the Ban Against Improper Movement. These syntactic ideas will be explained in a moment. For now, the key issue is that it is still unclear why natural languages uniformly obey this law. Syntactic formalisms usually have to stipulate it instead of deriving it from independently motivated aspects of syntax. I show that the Ban Against Improper Movement can be reduced to a general monotonicity requirement. The reduction is straight-forward, but it requires us to establish a bit of linguistic background first. Readers who are already familiar with selection, A-movement, and A'-movement can skip ahead to the very last paragraph of Sect. 2.1, which covers everything that is needed to derive the Ban Against Improper Movement (Sect. 2.2). I then argue that the same idea can also account for generalized versions of this ban, such as the Williams Cycle and the Ban Against Improper Case (Sect. 2.3).

2.1 Selection, A-Movement, and A′-Movement

Selection combines a head with its arguments. It is the basic mechanism for establishing head-argument dependencies. There are many ways to handle selection in the grammar. GSPG and HPSG use subcategorization frames [7,27], Tree Adjoining Grammar encodes selectional requirements directly in its elementary trees [19,20], and Minimalist Grammars (which are inspired by Chomsky's Minimalist Program [5]) annotate each lexical item with category and selector features to control the structure-building operation Merge [33,34]. For the purposes of this paper, we can completely abstract away from these technical details. It only matters that there is a broad consensus that syntax involves combining heads with their arguments, and that this phenomenon is what we refer to as *selection*.

There is also a broad consensus that selection is maximally local. That is to say, selection cannot target a phrase that is embedded inside another phrase:

(1) a. John cut [$_{\text{DP}}$ the carrot].

b. * John cut [$_{\text{VP}}$ bought [$_{\text{DP}}$ the carrot]].

While the verb *cut* can select the DP *the carrot* in (1a) , it cannot do so in (1b) where *the carrot* is embedded inside a VP.

An anonymous reviewer points out that this claim is at odds with the fact that *John greeted [[$_{DP}$ whoever] Mary invited]* is well-formed, whereas the minimally different *John greeted [[$_{DP}$ whatever] Mary invited]* is not. This suggests that the verb selects for the wh-phrase inside the complement clause. There are many ways this could be addressed. One might say that the second sentence is in fact syntactically well-formed and that its reduced acceptability is due to semantics. Other analyses allow the features that distinguish *whoever* from *whatever* to pass from the DP onto the head of the clausal complement, maintaining the locality of selection. The monotonicity approach can remain agnostic about this—the precise degree of locality of selection is immaterial as long as selection is less local than A-movement and A′-movement, which are discussed next.

A-movement and A′-movement both establish long-distance dependencies between a phrase and some other position in the sentence. *A-movement*, which is short for *argument movement*, targets positions that are in some way tied to a fixed grammatical function (the precise definition of A-movement is hotly debated, see [29] for an accessible overview). For instance, the promotion of an object to subject position in a passive sentence is commonly regarded as an instance of A-movement, and so is subject raising. Both are illustrated below, with t indicating the position that the phrase *John* is related to via A-movement.

(2) a. John was attacked t. *Passive*

b. John seems t to have cut the carrot. *Subject raising*

In (2a), *John* appears in the subject position but is interpreted as the object of *attacked*. In (2b), *John* is pronounced in the subject position of the matrix clause but is interpreted as the subject of the embedded verb *cut*. In both (2a) and (2b), we are dealing with A-movement because *John* appears in an argument position—a subject position, in this case—but the sentence is interpreted as if *John* resided in some other position.

Some readers may be puzzled that I describe A-movement as a dependency between positions and not as an operation. Admittedly the term originates from Transformational Grammar, where movement is construed as an operation that targets a phrase and puts it in a different position in the phrase structure tree. But just like selection can be implemented in many different ways, there are numerous ways of handling A-movement, many of which do not involve any kind of displacement. In fact, it is even possible to have a dedicated movement operation yet do not use it for A-movement [22]. Just as with selection, the pertinent point here is that syntax involves a cluster of phenomena that is subsumed under *A-movement*, not what specific mechanisms are the driving force behind these phenomena.

This leaves us with *A′-movement*, or *non-argument movement*. As the full name indicates, A′-movement establishes a dependency between positions that are not targeted by A-movement. This includes question formation and topicalization, among others. Neither construction involves a position that is tied to a specific grammatical function like subject or object.

(3) a. Who did Mary attack *t*. *Question formation*
b. John, Mary attacked *t*. *Topicalization*

A-movement and A′-movement differ in several respects, e.g. how they interact with semantic scope. But once again these details are largely immaterial for this paper, except that A-movement is more local than A′-movement; for example, only the latter can operate across finite clauses.

(4) a. * John said that Mary attacked *t*. *A-movement of object*
b. * John seems that *t* attacked Mary. *A-movement of subject*
c. John seems to have *t* attacked Mary. *infinitival A-movement*
d. Who did John say that Mary attacked *t*. *A′-movement*

Here (4a) is illicit under the intended reading that John said that Mary attacked him. We cannot establish an A-movement dependency between *John* and the object position of *attacked* because this dependency would span across the boundary of a finite clause. Similarly, (4b) is not well-formed as the A-movement dependency between *John* and the embedded subject would cross a finite clause boundary. Example (4c) shows that the problem is indeed the finiteness of the clause, as the same A-movement dependency can hold across an infinitival clause boundary. Finally, we see that the A′-movement dependency in (4d) is well-formed even though it holds across a finite clause boundary.

Depending on their theoretic priors, readers may object that the contrasts above can be explained on independent grounds that do not require A-movement to be more local than A′-movement (for instance the Case filter of Government-and-Binding theory). But this objection is based on construing the term "A-movement" as referring to a specific mechanism of the grammar, rather than a cluster of empirical phenomena. The claim is not that A-movement is intrinsically limited to be more local than A′-movement, but that syntax as a whole causes A-movement phenomena to be more limited than A′-movement phenomena. The source of this discrepancy and its causal mechanisms are deliberately abstracted away from, just like the monotonicity analysis in [11] posits a person hierarchy of $1 < 2 < 3$ while remaining agnostic about how (and even whether) person is represented in the grammar or what specific grammatical principles give rise to this order.

To sum up, there are three distinct types of syntactic phenomena that are commonly thought of in mechanical terms as selection, A-movement, and A′-movement. They differ in their locality, with selection as the most local option and A′-movement the least local one. I encode this fact in terms of a general locality hierarchy:

$$\text{selection} < \text{A-movement} < \text{A}'\text{-movement}$$

In the remainder of this section, I will refer to this hierarchy as the linear order $\mathcal{L} := \langle\{\text{selection}, \text{A-movement}, \text{A}'\text{-movement}\}, <\rangle$. In conjunction with monotonicity, $\mathcal{L}$ derives the Ban Against Improper Movement and several generalizations of this ban.

2.2 The Ban Against Improper Movement

The (simplified) examples in Sect. 2.1 may give the impression that a phrase participates in at most one instance of A-movement or at most one instance of A'-movement. But this is not the case. Quite often, a phrase participates in multiple instances of movement, and the manner in which it may do so is regulated by the Ban Against Improper Movement.

Let us consider a concrete example.

(5) [Which boy] does John think *t* impressed everyone?

Here the phrase *which boy* originated from the subject position of the embedded clause. Depending on one's analysis, though, many movement steps are involved in this. For the sake of exposition, I will present a Minimalist analysis of (5). In Minimalism, movement is indeed interpreted as an operation that displaces subtrees, and there are a few additional movement steps that are motivated by theoretical considerations. Consider, then, the sequence of steps that results in (5): First, *which boy* is selected by the verb *impressed* and undergoes A-movement to the embedded subject position. From there, it moves to the left edge of its clause, which is an instance of A′-movement. This is followed by another instance of A′-movement to the left edge of the matrix clause. The resulting phrase structure tree is depicted on the left of Fig. 3 (which also shows the A-movement of *John* to the matrix subject position).

Now contrast the well-formed (5) against the illicit (6).

(6) *[Which boy] does *t* think *t* impressed everyone?

The intended reading for this sentence is *which boy is such that he thinks that he impressed everyone*, but not only is this reading unavailable, the whole sentence is illicit. When we compare the phrase structure tree for (6) on the left of Fig. 3 to the one for (5) on the right, we can see that they differ in what types of movement take place.

In (6), *which boy* is once again selected by *impressed* and then undergoes A-movement to the embedded subject position and A′-movement to the left edge of the embedded clause. But then (5) and (6) diverge. Whereas (5) continues with A′-movement, (6) instead has *which boy* switch back to A-movement. Considered in isolation, this A-movement should be licit as it does not cross a clause boundary—the movement past the complementizer was an instance of A′-movement. Without further assumptions, then, there is no reason for (6) to be ill-formed.

Syntacticians have argued for a long time that the source of ill-formedness is the switch from A′-movement back to A-movement; this is what is commonly referred to as the Ban Against Improper Movement:

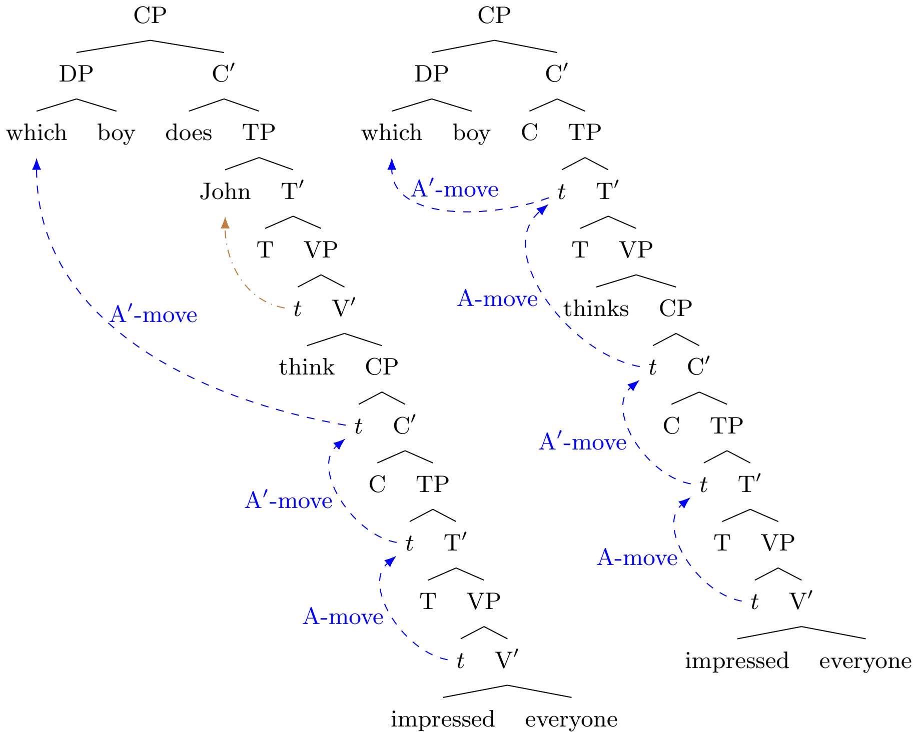


Fig. 3. Minimalist analyses of the licit (5) on the left and the illicit (6) on the right; only the latter intersperses A-movement and A′-movement.

(7) **Ban Against Improper Movement** (standard version)
A phrase that has already undergone A′-movement can no longer undergo A-movement.

Note that the Ban Against Improper Movement allows A-movement to take place after A′-movement as long as it is not the same phrase that undergoes both steps. In (5), for instance, *John* is allowed to participate in A-movement even though *which boy* has already A′-moved. In (6), on the other hand, the very same phrase *which boy* is supposed to A-move after it has already A′-moved. This violates the Ban Against Improper Movement, and hence (6) is ill-formed.

But the Ban Against Improper Movement is a stipulation, it cannot be naturally derived from other syntactic mechanisms (but see [24] for a recent attempt to do so). We can improve on this by reducing the ban to an instance of monotonicity, which is already known to be an important factor in semantics, morphology, and morphosyntax. To this end, let us consider the locality hiearchy $\mathcal{L}$, repeated here with the shorter names used in Fig. 3.

$$\text{Select} < \text{A-Move} < \text{A}'\text{-Move}$$

The Ban Against Improper Movement is, essentially, a requirement that the mapping from a phrase's sequence of operations into $\mathcal{L}$ be monotonic.

Let us look at this in detail. For any given phrase, we may record the sequence of operations it participates in. For example, *which boy* in (5) would have the sequence

$$\text{Select} < \text{A-Move} < \text{A}'\text{-Move} < \text{A}'\text{-Move}$$

while *which boy* in (6) would get the sequence

$$\text{Select} < \text{A-Move} < \text{A}'\text{-Move} < \text{A-Move} < \text{A}'\text{-Move}.$$

Note that we can also view these sequences as mappings from natural numbers into $\mathcal{L}$, where the natural number n denotes the n-th element of the sequence of operations. For example, the sequence for *which boy* in (5) above is equivalent to a mapping with $1 \mapsto$ Select, $2 \mapsto$ A-Move, $3 \mapsto$ A′-Move, and $4 \mapsto$ A′-Move. The Ban Against Improper Movement requires that the sequences, when viewed as such mappings, must obey the order of $\mathcal{L}$.

(8) **Ban Against Improper Movement** (monotonicity version)
Given some phrase p in some syntactic structure t, let f be a function from natural numbers into $\mathcal{L}$ such that f encodes the sequence of operations that applied to p in t. Then f must be monotonically increasing.

The function f for *which boy* in the illicit (6) violates this requirement: clearly $3 < 4$, yet $f(3) = \text{A}'\text{-Move} > \text{A-Move} = f(4)$.

In fact, the monotonicity version of the Ban Against Improper Movement also makes an additional prediction: once a phrase has undergone A-movement or A′-movement, it can no longer participate in selection. This is indeed the case. A phrase that has started moving can no longer select any arguments, nor can it be selected by anything else.[1] Syntacticians treat that as yet another law of syntax, whereas the monotonicity version of the Ban Against Improper Movement already rules out this kind of *Improper Selection*. Not only then can the Ban Against Improper Movement be related to monotonicity, doing so allows us to subsume another important constraint as just another special case.

2.3 Generalized Versions of the Ban Against Improper Movement

The Ban Against Improper Movement has been modified and generalized in several ways, and these generalizations also fit under the umbrella of monotonicity.

Perhaps the best-known generalization is the *Williams Cycle* [36,37]. It starts with the assumption of some linear order of all positions that a phrase can move from or into. In Minimalist syntax, for instance, a simplified version of this hierarchy could be VP $<$ vP $<$ TP $<$ CP (the vP position was skipped

[1] This of course depends on how one analyzes cases such as *John greeted whoever Mary invited*, which was discussed in Sect. 2.1. In addition, there have been proposals in the Minimalist literature that a mover can undergo *Late Merge* with some of its arguments [35].

in all phrase structure trees so far, but I include it here as it will matter in the discussion of case later on). The Williams Cycle then states that a phrase *p* cannot move into a position that is less prominent than the position that *p* currently resides in. For example, if *p* currently resides in CP, then it cannot move into a VP- or TP-position, but it could still move into another CP position. The Williams Cycle thus derives the ungrammaticality of (6) because, as we saw in Fig. 3, the phrase *which boy* moves from a CP position into a TP position. The minimally different (5), on the other hand, is correctly predicted to be well-formed as *which boy* moves from a VP-position to a TP-position, from there to a CP-position, and from there to another CP-position. The Williams Cycle thus constitutes a more fine-grained version of the Ban Against Improper Movement.

It should be readily apparent, though, that the Williams Cycle can be analyzed in exactly the same fashion as the Ban Against Improper Movement. Once again we keep a record of the relevant syntactic steps for each phrase. But now this record is no longer a sequence that lists the relevant operation/dependency (Select, A-Move, A′-Move). Instead, it lists the kind of position that the phrase resided in (VP, TP, CP, and so on). The Williams Cycle requires that this sequence must be a monotonic mapping from natural numbers into the hierarchy VP $<$ TP $<$ CP (or an extended version thereof with additional types of positions). Hence the sequence VP $<$ CP $<$ TP $<$ CP for (6) is forbidden because $f(2) = \text{CP} > f(3) = \text{TP}$ yet $2 < 3$. If anything, the Williams Cycle reveals the monotonic nature of the Ban Against Improper Movement even more clearly.

The Williams Cycle also provides the motivation for a recently proposed *Ban Against Improper Case* [28]. This principle starts with a specific analysis of how noun phrases receive morphological case, known as *Dependent Case Theory* (see [30] for a recent overview and a discussion of structural and lexical case in this theory). Dependent Case Theory posits that the case on one noun phrase can determine the case on another noun phrase. For example, direct objects typically receive accusative because of the nominative case on the subject, and indirect objects receive dative because of the accusative case on the direct object. Intuitively, there is a case hierarchy Nom $<$ Acc $<$ Dat $< \cdots$ and each noun phrase gets the next case that has not yet been claimed by a more prominent noun phrase. However, this kind of dependent case is not unrestricted. It is usually assumed to be clause bounded, so that the subject of the matrix clause cannot cause the subject of an embedded clause to receive accusative. The Ban Against Improper Case takes this idea and refines it in very much the same fashion that the Williams Cycle refines the Ban Against Improper Movement.

(9) **Ban Against Improper Case** (paraphrased from [28])
Assume that there is some ordering $<$ of syntactic positions. Then a noun phrase in position X cannot license dependent case on a noun phrase Y if there is some position Z between X and Y such that $X < Z$.

As a concrete example, consider the following sentence:

(10) $[_{\text{TP}}$ He $[_{v\text{P}}$ told $[_{\text{VP}}$ her $[_{\text{CP}}$ that $[_{\text{TP}}$ it had been stolen$]]]]]$.

English still displays remnants of case in its pronoun system. Here we see that the subjects *he* and *it* carry nominative, whereas the object *her* carries accusative case. The accusative case on the object *her* has to be licensed by the nominative on the subject *he*. Objects reside in VP-positions, and subjects in TP-positions. The only position between the two is vP. If we assume, as before, a hierarchy of the form VP < vP < TP < CP, then the presence of this vP does not violate the Ban Against Improper Case because it is not the case that vP > TP.

Now let us turn to the nominative case on the embedded subject *it*. Given what I said before about Dependent Case Theory, one might expect the accusative on the object *her* to cause *it* to receive dative case. That does not happen because of the Ban Against Improper Case. The subject *it* resides in a TP-position, and the object *her* in a VP-position. Between the two is a CP-position. Since TP < CP, the accusative on *her* cannot affect the case of *it* without triggering a violation of the Ban Against Improper Case. Hence the pronoun *it* appears with nominative case, effectively starting a new chain of dependent case licensing that is separate from whatever happened in the matrix clause.

The astute reader has probably figured out already how the Ban Against Improper Case reduces to monotonicity. For each phrase with licensed case, we look at the path of positions that starts right above said phrase and extends all the way up to its case licensor. When viewed as a mapping from natural numbers into the hierarchy of positions, the mapping must be monotonic. For the example above, the sequence for *her* is vP < TP, which is monotonically increasing. If *it* were to stand in a dependent case relation with *her*, then the corresponding sequence would be CP < VP, which is not monotonically increasing. For the same reason, *Bill* cannot stand in a case relation with *he* either, as this would yield the non-monotonic sequence CP < VP < vP < TP. When applied to such "case licensing paths", monotonicity does exactly the same work as the Ban Against Improper Case.

Overall, then, monotonicity can be regarded as the driving force behind the Ban Against Improper Movement/Williams Cycle, the Ban Against Improper Selection, and the Ban Against Improper Case. The treatment here is far from exhaustive. For example, I have said nothing about how head movement or sidewards movement [26] fit into this picture. Still, this is a promising start, and monotonicity can be pushed even farther.

3 Omnivorous Number

All the cases discussed so far involved a linear hierarchy. But the notion of monotonicity also applies to partial orders, and this, too, finds application in syntax. One concrete example comes from *omnivorous number* [25], to be discussed next (Sect. 3.1). The analysis of omnivorous number will also highlight some important methodological aspects of the monotonicity approach (Sect. 3.2).

3.1 Proposed Analysis

Omnivorous number is a rare phenomenon that only occurs in languages where verbal agreement is contingent on both the subject and the object. In languages with omnivorous number, a transitive verb displays plural agreement unless both its subject and its object are singular. In other words, once at least one argument of the verb is plural, the verb display plural agreement. This is illustrated by the following example from Georgian [25, p. 950].

(11) g- xedav- t
2ndObj- saw- pl

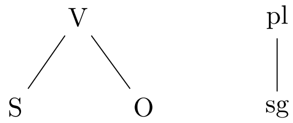
This utterance is highly ambiguous as it could mean "I saw you.pl", "We saw you.sg", and "We saw you.pl", among other options. All of these are potential readings because each one contains at least one plural argument that could be the source of the plural agreement on the verb.

Curiously, no known language displays the opposite system where verbal agreement depends on multiple arguments yet is singular if at least one argument is singular. The absence of this pattern is striking. One major goal of syntactic theories is to allow for the vast range of cross-linguistic variation while providing an explanation as to why some logically conceivable patterns never seem to occur. Ideally, the explanation for these typological gaps is simple and not specific to just a few phenomena. Both desiderata are met by a monotonicity-based analysis of omnivorous number—the analysis is simple, and it treats omnivorous number as yet another expression of a general monotonicity principle that also drives the Ban Against Improper Movement and many other syntactic constraints. As with all the constraints seen in Sect. 2, the monotonicity account of omnivorous number will restrict the mapping from some syntactic ordering to a fixed universal hierarchy. The major innovation of omnivorous number, though, is that the syntactic ordering is no longer linear, but a partial order.

First, let us assume a universal number hierarchy such that sg $<$ pl. This hierarchy is intuitively plausible in the sense that it replicates the ordering of quantities—a plural refers to more entities than a singular. There have been arguments in the literature that plural should be considered a semantic default from which the singular meaning is derived [31], but these do not necessarily conflict with the hierarchy above. These arguments make claims about how one meaning is derived from another, whereas the hierarchy I propose orders singular and plural in terms of their semantic extension. Moreover, we will see at the end of the section that the key insight of the monotonicity account is preserved even if one uses a hierarchy of the form pl $<$ sg.

The hierarchy sg $<$ pl gives us one ordering for monotonicity, but we still have to define a second ordering that represents the syntactic agreement mechanism that produces omnivorous number. Omnivorous number only arises in languages where the verb V agrees with both its subject S and its object O, and we will only consider such languages here (so English, for instance, would require a different model that omits O). Crucially, the number values of V, S, and O are not completely independent of each other. The number value of V depends on its two arguments S and O, but number values of S and O do not depend on each other. We can regard this as a partial order such that $S < V$ and $O < V$, but S and O are unordered with respect to each other.

We thus arrive at the two structures depicted below.



We can now ask what kind of mapping f can be defined from the partially ordered set on the left to the linear order on the right. Under the assumption that f must be total, there are 8 options, which are listed in Table 1. There are only three unattested patterns, all of which involve the verb displaying singular agreement even though at least one of its arguments is plural. These are exactly the cases that are ruled out if the mapping f must be monotonically increasing. Consider, for example, the case where $f(S) = f(V) = \text{sg} < \text{pl} = f(O)$. This contradicts $O < V$ and is hence ruled out. Minor variations of this equation show that the other unattested forms are not monotonic mappings either, whereas the attested patterns are.

Table 1. Potential agreement types in a language where verbs agree with subjects and objects in number

$f(S)$	$f(O)$	$f(V)$	Attested?
sg	sg	sg	yes (uniform agreement)
sg	sg	pl	yes (resolved agreement)
sg	pl	sg	no
sg	pl	pl	yes (omnivorous number)
pl	sg	sg	no
pl	sg	pl	yes (omnivorous number)
pl	pl	sg	no
pl	pl	pl	yes (uniform agreement)

We see then that monotonicity—when combined with intuitively plausible hierarchies that encode, respectively, the relation of singular and plural and how the value of the verb depends on its argument—is fully sufficient to derive the attested typology of verbal agreement systems with two arguments.

3.2 Addressing a Potential Objection

The reader might object that my account relies on two stipulations: I) the function must be monotonically increasing rather than monotonically decreasing, and II) the number hierarchy is sg < pl rather than pl < sg. It is instructive to fully explore this issue as it highlights in what ways the monotonicity approach to syntax can(not) enhance our linguistic understanding.

First, note that assumptions I and II are interlinked. If we alter both, we get exactly the same system because "monotonically increasing" is the dual of "monotonically decreasing", and sg < pl is the dual of pl < sg; the two duals cancel each other out. Suppose, then, that we alter only one of the two. No matter which one of the two assumptions we replace with its dual, we get the predictions in Table 2. These predictions do not line up with the typological landscape. Crucially, we do not just replace omnivorous number with its counterpart, we also predict that resolved agreement is impossible. Resolved agreement occurs when two singular arguments yield a single plural agreement marker, and this behavior is attested. Under the analysis proposed in Sect. 3.1, the existence of resolved agreement predicts the existence of omnivorous number (and the other way round).

Table 2. Predicted typology if either sg < pl or the mapping must be monotonically decreasing

$f(S)$	$f(O)$	$f(V)$	Attested?	Predicted to exist?
sg	sg	sg	yes (uniform agreement)	yes
sg	sg	pl	yes (resolved agreement)	no
sg	pl	sg	no	yes
sg	pl	pl	yes (omnivorous number)	no
pl	sg	sg	no	yes
pl	sg	pl	yes (omnivorous number)	no
pl	pl	sg	no	yes
pl	pl	pl	yes (uniform agreement)	yes

This kind of unification is the principal driver of the monotonicity approach, which otherwise could quickly devolve into arbitrariness. The approach relies on domain-specific hierarchies, but since hierarchies are an abstract encoding of linguistic substance, which is not nearly as well understood as linguistic form, they are necessarily tentative. Each hierarchy has to be motivated by independent considerations, e.g. locality or semantics, among others, but that is a soft constraint at best. However, one and the same hierarchy may affect many different phenomena, and thus linguistic typology acts as a much stronger constraint on the shape of hierarchies. The monotonicity perspective deliberately abstracts away from details of the grammar in order to maximize the impact of typology. If two phenomena revolve around, say, person, then they should both be describable in terms of the same person hierarchy, even if they involve vastly different mechanisms in the grammar. This way, the hierarchies can be put on a firm empirical foundation that minimizes arbitrariness.

We have seen several concrete instances of this principle throughout the paper. The analysis above combines resolved agreement and omnivorous number into a single package: if one can occur in some natural language, the other can occur in some (other) natural language. In the discussion of movement types (Sect. 2.2), the monotonicity analysis of the Ban Against Improper Movement also subsumes a Ban Against Improper Selection, and the Ban Against Improper Case uses the same hierarchy as the Williams Cycle. This is the ideal scenario: a hierarchy that is motivated by independent considerations can be combined with monotonicity to explain not just one specific phenomenon, but an array of phenomena.

4 Why Monotonicity?

By now, the reader is hopefully convinced that a number of syntactic phenomena can be insightfully analyzed in terms of monotonicity. This raises the question, though, why monotonicity should play a role in syntax.

The apparent importance of monotonicity is particularly puzzling because there seems to be no natural way to encode monotonicity in common syntactic formalisms such as Minimalism, HPSG, LFG, or TAG. This paper deliberately analyzed syntax at a high level of abstraction that completely factors out how the relevant orders and properties may be inferred by the syntactic machinery (or how said machinery could give rise to the observed orders). But this is in fact a common strategy in syntax. For example, syntactic accounts of NPI licensing frequently gloss over how syntax determines whether a phrase is an NPI-licensor. Sometimes the issue is sidestepped via lexicalization, e.g. via a specific feature, or by assuming that there is a finite list of NPI-licensors that can be queried by syntax. But this is just one specific way of syntacticizing a more abstract concept. Similarly, there is extensive work on island constraints, yet very little on how one encodes whether a specific phrase is an island or not—attempts to do so often require unusual encoding tricks (cf. [1]). Implementation details can obfuscate more than they illuminate, and syntacticians frequently do not provide formal implementations when there is reason to believe that the implementation would not yield novel insights. I have taken the same stance here with monotonicity, implicitly assuming that the issue of how monotonocity could be recast in terms of syntactic machinery would not help us understand the role of monotonicity in syntax. Seeing how some of the most fundamental aspects of syntax are rarely encoded directly in the syntactic formalism, it is not too troubling that the same holds of monotonicity and the proposed orders and hierarchies.

One should also keep in mind the following: while it is surprising for syntax to be sensitive to monotonicity, it would be even more surprising if syntax did not care about monotonicity at all. Monotonicity is already a major factor in semantics, and the work that this paper builds on suggests that monotonicity matters in morphology, too [11]. In addition, linguists have often noted the importance of structure-preservation principles, which can be regarded as an instance of monotonicity. And finally, work on grammatical inference points towards monotonicity greatly simplifying the learning problem (see [17]). Monotonicity has a role to play in many aspects of language, and it would be surprising for syntax to be exempt from that.

In the future, it will be interesting to see if broadening the scope of research on monotonicity from semantics to all linguistic domains yields a unifying cause for the prevalence of monotonicity in language. The answer may lie in learnability and grammatical inference, but I conjecture that computational complexity is also an important factor. The work that this paper builds on [11] grew out of [9], where typological gaps are explained in terms of how specific linguistic graph structures can and cannot be rewritten if the rewriting mechanism must fit a particular notion of *subregular complexity*. Subregular linguistics is concerned with the application of very restricted subclasses of finite-state machinery to natural language. There has been a flurry of promising results in computational phonology, morphology, syntax, and even semantics (see, among others, [2,4,6,10,12–16,18,23,32]). Monotonicity might be an elegant approximation of a more fine-grained, but also less intuitive notion of subregular complexity.

Syntax is the ideal candidate for probing the connection between monotonicity and computation. Monotonicity has been studied most extensively with respect to semantics, but this paper and related work show that morphology and syntax also seem to be exquisitely sensitive to monotonicity. Between morphology and syntax, the latter has seen a lot more work on its subregular complexity. Consequently, syntax is the only area of language right now that provides a fertile ground for both monotonicity and subregular complexity. If there is some connection between monotonicity and subregular complexity, some computational driver towards monotonicity, it should be easier to find in syntax than in phonology, morphology, or semantics.

5 Conclusion

I have presented several syntactic phenomena that can be analyzed in terms of monotonicity: the Ban Against Improper Movement, the Williams Cycle, the Ban Against Improper Case, and omnivorous number. Due to space constraints, many others had to be omitted, such as the Keenan-Comrie hierarchy [21]. There is also a plethora of work on 3/4-splits in typology, where only 3 out of 4 conceivable options ever show up in natural language. These can be regarded as monotonic maps from an order with two elements into another order with two elements. In addition, existing work such as the algebraic account of adjunct islands in [8] implicitly use monotonicity. A large number of seemingly unrelated phenomena thus fall under the purview of the same universal principle. They all can be explained in terms of monotonic mappings from some kind of abstract syntactic representation to a universal hierarchy.

That said, the work reported here is but a starting point. The posited hierarchies require a more rigorous and insightful motivation, and it will be important to also identify phenomena that do not obey monotonicity. This will give us a deeper understanding of the place of monotonicity in natural language, and may ultimately answer the question why any aspect of language, be it semantics, syntax, or something else, should care about monotonicity in the first place.

Acknowledgments. The work reported in this paper was supported by the National Science Foundation under Grant No. BCS-1845344. The paper benefited tremendously from discussion with Aniello De Santo, Alëna Aksënova, Ayla Karakas, Sedigheh Moradi, and Nazila Shafiei, as well as the detailed feedback of two anonymous reviewers.

References

1. Abels, K.: Successive cyclicity, anti-locality, and adposition stranding. Ph.D. thesis, University of Connecticut (2003)
2. Aksënova, A., Graf, T., Moradi, S.: Morphotactics as tier-based strictly local dependencies. In: Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pp. 121–130 (2016). https://www.aclweb.org/anthology/W/W16/W16-2019.pdf
3. Bobaljik, J.D.: Universals in Comparative Morphology: Suppletion, Superlatives, and the Structure of Words. MIT Press, Cambridge (2012)
4. Chandlee, J., Heinz, J.: Strict locality and phonological maps. Linguistic Inquiry **49**, 23–60 (2018)
5. Chomsky, N.: The Minimalist Program. MIT Press, Cambridge (1995)
6. De Santo, A., Graf, T.: Structure sensitive tier projection: applications and formal properties. In: Bernardi, R., Kobele, G., Pogodalla, S. (eds.) Formal Grammar, pp. 35–50. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-662-59648-7_3
7. Gazdar, G., Klein, E., Pullum, G.K., Sag, I.A.: Generalized Phrase Structure Grammar. Blackwell, Oxford (1985)
8. Graf, T.: The syntactic algebra of adjuncts. In: Proceedings of CLS 49 (2013, to appear)
9. Graf, T.: Graph transductions and typological gaps in morphological paradigms. In: Proceedings of the 15th Meeting on the Mathematics of Language (MOL 2017), pp. 114–126 (2017). http://www.aclweb.org/anthology/W17-3411
10. Graf, T.: Why movement comes for free once you have adjunction. In: Edmiston, D., et al. (eds.) Proceedings of CLS 53, pp. 117–136 (2018)
11. Graf, T.: Monotonicity as an effective theory of morphosyntactic variation. J. Lang. Modelling **7**, 3–47 (2019)
12. Graf, T.: A subregular bound on the complexity of lexical quantifiers. In: Schlöder, J.J., McHugh, D., Roelofsen, F. (eds.) Proceedings of the 22nd Amsterdam Colloquium, pp. 455–464 (2019)
13. Graf, T.: Curbing feature coding: strictly local feature assignment. Proc. Soc. Comput. Linguist. (SCiL) **2020**, 362–371 (2020)
14. Graf, T., De Santo, A.: Sensing tree automata as a model of syntactic dependencies. In: Proceedings of the 16th Meeting on the Mathematics of Language, pp. 12–26. Association for Computational Linguistics, Toronto, Canada (2019). https://www.aclweb.org/anthology/W19-5702
15. Heinz, J.: The computational nature of phonological generalizations. In: Hyman, L., Plank, F. (eds.) Phonological Typology, Chap. 5, pp. 126–195. Phonetics and Phonology, Mouton De Gruyter (2018)
16. Heinz, J., Idsardi, W.: What complexity differences reveal about domains in language. Topics Cogn. Sci. **5**(1), 111–131 (2013)
17. Heinz, J., Kasprzik, A., Kötzing, T.: Learning in the limit with lattice-structured hypothesis spaces. Theoret. Comput. Sci. **457**, 111–127 (2012). https://doi.org/10.1016/j.tcs.2012.07.017

18. Jardine, A.: Computationally, tone is different. Phonology **33**, 247–283 (2016). https://doi.org/10.1017/S0952675716000129
19. Joshi, A.: Tree-adjoining grammars: How much context sensitivity is required to provide reasonable structural descriptions? In: Dowty, D., Karttunen, L., Zwicky, A. (eds.) Natural Language Parsing, pp. 206–250. Cambridge University Press, Cambridge (1985)
20. Joshi, A., Schabes, Y.: Tree-adjoining grammars. In: Rosenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, pp. 69–123. Springer, Berlin (1997). https://doi.org/10.1007/978-3-642-59126-6_2
21. Keenan, E.L., Comrie, B.: Noun phrase accessiblity and universal grammar. Linguistic Inquiry **8**, 63–99 (1977)
22. Kobele, G.M.: A formal foundation for A and A-bar movement. In: Ebert, C., Jäger, G., Michaelis, J. (eds.) The Mathematics of Language. Lecture Notes in Computer Science, vol. 6149, pp. 145–159. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14322-9_12
23. McMullin, K.: Tier-based locality in long-distance phonotactics: learnability and typology. Ph.D. thesis, University of British Columbia (2016)
24. Müller, G.: A local reformulation of the Williams cycle. In: Heck, F., Assmann, A. (eds.) Rule Interaction in Grammar, Linguistische Arbeitsberichte, vol. 90, pp. 247–299 (2013)
25. Nevins, A.: Multiple agree with clitics: person complementarity vs. omnivorous number. Nat. Lang. Linguistic Theory **28**, 939–971 (2011). https://doi.org/10.1007/s11049-006-9017-2
26. Nunes, J.: Linearization of Chains and Sideward Movement. MIT Press, Cambridge (2004)
27. Pollard, C., Sag, I.: Head-Driven Phrase Structure Grammar. CSLI and The University of Chicago Press, Stanford and Chicago (1994)
28. Poole, E.: Improper case (2020). https://ling.auf.net/lingbuzz/004148
29. Preminger, O.: Phi features, binding, and A-positions (2018). https://facultyoflanguage.blogspot.com/2018/01/phi-features-binding-and-positions.html, blog post on Faculty of Language
30. Puškar, Z., Müller, G.: Unifying structural and lexical case assignment in dependent case theory. In: Lenertová, D., Meyer, R., Šimík, R., Szucsich, L. (eds.) Advances in Formal Slavic Linguistic 2016, pp. 357–379 (2018)
31. Sauerland, U.: A new semantics for number. In: Youn, R.B., Zhou, Y. (eds.) SALT 13. CLC Publications, Ithaca (2003)
32. Shafiei, N., Graf, T.: The subregular complexity of syntactic islands. Proc. Soc. Comput. Linguist. (SCiL) **2020**, 272–281 (2020)
33. Stabler, E.P.: Derivational Minimalism. In: Retoré, C. (ed.) Logical Aspects of Computational Linguistics, Lecture Notes in Computer Science, vol. 1328, pp. 68–95. Springer, Berlin (1997). https://doi.org/10.1007/BFb0052152
34. Stabler, E.P.: Computational perspectives on Minimalism. In: Boeckx, C. (ed.) Oxford Handbook of Linguistic Minimalism, pp. 617–643. Oxford University Press, Oxford (2011)
35. Takahashi, S., Hulsey, S.: Wholesale late merger: beyond the A/$\overline{\text{A}}$ distinction. Linguistc Inquiry **40**, 387–426 (2009)
36. Williams, E.: Rule ordering in syntax. Ph.D. thesis, MIT, Cambridge (1974)
37. Williams, E.: Representation Theory. MIT Press, Cambridge (2003)