AMERICAN PSYCHOLOGICAL ASSOCIATION

Psychological Review

ISSN: 0033-295X

2023, Vol. 130, No. 4, 935–952
https://doi.org/10.1037/rev0000429

Hierarchical Structure in Language and Action: A Formal Comparison

Cas W. Coopmans[1, 2], Karthikeya Kaushik[1, 3], and Andrea E. Martin[1, 3]
[1] Max Planck Institute for Psycholinguistics
[2] Centre for Language Studies, Radboud University, Nijmegen
[3] Donders Institute for Brain, Cognition and Behaviour, Radboud University, Nijmegen

Since the cognitive revolution, language and action have been compared as cognitive systems, with cross-domain convergent views recently gaining renewed interest in biology, neuroscience, and cognitive science. Language and action are both combinatorial systems whose mode of combination has been argued to be hierarchical, combining elements into constituents of increasingly larger size. This structural similarity has led to the suggestion that they rely on shared cognitive and neural resources. In this article, we compare the conceptual and formal properties of hierarchy in language and action using set theory. We show that the strong compositionality of language requires a particular formalism, a magma, to describe the algebraic structure corresponding to the set of hierarchical structures underlying sentences. When this formalism is applied to actions, it appears to be both too strong and too weak. To overcome these limitations, which are related to the weak compositionality and sequential nature of action structures, we formalize the algebraic structure corresponding to the set of actions as a trace monoid. We aim to capture the different system properties of language and action in terms of the distinction between hierarchical sets and hierarchical sequences and discuss the implications for the way both systems could be represented in the brain.

Keywords: hierarchy, syntax, compositionality, formal modeling, set theory

It has long been recognized that both language and action are structurally organized in a way that is not immediately evident from their serial appearance. In the 1950s, Lashley (1951) and Chomsky (1959) separately showed that then dominant behaviorist "chaining" theories based on contiguous stimulus–response associations could not account for serial behavior, such as language production and action execution. Instead, these behaviors appear to be controlled by internal, hierarchically organized plans, which allow human behavior to be creative, productive, and flexible. Since then, similarities between language and action have often been noted (Greenfield, 1991; Holloway, 1969; Miller et al., 1960), and more recent studies propose that the two systems are analogous in their hierarchical organization (Fitch & Martins, 2014; Fujita, 2014; Jackendoff, 2007; Pulvermüller & Fadiga, 2010; Stout & Chaminade, 2009).

Such proposals about cross-domain convergence are desirable from an evolutionary perspective, in which one seeks to find a set of primitives that account for the distinguishing features of the human mind (Boeckx & Fujita, 2014; de Waal & Ferrari, 2010; Hauser et al., 2002; Marcus, 2006). However, arguments in favor of the analogy between language and action are formally underspecified. It is possible to draw a hierarchical tree structure over any sequence, but what is needed is independent empirical evidence that this structure explains or describes a phenomenon in the natural world (Berwick & Chomsky, 2017; Bloom, 1994; Fitch & Martins, 2014; Moro, 2014a). In other words, superficial resemblance is insufficient:

> we cannot just observe that hierarchical structures are found in motor control (e.g., tool construction), and thereby claim that these are directly related to the hierarchical structures of language … Rather, it is necessary to develop a functional description of the cognitive structures in question, parallel to that for language … so we can look for finer-scale commonalities. (Jackendoff, 2002, p. 80)

While formal linguistics has provided many accounts of the specific properties of hierarchy in language, such a formal characterization in the domain of actions and action plans is lacking (but see Steedman, 2002, for an exception). To this end, the aim of this article is to characterize the similarities and differences between the hierarchical structures in language and action in both conceptual and formal terms. The article is structured as follows: In Hierarchical Structure in Language section, we discuss the type of data that shows that the syntax of natural languages is organized hierarchically, after which we list the core properties of such hierarchical

This article was published Online First May 11, 2023.
Cas W. Coopmans https://orcid.org/0000-0001-7622-3161
Karthikeya Kaushik https://orcid.org/0000-0001-8501-4134
Andrea E. Martin https://orcid.org/0000-0002-3395-7234
The authors thank Giosuè Baggio, Cedric Boeckx, Helen de Hoop, and Bob van Tiel for their helpful comments on earlier versions of this work. They also thank Steven Phillips for his help with aspects of the formal work. For annotating the equations in the formal work, the authors used the code from https://github.com/synercys/annotated_latex_equations. Andrea E. Martin was supported by a Max Planck Research Group and a Lisa Meitner Research Group "Language and Computation in Neural Systems" from the Max Planck Society and by the Netherlands Organization for Scientific Research (Grant 016.Vidi.188.029).

This work has been presented at the Joint Conference on Language Evolution in Kanazawa, Japan, in 2022. A preprint of this article has been posted on PsyArXiv (https://osf.io/x59un).

Correspondence concerning this article should be addressed to Cas W. Coopmans, Max Planck Institute for Psycholinguistics, Wundtlaan 1, 6525 XD Nijmegen, The Netherlands. Email: cas.coopmans@mpi.nl

syntactic structure (Properties of Syntactic Structure section). In Formalizing Linguistic Structure section, we formally describe these structures in a domain-neutral way using the mathematical language of set theory. We then show that this formalism is inadequate for describing the action system (Formalizing Action Structure (1) section) and suggest an alternative formalism to characterize its properties (Formalizing Action Structure (2) section). In Language Versus Action section, we conclude that the properties of syntactic hierarchy are not found in action structures (The Nature of Structure section) and discuss this conclusion in light of the idea that syntactic representations are fundamentally hierarchical sets, while actions are better conceived of as hierarchical sequences (Levels of Abstraction section). We end by discussing the implications for how language and action might be represented in the brain.

Hierarchical Structure in Language

In linguistics, the term hierarchy refers to the format of linguistic representations. At all levels of organization (phrases, words, and syllables), linguistic structure is organized hierarchically (see Everaert et al., 2015, for a recent overview). In the domain of syntax specifically, it refers to the fact that words are embedded into constituents, which are in turn recursively embedded into larger constituents, creating the hierarchically organized syntactic structures that are often visually denoted by means of tree structures. These tree structures are graphic representations of relations that are essentially set-theoretic (Lasnik, 2000).

A main source of evidence for constituency is the observation that the interpretation of phrases and sentences is often determined by structural relationships. For example, the sentence "the woman saw the man with binoculars" has two meanings. Either the woman has binoculars, which she uses to look at the man, or the man has binoculars. The sentence is ambiguous because it corresponds to two possible structures, which differ in terms of the attachment site of the prepositional phrase (PP) "with binoculars" (see Figure 1). If it attaches to "the man," forming a complex noun phrase (NP) constituent (Figure 1A), the man has the binoculars, but if it attaches to the verb phrase (VP) "saw the man" (Figure 1B), the woman must be holding the binoculars. Here, it is the structural relationship between the PP and the other constituents that determines how the sentence is interpreted.

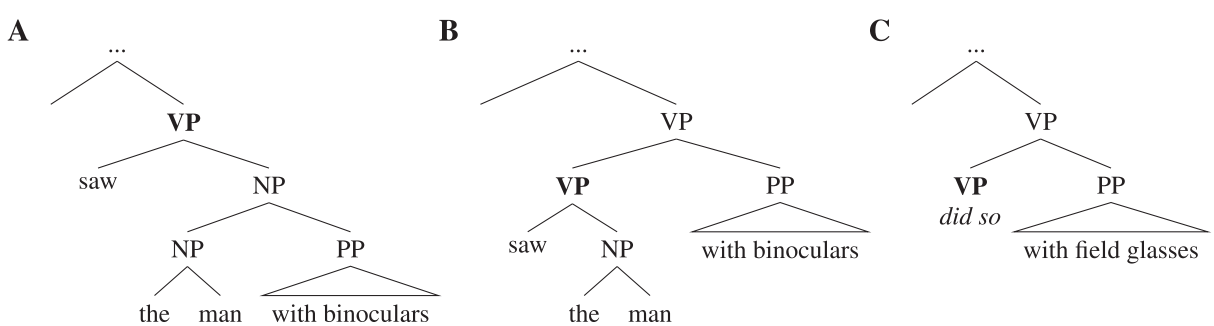
The structure dependence of meaning shows that language is compositional. To be able to compare combinatorial systems, such as language and action, we make a distinction between strong and weak compositionality (Pagin & Westerståhl, 2010). In a strongly compositional system, the meaning of a constructed unit is a function of the meanings of its constituents and the way in which these are structurally combined (Partee, 1995; Partee et al., 1993). In a weakly compositional system, instead, the meaning of a constructed unit is a function of the meaning of the elements and the total construction (i.e., the result of an operation applied over the total construction of ordered elements; Pagin & Westerståhl, 2010). A weakly compositional system can thus distinguish the meanings of "John likes Mary" and "Mary likes John" because their total constructions differ. However, weakly compositional systems cannot capture structural ambiguity. Because they do not take into account the structural relationships between intermediate representations, such as between the different constituents in Figure 1, they are unable to distinguish the two interpretations of "the woman saw the man with binoculars."

A second source of evidence for constituent structure is that syntactic operations, such as deletion and substitution, target constituents rather than words or mere word sequences. For instance, the phrase *did so* can substitute for a verbal word sequence, such as "saw the man," if this sequence forms a constituent. Because the words "saw the man" form an isolated constituent only in the structure of Figure 1B, the sentence "the woman saw the man with binoculars and the boy *did so* with field glasses" (corresponding to Figure 1C) can only mean that the boy is holding the field glasses (analogous to the interpretation of Figure 1B), not the man. In sum, both semantic interpretation and syntactic operations are structure-dependent: They refer to hierarchical constituent structures rather than to linear sequences of words, with the result that word sequences that do not form constituents are not available to semantic interpretation nor to syntactic operations.

Properties of Syntactic Structure

To generate such hierarchical structure, (any theory of) the language faculty must include, at a minimum, a computational procedure for combining smaller elements into larger elements. The properties of this procedure are debated, but all linguistic frameworks assume it in one form or another: Merge in the Minimalist Program (Chomsky, 1995b), Unify in the Parallel Architecture (Jackendoff, 2002), Forward/Backward Application in Combinatory

Figure 1
Hierarchical Syntactic Structures



Note. The structures correspond to the sentences "(the woman) saw the man with binoculars" (A and B) and "(the boy) did so with field glasses" (C). PP = prepositional phrase; VP = verb phrase; NP = noun phrase.

Categorial Grammar (Steedman, 2000), and Substitution in Tree-Adjoining Grammar (Joshi & Schabes, 1997). For our purposes, it is important that the properties of this combinatorial operator are formally defined, and that the operator is computationally general enough so that it could play a role in cognitive domains beyond language. Merge is one of the operators that meets these requirements, as it is formally defined as binary set formation: Merge(α, β) takes two elements α and β and forms the unordered set {α, β} (Chomsky, 2013; Collins, 2017). It can be applied recursively, such that it takes its own output as input: further combining the already formed set {α, β} with γ yields the set {{α, β}, γ}. As should be clear, recursive application of this combinatorial operation yields a structure that is hierarchical: The smaller set is contained in the larger set. Because the generated set is unordered (i.e., {{α, β}, γ} is identical to {γ, {β, α}}), the elements in the set cannot be described in terms of linear precedence. Rather, the relevant relationships are established with respect to structure: The element γ is higher in the structure and has a structurally more prominent position than the elements α and β.

In the remainder of this article, we will assume that the combinatorial procedure for generating syntactic structure is binary set formation. On this assumption, the hierarchical structure of syntax has the following properties: Self-similar, endocentric, and unordered.[1]

Self-Similar

Human language use is creative: Language users can produce and understand sentences that have never been produced before. Specifying such an open-ended capacity using finite means requires recursive procedures, such as the combinatorial operation defined above. While this operation both generates hierarchical structure and applies recursively, hierarchy and recursion are two independent properties. Hierarchy is a property of the output generated by the combinatorial operation (i.e., a property of its extension). Recursion, instead, is a property of a function defined in intension. A recursive function is a function that can apply indefinitely to its own output, leading to structurally self-similar output in which a unit of a specific type is contained in another unit of the same type (in linguistics, this is called self-embedding: The embedding of one thing into another thing *of the same kind*). Recursively generated hierarchy will therefore often display similar properties across different levels of embedding, as can be seen in the repetition of complement clauses like "he said that she believes that he thought …," which is a sentence within a sentence within a sentence. Note, however, that because a recursive function is defined in intension rather than in extension, the recursivity of a function should not be equated with its output. Absence of self-similar output therefore does not warrant the conclusion that the function generating the output is not recursive (Hauser et al., 2014; Watumull et al., 2014).

The independence of hierarchy and recursion is further illustrated by the fact that they doubly dissociate: Not all hierarchical objects are generated by recursion and not all recursive functions generate hierarchical structure. For instance, artificial grammars that generate sequences of the type $(ab)^n$ and a^nb^n can be recursive, via respectively f: $S \rightarrow abS$ and f: $S \rightarrow aSb$, but only the latter generates hierarchical structure.[2] Conversely, the syllable structure in phonology is hierarchical but not recursive. A syllable contains an onset and a rhyme, with the latter consisting of a nucleus and a coda. This hierarchy is not recursive: A syllable cannot be embedded in another syllable.

Endocentric

The categorial status of a constituent is determined by one of its elements (the "head"): The set {α, β} can be of type α or β but not of type γ. Endocentric structures are contrasted with exocentric structures, in which the label of a composed unit is not determined by one of its elements.[3] Labels allow phrases to be called upon by interpretive and formal procedures, thereby determining their distributional behavior. To give an example, the set {eat, cookies} is a verb phrase, which has "eat-like" (interpretive) semantic properties and "verb-like" (formal) syntactic properties, both inherited from the verb "eat." That this is the case can be seen by the fact that "eat cookies" can take the place of the verb "eat" in "he likes to eat," yielding "he likes to eat cookies." It cannot, however, take the place of the noun "cookies" in "he likes chocolate cookies," as is clear from the ill-formedness of "he likes chocolate eat cookies." The label of a composed unit thus places a constraint on further computation, restricting the elements with which it can combine: given that {eat, cookies} is a VP and not a NP, it can combine with adverbs but not with adjectives.

Endocentricity is intricately linked to recursivity, because the combinatorial operation can only be said to apply recursively if its output is of the same type as its input (Boeckx, 2009; Hornstein, 2009; Watumull et al., 2014). Similar to recursivity, endocentricity is a distinctive property of syntactic hierarchy, as not all linguistic structures are endocentric.

Unordered

Because the combinatorial operation is defined as binary set formation, no order is imposed on the members of the combined set. While the unordered structure has to be linearized for spoken language production, differences in linear order do not feed differences in semantic interpretation and syntactic operations do not refer to linear order. Different languages (and different modalities) can seem highly different in terms of the linear ordering of their words (e.g., whether heads precede or follow their dependents), which is a fundamental source of cross-linguistic variation. In terms of the compositional

[1] More properties of language can be derived from the minimal assumption that the structure-building procedure is binary set formation (see Hornstein, 2017; Rizzi, 2013, for comprehensive lists of properties). However, many of these properties, such as displacement, do not have clear analogues in actions (Moro, 2015; Pulvermüller, 2014). Because our aim is to compare the (formal) properties that language and actions might share, we focus on the properties of hierarchy listed here.

[2] The grammars that generate $(ab)^n$ and a^nb^n sequences *can be* implemented recursively, though they do not have to be. These sequences can also be generated with iterative functions that are not recursive, that is, do not call themselves (Fitch, 2010; Jackendoff, 2011).

[3] How it is determined which element defines the label of the phrase is still a much-debated question and is outside the scope of this article (see Boeckx, 2009; Chomsky, 2013; Fukui, 2011). What is important here is not how phrases get their labels but that they get them from one of their elements. Moreover, by using the term labels, we only refer to the fact that the combined unit is of the same type as one of its elements. Whether these labels reflect phrasal projections from the syntactic category of a lexical item (as in *X*-bar theory; Jackendoff, 1977) or rather the lexical item itself (as in bare phrase structure; Chomsky, 1995a) is not critical for our purposes.

properties of the hierarchical structure generated by Merge, however, these languages show consistent similarities.

Note that the assumption about unorderedness is specific to the definition of Merge as binary set formation and might not be shared in other linguistic frameworks.[4] What these frameworks do agree on, however, is that syntactic operations are structure-dependent, not order-dependent.

Defining structure building as binary set formation allows us to derive both compositionality and structure dependence. First, the structure of the input to the combinatorial operation is preserved in its output. Thus, if α and β are constituents (or sets) in the input, they are constituents (or sets) in the output as well: New elements can only be added on top of the already formed set, not inside it. Because the structure of every combination is retained at each level of the hierarchy, the hierarchical structure is strongly compositional. This can be shown with a structurally ambiguous phrase: {deep, {blue, sea}} is not the same as {{deep, blue}, sea}. Note that if the structure were not retained after recursive combination, it would be possible to derive from {blue, sea} not only {deep, {blue, sea}} but also {{deep, blue}, sea}. That would make it impossible to account for the ambiguity of the phrase.

Moreover, recursively generated sets describe hierarchical relations but not sequential relations. Therefore, syntactic operations that refer to these sets can only refer to their structure, and hence be structure-dependent, but not to their sequential order. Rules referring to a word's linear (ordinal) position are also ruled out by recursion: Because it is always possible to recursively insert material *between* two items and thereby change the linear position of the words (e.g., "the boy swims" → "the boy *with muscular arms* swims"), no operation can refer to the linear position of elements in a sequence.

We should note that the properties we described above are properties of a cognitive capacity, which can be expressed in varying degrees in natural languages (e.g., exocentricity might be found in certain subject–predicate relations). Moreover, the faculty of language is capable of assigning strongly compositional interpretations to most sentences, but it can assign other interpretations as well (e.g., to nondecomposable idioms; Baggio, 2021; Jackendoff, 2002). In other words, we listed properties that a model of (the faculty of) language must have, even though these need not be found in all constructions in all languages.

Formalizing Linguistic Structure

In order to be able to evaluate the similarities and differences between the hierarchical structure of language and action in a transparent way, we need a theory-neutral conceptual vocabulary to describe these structures. Ideally, this description should be accompanied by a formal analysis of the similarities and differences, as well as an evaluation of their implications (Guest & Martin, 2021; Martin, 2016, 2020; O'Donnell et al., 2005; Partee et al., 1993; van Rooij & Blokpoel, 2020). To this end, the following paragraphs will present a formal model in which we incorporate the properties of syntactic structure as defined in Properties of Syntactic Structure section.

Generating Structures

Definition 1. $(M, \oplus, \oslash)$ is a unital, commutative magma generated from W, where:

1. W is the set of words that represents the lexicon of a language.
2. M is a set of elements that are generated from W, with $W \subset M$.
3. $\oplus$ is a binary set formation operation, such that for $\forall a, b \in M$, $a \oplus b = \{a, b\} = b \oplus a \in M$. Additionally, $\oplus$ is nonassociative, so $\forall a, b, c \in M$, $(a \oplus b) \oplus c \neq a \oplus (b \oplus c)$.
4. $\oslash$ is the identity element, such that $\forall m \in M$, $m \oplus \oslash = m = \oslash \oplus m$.

A unital, commutative magma (henceforth referred to as a magma for conciseness) is an algebraic structure (see Box 1), whose operation we define as binary set formation following the formal definition of Merge described in Properties of Syntactic Structure section. This allows us to derive a number of important properties. First, as the magma axiom states that for any two members $a, b \in M$, application of this operator to a and b generates a member of M, thus yielding unbounded generation. Second, $\oplus$ does not introduce labels, so the label of each set is derived from one of its elements (i.e., endocentricity; see Chomsky, 2013; Collins, 2017).[5] Third, all elements in M are unordered sets. And fourth, $\oplus$ is nonassociative, which means that the order in which it is applied affects the structure that is generated (Fukui & Zushi, 2004). In other words, the structures that are generated are strongly compositional: Their meaning is a function of the meanings of their parts and the way in which they are structurally combined.

Without further constraints, a freely generated magma would contain elements that should not be constituents, such as {{*eat*}, {*happy*}}. To avoid this without modifying the formal properties of $\oplus$, the lexical items themselves must determine which combinations are licensed and which are not. That is, the application of $\oplus$ is constrained by selectional restrictions on its input (i.e., which categories can(not) combine with which other categories). For instance, {{*eat*}, {*happy*}} is excluded because verbs do not combine with adjectives. The same restrictions apply when the output of $\oplus$ is recursively used as its input. For example, the set $\{_V\{eat\}, \{cookies\}\}$ cannot combine with the adjective "happy" because the former is labeled as a type of verb rather than as a type of noun. Such illegitimate combinations are excluded by taking the (grammatically licensed) subset of the freely generated magma.

We make the relationship between these constituent structures explicit by defining a binary relationship between the elements of the magma, turning it into a partially ordered magma (see Box 2).

Definition 2. $(M, \oplus, \oslash, \leq)$ is a partially ordered magma, where $\leq$ is a *containment* relationship between the elements in M that is reflexive, transitive, and antisymmetric.

The relation $\leq$ on the set M reflects containment or set inclusion, which corresponds to the dominance relation commonly used in linguistics. Thus, $x_1 \leq x_2$ means that x_2 contains (and thus dominates) x_1. As a visualization of this partially ordered magma, consider the Hasse diagram

[4] See Saito and Fukui (1998) and Kayne (2011), who argue that Merge (α, β) forms the ordered pair ⟨α, β⟩. This makes *immediate* precedence part of syntax.

[5] Labels are a convenient way to group together structures with identical formal properties. In our formal setup, constituent labels are simply part labels whose union produces the set of all grammatical structures. For example, with $W = \{dog, man, big\}$, the label N would be the part $\{man, dog\}$, A is $\{big\}$, and NP is $\{\{big, dog\}, \{big, man\}\}$. Therefore, $M = M_N \bigcup M_A \bigcup M_{NP}$.

Box 1
Algebraic Structures

An algebraic structure consists of a nonempty set X (called the carrier set), a collection of finitary operations on X (typically binary operations), and a finite set of axioms that these operations must satisfy. To illustrate the relevant axioms for the current work, we consider X as the carrier set and $\odot$ as a binary operation acting on the elements of X.

1. $\forall x_1, x_2 \in X, x_1 \odot x_2 \in X$	*closed*
2. $\forall x \in X, \exists! i \in X$ such that $x \odot i = i \odot x = x$	*unital*
3. $\forall x_1, x_2, x_3 \in X, (x_1 \odot x_2) \odot x_3 = x_1 \odot (x_2 \odot x_3)$	*associative*
4. $\forall x_1, x_2 \in X, x_1 \odot x_2 = x_2 \odot x_1$	*commutative*

Depending upon the axioms they satisfy, the algebraic structures form a taxonomy. Presented below is a subset of this taxonomy, in which we highlight both the algebraic structures that are relevant for the current work as well as their corresponding axioms.

Magma	**Unital commutative magma**	**Monoid**	**Commutative monoid**	**Trace monoid**
closed	*closed*	*closed*	*closed*	*closed*
	unital	*unital*	*unital*	*unital*
	commutative	*associative*	*associative*	*associative*
			commutative	*partially commutative*

Note. The algebraic structures are indicated in bold. The axioms are indicated in italics.

in Figure 2, which displays the containment relationship for two structures that map onto the sequence "woman saw man with binoculars."

Besides containment, there is another relevant structural relation between the elements in constituent structures. This relationship, called c-command (Reinhart, 1983), describes the scope domain of a node in the tree structure. Specifically, a node α is said to asymmetrically c-command a node β iff β is contained in the sister node of α (e.g., in Figure 1A "saw" asymmetrically c-commands every node contained in the higher NP).

Definition 3. For $m_1, m_2 \in M$, m_1 c-commands m_2 (denoted $m_1 \gg m_2$) if $m_1 \nleq m_2$, and $m_2 \nleq m_1$, and $\exists! m = \{m_1, x\} \in M$, and $m_2 \leq x$. An *asymmetric c-command* relationship exists between m_1 and m_2 if $m_1 \gg m_2$ and $m_2 \not\gg m_1$. Asymmetric c-command is irreflexive, transitive, antisymmetric, and locally total.

Given Definition 3, asymmetric c-command is a locally total relation on nonterminal nodes in the tree structure.[6] The Hasse diagram in Figure 3 visualizes the asymmetric c-command relationship for the two structures that map onto the sequence "woman saw man with binoculars."

Sequences

Definition 4. (S, $*$, ‘’) is a monoid generated from W, where:

1. W is the set of words that represents the lexicon of a language.
2. S is the set of sequences generated from W, with $W \subset S$.
3. $*$ is the concatenation operation, which is unital and associative.
4. The empty sequence ‘’ is the identity element.

Definition 5. We define a binary relation ($\prec$) on the elements in $s = (x_1, x_2, \ldots x_n) \in S$, which we call *precedence*, where $x_1 \prec x_2 \prec \ldots x_n$. Precedence is irreflexive, transitive, antisymmetric, and locally total.

Given Definition 5, precedence is a locally total relation on the set of elements in a sequence (i.e., corresponding to the terminal nodes in the tree structure). The Hasse diagram in Figure 4 visualizes the precedence relationship for the sequence "woman saw man with binoculars."

Mapping Structures to Sequences

Following Kayne (1994), we assume that there exists a rigid mapping between hierarchical structure and linear order, such that only one linear sequence can be derived from a given hierarchical structure. As noted above, asymmetric c-command and precedence are locally total orders on the set of nonterminals and the set of terminals, respectively. Kayne (1994) formalizes the mapping between these two orders in the Linear Correspondence Axiom (LCA).

Linear Correspondence Axiom. A lexical item α precedes a lexical item β iff

1. α asymmetrically c-commands β or
2. an XP dominating α asymmetrically c-commands β

[6] Strictly speaking, the relation is left-locally total (Kayne, 1994). A left-locally total relation is total only on the elements to the left of the relation (i.e., for aRb, R is left-locally total for a).

Box 2
Ordered Sets

An ordered set X is a set ordered by a binary relation, denoted here with infix notation $\leq$, such that $\forall x, y, z \in X$, the following axioms may hold (depending on the kind of order):

1. $x \leq x$;	*reflexive*
2. if $x \leq y$ and $y \leq z$, then $x \leq z$;	*transitive*
3. if $x \leq y$ and $y \leq x$, then $x = y$;	*antisymmetric*
4. $x \leq y$ or $y \leq x$	*total*

When the binary relation is transitive and antisymmetric, the set is called partially ordered. A totally ordered set is an ordered set whose binary relation holds between all elements of the set. When a relationship is only total when restricted to X', which is a subset of X, we consider X' locally total (Kayne, 1994). We therefore say that $\forall x, y \in X' \subset X$, $x \leq y$ or $y \leq x$.

Definition 6. We adopt the LCA as a surjective function f: $M \rightarrow S$, defining f for a pair of lexical items $\alpha, \beta \in m \in M$, which holds for all elements of the sequence by induction:

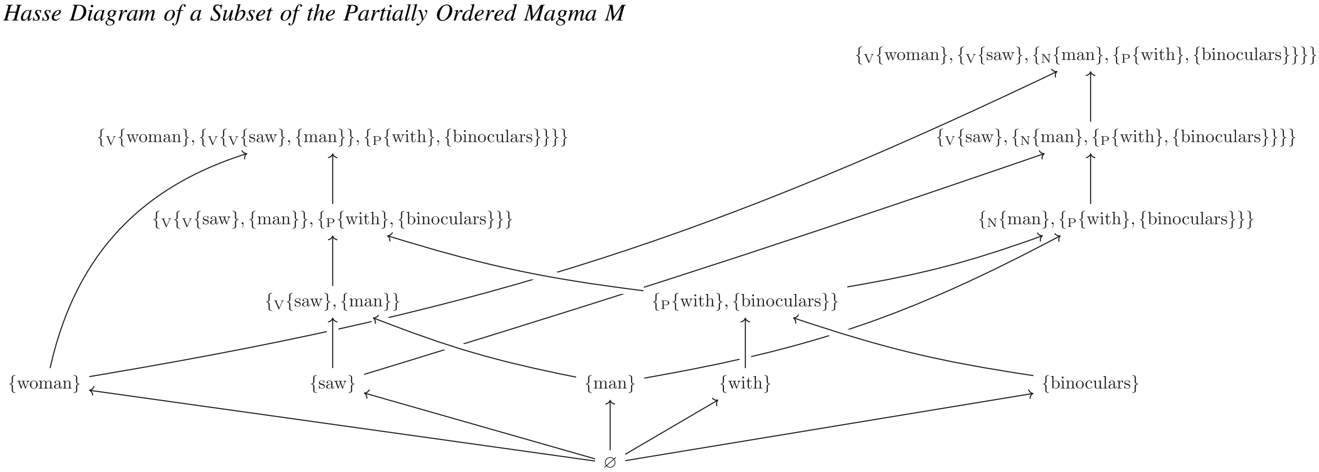
$$\textit{if } \overbrace{\boxed{\alpha \gg \beta \textit{ and } \beta \not\gg \alpha}}^{\alpha \text{ asymmetrically c-commands } \beta}, \textit{ or if } \overbrace{\boxed{\alpha \leq \hat{m} \textit{ and } \hat{m} \gg \beta \textit{ and } \beta \not\gg \hat{m}}}^{\text{a node dominating } \alpha \text{ asymmetrically c-commands } \beta}, \textit{ then } \overbrace{\boxed{\alpha \prec \beta \in f(m)}}^{\alpha \text{ precedes } \beta}$$

In short, Definition 6 states that a word α precedes a word β if it asymmetrically c-commands β or if a node dominating α asymmetrically c-commands β. The result of this mapping is a full total ordering of the terminals of the hierarchical structure in question. It is important to note that this mapping can be defined as a proper function because, under the LCA, only one linear sequence can be derived from any given hierarchical structure. Conversely, multiple hierarchical structures can map onto the same linear sequence. For instance, the precedence relations that are derived from the asymmetric c-command relations in the two structures in Figure 3 are the same, which illustrates the fact that the corresponding sequence is structurally ambiguous.

Ordering Sequences via Structures

When the sets in M are mapped to sequences in S, these sequences are imbued with grammatical properties. What these grammatical properties are can be understood in terms of the ordering that is carried over from the containment relation in M. Consider Figure 5,

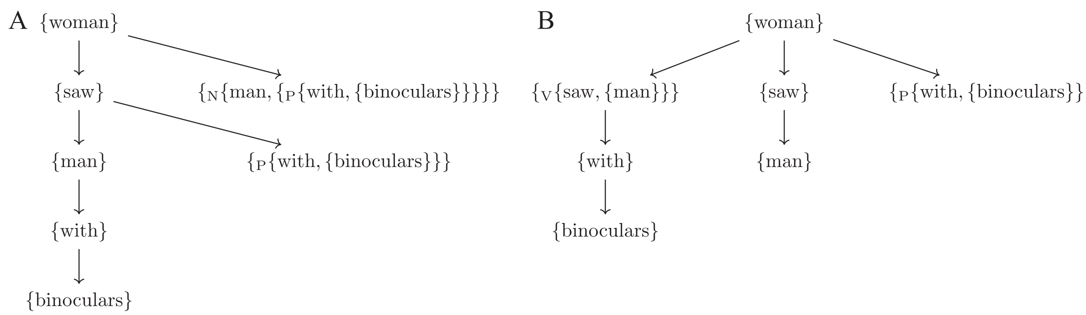
Figure 2
Hasse Diagram of a Subset of the Partially Ordered Magma M



Note. The Hasse diagram displays two different structures that map onto the sequence "woman saw man with binoculars". Arrows indicate direct containment. The subscript at the opening curly bracket of each binary set indicates the label of that set.

Figure 3

Hasse Diagram of a Subset of the Partially Ordered Magma M



Note. The Hasse diagram displays two different structures that map onto the sequence "woman saw man with binoculars". Arrows indicate asymmetric c-command. The subscript at the opening curly bracket of each binary set indicates the label of that set. The diagram in (A) corresponds to the structure in Figure 1A, where the man has the binoculars. The diagram in (B) corresponds to the structure in Figure 1B, in which the woman is holding the binoculars.

where the constituent structures in M (left panel) are mapped to the sequences in S (middle panel) via the LCA. By virtue of the containment relation by which the elements of M are ordered, this mapping imposes structure on the set of sequences (right panel) that is not there if only their sequential properties are considered.

If we only consider the sequential properties of the elements in S, a partial ordering already exists. This partial ordering is based on string containment. For example, both "woman with" and "with binoculars" can be said to be contained in the sequence "woman with binoculars." Using the Map f: $M \rightarrow S$, we impose a restriction on this ordering: for two elements $m_1, m_2 \in M$, $f(m_1) \leq f(m_2)$ iff $m_1 \leq m_2$. That is, two sequences in S are contained in one another only if their constituent structures in M are contained in one another. This imposed ordering restricts the initial ordering by excluding both ungrammatical sequences as well as containment relations that are not the result of a structural relationship. For example, in the middle panel of Figure 5, the subsequences s_8, s_9, and s_{10} do not appear in the imposed partial ordering. s_{10} is an ungrammatical sequence and therefore has no structural analog in M. s_8 and s_9 are subsequences of a grammatical sequence, yet they do not correspond to constituents and are therefore not retained in the ordering. Thus, only strings that correspond to constituents are retained in the partial ordering, and this partial ordering is based on constituent containment, as can be seen in the substructure in the right panel of Figure 5.

To sum up, we used the binary set formation operator $\oplus$ to generate hierarchical constituent structure. From the resulting structure, whose containment relationships are visualized in Figure 2, we derive all c-command relationships (see Figure 3). From these c-command relationships, we derive a linear sequence with precedence relationships using the LCA. Using the containment relationship in the partially ordered magma (see Figure 2), we impose an ordering relation on the resulting set of sequences (see Figure 5). The latter is possible because we define the algebraic structure corresponding to the set of structures as a magma, whose combinatorial operator is nonassociative. This allows us to generate strongly compositional structure, which is a necessary requirement for any description of (the faculty of) language.

Figure 4

Hasse Diagram of an Element of the Set of Sequences S

$$\text{woman} \longrightarrow \text{saw} \longrightarrow \text{man} \longrightarrow \text{with} \longrightarrow \text{binoculars}$$

Note. The Hasse diagram displays the sequential structure of the sequence "woman saw man with binoculars". Arrows indicate precedence.

Hierarchical Structure in Actions

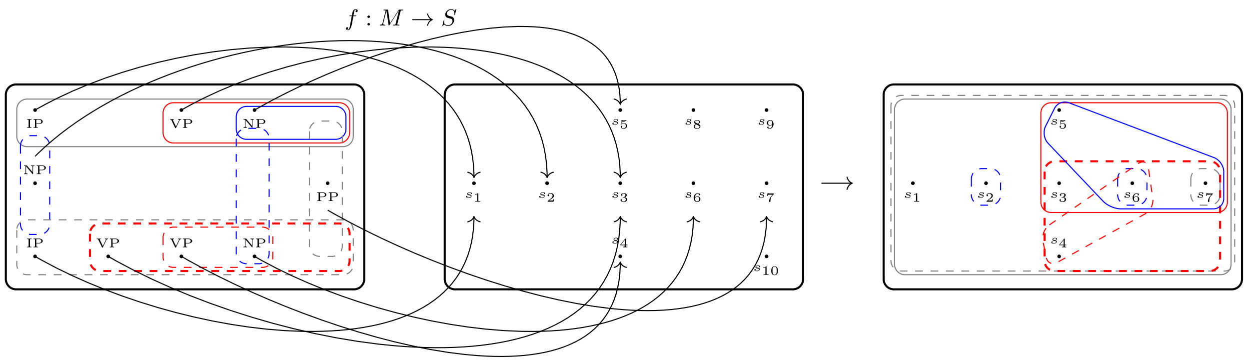
Having defined and formalized the properties of hierarchical structure in syntax, we will now consider whether action hierarchies are analogous to syntactic hierarchies. Similar to the hierarchical structure underlying sentences, action sequences are thought to be governed by hierarchically organized action plans (Botvinick, 2008; Cooper & Shallice, 2000, 2006; Holloway, 1969; Koechlin & Jubault, 2006; Lashley, 1951; Miller et al., 1960; Rosenbaum et al., 2007).[7] This structural analogy between linguistic syntax and actions has received considerable attention from several corners of cognitive science (Boeckx & Fujita, 2014; Fadiga et al., 2009; Jackendoff, 2007, 2009; Moro, 2014a, 2014b; Stout & Chaminade, 2009), in which the hierarchical structure of actions is thought to be generated by an *action syntax* (Fitch & Martins, 2014; Fujita, 2014; Maffongelli et al., 2019; Pulvermüller, 2014).

The idea is often illustrated using the example of tea- or coffee-making as a goal-directed behavioral routine (Cooper & Shallice, 2000; Fischmeister et al., 2017; Fitch & Martins, 2014; Humphreys & Forde, 1998; Jackendoff, 2007, 2009; Kuperberg, 2020). A multistep action such as tea-making can be decomposed into discrete subsequences of actions, which in turn can be decomposed into subsubsequences, and so forth. Figure 6 shows a visual representation of the hierarchical part-whole structure of "making tea." The highest level in the hierarchy represents the complex, temporally extended and goal-directed action, middle levels represent short-term, less complex subactions with their own subgoals, and the lowest level (terminal nodes) contains atomic actions with immediate subgoals. Decomposing complex actions into these embedded subsequences is theoretically and empirically warranted because the subsequences may be used in different tasks, because they are sometimes omitted,

[7] Note that we are concerned with describing the structure of actions rather than with describing how action sequences come about in a processing system (Badre, 2008; Tettamanti & Moro, 2012). The latter question belongs to the study of motor control, which is also hierarchically organized but has different properties: Motor control is based on causal relations ("processing" hierarchy), while actions should be described in terms of part-whole relations ("representational" hierarchy; see Uithol et al., 2012, for discussion).

Figure 5

An Ordering Relationship Is Imposed on S via the Structure in M



s_1 : woman saw man with binoculars
s_2 : woman
s_3 : saw man with binoculars
s_4 : saw man
s_5 : man with binoculars
s_6 : man
s_7 : with binoculars
s_8 : man with
s_9 : saw man with
s_{10}: man man with with

Note. PP = prepositional phrase; VP = verb phrase; NP = noun phrase; LCA = Linear Correspondence Axiom. The leftmost panel contains a subset of the partially ordered magma *M*, with the elements (denoted by their labels) ordered by containment. The labels IP, VP, NP, and PP refer to the labels of the constituents. Here, IP stands for Inflectional Phrase (whose head contains information about tense and inflection), which is often used as the top-most constituent in the structural representation of a sentence. A subset of *S* is shown in the middle panel, with example sequences presented below the figure. The LCA function *f*: *M* → *S* maps elements in *M* to elements in *S*, thus imposing a structural ordering relation on the sequential elements in *S* (rightmost panel). See the online article for the color version of this figure.

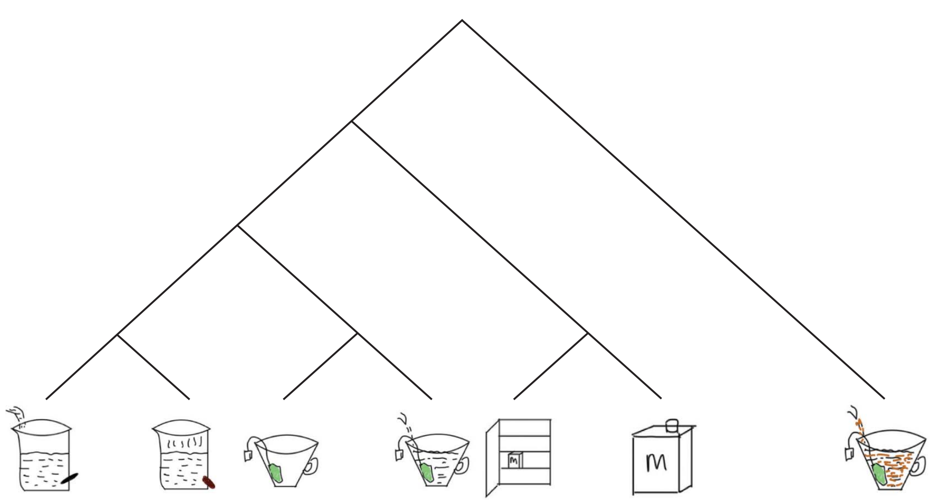
repeated, or substituted as a whole, and because they all have their own subgoal, which must be fulfilled in order to achieve the overarching goal (Cooper & Shallice, 2000, 2006; Humphreys & Forde, 1998; Lashley, 1951; Norman, 1981; Reason, 1979; Rosenbaum et al., 2007; Schwartz, 2006).

Formalizing Action Structure (1)

The following sections describe the structure of actions using the same mathematical formalism used to describe language in Formalizing Linguistic Structure section. We first show that this

Figure 6

An Example of a Hierarchical Decomposition of an Action Sequence, Such as Making Tea



Note. The terminal nodes correspond to the atomic actions in Figure 7. See the online article for the color version of this figure.

formalism is inadequate for describing actions. Formalizing Action Structure (2) section then proposes an alternative way to describe the structure of action sequences.

Definition 7. (M, $\oplus$, $\varnothing$) is a unital, commutative magma generated from A, where:

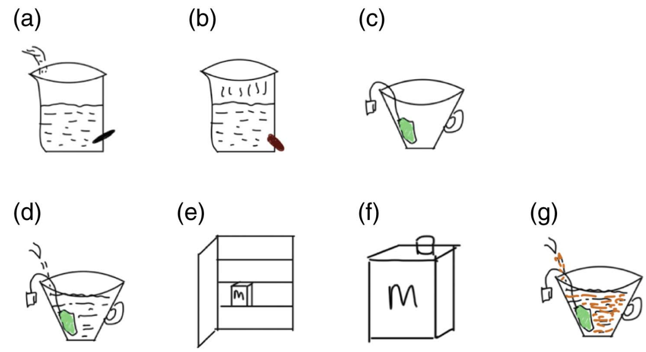
1. A is a set containing atomic actions, such as the examples presented in Figure 7.
2. M is a set of elements that are generated from A, with $A \subset M$.
3. $\oplus$ is a binary set formation operation, which is commutative, nonassociative, and closed.
4. $\varnothing$ is the identity element.

By defining the same binary relationship as used in Definition 2, we derive a partially ordered magma in which the actions and action sets are partially ordered by containment. A subset of this partially ordered magma is visualized in the Hasse diagram in Figure 8, which displays the containment relationship for two structures that map onto the same action sequence for making tea. Note that the two topmost action structures are derived in a different way. This figure illustrates a crucial point about the (ir)relevance of hierarchical structure in the interpretation of action sequences. That is, because the $\oplus$ operator is nonassociative, the order in which actions are combined using $\oplus$ affects the structure that is generated. Therefore, if we were to interpret these structures in a strongly compositional way, we would have to conclude that they correspond to different actions. This is clearly an undesirable conclusion, because the two structures correspond to one and the same action sequence. In other words, adopting a nonassociative combinatorial operator for generating action structures makes the model too strong: It will differentiate two action structures that should not be distinguished because they map onto the same action sequence and thus achieve the same goal in effectively the same way.

Compositionality in Language and Action

The fact that a strongly compositional formal model does not accurately describe actions indicates that the action system is not strongly compositional. If the action system is weakly compositional instead, it follows that one action sequence cannot be associated with multiple hierarchical structures. This prediction is borne out: structurally ambiguous actions, where one action sequence is associated with more than one structural representation and therefore more than one goal, do not seem to exist. This does not mean that actions cannot be ambiguous. Any given action may be characterized in terms of different goals, but these different goals are not a function of a decomposition of the action sequence in terms of hierarchically organized "action constituents." Whether the action's goal is achieved depends on the temporal order of its constituent actions, not on their hierarchical organization.

Figure 7
Atomic Actions for Tea Making



Note. (a) Fill kettle with water. (b) Turn on kettle. (c) Put teabag in cup. (d) Pour hot water into cup. (e) Open fridge. (f) Grab milk. (g) Pour milk into cup. In the context of Definition 7, $A = \{a, b, c, d, e, f, g\}$. See the online article for the color version of this figure.

Formalizing Action Structure (2)

In the previous section, we showed that when the language formalism is applied to actions, it appears to be too strong: It makes a distinction that should not be made. The model is also too weak: Actions and action plans are structured by temporal (precedence) relations, but the model does not take temporal order into account. In the current section, we therefore propose an alternative way to describe the structure of actions. The operator used to generate action structures must meet at least two requirements. First, it must generate sequential structure, because actions are temporally ordered. Second, it must not be nonassociative, because actions are not strongly compositional.

A Set of Partitioned Sequences

We have already defined the set of sequences as S (see Definition 4). The elements in each sequence are in a total, transitive, and antisymmetric ordering (see Box 2). This set is partitioned according to the following criterion: Sequences are deemed equivalent if they bring about a particular change in the environment (i.e., they achieve the same "goal"). All equivalent sequences are part of a single equivalence class whose label corresponds to the goal achieved by the sequences in it.

Definition 8. Given the set S, a partition of S contains a set G, and for each $g \in G$, a nonempty subset $S_g \subseteq S$ exists, such that:

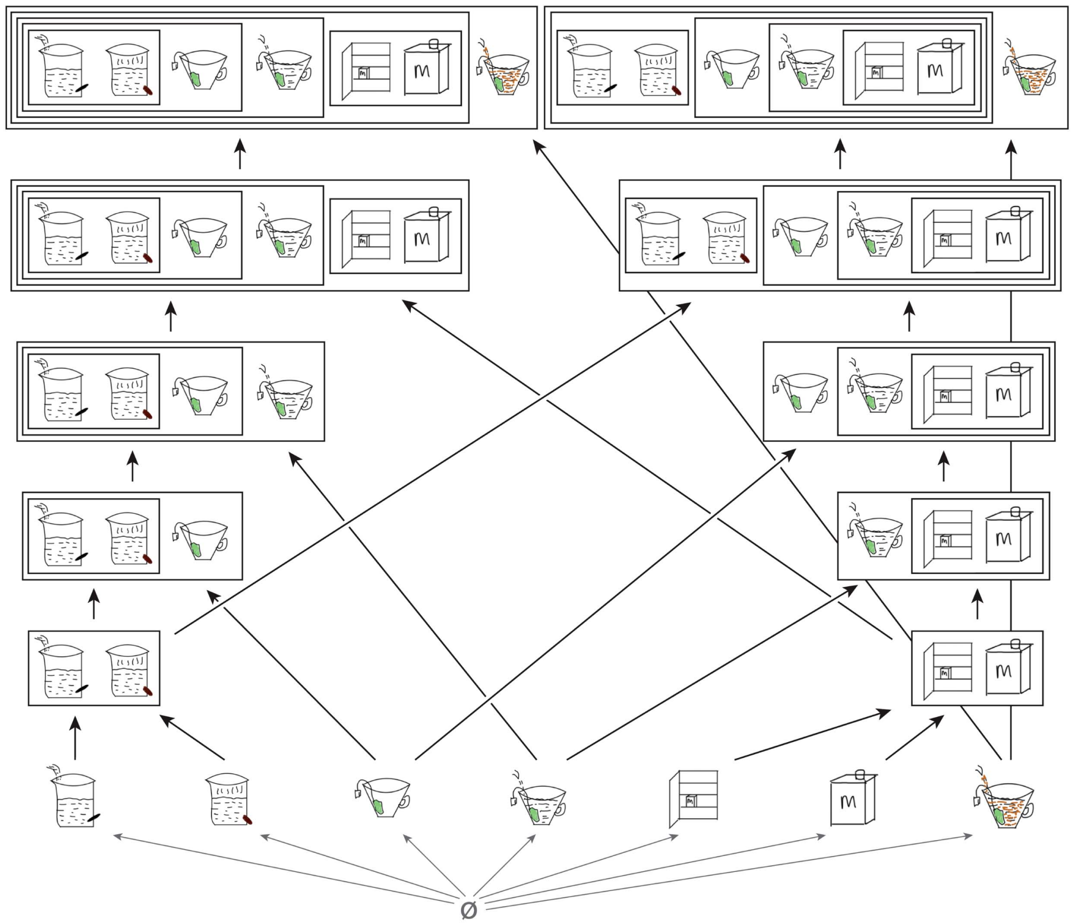
the set of sequences is the union of all parts

$$\boxed{S = \bigcup_{g \in G} S_g} \ \textit{and if } g \neq h, \ \boxed{S_g \cap S_h = \varnothing}$$

the parts do not overlap

Figure 8

Hasse Diagram of a Partially Ordered Magma, Which Displays Two Different Structures That Map Onto the Same Action Sequence for Making Tea



Note. The boxes around action combinations represent binary sets, and the arrows indicate direct containment. See the online article for the color version of this figure.

Here, we take $g \in G$ as the set of part labels (i.e., the labels given to each element in the partition). Because all sequences in a given part are equivalent, we call every element $s \in S_g$ a representative sequence of that part S_g.

The partitioning of *S* yields a set of part labels that correspond to the set of goals they accomplish. These goals can be interpreted as abstractions over action sequences that have something in common, namely the change they bring about in the environment (see e.g., Cooper & Shallice, 2000).

Generating Structured Sequences

Definition 9. We define action structure as (G, ⊗, ∗, ∅), where:

1. The elements of *G* are part labels (see Definition 8) corresponding to action sequences that achieve a particular goal.
2. ⊗ and ∗ are two sequence-building operators that generate the elements of G, with ∅ as the identity element.

Note that we include the set of atomic actions in *G*, because atomic actions achieve a particular change in the environment and therefore have their own subgoal. Therefore, an atomic action is simply an equivalence class with only one element.

A goal can often be achieved in several ways. For example, given the actions in Figure 7, the goal "make black tea" corresponds to the part S_b, where $S_b = \{(a, b, c, d), (a, c, b, d), (c, a, b, d)\}$. Here, *a* ("fill kettle with water") must precede *b* ("turn on kettle"), which in turn must precede *d* ("pour hot water into cup"), so the relative temporal ordering of (*a*, *b*, *d*) is fixed. However, the position of action *c* ("put teabag into cup") within this action sequence is specified only in relation to *d*; it can be placed at any position before *d* within (*a*, *b*, *d*), thus yielding three action sequences. In other words, for a given goal to be achieved, the temporal ordering of some actions must be specified, whereas it need not be specified for other actions. We achieve this combination of the requirement of strict temporal ordering with temporal flexibility via the use of two sequence-building operators.

Definition 10. $*$ is a sequence-building operation $*$: $G \times G \rightarrow G$. Let $a, b, c \in G$ be three part labels, and let $s_a \in S_a$, $s_b \in S_b$, $s_c \in S_c$ be three representative sequences, where $s_a = (a_1, a_2, \ldots a_n)$, $s_b = (b_1, b_2, \ldots b_m)$.

$*$ concatenates two sequences

$$s_a * s_b = (a_1, a_2, \ldots a_n, b_1, b_2, \ldots b_m) \quad where\ \forall s_a, s_b, s_c \in S,\ s_a * (s_b * s_c) = (s_a * s_b) * s_c$$

$*$ is associative

Definition 11. $\otimes$ is a sequence-building operation $\otimes$: $G \times G \rightarrow G$. Take s_a and s_b as defined in Definition 10. Then $s_a \otimes s_b = \{(c_1, c_2, \ldots c_{m+n})\}$, such that

$\otimes$ retains precedence relations in its input

$$c_i = \begin{cases} a_j & if\ i > 1\ and\ a_{j-1} \in (c_1 \ldots c_{i-1}) \\ b_j & if\ i > 1\ and\ b_{j-1} \in (c_1 \ldots c_{i-1}) \\ a_1\ or\ b_1 & otherwise \end{cases}$$

The operator $*$ is simple concatenation. This operator is required because the temporal ordering of some actions must be specified. For example, if $h \in G$ represents the (sub)goal "obtain hot water", then we must define $S_h = a * b$ due to the requirement that the kettle should be filled with water (action a) *before* it is turned on (action b). The temporal precedence relationship between these two actions requires an operator that yields strict sequential orders. Clearly, concatenation is not commutative: The sequence generated by $s_a * s_b$ is different from the output of $s_b * s_a$. Moreover, because $*$ generates sequences whose only relationship is precedence, it is associative: $s_a * (s_b * s_c) = (s_a * s_b) * s_c$. In sum, $(G, *)$ forms a monoid, which is an algebraic structure consisting of a set equipped with an operation that is closed, unital, and associative (see Box 1).

The operator $\otimes$ generates sets of sequences whose orders vary, with the only constraint that the relative ordering within its arguments is retained. For example, for two sequences, $s_a = (a, b, d)$, $s_b = (c)$, $s_a \otimes s_b = \{(a, b, d, c), (a, b, c, d), (a, c, b, d), (c, a, b, d)\}$. In each of the sequences generated by $s_a \otimes s_b$, a precedes b, which precedes d. So, while $\otimes$ allows for flexibility in terms of the order of the actions in the sequences, the flexibility is constrained by the sequential properties of s_a, whose precedence relations must be retained in the output of $s_a \otimes s_b$. Because $\otimes$ is a sequence-building operator that is constrained only by the sequential properties of its input (i.e., the ordering *within* its input arguments), $\otimes$ is associative. But as it does not specify the ordering *among* its input arguments, $\otimes$ is also commutative. $(G, \otimes)$ therefore forms a commutative monoid (see Box 1).

The two notions of precedence and flexibility are combined in $(G, \otimes, *)$, which is an algebraic structure called a trace monoid (also called partially commutative monoid; see Box 1). A trace monoid is a monoid of traces, which are sets of sequences that form equivalence classes (Mazurkiewicz, 1995). In $(G, \otimes, *)$, the traces contain equivalent sequences generated by $\otimes$ and $*$. In a trace monoid, two sequences are equivalent if they only differ in the order of a pair of elements for which an independency relation is defined.[8] These independent elements are allowed to commute in the sequences of the equivalence class.[9] Consider the independency $I = \{(b, c), (c, b)\}$, which holds that the actions b and c are allowed to commute; no precedence relation between them is specified. Given $(b, c) \in I$, we say that two action sequences are equivalent if they differ only in the ordering of b and c. The trace monoid is then said to contain a trace where $acb \sim abc$.

To sum up, defining the trace monoid $(G, \otimes, *)$ allows us to achieve simultaneously temporal precedence *and* temporal flexibility. The operator $*$ is required to build sequences where temporal precedence is necessary (e.g., "grab milk" must precede "pour milk into cup"), and $\otimes$ is used to generate action sequences whose temporal relationship is not specified (e.g., the action "grab milk" can precede or follow "put teabag in cup"). The combined use of $*$ and $\otimes$ leads to equivalence classes of sequences that contain a mixing of intermediate goals that are temporally independent of other intermediate goals. The mixing procedure introduced by $\otimes$ might destroy immediate precedence (or temporal adjacency) relationships in the output of $*$, but this is unproblematic: While it makes sense to let "open fridge" be directly followed by "grab milk", this is not necessary. One could open the fridge, perform all other tea-making preparations, and then grab the milk.

Hierarchical Relations Between Action Sequences

While the output of the two associative operators is a set of sequences, these sequences contain underlying structure if we take their derivational history into account (cf. "configurational properties" in Miller et al., 1960). For instance, given the atomic actions in

[8] Independency relations are symmetric (i.e., if (a, b) is present, then so is (b, a)) and irreflexive (i.e., there are no relations of type (a, a)) and can be extended to relations between sequences (see Mazurkiewicz, 1995).

[9] Commutativity in the general sense is slightly different from the way it is used in the context of traces. In the general sense (as used in Box 1), it refers to an operation that produces the same output if the order of the operands is changed, such as in $a \otimes b = b \otimes a$. In the context of a trace monoid, the notion of sameness is replaced by equivalence, where $a \otimes b = \{ab, ba\}$, and $ab \neq ba$ but $ab \sim ba$.

Figure 7, the sequence (a, b, c, d) corresponds to the goal of making black tea. By itself, this sequence does not provide a lot of information about the precedence relations that might hold for the complex action; it could in principle have been generated via $(((a \otimes b) \otimes c) \otimes d)$. Such information can be inferred only if additional action sequences are observed that achieve the same goal (see Box 3). Knowing how the action sequence was derived allows us to specify the temporal constraints to which it must adhere, which in turn provides information about the causal structure of the action plan. Thus, by deriving $[(a, b, c, d)] \in (a * b) \otimes (c * d)$, we make temporal precedence relations concrete: a must precede b, and c must precede d.

The derivational history of the sequence provides information about which other sequences are also possible. From observing only (a, b, c, d), it would be impossible to know whether (c, a, b, d) is also a fine sequence. However, that knowledge can be deduced if we know how the sequence was derived, because $(a * b) \otimes (c * d)$ also generates the sequence (c, a, b, d). The derivational history thus provides information that is not present in the temporal structure of the sequence, including information about the relationship between the output sequence and its subsequences. In (c, a, b, d), it is still the case that c precedes d, even though they are not adjacent anymore. But by taking into account the derivational steps leading to (c, a, b, d), we can specify a hierarchical relationship between (c, d) and (c, a, b, d), which states that $(c, d) \in (c, a, b, d)$ because (c, a, b, d) was generated via $(a, b) \otimes (c, d)$. This relationship holds even though (c, d) is not a subsequence of (c, a, b, d). Action sequences can be seen as hierarchical sequences, which are sequences with a derivational history that specifies how they relate to the action sequences from which they are derived. This allows us to go beyond the sequential structure of actions in a system that is still weakly compositional.

Language Versus Action

A Formal Comparison

In the previous sections, we described the properties of hierarchical linguistic structure (generated by $\oplus$) using a magma. When this formalism was applied to actions, it appeared to be too strong, deriving multiple "interpretations" from unambiguous action sequences, and too weak, as it does not generate temporally ordered structures. To overcome these limitations, our alternative formalism of actions described action structures as a trace monoid (generated by noncommutative $*$ and commutative $\otimes$). A crucial difference between these algebraic structures (see Box 1) is that the operation associated with magmas is nonassociative, whereas that associated with monoids is associative. As a consequence, the structure generated by $\oplus$ and represented in the magma is strongly compositional: The constituent structure of the input to $\oplus$ is retained in its output. This is important because both syntactic operations and semantic interpretation are structure-dependent. If the internal structure of each combination would be lost, syntactic rules could not target constituents. Moreover, meaning could not be derived from constituent structure, so sentences could not be structurally ambiguous; the system would generate only one output for ((deep $\oplus$ blue) $\oplus$ sea) and (deep $\oplus$ (blue $\oplus$ sea)). In contrast, the action structure generated by the associative operators $*$ and $\otimes$ and represented in the trace monoid is weakly compositional. This weakly compositional, order-sensitive model can account for the relevant properties of action structures. To conclude the formal comparison, by comparing language and action in terms of the combinatorial operations underlying their structures, this work paves the way for further, formally grounded cross-domain comparisons of the features of human cognition.

The Nature of Structure

Our formal characterizations of language and action show that their structural representations are different, in particular, concerning the relevance of constituency (see Papitto et al., 2020; Zaccarella et al., 2021, for a similar conclusion from the neuroimaging literature). The same conclusion is reached when we compare language and action in terms of the properties of syntactic structure discussed in Properties of Syntactic Structure section.

Self-Similar

It has been argued that the combinatorial operation involved in building syntactic structures evolved from a preexisting system for tool

Box 3
Inferring Plans From Action Sequences

In order to achieve a given goal, the relative order of some related actions must be specified, whereas that of some unrelated actions can be left undefined. Given a set of observed action sequences that successfully reach the same goal, the abstract plan to reach that goal can be extracted via the intersection of the sets of binary relations representing the sequences.

As a simple illustration, consider a sequence $x = (a, b, c, d)$ consisting of nonrepeating atomic actions. The precedence relations for x are $a \prec b \prec c \prec d$. The sequence can be represented as a set of binary relations. If we take these binary relations to represent precedence, x will be represented as $\{(a, b), (b, c), (c, d), (a, d), (a, c), (b, d)\}$. By observing only x, it is not immediately clear which of these elements are dependent and which are not. However, observing the sequence $y = (c, a, b, d)$ (represented as $y = \{(c, a), (a, b), (b, d), (c, d), (c, b), (a, d)\}$)), which achieves the same goal, provides more information. The plan to reach the goal is represented by the intersection of the sets of binary relations:

$$x \cap y = \{(a, b), (c, d), (b, d)\}$$

This intersection corresponds to the plan of making black tea (see Figure 7). Notice how this partial order is compatible with the previously unseen sequence (a, c, b, d), which reaches the same goal successfully as well.

use, also called Action Merge, which is thought to apply recursively (Fujita, 2014, 2017; Pulvermüller, 2014; Stout & Chaminade, 2009). As recursively generated hierarchy is characterized by self-similarity across levels of the hierarchy (Martins, 2012), one approach toward determining whether action structures are recursively generated is to examine whether they display self-embedding of tokens of the same type. That requires knowing what the types are. Consider the structure in Figure 6. One could combine "open fridge" and "grab milk" into an action constituent, which could be labeled "get milk". Here, it is unclear whether "get milk" is of the same type as "grab milk". Moreover, it seems plausible that the action "pour hot water into cup" is similar to "pour milk into cup," but that is because the tokens are similar (both involve pouring), not necessarily because their types are. To determine whether action structures are recursively generated, a theoretical specification of the types of actions is needed.

Our primary goal is to evaluate the claim that the hierarchical structures found in language and action are analogous. The validity of this claim rests on positive evidence that actions, like language, are recursively generated. In the absence of such evidence (e.g., in the form of self-similar hierarchy), it is premature to conclude that actions are structurally analogous to language.

Endocentric

Some of the hierarchical representations of actions that are proposed in the literature contain action constituents with one key element, or "head", which describes the core of the action and determines its (end)goal (e.g., Fischmeister et al., 2017; Jackendoff, 2007, 2009, 2011). While this makes the structures "headed," it does not make them endocentric. That is, it seems that this head merely serves to describe the main action of the action sequence rather than to provide a label for the constituent it is dominated by. In Figure 6, for instance, the action constituent formed by the combination of "open fridge" and "grab milk" is not a type of either of these actions. In line with the idea that endocentricity is unique to language (Boeckx, 2009; Hornstein, 2009), action hierarchies seem to be exocentric.

A plausible reason for the difficulty in assigning labels to action constituents is that actions do not have clear conceptual units, such as words (Berwick et al., 2011; Moro, 2014a), and that groups of actions do not obligatorily fall into a closed set of distinguishable categories, such as NP or VP (Jackendoff & Pinker, 2005). Without these categories, groupings of actions into constituents cannot be labeled or "syntactically" named, which means that there are no grammatical constraints on how the resulting constituents can be used in further combinations.

Unordered

Representations of actions are intimately tied to the physical environment in which the actions are performed (Graves, 1994; Kuperberg, 2020; Moro, 2014a; Zaccarella et al., 2021). As such, they are not order-independent: Some subactions must precede others in order for the action to achieve its goal (Fitch & Martins, 2014), and indeed, the output of Action Merge is inherently ordered (Fujita, 2014).[10] Comparing this to language, we see that the externalization of spoken language is also sequential, but that sequential order does not play a role in the representation of syntactic relations, which are invariably structure-dependent.

It has been proposed that closely related actions, which can be separated by arbitrarily many "embedded" actions (e.g., [open door [switch on light [brush teeth] switch off light] close door]), are similar to long-distance dependencies in language (Pulvermüller, 2014; Pulvermüller & Fadiga, 2010). This analogy is incorrect, however, regarding the hierarchical organization of constituent structure. While these action dependencies have serial and temporal properties (e.g., you cannot close a door *before* having opened it), they have no hierarchical properties. If they were truly hierarchical, the embedded action would be expected to adhere to structural restrictions on its distribution, which would be the case if the embedding of [brush teeth] at a different position, like in [open door [brush teeth] [switch on light switch off light] close door], were not allowed. Moreover, if the dependency between "switch on light" and "switch off light" were hierarchical, it should not be affected by linearly or temporally intervening actions, so whatever happens during "brush teeth" should not be able to affect the action "switch off light." As neither appears to be the case, it is more appropriate to label the dependency between two actions temporal (or causal) rather than hierarchical (Moro, 2014b, 2015). Indeed, actions and events can be understood in terms of temporal (and causal) structure (McRae et al., 2019; Zacks & Tversky, 2001), and oddly ordered complex actions, which are thought of as ungrammatical actions (e.g., Maffongelli et al., 2019), reflect the violation of "temporal rules" rather than phrase structure rules (Zaccarella et al., 2021).

The observation that none of the properties of hierarchy in syntax are found in actions suggests that the analogy between language and action is not to be found in syntactic structure. One plausible reason for this is that the syntax of actions cannot be independently defined. In language, syntax is computationally autonomous, having its own principles and properties that cannot be reduced to other factors, such as meaning (Adger, 2018; Berwick, 2018; Chomsky, 1957). The application of these principles is constrained by economy conditions (e.g., locality, minimality; see Collins, 2001), but not by whether they generate semantically interpretable or coherent output. Therefore, in language, there is an independent notion of grammaticality: Sentences are ungrammatical if their structures cannot be generated by the rules of syntax, or if they violate conditions on these rules. One way to illustrate this is by means of interpretable but nevertheless ungrammatical sentences. A sentence such as "which boy did they meet the girl who insulted?" is ungrammatical but can be interpreted (i.e., corresponding to the logical statement "for which x, x a boy, did they meet the girl who insulted x?"). Its deviance is due to the violation of a purely formal (locality) principle constraining the grammar, which is unrelated to its interpretability. Conversely, the sentence "colorless green ideas sleep furiously" is semantically odd, yet

[10] Even under an analysis in which *immediate* precedence plays a role in syntax (as in Kayne, 2011), the crucial difference between language and actions remains: If two linguistic objects α and β are not adjacent in their base-generated position (i.e., they do not form the ordered pair $\langle\alpha, \beta\rangle$), their relationship is defined as a relationship that refers to the (hierarchical) constituents they are contained in, not as a relationship that refers to their linear or temporal order. There is no such constraint in actions, where some actions must precede (distant) others, regardless of the relationship between the action constituents in which they are contained.

fully grammatical, showing that grammaticality does not boil down to meaningfulness or interpretability.

In contrast, the validity of action sequences seems to be related to their coherence, in terms of both logical consistency and environmental appropriateness. It has been suggested that a complex action is "ungrammatical" or "ill-formed" if its subparts are ordered in such a way that the action's overall goal cannot be achieved (Jackendoff, 2007; Maffongelli et al., 2019). The "grammaticality" of an action is thus intimately tied to the fulfillment of its goal, showing that the notion "ungrammatical" is very different for action sequences and sentences. On this interpretation, an "ungrammatical" action is similar to a sentence that does not convey the intended meaning, either because it is logically incoherent or because it is situationally inappropriate. The action equivalent of a logically incoherent sentence could be an action sequence in which a coffee grinder is turned on before the coffee beans are added. This is logically incoherent because it violates causality principles of the physical environment. An action like turning off the light when walking into your office during nighttime, instead, does not violate such principles, but it would be situationally inappropriate because it would preclude you from seeing anything.

Because there is no autonomous action syntax, there is no independent notion of grammaticality, devoid of goal-dependent meaning. As a result, it is unclear how to evaluate whether a given structural decomposition of complex actions into constituents is veridical unless we know the goal or conceptual content of the action (Berwick & Chomsky, 2017; Jackendoff, 2007). It seems that the decomposition of an action sequence into a hierarchical tree structure only works to the extent that the subactions are meaningful or coherent (i.e., represent subgoals).

Levels of Abstraction

The difference between language and actions in terms of their dependence on hierarchical and sequential structure can be captured quite naturally under the distinction between hierarchical sets and hierarchical sequences, a terminological contrast adopted by Fitch and Martins (2014) to distinguish possible interpretations of the term hierarchy.[11] Fitch and Martins (2014) describe hierarchical sets as structures that specify the superior/inferior relation between their elements (i.e., specifying containment), but whose elements are unordered at any given level. Hierarchical sequences, instead, are hierarchically generated structures in which sequential order matters: At least some elements at any given level represent a sequence rather than a set.

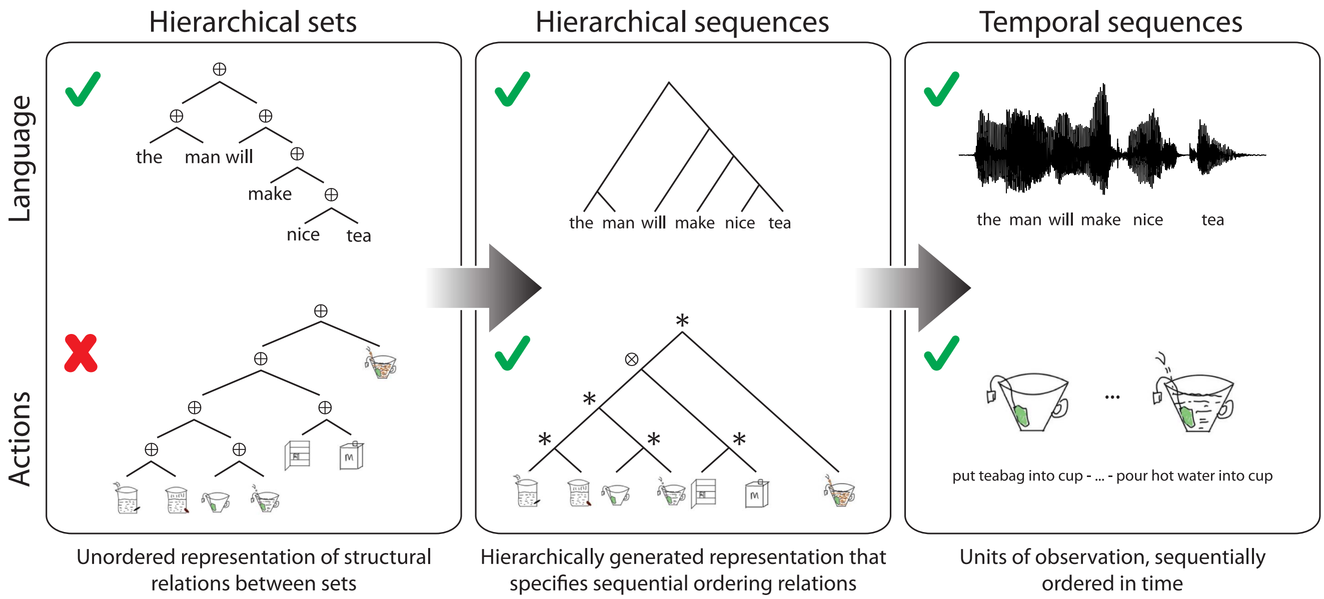
We will argue that language needs to be described in terms of both hierarchical sets and hierarchical sequences, but that actions can be described as hierarchical sequences only (see Figure 9). Regarding language, the level of hierarchical sets directly corresponds to the output of the combinatorial operator (left panel, top row in Figure 9). This level is explanatorily relevant for syntactic theory because it naturally captures the properties of syntax described in Properties of Syntactic Structure section: Hierarchical sets generated by Merge are self-similar, endocentric, and unordered (Lasnik, 2000). Explanations for both structural relations within languages and structural generalizations between languages should thus be stated at the level of hierarchical sets (e.g., the head-dependent relations in English and Japanese are identical at the level of hierarchical sets). These structural relations, however, are not directly realized in the external properties of the linguistic signal; externally, language does not contain sets. Instead, phrases and sentences are sequences that contain information about the way in which they were derived (see middle panel, top row in Figure 9). At this level of hierarchical sequences, we can describe a speaker's knowledge about those syntactic properties of their language that are realized in the sequential structure of words and phrases (e.g., morphosyntax, word order), such as whether heads precede or follow their dependents (e.g., English vs. Japanese).

As we noted at the end of Hierarchical Relations Between Action Sequences section, actions might be seen as structured sequences of events, whose derivational history is informative about the causal structure of the underlying action plan (middle panel, bottom row of Figure 9). However, the structural properties of actions cannot be described in a way that is completely detached from the physical instantiation of the action sequence, most clearly because (the representations of) action sequences contain information about temporal order. Because the properties of syntactic structure are not found in actions, it is not necessary to postulate hierarchical sets as an explanatorily relevant level of abstraction for actions.

The distinction between hierarchical sets and sequences is useful in explaining why it has been found that the brain areas involved in language processing (in particular, BA44 in the left inferior frontal gyrus) are also activated in response to tasks involving hierarchically organized actions (Higuchi et al., 2009; Koechlin & Jubault, 2006). At first thought, these neuroimaging results support the idea that there is a supramodal hierarchical processor in the brain that processes the hierarchical structures of cognitive systems such as language and action (Fadiga et al., 2009; Fazio et al., 2009; Fiebach & Schubotz, 2006; Higuchi et al., 2009; Jeon, 2014; Koechlin & Jubault, 2006; Tettamanti & Weniger, 2006). Crucially, however, instead of describing complex actions in terms of nonlinear relations defined over hierarchical structures, these accounts refer to the processing of structured sequences that were hierarchically generated (see Martins et al., 2019; Zaccarella et al., 2021, for related discussion). The overlapping activation patterns for language and action might therefore point not to shared brain regions processing hierarchical, nonlinear relations (operating over hierarchical sets), but rather to shared brain regions implicated in the processing of structured sequences (i.e., hierarchical sequences; see also Boeckx et al., 2014; Matchin & Hickok, 2020; Uddén & Bahlmann, 2012). As the externalized signal in both language (sentences) and action (action sequences) is a structured sequence, the processing requirements for both systems have to do with inferring relationships between temporally nonadjacent elements. We believe that a fruitful avenue for further investigation into the relationship between language and action concerns this type of structured sequence processing rather than the processing of the hierarchical structure itself. The overlap between language and action then has to do with the fact that, externally, both are structured sequences, even though their internal structures are quite different.

[11] A similar distinction is emphasized by Tettamanti and Moro (2012), who discuss the different meanings of hierarchical organization in terms of *sequential* versus *internal* hierarchy, describing respectively the computation of *sequential* hierarchical information (externalized) and the computation of *nonlinear* hierarchical relations (mind-internal).

Figure 9
Three Levels to Describe Hierarchically Structured Sequential Information



Note. The levels become increasingly concrete, from what is abstractly represented (on the left) to what is physically observed (on the right). The symbol at each node in the hierarchical structures indicates which operator was used to combine the elements, thus representing the derivational history of the sets in language (left) or the sequences in actions (middle). See the online article for the color version of this figure.

Conclusions

In response to the claim that language and action are analogous because they are both organized hierarchically, we argued in this article that the formal properties of hierarchy in both domains are fundamentally different. Our main argument is that the language system can embody strong compositionality, as both syntactic rules and semantic interpretation are structure-dependent. Structural analyses in language are thus concerned with nonterminal nodes in the hierarchical structure of syntax. Actions, instead, are weakly compositional: Regularities in action structures are dependent on the temporal order of the atomic actions, not on their hierarchical organization into action goals. Analyses of actions are thus concerned with terminal nodes in the action hierarchy. Based on this difference, we argue that the structure of syntax is best described as a system of hierarchical sets, whereas action structures can be described as hierarchical sequences.

In order to formally capture the strong compositionality of language, we described the algebraic structure corresponding to the ordered set of hierarchical structures in language as a magma, whose nonassociative combinatorial operator was defined as binary set formation. This set-based formalism integrates the three properties of syntactic structure (i.e., self-similarity, endocentricity, and unorderedness) with the description of syntax as a system of hierarchical sets and the fact that language exhibits strong compositionality. When this model was applied to actions, it appeared to be both too strong (i.e., it makes structural distinctions that should not be made) and too weak (i.e., it does not capture the importance of temporal precedence). We therefore proposed an alternative model for actions, which used two sequence-building operators that organize actions by sequential relations. This yielded an ordered set of action structures that could be described as a trace monoid. The associativity of the two operators formalizes the idea that actions exhibit a weaker form of compositionality, which is based on sequential rather than hierarchical structure. This aligns well with our argument that actions are best described in terms of hierarchical sequences. In sum, the formal tools needed to describe language are fundamentally different from those required to describe the action system. We believe that this result has important implications not only for comparative cognitive science but also for cognitive neuroscience, as it points to differences in the ways in which hierarchies are represented in the brain.

Glossary

- *Compositionality:* Property of a system which holds that the meaning of a complex expression is built up from the meanings of its parts and the way in which they are combined.
- *C-command:* Structural relation between nodes in a hierarchical structure. A node α c-commands a node β iff β is (contained in) the sister node of α.
- *Endocentricity:* Property of syntactic structures. A structural combination is endocentric if it fulfills the same grammatical function as one of its parts. Endocentric structures are contrasted with exocentric structures, in which the grammatical function of the combination is not the same as that of its parts.
- *Equivalence class:* A subset (of a set) containing elements that are related under the equivalence relation. An equivalence class is denoted by square brackets: $[a] = \{b | b \sim a\}$.

- *Equivalence relation:* An equivalence relation on a set is a binary relation that is reflexive, symmetric, and transitive. An equivalence relation is indicated by $\sim$, so $a \sim b$ denotes that a is equivalent to b.
- *Hasse diagram:* A mathematical diagram used to visualize an ordered set.
- *Linear Correspondence Axiom:* An algorithm that maps hierarchical structure onto linear order using the c-command relation.
- *Merge:* Combinatorial procedure to generate syntactic structure, formally defined as binary set formation: Merge(α, β) = $\{\alpha, \beta\}$.
- *Magma:* An algebraic structure consisting of a set equipped with a binary operation that is closed and not associative.
- *Monoid:* An algebraic structure consisting of a set equipped with a binary operation that is closed, unital, and associative.
- *Partial order:* A set with a binary relation that is transitive and antisymmetric.
- *Partition:* A partition of a set S is a set of nonempty subsets of S that cover S and are pairwise disjoint.
- *Recursion:* A recursive function is a function that can apply to its own output.
- *Self-similarity:* A self-similar object is similar to a part of itself.
- *Surjective:* Mathematical property of a function f: $X \rightarrow Y$ which holds that for every element $y \in Y$, there exists at least one element $x \in X$ such that $f(x) = y$.
- *Total order:* A set with a binary relation that is transitive, antisymmetric, and total.
- *Trace monoid:* A monoid whose operation is partially commutative.

References

Adger, D. (2018). The autonomy of syntax. In N. Hornstein, H. Lasnik, P. Patel-Grosz, & C. Yang (Eds.), *Syntactic structures after 60 Years: The impact of the Chomskyan revolution in linguistics* (pp. 153–175). De Gruyter Mouton.

Badre, D. (2008). Cognitive control, hierarchy, and the rostro-caudal organization of the frontal lobes. *Trends in Cognitive Sciences*, *12*(5), 193–200. https://doi.org/10.1016/j.tics.2008.02.004

Baggio, G. (2021). Compositionality in a parallel architecture for language processing. *Cognitive Science*, *45*(5), Article e12949. https://doi.org/10.1111/cogs.12949

Berwick, R. C. (2018). Revolutionary new ideas appear infrequently. In N. Hornstein, H. Lasnik, P. Patel-Grosz, & C. Yang (Eds.), *Syntactic structures after 60 years: The impact of the Chomskyan revolution in linguistics* (pp. 177–194). De Gruyter Mouton.

Berwick, R. C., & Chomsky, N. (2017). Why only us: Recent questions and answers. *Journal of Neurolinguistics*, *43*, 166–177. https://doi.org/10.1016/j.jneuroling.2016.12.002

Berwick, R. C., Okanoya, K., Beckers, G. J. L., & Bolhuis, J. J. (2011). Songs to syntax: The linguistics of birdsong. *Trends in Cognitive Sciences*, *15*(3), 113–121. https://doi.org/10.1016/j.tics.2011.01.002

Bloom, P. (1994). Generativity within language and other cognitive domains. *Cognition*, *51*(2), 177–189. https://doi.org/10.1016/0010-0277(94)90014-0

Boeckx, C. (2009). The nature of Merge: Consequences for language, mind, and biology. In M. Piattelli-Palmarini, J. Uriagereka, & P. Salaburu (Eds.), *Of minds and language: A dialogue with Noam Chomsky in the Basque Country* (pp. 44–57). Oxford University Press.

Boeckx, C., & Fujita, K. (2014). Syntax, action, comparative cognitive science, and Darwinian thinking. *Frontiers in Psychology*, *5*, Article 627. https://doi.org/10.3389/fpsyg.2014.00627

Boeckx, C., Martinez-Alvarez, A., & Leivada, E. (2014). The functional neuroanatomy of serial order in language. *Journal of Neurolinguistics*, *32*, 1–15. https://doi.org/10.1016/j.jneuroling.2014.07.001

Botvinick, M. M. (2008). Hierarchical models of behavior and prefrontal function. *Trends in Cognitive Sciences*, *12* (5), 201–208. https://doi.org/10.1016/j.tics.2008.02.009

Chomsky, N. (1957). *Syntactic structures*. Mouton de Gruyter.

Chomsky, N. (1959). A review of B. F. Skinner's verbal behavior. *Language*, *35*(1), 26–58. https://doi.org/10.2307/411334

Chomsky, N. (1995a). Bare phrase structure. In H. R. Campos & P. M. Kempchinsky (Eds.), *Evolution and revolution in linguistic theory: Essays in honor of Carlos Otero* (pp. 51–109). Georgetown University Press.

Chomsky, N. (1995b). *The minimalist program*. MIT Press.

Chomsky, N. (2013). Problems of projection. *Lingua*, *130*, 33–49. https://doi.org/10.1016/j.lingua.2012.12.003

Collins, C. (2001). Economy conditions in syntax. In M. Baltin & C. Collins (Eds.), *The Handbook of contemporary syntactic theory* (pp. 45–61). Blackwell Publishers.

Collins, C. (2017). Merge(X,Y) = {X,Y}. In L. Bauke & A. Blümel (Eds.), *Labels and roots* (pp. 47–68). De Gruyter Mouton. https://doi.org/10.1515/9781501502118-003

Cooper, R. P., & Shallice, T. (2000). Contention scheduling and the control of routine activities. *Cognitive Neuropsychology*, *17*(4), 297–338. https://doi.org/10.1080/026432900380427

Cooper, R. P., & Shallice, T. (2006). Hierarchical schemas and goals in the control of sequential behavior. *Psychological Review*, *113*(4), Article 887. https://doi.org/10.1037/0033-295X.113.4.887

de Waal, F. B. M., & Ferrari, P. F. (2010). Towards a bottom-up perspective on animal and human cognition. *Trends in Cognitive Sciences*, *14*(5), 201–207. https://doi.org/10.1016/j.tics.2010.03.003

Everaert, M. B. H., Huybregts, M. A. C., Chomsky, N., Berwick, R. C., & Bolhuis, J. J. (2015). Structures, not strings: Linguistics as part of the cognitive sciences. *Trends in Cognitive Sciences*, *19*(12), 729–743. https://doi.org/10.1016/j.tics.2015.09.008

Fadiga, L., Craighero, L., & D'Ausilio, A. (2009). Broca's Area in language, action, and music. *Annals of the New York Academy of Sciences*, *1169*(1), 448–458. https://doi.org/10.1111/j.1749-6632.2009.04582.x

Fazio, P., Cantagallo, A., Craighero, L., D'Ausilio, A., Roy, A. C., Pozzo, T., Calzolari, F., Granieri, E., & Fadiga, L. (2009). Encoding of human action in Broca's area. *Brain*, *132*(7), 1980–1988. https://doi.org/10.1093/brain/awp118

Fiebach, C. J., & Schubotz, R. I. (2006). Dynamic anticipatory processing of hierarchical sequential events: A common role for broca's area and ventral premotor cortex across domains? *Cortex*, *42*(4), 499–502. https://doi.org/10.1016/S0010-9452(08)70386-1

Fischmeister, F. P., Martins, M. J. D., Beisteiner, R., & Fitch, W. T. (2017). Self-similarity and recursion as default modes in human cognition. *Cortex*, *97*, 183–201. https://doi.org/10.1016/j.cortex.2016.08.016

Fitch, W. T. (2010). Three meanings of "recursion": Key distinctions for biolinguistics. In H. Yamakido, R. K. Larson, & V. Déprez (Eds.), *The* *evolution of human language: Biolinguistic perspectives* (pp. 73–90). Cambridge University Press. https://doi.org/10.1017/CBO9780511817755.005

Fitch, W. T., & Martins, M. D. (2014). Hierarchical processing in music, language, and action: Lashley revisited. *Annals of the New York Academy of Sciences*, *1316*(1), 87–104. https://doi.org/10.1111/nyas.12406

Fujita, K. (2014). Recursive Merge and human language evolution. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. 243–264). Springer.

Fujita, K. (2017). On the parallel evolution of syntax and lexicon: A Merge-only view. *Journal of Neurolinguistics*, *43*, 178–192. https://doi.org/10.1016/j.jneuroling.2016.05.001

Fukui, N. (2011). Merge and Bare phrase structure. In C. Boeckx (Ed.), *The Oxford handbook of linguistic minimalism* (pp. 73–95). Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199549368.013.0004

Fukui, N., & Zushi, M. (2004). Introduction. In N. Chomsky (Ed.), *The generative enterprise revisited: Discussions with Riny Huybregts, Henk van Riemsdijk, Naoki Fukui and Mihoko Zushi* (pp. 1–25). Walter de Gruyter.

Graves, P. (1994). Flakes and ladders: What the archaeological record cannot tell us about the origins of language. *World Archaeology*, *26*(2), 158–171. https://doi.org/10.1080/00438243.1994.9980270

Greenfield, P. M. (1991). Language, tools and brain: The ontogeny and phylogeny of hierarchically organized sequential behavior. *Behavioral and Brain Sciences*, *14*(4), 531–551. https://doi.org/10.1017/S0140525X00071235

Guest, O., & Martin, A. E. (2021). How computational modeling can force theory building in psychological science. *Perspectives on Psychological Science*, *16*(4), 789–802. https://doi.org/10.1177/1745691620970585

Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science*, *298*(5598), 1569–1579. https://doi.org/10.1126/science.298.5598.1569

Hauser, M. D., Yang, C., Berwick, R. C., Tattersall, I., Ryan, M. J., Watumull, J., Chomsky, N., & Lewontin, R. C. (2014). The mystery of language evolution. *Frontiers in Psychology*, *5*, Article 401. https://doi.org/10.3389/fpsyg.2014.00401

Higuchi, S., Chaminade, T., Imamizu, H., & Kawato, M. (2009). Shared neural correlates for language and tool use in Broca's area. *NeuroReport*, *20*(15), 1376–1381. https://doi.org/10.1097/WNR.0b013e3283315570

Holloway, R. L. (1969). Culture: A human domain. *Current Anthropology*, *10*(4), 395–412. https://doi.org/10.1086/201036

Hornstein, N. (2009). *A theory of syntax: Minimal operations and universal grammar*. Cambridge University Press.

Hornstein, N. (2017). On merge. In J. McGilvray (Ed.), *The Cambridge companion to Chomsky* (2nd ed., pp. 69–86). Cambridge University Press. https://doi.org/10.1017/9781316716694.004

Humphreys, G. W., & Forde, E. M. E. (1998). Disordered action schema and action disorganisation syndrome. *Cognitive Neuropsychology*, *15*(6–8), 771–811.

Jackendoff, R. (1977). *X' syntax: A theory of phrase structure*. MIT Press.

Jackendoff, R. (2002). *Foundations of language: Brain, meaning, grammar, evolution*. Oxford University Press.

Jackendoff, R. (2007). *Language, consciousness, culture: Essays on mental structure*. MIT Press.

Jackendoff, R. (2009). Parallels and nonparallels between language and music. *Music Perception*, *26*(3), 195–204. https://doi.org/10.1525/mp.2009.26.3.195

Jackendoff, R. (2011). What is the human language faculty? Two views. *Language*, *87*(3), 586–624. https://doi.org/10.1353/lan.2011.0063

Jackendoff, R., & Pinker, S. (2005). The nature of the language faculty and its implications for evolution of language (Reply to Fitch, Hauser, and Chomsky). *Cognition*, *97*(2), 211–225. https://doi.org/10.1016/j.cognition.2005.04.006

Jeon, H.-A. (2014). Hierarchical processing in the prefrontal cortex in a variety of cognitive domains. *Frontiers in Systems Neuroscience*, *8*, Article 223. https://doi.org/10.3389/fnsys.2014.00223

Joshi, A. K., & Schabes, Y. (1997). Tree-adjoining grammars. In G. Rozenberg & A. Salomaa (Eds.), *Handbook of formal languages: Volume 3 beyond words* (pp. 69–123). Springer. https://doi.org/10.1007/978-3-642-59126-6_2

Kayne, R. S. (1994). *The antisymmetry of syntax*. MIT Press.

Kayne, R. S. (2011). *Why are there no directionality parameters?* [Conference session]. Proceedings of the 28th West Coast Conference on Formal Linguistics, Somerville, Massachusetts, United States.

Koechlin, E., & Jubault, T. (2006). Broca's area and the hierarchical organization of human behavior. *Neuron*, *50*(6), 963–974. https://doi.org/10.1016/j.neuron.2006.05.017

Kuperberg, G. R. (2020). Tea with milk? A hierarchical generative framework of sequential event comprehension. *Topics in Cognitive Science*, *13*(1), 256–298. https://doi.org/10.1111/tops.12518

Lashley, K. S. (1951). The problem of serial order in behavior. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behavior* (pp. 112–131). Wiley.

Lasnik, H. (2000). *Syntactic structures revisited: Contemporary lectures on classic transformational theory*. MIT Press.

Maffongelli, L., D'Ausilio, A., Fadiga, L., & Daum, M. M. (2019). The ontogenesis of action syntax. *Collabra: Psychology*, *5*(1), Article 21. https://doi.org/10.1525/collabra.215

Marcus, G. F. (2006). Cognitive architecture and descent with modification. *Cognition*, *101*(2), 443–465. https://doi.org/10.1016/j.cognition.2006.04.009

Martin, A. E. (2016). Language processing as cue integration: Grounding the psychology of language in perception and neurophysiology. *Frontiers in Psychology*, *7*, Article 120. https://doi.org/10.3389/fpsyg.2016.00120

Martin, A. E. (2020). A compositional neural architecture for language. *Journal of Cognitive Neuroscience*, *32*(8), 1407–1427. https://doi.org/10.1162/jocn_a_01552

Martins, M. D. (2012). Distinctive signatures of recursion. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *367*(1598), 2055–2064. https://doi.org/10.1098/rstb.2012.0097

Martins, M. D., Bianco, R., Sammler, D., & Villringer, A. (2019). Recursion in action: An fMRI study on the generation of new hierarchical levels in motor sequences. *Human Brain Mapping*, *40*(9), 2623–2638. https://doi.org/10.1002/hbm.24549

Matchin, W., & Hickok, G. (2020). The cortical organization of syntax. *Cerebral Cortex*, *30*(3), 1481–1498. https://doi.org/10.1093/cercor/bhz180

Mazurkiewicz, A. (1995). Introduction to trace theory. In V. Diekert & G. Rozenberg (Eds.), *The book of traces* (pp. 3–41). World Scientific. https://doi.org/10.1142/9789814261456_0001

McRae, K., Brown, K. S., & Elman, J. L. (2019). Prediction-based learning and processing of event knowledge. *Topics in Cognitive Science*, *13*(1), 206–223. https://doi.org/10.1111/tops.12482

Miller, G. A., Galanter, E., & Pribram, K. H. (1960). *Plans and the structure of behavior*. Holt, Rinehart and Winston.

Moro, A. (2014a). On the similarity between syntax and actions. *Trends in Cognitive Sciences*, *18*(3), 109–110. https://doi.org/10.1016/j.tics.2013.11.006

Moro, A. (2014b). Response to Pulvermüller: The syntax of actions and other metaphors. *Trends in Cognitive Sciences*, *18*(5), Article 221. https://doi.org/10.1016/j.tics.2014.01.012

Moro, A. (2015). *The boundaries of Babel: The brain and the enigma of impossible languages*. MIT Press.

Norman, D. A. (1981). Categorization of action slips. *Psychological Review*, *88*(1), 1–15. https://doi.org/10.1037/0033-295X.88.1.1

O'Donnell, T. J., Hauser, M. D., & Fitch, W. T. (2005). Using mathematical models of language experimentally. *Trends in Cognitive Sciences*, *9*(6), 284–289. https://doi.org/10.1016/j.tics.2005.04.011

Pagin, P., & Westerståhl, D. (2010). Compositionality I: Definitions and variants. *Philosophy Compass*, *5*(3), 250–264. https://doi.org/10.1111/j.1747-9991.2009.00228.x

Papitto, G., Friederici, A. D., & Zaccarella, E. (2020). The topographical organization of motor processing: An ALE meta-analysis on six action domains and the relevance of Broca's region. *NeuroImage*, *206*, Article 116321. https://doi.org/10.1016/j.neuroimage.2019.116321

Partee, B. H. (1995). Lexical semantics and compositionality. In L. R. Gleitman & M. Liberman (Eds.), *Language: An invitation to cognitive science* (Vol. 1, 2nd ed., pp. 311–360). The MIT Press.

Partee, B. H., ter Meulen, A., & Wall, R. E. (1993). *Mathematical methods in linguistics* (Vol. 30). Kluwer Academic Publishers.

Pulvermüller, F. (2014). The syntax of action. *Trends in Cognitive Sciences*, *18*(5), 219–220. https://doi.org/10.1016/j.tics.2014.01.001

Pulvermüller, F., & Fadiga, L. (2010). Active perception: Sensorimotor circuits as a cortical basis for language. *Nature Reviews Neuroscience*, *11*(5), 351–360. https://doi.org/10.1038/nrn2811

Reason, J. T. (1979). Actions not as planned. In G. Underwood & R. Stevens (Eds.), *Aspects of consciousness* (pp. 67–90). Academic Press.

Reinhart, T. (1983). *Anaphora and semantic interpretation*. Croom Helm.

Rizzi, L. (2013). Introduction: Core computational principles in natural language syntax. *Lingua*, *130*, 1–13. https://doi.org/10.1016/j.lingua.2012.12.001

Rosenbaum, D. A., Cohen, R. G., Jax, S. A., Weiss, D. J., & van der Wel, R. (2007). The problem of serial order in behavior: Lashley's legacy. *Human Movement Science*, *26*(4), 525–554. https://doi.org/10.1016/j.humov.2007.04.001

Saito, M., & Fukui, N. (1998). Order in phrase structure and movement. *Linguistic Inquiry*, *29*(3), 439–474. https://doi.org/10.1162/002438998553815

Schwartz, M. F. (2006). The cognitive neuropsychology of everyday action and planning. *Cognitive Neuropsychology*, *23*(1), 202–221. https://doi.org/10.1080/02643290500202623

Steedman, M. (2000). *The syntactic process*. MIT press.

Steedman, M. (2002). Plans, affordances, and combinatory grammar. *Linguistics and Philosophy*, *25*(5), 723–753. https://doi.org/10.1023/A:1020820000972

Stout, D., & Chaminade, T. (2009). Making tools and making sense: Complex, intentional behaviour in human evolution. *Cambridge Archaeological Journal*, *19*(1), 85–96. https://doi.org/10.1017/S0959774309000055

Tettamanti, M., & Moro, A. (2012). Can syntax appear in a mirror (system)? *Cortex*, *48*(7), 923–935. https://doi.org/10.1016/j.cortex.2011.05.020

Tettamanti, M., & Weniger, D. (2006). Broca's area: A supramodal hierarchical processor? *Cortex*, *42*(4), 491–494. https://doi.org/10.1016/S0010-9452(08)70384-8

Uddén, J., & Bahlmann, J. (2012). A rostro-caudal gradient of structured sequence processing in the left inferior frontal gyrus. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *367*(1598), 2023–2032. https://doi.org/10.1098/rstb.2012.0009

Uithol, S., van Rooij, I., Bekkering, H., & Haselager, P. (2012). Hierarchies in action and motor control. *Journal of Cognitive Neuroscience*, *24*(5), 1077–1086. https://doi.org/10.1162/jocn_a_00204

van Rooij, I., & Blokpoel, M. (2020). Formalizing verbal theories. *Social Psychology*, *51*(5), 285–298. https://doi.org/10.1027/1864-9335/a000428

Watumull, J., Hauser, M. D., Roberts, I. G., & Hornstein, N. (2014). On recursion. *Frontiers in Psychology*, *4*, Article 1017. https://doi.org/10.3389/fpsyg.2013.01017

Zaccarella, E., Papitto, G., & Friederici, A. D. (2021). Language and action in Broca's area: Computational differentiation and cortical segregation. *Brain and Cognition*, *147*, Article 105651. https://doi.org/10.1016/j.bandc.2020.105651

Zacks, J. M., & Tversky, B. (2001). Event structure in perception and conception. *Psychological Bulletin*, *127*(1), Article 3. https://doi.org/10.1037/0033-2909.127.1.3

Received July 28, 2021
Revision received January 27, 2023
Accepted February 16, 2023 ■