Finding hierarchical structure in binary sequences: evidence from Lindenmayer grammar learning

COGNITIVE SCIENCE
A Multidisciplinary Journal

Cognitive Science 47 (2023) e13242

ISSN: 1551-6709 online
DOI: 10.1111/cogs.13242

Finding Hierarchical Structure in Binary Sequences: Evidence from Lindenmayer Grammar Learning

Samuel Schmid,[a] Douglas Saddy,[b] Julie Franck[a]

[a]*Faculty of Psychology, University of Geneva*
[b]*Centre for Integrative Neuroscience and Neurodynamics, University of Reading*

Received 22 February 2022; received in revised form 15 December 2022; accepted 4 January 2023

Abstract

In this article, we explore the extraction of recursive nested structure in the processing of binary sequences. Our aim was to determine whether humans learn the higher-order regularities of a highly simplified input where only sequential-order information marks the hierarchical structure. To this end, we implemented a sequence generated by the Fibonacci grammar in a serial reaction time task. This deterministic grammar generates aperiodic but self-similar sequences. The combination of these two properties allowed us to evaluate hierarchical learning while controlling for the use of low-level strategies like detecting recurring patterns. The deterministic aspect of the grammar allowed us to predict precisely which points in the sequence should be subject to anticipation. Results showed that participants' pattern of anticipation could not be accounted for by "flat" statistical learning processes and was consistent with them anticipating upcoming points based on hierarchical assumptions. We also found that participants were sensitive to the structure constituency, suggesting that they organized the signal into embedded constituents. We hypothesized that the participants built this structure by merging recursively deterministic transitions.

Keywords: Hierarchical representations; L-systems; Artificial grammar learning; Serial reaction times; Self-similarity; Fibonacci; Recursion; Nested structure

Correspondence should be sent to Samuel Schmid, Faculty of Psychology, University of Geneva, 1205 Geneva, Switzerland. E-mail: samuel.schmid@unige.ch

1. Introduction

How do humans extract hierarchical structure from a sequentially presented input? This question lies at the core of multiple domains of cognitive psychology and neuroscience. The most prominent is probably language processing where most linguistic theories assume that the sequences that humans produce and remember cannot be reduced to mere associations of consecutive items but must be mentally represented as recursively nested structures (Chomsky, 1957; Lashley, 1951; Simon, 1962). Nested tree structure is a form of representation generated by symbolic rules allowing recursion when they are embedded such that the same element can appear at multiple levels.

There's a plethora of evidence that nested structures are represented and used by adults in sentence processing (e.g., Lewis & Phillips, 2015) as well as in other cognitive domains like mathematical expressions (Maruyama, Pallier, Jobert, Sigman, & Dehaene, 2012; Monti, Parsons, & Osherson, 2012; Nakai & Sakai, 2014), motor action (Hunt & Aslin, 2001; Martins, Bianco, Sammler, & Villringer, 2019), musical melody (Koelsch, 2005), and rhythm (Fitch & Martins, 2014; Kotz, Ravignani, & Fitch, 2018). Nevertheless, the experimental demonstration of the learning of nested structures in sequence processing has proven difficult, and the field of artificial grammar learning (AGL) has produced very few empirical studies showing conclusive evidence (Fitch, 2014; Honing & Zuidema, 2014; Kovács & Endress, 2014; Levelt, 2019).

This difficulty comes from the fact that in the test cases classically used, a sequence can be processed without necessarily building a nested structure as other, possibly simpler ways of representing it can give rise to similar learning performance. Dehaene, Meyniel, Wacongne, Wang, and Pallier (2015) proposed a taxonomy of the different types of internal representations that can be generated from a sequence. In particular, they distinguish between two kinds of hierarchical representations: nested representations and algebraic patterns. Algebraic patterns refer to a type of representation where the input is coded as sequential abstract relationships or categories, thus allowing generalization to new exemplars irrespective of their specific identity. For example, the pseudowords "*duduba*" and "*pipiro*" share the algebraic pattern AAB that can be coded as a repetition followed by an alternation. Marcus, Vijayan, Rao, and Vishton (1999) showed that at 7 months, children were able to generalize this pattern to new unseen pseudowords, suggesting that they had a representation of the AAB rule. Algebraic patterns are hierarchical in the sense that they consist in variables that can take different values. Nevertheless, patterns, even though abstract, are insufficient to account for complex structural dependencies that characterize natural languages, like the subject–verb agreement dependency. For example, in the sentence "[The cats [the car avoided] ran away]" the plural subject (cats) agrees with the verb (ran) irrespective of the intervention of the relative clause (the car avoided). Long-distance dependencies in natural language are impossible to express with a system that only captures local order relations because arbitrarily large materials can intervene between the subject and the verb. In other words, the nesting of constituents where (cats) and (ran) are directly linked is necessary to account for long-distance dependencies.

Many AGL attempts to study the learning of nested structures have focused on the ability to learn and generalize center-embedding (de Vries, Petersson, Geukes, Zwit-

serlood, & Christiansen, 2012; Friederici, Bahlmann, Heim, Schubotz, & Anwander, 2006; Lai & Poletiek, 2011, 2013; Mueller, Bahlmann, & Friederici, 2010). Center-embedding is the nesting of an arbitrary number of phrases into higher-order phrases (e.g., [The cat [the dog chased] ran away]). Context-free grammars (CFG) represent the minimal level in the Chomsky hierarchy because of their unbounded memory that allows the binding of an unlimited number of constituents (Chomsky & Lightfoot, 2002). A well-studied instance of CFG is the a(n)b(n) grammar that generates strings like AB, A[AB]B, A[A[AB]B]B, and so forth. In order to assess if recursion can be induced by participants after exposure to sequences generated from this grammar, the test contrast is provided by strings generated from a finite state grammar (FSG) like (ab)n. FSGs cannot generate center-embedding because they have no memory; transitions are determined by the current state and the input only. They are therefore unable to describe nested structures. Fitch and Hauser (2004) compared in a habituation/discrimination task the ability of humans and cotton-top tamarins to discriminate between the a(n)b(n) grammar and the (ab)n grammar. The authors discovered that humans were able to notice the change from one grammar to the other, while cotton-top tamarins were not able to discriminate (ab)n from a(n)b(n) after training on a(n)b(n). The results were interpreted as evidence that humans possess a unique ability to induce the hierarchical structure needed to process CFG, while cotton-top tamarins are limited to the processing of less complex grammars.

However, the conclusion that participants can represent a(n)b(n) as a nested structure has been challenged. Perruchet and Rey (2005) noted that it was not necessary to pair As and Bs to discriminate between the two kinds of test strings; a simpler strategy based on counting and detection of repetition could also explain performance. They showed that participants were unable to pair As and Bs in structures involving mirror recursion (center-embedding with systematic pairing of As and Bs that generate strings such as A3[A2[A1B1]B2]B3). Although later studies reported successful learning of mirror recursion under specific conditions (Bahlmann & Friederici, ; de Vries, Monaghan, Knecht, & Zwitserlood, 2008, 2012), the authors of these studies all acknowledged that the processing of surface distinctions could also account for performance. This comes from the fact that the ungrammatical test strings necessarily differ in their surface expression from the grammatical string: The correct rejection of an ungrammatical string can therefore also be due to the representation of those surface properties.

Recent work has used fractal stimuli to explore hierarchical processing in the visual modality (Martins et al., 2014; Martins et al., 2019 ; Martins, de, Muršič, Oh, & Fitch, 2015), the auditory modality (Martins et al., 2020; Martins, Gingras, Puig-Waldmueller, & Fitch, 2017), and in the motor domain (Martins et al., 2019). In this series of studies, participants were performing a completion task on periodic fractals. For example, in Martins et al. (2017), participants were first exposed to three auditory stimuli that were generated by the application of a recursive rule. Participants were then asked to choose between two stimuli from the one that followed the rule at the higher hierarchical level. Each application of the rule added a hierarchical level to the existing stimulus. Each hierarchical level consisted of three notes that formed an ascending contour. The application of the recursive rule superimposed on each note of the preceding level three shorter higher pitch notes that also formed an ascending contour.

For example, the first stimulus was a low-pitch note with a duration of 7.3 s (Level 1). The second stimulus (Level 1 + Level 2) superimposed three shorter medium-pitch notes on the low-pitch note of Level 1. The third stimulus (Level 1 + Level 2 + Level 3) superimposed nine shorter high-pitch notes on each of the medium-pitch notes of Level 2. The authors found that participants were able to select the correct continuation when presented along with different foils and interpreted this result as an indication that participants were able to apply rules to new hierarchical levels. However, these results do not demonstrate that rules were embedded because it was sufficient to apply the rule only to the highest hierarchical level to solve the task. Indeed, a rule of the type "the notes follow an ascending pattern" was enough to reject the foils because Level 3 of each foil violated this rule. In other words, it was not necessary to apply the rule simultaneously at all the hierarchical levels to succeed.

As we have seen, it has proven challenging to create foils that allow to distinguish between learning based on surface regularities from learning based on higher-order structural properties in the habituation/discrimination paradigm. Furthermore, the presentation of ungrammatical strings may contaminate participants' mental representations throughout the testing phase. To avoid these difficulties, one should be able to assess learning without having to present ungrammatical strings to participants. To this end, the grammar should generate sequences in which the learning of one regularity is conditioned by the learning of another, lower-level regularity. This makes it possible to evaluate the depth of learning by comparing which regularities the learner has identified. Assessing learning of such a grammar that contains its own test can be done with a procedure that measures the evolution of performance throughout the task, avoiding the use of ungrammatical strings and explicit grammaticality judgments. The serial reaction time (SRT) paradigm (Nissen & Bullemer, 1987) allows such on-line monitoring of the participants' learning performance. In the SRT task, participants respond as quickly as possible to successively presented stimuli, usually by pressing response keys. Each response triggers the presentation of the next stimulus, to which participants respond anew. Learning typically manifests by a reduction in reaction times and is expected to take place when a given trial is subject to anticipation.

Only a few studies have made use of this paradigm to explore the learning of hierarchical structure, and for most of them, the kind of knowledge developed by participants involves algebraic patterns and not nested structures. Koch and Hoffmann (2000) were the first to report evidence suggesting sensitivity to higher-order properties of sequences in SRT. Participants were presented with sequences consisting of six different digits. The sequences were periodic and 24 digits in length. The participants' task was to respond to the digit presented on the screen with one of the six response keys. The authors manipulated the relational structure of the sequences. In the third experiment, the highly structured sequences were composed of four pairs of three elements that followed two relational patterns. The first two pairs corresponded to a mirror relationship of an ascending and descending order (e.g., 123–321), and the last two pairs corresponded to a transposition (e.g., 123–234). Participants in this condition therefore saw a sequence like 123–321–456–654–123–234–345–456 (e.g., mirror, mirror, transposition, transposition). The unstructured sequences were created by the permutation of the triplets in such a way as to break the relational patterns while keeping the statistical distribution identical (e.g., 123–345–456–123–234–321–456–654). The results showed

a greater decrease in reaction times for participants in the structured than in the unstructured condition, suggesting that they were sensitive to the sequences' higher-order relational structure. The participants thus went beyond the surface statistical properties and seem to have organized the sequence according to relational patterns. However, an algebraic rule like "two mirror relations followed by two transposition relations" is actually sufficient to account for the results: It is thus not necessary to assume that the representation developed by the participants corresponds to a nested structure in which an algebraic rule is nested within another algebraic rule since the relational patterns were not embedded in multiple levels.

In a slightly different task, the discrete sequence production task, Verwey and Wright (2014) trained participants by repeatedly presenting them with short sequences of six elements, each associated with the location of an illuminated square. During training, each sequence was presented with one of the six elements positioned in a random location, while all other elements occupied a position following a pattern, which could not be extracted from a single sequence but required combining positional information across sequences. In the test phase, participants were presented with the sequence without deviations (i.e., the "true" but never seen sequence) as well as an unfamiliar sequence (i.e., a sequence where the order of elements never matched the training phase). Participants were faster in the no-deviation sequence than in the unfamiliar sequence, although they did not practice either during the training phase. This suggests that during the training phase, participants extracted probabilities related to the order of appearance (i.e., the probability that an element appears in Position 1, Position 2, etc.) and combined that information into a representation capturing the underlying pattern of the sequence. Although those results demonstrate learning of an algebraic pattern, like in the study of Koch and Hoffmann (2000), they do not attest to learning of nested structures.

To our knowledge, only one SRT study reported results suggesting the use of nested structures, which is that from Hunt and Aslin (2001). These authors presented probabilistic sequences in a visual SRT task. The sequences were presented by illuminating buttons occupying different spatial positions. In their Experiment 3, the sequence consisted of four pairs of elements where the transitional probability from the first to the second element was 1, so the second element of a pair could always be anticipated with certainty by the participants. On the other hand, the transition between pairs was governed by the following probabilities: Pairs A and B were each followed in 50% of the cases by pair C and in the remaining 50% by pair D. Pairs C and D were each followed in 25% of the cases by pair A and in 25% of the cases by pair B. Pair C was followed in 50% of the cases by pair D and pair D in 50% of the cases by pair C. An additional restriction was that when pairs C and D were contingent, the next pair had to be either A or B (thus prohibiting alternating CDC or DCD). The authors observed that some participants became sensitive to the cumulative probability of the two most frequent pairs. When pairs C and D were contingent, reaction times for the second element of the pair in Position 2 were faster than those for the second item of the same pair when it was in Position 1. Since the transitional probability was always 1 for the second element of a pair, the effect can be explained only if participants have acquired the knowledge that the transition between elements of a pair is embedded in the transition between pairs. This embedding of transition seems more in line with a nested representation than a representation

of an algebraic pattern; however, this interpretation has some limitations. First, only three participants out of 10 showed the effect. Second, the alternation CDC and DCD being prohibited, the transition following CD or DC was at chance level (50% A and 50% B). Thus, the design of the materials prevented determining if participants nested more than one relation, that is, if the transitions between pairs were themselves embedded into transitions between multiple pairs. Nevertheless, the results suggest that transitional information is sufficient to bootstrap the construction of nested representations.

In a recent study, Planton et al. (2021) went further and explored if a simple form of temporal sequence could give rise to nested representations. One of the simplest forms of temporal sequences is binary sequences, and unlike more complex sequences like music or natural language, they have the advantage of allowing maximal control of the input presented to the participants. This apparent simplicity however preserves the possibility of creating highly complex sequences, which can be expressed as nested tree structures. The authors presented short binary sequences in a violation detection task. After an exposure phase, altered sequences that deviated by one item from the initial sequences were presented to the participants. The participants' task was to report as quickly as possible if they detected a violation. In order to vary the complexity of the sequences, the authors developed a formal language containing a limited number of primitive instructions that could generate any binary sequence. This allowed them to characterize each binary sequence in terms of *Kolmogorov Complexity*. Kolmogorov complexity is a theoretical measure where the complexity of a sequence is equal to the size of the shortest computer program that can generate it. Thus, the complexity of a sequence was defined by the minimal number of primitive instructions needed to generate it in the proposed language. The more the complexity of a sequence increases, the more its most compressed representation requires the use of instruction nesting. The authors therefore wanted to know if the participants' sequence representations were compressed in a similar way. To separate the part of the performance explained by this compression process and the part that can be attributed to the learning of transitional probabilities, the authors also measured in each test sequence the *Shannon surprise* induced by the deviant stimuli. Shannon surprise (Shannon, 1948) measures the degree of uncertainty of observing an item given the history of previous items and thus reflects statistical learning. Since surprise is independent of complexity (it varies with the position of the deviant within a sequence and is insensitive to sequence complexity that characterizes a sequence as a whole), if participants process only the transitional probabilities of the sequences, the degree of surprise of the deviant stimuli should be the only predictor of performance. Conversely, the use of compression by participants should result in a significant portion of the variance being explained by the degree of complexity of the sequences. The results showed that both surprise and complexity were significant predictors of performance suggesting that compression occurred along with statistical learning. This finding demonstrates that statistical learning is insufficient to fully account for sequence processing: even when processing sequences as simple as binary sequences, participants recode the sequence using a recursive compression algorithm. However, this study did not assess the degree of compression of the participants. Indeed, sensitivity to complexity, demonstrated by slower violation detection times in the most complex sequences, does not imply that participants have compressed the sequence to the maximum nor that the primitive instructions of

their formal language correspond to the mental operations of the participants. Our study aims to go further by trying to characterize more precisely the mechanism used by the participants to compress the signal.

1.1. Present study

The purpose of the present study is to evaluate, with the SRT paradigm, if participants represent binary sequences of events as nested structures. In theory, recursive compression algorithms allow an infinite number of hierarchical levels. This is obviously not the case for humans whose processing capacity is finite, limiting the number of hierarchical levels it can represent. Nevertheless, this limit cannot be defined a priori and can vary from one participant to another. Thus, predefining in advance the hierarchical structure of a sequence and setting a maximum number of levels does not allow for finely evaluating the hierarchical depth reached by the participants. We avoided this problem by using sequences generated by the Fibonacci grammar that are self-similar and aperiodic. The investigation of hierarchical processing with sequences having these two properties has several advantages. First, the self-similar character of the sequences does not limit a priori the hierarchical depth, which is theoretically infinite.[1] Second, the aperiodic character of the sequences means that no matter how deep the hierarchical representations are, they will necessarily be incomplete and will only explain part of the signal. Thus, the part not explained by the hierarchical structure corresponds to the maximum hierarchical level reached. In this way, it is not necessary to compare performance between grammatical and ungrammatical stimuli because the learning is evaluated within the sequence. Crucially, the linear distribution of units (henceforth referred to as *points*) in the sequences is aperiodic, meaning that there is no linear function that can be used to linearly predict *when* a point will occur. This prevents the use of low-level strategies like detecting recurring patterns.

The sequences we will use are generated by a grammar derived from the Lindenmayer formalism (L-systems). These grammars show interesting properties: There is no distinction between rewriteable and non-rewriteable symbols, and rewrite rules apply simultaneously to all symbols[2] rather than sequentially from left to right in a string (Lindenmayer, 1968; Vitányi & Walker, 1978). Because L-systems do not distinguish rewriteable from non-rewriteable symbols, rule systems are simplified but still produce complex structural patterns. One instantiation of L-systems used in AGL paradigms is the so-called Fibonacci grammar, which consists of two rewrite rules (Geambaşu, Ravignani, & Levelt, 2016; Saddy, 2009; Shirley, 2014):

$$0 \rightarrow 1$$

$$1 \rightarrow 0\ 1.$$

The interpretation of such a formalism is very simple: Every instance of [0] in a sequence must be "rewritten as" [1], and every instance of [1] in the same sequence must be rewritten as [01]. Applying these rules over and over again generates longer and longer sequences of points, each of which corresponds to a "generation" of the grammar. The name of this

grammar comes from the fact that the number of points in each generation actually follows the Fibonacci sequence (Fig. 1c). Moreover, in each generation, the distribution of 0s and 1s is asymmetric, with more 1s than 0s: The ratio between the number of 1s and 0s approximates the golden ratio (1.618). If we consider a sequence (i.e., a string generated by the grammar) from left to right, two transitions are possible (from 0 and the next point and from 1 and the next point), and the probability of those transitions is also asymmetric. The transition from 0 to 1 is deterministic: 0 is always followed by 1. The transition from 1 to the next point is probabilistic: 1 is followed by 0 in 61.8% of the cases and by 1 in 38.2% of the cases.

The most important property of this grammar with respect to our research question is its self-similarity. Each generation of this grammar constitutes by definition a natural constituent (Krivochen et al., 2018). Because of the recursive nature of the generative process, any generation is the concatenation of the two previous generations (Fig. 1c). This means that any generation can be parsed with two consecutive smaller generations that are natural constituents of the grammar. For example, Generation 4 [01101] can be divided into Generations 2 and 3 [[01][101]], which can be further divided into Generations 1 and 2 [[01][1][01]], which can (trivially) be further divided into Generations 0 and 1 [[[0][1]][[1][[0][1]]]]. Thus, any generation can be seen as a multiple embedding of constituents reflecting the hierarchical structure of the grammar. Transitions in the Fibonacci grammar are scale-free: The transitional probabilities between points at the surface level are identical to the transitional probabilities between constituents (Fig. 1a, right panel). Crucially, points/constituents surrounding a deterministic transition at level n always form a bigger constituent at level $n + 1$. For example, at the surface level, 0 is always followed by 1, and the concatenation of these two points results in the higher-order constituent [01], which is a natural constituent of the grammar. At Level 1, the constituent [1] is always followed by the constituent [01], and their concatenation results in the higher-order constituent [101]. Thus, because of the grammar's self-similarity, transitional probabilities at each level provide the parser a way to access the constituent structure of the grammar. The processing mechanism may start by merging the points linked by a deterministic transition, and then use the output of this process, that is, the higher-order constituents, to detect the deterministic transitions at the next hierarchical level. This process of recursive combination would progressively transform the representation of the sequence into a complex hierarchical structure of embedded constituents (Fig. 1a, left panel).

This leads to an interesting observation: Points that follow a probabilistic transition at level n can appear inside a constituent that follows a deterministic transition at level $n + 1$. For example, all 0s follow a probabilistic transition at the surface level: $p(0|1) = 0.62$ and $1 - p(0|1) = 0.38$. However, 0s always appear at Level 1 in the constituent [01], and some instances of this constituent follow a higher-order deterministic transition: The constituent [1] is always followed by the constituent [01] ($p([01]|[1]) = 1$). Thus, although at the surface level, all 0s are ambiguous (i.e., they follow a probabilistic transition) a subset of them are *disambiguated* at Level 1 (i.e., the 0s that follow a higher-order deterministic transition). Therefore, the detection of higher-order deterministic transitions serves to disambiguate some of the points that were ambiguous at the lower level. Crucially, the higher the hierarchical structure is, the more ambiguous points will be disambiguated. Nevertheless, due to the aperiodicity of the string, there will always remain a subset of non-disambiguated points that

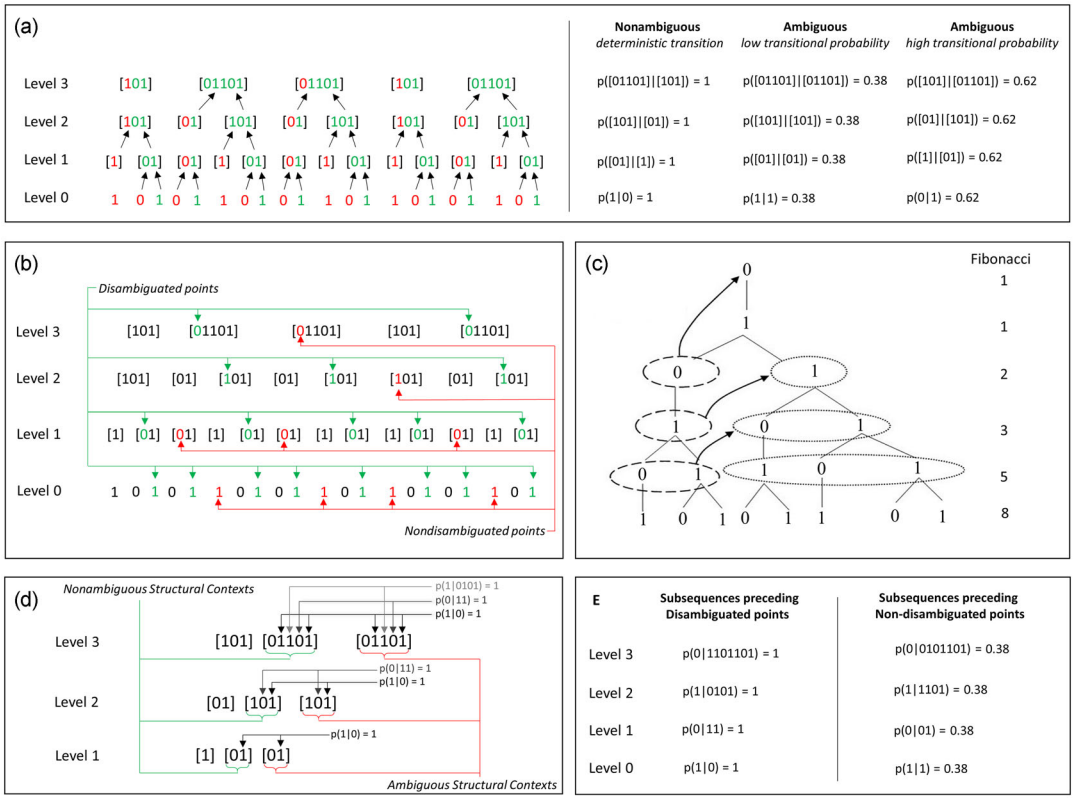

Fig. 1. (a) Left panel: depiction of the first three hierarchical levels of Generation 7 of the Fibonacci grammar. Non-disambiguated points at each level are highlighted in red and disambiguated points in green. To form a new hierarchical level, points that span across a deterministic transition are combined together (this is illustrated by the arrows). The result is a new representation of the string that consists in the combination of points corresponding to natural higher-order constituents of the grammar (illustrated by the brackets). At each level, constituents spanning a deterministic transition can be combined to form an embedded hierarchy. Right panel: transition probabilities between constituents at each level. (b) Disambiguated points (green) and non-disambiguated points (red) for each hierarchical level for Generation 7 of the Fibonacci grammar. In the present study, we used Generation 12 of the Fibonacci grammar, which consists of 233 points. We did not illustrate this generation due to space limitation, but the rationale is identical. (c) Derivation of the Fibonacci grammar for the first five generations. The right column shows the number of symbols at each generation, which maps the Fibonacci sequence. Arrows and circles highlight the hierarchical constituency of the grammar. (d) Structural contexts at Levels 1, 2, and 3. Green bars point to the constituents in non-ambiguous structural contexts at each level, and red bars point to the same constituents when in ambiguous structural contexts. Arrows illustrate the fact that, with the exception of the first point, points that occur inside constituents have the same transitional probability regardless if the constituent is in an ambiguous or non-ambiguous structural context. (e) Transitional probabilities for disambiguated and non-disambiguated points at each level given the subsequence that precedes them. We see that the transitional probability of the subsequence that precedes a disambiguated point is equal to 1, whereas the transitional probability of the subsequence that precedes a non-disambiguated point is equal to 0.38.

can lead to new embedding, no matter the depth of the hierarchy. Thus, each hierarchical level corresponds to a specific learning pattern of points: points that are still ambiguous at this level (i.e., non-disambiguated points) and points that are disambiguated at this level and lower levels.

Structural processing in the Fibonacci grammar has already been explored via the classical AGL paradigm (Geambaşu et al., 2016, 2020). However, these studies have run into the problem inherent to the habituation/discrimination paradigm of creating non-grammatical test strings that respect the surface properties of grammar. In a first study, Geambaşu et al. (2016) found that participants exposed to the Fibonacci grammar were unable to distinguish between grammatical and ungrammatical strings and attributed that failure to the fact that some of the foils were in fact Fib-grammatical (i.e., they were possible subsequences of the Fibonacci grammar). In a follow-up study using a different set of non-grammatical test strings, Geambaşu, Toron, Ravignani, and Levelt (2020) found that participants were able to discriminate them from grammatical strings and concluded that the grammar was successfully learned. However, closer inspection shows that 16 of the 18 foils contained the non-grammatical subsequence [01010], which is impossible in the Fibonacci grammar. Hence, participants may have rejected the foils on the basis of a low-level strategy without having learned the Fibonacci grammar. Two other studies (Vender, Krivochen, Phillips, Saddy, & Delfitto, 2019, 2020) explored the Fibonacci grammar by way of an SRT task: A sequence of blue and red dots generated by the Fibonacci grammar was presented to the participants whose task was to press the left or right button corresponding to the color of each dot. Sequences of dots were implemented in a Simon task: dots appeared to the left or to the right side of the screen, such that the colored dot sometimes appeared to the opposite side of the corresponding key. Such incongruent trials occurred every sixth trial. The Simon task was introduced to make the task less repetitive for participants. In the 2020 study, the authors added a final block within which the order of appearance of stimuli followed an alternative grammar called Skip, which has similar surface properties to Fib: 0 is always followed by 1 ($p(1|0) = 1$), the subsequence 11 is always followed by 0 ($p(0|11) = 1$) and the first order transitional probabilities are relatively similar: $p(0|1) = 0.73$ and $p(1|1) = 0.27$ but differ from the latter from a formal point of view. The authors proposed that within the Fibonacci grammar, the identification of certain points, called k-points, would allow the reconstruction of the local hierarchical structure of the sequence due to their specific structural status. Indeed, the distance between two k-points exactly mirrors the transitional probability of the minimal units of the sequence (see Krivochen et al., 2018, for a detailed explanation). Linearly, k-points are the last 1 of the 3-gram [011] and correspond to the constituent [1] of Level 1 (shown in Fig. 1a, left panel) whose transitional probability is $p(1|1) = 0.38$. In Skip, although the surface expression of the k-points is present (Skip has the 3-gram [011]), their identification would not allow the reconstruction of the local hierarchical structure because the distance between them does not mirror the statistical distribution of minimal units. In other words, in contrast to Fib, the self-similarity of Skip does not allow to extend the local statistical regularities at a higher hierarchical level. Vender et al. (2020) found faster processing for the last 1 of the 3-gram [011] in Fib blocks than in the Skip block. They interpreted this as

evidence that participants had granted a special status to k-points, suggesting that they partially reconstructed the hierarchical structure of the Fibonacci grammar.

However, a more detailed analysis of the sequences generated by the Skip grammar shows an inversion of the second-order transitional probabilities. In Skip, k-points have a second-order conditional probability of $p(1|01) = 0.36$, while in Fib, it is equal to $p(1|01) = 0.62$. Thus, the slower processing observed for the last 1 of the 3-gram [011] in Skip block could also be explained by participants becoming sensitive to the fact that 01 is more frequently followed by 0 than by 1. The effect can therefore also be explained by "*flat*" statistical learning processes. Moreover, the Simon task introduces a factor that occurs periodically (i.e., incongruent trials occurred every sixth trial); Fibonacci grammar being aperiodic, incongruent trials are not distributed evenly in the sequence, which makes the impact of this factor difficult to evaluate.

In the present study, we implemented Fibonacci sequences in an SRT task, thus avoiding the need to create non-grammatical Fib-strings (like in Geambaşu et al., 2016, 2020). In contrast to Vender et al. (2019), 2020), dots were presented in the center of the screen, to avoid the interfering congruency factor introduced by the Simon task. Importantly, we developed new analyses, substantially different from those conducted in these four papers, which allowed us to evaluate hierarchical learning within the Fibonacci grammar without having to compare the performance of participants to another grammar or to a random block. Sequence learning in the SRT task is traditionally assessed by inserting a so-called "transfer block" at the end of the experiment in which trials follow a random order or an alternative sequence. A slowdown in the transfer block relative to the block that precedes it is interpreted as indicating that participants have acquired the target sequence (Schwarb & Schumacher, 2012). However, when it comes to interpreting the origin of a slowdown in the transfer block, this methodology encounters the same limitation as the habituation/discrimination paradigm. The slowdown can be either due to a change in surface properties or to a change in more abstract properties. The use of the Fibonacci grammar aims precisely at avoiding this problem because it allows us to evaluate the learning during the processing without having to compare the performance to an alternative sequence. Our conceptual framework critically diverges from Vender et al. (2020) in that rather than hypothesizing that the parser extracts some formal properties of the Fibonacci grammar (k-points), we hypothesize that it proceeds through recursively merging points that span across deterministic transitions, and then using the output of this process to merge new deterministic transitions between groups of points, resulting in the progressive building of a hierarchical structure. Participants may also develop knowledge of formal properties of the Fibonacci grammar; however, this question is beyond the scope of the present study.

We carried out two analyses to assess whether participants built a hierarchical structure from the Fibonacci grammar through the recursive combination of points/constituents surrounding deterministic transitions. The first analysis (*Processing of hierarchical structure*) explored whether disambiguated points (i.e., points following a higher-order deterministic transition) were anticipated better than non-disambiguated points (i.e., points following a higher-order probabilistic transition). To this end, we compared reaction times and accuracy for points disambiguated at a particular hierarchical level to points not disambiguated at the

same level (Fig. 1b). Hierarchical processing should result in a larger decrease in reaction times and better accuracy for disambiguated points, compared to non-disambiguated points. We do not have any prior expectation with respect to how many levels the participants might reach. We will therefore evaluate each level successively until the effects disappear at the group level (see Fig. 1a, left panel, for levels descriptions). In order to control for frequency effects that could be due to the asymmetry of the sequence (1s being more frequent than 0s), we compared, for each hierarchical level, only 1s to 1s and 0s to 0s. Anticipating the results, we found evidence of learning at Levels 1, 2, and 3 but not at Level 4 (which is why this level is not presented in Fig. 1a,b).

The second analysis (*Processing of hierarchical constituency*) aimed at specifying further whether participants have processed the Fibonacci grammar as a nested structure. To this end, we explored the influence of the constituent structure at level n on the processing of disambiguated points at level $n - 1$. This analysis is a logical continuation of the first: If participants use deterministic transitions between constituents to anticipate disambiguated points, then the processing of a disambiguated point should depend not only on the level at which it is disambiguated but also on the constituent in which it appears higher in the hierarchy. If we examine closely the constituents of each level, we see that the first position (from left to right) is always occupied by either a disambiguated or a non-disambiguated point (Fig. 1a, left panel), whereas the following positions are composed of points disambiguated at the previous levels. Crucially, the remaining positions of the constituent following a deterministic transition and of the constituent following a low probabilistic transition (Fig. 1a, right panel) are occupied by points disambiguated at the same levels (Fig. 1d). In other words, a point disambiguated at level n can appear at level $n + 1$ in either a constituent that follows a deterministic transition or in a constituent that follows a probabilistic transition, while the composition of the constituents is identical (except for the point in the first position). Thus, the same disambiguated point appears higher in the hierarchy subsumed in a different structural context. We refer to the condition where a disambiguated point appears at a higher level inside a constituent that follows a deterministic transition as a *non-ambiguous structural context* and to the condition where a disambiguated point appears at a higher level in a constituent following a probabilistic transition as an *ambiguous structural context* (Fig. 1d). If the system is sensitive to the hierarchical constituency of the sequence, disambiguated points appearing at the upper level in a non-ambiguous structural context should be processed faster than the same disambiguated points appearing in an ambiguous structural context. Anticipating the results, we found a significant processing advantage for points occurring in non-ambiguous structural contexts, compared to points occurring in ambiguous structural contexts at Levels 1 and 3.

2. Methods

2.1. Participants

One hundred seventy-four students (33 men and 141 women; mean age 22.8 years old) participated in the experiment. They were recruited either from an introductory psycholinguistics

course from the University of Geneva or through announcements at the University of Geneva. All participants reported normal or corrected-to-normal vision.

2.2. Materials

The training sequence was composed of two elements and had a length of 50. The order was pseudo-randomized and elements had the same frequency. The training sequence included multiple non-grammatical subsequences such as 00 or 111. The longest Fib-grammatical subsequence had a length of 4. In the experimental blocks, the sequence consisted of Generation 12 of the Fibonacci grammar, which has 233 points. Each block corresponded to the full generation.

2.3. Design and procedure

Each trial consisted of a red or blue circle 100px in diameter presented at the center of the screen that correspond to 0 and 1 in a string generated by the Fib grammar. The circles disappeared after the response of the participant, or after 1200 ms, if no response was given. The response-to-stimulus interval lasted 500 ms. Participants were instructed to press as quickly as possible the button corresponding to the color of the circle they saw on the screen (X = blue, N = red). Keys X and N were chosen because they had a similar position on QWERTZ and AZERTY keyboards. No information about the grammar was given. The experiment started with a training block that was identical for all the participants. During the training block, when the participants made an error, the experiment stopped and a message appeared to remind them the color–key association, the experiment resumed after 3000 ms. In the experimental blocks, no message appeared when they made an error. After the training block, participants did five experimental blocks of 233 trials. The experiment was conducted online on the website Testable (https://www.testable.org/ Rezlescu, Danaila, Miron, & Amariei, 2020). Pre-testing showed that the error rate in the task was extremely low, which is not surprising given the simplicity of the task, so the emphasis on speed alone was intended to increase the error rate and avoid ceiling effects. Participants were asked to perform the experiment in a quiet environment where they could not be disturbed. Instructions were displayed on the screen, and participants had to click on a button to start the experiment. The experiment lasted approximately 25 min.

2.4. Data analyses

Four participants were removed due to technical failures. We also removed participants who had an error rate superior to 3 *SD* to the mean error rate in at least one block. This led to the removal of 11 additional participants. Due to an error in the experiment code, the data of the training block were not recorded. Reaction times and accuracy were both modeled as dependent variables. We removed from the analysis all the trials where participants did not respond after 1200 ms (699 trials). For the analysis of reaction times, only trials with a correct answer were included. Homoscedasticity and normality were checked by visual inspection of residual plots. Data from the remaining 159 participants were analyzed with linear

mixed-effects models as implemented in the lme4 package for R (Bates, Maechler, Bolker, & Walker, 2014; R Development Core Team, 2021).

For the analysis *Processing of hierarchical structure*, models included two fixed-effect factors and their interaction: *Exposure, Ambiguity*, and *Exposure*Ambiguity*. Because our predictions focus on reaction times (RT) slopes throughout the experiment, *Exposure* was treated as a continuous variable with a value of 0 for trials in the first experimental block, and 1, 2, 3, and 4 for trials in the second, third, fourth, and fifth blocks. Treating this factor as continuous allowed us to have a single estimate that represents the evolution (i.e., the slope) of performance throughout the experiment across all participants. Ambiguity is a discrete variable contrasting disambiguated and non-disambiguated points and operationalized differently depending on the level at which its effect is explored (it is labeled *Ambiguity level*$_n$ according to the level at which it has been operationalized). The modality "non-disambiguated" of the factor *Ambiguity level*$_n$ was always set as the intercept of the models. As random effects, the models had intercepts for *Participants*. *p*-values were calculated by way of the Satterthwaites's approximation to degrees of freedom with the lmerTest package (Kuznetsova, Brockhoff, & Christensen, 2015). We conducted separate analyses for RTs and accuracy instead of using a composite score because there is no consensus in the literature on the optimal method of calculation (Liesefeld & Janczyk, 2019; Vandierendonck, 2017, 2018). Moreover, composite measures that integrate RTs and accuracy cannot be calculated per trial but only per condition (for each participant). Since the factor *Ambiguity* is nested within blocks (i.e., each block contains several disambiguated and non-disambiguated points of the same level), using a composite score would drastically reduce the number of observations per participant and thus the statistical power of the analyses.

For the analysis *Processing of hierarchical constituency*, models included two fixed-effect factors and their interaction: *Exposure*, *Structural context*, and *Exposure*Structural context*. Structural context is a discrete variable contrasting disambiguated points that appeared at the next level in constituents that either followed a deterministic transition (non-ambiguous) or a probabilistic transition (ambiguous). This variable is operationalized differently depending on the level at which its effect is explored (it is labeled *Structural context level*$_n$ according to the level at which it has been operationalized). The same mixed models were ran as in previous analysis, including *Structural context* and *Exposure* as fixed factors, and the modality "Ambiguous" of the factor *Structural context level*$_n$ was always set as the intercept. Since at each level, the first point of a constituent is either a disambiguated or a non-disambiguated point, we excluded those first points when we computed the mean RTs and accuracy of the constituents (Fig. 1d). At Level 1, the constituent of interest is [01] and contains two points. This constituent is in a non-ambiguous structural context when it is preceded by [1] but in an ambiguous structural context when preceded by [01]. Since the first point of [01] can be either a disambiguated or a non-disambiguated point, we have included in the Level 1 analysis only the second point of constituent [01] (i.e., the 1). We excluded from the RT analyses all the constituents containing at least one error. At Level 2, the constituent of interest is [101] and contains three points. It is in a non-ambiguous structural context when it is preceded by [01] and in an ambiguous structural context when it is preceded by [101]. We have included in the analysis only the last two points of constituent [101] (i.e., 01) for the

same reason explained above. In the RT analysis, we first excluded all constituents containing at least one error. Insofar as the distribution of 0s and 1s is identical in each modality of the factor *Structural context* (i.e., there is exactly one 0 and one 1 in both the non-ambiguous and the ambiguous structural context at Level 2), there is no more asymmetry between the number of 0s and 1s. We thus calculated for each occurrence of the constituent [101] the mean of the last two points and took this measure as the dependent variable. For analyzing accuracy, we computed the mean number of correct answers for the last two points of [101] (i.e., the two disambiguated points that appeared in both structural contexts) and divided it by 2 in order to have a value that ranged from 0 to 1 (we did not consider the first point of [101] because it could either be disambiguated or non-disambiguated point depending on the structural context). At Level 2, the accuracy value for the constituent was either 1 (no error), 0.5 (1 error), or 0 (2 errors). At Level 3, the constituent of interest is [01101] and contains five points. It is in a non-ambiguous structural context when it is preceded by [101] and in an ambiguous structural context when it is preceded by [01101]. We have included in the analysis only the last four points of constituent [01101]. To analyze RTs, we first excluded all constituents containing at least one error. We then calculated for each occurrence of the constituent [01101] the mean of the last four points (i.e., 1101) and took this measure as the dependent variable. For analyzing accuracy, we followed the same logic as in Level 2 but with the constituent [01101]. We computed the mean number of correct answers for the four disambiguated points that appeared in both structural contexts and divided it by 4 in order to have a value that ranged from 0 to 1. At Level 3, the accuracy value for the constituent could either be 1 (no error), 0.75 (one error), 0.5 (two errors), 0.25 (three errors), or 0 (four errors).

We first explored if participants were sensitive to the surface statistical properties of the sequence, corresponding to Level 0, and then if they were able to detect the higher-order deterministic transitions at Levels 1–4 (see Fig. 1b). We then explored if participants were sensitive to the constituent structure of the grammar by comparing, at each level, disambiguated points occurring in different structural contexts (see Fig. 1d). Finally, we analyzed performance at the individual level to more finely explore the effect of structural context at Level 3 found at the group level.

3. Results

3.1. Processing of hierarchical structure

3.1.1. Processing of surface statistical regularities (Level 0)

Analyses of reaction times showed a main effect of *Exposure* ($\beta = -21.53$, $SE = 0.25$, $t = -87.23$, $p < .000$) with a mean reduction of reaction times of 86 ms from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_0$ ($\beta = -57.45$, $SE = 0.73$, $t = -78.52$, $p < .000$) with disambiguated points being faster than non-disambiguated ones by 57 ms. The interaction *Ambiguity level*$_0$* *Exposure* was also significant ($\beta = -14.16$, $SE = 0.50$, $t = -28.48$, $p < .000$) with a more important reduction over exposure for disambiguated points

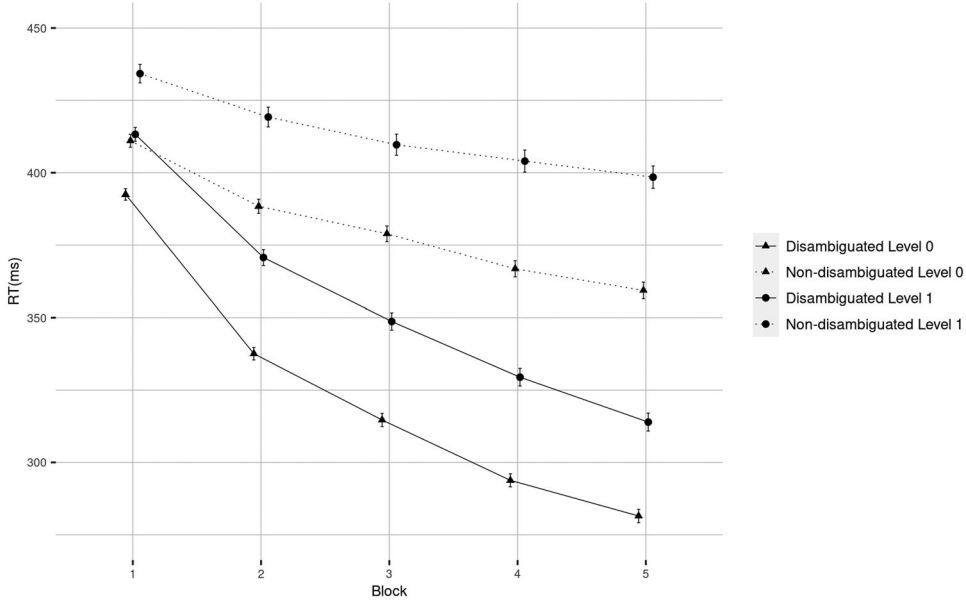

Fig. 2. Mean RT (ms) for disambiguated and non-disambiguated points of Hierarchical Levels 0 and 1 by block. Errors bars denote the 95% confidence interval.

($M_{block1\ -\ block5} = -106$ ms) than non-disambiguated points ($M_{block1\ -\ block5} = -49$ ms; $M_{block1\ -\ block5}$ indicates the mean difference between Blocks 1 and 5). Results are shown in Fig. 2.

Concerning accuracy, we found a main effect of *Exposure* ($\beta = -0.06$, $SE = 0.01$, $z = -6.302$, $p < .000$) with a mean reduction of accuracy of 1% from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_0$ ($\beta = 2.26$, $SE = 0.04$, $z = 57.80$, $p < .000$) with higher accuracy for disambiguated points ($M = 0.98$) than for non-disambiguated points ($M = 0.90$). The effect of *Exposure* significantly interacted with *Ambiguity level*$_0$ ($\beta = 0.23$, $SE = 0.03$, $z = 8.354$, $p < .000$) with accuracy increasing for disambiguated points over exposure ($M_{block1\ -\ block5} = 0.006$) and decreasing for non-disambiguated points ($M_{block1\ -\ block5} = -0.037$). Results are shown in Table 1.

3.1.2. Processing of hierarchical regularities (Levels 1–4)

3.1.2.1. Hierarchical processing at Level 1: Analyses of reaction times showed a main effect of *Exposure* ($\beta = -18.39$, $SE = 0.31$, $t = -59.45$, $p < .000$) with a mean reduction of reaction times of 73 ms from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_1$ ($\beta = -56.19$, $SE = 0.92$, $t = -61.31$, $p < .000$) with disambiguated points being faster than non-disambiguated ones by 56 ms. The interaction *Ambiguity level*$_1$* *Exposure* was also significant ($\beta = -15.20$, $SE = 0.64$, $t = -23.62$, $p < .000$) with a more important reduction over exposure for disambiguated points ($M_{block1\ -\ block5} = -95$ ms) than non-disambiguated points ($M_{block1\ -\ block5} = -34$ ms). Results are shown in Fig. 2.

Table 1
Mean proportion (*M*) and standard deviation (*SD*) of correct responses for disambiguated and non-disambiguated points by hierarchical levels and blocks

		Block 1		Block 2		Block 3		Block 4		Block 5	
		M	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Level 0	Disambiguated	0.98	0.13	0.99	0.09	0.99	0.11	0.99	0.10	0.99	0.09
	Non-disambiguated	0.93	0.26	0.91	0.28	0.90	0.30	0.89	0.31	0.89	0.31
Level 1	Disambiguated	0.96	0.20	0.96	0.19	0.96	0.19	0.97	0.18	0.97	0.17
	Non-disambiguated	0.91	0.28	0.89	0.32	0.87	0.34	0.86	0.35	0.86	0.35
Level 2	Disambiguated	0.92	0.27	0.91	0.28	0.9	0.30	0.9	0.30	0.9	0.24
	Non-disambiguated	0.94	0.23	0.90	0.30	0.89	0.31	0.88	0.32	0.88	0.33
Level 3	Disambiguated	0.91	0.28	0.89	0.32	0.87	0.33	0.87	0.34	0.87	0.34
	Non-disambiguated	0.91	0.28	0.89	0.32	0.88	0.34	0.84	0.36	0.85	0.36

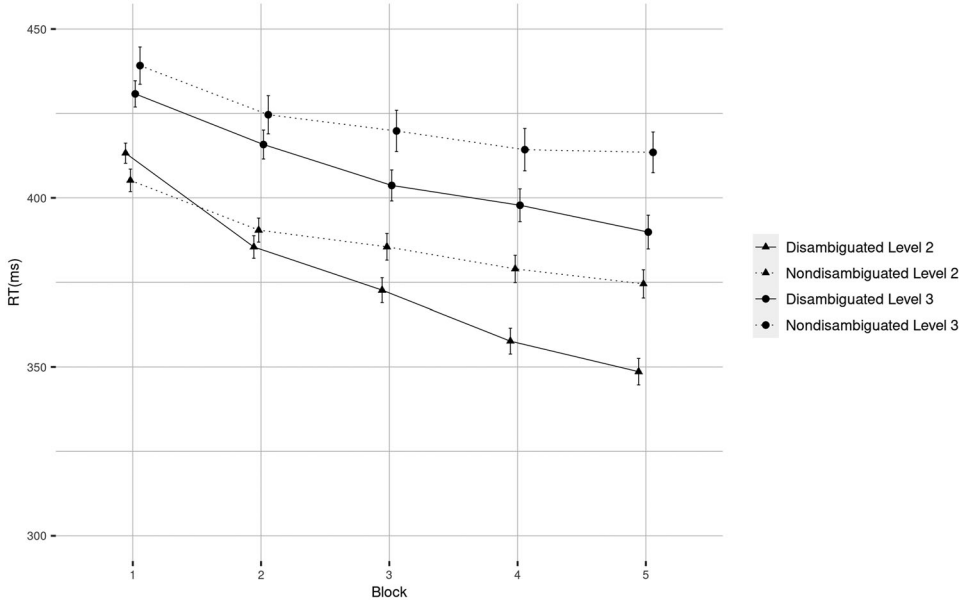

Fig. 3. Mean RT (ms) for disambiguated and non-disambiguated points of Hierarchical Levels 2 and 3 by block. Errors bars denote the 95% confidence interval.

Concerning accuracy, we found a main effect of *Exposure* ($\beta = -0.055$, $SE = 0.01$, $z = -5.022$, $p < .000$) with a mean reduction of accuracy of 1.2% from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_1$ ($\beta = 1.35$, $SE = 0.03$, $z = 41.903$, $p < .000$) with accuracy higher for disambiguated points ($M = 0.96$) than for non-disambiguated points ($M = 0.88$). The effect of *Exposure* significantly interacted with *Ambiguity level*$_1$ ($\beta = 0.20$, $SE = 0.02$, $z = 8.759$, $p < .000$) with accuracy increasing for disambiguated points over exposure ($M_{block1\ -\ block5} = 0.01$) and decreasing for non-disambiguated points ($M_{block1\ -\ block5} = -0.05$). Results are shown in Table 1.

3.1.2.2. Hierarchical processing at Level 2: Analyses of reaction times showed a main effect of *Exposure* ($\beta = -12.32$, $SE = 0.36$, $t = -34.036$, $p < .000$) with a mean reduction of reaction times of 49 ms from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_2$ ($\beta = -8.10$, $SE = 1.05$, $t = -7.693$, $p < .000$) with disambiguated points being faster than non-disambiguated ones by 8 ms. The interaction *Ambiguity level*$_2$* *Exposure* was also significant ($\beta = -7.75$, $SE = 0.74$, $t = -10.44$, $p < .000$) with a more important reduction over exposure for disambiguated points ($M_{block1\ -\ block5} = -61$ ms) than non-disambiguated points ($M_{block1\ -\ block5} = -31$ ms). Results are shown in Fig. 3.

Concerning accuracy, we found a main effect of *Exposure* ($\beta = -0.10$, $SE = 0.01$, $z = -8.905$, $p < .000$) with a mean reduction of accuracy of 3.5% from block 1 to block 5. There was also a main effect of *Ambiguity level*$_2$ ($\beta = 0.07$, $SE = 0.03$, $z = 2.195$, $p = .028$) with accuracy higher for disambiguated points ($M = 0.91$) than for non-disambiguated points (*M*

= 0.89). The effect of *Exposure* significantly interacted with *Ambiguity level*$_2$ ($\beta = 0.09$, $SE = 0.02$, $z = 3.811$, $p < .000$) with accuracy decreasing less for disambiguated points over exposure ($M_{block1 - block5}$ = -0.02) than for non-disambiguated points ($M_{block1 - block5}$ = -0.06). Results are shown in Table 1.

3.1.2.3. Hierarchical processing at Level 3: Analyses of reaction times showed a main effect of *Exposure* ($\beta = -8.60$, $SE = 0.50$, $t = -17.314$, $p < .000$) with a mean reduction of reaction times of 34 ms from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_3$ ($\beta = -12.86$, $SE = 1.47$, $t = -8.769$, $p < .000$) with disambiguated points being faster than non-disambiguated ones by 12 ms. The interaction *Ambiguity level*$_3$* *Exposure* was also significant ($\beta = -3.224$, $SE = 1.03$, $t = -3.120$, $p = .002$) with a more important reduction over exposure for disambiguated points ($M_{block1 - block5} = -38$ ms) than non-disambiguated points ($M_{block1 - block5} = -27$ ms). Results are shown in Fig. 3.

Concerning accuracy, we found a main effect of *Exposure* ($\beta = -0.13$, $SE = 0.01$, $z = -9.215$, $p < .000$) with a mean reduction of accuracy of 5.2% from Block 1 to Block 5. There was also a main effect of *Ambiguity level*$_3$ ($\beta = 0.09$, $SE = 0.04$, $z = 2.187$, $p = .029$) with accuracy higher for disambiguated points ($M = 0.88$) than for non-disambiguated points ($M = 0.87$). The interaction *Exposure** *Ambiguity level*$_3$ did not reach significance level ($\beta = 0.05$, $SE = 0.03$, $z = 1.651$, $p < .098$). Results are shown in Table 1.

3.1.2.4. Hierarchical processing at Level 4: Analyses of reaction times showed a main effect of *Exposure* ($\beta = -7.46$, $SE = 0.54$, $t = -13.911$, $p < .000$) with a mean reduction of reaction times of 30 ms from Block 1 to Block 5. There was no main effect of *Ambiguity level*$_4$ ($\beta = 0.03$, $SE = 1.56$, $t = 0.023$, $p = .981$) and the interaction *Ambiguity level*$_4$* *Exposure* was also not significant ($\beta = 1.59$, $SE = 1.10$, $t = 1.442$, $p = .149$).

Concerning accuracy, we found a main effect of *Exposure* ($\beta = -0.16$, $SE = 0.02$, $z = -8.617$, $p < .000$) with a mean reduction of accuracy of 6% from Block 1 to Block 5. There was no main effect of *Ambiguity level*$_4$ ($\beta = -0.02$, $SE = 0.05$, $z = -0.295$, $p = .768$) and the interaction *Ambiguity level*$_4$* *Exposure* was also not significant ($\beta = 0.023$, $SE = 0.04$, $z = 0.603$, $p = .546$).

3.2. Processing of hierarchical constituency

The results above suggest that participants were sensitive to the higher-order regularities of the sequence up to the third level, we thus restricted the analysis of the structure constituency to Levels 1, 2, and 3.

3.2.1. Hierarchical constituency at Level 1

Analyses of reaction times showed a main effect of *Exposure* ($\beta = -26.52$, $SE = 0.31$, $t = -85.703$, $p < .000$) with a mean reduction of reaction times of 106 ms from Block 1 to Block 5. There was also a main effect of *Structural context*$_{level1}$ ($\beta = 4.89$, $SE = 0.90$, $t = 5.416$, $p < .000$) with points in an ambiguous structural context faster than points in a non-

ambiguous structural context by 4.9 ms. The interaction *Structural context*$_{level1}$* Exposure* was not significant ($\beta = -0.86$, $SE = 0.64$ $t = -1.345$, $p = .178$).

Concerning accuracy, we found a main effect of *Exposure* ($\beta = 0.13$, $SE = 0.03$, z = 5.192, $p < .000$) with accuracy increasing of 0.7% from Block 1 to Block 5. There was no main effect of *Structural context*$_{level1}$ ($\beta = -0.02$, $SE = 0.07$, z = −0.280, $p = .779$). However, the interaction *Structural context*$_{level1}$* Exposure* was significant ($\beta = 0.14$, $SE = 0.05$, z = 2.649, $p = .008$) with accuracy increasing more for points in non-ambiguous structural context ($M_{block1\ -\ block5} = 0.009$) than points in ambiguous structural context ($M_{block1\ -\ block5} = 0.004$). Results are shown in Table 2.

3.2.2. Hierarchical constituency at Level 2

Analyses of reaction times showed a main effect of *Exposure* ($\beta = -25.32$, $SE = 0.30$, $t = -83.536$, $p < .000$) with a mean reduction of reaction times of 101 ms from Block 1 to Block 5. There was no effect of *Structural context*$_{level2}$ ($\beta = -1.29$, $SE = 0.88$, $t = -1.464$, $p = .143$). The interaction *Structural context*$_{level2}$* Exposure* was also not significant ($\beta = -0.18$, $SE = 0.58$, $t = -0.311$, $p = .756$).

Concerning accuracy, we found a main effect of *Exposure* ($\beta = 0.002$, $SE = 0.0004$, $t = 4.802$, $p < .000$) with accuracy increasing of 0.8% from Block 1 to Block 5. *Structural context*$_{level2}$ did not reach significance level ($\beta = -0.002$, $SE = 0.001$, $t = -1.703$, $p = .088$) and the interaction *Structural context*$_{level2}$ * Exposure* was also not significant ($\beta = 0.0008$, $SE = 0.0008$, $t = 0.930$, $p = .352$). Results are shown in Table 2.

3.2.3. Hierarchical constituency at Level 3

Analyses of reaction times showed a main effect of *Exposure* ($\beta = -23.18$, $SE = 0.33$, $t = -68.782$, $p < .000$) with a mean reduction of reaction times of 92 ms from Block 1 to Block 5. There was also a main effect of *Structural context*$_{level3}$ ($\beta = -4.01$, $SE = 0.98$, $t = -4.08$, $p < .000$) with points in non-ambiguous structural context faster than points in ambiguous structural context by 4 ms. The interaction *Structural context*$_{level3}$* Exposure* was significant ($\beta = -1.58$, $SE = 0.69$ $t = -2.279$, $p = .022$) with a more important reduction over exposure for points in a non-ambiguous structural context ($M_{block1\ -\ block5} = -94$ ms) than for points in an ambiguous structural context ($M_{block1\ -\ block5} = -88$ms). Fig. 4 shows the results plotted for each disambiguated point of the ambiguous and non-ambiguous structural context.

With respect to accuracy, we found no main effect of *Exposure* ($\beta = 0.0003$, $SE = 0.0005$, $t = -0.740$, $p = .459$). There was a significant main effect of *Structural context*$_{level3}$ ($\beta = 0.003$, $SE = 0.001$, $t = 2.222$, $p = .026$) with accuracy better for points in a non-ambiguous structural context ($M = 0.96$) than for points in an ambiguous structural context ($M = 0.95$). The interaction *Structural context*$_{level3}$ * Exposure* was significant ($\beta = 0.002$, $SE = 0.001$, $t = 2.371$, $p = .018$) with accuracy increasing for points in a non-ambiguous structural context over exposure ($M_{block1\ -\ block5} = 0.004$) and decreasing for points in an ambiguous structural context ($M_{block1\ -\ block5} = -0.005$). Results are shown in Table 2.

Table 2
Mean proportion (*M*) and standard deviation (*SD*) of correct responses for ambiguous and non-ambiguous structural context by hierarchical levels and blocks

		Block 1		Block 2		Block 3		Block 4		Block 5	
		M	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Level 1	Non-ambiguous Structural Context	0.98	0.13	0.99	0.10	0.99	0.09	0.99	0.09	0.99	0.09
	Ambiguous Structural Context	0.99	0.11	0.99	0.09	0.99	0.11	0.99	0.10	0.99	0.10
Level 2	Non-ambiguous Structural Context	0.97	0.13	0.97	0.12	0.97	0.12	0.98	0.12	0.98	0.12
	Ambiguous Structural Context	0.97	0.12	0.97	0.11	0.97	0.11	0.98	0.11	0.98	0.11
Level 3	Non-ambiguous Structural Context	0.95	0.11	0.96	0.11	0.96	0.12	0.96	0.11	0.96	0.11
	Ambiguous Structural Context	0.95	0.11	0.96	0.11	0.96	0.11	0.95	0.12	0.95	0.12

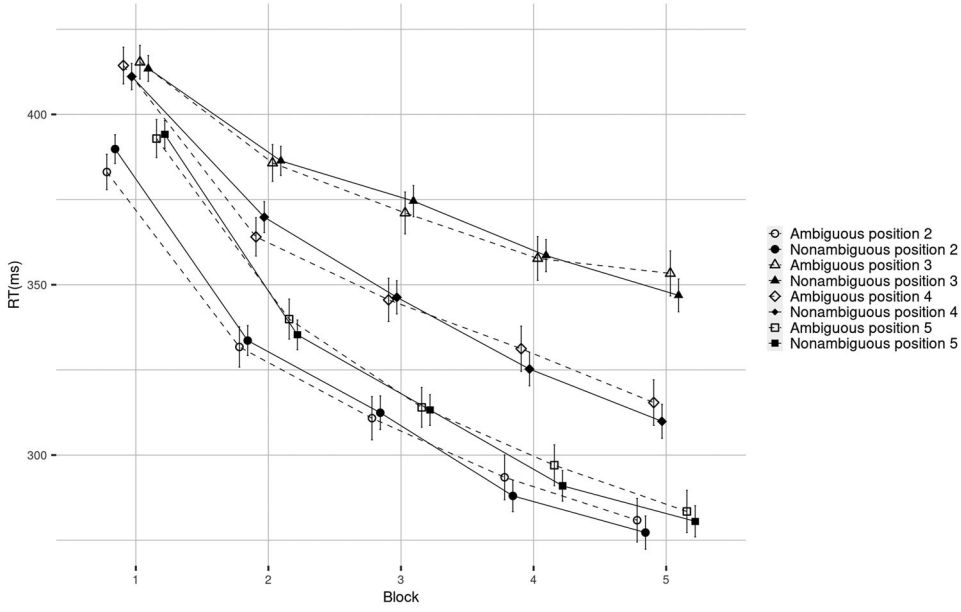

Fig. 4. Mean RT (ms) of disambiguated points occurring in ambiguous (dashed lines) and non-ambiguous (solid lines) structural contexts at Level 3 by position and blocks. The position number indicates the serial order in the constituent [01101], from left to right. Errors bars denote the 95% confidence interval.

4. Discussion

The aim of the present study was to evaluate if binary sequences can be processed as nested structures. To do so, we created aperiodic self-similar sequences from the Fibonacci grammar and tested adult participants' learning of their properties in an SRT task. The transitions within these sequences can be considered from a hierarchical point of view. Sequences being self-similar, transitions between units at level n are identical to transitions between constituents at level $n + 1$. At each level, the transitions are either probabilistic or deterministic. Crucially, the probabilistic transitions at level n are embedded in deterministic transitions at level $n + 1$. It is thus possible to reduce the number of probabilistic transitions by recursively embedding deterministic transitions. This recursive structure allows us to predict precisely, which unit can be anticipated if the underlying hierarchical structure of the sequence is processed.

We hypothesized that hierarchical processing would result in a progressive construction of the underlying, nested structure. This should be reflected by (a) a progressive ability to anticipate specific points in the sequence that are ambiguous at level n but disambiguated at level $n + 1$ and (b) a better anticipation for disambiguated points appearing at level $n + 1$ in a constituent following a deterministic transition (non-ambiguous structural context), compared to the same disambiguated points occurring at level $n + 1$ in a constituent following a probabilistic transition (ambiguous structural context).

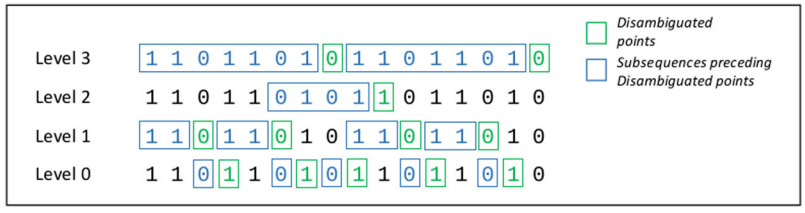

Fig. 5. Subsequences (blue) preceding disambiguated points (green) by hierarchical levels. We see that the linear subsequences necessary to anticipate the disambiguated points of each level overlap.

In line with the first prediction, we found that for Levels 0, 1, 2, and 3, disambiguated points showed a steeper reduction of RTs through exposure than their non-disambiguated counterparts. At Levels 0 and 1, we also found that through exposure, accuracy increased for disambiguated points, while it decreased for non-disambiguated points. However, at Levels 2 and 3, accuracy decreased through exposure for both disambiguated and non-disambiguated points suggesting a speed accuracy trade-off. Critically, this decrease in accuracy does not invalidate our predictions since at Level 2, it was significantly greater for non-disambiguated than disambiguated points, and at Level 3, accuracy was overall higher for disambiguated points. The decrease in accuracy could be due to the boredom of the participants caused by the simplicity of the task. It could also be due to the instructions that only concerned the speed of response. It should also be noted that the magnitude of this decrease remains relatively small, and it was at most at 6% between the first and the last block of the experiment. Finally, we found no sign of anticipation at Level 4. Taken all together, the results of the first analysis suggest that participants were able to build the structure up to the third hierarchical level.

An alternative explanation based on linear precedence may account for the better anticipation of disambiguated points, compared to non-disambiguated points. This explanation is based on the fact that disambiguated points are systematically preceded by a specific subsequence that never precedes non-disambiguated points of the same level, whereas transitions between subsequences of identical length and their following non-disambiguated points are probabilistic (Fig. 1e). Thus, the better anticipation of disambiguated points can potentially come from their linear precedence. However, accounting for the anticipation of disambiguated points with linear precedence faces numerous challenges. First, such explanations would be very costly in terms of memory resource. The linear subsequences needed to anticipate the disambiguated points overlaps (see Fig. 5); hence, the parser would need to track in parallel all the different patterns. Second, the sequence being binary, the patterns are distinguishable only by their positional order; the parser must therefore also be able to deal with the interference caused by the similarity in the patterns' elements. Finally, the pattern allowing anticipation of disambiguated points would have to be held in memory for a relatively long time. In the present experiment, the pattern retention time would include the 500 ms of the response-to-stimulus interval and the time to answer the trial. If we consider a mean reaction time of 300 ms per trials, the patterns allowing anticipation of disambiguated points at Levels 1, 2, and 3 should be held in memory for 1.6, 3.2, and 5.5 s, respectively. Thus, in order to account for the results, a linear precedence parser would have to overcome these three

requirements: overlapping patterns, interference caused by item similarity, and long retention time in working memory. The attentional cost induced by these constraints' casts doubt that a simple pattern recognition mechanism could be a plausible candidate to account for anticipation of disambiguated points.

These results also seem to contradict the hypothesis put forward by Vender et al. (2020) to explain the processing of the Fibonacci grammar. According to this hypothesis, participants would identify certain points of the grammar, called k-points, as relevant structural units and would rely on these k-points to build the hierarchical structure. According to our notation, k-points are the non-disambiguated points of Level 0 (i.e., they are all the 1s that appear after [01]), and therefore Level 2 contrast different instance of k-points (i.e., the disambiguated and non-disambiguated points of Level 2 are all and only k-points). If the formal status of k-points was at the origin of the building of the hierarchical structure, then they should all be identified in the same way, which should translate into an identical processing advantage for all k-points. Therefore, the difference between disambiguated and non-disambiguated points we found at Level 2 cannot be explained by Vender et al.'s (2020) hypothesis. Moreover, since k-points are by definition 1s, this hypothesis cannot explain the effects we found at Level 3, which concern differences between 0s. Since Vender et al.'s (2020) core argument in favor of k-points relies on the comparison between the processing of k-points in the Fibonacci grammar and in an alternative grammar, we cannot assess the validity of this hypothesis; however, the formal approach adopted by these authors needs to be further elaborated to account for our results.

Concerning the second prediction, we found that accuracy increased significantly more in non-ambiguous structural contexts than in ambiguous structural contexts at Level 1,[3] suggesting progressive learning of the constituent structure at Level 1. Results also showed that points occurring in an ambiguous structural context were overall faster than when they appeared in a non-ambiguous structural context. However, that effect was there from the beginning of the sequence, that is, it did not interact with exposure, which suggests that it does not reflect learning. Level 3 showed the predicted effect of structural context in both RTs and accuracy, with a significant reduction of RTs and a significant accuracy increase for the non-ambiguous structural context, compared to the ambiguous structural context. However, at Level 2, we found no effect of structural context in either RTs or accuracy, although a trend was found in the expected direction. Before reasoning about the possible explanation to the lack of effect at Level 2, it is important to highlight that the effects found at Levels 1 and 3 already exclude the possibility that performance is *only* due to "flat" statistical learning processes (i.e., linear precedence). If better anticipation for the disambiguated points was due to participants memorizing the subsequence preceding them, the structural context in which they occur should have no influence given that in both ambiguous and non-ambiguous structural contexts, disambiguated points were preceded by exactly the same subsequences. These effects can only be accounted for by a strategy that incorporates in one way or another the notion of hierarchy. But why did structural context fail to significantly affect performance at Level 2? Although we are currently unable to provide one fully satisfying explanation, we can sketch different lines of reasoning. First, it should be kept in mind that for the analysis of structural context, we compared at each level different instances of the same disambiguated points. At Level 2, we

compared two subsets of disambiguated points from Levels 0 and 1 whose transitional probabilities were $p(1|0) = 1$ and $p(0|11) = 1$, respectively. It could be that the linear precedence of the points involved in this comparison has hidden the effects of structural context. In line with this interpretation, the first analysis showed that these disambiguated points were learned very early in the experiment, already in Block 1 (see Fig. 2). Moreover, these disambiguated points were the ones that showed the highest RT decrease. It is thus possible that a floor level was reached, making the effect of structural context undetectable. However, according to this interpretation, the effects should be weaker for the lower level than for higher levels (i.e., it should be the strongest at Level 3, followed by Level 2 and then Level 1) because higher-level constituents contain points that are also disambiguated at higher levels, which imply that the influence of the linear precedence should decrease when the higher one progresses in the hierarchy. The fact that we observed an effect at Level 1 therefore tempers this interpretation, although the effect size was small. Finally, Fig. 4 shows that the effect of structural context at Level 3 is distributed across all the points of the constituent. In particular, the RTs of the points at Positions 4 and 5, which correspond respectively to disambiguated points at Levels 1 and 0, decrease more strongly in the non-ambiguous structural context than in the ambiguous structural context. These points are precisely the disambiguated points taken in the analysis of the structural context of Level 2. Thus, it might be that the null result found at that level was due to a lack of statistical power.

Taken together, those results suggest that participants have organized the input in a hierarchical way. However, the exact nature of the representations that have been acquired remains to be explored. Fig. 4. shows that the advantage for the non-ambiguous structural context was not driven by one particular point but was distributed across all the points that appeared in that context. This last finding is interesting as it tells us something about the type of hierarchical structure participants built. We have suggested that the process by which participants anticipate higher-order regularities would consist in the recursive combination of units linked through deterministic transitions. However, such a mechanism does not necessarily need to represent a unit as embedded in multiple hierarchical levels; the parser could only retain a representation of the highest level's constituents and anticipate the constituents as wholes. In that view, lower levels' constituents are dissolved into higher levels' constituents and become inaccessible once these higher levels' constituents are represented. In other words, the internal hierarchical structure of the constituents might dissolve as hierarchical building progresses. Such a hypothesis is assumed in different models of chunking in which there is no record of the sequential steps by which a chunk is formed (French, Addyman, & Mareschal, 2011; Goldwater, Griffiths, & Johnson, 2009; McCauley & Christiansen, 2014; Perruchet & Vinter, 1998; Robinet, Lemaire, & Gordon, 2011). For example, in PARSER (Perruchet & Vinter, 1998), the system chunks together units that are present in the focus of attention. The span of this focus changes randomly at each trial (encompassing 1, 2, or 3 units). Once a chunk is created, it is processed as a single unit in the focus of attention. Thus, if a chunk reoccurs in the signal, it will occupy only one slot in the focus of attention. This allows the model to chunk multiple chunks together if they are present at the same time in the focus of attention. The activation value of a chunk decreases at each trial if it is not in the focus of attention and increases each time the chunk is encountered. When multiple chunks in memory correspond

to the signal (i.e., when the signal could fit with chunks of different sizes) the activation value of the chunk with the best fit increases, while the activation value of the chunks with a lower fit decrease. In this way, the small chunks that are created in the early phases of learning have their activation values progressively tend to 0 as bigger chunks that embed them are created. This results in a representation where only the biggest chunks that fit the signal are kept in memory whereas the smaller chunks that allowed the creation of these bigger chunks are progressively erased from memory. In this view, cognitive representations are limited to chunks with no internal hierarchical structure.

Evidence supporting this claim comes from the so-called *subunit effect* that shows that subunits of a chunk are less accessible once a chunk is learned (Fiser & Aslin, 2005; Giroux & Rey, 2009; Orbán, Fiser, Aslin, & Lengyel, 2008; Slone & Johnson, 2015; 2018). In SRT experiments, this manifests as relatively slow RTs for the first unit of a chunk followed by an acceleration for the remaining units (Hunt & Aslin, 2001; Jiménez, Méndez, Pasquali, Abrahamse, & Verwey, 2011; Sakai, Kitaguchi, & Hikosaka, 2003). In our experiment, if participants were processing constituents as single units without internal structure, RTs should progressively diminish through the constituent. This should be especially true for constituents appearing in the non-ambiguous structural contexts at Level 3. This constituent (01101) is composed of five points and four transitions: If it were processed as a single unit, the transition from one point to the next should result in a progressive reduction of RTs, and the transitional pattern should thus be (- - - -) (where "-"corresponds to a diminution of RTs from each unit to the following). In contrast to that prediction, the transitional pattern observed for this constituent in the last two blocks is (- + - -) (where "+"corresponds to an increase of RTs), that is, there was a strong deceleration at the second transition. Crucially, that deceleration appears precisely at the border between two constituents at the lower level: The internal structure of [01101] is indeed [[01][101]]. The pattern of acceleration/deceleration therefore provides further evidence that participants represent the internal structure of constituent [01101].

In order to make sure that the deceleration at the second transition observed at the group level was not driven by a subset of participants, we computed for each participant the direction of the four transitions of the constituent in the non-ambiguous structural context at Level 3. We ran by-participants comparisons with four linear models (one for each transition). The factor *Position* had two modalities (before, after), "before" coded for the points that were before the transition and "after" coded for the point after the transition. Each model had as predictor the factors *Participants* and the interaction *Participants* Position* (the factor *Position* was entered only in the interaction term in order to compare the effect of position for the same individual and not across individuals). In order to increase statistical power, we computed transitions for Blocks 4 and 5 jointly (see Supporting Information for detailed results). Table 3 shows the number of participants by transition pattern. We see that 78% of the participants show a deceleration at the second transition, 22% show no variation in RTs, and critically none show acceleration. This shows that the transition pattern (- + - -) found at the group level is replicated at the individual level and is therefore not due to a mix of different patterns across participants. We also see that the transitional pattern (- - - -), expected if chunks lost their internal structure, was found in no participants, suggesting that the constituent [01101] was never processed as a single unit. Crucially, 93% of the slow-downs occurred at the second

Table 3
Distribution of the statistical effects for the four transitional patterns in Blocks 4 and 5 combined for the constituent [01101] in non-ambiguous structural context

Transitional Pattern				
Transition 1	Transition 2	Transition 3	Transition 4	No. of Obs.
−	+	−	−	30
−	+	=	−	32
−	+	−	=	31
−	+	=	=	22
−	=	=	=	20
−	+	−	+	5
−	=	−	+	3
−	=	−	=	3
=	=	=	=	3
−	+	+	−	2
=	=	−	=	2
−	=	+	−	1
−	=	=	−	1
−	=	=	+	1
=	+	−	=	1
=	+	=	=	1
=	=	−	+	1
−	−	−	−	0

Note. The + sign indicate a significant increase in reaction times. The − sign indicate a significant decrease in reaction times. The = sign indicate no significant differences in reaction times. Significant differences were considered at the $p < .05$ level.

and third transitions, that is, at the boundary between lower-level constituents. This suggests that participants represent several hierarchical levels simultaneously: The pattern reflects the processing of the internal structure [[01][[1][01]]] of the constituent [01101]. This observation brings further support to our hypothesis that sequences are represented as recursive embedding of constituents.

In the present study, we proposed that the cognitive system would build a hierarchical structure by recursively combining deterministic transitions in the Fibonacci grammar. This mechanism does not require that participants have access to the rewriting rules of the grammar. Because of the Fib-specific self-similarity, which makes the transitional probabilities perfectly scale-free, the surface properties (i.e., the transitional probabilities) lead the parser to a structure that is identical to the natural structure of the Fibonacci grammar. We would like to emphasize that there may be different strategies to build a hierarchical structure from the Fibonacci grammar. However, our results can only be explained by a single family of strategies: those that are sensitive to hierarchically organized substrings.

Our results also confirm the finding of Planton et al. (2021) that even sequences as simple as binary sequences can be processed hierarchically. Our proposal that the parser relies on the statistical regularities of the signal to access higher-level constituents is also consistent

with the results reported by these authors regarding the involvement of statistical learning. Indeed, this component explained a significant part of the variance even in sequences with high Kolmogorov complexity. The idea that the degree of complexity of the input is the factor that will lead the system to recode the information has also been put forward to explain how the system induces rules from a set of exemplars (Pothos, 2010; Radulescu, Wijnen, & Avrutin, 2019, 2021). In particular, Radulescu et al. (2019, 2021) proposed that the recoding of information into a more abstract format depends on the complexity of the signal and the finite encoding capabilities of the cognitive system. The degree of entropy of a signal (i.e., its complexity) depends on the number of items that compose it as well as on the homogeneity of the distribution of these items. The more homogeneous the distribution (i.e., all items have the same probability) and the longer the signal, the higher the entropy is. Radulescu argues that rule induction arises when the entropy level exceeds the encoding capacity of the system. This upper limit of the amount of information that can be sent through the channel per unit of time forces the system to compress the information into a more abstract format in order to reduce the level of entropy. We suggest that the construction of a hierarchical structure can be seen as a way to reduce the entropic state of the parser: Uncertainty is reduced as the hierarchical structure of the signal is built, in line with the proposition of Radulescu et al. (2019, 2021). However, the particularity of the Fibonacci grammar is that at each level, the statistical distribution of the constituents is identical due to the specific flavor of self-similarity of the Fibonacci grammar. An interesting line for future research could be to ask whether and how self-similarity may play a role in the compression of the input since it is independent of the entropy of the signal. The rich world of L-systems allows such manipulation, that is manipulating the degree of isomorphism of the self-similarity while keeping entropy constant.

Acknowledgments

We thank Diego Krivochen for his ongoing support, in-depth conversations, and helpful comments. We also thank Denis Delfitto, Maria Vender, and Beth Phillips for their valuable discussions and comments.

Open access funding provided by Universite de Geneve.

Notes

1 Note that the hierarchical depth can of course only be infinite for an infinite chain. In the present study, the presented sequences were 233 points long and had potentially up to 12 hierarchical levels, which is presumably well beyond the processing capacity of the cognitive system.

2 Formally, the rewriting rules of a grammar operate on the "symbols" of an alphabet. The expression of the symbols (i.e., their actual realization) can however vary. For example, 0s and 1s can be replaced arbitrarily by As and Bs without any impact. In this article, we use the term "point" to refer to the actual realization of the symbols of the Fibonacci grammar.

3 The reader may find it surprising that accuracy increases with exposure at Levels 1 and 2 in the analysis *Processing of hierarchical constituency*, while it decreases for the same levels in the analysis *Processing of hierarchical structure*. This is explained by the fact that the two analyses rely on different contrasts. In the analysis *Processing of hierarchical structure*, the two modalities of the factor *Ambiguity* contrast disambiguated and non-disambiguated points at given level. In the analysis *Processing of hierarchical constituency*, the two modalities of the factor *Structural context* contrast different instances of disambiguated points; non-disambiguated points are not taken into account in this analysis. *Structural context*$_{level1}$ contrasts disambiguated points at Level 0 and *Structural context*$_{level2}$ contrasts disambiguated points at Level 0 combined with disambiguated points at Level 1. The analysis *Processing of hierarchical structure* show that accuracy increases with exposure for the disambiguated points at Levels 0 and 1 (see Table 1); therefore, it is logical that accuracy also increases at Levels 1 and 2 in the analysis *Processing of hierarchical constituency*.

References

Bates, D., Maechler, M., Bolker, B., & Walker, S. (2014). Lme4: Linear mixed-effects models using Eigen and S4. R package version 1.1-7. https://doi.org/10.48550/arXiv.1406.5823 access date "4th april 2021"

Chomsky, N. (1957). Logical structures in language. *American Documentation (pre-1986)*, *8*(4), 284.

Chomsky, N., & Lightfoot, D. W. (2002). *Syntactic structures*. Berlin: Walter de Gruyter.

Dehaene, S., Meyniel, F., Wacongne, C., Wang, L., & Pallier, C. (2015). The neural representation of sequences: From transition probabilities to algebraic patterns and linguistic trees. *Neuron*, *88*(1), 219. https://doi.org/10.1016/j.neuron.2015.09.019

de Vries, M. H., Monaghan, P., Knecht, S., & Zwitserlood, P. (2008). Syntactic structure and artificial grammar learning: The learnability of embedded hierarchical structures. *Cognition*, *107*(2), 763–774. https://doi.org/10.1016/j.cognition.2007.09.002

de Vries, M. H., Petersson, K. M., Geukes, S., Zwitserlood, P., & Christiansen, M. H. (2012). Processing multiple non-adjacent dependencies: Evidence from sequence learning. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *367*(1598), 2065–2076. https://doi.org/10.1098/rstb.2011.0414

Fiser, J., & Aslin, R. N. (2005). Encoding multielement scenes: Statistical learning of visual feature hierarchies. *Journal of Experimental Psychology: General*, *134*(4), 521–537. https://doi.org/10.1037/0096-3445.134.4.521

Fitch, W. T. (2014). Toward a computational framework for cognitive biology: Unifying approaches from cognitive neuroscience and comparative cognition. *Physics of Life Reviews*, *11*(3), 329–364. https://doi.org/10.1016/j.plrev.2014.04.005

Fitch, W. T., & Hauser, M. D. (2004). Computational constraints on syntactic processing in a nonhuman primate. *Science*, *303*(5656), 377–380. https://doi.org/10.1126/science.1089401

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

French, R. M., Addyman, C., & Mareschal, D. (2011). TRACX: A recognition-based connectionist framework for sequence segmentation and chunk extraction. *Psychological Review*, *118*(4), 614–636. https://doi.org/10.1037/a0025255

Friederici, A.D., Bahlmann, J., Heim, S., Schubotz, S.I., & Anwander, A. (2006). The brain differentiates human and non-humangrammars: Functional localization and structural connectivity. *Proceedings of the National Academy of Sciences*, *103*(7), 2458–2463. https://doi.org/10.1073/pnas.0509389103

Geambaşu, A., Ravignani, A., & Levelt, C. C. (2016). Preliminary experiments on human sensitivity to rhythmic structure in a grammar with recursive self-similarity. *Frontiers in Neuroscience*, *10*, 281. https://doi.org/10.3389/fnins.2016.00281

Geambaşu, A., Toron, L., Ravignani, A., & Levelt, C. C. (2020). Rhythmic recursion? Human sensitivity to a Lindenmayer grammar with self-similar structure in a musical task. *Music & Science*, *3*, 205920432094661. https://doi.org/10.1177/2059204320946615

Giroux, I., & Rey, A. (2009). Lexical and sublexical units in speech perception. *Cognitive Science*, *33*(2), 260–272. https://doi.org/10.1111/j.1551-6709.2009.01012.x

Goldwater, S., Griffiths, T. L., & Johnson, M. (2009). A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, *112*(1), 21–54. https://doi.org/10.1016/j.cognition.2009.03.008

Honing, H., & Zuidema, W. (2014). Decomposing dendrophilia. *Physics of Life Reviews*, *11*(3), 375–376. https://doi.org/10.1016/j.plrev.2014.06.020

Hunt, R. H., & Aslin, R. N. (2001). Statistical learning in a serial reaction time task: Access to separable statistical cues by individual learners. *Journal of Experimental Psychology: General*, *130*(4), 658–680. https://doi.org/10.1037/0096-3445.130.4.658

Jiménez, L., Méndez, A., Pasquali, A., Abrahamse, E., & Verwey, W. (2011). Chunking by colors: Assessing discrete learning in a continuous serial reaction-time task. *Acta Psychologica*, *137*(3), 318–329. https://doi.org/10.1016/j.actpsy.2011.03.013

Koch, I., & Hoffmann, J. (2000). Patterns, chunks, and hierarchies in serial reaction-time tasks. *Psychological Research*, *63*(1), 22–35. https://doi.org/10.1007/PL00008165

Koelsch, S. (2005). Neural substrates of processing syntax and semantics in music. *Current Opinion in Neurobiology*, *15*(2), 207–212. https://doi.org/10.1016/j.conb.2005.03.005

Kotz, S. A., Ravignani, A., & Fitch, W. T. (2018). The evolution of rhythm processing. *Trends in Cognitive Sciences*, *22*(10), 896–910. https://doi.org/10.1016/j.tics.2018.08.002

Kovács, Á. M., & Endress, A. D. (2014). Hierarchical processing in seven-month-old infants. *Infancy*, *19*(4), 409–425. https://doi.org/10.1111/infa.12052

Krivochen, D., Phillips, B., & Saddy, J. (2018). Classifying points in Lindenmayer systems: Transition probabilities and structure reconstruction (v. 1.1). Retrieved March 1, 2020, from https://doi.org/10.13140/RG.2.2.25719.88484

Kuznetsova, A., Brockhoff, P. B., & Christensen, R. H. B. (2015). Package 'lmertest'. *R package version*, *2*(0), 734.

Lai, J., & Poletiek, F. H. (2011). The impact of adjacent-dependencies and staged-input on the learnability of center-embedded hierarchical structures. *Cognition*, *118*(2), 265–273. https://doi.org/10.1016/j.cognition.2010.11.011

Lai, J., & Poletiek, F. H. (2013). How "small" is "starting small" for learning hierarchical centre-embedded structures? *Journal of Cognitive Psychology*, *25*(4), 423–435. https://doi.org/10.1080/20445911.2013.779247

Lashley, K. S. (1951). *The problem of serial order in behavior* (Vol. 21). Oxford, England: Bobbs-Merrill.

Levelt, W. J. M. (2019). On empirical methodology, constraints, and hierarchy in artificial grammar learning. *Topics in Cognitive Science*, *12*(3), 942–956. https://doi.org/10.1111/tops.12441

Lewis, S., & Phillips, C. (2015). Aligning grammatical theories and language processing models. *Journal of Psycholinguistic Research*, *44*(1), 27–46. https://doi.org/10.1007/s10936-014-9329-z

Liesefeld, H. R., & Janczyk, M. (2019). Combining speed and accuracy to control for speed-accuracy trade-offs(?). *Behavior Research Methods*, *51*(1), 40–60. https://doi.org/10.3758/s13428-018-1076-x

Lindenmayer, A. (1968). Mathematical models for cellular interactions in development II. Simple and branching filaments with two-sided inputs. *Journal of Theoretical Biology*, *18*(3), 300–315. https://doi.org/10.1016/0022-5193(68)90080-5

Marcus, G. F., Vijayan, S., Rao, S. B., & Vishton, P. M. (1999). Rule Learning by Seven-Month-Old Infants. *Science*, *283*(5398), 77–80. https://doi.org/10.1126/science.283.5398.77

Martins, M. D., Gingras, B., Puig-Waldmueller, E., & Fitch, W. T. (2017). Cognitive representation of "musical fractals": Processing hierarchy and recursion in the auditory domain. *Cognition*, *161*, 31–45. https://doi.org/10.1016/j.cognition.2017.01.001

Martins, M., de, J. D., Muršič, Z., Oh, J., & Fitch, W. T. (2015). Representing visual recursion does not require verbal or motor resources. *Cognitive Psychology*, *77*, 20–41. https://doi.org/10.1016/j.cogpsych.2015.01.004

Martins, M. J. 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

Martins, M. J. D., Fischmeister, F. Ph. S., Gingras, B., Bianco, R., Puig-Waldmueller, E., Villringer, A., … Beisteiner, R. (2020). Recursive music elucidates neural mechanisms supporting the generation and detection of melodic hierarchies. *Brain Structure and Function*, *225*(7), 1997–2015. https://doi.org/10.1007/s00429-020-02105-7

Martins, M. J. D., Krause, C., Neville, D. A., Pino, D., Villringer, A., & Obrig, H. (2019). Recursive hierarchical embedding in vision is impaired by posterior middle temporal gyrus lesions. *Brain*, *142*(10), 3217–3229. https://doi.org/10.1093/brain/awz242

Martins, M. J., Fischmeister, F. P., Puig-Waldmüller, E., Oh, J., Geißler, A., Robinson, S., Fitch, W. T., & Beisteiner, R. (2014). Fractal image perception provides novel insights into hierarchical cognition. *NeuroImage*, *96*, 300–308. https://doi.org/10.1016/j.neuroimage.2014.03.064

Maruyama, M., Pallier, C., Jobert, A., Sigman, M., & Dehaene, S. (2012). The cortical representation of simple mathematical expressions. *NeuroImage*, *61*(4), 1444–1460. https://doi.org/10.1016/j.neuroimage.2012.04.020

McCauley, S. M., & Christiansen, M. H. (2014). Acquiring formulaic language: A computational model. *The Mental Lexicon*, *9*(3), 419–436. https://doi.org/10.1075/ml.9.3.03mcc

Monti, M. M., Parsons, L. M., & Osherson, D. N. (2012). Thought beyond language: Neural dissociation of algebra and natural language. *Psychological Science*, *23*(8), 914–922. https://doi.org/10.1177/0956797612437427

Mueller, J. L., Bahlmann, J., & Friederici, A. D. (2010). Learnability of embedded syntactic structures depends on prosodic cues. *Cognitive Science*, *34*(2), 338–349. https://doi.org/10.1111/j.1551-6709.2009.01093.x

Nakai, T., & Sakai, K. L. (2014). Neural mechanisms underlying the computation of hierarchical tree structures in mathematics. *PLOS ONE*, *9*(11), e111439. https://doi.org/10.1371/journal.pone.0111439

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, *19*(1), 132. https://doi.org/10.1016/0010-0285(87)90002-8

Orbán, G., Fiser, J., Aslin, R. N., & Lengyel, M. (2008). Bayesian learning of visual chunks by human observers. *Proceedings of the National Academy of Sciences*, *105*(7), 2745–2750.

Perruchet, P., & Rey, A. (2005). Does the mastery of center-embedded linguistic structures distinguish humans from nonhuman primates? *Psychonomic Bulletin & Review*, *12*(2), 307–313. https://doi.org/10.3758/BF03196377

Perruchet, P., & Vinter, A. (1998). PARSER: A model for word segmentation. *Journal of Memory and Language*, *39*(2), 246–263. https://doi.org/10.1006/jmla.1998.2576

Planton, S., Kerkoerle, T. V., Abbih, L., Maheu, M., Meyniel, F., Sigman, M., Wang, L., Figueira, S., Romano, S., & Dehaene, S. (2021). A theory of memory for binary sequences: Evidence for a mental compression algorithm in humans. *PLOS Computational Biology*, *17*(1), e1008598. https://doi.org/10.1371/journal.pcbi.1008598

Pothos, E. (2010). An entropy model for artificial grammar learning. *Frontiers in Psychology*, *1*, 16. https://doi.org/10.3389/fpsyg.2010.00016

R Core Team. (2021). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Available at: https://www.R-project.org/ access date "2 january 2021"

Radulescu, S., Kotsolakou, A., Wijnen, F., Avrutin, S., & Grama, I. (2021). Fast but not furious. When sped up bit rate of information drives rule induction. *Frontiers in Psychology*, *12*, 4987. https://doi.org/10.3389/fpsyg.2021.661785

Radulescu, S., Wijnen, F., & Avrutin, S. (2019). Patterns bit by bit. An entropy model for rule induction. *Language Learning and Development*, *16*, 109–140. https://doi.org/10.1080/15475441.2019.1695620

Rezlescu, C., Danaila, I., Miron, A., & Amariei, C. (2020). More time for science: Using Testable to create and share behavioral experiments faster, recruit better participants, and engage students in hands-on research. In B. L. Parkin (Ed.), *Progress in brain research* (Vol. 253, pp. 243–262). Amsterdam: Elsevier. https://doi.org/10.1016/bs.pbr.2020.06.005

Robinet, V., Lemaire, B., & Gordon, M. B. (2011). MDLChunker: A MDL-based cognitive model of inductive learning. *Cognitive Science*, *35*(7), 1352–1389. https://doi.org/10.1111/j.1551-6709.2011.01188.x

Saddy, J. D. (2009). Perceiving and processing recursion in formal grammars. *Recursion: Structural Complexity in Language and Cognition Conference at the University of Massachusetts (Amherst)*, Amherst, MA.

Sakai, K., Kitaguchi, K., & Hikosaka, O. (2003). Chunking during human visuomotor sequence learning. *Experimental Brain Research*, *152*(2), 229–242. https://doi.org/10.1007/s00221-003-1548-8

Schwarb, H., & Schumacher, E. H. (2012). Generalized lessons about sequence learning from the study of the serial reaction time task. *Advances in Cognitive Psychology*, *8*(2), 165–178. https://doi.org/10.5709/acp-0113-1

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(3), 379–423.

Shirley, E. J. (2014). *Representing and remembering Lindenmayer-grammars*. (Doctoral dissertation), University of Reading https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.658878

Simon, H. A. (1962). An information processing theory of intellectual development. *Monographs of the Society for Research in Child Development*, *27*(2), 150–161.

Slone, L., & Johnson, S. P. (2015). Statistical and chunking processes in adults' visual sequence learning. *Proceedings of the 37th Annual Conference of the Cognitive Science Society*, Pasadena, CA (pp. 2218–2223).

Slone, L. K., & Johnson, S. P. (2018). When learning goes beyond statistics: Infants represent visual sequences in terms of chunks. *Cognition*, *178*, 92102. https://doi.org/10.1016/j.cognition.2018.05.016

Vandierendonck, A. (2017). A comparison of methods to combine speed and accuracy measures of performance: A rejoinder on the binning procedure. *Behavior Research Methods*, *49*(2), 653–673. https://doi.org/10.3758/s13428-016-0721-5

Vandierendonck, A. (2018). Further tests of the utility of integrated speed-accuracy measures in task switching. *Journal of Cognition*, *1*(1), 8. https://doi.org/10.5334/joc.6

Vender, M., Krivochen, D. G., Compostella, A., Phillips, B., Delfitto, D., & Saddy, D. (2020). Disentangling sequential from hierarchical learning in Artificial Grammar Learning: Evidence from a modified Simon Task. *PLOS ONE*, *15*(5), e0232687. https://doi.org/10.1371/journal.pone.0232687

Vender, M., Krivochen, D. G., Phillips, B., Saddy, D., & Delfitto, D. (2019). Implicit learning, bilingualism, and dyslexia: Insights from a study assessing AGL with a modified Simon Task. *Frontiers in Psychology*, *10*, 1647.https://doi.org/10.3389/fpsyg.2019.01647

Verwey, W. B., & Wright, D. L. (2014). Learning a keying sequence you never executed: Evidence for independent associative and motor chunk learning. *Acta Psychologica*, *151*, 2431. https://doi.org/10.1016/j.actpsy.2014.05.017

Vitányi, P. M. B., & Walker, A. (1978). Stable string languages of lindenmayer systems. *Information and Control*, *37*(2), 134–149. https://doi.org/10.1016/S0019-9958(78)90483-7

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Appendix A. Supplementary material

Data associated with this article can be found electronically at https://osf.io/8n9he/?view_only=ce203fa912294af0b9391f0ad19be392