Cognition xxx (2012) xxx–xxx

Contents lists available at SciVerse ScienceDirect

Cognition

journal homepage: www.elsevier.com/locate/COGNIT

Eye gaze reveals a fast, parallel extraction of the syntax of arithmetic formulas

Elisa Schneider[a,*], Masaki Maruyama[b,c,d,e], Stanislas Dehaene[b,c,d,e], Mariano Sigman[a]

[a] Laboratory of Integrative Neuroscience, Physics Department, University of Buenos Aires, Buenos Aires, Argentina
[b] INSERM, U992, Cognitive Neuroimaging Unit, F-91191 Gif/Yvette, France
[c] CEA, DSV/I2BM, NeuroSpin Center, F-91191 Gif/Yvette, France
[d] Univ Paris-Sud, Cognitive Neuroimaging Unit, F-91191 Gif/Yvette, France
[e] Collège de France, F-75005 Paris, France

ARTICLE INFO

Article history:
Received 11 July 2011
Revised 25 May 2012
Accepted 29 June 2012
Available online xxxx

Keywords:
Arithmetic
Gestalt
Cognitive architecture
Language
Mathematical education

ABSTRACT

Mathematics shares with language an essential reliance on the human capacity for recursion, permitting the generation of an infinite range of embedded expressions from a finite set of symbols. We studied the role of syntax in arithmetic thinking, a neglected component of numerical cognition, by examining eye movement sequences during the calculation of arithmetic expressions. Specifically, we investigated whether, similar to language, an expression has to be scanned sequentially while the nested syntactic structure is being computed or, alternatively, whether this structure can be extracted quickly and in parallel. Our data provide evidence for the latter: fixations sequences were stereotypically organized in clusters that reflected a fast identification of syntactic embeddings. A syntactically relevant pattern of eye movement was observed even when syntax was defined by implicit procedural rules (precedence of multiplication over addition) rather than explicit parentheses. While the total number of fixations was determined by syntax, the duration of each fixation varied with the complexity of the arithmetic operation at each step. These findings provide strong evidence for a syntactic organization for arithmetic thinking, paving the way for further comparative analysis of differences and coincidences in the instantiation of recursion in language and mathematics.

1. Introduction

Human language is characterized by a generative capacity: by embedding linguistic constituents inside each other, we can generate an infinite number of sentences. This capacity does not seem to exist in other animal communication systems (Chomsky, 1957). It has been suggested to arise from a core process of recursion which allows the formation of tree-like mental structures, not only in the field of language, but also in other cognitive domains such as music and mathematics (Hauser, Chomsky, & Fitch, 2002). Recursion would be a key distinctive feature separating humans from non-human primates.

While there is a vast corpus of evidence demonstrating a syntactic organization of language (Chomsky, 1988), and similar analyses of music (Lerdahl & Jackendoff, 1996; Patel, 2003), only a handful of studies have explored whether and how mathematical reasoning is organized at the syntactic level. These studies focused on the perception and recognition memory of mathematical expressions (Ernest, 1987; Nowak, Plotkin, & Jansen, 2000; Posner, Walker, Friedrich, & Rafal, 1984; Ranney, 1987). They demonstrated that syntactically well-formed substrings such as $4-x$ can be more easily memorized than random non-grammatical strings such as $x3$ (Nowak et al., 2000). Providing further evidence in favor of a syntactic encoding of equations, Nowak et al. (2000) showed that within the

* Corresponding author.
E-mail address: ichischneider@gmail.com (E. Schneider).

0010-0277/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cognition.2012.06.015

well-formed sub-expressions contained in a string, those which form a phrasal node on the parse tree are more readily remembered. For instance, if the string $4 - x^2(y + 7)$ is presented, the substring $y + 7$ is more likely to be memorized than the substring $4 - x^2$ (Nowak et al., 2000). In a recent study, Jansen, Marriott, and Yelland (2003) explored the scanning sequence of algebraic expressions using the Restricted Focus Viewer (RFV), whereas participants see at any time only a small region of the image in focus. The window can be moved using the computer mouse. As observed in language reading (Mitchell et al., 2008), symbols at the end of a phrasal constituent were scanned for significantly longer durations than symbols at the start or middle of the phrasal constituent (Nowak et al., 2000).

Landy and Goldstone (2007a, 2007b) have performed a series of experiments to measure the temporal and spatial allocation of attention in arithmetical expressions (Goldstone, Landy, & Son, 2010; Landy & Goldstone, 2007b; Landy, Jones, & Goldstone, 2008). These experiments have shown that instead of memorizing that procedurally multiplication comes before addition, saliency maps may direct eye movements in such a way as to automatically instantiate this rule (Goldstone et al., 2010). In an experiment particularly relevant for the present study, Landy et al. (2008) measured eye-movements while subjects solve "multiplication–addition" problems ('$2 \cdot 3 + 4$') and addition–multiplication ('$3 + 1 \cdot 4$') problems. The results showed that saccades towards the multiplication sign tended to be earlier in the trial and to last longer than saccades to the addition sign (Landy et al., 2008).

In the present work we study how people solve arithmetic expressions with topologically equivalent parse trees, but with different spatial layouts such as $3 - (2 + (1 - 4))$ or $((1 - 4) + 2) + 3$. We contrast three alternative hypotheses on how syntax is processed in mathematical expressions, in relation to language.

H1. Mathematical expressions are similar to language: an equation has to be scanned from left to right, sequentially, much like a sentence, while the nested syntactic structure is being computed.

H2. The syntactic structure can be extracted very quickly and in parallel, but the typical left-to-right sequential organization of language is imprinted in mathematical thinking. This hypothesis predicts an initial bias which favors left to right parsing sequences.

H3. The syntactic structure can be extracted in parallel, and language sets no bias on how mathematical expressions are read, predicting that all aspects of performance depend only on the topology of the parse-tree but are independent of their spatial layout.

We use eye movement patterns and chronometric measures to resolve these alternatives. Response-time data can distinguish H1 and H2 from H3 by demonstrating a cost for expressions whose parse tree does not have a left to right structure. Furthermore, hypotheses H1 and H2 establish distinct predictions on the precise trajectory of eye movements. H1 predicts a left-to right linear exploration in space. Instead, H2 predicts that only the initial fixations may reflect a reading bias, and very rapidly, the trajectory should reflect the hierarchical structure of the expression independently of its spatial layout.

Our data show strong evidence in favor of the second hypothesis: gaze is directed to the left of the equation, reflecting a bias which is consistent with language structuring, and then very rapidly follows the syntactic organization of the expression.

2. General method

2.1. Participants

A total of 35 people participated in three independent experiments. All participants had finished high-school and thus had substantial educational practice in arithmetic. All the participants were native Spanish speakers. The experiment was approved by the local institutional ethics committee. Informed consent was obtained from all of the participants after the purpose and procedures of the experiment were fully explained.

2.2. Stimuli and procedure

Every trial started with the presentation of a fixation cross positioned 8.7° below the center of the screen. After participants had sustain fixation for 500 ms, an arithmetic expression appeared in a window centered 8.7° above the center of the screen.

Arithmetic expressions were written using the Courier New font with fixed character width of 1.67°. The distance between the centers of two adjacent symbols in the expression was 2.2°. The expression was displayed in a rectangular area which covered 2.3° of vertical span. The horizontal span varied with the number of symbols of the expression. In the different experiments expressions could have 11, 9, 4 or 3 symbols. The horizontal span was 24.2°, 19.4°, 8.8° and 6.6° respectively.

Only fixations inside the expression span were analyzed. In fixation sequences, fixation 1 is referred as the first fixation of the trial directed to the equations. For each fixation we calculated the distance to the center of all symbols. For analysis, we labeled each fixation according to the closest symbol (Fig. 1).

Experiment 1 involved additions and subtractions. The expressions were built using numerals 1–4. In Experiment 2 participants solved multiplications, therefore numerals 2–5 were used to avoid the trivial multiplication by one.

Participants were asked to solve the arithmetic expression by uttering the final result rapidly but taking sufficient time to avoid committing errors. The response time was defined as the time interval from the appearance of the expression to the onset of the vocal response. The proportion of correct trials was above 85% for all participants and for all experiments. On average, error rates were below 6%. Unless explicitly mentioned (in Figs. 3b, and 5e and Supplementary Fig. 2) all analyses are performed on correct trials.

Movements of the observers' left eye were recorded with a video-based eye tracker (SR Research EyeLink 2K,

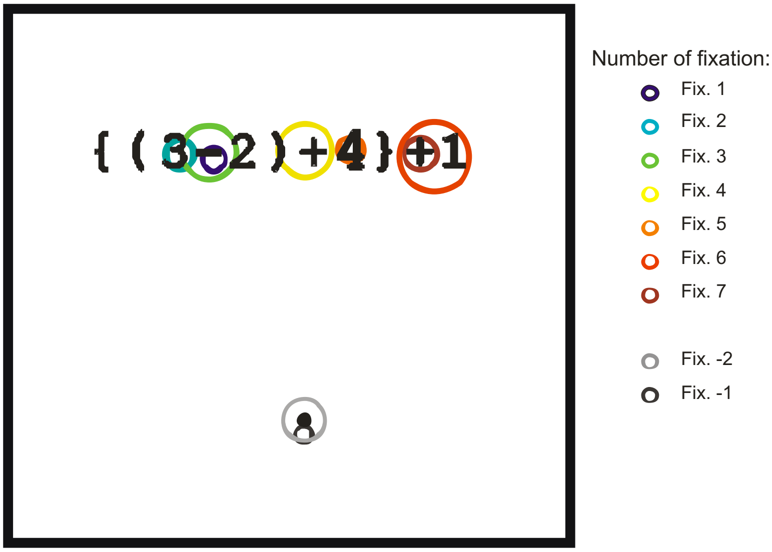

Fig. 1. Sequence of fixations of a representative trial. The temporal order of fixations is color-coded. The diameter indicates the duration of each fixation. Subjects fixate to the center of the screen (grey fixations) before the expression appear and then scan the expression. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

http://www.sr-research.com/) at a sample rate of 2000 samples/s. The participant voice was recorded simultaneously with the recording interface M-Audio Transit Sound Card at a sampling frequency of 44,100 Hz.

3. Experiment 1

3.1. Method

3.1.1. Participants

Thirteen people participated in the experiment 1 (mean age 24).

3.1.2. Stimuli and procedure

Arithmetic expressions were structured in a nested series of three operations delimited by parentheses and brackets. They belonged to two different classes: left branching "$\{(n1 \pm n2) \pm n3\} \pm n4$" and right branching "$n1 \pm \{n2 \pm (n3 \pm n4)\}$". In all cases, the first operation to be performed according to syntactic structuring of the expression (referred as the deepest node of the hierarchy) was of the form ($n_i \pm n_{i+1}$). In the left-branching expressions, this deepest node was positioned at the left of the expression while in the right-branching it was on the right. We denoted left and right branching expressions with L and R respectively (Fig. 2). To keep the calculations simple, the numbers only took values between 1 and 4 (without repetition). In each expression, two of the operators were additions and the remaining one was a subtraction. These constraints define a total of 72 different expressions for each branching class. Participants solved all possible expressions, thus performing a total of 144 trials.

3.2. Results

3.2.1. Vocal response times, error distributions and total number of fixations

We first analyzed the response time (RT) as a function of expression type (Fig. 3a). RTs were significant longer for right branching expressions (mean RT = 5.55 ± 0.56)

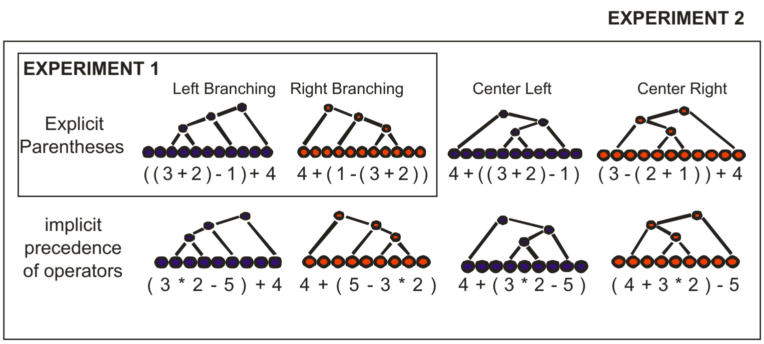

Fig. 2. Experimental design. Experiment 1 investigated formulas with left and right ranching syntactic structures. Syntax was indicated explicitly by parentheses. Experiment 2 extended Experiment 1 in two different dimensions: inclusion of non-linear syntactic structures (CL and CR) and formulas in which syntax is partly determined implicitly by precedence rules (multiplication first). The design is factorial, with syntactic structure (L, R, CL and CR formula) and equation type (*explicit* or *implicit*) as independent factors.

Please cite this article in press as: Schneider, E., et al. Eye gaze reveals a fast, parallel extraction of the syntax of arithmetic formulas. *Cognition* (2012), http://dx.doi.org/10.1016/j.cognition.2012.06.015

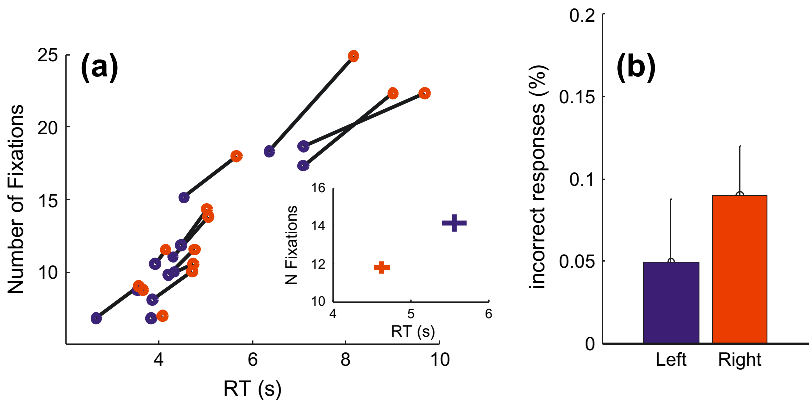

Fig. 3. (A) Effect of syntax on calculation time (RT, x axis) and number of eye fixations (y axis). Each pair of linked points represents the mean data from a single participant, with blue and red symbols indicating respectively left and right-branching formulas. The inset shows the mean and standard error across all participants. (B) Effect of syntax on the fraction of errors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

than for left branching expressions (mean RT = 4.62 ± 0.37 s; paired t-test, $t(12) = 4.45$, $p < 0.001$). This effect was extremely robust: all participants showed slower RTs for right branching than for left-branching expressions. The same pattern was observed when measuring the effect of expression type on the total number of fixations before responding (Fig. 3a) (Average number of fixations for left-branching 11.78 ± 1.17 and for right branching 14.17 ± 1.60; $t = 4.45$, df = 12, $p < 0.001$). For each participant the RTs and the number of fixations were highly correlated: the correlation coefficient between RT and number of fixations across trials averaged across all subjects was 0.93 (significant in every subject with $p < 0.001$). The effect of branching on both measures was greater for subjects with slower RT (Fig. 3a). The mean RT of the six (out of thirteen) slower participants (averaged across branching conditions) was 6.3 ± 0.6 s and of the six faster participants was 3.9 ± 0.2 s, respectively. The effect of branching side for both groups was, respectively, 1.45 ± 0.32 s and 0.48 ± 0.14 s. The correlation coefficient between the mean RT and the effect of branching across all participants was 0.91; a permutation test showed that this correlation was significant ($p < 0.001$).

The previous analysis is based on correct trials. Error rates were, on average, very low and showed the same dependency with syntax as RTs. Errors were more frequent in right-branching expressions (10.3 ± 1.2%) than in left-branching expressions (5.1 ± 1.0%; paired t-test, $t(12) = 3.3$, $p < 0.01$). This result can be explained by the fact that right-branching expressions involve subtractions of bigger numbers and more frequently have negative results (See Experiment 3). An analysis of all the errors pooled together confirmed this trend. The total number of errors were 14 for expressions of the form $n1 - \{n2 + (n3 + n4)$, 9 for expressions of the form $n1 + \{n2 - (n3 + n4)$ and only 2 for expressions of the form $n1 + \{n2 + (n3 - n4)$. While this shows a clear trend, the number of errors was not sufficient to conduct a robust statistic varying this factor for individual subjects.

An analysis of the distribution of numerical distance between erroneous and correct responses ($d_{E\text{-}C}$) revealed that this distribution was not symmetrical. Instead, it was biased towards positive values, mean $d_{E\text{-}C} = 1.98 \pm 0.6$, paired t-test, $t(12) = 3.033$, $p = 0.01$, i.e. participants tended to respond with values which were larger than the correct result. We reasoned that this result may be biased by a tendency of participants to erroneously respond with the absolute value (unsigned response, responding x when the correct response was $-x$). However, the significant bias towards positive $d_{E\text{-}C}$ values persisted when these trials were discarded from analysis.

3.2.2. Sequence of fixations during arithmetic calculation

The distribution of fixations on the equation clustered in three regions, each corresponding to a level in the hierarchic tree. This was clearly seen in the fixation sequence of two representative trials (Fig. 4a) or after merging the fixations of all trials of a representative subject (Fig. 4b). The first fixation on the equation was directed slightly to the left of the center of the expression. Subsequent fixations progressed monotonically starting at the deepest level of the hierarchy and in the appropriate arithmetic sequence, with a strong influence of whether the tree was left- or right-branching.

To quantify this observation, we examined the average position of each fixation in the sequence, for all trials and participants in the experiment (Fig. 4c). The position of the first fixation was unaffected by expression type (Average positions of first fixation, right-branching trials [RB]: 4.7 ± 0.14, left-branching trials [LB]: 4.9 ± 0.27; $t = 1.57$, df = 12, $p > 0.1$). Following this initial fixation, trajectories rapidly bifurcated: LB trajectories progressed slightly to the left (towards the deepest node in the hierarchical tree for LB expressions) while RB trajectories jumped towards the right. This result can explain why RTs for the LB expressions are faster than RTs of the RB expressions. After the first saccade directed spontaneously to the left of the expression, a small correction of eye position is required

Please cite this article in press as: Schneider, E., et al. Eye gaze reveals a fast, parallel extraction of the syntax of arithmetic formulas. *Cognition* (2012), http://dx.doi.org/10.1016/j.cognition.2012.06.015

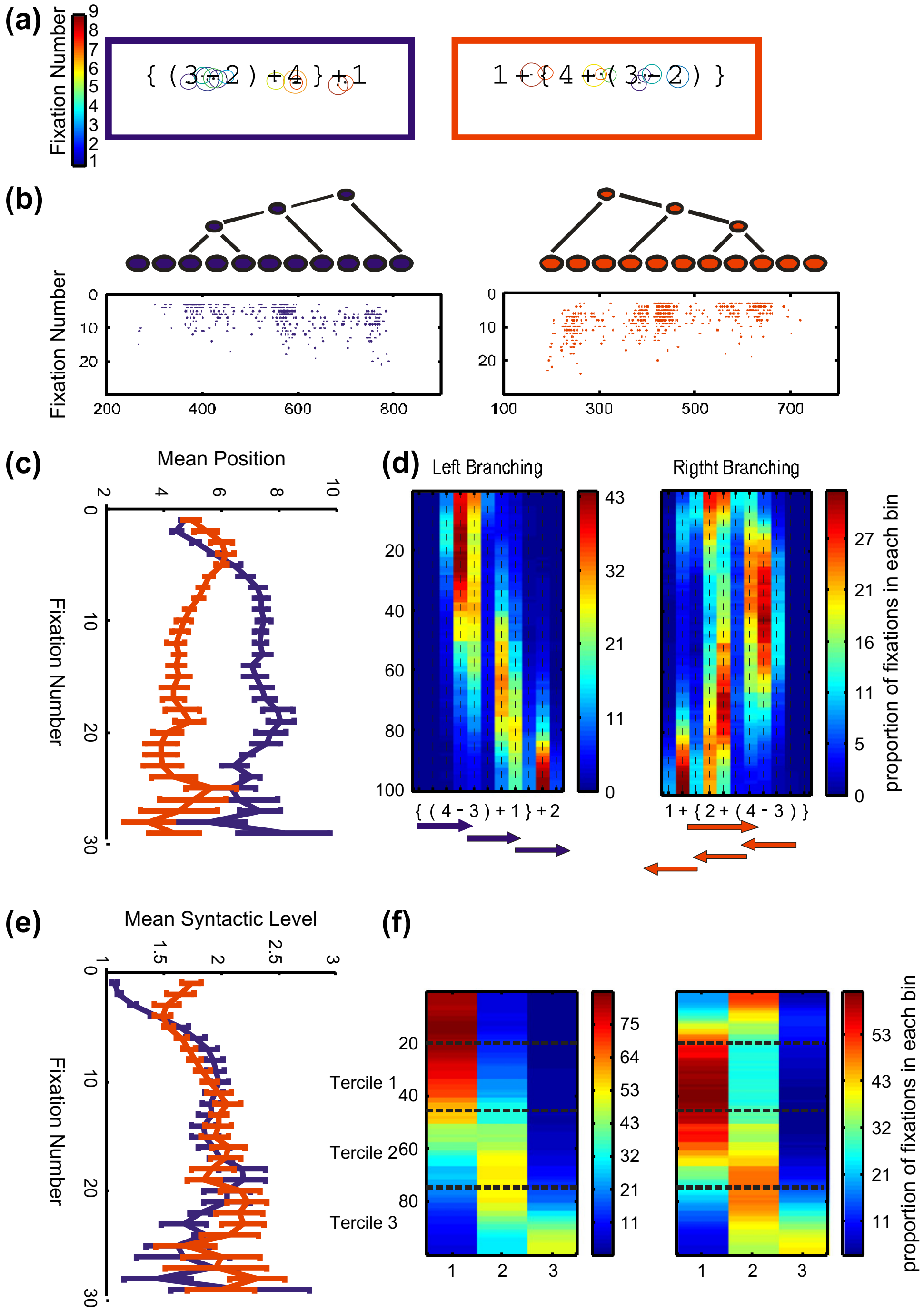

Fig. 4. Sequential progression of fixations through the syntactic levels of the proposed formula. In all panels, eye position data is shown in blue symbols for left-branching formulas, and in red symbols for right-branching formulas. (A) Two examples of single trial data showing how fixations cluster at the arithmetic symbols and in a syntactically relevant order (from left to right for the left-branching formula, and from right to left for the right branching formula). (B) Distribution of fixation positions over all trials in a representative participant for left-branching formulas (left panel) and right-branching formulas (right panel). Horizontal axis indicates each fixation position. The vertical axis indicates the order of fixations on each trial. Note that in both types of formulas, fixations cluster around three locations. (C and E) Mean position (C) and mean level (E) as a function of fixation number, separately for left-branching (blue) and right-branching (red) formulas. (D and F) Distribution of fixated positions (D) and of syntactic level (F) during the course of the trial. In matrices (D and F), the vertical axis is normalized time (fraction of the trial). Colors code the fraction of fixations to the corresponding position (D) or level (F). The arrow lines in panel (F) indicate the three terciles of the trial (excluding the first 20%) considered for statistical analysis. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

for LB-expressions whereas a considerably larger correction is needed for RB-expressions.

While this analysis is quite informative to identify the point at which eye trajectories bifurcate, it considerably blurs the analysis of subsequent fixations since the number of fixations varies widely from one trial to another. To reliably compare sequence dynamics in trials with variable number of fixations, we resampled the data in the following manner: We first normalized trial duration into 100 bins such that bin X corresponds to the time at which X% of the trial has been completed. We then measured for each trial and for each bin the location of the corresponding fixation and finally, for each time bin, we computed a histogram of locations (Fig. 4d). This normalized analysis reflects, in a much clearer manner, the dynamics of fixation during the course of the trial: for both L and R expressions, fixations are directed to positions 4 and 5 of the expression and then progress sequentially through the levels of the expression.

To assert the statistical significance of this observation, we measured, for every participant, the correlation between fixation rank and syntactic level. For Left Branching expressions, the resulting correlation was positive for every single participant in the study (average correlation 0.51 ± 0.04; $t = 10.7$, $df = 12$, $p < 0.001$). For right-branching, 12 out of 13 participants showed a positive correlation between fixation rank and hierarchical level (average correlation 0.25 ± 0.05; $t = 4.3$, $df = 12$, $p < 0.001$). Note that for right-branching, the first portion of the trial involves an adjustment of the first saccade which is directed to the left half of the expression. When we measured the correlation between fixation rank and hierarchical level during the last 80% of the trial, all participants showed a positive correlation for right branching (average correlation 0.37 ± 0.04; $t = 8.7$, $df = 12$, $p < 0.001$).

Detailed examination of the particular locations to which fixations were directed was also informative. First, the data refute the possibility of a sequential and homogeneous scanning of the expression, because fixations are almost never directed to parentheses (<10% of fixations), leaving marked gaps in the scanned area of the expression. Second, within each level, fixations also show a remarkably reliable organization. Fixations to the deepest node (positions 2–6 in LB and positions 6–10 in RB) were biased towards the center of the expression (positions 4 and 5 for LB and symmetrically correspondingly, positions 7 and 8 for RB). We collapsed this histogram across time, counting the total fraction of fixations directed to each of the three different classes of symbols. This was calculated for each participant and then averaged and submitted to statistical analysis. The majority of fixations were directed to the operators. (57 ± 2% of fixations), a substantial but smaller fraction were directed to numbers (36 ± 2%) and only a small fraction (7 ± 1%) were directed to parentheses. To quantify this observation we submitted frequency of fixations to an ANOVA with symbol type (number, operator or parenthesis) as main factor and subjects as random factor. We observed a highly significant main effect of symbol ($df = 2$, $F = 113$, $p < 10^{-10}$). To directly observe whether, for all expressions, the dynamics of eye fixation was primarily governed by syntax (with an initial delay due to the location of the first fixation), we repeated the previous analysis while collapsing spatial locations for each fixation number to their corresponding syntactic levels 1, 2 or 3 (Fig. 4e). We then mapped each percentile of a trial to its corresponding level as done before (Fig. 4f). Both analyses showed that, when analyzed in syntactic space, exploration was virtually identical for left-branching and right-branching expressions, except for a small initial delay.

Finally, we investigate whether the fixation patterns were different in expressions in which the parentheses change the outcome of the calculation (e.g. $2 + (3 - (4 + 1)) = 0$, while $2 + 3 - 4 + 1 = 2$), compared to expressions in which the parentheses have no impact on the outcome (e.g. $2 + (3 + (4 - 1))$). Note that this analysis can only be done for right-branching expressions, since in the left-branching expressions the parentheses never affect the solution. Fixation sequences for expressions where the parentheses changed the solution and where they had no outcome on the final answer were virtually identical, whether observed in real (Supplementary Fig. 1a) or in syntactic space (Supplementary Fig. 1b). To quantify this comparison we submit each fixation to an ANOVA with tercile of the trial (excluding the first 20% of the trial, i.e. fixations belonging to the beginning (20–46.66%) the mid (46.66–83.32%) or in the end (83.32–100.00%) of the trial, Supplementary Fig. 1b), relevance (whether the fixation belongs to a trial for which parenthesis are relevant or not for the solution) as main factors and subjects as random factors. We observed a main effect of tercile ($df = 2$, $F = 64$, $p < 10^{-10}$). The effect of relevance was not significant ($df = 1$, $F = 0.66$ $p > 0.1$) and the interaction between both factors was significant ($df = 2$, $F = 96$, $p < 10^{-10}$), which may reflect the fact that in expressions in which parentheses are relevant, the minus sign distributes on a compound operation and thus may require some backtracking in fixations. However, as seen in Supplementary Fig. 1 the effect is very small and the pattern of fixations look almost identical.

3.3. Discussion

Experiment 1 was conceived to distinguish three possible mechanisms of how syntax is processed in arithmetic expressions compared to language: (H1) Syntax is extracted linearly, scanning the expressions from left to right, as done in language. (H2) Syntax is extracted rapidly and may interact with an existing bias to scan expressions from left to right inherited from language and (H3) Syntax is extracted rapidly and computed without any prior bias inherited from language. Here we outline how our data clearly favors the second hypothesis:

First, analysis of response times and error rates refuted the third hypothesis. All participants responded more slowly and made more errors when calculating expressions which were parsed from right to left. This suggests an imprint of the language system in mathematics (see the general discussion for whether this may relate to syntax, reading, or both).

Behavioral data cannot distinguish whether this effect results from the difficulty of reading an equation contrary to its procedural order as in H1 (and hence having to keep more variables in memory, storing temporary operations, etc.) or instead whether it is the cost of a rapid re-orienting of an initial bias as predicted by H2. However, the eye-movement data discard H1 and are perfectly consistent with H2: The eye-sequence patterns reflect an initial fixation which is identical for all expressions, reflecting a bias towards the left of the equation. Very rapidly (in a few hundred milliseconds) both trajectories bifurcate and reveal patterns which are opposite in space but identically in syntactic space. Because the initial spontaneous fixation coincides with syntactic order in Left Branching expressions, there is no re-orienting cost. In Right Branching expressions, however, the initial fixation does not coincide with the first operation in the sequence defined by syntax and hence participants need to correct. Eye-sequences reflect exactly this pattern, showing that sequences for Right branching expressions are almost identical in syntactic space to left-branching ones, only delayed because the first fixation was not in target.

One possibility is that the differences in RT between left and right branching expressions resides in the fact that left-branching expressions are in fact linear structures while right branching ones are often not. However, this is unlikely given our analysis comparing fixation patterns in expressions in which the parentheses change the outcome of the calculation (strictly hierarchical) compared to expressions in which the parentheses have no impact on the outcome (linear expressions, where subjects could simply execute the operation from left to right) (Supplementary Fig. 1). In fact, the data shows robustly that ocular movements in both classes of expressions are virtually identical. This implies that subjects treat linear structures as if they were in fact hierarchical, indicating that parentheses have a very strong spontaneous influence on how arithmetic expressions are parsed, even when they do not convey any relevant procedural information.

Finally, analysis of individual variability (Fig. 3a) also argues in favor of Hypothesis 2, suggesting that the cost in RT observed when comparing Right to Left branching expressions relates to a postponement in the execution of just a few saccades to correct the initial bias. This predicts that subjects that show a greater difference in RT between both conditions should also show a difference in the number of eye-movements, exactly as described in Fig. 3a.

4. Experiment 2

Experiment 1 did not address whether the observed eye-movement patterns are driven by the abstract hierarchical structures of the equations 'per se' or merely by the presence of explicit visual cues. The equations were always hierarchically organized in terms of parentheses; with 'round' and 'curly' brackets further differentiating between the two nested levels of structural embedding. This raises the question of whether the same results would be obtained if instead of brackets, levels of embedding were distinguished solely by different kinds of operators (taking advantage of the precedence of multiplication over addition). In Experiment 2 we used two types of expressions: (1) expressions using addition and subtraction in which syntax is conveyed by parentheses and brackets (concrete visual cues to parse syntax) as in Experiment 1, henceforth referred as *explicit* and (2) expressions in which the expression includes a multiplication which defines the parse tree based on implicit rules of arithmetic (first multiply and then add or subtract) to which we refer as *implicit*.

Note that as a result of this manipulation, expressions with multiplications also involved larger numbers and hence were expected to generate slower RTs. Experiment 2 was also designed to investigate whether eye movements reflect a hierarchical exploration of the syntactic tree even in non-linear center-embedded structures. In these structures, an exploration of the hierarchical tree requires a first fixation to the center, then to the left and then to the right (referred as Center Right (CR), Fig. 2) or a first fixation to the center, then to the right and then to the left (referred as Center Left (CL), Fig. 2), instead of simply scanning the equation in a constant direction of movements as in L and R-expressions. Experiment 2 was designed in a factorial manner, with syntactic structure (L, R, CL and CR expression) and equation type ("explicit" or "implicit") as independent factors (Fig. 2).

4.1. Methods

4.1.1. Participants

A different group of 13 people participated in Experiment 2 (mean age 23).

4.1.2. Procedure and stimuli

Experiment 2 was an extension of Experiment 1, in which we included two new experimental factors. First, we increased the number of syntactic structures. Experiment 1 only explored linear trees (left to right or right to left, but without insertions). In Experiment 2 we investigated non-linear syntactic structures, in which the first level was in the center of the expression. We denoted these center-embedded expressions central left (CL) branching expressions "$n4 \pm ((n1 \pm n2) \pm n3)$" and central right (CR) branching expressions "$(n3 \pm (n1 \pm n2)) \pm n4$" (Fig. 2). The distribution of numbers and symbols was identical to Experiment 1. In Experiment 2 we also investigated expressions where the syntactic structure was partly implied by operators, i.e. by the precedence of multiplication over addition. In these expressions the deepest levels was defined without parentheses, but solely using a multiplication symbol (Fig. 2). Experiment 2 was therefore organized in a factorial design, with four syntactic structures (LB, CL, CR and RB) and two indicators of syntax, parentheses and multiplications which are henceforth referred as *explicit* or *implicit* simply to denote that the former provide easily detectable visual cues to parse the equations and the latter involve in part a conventionally abstract system to define syntax. The four types of *implicit* expressions were: left branching "$(n1 * n2 + n3) + n4$", right branching "$n1 + (n2 + n3 * n4)$", center left branching expressions "$n4 + (n1 * n2 + \mathrm{n}3)$" and center right branching expressions "$(n3 + n1 * n2) + n4$".

In *implicit* expressions we used numbers two to five to avoid the trivial multiplication by one. The results of these equations are consequently numerically larger than those of expressions constructed using only plus and minus symbols. Each experiment had 200 trials, 25 trials of each structure which were chosen randomly from the 72 possible combinations.

4.2. Results

4.2.1. Vocal response times error rates and total number of fixations

RTs were calculated within each condition and each subject, and submitted to ANOVAs with subjects as a random factor and syntactic structure and equation type as within-subject factors. The ANOVAs on RTs revealed an effect of both main factors. The effect of equation type was significant (df = 1, $F = 9.82$ $p < 0.001$). As expected, equations which involved multiplications were responded more slowly (Fig. 5a). The effect of syntactic structure was also significant (df = 3, $F = 9.89$ $p < 0.001$) although the variation with the different syntactic trees was more pronounced when syntax was purely delimited by parentheses, as in Experiment 1, as revealed by an interaction between these two factors (df = 3, $F = 2.79$, $p < 0.05$). This interaction may indicate that, although symbolic precedence is also extracted very rapidly, parentheses (and probably other Gestalt makers of syntax) make the syntactic parsing process more efficient. This is compatible with the idea suggested by Landy et al. (2008) and Landy and Goldstone (2010) that we selectively attend to high-precedence operations automatically. Consistently with what we had observed in experiment 1, the majority of fixations were directed to the operators. (52 ± 3% of fixations), a substantial but smaller fraction were directed to numbers (30 ± 2%) and only a small fraction (18 ± 2%) were directed to parentheses. To quantify this observation we submitted the frequency of fixations to an ANOVA with the symbol type (number, an operator or a parenthesis) as main factor and subjects as random factor. We observed a highly significant main effect of symbol (df = 2, $F = 37$, $p < 10^{-10}$).

Similarly, the number of fixations was calculated within each condition and each subject, and submitted to ANOVA with subjects as a random factor and syntactic structure and equation type as within-subject factors. The number of fixations varied with syntactic structure (df = 3, $F = 9.31$ $p < 0.001$). The dependencies of the number of fixations and of RT with syntactic structure were very similar. However, in striking contrast with RT, there was virtually no effect of equation type on the number of fixations (Fig. 5b, df = 1, $F = 0.1$ $p > 0.1$). The interaction between both factors for fixation number did not reach significance (df = 3, $F = 2.02$, $p > 0.1$).

An analysis of the error distribution yielded a very similar pattern (Fig. 5c) than for the number of fixations. Error rates were calculated within each condition and each subject, and submitted to ANOVAs with subjects as a random factor and syntactic structure and equation type as within-subject factors. Error rates varied with syntactic structure (df = 3 $F = 3.03$ $p < 0.05$). As observed for the number of fixations, error rates were not sensitive to equation type (df = 1, $F = 1.49$ $p > 0.1$). These results stress the importance of syntax since errors are not affected by the operation type (even if it involves greater numbers) but on the contrary is quite sensitive to the syntactic organization of the expression. The interaction between both factors for fixation number was not significant (df = 3, $F = 0.62$, $p > 0.1$).

Together, these results indicate that, when solving an arithmetic expression, the number of fixations depends on the syntactic structure but not on the intrinsic complexity of each operation within the expression (i.e. if it is an addition or a multiplication). Response Time is affected by both factors, implying that the duration of each individual fixation increases with the complexity of the operation. This is testified by a measure of fixation duration as a function

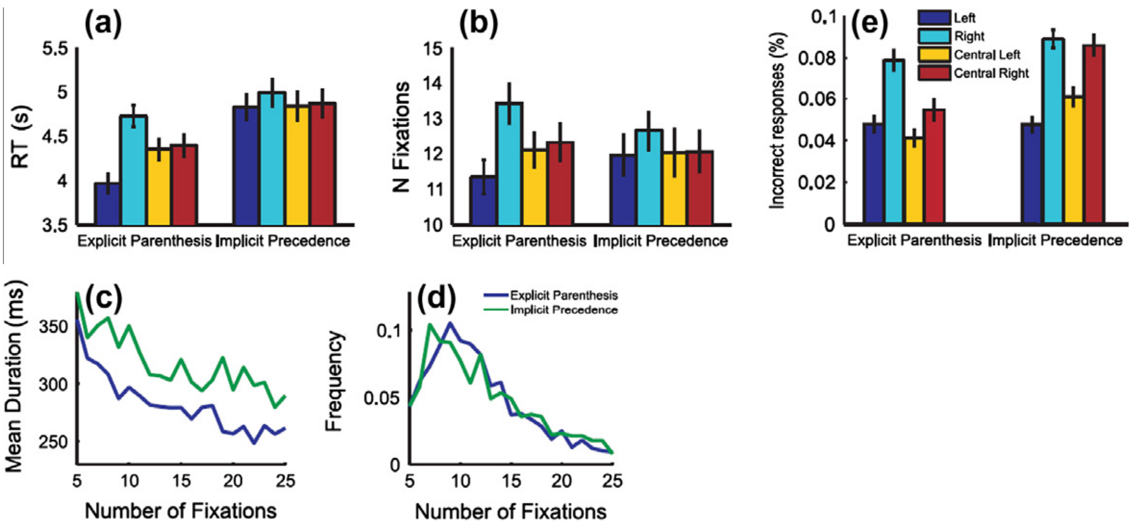

Fig. 5. Effect of syntax and syntactic marker on calculation time, number of eye fixations and errors (A–C). Calculation time is affected by both factors (A). The number of fixations (B) and error rates (C) are mainly determined by syntactic structure (B). The mean duration of each fixation decreased with the total number of fixations in each trial and was slower for *implicit* than for explicit formulas (D). The distribution of number of fixations per trial was not different in *implicit* and *explicit* formulas (E).

Please cite this article in press as: Schneider, E., et al. Eye gaze reveals a fast, parallel extraction of the syntax of arithmetic formulas. *Cognition* (2012), http://dx.doi.org/10.1016/j.cognition.2012.06.015

of the numbers of fixation, which revealed an exponential decaying function (trials with more fixations have comparatively slower fixations) and an additive effect of around 30 ms per fixation of equation type (Fig. 5d). In comparison, the distribution of trials with a given number of fixations was virtually identical for both equation types (Fig. 5e).

4.2.2. Sequence of fixations during arithmetic

To compare eye movement dynamics in trials with variable number of fixations, we proceeded as in Experiment 1: We first normalized trial duration into 100 bins such that bin X corresponds to the time at which X% of the trial has been completed. We then measured the location of the corresponding fixation for each trial and each bin, and finally we computed a histogram of locations for each time bin (Fig. 6, rasters). Normalized analysis revealed that eye-movements reflected the syntactic organization of expressions for all the classes explored in this experiment. We replicated experiment one (Fig. 6, first row, first and second raster starting from the left) and we extended this observation to center–left and center–right trees (third and fourth column). *Implicit* and *explicit* equation showed very similar patterns. For each expression type we calculated the syntactic level during the course of the trial and averaged it for all trials and subjects to obtain the mean syntactic level at different stages of the trial. This analysis revealed a monotonic exploration of the tree (excluding the first 20% of the trial where, as in Experiment 1 the first fixation is directed to a fixed position in the expression). The patterns of exploration were very similar for all equation types, with the exception of the CL and CR expressions of *explicit equations* (Fig. 6, first row, third and fourth column) for which the last level of the hierarchy was reached later in the trial. This observation is confirmed by an ANOVA analysis in which we analyzed mean syntactic level with subjects as a random factor and three independent within-subject factors: syntactic structure, equation type and the tercile of the trial (excluding the first 20% of the trial, as in Experiment 1). All factors were significant (tercile, df = 2, $F = 80$, $p < 10^{-10}$, syntactic structure df = 3, $F = 7.1$, $p < 0.001$ and equation type df = 1, $F = 12.6$, $p < 0.001$). All interactions were also significant. Fixation sequences for correct and incorrect trials showed very similar patterns (Supplementary Fig. 2a and b). We confirmed this observation with an ANOVA with tercile of the trial (whether fixations belong to the beginning (20–46.66%) the mid (46.66–83.32%) or in the end (83.32–100.00%) of the trial, Supplementary Fig. 2b) and correctness (whether the answer was correct or not) as main factors and subjects as random factors. We observed a main effect of tercile for left and right branching structures (Left: df = 2, $F = 88$, $p < 10^{-10}$; Right: df = 2, $F = 26$, $p < 10^{-10}$). The effect of correctness did not reach significance (Left: df = 1, $F = 0.09$ $p > 0.1$; Right: df = 1, $F = 0.03$ $p > 0.1$). The interaction between both factors was not significant (Left: df = 2, $F = 1.79$ $p > 0.1$; Right: df = 2, $F = 0.27$ $p > 0.1$).

4.3. Discussion

Experiment 2 confirms the main-result of experiment 1 – a rapid extraction of the syntactic backbone is reflected in the eye-movement sequence – and extends it to syntactic trees with non-linear center-embedded structures. As in experiment 1, we observe a bias towards the left which is rapidly corrected to accommodate the syntactic ordering of the expression. This shows that rapid extraction of syntax is not limited to the identification of a left-to-right or right-to-left parse, but generalizes to the detection of more complex spatial layouts of a hierarchical syntactic tree. Experiment 2 also shows that rapid extraction of syntax does not require explicit parentheses and still occurs very rapidly when syntax is determined by precedence rules. These two results can be simply seen as a generalization of Experiment 1. They support hypothesis H2 (an initial language bias followed by the fast parallel extraction of syntax, independently of spatial layout) extends in a wide range of parameters, including: (1) branches which are not monotonically arranged in space and (2) non-explicit markers of syntax. The latter constitutes an important handle to investigate the development of this rapid parsing ability, since implicit markers are clearly cultural conventions highly practiced during education. Future perspectives are analyzed in the general discussion.

Our results also show that participants fixate much more frequently the operators than the parentheses. In pure syntactic reasoning, this observation seems to reflect an interesting departure between gaze and normative functions. Both operators and parentheses serve to parse the syntactic tree, and yet participants are more likely to attend to the operators than the parentheses. At least two non-exclusive interpretations may be proposed for this observation: (1) symbols are centered in the arithmetic phrases while parentheses delimit their borders, and gaze is naturally directed to the center of constituents; or (2) symbols convey information about syntax but also about the specific operation which has to be performed, and gaze is biased to the symbols carrying semantic information about operations. Future experiments can be done, manipulating the spatial proximity and layout of symbols – departing them from the center of constituents – to disambiguate between these hypotheses.

Finally, our data revealed that expressions which involve multiplications and hence larger numbers are solved more slowly but do not require more fixations. A double dissociation is observed, in which the number of fixations is determined by the syntactic backbone while the duration of each fixation is modulated by the magnitude of the numbers involved. This observation suggests that the solution of an arithmetic program can be described by a simple architecture in which each fixation corresponds to one operation. In this model, the number of fixations may vary in different layouts as described in Experiment 1 and 2, to accommodate for initial biases. It predicts that the duration of each fixation in one level should be accounted by the time it takes to calculate the corresponding result. Experiment 3 aimed to corroborate this prediction by investigating whether the time spent solving an arithmetic expression can be accounted by the duration of the internal operations involved at each step of the sequence or if, instead, the interactions between processing steps blur this dependence.

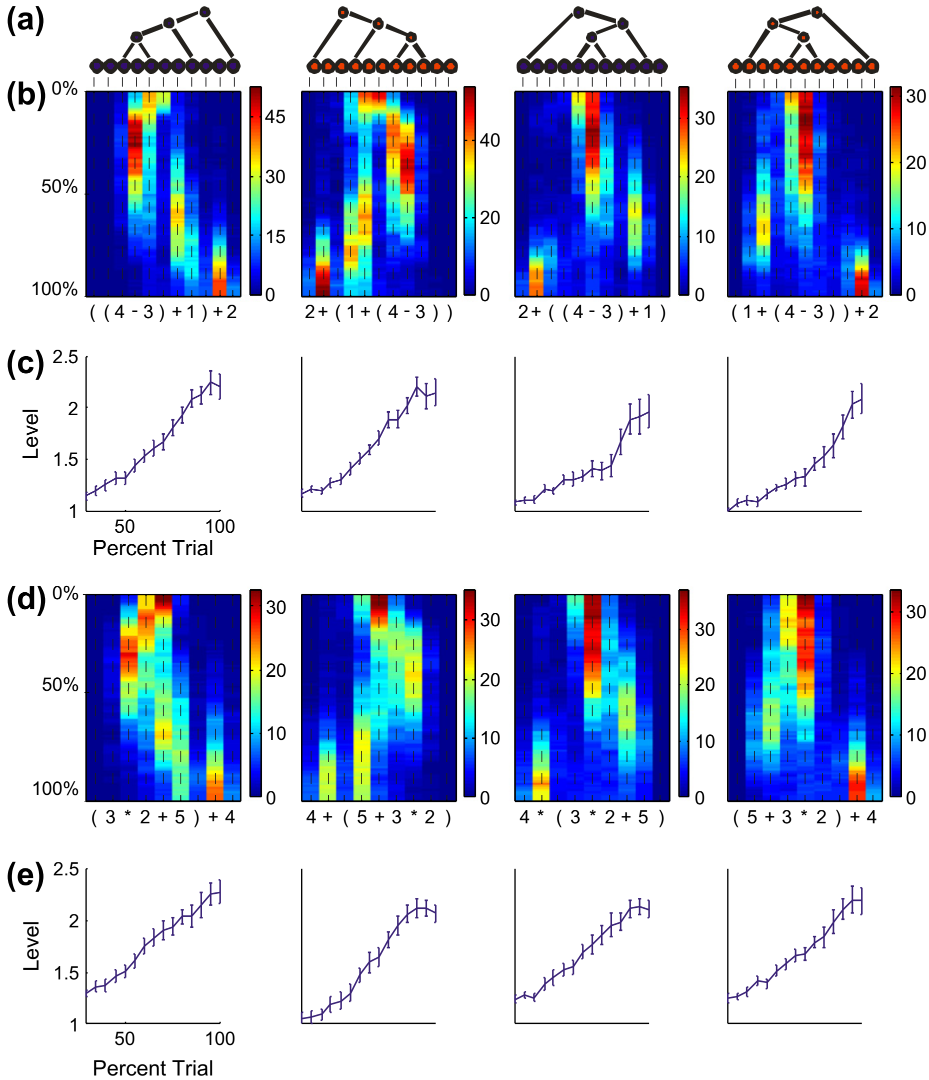

Fig. 6. Sequential progression of eye fixations. (A) Sketch of the four syntactic structures. (B and D) Mean position as a function of fixation number, for *explicit* (B) and *implicit* formulae (D). Each column corresponds to the syntactic structure described in (A). (C and D) When projected to syntactic levels *explicit* (C) and *implicit* (E) formulas reflect a monotonic progression (excluding the first 30% of total trial time in which the first saccades were directed to a position independent of syntax). Colors code the fraction of fixations to the corresponding position. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

5. Experiment 3

Algebra and arithmetic have been considered a case study for the investigation of tasks involving nested sequences of operations where intermediate results are stored and reused in subsequent steps (Anderson & Lebiere, 1998). Multi-step human cognition has been modeled by applying the computer-science notion of "production system" (Anderson et al., 2004). It consists of a general mechanism for selecting production rules fuelled by sensory, motor, goal and memory modules. These models, as well as other similar enterprises (Meyer & Kieras, 1997) emphasize the chained nature of cognition: at any moment in the execution of a task, information placed in buffers of specialized modules acts as data for the central production system, which in turn outputs new information to the buffers. Mental chronometry partially validated this approach by showing that total calculation time can be approximated by a sum of multiple stages, each sensitive to the duration of individual calculations (Ashcraft & Stazyk, 1981; Sigman & Dehaene, 2005; Timmers & Claeys, 1990; Widaman, Geary, Cormier, & Little, 1989; Zylberberg, Fernandez Slezak, Roelfsema, Dehaene, & Sigman, 2010).

Experiment 3 aimed to relate this serial chaining of operations to the syntactic organization investigated in Experiments 1 and 2. In these experiments we showed that participants can extract rapidly and in parallel the syntactic backbone needed to precisely program the sequence of operations. Here we examine whether the solution of arithmetic expressions involve a strictly serial execution of each arithmetic step. The prediction of this hypothesis is that the duration in each level should be tightly correlated with the duration of the corresponding isolated operation. As in Experiment 1 and 2, we used expressions which required a total of 3 binary arithmetic facts. For instance $3-(2+(4+1))$ can be parsed in the following sequence: $4+1=5$; $2+5=7$; $3-7=-4$. In Experiment 3 participants solved all possible LB arithmetic expressions presented in experiment 1, plus all of the corresponding isolated arithmetic problems, allowing the former to be regressed on the latter.

5.1. Methods

5.1.1. Participants

Nine people participated in Experiment 3 (mean age 24).

5.1.2. Procedure and stimuli

Experiment 3 was conducted to measure the covariation of the RTs to simple arithmetic facts and to the complex arithmetic expressions used in Experiment 1. On each trial participants solved either an arithmetic fact (i.e. $3+2$) or a three level expression i.e. $((3+2)-1)+4$. Arithmetic facts included all the operations appearing during the hierarchical resolution of one or more of the syntactic trees (e.g. for $((3+2)-1)+4$, the arithmetic facts are $3+2$, $5-1$ and $4+4$). Participants solved all the possible left branching structures twice. Therefore in every experiment there were 144 left branching trials. There are 63 different arithmetic facts which result from unfolding all 72 three-step expressions. During the experiment, each participant solved each of these facts five times. Consequently each experiment consisted of 459 trials, 144 left branching expressions and 315 facts.

The 63 facts were grouped in nine classes of operations depending on whether or not they involved negative numbers, whether one of the addends was a zero, whether the two numbers in the operation were identical, and whether the result was positive or negative. This resulted in a total of 9 classes, which qualitatively represented the different types of operations: (1) Additions with non-identical numbers $(1+2)$, (2) additions with identical numbers $(2+2)$, (3) additions of zero $(3+0)$, (4) subtractions of identical numbers in $+\,-$ order $(3-3)$, (5) subtractions of identical numbers in $-\,+$ order $(-3+3)$, (6) subtractions $+\,-=+$ (i.e. $3-2=1$), (7) subtractions $+\,-=-$ (i.e. $2-3=-1$), (8) subtractions $-\,+=+$ (i.e. $-2+3=1$), and (9) subtractions $-\,+=-$ (i.e. $-3+2=-1$).

Not all classes are performed in all levels, for instance operations with equal operands or with zero as an operand are never performed in the first step, which can be any addition or subtraction of two different numbers between one and four. For the regression analysis of experiment 3 we grouped measures from all levels together. The results remained unchanged when the regression depended on the syntactic level of the arithmetic calculation in which the operation was performed.

5.2. Results

We first classified all the binary operations in nine categories, based on the type of operation and the sign of the numbers involved (see methods and Fig. 7a). Response time was faster for classes involving additions only, than for any other class of operations; the fastest class corresponded to an addition of zero, which occurs for instance in the last step of $4+(3-(2+1))$ which is $4+0$. The slowest category was a subtraction in which the subtracted number is first (or an addition in which the first term is negative) and which has a negative result $(-3+1)$.

To model the variability of this data, following the procedure described by Aschraft and colleagues (Ashcraft, 1992; Ashcraft & Stazyk, 1981), we performed a multiple regression using five binary factors: (1) Whether the operation involves only positive numbers, (2) whether one of the two terms is zero, (3) whether the two terms were identical, (4) whether the negative number was the first or the second (i.e. $-2+3$ or $3-2$), and (5) whether the result was positive or negative. To avoid performing too many regressions, we modeled the main effects but not the interactions. This was clearly sufficient to show that the numerical complexity of each operation affect response time. This analysis revealed that all these factors were significant (the p values of all factors were <0.01) and had expected main effects on RT. Expressions involving additions of positive (non-zero) addends were faster by 95 ± 10 ms than expression involving subtractions or additions of a positive and a negative number. The occurrence of a zero term reduced RTs by 85 ± 35 ms. Additions were 54 ± 20 ms faster when the two numbers were identical. Subtractions in which the negative number was the first operand were 152 ± 57 ms slower than RT in subtractions where the negative number was in the second position. Operations for which the result was negative were 87 ± 36 ms slower than those for which the result was positive.

Having classified the dispersion of response times to individual arithmetic facts, we then investigated whether the time to perform each step of the expression could be accounted by the time to perform the corresponding operation. By definition, in multiple-step arithmetic one does not have direct access to the time involved in each step since a response is emitted when all the steps have been completed. However, based on the results of Experiment 2 we hypothesized that the total time of gaze in each syntactic level constitutes a marker of the time spent in this step of the sequence. To test this hypothesis we performed a linear regression $RT_{single}(i) = \alpha + \beta \cdot RT_{compound}(i)$ where i indexes the nine classes of arithmetic operations defined above, RT_{single} is the mean RT for all the arithmetic facts in the given class, and $RT_{compound}$ is defined as the total gaze time for which the corresponding operation belonged to a class (Fig. 7b). The linear regression was performed independently for each participant. The resulting mean values of the regression were $\alpha = -362 \pm 82$ ms and $\beta = 0.94 \pm 0.08$. These results indicate that total gaze time to a step of the calculation scales linearly with the time it takes to perform the operation in isolation, with a slope that does not differ significantly from 1 ($p > 0.1$). This observation validates our hypothesis that gaze duration relates to the calculation time at a given step. The intercept, however, indicates that on average, gaze time during the compound calculation is much shorter that the calculation time for a single arithmetic fact, indicating that the transi-

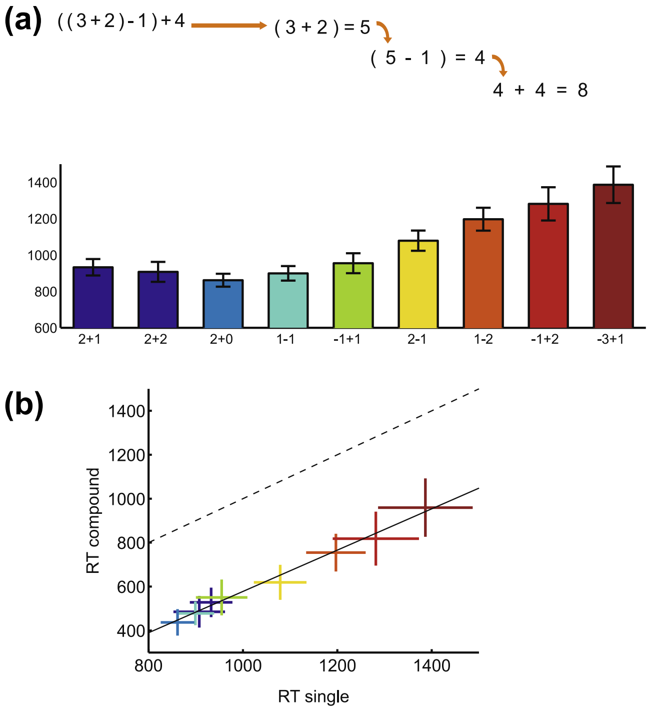

Fig. 7. Unfolding the resolution of arithmetic formulas as a sequence of individual arithmetic operations. (A) Duration of calculation time for nine different classes of operations. An exemplar is given as a legend and the description of the category is found in the methods section. (B) For each class, we averaged the time spent at the corresponding syntactic level during the arithmetic formula (ordinate) and compared it with the time needed to solve the same calculation problem in isolation (abscissa) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

tion from one calculation step to the next (as indexed by saccades) is considerably more rapid than the time it takes to receive the single problem visually and to manually convey the response.

5.3. Discussion

This experiment confirmed that, as suggested by the results of experiment 2, the time of gaze at each syntactic level constitutes a good marker of computation time at this step of the sequence. This is inline with the prediction derived from our hypothesis that syntax is extracted rapidly and configures a sequence of operations which is then executed serially without additional cost. Our observations are incompatible with alternative architectures. For instance, it rules out the possibility that subjects move the eyes from one level to the other before having completed the calculation. A similar spillover effect has been found during reading (Binder, Pollatsek, & Rayner, 1999; Schroyens, Vitu, Brysbaert, & D'Ydewalle, 1999). Similarly, our observation is incompatible with a model in which subjects first scan and encode the entire expression and only then compute the series of arithmetic facts. Beyond the theoretical implications, this specific result has also methodological relevance, since it shows that eye-movements can be used to tag individual arithmetic operations which are embedded in multiple-step arithmetic where, by definition, one does not have direct access to the time involved in each step.

6. General discussion

6.1. Main theoretical conclusion: extracting syntax at a glance

We showed that fixation sequences during arithmetic calculations are highly stereotyped and universal in a population of subjects with substantial mathematical training. The first fixation is directed slightly towards the left of the center of the expression. Trajectories then rapidly bifurcate in accordance with a hierarchical exploration of the syntactic tree. These trajectories are not affected by whether syntax is conveyed by brackets and parentheses, or by the precedence of arithmetic operators, and are only affected by syntactic structure and not by the total calculation time. This data coherently shows that the human brain can extract, at a glance, the hierarchical structure of an arithmetic expression. As opposed to language, where a sentence has to be scanned progressively while the nested syntactic structure is being computed, syntax in arithmetic can be extracted automatically and in parallel. Our study also reveals an initial bias towards expressions that unfold from left to right, which is rapidly corrected for expressions with other spatial layout.

Together our data can distinguish between three alternative hypotheses which motivated this study: (H1) mathematical expressions are similar to language: an equation has to be scanned from left to right, sequentially; (H2) the syntactic structure can be extracted very quickly and in parallel, but the typical left-to-right sequential organization of language is imprinted in mathematical thinking. This hypothesis predicts an initial bias for left to right parsing which is rapidly corrected; or (H3) the syntactic structure can be extracted in parallel, and language sets no bias on how mathematical expressions are read. Our observations are highly compatible with H2 and incompatible with the other views.

The observation of a rapid identification of syntax fits well with a series of studies by Landy and Goldstone (2007a, 2007b), and Landy et al. (2008), who have demonstrated that humans selectively attend to high-precedence operations automatically and direct the first fixations towards the multiplication symbol during the solution of "addition–multiplication" or "multiplication–addition" expressions (Landy & Goldstone, 2007a, 2007b, 2010; Goldstone et al., 2010; Landy et al., 2008). Experiment 2 shows the generality of this phenomenon, demonstrating that a rapid structuring of eye movements follows syntax in a broad variety of hierarchical trees and using different markers to convey syntax. More generally, our results are consistent with the theoretical proposal that mathematical notation is founded upon early perceptual mechanisms yielding efficient pre-attentive and parallel mechanisms for rapid parsing of mathematical structures (Goldstone & Barsalou, 1998).

Experiment 3 may explain why the syntactic backbone of the expression is extracted rapidly and in parallel. The regression analysis indicates that the time spent at each level of an expression is tightly correlated, with a slope of 1, with the time it takes to compute the corresponding arithmetic fact in isolation. This is consistent with a strict sequential unfolding of a nested series of operations. Instead, it is incompatible with models in which processing time of certain steps is slower (than the fact in isolation) while queued until syntax is extracted, as observed in sentence comprehension (Waters, Caplan, & Hildebrandt, 1987).

Our observations are also incompatible with distributed or cascaded processing mechanisms (an operation is performed through many fixations) which have been observed in recent studies of eye-movements during reading (Kliegl, 2007; Kliegl, Nuthmann, & Engbert, 2006) or, similarly, with architectures based on accumulation of memory buffers or other non-stationary mechanisms (such as fatigue) which predict that the last steps of the calculation should take more time than when performed as isolated facts.

This data instead support a simple architecture by which the order of operations is quickly identified and then executed serially, step by step, with no further delays. This architecture brings together the ideas of Goldstone and colleagues of a saliency of syntactic markers with the notions of cognitive architecture based on "production systems" (Anderson et al., 2004). This architecture is equivalent to a Turing Machine (Zylberberg et al., 2010), gaining computational versatility by relying on a sequential implementation of simple steps. The sequence of steps is determined by a program, whose role here is played by the syntax of the expression. As an interesting historical coincidence, in the implementation of his celebrated computing device, Alan Turing did not conceive an abstract thinking machine. Instead, he developed a device observing his own thought, explicitly attempting to build a machine which could imitate a mathematician (Turing, 1938).

6.2. Spatial asymmetries in mathematical notation

The asymmetry observed between the first fixation of left- and right-branching expressions (Figs. 2 and 3) might suggest an influence of language in the processing of algebraic expressions. All our participants were native Argentinean Spanish speakers. Spanish is read from left to right, which can explain the observed bias for left-branching structures. Another possible bias may originate in sentence structure: Spanish has a subject–verb–object (SVO) sentence structure (as opposed to Turkish, Basque or Japanese among others which are subject–object–verb (SOV) languages) which may favor a left-starting parse. Our data are not conclusive as to which of these two aspects of language contribute to the slight leftward bias of the first fixation to arithmetic expressions.

The leftward bias may also originate from the non-commutative nature of some arithmetic operations ($3 - 2$ is different from $2 - 3$) as well as their non-associative nature; $2 * 3 + 4$ is different from $2 * (3 + 4)$. In this case, by convention, operations proceed from left to right (a convention which in turn may also originate from a language bias; see below). In the future, it may be possible to investigate these hypotheses by studying arithmetic in reverse polish notation, where an operator follows all its operands (i.e. $2\ 6 - 1\ +$ denotes: subtract 6 from 2 and add 1 to the result).

The leftward bias may also be specific to the spatial organization of mathematical notation. Much of the development of mathematical notations occurred in Latin (Cajori, 1929), and as a result, algebraic notations are most often organized in a left-to-right and top-to-bottom fashion. These cultural choices, which are probably largely arbitrary, set a concrete spatial layout for the organization of arithmetical thought as demonstrated by several findings. Kirshner (1989) demonstrated that spatial information plays a role in processing mathematical notations, because participants performed better when processing expressions that maintained the spacing found in algebraic notation (Kirshner, 1989). Landy and Goldstone (2007b) further showed that spatial proximity, as well as other notions of perceptual similarity, guide parsing of expressions and equations (Landy & Goldstone, 2007b). For instance subjects are likely to compute $5 * 5$ in response to "$2 + 3 * 5 = ?$" despite the procedural rule which states that multiplication precedes addition, simply because of the proximity of the addends 2 and 3. This error is much less frequent when responding to "$2 + 3 * 5 = ?$" when spatial proximity is congruent with procedural rules (Landy & Goldstone, 2007b; Goldstone et al., 2010; Landy et al., 2008).

Indirect measures of spatial mental movement during algebraic thinking can also be obtained by superposing an equation like "$4 * Y + 8 = 24$" with a moving grating (Goldstone et al., 2010). Accuracy increases if the movement is to the right, when the direction of motion is congruent with the algebraic transformation needed to isolate the *Y* on the left side (the 8 and then the 4 must be "moved" to the right-hand side of the equation). In a comparable study, Jansen et al. (2003) used a Restricted Focus Viewer (RFV), where participants only see a small region of the equation at a time. Compared to our study, Jansen and colleagues also changed the original fixation (to the corner instead of the center of the equation as used here) and the specific task (participants read the equation to later determine whether a statement was true or false). Interestingly, even under this restricted circumstances, participants showed a scan path which was heavily determined by syntax, although with a stronger LR bias than we observed (which was only seen in the first fixation).

The interaction between the spatial layout of the expression and its syntactic structure are reminiscent of the tight relation between syntax and prosody observed in spoken language (Mehler, Nespor, Shukla, & Pena, 2006; Nespor & Vogel, 2007). It argues in favor of a tight relationship between perceptual organization and conceptual reasoning (Goldstone & Barsalou, 1998). According to this view, the cues needed to parse mathematical expressions do not rely on purely amodal and abstract symbol representations (Fodor, 1975) but instead recycle evolutionarily older mechanisms of perceptual grouping (Goldstone & Barsalou, 1998; Dehaene & Cohen, 2007; Goldstone et al., 2010). This point seems quite evident when syntax is indicated by parentheses, which constitute one of the earlier examples used by explicit psychologists to indicate perceptual grouping and segmentation (Wertheimer, 1938; Westheimer, 1999). For instance the sequence [ ] [ ] [ ] is perceived as three objects because the open and close bracket are grouped spontaneously (Kofka, 1935). Our observation that eye-movement patterns are virtually identical in equations in which parsing is encoded by brackets and parentheses or by precedence of operators suggests an additional role for perceptual learning as a relevant aspect of mathematical training (Goldstone et al., 2010). Hence, perceptual learning mechanisms may extend beyond the early assimilation of regularities (Sigman, Cecchi, Gilbert, & Magnasco, 2001) and have a direct impact on reasoning and on the organization of cultural conventions including reading and mathematics (Goldstone & Barsalou, 1998; Gilbert, Sigman, & Crist, 2001; Sigman & Gilbert, 2000; Dehaene & Cohen, 2007; Goldstone et al., 2010).

6.3. Universal syntax? Comparing syntax in arithmetic and language

Is the syntax of arithmetic closely related to the syntax of language? In his pioneering study, Ranney made a strong claim about the relative advantages of using algebraic expressions to investigate syntax (Ranney, 1987). "The syntax of algebra places certain conventional ordering constraints on its elements, much as English syntax does [. . .] It is algebra's compactness, however, that makes it particularly appropriate for the study of the perceptual effects of structural context. Even rather complex expressions can be formed with the appropriate concatenation of relatively few characters. In algebra, then, we have a formal, syntactic language that is compact enough to be briefly perceived (i.e., with fewer eye fixations and saccades than are required to perceive English sentences)."

Ranney's remarks refer to the spatial layout of algebra which presents substantial differences with the arithmetic expressions studied here (etymologically arithmetic relates to *counting* and algebra to *movement* or *reunion of broken parts*). Nevertheless, algebra and arithmetic use a similar spatial layout and both are, in the words of Ranney, "a formal, syntactic language that is compact enough to be briefly perceived". Hence, while the extension of the observations of this study to algebra should be a matter of further study, here we built upon some of the similarities between arithmetic and algebra which result from their spatial representation and symbolic expression.

Our study can be seen as an empirical corroboration, in the domain of arithmetic, of Ranney's intuition: language and mathematics differ in the speed and compactness with which they convey syntax. Written forms of natural language are read primarily in a specific direction (left-to-right in English), with occasional backtracking to regions of the text that have already been scanned (Rayner, 1998). Syntax therefore primarily impacts on the latencies of sequential scanning, but not on the order of fixations (Frazier, Carminati, Cook, Majewski, & Rayner, 2006; Rayner, 1983). Instead, here we show that syntactic structure during arithmetic is directly reflected in the sequence of fixations itself, reflecting a global extraction of syntax immediately after the first fixation to the equation.

In the future, a comparative analysis between language, arithmetic and algebra may constitute a useful vehicle to probe syntactic universals. Here, we merely make a few observations. An important difference between syntax in arithmetic is the necessity for explicit syntactic markers such as parentheses. In language, such an explicit marking of syntactic trees is often unnecessary (although prosody plays a role), among other things because words provide syntactic tags (they are verbs, nouns, adjectives. . .). Hence, in the sentence "John quickly decides to walk", or even its Jabberwocky equivalent "John pradly hinfers to jolk", word order, grammatical category and morphological endings serve as markers for syntax. In mathematical notation, inflectional morphology is very rarely used. The sequence 2 3 is completely ambiguous and hence requires explicit operators to assign syntax to this sequence of numbers.

In natural languages, branching patterns have been identified as a factor influencing syntactic complexity since the pioneering studies of Miller and Chomsky (Miller & Chomsky, 1963). For instance, subjects make more errors and take longer to process sentences that contain center-embedded relative clauses than sentences that contain right-branching relative clauses (Waters, Caplan, & Hildebrandt, 1991) which is believed to result from the memory load associated with holding the unassigned phrase in a parsing buffer until it is assigned a thematic role, posing specific comprehension difficulties to aphasic patients suffering from a lesion in Broca's region (Ben-Shachar, Hendler, Kahn, Ben-Bashat, & Grodzinsky, 2003; Grodzinsky, 2000; Grodzinsky & Finkel, 1998; Grodzinsky & Friederici, 2006). This constitutes an example of a more fundamental process, referred as syntactic movement, which describes the fact that in all natural languages constituents may be displaced from the position where they receive features of interpretation (Chomsky, 1975, 1995). The origin of syntactic movement remains an open theoretical discussion (Moro, 2000). A recent view, developed by Andrea Moro, is that it is driven by purely structural reasons related to the necessity to linearize words in a string. The hypothesis is that the structural rationale of movement is to reduce symmetry created by the binary recursive operation (*merge*) which assembles words together (Moro, 2000). A potentially interesting domain to probe these ideas is arithmetic in reverse polish notation in which, contrary to standard notation, the operator is in the postfix position. For instance 2 5 + denotes 2 + 5. Interestingly, this notation (assuming that all operations are binary) reduces the necessity for syntactic markers. For instance, the arithmetic phrase: "2 3 4 $*$ $-$ " is read unambiguously as "2 (3 4 $*$) $-$ ". Hence an analysis of eye-movement in reverse polish notation arithmetic may be shed light on the impact of asymmetries in the syntactic organization of arithmetic.

6.4. The development of arithmetic abilities

The present study was conducted in participants who had completed high school and hence had substantial mathematical training. Those results are not expected to be universal since, among other things, they rely on arbitrary procedural rules which are extensively trained during mathematical instruction. Our results pave the path for developmental studies aiming to dissect the development of the mathematical ability to perform individual facts and to organize these facts into nested operations. A particular fruitful experimental tag is the possibility to convey syntax by explicit symbols (parentheses) compared to implicit rules (precedence rules) which result in comparable sequences in adults but which might be expected to develop at different moments during mathematical instruction.

6.5. Conclusion

In conclusion, our results show that arithmetic computation, although fast, is organized into a syntactic structure which is partly reminiscent of the organization observed in language, yet preserving some specific idiosyncrasies. The most striking difference is that the syntactic structure is almost immediately evidenced in eye movement patterns. The present study provides a new method to probe syntactic organization in mathematics and, in future studies, explore Hauser et al.'s (2002) conjecture that recursion is a distinctively human trait with broad manifestations in various cognitive domains.

Acknowledgements

This work was funded by the Human Frontiers Science Program. We thank Andrea Moro for reading the manuscript and for stimulating discussions on asymmetries on language and syntax.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.cognition.2012.06.015.

References

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*(4), 1036–1060.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates.

Ashcraft, M. H., & Stazyk, E. H. (1981). Mental addition: A test of three verification models. *Memory and Cognition, 9*(2), 185–196.

Ashcraft, M. H. (1992). Cognitive arithmetic: A review of data and theory. *Cognition, 44*(1–2), 75–106.

Ben-Shachar, M., Hendler, T., Kahn, I., Ben-Bashat, D., & Grodzinsky, Y. (2003). The neural reality of syntactic transformations: Evidence from functional magnetic resonance imaging. *Psychological Science, 14*(5), 433–440.

Binder, K. S., Pollatsek, A., & Rayner, K. (1999). Extraction of information to the left of the fixated word in reading. *Journal of Experimental Psychology: Human Perception and Performance, 25*(4), 1162–1172.

Cajori, F. (1929). *A history of mathematical notations*. The Open Court Publishing Company.

Chomsky, N. (1957). *Sytactic structures*. Mouton.

Chomsky, N. (1975). *Reflections on language*. New York: Pantheon Books.

Chomsky, N. (1988). *Language and problems of language*. Cambridge: MIT Press.

Chomsky, N. (1995). *The minimalist program*. Cambridge, Massachusetts: MIT Press.

Dehaene, S., & Cohen, L. (2007). Cultural recycling of cortical maps. *Neuron, 56*(2), 384–398.

Ernest, P. (1987). A model of the cognitive meaning of mathematical expressions. *British Journal of Educational Psychology, 57*(3), 343–370.

Fodor, J. A. (1975). *The language of thought*. New York: Crowell.

Frazier, L., Carminati, M. N., Cook, A. E., Majewski, H., & Rayner, K. (2006). Semantic evaluation of syntactic structure: Evidence from eye movements. *Cognition, 99*(2), B53–B62.

Gilbert, C. D., Sigman, M., & Crist, R. E. (2001). The neural basis of perceptual learning. *Neuron, 31*(5), 681–697.

Goldstone, R. L., & Barsalou, L. W. (1998). Reuniting perception and conception. *Cognition, 65*(2–3), 231–262.

Goldstone, R. L., Landy, D. H., & Son, J. Y. (2010). The education of perception. *Topics in Cognitive Science, 2*(2), 265–284.

Grodzinsky, Y. (2000). The neurology of syntax: Language use without Broca's area. *Behavioral and Brain Sciences, 23*(1), 1–21 (discussion 21–71).

Grodzinsky, Y., & Finkel, L. (1998). The neurology of empty categories aphasics' failure to detect ungrammaticality. *Journal of Cognitive Neuroscience, 10*(2), 281–292.

Grodzinsky, Y., & Friederici, A. D. (2006). Neuroimaging of syntax and syntactic processing. *Current Opinion in Neurobiology, 16*(2), 240–246.

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

Jansen, A. R., Marriott, K., & Yelland, G. W. (2003). Comprehension of algebraic expressions by experienced users of mathematics. *Quarterly Journal of Experimental Psychology: A, Human Experimental Psychology, 56*(1), 3–30.

Kirshner, D. (1989). The visual syntax of algebra. *Journal for Research in Mathematics Education, 20*(3), 274–287.

Kliegl, R. (2007). Toward a perceptual-span theory of distributed processing in reading: A reply to Rayner, Pollatsek, Drieghe, Slattery, and Reichle (2007). *Journal of Experimental Psychology: General, 136*(3), 530–537.

Kliegl, R., Nuthmann, A., & Engbert, R. (2006). Tracking the mind during reading: The influence of past, present, and future words on fixation durations. *Journal of Experimental Psychology: General, 135*(1), 12–35.

Kofka, K. (1935). *Principles of Gestalt psychology*. New York: Harcourt, Brace.

Landy, D., & Goldstone, R. L. (2007a). Formal notations are diagrams: Evidence from a production task. *Memory and Cognition, 35*(8), 2033–2040.

Landy, D., & Goldstone, R. L. (2007b). How abstract is symbolic thought? *Journal of Experimental Psychology: Learning, Memory, and Cognition, 33*(4), 720–733.

Landy, D., & Goldstone, R. L. (2010). Proximity and precedence in arithmetic. *Quarterly Journal of Experimental Psychology (Hove), 63*(10), 1953–1968.

Landy, D. H., Jones, M. N., & Goldstone, R. L. (2008). How the appearance of an operator affects its formal precedence. In *Proceedings of the thirtieth annual conference of the cognitive science society*, Washington, DC.

Lerdahl, F., & Jackendoff, R. (1996). *A generative theory of tonal music*. The MIT Press.

Mehler, J., Nespor, M., Shukla, M., & Pena, M. (2006). Why is language unique to humans? *Novartis Foundation Symposium, 270*, 251–280 (discussion 280–292).

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: Part 1. Basic mechanisms. *Psychological Review, 104*(1), 3–65.

Miller, G. A., & Chomsky, N. (1963). *Finitary models of language users. Handbook of mathematical psychology* (Vol. 2, pp. 419). Wiley.

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K. M., Malave, V. L., Mason, R. A., et al. (2008). Predicting human brain activity associated with the meanings of nouns. *Science, 320*(5880), 1191–1195.

Moro, A. (2000). *Dynamic antisymmetry*. Cambridge, Massachusetts: The MIT Press.

Nespor, M., & Vogel, I. (2007). *Prosodic phonology*. Mouton de Gruyter.

Nowak, M. A., Plotkin, J. B., & Jansen, V. A. (2000). The evolution of syntactic communication. *Nature, 404*(6777), 495–498.

Patel, A. D. (2003). Language, music, syntax and the brain. *Nature Neuroscience, 6*(7), 674–681.

Posner, M. I., Walker, J. A., Friedrich, F. J., & Rafal, R. D. (1984). Effects of parietal injury on covert orienting of attention. *Journal of Neuroscience, 4*(7), 1863–1874.

Ranney, M. (1987). The role of structural context in perception: Syntax in the recognition of algebraic expressions. *Memory and Cognition, 15*(1), 29–41.

Rayner, K. (1983). *Eye movements in reading: Perceptual and language*. Academic Press.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin, 124*(3), 372–422.

Schroyens, W., Vitu, F., Brysbaert, M., & D'Ydewalle, G. (1999). Eye movement control during reading: Foveal load and parafoveal processing. *The Quarterly Journal of Experimental Psychology, 52*(4), 1021–1046.

Sigman, M., Cecchi, G. A., Gilbert, C. D., & Magnasco, M. O. (2001). On a common circle: Natural scenes and Gestalt rules. *Proceedings of the National Academy of Sciences of the United States of America, 98*(4), 1935–1940.

Sigman, M., & Dehaene, S. (2005). Parsing a cognitive task: A characterization of the mind's bottleneck. *PLoS Biology, 3*(2), e37.

Sigman, M., & Gilbert, C. D. (2000). Learning to find a shape. *Nature Neuroscience, 3*(3), 264–269.

Timmers, L., & Claeys, W. (1990). The generality of mental addition models: Simple and complex addition in a decision–production task. *Memory and Cognition, 18*(3), 310–320.

Turing, A. M. (1938). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society, 2*(42), 230–265.

Waters, G., Caplan, D., & Hildebrandt, N. (1987). *Working memory and written sentence comprehension. Attention and performance 12: The psychology of reading* (pp. 531–555). N. Hillsdale, England: Lawrence Erlbaum Associates, IncColtheart, Max Ed.

Waters, G., Caplan, D., & Hildebrandt, N. (1991). On the structure of verbal short-term memory and its functional role in sentence comprehension: Evidence from neuropsychology. *Cognitive Neuropsychology, 8*(2), 81–126.

Wertheimer, M. (1938). *Laws of organization in perceptual forms*. Harcourt, Brace & Jovanovitch, London.

Westheimer, G. (1999). Gestalt theory reconfigured: Max Wertheimer's anticipation of recent developments in visual neuroscience. *Perception, 28*(1), 5–15.

Widaman, K. F., Geary, D. C., Cormier, P., & Little, T. D. (1989). A componential model for mental addition. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*(5), 898–919.

Zylberberg, A., Fernandez Slezak, D., Roelfsema, P. R., Dehaene, S., & Sigman, M. (2010). The brain's router: A cortical network model of serial processing in the primate brain. *PLoS Computational Biology, 6*(4), e1000765.