Automata and Complexity in Multiple-Quantifier Sentence Verification

Jakub Szymanik (J.K.Szymanik@uva.nl)
Institute for Logic, Language and Computation, University of Amsterdam

Shane Steinert-Threlkeld (Shanest@stanford.edu)
Department of Philosophy, Stanford University

Marcin Zajenkowski (Zajenkowski@psych.uw.edu.pl)
Faculty of Psychology, University of Warsaw

Thomas F. Icard III (Icard@stanford.edu)
Department of Philosophy, Stanford University

Abstract

We study possible algorithmic models for the picture verification task with double-quantified sentences of the form 'Some X are connected with every Y'. We show that the ordering of quantifiers, either Some ∘ Every or Every ∘ Some, influences the cognitive difficulty of the task. We discuss how computational modeling can account for the varying cognitive load in quantifier verification.

Keywords: generalized quantifiers, automata theory, working memory, sentence-picture verification, logic

Introduction

A central area of cognitive science is the study of our linguistic abilities, including the understanding and evaluation of natural language sentences. Given the richness and the variety of natural language constructions it is almost an impossible task to model those cognitive abilities in their full generality. However, there are some fragments of language which have formal semantics and which are well understood and rigorously described by linguists. Those fragments are good candidates for cognitive computational modeling building upon the formal results. In particular, linguistic expressions of quantities deserve special attention. This is because the study of such expressions (determiner phrases) in the framework of Generalized Quantifier Theory (GQT) marks one of the most well-developed branches of formal semantics. Recently, it has been shown how GQT can give rise to a computational model delivering neuropsychological predictions on verification tasks (see, e.g, McMillan et al., 2005; Szymanik and Zajenkowski, 2010a). However, the model could only account for sentences with a single quantifier, like 'More than 5 boys played the game'. In this paper we discuss an extension of the model that covers sentences with multiple-quantifiers, like 'Some boy kissed every girl'. We also test empirically some predictions of the model about the cognitive complexity of various sentences with embedded quantifiers.

Generalized Quantifiers

Generalized quantifiers (GQs) are one of the basic tools of today's linguistics; their mathematical properties have been extensively studied since the 1950s (see, e.g., Peters and Westerståhl, 2006). GQT assigns meanings to statements by defining the semantics of the quantifiers in terms of relations between subsets of the universe. Let us consider sentence (1) as an example:

(1) Every poet has low self-esteem.

GQT takes 'every' as a binary relation between (in this case) the set of poets and the set of people having low self-esteem. Following the natural linguistic intuition we will say that sentence (1) is true if and only if the set of poets is included in the set of people having low self-esteem. Hence, the quantifier 'every' corresponds in this sense to the inclusion relation.

Mathematically, such notion of GQs may be captured by identifying sentences of the form $\mathsf{Q}AB$ with the situations (models) in which those sentences are true (Lindström, 1966). For instance, we want to uniformly express the meaning of 'most' independently from the situation. Let us explain this approach further by giving a few examples. Sentence (1) is of the form Every A *is* B, where A stands for poets and B for people having low self-esteem. As we explained above the sentence is true if and only if $A \subseteq B$. Therefore, the quantifier 'every' corresponds to the class of models (M, A, B) in which $A \subseteq B$. For the same reasons the quantifier 'some' corresponds to the class of models in which the set $A \cap B$ is nonempty. Finally, let us consider the quantifier 'most'. The sentence Most *As are B* is true if and only if the cardinality of set $(A \cap B)$ is greater than the cardinality of set $(A - B)$. Therefore, formally speaking:

$$\mathsf{Every} = \{(M, A, B) \mid A, B \subseteq M \text{ and } A \subseteq B\}.$$
$$\mathsf{Some} = \{(M, A, B) \mid A, B \subseteq M \text{ and } |(A \cap B)| > 0\}.$$
$$\mathsf{Most} = \{(M, A, B) \mid A, B \subseteq M \text{ and } |(A \cap B)| > |(A - B)|\}.$$

Hence, if we fix a model $\mathbb{M}$, then we can treat a generalized quantifier as a relation between relations over the universe, and this is the familiar notion from natural language semantics. For instance, in a

given model $\mathbb{M}$ the statement $\mathsf{Most}_M(A, B)$ says that $|(A^M \cap B^M)| > |(A^M - B^M)|$, where $A^M, B^M \subseteq M$.

The above formalization links tightly with many computational models and results on GQs (see, e.g., Szymanik, 2009). Even though it has not been designed as a cognitive model, it can raise precise processing predictions (see Szymanik, 2007). In the reminder of the paper we study the computational model of the verification tasks for sentences containing combinations of 'every' and 'some' quantifiers.

Model

In the semantic automata approach to GQs, quantifiers are modeled as automata of various types. This works by associating each finite model[1] – corresponding, say, to a visual scene – with a string in a formal language and designing an automaton that accepts a string if and only if the model which was translated into that string makes the quantifier sentence true. As an example, consider Every, the model-theoretic interpretation of which is given above. Given a model $\mathbb{M}$, we can write a word in $\{a, b\}^{|\mathbb{M}|}$[2] by enumerating A and writing an a for every element in $A \setminus B$, and a b for every element in $A \cap B$. To verify whether $\mathbb{M} \in$ Every, we must then simply check whether the word consists entirely of bs since $A \subseteq B$ iff $A \setminus B = \emptyset$. An automaton that recognizes strings corresponding to models in Some should accept any string in which at least one b appears. These two automata are pictured below in Fig. 1.

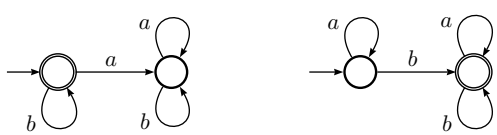

Figure 1: Finite state automata for Every (left) and Some (right).

A detailed theory of automata for GQs of this form has been developed by Van Benthem (1986) (but see also Mostowski (1998)). Certain quantifiers, such as Most can only be modeled by pushdown automata. These automata add to the ones above a stack data structure, a.k.a. a form of memory. A slew of both neuroimaging and behavioral studies (see, e.g., McMillan et al., 2005; Szymanik and Zajenkowski, 2010a,b; Zajenkowski et al., 2011) have shown that this formal distinction is mirrored in the way patients process quantifiers: proportional quantifiers like Most take longer to process, are processed less accurately, and place more demand on working memory than do simpler quantifiers.

Most of the theoretical and experimental work thus far has focused, however, on sentences of the form QAB. But natural language contains the ability to embed quantifiers, as in "some student read every book" or "most students take three classes". These are modeled by *iterated quantifiers*. For instance:

$$\mathsf{Some} \circ \mathsf{Every} = \{(M, A, B, R) \mid (M, A, \{x \in M \mid (M, B, R_x) \in \mathsf{Every}\}) \in \mathsf{Some}\},$$

where $A, B \subseteq M$, $R \subseteq M \times M$, and $R_x = \{y \in M \mid Rxy\}$. Unpacking this definition, we find that "some student read every book" will come out true just when the set of students (x) who are such that every book was read by them (R_x) is non-empty (see Fig. 4 for an example).

Szymanik (2010) has studied the computational complexity of the various readings of multi-quantifier sentences, including iterated quantifiers. He identified the border between tractable and intractable constructions that has been later tested experimentally by Schlotterbeck and Bott (2012). It has turned out that the computational differences approximate cognitive complexity of the corresponding verification task. However, the results of Szymanik for multi-quantifier sentences have been formally disconnected from the automata model for simple quantifier sentences. Recently, that gap has been bridged by Steinert-Threlkeld and Icard (forthcoming) who have extended the semantic automata approach to handle cases of iterated quantification. While we refer the reader to the paper for formal details, we here present one example, a machine for Some ∘ Every (Fig. 2)[3]. Let

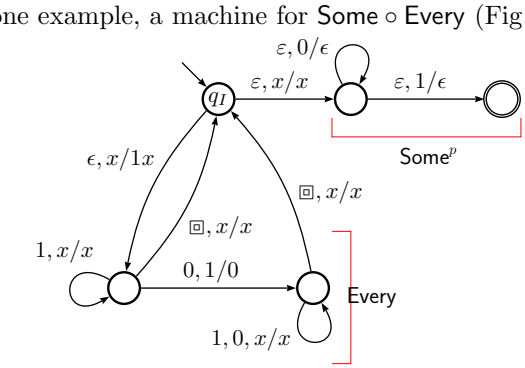

Figure 2: A pushdown automaton for Some ∘ Every.

us assume that we want to verify sentence 'Some boy reads every book'. The idea is this: for every boy we check the set of books he reads (the string generated by B and R_x). We run the Every automaton on it to check

[1] As we work with natural language quantifier the restriction to finite models is arguably innocent (see Szymanik, 2009).

[2] To use a two-letter alphabet, the quantifiers must satisfy certain properties, like topic neutrality, domain independence and conservativity (see, e.g., Peters and Westerståhl, 2006). All the quantifiers considered in this study do.

[3] ⊡ is a special separator symbol needed in the encoding of the models.

whether that particular boy actually reads all the books. If he does we push a 1 onto the stack, otherwise we push a 0. Next, we move to the books read by the next boy (⊡ separates the corresponding substrings in the encoding) and run the same algorithm. Once we analyzed books read by every boy in that way, we then run the Some machine but using the stack contents as the tape. So, this machine will reach the accepting state if and only if there is at least one 1 on the stack, which will happen if and only if at least one $x \in A$ is such that the Every automaton accepts the string generated by B and R_x, meaning that there is a boy who reads all books.

This raises an empirical question: do iterated quantifiers place demands on working memory as the pushdown automata model would predict?

Predictions

We address this question by performing sentence-verification tasks on sentences with Some ∘ Every and Every ∘ Some iterations. First, we must note that it turns out that these iterations can be also modeled by deterministic finite state automata.[4] Fig. 3 depicts the minimal DFAs accepting Every ∘ Some and Some ∘ Every. Nevertheless, we predict that processing iterated quan-

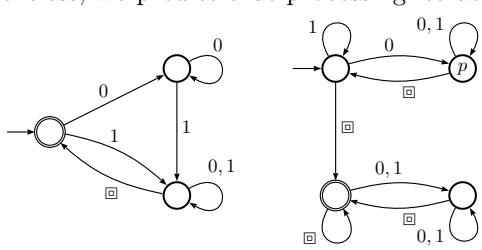

Figure 3: Minimal DFAs accepting the language for Every ∘ Some (left) and Some ∘ Every (right). To compare with PDA let us explain the run of Some ∘ Every automata for the sentence 'Some boy reads every book'. Again, we pick the first boy and look at the books he reads. If he reads all the books we move to accepting state; otherwise, if we find one book he doesn't read we move to state p. From here we can move back to the initial state only by starting to check books read by another boy (marked by ⊡ in the encoding of the model). From the initial state we can go to the double-circled accepting state if and only if we find a boy who reads all books. The bottom row indicates that once we have found one such student, it does not matter whether or not any of the others have read every book.

tifiers will place similar demands on working memory to proportional quantifiers. This is because the construction outlined in the previous section provides a general mechanism for converting automata for any two quantifiers into an automaton for their iteration. No such general mechanism is known to exist for generating DFAs for iteration. To see this, compare the two automata in Fig. 3. Although the top row of the Some ∘ Every machine does contain a copy of the Every DFA, no such copy of either DFA can be found in Fig. 3; more importantly, there doesn't appear to be a uniform construction generating the two iterated DFAs. It is plausible that people learn procedures for assessing basic quantifier meanings and then develop a general mechanism for processing embedded quantifiers. Since iteration of quantifiers is one semantic operation which is independent of the two basic quantifiers, we predict that there is a single corresponding mental mechanism for constructing procedures to process iterated quantifiers from basic procedures. This mechanism generates pushdown automata and so we expect to find strong working memory demands when processing iterated quantifiers.

Furthermore, the model allows us to predict that true instances of Every∘Some might be more complex to verify than true instances of Some ∘ Every. This is because in the first case subjects have to run through every element of A, whereas in the latter, they needed only find one example; this example might be salient in the image. But even if not, a subject verifying Some∘Every can stop processing once he finds one appropriate A and need not continue to the rest. Of course, the model predicts that the situation is opposite for false instances: namely false Every∘Some are simpler to verify than false Some∘Every since the former require just finding one counterexample.

Experimental results

Method

To test the theoretical predictions we studied how people verify sentences of the form 'Every X is connected with some Y' and 'Some X is connected with every Y'. We also compared the performance on these sentences with other cognitive tasks. In particular, we measured memory span and cognitive control. According to the multi-component model of working memory as well as empirical findings, these two cognitive functions reflect central aspects of working memory (see, e.g. Logie, 2011). Additionally, we looked at proportional judgments of the form 'More than half of the dots are yellow'. According to the model, proportional sentences are only computable by PDAs and, therefore, they engage working memory.

Participants Seventy-six Polish-speaking students from University of Warsaw (46 females and 30 males) between 19 and 31 years (mean age was 22.64 years, SD=2.65) were recruited for this experiment. Each subject received a small financial reward for participating.

Materials and procedure

[4] A key result of Steinert-Threlkeld and Icard (forthcoming) is that if Q_1 and Q_2 are computable by deterministic finite state automata, then so too is $Q_1 \circ Q_2$, i.e., a pushdown automata from Fig. 2 can be actually transformed into a finite-state automata.

Iterations The task tested how subjects verify two types of sentences against simple pictures (see Figure 4). Each sentence was repeated eight times. Half of the trials

Figure 4: Examples of stimuli used in the study. Sentence 'Every circle is connected with some square' is true in situation 1. Sentence 'Some circle is connected with every square' is true in situation 2.

were true. At the beginning of each trial a sentence was displayed. Subjects had as much time as they needed to read it. Next, a picture was presented, and participants were asked to decide within 20000 ms if the proposition accurately describes the picture. All stimuli were counterbalanced and randomly distributed throughout the experiment. For every sentence we mesured mean reading time, mean verification time, and accuracy (number of correct answers; maximum=8).

Memory span The memory span task was a computerized version of Sternberg's (1966) short-term memory measure. On each trial of the test, the subjects were presented with a random series of different digits, one at a time, for 300 ms, followed by a blank screen and the test digit. Participants had to decide whether the test digit had appeared in the previously displayed string. Sequences of digits of three lengths (four, six, or eight) were repeated eight times each; hence, there were 24 trials overall. The score was the total of correct responses from all conditions (range 0 to 24).

Cognitive control Cognitive control was measured with the short version of the Attention Networks Test (ANT) designed by Fan et al. (2002).

The authors' starting point was the assumption that the attentional system can be divided into three functionally independent networks: alerting, orienting, and executive control. In the present study we focused on the latter network (the monitoring and resolution of conflict between expectation, stimulus, and response) as an index of cognitive control. In the ANT task, on each trial, the participant has to decide, by pressing a button, whether a central arrow stimulus (the target) points left or right. The target is flanked by distractor stimuli, which may be congruent with the target (arrow points in same direction) or incongruent (arrow points in opposite direction). In each case, two flankers are presented on either side of the target. The control index is calculated by subtracting the RT median of the congruent flanking conditions from the RT median of incongruent flanking conditions.

Proportional judgements This task measured the reaction time and accuracy of proportional judgments, such as 'Less than half of the dots are blue', against color pictures presenting dots. The pictures accompanying sentences differed in terms of the number of objects (15 dots or 17 dots), but not the distance between the cardinalities of two sets of dots (7 vs 8 and 8 vs 9). Within each condition, subjects had to solve eight trials. Half of them were true. Participants were asked to decide, by pressing a button, whether or not the proposition accurately describes the picture. We analyzed mean reaction time (RT) as well as accuracy level (number of correct answers; maximum=8) of each condition.

Results

Iterations First we compared the processing of two types of sentences used in the task. ANOVA with type of sentence (2 levels) and statements truth-value (2 levels) as two within-subject factors was used to examine differences in mean verification times and accuracy (see Table 1 and Table 2 for means and standard deviations). The main effect of sentence type was significant indicating that sentences containing quantifiers ordered as every-some were verified significantly longer ($F(1, 75) = 17.01$, $p < 0.001$, η^2=0.19) and less accurately ($F(1, 75) = 22.48$, $p < 0.001$, η^2=0.23) than sentences with some-every order. Moreover, the analysis revealed the significant main effect of the interaction between sentence type and truth-value in case of verification time ($F(1, 75) = 42.02$, $p < 0.001$, η^2=0.36) as well as accuracy ($F(1, 75) = 25.63$, $p < 0.001$, η^2=0.26). Further comparisons among means indicated that true sentences with every-some were processed longer and worse than all other situations. Both false conditions did not differ from one another, and were medium difficult, while true some-every sentences had shortest mean RT and the highest correctness.

Finally, for reading time we analyzed only difference between sentence types. ANOVA reached the tendency level ($F(1, 75) = 2.85$, $p = 0.095$, η^2=0.04) and indicated that participants needed more time for every-some than some-every constructions.

Correlations Next, we correlated the scores obtained in the iteration verification task with other cognitive measures (see Table 1). Analyzing accuracy, we found that only some-every sentences were highly and positively correlated with scores obtained in the memory task and proportional judgements, while in the case of cognitive control the relationship was negative. The latter result is negative since the high result on control network indicates delay in inhibiting response to competing stimuli, and hence poor executive functioning. Interest-

ingly, similar correlations were obtained between accuracy on proportional judgements and both memory span and control tasks. We also found that the verification times for sentences with two quantifiers are positively associated with the verification times of proportional judgments.

When the correlations are conducted separately for true and false iterated statements, the general pattern of significant correlations remains the same (see Table 2). Specifically, only sentences with some-every order were significantly associated with cognitive control, memory span, and proportional quantifiers. This relationship was independent of truth-value.

Table 1: Means (SD) of all variables and correlations between iteration task and other cognitive measures

	Control	Memory	Prop15 acc	Prop17 acc	Mean (SD)
Every-some read	-.12	-.04	-.14	.02	5019 (2520)
Some-every read	-.05	-.01	-.15	-.07	4738 (2140)
Every-some ver	.11	.11	-.01	.01	2506 (1129)
Some-every ver	.10	-.13	-.02	-.10	2079 (867)
Every-some acc	-.06	.02	.10	-.05	6.48 (1.76)
Some-every acc	-.38**	.29*	.32**	.45**	7.61 (.92)
Control		-.26*	-.33**	-.29*	95.68 (38.22)
Memory			.25*	.30**	20.89 (2.15)
Prop15 acc				.49**	6.86 (1.22)
Prop17 acc					6.88 (1.24)

* $p < 0.05$; ** $p < 0.01$; *Note* Read - reading time; ver - verification time; acc - accuracy; prop15 - proportional quantifiers presented with 15 objects, prop17 - proportional quantifiers presented with 17 objects.

Table 2: Means (SD) of iterated sentences in true and false conditions, and their correlations with other cognitive measures

	Control	Memory	Prop15 acc	Prop17 acc	Mean (SD)
Every-some ver false	.12	.05	-.02	.05	2231 (870)
Every-some ver true	.09	.20	-.02	-.06	2781 (1569)
Some-every ver false	.03	-.12	.10	.05	2468 (1315)
Some-every ver true	.16	-.20	-.21	-.19	1690 (752)
Every-some acc false	-.06	.18	.03	.03	3.5 (0.80)
Every-some acc true	-.05	-.09	.05	-.11	2.96 (1.35)
Some-every acc false	-.30**	.24*	.23*	.41**	3.72 (0.62)
Some-every acc true	-.38**	.28*	.31**	.36**	3.90 (0.42)

Discussion

We have studied the computational model of verifying sentences containing embedded quantifiers. We confirmed the prediction that for true instances Every∘Some is harder than Some∘Every but we did not find the opposite relation for false instances. Most importantly, while the model suggests that sentences with Some∘Every and Every ∘ Some iterations are equally difficult with respect to working memory engagement, we found some differences in subjects' performance: 'Every-some' sentences are more difficult in terms of reaction time and accuracy. On the other hand, only verification of 'some-every' sentences correlates with other tasks engaging working memory resources, like cognitive control and memory span, as well as with accuracy of proportional judgments. Moreover, the latter are also associated with both working memory aspects. These findings point towards an alternative model under which Some ∘ Every gets associated with a canonical push-down automata from Fig. 2 and Every ∘ Some iterations are processed with a strategy resembling a finite-state automaton from Fig. 3. That could explain, on the one hand, the qualitatively different engagement of working memory in the verification of 'Some X is connected with every Y', and on the other hand, the longer reaction time and higher error-rate in the judgments of 'Every X is connected with some Y'. The idea here would be that even though the push-down automata strategy engages more cognitive resources, it is more effective than the corresponding finite-state automata. A related empirical finding is that the reading time (comprehension) for 'every-some' sentences is longer than for 'some-every' sentences. Therefore, an alternative model should also predict that deriving the push-down automata verification strategy for Some ∘ Every iteration is easier than constructing the finite-state automata strategy for Every∘Some iteration. This seems to be a natural direction for future research.

Outlook

We think that one of the best strategies for subsequent research would be to embed the formal theory in a proper computational cognitive model or implement it within some cognitive architecture, like ACT-R. The general aim of the project would be to build a psychologically and neurally plausible theory of quantifier meaning and compare it with other proposals, such as Johnson-Laird's mental models (1983) or Clark's Comparison Theory (1976). There are many questions about the correspondence between the formal models of quantifier verification and the cognitive resources the subjects need to use in order to solve the task. Building a computational cognitive model will lead to new experimental predictions that can be consequently tested. Moreover, none of the empirical research so far has looked into actual strategies the subjects are applying in order to verify quan-

tifier sentences. Eye-tracking studies could fill the gap and provide additional data to assess whether models of quantifier verification postulate psychologically plausible strategies. We hope that such tasks could be successfully carried out in a collaboration between cognitive modelers and logicians studying GQT.

General Conclusions

The paper describes an abstract and purely quantitative model of quantifier verification motivated by logical investigations in the semantics of natural language. From a cognitive computational perspective, this is a sort of conceptual pre-modeling, mathematically delimiting the class of all possible cognitive strategies that could be further implemented in a proper cognitive computational model, giving raise to qualitative predictions. In other words, our approach is to analyze human cognitive behavior by investigating formal computational properties of the task (cf. Marr, 1983; Anderson, 1990). One could say that our work is positioned between the computational and algorithmic levels of Marr at level 1.5 explanation (see, e.g., Steinert-Threlkeld and Icard, 2013; Isaac et al., 2013) We do not specify the actual verification strategies people use but we do more than only formal computational characterization of the task, namely we delimit a class of 'reasonable' strategies. Our toolbox in doing that is modern logic and computation theory which focuses on processes rather than 'logical correctness'. One natural application of this toolbox – that we have explored in the paper – is in estimating cognitive difficulty of a task. We believe that the formal insights logic and computation theory have to offer are instrumental for building plausible cognitive computational models.

Acknowledgments

JS acknowledges NWO Veni Grant 639.021.232. The work of MZ was supported by a grant no. 2011/01/D/HS6/01920 funded by the National Science Centre in Poland.

References

Anderson, J. (1990). *The Adaptive Character of Thought*. Studies in Cognition. Lawrence Erlbaum.

van Benthem, J. (1986). *Essays in Logical Semantics*. D. Reidel, Dordrecht.

Clark, H. (1976). *Semantics and Comprehension*. Mouton.

Fan, J., McCandliss, B. D., Sommer, T., Raz, A., and Posner, M. I. (2002). Testing the Efficiency and Independence of Attentional Networks. *Journal of Cognitive Neuroscience*, 14(3):340–347.

Isaac, A., Szymanik, J., and Verbrugge, R. (2013). Logic and complexity in cognitive science. In Baltag, A. and Smets, S., editors, *Johan van Benthem on Logical and Informational Dynamics*. Springer.

Johnson-Laird, P. N. (1983). *Mental Models: Toward a Cognitive Science of Language, Inference and Consciousness*. Harvard University Press.

Lindström, P. (1966). First order predicate logic with generalized quantifiers. *Theoria*, 32:186–195.

Logie, R. (2011). The functional organisation and the capacity limits of working memory. *Current Directions in Psychological Science*, 20:240–245.

Marr, D. (1983). *Vision: A Computational Investigation into the Human Representation and Processing Visual Information*. W.H. Freeman, San Francisco.

McMillan, C. T., Clark, R., Moore, P., Devita, C., and Grossman, M. (2005). Neural basis for generalized quantifier comprehension. *Neuropsychologia*, 43:1729–1737.

Mostowski, M. (1998). Computational semantics for monadic quantifiers. *Journal of Applied Non-Classical Logics*, 8:107–121.

Peters, S. and Westerståhl, D. (2006). *Quantifiers in Language and Logic*. Clarendon Press, Oxford.

Schlotterbeck, F. and Bott, O. (2012). Easy solutions for a hard problem? The computational complexity of reciprocals with quantificational antecedents. In Szymanik, J. and Verbrugge, R., editors, *Proceedings of the Logic and Cognition Workshop at ESSLLI 2012*, pages 60–72. CEUR Workshop Proceedings.

Steinert-Threlkeld, S. and Icard, T. (2013). Iterating semantic automata. *Linguistics and Philosophy*.

Sternberg, S. (1966). High-speed scanning in human memory. *Science*, 153:652–654.

Szymanik, J. (2007). A comment on a neuroimaging study of natural language quantifier comprehension. *Neuropsychologia*, 45(9):2158–2160.

Szymanik, J. (2009). *Quantifiers in TIME and SPACE. Computational Complexity of Generalized Quantifiers in Natural Language*. PhD thesis, University of Amsterdam, Amsterdam.

Szymanik, J. (2010). Computational complexity of polyadic lifts of generalized quantifiers in natural language. *Linguistics and Philosophy*, 33:215–250.

Szymanik, J. and Zajenkowski, M. (2010a). Comprehension of simple quantifiers. Empirical evaluation of a computational model. *Cognitive Science: A Multidisciplinary Journal*, 34(3):521–532.

Szymanik, J. and Zajenkowski, M. (2010b). Quantifiers and working memory. In Aloni, M. and Schulz, K., editors, *Amsterdam Colloquium 2009, Lecture Notes In Artificial Intelligence 6042*, pages 456–464. Springer.

Zajenkowski, M., Styła, R., and Szymanik, J. (2011). A computational approach to quantifiers as an explanation for some language impairments in schizophrenia. *Journal of Communication Disorders*, 44(6):595 – 600.