Hierarchy and Recursion in the Brain.

Cristiano Chesi & Andrea Moro
Ne.T.S. - IUSS Center for Neurolinguistics and Theoretical Syntax, Pavia

1. Syntax in the brain.

Neuroimaging techniques has offered interesting opportunities to deepen our understanding of the relationship between syntax and the brain (Cappa 2012). Two issues appear to be well-established: first, syntactic computation activates a dedicated network (Embick et al. 2000, Moro et al. 2001); second, the format of rules cannot be traced to arbitrary, cultural or conventional facts but it reflects the neuropsychological architecture of the brain circuitry (Tettamanti et al. 2002, Musso et al. 2003, Tettamanti et al. 2008). In this paper we address a specific issue that raises from these studies on a computational perspective: the core result of the last three experiments mentioned is that the theoretical distinction between grammatical vs. non-grammatical rules is reflected in the brain activity. More specifically, the activity of (a deep component of) Broca's area within a more complex network including subcortical elements such as the left nucleus caudatus appears to be sensitive to this distinction (structure-dependent vs. position-dependent) as the BOLD signal is increased in this area only when the subjects increase their performance in manipulating grammatical (i.e. structure-dependent) rules. Here we want to discuss how this result relate to the nature of recursion and hierarchy in linguistic processing (Chomsky 1995, Berwick & Chomsky 2001).

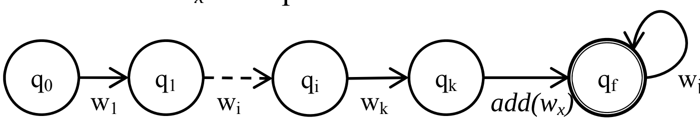
2. Disentangling hierarchy from recursion: a computational complexity perspective

Although there are subtle discrepancies looking at reaction times, there is surely complete convergence with respect to performance: all subjects rapidly acquire the same capability to manipulate both grammatical (e.g. passive construction, Musso et al. 2003) vs. non-grammatical rules (article, Tettamanti et al. 2002, or negation, Musso et al. 2003, placement in a fixed-position; question formation by complete word-sequence inversion, Musso et al. 2003). This fact already constitutes a puzzle, since the broad distinction between hierarchical (grammatical) vs. non-hierarchical (non-grammatical, e.g. sequential) rules correspond to a different degree of complexity: assuming that each rule can be expressed as a set of (computational) states traversals, being the number of states to be explored somehow proportional to the memory required to perform a certain computation, hierarchical rules are less memory demanding than sequential rules, since in the vast majority of contexts, hierarchical rules can deal with lexical clusters rather than single items, then operating only on the relevant chunk(s) level. If the hierarchical rules are also recursive (e.g. $X \rightarrow aXb$) the very same state can be re-used more times, inducing extra memory saving. Similar considerations on complexity also extend to non-hierarchical, non-recursive rules, that, in this sense are more "expensive". To explain this we must preliminarily define, from a computational perspective, the typology of (non-)recursive/(non-)hierarchical rules. Here we assume that the rules/computations are subsumed by different automata.

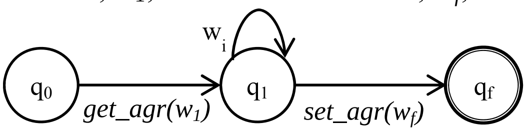
3. Ranking complexities

The (computational) complexity of a task is measured in terms of resources (memory and time) used by a computation while attempting to complete that task. This definition of complexity requires a precise formalization of the computation in order to understand the amount of resources used by the task we want to analyze. Assuming that the rules are computed by a simple Push-Down Automata (i. e. a "PDA", a Finite State Automata endowed with a Last In First Out memory buffer), we could characterize the rule typology as follows:

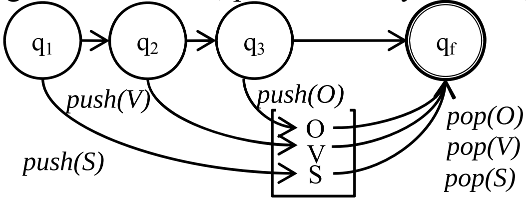
1. rule (1) *(non-recursive, non-hierarchical*):
   insert a word w_x at k^{th} position



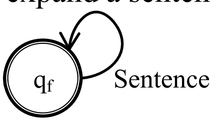
2. rule (2) (*recursive, non-hierarchical*):
   the first, w_1, and the last element, w_f, in the string should agree



3. rule (3) (*non-recursive, hierarchical*)
   given a sentence, passivize it by inverting the subject and the object



4. rule (4) (*recursive, hierarchical*)
   expand a sentence with another sentence by complementation



The prediction is that (4), once a sentence is recognized/expected, is the simplest computation, while (2) is generally simpler (it requires 3 states) than (1) (this requires $k+1$ states). On the other hand, (3) is generally simpler than (1), in terms of state traversal numbers, but since it uses the memory buffer, we need a more articulated complexity cost function: if we assume that adding an extra state has a linear cost and that using an extra slot in the memory buffer has an exponential cost (cf. Gibson 1998), (3) will be harder than (2) and, in most cases, also harder than (1). What is interesting, is that these distinctions do not (yet) correlate in terms of brain activity nor behavioral measures.

4. Complexity, recursion and the brain

The scenario discussed here raises at least two delicate questions that should be put on the agenda for those who study the biological foundations of language, and syntax in particular. The first one amounts to explain how there are no significant behavioral different outcomes in achieving tasks when manipulating recursive vs. non-recursive rules tout court. The second one, on the other hand, raises a deep methodological issue: being able to measure the complexity of all typologies of rules (§3) with simple computational models (PDAs as baseline), this allows us to provide precise and comparable complexity metrics. Since all the sentences we can test are finite, it is logically impossible to test recursion directly: what we should aim at verifying, then, is whether the complexity reduction we expect with recursive and/or hierarchical rules, and the cost of using devices like memory buffers, is proportional to the behavioral/learning data. Since now we have a reliable brain signature of linguistic rules usage, we think we are ready to deepen our understanding of hierarchy and recursion in a rather new way, reconciling grammar with processing models (Sprouse et al. 2012).

Selected references

Berwick & Chomsky (2011) *Biolinguistic investigations:*19-41. **Cappa** (2012) *Neuroimage* 61(2):427-31. **Chesi** (2012) Unipress. **Chomsky** (1995) The Minimalist Program. **Embick et al.** (2000) *PNAS* 97(11):6150-6154. **Gibson** (1998) *Cognition* 68:1–76. **Moro et al.** (2001) *NeuroImage* 13:110-118. **Musso, et al.** (2003) *Nature neuroscience* 6:774-781. **Sprouse et al.** (2012) Language 88(2):401-407. **Tettamanti et al.** (2002) *NeuroImage* 17:700-709. **Tettamanti, et al.** (2008) *Cortex*, 45(7):825-38.