

Complexity

Complexity of phonology is a term with many meanings:

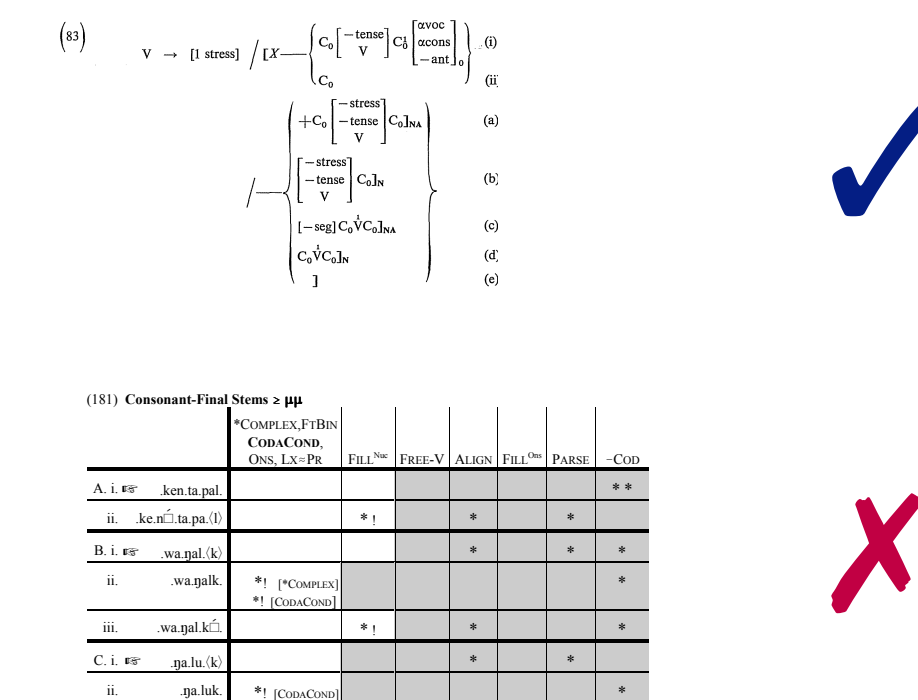
- how big is the inventory?
- how many rules/constraints are there?
- how long is the biggest rule/constraint?
- how long might it take to apply rules/constraints?
- how long might it take to learn rules/constraints?

Here we're using **computational complexity** of **evaluating rules/constraints**.

SPE and OT

Under slight restrictions, an SPE grammar is a finite-state transducer, so produces surface form in **linear** time ($t \propto n$) [5]. Even with weaker assumptions, we can see it's $\propto n^2$.

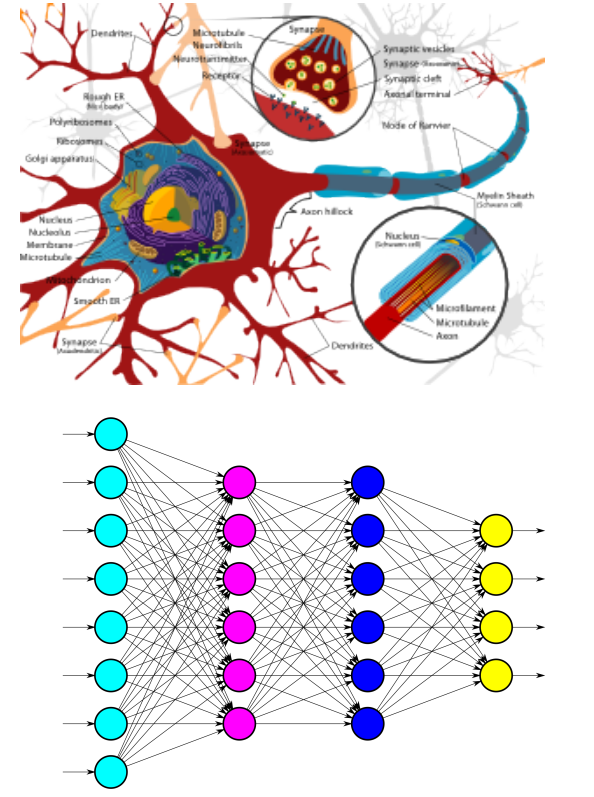
Analysing OT is awkward (richness of the base!). Simplified OT was early shown to be **NP-hard** (i.e. probably intractable) when constraints are part of problem [1].



Complexity of the brain

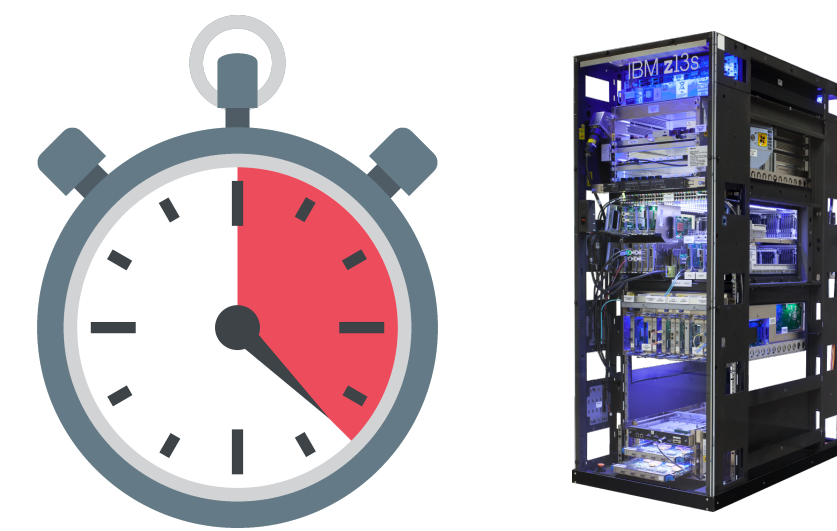
Our understanding of low-level neuronal connections is quite good, and that's about all. Some evidence that phonemes, even features, locate in mm-sized regions (ca. 10^4 neurons) [6].

Neural nets are a fashionable computer model used for machine learning (mis-called AI). They are far simpler than the brain – but we don't even have good ways of analysing their complexity. So what can we do? Give up?



Computational Complexity ...

... in computer science, is how the time (or other resource) taken to compute the answer to a problem depends on the size of the problem.

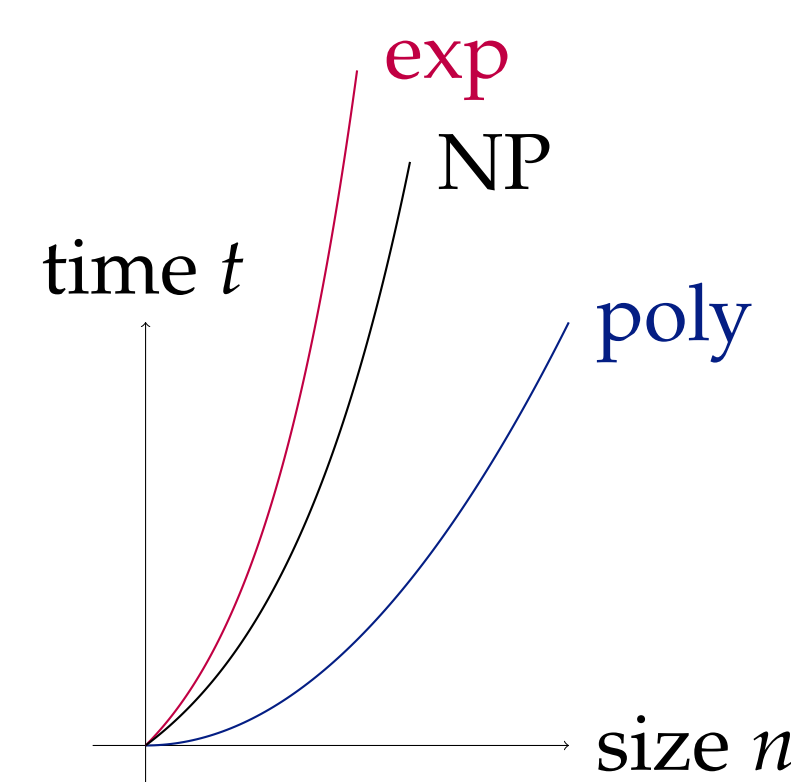


Polynomial or not

Generally, if time is polynomial (P) in problem size (e.g. $t \propto n^2$), we think of the problem as **tractable**.

If it grows faster (e.g. exponentially) we consider it **intractable**.

Many real problems are 'in-between' in **NP**, which we think is intractable, but we don't know.



Is OT really that bad?

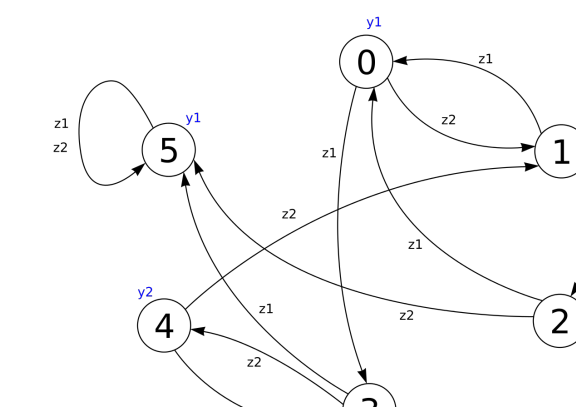
[1] shows OT is NP-hard (at least as bad as NP). But is it only NP, or is it worse? We can't (yet) show that OT is better than **ExpTime** – very bad!



[4] says this is wrong because:

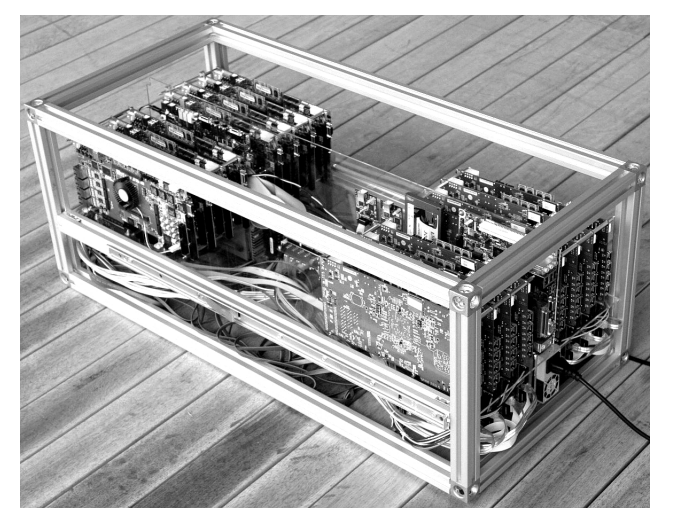
- constraint set should be treated as fixed, not part of the problem size – evaluation is linear in input string length [2]
- OT grammars don't have to be lists of constraints – there are more efficient representations as automata

$$t \propto \exp(|\text{constraints}|) \times |\text{input}|$$



Hardware models of the brain

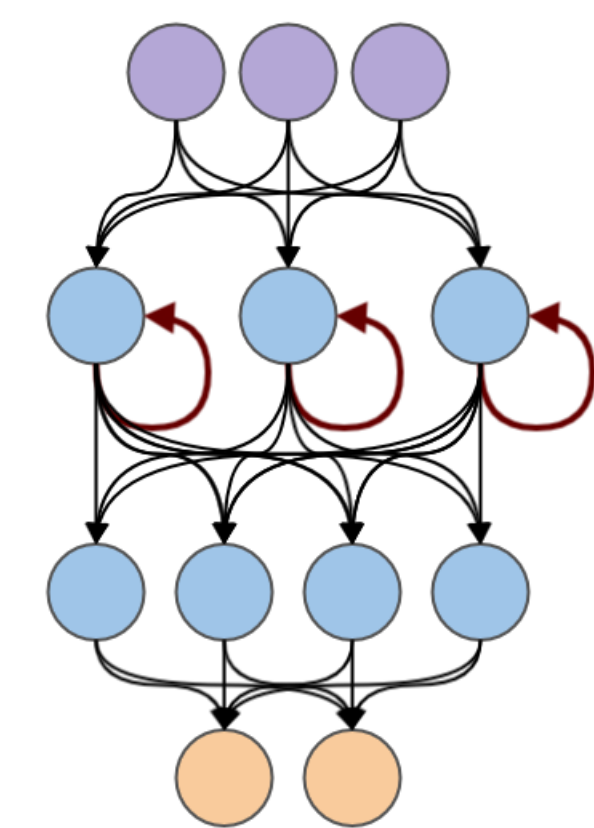
[3] designed an FPGA-based simplified model with high (1000) connectivity and semi-realistic inter-neuron communication. The base model has 256k 'neurons', enough for playing with small phonology-sized systems.



OT is parallel

Tentative model in [3]-style model:

- input strings as chains of neuron ensembles forming a buffer (cf. short-term memory)
 - feature-computing ensembles fed from input, and feeding in parallel to
 - constraint ensembles also fed from input, and activating
 - output buffer ensembles
- Complexity is irrelevant! Need to implement in simulation, and then get the hardware ...



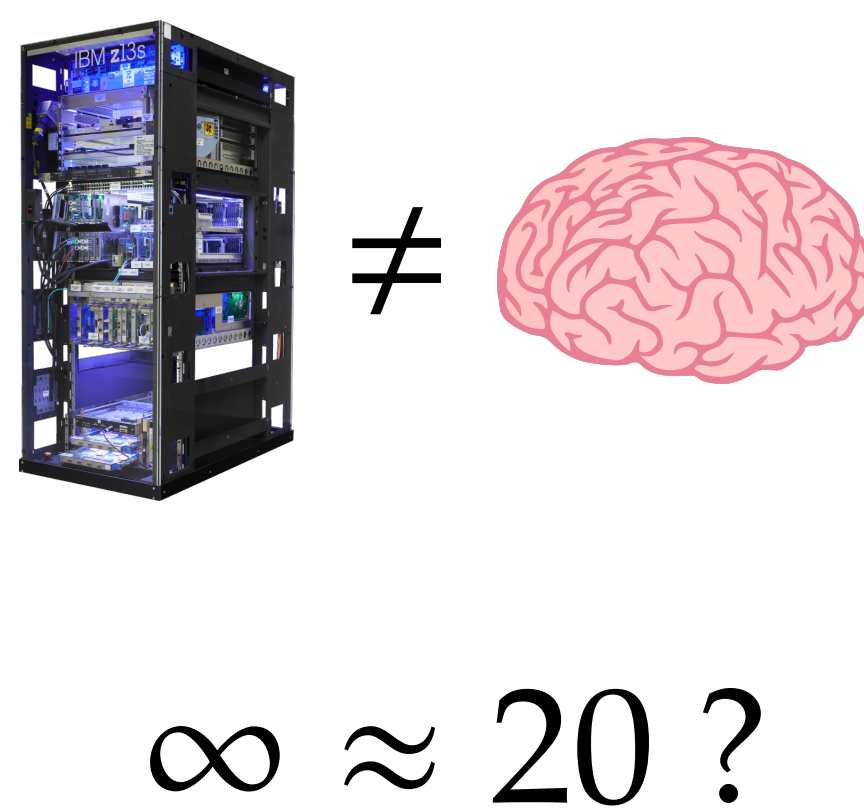
... is it useful?

Usually, complexity means the number of steps on a simple single-processor, unlimited memory, model of computation.

The brain is not such – it is a massively parallel highly connected collection of neurons.

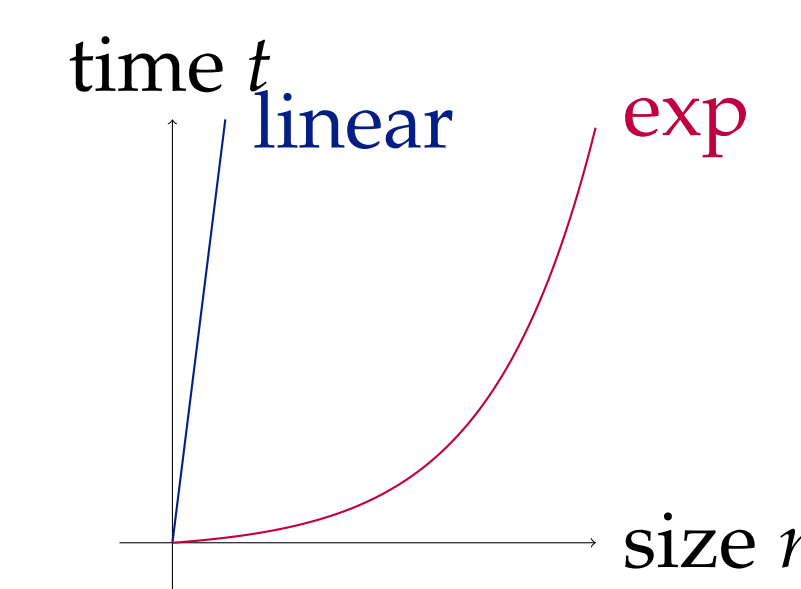
A logic gate may connect to 2 or 3 other gates; a neuron may connect to 10000 other neurons!

Anyway, complexity is usually only interesting **asymptotically** as n goes to infinity. In phonology, who cares about infinity?

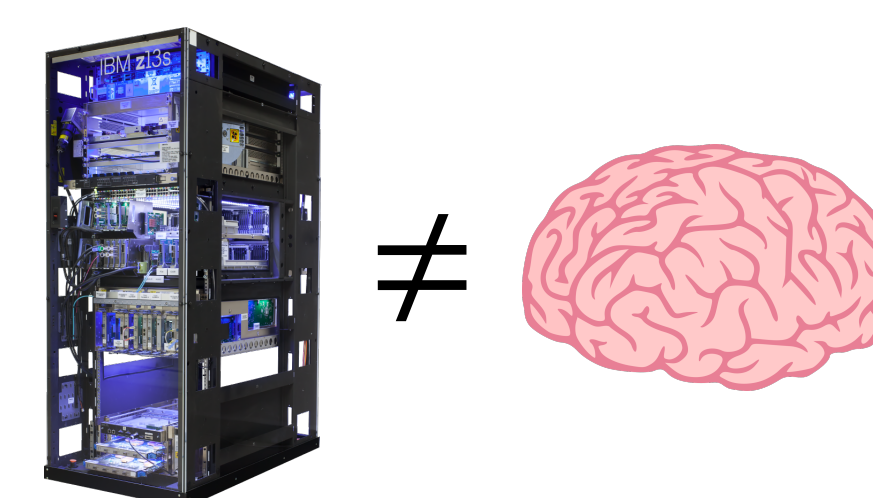


Constants do matter

[4] relies on 'constants don't matter': in complexity theory, $t = 10^{10}n$ is tractable. Sorry: they do. The automata are exponentially bigger than the constraint lists. And $\infty \approx 20$



But ...



References

- [1] Jason Eisner. Efficient generation in primitive Optimality Theory. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 313–320, Madrid, July 1997. URL: <http://cs.jhu.edu/~jason/papers/#acl97>.
- [2] T. Mark Ellison. Phonological derivation in optimality theory. In *15th International Conference on Computational Linguistics, COLING 1994, Kyoto, Japan, August 5-9, 1994*, pages 1007–1013, 1994. URL: <https://aclanthology.org/C94-2163/>.
- [3] Paul J. Fox. Massively parallel neural computation. Technical report UCAM-CL-TR-830, University of Cambridge, Computer Laboratory, March 2013. DOI: 10.48456/tr-830. URL: <https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-830.pdf>.
- [4] Jeffrey Heinz, Gregory Koble, and Jason Riggle. Evaluating the complexity of Optimality Theory. *Linguistic Inquiry*, 40(2):277–288, 2009.
- [5] Ronald M. Kaplan and Martin Kay. Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378, 1994. URL: <https://aclanthology.org/394-3001>.
- [6] Nima Mesgarani, Connie Cheung, Keith Johnson, and Edward F Chang. Phonetic feature encoding in human superior temporal gyrus. *Science*, 343(6174):1006–1010, 2014. DOI: 10.1126/science.1245994.