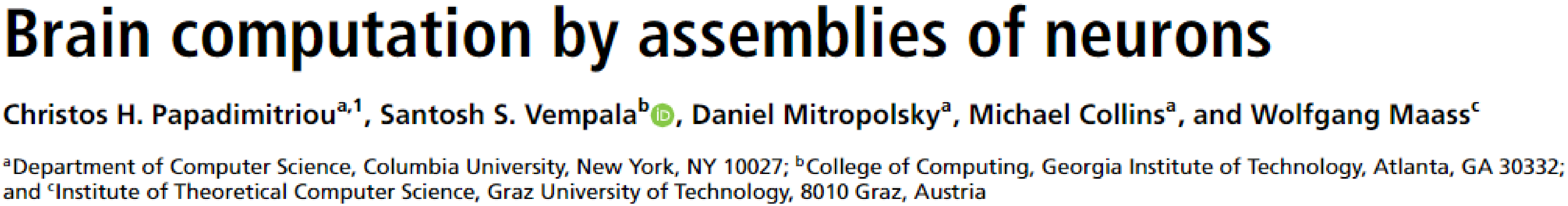


Brain computation by assemblies of neurons

Christos H. Papadimitriou[a,1], Santosh S. Vempala[b], Daniel Mitropolsky[a], Michael Collins[a], and Wolfgang Maass[c]

[a]Department of Computer Science, Columbia University, New York, NY 10027; [b]College of Computing, Georgia Institute of Technology, Atlanta, GA 30332; and [c]Institute of Theoretical Computer Science, Graz University of Technology, 8010 Graz, Austria

Victoria Zhang

2022.5

Motivation

“We do not have a logic for the transformation of neural activity into thought and action. “

Richard Axel

What kind of formal system, abstracting the realities of neural activity, would qualify as the sought “logic”?

Motivation

Assembly:

Large, density interconnected populations of excitatory neurons in a brain area, whose loosely synchronized firing in a pattern is coterminous with the subject thinking of a particular concept or idea.

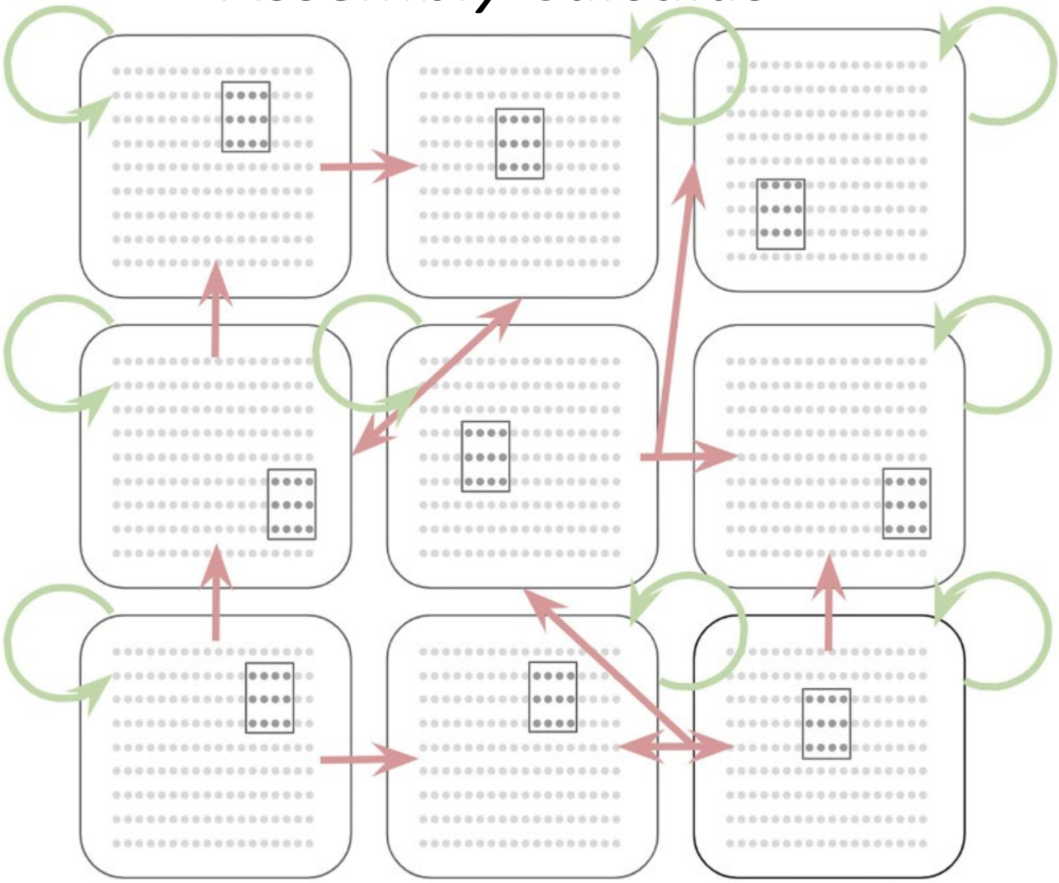
Donald O. Hebb 1949

What kind of formal system, abstracting the realities of neural activity, would qualify as the sought "logic"?

A simple model of the cortex

- Finite number of brain areas
- Each contains n excitatory neurons
- Inhibition: only $k < n$ fire
- Some pairs of areas have sparse random connectivity ⟶
- All have recurrent random connectivity with probability $= p$ ↻

Assembly Calculus



A simple model of the cortex (cont.)

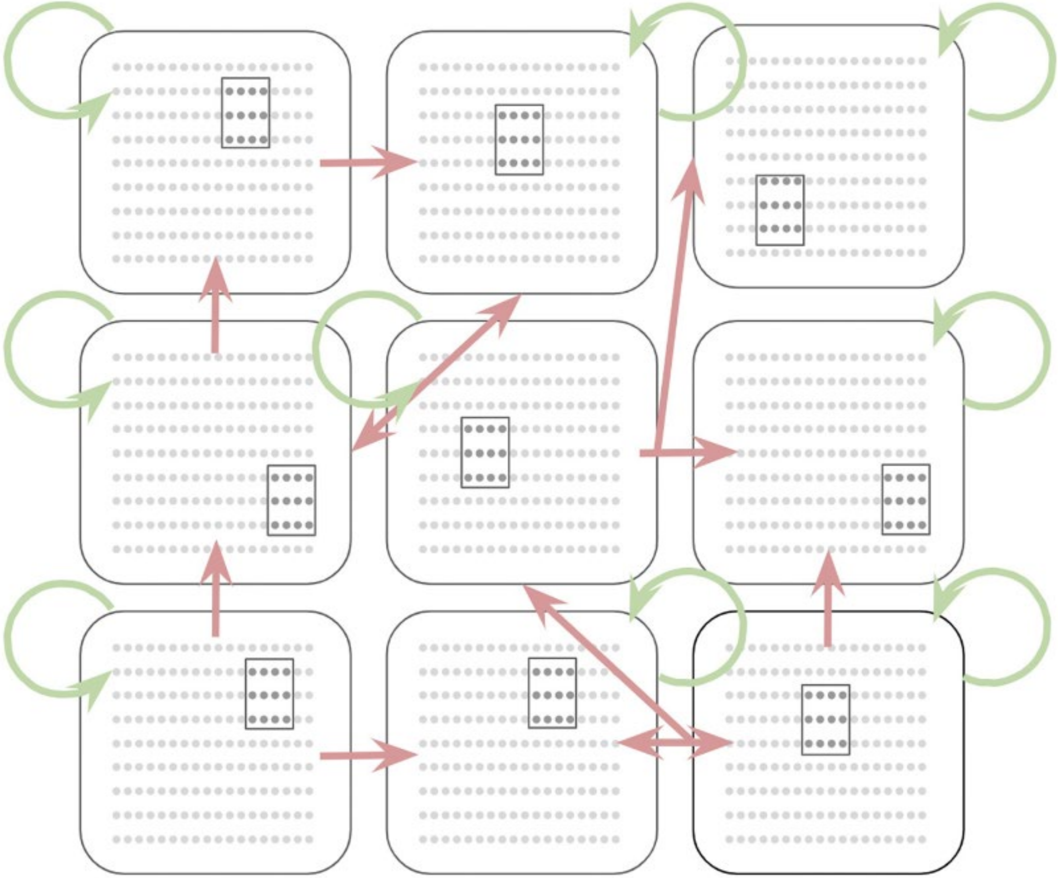
- Assume neurons fire in discrete steps
- At each step, $k < n$ neurons fire
- Areas can be inhibited/disinhibited
- Hebbian plasticity:

  If $i \to j$

  i fires;

  next j fires;

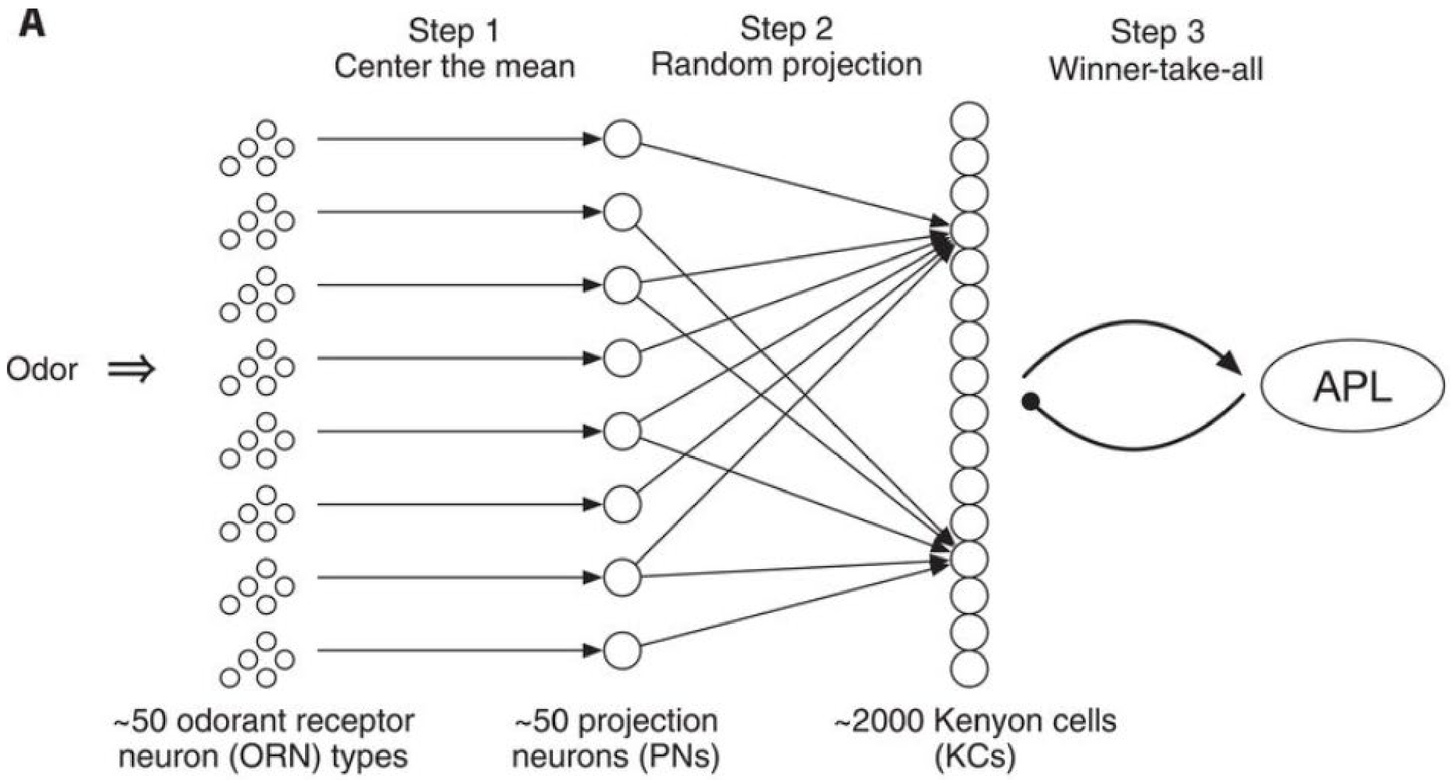
  the weight of $i \to j$ is multiplied by$(1 + \beta)$



Main parameters

- $n \sim 10^7$ #excitatory neurons in an area
- $k \sim 10^{3-4}$ maximum # firing neurons in any area
- $p \sim 0.001$ probability of recurrent and afferent synaptic connectivity
- $\beta \sim 0.1$ plasticity coefficient

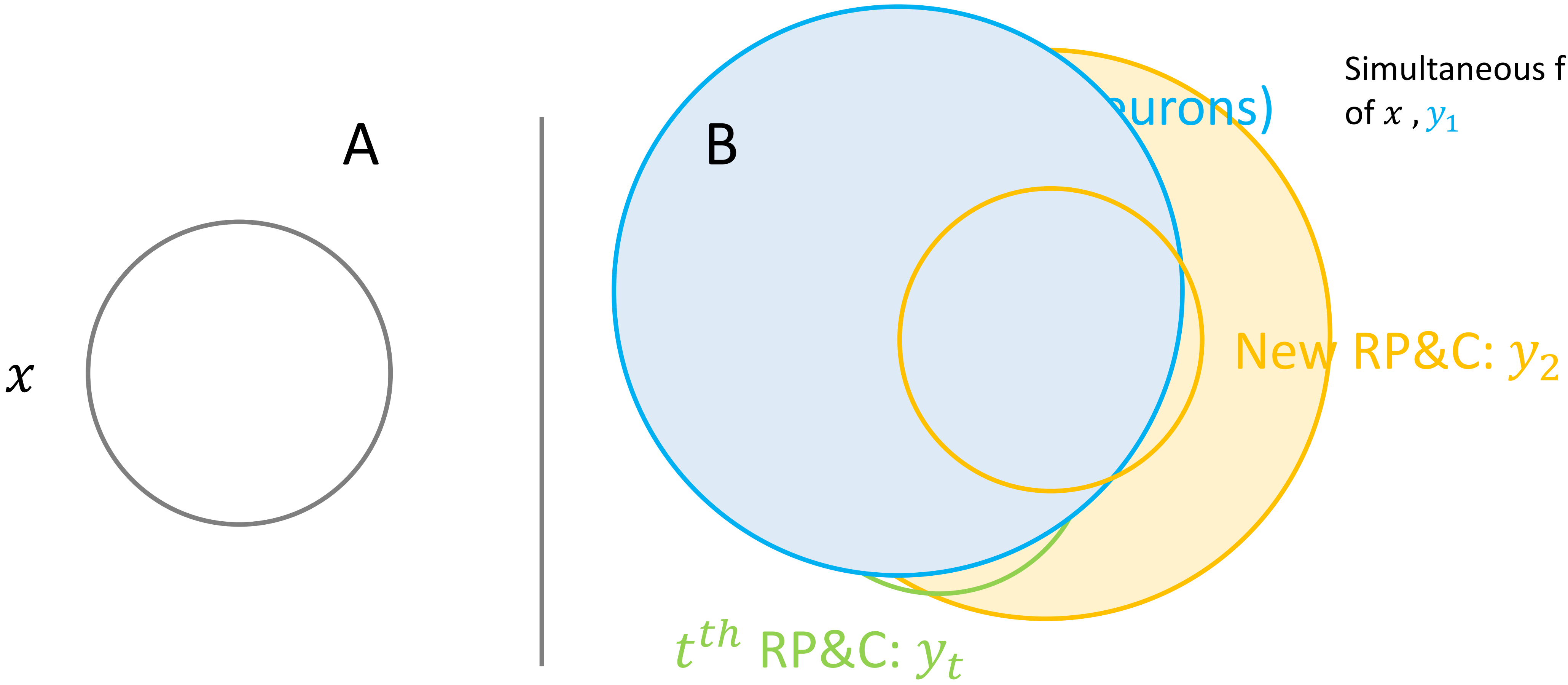
Random Projection and Cap Primitive (RP&C)



The selection of the k neurons (among n neurons) with highest synaptic input

Dasgupta, Sanjoy, Charles F. Stevens, and Saket Navlakha. “A Neural Algorithm for a Fundamental Computing Problem.” *Science* 358, no. 6364 (2017): 793–96. https://doi.org/10.1126/science.aam9868.

Assembly Projection

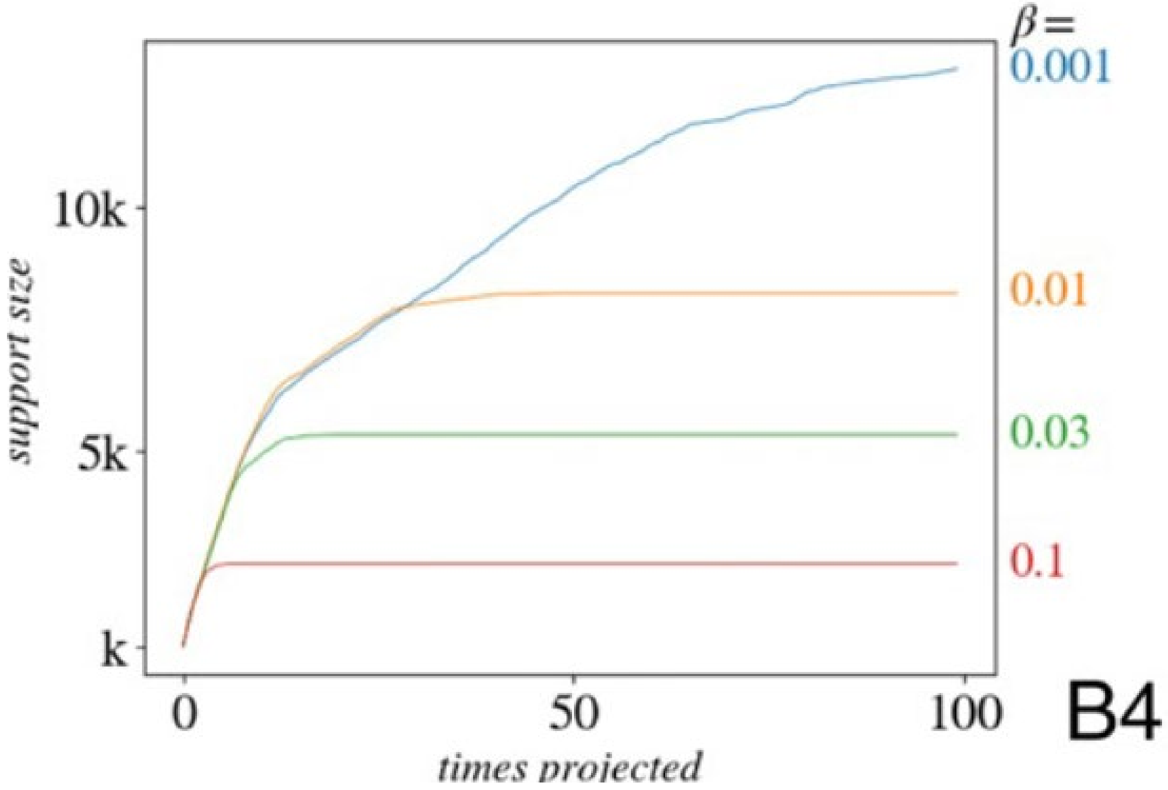


Simultaneous firing of x , y_1

Does this process converge? Is the result stable?

Assembly Projection (cont.)

Total number of neurons in area B that fired in the process



The number of times assembly x fires

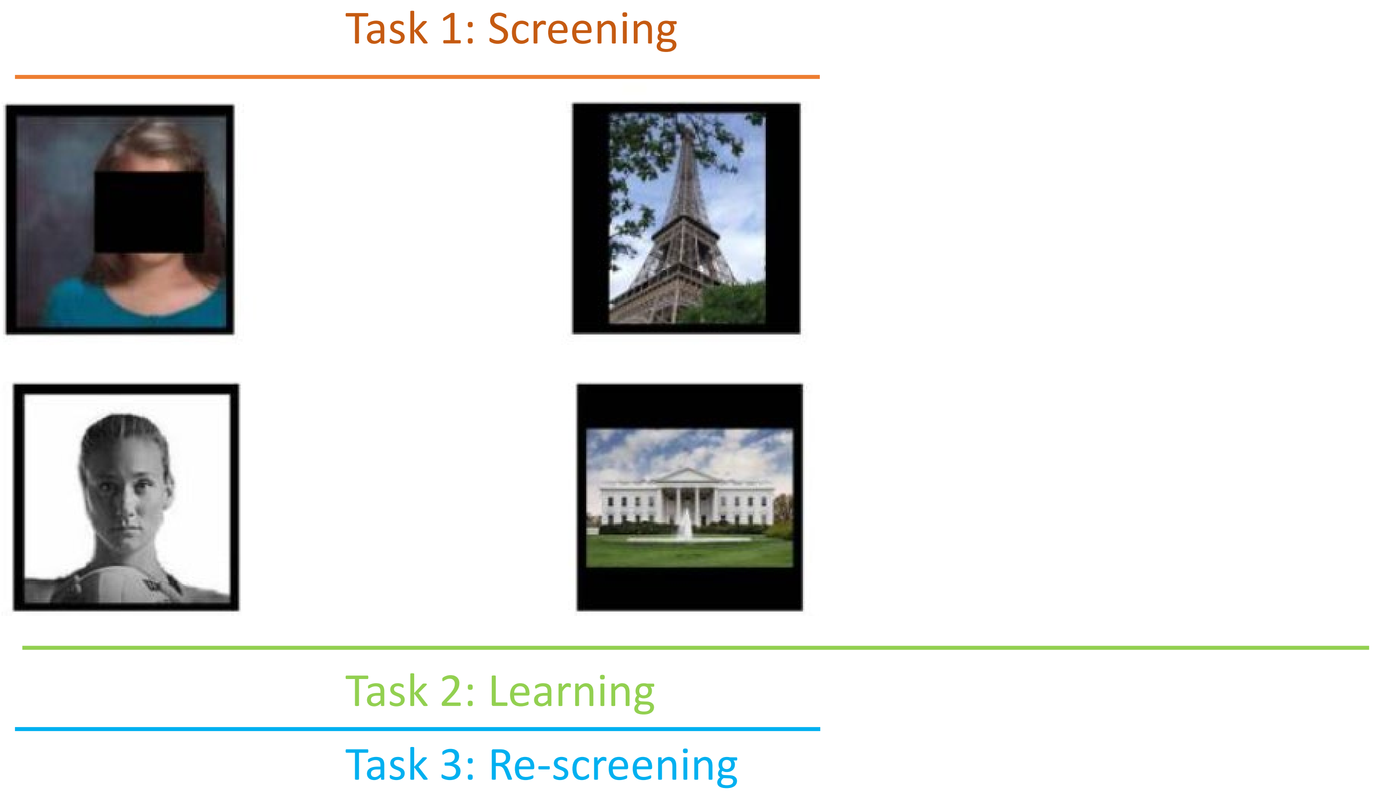
The process **converge** exponentially fast, with high probability, to create a new **stable** assembly y

Association

After exposure of the composite picture

- The subject learned the association (family member at the Eiffel tower) and the MTL neuron firing rate in response to the Eiffel tower increased.

- MTL units fired to White House and not to American beach volleyball player Kerri Walsh increased firing to Kerri Walsh.



Task 1: Screening

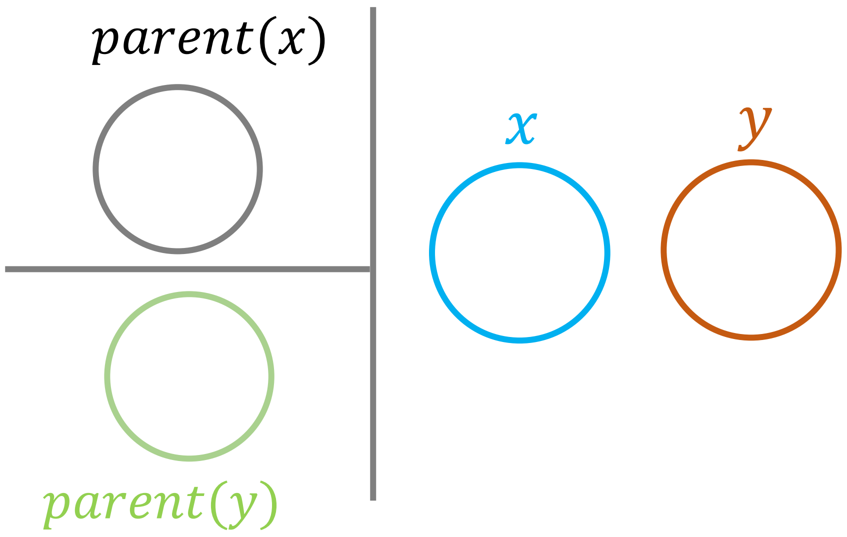
Task 2: Learning

Task 3: Re-screening

Ison, Matias J., Rodrigo Quian Quiroga, and Itzhak Fried. “Rapid Encoding of New Memories by Individual Neurons in the Human Brain.” *Neuron* 87, no. 1 (2015): 220–30. https://doi.org/10.1016/j.neuron.2015.06.016.

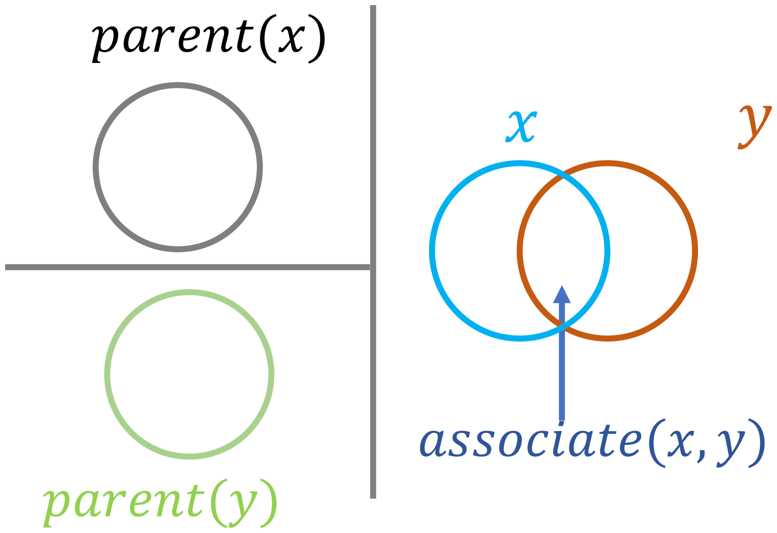
Association & Pattern Completion

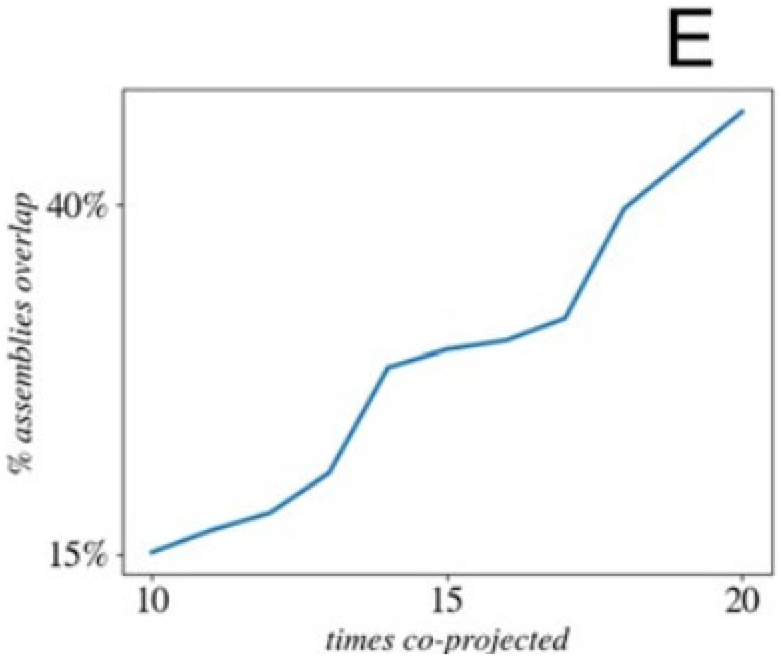
- Association: If two assemblies imprinting two different entities *co-occur*, then the *overlap* of the projected assemblies *increases*.



Association & Pattern Completion

- Association: If two assemblies imprinting two different entities *co-occur*, then the *overlap* of the projected assemblies *increases*.

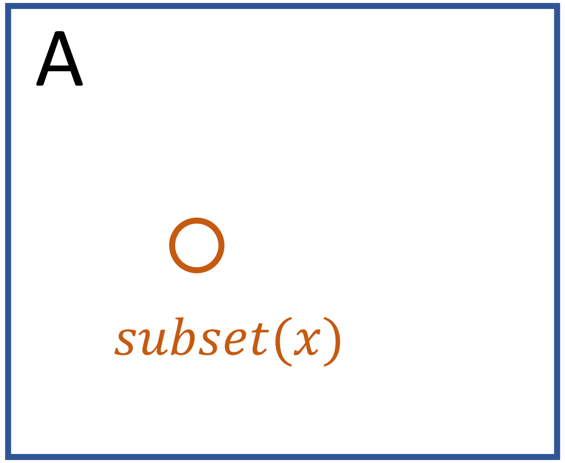




The number of times assembly pa(x),pa(y) fires

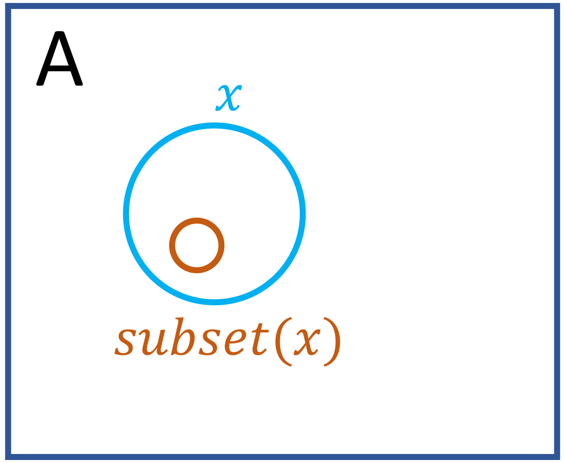
Association & Pattern Completion

- Association: If two assemblies imprinting two different entities *co-occur*, then the *overlap* of the projected assemblies *increases*.
- Pattern Completion: the firing of the whole assembly *x* in response to the firing of a small subset of its cells.



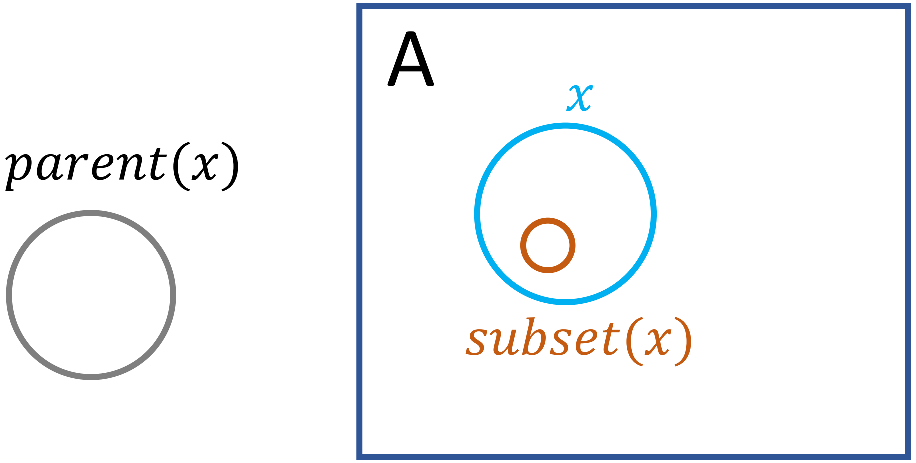
Association & Pattern Completion

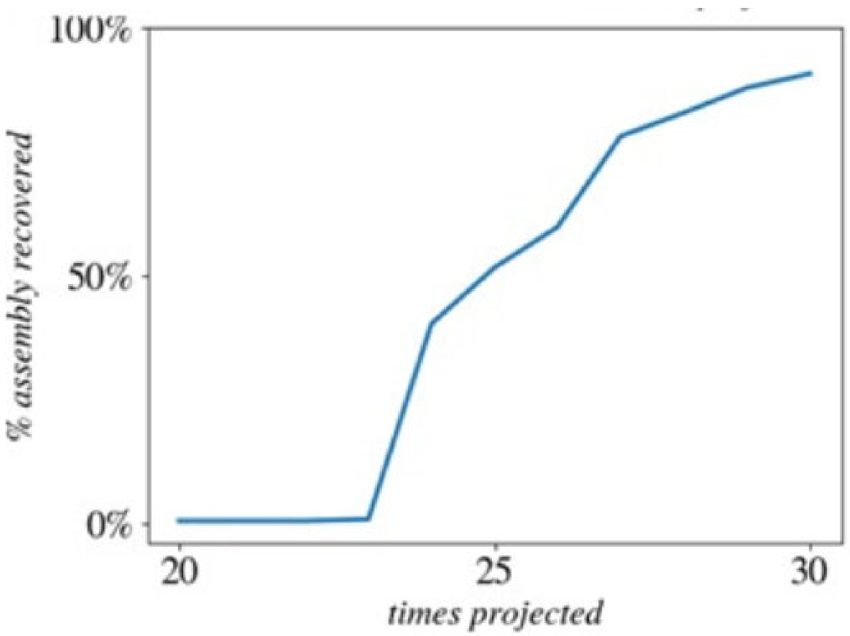
- Association: If two assemblies imprinting two different entities *co-occur*, then the *overlap* of the projected assemblies *increases*.
- Pattern Completion: the firing of the whole assembly *x* in response to the firing of a small subset of its cells.



Association & Pattern Completion

- Association: If two assemblies imprinting two different entities *co-occur*, then the *overlap* of the projected assemblies *increases*.
- Pattern Completion: Small parts of the assembly being able to *complete*, very accurately, the whole assembly.



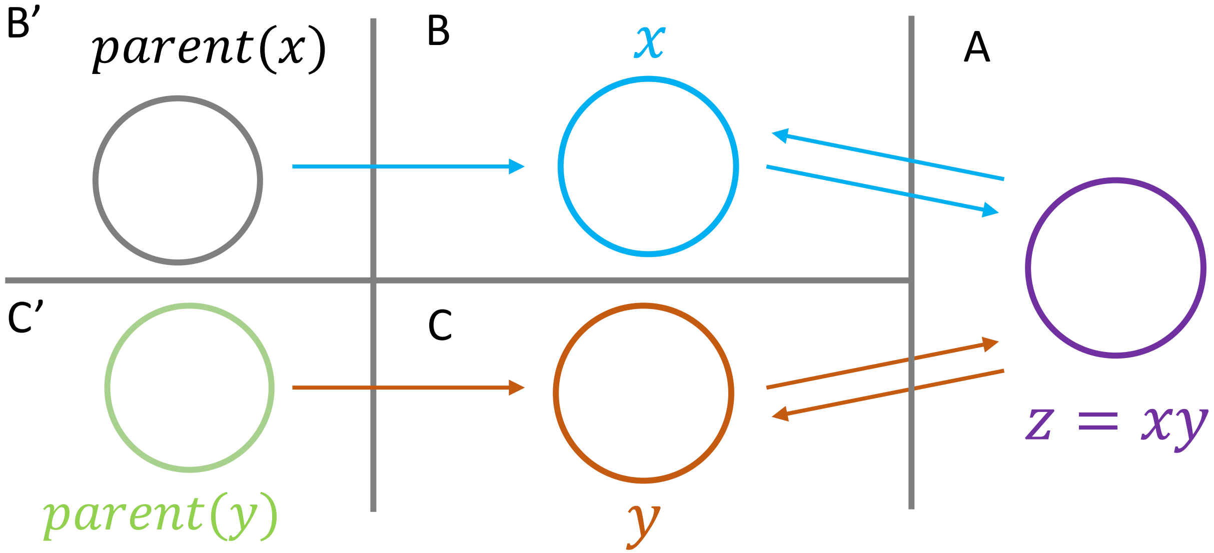


The number of times assembly x fires

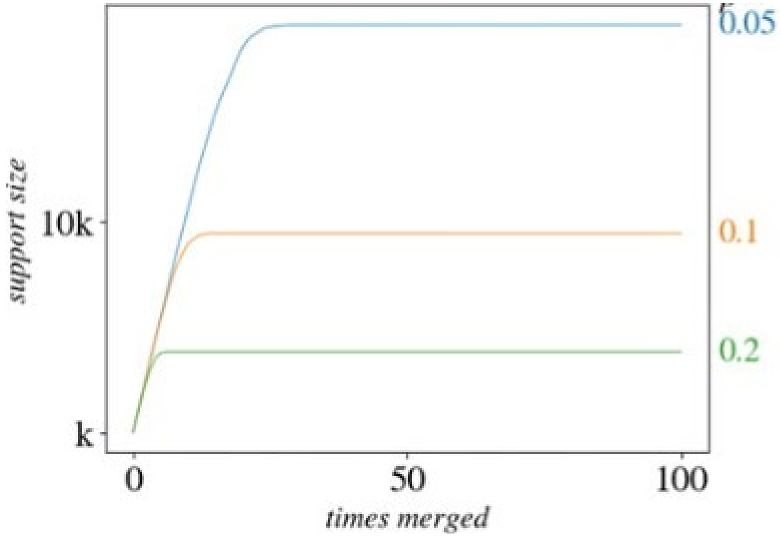
Merge

Areas to be merged

New assembly



Total number of neurons in area A that fired in the process

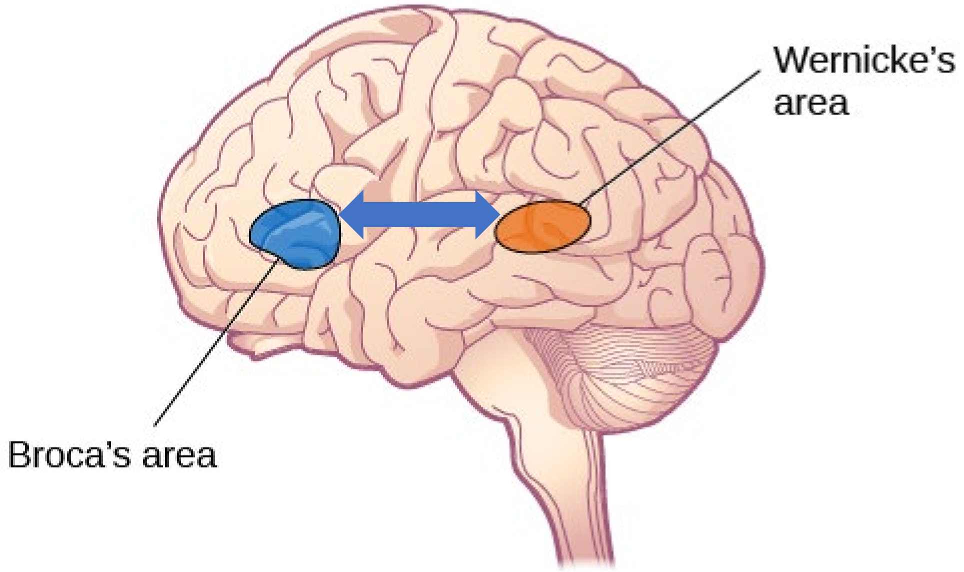


The number of times assembly x, y fires

Merge

Merge

Hierarchy type	+ (syntax)	– (word list)
	SE (+M) DAS SCHIFF SINKT (the ship sinks)	SE (–M) HALM SCHIFF SAFT (stem ship juice)
	PH (+M) AUF DAS SCHIFF (on the ship)	PH (–M) LAUCH MUND SCHIFF (leek mouth ship)



- Phrases and sentences activates parts of Broca's areas (implicated in syntactic processing)
- Words activates parts of Wernicke's area (implicated in word selection).

E. Zaccarella, L. Meyer, M. Makuuchi, A. D. Friederici, Building by syntax: The neuralbasis of minimal linguistic structures.Cerebr. Cortex27, 411–421 (2017).

Low-level operations

- `read(A)`: identify the assembly which has just fired in area A, and returns null otherwise
- `fire(x)`: fire assembly x in an area A
- `disinhibit(A)`: by default, the excitatory cells in an area A are inhibited unless explicitly disinhibited for a limited time period whose end is marked by the operation `inhibit(A)`
- for programming purposes, lack in justification

Assembly operations summary

- High level operations
  - `project(y,B,x)`
  - `pattern_complete(x,y)`
  - `associate(x,y)`
  - `merge(x,y,B,z)`
- Low level operations
  - `read(x)`
  - `fire(x)`
  - `disinhibit(x)`
  - `inhibit(x)`

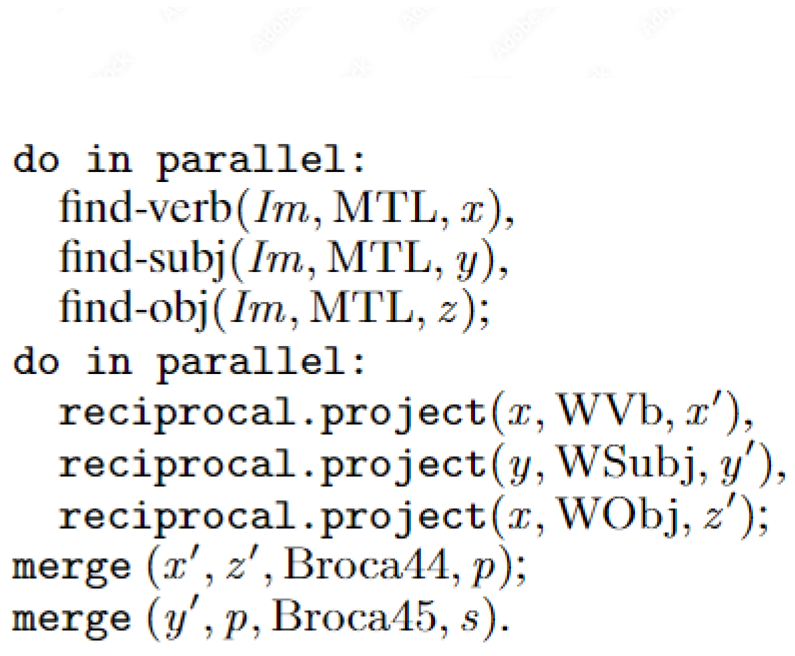
Discussion

- **Q:** Are the assembly operations real?
- **A:** observed/strongly suggested by experiments; can be compiled down to activity of neurons and synapses; mathematically and in simulations.

- **Q:** How powerful is this system
- **A:** Capable of implementing, under some assumptions, arbitrary computations on $O(\sqrt{n}/k)$ bits of memory.

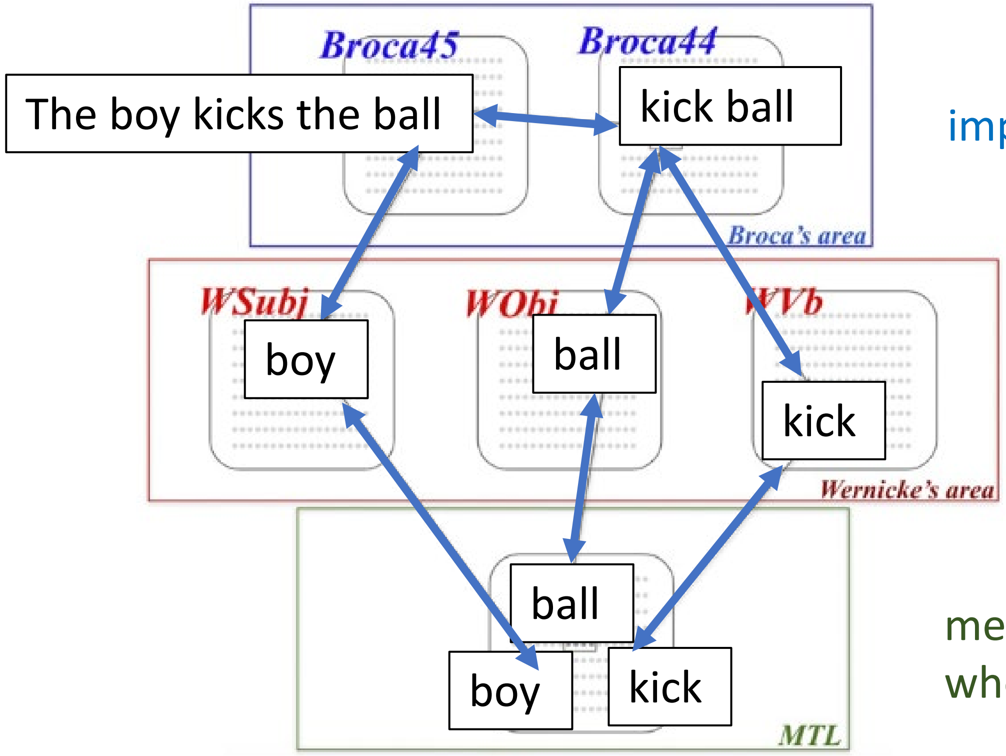
Discussion (cont.)

- **Q:** Can the Assembly Calculus help elucidate the mystery of language?
- **A:** Example of syntax in language generation



```
do in parallel:
   find-verb($Im$, MTL, $x$),
   find-subj($Im$, MTL, $y$),
   find-obj($Im$, MTL, $z$);
do in parallel:
   reciprocal.project($x$, WVb, $x'$),
   reciprocal.project($y$, WSubj, $y'$),
   reciprocal.project($x$, WObj, $z'$);
merge ($x'$, $z'$, Broca44, $p$);
merge ($y'$, $p$, Broca45, $s$).
```

Image: Adobe Stock



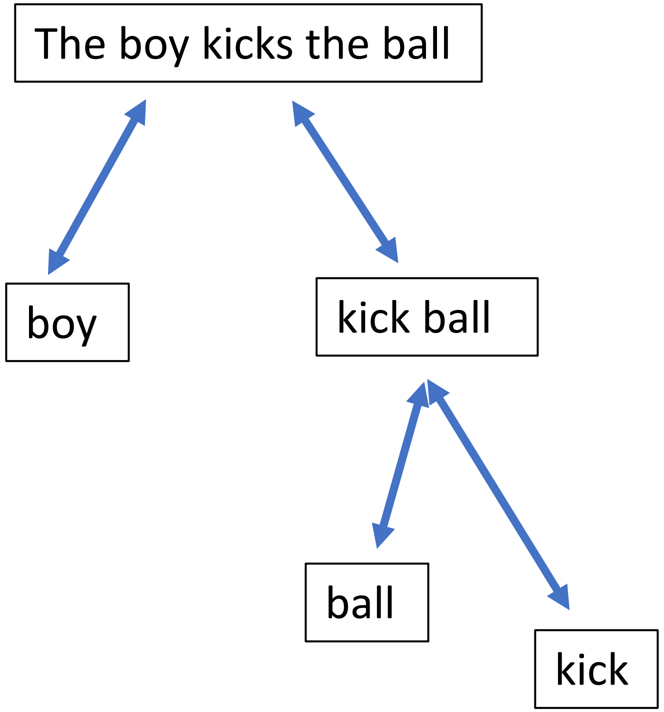
implicated in syntax

implicated in the use of words

medial temporal lobe; where lexicon are stored

Discussion (cont.)

- **Q:** Can the Assembly Calculus help elucidate the mystery of language?
- **A:** Example of sentence articulation (activation of assembly from root to leaf)



The boy kicks the ball

boy

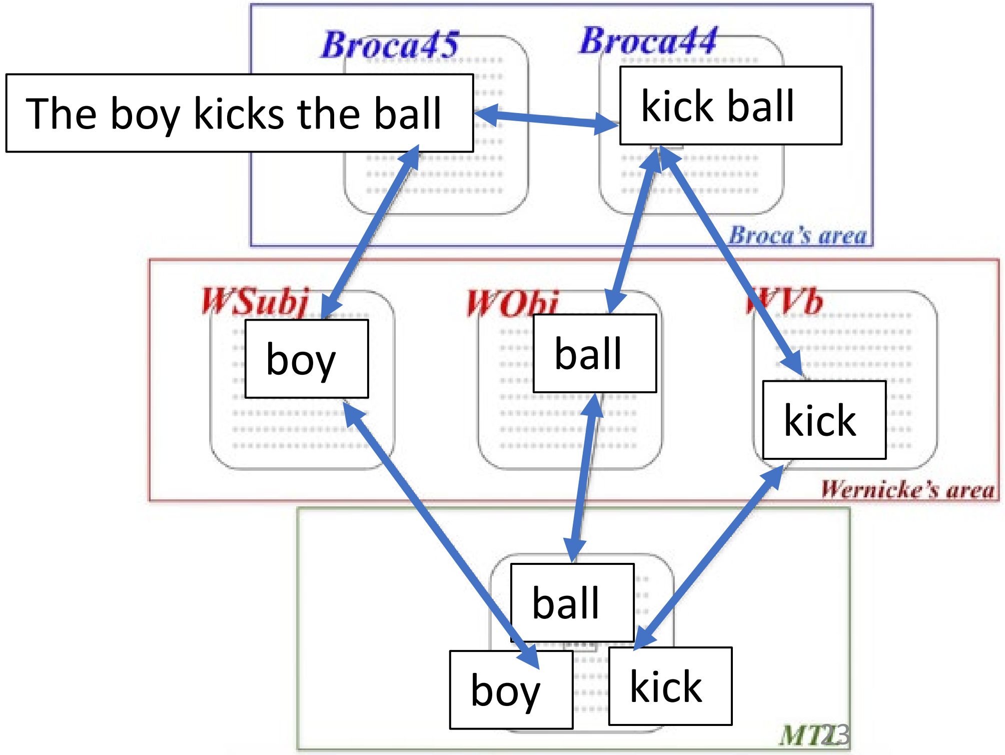
kick ball

ball

kick

Discussion (cont.)

- **Q:** Can the Assembly Calculus help elucidate the mystery of language?
- **A:** many aspects still unknown
- Find-tasks implementation;
- Articles in front of nouns (the, an, a);
- Verb tenses (kicks, kicking, kicked);



Thank you