What are the Computational Challenges for Cortex?

Leslie Valiant
Harvard University

Light Bulb Problem

Large *N*

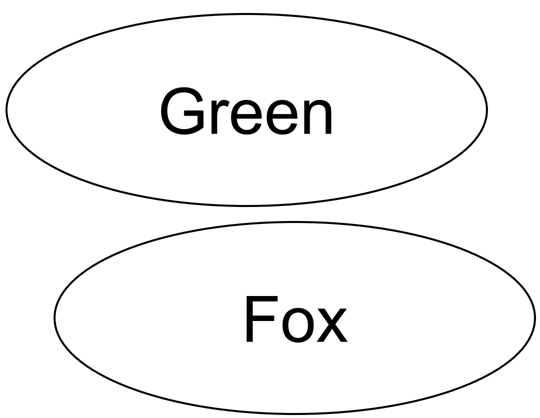
Cognitive Tasks?

E.g. How many neurons do you use to remember each new person you meet at HLF7?

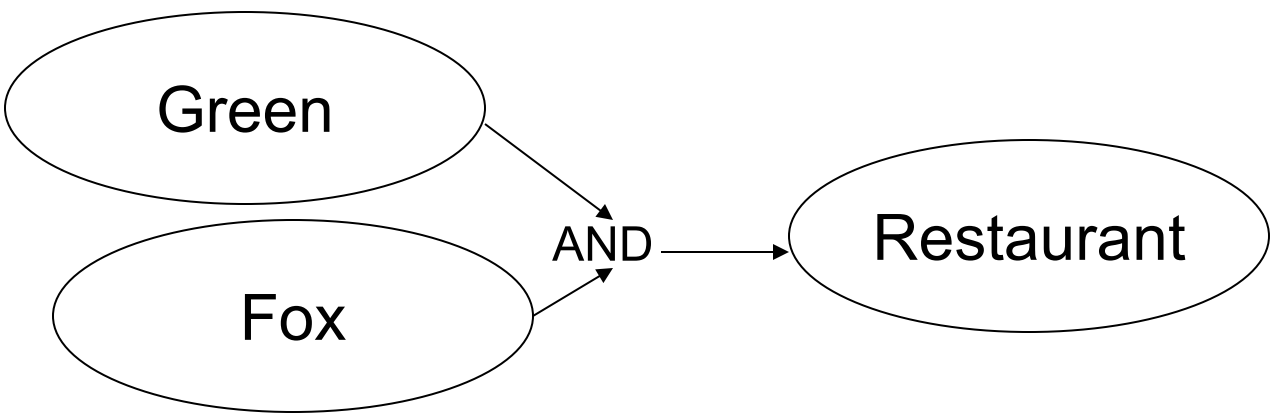
1, 10, 10^2, …. , or 10^7?

Symbolic Processing

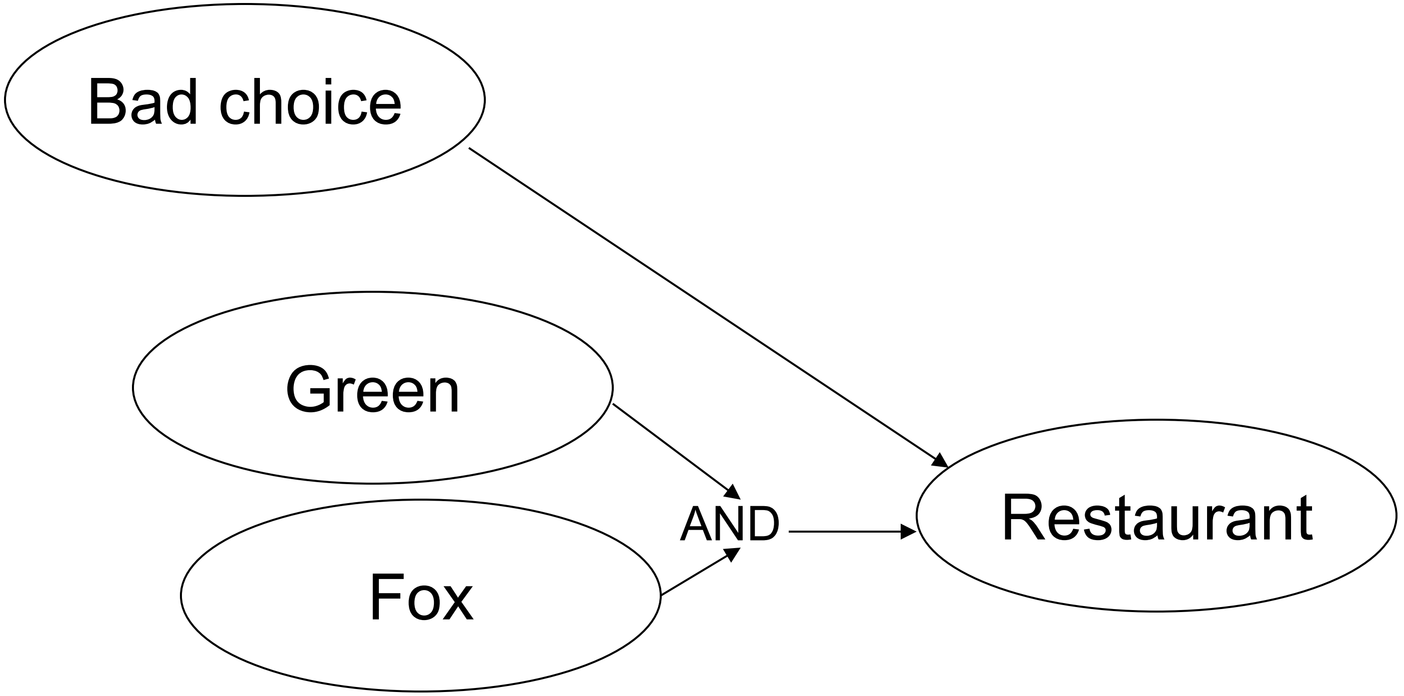
Symbolic Processing



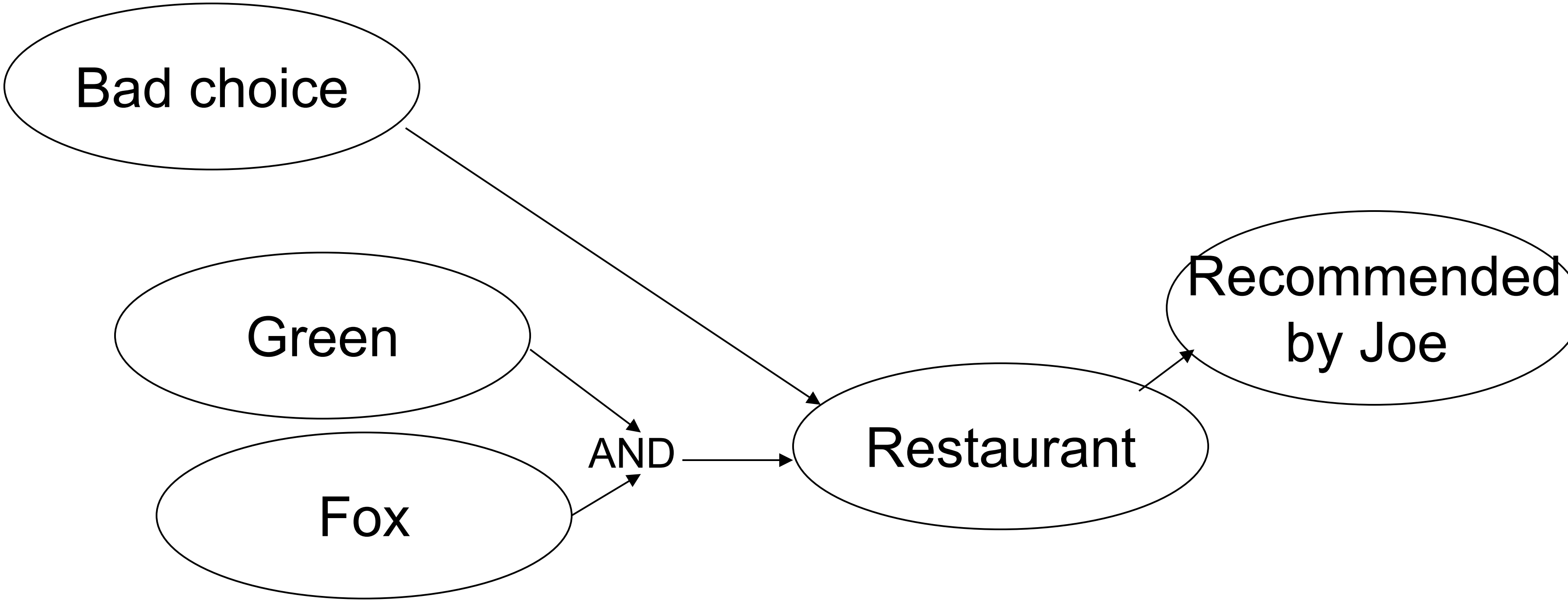
Symbolic Processing



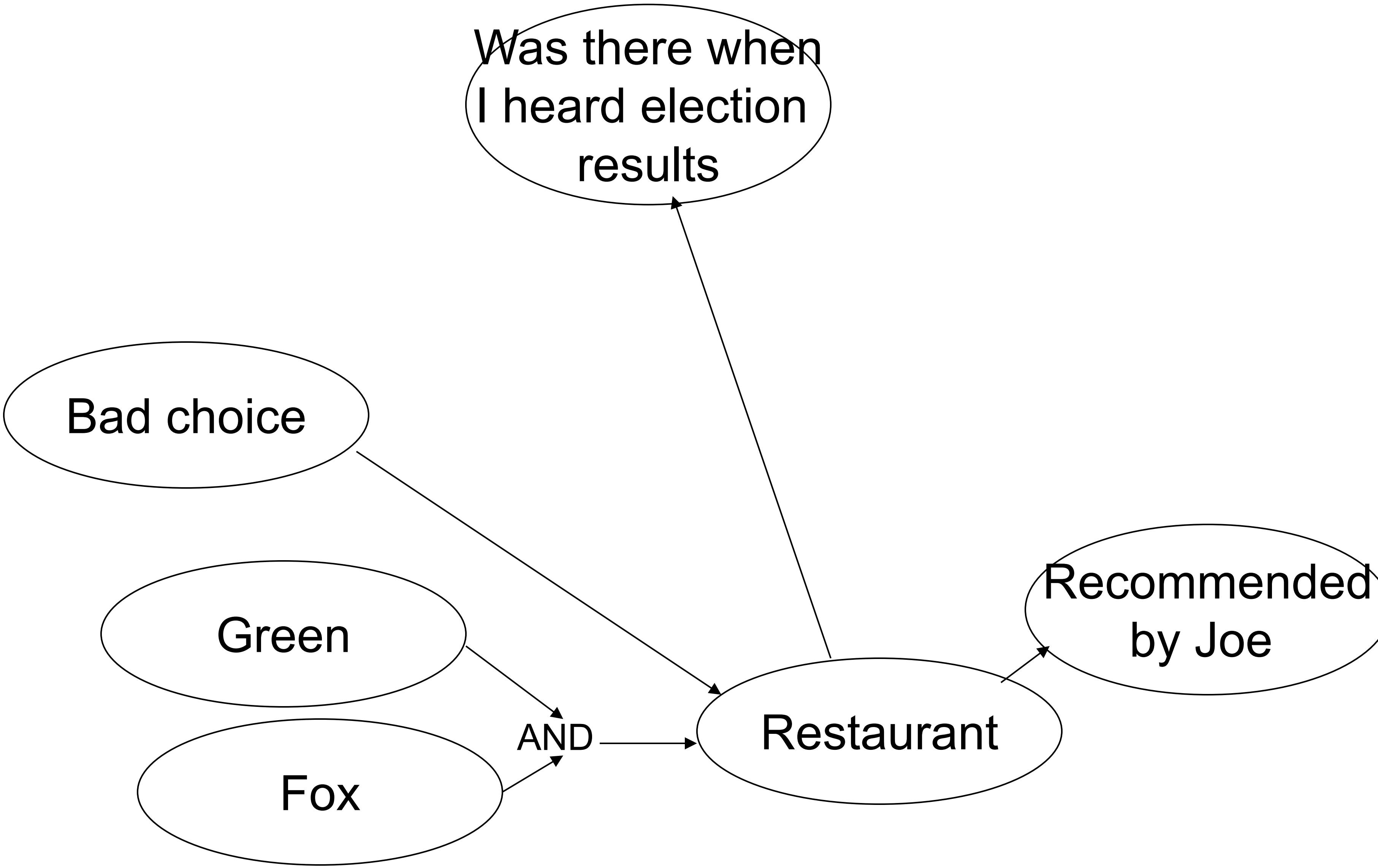
Symbolic Processing



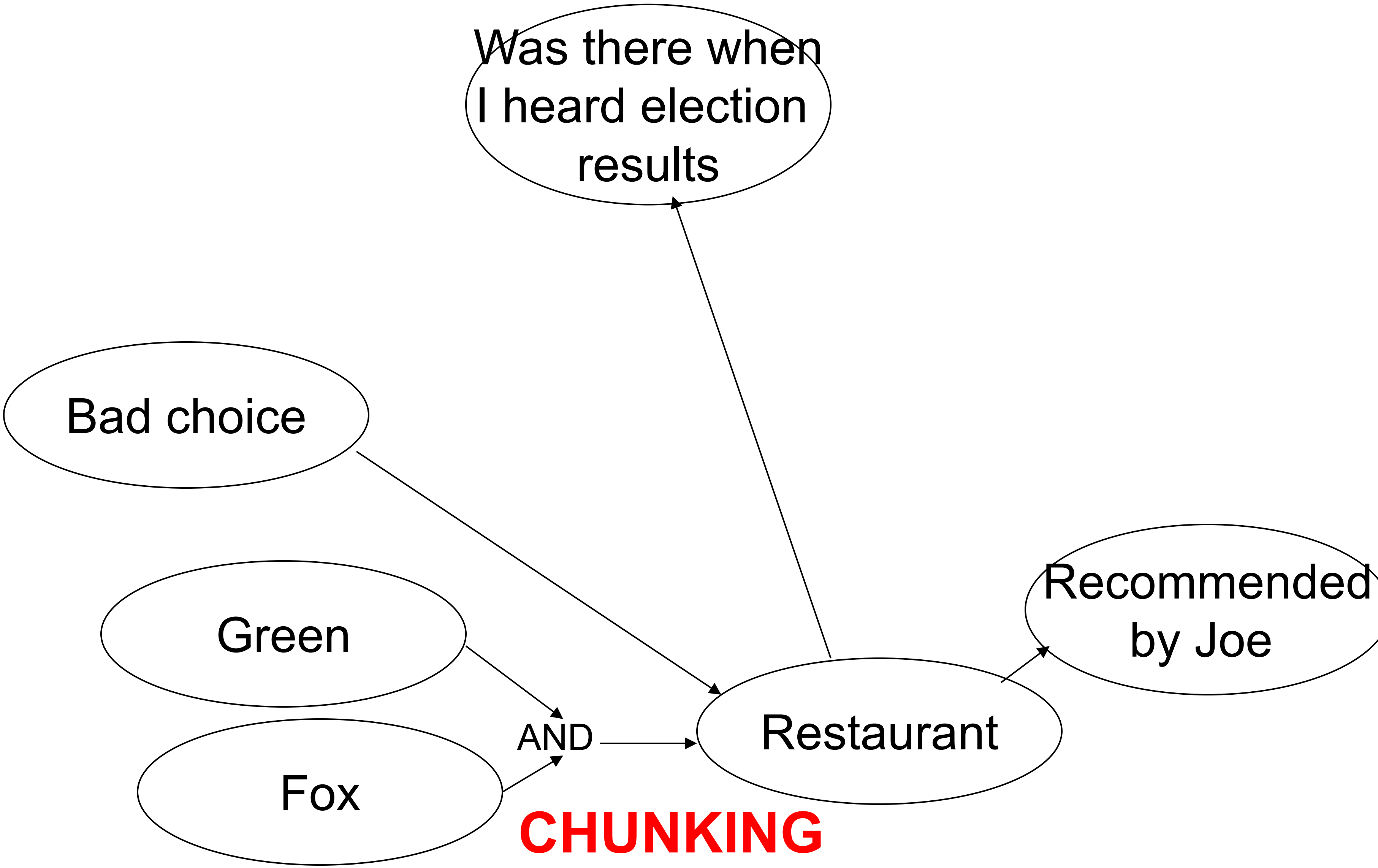
Symbolic Processing



Symbolic Processing



Symbolic Processing



Cortex can achieve this, for hundreds of thousands of such tasks in succession.

Cortex can achieve this, for hundreds of thousands of such tasks in succession.

No *generally agreed* theory known of how it can, even in principle.

Some Requirements

1. One-shot learning.

Some Requirements

1. One-shot learning.
2. Lifelong learning of $\sim 10^5$ tasks.

Some Requirements

1. One-shot learning.
2. Lifelong learning of ~10^5 tasks.
3. Learning results in *subcircuit creation*.

Some Requirements

1. One-shot learning.
2. Lifelong learning of ~10^5 tasks.
3. Learning results in *subcircuit creation*.
4. Subcircuits composed on-the-fly when evaluated.

Some Requirements

1. One-shot learning.
2. Lifelong learning of ~10^5 tasks.
3. Learning results in *subcircuit creation*.
4. Subcircuits composed on-the-fly when evaluated.
5. Quantitative plausibility as far as neuron numbers, connection numbers, connection strengths.

Some Requirements

1. One-shot learning.
2. Lifelong learning of $\sim 10^5$ tasks.
3. Learning results in *subcircuit creation*.
4. Subcircuits composed on-the-fly when evaluated.
5. Quantitative plausibility as far as neuron numbers, connection numbers, connection strengths.
6. Architectural plausibility (e.g. one-way connections.)

Some Requirements

1. One-shot learning.
2. Lifelong learning of ~10^5 tasks.
3. Learning results in *subcircuit creation*.
4. Subcircuits composed on-the-fly when evaluated.
5. Quantitative plausibility as far as neuron numbers, connection numbers, connection strengths.
6. Architectural plausibility (e.g. one-way connections.)
7. Cognitively adequate set of primitives

Constraints on Brain Computation

- No addressing mechanism!
- Slow – has to do much in 100 steps.
- Neurons sparsely connected, communication – challenged.
- Resource constraints:

  n neurons,

  d connections to/from each,

  maximum synaptic strength $1/k$.
- But long distance communication by stylized spikes – information carried in the timing.

Neuroidal Model as Programming Language

- Directed graph, spikes for communication.

Neuroidal Model as Programming Language

- Directed graph, spikes for communication.
- Each neuron executes a program locally (has states and can do thresholds on inputs.) $\leftrightarrow$ Many genetically different neurons.

Neuroidal Model as Programming Language

- Directed graph, spikes for communication.
- Each neuron executes a program locally (has states and can do thresholds on inputs.) ↔ Many genetically different neurons.
- Synapses also have state beyond weight ↔ Many genetically different synapses.

Neuroidal Model as Programming Language

- Directed graph, spikes for communication.
- Each neuron executes a program locally (has states and can do thresholds on inputs.) ↔ Many genetically different neurons.
- Synapses also have state beyond weight ↔ Many genetically different synapses.
- Timing mechanisms ↔ spike time dependent plasticity.

Neuroidal Model as Programming Language

- Directed graph, spikes for communication.
- Each neuron executes a program locally (has states and can do thresholds on inputs.) ↔ Many genetically different neurons.
- Synapses also have state beyond weight ↔ Many genetically different synapses.
- Timing mechanisms ↔ spike time dependent plasticity.

- (Aim to underestimate cortex.)

Neuroidal Model as Resource Model

- n neurons
- each connected to and from d others.
- max. synaptic weights $1/k \times$ threshold.
- time

Representations

Each real world "item" corresponds to a set S of $\approx r$ randomly chosen neuron in certain area.

Representations

Each real world "item" corresponds to a set S of $\approx r$ randomly chosen neuron in certain area.

(This and above modeling choices all made for theoretical reasons in 1994.)

Representations

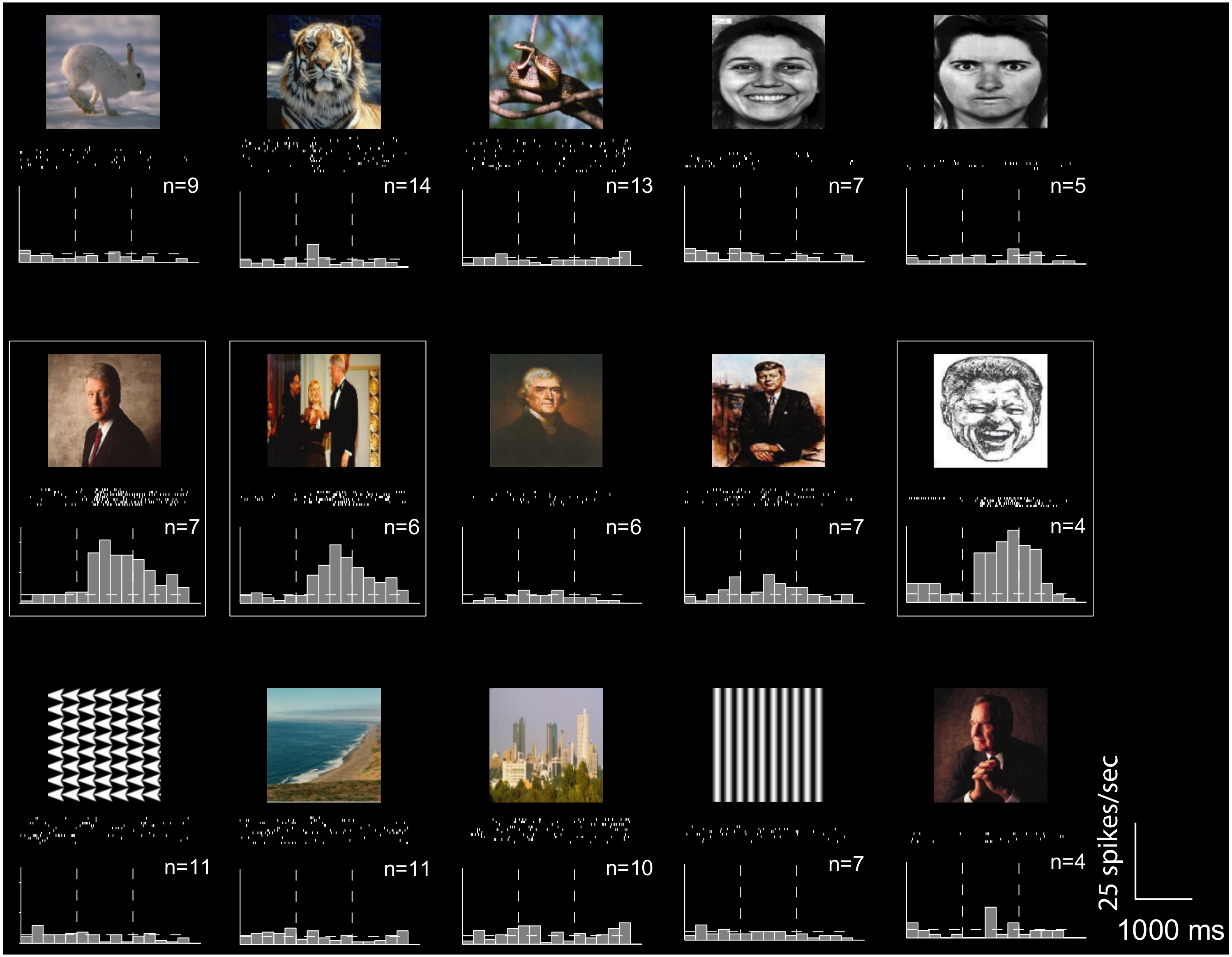
Each real world "item" corresponds to a set S of $\approx r$ randomly chosen neuron in certain area.

(This and above modeling choices all made for theoretical reasons in 1994.)

Note: Correspondences between items and neurons are "experimentally determinable"

r is large.

(e.g. in hippocampus, IT, olfactory bulb)



Kreiman, Fried, Koch (2002) *PNAS* **99**:8378; Crick, Koch, Kreiman, Fried (2004) *Neurosurgery* **55**:273

Are Sets Random?

The set of grid cells for any one offset looks random.

Random Access Tasks: Type I

Allocate Storage to New Concept

Random Access Tasks: Type I

Allocate Storage to New Concept

(1) e.g. First time you heard of “Boris Johnson”.

Random Access Tasks: Type I

Allocate Storage to New Concept

(1) e.g. First time you heard of “Boris Johnson”.

Hierarchical Memorization: For any stored items A, B, allocate neurons to new item C and change synaptic weights so that in future A and B active will cause C to be active also.

Random Access Tasks: Type I
Allocate Storage to New Concept

(1) e.g. First time you heard of "Boris Johnson".

Hierarchical Memorization: For any stored items A, B, allocate neurons to new item C and change synaptic weights so that in future A and B active will cause C to be active also. (Chunking)

Random Access Tasks: Type I

Allocate Storage to New Concept

(1) e.g. First time you heard of “Boris Johnson”.

Hierarchical Memorization: For any stored items A, B, allocate neurons to new item C and change synaptic weights so that in future A and B active will cause C to be active also. (Chunking)

*Constructs or modifies a **circuit** in response to stimulus.*

Random Access Tasks: Type II

Add relationships among represented concepts

(2) (e.g. Boris Johnson→ Prime Minister)

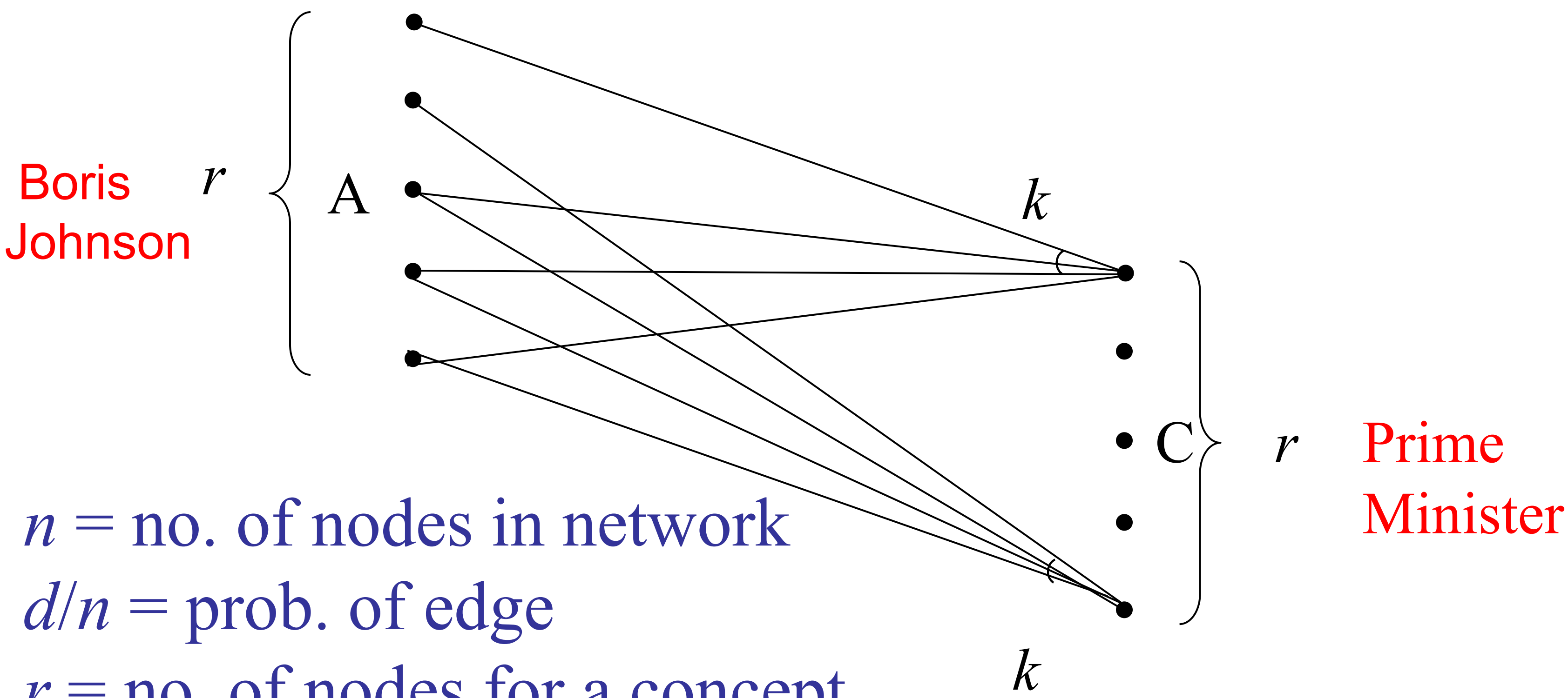
Association: For any stored items A, B, change synaptic weights so that in future when A is active then B will be caused to be also.

(c.f. Willshaw 1969)

Representations: Disjoint or Shared?

- Disjoint: Each neuron represents just one, possibly complex, item.
- Shared: Each neuron may represent many items.

Network Requirement for **Association** on Random Graph



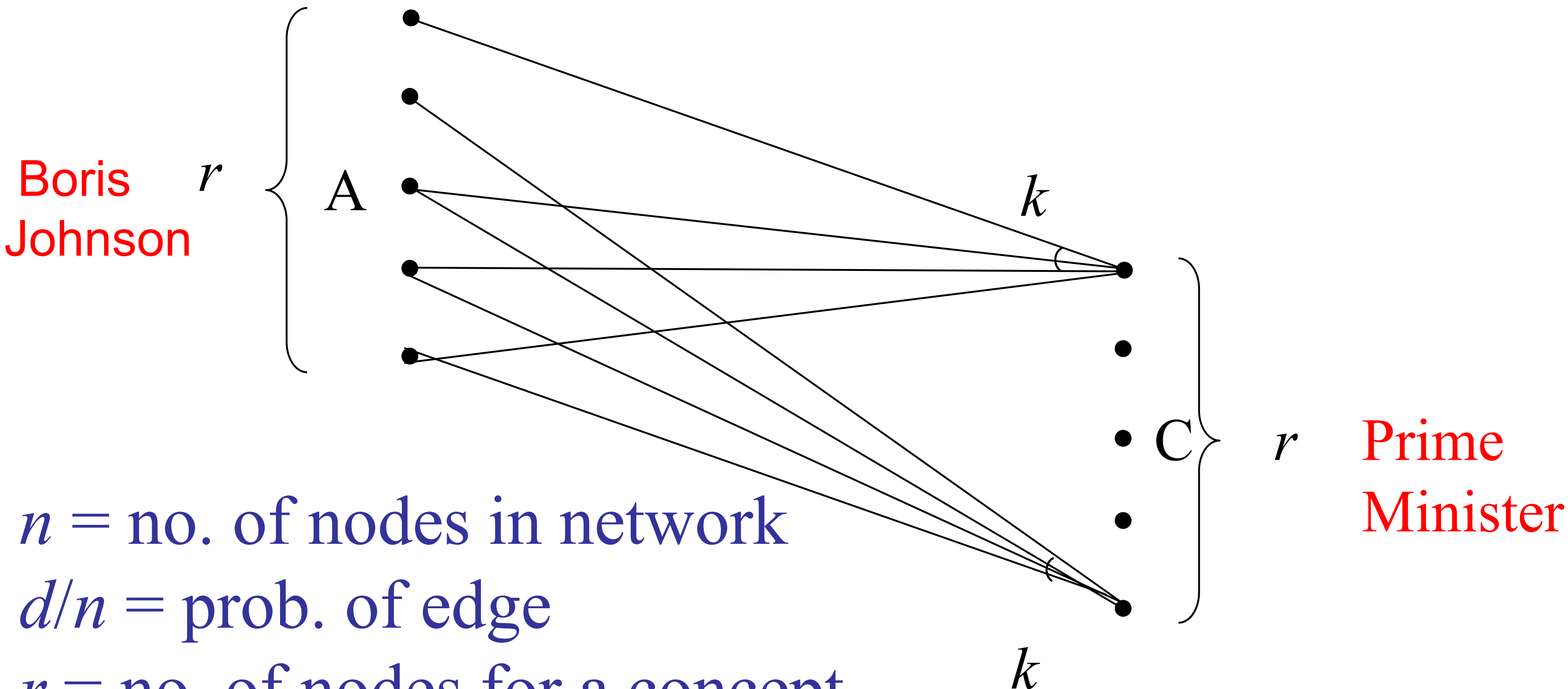
n = no. of nodes in network

d/n = prob. of edge

r = no. of nodes for a concept

k = min. no. of inputs to fire a node

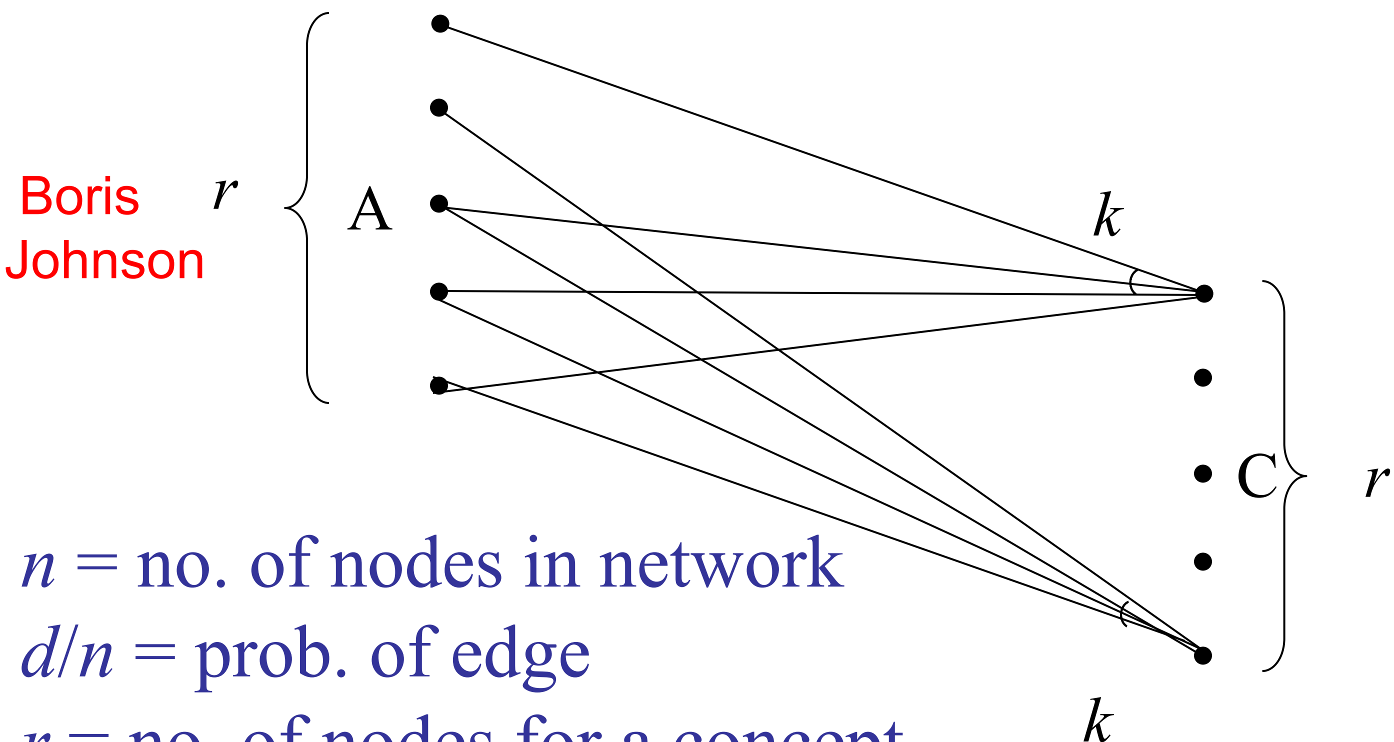
Network Requirement for **Association** on Random Graph



n = no. of nodes in network
d/n = prob. of edge
r = no. of nodes for a concept
k = min. no. of inputs to fire a node

$$\beta(r, d/n, k) \sim 1 \text{ if } rd/n \sim k.$$

Network Requirement for **Association** on Random Graph

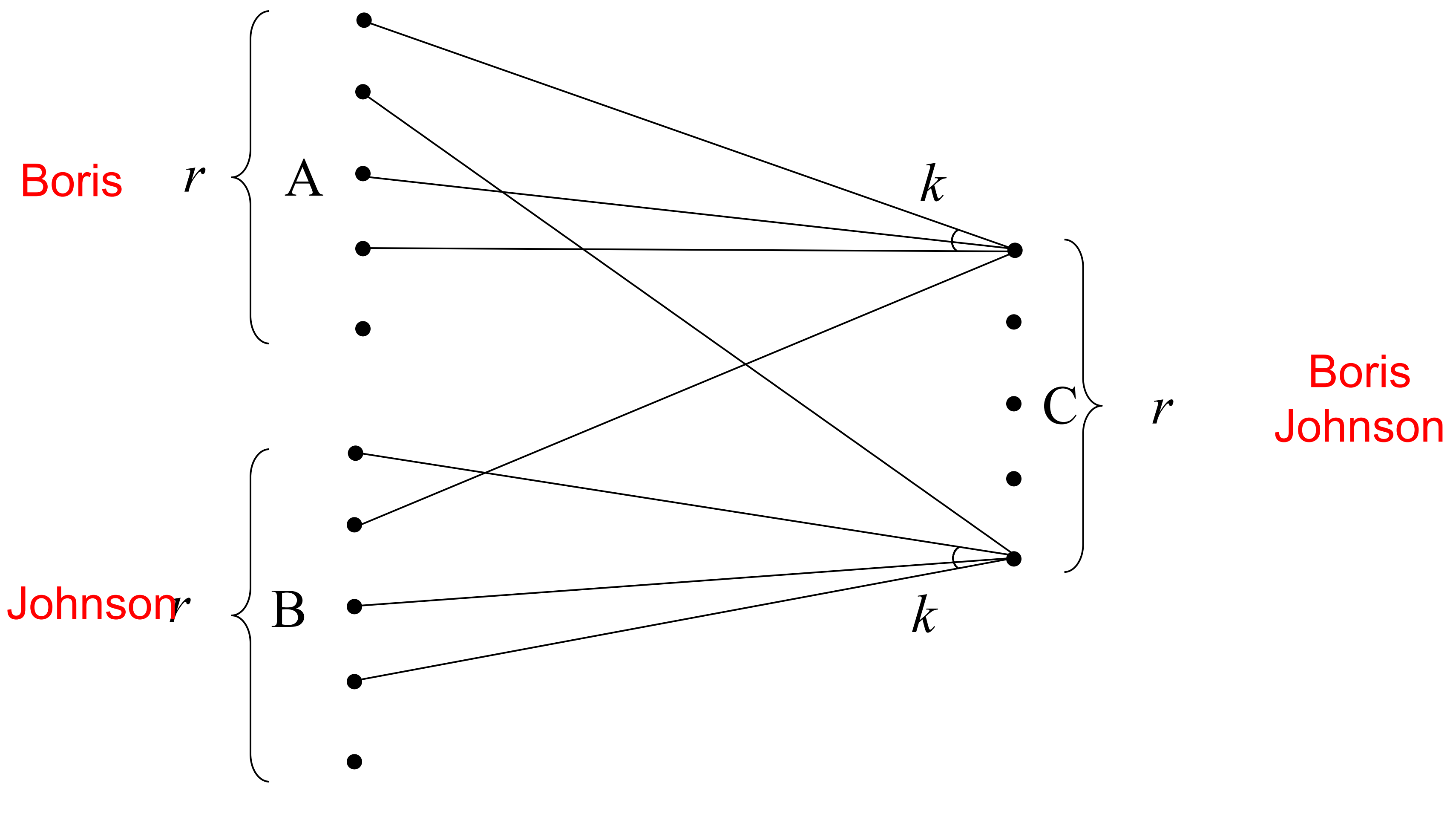


n = no. of nodes in network
d/n = prob. of edge
r = no. of nodes for a concept
k = min. no. of inputs to fire a node

$$\beta(r, d/n, k) \sim 1 \text{ if } rd/n \sim k.$$

Association: a systems level Hebb's Rule?

Hierarchical Memorization



Actual Conditions Are More Complex

…..

Finding Capacity by Simulations

(V. Feldman & LV, *Neural Computation*, 2009)

Simulate mixed sequences of associations, supervised memorization, and inductive learning tasks, on initial allocation by hierarchical memorization.

Results of Simulations: Regime α

(V.Feldman & LV, *Neural Computation*, 2009)

$n = 10^8$ neurons.

$d = 8{,}000$ connections per neuron.

$k = 16$ (i.e. inputs from 16 needed for a.p.)

$r = 360000$ neurons per item, shared.

$\downarrow$

Sequences of 3,200 actions can be supported with small interference.

Results of Simulations: Regime β

(V.Feldman & LV, *Neural Computation*, 2009)

$n = 10^8$ neurons.

$d = 4{,}000$ connections per neuron.

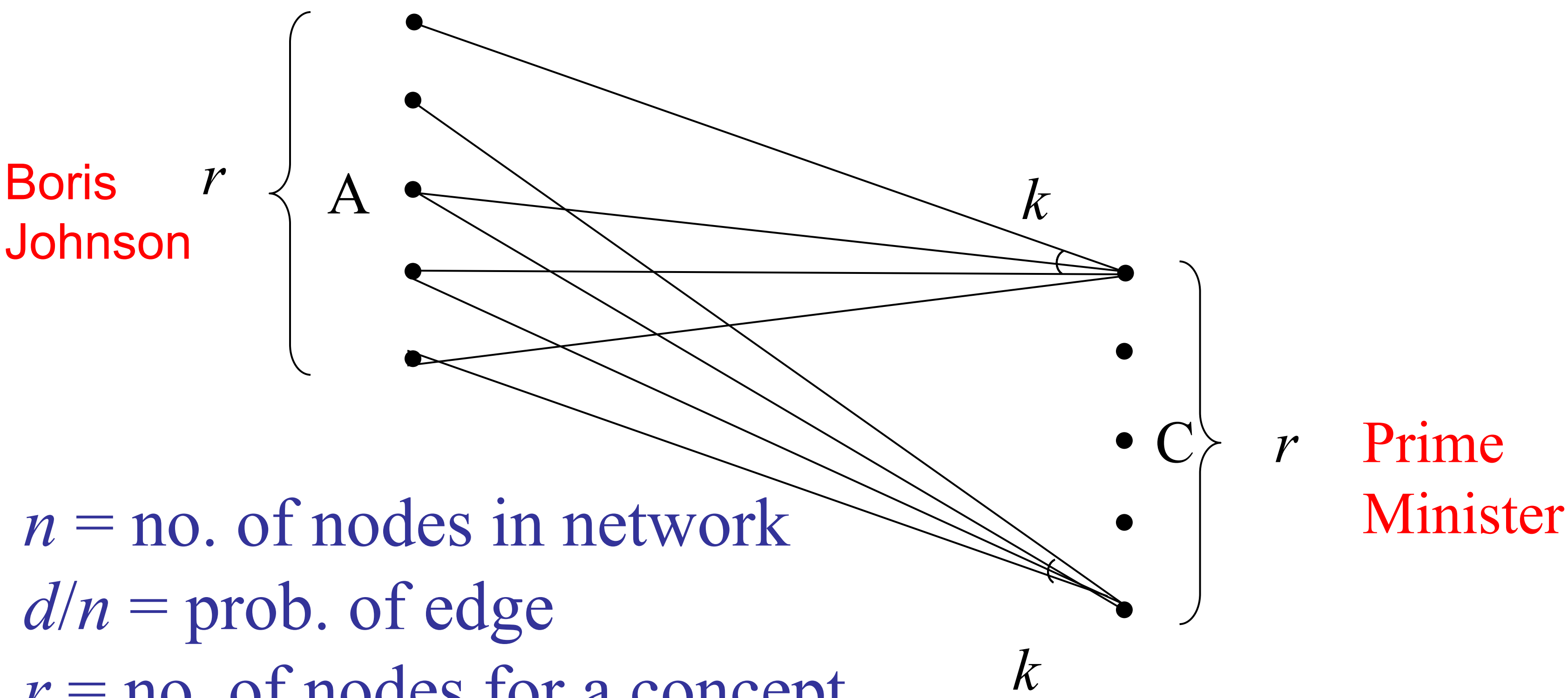
$k = 1$ (i.e. maximally strong synapses)

$r = 100$ neurons per item, disjoint.

↓

Sequences of 250,000 actions can be supported with small interference.

Network Requirement for **Basic Mechanism** for Association



n = no. of nodes in network

d/n = prob. of edge

r = no. of nodes for a concept

k = min. no. of inputs to fire a node

Complexity analysis for realizing Association

Association instance:

$(X_1 \rightarrow Y_1, \ldots, X_C \rightarrow Y_C)$: w.h.p. if X_i all fire Y_i all fire, and ~0 of rest.

How large can C be in terms of n, d, k?

In general $|X_i| = R$, $|Y_i| = r$.

For composability $R = r$.

Can $C \sim dn$ be achieved? E.g. Can $C \sim n^{3/2}$ if $d = n^{1/2}$?

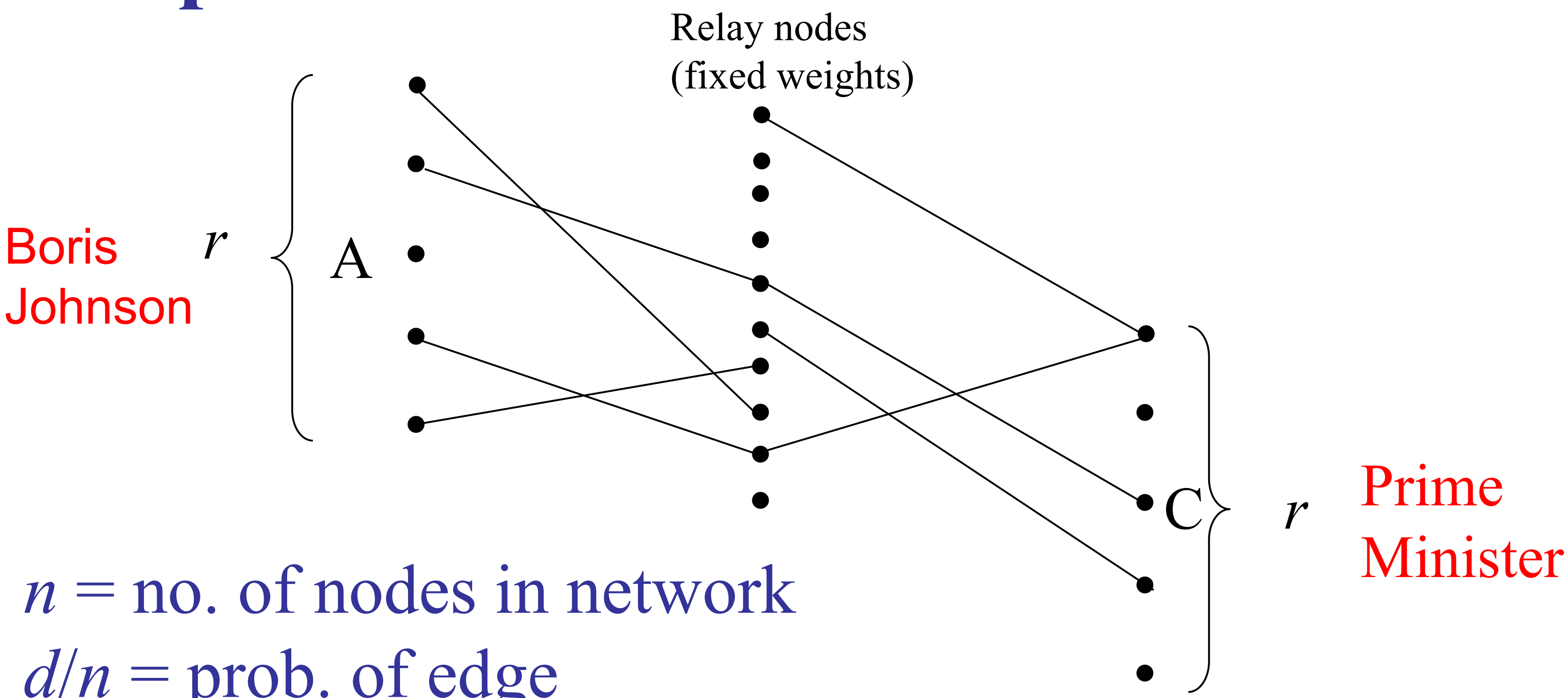
Theorem 1 *If … $C = \Theta(dn)$ achieved by* ***Basic Mechanism*** *to polylog with $R = nr/d$, $r = 3k$, $k = K\log_2 n$, for K large enough.*

Theorem 2 *If … for the* ***Basic Mechanism*** $C \leq (d^2R)/(k^2r)$.

($O(n)$ if $d = n^{1/2}$ and $r = R$).

(c.f. Willshaw 1969)

Network Requirement for **Expansive Mechanism** for Association



n = no. of nodes in network
d/n = prob. of edge
r = no. of nodes for a concept
k = min. no. of inputs to fire a node

Locust

How might these “systems level primitives” be validated experimentally?

“In-circuit” testing.

Validating the Association Primitive

Association: Stimulate sets *A*, *B* so that result is: If in future *A* stimulated then *B* will become active.

Validating the Association Primitive

Association: Stimulate sets *A*, *B* so that result is: If in future *A* stimulated then *B* will become active.

Very Strong Hypothesis: For *random A, B* of right sizes.

Validating the Association Primitive

Association: Stimulate sets *A*, *B* so that result is: If in future *A* stimulated then *B* will become active.

Very Strong Hypothesis: For *random A, B* of right sizes.

Want to determine whether cortex is capable of this building block of computation.

THANK YOU