
welcome
to Lecture 7
Computation and the Brain

Talk by Dr. Kenneth Kay

What happened last Wednesday

[Bengio et al. 2016] on STDP as biologically plausible backprop

- cf: [Lillicrap et al. 2016] on random (and thus more biologically plausible) backprop

My take on biological plausibility of backprop

- Backpropagation can be thought as evolution
- The feedforward circuit and/or the weights are a phenotype that depends on *many* genes
- Minibatch: the collective experience of a generation
- Selection changes the allele statistics of the next generation through a variant of gradient descent
- ***Can learning happen this way?***

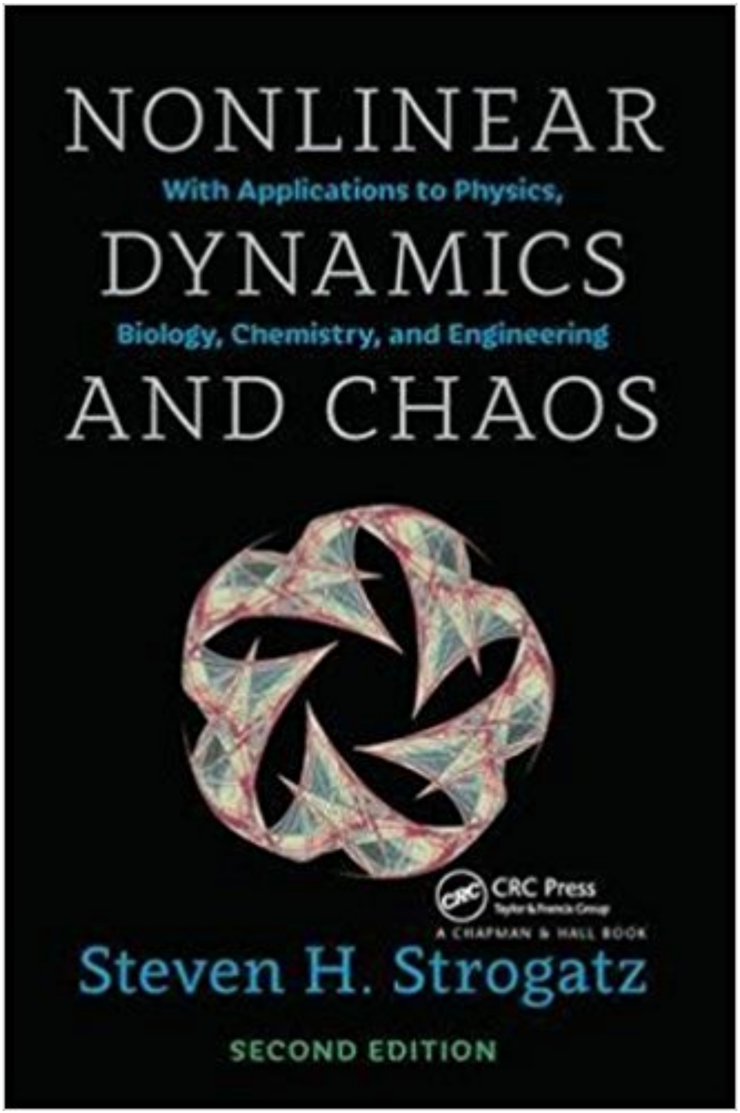
How to measure information in spike trains (a bit array for each Δt, then subtract noise)

Result: Frog auditory

respond with much hi

to sounds that resemble frog

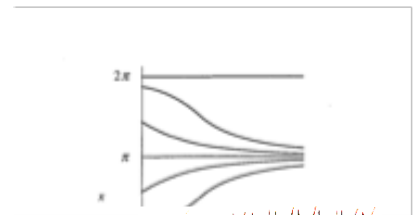
calls than to white noise



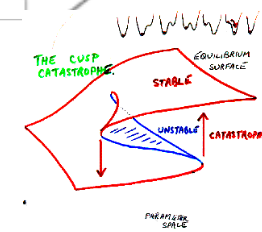
Dynamical systems!

Summary of dynamical systems:

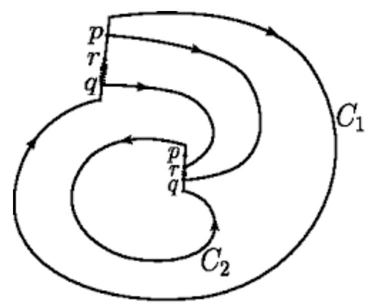
- 1D: No cycles, just fixpoints (or growth)



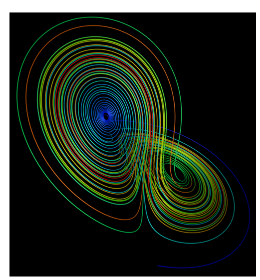
- Bifurcations: change in parameters qualitatively changes behavior



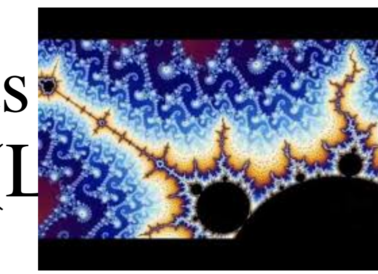
- 2D: cycles happen (oscillators) Poincare'-Bendixson



- 3D: chaos (Lorenz)

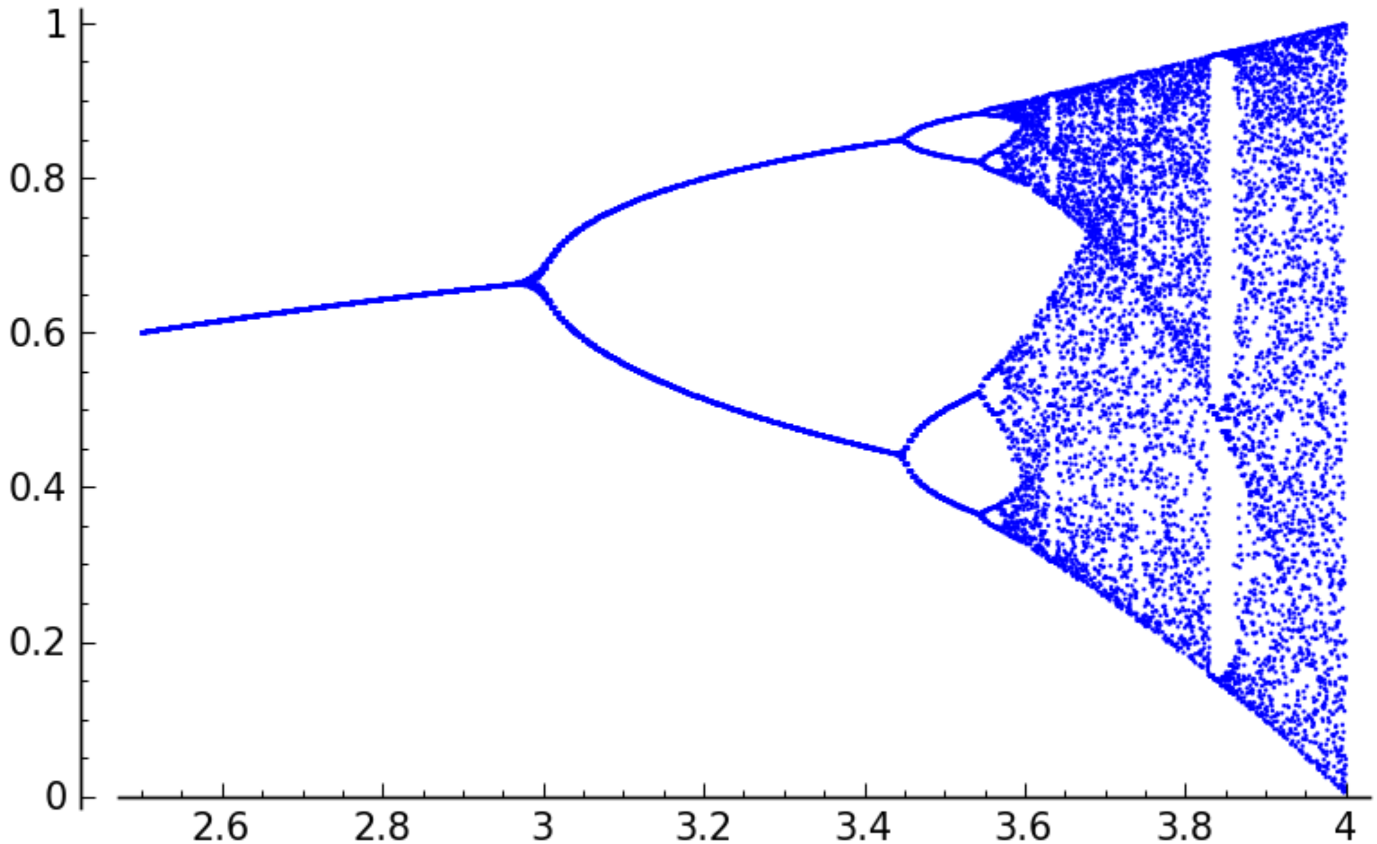


What is Chaos?

- Exponentially small perturbations in parameters and initial conditions lead to qualitatively different behaviors
- A seemingly periodic behavior repeats  cept that the system never exactly cycles (L
- An attractor is strange (fractal-like)
- In discrete time: there cycles of all

kinds of periods (but a cycle of period three is enough….)

$x_{t+1} = a\, x_t\, (1 - x_t)$: already chaos…



Against chaos: Properties you want your dynamical system to have

- Linear: can solve closed form: **eAt**
- 2D: (P-B to the rescue)
- Conservative systems: conserve "energy"
- Reversible systems: they can be "run backwards"
- Systems that have a Lyapunov function (progress toward convergence)

The fundamental theorem of dynamical systems (Conley 1984)

- Redefine cycle in discrete time systems
- Infinitesimal jumps between steps are allowed (think round-off error).
- Then the (compact) domain is decomposed into the “chain connected components” and the “transient part”; in the latter a Lyapunov function is at work…

“if you squint a little, chaos goes away”

- Next: dynamical systems for modeling parts of the Brain
- Continuous and discrete
- A few representative examples
- Avoiding chaos

The classic

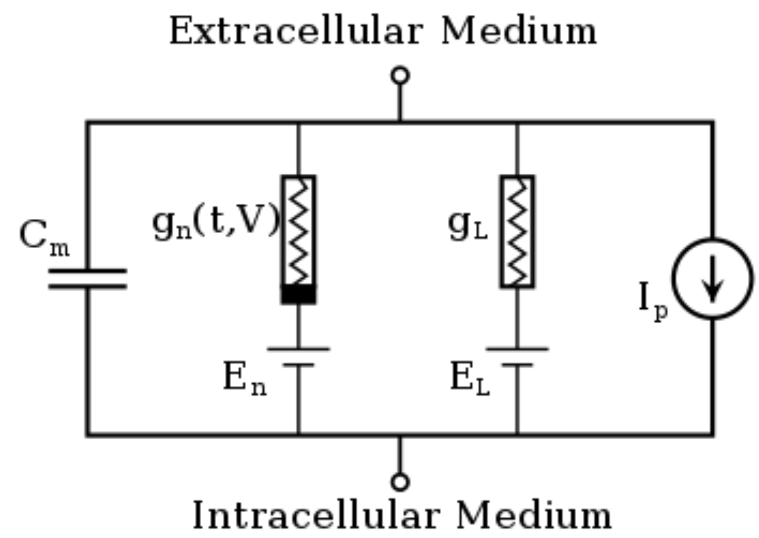
- The Hodgkin-Huxley oscillator

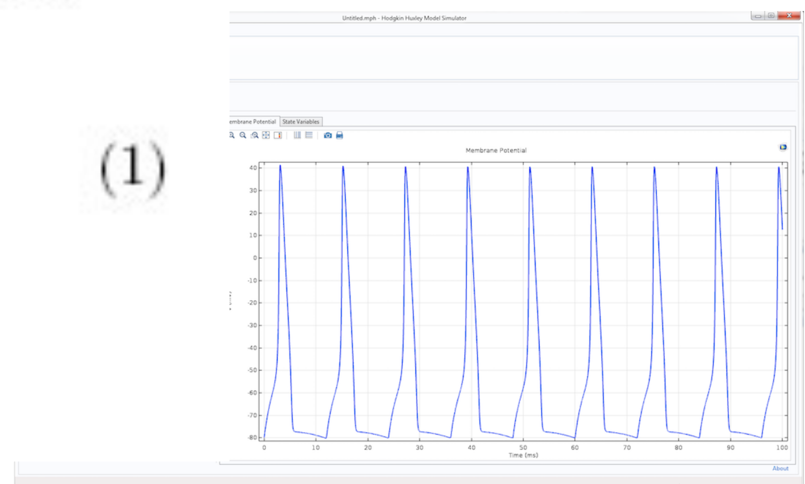
$$\frac{dv}{dt} = \frac{1}{C_m}[I - g_{Na}m^3h(v - E_{Na}) - g_k n^4(v - E_K) \\ -g_L(v - E_L)]$$

$$\frac{dm}{dt} = \alpha_m(v)(1 - m) - \beta_m(v)m \qquad (1)$$

$$\frac{dm}{dt} = \alpha_m(v)(1 - m) - \beta_m(v)m$$

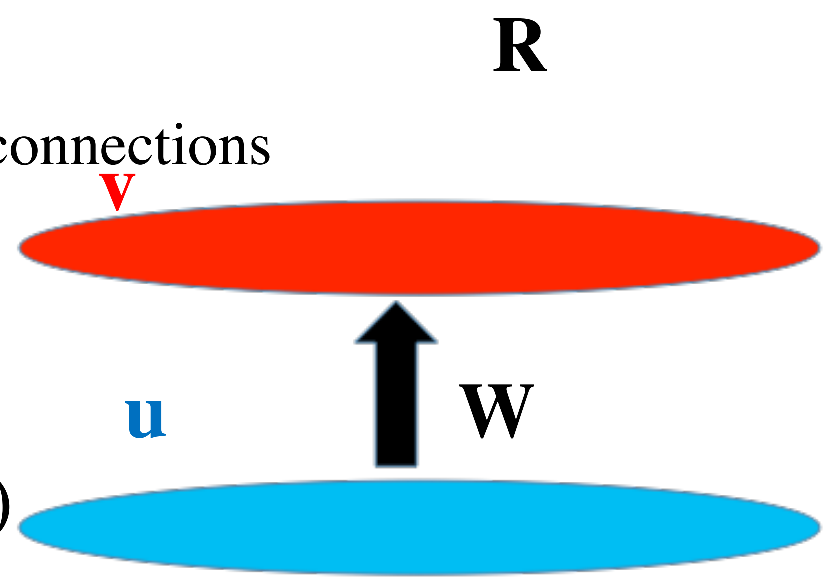
$$\frac{dh}{dt} = \alpha_h(v)(1 - h) - \beta_h(v)h$$





The Feedforward + Recurrent network

- Two populations of neurons
- Feedforward and recurrent synaptic connections
- **u**, **v**: vectors of firing rates
- **W**: matrix of synaptic weights



$$\tau \cdot dv/dt = -v + F(W \cdot u + R \cdot v)$$

- Interesting case: Inhibitory and excitatory neurons in **R**

Hopfield networks

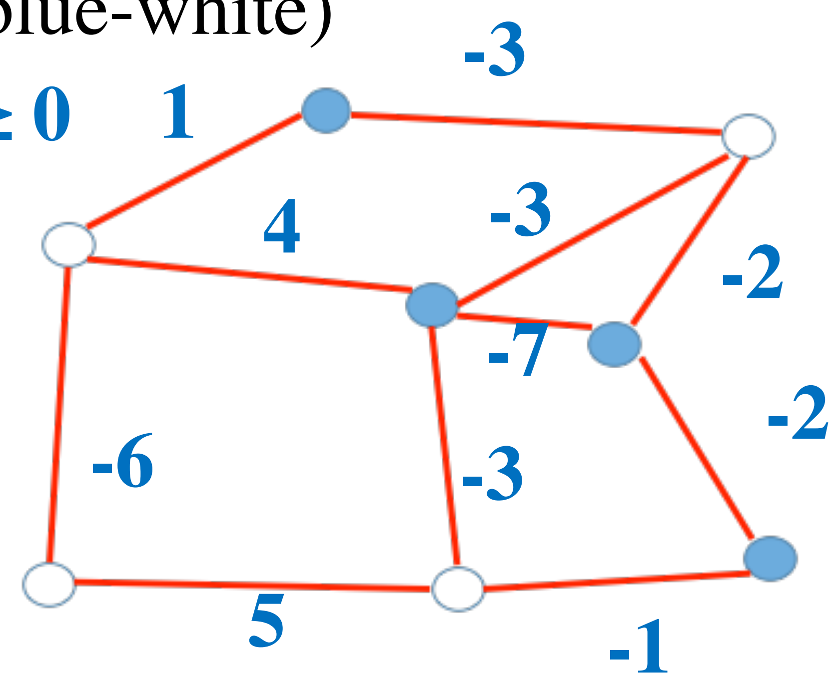
Nodes have two values: +1, -1 (blue-white)

Node i is happy if $\Sigma_j v_i v_j w_{ij} \geq 0$

$\geq \Theta_i$

Algorithm/dynamical system:

while there is
an unhappy node
flip it



Questions? Thoughts? Feedback?

Today (and next Wednesday):

- Further dynamical systems in the brain
- Hopfield networks (and Boltzmann machines)
- Feedforward + recurrent networks in detail
- FFR networks with random synapses
- E-I balance
- Animal motion and dynamical systems

A discrete-time dynamical system: The Hopfield network

Nodes have two values: +1, -1 (blue-white)

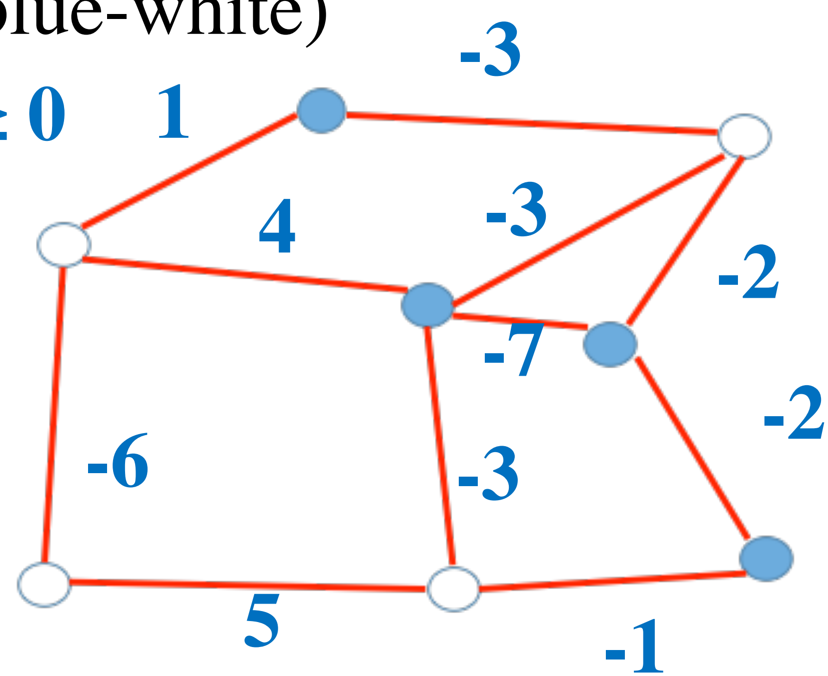
Node i is happy if **$\Sigma_j v_i v_j w_{ij} \geq 0$**

$\geq$

Θ_i

Algorithm/dynamical system:

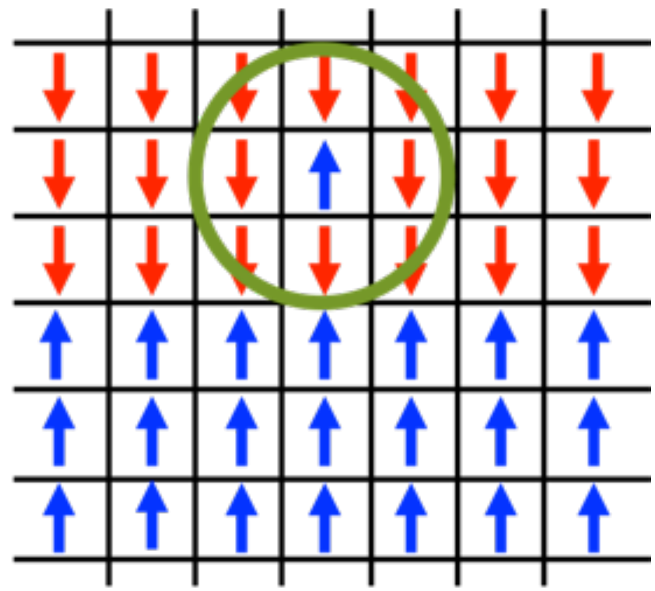
while there is
an unhappy node
flip it



cf: the Ising model: ***all edges are +1***

Algorithm/dynamical system:

repeat
pick a node at random
if unhappy, flip
if k-happy, flip with
probability e-k/T
until ferromagnetic (= all same)



A discrete-time system: Hopfield net (cont)

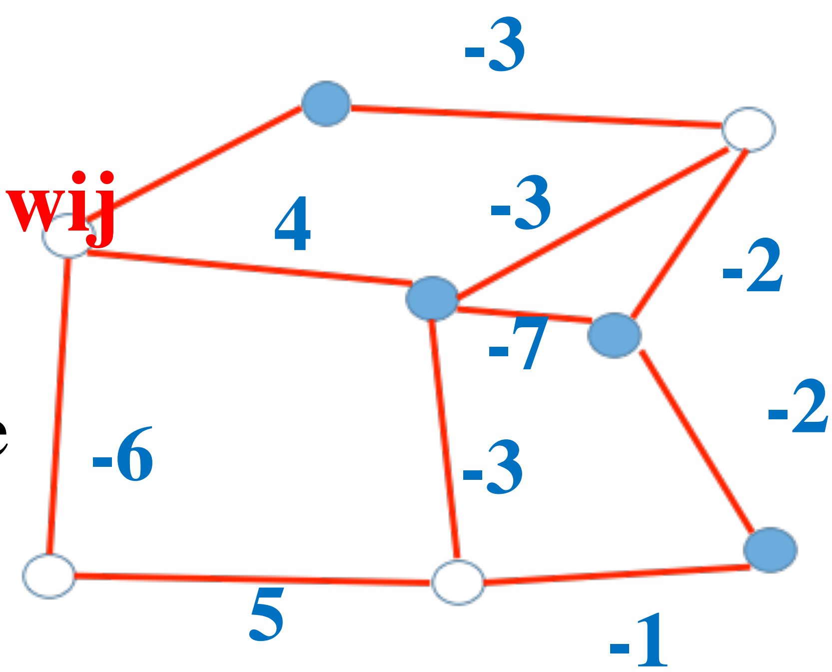
Theorem [Hopfield 1982]: Dynamical system converges

Proof: **Lyapunov** function

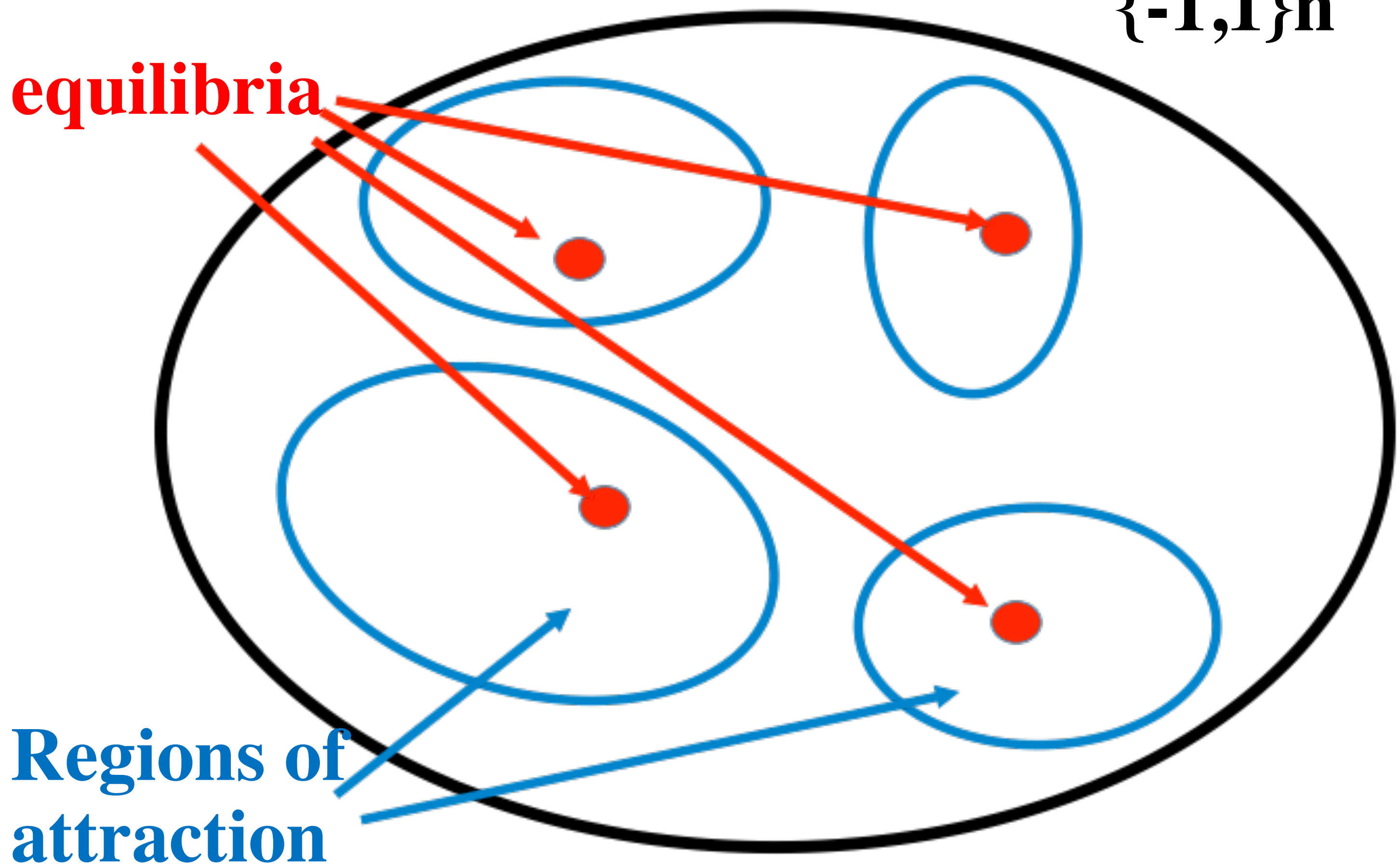
"total happiness" **= Σi,j vi vj wij**

(= - energy)

At each step increases by 2 × the unhappiness of the flipped node



Goal: Pattern completion



How do you train a Hopfield net so that it pattern completes?

- Given a set of desired memories M1, … , Mm ϵ {-1, +1}n
- Set the **weight** of edge a-b to **Σk Mka × Mkb**
- That is, for every memory k we bias the weight in the direction "memory k wants the weight to be".
- Question: does in work?

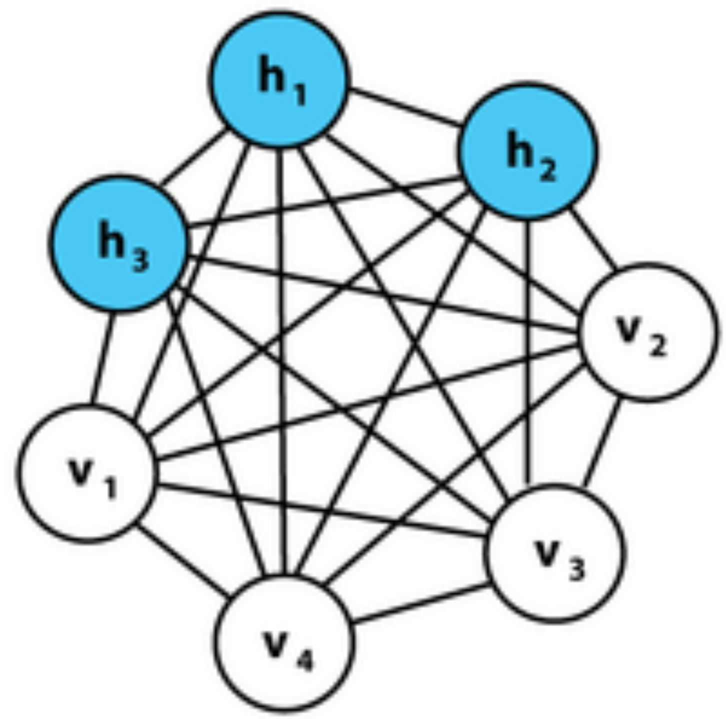
Theorem: with n nodes, 0.138n random patterns can be stored with probability of erroneous retrieval < 0.4%

Do Hopfield nets work?

- Spurious memories: if **M** is stored, **-M** is also retrieved
- Also, if **M, M', M''** are stored, so are **± M ± M' ± M''**
- Finally, if you store **M** K times, it will be retrieved L >> K times more often than other memories
- Fun: [Hopfield and Tank 1984] show that Hopfield nets solve the traveling salesman problem with 10 cities

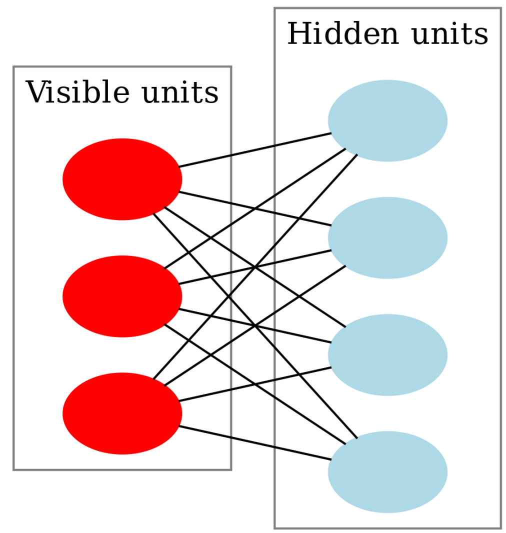
Boltzmann Machines

- Some of the nodes are **hidden**
- The visible nodes receive the training data by clamping
- After each training data vector is input, the whole network is left free to run until it reaches "thermal equilibrium"



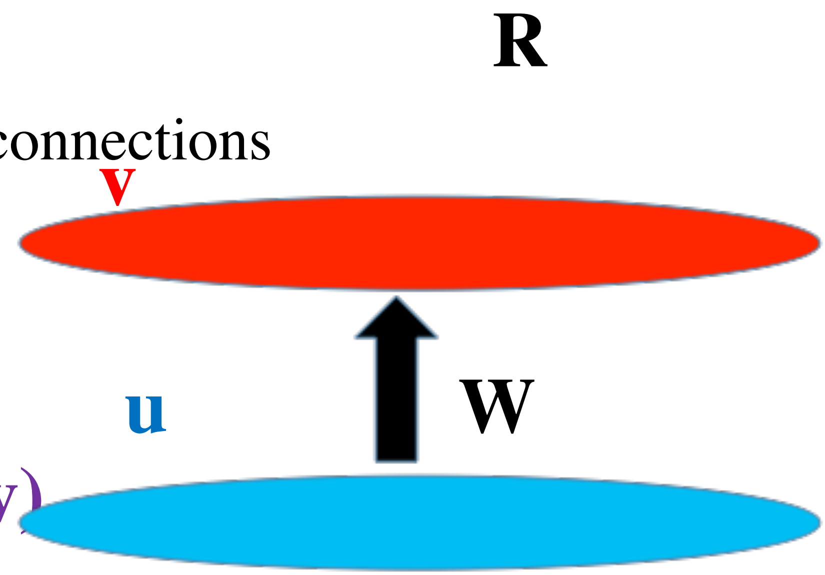
Boltzmann Machines (cont.)

- With some engineering, learning can happen
- Restricted Boltzmann machine: graph is bipartite [Hinton 2005]
- Can be stacked to deep net…



Back to modeling brain networks: The Feedforward + Recurrent (FFR) network

- Two populations of neurons
- Feedforward and recurrent synaptic connections
- **u**, **v**: vectors of firing rates
- **W, R**: matrices of synaptic weights



$$\tau \cdot d\mathbf{v}/dt = -\mathbf{v} + F(\mathbf{W}\cdot\mathbf{u} + \mathbf{R}\cdot\mathbf{v})$$

F is the response function of the red neurons

what if F is linear?

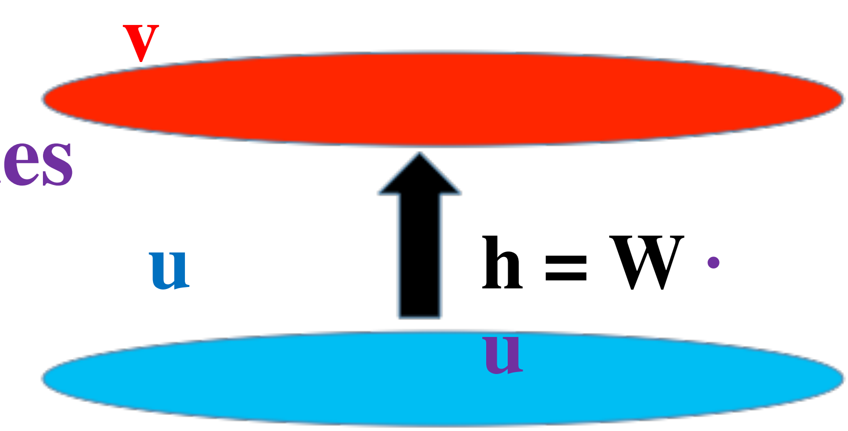
Linear FFR network

$\tau \cdot dv/dt = h + (M - I) \cdot v$

M has **positive** eigenvalues λk with **orthonormal** eigenvectors vk

$\lambda 1 \geq \lambda 2 \geq \ldots$

symmetric synaptic matrix **M**



What is the solution?

Write $v(t) = \sum_k c_k(t)\, e_k$ and write ODEs for the $c_k(t)$'s :

$\tau \cdot dc_k/dt = -(1 - \lambda_k) \cdot c_k(t) + e_k \cdot h$ Solve (assume $\lambda_k \neq 1$)

$$c_k(t) = \exp(-t\,(1 - \lambda_k)/\tau) \cdot (e_k \cdot (v(0) - h/(1 - \lambda_k))) + e_k \cdot h/(1 - \lambda_k)$$

First term: $\lambda_k > 1$ ➔ **exponential growth** (hence

Suppose that **$\lambda_2 << \lambda_1 \approx 1$**

We can ignore all other terms, and we have the asymptotic solution

$v^* \approx (e_1 \cdot h) \cdot e_1 / (1 - \lambda_1)$

So, this circuit takes the feedforward input h and projects it on **e_1**, amplifying it by a large number

(If two eigenvalues of M are close to one, it projects on a plane…)

Finally, suppose $\lambda 1 = 1$

Recall : $\tau \cdot dc1/dt = -(1 - \lambda 1) \cdot c1(t) \; + \quad e1 \cdot h$

So, the circuit ***integrates*** the feedforward input's projection on **e1**

NB: Integration means memory. (Why?)

A system like this seems to be at work in the brain stem of verterbrates, remembering eye position

Nonlinear FFR networks

$\boldsymbol{\tau \cdot dv/dt = -v + [h + M \cdot v - T]_+}$

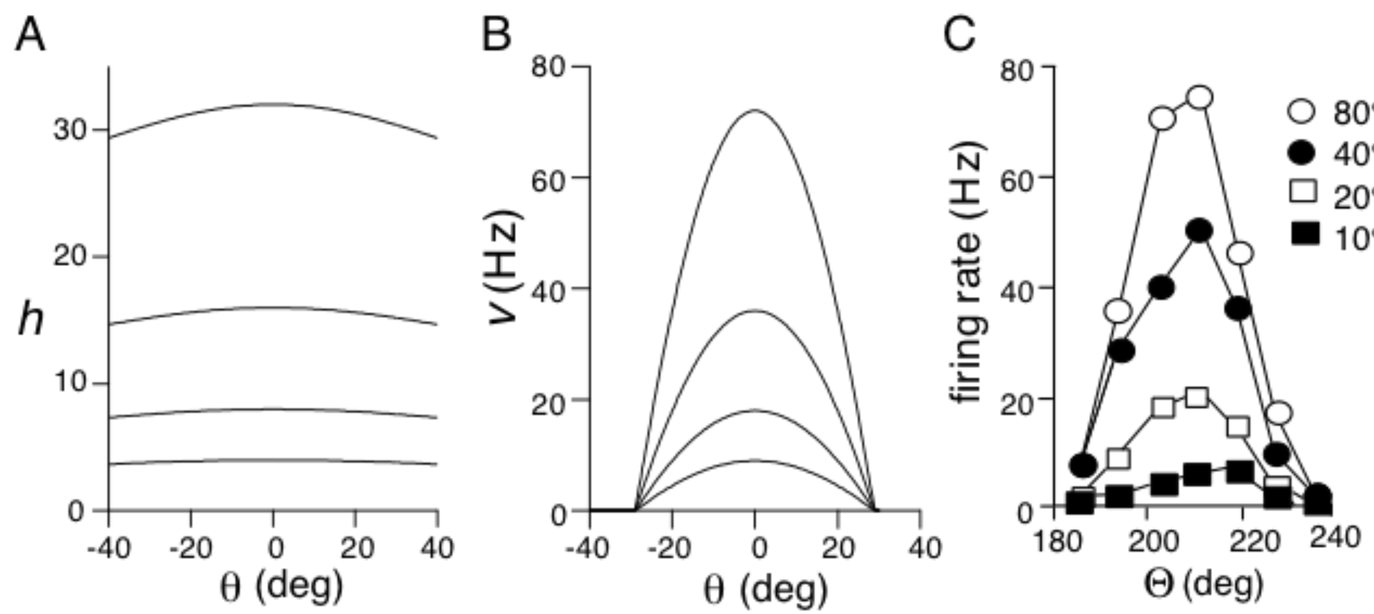
Subtract a vector of thresholds **T** and set to zero if negative

Can model simple and complex cells

Recurrent input from V1-V2 strong, must be modeled

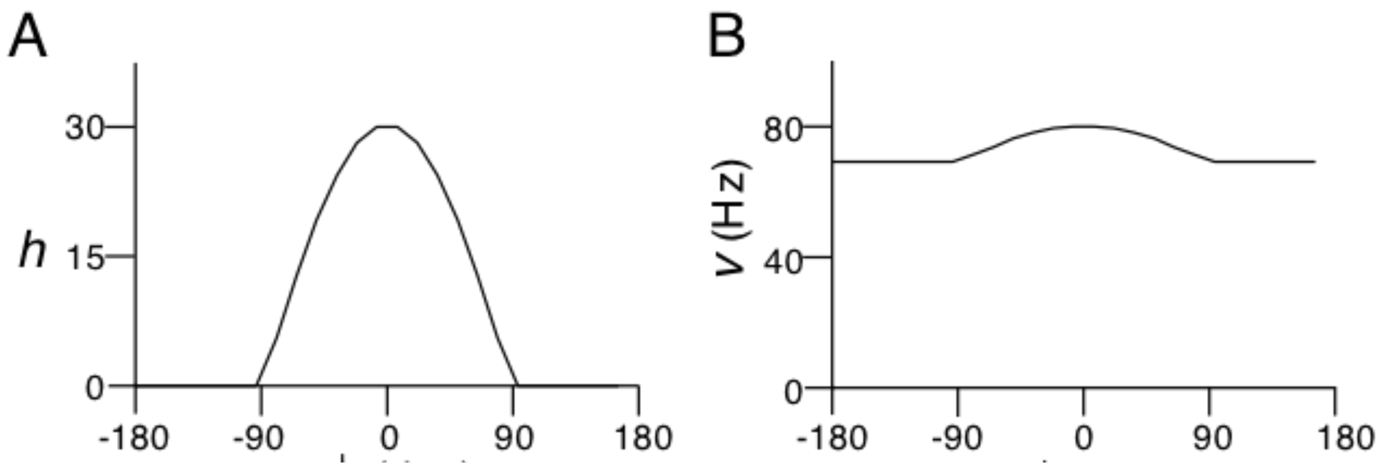
FFR network with array of cells indexed by angle θ in -40^o to 40^o, with synaptic weight $M(\theta_1,\theta_2) \sim \cos(\theta_1-\theta_2)$

Modeling simple cells in V1



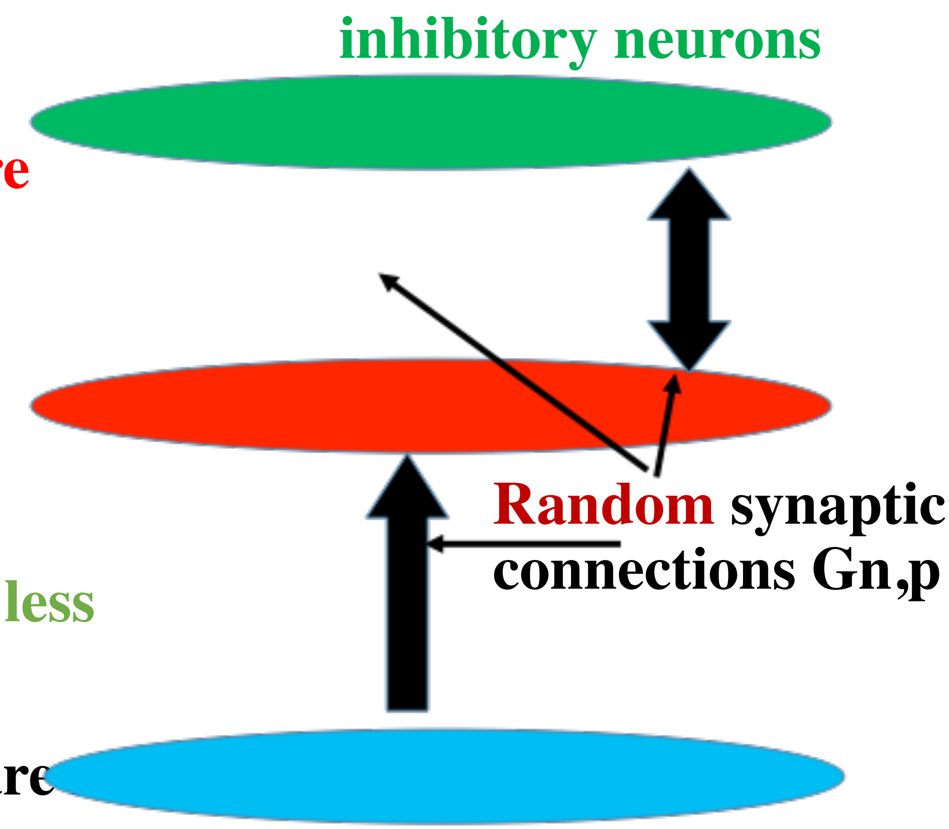
Modeling complex cells

- Again, an array of neurons indexed by θ
- Input $h(\theta)$ is a simple cell with preferred angle θ (here zero). It feeds the θ-neuron in the array
- Constant recurrent synaptic weights M

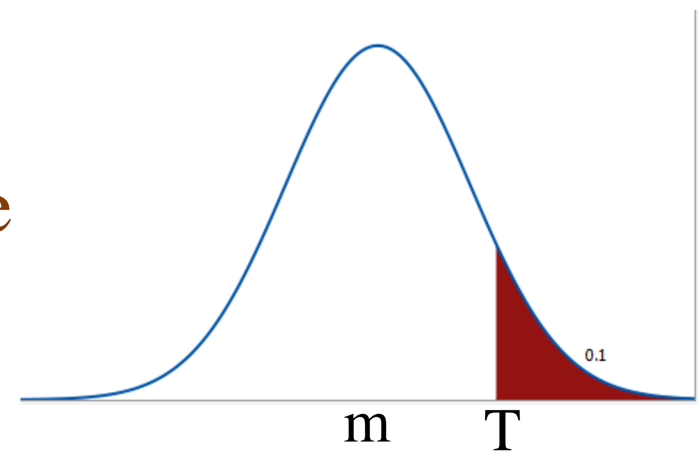


Another FFR network: Excitation-Inhibition balance

- **The blue cells spike**
- **Many red cells receive input and fire**
- **The green cells receive input**
- **They fire, and inhibit the red**
- **Maybe too much**
- **Now they receive less input from the red, and they inhibit less**
- **All these inputs are random**
- **By the law of large numbers, they are**



E – I balance



Notation: GT(T, m, var) is the Gaussian tail with parameters m and var above threshold T

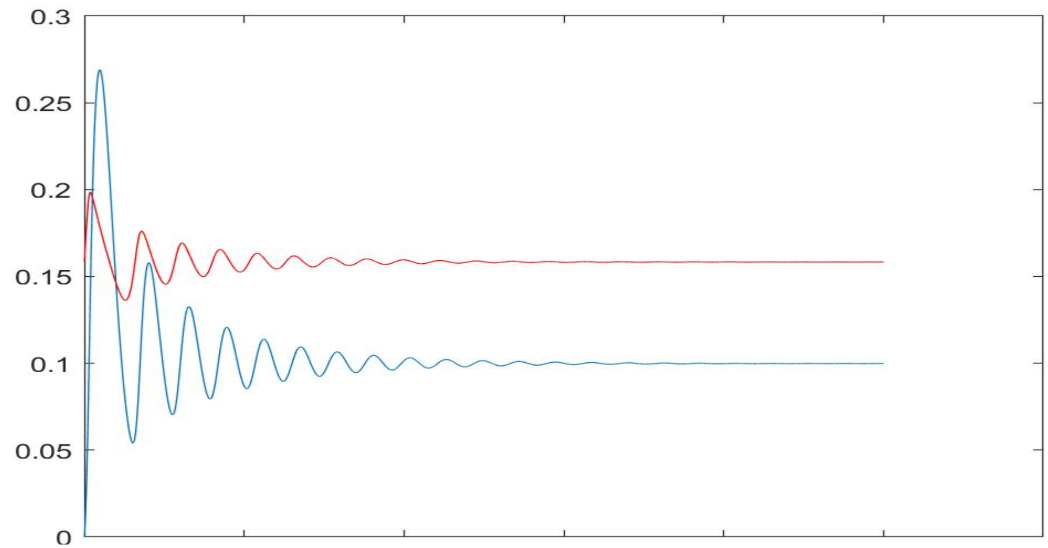
Nonlinear ODE

$$\tau E\, \dot{E} = GT(TE,\ np(E - I),\ np(1-p)(E + I)) - E$$

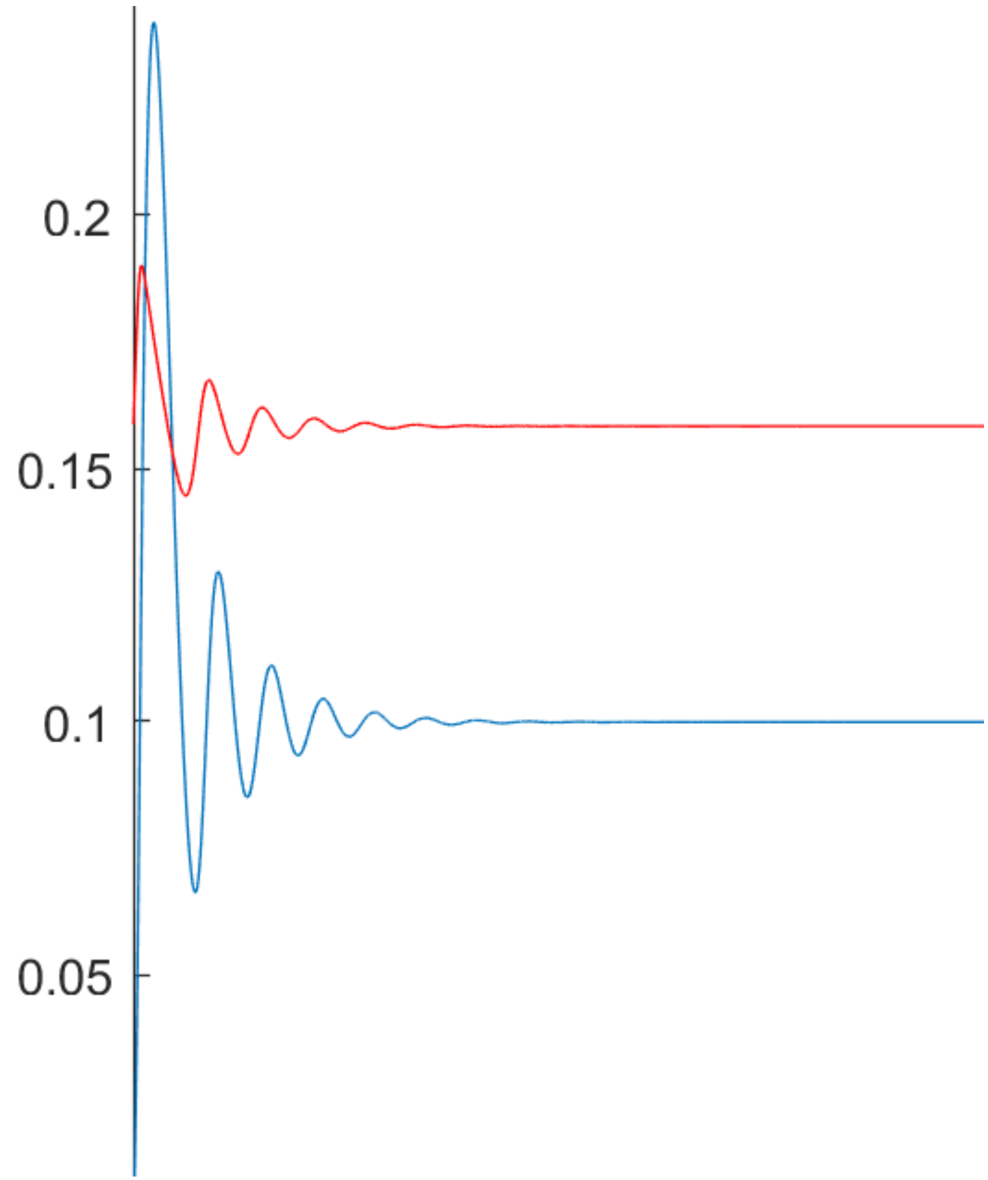
$$\tau I\, \dot{I} = GT(TI,\ npI,\ np(1-p)I) - I$$

E – I balance (cont.)

If τE is sufficiently larger than τI, an E – I balance will be reached after a few up and down oscillations



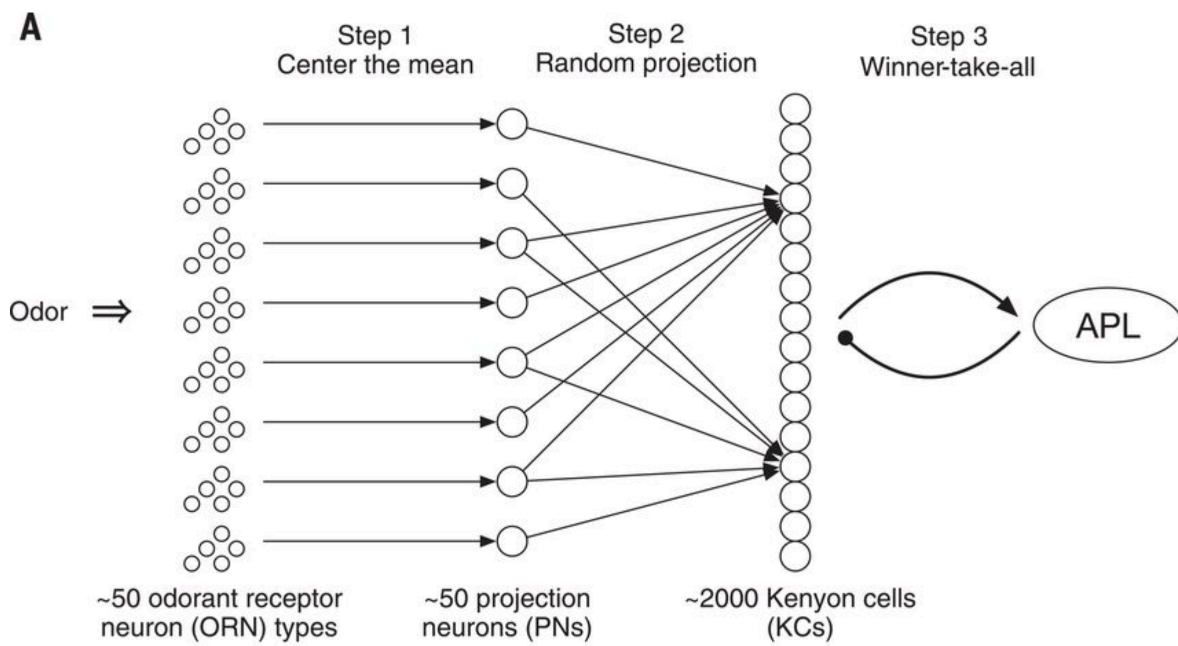
E – I balance (cont.)



btw: Random synaptic connections?

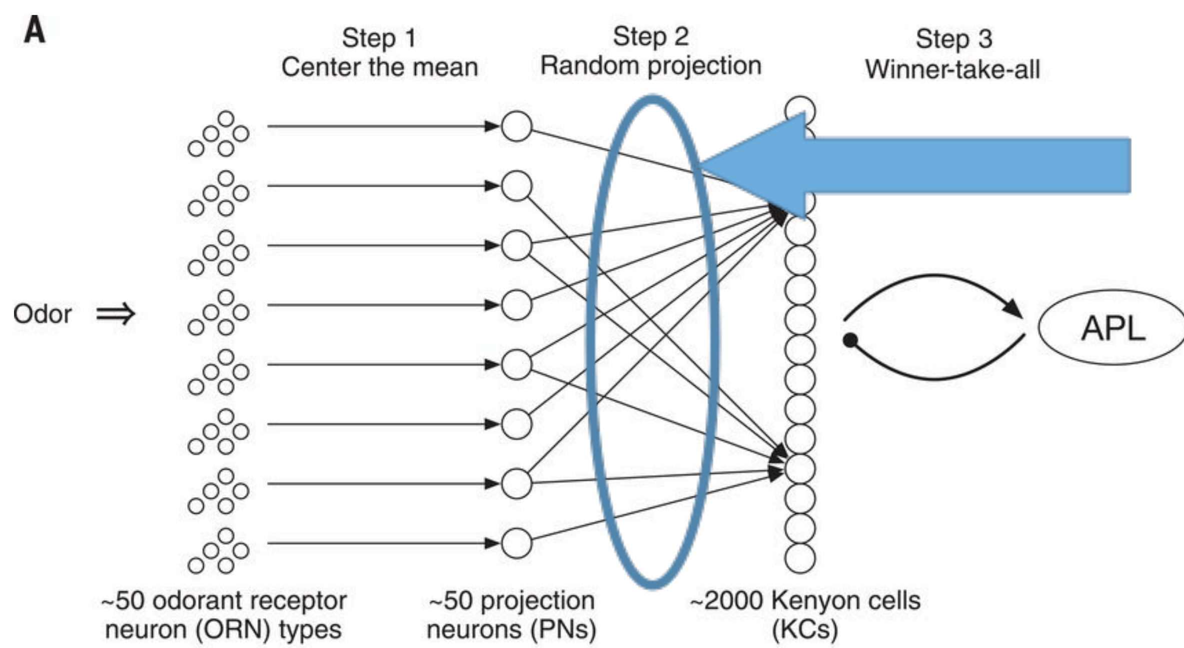
- We know (from reading Guzman et al.) that it's not **Gn,p** -- at least hippocampus CA3 is not, neither is V2 [Song et al. 2005]
- But, intuitively, it must be close: **Gn,p++**
- **Gn,p** assumption: refreshing analytical empowerment through the GT(T,m,var) approximation

RP&C: How fruit flies remember smells



random projection followed by ***cap (RP&C):*** 100 winners (out of 2000) take all

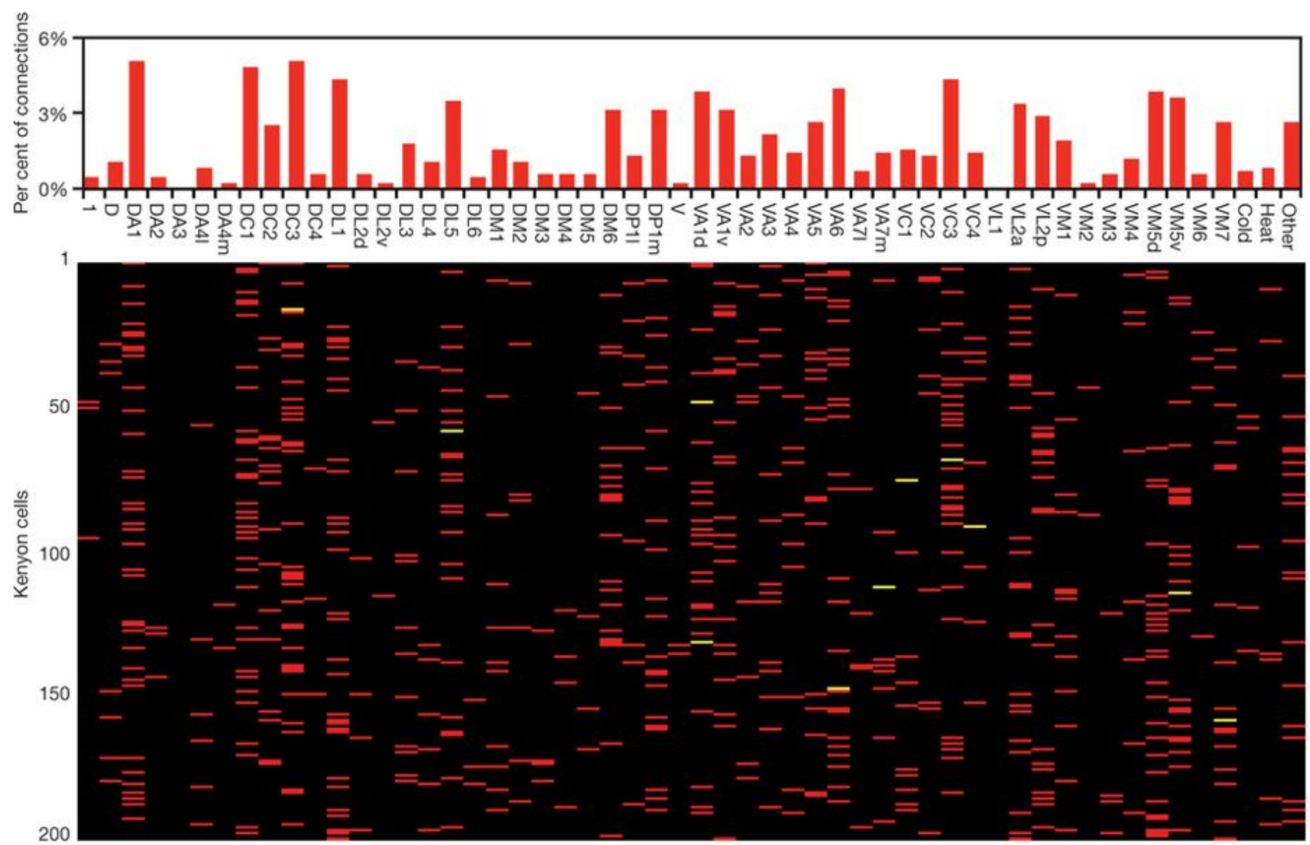
How fruit flies remember smells

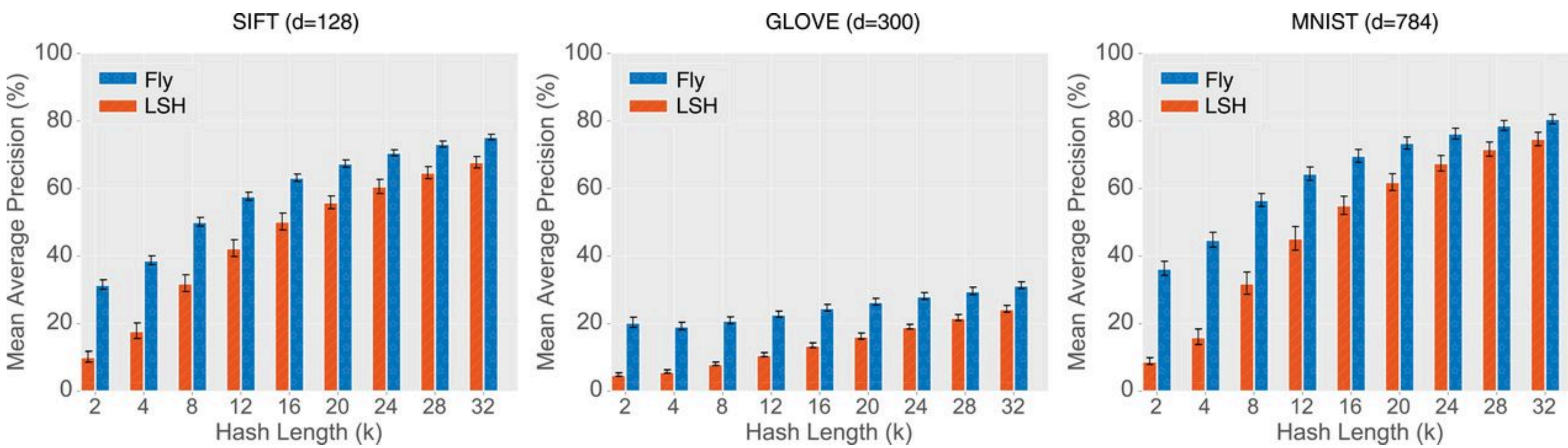


Q: but wait a minute! Is this truly a random bipartite graph?

A: ***Random convergence of olfactory input in the Drosophila mushroom body*** by S. Caron, V. Ruta, L. Abbott, R. Axel,2013

Bottom line: looks like a random bipartite graph, except that the degree distribution of the LHS is not uniform

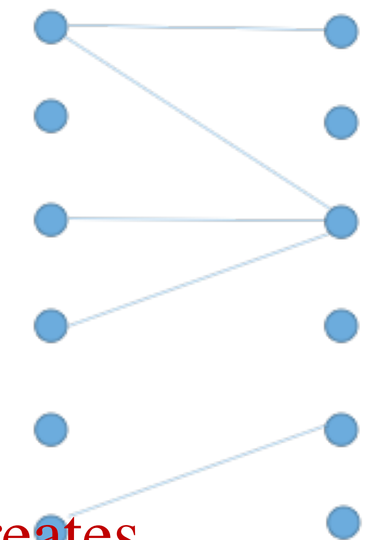




Surprise! The fly's algorithm (RP&C) preserves similarity in standard datasets to a degree that is competitive with the best similarity preserving algorithms (AKA locality sensitive hashing, LSH)* [Dasgupta et al. Nov 2017]

*Alex Andoni 2018: but not with the latest versions…

The underlying mathematical reason?

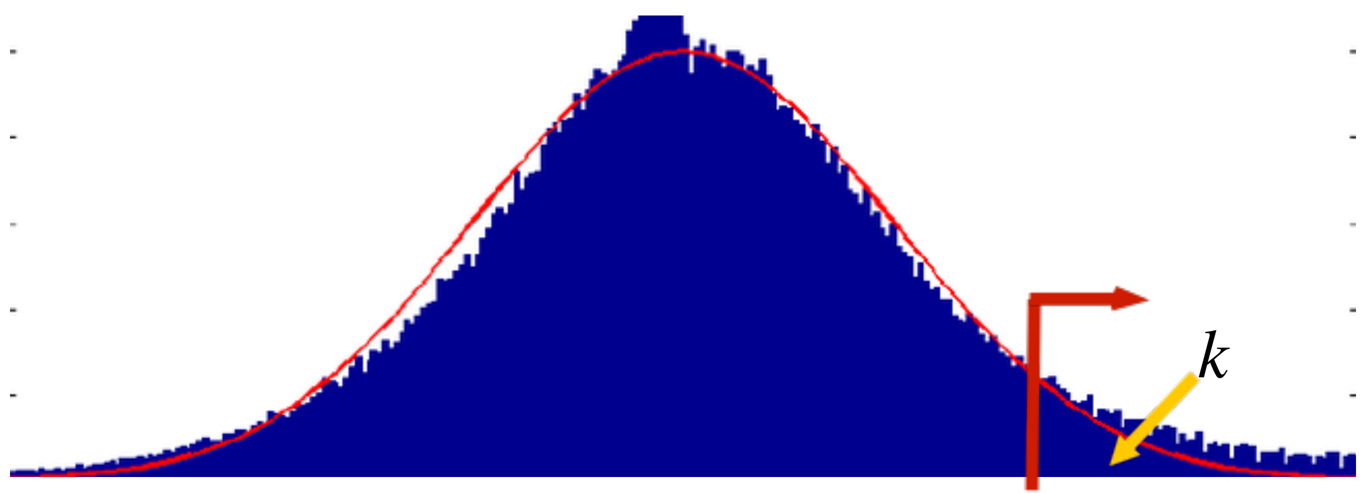


Btw: why am I taking the two sides to be *symmetric* (n nodes, k-sparse sets)?

1. It is helpful in math

2. It doesn't matter, anything on the LHS creates a "Bernoulli shower" (the GT(T,m,var) function) on the RHS

RP&C

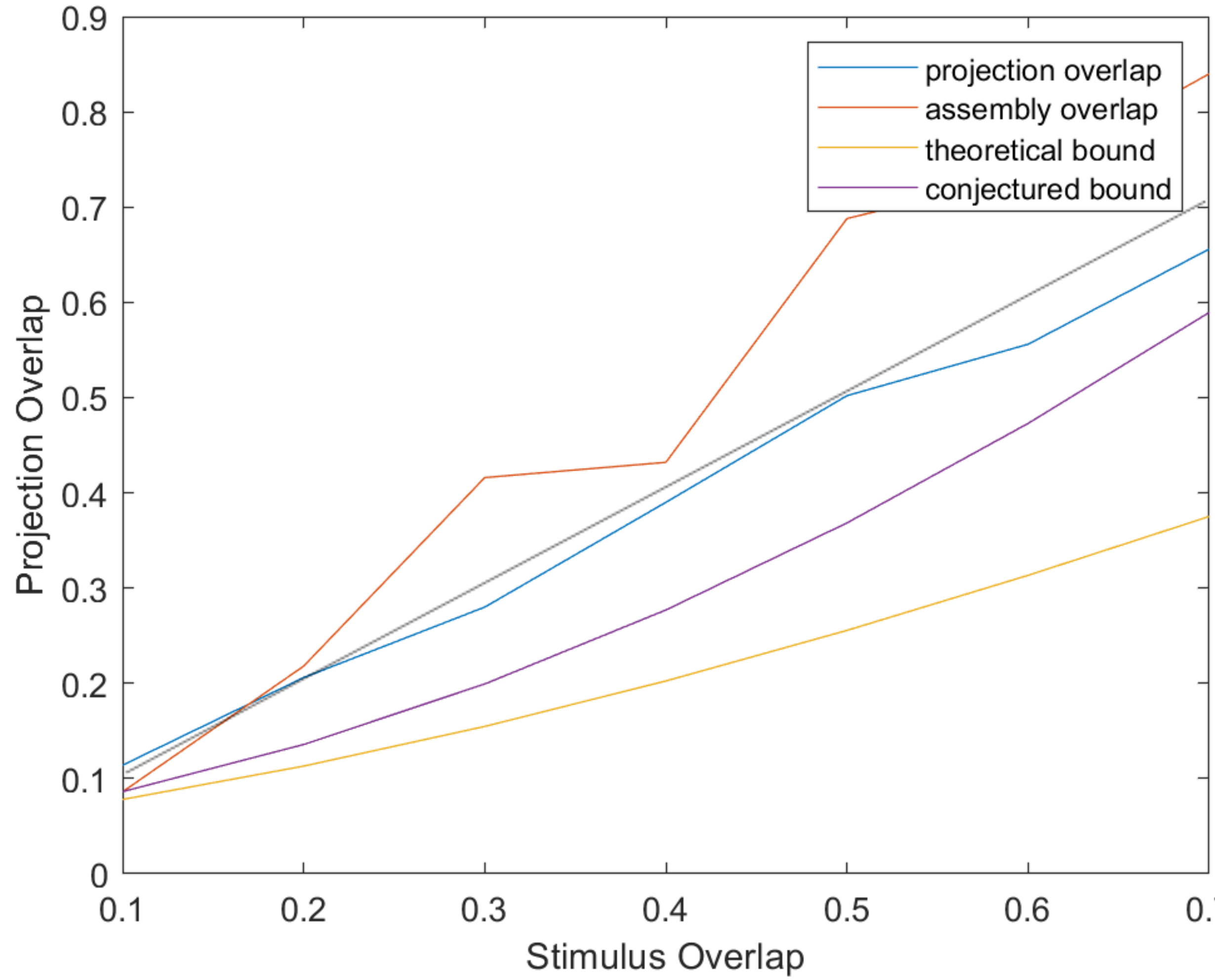


The underlying mathematical reason:

- **Theorem** [P., Vempala, 2018
  The intersection of **cap(A)** ar
  probability, at least

Conjecture:
no denominator

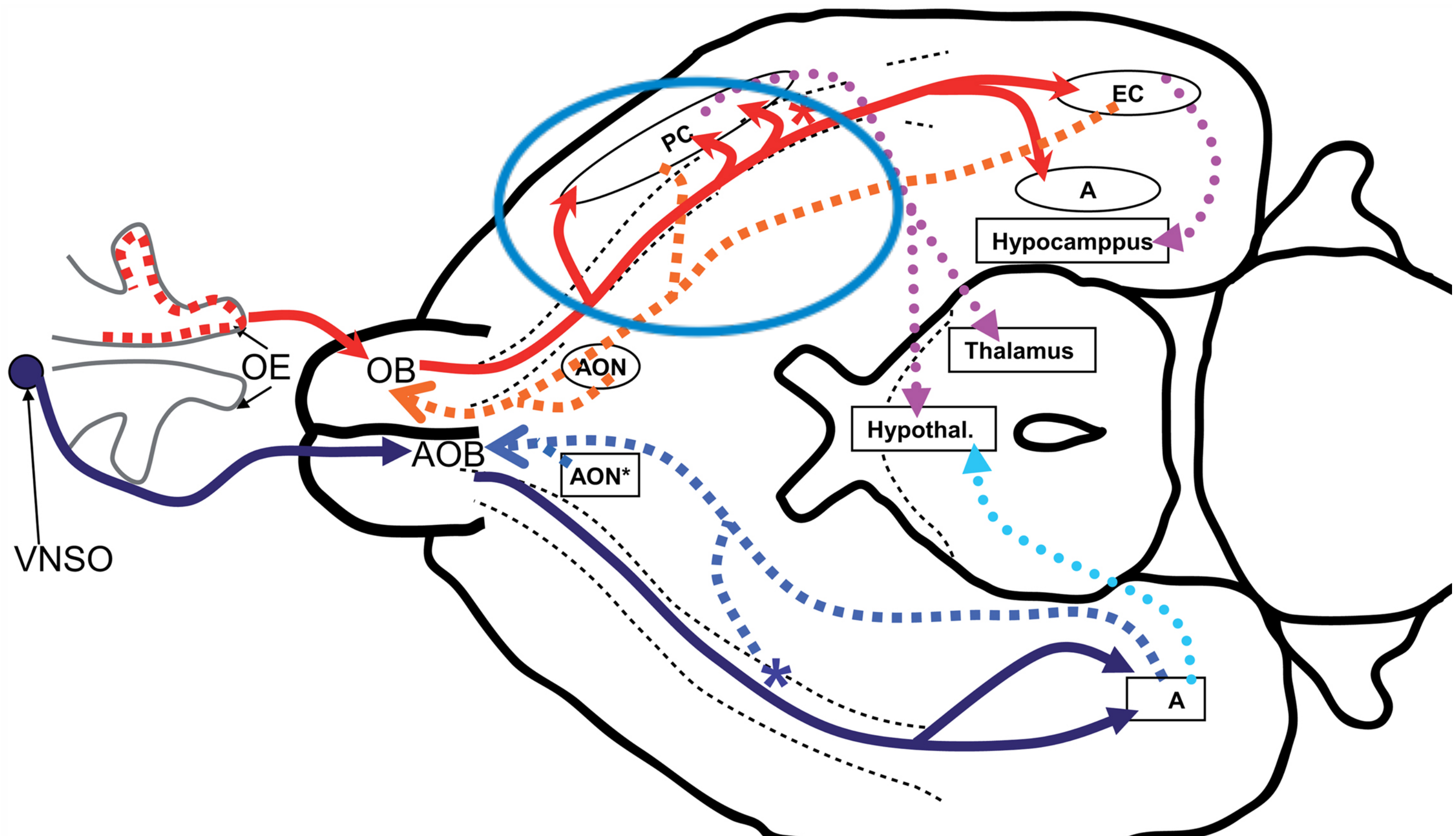
The underlying mathematical reason: compare with simulations



So much for the fruit fly…

- Q: Does something homologous happen in mammals?

Yes!



K. Franks, M. Russo, S. Sosulki, A. Mulligan, S. Siegelbaum, R. Axel
"Recurrent Circuitry Dynamically Shapes the Activation of Piriform Cortex." Neuron October 2011

From the *Discussion* section of Franks *et al*.

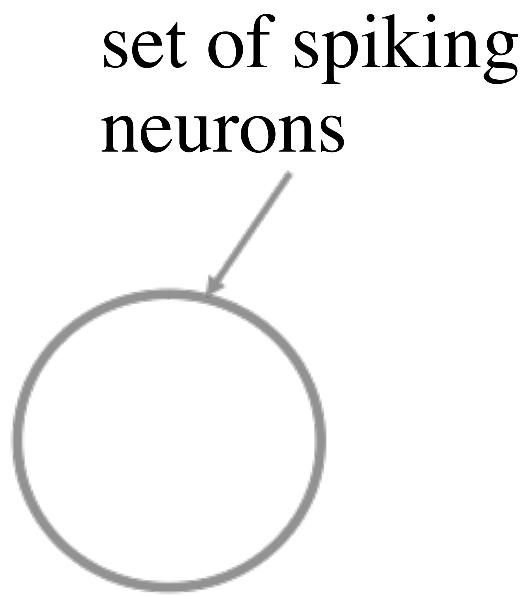
An odorant may [cause] a small subset of [PC] neurons [to fire].

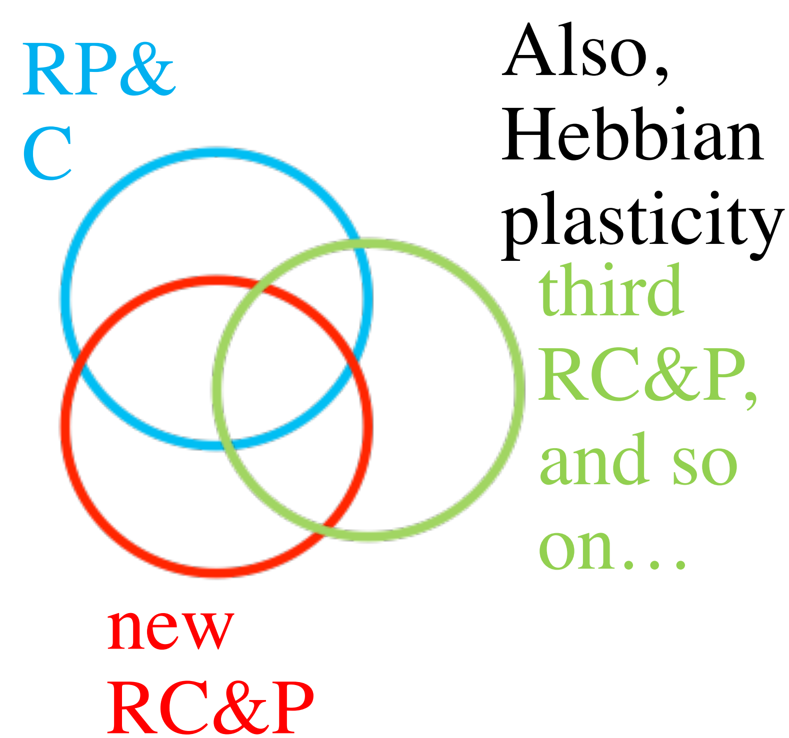
Inhibition triggered by this activity will prevent further firing

This small fraction of ... cells would then generate sufficient recurrent excitation to recruit a larger population of neurons.

In the extreme, some cells could receive enough recurrent input to fire … without receiving [initial] input…

In pictures…

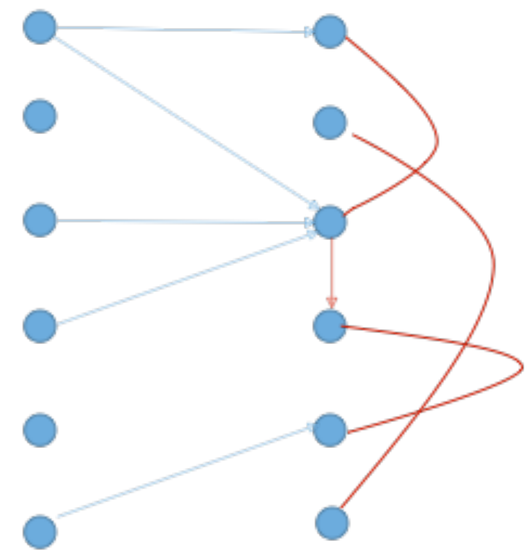

set of spiking
neurons


RP&
C
new
RC&P
Also,
Hebbian
plasticity
third
RC&P,
and so
on…

Does this process converge?

And does it preserve similarity?

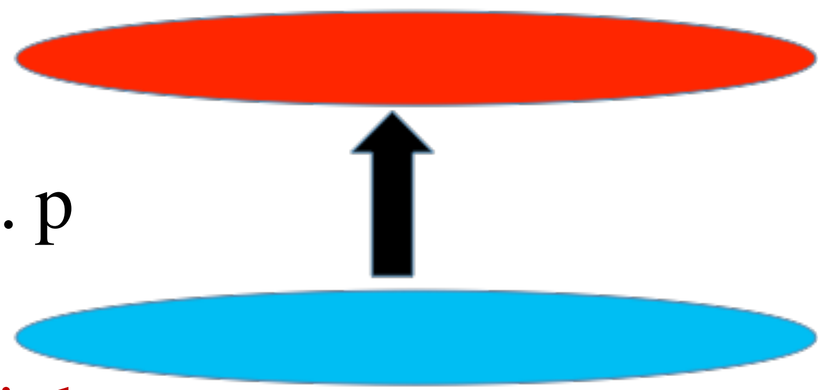
Upgrade the model: the GNP model



- Fruit fly, plus:
- Recurrent synapses
- All random connections with prob. p
- Discrete time
- Hebbian plasticity: i-j synaptic weight increased by β – or multiplied by (1 + β)
- A fixed number of brain areas, each with n excitatory neurons and recurrent connectivity, plus

Upgrade the model: the GNP model

- Fruit fly, plus:
- Recurrent synapses
- All random connections with prob. p
- Discrete time
- Hebbian plasticity: i-j synaptic weight increased by β – or multiplied by $(1 + \beta)$
- A fixed number of brain areas, each with n excitatory neurons and recurrent connectivity, plus



Main parameters, intended values

- $n \sim 107$
- $k \sim 103 - 4$
- $p \sim 0.001$
- $\beta \sim 0.20$

So, can this model predict what happens in the piriform cortex?

Lineariz

“probability”
of activation

Input from stimulus

$$xi(t+1) = si + \Sigma i \rightarrow i \;\; xi(t)\, wij(t)$$

random synaptic
weights

(additive) plasticity

$$wij(t+1) = wij(t) + \beta\, xi(t)\, xj(t+1)$$

Linearized model: solution

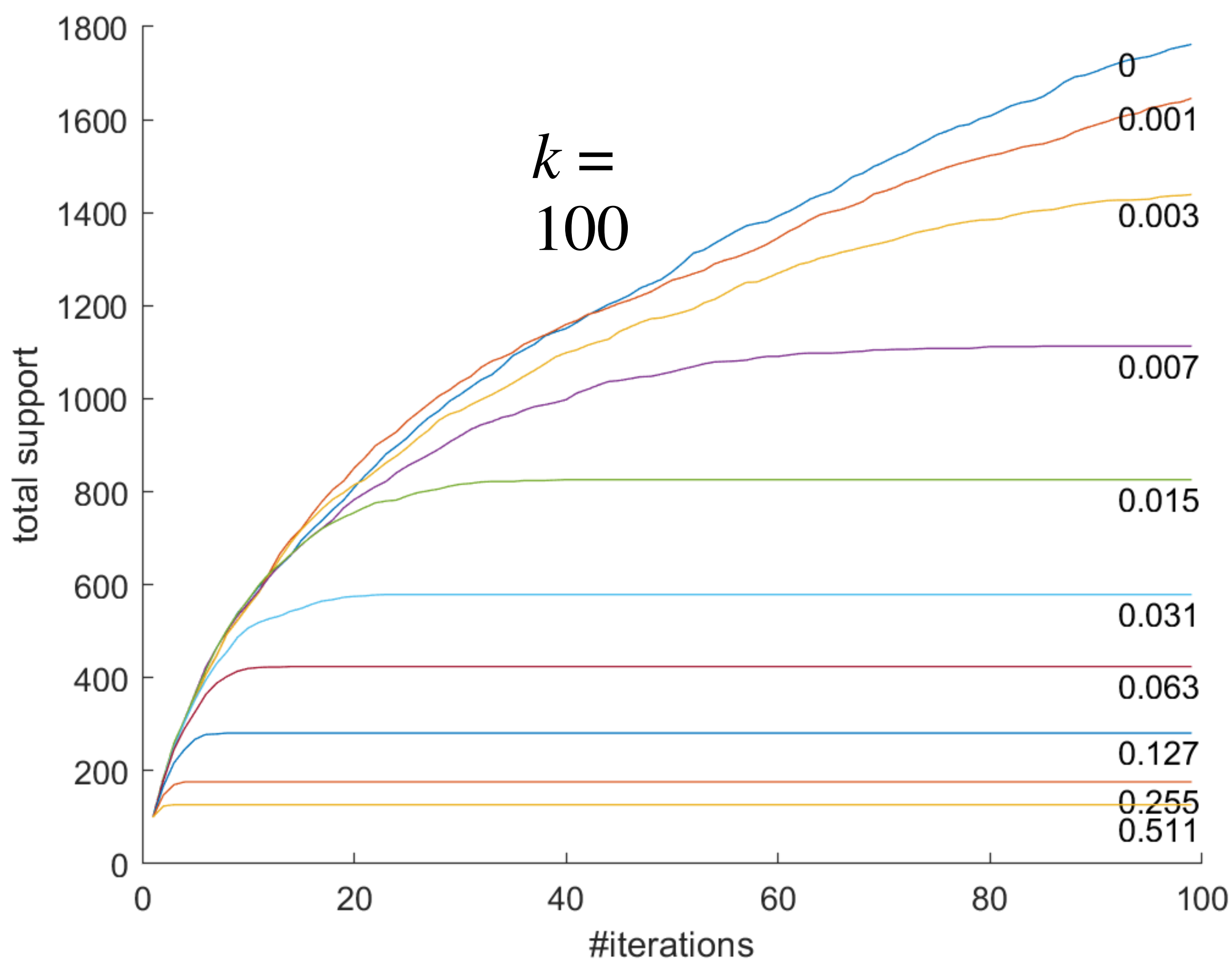
Theorem (P., Vempala, Maass, 2018]: The linearized dynamics converges geometrically and with high probability to

OK, how about the real, nonlinear system?

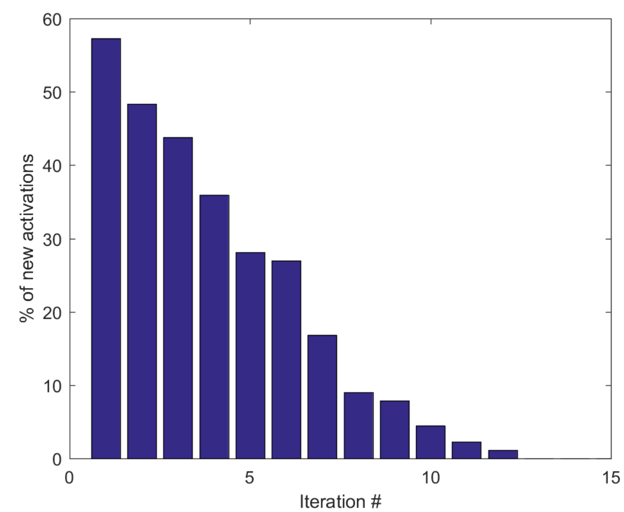
Theorem (P., Vempala 2016-18): The process converges exponentially fast, with high probability, and the *total number of cells involved* is at most:

- If $\beta \geq \beta^*$: $k + o(k)$
- If $0 < \beta < \beta^*$: $k \cdot \exp(0.17 \cdot \ln(n/k) / \beta)$

- NB: $\beta^* = (\sqrt{2} - 1)/(1 + \sqrt{pk/\ln n})$



$k = 100$
total support
#iterations
0
0.001
0.003
0.007
0.015
0.031
0.063
0.127
0.255
0.511



% of new activations
Iteration #

The result of such projection: an ***Assembly***

- Set of $\approx k$ neurons in a brain area whose firing (in a pattern) is tantamount to our thinking of a particular memory, concept, name, word, episode, etc.
- [Hebb 1949, Harris 2003, 2005; Buzsaki 2008, 2010]
- Also, simulations of a far more biologically accurate STDP model [Pokorny et al 2018, under submission]
- Presumably highly connected