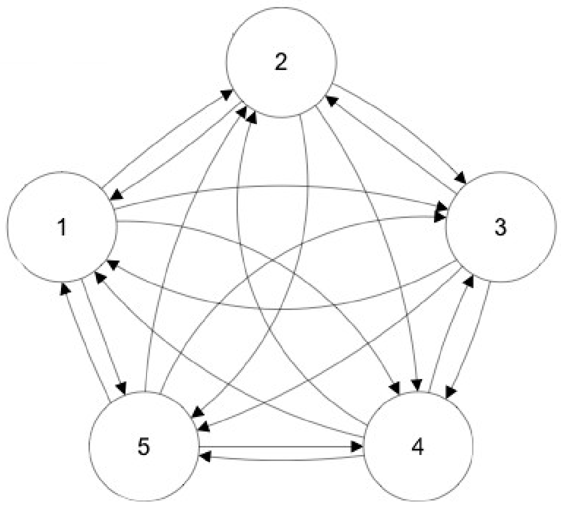
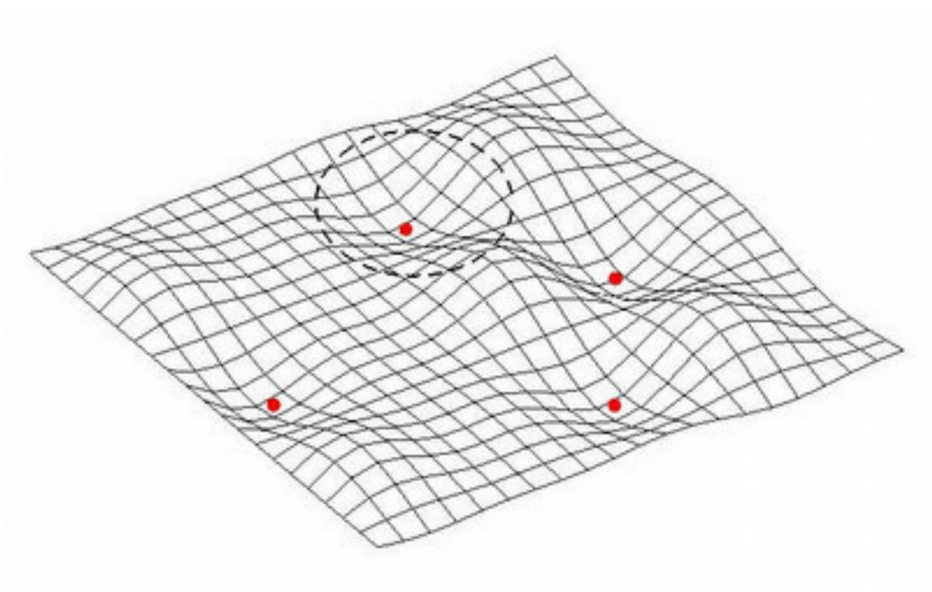
A purely local, distributed, simple learning scheme achieves near-optimal capacity in recurrent neural networks without explicit supervision

Alireza Alemi, Carlo Baldassi, Nicolas Brunel, Riccardo Zecchina

BDA 2015

Attractor networks

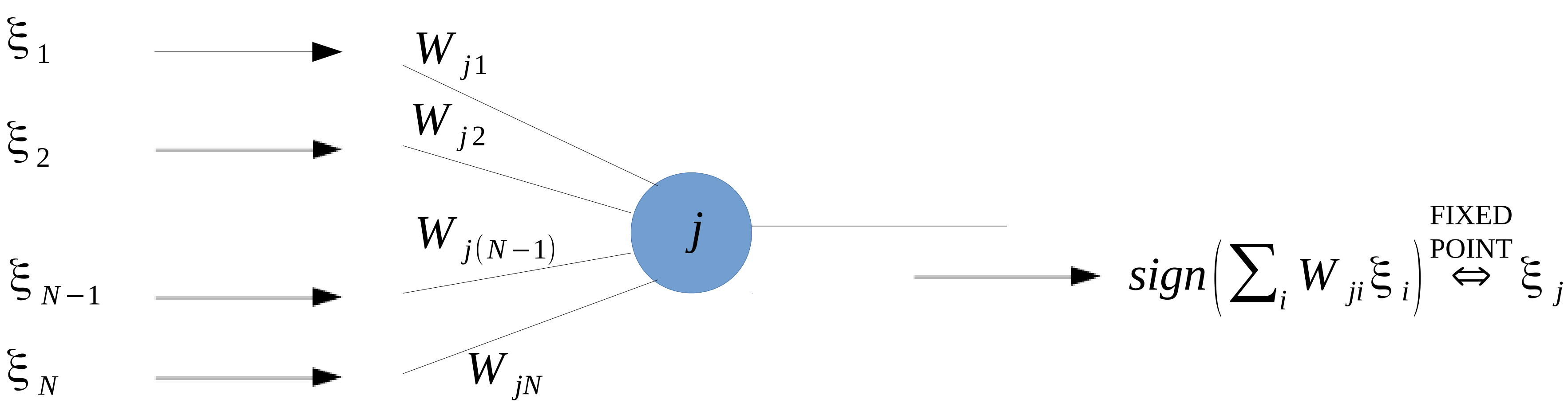




- Popular model for information storage in the brain (memorization – working memory, recognition, error-correction, ...)
- Recurrent neural networks
- Distributed model (each unit behaves independently, information is stored in the collective behaviour)
- Learning → patterns of activity are encoded as fixed points of the network dynamics
- Robustness → basins of attraction around the fixed points

Hopfield network

- First, most popular model (1984), with many later variants
- Binary ±1 units and patterns (perceptrons → $output_j = sign\left(\sum_i W_{ji} \xi_i\right)$)
- Hebbian learning ("fire together → wire together, out-of-sync → fail to link")
  - Rule: $W_{ij} = \frac{1}{M} \sum_a \xi_j^a \xi_i^a$
- **Pros**: simple, local, distributed, unsupervised, some experimental support
- **Cons**: symmetric, low capacity (~0.138N), catastrophic forgetting beyond capacity



Perceptron learning rule (PLR)

- On line, supervised learning rule for training individual units:
  1. Present a pattern at random: ξ^a
  2. In case of error, change W_{ji} in the opposite direction, modulated by ξ^a

$$\Delta W_{ji} = \eta\, \xi_i^a \left(\xi_j^a - sign\left(\sum_i W_{ji} \xi_i^a \right) \right)$$

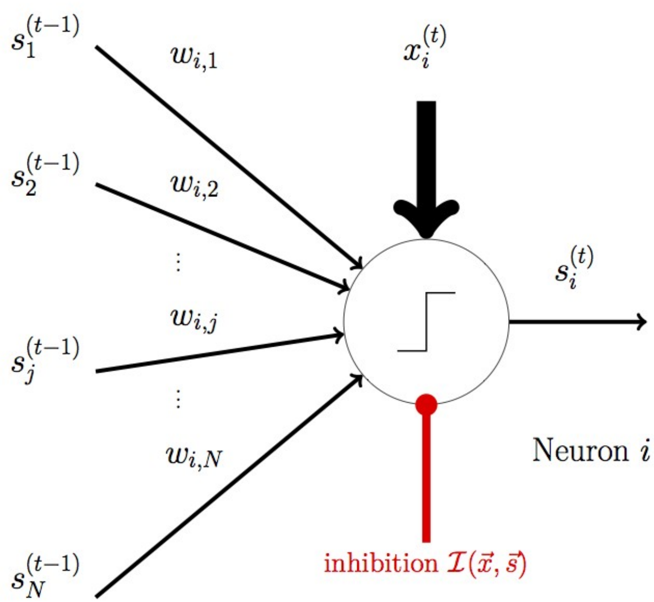
- **Pros**: able to achieve the maximal capacity (~2N) (even with **correlated patterns**), no catastrophic forgetting, allows asymmetric weights
- **Cons**: requires an explicit supervisory error signal
  - i.e. compare “output in absence of the pattern” vs “pattern itself”

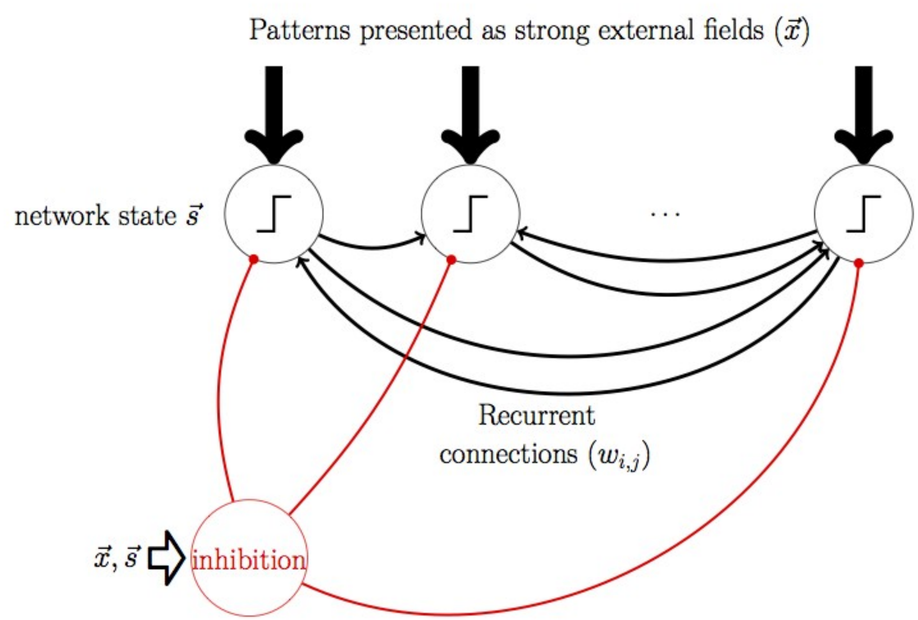
Best of both worlds?

- **Goal**: get the best of both worlds, a distributed, local, simple, unsupervised rule which achieves maximal capacity, allows asymmetric weights, has no catastrophic forgetting – in a **more realistic** setting
- **Means**: convert the PLR in an unsupervised setting, using statistical properties of the inputs
- **Main observation**: the statistic of the depolarization fields carries enough information about the error type → no need for an explicit error signal

Our network model

- **Network model**: excitatory population, state-dependent inhibitory feedback for stabilization, patterns presented via external inputs





$$s_j^{(t)} = \Theta\left(\sum_i W_{ji} s_i^{(t-1)} + x_j^{(t)} - I\left(\vec{x}^{(t-1)}, \vec{s}^{(t-1)}\right) - \theta\right)$$

threshold

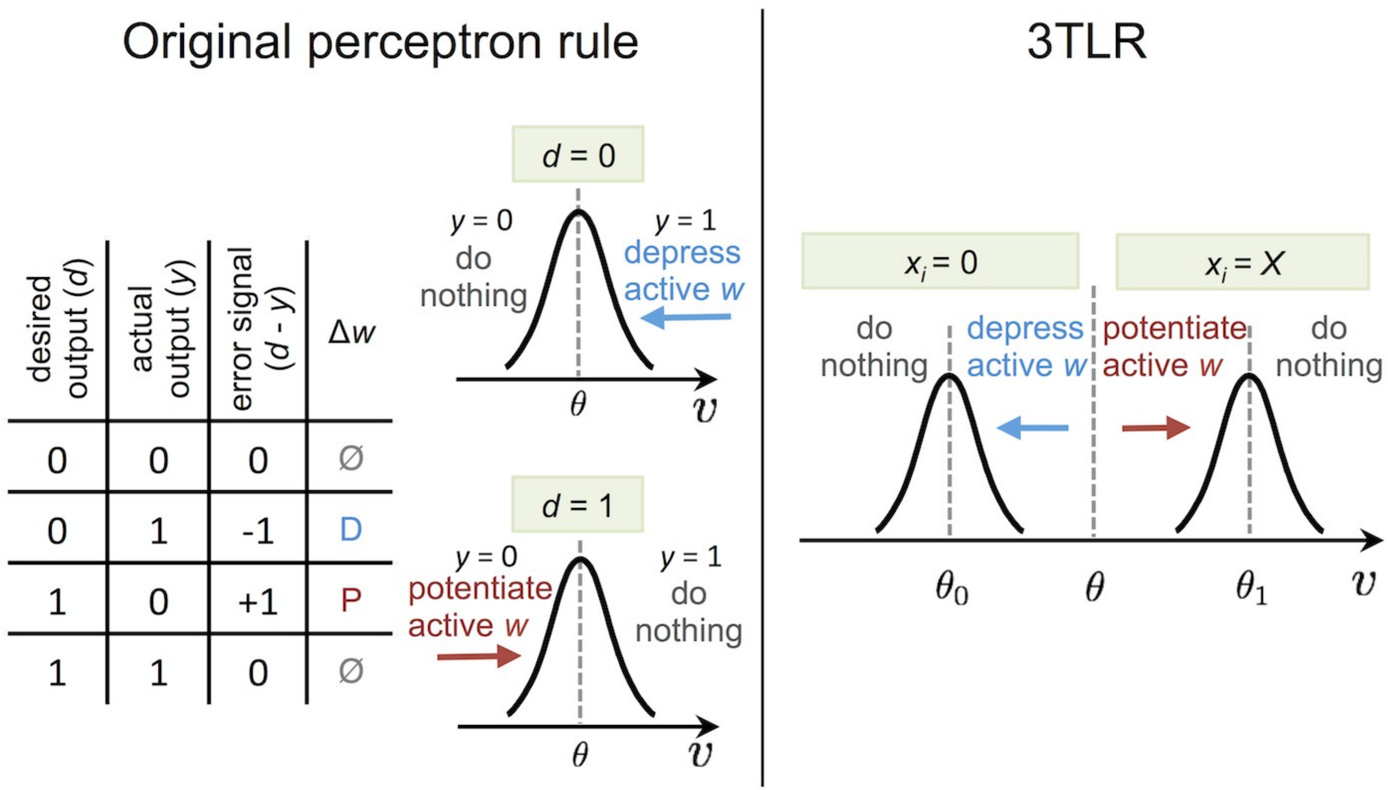
recurrent

external

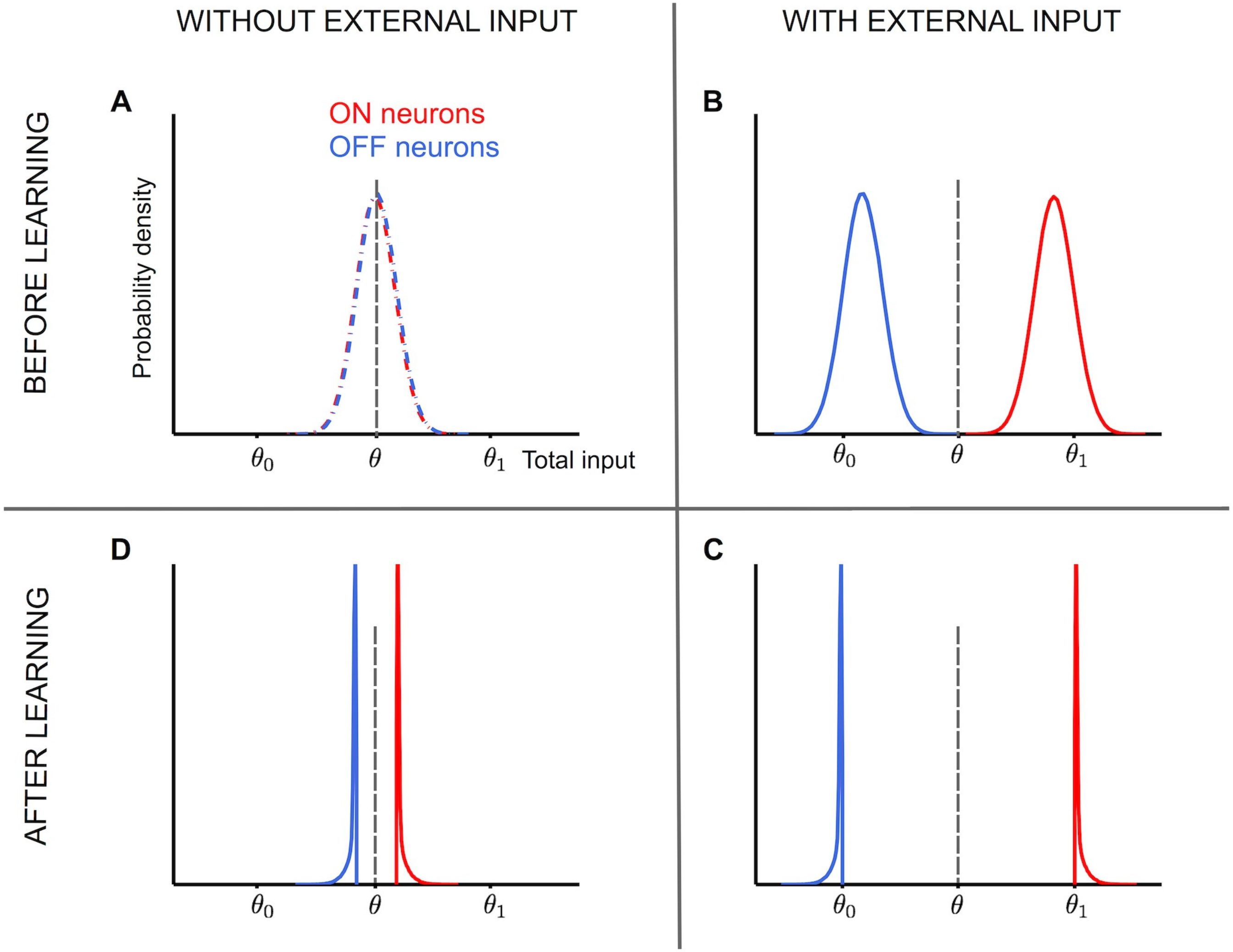
inhibition

A three-threshold learning rule (3TLR)

- Converting the PLR into an **unsupervised** rule: 3TLR
- **Crucial observation**: depolarizations $\sum_i W_{ji} s_i$ are distributed according to a Gaussian of width $O\left(\sqrt{N}\right)$; external inputs x_j make them bimodal

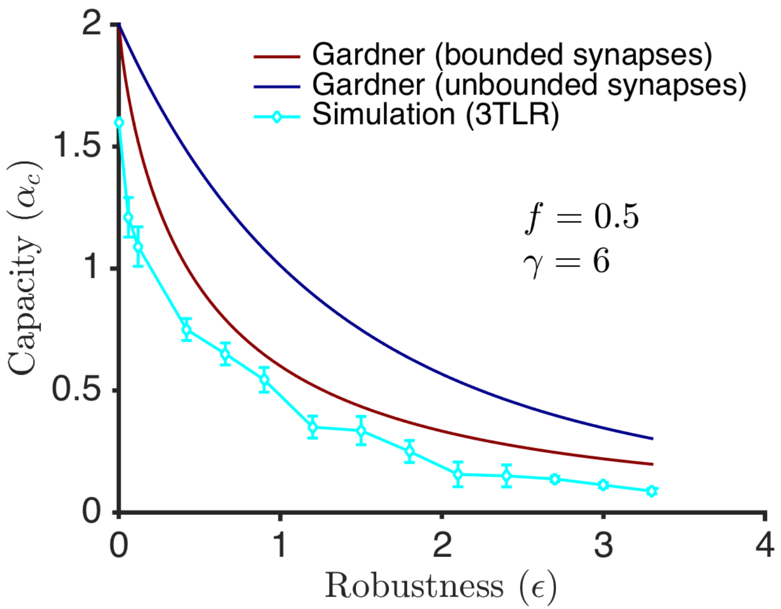


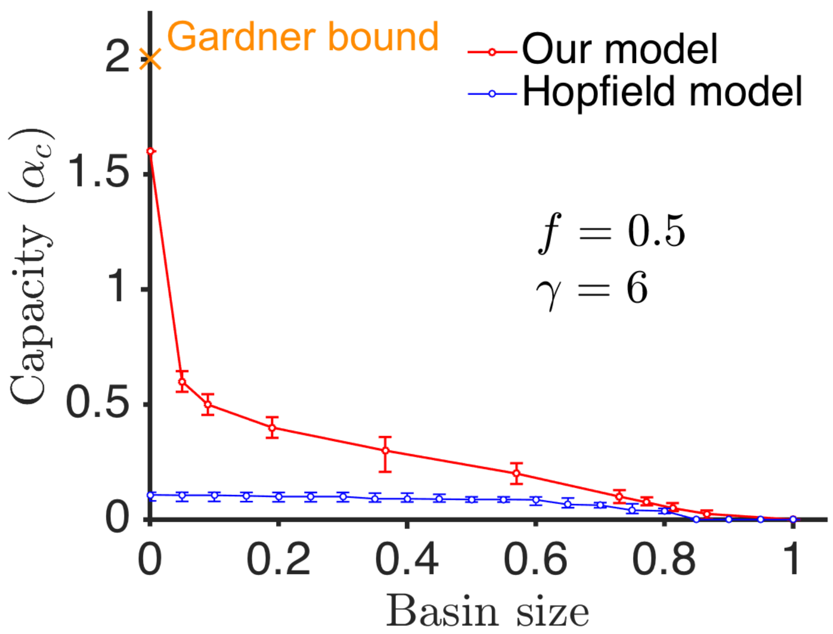
3TLR in action



Simulation results

- Many-fold increase in capacity w.r.t. Hopfield network, even though the theoretical capacity is lower
- Works in sparse regime, with correlated patterns etc.
- Depends on the external inputs being strong enough (although...)
- After learning, many synapses are off (“silent”) → sparsification





Further comments, future directions

- Parameter tuning → unsupervised pre-training phase (learn the general statistics of the inputs)
- 3TLR can be framed within the BCM theory (with additional specifications)
- Experimental evidence?
- The transformation could be applied to (essentially) any supervised rule (e.g. discrete synapses)
- Future direction: more realistic neurons (firing-rate, integrate and fire, HH) and general scenario

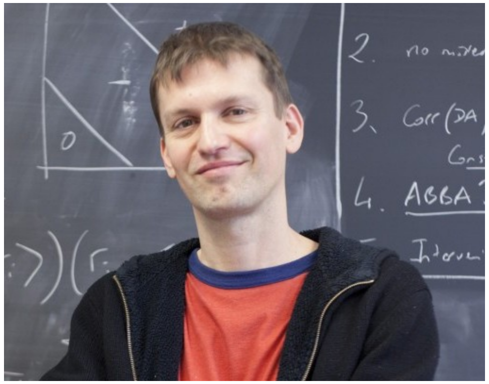
Thanks!

see details in (to be published soon):

A Three-Threshold Learning Rule Approaches the Maximal Capacity of Recurrent Neural Networks, A. Alemi, C. Baldassi, N. Brunel and R. Zecchina, *Plos. Comp. Biol.* 2015

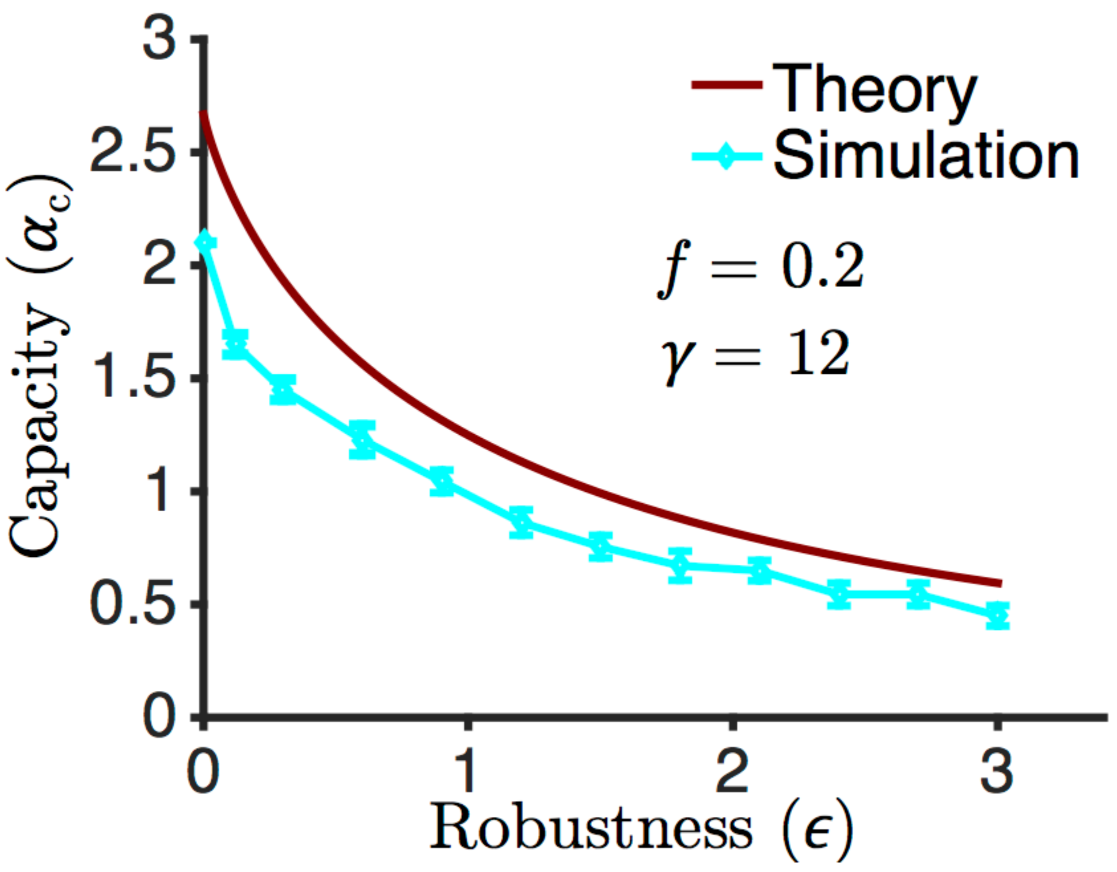
Alireza Alemi

Riccardo Zecchina



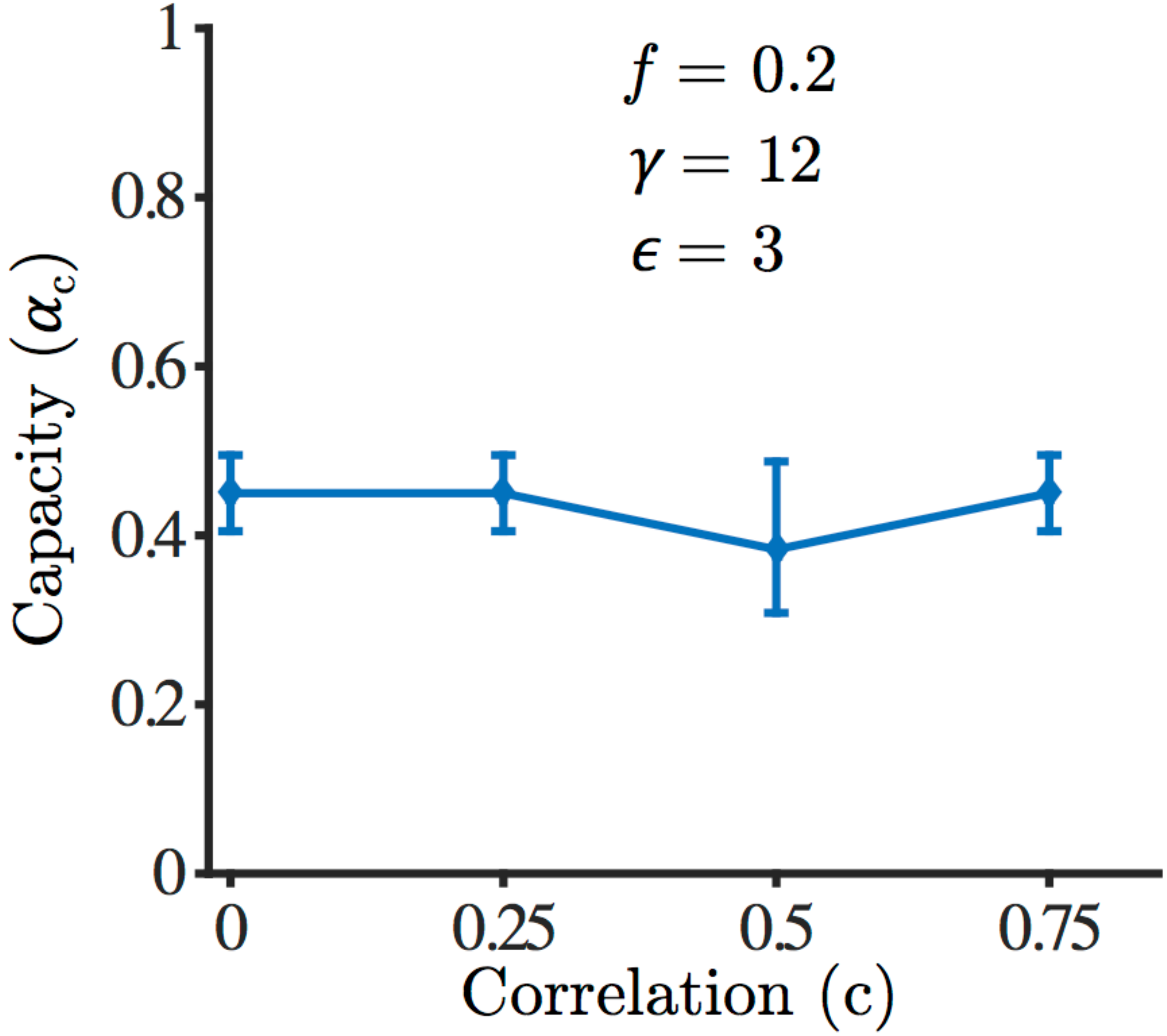
Nicolas Brunel

Sparse case

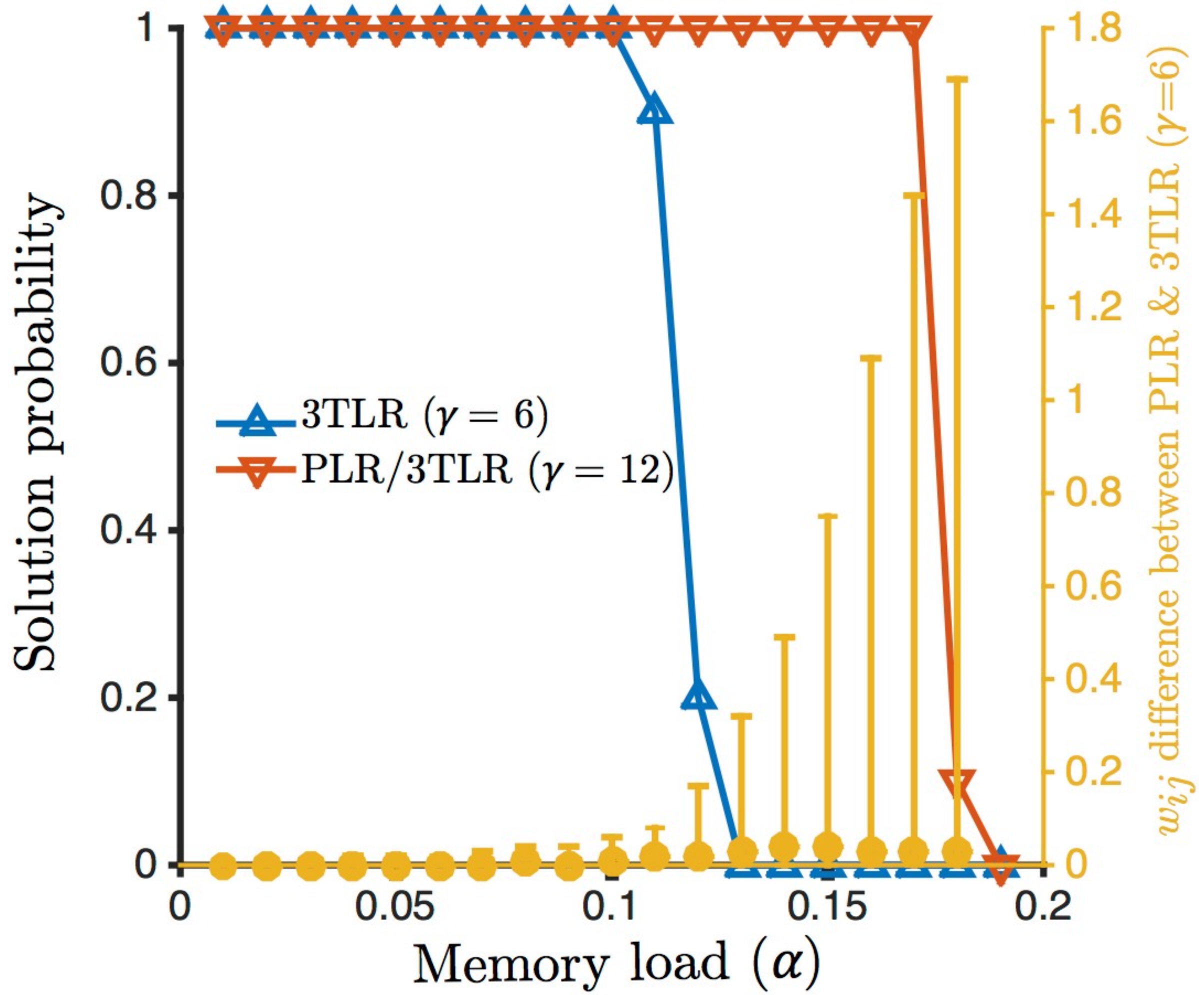


Theory
Simulation
$f = 0.2$
$\gamma = 12$
Capacity (α_c)
Robustness (ϵ)
0
0.5
1
1.5
2
2.5
3
0
1
2
3

Correlated patterns

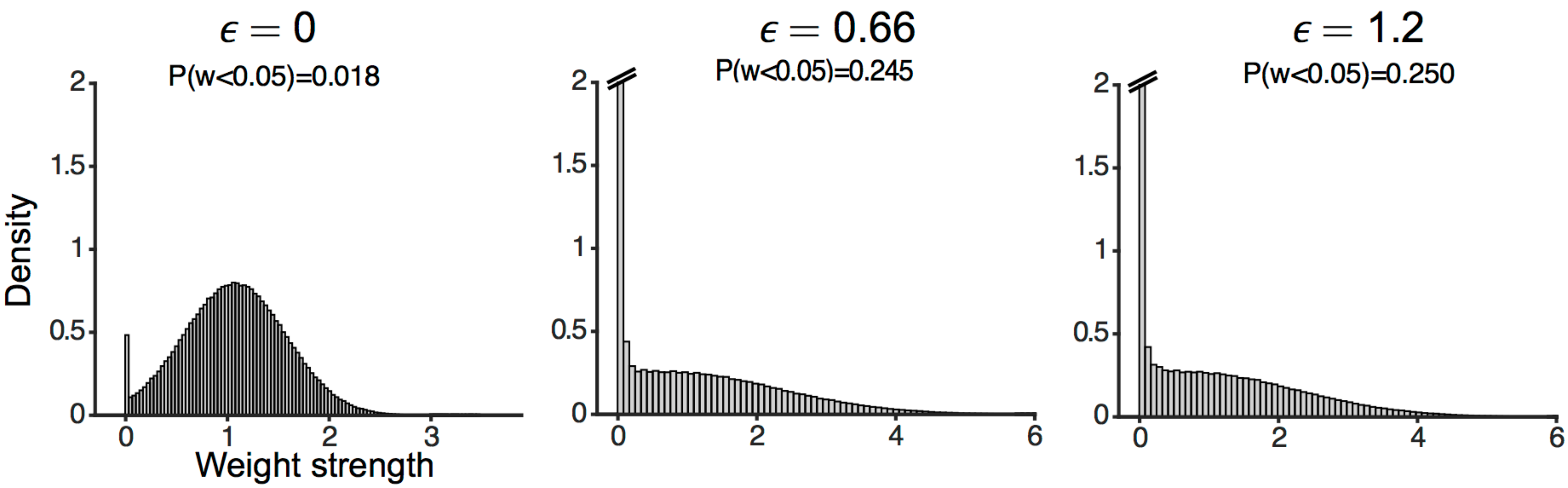


3TLR vs PRL

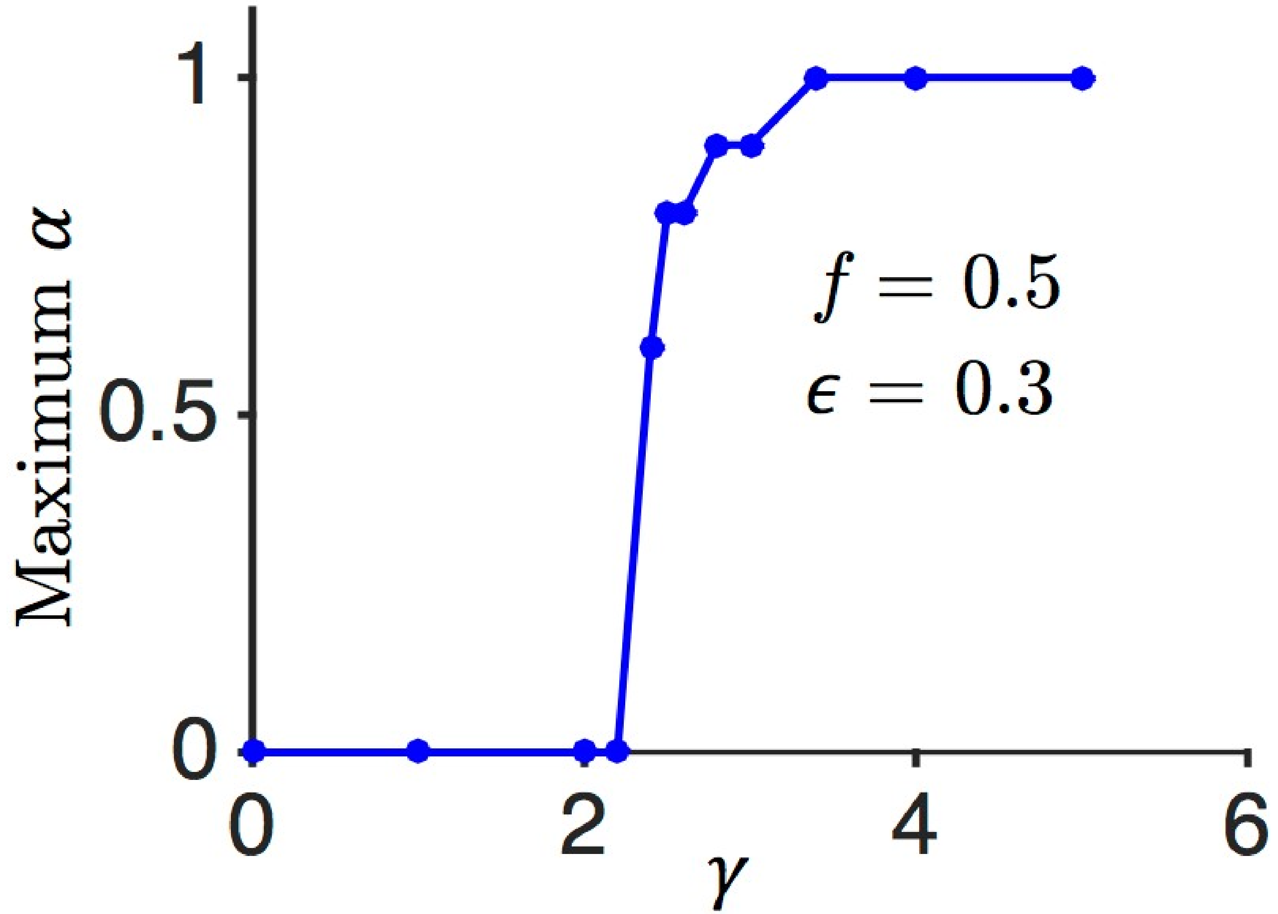


Solution probability
Memory load (α)
w_{ij} difference between PLR & 3TLR (γ=6)
3TLR ($\gamma = 6$)
PLR/3TLR ($\gamma = 12$)
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
0.05
0.1
0.15

Distribution of synaptic weights



External field strength



Weights symmetry