Transferring State Representations in Hierarchical Spiking Neural Networks

Barna Zajzon*, Renato Duarte and Abigail Morrison

10.07.2018 | IJCNN, SS7: ADVANCES IN RESERVOIR COMPUTING

b.zajzon@fz-juelich.de

IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE
WCCI 2018
Rio de Janeiro, Brazil | July 8-13, 2018

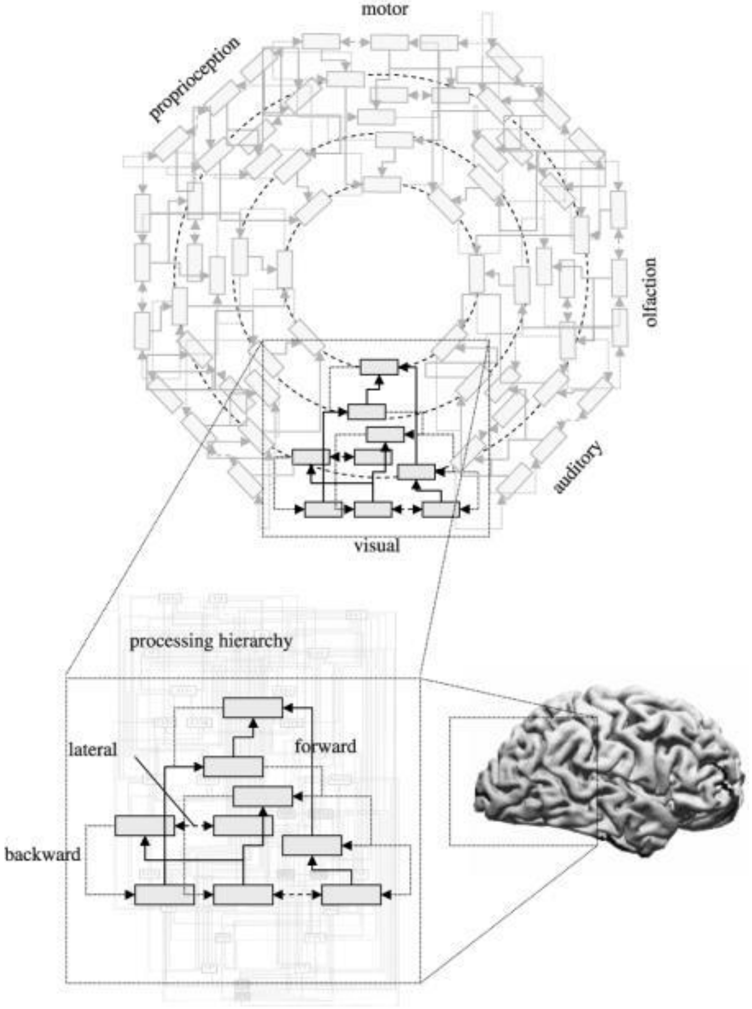
JÜLICH
Forschungszentrum

Introduction

- Hierarchical modularity as a design principle
- Neocortex as a large distributed hierarchy of recurrent spiking networks
- Mechanism for information transfer

What features enable efficient information transfer? … in a computationally useful way?

Context: Reservoir Computing



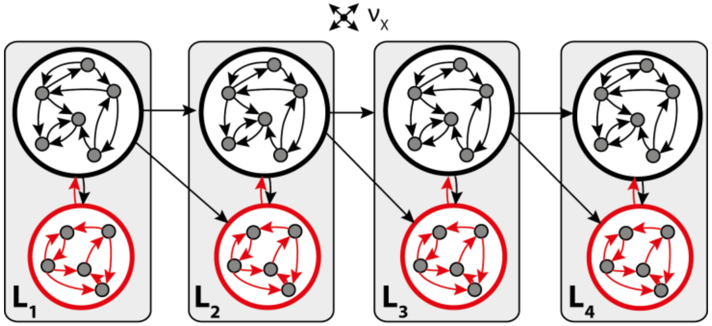
Friston, K. (2005)

JÜLICH Forschungszentrum

Setup

- 4 identical layers of **balanced random networks**
- 10000 spiking neurons (LIF)
- Static synaptic weights

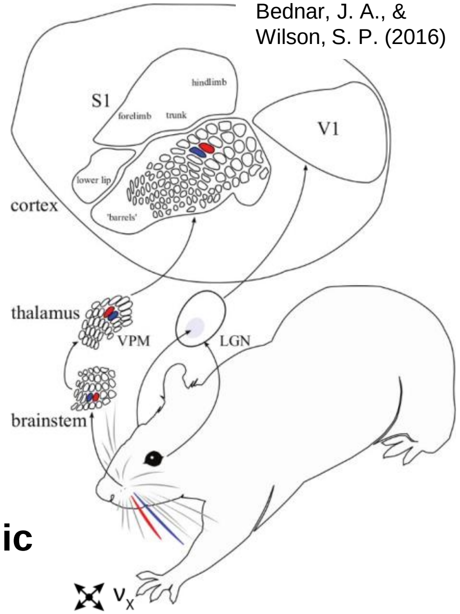
random feed-forward



- Each neuron → same amount of excitatory input
- Tune parameters for **asynchronous irregular activity** (first 2 layers)

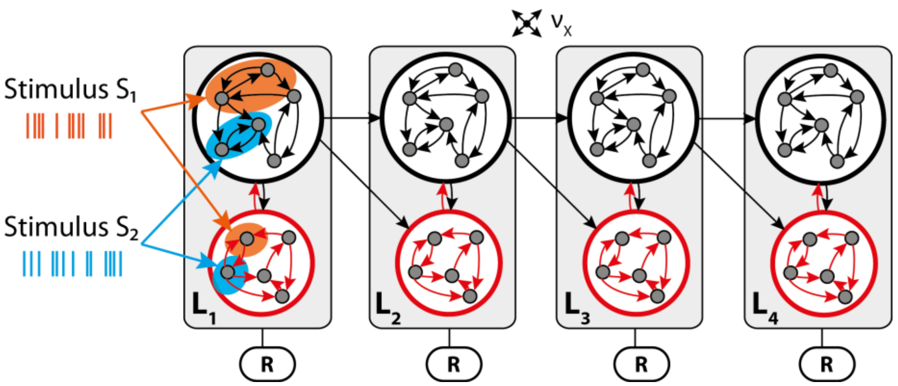
Setup

Bednar, J. A., & Wilson, S. P. (2016)

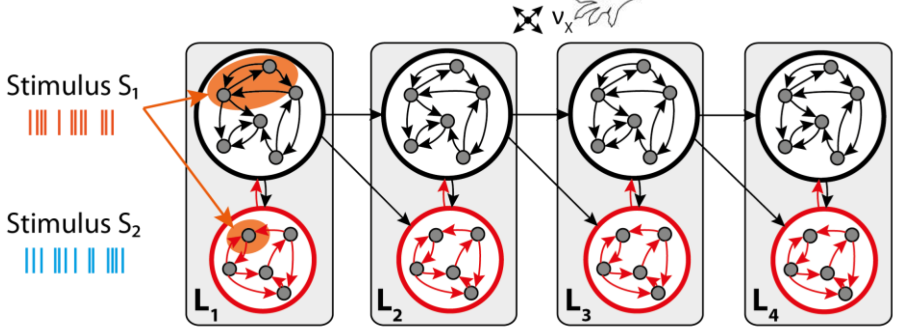


- 4 identical layers
- 10000 spiking neurons (LIF)
- Static synaptic weights
- Poisson input to sub-populations in L_1

random feed-forward

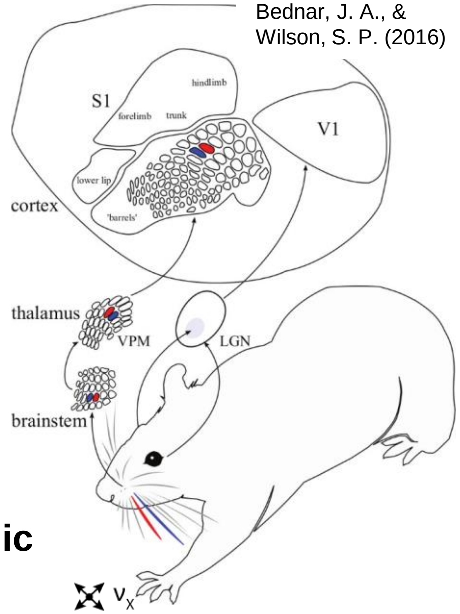


topographic



- ➢ Each neuron → same amount of excitatory input
- ➢ Tune parameters for **asynchronous irregular activity** (first 2 layers)
- ➢ Readout from E populations in each layer

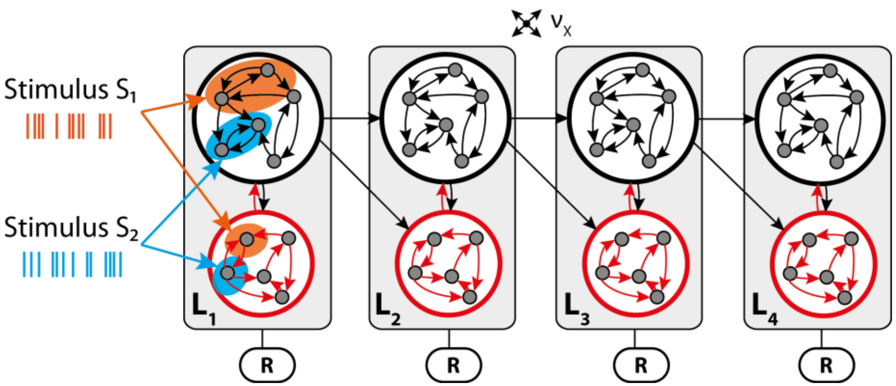
Setup



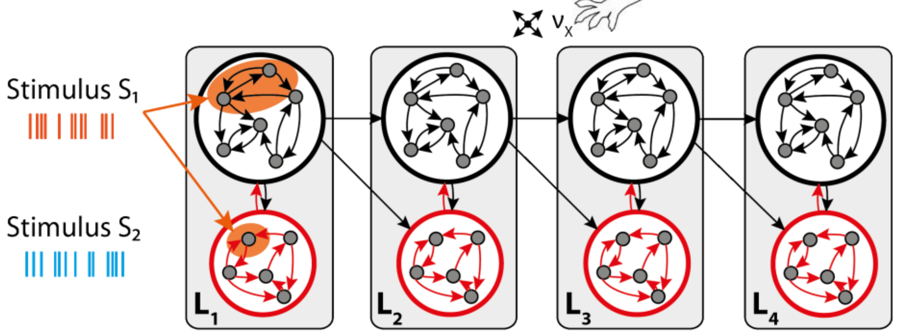
Bednar, J. A., & Wilson, S. P. (2016)

- 4 identical layers
- 10000 spiking neurons (LIF)
- Static synaptic weights
- Poisson input to sub-populations in L_1

random feed-forward



topographic



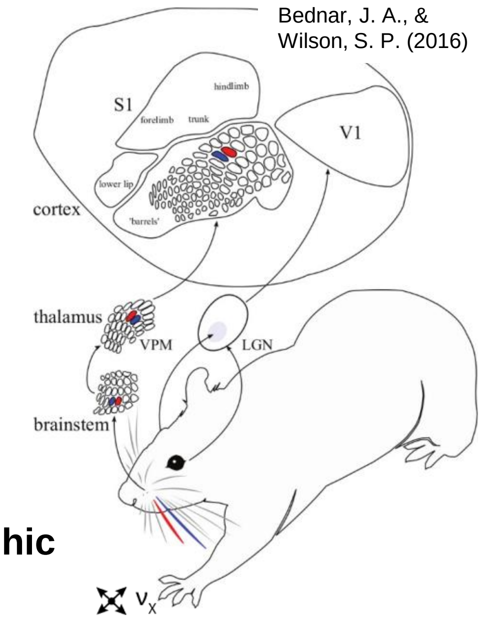
- ➢ Each neuron → same amount of excitatory input
- ➢ Tune parameters for **asynchronous irregular activity** (first 2 layers)
- ➢ Readout from E populations in each layer

- ➢ Stimulus-specific sub-populations conserved across the hierarchy through structured connectivity
- ➢ Randomly selected, overlap allowed
- ➢ Receptive field size fixed across the hierarchy

JÜLICH
Forschungszentrum

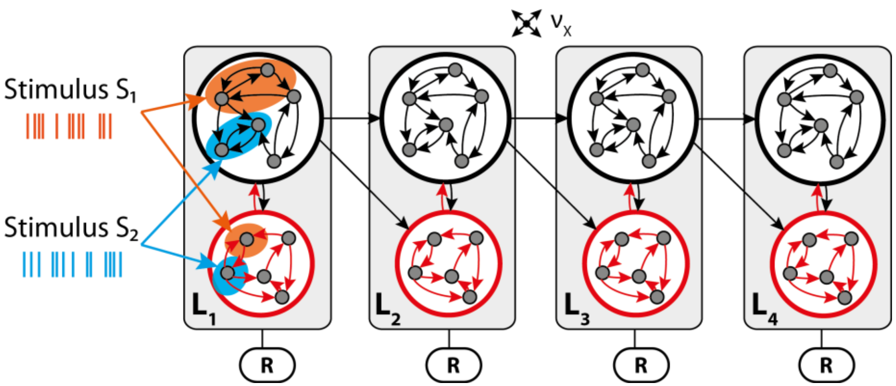
Setup

Bednar, J. A., & Wilson, S. P. (2016)

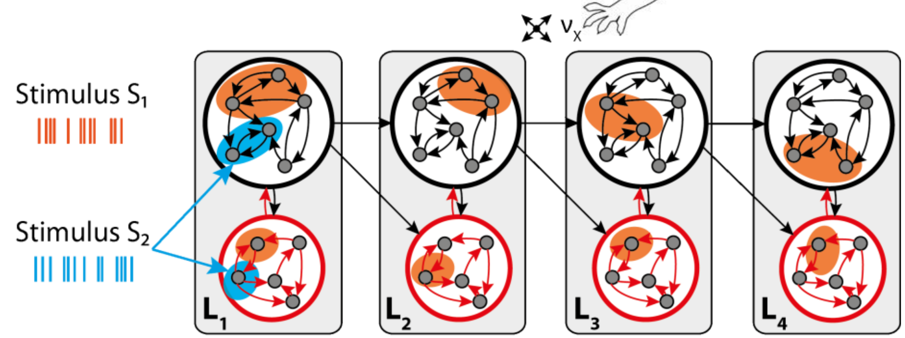


- 4 identical layers
- 10000 spiking neurons (LIF)
- Static synaptic weights
- Poisson input to sub-populations in L_1

random feed-forward



topographic

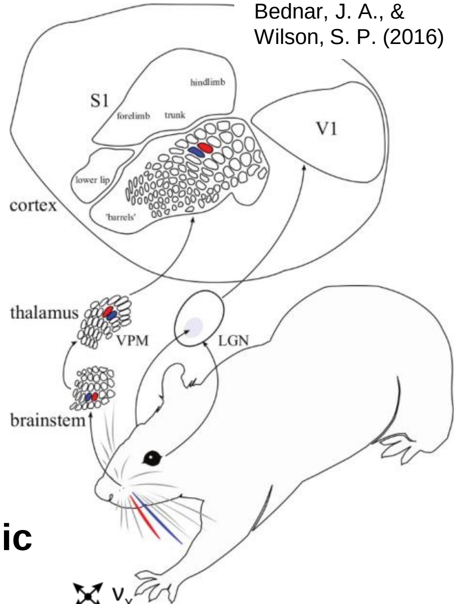


- ➢ Each neuron → same amount of excitatory input
- ➢ Tune parameters for **asynchronous irregular activity** (first 2 layers)
- ➢ Readout from E populations in each layer

- ➢ Stimulus-specific sub-populations conserved across the hierarchy through structured connectivity
- ➢ Randomly selected, overlap allowed
- ➢ Receptive field size fixed across the hierarchy

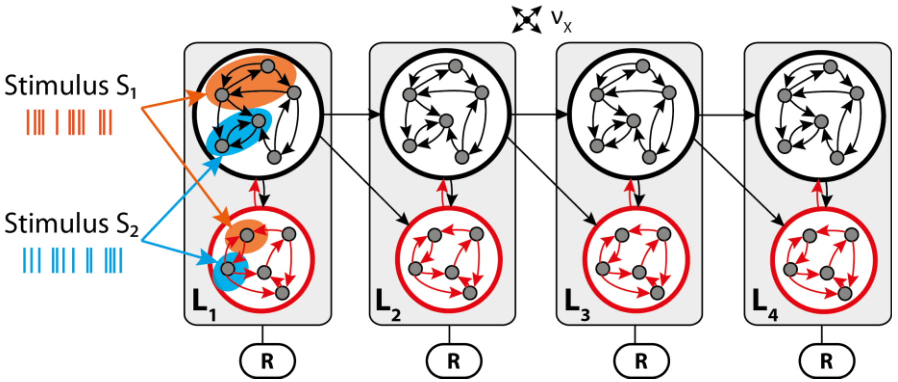
Setup

Bednar, J. A., & Wilson, S. P. (2016)

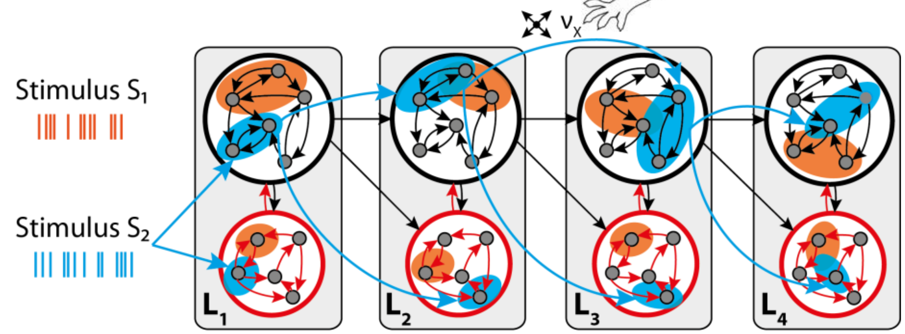


- 4 identical layers
- 10000 spiking neurons (LIF)
- Static synaptic weights
- Poisson input to sub-populations in L_1

random feed-forward



topographic



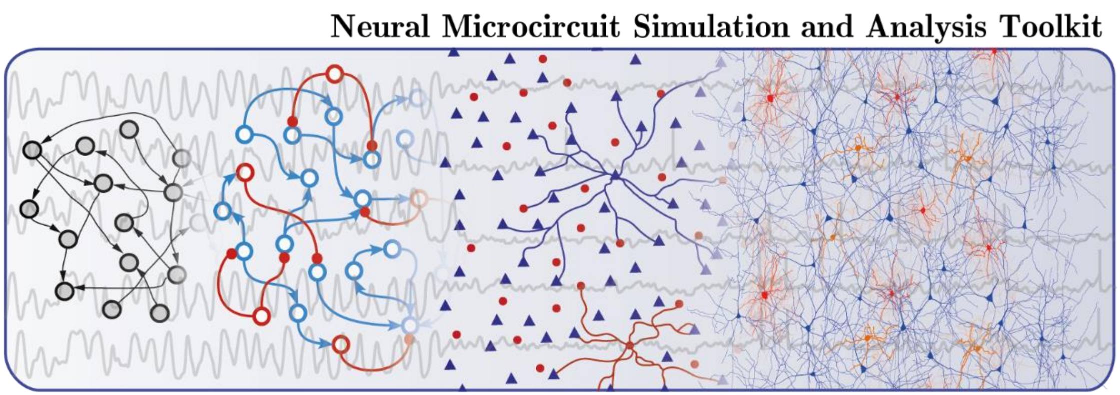
- ➢ Each neuron → same amount of excitatory input
- ➢ Tune parameters for **asynchronous irregular activity** (first 2 layers)
- ➢ Readout from E populations in each layer

- ➢ Stimulus-specific sub-populations conserved across the hierarchy through structured connectivity
- ➢ Randomly selected, overlap allowed
- ➢ Receptive field size fixed across the hierarchy

Neural Microcircuit Simulation and Analysis Toolkit

Python package to **build**, **simulate** and **analyze** complex **neuronal microcircuits**

scalable and ***reproducible***



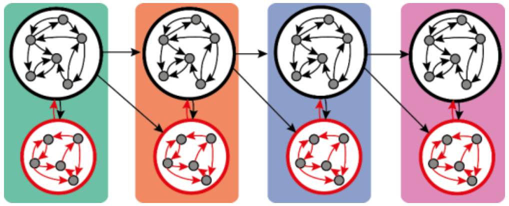
GitHub

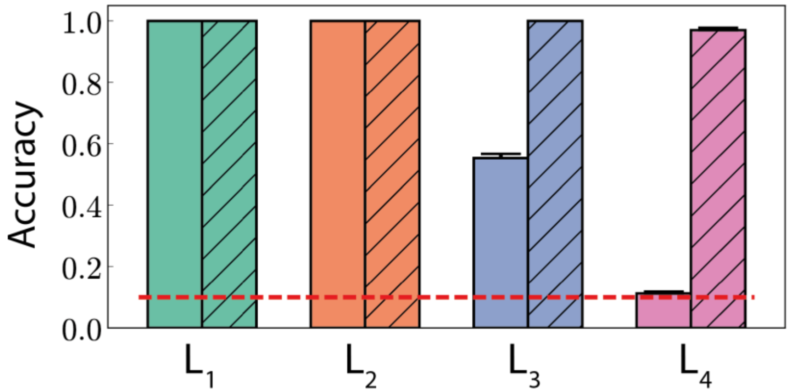
https://github.com/rcfduarte/nmsat

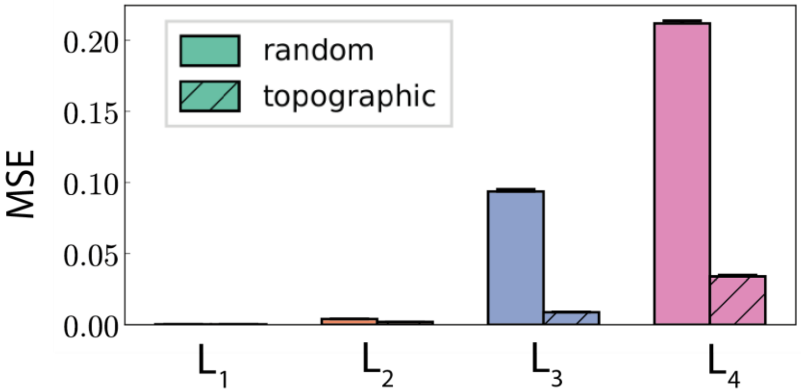
DOI 10.5281/zenodo.582645

Renato Duarte, Barna Zajzon, & Abigail Morrison. (2017). Neural Microcircuit Simulation and Analysis Toolkit

Stimulus representation





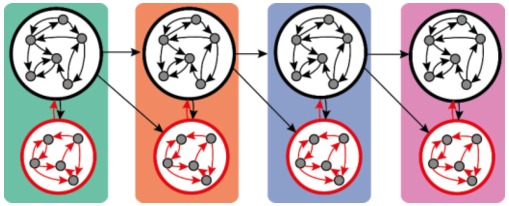


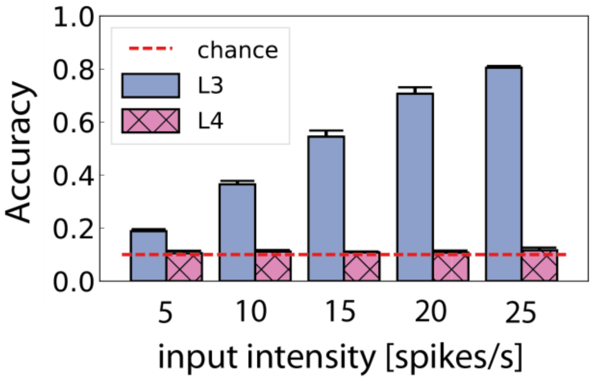
Classification of stimulus identity with $\nu_{\mathrm{stim}} = 15\ \mathrm{spk/sec}$, 10 stimuli

- Random connectivity enables stimulus decoding **only up to L_3**
- Topography improves accuracy in last 2 layers

JÜLICH Forschungszentrum

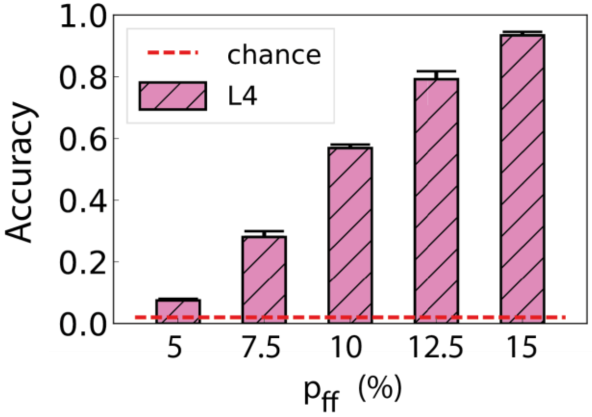
Stimulus representation





Random network

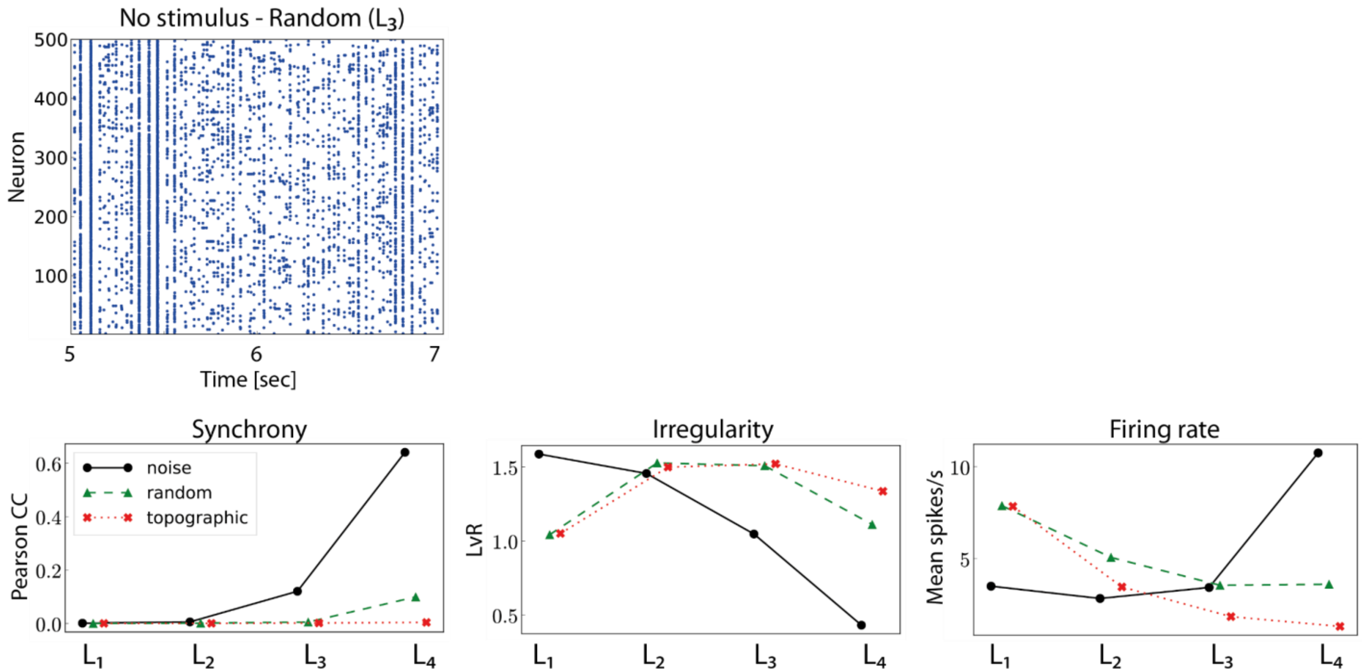
- Propagation depends on **input rate**
- No effect on L_4



Topographic maps

- Capacity depends on **connection density** within topographic projections
- Compensates for increased overlap in case of more stimuli

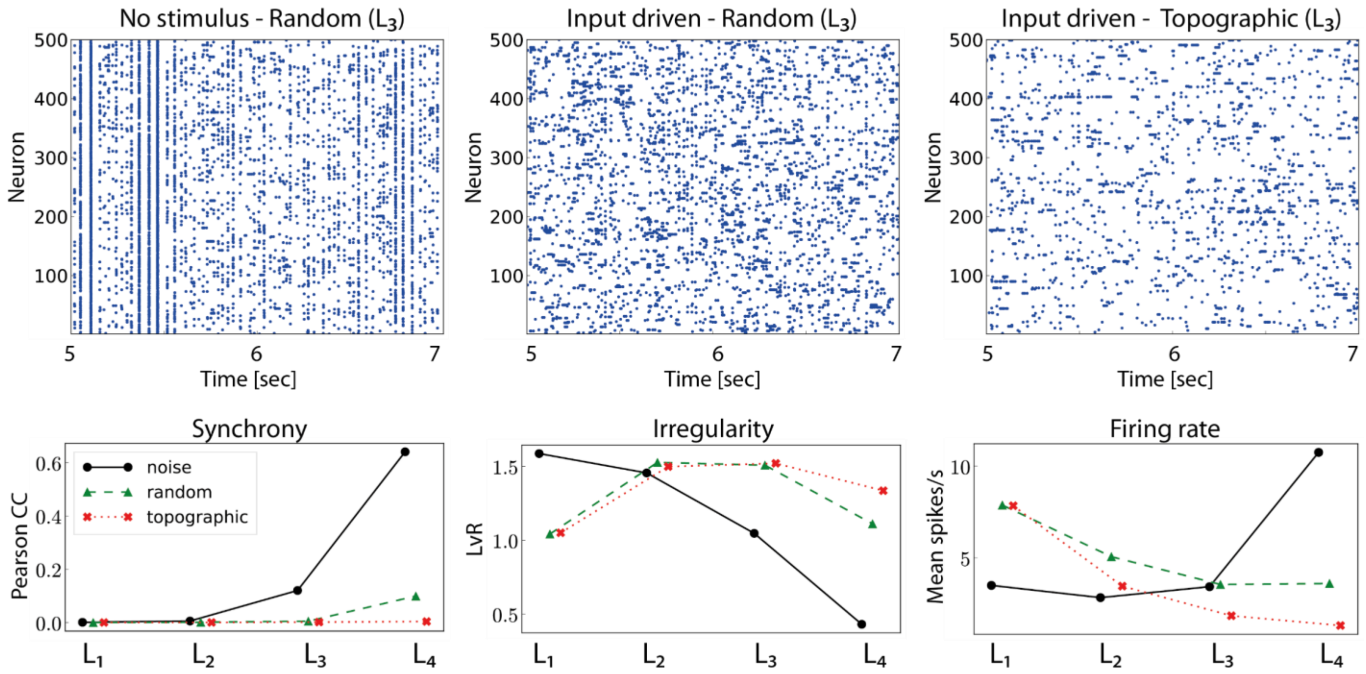
Activity statistics



Noise-driven, no stimulus

- Firing is more regular
- Synchrony increases downstream (shared input effect)

Activity statistics



Noise-driven, no stimulus

- Firing is more regular
- Synchrony increases downstream (shared input effect)

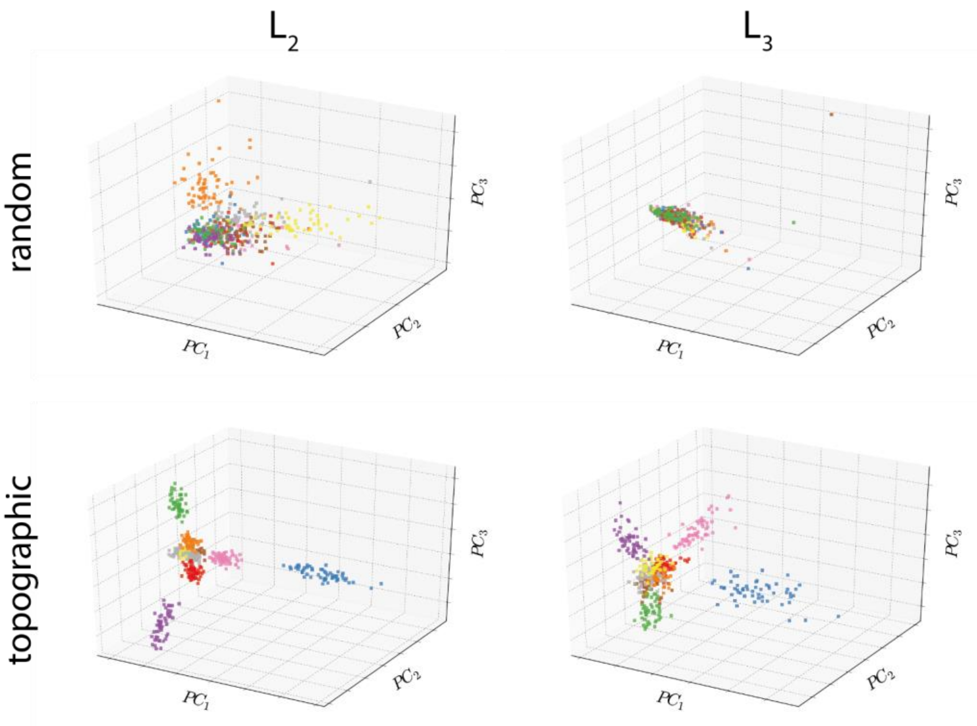
Input-driven

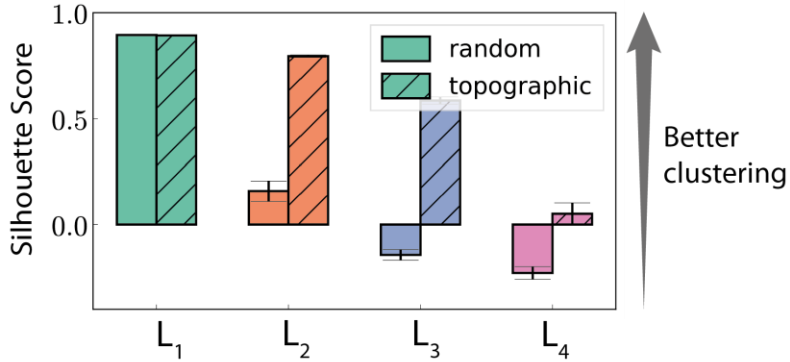
- Global AI state maintained
- Synchrony reduced (topographic)
- Increased irregularity except L_1

Topographic networks are more resource-efficient

State space organization

Ideally → clearly segregated stimulus-specific clusters





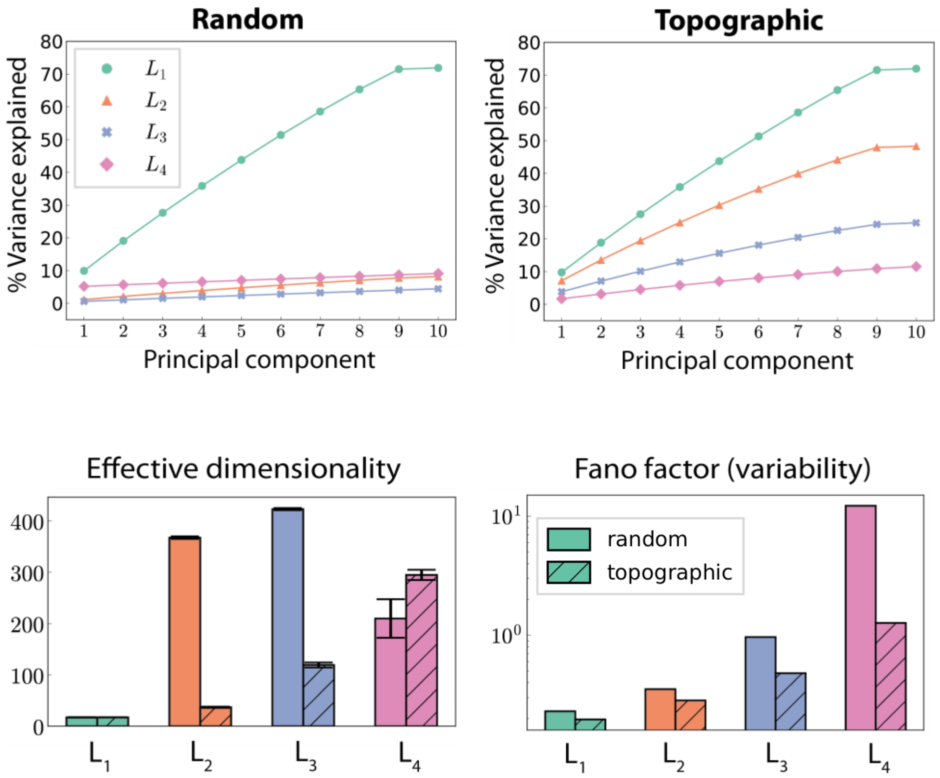
- Clustering quality decays through the hierarchy
- Topography leads to better state separation (silhouette score)
- In line with computational performance

JÜLICH Forschungszentrum

State space organization

How are the networks exploring the high-dimensional state space?

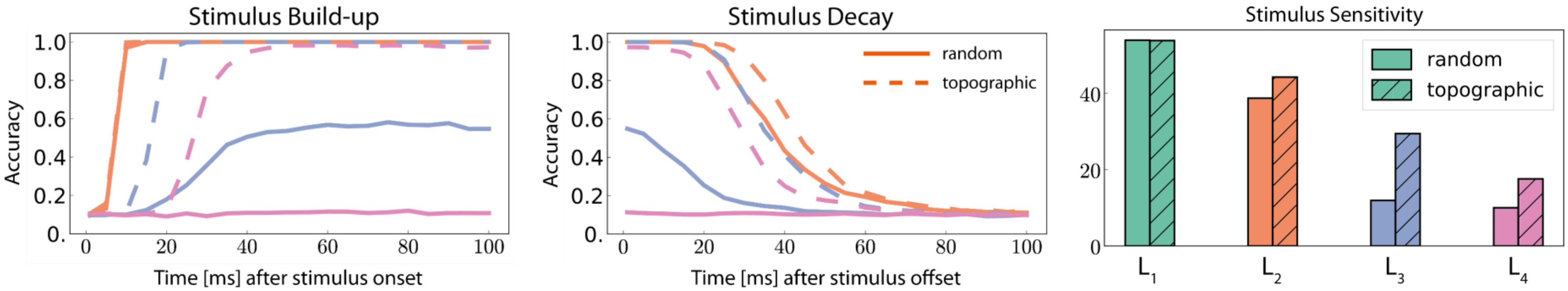
- Stimulus “enslaves” neural activity
- Effective dimensionality[1] increases with hierarchical depth
- Topography extends duration of stimulus representations
- Dimensionality and variability inversely related to performance



[1] Mazzucato et al, (2016)

Memory capacity and stimulus sensitivity

Stimuli presented sequentially



- Representations build-up over exposure time, slower downstream
- Topographic networks react faster

- Input representations gradually disappear (fading memory)
- Memory capacity decays with hierarchical depth
- Topography (marginally) increases memory

- **Stimulus sensitivity**: how long non-interfering representations are maintained
  - ➢ decreases through hierarchy, better with topography

Summary

Useful constraints for building hierarchical (balanced) spiking networks

- Random connectivity enough for local transmission, longer distances require topographic precision

- Topographic maps

  - better performance
  - memory capacity
  - more efficient
  - low-dimensional responses
  - less variability
  - more robust

Thank you!