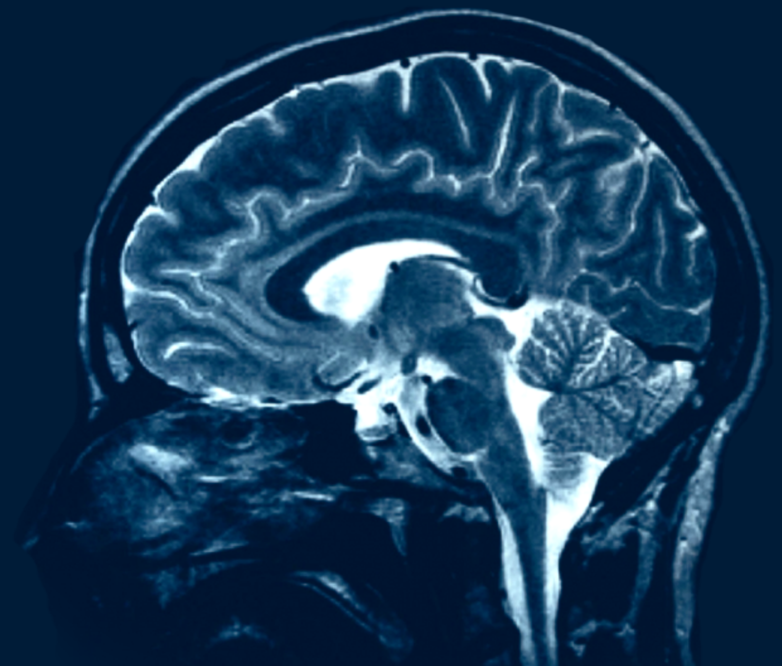
Using Spiking Neural Networks for Pattern Recognition

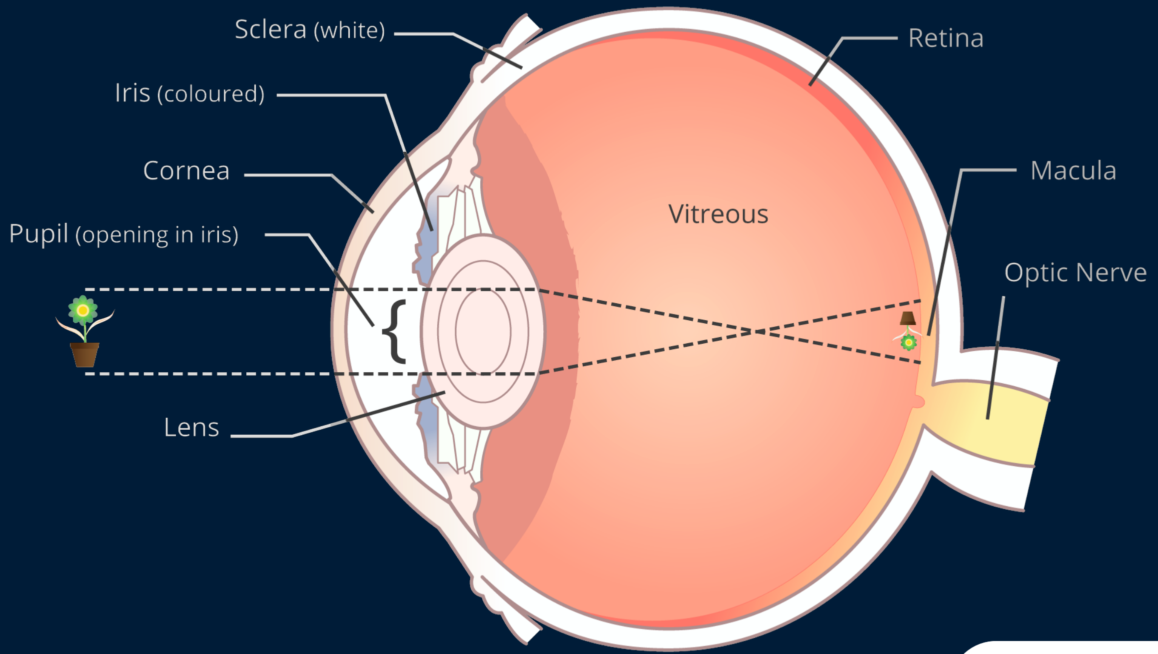
Nima Mohammadi

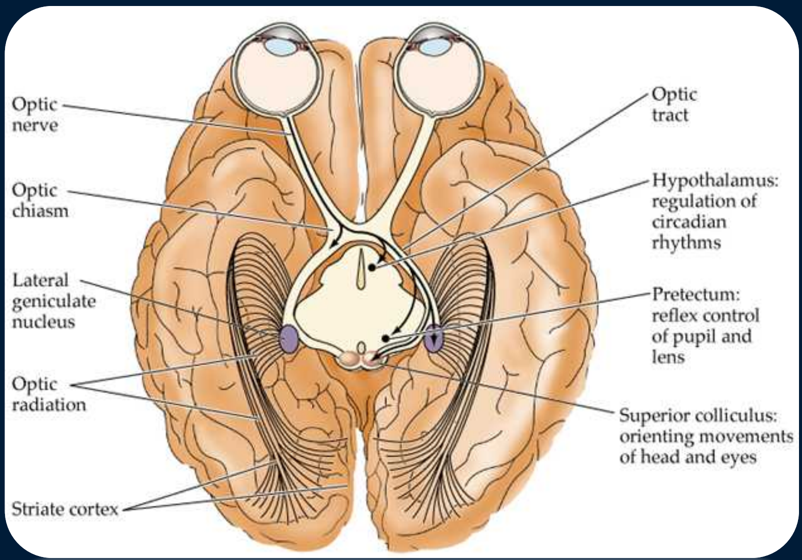
Under Supervision of **Dr. Ganjtabesh**

September 2016

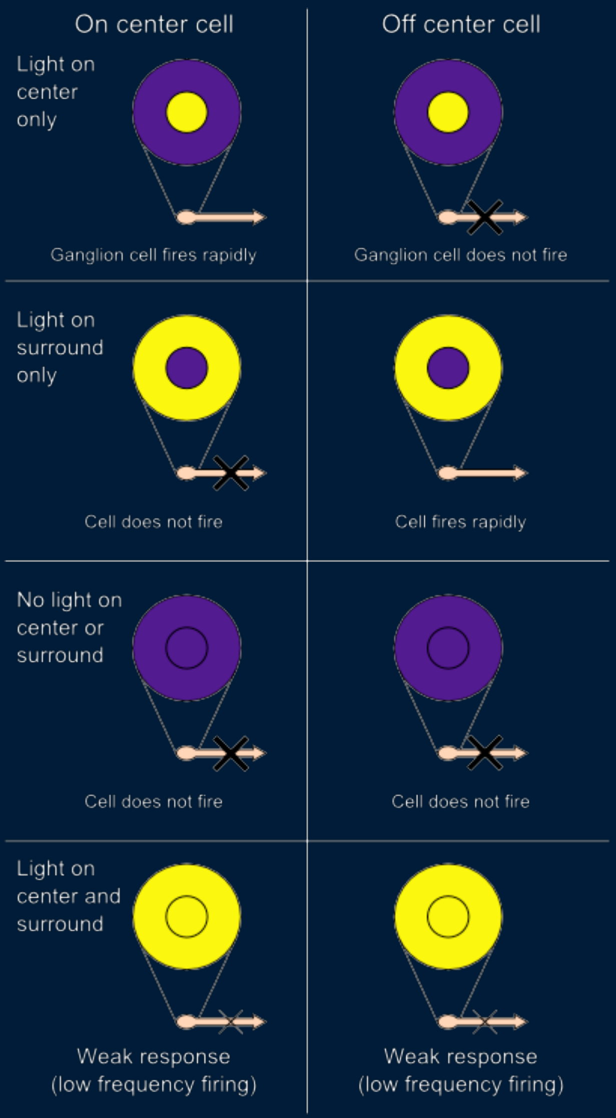


Brain | Retina

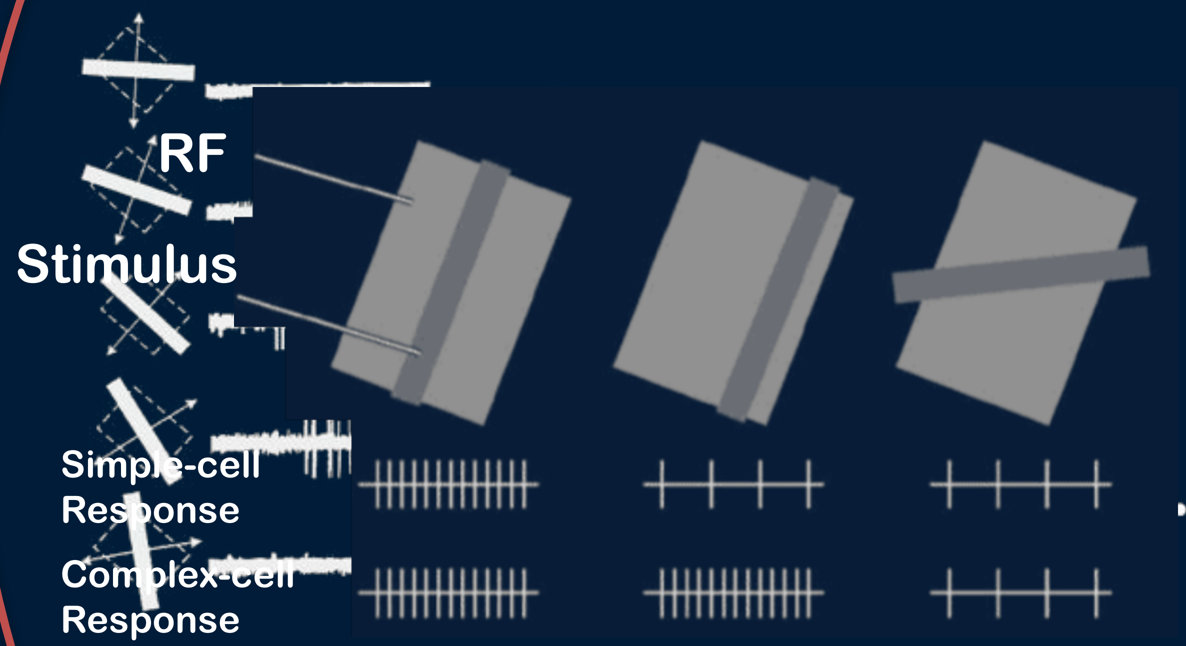




Lateral Geniculate Nucleus and V1

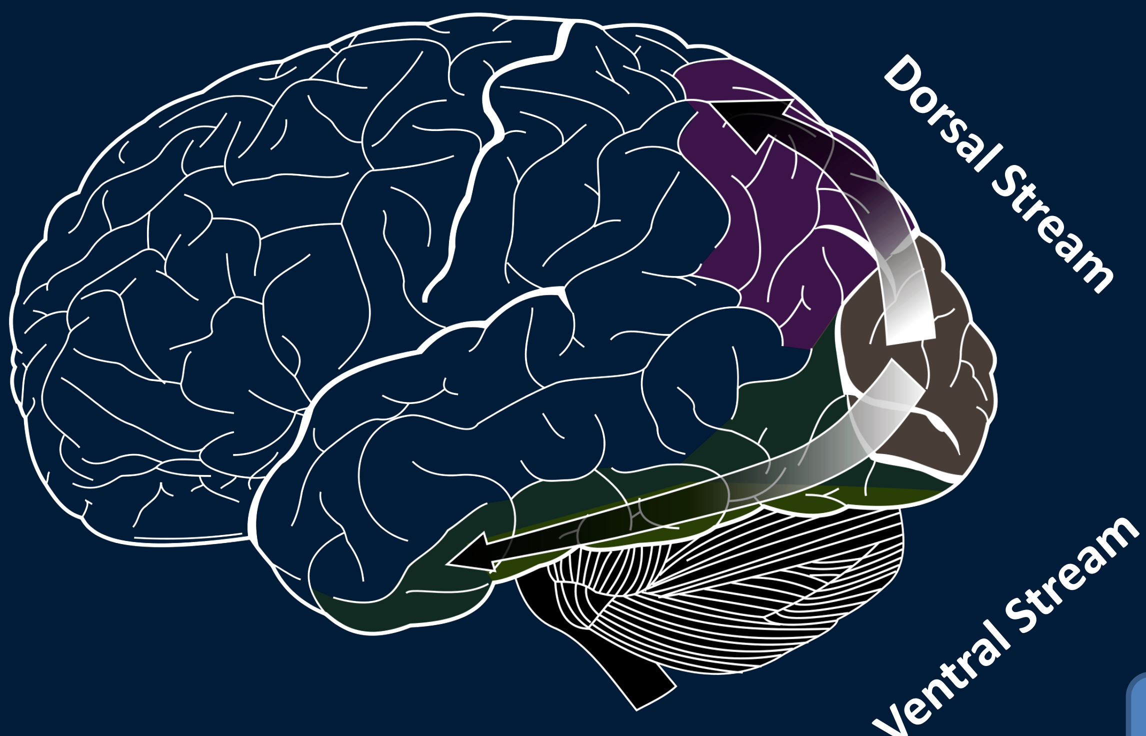


Ganglion Bipolar RF

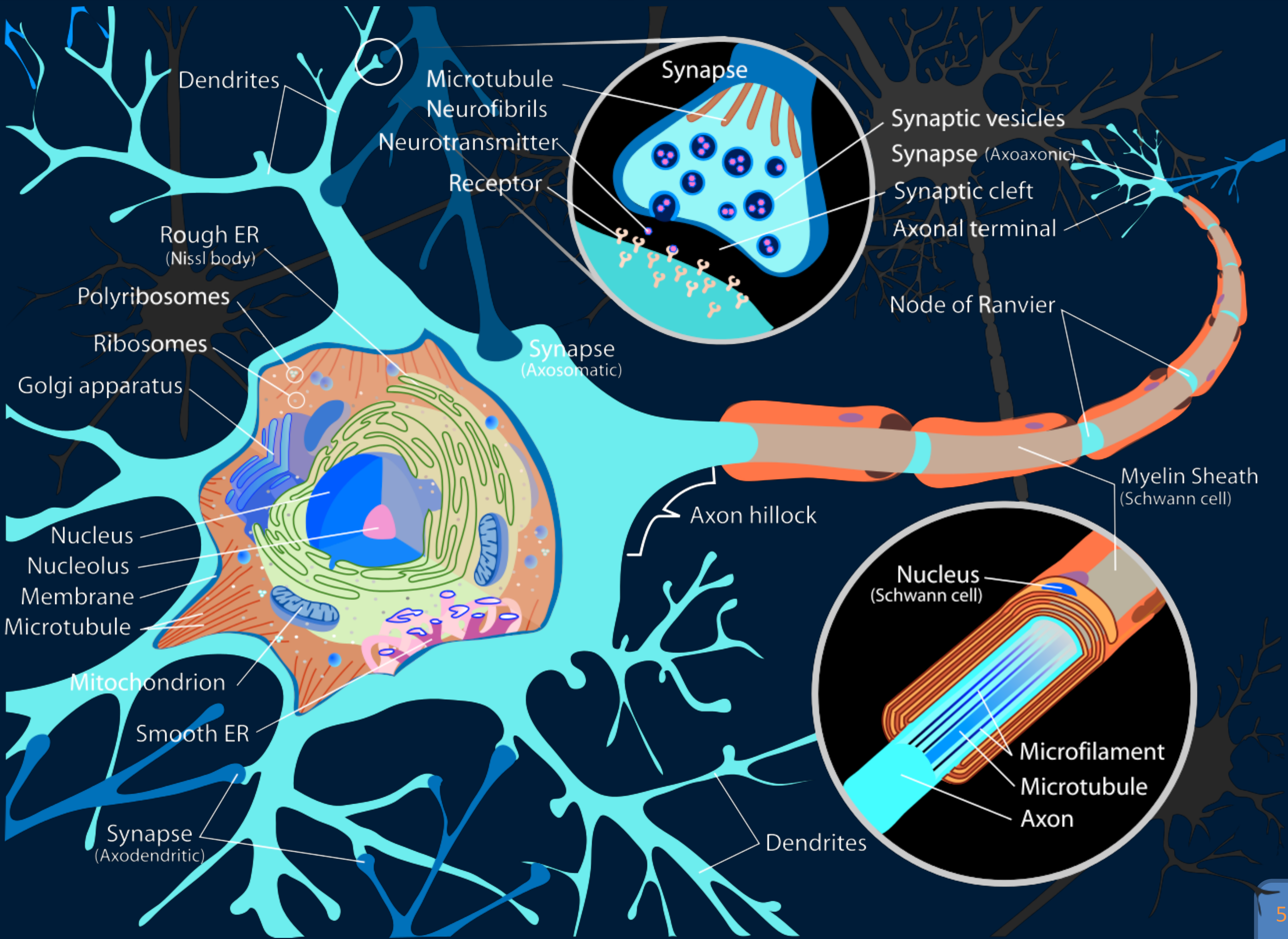


V1 Orientational RF

- **Ventral stream** (what pathway) including V1, V2, V4, IT
- **Dorsal stream** (where pathway) including V1, V2, V3, MT, MST



Brain | Neuron



- Experience shows that a model should be as simple as possible in order to be tractable (possible to analyze, easy to do computations with) and in order to make strong statements about the physical system being modeled.

> *"It can scarcely be denied that the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience."*
>
> -Einstein

- With increased complexity of model, uncertainty of modeling results increases. Also the explanatory power of model is lost. Of course the degree of simplicity that is achievable is dependent on the scientific question posed.

- It seems that a higher level model should be composed of component models at a lower level, and in turn must be much more complex and have more parameters than a model of a neuron. If true, one might ask what is the proper lowest level? After all, the deeper the level, the more realistic, right?!

  Wrong! Including more details from lower levels leads to more model parameters which makes it harder to obtain a realistic model since the realism of a model is related to how well-constrained it is by experimental data.

- More model parameters means that more data is required to determine them, data which can often contain uncertainties and be hard to acquire.

Example: The propagation of sound through air.

- Modeler’s wrench, abstraction: taking away aspects not important for answering the scientific questions which the model is designed to address, useful models can be formulated at different levels of organization without loss of tractability.

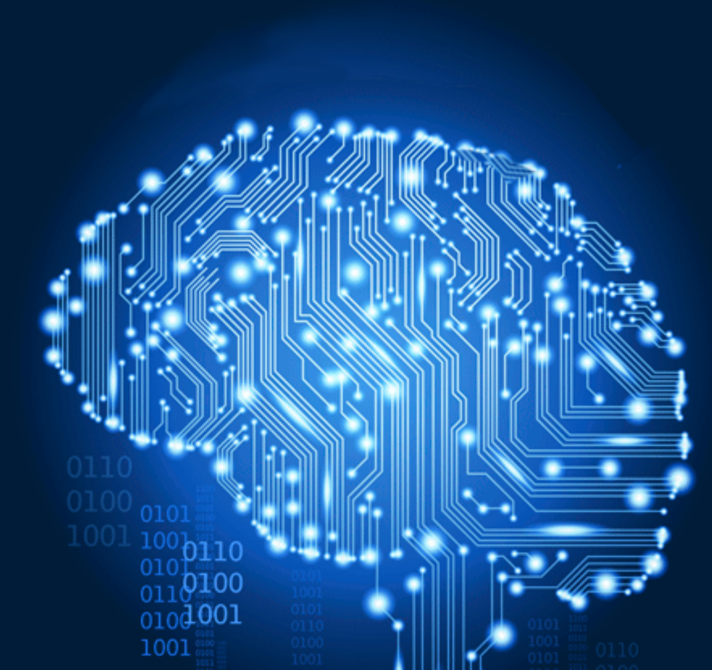
- Subsampling is often employed to decrease model size. With fewer pre-synaptic units providing synapses, it becomes necessary to exaggerate connection density or synaptic conductance. This results in a network with unnaturally few and strong signals circulating, in contrast to the real network, where many weak signals interact. Therefore differences might arise such as artificial synchronization which is a problem especially since synchronization is one of the more important phenomena.

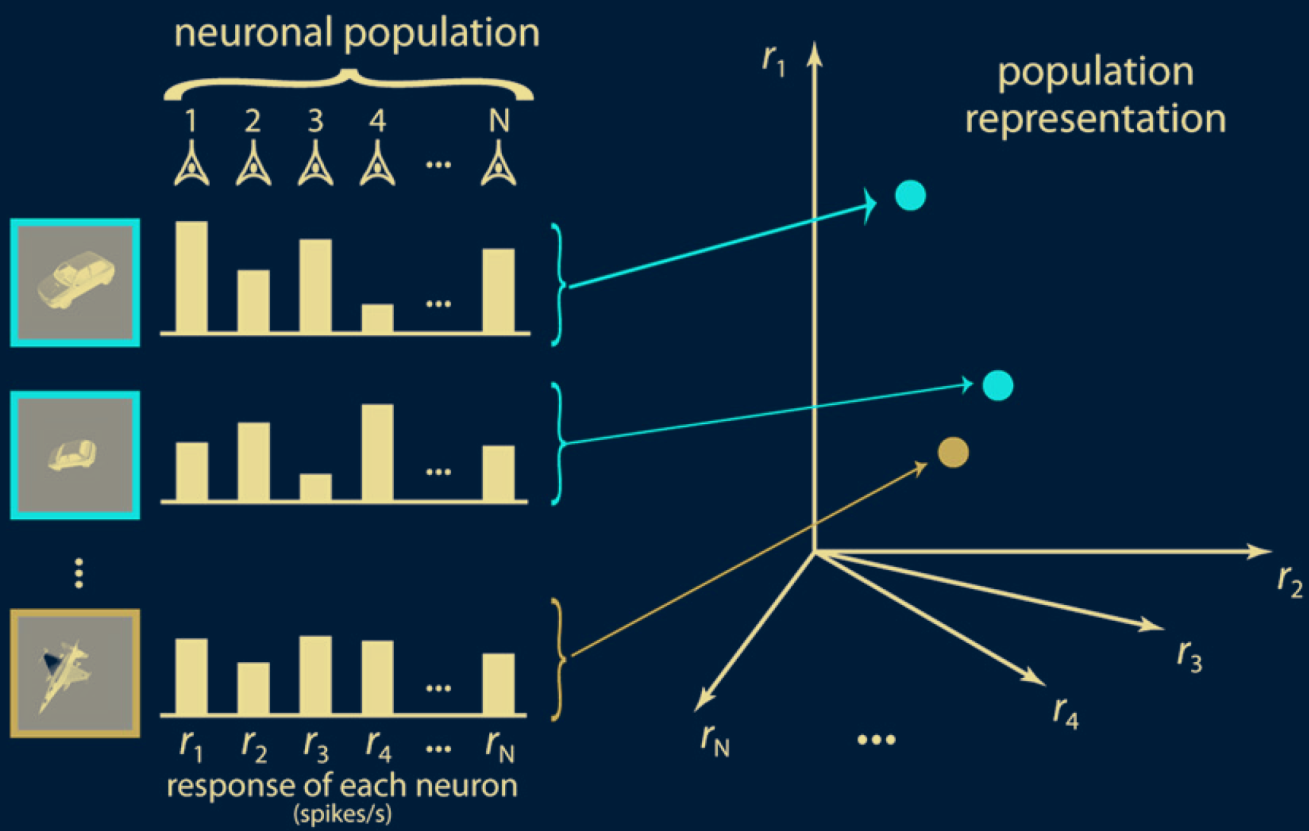
Neuronal Networks

- Improve understanding of brain functionality involving interactions of billions of neuronal and synaptic processes.

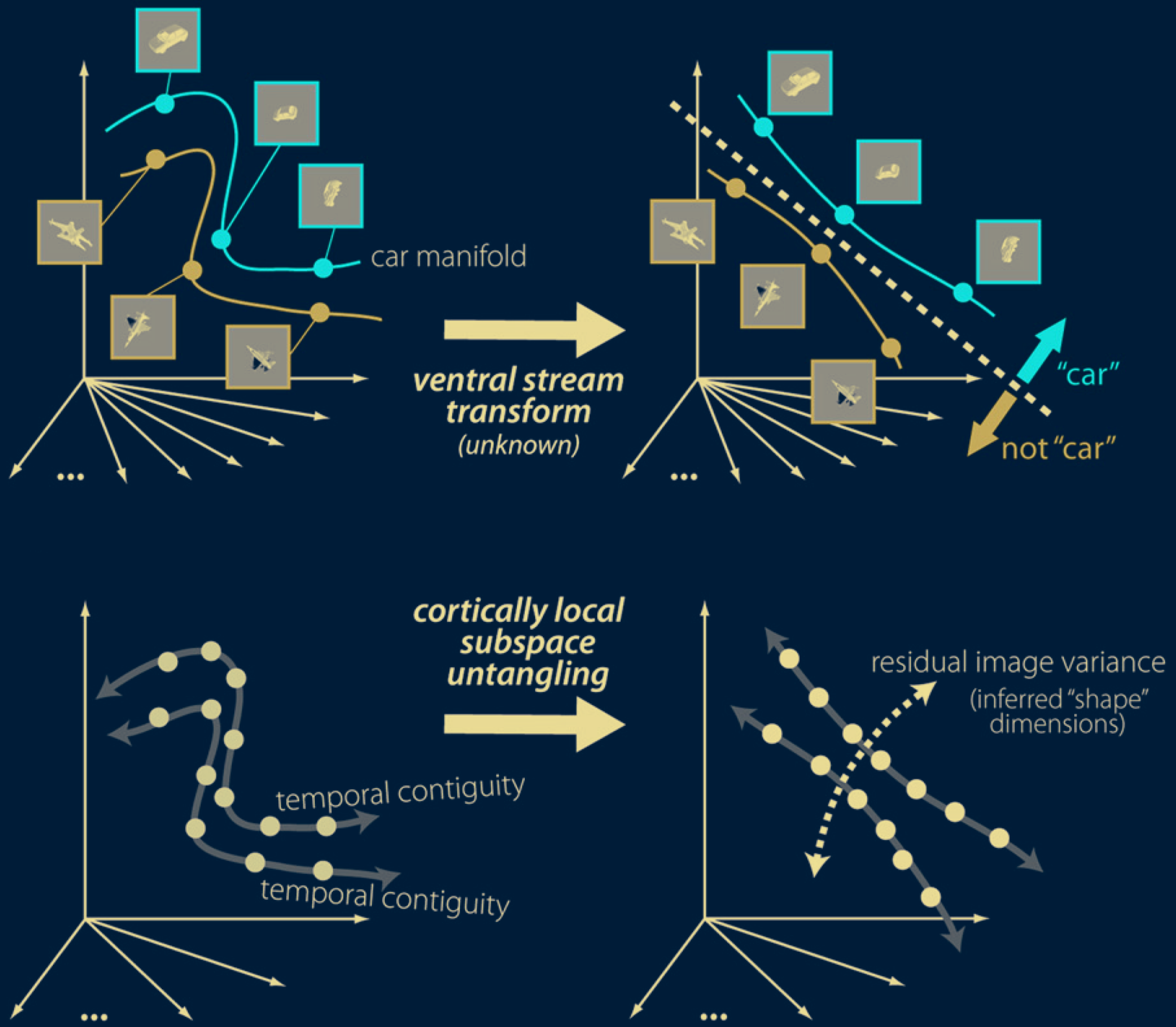
- Perform experiments (on a computer) that are impossible (experimentally or ethically) to be done on humans or animals.

- Eventually improve and test hypotheses about complex behaviors:
  Perception • Attention • Learning • Memory • Consciousness • Sleep and wakefulness

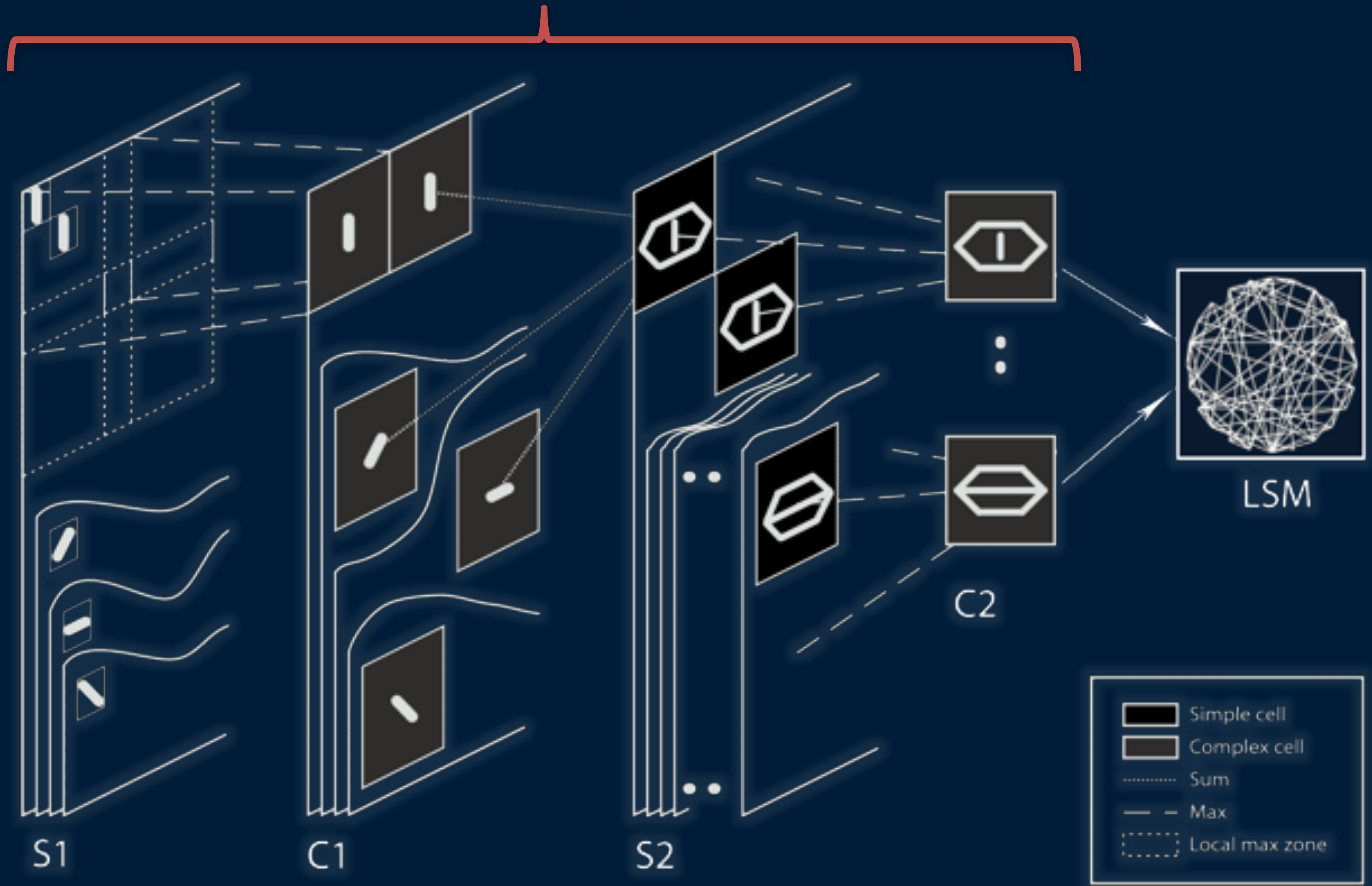




DiCarlo, James J., Davide Zoccolan, and Nicole C. Rust. (2012). How does the brain solve visual object recognition? Neuron


car manifold
ventral stream
transform
(unknown)
"car"
not "car"
cortically local
subspace
untangling
temporal contiguity
temporal contiguity
residual image variance
(inferred "shape"
dimensions)

Experiment | Model Schema



Dataset

- **Evaluated the neural network on two Caltech datasets (www.vision.caltech.edu):**
  - **Faces 1999**
  - **Motor-cycles 2001**

- **All images were rescaled to be 300 pixels in height (preserving the aspect ratio) and converted to gray-scale values.**

Four layers:

- S_1:
  - detect edges by performing a convolution on the input images
  - 5x5 convolution kernels corresponding Gabor filters
  - four preferred orientations: $\frac{\pi}{8}, \frac{\pi}{4}+\frac{\pi}{8}, \frac{\pi}{2}+\frac{\pi}{8}, \frac{3\pi}{4}+\frac{\pi}{8}$
  - five scaled versions of original image: 100%, 71%, 50%, 35% and 25%
  - 4 x 5 = 20 S_1 maps
  - spikes with a latency that is inversely proportional to the absolute value of the convolution
  - Limiting activity: at a given processing scale and location only the spike corresponding to the best matching orientation is propagated
- C_1
- S_2
- C_2

Four layers:

- S_1
- C_1:
  - Subsampling S_1 maps (36 times fewer C_1 cells)
  - propagate the first spike emitted by S1 cells in a 7x7 square of a given S1 map (corresponding to one preferred orientation and one processing scale
  - Max operation is a biologically plausible way to gain local shift invariance
  - Lateral inhibition: Upon emitting a spike, the latency of neighbors within a 11x11 square in the map is increased (latency decreases with distance from the spike)
- S_2
- C_2

Four layers:

- S_1
- C_1
- S_2:
  - intermediate complexity visual features
  - Each prototype cell is duplicated in 5 maps (weight sharing), each map corresponding to one processing scale
  - Within those maps the S2 cells can only integrate spikes from the four C1 maps of the corresponding processing scale
  - RF size is 16 x 16 C1 cells
  - C1-S2 synaptic connections are set by STDP
  - No leakage in neuron model
  - kWTA ensures that at most two cells can fire for each processing scale (only during learning phase). Without it there would be a bias toward small patterns (large scales)
- C_2

Four layers:

- S_1
- C_1
- S_2
- C_2:
  - take the maximum response (i.e. first spike) for each prototype S2 cells over all positions and processing scales
  - shift and scale invariant cells

- Integrate and Fire (IF) neural model is used as building blocks for the hierarchical network
- No Leakage needed!
- Synapses are current based (CUBA)
- A simplified STDP is used for learning.

For our task we have restricted the feature to only four!
The hierarchical network is tested across various parameters:

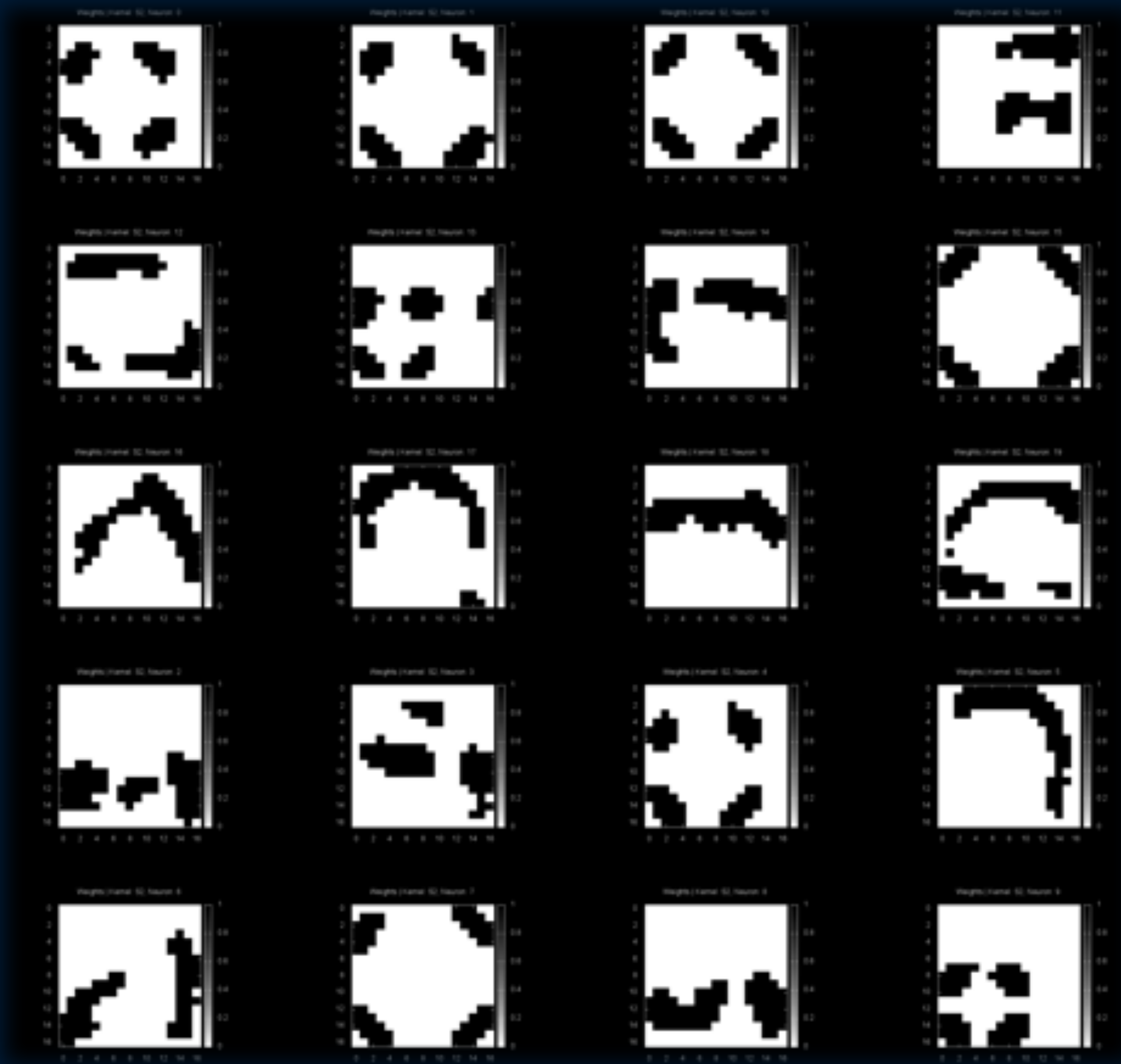
Parameter	Values
SRF	15, **17**
Threshold	100, 80, 65, **30**, 10
Epochs	20, 10, **5**

Masquelier and Thorpe (2007). Unsupervised learning of visual features through spike timing dependent plasticity. PLoS Comput Biol

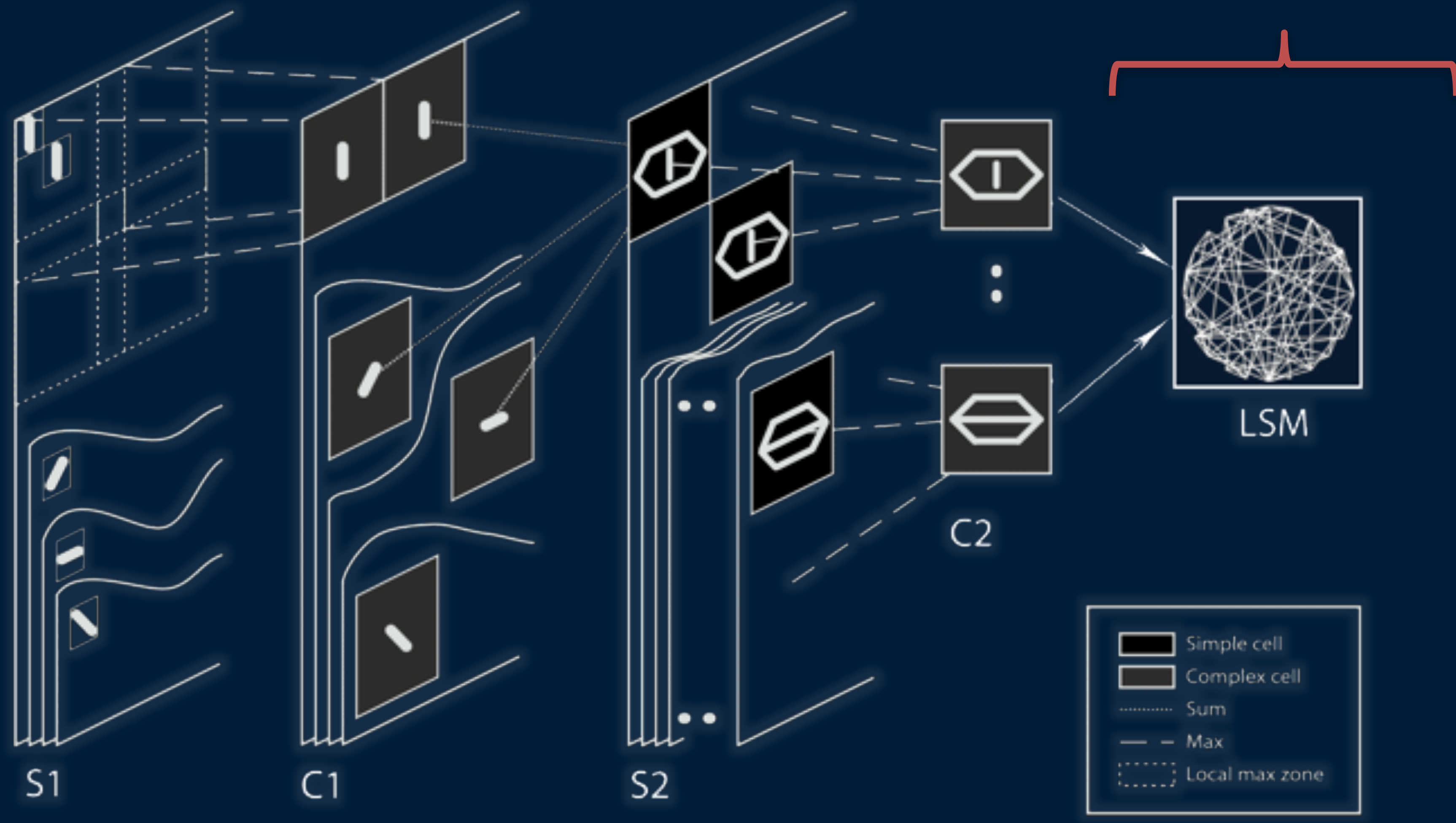
A simplified STDP rule:

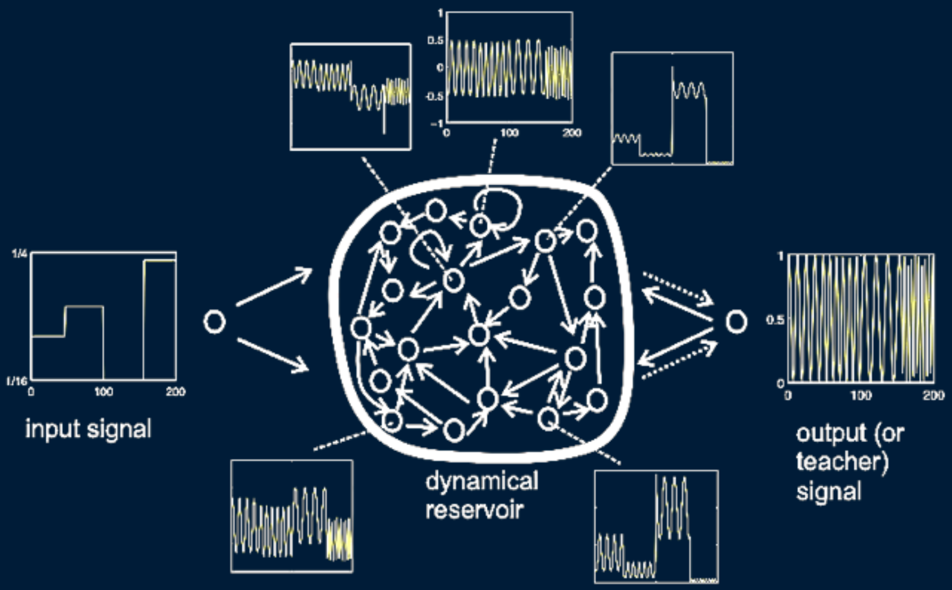
$$\Delta w_{ij} = \begin{cases} a^{+}.w_{ij}.\left(1 - w_{ij}\right) & \text{if} \quad t_j - t_i \leq 0 \qquad (LTP) \\ a^{-}.w_{ij}.\left(1 - w_{ij}\right) & \text{if} \quad t_j - t_i > 0 \qquad (LTD) \end{cases}$$

- **With this rule only the order of the spikes matters, not their precise timings**
- **As a result the intensity-latency conversion function of S1 cells has no impact, any monotonously decreasing function give the same results**
- **a^+ and $|a^-|$ are increased during simulation as STDP learning progresses (multiplying by 2 every 400 postsynaptic spikes)**



Experiment | Model Schema



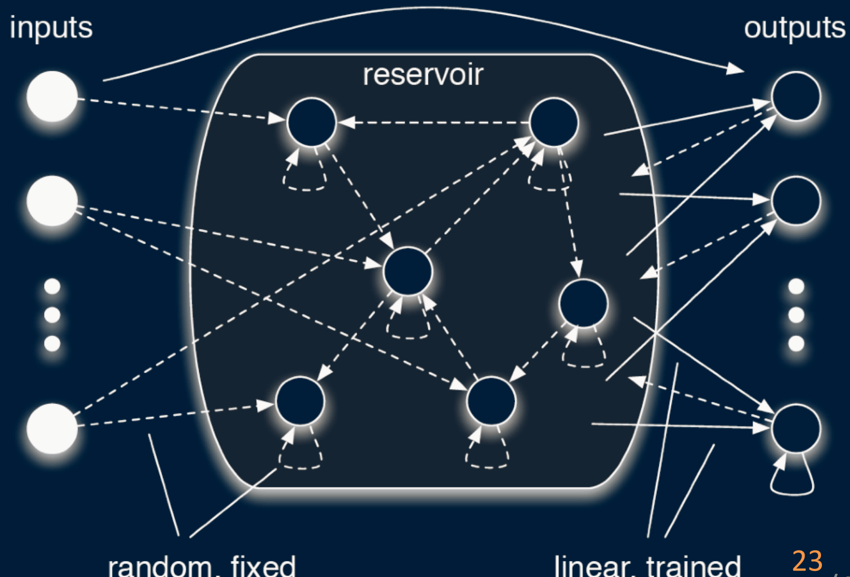


Principles of RC

- Use large, fixed, random recurrent network as excitable medium
- Excite by input signal
- Read out desired output

- Also known as *Echo State Networks* and *Liquid State Machines*
- Discovered in 2000, now an established paradigm in computational neuroscience and machine learning
- RC makes, for the first time, training of recurrent neural networks practically feasible: a major enabling technology
- RC is biologically plausible

- Recurrent structures without the training: Reservoir Computing
  - Jaeger (2001): Echo State Networks (engineering)
  - Maass (2002): Liquid State Machines (neuroscience)
  - Steil (2003): weight dynamics of Atiya-Parlos equivalent
- Fixed, random topology operated in correct dynamic regime
- Different node types possible: THG, linear, tanh, spiking, …
- Linear "readout" function which is trained (No local minima, no problems with recurrent structure, one shot learning)
- On-line computing: prediction at every time-step
- Any time-invariant filter with fading memory can be learned (with output feedback, universal computing)



Leaky Integrate and Fire model is used for this network.

$$\tau_m \frac{dV_m}{dt} = -(V_m - v_{rest}) + R_m(I_{syn} + I_{inject} + I_{noise})$$

The refractory period for Exc and Inh neurons is set to 3ms and 2ms, respectively.

Dynamic synapses (short-term plasticity) are connecting neurons to each other.

Synaptic weights are static and initialized to $W \times W_{scale}$, where W is randomly drawn from a Normal distribution as below:

	μ	σ
EE	3×10^{-8}	half of μ
EI	6×10^{-8}	
IE and II	-1.9×10^{-8}	

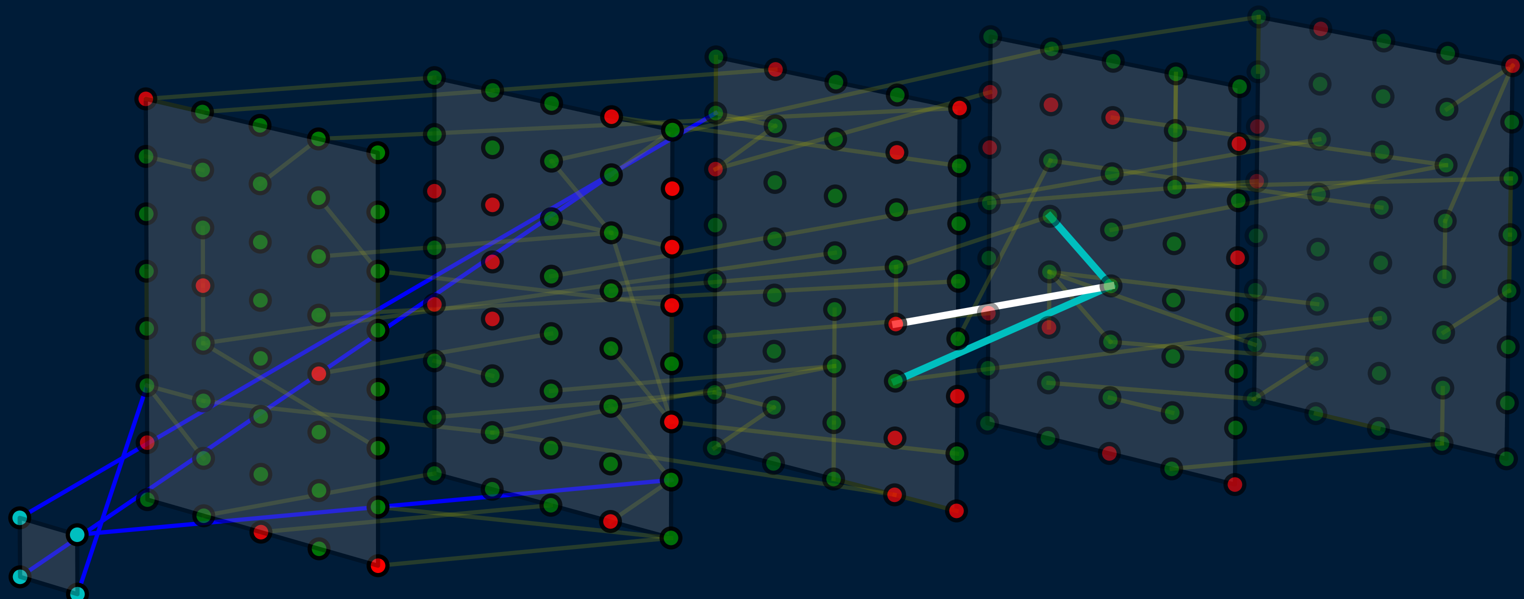
Maas et al. (2002). Real-time computing without stable states: A new framework for neural... Neural computation

The original connection technique proposed by Maas for Liquid State Machines:

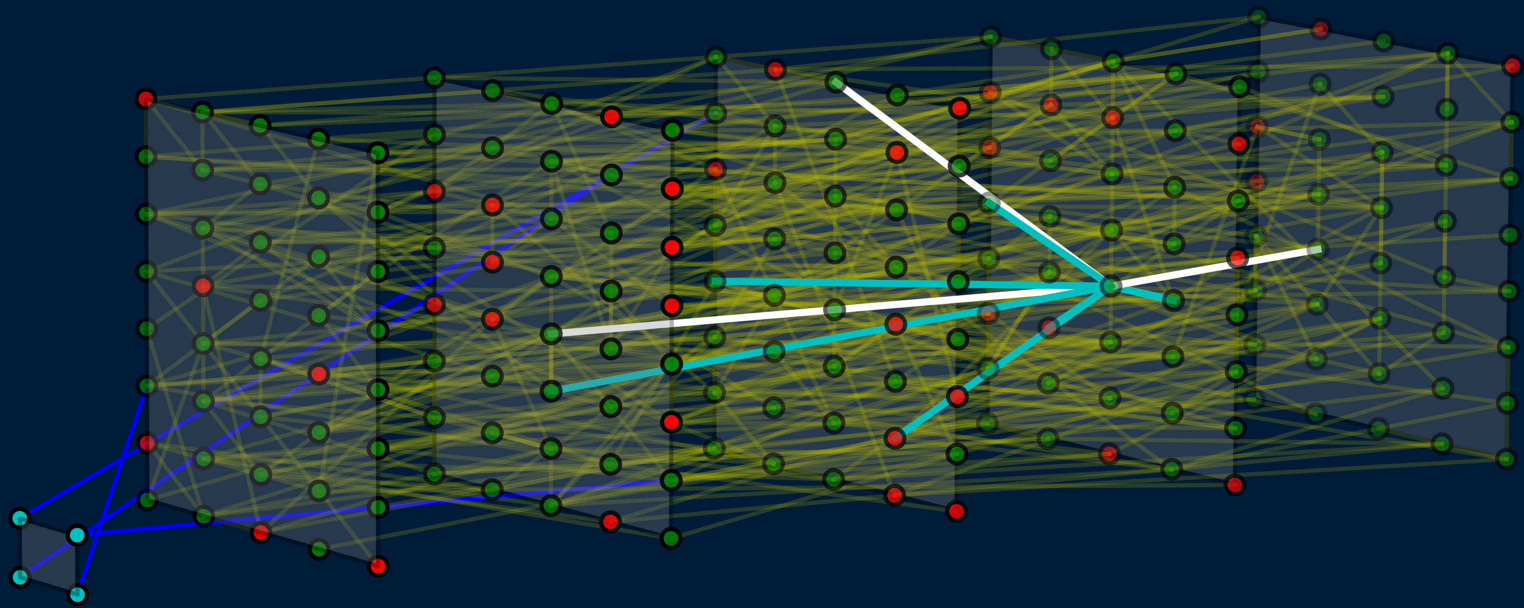
$$p = C.\exp[-(\frac{D(a,b)}{\lambda})^2]$$

Where
D(a, b) is the Euclidean distance between neurons *a* and *b*,
λ is a connection parameter
and C is set depending on whether neurons are inhibitory (I) or excitatory (E):

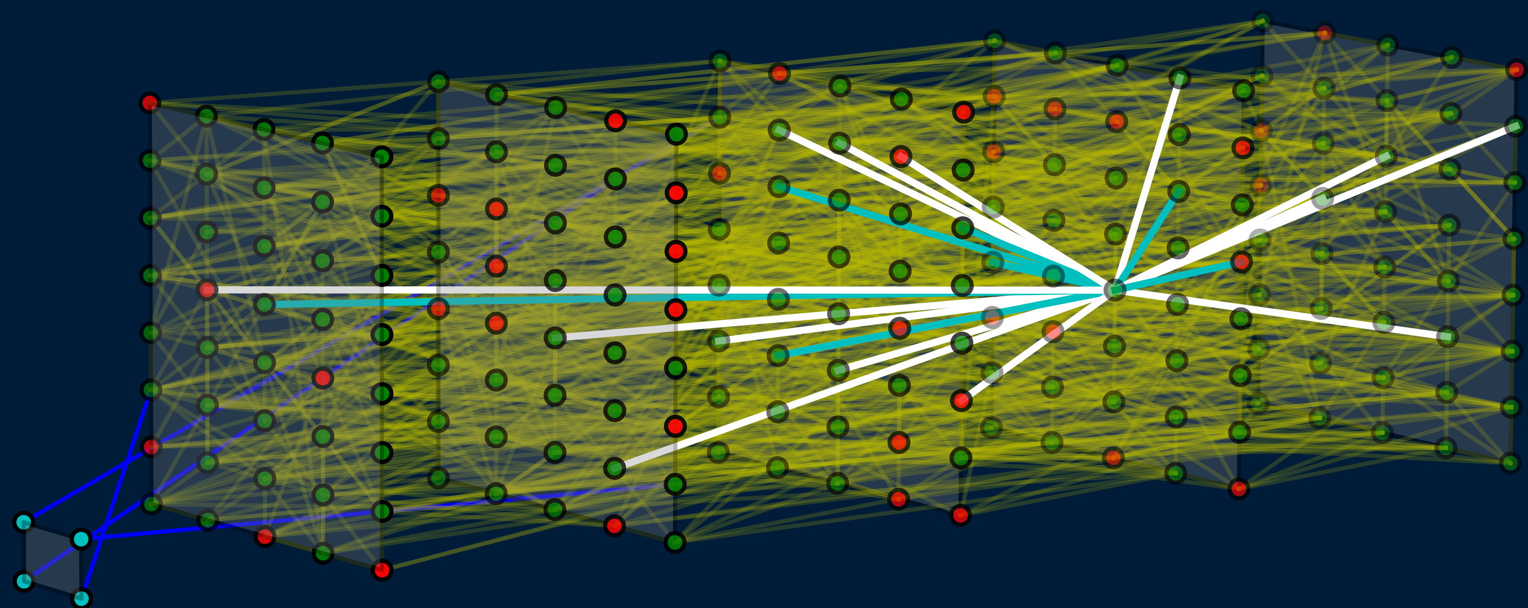
```
0.3    (EE)
0.2    (EI)
0.4    (IE)
0.1    (II)
```



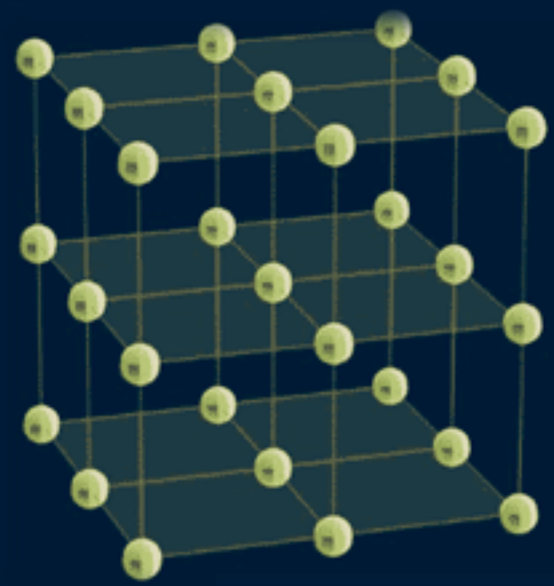
$\lambda = 0.5$



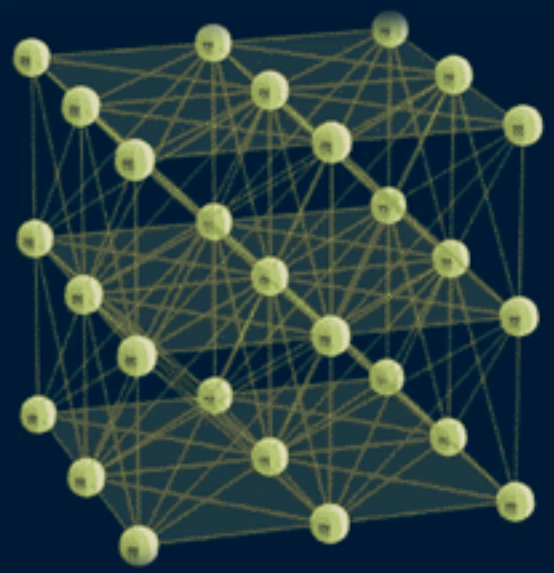
$$\lambda = 1.0$$



$\lambda = 1.5$



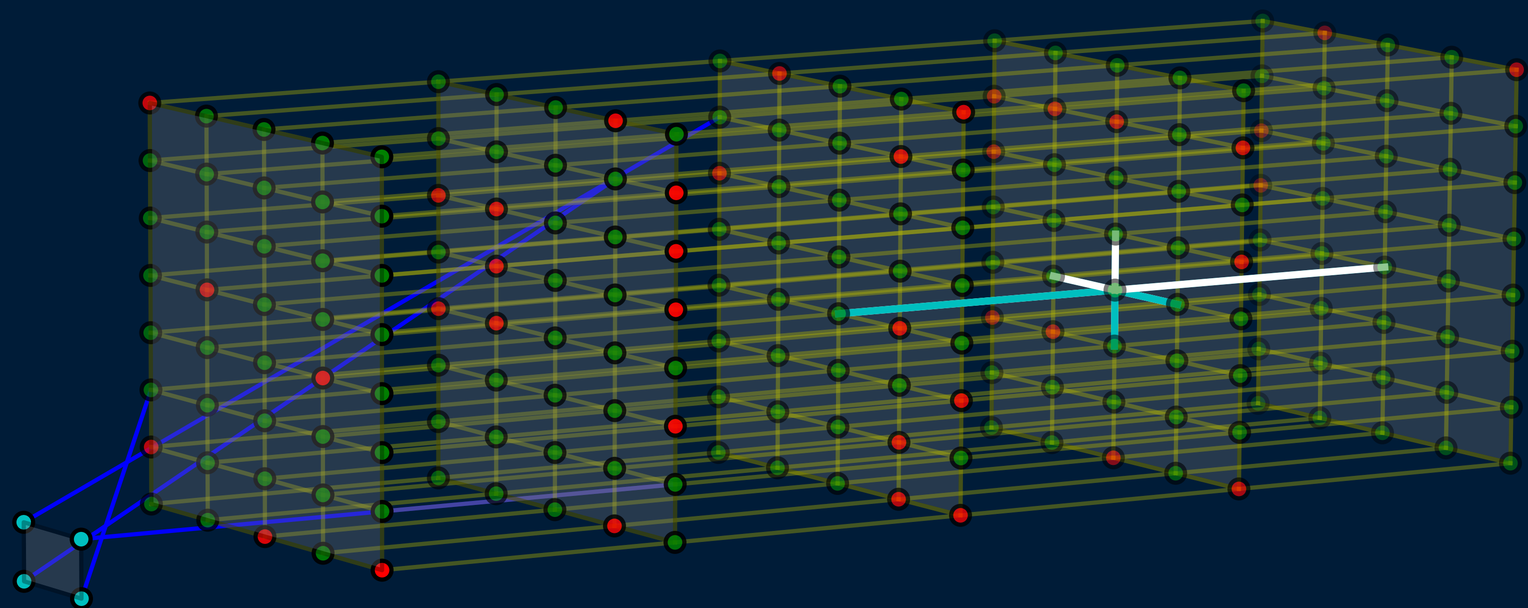
Lattice A



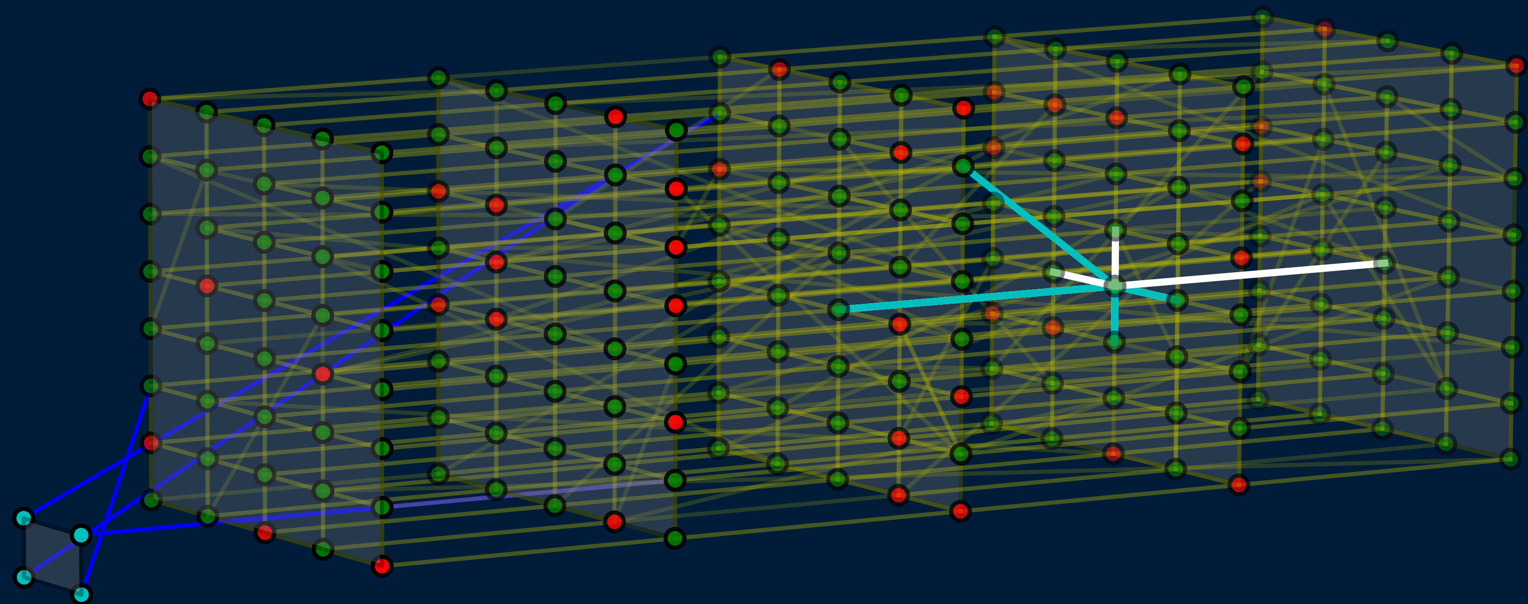
Lattice B

Watts–Strogatz Algorithm:

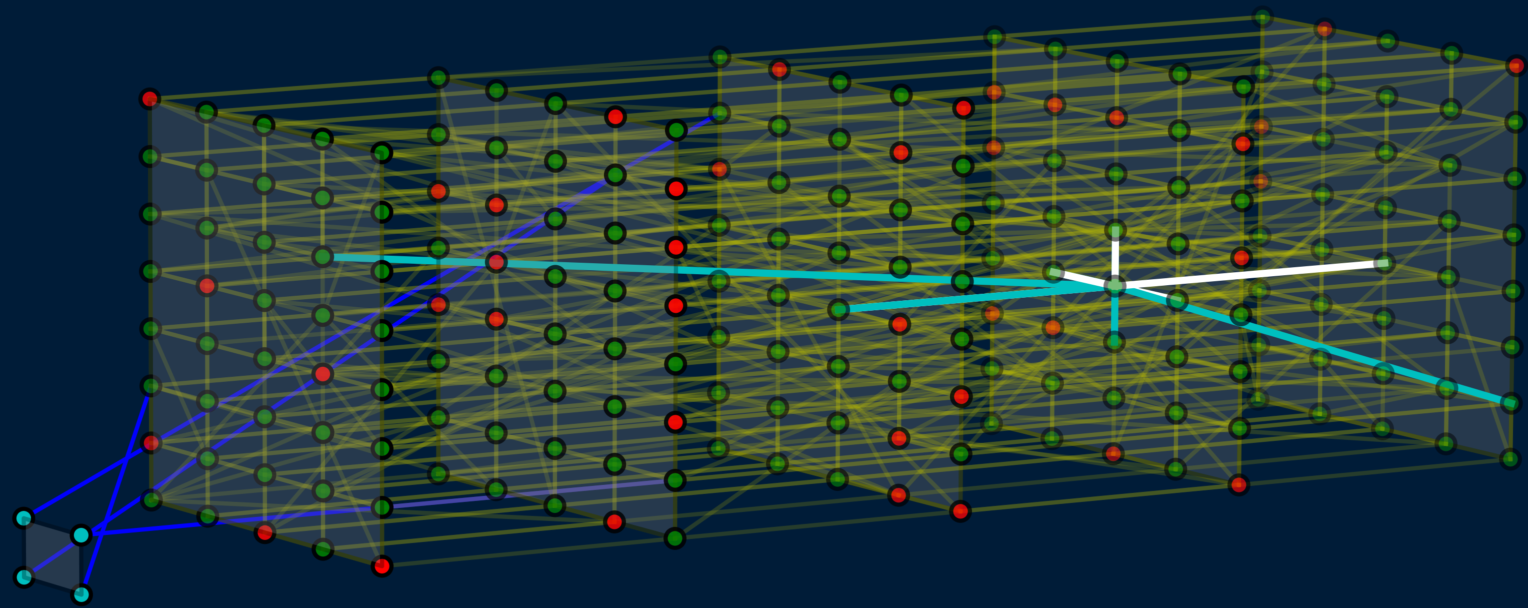
The network is created by removing each edge with uniform, independent probability *p* and rewiring it to yield an edge between a pair of nodes that are chosen uniformly at random.



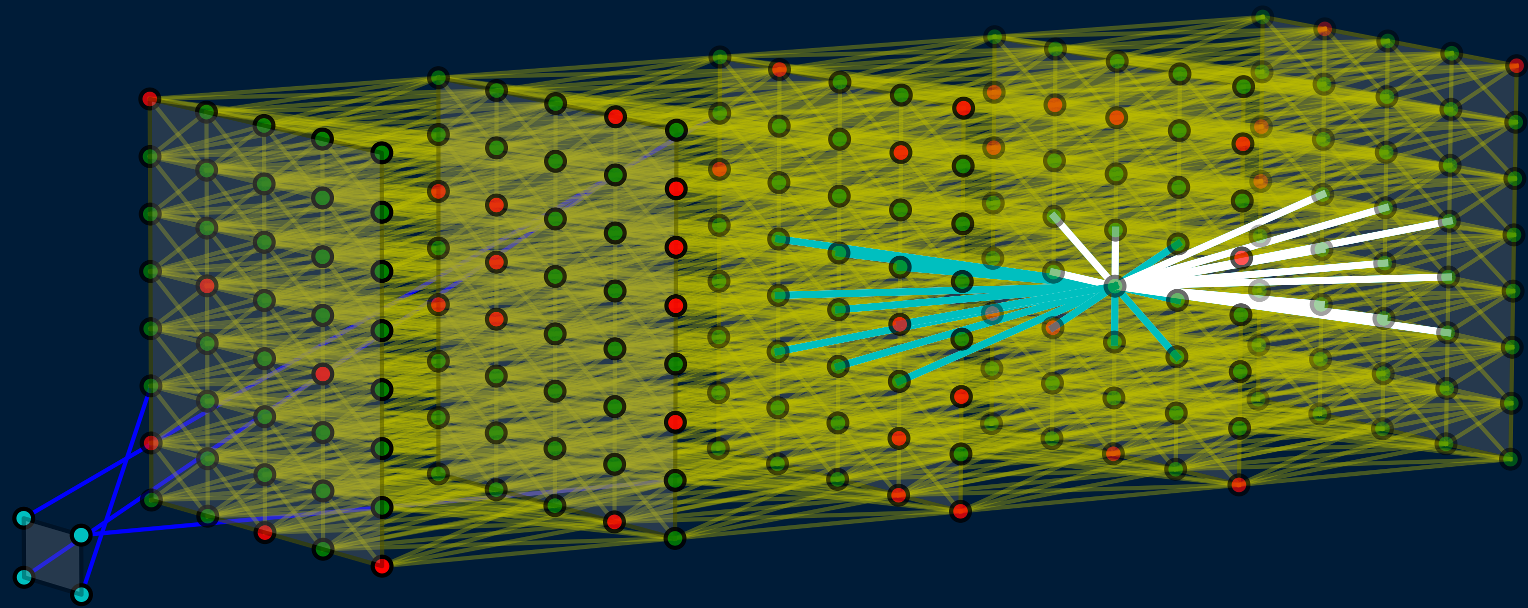
Type A, $P = 0$



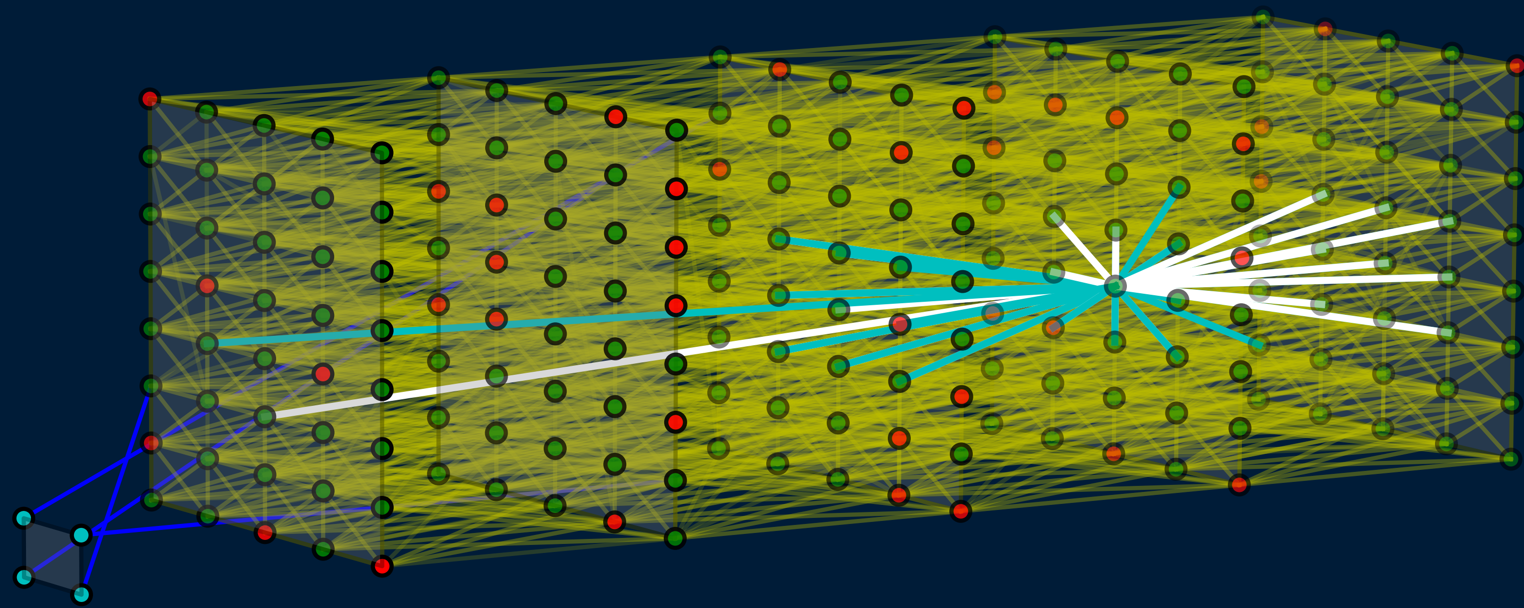
Type A, $P = 0.05$



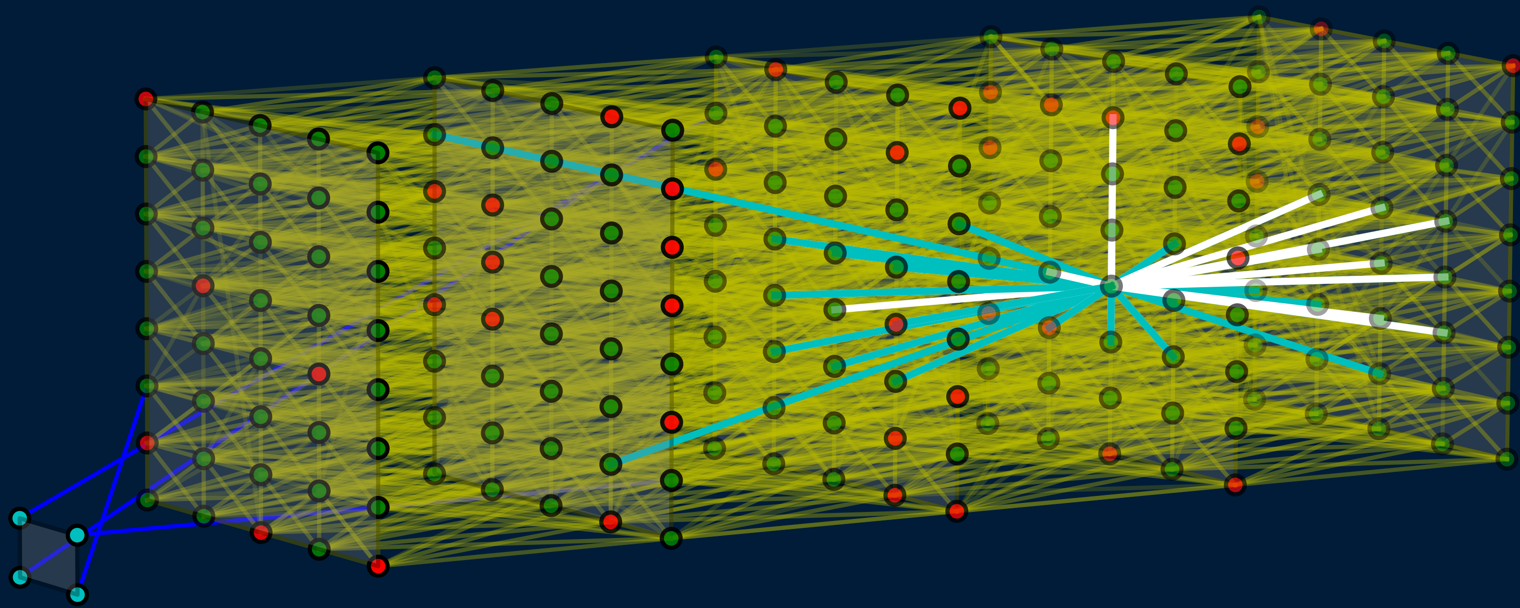
Type A, $P = 0.1$



Type B, $P = 0$

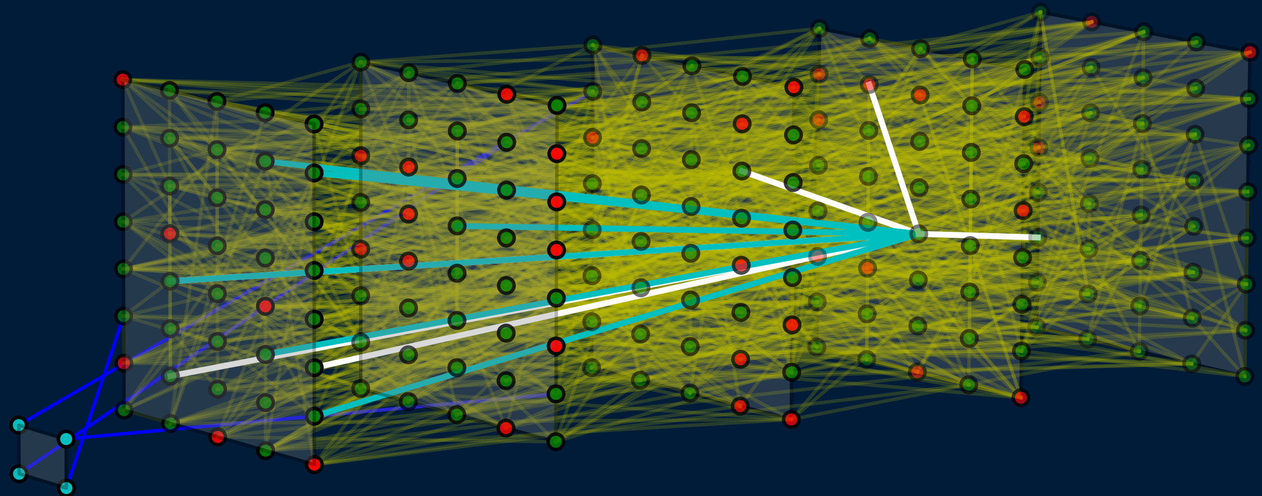


Type B, $P = 0.05$

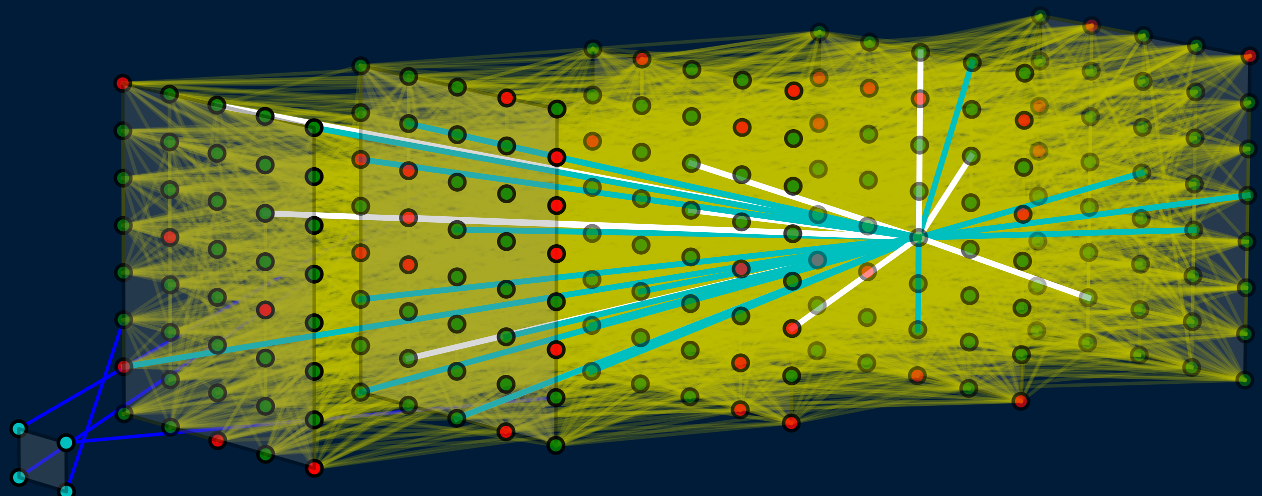


Type B, $P = 0.1$

An extreme case of small-world networks where P=1 is called Erdős–Rényi (fully random)



Type A, $P = 1$



Type B, $P = 1$

The state of Liquid Filter is calculated as below:

$$s_m^f(i) = \sum_n \exp(-\frac{t_i^n}{\tau})$$

Where τ is time constant equal to 50ms. t_i^n is time of *n*th spike for each stimulus.

This is as opposed to simply counting spikes which neglects timing of spikes within spike trains.

We used two widely-known linear classifier (that is LDA and SVM) as the readout layer to evaluate how well the Liquid Filter helps in distinguishing patterns.

We also propose a new separation criteria.

Using LDA and SVM has the drawback of performance dependence on an ML technique which is not inherent to reservoir computing scheme. It is also time consuming to use an actual classifier.

The classic way to assess separation capability of network is to use Fisher Discriminant Ratio:

$$S_b = \sum_{i=1}^{M} P_i(\mu_i - \mu_0)(\mu_i - \mu_0)^T$$

$$S_w = \sum_{i=1}^{M} P_i \Sigma_i$$

$$S_m = S_w + S_b$$

Trace of S_m calculates the variance of features across global mean while trace of S_w is sum of features variances for all classes

$$FDR = trace(S_w^{-1} S_m)$$

However we have not used FDR. As Liquid State may be comprised of collinear features, S_w might be a singular matrix, and therefore its inverse can not be calculated.
We may overcome this by adding a negligible number to main diagonal or use pseudo-inverse technique, both of which fail to fully depict the separation capability.

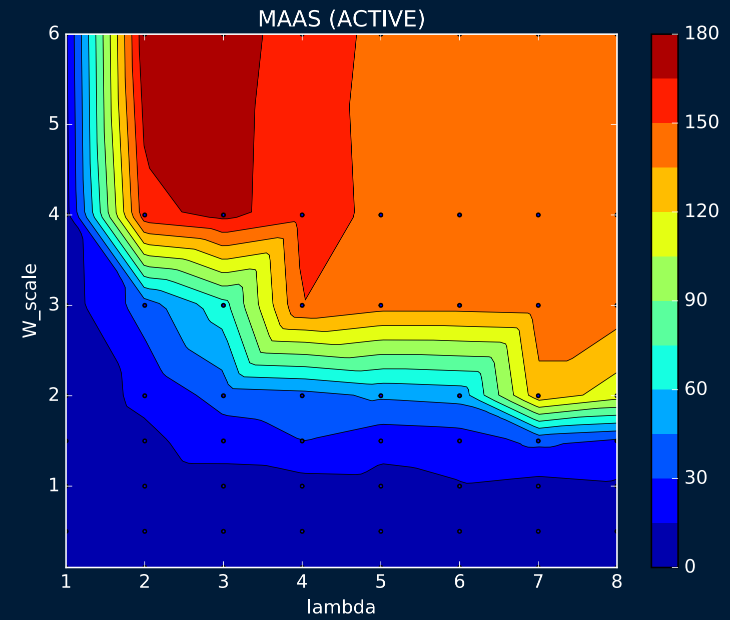
Other separation measure use Euclidean or Bray-Curtis distance. Based on previous works, we propose the following:

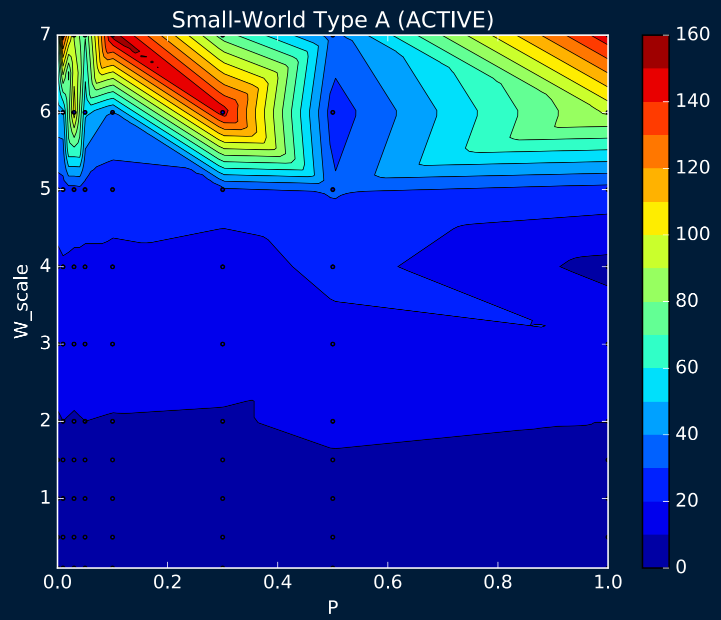
$$D_b = \frac{\sum_{i=1}^{M} \sum_{j=1}^{M} \left\| \mu_i - \mu_j \right\|_2}{M^2}$$

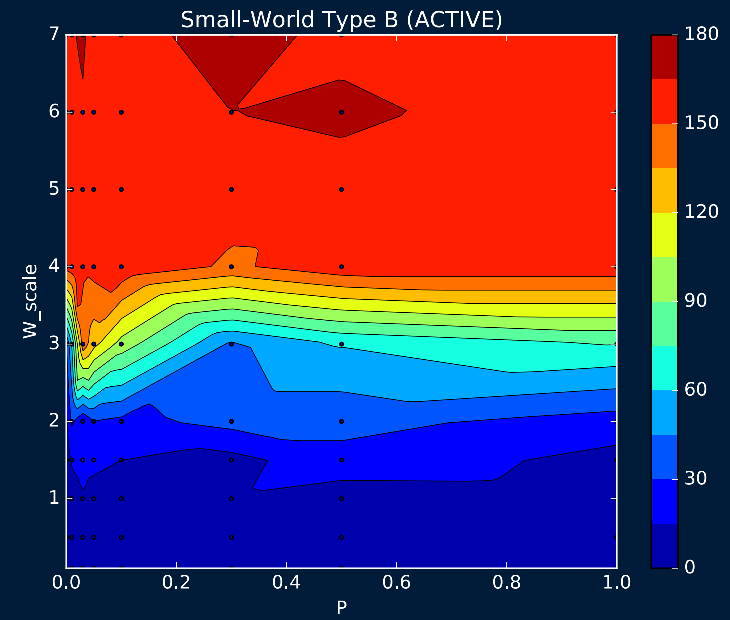
$$D_w = \sum_{i=1}^{M} \sum_{j=1}^{N} \sigma_{i,j}$$

$$Separation = \frac{D_b}{D_w + 1}$$

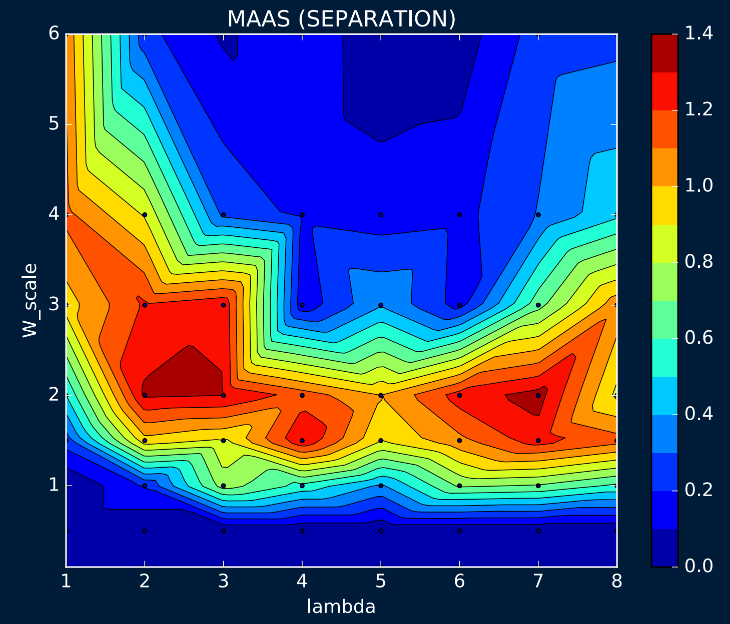
The plots are obtained by interpolation of a number of points (black dots) whose performance are calculated for the corresponding parameters.

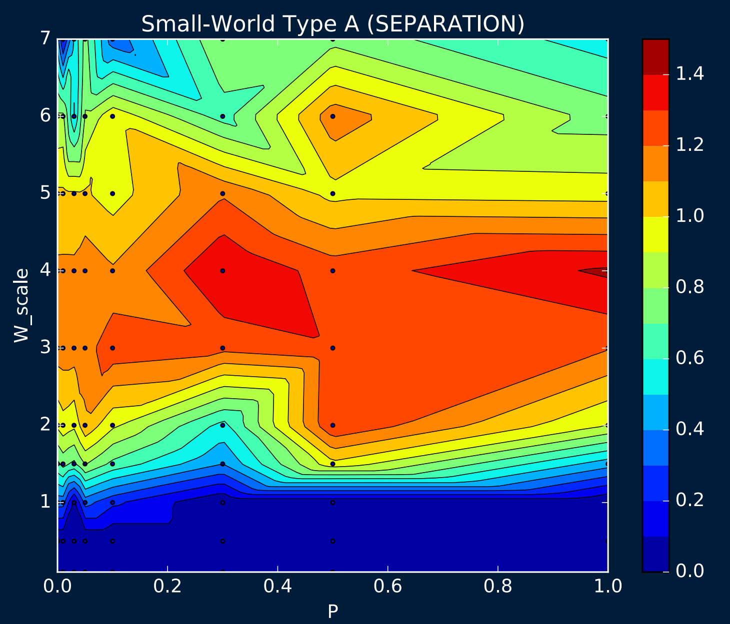


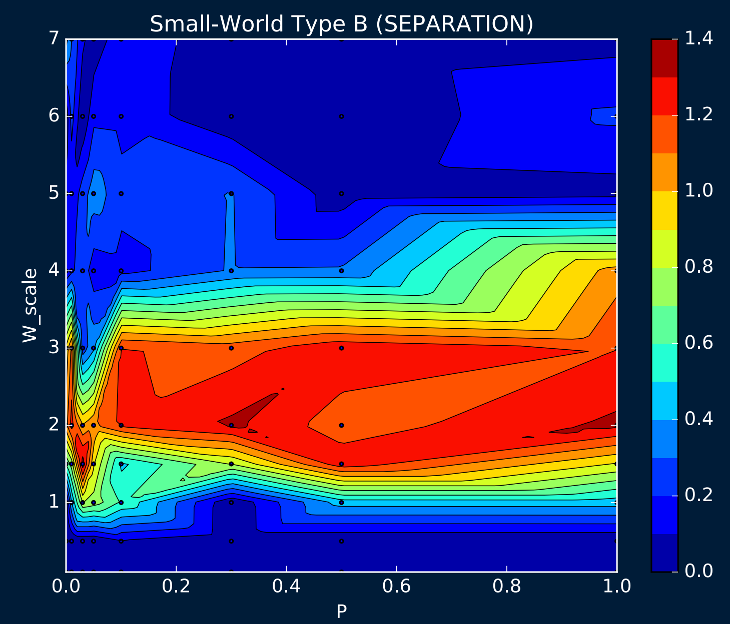




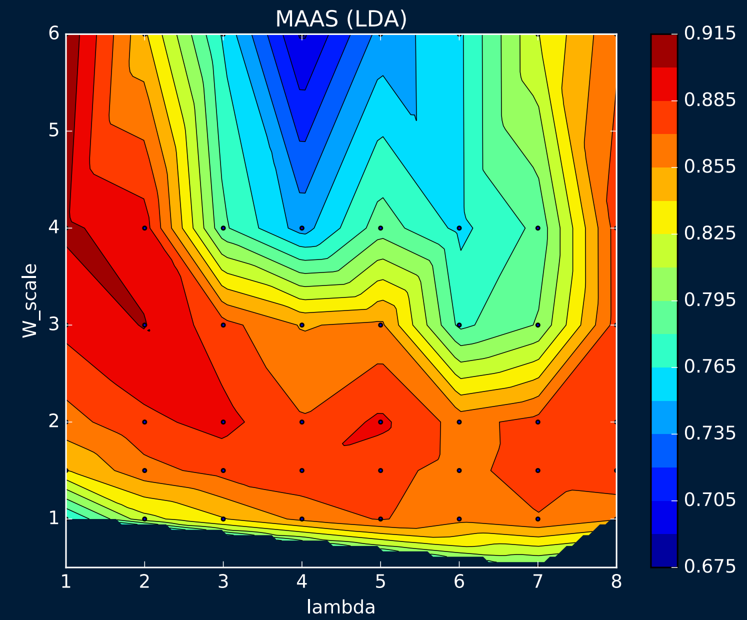
For each set of parameters the network has been evaluated six times.

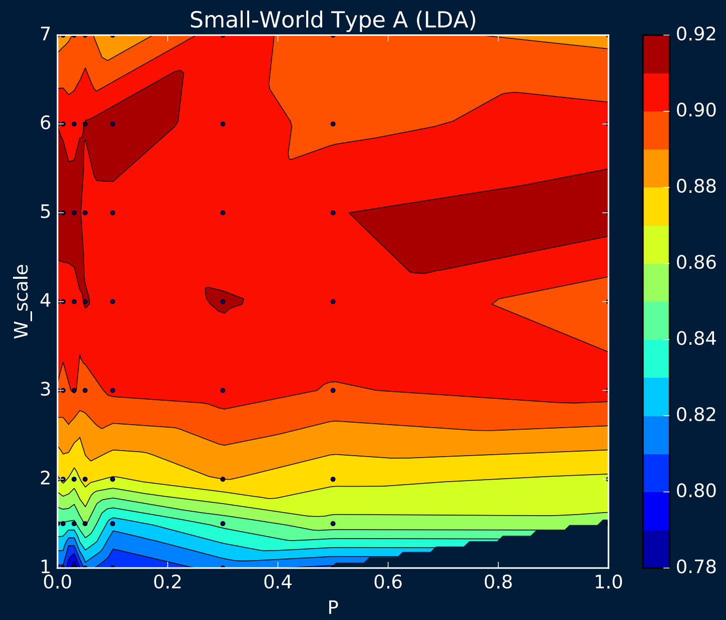


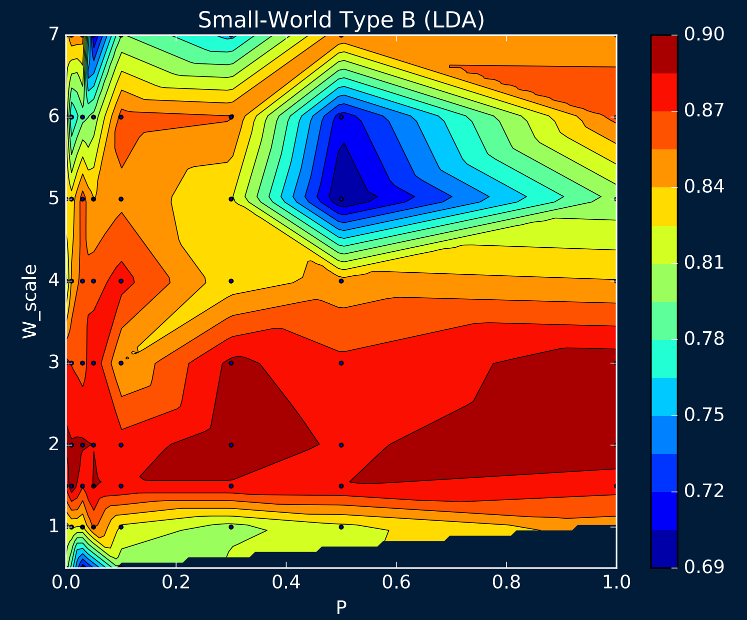




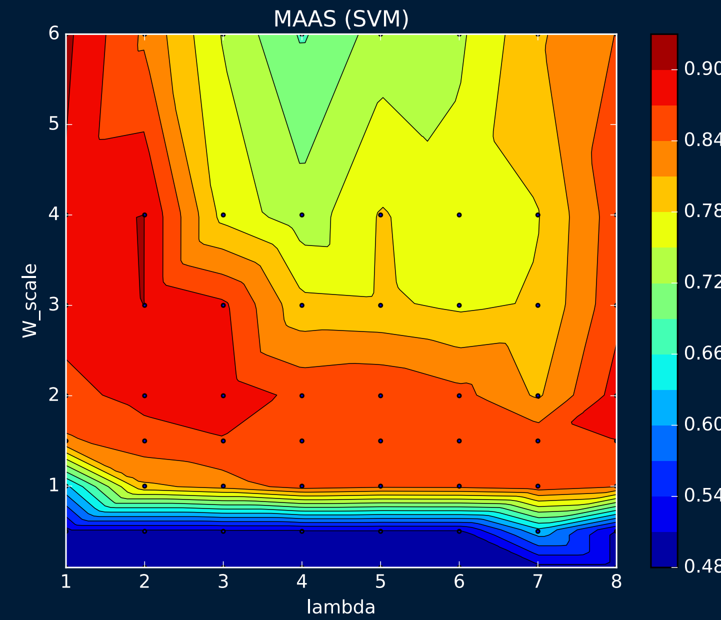
Results LDA


MAAS (LDA)
W_scale
lambda
1
2
3
4
5
6
7
8
0.915
0.885
0.855
0.825
0.795
0.765
0.735
0.705
0.675

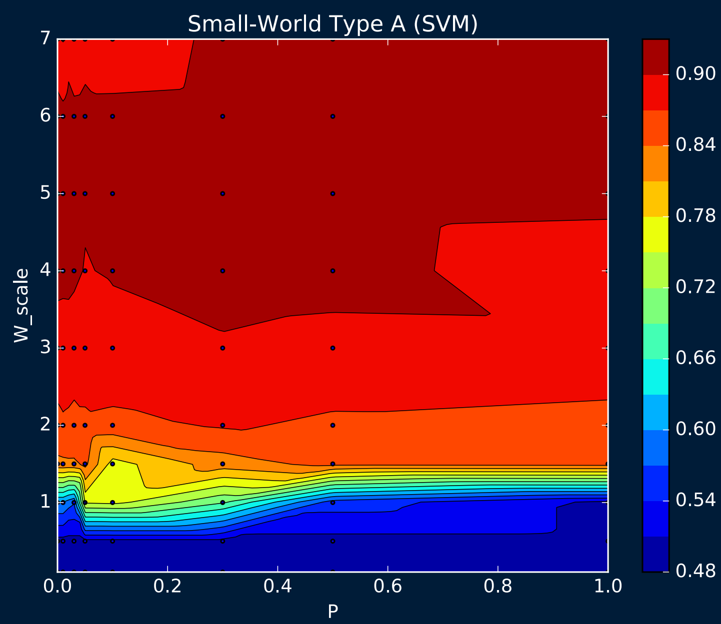

Small-World Type A (LDA)
W_scale
P
0.0
0.2
0.4
0.6
0.8
1.0
0.92
0.90
0.88
0.86
0.84
0.82
0.80
0.78


Small-World Type B (LDA)
W_scale
P
0.0
0.2
0.4
0.6
0.8
1.0
0.90
0.87
0.84
0.81
0.78
0.75
0.72
0.69

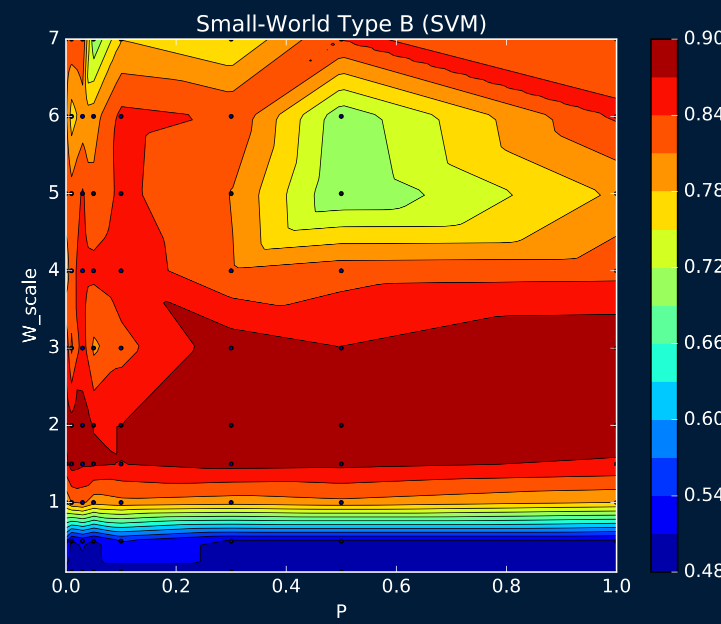
Results | SVM



MAAS (SVM)
W_scale
lambda
1
2
3
4
5
6
7
8
0.90
0.84
0.78
0.72
0.66
0.60
0.54
0.48



Small-World Type A (SVM)
W_scale
P
0.0
0.2
0.4
0.6
0.8
1.0
0.90
0.84
0.78
0.72
0.66
0.60
0.54
0.48



Small-World Type B (SVM)
W_scale
P
0.0
0.2
0.4
0.6
0.8
1.0
0.90
0.84
0.78
0.72
0.66
0.60
0.54
0.48

Connection Topology	Separation	LDA	SVM
Maas	1.34	92.5	91.8
SW-A	1.41	92.6	92
SW-B	1.36	92.4	91.6

Future Works

- Incorporating multiple readouts and partitioning liquid state to overlapping or non-overlapping smaller snapshots.

- There are plenty more topologies to consider (scale-free, etc.). There are also technique to mimic non-spherical directional axonal growth of synapses.

- LSMs generally do not incorporate synaptic plasticity. We may still investigate the possibility of evolutionary finding the best connection topology by dynamically adding/removing synapses during simulation.

- No work done on motion pictures! LSMs may be applicable for motion-related tasks (dorsal stream) as they are appropriate for time-series.

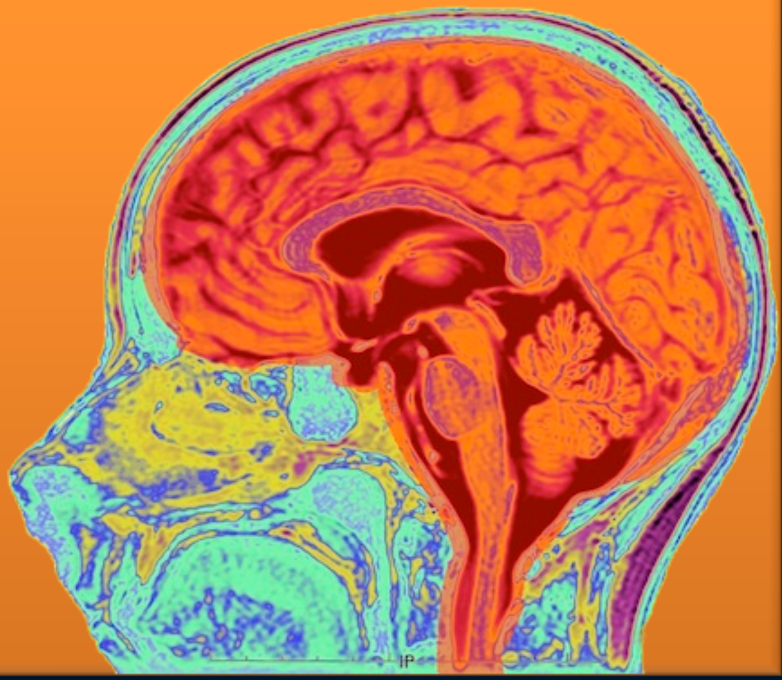
- More intricate neuron models (e.g. Hodgkin-Huxley or Izhikevich) may be investigated for more biologically plausible models. Polychronization is also interesting!

Selected References

1. Kaiser, Marcus, Hilgetag, Claus C, and Van Ooyen, Arjen. A simple rule for axon outgrowth and synaptic competition generates realistic connection lengths and filling fractions. Cerebral Cortex, 19(12):3001–3010, 2009.
2. Hourdakis, Emmanouil and Trahanias, Panos. Improving the classification performance of liquid state machines based on the separation property. in Engineering Applications of Neural Networks, pp. 52–62. Springer, 2011.
3. Wojcik, Grzegorz M and Ważny, Marcin. Bray-curtis metrics as measure of liquid state machine separation ability in function of connections density. Procedia Computer Science, 51:2979–2983, 2015.
4. Masquelier, Timothée, and Simon J. Thorpe. "Unsupervised learning of visual features through spike timing dependent plasticity." PLoS Comput Biol 3.2 (2007): e31.
5. Maass, Wolfgang, Thomas Natschläger, and Henry Markram. "Real-time computing without stable states: A new framework for neural computation based on perturbations." Neural computation 14.11 (2002): 2531-2560.
6. Ju, Han, et al. "Effects of synaptic connectivity on liquid state machine performance." Neural Networks 38 (2013): 39-51.
7. Hourdakis, Emmanouil, and Panos Trahanias. "Improving the classification performance of Liquid State Machines based on the separation property." Engineering Applications of Neural Networks. Springer Berlin Heidelberg, 2011. 52-62.

Thanks for your attention :)

Any Questions?!



Neural Modeling

- Principle of hierarchical reductionism: study smaller and smaller parts and attempt to understand the whole .

- While it remains uncertain whether a brain system can be understood as the interaction between independently describable subsystems, the brain does display a hierarchy of spatial scales with repeated structure such as molecules, synapses, neurons, microcircuits, networks, regions and systems.

- How the advent of modeling in computational neuroscience opens the possibility to study the dynamics of models simultaneously incorporating various levels from molecules to regions?