

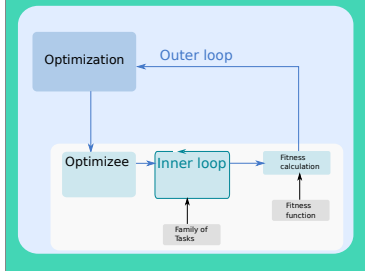
# Optimizing Spiking Neural Networks with L2L on HPC systems

## End of year colloquium

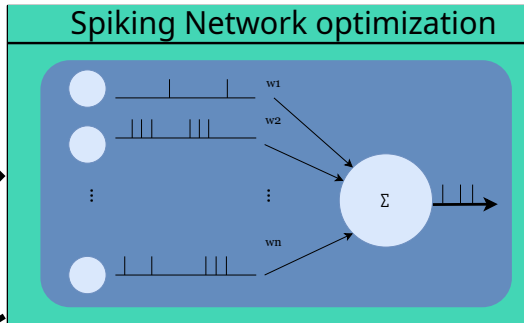
December 8, 2022 | **Alper Yeğenoğlu**<sup>1,2</sup> |

- 1. SDL Neuroscience, Jülich Supercomputing Centre (JSC), Forschungszentrum Jülich
- 2. Institute of Geometry and Applied Mathematics, Department of Mathematics, RWTH Aachen|  
[a.yegenoglu@fz-juelich.de](mailto:a.yegenoglu@fz-juelich.de)

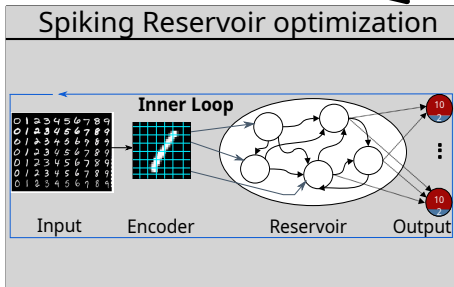
# Learning to Learn



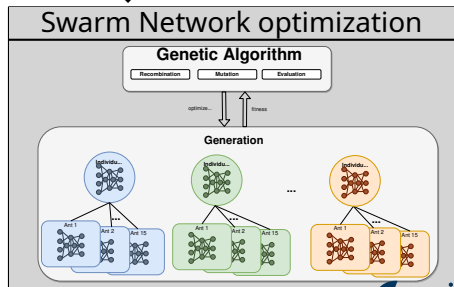
# Spiking Network optimization



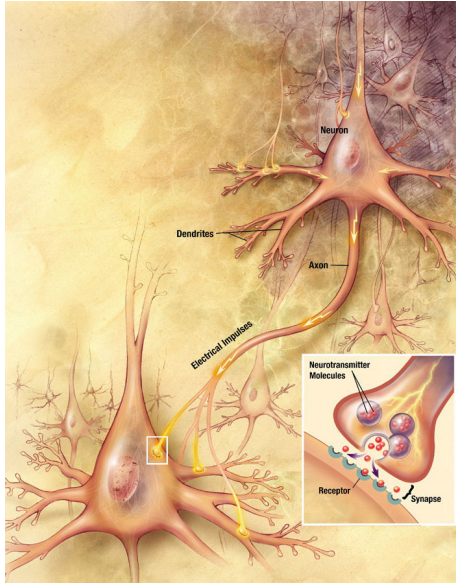
# Spiking Reservoir optimization



# Swarm Network optimization



# Neurons and Action Potentials



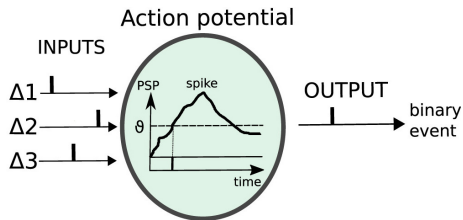
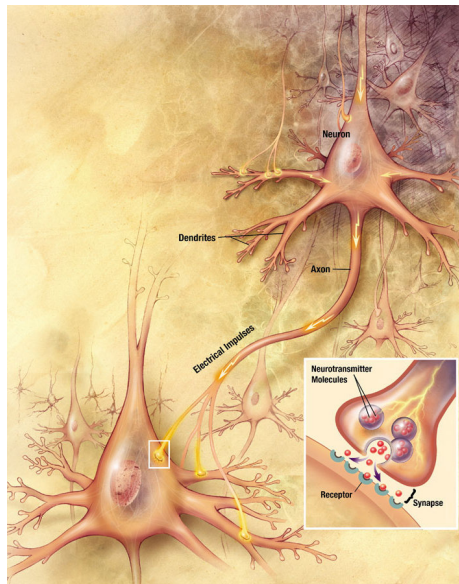
[US National Institute of Health]

Member of the Helmholtz Association

December 8, 2022

Slide 2

# Neurons and Action Potentials



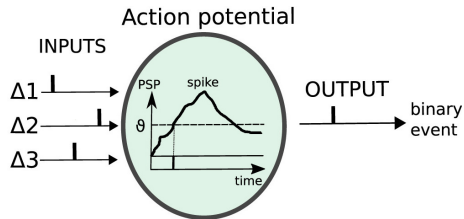
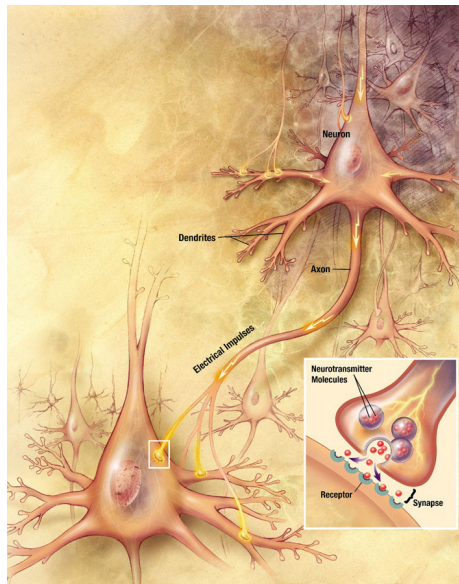
[US National Institute of Health]

Member of the Helmholtz Association

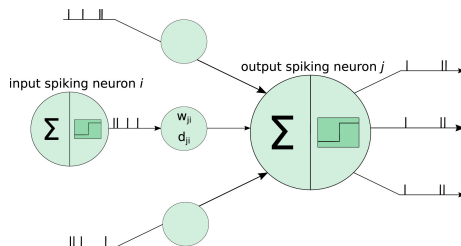
December 8, 2022

Slide 2

# Neurons and Action Potentials



## Spiking Neural Network (SNN)

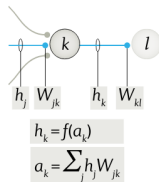
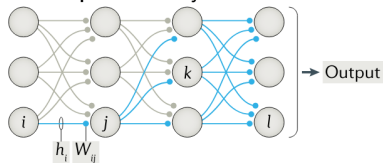


[Lobo et al., Neural Networks 121, 2020]

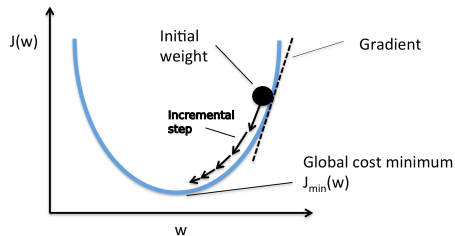
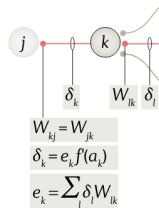
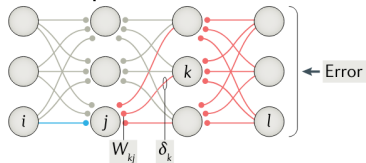
# Gradient Descent not applicable on SNNs

## ANN

### Forward pass of activity



### Backward pass of errors



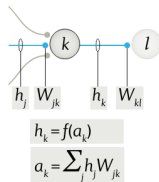
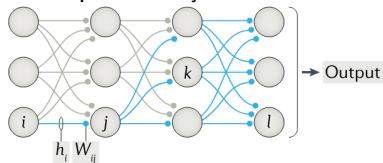
<https://rasbt.github.io/mlxtend>

[Lillicrap et al., Nature Reviews NS, 2020]

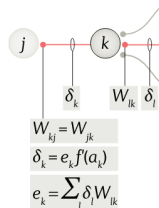
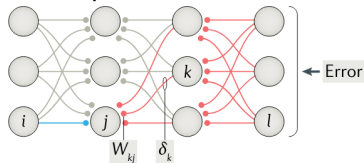
# Gradient Descent not applicable on SNNs

## ANN

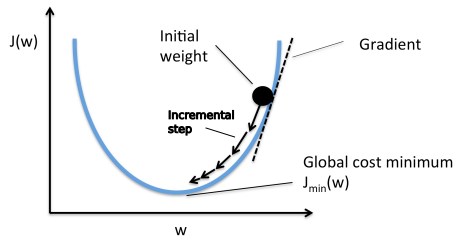
Forward pass of activity



Backward pass of errors

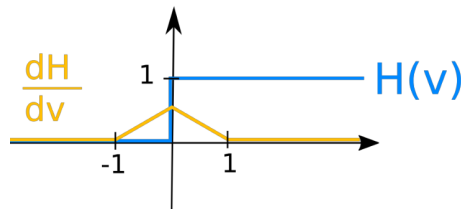


[Lillicrap et al., Nature Reviews NS, 2020]



<https://rasbt.github.io/mlxtend>

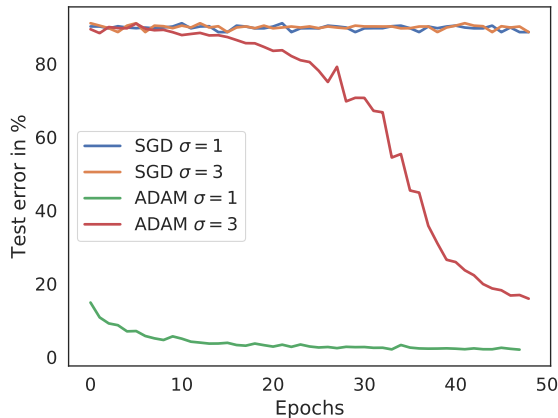
## SNN



Step function over spike. Gradient descent and backprop not possible.

# Gradient Descent Issues with ANNs

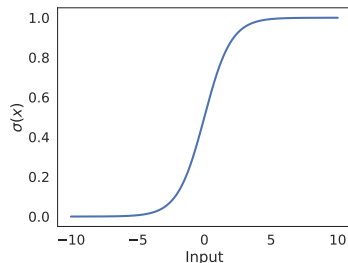
- Problem of Vanishing and Exploding Gradients
- In backpropagation step  $\rightarrow$  zero or huge gradients



[Yegenoglu et al., LOD 2020]

Problem depends on:

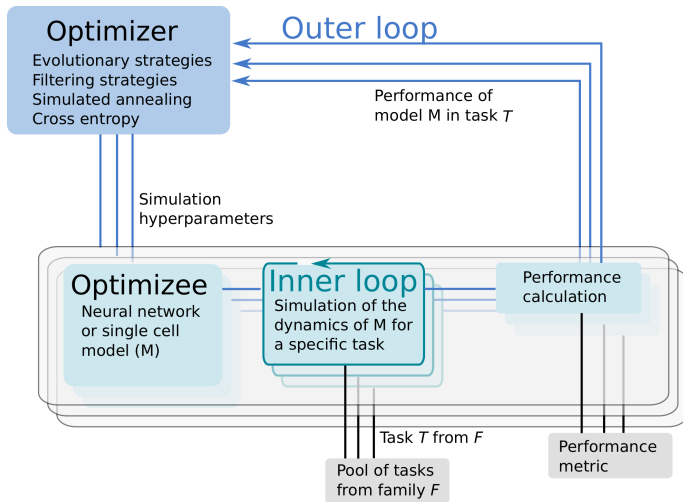
- Initialization of weights e.g.  $w_{i,j} \sim \mathcal{N}(0, 1)$
- Activation Functions



Logistic Function:  $\sigma(x) = \frac{1}{1+e^{-x}}$



# Optimization with Learning to Learn

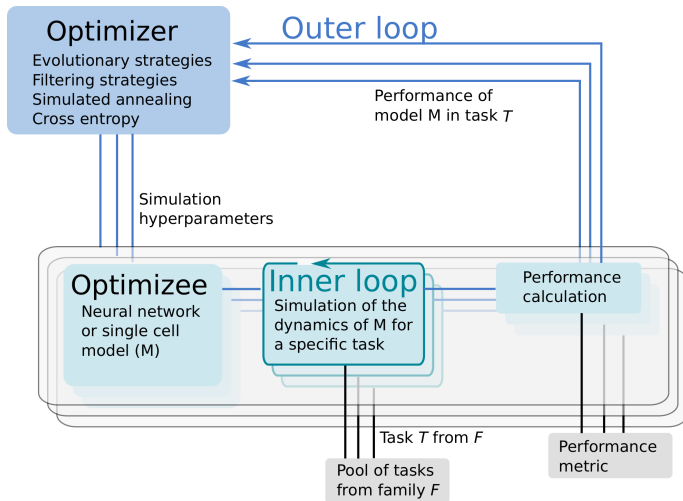


## Learning to Learn (L2L)

- Generalization on new data sets via experience
- Parameter space exploration
- Variety of optimization algorithms
- e.g. ensemble Kalman filter (**EnKF**)

[Yegenoglu et al., Front. Comput. Neurosci. 2022]

# Optimization with Learning to Learn



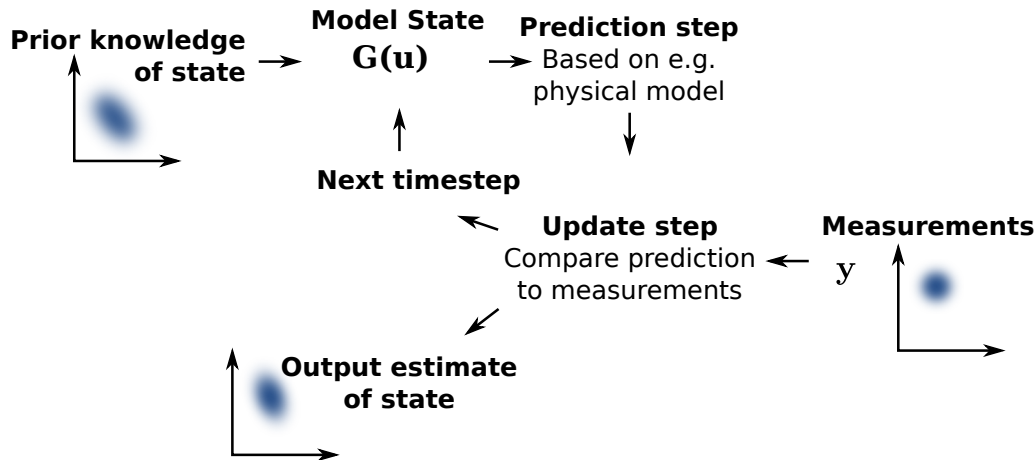
[Yegenoglu et al., Front. Comput. Neurosci. 2022]

## Learning to Learn (L2L)

- Generalization on new data sets via experience
- Parameter space exploration
- Variety of optimization algorithms
- e.g. ensemble Kalman filter (EnKF)



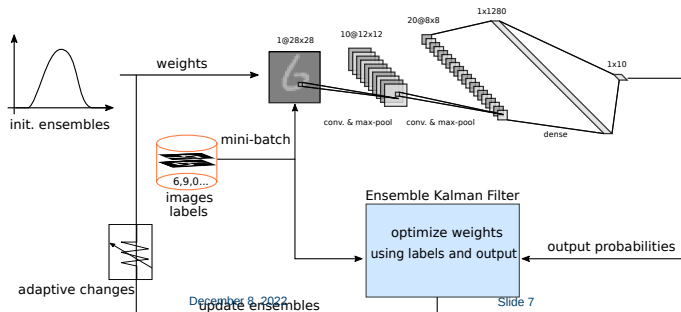
# Kalman Filter - Intuition



[[https://en.wikipedia.org/wiki/Kalman\\_filter](https://en.wikipedia.org/wiki/Kalman_filter)] modified

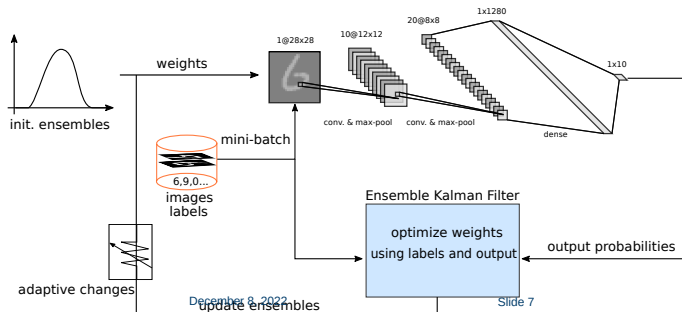
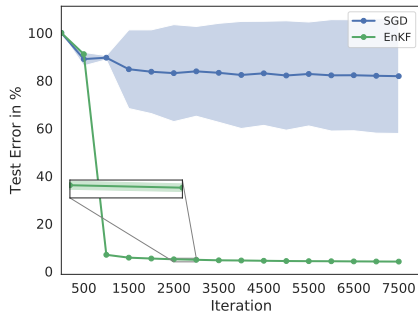
# Classification with EnKF

- MNIST & Letter dataset
- Logistic function
- Optimizer: EnKF
- [Yegenoglu et al., LOD 2020]

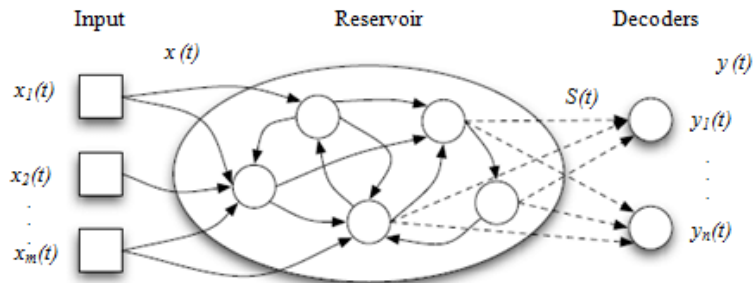


# Classification with EnKF

- MNIST & Letter dataset
- Logistic function
- Optimizer: EnKF
- [Yegenoglu et al., LOD 2020]



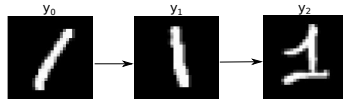
# Reservoir Computing



[Avesani et al., Neural Networks 2015]

- fixed reservoir
- connections to output are trained
- here: input encoded into firing rates

# Optimizing with L2L a Spiking Reservoir



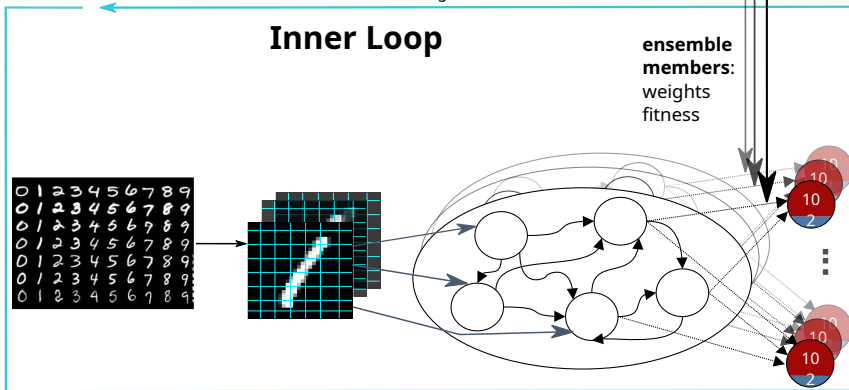
**Outer Loop**

optimize  
weights

EnKF

**Inner Loop**

ensemble  
members:  
weights  
fitness



# Reservoir Fitness



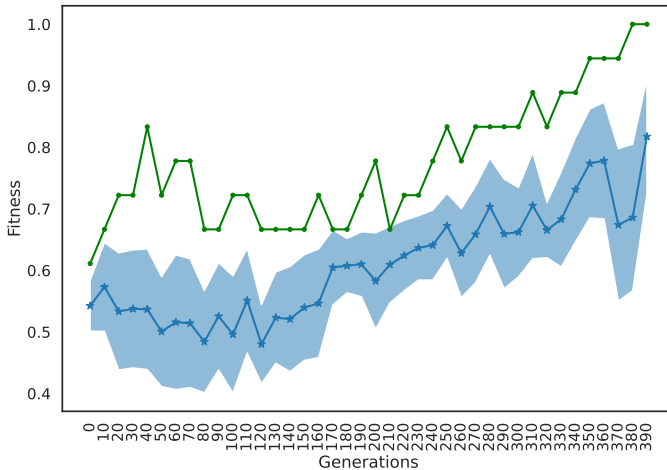
- 98 individuals
- connection weights optimized
- 7 nodes à 16 tasks



# Reservoir Fitness



- 98 individuals
- connection weights optimized
- 7 nodes à 16 tasks



# Swarm Optimization



- Foraging for food and avoiding obstacles
- Collaboration and communication
- Evolution over (long) time/generations

# Swarm Optimization

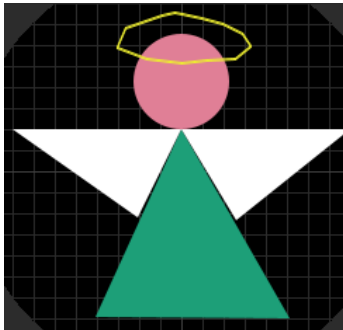


- Foraging for food and avoiding obstacles
- Collaboration and communication
- Evolution over (long) time/generations
- **here:** optimize *agents* to help *Nikolaus* to collect *presents*

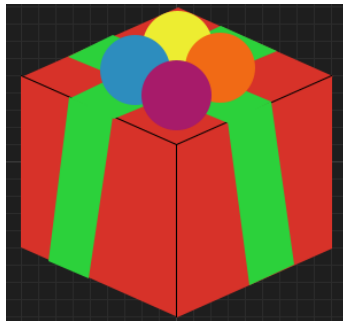
# Setting – Agents



(a) Nikolaus

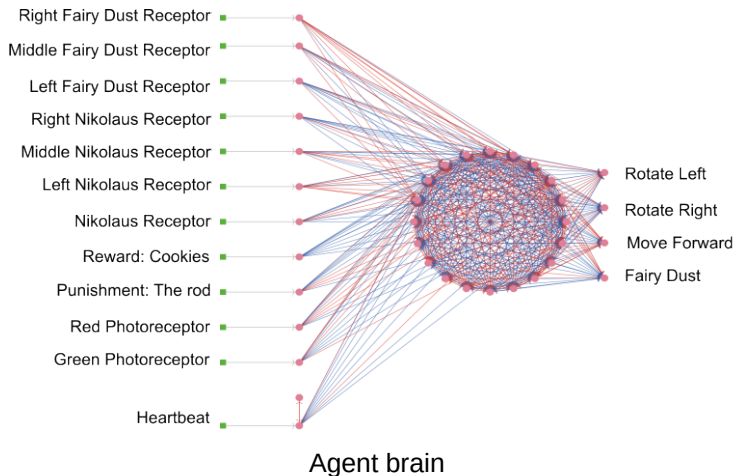


(b) Agent of Nikolaus aka angel

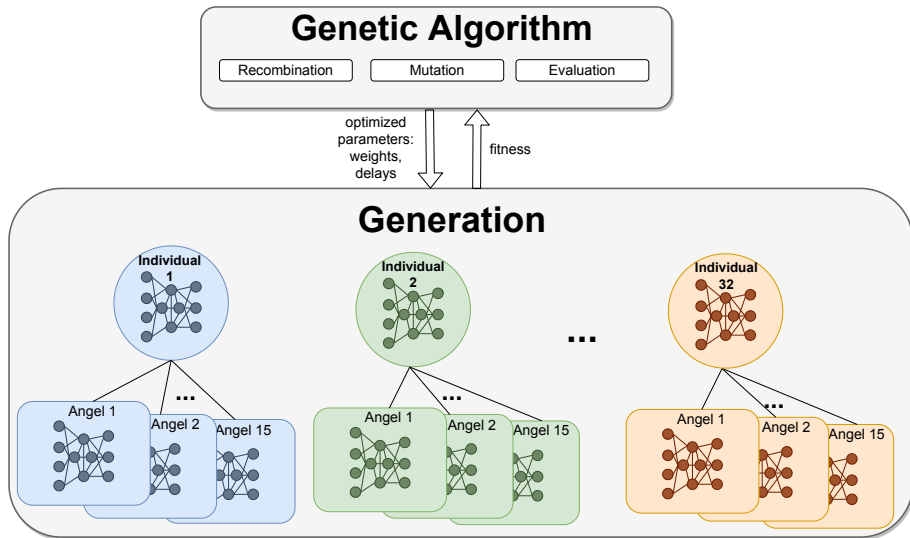


(c) Presents

# Setting – Network



# Optimization



# Video



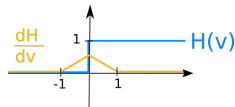
# Outlook

- Extend to different datasets
- Learning parameter mapping with ANNs (e.g. auto-encoder)
- Neuro-architecture search with evolutionary algorithms (neuroevolution)
- Swarm evolution using stigmergy: ants (multiple pheromones), drones



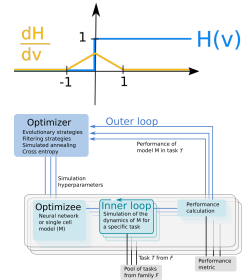
# Summary

- Training SNNs is not straightforward



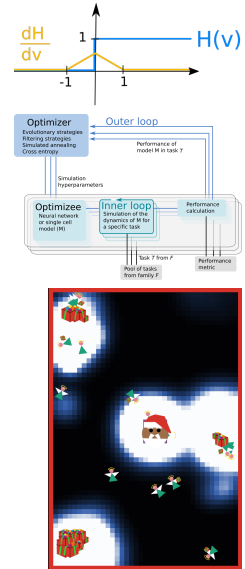
# Summary

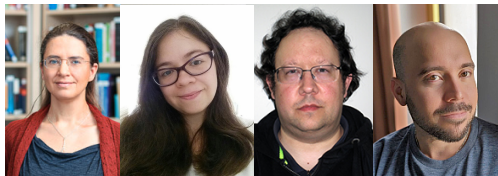
- Training SNNs is not straightforward
- Optimization via L2L and EnKF



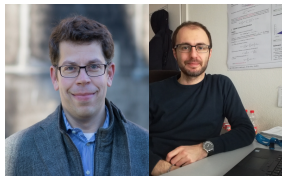
# Summary

- Training SNNs is not straightforward
- Optimization via L2L and EnKF
- Different learning tasks/applications





Abigail Morrison  
Sandra Diaz Pier  
Kai Krajsek  
Cristian Jimenez Romero



Michael Herty  
Giuseppe Visconti



Thank you for your attention  
and a merry Christmas & a happy new year  
contact: [a.yegenoglu@fz-juelich.de](mailto:a.yegenoglu@fz-juelich.de)