# **Research** Project

Master Artificial Intelligence & Robotics
2023/2024

Sterley Gilbert LABADY

**Tutors :**
- Pr. Thanos MANOS,
- Dr. Cristian JIMENEZ-ROMERO,
- Pr. Mathias QUOY

Friday 29th March 2024

# Exploring Plasticity-Driven SNNs for Enhanced ML

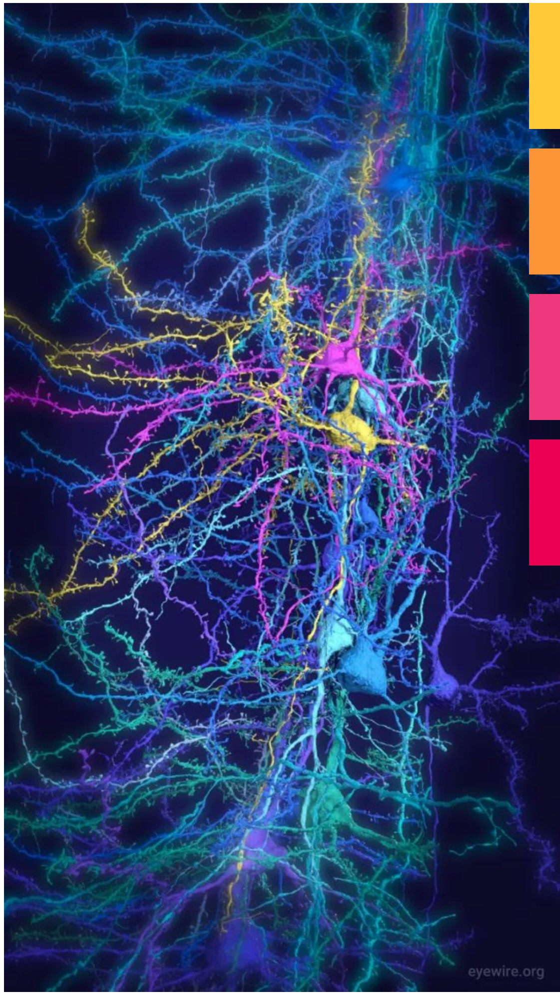
Équipes Traitement
de l'Information
et Systèmes

CERGY PARIS UNIVERSITÉ

Équipes Traitement de l'Information et Systèmes

# Contents





# Introduction
History & context
Objectives & Challenges

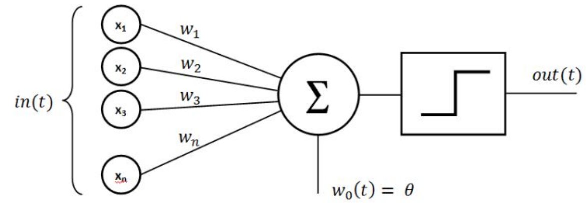
# Introduction
## History & Context

### Generations

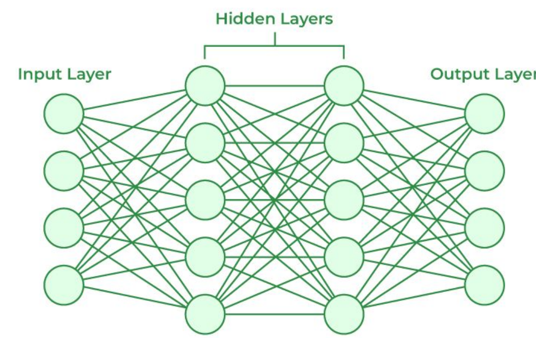
**1st** — **Perceptron & ADALINE**
1940 - Adaptive Filters

**2nd** — **Artificial Neural Networks (ANNs)**
1980 - Non-linear problem

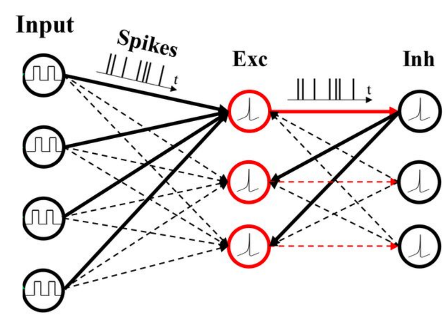
**3rd** — **Spiking Neural Networks (SNNs)**
1990 - Biological Process



souce : datascientest.com



souce : geeksforgeeks.org



souce : xialod.top

# Introduction
## Objectives & Challenges

### Objectives

**Exploring SNNs:**
Boost their application in machine learning.

**Mechanisms of Adaptability:**
Learning through local plasticity rules.

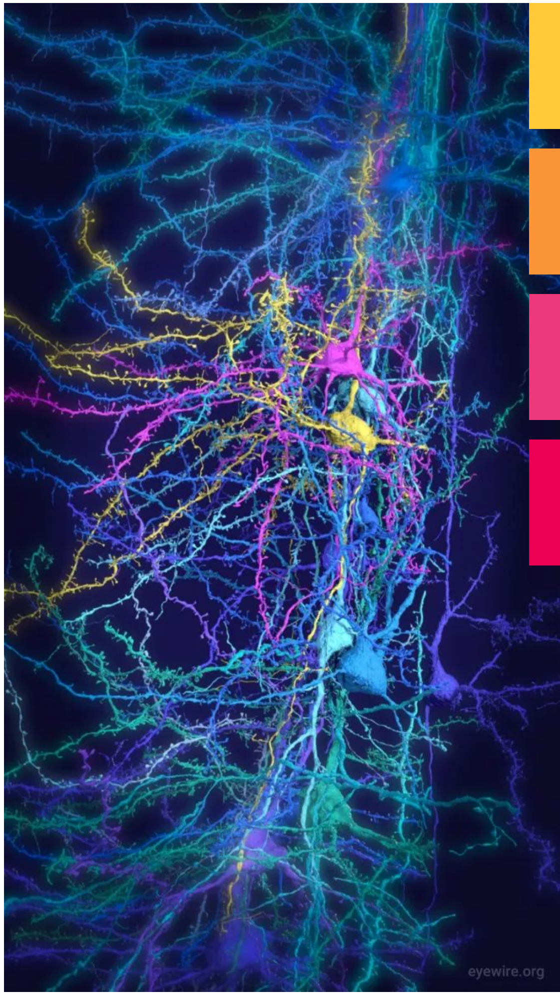
**Practical Application:**
Demonstrate the effectiveness of enhanced SNNs in machine learning.

### Challenges

**Training SNNs:**
Machine learning tasks without gradient-based methods.

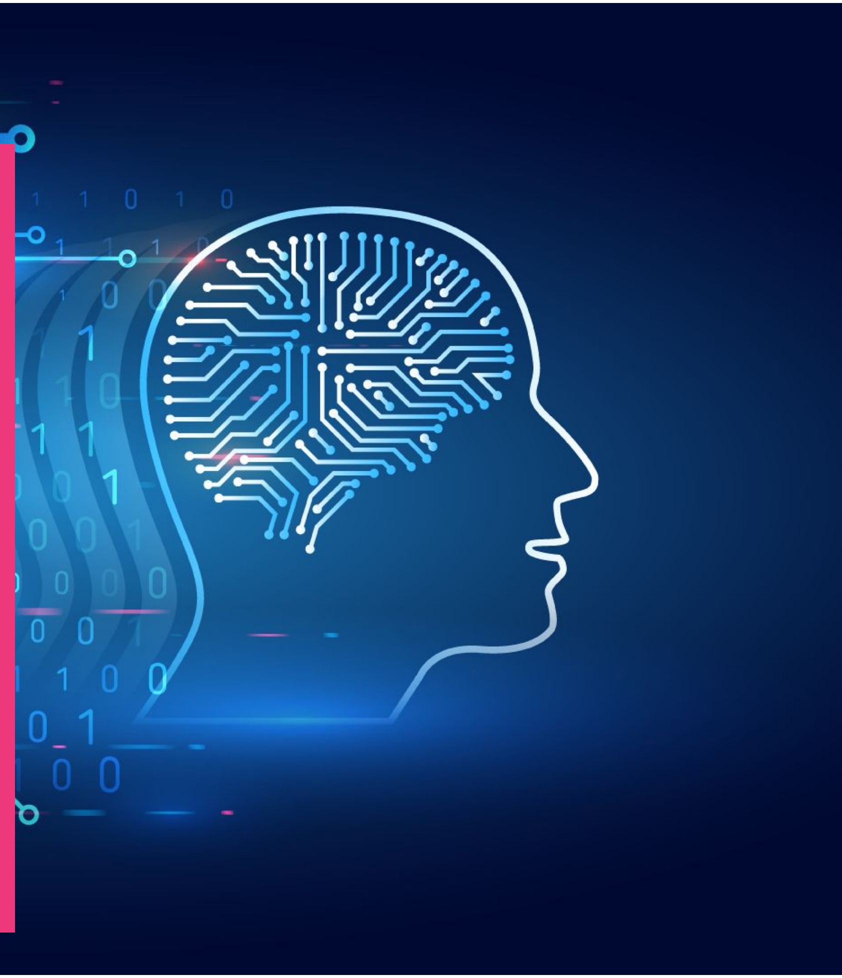
**Nature of Spikes:**
Spikes complicates the use of backpropagation.

**Incorporating Plasticity:**
Local plasticity rules adds complexity to network architecture and training.



## Literature Review

- Leaky Integrate-and-Fire (LIF)
- Spike-Timing- Dependent Plasticity (STDP)
- Neural Simulation Tool (NEST)
- Mountain Car (MC)
- Genetic Algorithms (GAs)
- Learning to learn (L2L)

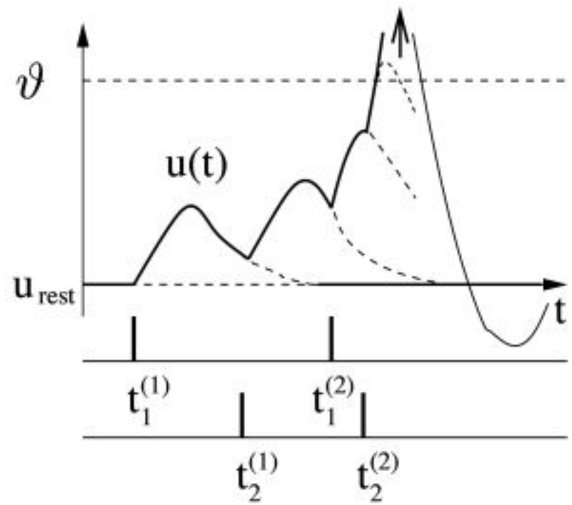


# Literature Review
## LIF

### Neuron model : LIF

$$\frac{dV_m}{dt} = \frac{V_m - E_L}{\tau_m} + \frac{I_{syn} + I_e}{C_m}$$

| Parameter | Unit | Description | Value |
|---|---|---|---|
| $V_{th}$ | mV | Spike threshold | -55 |
| $E_L$ | mV | Resting membrane potential | -70 |
| $C_m$ | pF | Membrane capacitance | 250 |
| $\tau_m$ | ms | Membrane time constant | 10.0 |
| $\tau_{ref}$ | ms | Refractory period duration | 1.0 |
| $V_{reset}$ | mV | Reset potential | -70 |
| $\tau_{syn\_ex}$ | ms | Excitatory synaptic time constant | 2.0 |
| $\tau_{syn\_in}$ | ms | Inhibitory synaptic time constant | 2.0 |
| $I_e$ | pA | Constant input current | — |




souce : neuronaldynamics.epfl.ch/index.html

**State of the art :** Hodgkin-Huxley Model, Exponential Integrate-and-Fire (EIF), Adaptive Exponential Integrate-and-Fire (AdEx)

# Literature Review
## STDP

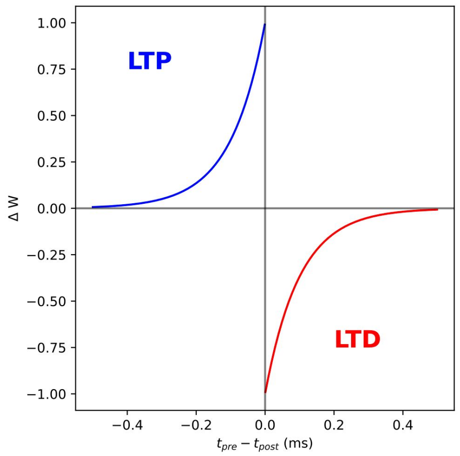
### Synapse model : STDP

Synaptic weight potentiation:

$$w_{pot} = W_{max}\left(\frac{w}{W_{max}} + \lambda\left(1 - \frac{w}{W_{max}}\right)^{\mu_+} pre\_trace\right) \tag{8}$$

Synaptic weight depression:

$$w_{dep} = W_{max}\left(\frac{w}{W_{max}} - \alpha\lambda\left(\frac{w}{W_{max}}\right)^{\mu_-} post\_trace\right) \tag{9}$$




souce : wikipedia.org

- **Weight** ($w$): The strength of the synaptic connection, set by the plasticity mechanism.
- **Transmission delay** ($d$): The time taken for the signal to traverse the synapse, set to 1.0 ms.
- **Learning rate** ($\lambda$): Determines the rate at which the synapse can change, set to 0.01.
- **Pre-trace and post-trace time constants** ($\tau_{\text{plus}}$, $\tau_{\text{minus}}$): Governs the decay of spike traces, both set to 20.0 ms.
- **Potentiation factor** ($\mu_{\text{plus}}$): Modulates the weight change during potentiation, set to 1.0.
- **Depression factor** ($\mu_{\text{minus}}$): Modulates the weight change during depression, set to 1.0.
- **Maximum synaptic weight** ($W_{\text{max}}$): Defines the upper limit within which the synaptic weight can vary, set to 3000.0.

**State of the art :** Hebbian Learning, Rate-Based Plasticity, Short-Term Plasticity (STP), R-STDP

Équipes Traitement de l'Information et Systèmes

# Literature Review

## Simulation

### Neural Simulation Tool (NEST)

- **Simulation software for SNNs.**
- **Highly regarded simulator in computational neuroscience.**
- **Rich library of neuron & synapse models and dynamics.**

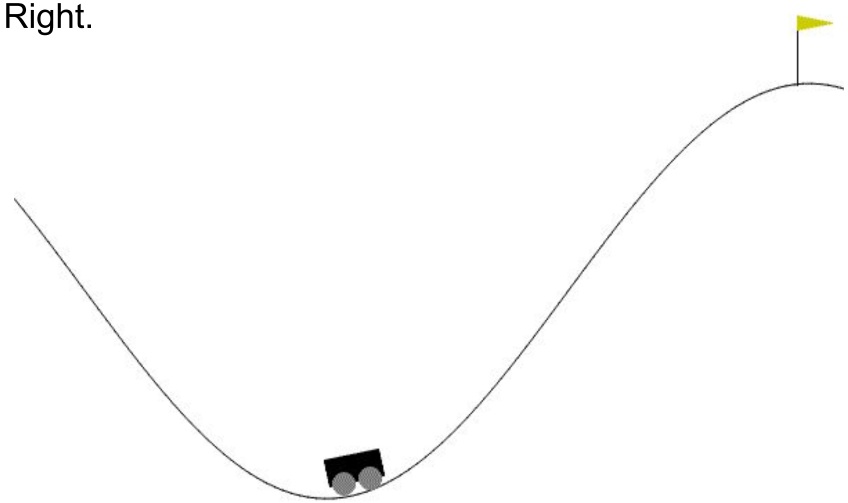
**State of the art :** NEURON, Brian, Nengo, PyNN, CARLsim

### Mountain Car (MC)

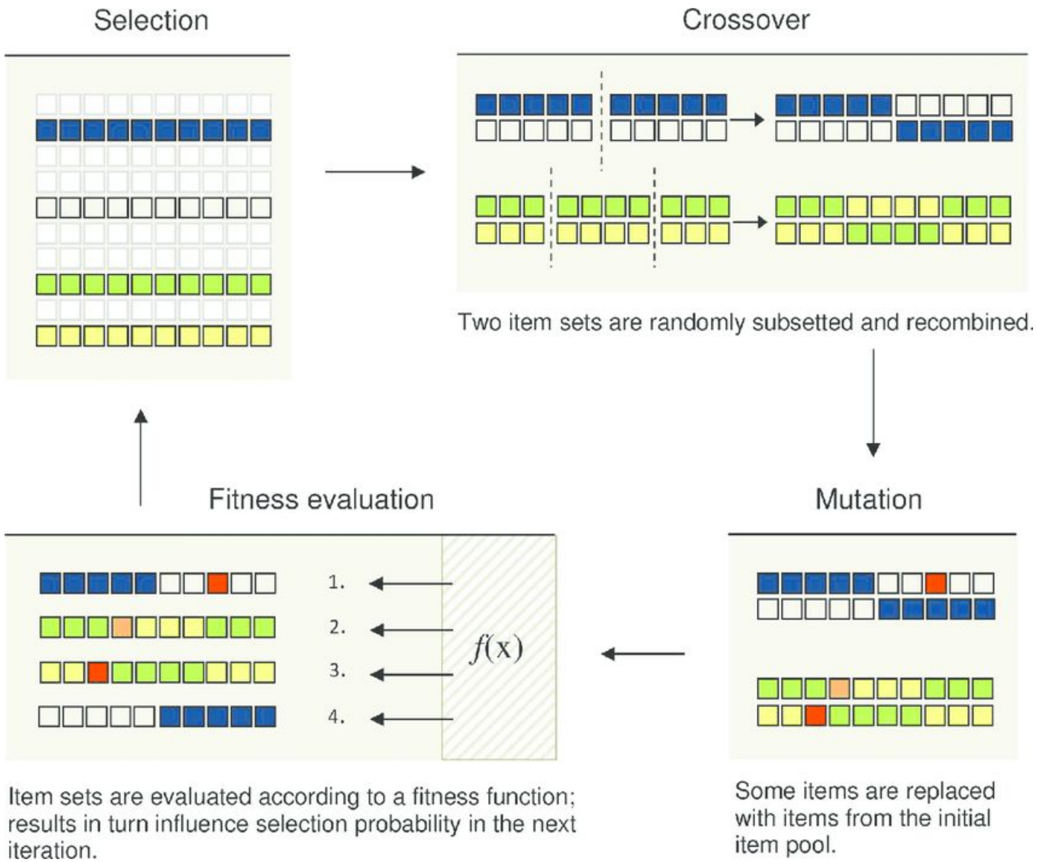
**The MC environment**, a benchmark in RL, involves a car in a valley, requiring strategic accelerations to reach a goal atop a hill. The agent must learn to navigate to the goal efficiently.

**Observations :** Position & Velocity

**Actions :** Accelerate Left, No Acceleration, Accelerate Right.



# Literature Review
## Genetic Algorithms (GAs)




souce : researchgate.net

- Population size (`32`): Determines the number of individuals in each generation. A larger population size allows for more diverse solutions but increases computational load.
- Crossover probability (`0.7`): The likelihood of two individuals exchanging genetic information, promoting diversity in the population. (`Exploration`)
- Mutation probability (`0.7`): Governs the chance of random alterations in an individual's genetic, introducing variability.($\Delta\text{Mut} \sim \mathcal{N}(\mu = 0, \sigma = 1)$) (`Exploration`)
- Number of generations (`1000`): The total number of iterations for the GA, affecting the convergence of the algorithm.
- Tournament size (`4`): Used in the selection process to choose individuals for reproduction. (`Exploitation`)
- Mating Bend (`0.5`)): Half of each parent genetics is used for crossover operation.

# Literature Review
## Learning to learn (L2L)

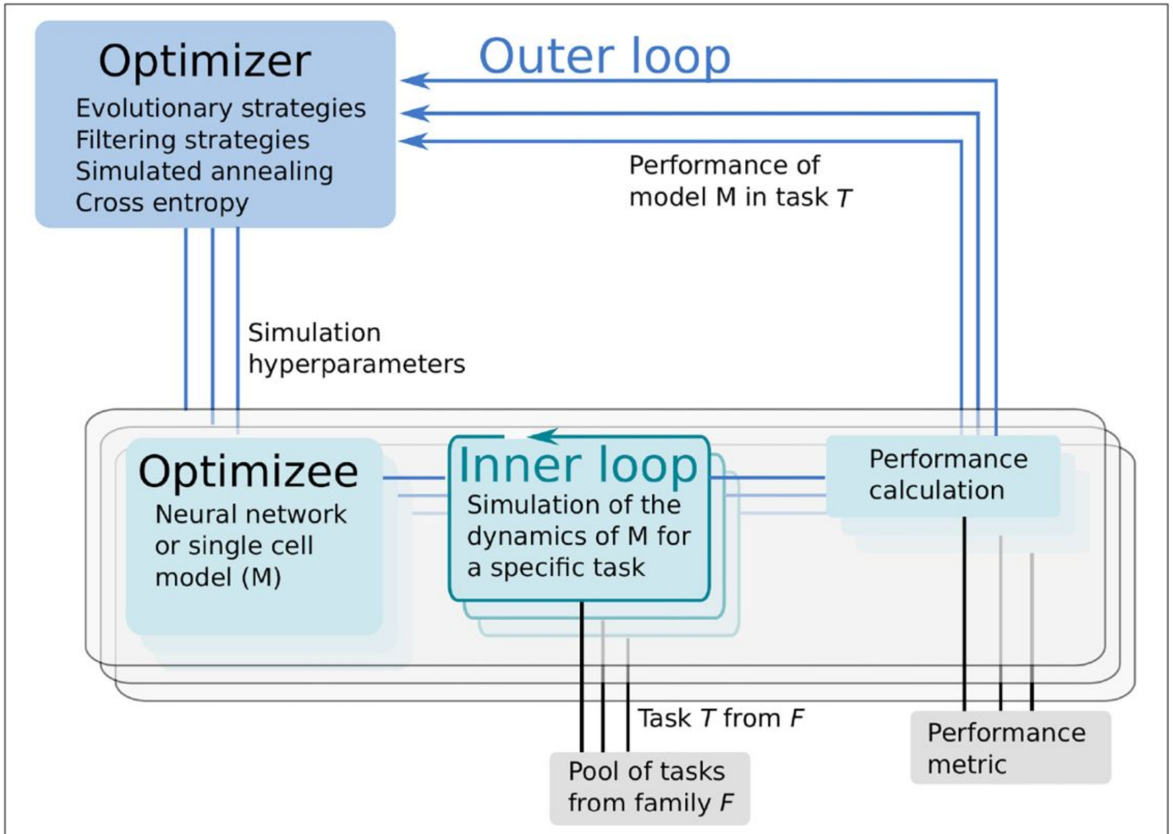
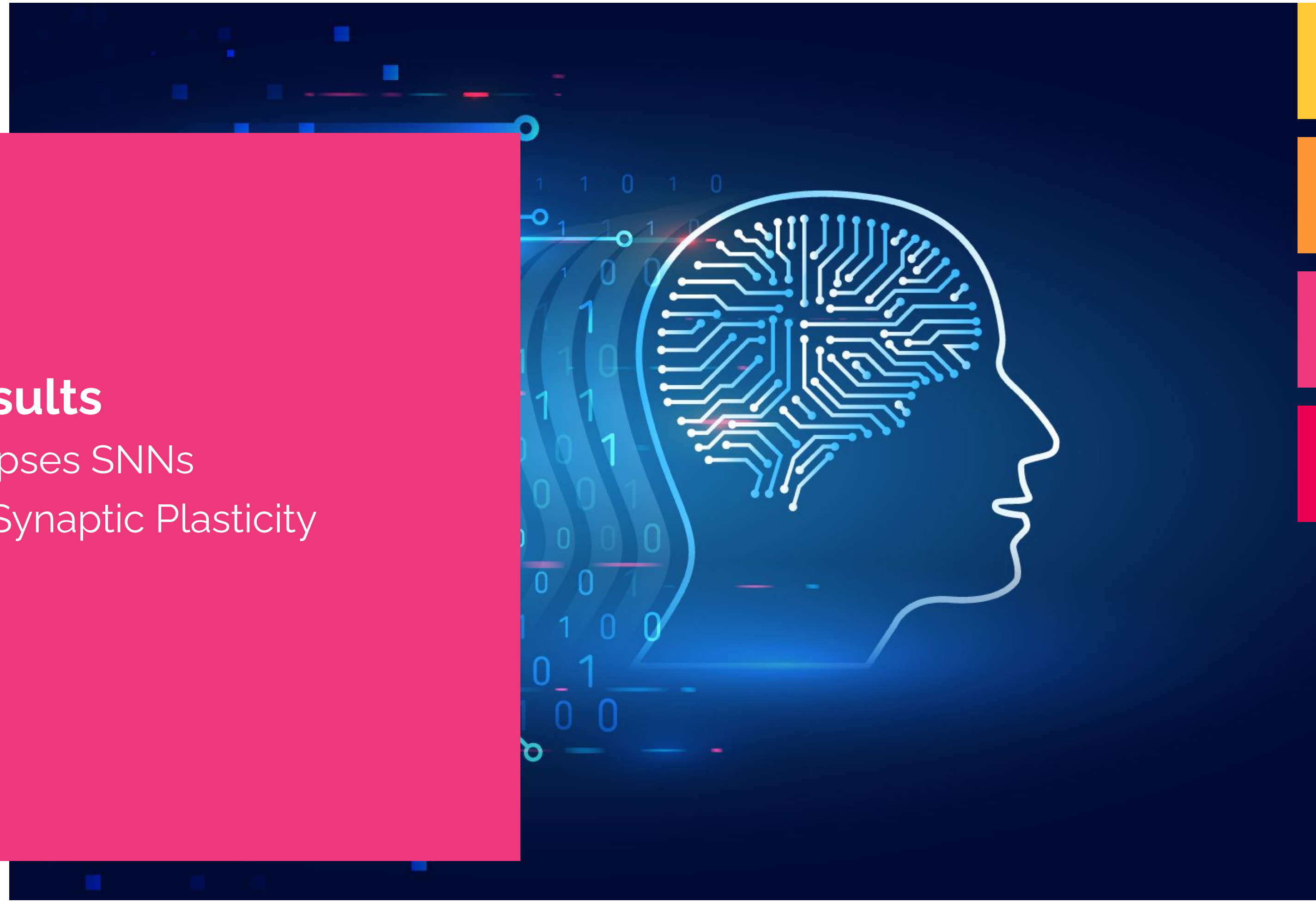



**FIGURE 1 |** Learning to learn (L2L) consists of two loops. In the inner loop, the optimizee, an algorithm with learning capabilities is trained on a family of tasks. A fitness function evaluates the performance of the algorithm. The (hyper-) parameters and the fitness value of the algorithm are sent to the optimizer in the outer loop. Several optimization methods are available to optimize the parameters, which are fed back to the optimizee and the algorithm.

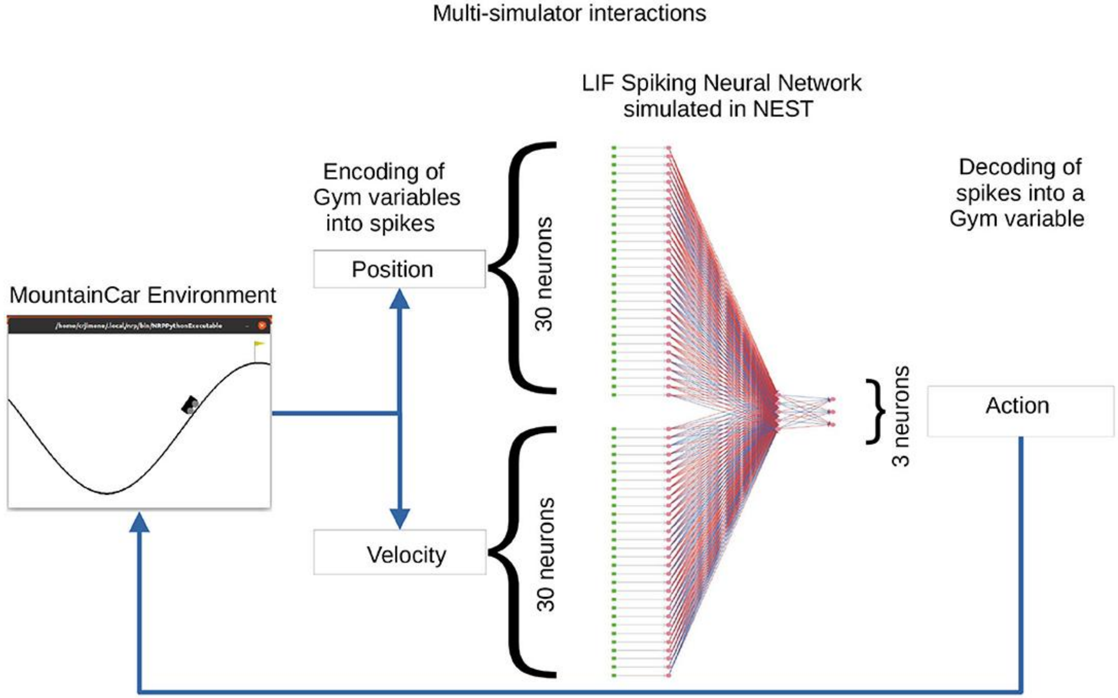
souce : frontiersin.org/articles/10.3389/fncom.2022.885207/full



## Methodology & Results

- Optimizing Static-synapses SNNs
- Enhancing SNNs with Synaptic Plasticity

# Methodology
## Optimizing Static-synapses SNNs




souce : frontiersin.org/articles/10.3389/fncom.2022.885207/full

### Setup

**Population size : 32**
Number of individuals per generation

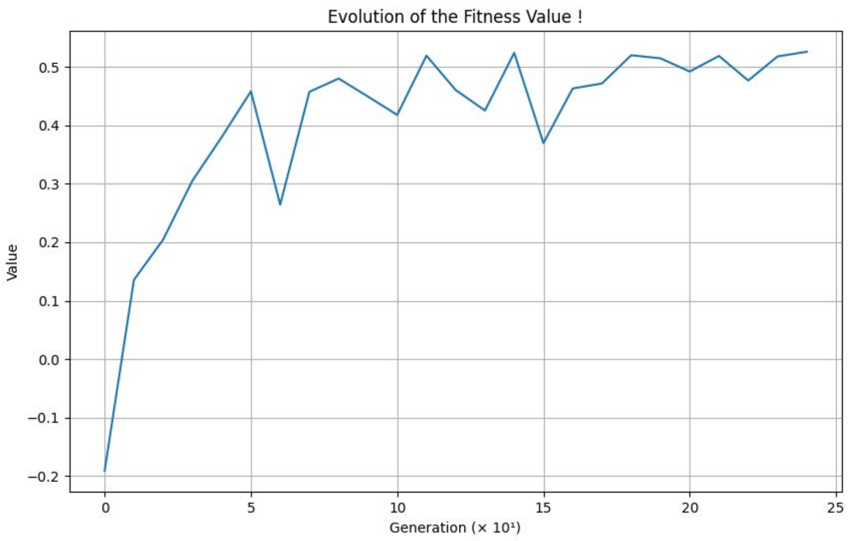
**Training parameters : Weights**
Static synapses -> Static weights value

**Simulation : 110 steps :**
Number of steps allowed for the agent to solve the challenge

$$\text{Fitness}(I) = f(\text{paramètres}(I)) = \max_{T}(PET)$$

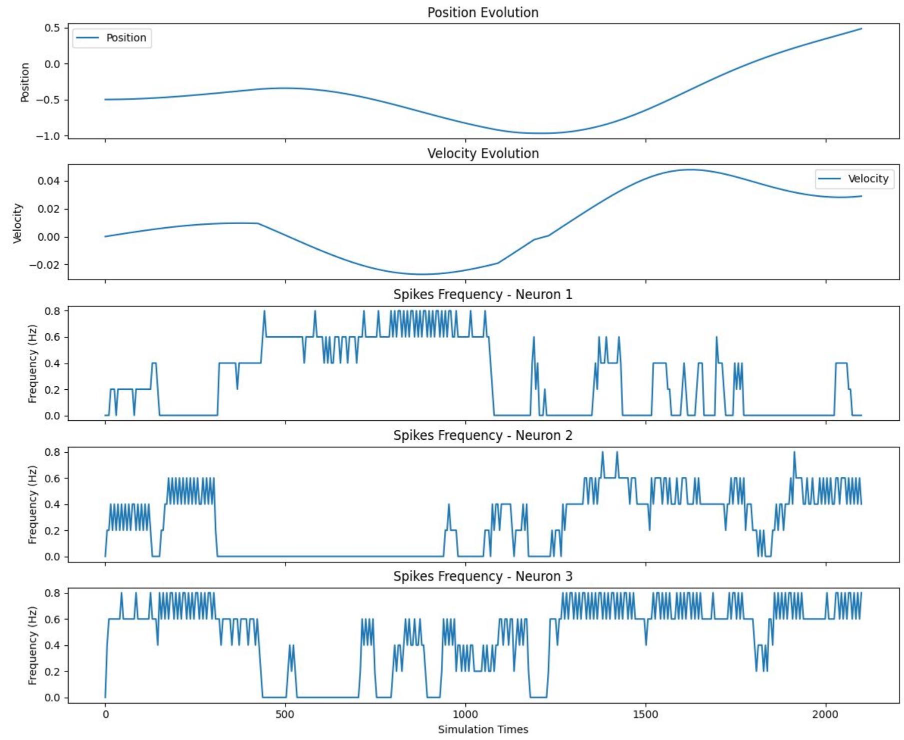
# Results
## Optimizing Static-synapses SNNs

### Training Fitness Evolution

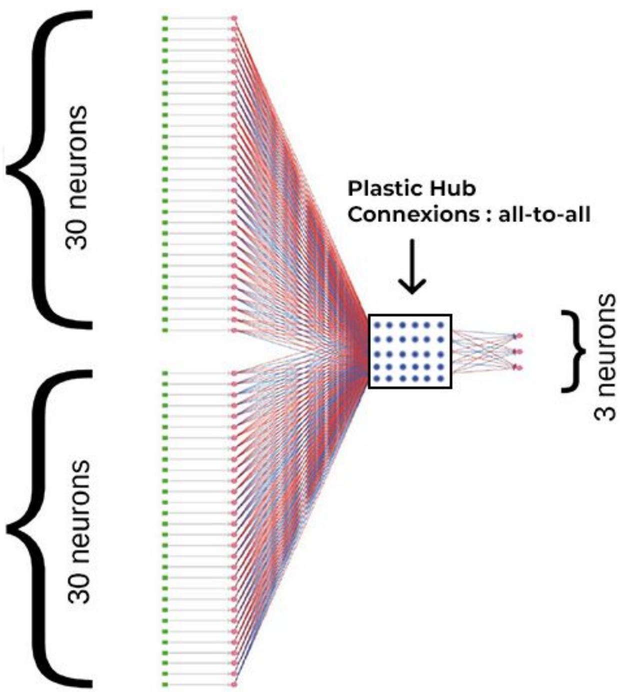



- The x-axis represents the Generation Index (times 10)
- The y-axis denotes the best individual fitness value of each generation.
- Convergence of the GA from the 110th Generation.

### Environment Dynamics



# Methodology
## Enhancing SNNs with Synaptic Plasticity




### Updates

**Population size : 96**
This configuration aligns with the hardware's capacity : 96 CPUs

**Training parameters :**
Plasticity_flags & Initial Weights value

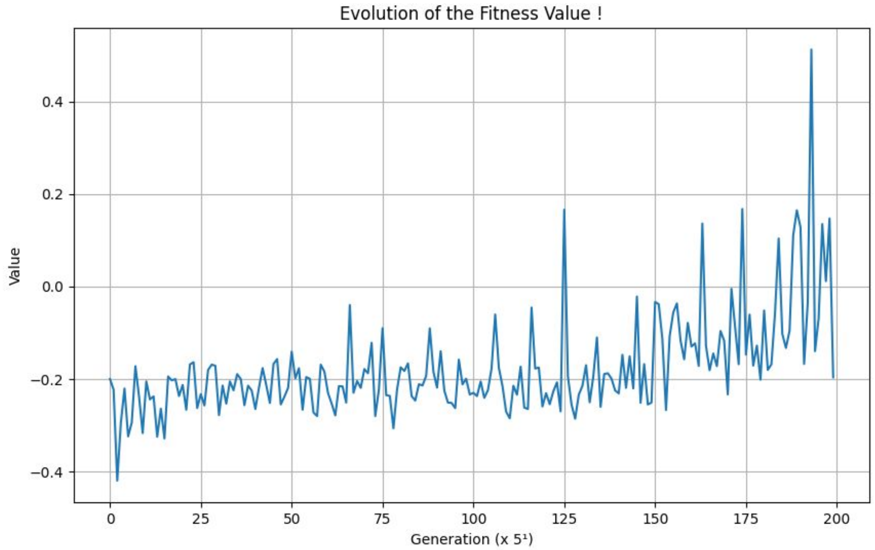
**Simulation : 600 steps allowed :**
Requires a higher number of simulation steps.

**Plastic Hub :**
All-to-all connectivity pattern with no self-connections.

$$\text{Fitness}(I) = f(\text{paramètres}(I)) = \max_{T}(PET)$$

# Results
## Enhancing SNNs with Synaptic Plasticity

### Training Fitness Evolution




- The x-axis indicates the generation index (factor of 5)
- The y-axis denotes the best individual fitness value of each generation.
- This peak signifies the emergence of an individual that successfully solved the challenge.

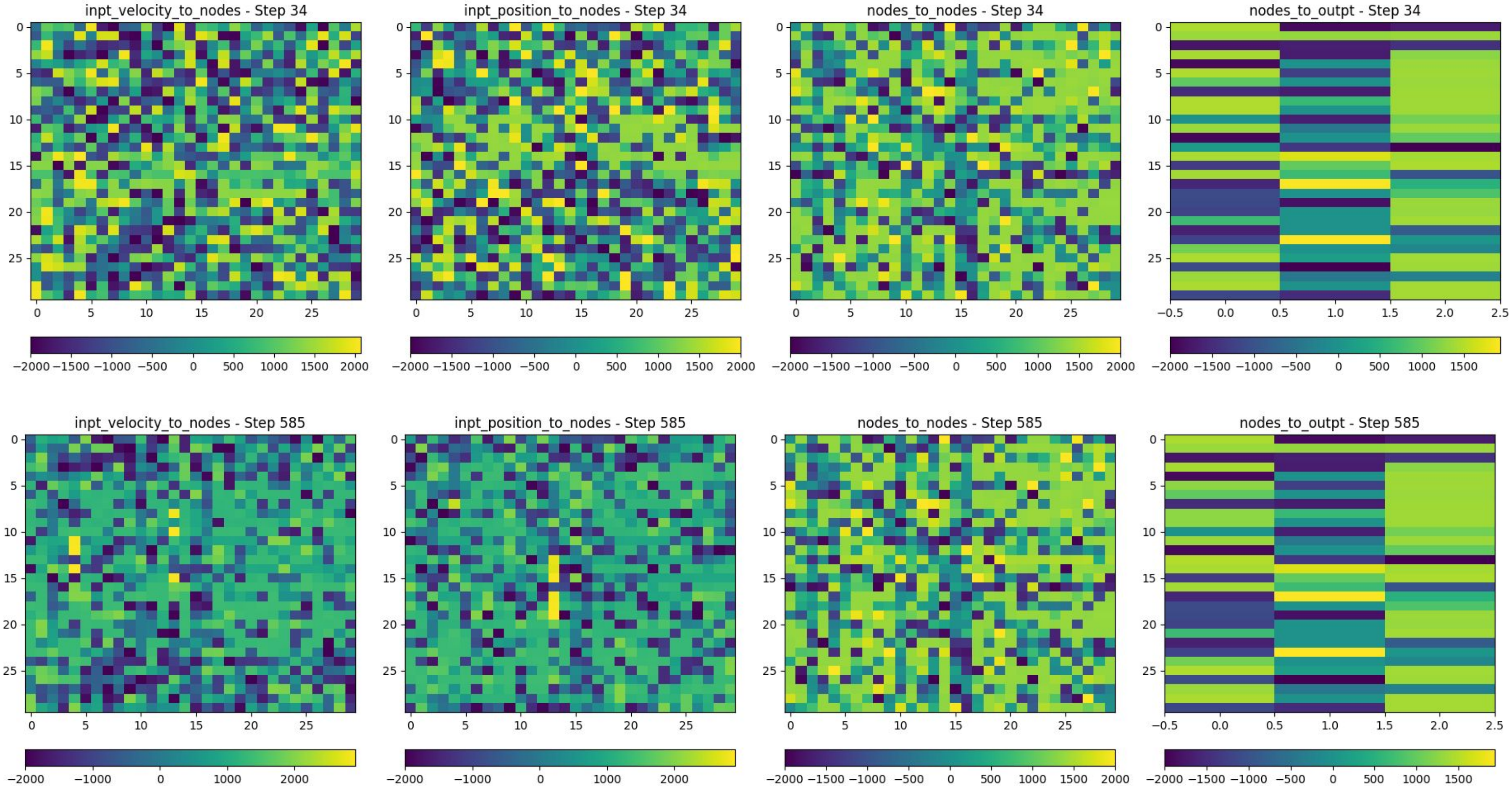
### Best individual

**Plastic synapses :**
1389/2700 (51.44%) plasticity flag = 1

**Simulation steps :**
585 over 600 steps

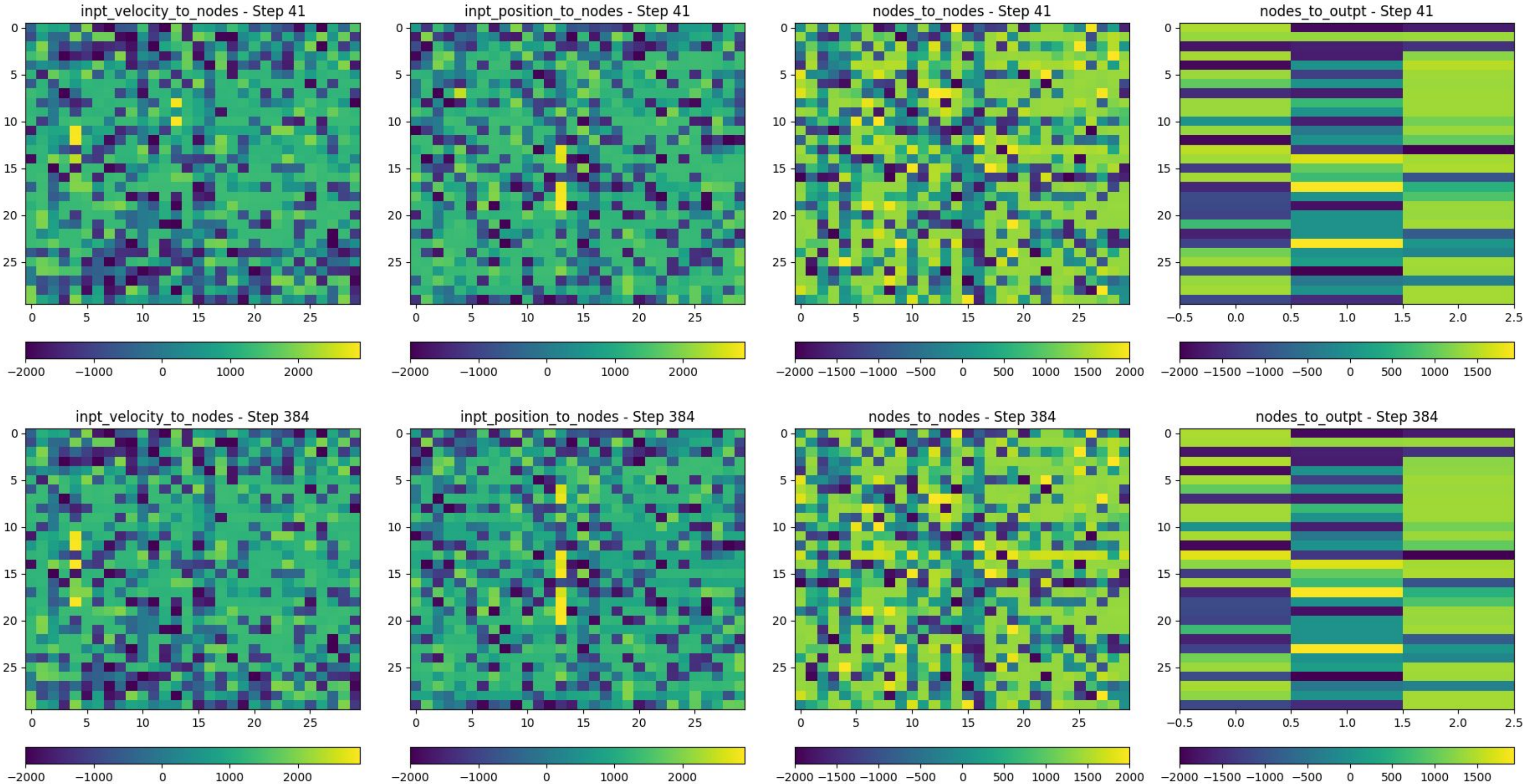
Équipes Traitement de l'Information et Systèmes

# Results
## Network Evolution Simulation 1



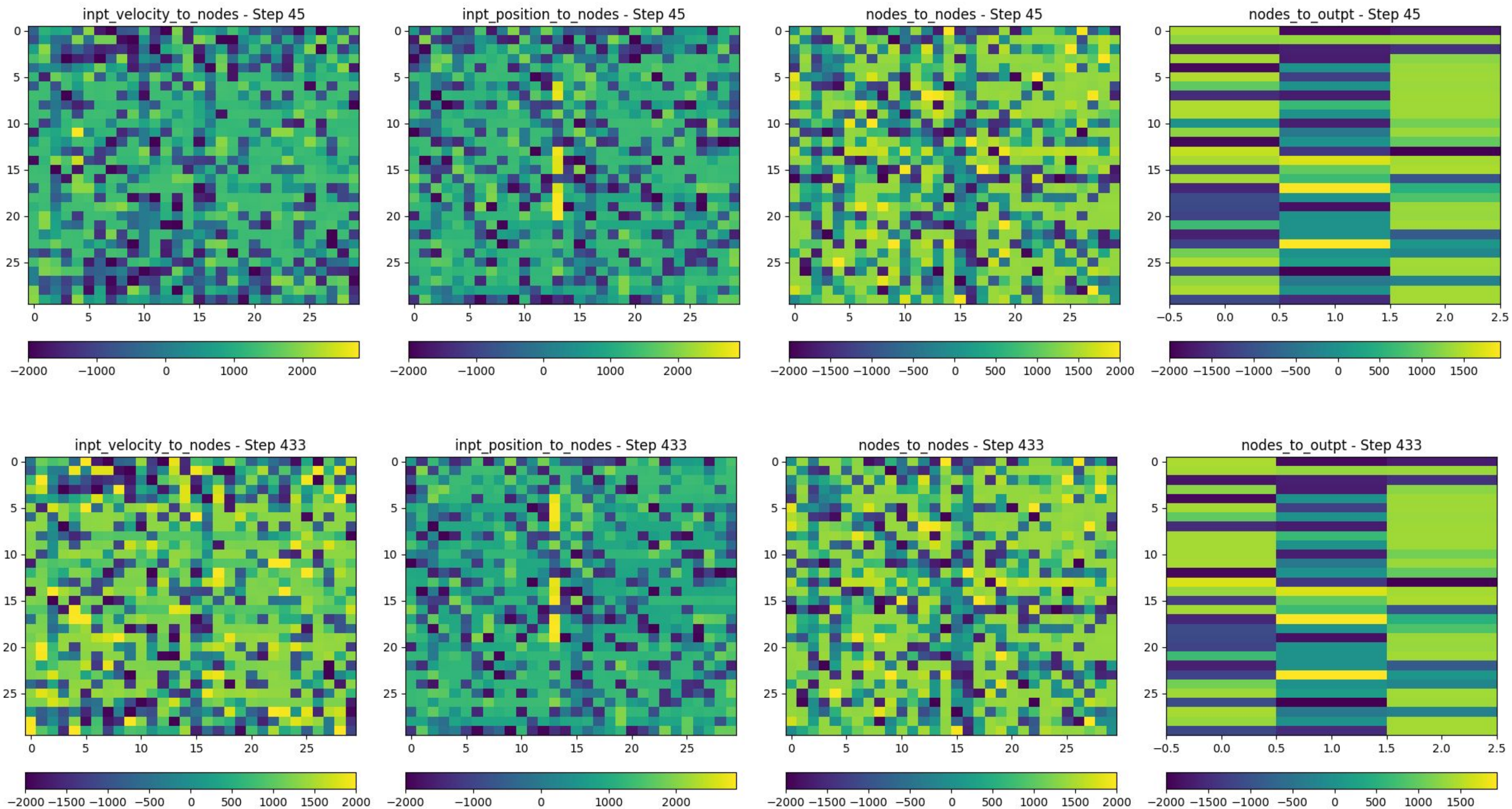
Équipes Traitement de l'Information et Systèmes

# Results
## Network Evolution Simulation 2



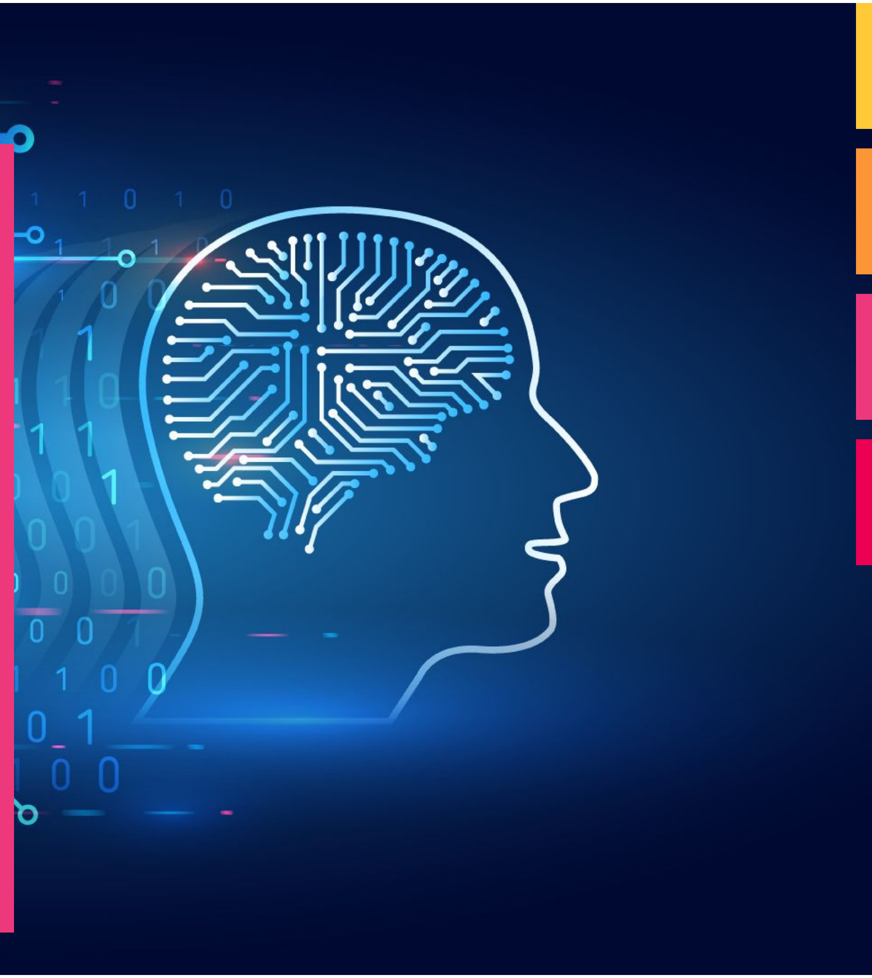
# Results

## Network Evolution Simulation 3



## Conclusion

- Discussion
- Future work insights



# Conclusion
## Discussion

### 01 Feasibility Demonstration
Our experiments show that plasticity-driven SNNs can effectively solve complex RL problems.

### 02 Potentials & Benefits
Opens up new possibilities for creating more adaptable and efficient AI systems.

### 03 Generation
Pushing the boundaries of what these networks can achieve with several different random initial state.

### 04 Scalability
Adapt for more complex tasks and larger datasets in future applications.

# Conclusion
## Future work insights

**01**

### Allow more steps to the SNN : 600 -> 1000

As a plastic network requires a higher number of simulation steps compared to a static one.

**03**

### Novel fitness function

**If MaxPosition < 0.5 : Fitness = MaxPosition**
**If MaxPosition >= 0.5 : Fitness = MaxPosition + 100/sim_step**

Équipes Traitement de l'Information et Systèmes

# Relevant References


01

**2014**

Wulfram et al. Neuronal dynamics - a neuroscience textbook, cambridge university press.

02

**2022**

Yegenoglu et al. Exploring Parameter and Hyper-Parameter Spaces of Neuroscience Models on High Performance Computers With Learning to Learn. Frontiers in Computational Neuroscience

03

**2022**

Haşegan et al. Evolutionary and spike-timing-dependent reinforcement learning train spiking neuronal network motor control,

04

**2019**

Wunderlich et al. Demonstrating Advantages of Neuromorphic Computation.

05

**2007**

Marc-Oliver Gewaltig and Markus Diesmann, NEST (NEural Simulation Tool)

06

**2019**

Bing et al. Supervised learning in SNN via reward-modulated spike-timing- dependent plasticity for a target reaching vehicle.

# Questions

ETIS
Équipes Traitement de l'Information et Systèmes

Laboratoire Etis
6 Rue du Ponceau
95000 Cergy

T.07 61 76 91 47

etis-lab.fr