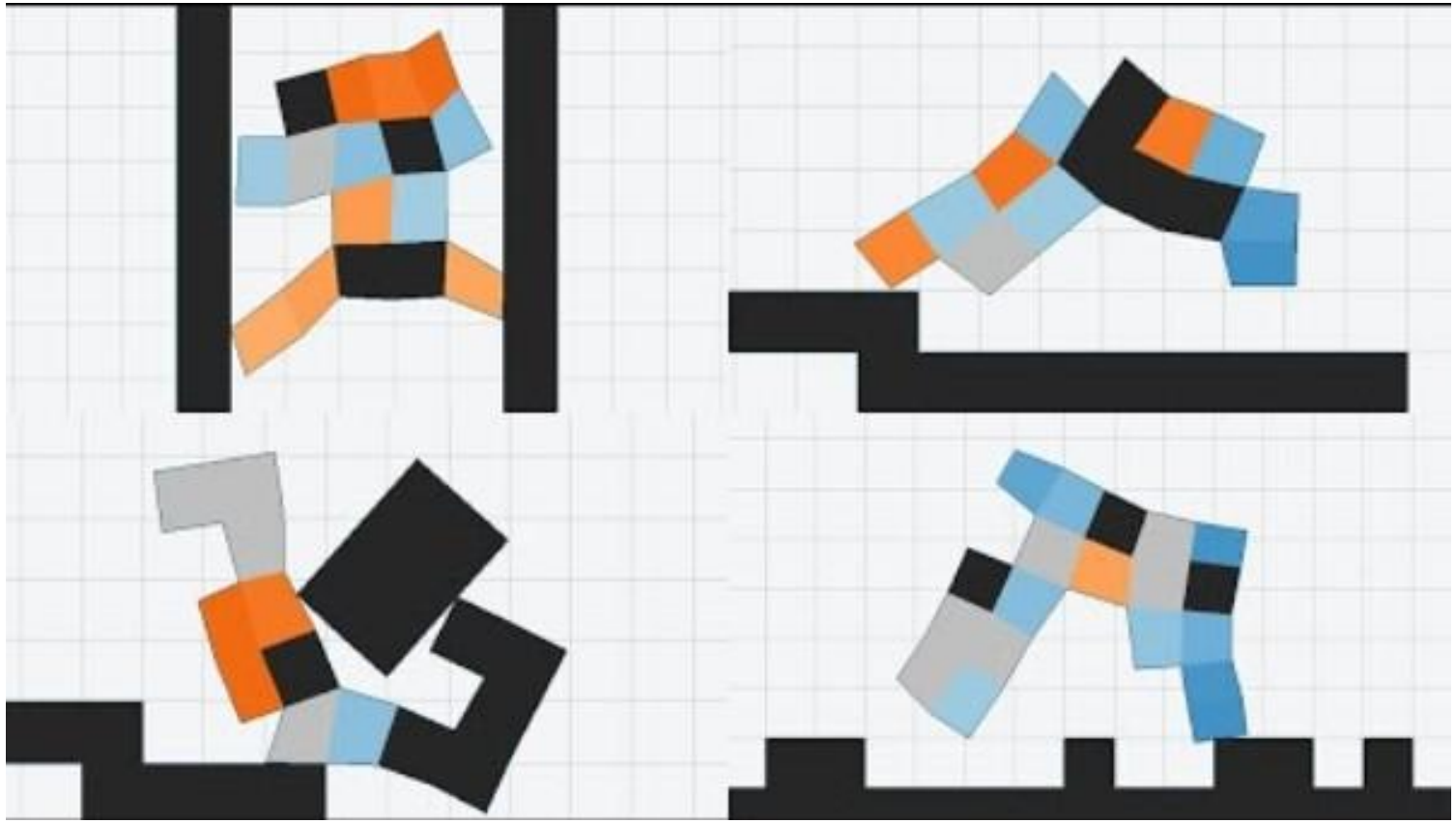
Diploidy and Indirect-Encoding, new ways to explore Body-Brain co-optimization in 2D single genome soft-robot

MATTEO VACHER

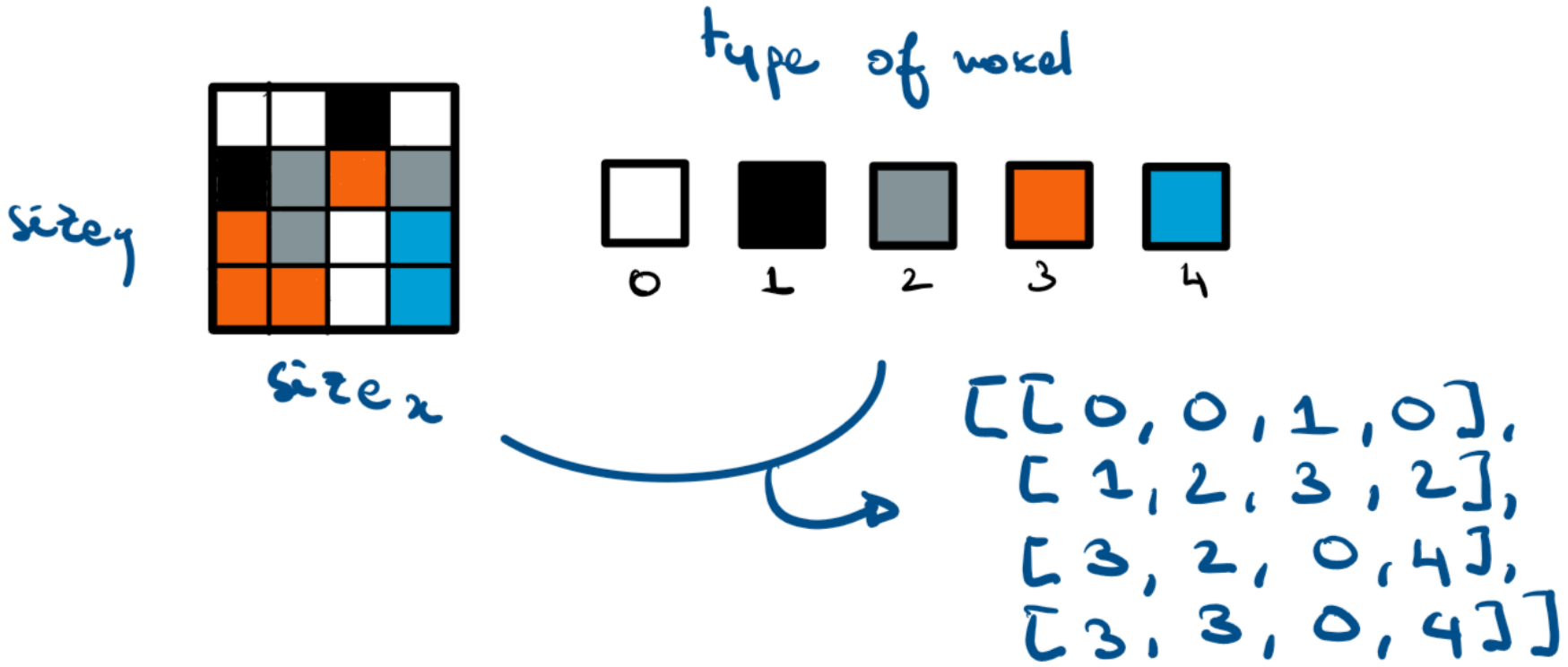
ENGINEERING STUDENT AT ISAE-SUPAERO

EXCHANGE RESEARCH STUDENT AT TSUKUBA UNIVERSITY

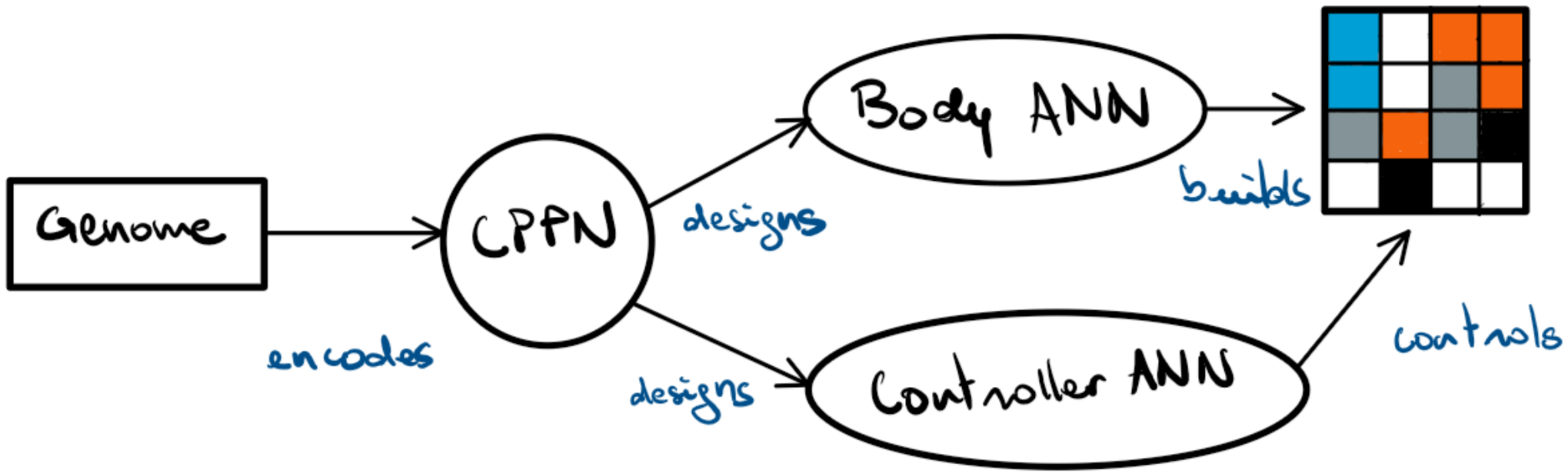
Evogym



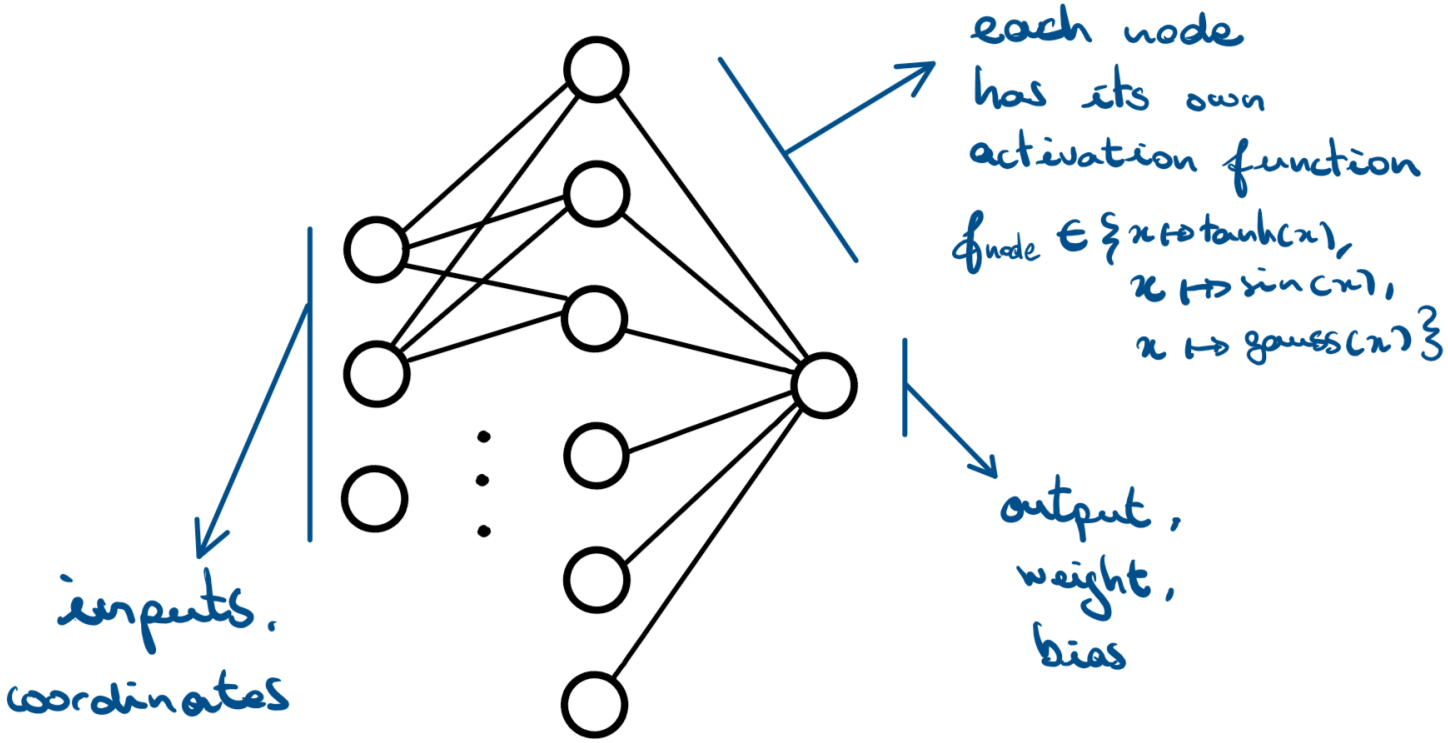
What is a Soft-Robot



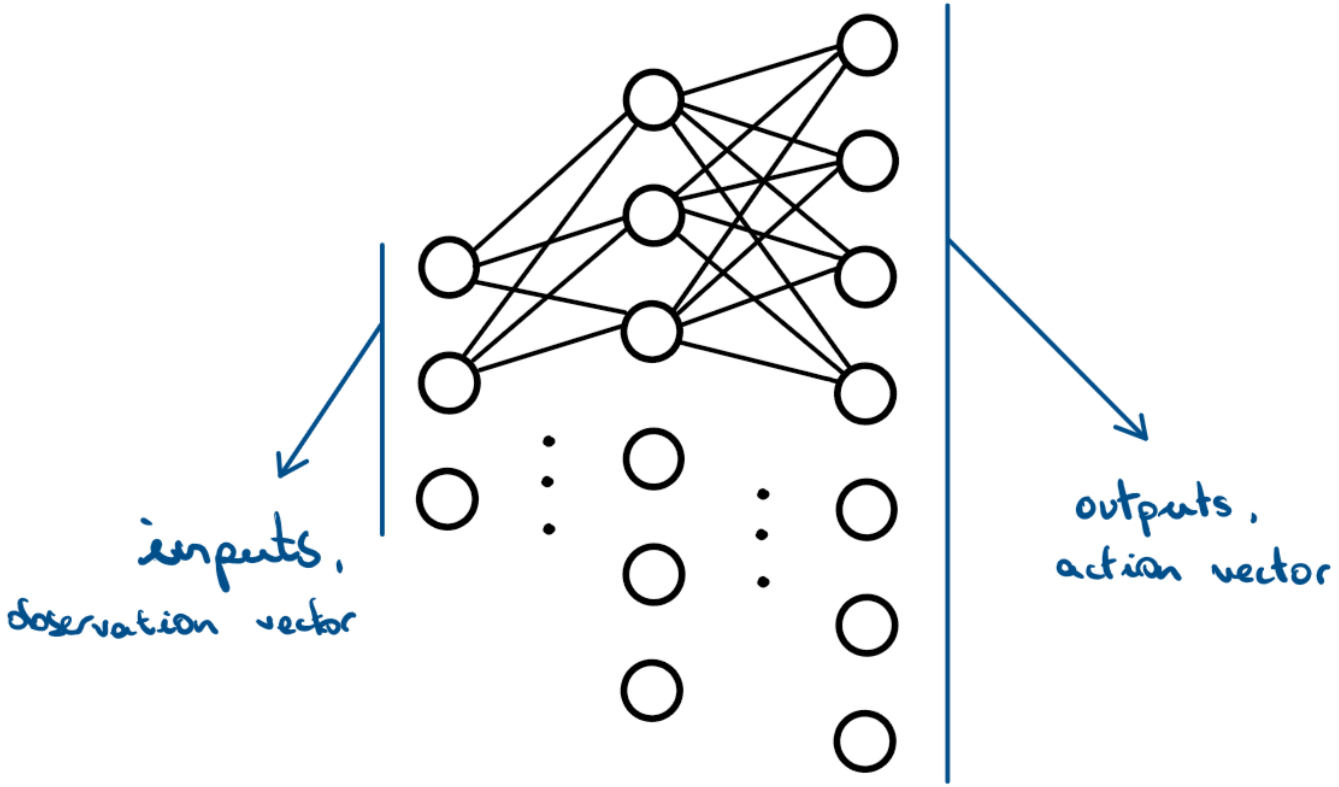
Indirect Encoding



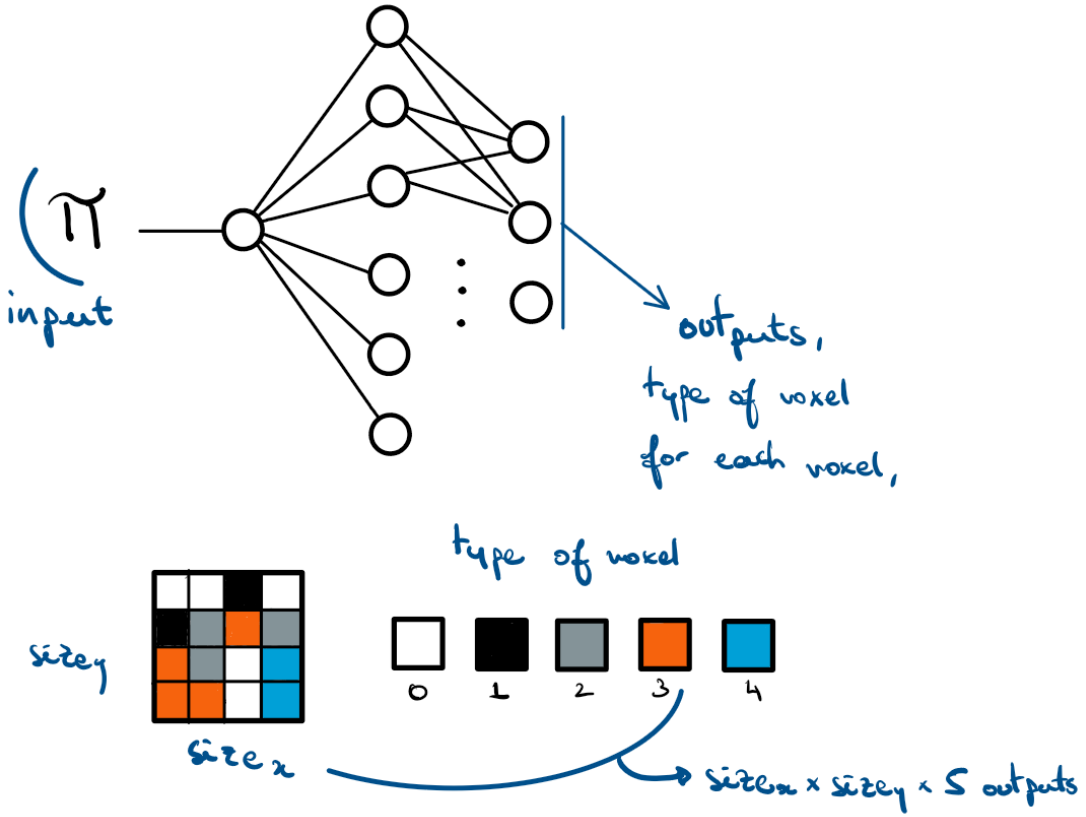
Compositional Pattern Producing Network



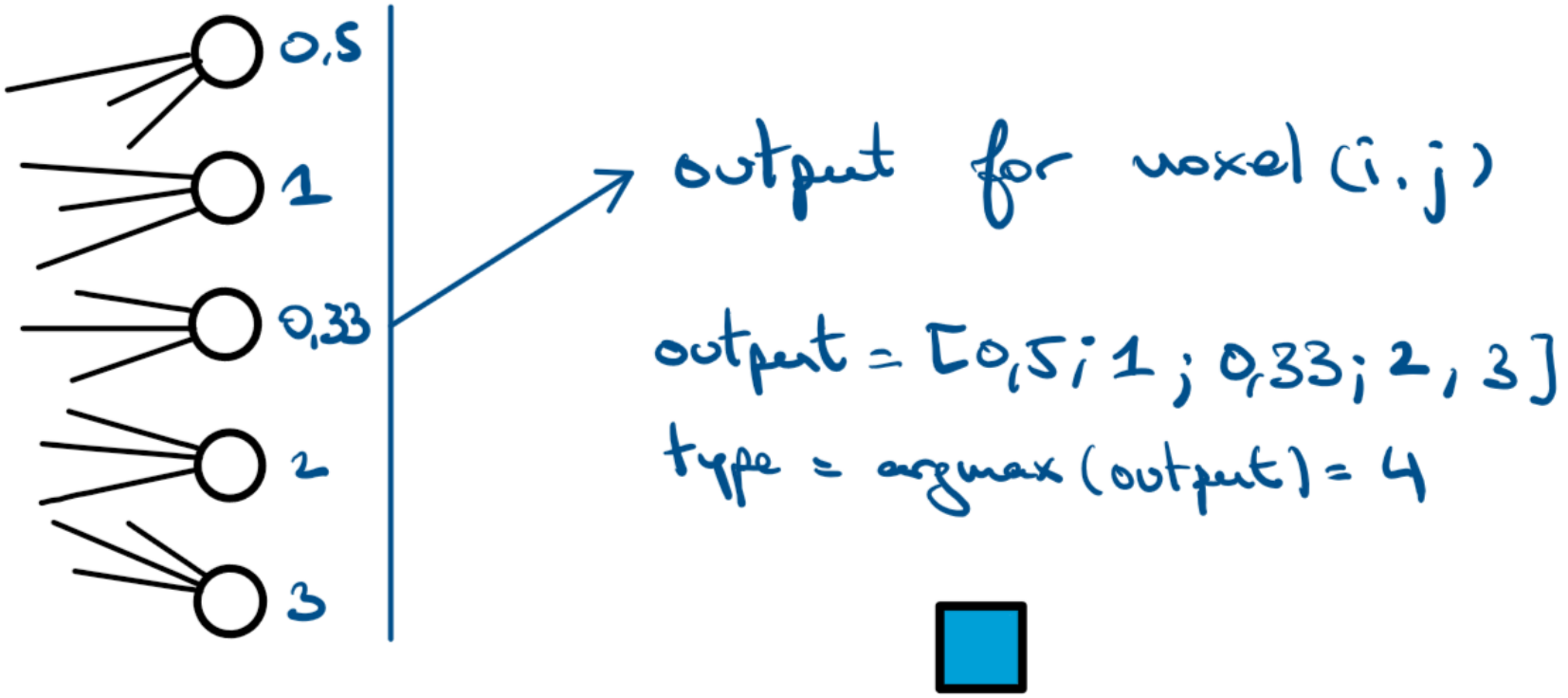
Controller Artificial Neural Network



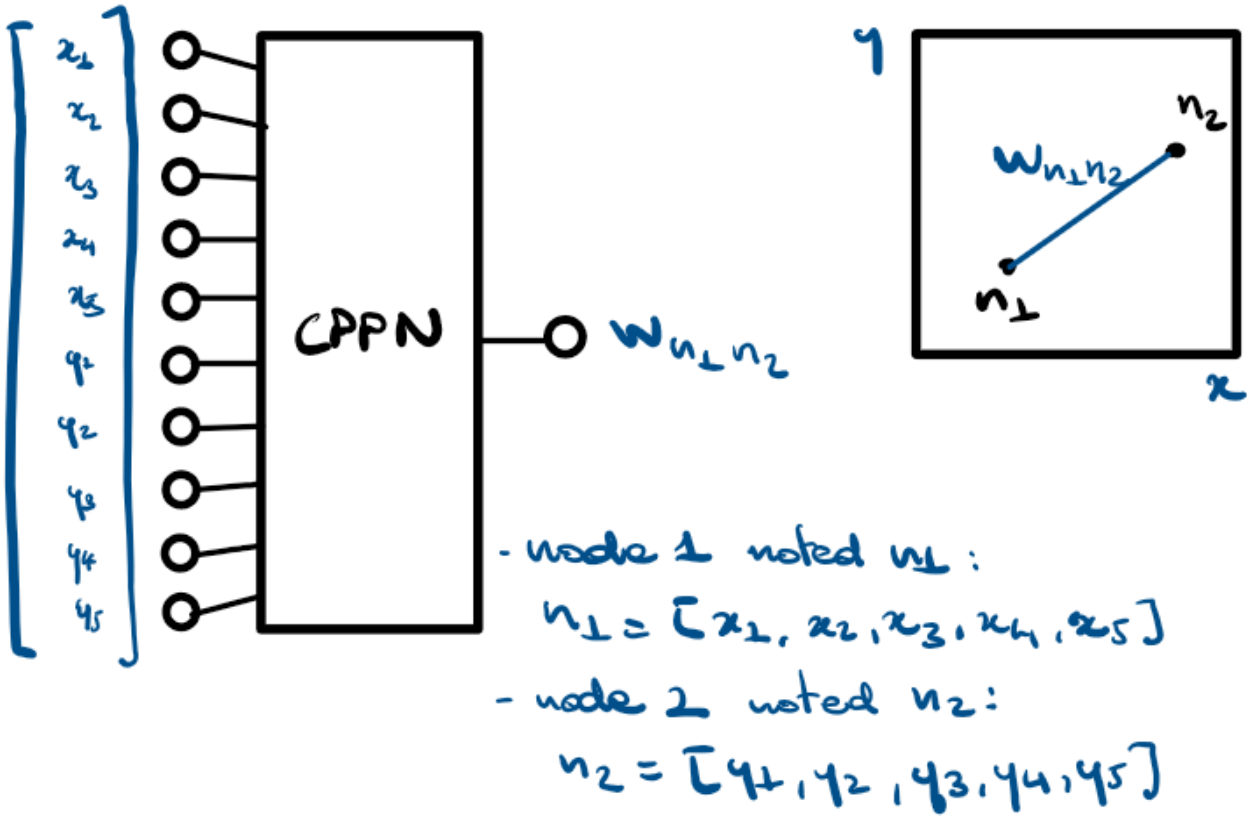
Body Producer Neural Network



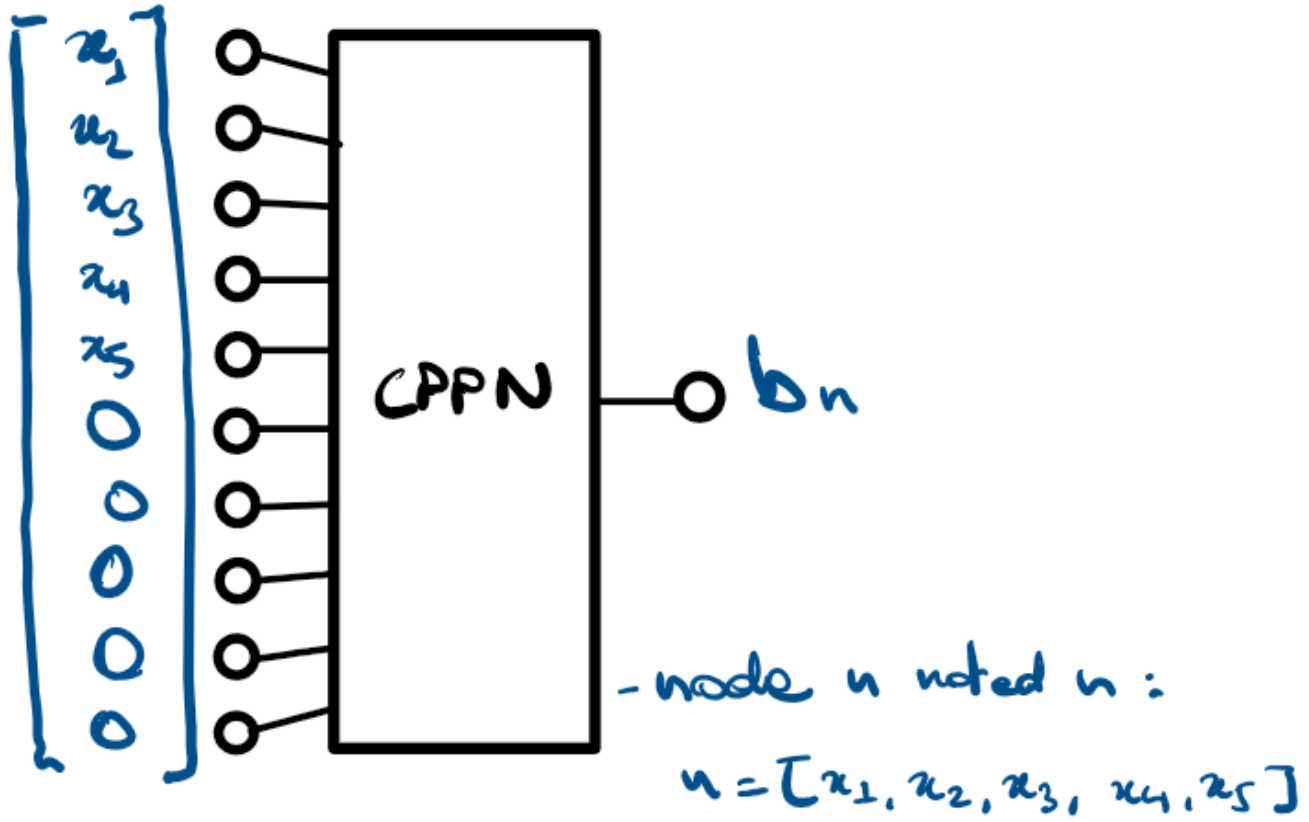
Example


0,5
1
0,33
2
3
output for voxel (i, j)
output = [0,5; 1; 0,33; 2, 3]
type = argmax (output) = 4

CPPN and Weights



CPPN and Biases



The Substrate

Cartesian product :

$E_1, E_2, \dots, E_n$ set of elements :

$E_1 \times E_2 \times \dots \times E_n = (x_1, x_2, \dots, x_n)$

for $x_1 \in E_1, x_2 \in E_2, \dots, x_n \in E_n$

Body ANN substrate :

for n node $\in$ neural network

$n = [x_1, x_2, x_3, x_4] \in E_1 \times \dots \times E_4$

How can every point represent the reality of the problem ?

Building $E_1, E_2, \dots, E_4$

$E_1 = \{0,33; 0,66; 1\}$

$E_2 = \{-1; -0,5; 0; 0,5; 1\}$

$E_3 = \{-1; -0,5; 0; 0,5; 1\}$

$E_4 = \{(\cos(k\frac{2\pi}{5}); \sin(k\frac{2\pi}{5})); k \in [\![0; 4]\!]\}$

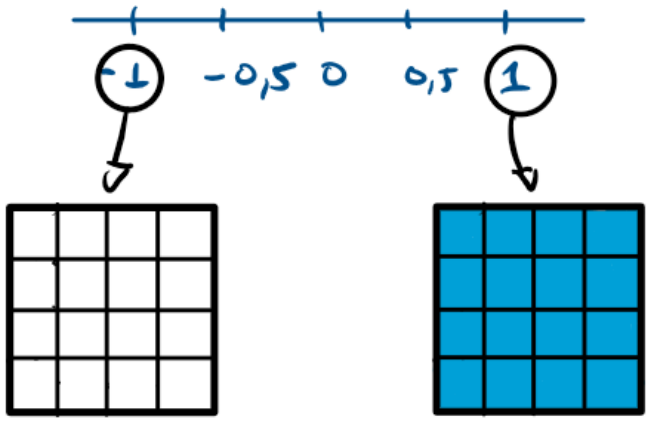
↳ $n = \left[0,66 ; -0,5 ; 1 ; \cos(2\frac{2\pi}{5}) ; \sin(2\frac{2\pi}{5})\right]$

The problem

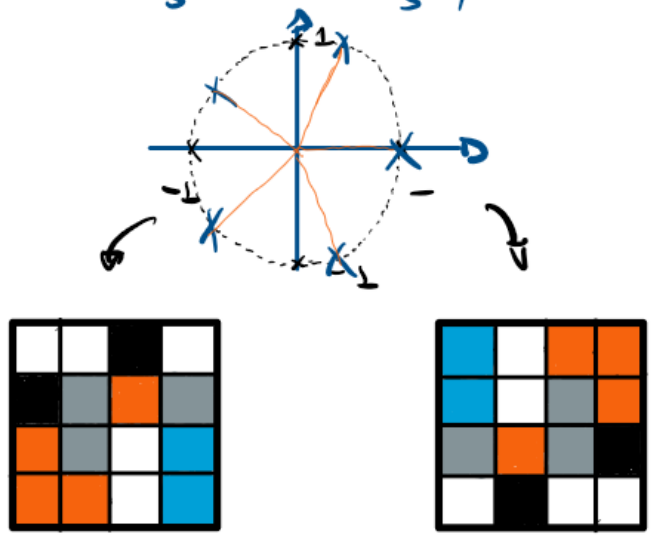
Side effect

E_4 = type of voxel

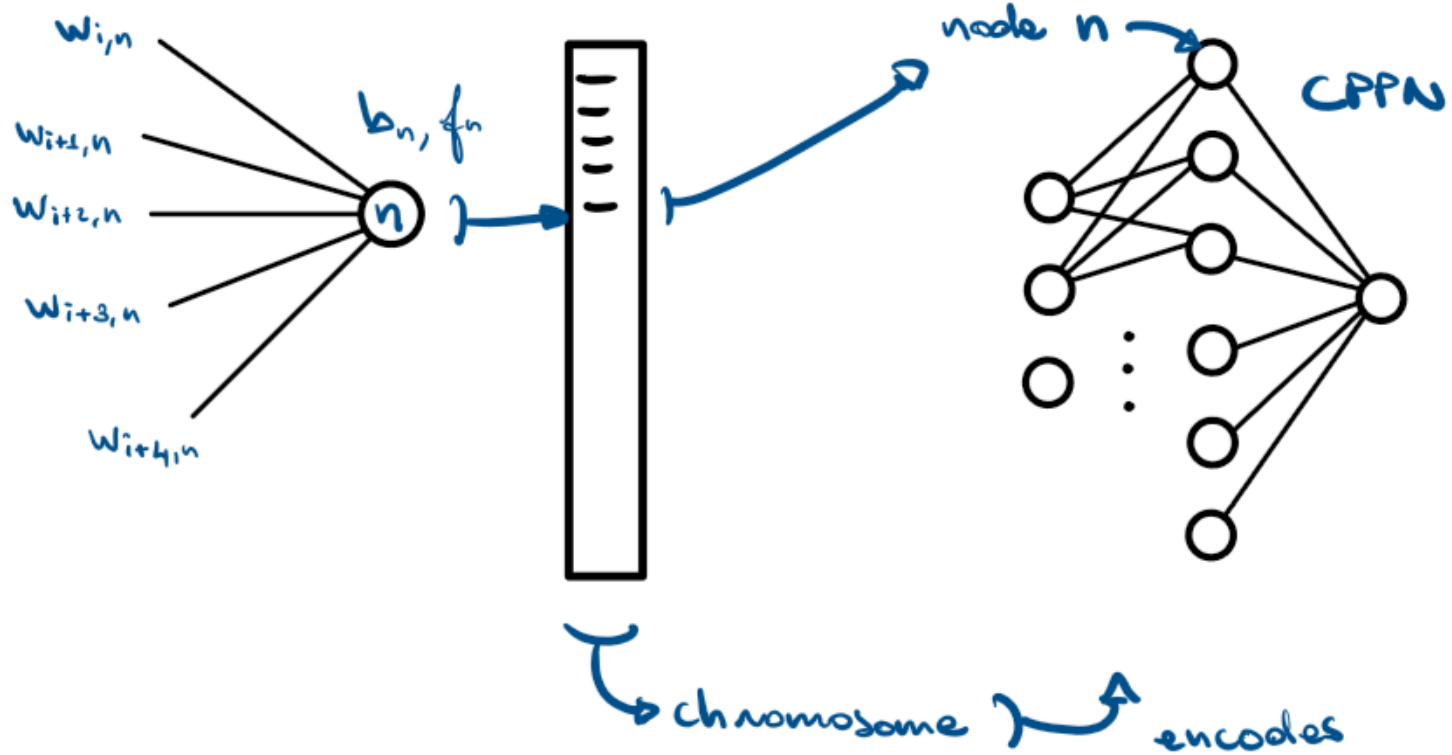
$E_4 = \{-1; -0,5; 0; 0,5; 1\}$



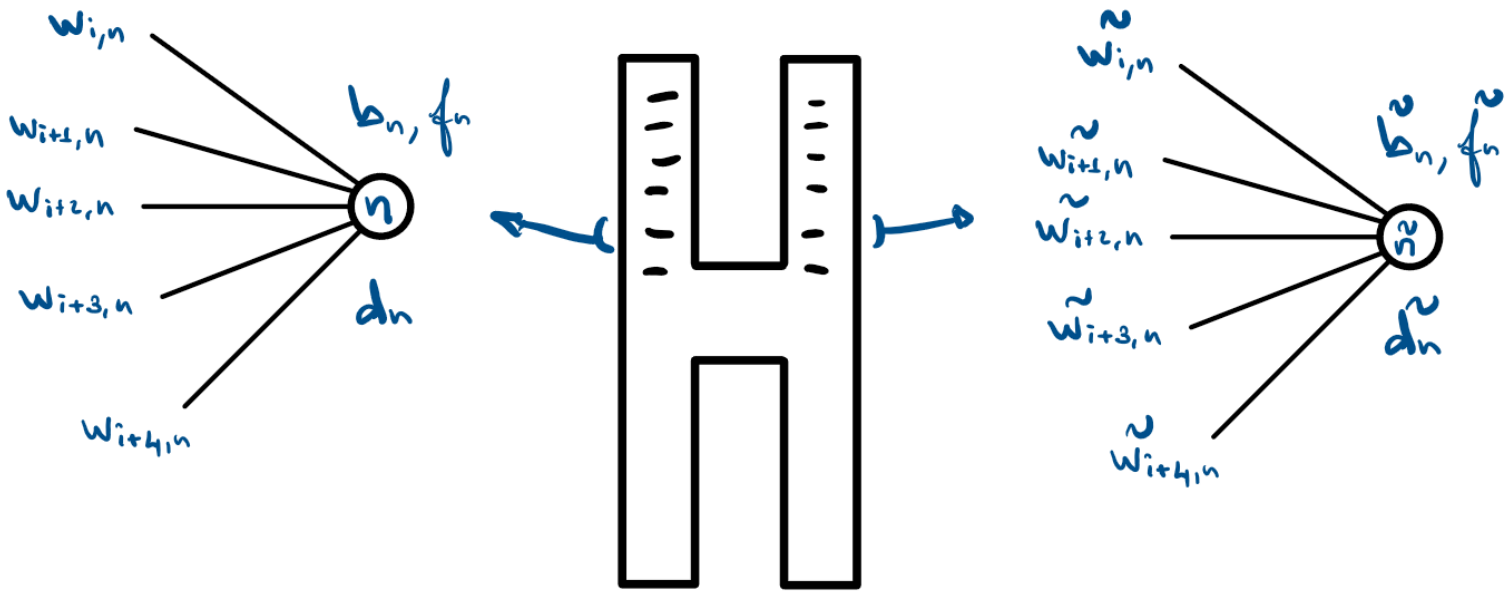
$E_4 = \{(\cos(k\frac{2\pi}{5}); \sin(k\frac{2\pi}{5})) ; k \in [\![0;4]\!]\}$



Haploidy

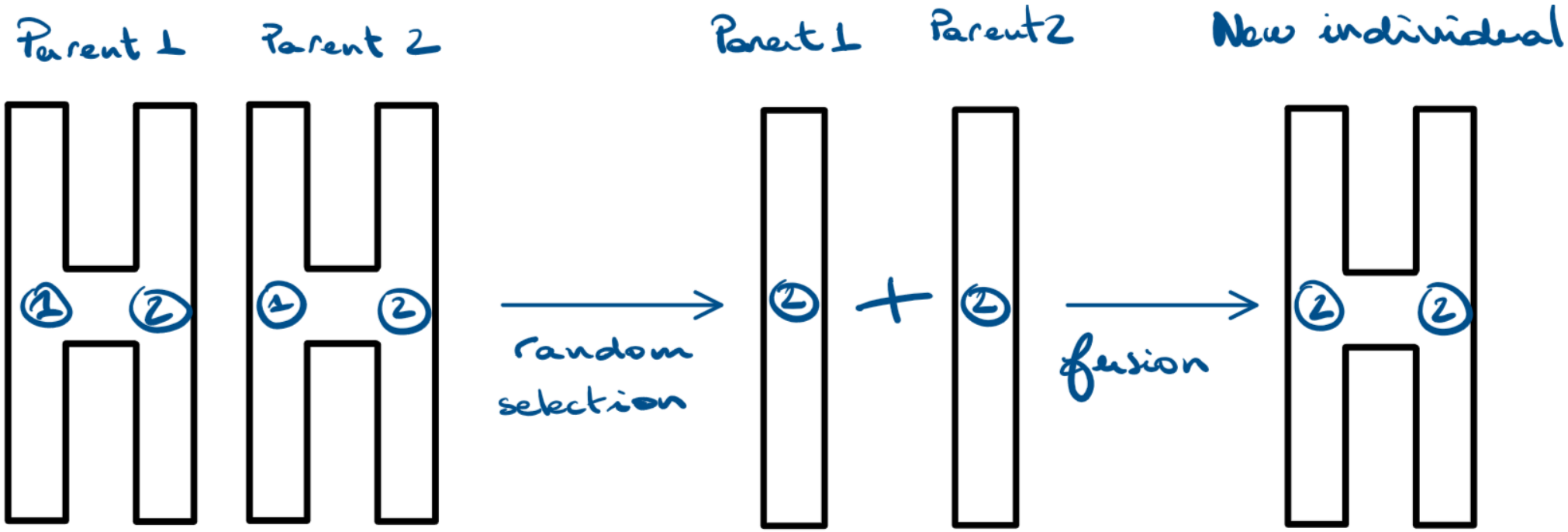

$w_{i,n}$
$w_{i+1,n}$
$w_{i+2,n}$
$w_{i+3,n}$
$w_{i+4,n}$
b_n, f_n
n
node n
CPPN
chromosome
encodes

Diploidy

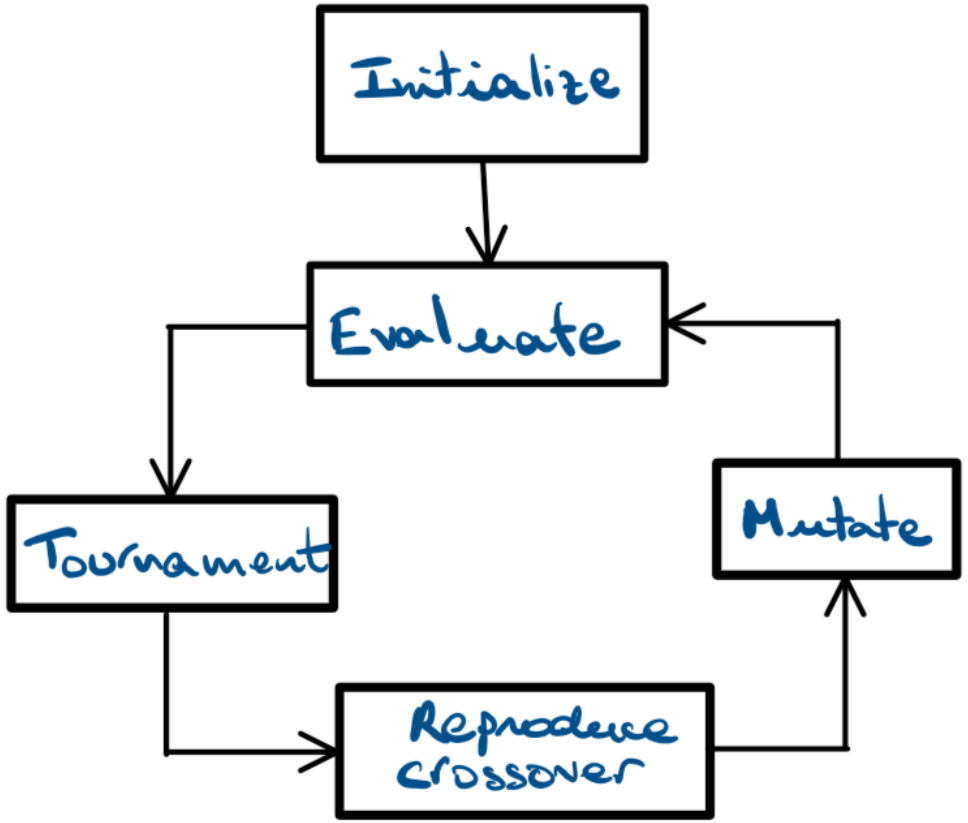


$d_n, \tilde{d}_n \in [0,5]$, compare $(d_n, \tilde{d}_n)$

Crossover


Parent 1
Parent 2
Parent 1
Parent 2
New individual
random selection
fusion

The Algotrithm



The benefits

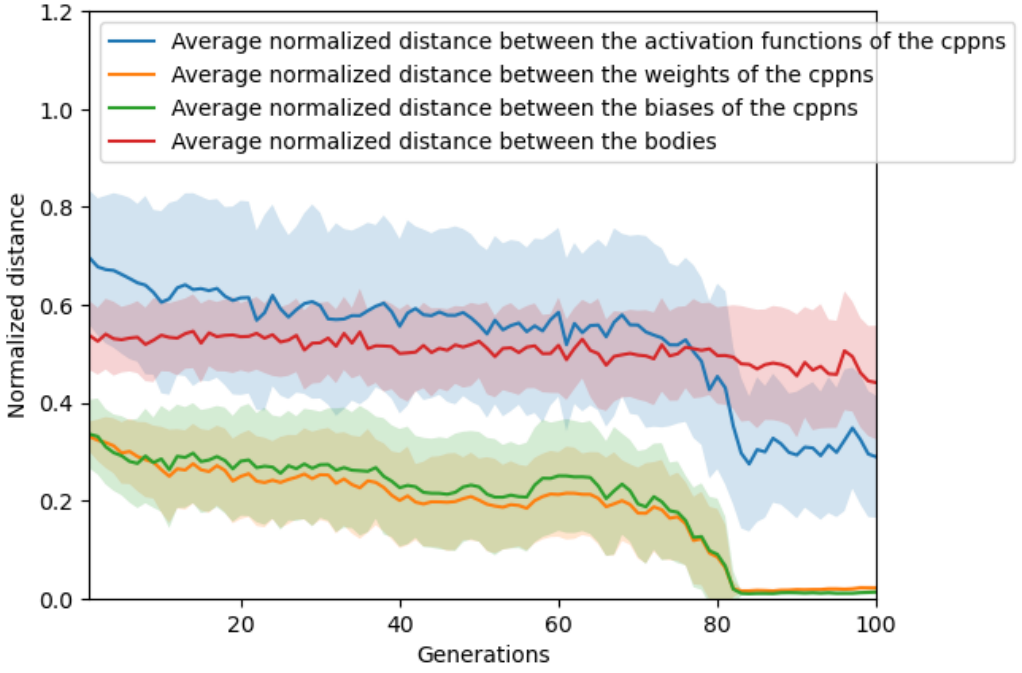
- Greater ability to search complex spaces and find better optima.
- Better adaptation to changing environments
- Less sensitive to mutation rate, i.e. to destructive mutations
- Depending on the crossover, can match the results of haploid systems on stationnary problems

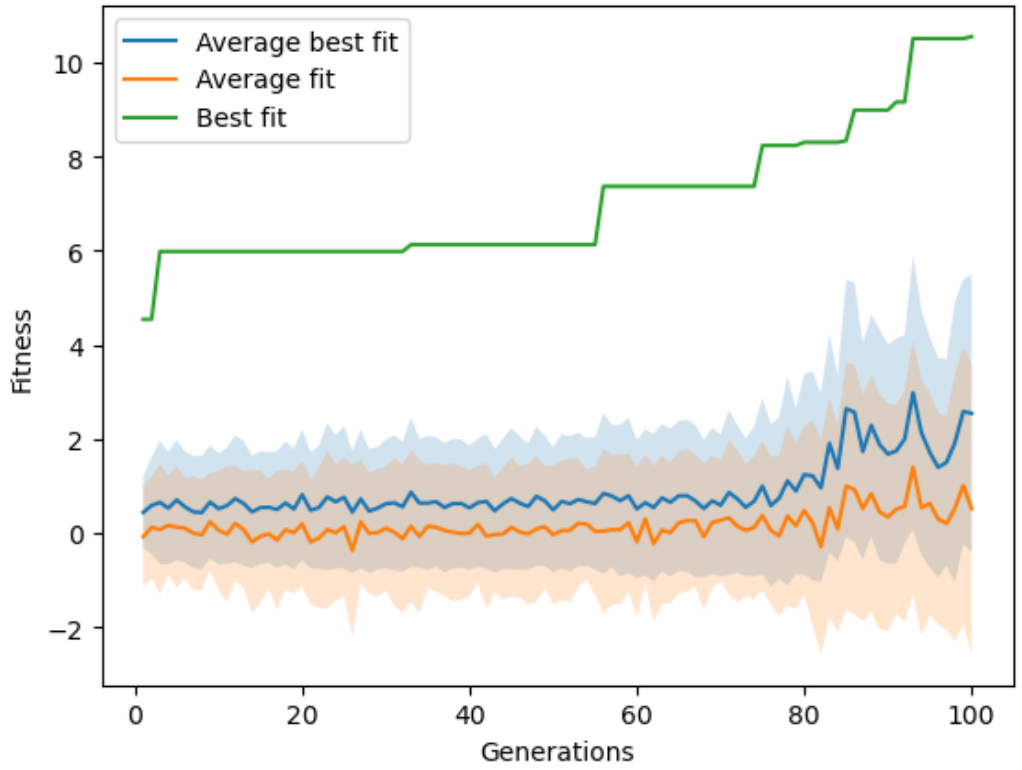
The Parameters

```
{
    "body_shape" : [5, 5],
    "max_weight" : 20,
    "max_bias" : 20,
    "response" : 1,

    "range_weight" : 12,
    "range_bias" : 12,
    "sigma_weight" : 0.5,
    "sigma_bias" : 0.3,
    "threshold_weight" : 0.1,
    "threshold_bias" : 0.1,
    "threshold_function" : 0.05,
    "threshold_dominance" : 0.1,

    "number_of_dominances" : 5,
```

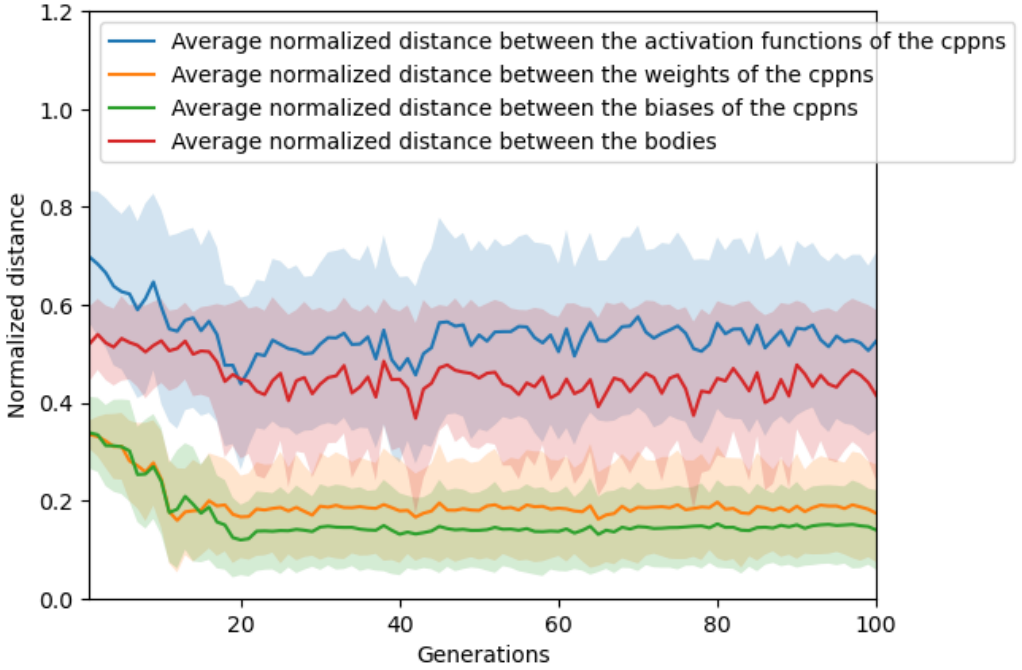
```
    "number_of_dominances" : 5,

    "generations" : 100,
    "population" : 100,
    "number_of_winner" : 1,
    "number_in_tournament" : 5,
    "number_of_elites" : 2,

    "shape_of_cppn" : [10, 4, 4, 2, 1],

    "function_pool" : ["gaussian", "sin", "tanh"],

    "n_steps" : 800,
    "number_of_reported_individuals" : 40,

    "cpus" : 7,

    "env_name" : "Walker-v0"

}
```

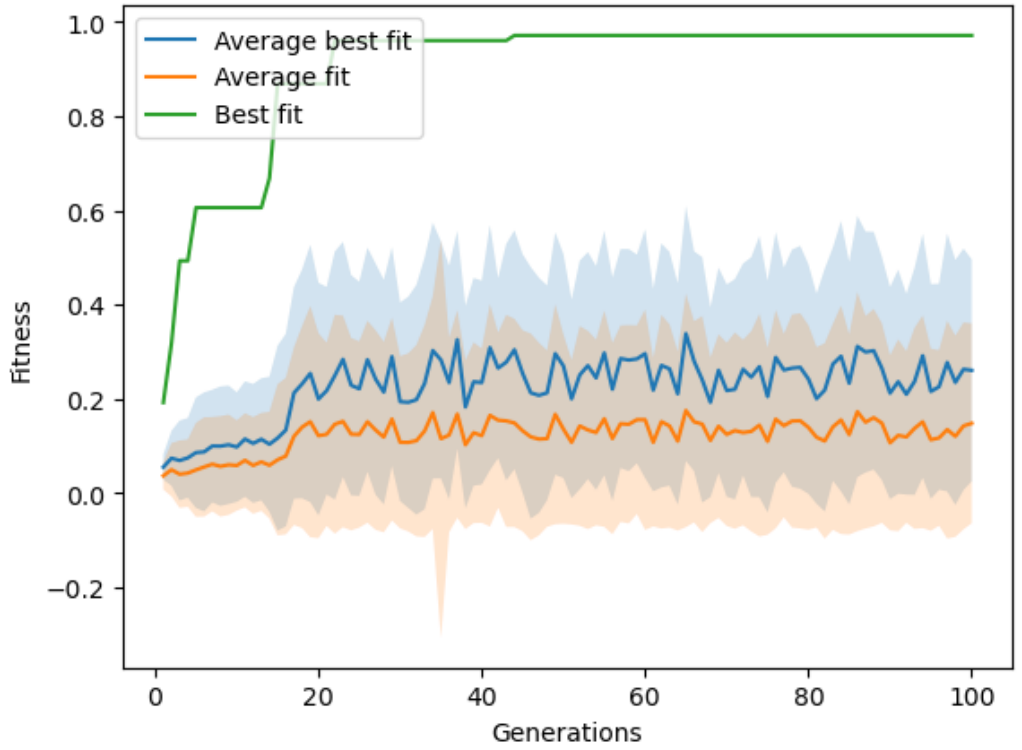
Initial Results (Stationary, Diploid, Walker-v0)



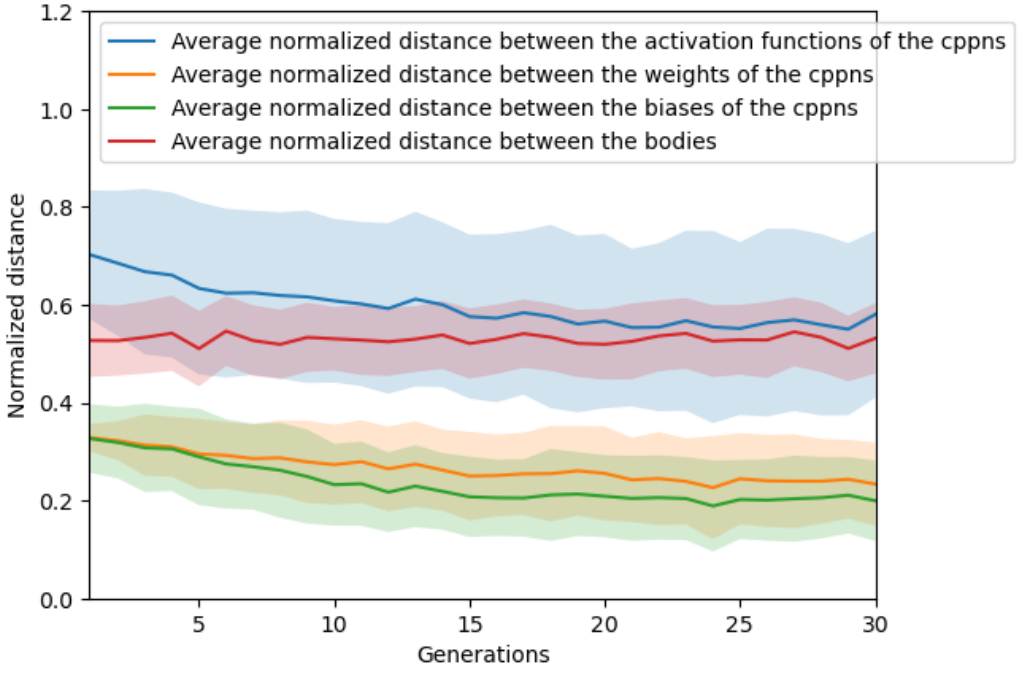


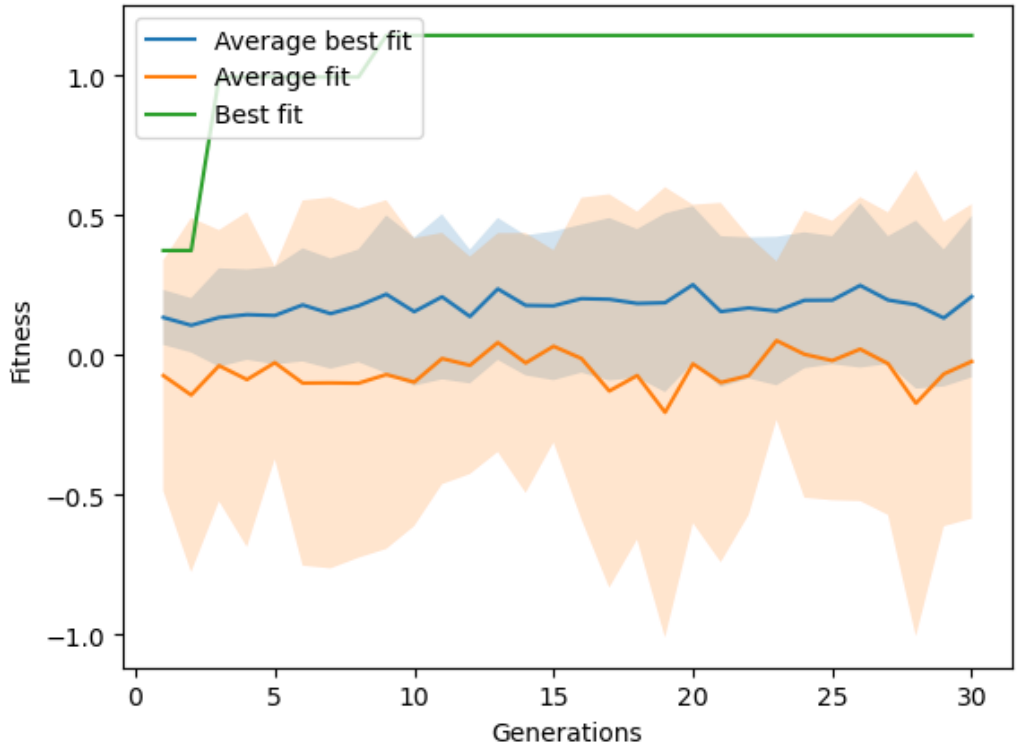
Initial Results (Stationary, Diploid, Climber-v0)





Initial Results (Stationary, Diploid, BridgeWalker-v0)





Videos

- Walker-v0 : https://www.youtube.com/playlist?list=PL-fKXFhkBjTNbn6cMiTNqUD7EEFH63u7w

- Climber-v0 : https://www.youtube.com/playlist?list=PL-fKXFhkBjTPiS3WBmSGBMzGQ9fQO4ZoW

- BridgeWalker-v0 : https://www.youtube.com/playlist?list=PL-fKXFhkBjTM0OOiY-H2yw-BHk5M6r9cK

Next

→ Experiment with haploid individual and compare on stationnary / changing environment

→ Build and compare other forms of substrate at generation 0.

Thank you

ありがとうございます