IA pour la Robotique

M2 ANDROIDE 2021 – 2022

Encadrant : Benoît Girard

SORBONNE UNIVERSITÉ

CRÉATEURS DE FUTURS DEPUIS 1257

Self-Improving Reactive Agents Based On Reinforcement Learning, Planning and Teaching

Long-Ji LIN (1992)

LIM Vincent

WANG Tianyu

SOMMAIRE

SORBONNE UNIVERSITÉ
CRÉATEURS DE FUTURS DEPUIS 1257

1

Introduction

Contexte

- Etat de l'art en 1992 :
  - Adaptative Heuristic Critic (Sutton, 1984 et Barto, 1990)
  - Connectionist error backpropagation algorithm (Rumelhart, 1986)
  - Temporal difference learning (Sutton, 1988)
  - Q-Learning (Watkins, 1989)

- Testées sur des tâches simples

- Convergence lente

Objet de l'article

- Deux aspects :
  - Analyse des performances du RL sur des tâches plus complexes
  - Chercher des méthodes pour rendre la convergence plus rapide

- Comparer QCON et AHCON

- Trois méthodes pour améliorer la convergence :
  - Experience replay
  - Learning action model
  - Teaching

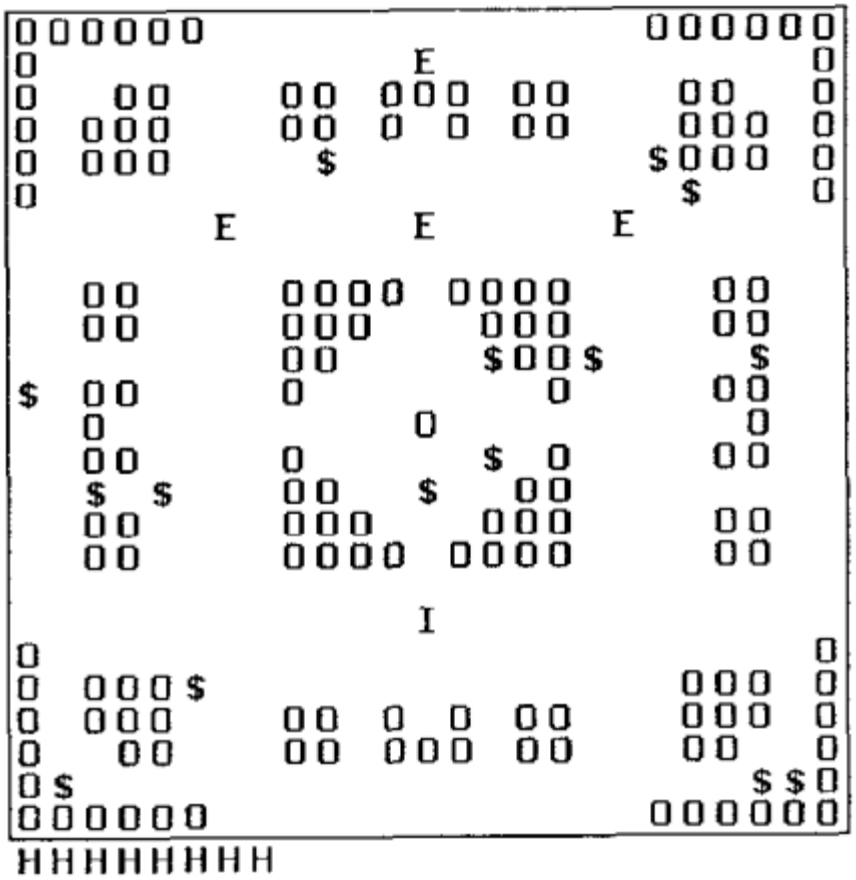
Les objectifs

- Reproduire l'environnement de simulation
- Implémenter l'algorithme d'apprentissage QCON ainsi que la variante Experience Replay
  - Q-Learning
  - Réseau de neurones
- Reproduire le protocole expérimental et obtenir des résultats similaires à ceux de l'article

2

L’environnement

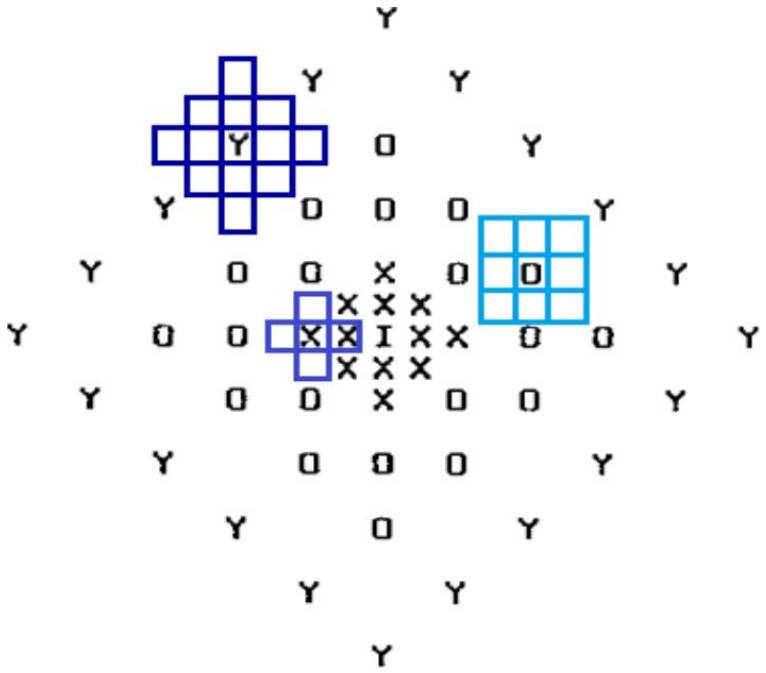
L’environnement



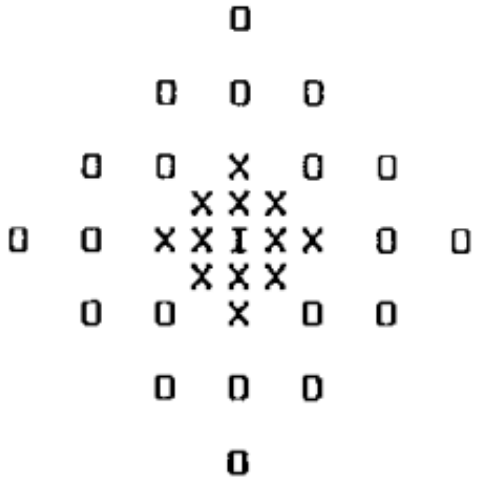
Représentation de l’environnement

- Grille 25x25
- 1 agent (I)
- 4 ennemies (E)
- 15 nourritures ($)
- Obstacles (O)
- Energie de l’agent (H)
- Déplacement possible dans 4 directions
- Trois rewards :
  - + 0.4 lorsque l’agent collecte de la nourriture
  - - 1.0 lorsque l’agent meurt
  - 0.0 sinon

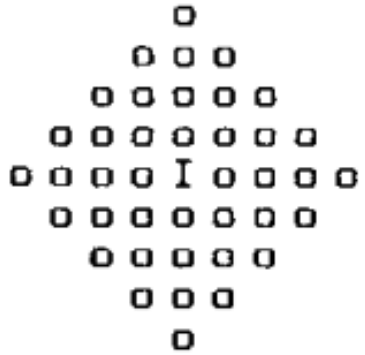
Les capteurs



Capteurs de nourritures



Capteurs d’ennemies

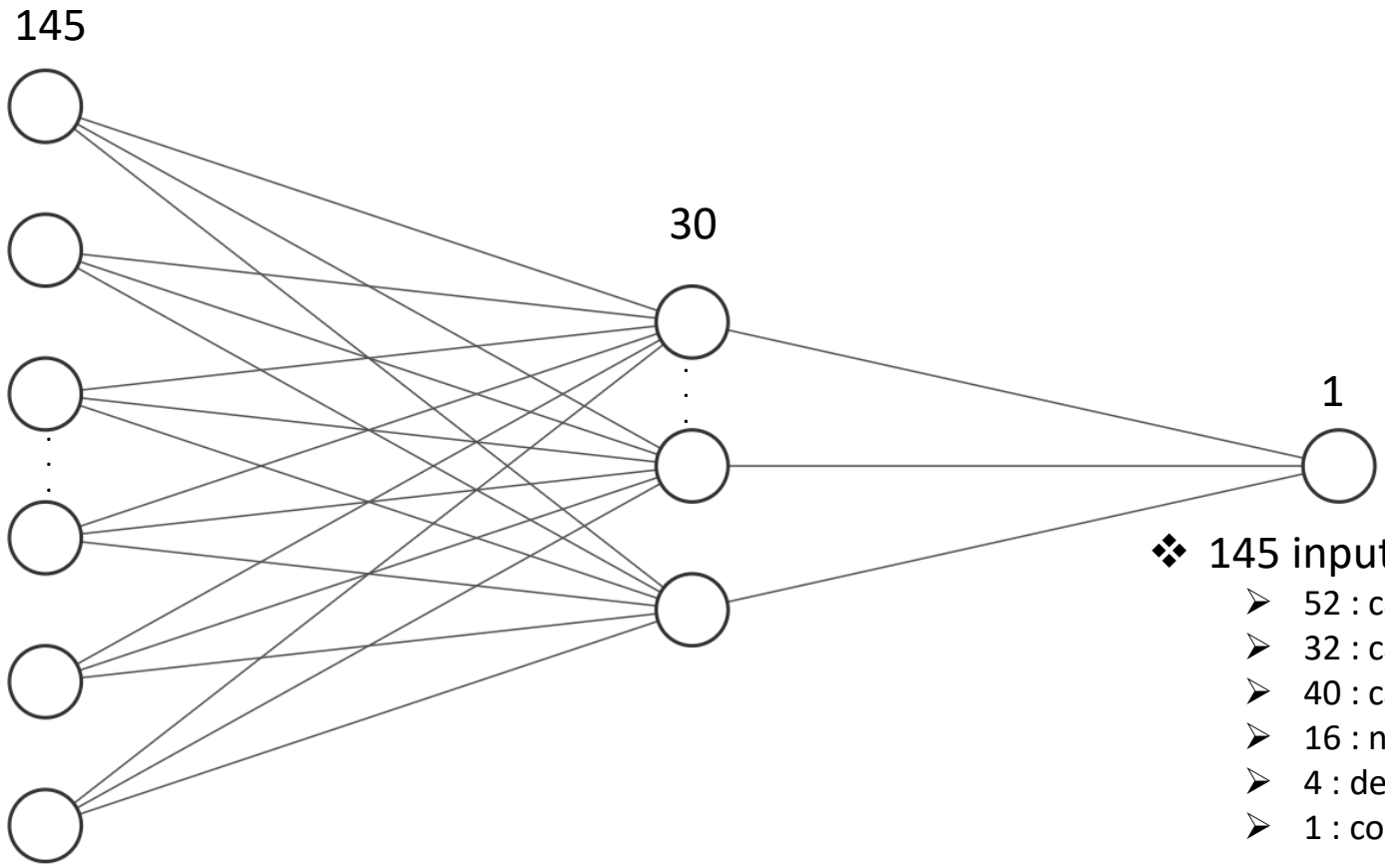


Capteurs d’obstacles

3

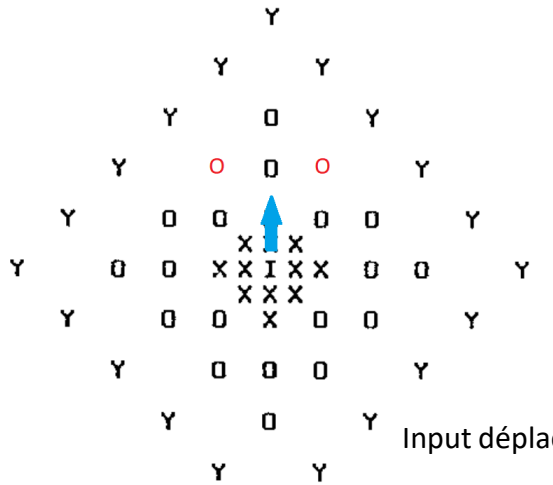
Le réseau de neurones

Le réseau

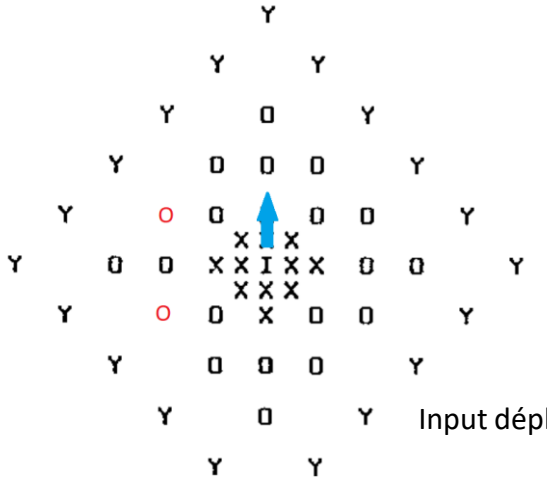


- 145 inputs :
  - 52 : capteurs de nourritures
  - 32 : capteurs d'ennemies
  - 40 : capteurs d'obstacles
  - 16 : niveau d'énergie
  - 4 : dernière action
  - 1 : collision à la dernière action
- 30 neurones couche caché
- 1 output :
  - « Mérite » de chaque action

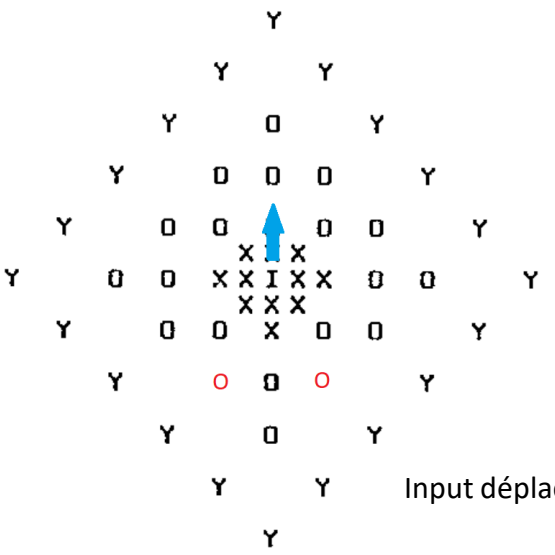
Rotation des inputs



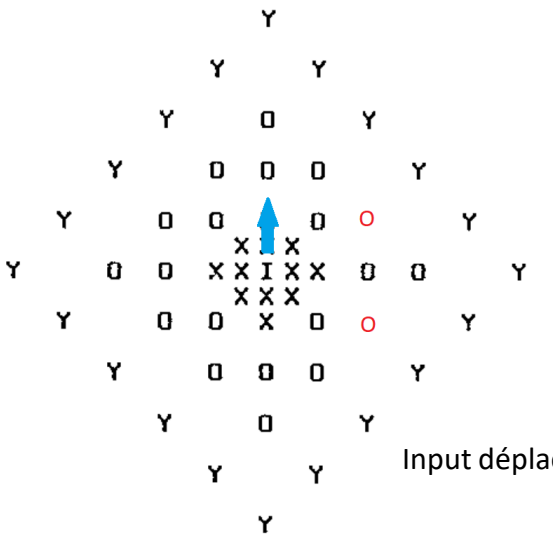
Input déplacement Nord



Input déplacement Est



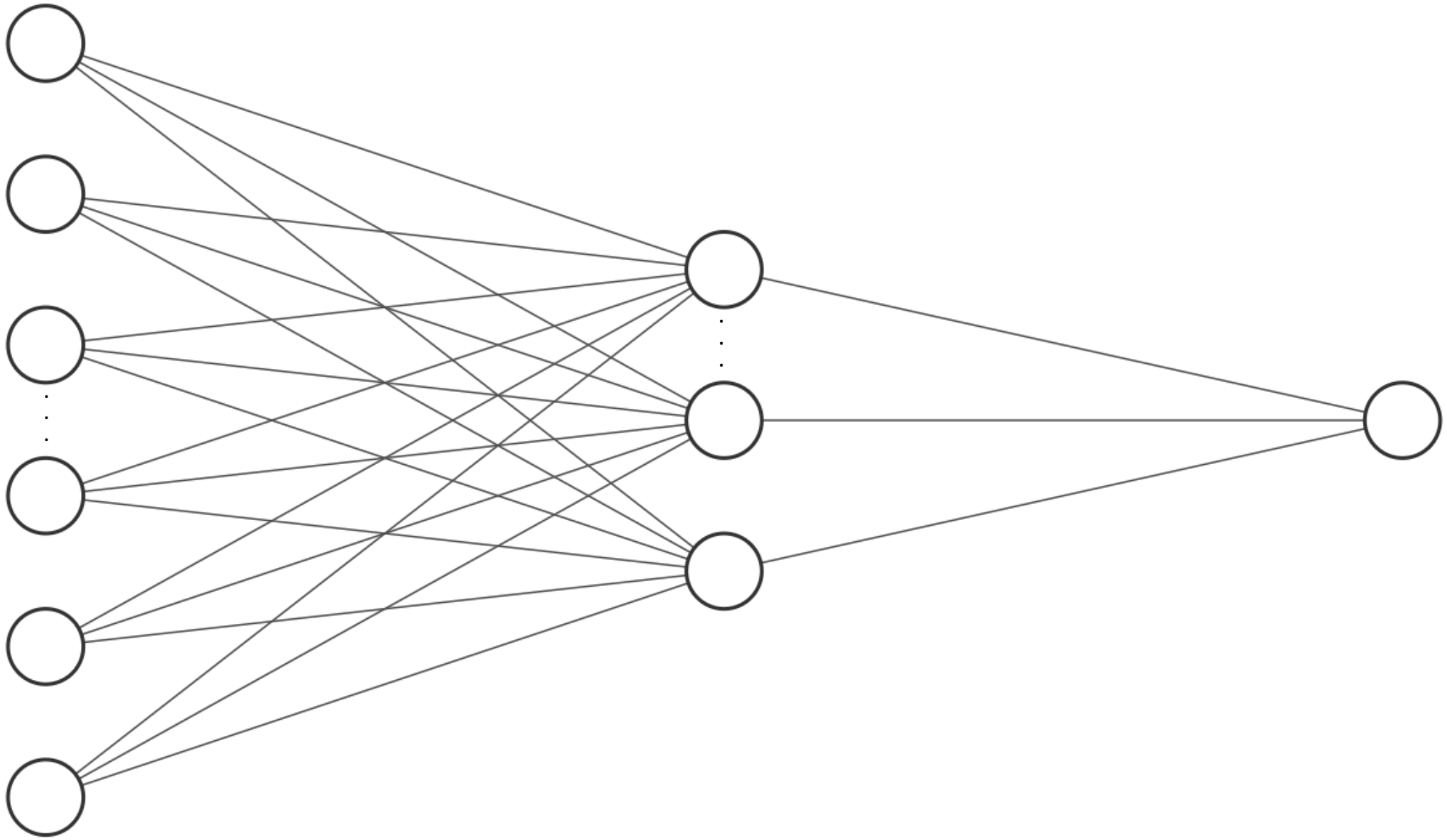
Input déplacement Sud



Input déplacement Ouest

La propagation

Fonction d'activation : $f(x) = (\frac{1}{1+e^{-x}} - 0.5) \cdot 2$

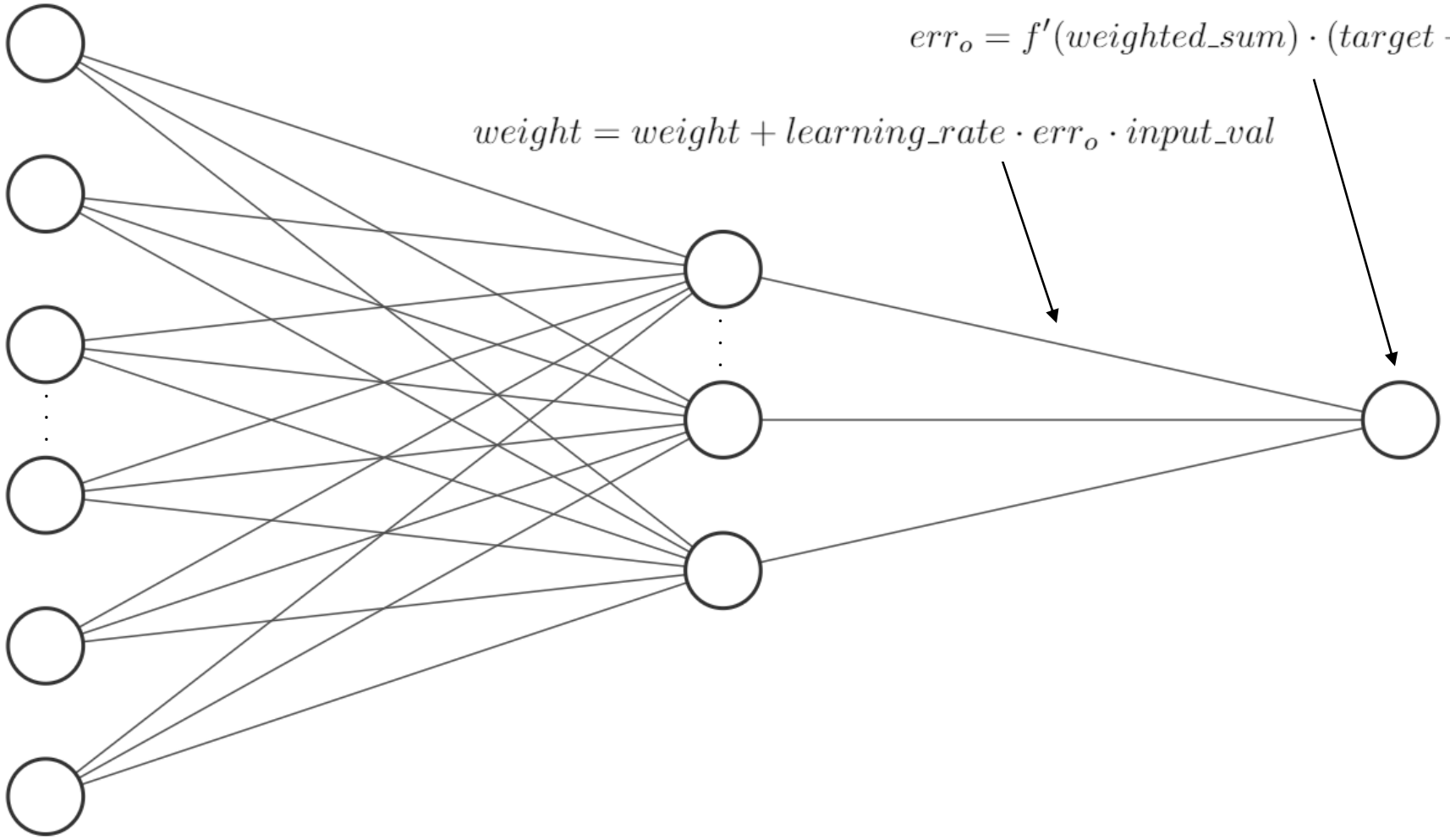


La rétropropagation

Fonction d'activation : $f(x) = (\frac{1}{1+e^{-x}} - 0.5) \cdot 2$

$$err_o = f'(weighted\_sum) \cdot (target - output)$$

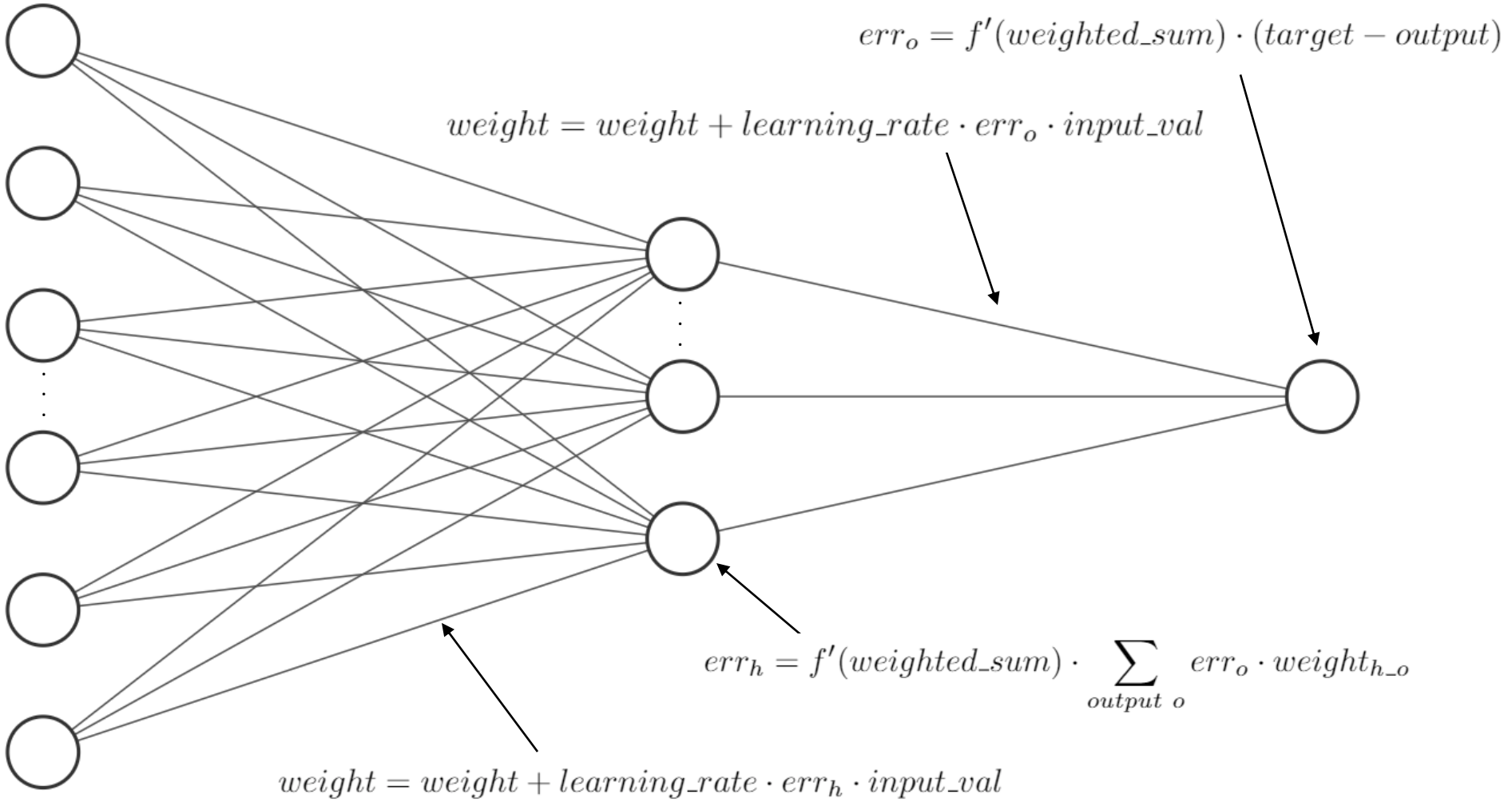
$$weight = weight + learning\_rate \cdot err_o \cdot input\_val$$



https://helios2.mi.parisdescartes.fr/~bouzy/Doc/AA1/ReseauxDeNeurones1.pdf

La rétropropagation

Fonction d'activation : $f(x) = (\frac{1}{1+e^{-x}} - 0.5) \cdot 2$

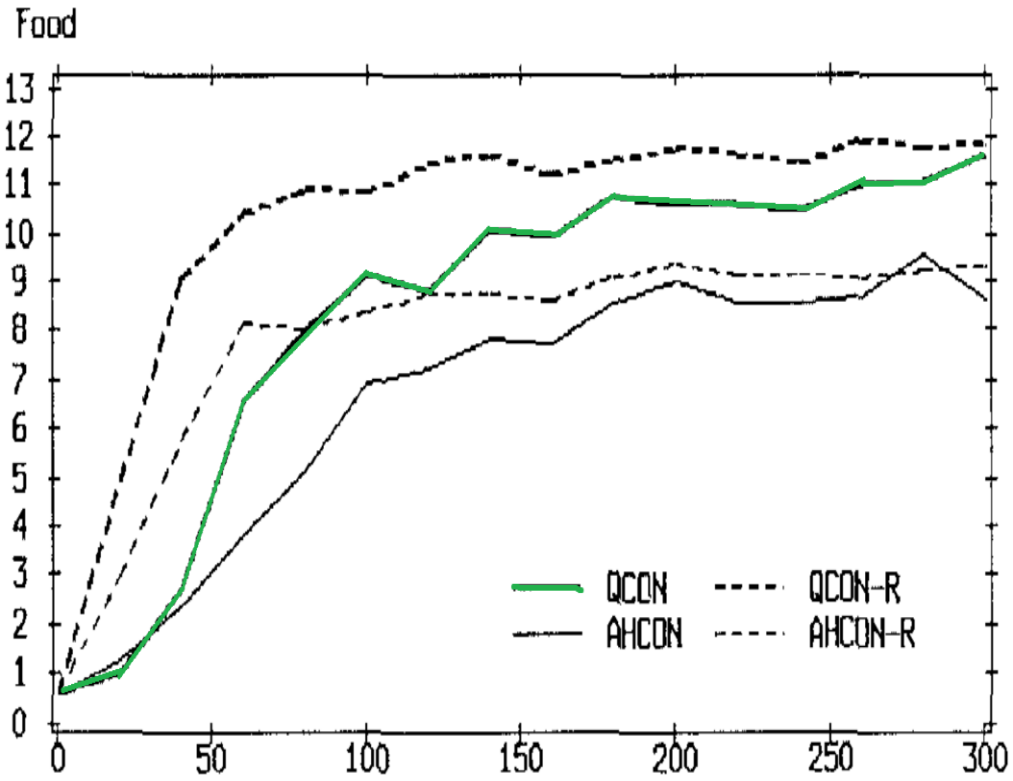


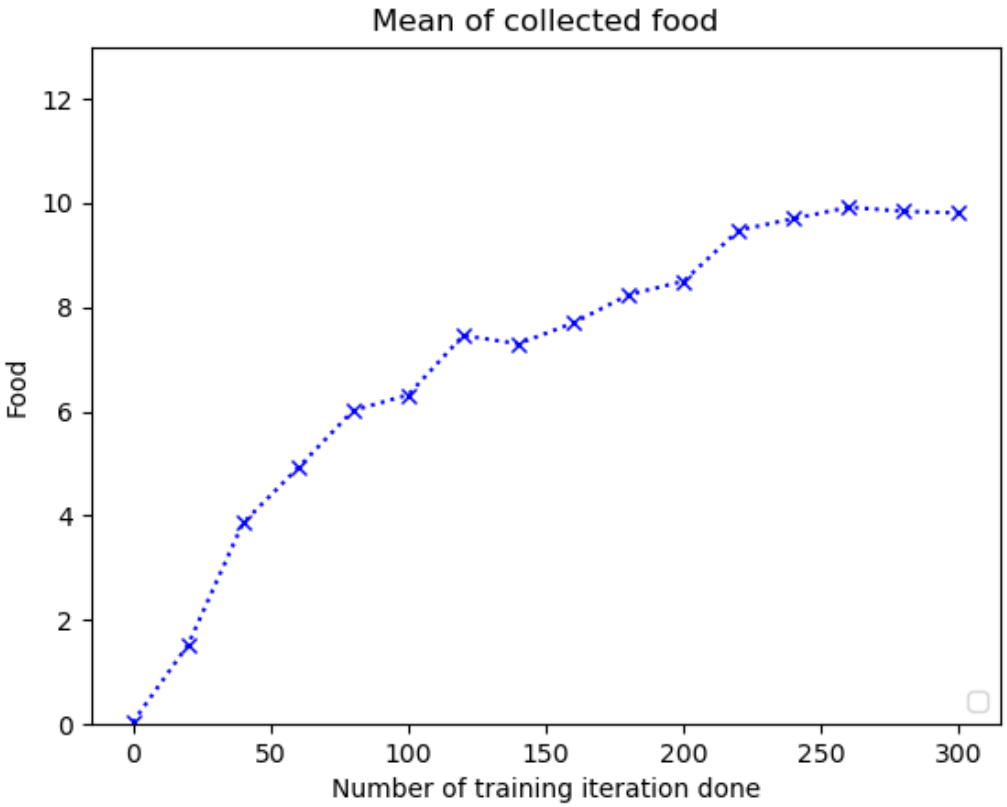
https://helios2.mi.parisdescartes.fr/~bouzy/Doc/AA1/ReseauxDeNeurones1.pdf

4

Résultats

Résultats


Food
13
12
11
10
9
8
7
6
5
4
3
2
1
0
0
50
100
150
200
250
300
QCON
QCON-R
AHCON
AHCON-R

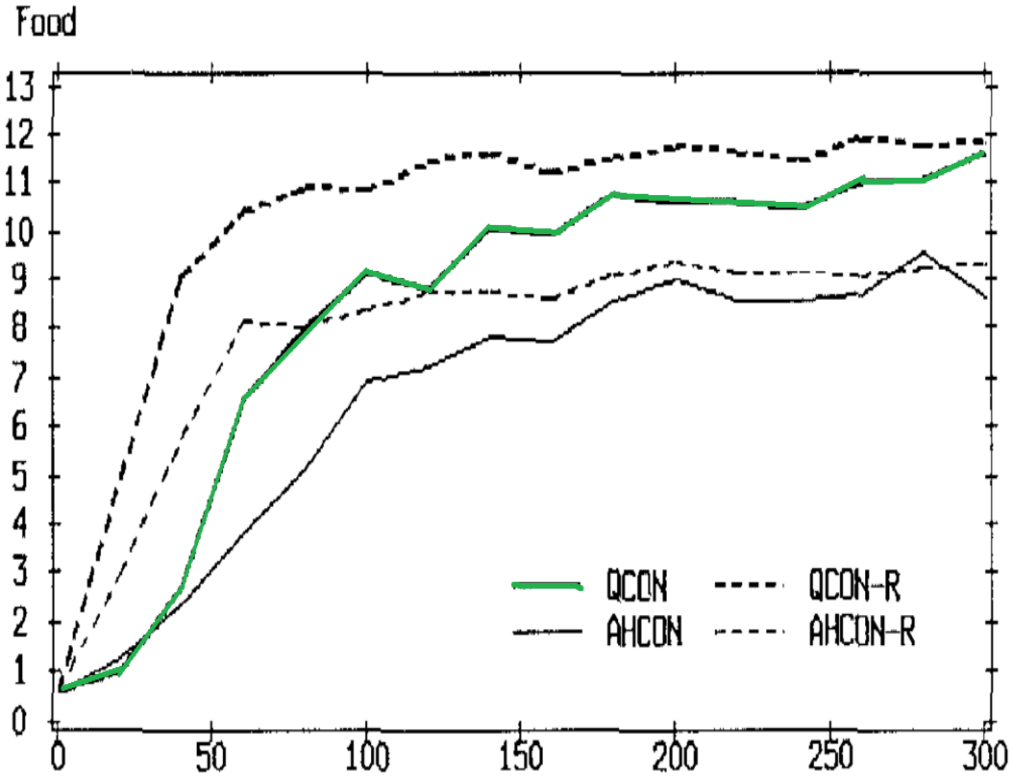

Mean of collected food
Food
12
10
8
6
4
2
0
0
50
100
150
200
250
300
Number of training iteration done

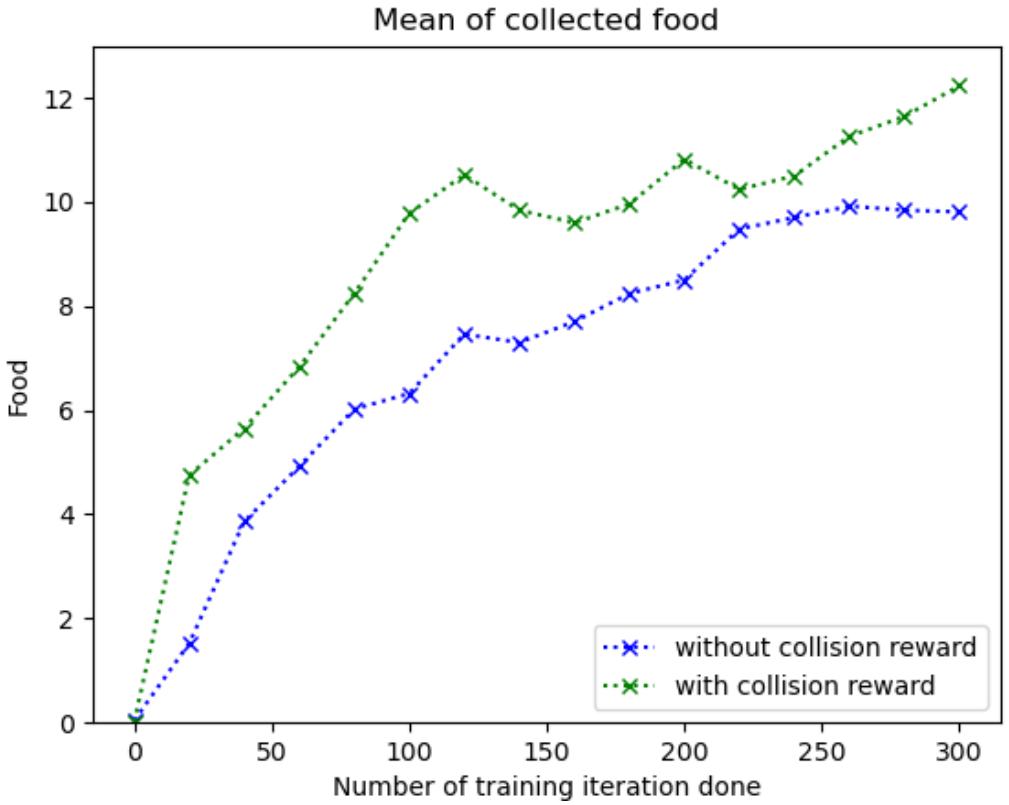
Nouvelles rewards

- “*Programming robots using reinforcement learning and teaching*”, Long-Ji LIN, 1991

- Cinq rewards :
  - + 0.4 lorsque l’agent collecte de la nourriture
  - - 1.0 lorsque l’agent meurt
  - - 0.5 lorsque l’agent rentre dans un mur
  - - 0.5 lorsque l’agent revient à son ancienne position
  - 0.0 sinon

Résultats


Food
13
12
11
10
9
8
7
6
5
4
3
2
1
0
0
50
100
150
200
250
300
QCON
QCON-R
AHCON
AHCON-R


Mean of collected food
Food
12
10
8
6
4
2
0
0
50
100
150
200
250
300
Number of training iteration done
without collision reward
with collision reward

Merci de votre attention !
Des questions ?

SORBONNE UNIVERSITÉ
CRÉATEURS DE FUTURS
DEPUIS 1257