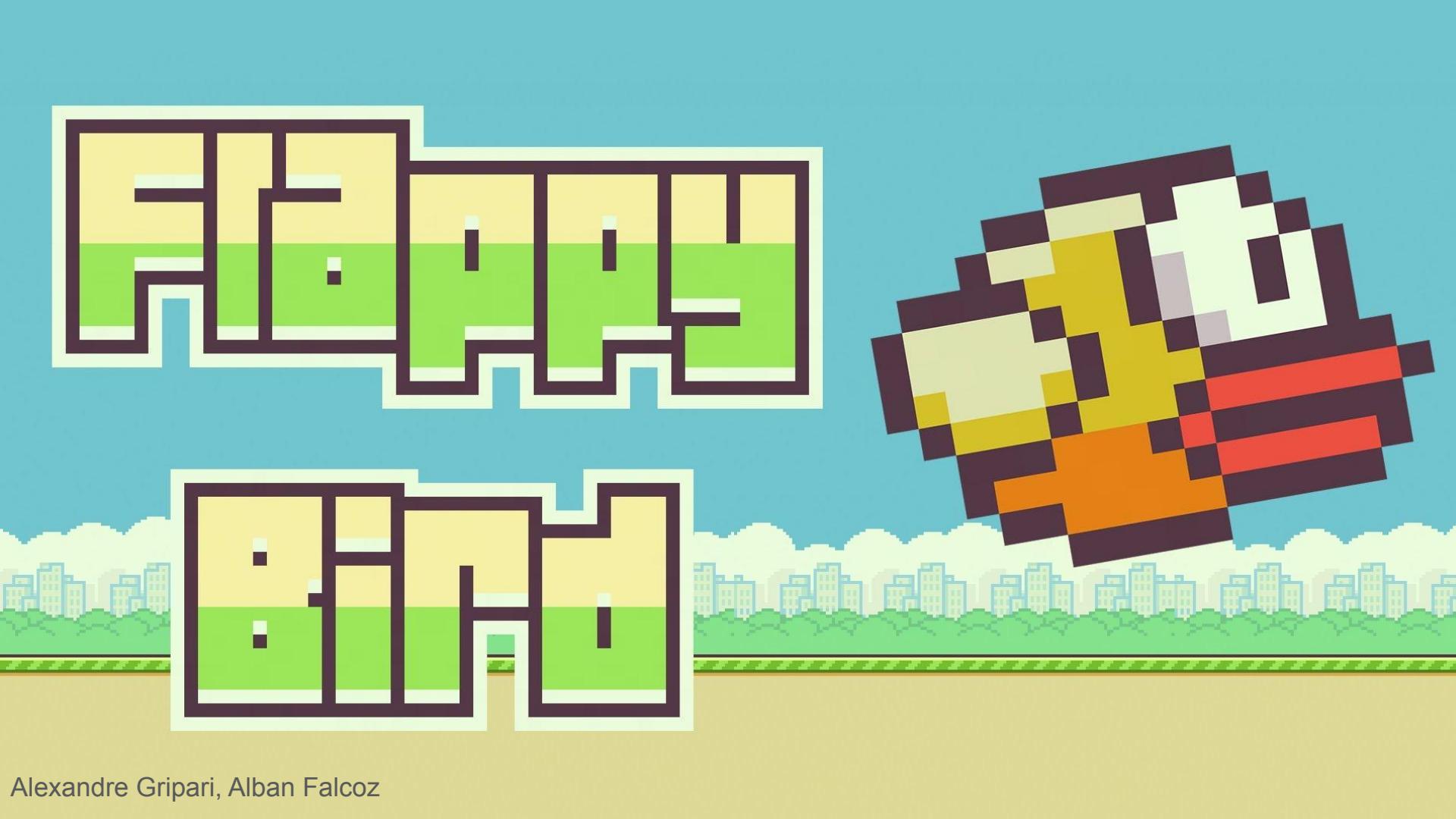
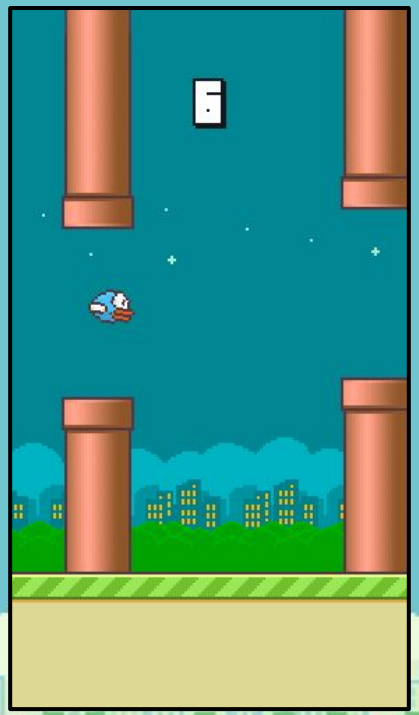


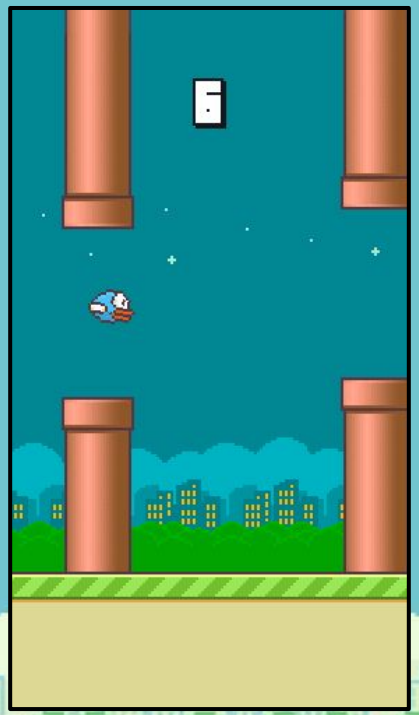
Alexandre Gripari, Alban Falcoz

RL appliqué à Flappy Bird



- 3 observations en entrées : **dx, dy, vel_y**
  - dx : distance en X jusqu'au prochain tuyau.
  - dy : distance en Y jusqu'au prochain tuyau.
  - vel_y : Vitesse verticale de l'oiseau.
- 2 actions en sorties : **flap, skip**
  - flap (1) : saut
  - skip (0) : ne rien faire
- Jeu infini mais avec un nombre d'état fini => possibilité d'utiliser Q-Learning

Q-Learning



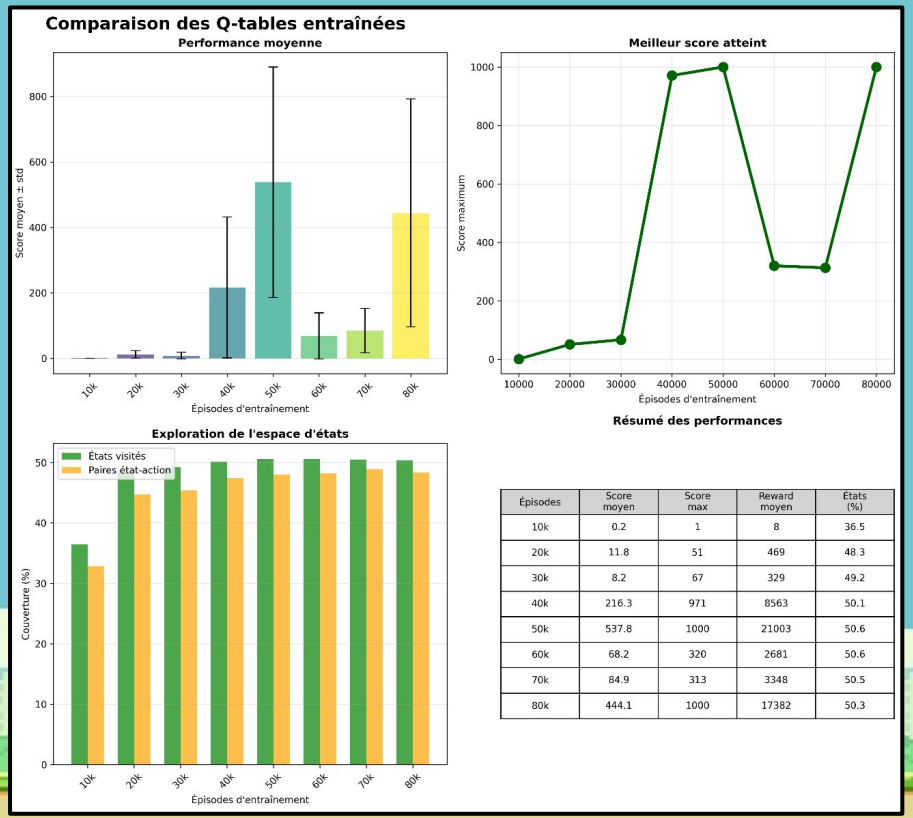
- **Nombre d’états fini, mais combien ?**
- Plages de valeurs :
  - dx ∈ [0, 212]
  - dy ∈ [-104, 256]
  - vel_y ∈ [-8, 10]
- → Espace d’état fini : (213 × 361 × 19) **1,6 million** de combinaisons possibles × 2 = **2.9 million** de paires états-actions.
- Discrétisation :
  - dx ∈ [0, 23]
  - dy ∈ [0, 35]
  - vel_y ∈ [0, 4]
- (24*36*5) = 4320 * 2 = **8640**

Reward function and hyper parameters

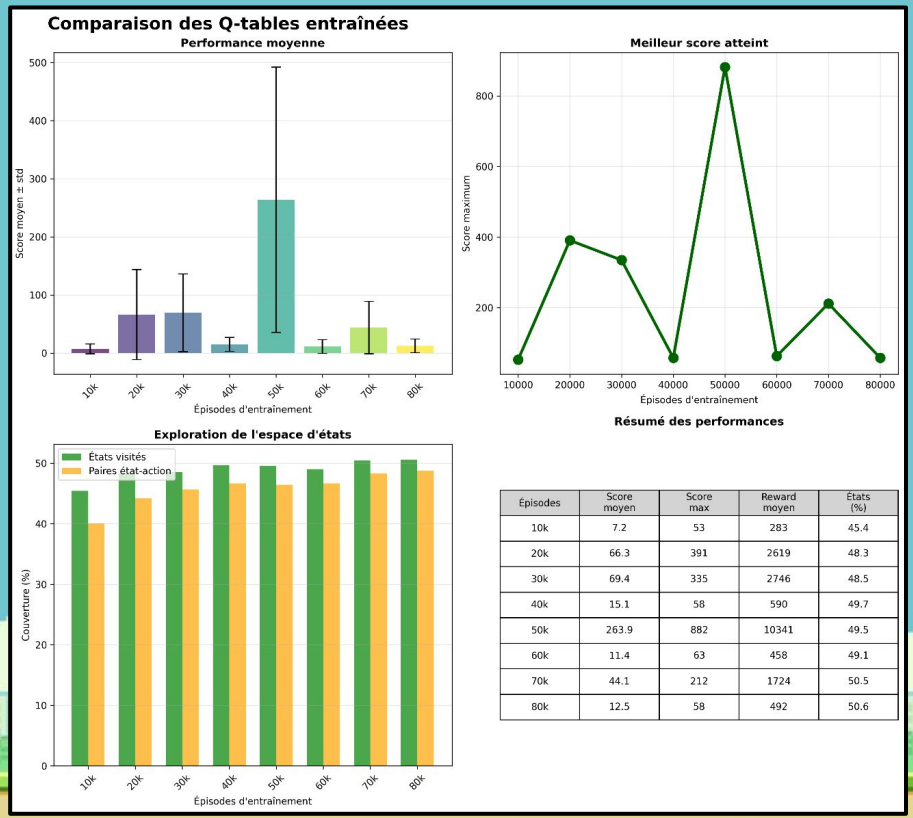
```python
@dataclass
class RewardConfig:
    frame_alive: float = 0.1
    near_pipe_gap: float = 0.2
    pass_near_pipe: float = 5.0
    pass_pipe: float = 20.0
    flap: float = -0.05
    die: float = -10.0
```

```python
@dataclass  9 usages  Alexandre-Gripari
class TrainingConfig:
    """Configuration for Q-learning training"""
    # Training hyperparameters
    episodes: int = 50000
    alpha: float = 1.0
    gamma: float = 0.95
    epsilon_start: float = 1.0
    epsilon_end: float = 0.005
    epsilon_decay_power: float = 1.0
    use_linear_decay: bool = True
```

Some experiments

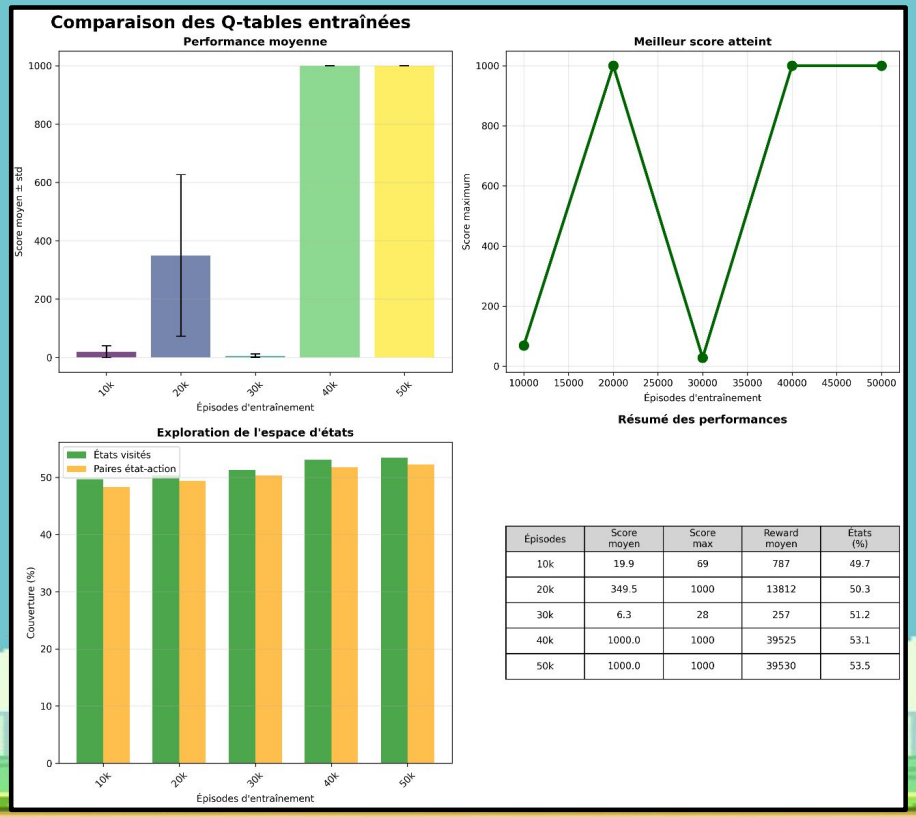


Épisodes	Score moyen	Score max	Reward moyen	États (%)
10k	0.2	1	8	36.5
20k	11.8	51	469	48.3
30k	8.2	67	329	49.2
40k	216.3	971	8563	50.1
50k	537.8	1000	21003	50.6
60k	68.2	320	2681	50.6
70k	84.9	313	3348	50.5
80k	444.1	1000	17382	50.3



Épisodes	Score moyen	Score max	Reward moyen	États (%)
10k	7.2	53	283	45.4
20k	66.3	391	2619	48.3
30k	69.4	335	2746	48.5
40k	15.1	58	590	49.7
50k	263.9	882	10341	49.5
60k	11.4	63	458	49.1
70k	44.1	212	1724	50.5
80k	12.5	58	492	50.6

Some experiments

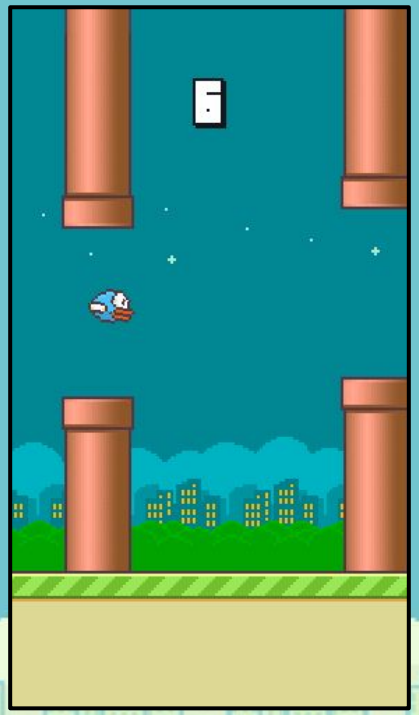


Épisodes	Score moyen	Score max	Reward moyen	États (%)
10k	19.9	69	787	49.7
20k	349.5	1000	13812	50.3
30k	6.3	28	257	51.2
40k	1000.0	1000	39525	53.1
50k	1000.0	1000	39530	53.5

- 16 * 28 * 3 = 1 344
- 1 344 * 2 = 2688

From Q-Learning to Deep Q-Learning

Principaux défis relevés par Deep Q Learning

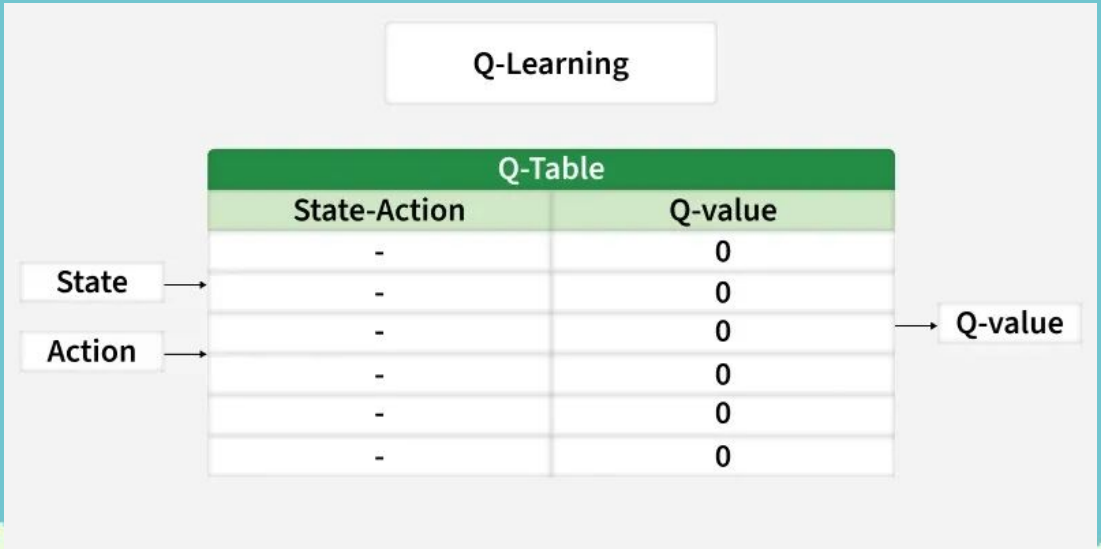


- **La taille de la Q-table présente une limite** : plus le nombre d’états augmente, plus sa dimension devient importante.

- Dans un espace d’états continu, **une discrétisation est nécessaire**, car la Q-table ne peut représenter que les états discrétisés.

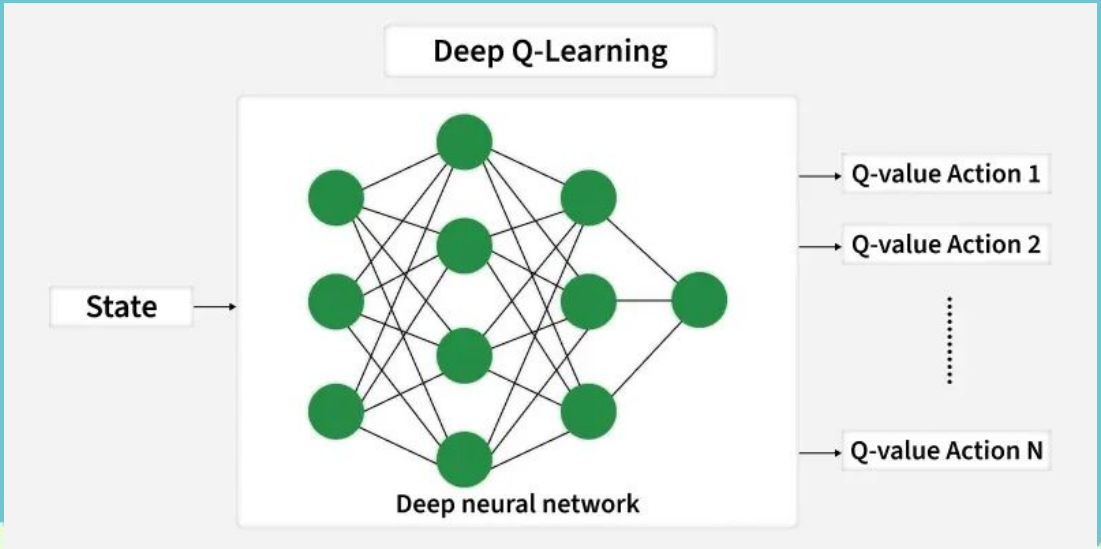
Les principaux composants de Deep Q-Network

Architecture Deep Q-Network



- Utilise une **Q-Table** comme Q value function pour approximer les q values

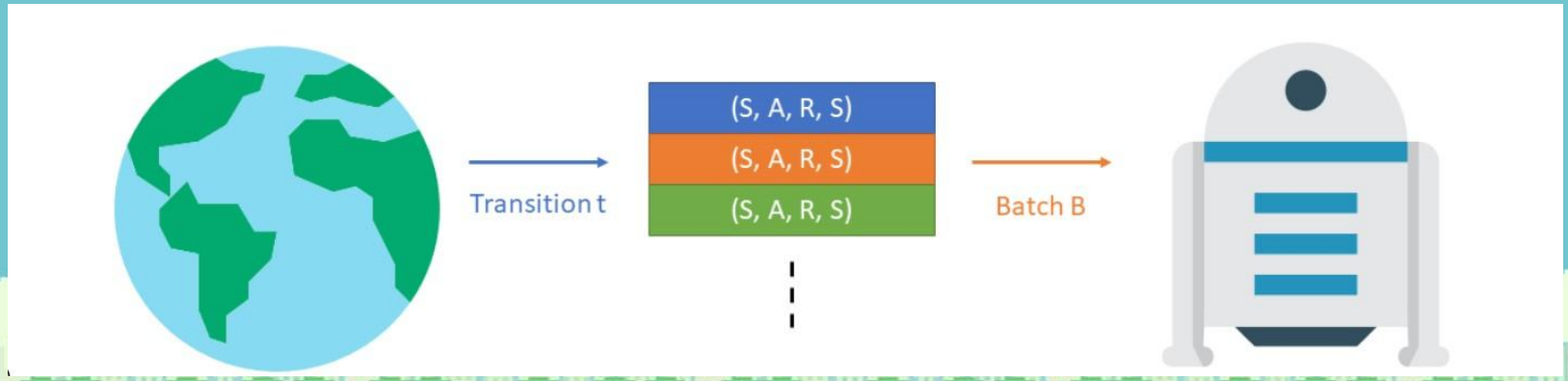
Architecture Deep Q-Network



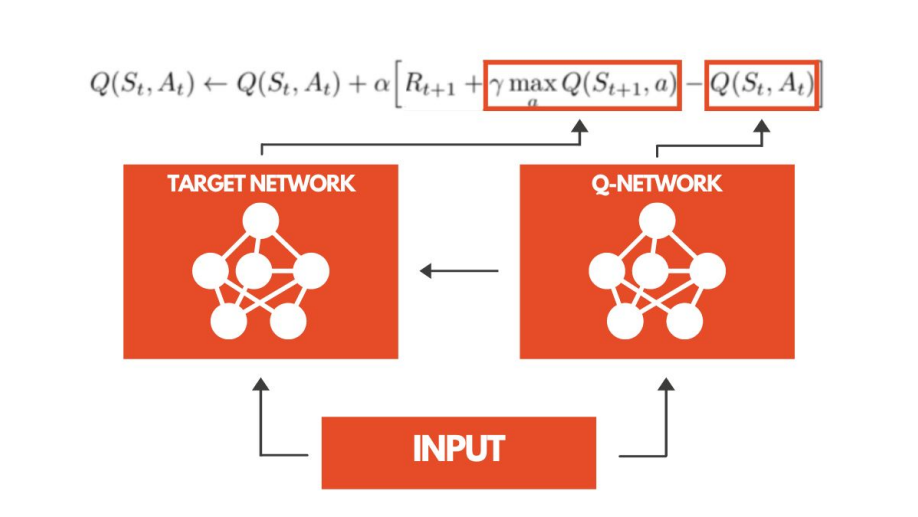
- Utilise un **réseau de neurones** pour estimer les q values

Experience replay

- Sauvegarde les transitions (State, Action, Reward, State+1) dans un buffer
- Elles sont ensuite sélectionnées aléatoirement lors de l'entraînement

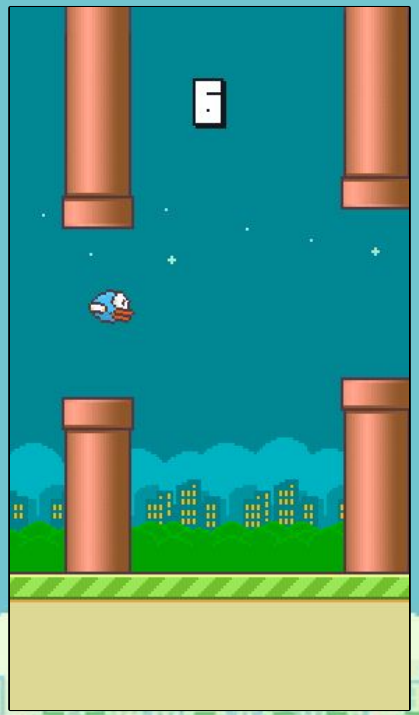


Target Network



- Le **target network**, un deuxième réseau utilisé pour calculer les valeurs cibles
- Régulièrement mis à jour avec les poids du réseau principal
- Permet une plus stabilité lors de l'entraînement.

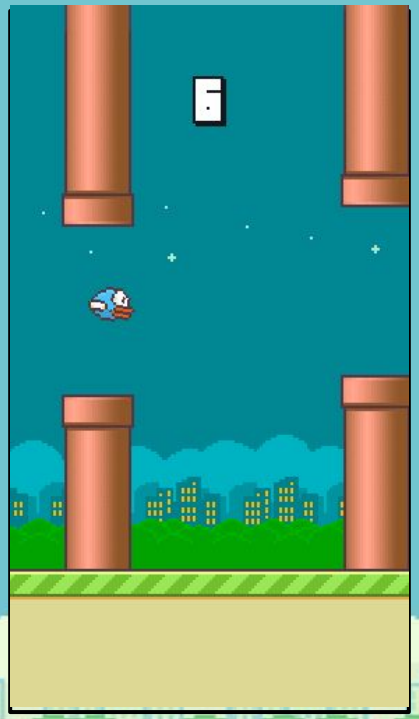
Processus d’entraînement



- **Initialisation** (Expérience replay, réseau principal et target, …)
- ε-greedy policy pour **exploration** vs **exploitation**
- Interaction avec **l’environnement** (sauter, faire avancer les tuyaux, …)
- Ajout des **transitions** pour le replay buffer
- Entraînement sur un mini-batch des **expériences**, calcul des **valeurs cibles** par le target network et **mise à jour** du réseau principal
- **Mise à jour** périodique du target network avec les poids du réseau principal

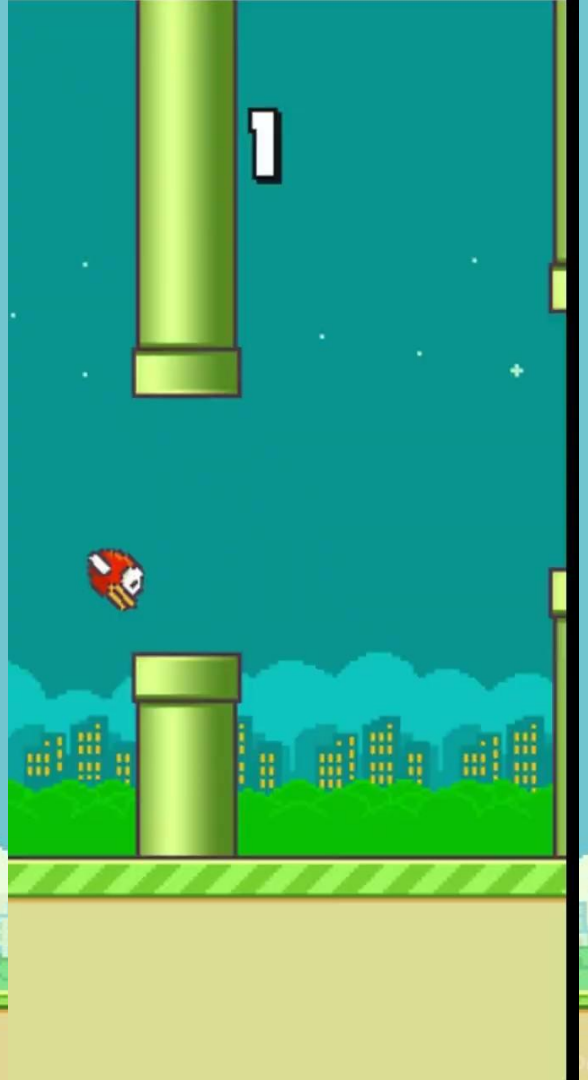
Entraînement du modèle

Difficultés lors de l’entraînement

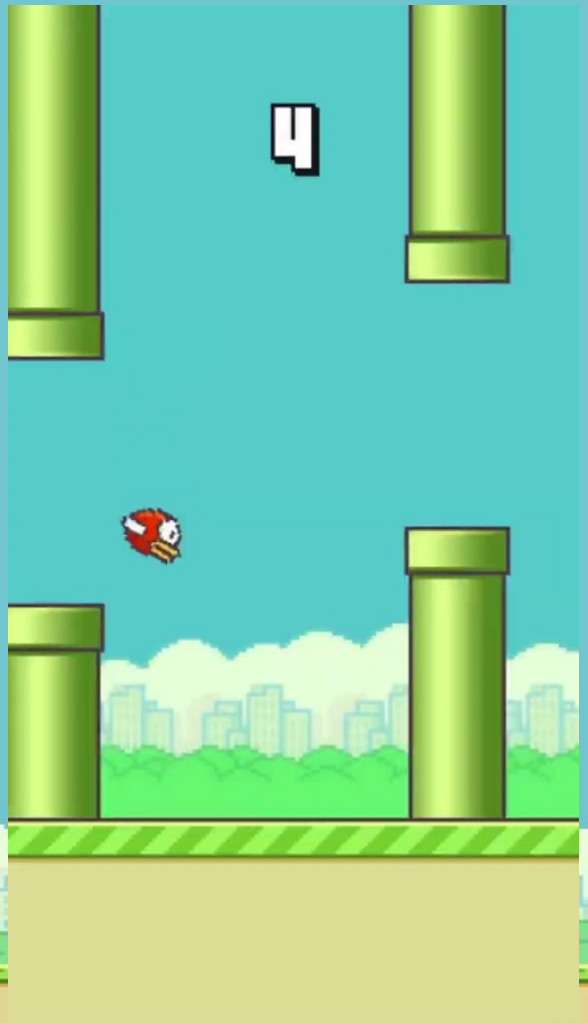


- Calcul des rewards plus adapté
- Ajout d’une couche de dropout => meilleure généralisation
- Ajout d’un scheduler pour learning rate décroissant

Résultats



Résultats



Difficulté croissante

Difficulté croissante

Merci de votre attention