UNIVERSITÉ
Abdelhamid MEHRI
CONSTANTINE 2

LISIA
Laboratoire Informatique en Sciences de données et Intelligence Artificielle

Module : IAD & AI

L'Intelligence Artificielle Distribuée

& Agent Intelligent

Master 1 : Sciences de Données et Intelligence Artificielle

2023 - 2024

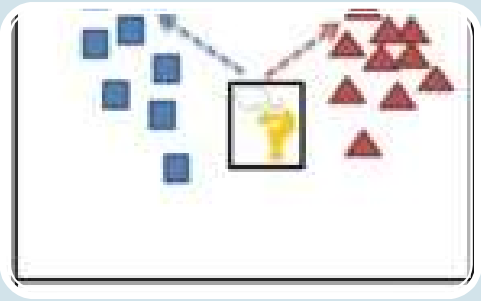
Plan de Présentation

- **Apprentissage par Renforcement :**

  Définitions , Types d'Apprentissage, Comparaison, RL pour Agent ..

- **Exploration & Exploitation.**
- **La Stratégie : ε-Greedy.**
- **Fonction de Valeur (Value Function).**
- **Algorithmes d'Agent :**
  - **L'Algorithme : Q-Learning.**
  - **L'Algorithme : Deep Q-Learning.**
  - **L'Algorithme : Policy Gradient.**

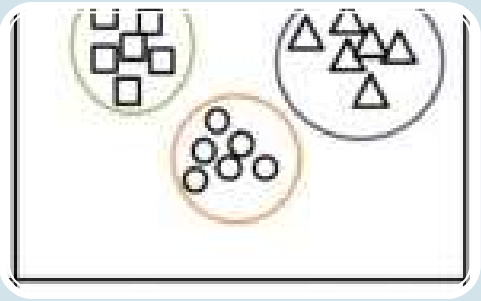
SL vs USL vs RL



Apprentissage

Supervisé

Faire des Prédictions



Apprentissage

Non Supervisé

Avoir des Corrélations

Apprentissage

Par Renforcement

Procéder selon Essais & Erreurs

Types d'Apprentissage

Avoir un Dataset

(Prédiction)

Pas de Target

(Corrélation)

Nouveau Paradigme

Apprentissage Supervisé

Avoir des Données X & Target T

Proposer des Modèles Prédictifs

But : La prédiction Y réduit l'erreur du Target T.

Apprentissage Non Supervisé

Avoir des Données X

Sans Target

Données Modifiées

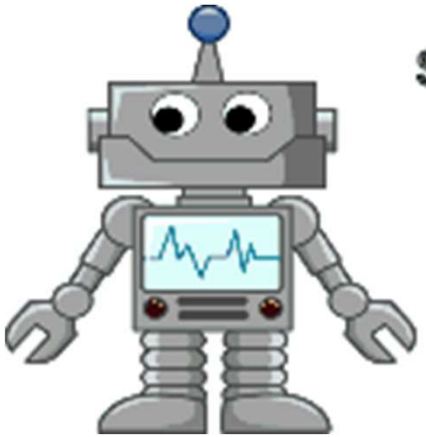
Avec du Sens

But : Créer des modèles capables de trouver des corrélations (relations) dans les données.

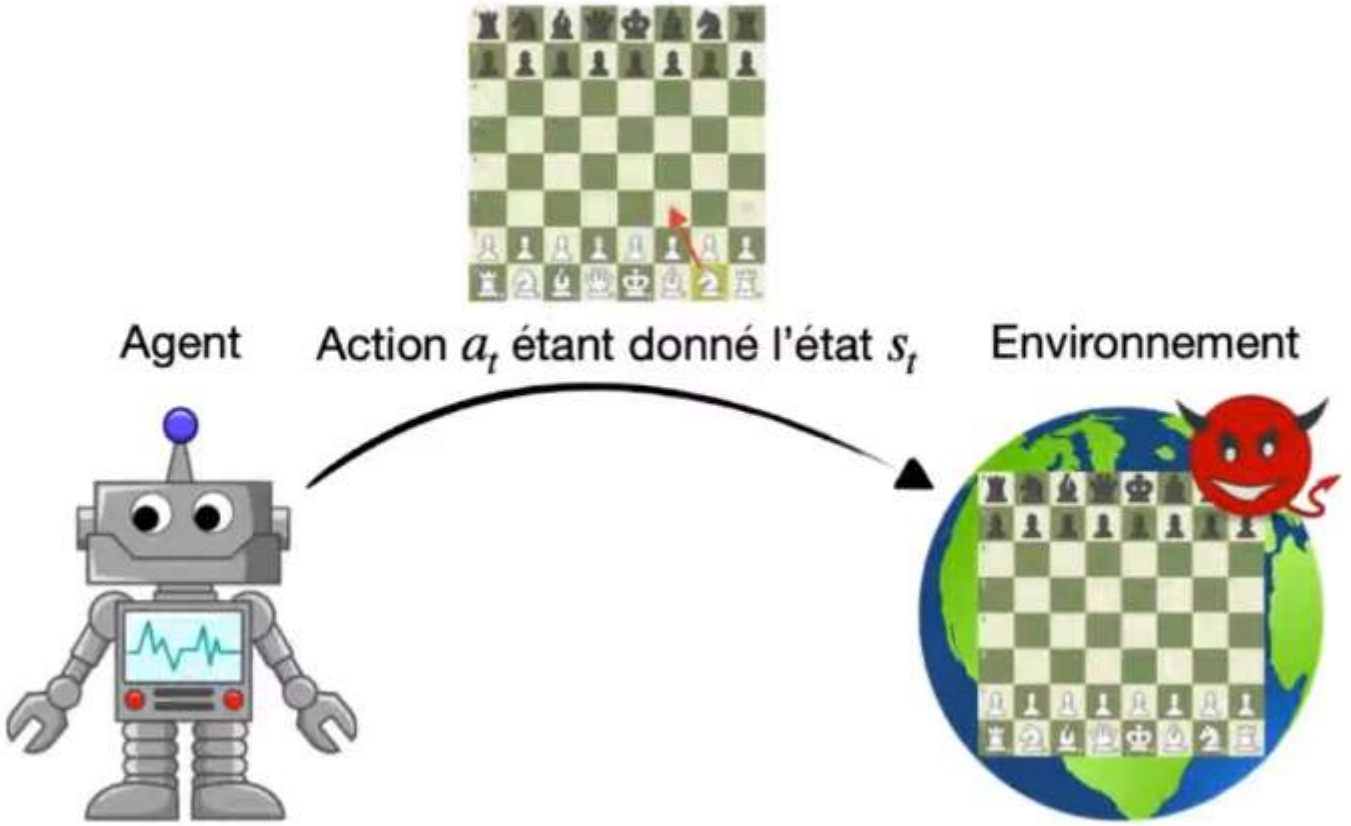
Apprentissage par Renforcement

- **RL est utilisé pour apprendre à un agent comment se comporter dans un environnement.**
- **Les données d'entrainement proviennent de l'environnement.**
- **L'environnement peut être réel ou virtuel.**
- **L'entrainement est très différent.**
- **Trop similaire à l'apprentissage humain.**
- **SL : minimiser le cout (fonction d'erreur).**
- **RL : maximiser le nombre de récompenses.**
- **Les récompenses sont données par l'environnement.**

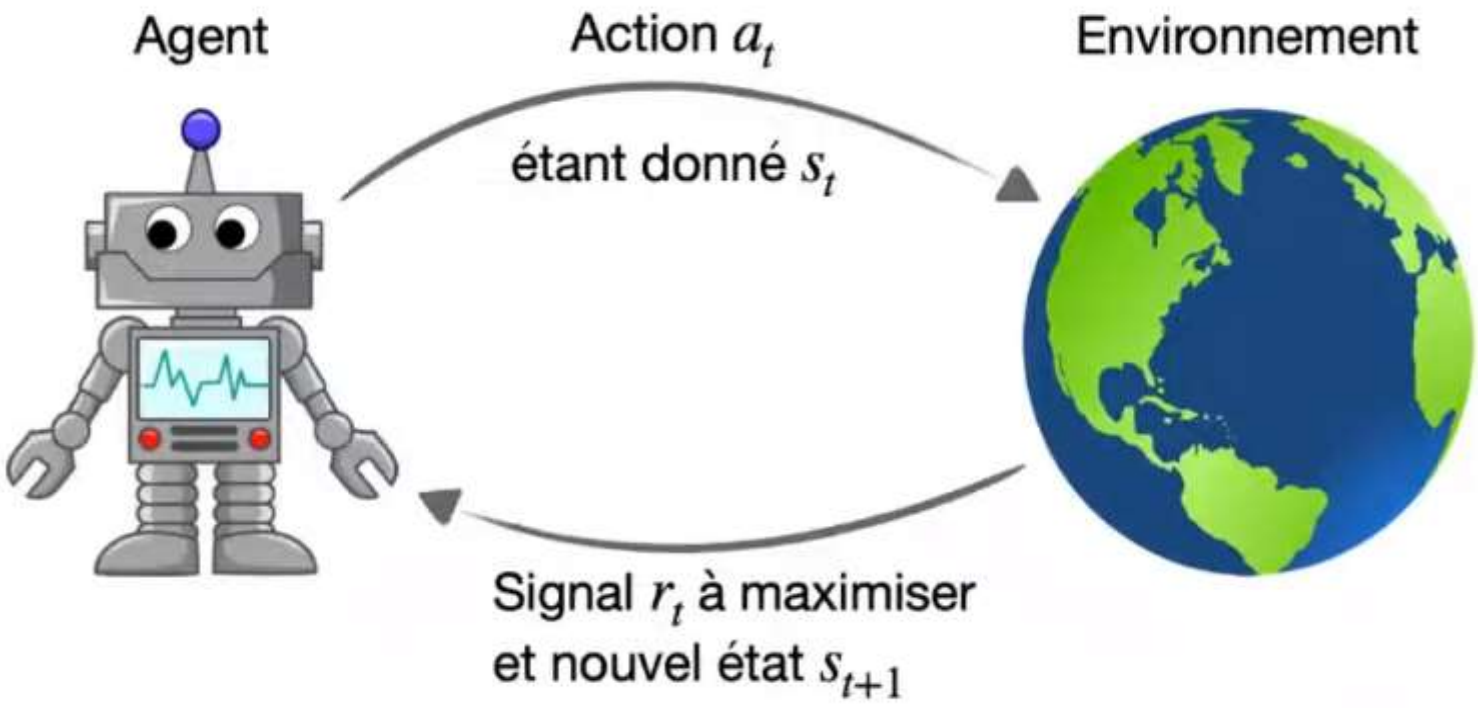
Principe du RL



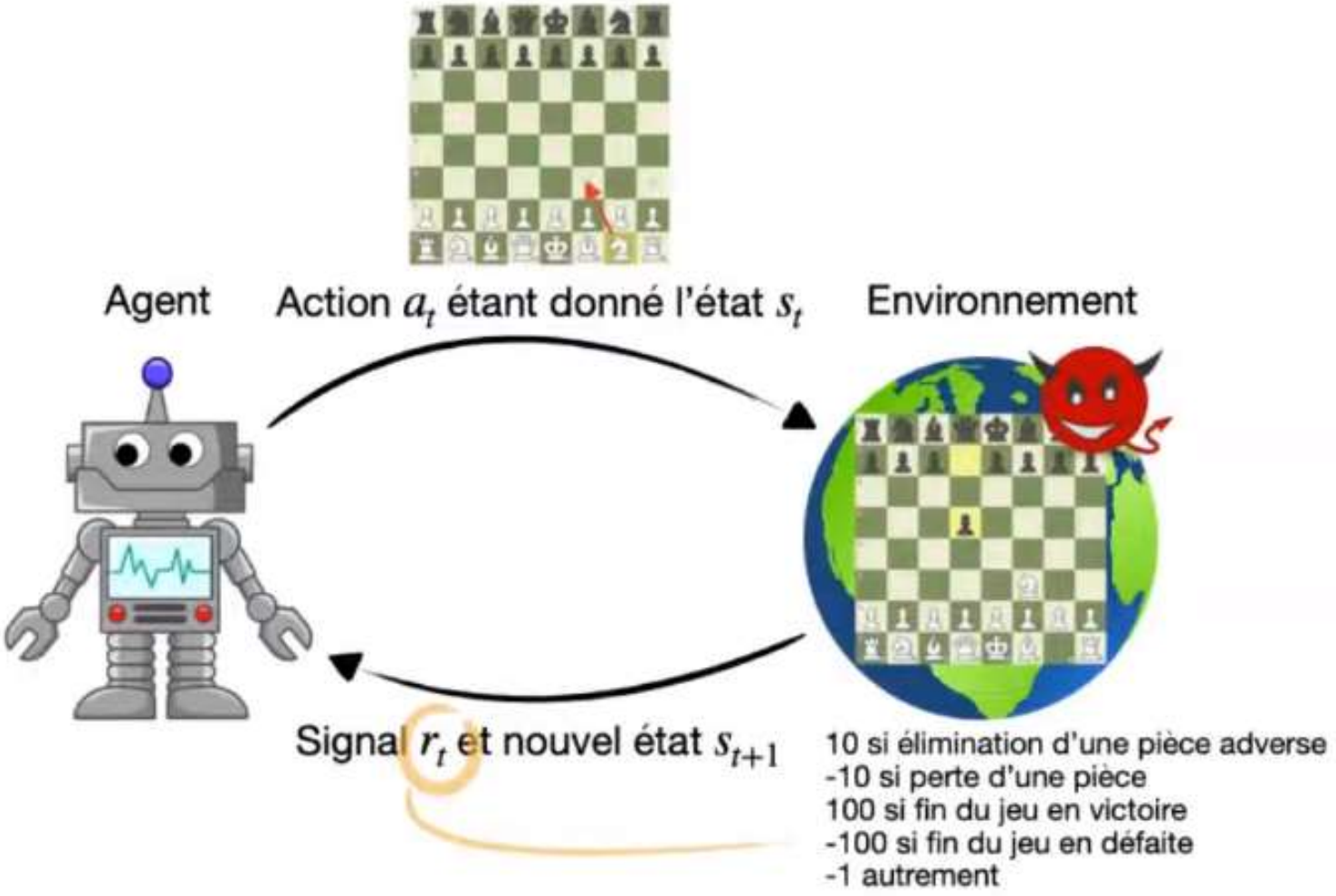
Problème de RL



Exemple du RL



Exemple du RL



Terrains d'Expérimentation

AlphaGo

(Google)

Dota 2

(OpenAI Five)

Notion d'Épisode

- Définition :

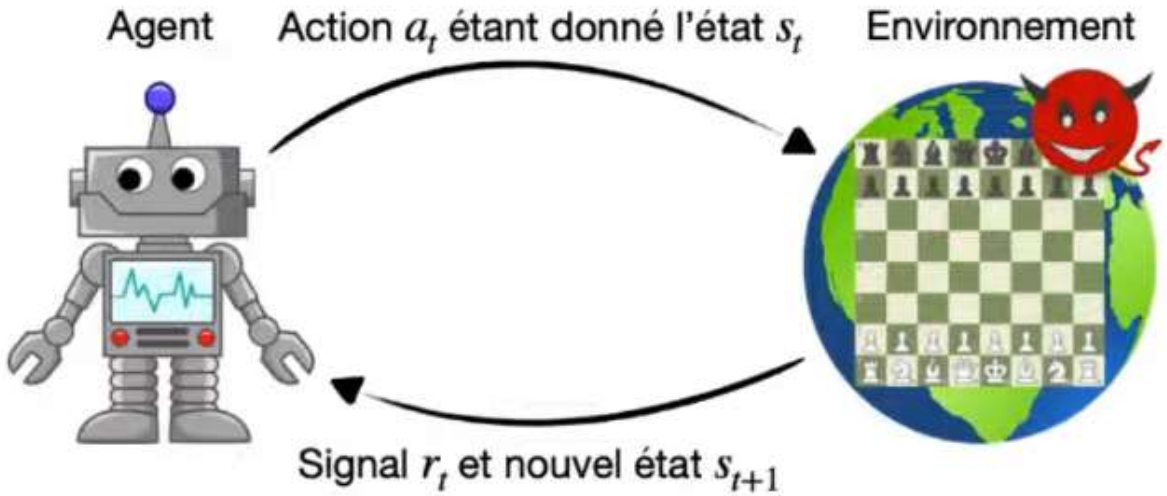
  Une séquence complète d'interactions entre un agent et un environnement. Cela commence par un agent exécutant une action, l'environnement lui répond, jusqu'à ce que l'objectif soit atteint ou l'épisode se termine autrement.

Simulation d'une Episode

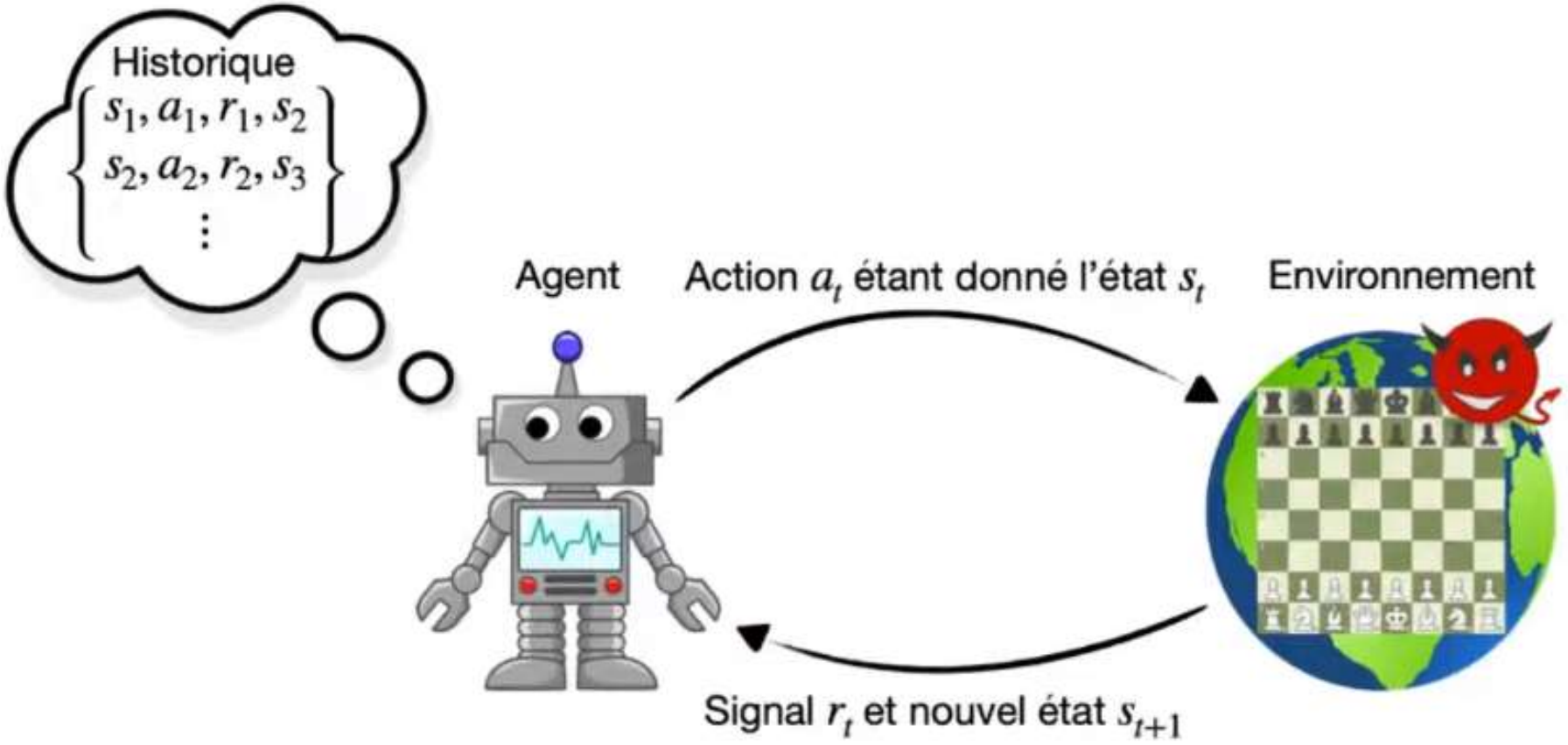
Exemple d'Épisode

1 épisode = 1 partie

- Partie 1 :  ... Défaite!
- Partie 2 :  ... Défaite!
- Partie N :  ... Victoire!



Notion d'Historique



- *A travers cet historique, l'agent va identifier les comportements optimaux.*

RL vs SL

- Exploration : consiste à la recherche de nouvelles actions ou de nouvelles zones de l'espace des actions dans le but de découvrir des informations sur l'environnement et d'améliorer la compréhension de l'agent (gain à long terme).

- Exploitation : consiste à l'utilisation des connaissances actuelles de l'agent pour maximiser les récompenses immédiates, mais cela comporte le risque de manquer de nouvelles opportunités ou de rester coincé dans des stratégies (gain à court terme).

Exploration & Exploitation

- Données d'entrainement :
  - Supervisé : Crées à l'avance et couvre bien l'espace.
  - Renforcement : L'agent construit lui-même son historique.

- Étiquettes (Classes) :
  - Supervisé : On connait la bonne réponse à l'avance.
  - Renforcement : L'agent ne connait pas l'action optimale, tant qu'il ne l'a pas essayé.

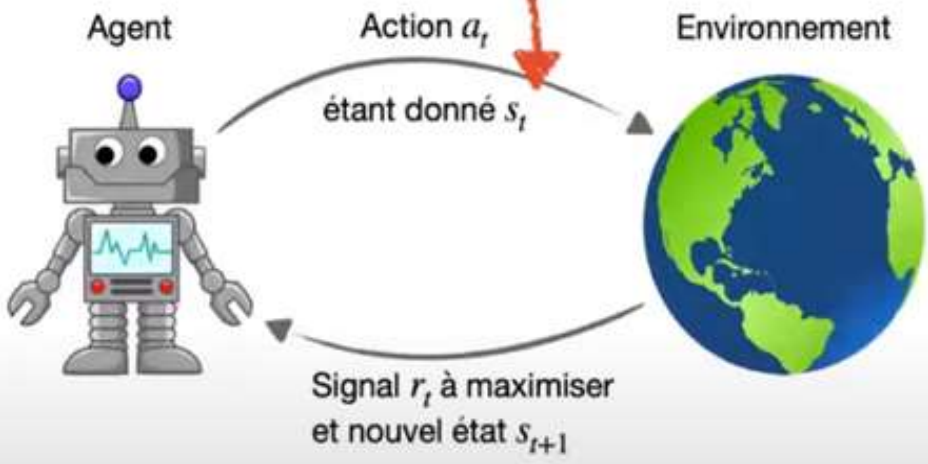
Formulation d'un Problème RL

Les Problèmes RL sont souvent formulés :

Processus de Décision de Marcov.

Processus de décision Markov:

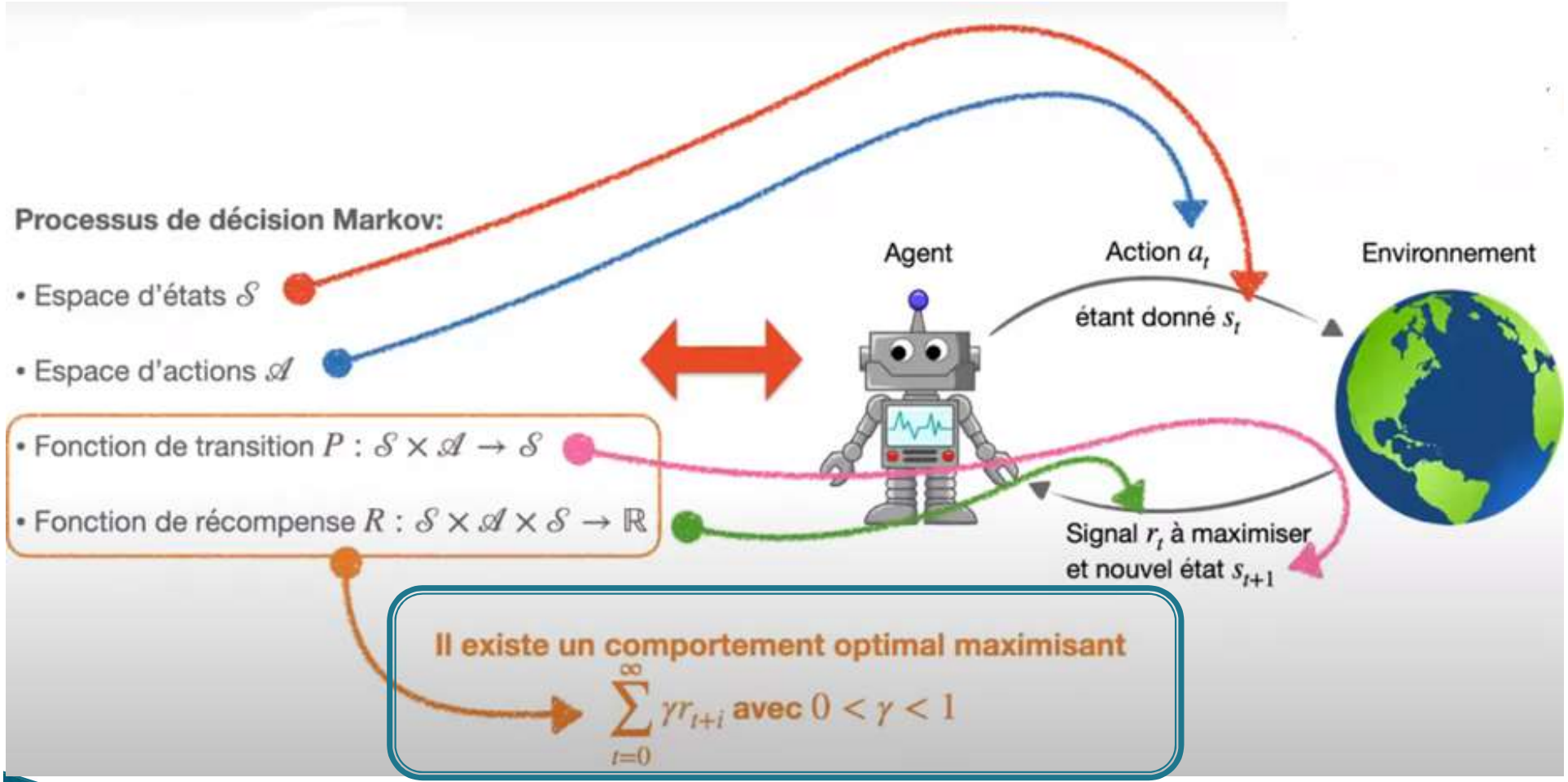
- Espace d'états $\mathcal{S}$
- Espace d'actions $\mathcal{A}$
- Fonction de transition $P : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$
- Fonction de récompense $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$



But : Caractériser le gain.

Après chaque épisode.

Formulation d'un Problème RL



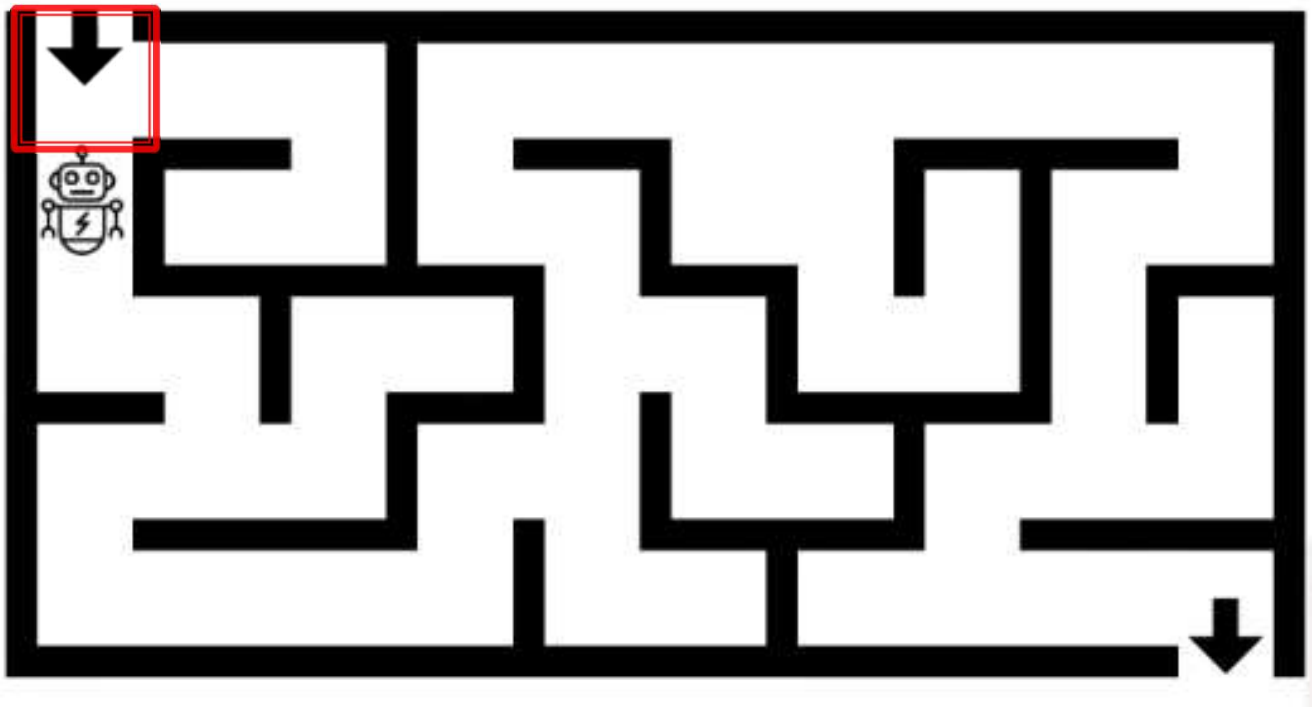
En apprenant, l'agent essaye d'avoir un comportement :

Maximiser la somme des Gains Actualisés.

Exemple

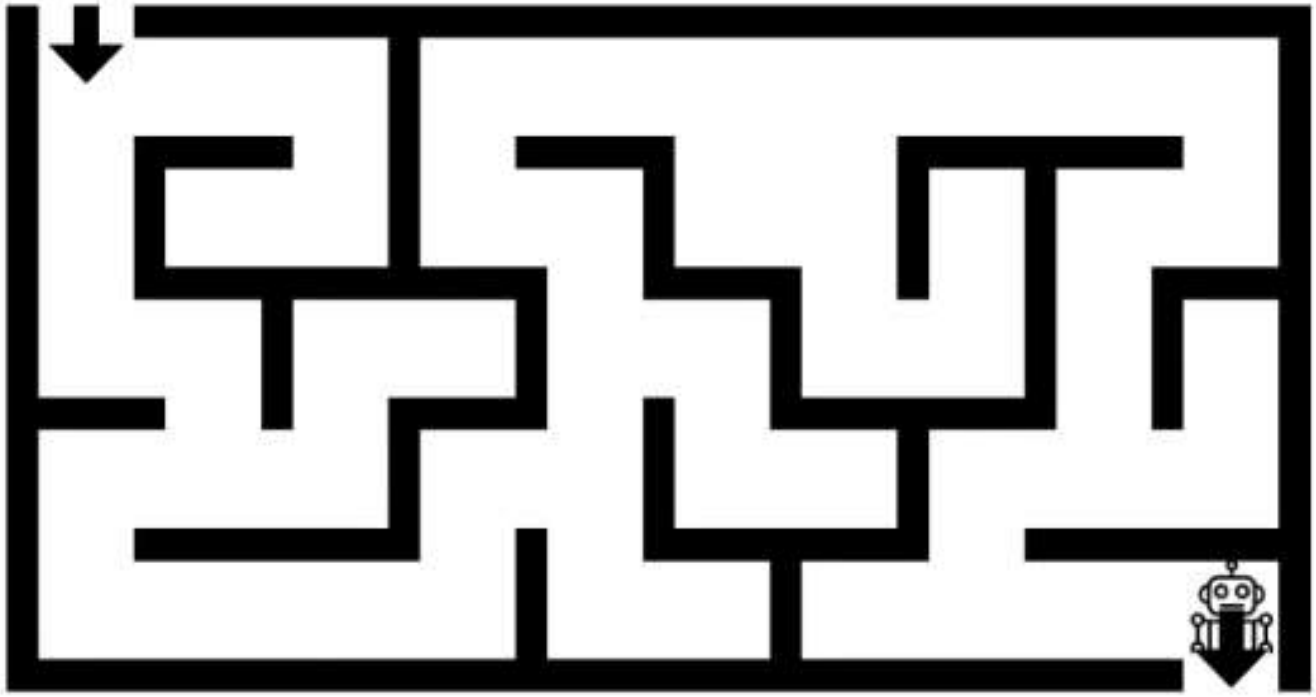
X : 0,1,2, … ,9
Y : 0,1, … , 4

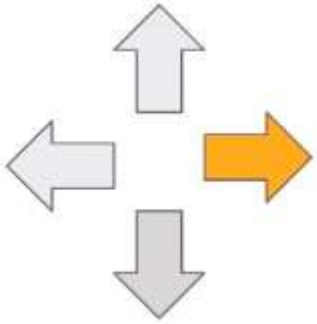
(x, y) => (0, 1)



Exemple

(x, y) => (9, 4)





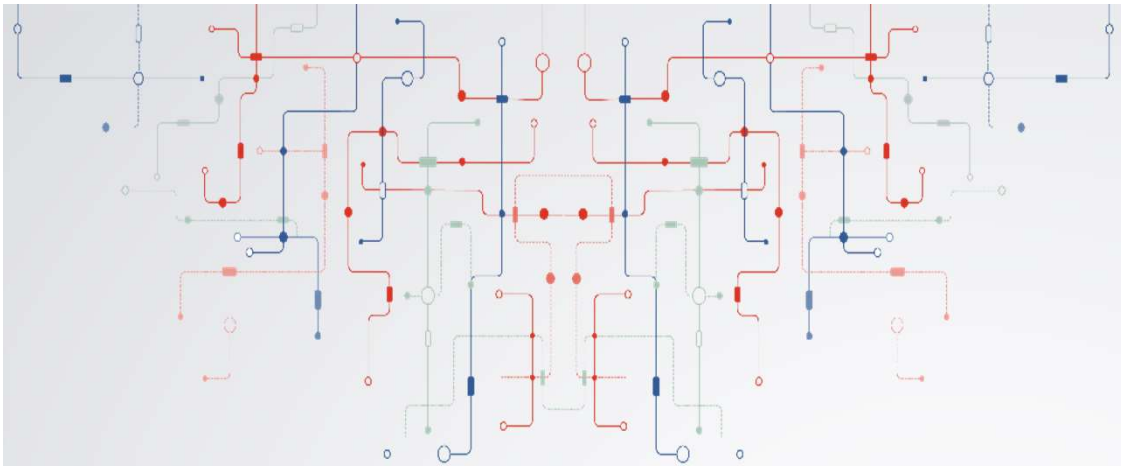
r: 1

Différentes Stratégies en RL

- **Exploration vs Exploitation :** Trouver un équilibre pour maximiser les récompenses actuelles.
- **Politique :** Détermine comment l'agent choisit ses actions.
- **Fonction de Valeur :** l'agent évalue la qualité d'une action ou d'un état.
- **Méthodes Monte Carlo :** l'agent estime les valeurs en observant des séquences complètes d'épisodes.
- **Méthodes Temporelles :** l'agent met à jour les estimations à chaque étape.

Différents Algorithmes en RL

- Q-learning : Algorithme basé sur la programmations dynamique pour l'apprentissage hors ligne .
- SARSA : Similaire, mais en mettant à jour les valeurs Q en fonction des actions réellement prises.
- Deep Q Network : Extension de Q-learning, en utilisant des réseaux de neurones profonds.
- Policy Gradient Methods : Famille d'algorithmes optimisant directement la politique de l'agent .
- SAC : Algorithme permettant l'apprentissage en continu, tout en maintenant une exploration efficace.



Merci pour votre attention ..