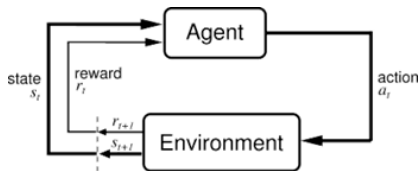


# Possibilités offertes par le machine learning

Apprentissage par renforcement

---

# Apprentissage par Renforcement



où :

$S_t$  est l'état de l'environnement,

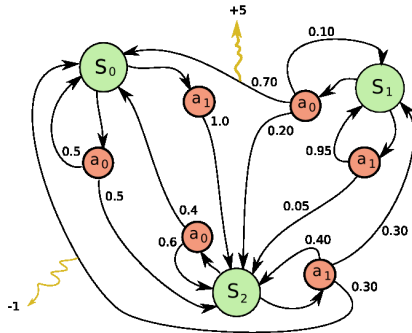
$A_t$  l'action effectué par l'agent et

$R_t$  la récompense de l'environnement à l'agent (conséquence de  $A_t$ )

# Apprentissage par Renforcement

Markov Decision Process :

L'effet des actions sur l'environnement est modélisé par des probabilités de transition



Équations de Bellman :

- Une politique  $\pi$ 
  - $\pi(s_t) = a_t$  pour une politique déterministe
  - $\pi(a|s) = \mathbb{P}[a|s]$  dans le cadre d'une politique stochastique
- Une modélisation de l'environnement  $M(s_t, a_t) = s_{t+1}, r_{t+1}$
- Une fonction d'évaluation  $v_\pi(s_t) = \mathbb{E}[r_{t+1} + r_{t+2} + \dots | a_t]$

Objectifs :

Trouver  $\pi^*(s)$  tel que

$$\forall s \in S, \forall \pi \neq \pi^*, v_{\pi^*}^* \geq v_{\pi}^*$$

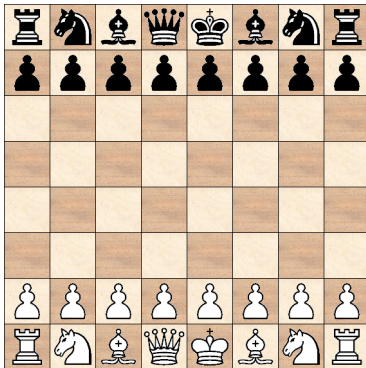
## Des contraintes techniques :

- L'environnement n'est pas forcément parfaitement modélisable
- La récompense n'est pas forcément calculable immédiatement

# Apprentissage par Renforcement

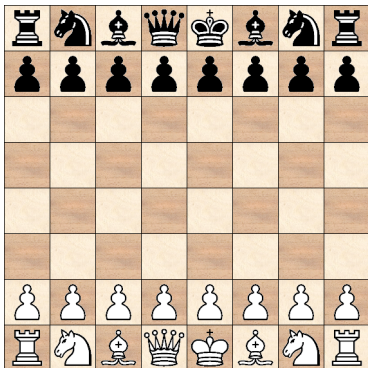


Environnement modélisable ? Récompense calculable ?





Environnement modélisable ? Récompense calculable ?



$\approx 10^{120}$  parties possibles  $\ggg 6 \times 10^{85}$   
(nombre d'atomes dans l'univers observable)

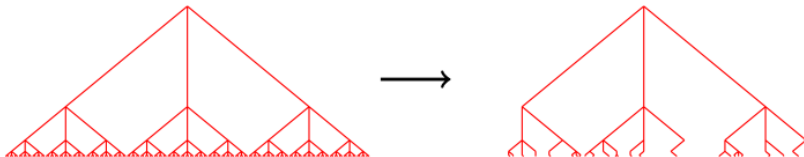


$$\approx 10^{600}$$

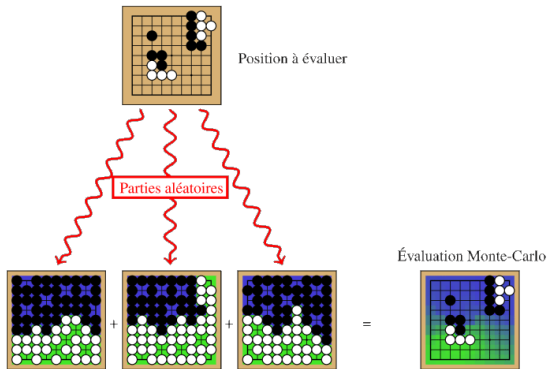
Une solution naturelle :

LE HASARD !

## Monte Carlos Tree Search



## Monte Carlos Tree Search



Crazy Stone (Rémi Coulom) et MoGo (Yizao Wang)

**octobre 2006** : MoGo est à  $\approx 10^6$  parties générées par coup (9x9)

**mars 2008** : MoGo bat Catalin Taranu (5 dan) (9x9)

**août 2008** : MoGo bat Kim Myungwan (9 dan) à 9 pierres

**septembre 2008** : Crazy Stone bat Kaori Aoba (4 dan) à 8 pierres

**décembre 2008** : Crazy Stone bat Kaori Aoba (4 dan) à 7 pierres

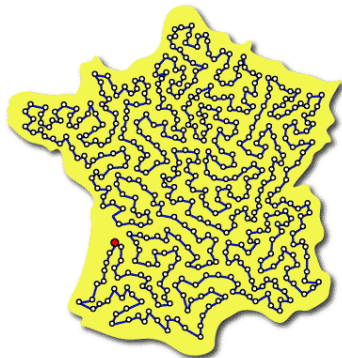
**février 2009** : MoGo bat Li-Chen Chien (1 dan) à 6 pierres

**mai 2014** : Crazy Stone bat Norimoto Yoda (9 dan) avec 4 pierres  
( $\approx 10^6$  parties générées pour chaque coup)

Progrès de + en + lents et difficiles

## Des contraintes techniques :

- L'environnement n'est pas forcément parfaitement modélisable
- La récompense n'est pas forcément calculable immédiatement
- **Plannification**





Inverse Reinforcement Learning (Andrew Ng & Peter Abbeel 2000)

- la fonction de récompense est inconnue
- Accès à des séquences d'action d'expert
- $\Rightarrow$  Apprentissage de la fonction de récompense dans une modélisation de l'environnement

Hélicoptère de modélisme