



**IA** Intelligence  
**A**rtificielle

---

Stratégie d'exploration  
non informée

---

Réalisé par

**ARNANASH**ind

**BENLAHCENK**hadija

## ➤ Les types des agents :

En intelligence artificielle, un agent est une entité autonome qui perçoit son environnement et réagit sur cet environnement afin d'atteindre un objectif et résoudre un problème.

On a deux types d'agents :

- Agent réflexe :

Le choix de l'action à effectuer se fait en se basant sur l'état actuel des perceptions avec mémorisation. Cet agent ne considère pas les conséquences de ses actions, car il n'exploite pas toutes les possibilités.

- Agent de planification :

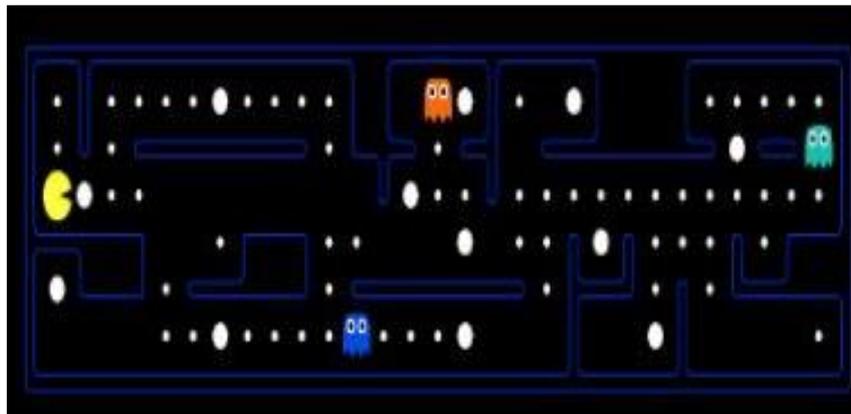
Le choix de l'action se fait selon une séquence d'actions qui permet d'atteindre l'objectif. Il considère l'environnement dans l'état actuel et l'état futur. Donc il utilise une représentation atomique de l'environnement qui permet de connaître l'état après le choix de l'action.

## ➤ Le problème d'exploration :

Un problème d'exploration est composé de :

- Un espace d'état :

Il doit inclure tous les détails de l'environnement pour que l'agent puisse connaître la dynamique de l'environnement.



- Fonction de succession :

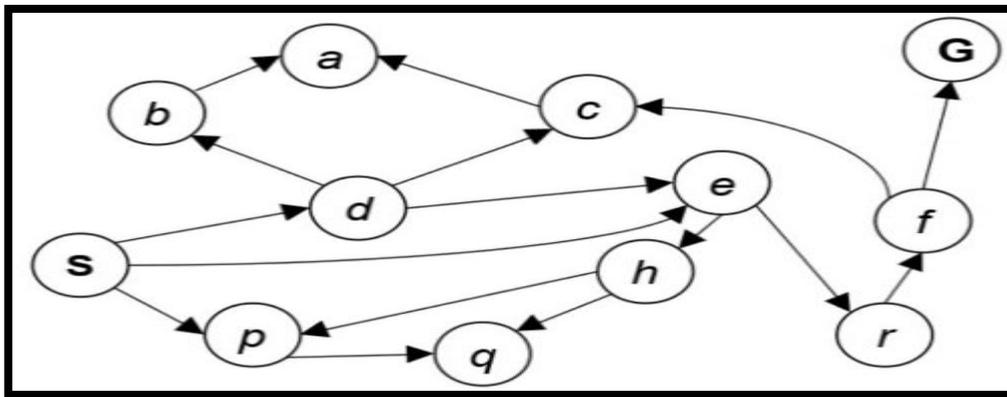
Elle définit une action à effectuer (aller à droite, aller à gauche ...) et le coût associé à cette action.

- Un état de départ et un objectif.

La solution du problème consiste à trouver la séquence d'action (plan) la plus optimisée afin d'atteindre l'objectif à partir de l'état initiale.

### ➤ Graphe d'espace d'état :

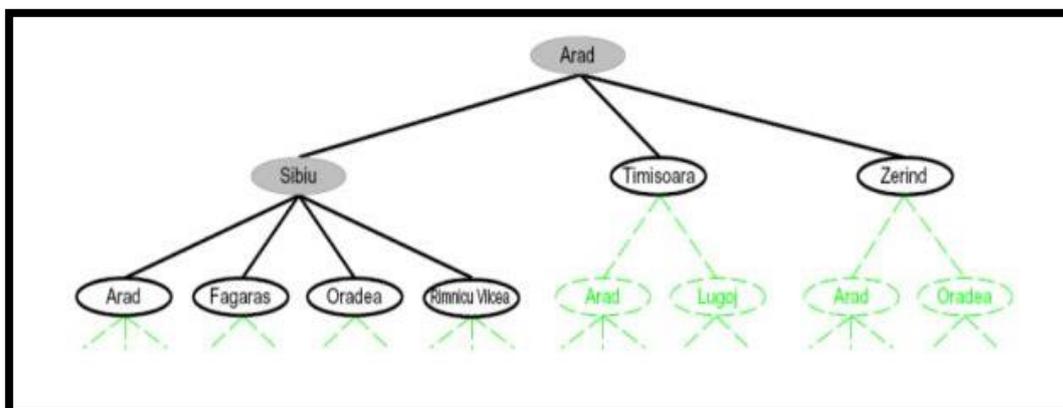
Le graphe d'espace d'état est une représentation mathématique du problème d'exploration :



- Les nœuds : ils représentent des abstractions des configurations possibles. Un nœud contient le minimum d'informations pour résoudre le problème.
- Les arrêtes : elles représentent des successions selon une action.
- Le nœud objectif est un ensemble des nœuds du graphe.

### ➤ Arbre d'exploration :

C'est une liste des configurations possibles suite à une action, où la racine est l'état initiale et les fils sont des successions. Chaque nœud représente un chemin complet pour atteindre le nœud initial. Dans ce graphe, chaque nœud est unique.



Un arbre d'exploration permet de :

- Explorer des plans potentiels.
- Maintenir une frontière des plans partiels qui doivent être considérés.
- Explorer le minimum de nœuds possibles.

## ➤ Algorithme d'exploration d'arbres :

**Function** explorer\_arbre (problème, stratégie)

Initialiser la frontière avec l'état initiale.

**Loop**

if frontière vide

then return échec ;

end if ;

Choisir une feuille de la frontière selon la stratégie ;

if feuille contient Objectif ;

then return Success ;

end if ;

Développer le nœud en ajoutant ces fils à la frontière ;

end loop

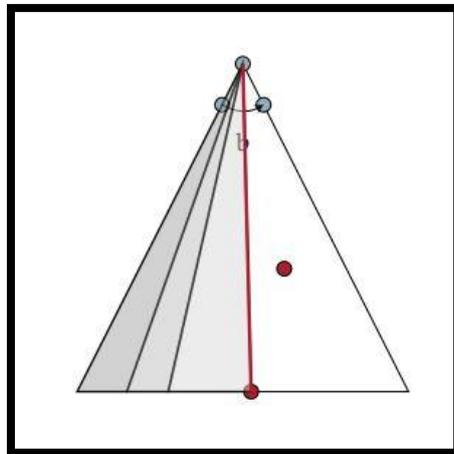
**EndProcedure**

## ➤ Les stratégies d'exploration des arbres :

Pour explorer les arbres, on a recours à trois stratégies :

- Recherche en profondeur :

Cette stratégie consiste à développer d'abord le nœud le plus profond de la frontière.

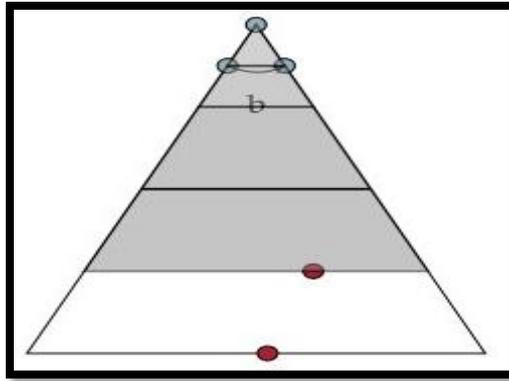


Cette stratégie assure de trouver l'objectif s'il existe, et le trouve avec un coût minimal. Elle n'est pas optimale car elle cherche à gauche.

La frontière est une pile.

- Recherche en largeur :

La stratégie de recherche en largeur consiste à développer tous les nœuds d'une superficie avant de passer à la superficie suivante.

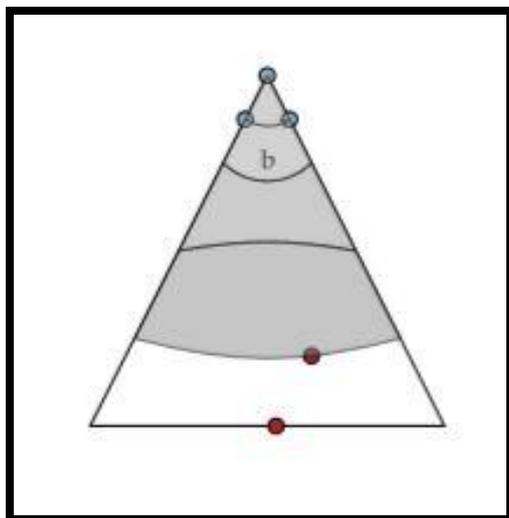


Elle permet de trouver le chemin optimal lorsque le coût de toutes les transitions est égal à 1.

La frontière est une file.

- **Recherche uniforme en coût :**

Cette méthode consiste à développer les nœuds les moins coûteux.



Cette stratégie est optimale car elle trouve le chemin ayant un coût minimal.

Dans ce cas la frontière est une file d'attente.