

EFREI 2011/2012 L3

Aide à la Décision Recherche dans un espace d'états - Généralités

Support de cours

Intelligence Artificielle
Artificial Intelligence

*Recherche
dans un espace d'états*

*Search
in a states space*

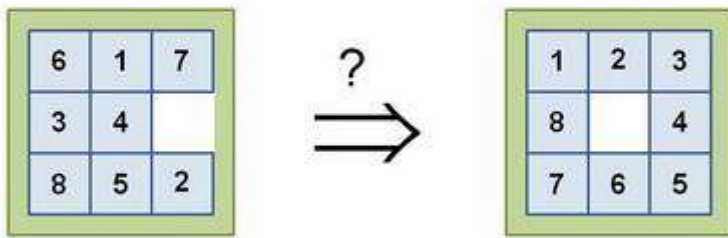
© Hervé BARBOT, 2005-2011 – www.proactitude.com

- Introduction
- Description formelle d'un problème de recherche
- Exploration d'arbres
- Recherches aveugles
 - Principes généraux
 - Illustration
 - Recherche en profondeur d'abord
 - Recherche en profondeur limitée
 - Approfondissement itératif
 - Recherche en largeur d'abord
 - Recherche en coût uniforme
- Recherches heuristiques
 - Principes généraux
 - Recherche gloutonne / recherche gourmande
 - A*
- Propriétés des méthodes de recherche – Comparaison
- Conclusion & ouverture

(C) Hervé Barbot, 2005-2011

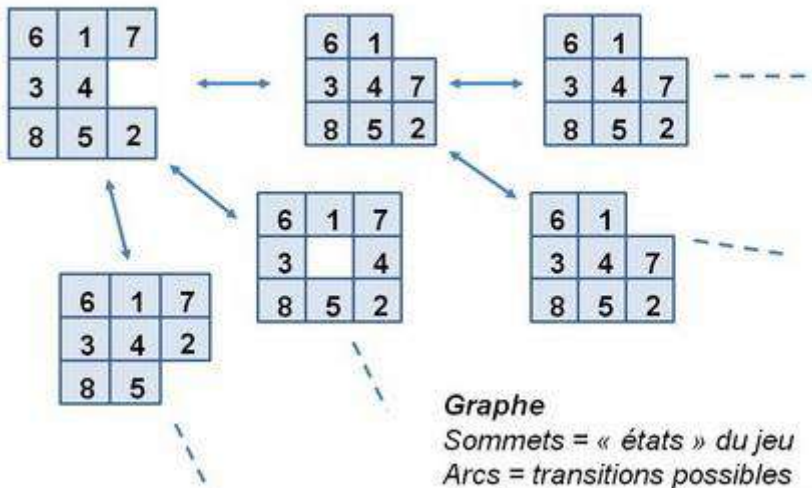
2

▪ Taquin (« 8-puzzle »)



(C) Hervé Barbot, 2005-2011

3



Graphe
 Sommets = « états » du jeu
 Arcs = transitions possibles
 Recherche de (meilleur) chemin

(C) Hervé Barbot, 2005-2011

4

▪ Taquin – Exploration de l'espace d'états

- | | Ordre de grandeur |
|------------------------------|------------------------------|
| • 8-puzzle | |
| • $9!/2 = 181\ 440$ états | <i>0,18 sec</i> |
| • 15-puzzle | |
| • $.65 \times 10^{12}$ états | <i>6 jours</i> |
| • 24-puzzle | |
| • $.5 \times 10^{25}$ états | <i>12 milliards d'années</i> |

*si analyse de
 10 millions d'états / seconde*

(C) Hervé Barbot, 2005-2011

5

Hypothèses de discussion

- **Objectif :** partant d'une situation donnée, trouver une suite d'actions menant à un « but »
 - « But » = situation finale, situation acceptable
 - **Connu :**
 - explicitement
la situation peut être décrite précisément
 - par une notion d' « acceptabilité »
une fonction est applicable à chaque situation et détermine son « acceptabilité » ou non

(C) Hervé Barbot, 2005-2011

6

Hypothèses de discussion

- **Conditions de recherche :**
 - Nombre de situations, d'étapes très grands
 - Représentation exhaustive irréaliste
 - Temps
 - Espace
 - Ensemble des situations possibles éventuellement inconnu
 - 0, 1, plusieurs, une infinité ...?

(C) Hervé Barbot, 2005-2011

7

- **Introduction**
- **Description formelle d'un problème de recherche**
- **Exploration d'arbres**
- **Recherches aveugles**
 - Principes généraux
 - Illustration
 - Recherche en profondeur d'abord
 - Recherche en profondeur limitée
 - Approfondissement itératif
 - Recherche en largeur d'abord
 - Recherche en coût uniforme
- **Recherches heuristiques**
 - Principes généraux
 - Recherche gloutonne / recherche gourmande
 - A*
- **Propriétés des méthodes de recherche – Comparaison**
- **Conclusion & ouverture**

(C) Hervé Barbot, 2005-2011

8

Recherche dans un espace d'états
Search in a states space

Formalisation

Formal description

(C) Hervé Barbot, 2005-2011

9

Algorithme de recherche

- Basé sur un mécanisme général reposant sur :
 - Un espace d'états
 - L'exploration d'alternatives
 - La recherche de la séquence d'états, donc de la séquences de transitions d'un état à un autre, menant à « la / une » solution

Dès qu'une solution est trouvée, on s'arrête.

(C) Hervé Barbot, 2005-2011

10

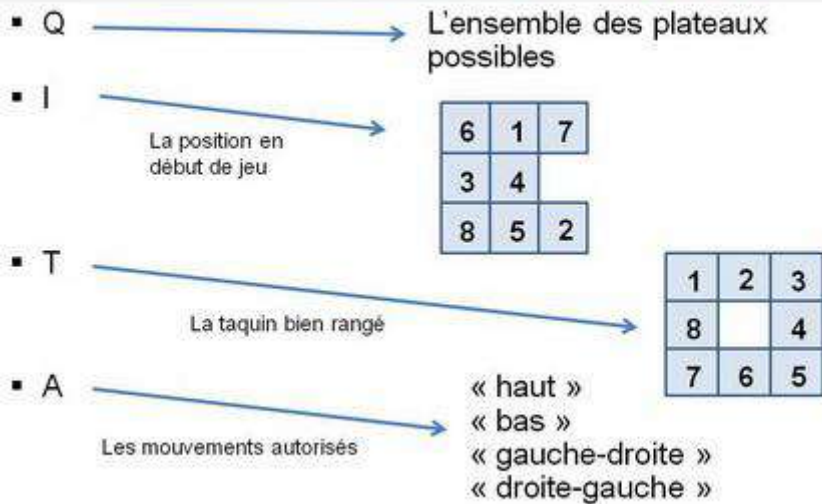
Algorithme de recherche

- Espace d'états
 - Représentation de l'ensemble des situations, des étapes, correspondant à la résolution du problème à traiter
 - Peut être infini
 - *Est découvert au fur et à mesure de la résolution du problème*
- Exploration des alternatives
 - Étant donné un état « courant » : quelle action effectuer, i.e. vers quel nouvel état se diriger ?
- Recherche de la séquence d'états menant à « la / une » solution
 - Un état acceptable au vu du problème à résoudre
- Dès qu'une solution est trouvée, on s'arrête
 - Puisqu'on ne connaît pas l'espace d'états, et on ne sait jamais si on trouverait « mieux » en continuant

(C) Hervé Barbot, 2005-2011

11

« Automate » $\langle Q, I, T, A \rangle$



(C) Hervé Barbot, 2005-2011

12

Paramètres d'un problème de recherche

- $Q = \{ \text{états} \}$
Non vide – En général, supposé fini (mais pas toujours...)
- $I = \{ \text{états initiaux} \}$... ou « { état initial } »
 $I \subseteq Q$
- $T = \{ \text{états « solution »} \}$
 $T \subseteq Q$
Connus explicitement ou déterminés implicitement par une fonction test-solution
- $A = \{ \text{actions} \}$
- $S : Q \times A \rightarrow Q$
Passage d'un état à un autre au moyen d'une certaine action
Notation : $S(q) = \{ q_i \in Q \mid \exists a_i \in A, S(q, a_i) = q_i \}$

(C) Hervé Barbot, 2005-2011

13

- Une solution à un problème =
 - une suite d'actions a_0, a_1, \dots, a_{n-1}
 - ou une suite d'états q_0, q_1, \dots, q_n
 Si l'on suppose que la connaissance de 2 états est suffisant pour déterminer l'action qui fait passer le système de l'un à l'autre...

telle que

$$\forall i, 0 \leq i \leq n, q_i \in Q$$

$$\forall i, 0 \leq i < n, a_i \in A, q_{i+1} = S(q_i, a_i)$$

$$q_0 \in I$$

$$q_n \in T$$

(C) Hervé Barbot, 2005-2011

14

- $S^{-1} : Q \times A \rightarrow Q$ fonction « prédécesseurs »
- $S^* : Q \times A^* \rightarrow Q$ fonction « descendants »
- On a parfois une notion de coût associé à chaque action
 - $C : Q \times A \times Q \rightarrow \mathbb{R}^+$
 - $C : Q \times Q \rightarrow \mathbb{R}^+$
 Coût d'une transition défini en relation avec la fonction S
 - $C^* : Q \times A^* \times Q \rightarrow \mathbb{R}^+$
 - $C^* : Q^* \rightarrow \mathbb{R}^+$
 Coût d'un chemin défini par :

$$\forall i, 0 \leq i < n, q_i \in Q, q_{i+1} = S(q_i, a_i)$$

$$C^*(q_0, q_1, \dots, q_n) = \sum_{0 \leq i < n} C(q_i, q_{i+1})$$

A^* = ensemble / suite d'actions

(C) Hervé Barbot, 2005-2011

15

Enoncé d'un problème

Etant donné une situation donnée q_0

- L'état actuel
- Le point de départ de la recherche

Comment faire pour aboutir à une situation q_n acceptable ?

- Quelles actions effectuer pour atteindre un état solution ?
- Eventuellement, comment le faire à « moindre coût » ?
(minimisation de la fonction C^)*

(C) Hervé Barbot, 2005-2011

16

Stratégies de recherche

- **Stratégies aveugles**
 - Largeur d'abord
 - Profondeur d'abord
 - Profondeur limitée
 - Approfondissement itératif
 - Coût uniforme

Aucune information particulière n'est contenue dans un nœud donné
- **Stratégies heuristiques**
 - « best-first »
 - Recherche gourmande
 - A^*

Des informations existent pour déterminer si un nœud est « plus intéressant » qu'un autre

(C) Hervé Barbot, 2005-2011

17

Critères d'évaluation des algorithmes

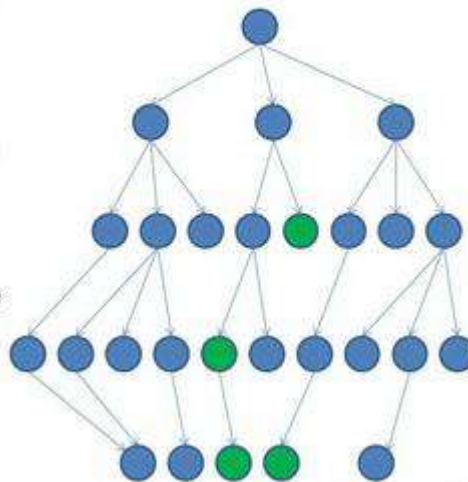
- **Complétude**
 - La méthode garantit-elle de trouver une solution si elle existe ?
- **Optimalité**
 - Si plusieurs solutions existent, la méthode garantit-elle de trouver la « meilleure » ?
- **Complexité en temps**
 - Combien de nœuds (d'états) faut-il produire / analyser pour trouver la solution ? *(ordre de grandeur)*
- **Complexité en espace**
 - Combien de nœuds faut-il conserver en mémoire pour trouver une solution ? *(ordre de grandeur)*

(C) Hervé Barbot, 2005-2011

18

▪ Valeurs numériques de référence :

- b = nombre (maximum) de successeurs pour un état donné
- d = profondeur de la solution trouvée en utilisant une stratégie de recherche particulière
- m = profondeur maximale de l'arbre de recherche

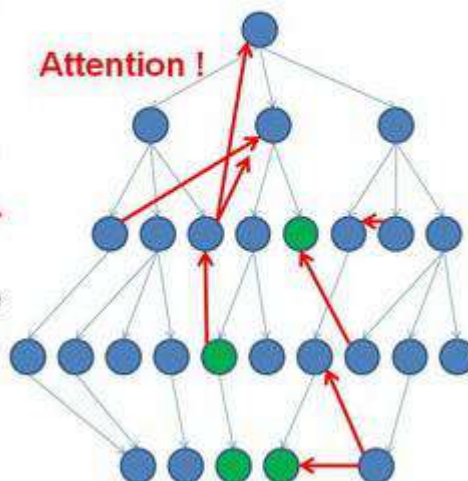


(C) Hervé Barbot, 2005-2011

19

▪ Valeurs numériques de référence :

- b = nombre (maximum) de successeurs pour un état donné
 - d = profondeur de la solution trouvée en utilisant une stratégie de recherche particulière
 - m = profondeur maximale de l'arbre de recherche
- Certains valeurs dépendent de la « façon » dont on découvre un état...!



(C) Hervé Barbot, 2005-2011

20

- Introduction
- Description formelle d'un problème de recherche
- **Exploration d'arbres**
- Recherches aveugles
 - Principes généraux
 - Illustration
 - Recherche en profondeur d'abord
 - Recherche en profondeur limitée
 - Approfondissement itératif
 - Recherche en largeur d'abord
 - Recherche en coût uniforme
- Recherches heuristiques
 - Principes généraux
 - Recherche gloutonne / recherche gourmande
 - A*
- Propriétés des méthodes de recherche – Comparaison
- Conclusion & ouverture

(C) Hervé Barbot, 2005-2011

21

Exploration d'arbres

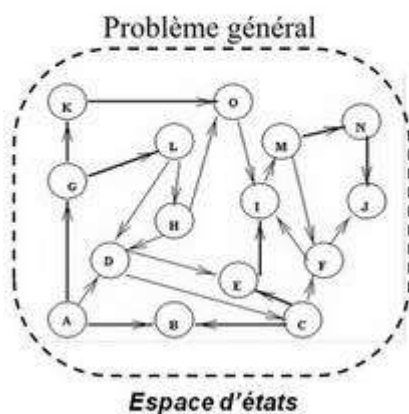
Exploring trees

(C) Hervé Barbot, 2005-2011

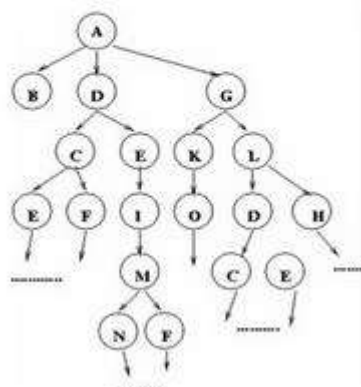
22

Représentation sous forme d'arbre

- Arbre « de dérivation » ou « de recherche »



Représentation par arbre



(C) Hervé Barbot, 2005-2011

23

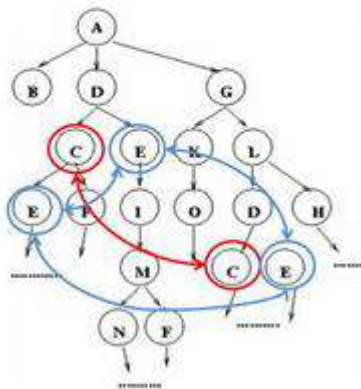
Représentation sous forme d'arbre

- Arbre « de dérivation » ou « de recherche »

Représentation par arbre

Attention :

Analyses multiples ?...



(C) Hervé Barbot, 2005-2011

24

- Remarque importante :

Pour des raisons didactiques, on suppose connaître l'espace d'états « à l'avance »,

ce qui n'est pas le cas !

(C) Hervé Barbot, 2005-2011

25

- Arbre « de dérivation » ou « de recherche »

- A chaque nœud : ses successeurs
- Certains nœuds représentent des solutions
- Choisir un successeur = se déplacer dans l'arbre
 - Selon la **stratégie** choisie

- Etat

- Configuration réelle de l'environnement, du problème à résoudre

Nœud

- Élément de la SdD qui représente un état
- Notions de parent / fils, de profondeur, de coût d'un chemin

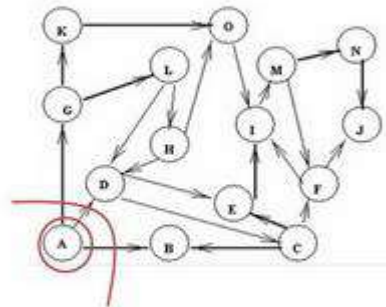
(C) Hervé Barbot, 2005-2011

26

A

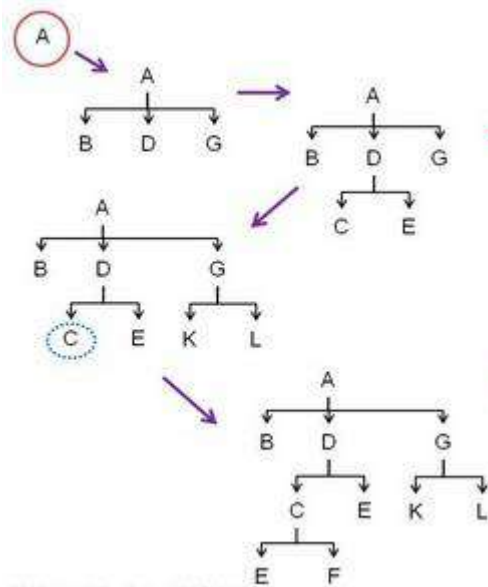
Exploration off-line / simulée

Problème général

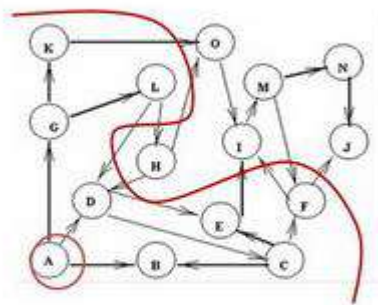


(C) Hervé Barbot, 2005-2011

27



Après quelques étapes :



La partie à gauche de la ligne représente l'espace d'états parcouru

(C) Hervé Barbot, 2005-2011

28

- Gestion d'un ensemble d'états « candidats à expansion »
 - Génération des successeurs des états déjà explorés
 - Stratégie de choix du successeur pour la progression de la recherche
 - Atteinte d'une solution

(C) Hervé Barbot, 2005-2011

29

Recherche « offline » - Algorithme

Function `tree_search (problem, strategy)`
Returns `solution` or `failure`

```
search_tree ← initial_state( problem )
```

Loop do :

```
  → ∃ candidate_for_expansion(search_tree)
     → return failure
```

```
leaf_node ← choose_node(search_tree, strategy)
```

```
is_solution(leaf_node, problem)
  → return path from initial state to leaf_node
```

```
  → is_solution(leaf_node, problem)
     → search_tree ← expand(search_tree, leaf_node,
                           problem, strategy)
```

(C) Hervé Barbot, 2005-2011

30