TD1 – Résolution de problèmes à l'aide de graphes d'états

Nathalie Guin & Marie Lefevre

Nous avons vu en cours que pour résoudre un problème, nous pouvons le modéliser de manière à construire un graphe d'états et utiliser des algorithmes permettant de rechercher une ou plusieurs solutions dans ce graphe. Une solution est donc un chemin dans le graphe, constitué de la liste des états par lesquels passer, associée aux opérateurs permettant de passer d'un état à un autre.

Nous avons également vu qu'il existe plusieurs classes de problèmes : les problèmes de satisfaction de contraintes où nous cherchons un état but respectant les contraintes, peu importe le chemin parcouru ; et les problèmes de planification où nous cherchons les étapes à effectuer pour arriver sur un état but connu.

Dans ce premier TD, nous allons donc modéliser un problème de planification et un problème de satisfaction de contraintes de façon à pouvoir construire un graphe d'états et à parcourir ce graphe avec l'algorithme de recherche informée A*.

PARTIE 1 - MODELISATION D'UN PROBLEME DE PLANIFICATION

TAQUIN

Le taquin est une sorte de puzzle où il faut remettre les nombres dans l'ordre. Il est formé d'une grille de N × N cases où sont placées N^2-1 tuiles étiquetées par les nombres 1 à N^2-1, une des cases restant vide. Une des tuiles situées à côté de la case vide peut être déplacée vers cette case. Les déplacements se font donc verticalement ou horizontalement, mais pas en diagonale.

Le taquin est souvent utilisé pour tester les algorithmes de recherche. En augmentant la taille de la grille, les problèmes deviennent de plus en plus complexes. Les algorithmes d'aujourd'hui arrivent à résoudre les taquins 3 × 3 et 4 × 4 (qui ont des espaces d'états respectivement de taille 181 440 et d'environ 1,3 milliards), mais les instances du taquin 5 × 5 (avec un espace d'états de taille 10^{25}) restent difficiles...

Nous souhaitons résoudre un taquin de taille 3 × 3, où les tuiles sont numérotées de 1 à 8. Les états initial et final sont donnés sur les figures ci-dessous.

2	8	3
1	6	4
7		5

État initial

1	2	3
8		4
7	6	5

État final

Q1 : Modélisez ce problème selon le canevas présenté en cours :

Pour rappel, le canevas de modélisation d'un problème est :

{ États / État initial / Opérateurs (action, condition d'application et fonction de successeur) / Test de but / Fonction de coût (simple ou avec heuristique) }

Q2 : Dessinez les 3 premiers niveaux du graphe d'états.

Q3 : Proposez une heuristique permettant de guider la recherche.

PARTIE 2 – RECHERCHE HEURISTIQUE DANS UN GRAPHE D'ETATS

Nous avons vu en cours qu'il existe plusieurs algorithmes pour rechercher une solution dans un graphe d'états. Dans ce TD, nous allons travailler sur l'algorithme A* qui est un algorithme de recherche informée.

L'ALGORITHME A*

```
Algorithme A*
1.     Initialisation : OUVERTS ← u₀ ; FERMES ← Ø ; g(u₀) ← 0 ; u ← u₀
2.     Itérer tant que [OUVERTS ≠ Ø et u non terminal]
2.1       Supprimer u de OUVERTS et le mettre dans FERMES
2.2       Itérer sur les nœuds v successeurs de u
        Si [v ∉ (OUVERTS ∪ FERMES) ou g(v) > g(u) + coût(u, v)] Alors faire :
           g(v) ← g(u) + coût(u, v)
           f(v) ← g(v) + h(v)
           père(v) ← u
           Ranger v dans OUVERTS, dans l'ordre f croissant, puis g décroissant
     Fin Itération 2.2
2.3       Si OUVERTS ≠ Ø Alors u ← tête(OUVERTS)
          Fin Itération 2
3.     Si OUVERTS = Ø   Alors le problème n'admet pas de solution
                        Sinon fournir la solution chemin(u)
```

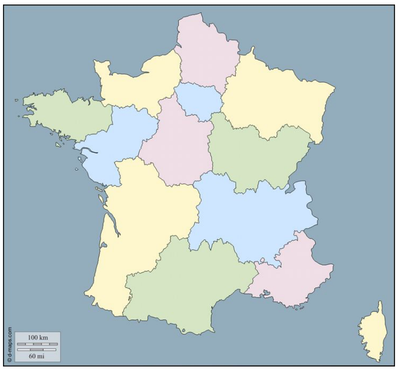
Q4 : Appliquez cet algorithme au problème du taquin tel que vous l'avez modélisé dans la partie 1, avec au moins une heuristique. Il s'agit de faire « tourner à la main » l'algorithme en traçant les différentes structures et variables utilisées.

PARTIE 3 - MODELISATION D'UN PROBLEME DE SATISFACTION DE CONTRAINTES

THEOREME DES QUATRE COULEURS

Le théorème des quatre couleurs indique qu'il est possible, en n'utilisant que quatre couleurs différentes, de colorer n'importe quelle carte découpée en régions connexes, de sorte que deux régions limitrophes, c'est-à-dire ayant une frontière (et non simplement un point) en commun, reçoivent toujours deux couleurs distinctes.

Nous souhaitons donc colorier une carte des 13 régions métropolitaines françaises en utilisant uniquement quatre couleurs.



Q5 : Modélisez ce problème selon le canevas présenté en cours.

Q6 : Dessinez les 3 premiers niveaux du graphe d'états.

Q7 : Proposez une heuristique permettant de guider la recherche.

Q8 : Appliquez l'algorithme A* à ce problème.