SOLUTION SESSION1 LUNDI 28/03/2022

EXERCICE 1 : LE SUPER-TAQUIN (9 PT)

1.1 Soit u_0 l'état initial et u_n l'état obtenu après n actions successives.
Définition de la fonction $g(u_n)$ dans A* : (1 pt)

Soit u_0, u_1, u_2, …, u_{n-1}, u_n la suite des états traversés pour aller de l'état initial u_0, à l'état courant u_n

$$g(u_n) = \sum_{i=0}^{i=n-1} k(u_i, u_{i+1})$$

ou bien (définition récursive)

$$g(u_0) = 0,$$
$$g(u_i) = g(u_{i-1}) + k(u_{i-1}, u_i)\ \forall i = 1..n$$

1.2 ***Distance de Manhattan***.

$$\boldsymbol{DM(p,u,T) = |x_p^T - x_p^u| + |y_p^T - y_p^u|} \qquad \boldsymbol{h(u) = \sum_{p=1}^{p=8} DM(p,u,T)}$$

h est-elle ***minorante*** dans le cas du super-Taquin ? (2 pt)

Soient *u* et *v*, 2 états tels que *v* est le successeur de *u* après réalisation de *A*, action simple ou super action.
On sait que chaque action modifie l'emplacement d'une **seule pièce du puzzle à la fois** ; soit *p*, la pièce qui est déplacée par l'action : les autres pièces, *p'*, ne changent pas de position ; on a : *DM*(*p'*,*u*,*T*) = *DM*(*p'*,*v*,*T*). Donc on peut écrire :

$$\boldsymbol{h(u) - h(v) = \sum_{p=1}^{p=8} DM(p,u,T) - \sum_{p=1}^{p=8} DM(p,v,T)}$$

$$\boldsymbol{= \sum_{p'=1}^{p'=8,\ p'\neq p} DM(p',u,T) - DM(p',v,T) \quad + \quad DM(p,u,T) - DM(p,v,T)}$$

$$\boldsymbol{= 0 + DM(p,u,T) - DM(p,v,T)}$$

Comme chaque action(ou super-action) est soit horizontale, soit verticale, seule une des 2 coordonnées de la pièce déplacée change de valeur ; soit ***z*** la coordonnée qui change de valeur au cours de l'action de déplacement (z=x ou bien *z*=y). On a :

$$\boldsymbol{DM(p,u,T) - DM(p,v,T) = |x_p^T - x_p^u| + |y_p^T - y_p^u| - |x_p^T - x_p^v| - |y_p^T - y_p^v| = |z_p^T - z_p^u| - |z_p^T - z_p^v|}$$

On analyse 3 cas :

- le cas où l'action *A* a comme effet de **rapprocher** la pièce déplacée *p* de sa position finale ; on a alors :
  h(*u*)–*h*(*v*) = -1 (cas d'une action simple) ou -2 (dans le cas d'une super-action).
  Le coût de l'action est : $\boldsymbol{k(u,v) = |z_p^u - z_p^v|}$**=1 ou 2** selon qu'il s'agit d'une action ou d'une super-action.

Dans les 2 cas, on vérifie donc que : *h*(*u*)–*h*(v) ≤ -1 ≤ 1 ≤ *k*(*u*,*v*)

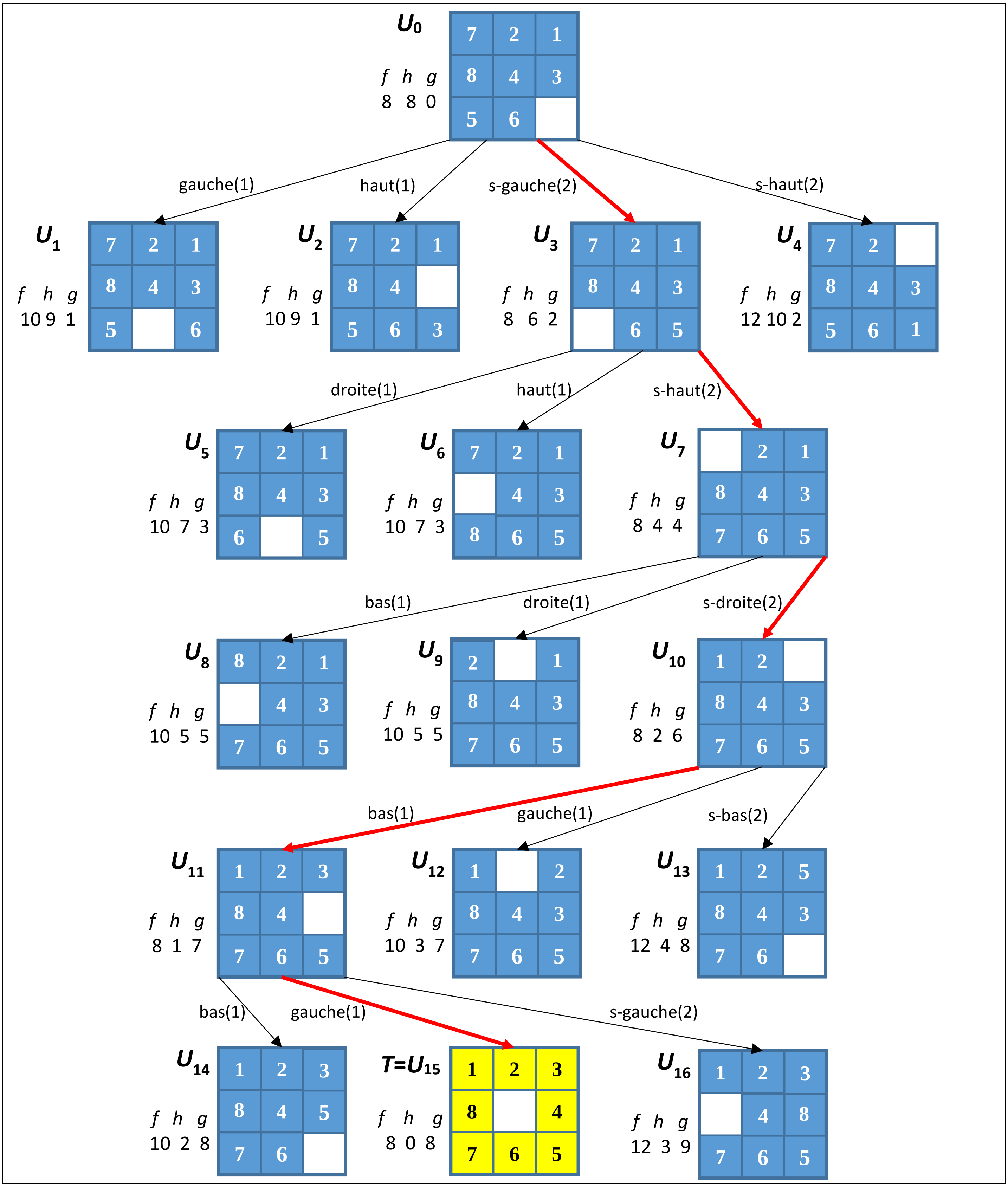
- le cas où l'action ***A* n'a aucun effet** : elle n'éloigne pas la pièce déplacée mais ne la rapproche pas non plus. Dans ce cas : *h*(*u*)–*h*(*v*) = 0. Cependant cette action a quand même un coût *k*(*U*,*V*) =1 ou 2. On vérifie encore que :
  h(*u*)–*h*(*v*) ≤ *k*(*u*,*v*)
- le cas où l'action *A* a comme effet d'**éloigner** la pièce déplacée de sa position finale ;
  si c'est une action simple, il y a un éloignement de *h*(*u*)–*h*(*v*) = 1
  l'action a un coût *k*(*u*,*v*)= 1. On respecte encore *h*(*u*)–*h*(*v*) ≤ *k*(*u*,*v*)
  s'il s'agit d'une super-action, il peut y avoir un éloignement de *h*(*u*)–*h*(*v*) = 2 ; mais dans ce cas l'action a un coût *k*(*u*,*v*)=2 . On respecte toujours *h*(*u*)–*h*(*v*) ≤ *k*(*u*,*v*).

Finalement, dans tous les cas (rapprochement, aucun effet, éloignement), on peut écrire : ***h*(*u*)–*h*(*v*) ≤ *k*(*u*,*v*)**
Donc la fonction *h* est monotone. (1)
Dans l'état terminal *T*, par définition, toutes les pièces sont bien placées (c'est le but) et donc ***h*(*T*)=0** :

***h* est coïncidente.** (2)

h vérifie (1) et (2) donc **est minorante** (théorème démontré en cours). CQFD

Mise en œuvre de A*

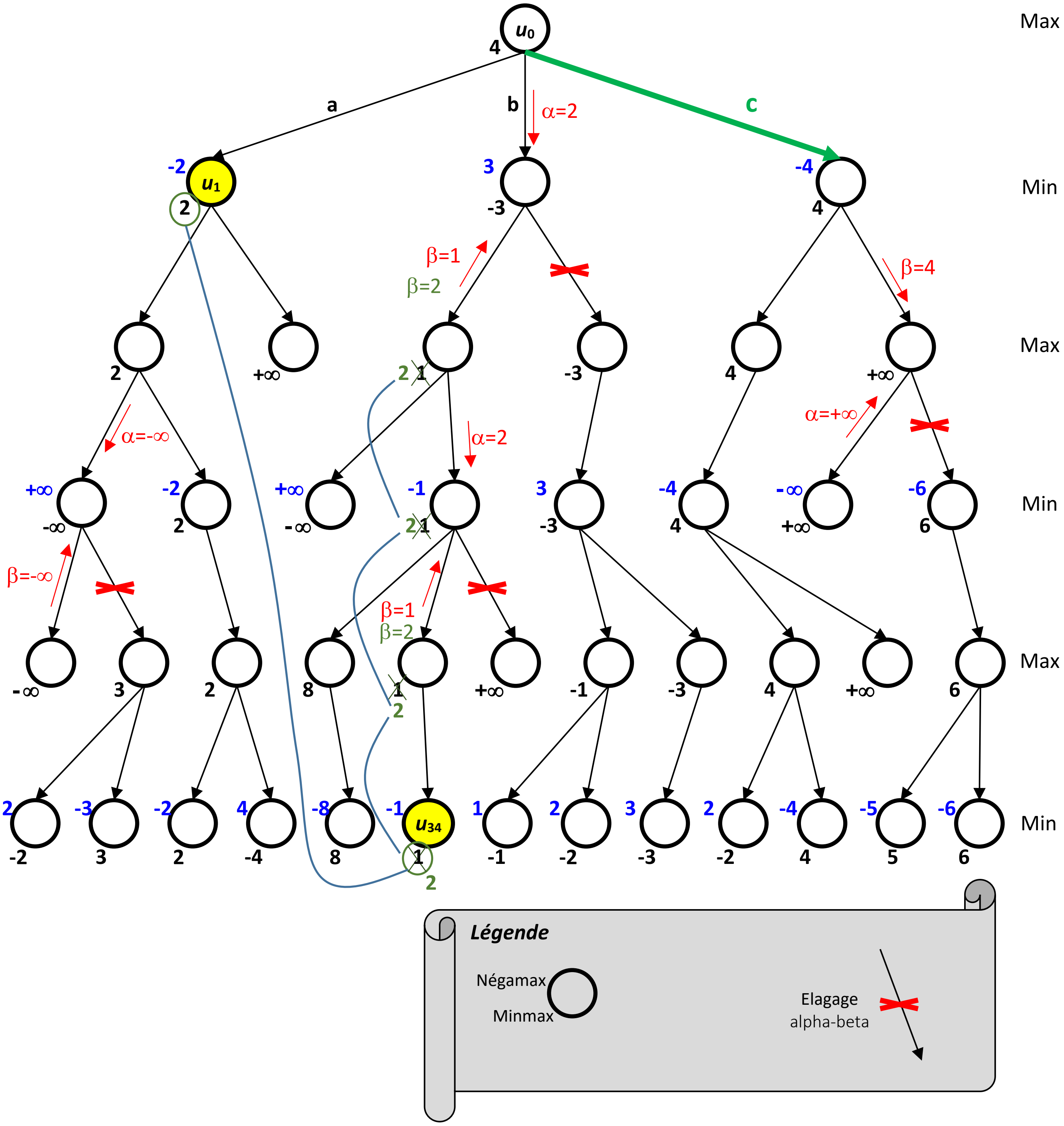


EXERCICE 2 : MINMAX – NEGAMAX – ALPHA-BETA (6 POINTS)

2.1 Le meilleur coup à jouer en u_0 indiqué par l'algorithme minmax est le coup **c**.

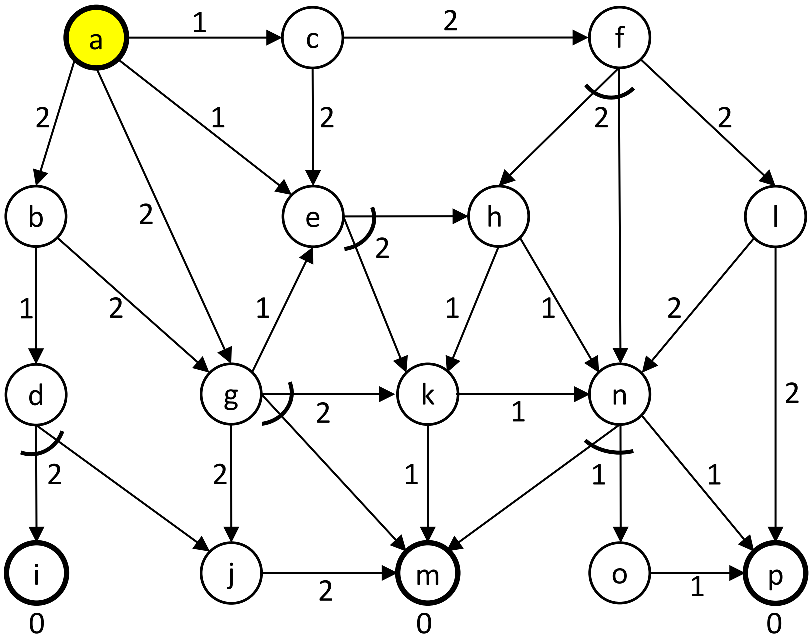
2.4 La situation u_1 a été développée sur 4 coups alors que la situation u_{34} n'a pas été développée ; elle est juste évaluée par une heuristique. C'est la valeur de u_1 qui doit remplacer la valeur de u_{34}.

Cette modification entraîne une actualisation de 3 nœuds situés au-dessus de u_{34} mais pas au-delà. La valeur du coup *b* reste -3 et donc le meilleur coup à jouer en u_0 demeure le coup **c** (+4).



EXERCICE 3 : AO* (5 POINTS)

U	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
h(*U*)	5	4	5	3	4	3	3	2	0	2	1	2	0	1	1	0



Sous-graphe solution de coût total minimal