Louise **Eymery**
Adrien **Sadé** 2A ENPC

OPTIM

Projet : Beating Can't Stop

Question 1.
Soit X_N la variable aléatoire discrète comptant les gains. Si le joueur tire N fois, soit il gagne N fois de suite soit il perd au moins une fois donc, $\mathbb{E}[X_N] = Np^N$. On introduit la fonction associée afin d'en déterminer le maxmimum

$$f : x \mapsto xp^x$$

On dérive,

$$f' : x \mapsto p^x(1 + x \ln p)$$

On en déduit les variations de f,

x	$-\infty$		$-\frac{1}{\ln p}$		$+\infty$
$f'(x)$		$+$	0	$-$	
f	$\nearrow$		$f\left(-\frac{1}{\ln p}\right)$		$\searrow$

Donc f est maximale en $\boxed{x^\star = -\frac{1}{\ln p}}$. Notons que $x^\star$ est bien positif ($p \in (0,1)$). Toutefois, ce résultat n'est pas pratique à utiliser : la valeur n'est pas entière et difficilement manipulable. Pour remédier à cela, on étudie directement la variation d'espérance. Cela revient à déterminer le premier tour à partir duquel l'espérance cesse de croître : on étudie à la place le signe de

$$\mathbf{D_N} = \mathbb{E}[X_{N+1}] - \mathbb{E}[X_N]$$

Cette différence s'écrit

$$\begin{aligned}\mathbf{D_N} &= (N+1)p^{N+1} - Np^N \\ &= p^N\left(N(p-1) + p\right)\end{aligned}$$

Et donc,

$$\mathbf{D_N} \leq 0 \iff N \geq \frac{p}{1-p} \tag{1}$$

En d'autres termes, le premier tour où la condition 1 est vérifiée, l'espérance atteint son maximum. La stratégie à adopter est de continuer à jouer jusqu'au tour

$$\boxed{N^\star = \left\lceil \frac{p}{1-p} \right\rceil}$$

En suivant cette politique, un joueur obtiendra en moyenne un gain optimal de

$$\begin{aligned}\mathbb{E}\left[X_{N^\star}\right] &= N^\star p^{N^\star} \\ &\boxed{= \left\lceil \frac{p}{1-p} \right\rceil p^{\left\lceil \frac{p}{1-p} \right\rceil}}\end{aligned}$$

On propose quelques valeurs pour fixer les idées :

p	0.25	0.5	0.75	0.9
$N^\star$	1	1	3	9
$\mathbb{E}\left[X_{N^\star}\right]$	0.25	0.5	$\simeq 1.27$	$\simeq 3.49$

Si l'on suit cette politique optimale, il n'y a que deux issues possibles : soit le joueur gagne tous ses lancers, soit il en perd au moins un. On en déduit que le gain sous la politique $N^\star$ suit la loi suivante :

$$\left|\begin{array}{ll}\mathbb{P}(X_{N^\star} = k) = 0 & \forall k \in \mathbb{N} \setminus \{0, N^\star\} \\ \mathbb{P}(X_{N^\star} = 0) = 1 - p^{N^\star} & \\ \mathbb{P}(X_{N^\star} = N^\star) = p^{N^\star} & \end{array}\right.$$

On en déduit la probabilité de terminer le tour sans gain sur cette tentative :

$$\boxed{\mathbb{P}(X_{N^\star} = 0) = 1 - p^{N^\star}}$$

Question 2.

Considérons que l'ensemble des états est l'ensemble des entiers compris entre 0 et G_{max}. Chaque état représente le gain courant à la fin d'un tour. Dans cette modélisation, une politique correspond au nombre de lancers à tenter à chaque tour. On a ainsi, $V_t(G_{\max}) = 0$ et pour $G \in [0, G_{\max}[$:

$$\begin{aligned} V_T(G) &= 0 \text{ pour } T \text{ grand, par exemple } T = 2G_{\max} \\ V_t(G) &= \min_{a \in \mathcal{A}} 1 + \mathbb{E}[V_{t+1}(G + \text{gain au tour t})] \text{ car il faut encore au moins un tour} \\ &= \min_{1 \leq k < G_{\max} - G} (1 + p^k V_{t+1}(G + k) + (1 - p^k) V_{t+1}(G)) \\ &\quad \text{avec } k \text{ le nombre de lancers avant de s'arrêter à ce tour} \end{aligned}$$

Avec cette méthode, pour $G_{\max} = 50$ et $p = 0.7$, l'action à effectuer est indépendante de l'itération t. Si le gain G est entre 0 et 45, on joue trois fois. S'il vaut 46, on ne lance que deux fois (il faudra statistiquement au moins un lancer supplémentaire pour arriver à $G_{\max}$, mais on peut terminer rapidement). Sinon, on lance $G_{\max} - G$ fois.
Intuitivement, lorsque le gain $G_{\max}$ à atteindre est très grand, il convient de chercher à maximiser son gain provisoire à chaque tentative afin de minimiser le nombre de tours nécessaires. Dans ce cadre $G_{\max} \to +\infty$, la politique optimale est donc donnée par la question 1 : il convient de s'arrêter et de changer de tour (et donc assurer son gain provisoire) après $N^\star(p)$ lancers à chaque tour.
En partant de $g = 0$, notre méthode nous indique donc de lancer la plupart du temps 3 coups par tour. A la question précédente, on terminait un tour après $N^\star$ lancers, avec $N^\star = \left\lceil \frac{p}{1-p} \right\rceil = 3$. On retrouve la même stratégie, pour $G_{\max}$ grand.

Étudions maintenant une deuxième méthode, plus facilement généralisable pour la suite du problème (notamment pour la question 4).
On cherche $\min_a \mathbb{E}_a(\sum_t c_t(X_t, X_{t+1}))$ où X_t est la variable aléatoire donnant le gain total (gain courant ou gain provisoire) à l'instant t. On note

$$V_t(x) = \min_a \mathbb{E}_a(c_t(X_t, X_{t+1}) + V_{t+1}(X_{t+1}) \mid X_t = x, a_t = a)$$

avec :

$$\begin{aligned} c_t(X_t, X_{t+1}) &= 0 \text{ si } X_{t+1} \text{ correspond à un gain provisoire} \\ &= 1 \text{ si } X_{t+1} \text{ correspond à un gain courant} \end{aligned}$$

On décrit l'espace des états et des actions avec la figure 1.
La fonction implémentant cette équation et renvoyant la politique optimale à suivre (donc minimisant en espérance le nombre de tours à effectuer pour atteindre $G_{\max}$) en fonction du gain à atteindre est disponible dans le fichier Julia attaché `Question2.jl`.
Observons les résultats de la programmation :

- Pour $G_{\max} = 10$ et $p = 0.5$, l'action à effectuer est indépendante de l'itération t. Si notre gain provisoire est égal au gain courant : on n'a rien à perdre donc on rejoue. Si le gain provisoire vaut 1 de plus que le gain courant (et que le gain courant est inférieur à 9), on rejoue. Dans le cas contraire, on s'arrête.
- Pour $G_{\max} = 50$ et $p < 0.5$, on n'effectue qu'un seul lancer par tour.
- Pour $G_{\max} = 50$ et $p = 0.7$, tant que le gain courant est inférieur ou égal à 17, on continue à lancer jusqu'à atteindre un gain provisoire de 23 (ou jusqu'à rater son lancer). Puis, pour un gain courant entre 17 et 23, le nombre de lancers varie. Enfin, pour un gain courant supérieur ou égal à 24, on lance trois fois (ou jusqu'à rater son lancer ou arriver à $G_{\max}$).

Cette méthode est cependant assez coûteuse en temps de calcul. Pour $p > 0.5$ (lorsqu'il est avantageux de lancer plus d'une fois par tour), on ne retrouve pas le même résultat qu'avec la méthode précédente ou la question 1.

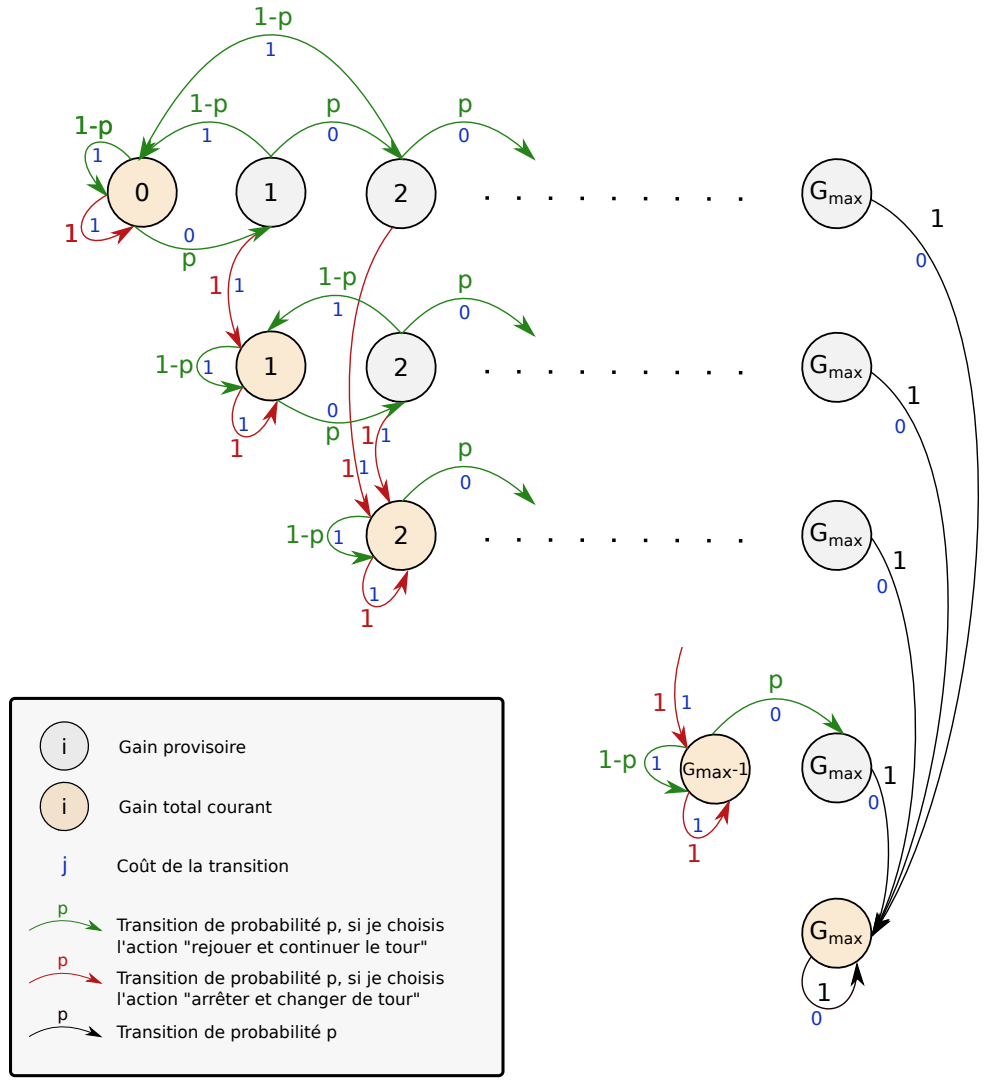

FIGURE 1 – Graphe des états de la chaîne de Markov représentant l'évolution du gain du joueur

Question 3.
Le fichier `Question3.jl` (ou encore `policies_15.jl`) contient la fonction `proba_legal_move(i,j,k)` qui permet de calculer la probabilité qu'il existe au moins une solution de déplacement des pions sur une colonne (« legal move »). On en propose quelques valeurs dans le tableau ci-dessous :

Colonnes (i,j,k)	$(6,7,8)$	$(2,5,9)$	$(2,3,12)$
Probabilité p	$\simeq 0.92$	$\simeq 0.76$	$\simeq 0.44$

On vient de déterminer la probabilité p d'obtenir un déplacement autorisé après un lancer de dés. Ainsi, tant que les 3 colonnes sont ouvertes (hypothèses de bornes infinies), on connaît la probabilité de pouvoir continuer à jouer, ou de perdre. On vient de décrire le cadre des premières questions : on en déduit la politique optimale (pour maximiser le nombre de déplacements avant de s'arrêter). La politique optimale à suivre est donc de s'arrêter après $N^\star(p)$ lancers de dés.
La probabilité de terminer le tour sans augmenter son gain courant, c'est-à-dire sans déplacer un pion définitif sous cette stratégie est donc donnée par la loi explicitée en question 1 :

$$\mathbb{P}(\text{« terminer le tour sans déplacer un pion définitif »}) = \mathbb{P}(X_{N^\star} = 0) = 1 - p^{N^\star}$$

Question 4.
Les colonnes i, j, k sont ouvertes. Notons $G_{max_i}, G_{max_j}, G_{max_k}$ les longueurs respectives de ces trois colonnes. Le gain courant sur une colonne désigne l'avancée du pion de couleur sur cette colonne. Le gain provisoire concerne l'avancée du pion noir sur cette colonne.

L'ensemble des états est $(G_{max_i} \times (t_g, c_g)) \times (G_{max_j} \times (t_g, c_g)) \times (G_{max_k} \times (t_g, c_g))$. Un état est donc de la forme $x = ((x_i, q_i), (x_j, q_j), (x_k, q_k))$ où x_m donne l'avancée sur la colonne m et q_m indique si cette avancée est provisoire ou définitive (c'est-à-dire si le gain est provisoire ou courant) On a alors :

$P^a(x, y) = 0$ si trois coordonnées changent entre x et y, càd si $x_i \neq y_i$ et $x_j \neq y_j$ et $x_k \neq y_k$
$= p_1(i, j, k)$ si deux coordonnées changent entre x et y, qu'elles augmentent toutes les deux de 1 et que ces coordonnées correspondent à des gains provisoires chez y
$= p_2(i, j, k)$ si une coordonnée change entre x et y, qu'elle augmente de deux et que cette coordonnée correspond à un gain provisoire chez y
$= p_3(i, j, k)$ si une coordonnée change entre x et y, qu'elle augmente d'un et que cette coordonnée correspond à un gain provisoire chez y
$= 1$ si certaines coordonnées de x correspondent à un gain provisoire, celles de y à un gain courant et que leurs coordonnées ont la même valeur

$c(x, y) = 1$ si x correspond à un gain provisoire, y à un gain courant et que leurs coordonnées ont la même valeur
$= 0$ sinon

Puis on pose, pour T grand (par exemple $T = 2 \times G_{max_i} \times G_{max_j} \times G_{max_k}$),

$$V_T(x) = 0$$
$$V_t(x) = \min_{a \in \mathcal{A}} \sum_{y \in \mathcal{X}} P^a(x, y) \left(c(x, y) + V_{t+1}(y)\right)$$

Cette équation nous donne, par programmation dynamique, le minimum de l'espérance du nombre de tours à jouer avant de fermer les 3 colonnes.

Question 5.
On propose une heuristique pour minimiser le nombre de tours avant de fermer 3 colonnes. L'heuristique retenue est décrite ci-après.

Heuristique Q5
L'heuristique est définie sur un tour. À chaque lancer de dés :
- Si moins de 3 colonnes sont ouvertes :
  - On calcule la « probabilité améliorée » des différents mouvements admissibles (le calcul est fait en sommant les probabilités améliorées de chacune des colonnes différentes du mouvement admissible, si la même colonne est proposée deux fois sa probabilité améliorée est ajustée d'un poids de 5/4 [a]).
  - On choisit le mouvement de plus grande probabilité améliorée selon ce calcul.
  - **On choisit de relancer les dés** systématiquement.
- Si 3 colonnes sont ouvertes :
  - On calcules les probabilités améliorées de chacun des mouvements admissibles [b].
  - On choisit le mouvement de plus grande probabilité améliorée selon ce calcul.
    - Si au mois une des 3 colonnes est fermée : **on arrête de lancer les dés et on finit le tour**.
    - Sinon :
      - On calcule $N^\star = \left\lceil \frac{p}{1-p} \right\rceil$, où p est la probabilité d'obtenir un mouvement autorisé (legal move) sur le triplet de colonnes ouvertes.
      - Si le nombre de lancers courant est inférieur à ce $N^\star$: **on choisit de relancer les dés**.
      - Sinon : **on arrête de lancer les dés et on finit le tour**.

a. On ne prend pas deux fois cette probabilité car il est certainement souvent plus intéressant d'ouvrir deux colonnes différentes sur le long terme. On lui attribue un poids supérieur à 1 pour prendre en compte le fait d'avancer de deux cases d'un coup. Le facteur 5/4 est obtenu après une descente locale.

b. Cette fois-ci, on somme les probabilités même si on a 2 fois la même colonne : on ne regarde plus l'avenir lointain ni l'ouverture de colonnes.

La « probabilité améliorée » d'une colonne correspond à la probabilité, suite à un lancer de 4 dés, de pouvoir se déplacer sur cette colonne, divisée par le nombre de cases restantes à parcourir pour fermer cette colonne. Ce score (qui n'est pas vraiment une probabilité) permet ainsi de prendre en compte l'avancement d'un joueur sur la plateau. Elle nous donne l'intérêt de se déplacer sur une colonne. Cette probabilité est calculée au début du fichier `policies_15.jl` à l'aide de la fonction

```
function proba_amelioree(n_avancement::Int, column::Int)
```

Cette heuristique permet d'obtenir les résultats donnés par le tableau 1 (obtenus sur 1000 simulations du jeu) :

Nombre de tours	**Proportion de tours « failed »**
8.87	48.1%

TABLE 1 – Résultats heuristique Q5

Remarque : Dans l'ensemble de ce projet, les résultats ont généralement une grande variance. D'une simulation à l'autre, on obtient parfois des écarts relativement conséquents (malgré les 1000 répétitions). Les résultats présentés correspondent aux meilleurs scores que l'on a réussi à obtenir. Ainsi, un joueur souhaitant utiliser nos heuristiques doit être conscient que ces valeurs sont presque des bornes, atteignables, sur la qualité de nos heuristiques. Un aspect intéressant, serait de chercher à réduire cette variance : l'objectif serait d'assurer une valeur à toutes les parties, plutôt que de maximiser une moyenne. Enfin, nous n'avons pas augmenté le nombre de simulations car nos heuristiques demandent en général déjà quelques minutes pour 1000 simulations.

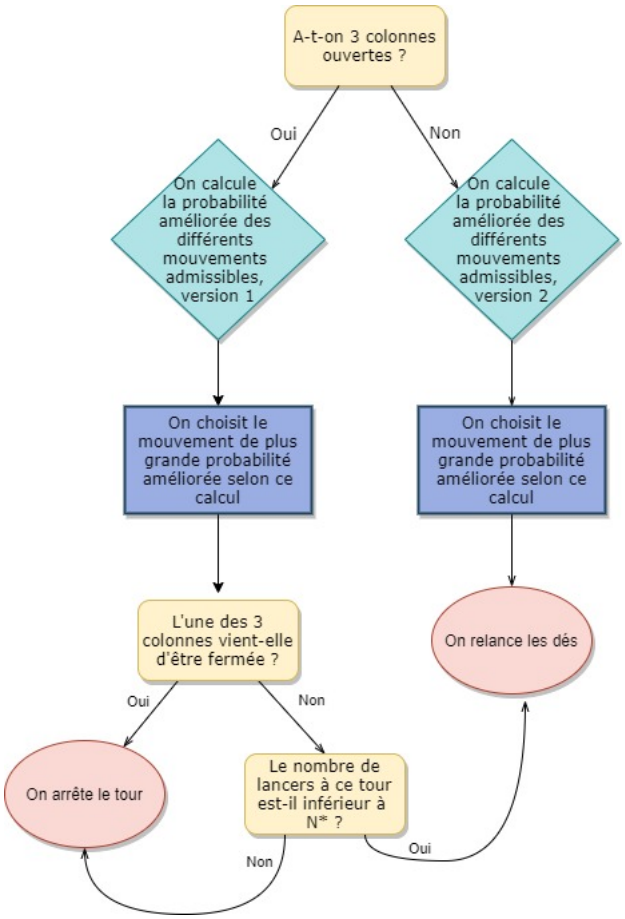

FIGURE 2 – Schéma expliquant l'heuristique Q5 pour un seul joueur

Question 6.
Première Heuristique
Pour établir une stratégie compétitive, on s'est appuyés sur l'heuristique de la question précédente tout en prenant en compte la présence de l'autre joueur, et de l'urgence qu'il peut imposer s'il est proche de fermer 3 colonnes. Ainsi, nous avons proposé l'heuristique suivante, dont la figure 3 permet de visualiser plus aisément son fonctionnement :

Heuristique 1 Q6
L'heuristique est définie sur un tour. À chaque lancer de dés :
- Si moins de 3 colonnes sont ouvertes :
  - On calcule la « probabilité améliorée » des différents mouvements admissibles (le calcul est fait en sommant les probabilités améliorées de chacune des colonnes différentes du mouvement admissible, si la même colonne est proposée deux fois sa probabilité améliorée est ajustée d'un poids de $5/4$ [a]).
  - On choisit le mouvement de plus grande probabilité améliorée selon ce calcul.
  - **On choisit de relancer les dés** systématiquement.
- Si 3 colonnes sont ouvertes :
  - On calcules les probabilités améliorées de chacun des mouvements admissibles [b].
  - On choisit le mouvement de plus grande probabilité améliorée selon ce calcul, parmi les colonnes où l'adversaire ne se trouve pas.
    - Si au mois une des 3 colonnes est fermée : **on arrête de lancer les dés et on finit le tour**.
    - Sinon :
      - Si l'adversaire est « proche » de gagner : **on choisit de relancer les dés**.
      - Sinon :
      - On calcule $N^\star = \left\lceil \frac{p}{1-p} \right\rceil$, où p est la probabilité d'obtenir un mouvement autorisé (legal move) sur le triplet de colonnes ouvertes.
      - Si le nombre de lancers courant est inférieur à ce $N^\star$: **on choisit de relancer les dés**.
      - Sinon : **on arrête de lancer les dés et on finit le tour**.

a. On ne prend pas deux fois cette probabilité car il est certainement souvent plus intéressant d'ouvrir deux colonnes différentes sur le long terme. On lui attribue un poids supérieur à 1 pour prendre en compte le fait d'avancer de deux cases d'un coup. Le facteur $5/4$ est obtenu après une descente locale.
b. Cette fois-ci, on somme les probabilités même si on a 2 fois la même colonne : on ne regarde plus l'avenir lointain ni l'ouverture de colonnes.

Remarque : Ici, on a préféré se déplacer sur des colonnes où l'adversaire ne se trouve pas, pour éviter que l'adversaire termine avant nous cette colonne et qu'on y perde notre progression. S'il n'existe pas de telle colonne, on se déplace sur le mouvement qui donne la plus grande probabilité améliorée. Si l'adversaire est « proche » de gagner au prochain tout, on décide de rejouer jusqu'à gagner ou obtenir un mouvement interdit : on tente le tout pour le tout. Cette notion de « proche » est définie par la comparaison de la probabilité améliorée du joueur adverse sur chacune des colonne avec un seuil : s'il a déjà fermé 2 colonnes et s'il existe une colonne où le joueur adverse est suffisamment avancé pour dépasser ce seuil, on considère qu'il est proche de gagner. Ce seuil est choisi à l'aide d'une descente locale, maximisant notre nombre de parties gagnées.

Pour évaluer notre stratégie, on la fait jouer contre la stratégie `First` qui, à chaque tour, choisit le premier mouvement admissible calculé et s'arrête après 2 lancers. Ainsi, pour 1000 simulations du jeu contre la stratégie `First`, cette heuristique permet d'obtenir les résultats répertoriés dans le tableau 2.

Nombre de parties gagnées	Nombre de tours en moyenne
864	17.67

Table 2 – Résultats heuristique 1 Q6

Remarque : En réalité, l'ordre des joueurs influe sur leur taux de victoire. Le premier joueur a un avantage sur les autres. Cet avantage ayant peu d'importance, on choisit de l'ignorer dans la suite du rapport (si les chiffres changent légèrement, les conclusions sur la puissance des heuristiques restent les mêmes, surtout d'un point de vue relatif aux autres heuristiques qui sont également soumises à ces

fluctuations). Dans l'ensemble du rapport, les résultats présentés ont été simulés tel que le premier joueur suit toujours notre stratégie.

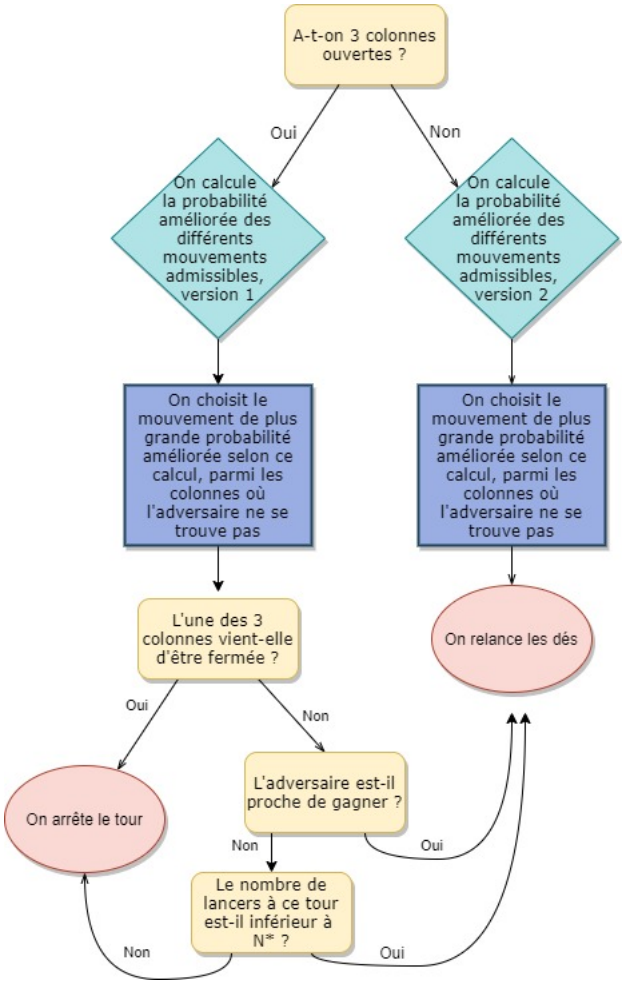

FIGURE 3 – Schéma expliquant l'heuristique 1 Q6 pour deux joueurs

Seconde Heuristique
À ce stade on obtient une heuristique qui fournit des résultats *a priori* corrects. Néanmoins, on se propose de l'améliorer encore un peu. Après avoir étudié son comportement, on comprend deux choses :

- l'adversaire a tendance à fermer des colonnes de plus grande probabilité que nous : cela vient du fait qu'on refuse d'avancer sur des colonnes où il est présent.
- on a un grand nombre de « failed throws », comme constaté lors de la question 5.

Le premier point implique que l'adversaire est laissé seul sur les « meilleures » colonnes, mais aussi qu'il ne risque jamais d'y perdre son avancement ! Le deuxième point suggère qu'on perd des tours et qu'on laisse le temps à l'adversaire d'avancer (et en plus il nous bloque l'accès à de plus en plus de colonnes d'après le premier point). Finalement, on cherche à pallier à ces deux problèmes de la façon qui suit :

- on simplifie notre première heuristique pour qu'elle ne prenne plus en compte la présence ou non de l'adversaire lorsqu'elle doit choisir un mouvement,
- le $N^\star$ est diminué de 3 lancers systématiquement.

Remarque : Un grand nombre de variantes ont été testées mais c'est celle-ci que nous avons retenue pour ses résultats. Pour diminuer le nombre de lancers limite $N^\star$, de nombreuses méthodes ont été testées (linéaires, non linéaires, clipping,...), c'est finalement une diminution constante qui a été retenue et déterminée à l'aide d'une descente locale.

Désormais, en jouant toujours contre la startégie `First`, sur 1000 simulations, on obtient les résultats du tableau 3 :

Nombre de parties gagnées	Nombre de tours en moyenne
911	17.37

Table 3 – Résultats heuristique 2 Q6

Les résultats obtenus sont donc bien meilleurs. Dans toute la suite du rapport, « l'heuristique de la question 6 » désignera donc cette seconde heuristique. Dans le fichier `policies_15.jl`, la fonction

```
function policy_q6(gs::game_state, adm_movement)
```

contient le code de cette heuristique, tandis que la fonction

```
function policy_q6_old(gs::game_state, adm_movement)
```

contient le code de la première heuristique.

Question 7.
Notre première idée est de regarder si les adversaires sont proches de gagner. Si non, alors on essaie d'avancer sur le plus de colonnes en même temps. Sinon, on essaie de terminer le plus vite possible. L'idée est que, lorsque l'on a encore « du temps », on choisit d'avancer sur un maximum de colonnes différentes : à terme, on aura peut-être la possibilité de fermer, non pas 3, mais 4 colonnes lors du tour gagnant (2 colonnes en un lancer après en avoir déjà fermé 2). Cette idée a donné une première heuristique. Malheureusement, après l'avoir implémentée, nous avons comparé les résultats qu'elle donnait à ceux obtenus directement avec l'heuristique de la question 8 (cf question suivante). Suite à cette constatation, on choisit de retenir l'heuristique de la question suivante pour cette question également. On obtient les résultats du tableau 4 (obtenus sur 1000 simulations du jeu contre un joueur suivant la stratégie `First`) :

	Nombre de colonnes fermées
Jeu à 3 joueurs	2.88
Jeu à 4 joueurs	2.89

Table 4 – Résultats heuristique Q7-8

Question 8.
L'objectif dans cette question est de gagner, c'est-à-dire, d'être le premier à fermer 3 colonnes dans un jeu multijoueur. L'objectif étant similaire à celui de la question 6, on réutilise l'heuristique de cette question, adaptée à plusieurs adversaires (dans la politique d'arrêt optimal, on s'assure qu'il n'existe aucun adversaire qui est proche de gagner).
Cette heuristique permet d'obtenir les résultats du tableau 5 (obtenus sur 1000 simulations du jeu contre des joueurs suivant tous la stratégie `First`) :

	Nombre de parties gagnées	Nombre de tours en moyenne
Jeu à 3 joueurs	883	25.73
Jeu à 4 joueurs	892	33.88

Table 5 – Résultats heuristique Q8