Sorbonne Université

UE 4IN200 - « MOGPL », Master Informatique

Rapport de projet DICE BATTLE

Calle Viera Andersson, Khati-Lefrançois Elias

Encadré par
ESCOFFIER Bruno

Table des matières

Introduction

Deux joueurs s'affrontent dans un jeu de dés. Le but étant d'être le premier à atteindre au moins N points (généralement $N = 100$). À chaque tour, on choisit de lancer entre 1 et D dés à six faces (par exemple, D = 10), alors :

— Si l'un des dés au moins tombe sur 1, on marque 1 **point**.

— Dans le cas contraire on marque le nombre de points equivalent à la somme des dés.

Nous allons étudier deux variantes (séquentielle et simultanée) de ce jeu. Le but du projet est de déterminer une stratégie de jeu optimale en testant différentes stratégies et on les comparant entre elles. On dit qu'un joueur a un gain égal à 1 s'il remporte la partie, un gain égal à 0 si la partie est nulle, un gain égal à -1 s'il perd. C'est donc en particulier un jeu à somme nulle.
L'implémentation du jeu sera faite en python 3.x dans un fichier **DB_LF_CV.py**

1 Probabilités

Notons $P(d,k)$ la probabilité qu'un joueur qui lance d dés obtienne k points, on rappel alors que :

— $\forall d \geq 1$, $P(d,1) = P($"obtenir 1 point | k dés$) = P($"obtenir tout sauf 1 point | k dés$)^C = 1 - \left(\frac{5}{6}\right)^d$

— $\forall k \in [\![2,\ 2d-1]\!] \cup [\![6d+1,\ +\infty[\![$, $P(d,k){=}0$

— $\forall k \in [\![2d,\ 6d]\!]$, $P(d,k) = \left(\frac{5}{6}\right)^d Q(d,k)$ où, $Q(d,k) = \sum_{i=2}^{6} \frac{Q(d-1,k-i)}{5}$

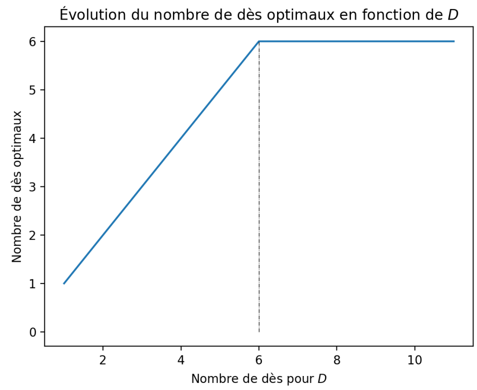
Cette formule de récurrence est itérée jusqu'à $(d-1) = 1$, qui est notre cas de base, et ce pour $(k-i) \in [\![2(d-1),\ 6(d-1)]\!]$, en effet pour tout les autres cas on a que $Q(d,k) = P(d,k) = 0$.
En notant les variables aléatoires, K le nombre de points obtenus et Δ le nombre de dès lancés, on a : Q(1, k) = P(K = k| Δ = 1, k $\neq$ 1) = $P(k \in [\![2,\ 6]\!]) = \frac{1}{5}$

2 Variante Séquentielle

Dans cette variante du jeu les deux joueurs jouent à tour de rôle. En particulier le premier joueur à jouer a donc bien sûr un avantage par rapport à l'autre.

Stratégie Aveugle :

Q2) On peut remarquer que $\forall\ D \in \mathbb{N}$: $\underset{d \in [\![1,D]\!]}{\operatorname{argmax}} EP(d)$ est une fonction croissante sur ce même intervalle. Entre 0 et 6 elle est plus exactement linéaire, puis elle atteint son maximum en 6 et elle devient constante. Pour $D < 6$, la stratégie aveugle consiste donc "seulement" à choisir d*(D) égale à D tandis que pour $D \geq 6$ d*(D) vaut toujours 6, ce qui nous fait nous interroger sur son efficacité.



Évolution de d*(D) selon D

Q3) On peut constater que choisir systématiquement d*(D) n'est pas souhaitable, par exemple : Dans un cas où J1 est a 98 points, que N=100 et que D est tel que la valeur de d*(D) est 4, il est alors plus intéressant pour J1 de ne lancer qu'un seul dé plutot que d*(D) dés. En effet, en lançant un seul dé J1 a une probalilité de finir la partie de $\frac{5}{6}$ contre $(\frac{5}{6})^4$ en en lançant 4.

Programmation Dynamique :

Q4) On a :

$$\begin{cases} EG(i,j) = 1 & \text{si } i \geq N\ \forall j \\ EG(i,j) = -1 & \text{si } i < N \text{ et } j \geq N \\ EG(i,j) = \max\limits_{d \in [\![1,D]\!]} \left(\sum_{k=1}^{6*d} -EG(j,i+k) * P(d,k) \right) & \text{si } i < N \text{ et } j < N \end{cases}$$

Concernant le cas où $i \geq N$, si $j \geq N$, alors J1 gagne tout de même car il s'agit de la variante séquentielle et par conséquent, J1 dépassera N en premier car la partie s'arretera imédiatement à cet instant.

Concernant le cas où i et j sont plus petits que N, on utilise le fait que le jeu est à somme nulle pour pouvoir dire que l'espérance dépend de l'opposé de l'espérance de gain de J2 avec la nouvelle situation.

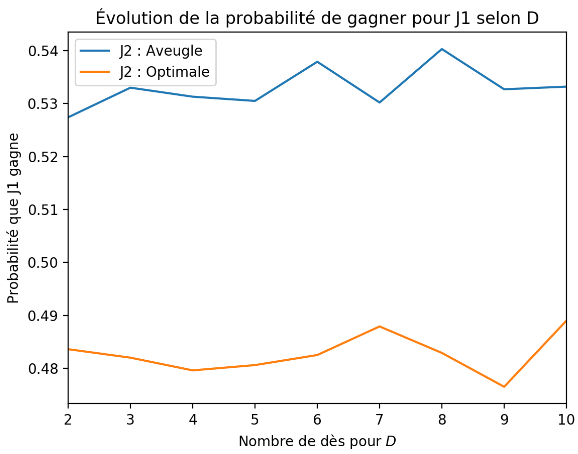
On itère donc cela pour toutes les valeurs que peuvent prendre le nouvel état afin d'en retirer la plus avantageuse pour J1.

En pratique, on va préférer initialiser le tableau EG avec les valeurs 1 et -1 au préalable, afin d'éviter des appels récursif peu utiles.

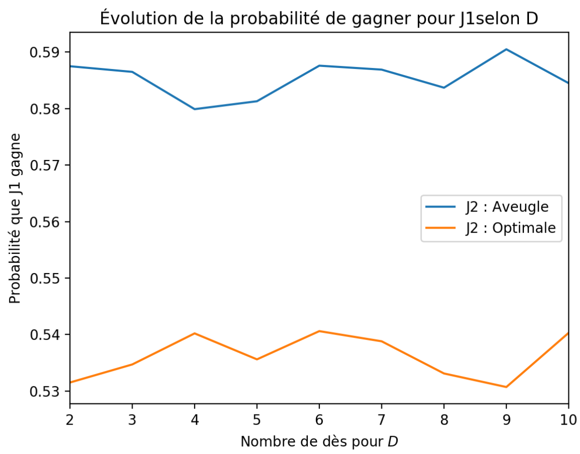
Q5) Pour adapter l'algorithme précédemment construit afin d'en retirer le nombre de dés à lancer pour jouer à chaque fois de manière optimale il suffit, lors de la création du tableau EG, de stocker dans un tableau annexe, aux indices (i,j) la valeur de d correspondant à la valeur de d donnant le calcul d'espérance maximum tel que décrit à la question précédente pour l'état de jeu (i,j). Par conséquent, lorsque l'on souhaite jouer de manière optimale, on prend l'état actuel (i,j) et l'on va chercher la valeur dans ce tableau annexe aux indices (i,j) qui va nous indiquer le nombre de dés à lancer.

Q6) Le fait d'avoir un cas où l'on gagne 0 points ne marchera pas avec l'idée de rendre récursive la fonction. En effet on se retrouvera dans un boucle infinie où pour un état (i,j) donné, l'espérance qui y est lié a besoin de ce même état (i,j) qui a lui même besoin de cet état *etc.*

Q8)

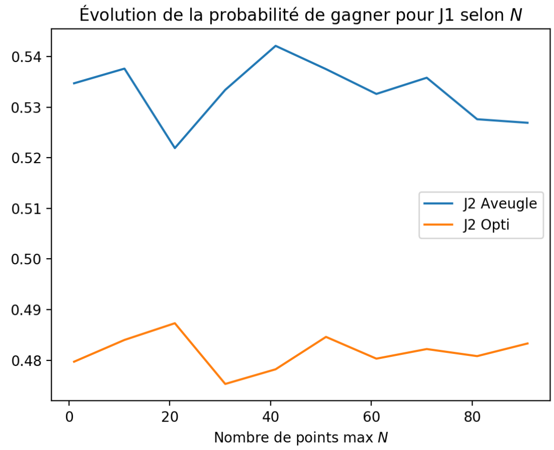


Stratégie de J1 : Aveugle.

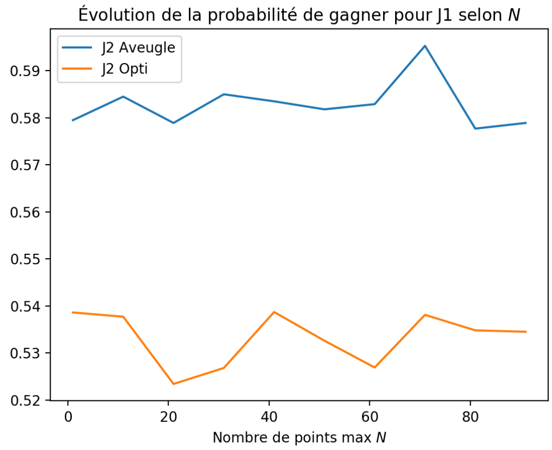


Stratégie de J1 : Optimale.

On peut remarquer que, étant le premier à jouer, le Joueur 1 commence déjà avec un avantage. Néanmoins quand il choisit la stratégie Aveugle il a moins d'une chance sur deux de gagner en comparaison du Joueur 2 qui lui a choisit la stratégie Optimale ce qui prouve l'efficacité de cette dernière. D'un autre coté si le premier joueur choisit d'adopter la stratégie Optimale il a non seulement toujours plus de 50% de chances de gagner mais aussi et surtout ses probabilités de gagner avec cette stratégie en comparaison de celles avec l'autre sont plus grandes.



Stratégie de J1 : Aveugle.



Stratégie de J1 : Optimale.

Ici aussi on peut observer le même phénomène que pour l'évolution de la probabilité de gagner selon D avec n"anmoins une plus grande variance pour les probabilités.
On peut donc en conclure que pour n'importe quelle valeur de N ou de D, même si le Joueur 2 choisit la stratégie Optimale le Joueur 1 gagne plus souvent en choisissant la stratégie Optimale qu'en choissisant l'autre. Ce que le Joueur 2 peut faire pour "limiter" les dégats est de choisir aussi la stratégie Optimale.

Q9) En temps que stratégie suplémentaire, nous avons programmé une stratégie aléatoire qui à chaque tour va lancer un nombre de dés compris entre 1 et D le choix de cette valeur étant équitable. La seconde stratégie supplémentaire que nous avons eu le temps d'implémenter et tester est une stratégie consistant à lancer à tout les tours le nombre maximum de dés, donc D dés.

Voici le tableau d'espérances de gains de J1 calculées de manière expérimentalement (réalisant 10000 parties pour chaque combinaisons) :

		J2			
		aléatoire	aveugle	optimale	maximum
J1	aléatoire	0.068	-0.170	-0.342	0.182
	aveugle	0.268	0.068	0.006	0.376
	optimale	0.404	0.144	0.072	0.412
	maximum	-0.118	-0.302	-0.384	0.094

Il est à noter que la startégie que le joueur 1 devrait éviter est celle qui consiste à jouer D dès à chaque tour, ce qui correspond à notre intuition. Par ailleur, pour J1 comme pour J2 la meilleur stratégie à choisir est celle "optimale".

3 Variante Simultanée

A présent, on va aborder la version du jeu où les deux joueurs choisissent en même temps le nombre de dés qu'ils vont lancer. Cela supprime entre autre l'avantage que le premier joueur avait sur le second dans la version séquentielle. Concernant le jeu en un coup, il suffit d'établir N à 1 et ainsi la partie s'arrêtera forcément après le premier tour.

Jeu en un coup :

Q10) Dans le cas énoncé à la question 10 (jeu en un seul coup, le plus grand score l'emporte), on a $EG_1(d1, d2)$ tel que :
$EG_1(d1, d2) = 1 * P_{J1} - 1 * P_{J2} + 0 * P_{Nul}$

Où P_{J1} est la probabilité que J1 gagne, P_{J2} celle que J2 gagne, et P_{Nul} celle qu'il y ai un match nul. On a donc :

$$P_{J1} = \sum_{x=1}^{6*d2} \sum_{y=x+1}^{6*d1} P(d1, y) * P(d2, x)$$

$$P_{J2} = \sum_{x=1}^{6*d1} \sum_{y=x+1}^{6*d2} P(d1, x) * P(d2, y)$$

$$P_{Nul} = \sum_{x=1}^{6*k} P(d1, x) * P(d2, x) \quad \text{où } k = min(d1, d2)$$

Par conséquent,

$$EG_1(d1, d2) = \sum_{x=1}^{6*d2} \sum_{y=x+1}^{6*d1} P(d1, y) * P(d2, x) - \sum_{x=1}^{6*d1} \sum_{y=x+1}^{6*d2} P(d1, x) * P(d2, y)$$

Voici la matrice des gains pour D=3 :

		d1		
		1	2	3
	1	0	0.375	0.2268
d2	2	-0.375	0	0.1988
	3	-0.2268	-0.1988	0

Q11) Si J2 a connaissance du vecteur de probabilité avec lequel J1 va choisir le nombre de dés qu'il va lancer, J2 connais donc l'expression de l'ésperance des gains de J1 en fonction d'une valeur d2 le nombre de dés lancés par J2 fixée :

$E = p_1(1) * EG_1(1, d2) + ... + p_1(D) * EG_1(D, d2)$

J2 dois donc chercher à minimiser les gains de J1 afin d'augmenter les siens, le jeu étant à somme nulle, et donc jouer d2 dés qui minimise l'expression E. Il faut avoir d2 tel que :

$d2 \in [\![1, D]\!]$

Q12) Si J1 doit choisir les probabilitées avec lesquels il va choisir le nombre de dés à lancer, mais qu'il sait également que J2 va choisir son nombre de dés en ayant connaissance de ces probabilités, on va chercher à maximiser l'expression précédente. On a donc, avec EG_1 tel que décrite précédemment et $p_1(i)$ la probabilité pour J1 de jouer i dés :

$$\begin{array}{ll} \text{maximize} & z \\ \text{s.c. :} & \begin{cases} \sum_{i=1}^{D} EG_1(i, j) * p_1(i) \geq z & \forall j \in [\![1, D]\!] \\ \sum_{i=1}^{D} p_1(i) = 1 & \\ p_1(i) \geq 0 & \forall i \in [\![1, D]\!] \end{cases} \end{array}$$

Q13) La fonction implémentée résolvant le programme linéaire nous permet d'obtenir le tableau, des stratégies optimales, suivant :

		Valeur de D		
		3	5	10
$p_1(i)$	1	0	0	0
	2	0	0.175	0.175
	3	1	0.053	0.053
	4	0	0	0
	5	0	0.771	0.771
	6	0	0	0
	7	0	0	0
	8	0	0	0
	9	0	0	0
	10	0	0	0

On constate qu'en dessous d'une valeur de 5 pour D, on aura toujours $p_(1)(D) = 1$ et les autres valeurs $p_1(i) = 0$. De plus, pour un D>5, on aura des valeurs de p_1 indentiques, du moins jusqu'à D=10.

Q14) Pour rappel, la stratégie aveugle consiste à choisir la valeur de d maximisant EP(d). En évaluant experimentalement les stratégie aveugles et optimales en variante simultanée, on obtient la matrice de gain suivante :

		J2	
		aveugle	optimale
J1	aveugle	-0.006	-0.002
	optimale	0.033	-0.014

Le tableau observé ne semble pas être réellement significatif. Cela peut être dû à plusieurs choses, entre autre une mauvaise interprétation de la stratégie aveugle dans cette variante simultanée. Cependant, le jeu dans cette variante est symétrique si les deux joueurs ont la même stratégie, ce qui doit ce refletter par un gain de 0 (ou environnant) sur la diagonale du tableau, ce qui est le cas ici. Le peu de différences pour les deux autres cases relève sans doutes également d'un nombre trop faible d'instance de jeu lancées (1000 pour le cas présent).

Cas général :

Questions facultatives) Bien que nous n'ayons pas eu le temps d'implémenter complétement le jeu simultané sur plus d'un tour, nous avons établis l'expression de $EG_1()$. Il s'agit de la même idée qu'à la question 4 à quelques détails près. En effet J1 va chercher à maximiser son espérance mais J2 cherchera en même temps à miniser cette même espérance. On a donc une expression max d'un min qui prend la valeur dépendant des valeurs des espérance $EG_1()$ aux états (i,j) qui sont atteignables.

Par ailleur, le jeu étant ici symétrique, on donc que la matrice EG_1 que l'on obtient est symétrique et que $EG_1(i,i) = 0$ et ce $\forall i \in \mathbb{N}^*$.

A noter également qu'ici $EG_1()$ reprend la notation du sujet mais n'est pas la même matrice que celle calculée à la question 10) qui dépendait elle des valeurs d1 et d2 le nombre de dés que lancent J1 et J2.

$$\begin{cases} EG_1(i,j) = 0 & \text{si } i \geq N \text{ et } i = j \\ EG_1(i,j) = 1 & \text{si } i \geq N \text{ et } j < N \\ EG_1(i,j) = -1 & \text{si } i < N \text{ et } j \geq N \\ EG_1(i,j) = \max\limits_{d1\in[\![1,D]\!]}(\min\limits_{d2\in[\![1,D]\!]}(\sum\limits_{k=1}^{6*d1}\sum\limits_{l=1}^{6*d2} EG_1(i+k,j+l)*P(d1,k)*P(d2,l)) & \text{si } i < N \text{ et } j < N \end{cases}$$

Conclusion

Si l'on décide de jouer avec la variante séquentielle, la programation dynamique nous permet d'affirmer qu'on a tout intéret de jouer la stratégie Optimale peu importe qu'on joue en premier ou en deuxième.

Pour la variante simultannée nous avons manqué de temps pour implémenter le jeu sur plusieurs tours avec le cas général. Malgré nos calculs nous n'avons pas pu mettre en exergue une stratégie vraiment suppérieure à une autre.

On peut s'interroger sur la formule de récurrence utilisée pour l'espérance du Joueur 1 dans l'état (i, j) et si une autre implémentation ne permettrait pas de l'augmenter. Différentes actions pourraient venir influer sur le jeu aussi comme le fait de pouvoir changer de stratégies entre les tours, comme implémenter dans notre fichier python, et donc de pouvoir choisir d'être averse au risque ou l'inverse en fonction de notre situation.