Un jeu de Nim

Année 2016- 2017

Robin JOURDE (1^{e}S) et Rafaël CATTIN (1^{e}S)

Établissement : Lycée Douanier Rousseau (Laval)

Encadrés par Stéphanie CHANCEREL et Anne DUVAL

Chercheur : François DUCROT, Université d'Angers

Présentation du sujet

Les jeux de Nim sont des jeux de stratégie relativement courants se jouant à 2. Ces jeux se jouent avec des bâtons, des billes, des jetons, des allumettes ou tout autre objet de ce type. Il en existe d'innombrables versions mais ... nous allons nous attarder sur l'une d'elles.

Dans notre version, chaque joueur doit prendre dans un tas d'un certain nombre de bâtons tour à tour. Le premier joueur peut en prendre autant qu'il le veut mais doit en laisser au moins un. Puis, chacun peut prendre le nombre de bâtons au plus égal au double du nombre de bâtons pris par le précédent.
Par exemple, si le joueur précédent a pris 7 bâtons, le suivant peut en prendre 14 au maximum.
Celui qui gagne est celui qui prend le dernier bâton.

On cherche à savoir s'il existe une méthode pour gagner à tous les coups, quel joueur gagne dans quel cas, etc. On considérera pour notre analyse que les deux joueurs jouent pour le mieux.

Sommaire

1/ CONSTATS EXPÉRIMENTAUX

Après des tests au cours desquels nous regardions les meilleurs coups à jouer et qui gagne (A ou B) en fonction du nombre de bâtons N (de 2 à 31), nous avons remarqué certaines choses.

A- Loi du tiers :

Nous avons premièrement remarqué que, **lorsqu'un joueur prend un nombre de bâtons supérieur à un tiers du total actuel, le joueur suivant peut prendre tout le reste.**
Par exemple, si, alors qu'il y a 28 bâtons et que le joueur précédent prend 11 bâtons, je peux prendre jusqu'à 22 bâtons, soit tout le reste (il reste 28 - 11 = 17 bâtons).

Démonstration:

Il y a N bâtons. Le premier joueur prend A bâtons. A est supérieur ou égal à un tiers de N.

$$\begin{aligned} A &\geq N/3 \\ -A &\leq -N/3 \\ N - A &\leq N - N/3 \\ N - A &\leq 2 \times N/3 \end{aligned}$$

N-A est le nombre de bâtons restants. Il est inférieur au double de N/3, soit ce qu'a pris le premier joueur. Le deuxième joueur peut donc prendre N-A bâtons et gagner.

B- Suite de Fibonacci

Nous avons remarqué que A gagne dans la plupart des cas, sauf lorsque N est égal à 2, 3, 5, 8, 13, 21,… soit des nombres de la suite de Fibonacci

La suite de Fibonacci est une suite de nombres entiers dont les deux termes initiaux sont 0 et 1. Chaque terme est égal à la somme des deux termes précédents. La suite débute donc par :

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597...

On la notera F. F_i est le terme de rang i dans F (le premier terme est au rang 0).
Par exemple : $F_0 = 0$; $F_9 = 34$; $F_{15} = 610$; …

C- Annonce des résultats

Nous avons également constaté que

- Si le premier joueur est devant un nombre de Fibonacci, alors il perdra forcément

- Si le premier joueur n'est pas devant un nombre de Fibonacci, alors il gagnera à coup sûr (1).

La stratégie pour gagner quand on n'est pas devant un nombre de Fibonacci est décrite par l'algorithme ci dessous :

Si le joueur peut prendre tout le paquet, il le fait (cf "A-Loi du tiers") (2)

Sinon :

- Trouver le plus grand terme de F inférieur à *N*.
- Faire la différence entre ce terme et *N*.
- Recommencer ces deux dernières étapes jusqu'à ce que la différence soit égale à un membre de F.
- Prendre ce nombre de bâtons.

2/ PREUVE DU FONCTIONNEMENT DE CETTE MÉTHODE

A- Décomposition d'un nombre en somme de nombres de Fibonacci

1. Explication

On peut noter que l'algorithme proposé aboutit à une décomposition de N comme une somme de nombres de Fibonacci, décroissants et de taille maximale :

$N = F_i + \ldots + F_k$ avec $F_i > \ldots > F_k$, soit $i > \ldots > k$ et $F_i, \ldots , F_k$ les plus grands possible (3)

OU

On pose : f_i est un nombre qui appartient à F, placé en $i^{ème}$ position dans la décomposition.

$N = f_1 + \ldots + f_j$ avec $f_1 > \ldots > f_j$ et $f_1, \ldots , f_j$ les plus grands possible

On a remarqué également que, lorsque $N \notin F$, les termes de cette décomposition ne sont pas consécutifs dans F.

On aboutit alors à une proposition qui définit cette décomposition :

Proposition : « Tout nombre entier positif se décompose en somme de nombres de Fibonacci non consécutifs. »

Remarque : Termes non consécutifs dans F
On appelle deux nombres « consécutifs dans F » deux nombres qui se suivent dans la suite de Fibonacci.
Exemples : F = { 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55 ... }
1 et 2 sont consécutifs ; 21 et 34 sont consécutifs ; 3 et 8 ne sont pas consécutifs …

2. Preuve

Cas n°1 : si $N \in F$

$N = F_i$ ou $N = f_0$

Si N est un terme de F on dira que la proposition est vérifiée, bien que ça n'ait pas beaucoup de sens de parler d'une somme ici.

Cas n°2 : si $N \notin F$

Pour le prouver, on utilise un raisonnement par récurrence.

Notre propriété est la suivante : P_N : « Tout nombre entier inférieur ou égal à N se décompose en somme de nombres de Fibonacci non consécutifs. »

Si N = 4, N = 3 + 1

P_N est donc vraie pour N = 4 car 4 est le plus petit entier naturel qui n'appartient pas à F.

On suppose que P_N est vraie et on cherche à prouver P_{N+1} . On cherche donc à décomposer N+1.

Il existe $f_0 \in F$ tel que f_0 soit le plus grand nombre de Fibonacci inférieur ou égal à N+1.

$N+1 = f_0 + N'$ avec $N' \leq f_0$, et donc $N' = N + 1 - f_0 < N$ (4)

Comme N'<N, d'après l'hypothèse de récurrence, N' se décompose en somme de nombres de Fibonacci non consécutifs.:

$N' = f_1 + \ldots + f_j$ avec $f_1 > \ldots > f_j$ et $f_1, \ldots, f_j$ les plus grands possible et non consécutifs dans F

$N+1 = f_0 + f_1 + \ldots + f_j$

N+1 peut donc se décomposer en une suite de nombres de Fibonacci, il reste à prouver que f_0 et f_1 ne sont pas consécutifs dans F.

On effectue une démonstration par l'absurde : on considère que f_0 et f_1 sont consécutifs et on cherche une contradiction.

Si f_0 et f_1 sont consécutifs, alors $f_0 + f_1 = f'$, et $f_0 < f'$

Soit :

$$N+1 = f_0 + f_1 + \ldots + f_j$$
$$N+1 \geq f_0 + f_1$$
$$N+1 \geq f'$$

Il existe alors un nombre de Fibonacci, f', inférieur ou égal à N+1 et supérieur à f_0
Or on a dit que : "Il existe $f_0 \in F$ tel que f_0 soit le plus grand nombre de Fibonacci inférieur ou égal à N+1."

Il y a une contradiction, f_0 et f_1 ne sont donc pas consécutifs.

Donc, notre propriété P_N est toujours vraie pour N+1, P_N est donc vraie pour tout N :

Pour tout N : P_N : « Tout nombre entier inférieur ou égal à N se se décompose en somme de nombres de Fibonacci non consécutifs. »

Soit : **Propriété : « Tout nombre entier positif se décompose en somme de nombres de Fibonacci non consécutifs. »**

Notre proposition étant démontrée, on a une propriété.

On appellera par la suite cette décomposition "Décomposition selon Fibonacci", abrégé en *DFibo*.

3. Le théorème de Zeckendorf

"Le théorème de Zeckendorf garantit que tout entier positif N peut être représenté, de manière unique, comme somme de nombres de Fibonacci distincts et non consécutifs. Cette représentation est appelée la représentation de Zeckendorf de N."
Source : Wikipedia (https://fr.wikipedia.org/wiki/Théorème_de_Zeckendorf) (5)

Notre décomposition est donc une application de ce théorème, démontré par le mathématicien belge Edouard Zeckendorf en 1972. Elle correspond à la représentation de Zeckendorf.

B- Coups de B

Pour $N \notin F$:

D'après la preuve précédente, on sait que si $N \notin F$:

$N = F_i + \ldots + F_{k'} + F_k$ avec $k' \geq k+2$ (on sait que $F_{k+2} = F_{k+1} + F_k$)

Soit :

$$\begin{aligned} F_{k'} &\geq F_{k+2} \\ F_{k+1} &> F_k \\ F_{k+1} + F_k &> 2\,F_k \\ F_{k+2} &> 2\,F_k \\ \mathbf{F_{k'} \geq F_{k+2}} &\mathbf{>} \mathbf{2\,F_k} \end{aligned}$$

Lorsque A prend F_k bâtons, B ne peut pas prendre $F_{k'}$ bâtons ou plus, soit le terme précédent de *DFibo*.

Pour $N \in F$:

$N = F_i = F_{i-1} + F_{i-2}$

Soit :

$$\begin{aligned} F_{i-3} &\leq F_{i-2} \\ F_{i-2} + F_{i-3} &\leq 2\,F_{i-2} \\ \mathbf{F_{i-1}} &\mathbf{\leq} \mathbf{2\,F_{i-2}} \end{aligned}$$

Lorsque A prend F_{i-2} bâtons, B peut prendre F_{i-1} bâtons (cf. Loi du tiers), et ainsi prendre tout ce qui reste et gagner.

C- Coups de A

On a constaté que si au début de la partie $N \notin F$, A pourra toujours prendre le dernier terme de *DFibo*, quel que soit le coup précédent

La preuve n'a pas été réalisée.

On considérera donc : "**A peut toujours prendre le dernier terme de *DFibo***"

D- Nombre de termes de la décomposition selon Fibonacci

On montre l'évolution du nombre de termes de *DFibo* lorsque le nombre de bâtons pris est inférieur au dernier terme de *DFibo*.

$N = f_1 + \dots + f_{j-1} + f_j$ avec j termes

N' est le nombre de bâtons restants après le coup.
C est le nombre de bâtons pris lors du coup, avec $C < f_j$

$N' = N - C = f_1 + \dots + f_{j-1} + f_j - C$

Comme $C < f_j$, $f_j - C$ est un entier positif qui peut se décomposer selon *DFibo*.

On remarque aussi que la première partie de *DFibo* (de f_1 à f_{j-1}) est identique (6).

La décomposition *DFibo* de N' comportera donc au moins (j-1) + 1 = j termes, soit autant ou plus que N.

Donc : **Lorsque le nombre de bâtons pris est inférieur au dernier terme de *DFibo*, la décomposition *DFibo* du nombre de bâtons restants comportera autant ou plus de termes que la décomposition précédente.**

E- Fin de la preuve

Si N∉F :

$N = f_1 + \dots + f_j + f_{j+1}$ avec j+1 termes

Le joueur A prend f_{j+1} bâtons. Il reste j termes.

Son adversaire B ne pourra pas prendre f_j bâtons ou plus. (cf. Coups de B)
Il devra donc en prendre moins. f_j sera donc remplacé par un plusieurs termes. Le nombre de termes n'est pas diminué. (cf. Nombre de termes de la décomposition selon Fibonacci)

Dans ce cas, le joueur A peut toujours prendre le dernier terme de *DFibo*. (cf. Coups de A) (7)

Le joueur A est donc le seul à pouvoir faire diminuer le nombre de termes.

A est donc le seul joueur à pouvoir prendre le dernier terme, et donc gagner.

Si N∈F :

$N = f_0 = f_1 + f_2$
ou
$N = F_i = F_{i-1} + F_{i-2}$

Le joueur A ne peut pas prendre F_{i-2} bâtons ou plus. (cf. Coups de B)

Le nombre de termes ne peut donc pas être diminué. (cf. Nombre de termes de la décomposition selon Fibonacci)

Les rôles sont inversés par rapport à l'autre cas : B doit jouer avec N∉F. Donc B peut prendre le dernier terme de *DFibo* du nombre de bâtons restants. (cf. Coups de A, rôles inversés)

Dans ce cas, B gagne toujours.

3/ PISTE DE GÉNÉRALISATION

On a ensuite cherché ce qu'il se passait quand le nombre de bâtons pris par un joueur doit être non pas inférieur ou égal à 2 fois le nombre de bâtons pris par le joueur précédent mais inférieur ou égal à *p* fois ce nombre pour différentes valeurs de *p*.

On trouve alors d'autres suites qui se comportent comme F. C'est à dire que, pour chaque valeur de p et donc chaque suite, l'algorithme fonctionne et que tout nombre peut se décomposer en une somme de terme de ces suites non consécutifs. (Ceci est pour l'instant une proposition) (8)

Ces suites (termes initiaux et expression par récurrence) ont été trouvées expérimentalement.

On a défini ces suites (u^p) ainsi :

u^p_0 ; … ; u^p_{q-1} *termes initiaux (trouvés expérimentalement)*

$u^p_i = u^p_{i-1} + u^p_{i-q}$ *expression de la suite par récurrence*

Exemples :

Pour p = 2 (suite de Fibonacci)

$u^2_o = 0$; $u^2_1 = 1$

$u^2_i = u^2_{i-1} + u^2_{i-2}$

OU

$F_o = 0$; $F_1 = 1$

$F_i = F_{i-1} + F_{i-2}$

Pour p = 3

$u^3_o = 0$; $u^3_1 = 1$; $u^3_2 = 0$; $u^3_3 = 1$

$u^3_i = u^3_{i-1} + u^3_{i-4}$

Notes d'édition

(1) Rappelons qu'on considère que les deux joueurs jouent au mieux.

(2) Cette stratégie s'applique à chaque coup au joueur qui peut gagner (*N* désigne le nombre de bâtons restant avant qu'il joue), à condition qu'il la suive depuis le début. Elle suppose qu'il pourra alors effectivement prendre le nombre de bâtons calculé par l'algorithme (voir plus loin).

(3) Plus précisément, F_i est le plus grand nombre de Fibonacci ≤ N, le deuxième terme est le plus grand nombre de Fibonacci ≤ $N – F_i$, et ainsi de suite.

(4) On suppose ici $N \geq 4$, $N+1 \geq 5$, donc f_0 est au moins égal à 5.

(5) En plus de ce qui est montré ici, le théorème de Zeckendorf garantit l'unicité de la décomposition, à condition d'exclure $F_0=0$ et $F_1=1$ (car on a aussi $F_2=1$). Pour une démonstration, voir l'article cité de *Wikipedia*.

(6) Plus précisément, si la décomposition de $f_j – C$ est

$$f_j – C = f'_j + \dots + f'_k$$

et celle de N', $N' = N - C = f_1 + \dots + f_{j-1} + f'_j + \dots + f'_k$,
on a $f'_j < f_j$, et $f_j – f'_j$ est supérieur ou égal au nombre précédant f'_j dans la suite de Fibonacci, lui-même strictement supérieur à f'_{j+1} ; donc $(f_j – f'_j) – f'_{j+1}$ est supérieur ou égal au nombre de Fibonacci précédant f'_{j+1}.
En continuant de même de proche en proche, on obtient

$$f_j – (f'_j + \dots + f'_k) = C \geq f',$$

où f' est le nombre précédant f'_k dans la suite de Fibonacci.
Mais on a vu (coups de B, pour $N \in F$) qu'un nombre de Fibonacci est toujours inférieur ou égal à 2 fois son prédécesseur.
Donc $f'_k \leq 2f' \leq 2C$, et l'autre joueur pourra bien prendre f'_k bâtons, c'est-à-dire le dernier terme de la décomposition de N'.

(7) Voir la note précédente. Une fois que A a pris le dernier terme de la décomposition, B doit prendre un nombre strictement inférieur au terme précédent, A peut de nouveau prendre le dernier terme de la nouvelle décomposition, et ainsi de suite…
Lorsqu'au départ $N \in F$, A doit prendre un nombre de bâtons strictement inférieur à N, et c'est B qui pourra appliquer la stratégie gagnante, A étant toujours contraint à prendre un nombre de bâtons strictement inférieur au dernier terme de la décomposition.

(8) Le relecteur comprend qu'il s'agit ici d'une simple piste de travail. Pour ces suites, on pourra bien trouver des décompositions des nombres sans termes consécutifs, et même avec plus de contraintes. Mais il n'est pas du tout certain que l'algorithme trouvé pour p = 2 fonctionne dans les autres cas, ni même que la suite donne tous les nombres de bâtons où le joueur A doit perdre.
Dans les exemples qui suivent, il manque aussi le choix de l'entier q, qui est pris égal à 2 pour p = 2 et égal à 4 pour p = 3.