ADS Informatique

Jeux combinatoires

Source : Éric Duchêne et Aline Parreau

Informatique-Mathématique. Une photographie en 2017, EJCIM 2017

Travail demandé

Il vous est demandé d'étudier puis de présenter le texte joint à travers un exposé de synthèse d'une durée comprise entre 15 et 20 minutes.

Si l'étude de la totalité du dossier et la préparation d'un exposé cohérent dans la durée impartie ne vous paraît pas possible, vous pouvez décider de vous limiter à une partie du dossier.

Remarques générales

1. Les textes proposés, quelle que soit leur origine, peuvent présenter des défauts (coquilles typographiques, négligences ou sous-entendus de l'auteur, voire erreurs...) qui, sauf exception, n'ont pas été corrigés.

2. Les textes proposés peuvent contenir des exercices qu'il n'est pas demandé de résoudre. Néanmoins, vous pouvez vous aider des énoncés de ces exercices pour enrichir votre exposé.

2.1 Introduction et premières briques

2.1.1 Jeu combinatoire : définition

Les jeux combinatoires — au sens large — sont des objets familiers pour beaucoup d'entre nous. Parmi les jeux les plus connus, on peut citer le jeu de Nim, le Go, les échecs ou encore les dames. La théorie mathématique qui permet d'envisager leur résolution est toujours en cours de construction aujourd'hui. Dans certains cas (jeux en convention normale), un cadre de résolution théorique a été construit par Conway à la fin des années 1970. C'est celui-ci que nous détaillerons dans ce cours. Dans d'autres cas (jeux avec boucles ou jeux à score par exemple), la compréhension de ces jeux s'avère un problème d'actualité pour la communauté scientifique concernée.

Une définition d'un jeu combinatoire (au sens strict du terme) a été donnée pour la première fois par Berlekamp, Conway et Guy dans leur ouvrage référence [9]. En voici les ingrédients :

- Il y a exactement deux joueurs, appelés "Left" et "Right", qui jouent à tour de rôle, sans passer leur tour ;
- l'information est totale : tous les coups possibles sont connus par les deux joueurs ;
- le hasard n'intervient pas dans les règles du jeu (sous forme de lancers de dés ou de tirages de cartes par exemple) ;
- les règles sont définies de sorte que le jeu ait toujours une fin, et le nombre de positions de jeu est finie ;
- le vainqueur est désigné par le dernier coup joué : en *convention normale*, le premier joueur qui ne peut plus jouer a perdu. C'est l'inverse en *convention misère*.

Le dernier point implique qu'il y a toujours un gagnant dans un jeu combinatoire. Nous verrons un peu plus loin qu'une définition plus formelle d'un jeu peut-être donnée en tant que jeu de déplacement sur un arbre.

Si l'on se réfère aux éléments ci-dessus, les conditions requises sont alors trop fortes pour que les jeux abstraits à deux joueurs les plus célèbres soient combinatoires. Par exemple, les échecs et les dames peuvent avoir des parties nulles. Cela provient du fait que dans ces jeux, certaines positions peuvent être visitées plusieurs fois au cours de la partie. De tels jeux sont appelés *loopy*. Au Go, à Dots and Boxes ou à Othello, le vainqueur est déterminé par un score et non en fonction du dernier coup joué. Cependant, tous ces jeux restent proches des jeux combinatoires. On peut trouver dans la littérature des éléments de réponse pour les résoudre (voir [122], chap. 6 pour les jeux loopy et [76] pour les jeux à score). Si certains des concepts et problématiques présentés dans ce cours restent pertinents dans ces contextes étendus, nous ne développerons toutefois pas les résultats qui leur sont spécifiques. De même, la théorie des jeux en convention misère ne sera pas abordée ici (très différente de celle des jeux en convention normale).

Pour le lecteur qui souhaite accéder à un état de l'art assez complet sur le domaine, les trois livres suivants remportent les suffrages de la communauté : *Winnning Ways* [9], *Lessons in Play* [1] et *Combinatorial Game Theory* [122]. Le premier cité est l'œuvre pionnière qui fait suite aux travaux de Conway. Constitué de quatre volumes, il contient un vaste panel de jeux avec résolution à l'appui. Le deuxième est un ouvrage très clair et pédagogique, utile notamment pour faire découvrir la théorie de façon ludique et bien organisée. Le dernier livre, plus récent et plus formel, couvre avec un spectre très large les grands résultats du domaine.

2.1.2 Deux exemples fil rouge : NIM et DOMINEERING

Il est temps désormais de donner deux exemples de "vrais" jeux combinatoires qui serviront de fil rouge à ce cours. Il s'agit du jeu de NIM [13] et de DOMINEERING [37]. Pour le premier, les positions de jeu sont des n-uplets d'entiers positifs ou nuls qu'on notera $(a_1, \ldots, a_n)$. Un coup consiste à soustraire à l'une des composantes a_i (avec $a_i > 0$) une valeur comprise entre 1 et a_i. Le joueur qui ne peut plus trouver un tel a_i perd la partie. En d'autres termes, celui qui joue vers la position $(0, \ldots, 0)$ est le vainqueur.

Le jeu DOMINEERING se joue quant à lui sur une grille rectangulaire. À tour de rôle, les deux joueurs placent un domino sur la grille selon les contraintes suivantes : Left ne peut placer ses dominos que verticalement, et Right qu'horizontalement. Ici encore nous jouons en convention normale et le premier joueur qui ne peut plus placer de domino perd la partie. La figure 2.1 illustre une position de jeu où Left a commencé et gagne, puisque Right ne peut plus placer aucun domino horizontal sur la grille.

2.1.3 Modélisation d'un jeu

Une première propriété très utile se dégage de la définition d'un jeu : tout jeu combinatoire peut se jouer sur un certain arbre fini décrit ci-dessous.

Définition 2.1.1 (Arbre de jeu). *Étant donné un jeu $\mathcal{G}$ ayant comme position de départ S, l'*arbre de jeu *associé à $\mathcal{G}$ est un arbre semi-ordonné (c'est-à-dire avec deux types de fils — gauches et droits) défini de la façon suivante :*

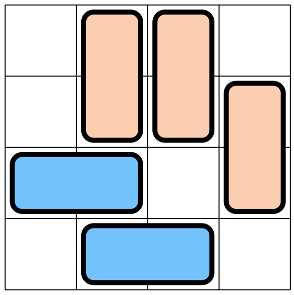


FIGURE 2.1 – *Une partie de* DOMINEERING : *Right est bloqué et perd.*

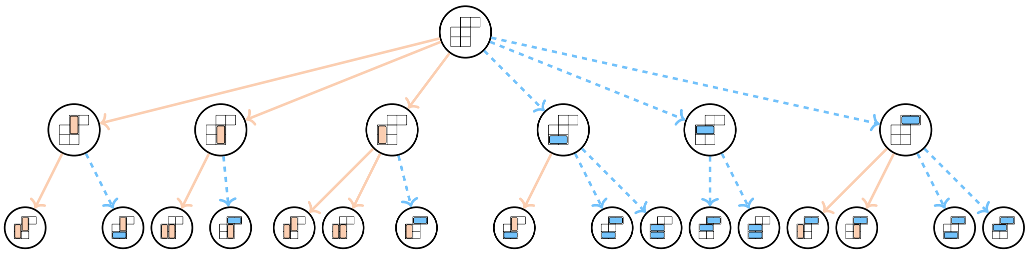
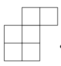


FIGURE 2.2 – *Arbre de jeu d'une position de* DOMINEERING

— *On construit un sommet racine qui correspond à la position de départ S ;*
— *on construit les fils de ce sommet de la façon suivante : toutes les positions de jeu atteignables par Left (resp. Right) en un coup sont les fils gauches (resp. droits) de la racine ;*
— *on applique la règle précédente récursivement pour tous les fils nouvellement créés.*

La figure 2.2 donne l'arbre de jeu de DOMINEERING avec comme position de départ . On notera que seuls les trois premiers niveaux de l'arbre ont été dessinés (l'arbre complet admet un niveau supplémentaire).

Jouer à un jeu sur son arbre consiste à déplacer à tour de rôle un jeton situé initialement à la racine. Chaque joueur le déplace le long d'un arc qui lui correspond (arcs orientés vers la gauche pour Left, et vers la droite pour Right). En convention normale, le premier joueur qui ne peut plus déplacer le jeton perd la partie.

Enfin, définissons maintenant la notion d'*option* qui correspond aux fils d'un sommet de l'arbre.

Définition 2.1.2 (Option)**.** *Pour une position de jeu donnée G, on appellera* options *de G l'ensemble des positions de jeu atteignables en un coup. Celles-ci peuvent être partitionnées en deux sous-ensembles : les options gauches (positions atteignables par Left) et les options droites (positions atteignables par Right).*

2.1.4 Problématiques en théorie des jeux combinatoires

Étant donné un jeu, les chercheurs en théorie des jeux combinatoires s'intéressent à trois problématiques majeures :

- Si l'on suppose que les deux joueurs jouent au mieux à chaque tour, qui sera le vainqueur ?
- quelle est la valeur du jeu ? Cette notion sera définie dans la partie 2.2 et constitue le cœur de la théorie ;
- peut-on construire une stratégie gagnante pour le vainqueur ? (c'est-à-dire une séquence de coups qui amène à la victoire, quels que soient les coups de son adversaire).

Grâce à l'arbre de jeu (dont la taille est finie), il n'est pas difficile de voir que ces problèmes sont tous décidables. C'est donc la complexité algorithmique de leur résolution qui va nous intéresser. Avant de rentrer dans plus de formalisme, abordons-les à travers quelques exemples.

Prenons le jeu de NIM avec $n = 2$. On peut remarquer que les positions de jeu de la forme (a_1, a_2) avec $a_1 \neq a_2$ sont gagnantes pour le joueur qui commence. La stratégie gagnante consiste à rendre les deux valeurs égales. L'autre joueur n'aura pas d'autre choix que de les rendre différentes, et ainsi de suite jusqu'à la position finale $(0, 0)$. Lorsque $n > 2$, nous verrons plus tard que les problèmes restent solubles en temps polynomial, bien que la résolution soit moins immédiate.

Il existe d'autres jeux pour lesquels on sait dire qui gagne, mais sans pour autant connaître une stratégie gagnante en temps raisonnable : c'est le cas par exemple de CHOMP ou de HEX. Dans le cas de DOMINEERING sur un grille $n \times m$, la complexité de ces trois problèmes est ouverte.

Nous donnons par la suite quelques éléments supplémentaires à propos des problèmes décrits précédemment.

2.1.5 Recherche de l'issue du jeu

L'*issue* d'un jeu combinatoire correspond au problème de la détermination du vainqueur. Celle-ci ne peut prendre que quatre valeurs possibles :

- $\mathcal{L}$ si Left admet une stratégie gagnante indépendamment de qui commence ;
- $\mathcal{R}$ si Right admet une stratégie gagnante indépendamment de qui commence ;
- $\mathcal{N}$ si le premier joueur admet une stratégie gagnante ;
- $\mathcal{P}$ si le deuxième joueur admet une stratégie gagnante.

Étant donné un jeu G, on notera par la suite $o(G)$ l'issue de G. Le fait que seulement quatre issues soient possibles se déduit directement de l'arbre de jeu en étiquetant les sommets des feuilles jusqu'à la racine. Toutes les feuilles sont initialement étiquetées $\mathcal{P}$ car celui qui doit jouer depuis une feuille est bloqué et donc déclaré perdant (en convention normale uniquement). Puis on étiquette récursivement les sommets des niveaux précédents en utilisant la proposition suivante :

Proposition 2.1.3. *Un nœud d'un arbre de jeu sera étiqueté :*
- *$\mathcal{P}$ si ses fils gauches sont tous $\mathcal{N}$ ou $\mathcal{R}$ et ses fils droits sont tous $\mathcal{N}$ ou $\mathcal{L}$;*
- *$\mathcal{N}$ si au moins un fils gauche est $\mathcal{L}$ ou $\mathcal{P}$ et au moins un fils droit est $\mathcal{R}$ ou $\mathcal{P}$;*
- *$\mathcal{L}$ si au moins un fils gauche est $\mathcal{L}$ ou $\mathcal{P}$ et ses fils droits sont tous $\mathcal{N}$ ou $\mathcal{L}$;*
- *$\mathcal{R}$ si ses fils gauches sont tous $\mathcal{N}$ ou $\mathcal{R}$ et au moins un fils droit est $\mathcal{R}$ ou $\mathcal{P}$.*

Cet algorithme permet donc de calculer l'issue d'un jeu en temps polynomial en fonction de la taille de l'arbre. Cependant, la taille d'une position de jeu est souvent bien plus petite que la taille de l'arbre correspondant. Par exemple, une position $(a_1, \ldots, a_n)$ du jeu de NIM (avec $a_i > 0$ pour tout i) admet un codage de taille $\mathcal{O}(\sum_{i=1}^{n} \log_2(a_i))$, ce qui est bien plus faible que le nombre de sommets dans l'arbre de jeu (de l'ordre de $\Pi_{i=1}^{n} a_i$). Par conséquent, la construction de l'arbre de jeu n'est souvent pas la solution pour calculer en temps polynomial l'issue d'un jeu. D'autres caractérisations doivent alors être envisagées. Dans le cas de NIM, il a été prouvé [13] qu'une position $(a_1, \ldots, a_n)$ est $\mathcal{P}$ si et seulement si $a_1 \oplus \cdots \oplus a_n = 0$, où $\oplus$ est l'opérateur d'addition bit à bit sans retenue (XOR), qu'on appellera *Nim-somme*. Toutes les autres positions sont $\mathcal{N}$. Donnons quelques éléments de preuve de ce résultat :

Théorème 2.1.4. *Une position $(a_1, \ldots, a_n)$ de* NIM *est $\mathcal{P}$ si et seulement si $a_1 \oplus \cdots \oplus a_n = 0$.*

Preuve. La seule position finale du jeu est la position nulle $(0, \ldots, 0)$, qui est trivialement $\mathcal{P}$, et vérifie bien la propriété de Nim-somme nulle. Considérons maintenant une position non nulle quelconque $(a_1, \ldots, a_n)$ telle que $a_1 \oplus \cdots \oplus a_n = 0$. D'après la proposition 2.1.3, montrons que tout coup depuis cette position produit une position où la Nim-somme est non-nulle. Le joueur qui joue à partir de cette position donne une position $(a'_1, \ldots, a'_n)$, où tous les a'_i sont égaux aux a_i sauf exactement pour une valeur a'_i telle que $a'_i < a_i$. Par conséquent, cette position vérifie bien $a'_1 \oplus \cdots \oplus a'_n \neq 0$.

Il nous reste alors à montrer que pour une position $(a_1, \ldots, a_n)$ de Nim-somme non nulle, il existe un coup vers une position de Nim-somme nulle. Soit $S = a_1 \oplus \cdots \oplus a_n \neq 0$, et notons $\{i_1, \ldots, i_k\}$ l'ensemble des indices des bits de S égaux à 1. Cet ensemble est non vide, et sans restreindre

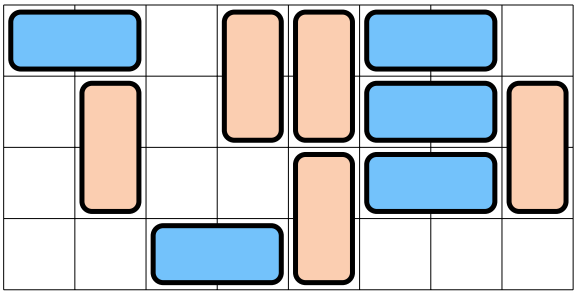


FIGURE 2.3 – *Somme de positions de* DOMINEERING

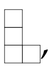
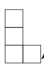
la généralité, supposons que i_1 est l'indice du bit de poids le plus fort. Par définition de S, il existe nécessairement une composante a_i dont le bit d'indice i_1 vaut 1. Il suffit alors de jouer dans cette composante a_i vers un certain a_i' obtenu en mettant tous les bits de a_i d'indices $\{i_1, \ldots, i_k\}$ à leur opposé. On a bien $a_i' < a_i$ et $a_1 \oplus \cdots \oplus a_{i-1} \oplus a_i' \oplus a_{i+1} \cdots \oplus a_n = 0$. ■

Ainsi, il n'existe aucune position $\mathcal{L}$ ou $\mathcal{R}$ au jeu de NIM. C'est d'ailleurs le cas pour tous les jeux dits *impartiaux*. Ce sont des jeux dont l'arbre est symétrique : tout coup disponible pour Left est aussi disponible pour Right, et réciproquement. Les jeux qui ne satisfont pas cette propriété (comme DOMINEERING) sont dits *partisans*.

Pour résumer la problématique de l'issue, on cherche à déterminer la complexité algorithmique pour décider si un jeu donné G est $\mathcal{P}$, $\mathcal{R}$, $\mathcal{L}$, ou $\mathcal{N}$. Dans le cas de DOMINEERING, on a vu précédemment que cette complexité n'était toujours pas connue. On connaît toutefois les issues pour certaines instances particulières (grilles avec un côté pair, grilles $5 \times n$, ou grilles carrées jusqu'au 10×10).

2.1.6 Somme de jeux

Si le calcul de l'issue est souvent un problème de haute complexité (PSPACE-difficile pour beaucoup de jeux, comme nous le verrons dans le chapitre 2.3), cette complexité peut parfois être moins élevée pour les jeux dont les positions peuvent se décomposer en une union de sous-jeux indépendants. Prenons par exemple la position de DOMINEERING de la figure 2.3.

Cette position peut être vue comme l'union (on parlera plutôt de *somme*) des trois composantes , , et . Jouer à une somme de deux jeux G_1 et G_2 revient alors à jouer à chaque tour soit sur G_1, soit sur G_2. La partie s'arrête quand les deux jeux sont terminés. Notons que cette définition a du sens même si les deux jeux ont des règles différentes. Par exemple, on peut considérer la somme $G_1 + G_2$ où G_1 est une position de NIM $(a_1, \ldots, a_n)$ et G_2 une position de DOMINEERING.

L'intérêt principal d'une décomposition en somme de sous-jeux est la détermination de l'issue sachant l'issue de chacun des sous-jeux. On peut par exemple assez facilement prouver la propriété suivante :

Proposition 2.1.5. *Étant donné un jeu G tel que* $o(G) = \mathcal{P}$ *, alors pour tout jeu* H*, on a* $o(G+H) = o(H)$.

Preuve. Sans restreindre la généralité, supposons que Left a une stratégie gagnante sur H. Alors sur $G+H$, Left applique exactement sa stratégie gagnante sur H, et à chaque fois que Right joue sur G, il répond aussi sur G en appliquant sa stratégie gagnante en tant que second joueur sur G. Ainsi, Left est certain de jouer le dernier coup à la fois sur G et H, et donc sur $G+H$. ∎

Autrement dit, toutes les composantes $\mathcal{P}$ d'une somme de jeu n'influent pas sur l'issue globale du jeu. Malheureusement, la connaissance des issues des termes d'une somme de jeux est souvent insuffisante. Le tableau 2.1 résume ce qu'on peut conclure :

+	$\mathcal{P}$	$\mathcal{N}$	$\mathcal{L}$	$\mathcal{R}$
$\mathcal{P}$	$\mathcal{P}$	$\mathcal{N}$	$\mathcal{L}$	$\mathcal{R}$
$\mathcal{N}$	$\mathcal{N}$	?	$\mathcal{N}$ ou $\mathcal{L}$	$\mathcal{N}$ ou $\mathcal{R}$
$\mathcal{L}$	$\mathcal{L}$	$\mathcal{L}$ ou $\mathcal{N}$	$\mathcal{L}$	?
$\mathcal{R}$	$\mathcal{R}$	$\mathcal{R}$ ou $\mathcal{N}$	?	$\mathcal{R}$

TABLE 2.1 – *Issues des sommes de jeux*

Ainsi, la somme de deux positions $\mathcal{L}$ et $\mathcal{R}$ peut être $\mathcal{P}$, $\mathcal{N}$, $\mathcal{L}$ ou $\mathcal{R}$ selon les cas. Dans de telles situations (comme celle de la figure 2.3 qui est une somme de type $\mathcal{L} + \mathcal{L} + \mathcal{R}$), la seule notion d'issue ne suffit pas et la caractérisation doit être affinée. Conway [21] a alors réussi à généraliser la notion d'issue en attribuant une valeur à chaque jeu. Cette valeur permettra notamment de mieux appréhender le traitement des sommes. Nous allons voir dans le chapitre suivant comment ces valeurs sont construites.

2.2 Les valeurs d'un jeu combinatoire

Nous donnons dans ce chapitre un aperçu de la théorie de Conway qui associe une valeur à tout jeu combinatoire. La plupart des résultats énoncés ci-dessous sont accessibles (avec leurs preuves) dans [122].

2.2.1 Relation d'équivalence entre jeux

Dans ce qui suit, on assimilera un jeu à une position particulière d'un jeu combinatoire. NIM et DOMINEERING seront quant à eux appelés des ensembles de règles. Par conséquent, une position $(1, 2, 5)$ de NIM est un jeu. On passe d'un jeu à un autre en jouant un coup. Le fait de considérer les positions de jeu comme les éléments atomiques de la théorie permettra notamment d'accéder à la clôture de l'ensemble des jeux pour la somme.

Comme nous l'avons vu précédemment, un jeu peut être représenté par un arbre dont il sera la racine. Ainsi, on peut le définir récursivement par la seule donnée de ses fils. Étant donné un jeu G, si l'on note $G_1^L, G_2^L, \ldots, G_m^L$ (resp. $G_1^R, G_2^R, \ldots, G_n^R$) les options gauches (resp. droites) de G, alors on écrira

$$G = \{G_1^L, G_2^L, \ldots, G_m^L | G_1^R, G_2^R, \ldots, G_n^R\}.$$

Pour simplifier, on écrira plus légèrement (avec un petit abus de notation) $G = \{G^L | G^R\}$ où G^L et G^R dénotent respectivement les ensembles d'options gauches et droites de G.

Le cas de base de cette construction correspond aux arbres avec un seul nœud, autrement dit au jeu où personne ne peut jouer. On appellera ce jeu le jeu 0, il sera noté $\{\ |\ \}$. Par la suite, on définira l'ensemble des jeux combinatoires comme l'union infinie des ensemble de jeux dont l'arbre est de hauteur n, pour $n \geqslant 0$. Ainsi, on posera :

$$\mathbb{G}_0 = \{0\}$$

et récursivement les jeux de *hauteur n* seront définis comme :

$$\mathbb{G}_{n+1} = \{\{G^L | G^R\} \mid G^L, G^R \subset \mathbb{G}_n\}.$$

L'ensemble des jeux combinatoires sera alors

$$\bigcup_{n \geqslant 0} \mathbb{G}_n.$$

Avec cette notation, la somme de deux jeux G et H se définit inductivement comme suit :

$$G + H = \{G^L + H, G + H^L | G^R + H, G + H^R\},$$

avec, comme cas de base, $G + \emptyset = G$.

Définition 2.2.1 (Négation). *La négation d'un jeu G, notée $-G$, est le jeu où les rôles de Left et Right sont inversés. On a $-G = \{-G^R | -G^L\}$ et cette définition est récursive.*

Avec cette notation, on pourra écrire $G - H$ pour $G + (-H)$. De plus, le jeu $-(-G)$ est isomorphe à G (leurs arbres respectifs sont les mêmes), et $o(G - G) = \mathcal{P}$.

Venons-en maintenant à la relation fondamentale des jeux en convention normale :

Définition 2.2.2 (Équivalence). *Étant donnés deux jeux G et H, on définit la relation $=$ suivante entre G et H :*

$$G = H \textit{ si pour tout jeu } X, o(G + X) = o(H + X).$$

Cette relation est clairement une relation d'équivalence. On notera toutefois que deux jeux isomorphes (ayant le même arbre) sont forcément égaux, mais que la réciproque n'est pas toujours vraie (on peut trouver deux jeux avec des arbres différents mais qui satisfont la relation ci-dessus).

Cette relation a permis à Conway de définir la valeur d'un jeu :

Définition 2.2.3 (Valeur d'un jeu). *Étant donné un jeu G, la* valeur *de G, notée $v(G)$, est la classe d'équivalence de G modulo la relation $=$.*

On notera $\mathbb{G}$ l'ensemble des valeurs de tous les jeux possibles.

L'un des premiers résultats faciles qui découle de cette définition est le fait que dans une somme de jeux, on peut toujours remplacer un terme par un autre jeu qui lui est équivalent :

Proposition 2.2.4. *Étant donné un jeu G et deux jeux H et H' tels que $H = H'$, on a $G + H = G + H'$.*

Notons que la relation d'équivalence $=$ peut tout-à-fait être étendue dans d'autres contextes que les jeux combinatoires en convention normale, en faisant varier les notions d'issue, de somme ou de "pour tout jeu". Dans de tels contextes (jeux misère, jeux à score par exemple), certaines des propriétés que nous allons découvrir maintenant ne sont cependant plus vraies.

2.2.2 $\mathbb{G}$ est un groupe

L'une des propriétés les plus remarquables est que l'ensemble $\mathbb{G}$ muni de l'opérateur $+$ forme un groupe abélien. Examinons rapidement pourquoi.

Commençons par une propriété que nous avons déjà rencontrée, celle de l'élément neutre. Par définition du jeu noté 0, on a clairement que pour tout jeu G, $G + 0 = G$ (puisqu'aucun coup n'est jouable depuis 0).

Une question naturelle est la détermination de la classe d'équivalence de 0. La réponse est donnée par le théorème suivant :

Théorème 2.2.5. *$G = 0$ si et seulement si $o(G) = \mathcal{P}$.*

Preuve. Directement déduite de la proposition 2.1.5. ■

Ce résultat implique que les jeux $\mathcal{P}$ peuvent être retirés d'une somme de jeux sans en changer la valeur.

Venons-en à la question de l'inverse. Pour tout jeu G, il existe un jeu H tel que $G + H = 0$. Le candidat naturel pour H est le jeu négatif de G, en jouant symétriquement :

Proposition 2.2.6. *Pour tout jeu G, on a $G - G = 0$.*

Preuve. Issue du théorème 2.2.5 avec $o(G - G) = \mathcal{P}$. ■

Comme l'opérateur $+$ est associatif et commutatif, on a bien une structure de groupe abélien. On peut ajouter que ce groupe peut être muni d'une relation d'ordre partiel basée sur une priorité donnée à Left. En définissant en effet l'ordre partiel suivant sur les issues :

$$\begin{tikzcd}
 & \mathcal{N} \arrow[dr, "\leqslant"] & \\
\mathcal{R} \arrow[ur, "\leqslant"] \arrow[dr, "\leqslant"'] & & \mathcal{L} \\
 & \mathcal{P} \arrow[ur, "\leqslant"'] &
\end{tikzcd}$$

on peut définir l'ordre partiel suivant sur les jeux :

Définition 2.2.7. *Étant donné deux jeux G et H, on a*

$$G \geqslant H \text{ si } o(G + X) \geqslant o(H + X) \text{ pour tout jeu } X.$$

Cette définition permet notamment de décider de l'issue d'un jeu en fonction de sa position vis à vis du 0. En effet, les jeux $\mathcal{L}$ vérifient $G > 0$, les jeux $\mathcal{R}$ vérifient $G < 0$, les jeux $\mathcal{P}$ vérifient $G = 0$ (on le savait déjà !), et les jeux $\mathcal{N}$ sont incomparables avec 0.

2.2.3 Forme canonique d'un jeu

Si la définition récursive de la valeur d'un jeu semble rendre difficile toute simplification, il existe toutefois deux règles générales qui permettent de simplifier l'arbre tout en conservant la valeur du jeu.

Dans ce qui suit on considère un jeu G :

$$G = \{G_1^L, G_2^L, \ldots, G_m^L | G_1^R, G_2^R, \ldots, G_n^R\}.$$

Proposition 2.2.8 (Domination)**.** *Supposons que parmi les options gauches de G, il existe une position inférieure ou égale à une autre. Sans restreindre la généralité, supposons $G_1^L \leqslant G_2^L$. Alors cette position est dite* dominée *et le jeu $H = \{G_2^L, \ldots, G_m^L | G_1^R, G_2^R, \ldots, G_n^R\}$ est équivalent à G.*

Moralement, le coup G_1^L étant moins favorable à Left que G_2^L, Left ne le choisira jamais et le sous-arbre qui lui correspond n'est pas utile. Notons qu'on a le symétrique de cette proposition pour Right (si $G_1^R \geqslant G_2^R$, on peut enlever G_1^R sans affecter la valeur du jeu).

La seconde propriété de simplification est moins évidente :

Proposition 2.2.9 (Réversibilité)**.** *Supposons que pour une option gauche de G, disons G_1^L, il existe une réponse de Right vers un jeu noté G_1^{LR}, de valeur inférieure ou égale à G. Alors la position G_1^L est dite réversible, et le jeu $H = \{G_1^{LR^L}, G_2^L, \ldots, G_m^L | G_1^R, G_2^R, \ldots, G_n^R\}$ est équivalent à G, où $G_1^{LR^L}$ est l'ensemble des jeux atteignables pour Left depuis G_1^{LR}.*

Preuve. La preuve formelle de ce résultat est accessible dans [1] (chapitre 4) ou [122] (p. 65). La signification de la réversibilité peut toutefois s'expliquer assez bien : comme $G_1^{LR} \leqslant G$, on peut considérer que si Left choisit de jouer G_1^L, alors Right décide de répondre automatiquement G_1^{LR} car la position obtenue est plus favorable pour lui que la position de départ. Dans ce cas, Left a conscience que s'il choisit de jouer G_1^L, cela revient à choisir parmi les options gauches de G_1^{LR}, d'où la substitution définie dans la proposition. ■

Ainsi, un arbre de jeu peut être réduit en supprimant tous les sommets correspondant à des positions dominées ou en substituant des sommets correspondant à des positions réversibles par d'autres jeux plus simples. On considérera qu'un jeu est sous sa *forme canonique* s'il n'existe plus aucun sommet de l'arbre qui soit dominé ou réversible. Autrement dit, la forme canonique est atteinte lorsqu'on a appliqué toutes les simplifications possibles de l'arbre de jeu. On peut alors montrer que tout jeu admet une et une seule forme canonique qui lui soit équivalente (preuve dans [122], Théorème 2.9).

2.2.4 Simplifier les valeurs d'un jeu

Rappelons que la problématique initiale qui a conduit à l'introduction des valeurs d'un jeu est celle du calcul de l'issue pour les sommes de jeux. De façon plus générale, on serait encore plus heureux de pouvoir évaluer la valeur de $G + H$ connaissant celles de G et H. Existe-t-il des jeux pour lesquels on aurait $v(G + H) = v(G) + v(H)$ en interprétant l'addition comme l'addition classique de nombres ? Cela nécessiterait bien entendu l'existence d'une application allant des classes d'équivalence de $=$ vers certains nombres.

Certains jeux sont des nombres

Dans certains cas, un plongement des valeurs de jeu vers certains nombres va permettre de répondre favorablement à cette question. Commençons par définir les jeux entiers positifs :

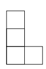
Définition 2.2.10 (Jeux entiers positifs)**.** *Le jeu* 1 *est défini comme le jeu de valeur* $\{0|\ \}$*. Pour tout entier* $i > 1$*, le jeu* i *est défini comme étant le jeu* $(i-1)+1$.

Cette définition traduit assez naturellement que les jeux i avec $i > 0$ sont des jeux où Left possède i coups d'avance sur Right avant d'être bloqué. En d'autres termes, on peut aussi écrire $i = \{i-1|\ \}$. Par exemple, dans DOMINEERING, la position  vaut 2 car Left peut jouer deux fois avant d'être bloqué (et Right est bloqué immédiatement).

On définit les jeux entiers négatifs de la même façon, le jeu $-i$ étant la négation du jeu i et peut s'écrire $-i = \{\ |-i+1\}$.

Pour ces jeux-là, on a bien $v(G + H) = v(G) + v(H)$. Mais peut-on avoir d'autres nombres que les entiers ? Par exemple, quelle serait la valeur du jeu $G = \{0|1\}$? Clairement, on peut montrer que $0 < G < 1$. On est alors tenté de lui donner la valeur $1/2$. Pour vérifier que cette valeur serait bien consistante avec l'addition, il faut démontrer que $G + G = 1$. On laissera le lecteur démontrer ce résultat et on considérera désormais :

Définition 2.2.11 (Jeu $1/2$)**.** *Le jeu* $1/2$ *est défini comme le jeu de valeur* $\{0|1\}$.

On peut constater que le jeu  vaut précisément $1/2$.

On peut ensuite récursivement définir tous les autres jeux de la forme $\frac{1}{2^m}$ comme étant les jeux de valeur $\{0|\frac{1}{2^{m-1}}\}$. Et de manière plus générale, en sommant de tels jeux, on peut définir tous les jeux de la forme $\frac{m}{2^n}$. Ce sont les nombres dyadiques.

Ainsi, l'ensemble des jeux qui ont pour valeur des nombres dyadiques forment un sous-groupe de $\mathbb{G}$. Plus généralement, on peut montrer que pour un jeu G dont toutes les valeurs de G^L sont des nombres dyadiques inférieurs à toutes les valeurs de G^R, sa valeur est elle-même un nombre dyadique. Ainsi, le jeu $\{1/2, 2, -2|3, 7\}$ sera un nombre dyadique. De plus, il est facile de calculer la valeur du jeu en connaissant celles de ses options via la règle de simplicité :

Proposition 2.2.12 (Règle de simplicité)**.** *Étant donné $G = \{G^L|G^R\}$ où G^L et G^R sont des nombres dyadiques tels que $G^L < G^R$, le jeu G a pour valeur le nombre dyadique le plus simple compris entre G^L et G^R (c'est-à-dire celui ayant le plus petit dénominateur, et à dénominateur égal, celui ayant le plus petit numérateur en valeur absolue).*

Preuve. Soit x le nombre dyadique le plus simple entre G^L et G^R. On montre que $G - x = 0$ en montrant successivement $G - x \geqslant 0$ et $G - x \leqslant 0$ (les preuves pour chaque inégalité sont similaires). Prouver que $G - x \geqslant 0$ revient à montrer que si Right commence, il perd. S'il joue vers $G^R - x$, il perd car cette position a une valeur positive. S'il joue vers $G - x^L$, alors la simplicité de x garantit le fait que x^L n'est pas compris entre G^L et G^R (en effet, la simplicité d'un nombre se traduit par la hauteur de son arbre, et x^L a un arbre moins haut que x comme il s'agit d'une de ses options). Plus précisément, on a nécessairement $x^L \leqslant G^L$, et ainsi, Left peut répondre vers $G^L - x^L$ en s'assurant la victoire. ■

Par exemple, le jeu $G = \{1|7\}$ a pour valeur 2.

Cette règle combinée à la règle de domination permet notamment de calculer la valeur des jeux dont toutes les options gauches ont des valeurs inférieures à toutes les options droites.

Ainsi, la somme de deux jeux qui sont des nombres dyadiques est aussi un nombre dyadique obtenu simplement en faisant l'addition usuelle de ces deux nombres. Par exemple, sur la figure 2.3, la somme vaut $1/2 + 1/2 - 1 = 0$ et donc la position globale est $\mathcal{P}$.

Remarque 2.2.13. *Dans sa théorie, Conway définit les nombres comme les jeux vérifiant la propriété $x < y$ pour toute option $x \in G^L$ et $y \in G^R$. Moralement, cela correspond aux jeux où personne n'a vraiment intérêt à jouer car pour les deux joueurs, les options ont des valeurs moins intéressantes que celle de la position initiale. Les nombres dyadiques vérifient bien cette propriété. Par ailleurs, la règle de simplicité garantit que les seuls nombres de la théorie de Conway (selon cette définition) sont les nombres dyadiques. Par ailleurs, Conway a également considéré les jeux ayant des options infinies (ce que nous ne présentons pas dans* *ce manuscrit), et c'est ainsi qu'il a pu définir tous les autres nombres — réels et même surréels — comme valeurs de tels jeux.*

Tous les jeux ne sont pas des nombres dyadiques

Certains jeux comme HACKENBUSH [9] ont la particularité que toutes leurs positions ont pour valeur des nombres dyadiques. Si les nombres dyadiques correspondent à des jeux faciles à traiter en termes de somme, tous les jeux ne correspondent pas à des nombres dyadiques. Par exemple, le jeu $\{0|0\}$ ne sera pas considéré dans la théorie de Conway comme un nombre dyadique, ni même comme un nombre réel ou surréel. Plus précisément, on peut montrer que $\{0|0\} < x$ pour tout jeu $x > 0$ qui est un nombre dyadique, et $\{0|0\} > y$ pour tout jeu $y < 0$ qui est un nombre dyadique. De tels jeux sont appelés des *infinitésimaux* :

Définition 2.2.14 (Infinitésimal). *Un jeu G est* infinitésimal *si $x > G > -x$ pour tout nombre positif dyadique x.*

Théorème 2.2.15 (Théorème de Lawnmower [9]). *Si un jeu G a son arbre qui vérifie que pour tout coup de Left il existe un coup pour Right (et vice versa), alors ce jeu admet une valeur infinitésimale.*

Preuve. Pour tout $x > 0$, on montre que Left gagne sur $x - G$ quel que soit le premier joueur. La preuve se fait en considérant les deux cas (Left commence et Right commence), et en utilisant le fait que par induction, les options de G^L et G^R vérifient aussi le théorème. ∎

Moralement, ce résultat signifie qu'aucun des deux joueurs n'aura de coup d'avance sur l'adversaire et donc que la valeur du jeu est proche de 0. C'est en particulier le cas pour les jeux impartiaux.

Voyons quelques exemples de jeux infinitésimaux. Prenons les positions de NIM avec $n = 1$. En notant $*i$ la valeur du jeu (i), on a :

$$*0 = \{\ |\ \}, \quad *1 = \{0|0\}, \quad *2 = \{*0, *1 | *0, *1\}, \quad *3 = \{*0, *1, *2 | *0, *1, *2\}, \quad \ldots$$

Ces valeurs sont appelées des *nimbers*. Tous les jeux impartiaux sont équivalents à un nimber et donc à un jeu de NIM à une seule composante. On notera au passage que $*0 = 0$, et on simplifiera plus souvent $*1$ en $*$. De plus, la valeur d'un jeu impartial dont les options sont des nimbers est connu :

Théorème 2.2.16. *Étant donné $G = \{G^L|G^R\}$ où $G^L = G^R$ est un ensemble de nimbers, G a pour valeur le nimber $*c$, où $*c$ est le plus petit nimber non présent dans G^L.*

Preuve. Il suffit de montrer que $G + *c = 0$, autrement dit que $G + *c$ est perdant pour le premier joueur. Si celui-ci choisit une option de G^L, alors il laisse à son adversaire un jeu $*a + *c$ avec $a \neq c$. Il s'agit là d'un jeu de Nim en dimension 2 qui est gagnant pour le premier joueur d'après le théorème 2.1.4. Si en revanche le premier joueur joue dans la composante $*c$, il laisse par définition une somme $G + *b$ avec $b < c$ à son adversaire. Par minimalité de c, le nimber $*b$ appartient à G^L et G^R. Le second joueur choisit alors de répondre dans G en jouant vers le jeu $*b + *b$, qui vaut 0. Le premier joueur perdra donc dans tous les cas. ■

Par exemple, on a $\{*0, *1, *4| * 0, *1, *4\} = *2$. À partir d'un arbre de jeu impartial, ce théorème permet de construire un algorithme qui calcule la valeur de chaque sommet de l'arbre avec un algorithme similaire à celui décrit en section 1.5.

Enfin, jouer sur une somme de jeux dont les valeurs sont des nimbers est facile à appréhender depuis les travaux de Sprague et Grundy [124] :

Théorème 2.2.17. *Étant donné deux jeux de valeurs $*a$ et $*b$, on a $*a + *b = *c$ où $c = a \oplus b$.*

Preuve. Directement déduite du théorème 2.1.4. ■

Ainsi, les nimbers sont des infinitésimaux qui correspondant à des positions de jeu $\mathcal{N}$ et ne sont donc pas comparables avec 0. Il en existe bien d'autres qui eux sont comparables avec 0, comme par exemple $\{0|*\}$ et $\{*|0\}$, qu'on notera respectivement $\uparrow$, et $\downarrow$. D'autres peuvent être construits par la somme, comme le jeu $\uparrow + *$ ou $\Uparrow = \uparrow + \uparrow$. Si certains infinitésimaux sont incomparables entre eux ($*$ et $\uparrow$), d'autres peuvent être comparés ($*2 < \uparrow$), permettant les simplifications d'écriture via les options dominées ou réversibles. Citons par exemple les égalités

$$\{\uparrow | \downarrow\} = \{\uparrow |0\} = \{0| \downarrow\} = *,$$

ou encore, si x est un nombre dyadique,

$$\{x|x\} = x + *.$$

Enfin, notons que la somme d'un nombre et d'un infinitésimal est dite *nombresque*, autrement dit une valeur infinitésimalement proche d'un nombre. Les infinitésimaux sont eux-mêmes des nombresques.

Les switches

Enfin, il existe d'autres jeux qui ne sont ni des nombres, ni des nombresques. C'est le cas des *switches*.

Définition 2.2.18 (switch). *Un* switch *est un jeu de la forme* $\{x|y\}$*, où* x *et* y *sont des nombre dyadiques tels que* $x > y$*. En particulier, le switch* $\{x| - x\}$ *sera noté* $\pm x$*.*

Les switches sont des jeux où chaque joueur a envie de jouer en premier car laisser l'autre commencer conduirait à une position de valeur moins favorable.

La somme d'un switch et d'un nombre se calcule bien. En l'occurrence, si G est un switch $\{x|y\}$ et z est un nombre dyadique, alors on a :

$$\{x|y\} + z = \{x + z|y + z\}.$$

De manière générale, les switches et autres valeurs (obtenues par récursion avec des switches — par exemple $\{\pm 2|\{0| - 2\}\}$) ou les sommes de switches avec des infinitésimaux sont difficiles à simplifier ou à comparer. Nous ne rentrerons ici pas dans les détails de leur traitement.

2.2.5 Comment jouer sur une somme de jeux ?

C'est la question principale qui nous anime, dès lors qu'on connaît les valeurs des composantes d'une somme. Par exemple, imaginons le jeu $3 + * \pm 2$, peut-on savoir qui gagne et quelle stratégie appliquer ?

Si on a vu que les sommes de nombres et de nimbers se simplifient bien et permettent de répondre favorablement à ces questions, cela devient plus compliqué lorsque l'on somme des jeux ayant des types de valeurs différentes. Pour savoir quelle composante est prioritaire sur quelle autre, Conway a défini la notion de *température* d'un jeu. Il s'agit d'une valeur numérique qui quantifie le degré d'urgence pour jouer dans ce jeu. Sans rentrer dans les détails de la définition de cette valeur ([122], Def. 5.2), on a les propriétés suivantes :

Proposition 2.2.19 (Température). *Soit* $G \in \mathbb{G}$*.*

- *Si* G *est un nombre* $\frac{m}{2^n}$ *avec* $m \neq 0$*, alors sa température est négative et vaut* $-\frac{1}{2^n}$ *;*
- *si* G *est nombresque, alors sa température est nulle ;*
- *si* G *n'est ni un nombre, ni nombresque, alors sa température est positive. En particulier si* G *est un switch* $\{x|y\}$ *(avec* $x > y$*), alors sa température vaut* $\frac{1}{2}(x - y)$*.*

Dans une somme de switches, de nombresques et de nombres, la stratégie est alors la suivante : jouer à chaque tour dans le jeu ayant la plus haute température. Les nombres sont donc considérés comme des jeux où il n'y a aucune urgence de jouer puisque d'une certaine manière, ce seront des coups "gâchés". On dit qu'il s'agit de jeux *froids*. Au contraire, les switches sont dits *chauds* et les nombresques sont dits *tièdes*. Par exemple, le jeu $\{2|-\frac{1}{2}\}+\{1|-1\}+\{0|-1\}+\{0|0\}+\frac{3}{4}$ est une somme de jeux de températures respectives $\frac{5}{4}, 1, \frac{1}{2}, 0$ et $-\frac{1}{4}$. Si Left commence, on arrivera après 4 coups à la position de valeur

$$2-1+0+0+\frac{3}{4}=\frac{7}{4}.$$

Si Right commence, on obtiendra le jeu de valeur

$$-\frac{1}{2}+1-1+0+\frac{3}{4}=\frac{1}{4}.$$

Dans tous les cas, la résultante sera strictement positive et ainsi le jeu global est $\mathcal{L}$.

Dans les cas plus complexes (avec des valeurs à température positive qui ne sont pas des switches), d'autres notions sont mises en place pour savoir comment jouer (cf. la notion de thermographe) mais nous ne les aborderons pas ici. De manière générale, décider si Left gagne sur une somme de jeux sous forme canonique reste un problème PSPACE-complet [137].

2.3 Résolution pratique de jeux

Dans la partie précédente, nous avons vu comment étudier de manière théorique un jeu combinatoire. Nous allons maintenant considérer la résolution pratique de jeux. Nous nous concentrons ici sur des jeux impartiaux (les deux joueurs ont les mêmes coups à jouer). Comme vu précédemment, les valeurs ne peuvent alors être que des nimbers et une fois ceux-ci connus, la résolution du jeu est assez aisée. En pratique, calculer ces nimbers ou simplement calculer l'issue du jeu se révèle la plupart du temps compliqué. Dans un premier temps, nous montrons en quoi la classe de complexité PSPACE est la classe pertinente pour l'étude de jeux et donnons un exemple de jeux PSPACE-complet qui devient polynomial en changeant légèrement les règles. Nous nous intéressons ensuite à une classe importante de jeux impartiaux, les jeux octaux, pour lesquels un comportement périodique est conjecturé. Il s'agit selon Guy [42] de l'un des problèmes non résolus les plus importants des jeux combinatoires.

2.3.1 Quelle classe de complexité pour les jeux ?

Vérifier qu'une stratégie est gagnante nécessite souvent un temps de calcul exponentiel car nous n'avons guère d'autre choix que de tester toutes les options de jeu de l'adversaire. Par contre, lorsqu'un jeu ne boucle pas, il est en général possible de tester une stratégie avec un espace de taille polynomial en la taille du jeu [1]. C'est principalement pour cette raison que la classe de complexité pertinente pour l'étude des jeux est la classe PSPACE, contenant tous les problèmes de décision pouvant se résoudre sur une machine de Turing avec un espace de taille polynomial. Un problème est PSPACE-complet s'il est dans PSPACE et que tout problème de PSPACE peut s'y réduire en temps polynomial.

Le problème canonique PSPACE-complet est le problème QBF — *Quantified Boolean Formula* [5]. Il s'agit de décider si une formule de la forme $Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \varphi(x_1, \ldots, x_n)$ est vraie, où les Q_i sont des quantificateurs, x_i des variables booléennes et φ est une formule close, c'est-à-dire que les seules variables apparaissant dans φ sont $x_1, \ldots, x_n$.

La nature de ce problème est exactement la même que celle de chercher une stratégie gagnante pour un jeu combinatoire. En effet, un certificat pour le problème QBF ressemble fort à un arbre de jeu donnant la stratégie gagnante d'un joueur. De manière plus formelle, on peut considérer le jeu QBF-GAME. Le plateau de ce jeu est une formule booléenne φ contenant des variables booléennes $x_1, \ldots, x_n$. Le premier joueur choisit la valeur de la première variable x_1. Puis le deuxième joueur fait de même avec x_2 et ainsi de suite. Le premier joueur gagne si et seulement si, à la fin, φ est vraie. Le problème est de décider l'issue de ce jeu : le premier joueur a-t-il une stratégie gagnante ? Clairement, le premier joueur a une stratégie gagnante si et seulement si la formule

$$\exists x_1 \forall x_2 \exists x_3 \cdots \varphi(x_1, x_2, \ldots, x_n)$$

est vraie. On peut démontrer assez facilement que ce problème est équivalent à QBF et donc que QBF-GAME est PSPACE-complet. Ainsi tout problème PSPACE-complet est dans sa nature proche d'un jeu.

Un des premiers jeux à avoir été démontré comme PSPACE-complet est le jeu GEOGRAPHY. C'est d'ailleurs à partir de ce jeu que Lichtenstein et Sipser ont montré que le Go dans une version généralisée était PSPACE-difficile [79]. GEOGRAPHY se joue sur un graphe orienté. Un jeton est déposé sur un sommet s du graphe. Puis les joueurs déplacent l'un après l'autre le

1. On considère en général pour la taille du jeu un encodage de la position initiale.

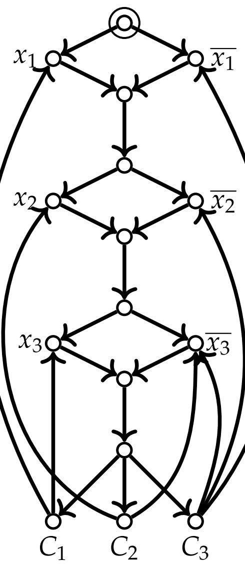


FIGURE 2.4 – *Réduction de* QBF-GAME *vers* GEOGRAPHY *pour la formule* $\exists x_1 \forall x_2 \exists x_3, (x_1 \vee x_3) \wedge (x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})$.

jeton en suivant les arcs et sans repasser par un arc déjà visité [2]. Le joueur ne pouvant plus jouer a perdu.

Théorème 2.3.1 (Schaeffer [120])**.** *Décider si le premier joueur a une stratégie gagnante dans* GEOGRAPHY *est* PSPACE-*complet.*

Preuve. La réduction se fait à partir de QBF-GAME joué sur des formules avec un nombre impair de variables et sous forme normale conjonctive (on se convaincra assez facilement que cela ne change pas la complexité du problème). À partir d'une telle formule, un graphe est construit de telle sorte que le premier joueur a une stratégie gagnante dans un des jeux si et seulement s'il a une stratégie gagnante dans l'autre jeu. Un exemple de construction est donné dans la figure 2.4 pour la formule $\exists x_1 \forall x_2 \exists x_3, (x_1 \vee x_3) \wedge (x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})$. Le premier joueur peut "décider" les valeurs des variables x_1 et x_3 en orientant le jeton vers le littéral correspondant tandis que le joueur 2 décide la valeur de x_2 et la clause finale. Le premier joueur gagne s'il peut revenir sur un littéral qui a été choisi, le deuxième joueur ne pouvant plus jouer. Cela signifie aussi que la clause est satisfaite. ■

2. L'origine de ce jeu vient du jeu populaire où l'on doit dire chacun son tour le nom d'un pays commençant par la dernière lettre du pays précédemment cité, d'où le nom GEOGRAPHY !

Cette preuve s'adapte facilement si l'on ne peut pas revenir sur un sommet déjà ou bien si l'on joue sur un graphe planaire [79]. Par contre, la variante suivante, jouée sur un graphe orienté, est polynomiale. Le jeu UV-GEOGRAPHY (Undirected-Vertex Geography) se joue sur un graphe non orienté et au lieu de ne pas réutiliser d'arcs, les joueurs ne peuvent pas réutiliser un sommet déjà visité. Rappelons qu'un *couplage* dans un graphe est une union d'arêtes deux à deux non adjacentes. Il *sature* un sommet si l'une des arêtes est adjacente à ce sommet.

Théorème 2.3.2 ([36]). *Le premier joueur gagne à* UV-GEOGRAPHY *sur* (G, s) *si et seulement si tout couplage de taille maximum de* G *sature* s.

Preuve. Supposons qu'il existe un couplage M de taille maximum qui ne sature pas s. Alors le second joueur joue à chaque fois sur une arête de M et gagne. Sinon, le premier joueur choisit un couplage M de taille maximum et joue sur ce couplage. Si le second joueur gagne, les arêtes formées par ses coups forment un couplage de taille maximum qui ne sature pas s. ■

Cela implique un algorithme polynomial pour décider si le premier joueur gagne à UV-GEOGRAPHY. En effet, on peut utiliser l'algorithme d'Edmonds [32] qui calcule en temps polynomial la taille maximum d'un couplage dans un graphe. Il suffit alors de comparer les tailles maximum des couplages dans G et $G \setminus \{s\}$.

Le cas de UV-GEOGRAPHY reste assez exceptionnel et la plupart des jeux combinatoires connus (au sens large) sont PSPACE-difficiles lorsque l'on généralise leurs règles sur des plateaux de taille quelconque [3]. C'est le cas de Hex [33], du Go [79] ou des échecs [35]. Ces deux derniers sont même EXPTIME-complet (du fait des boucles possibles dans les règles du jeu). Pour de nombreux jeux, la complexité reste un problème ouvert important. C'est par exemple le cas de ARC KAYLES et DOMINEERING. Pour en savoir plus, nous invitons le lecteur à consulter le tableau de complexité des jeux maintenu par Kyle Burke [4].

2.3.2 Conjecture de Guy sur les jeux octaux

Il existe tout de même une classe importante de jeux pour lesquels on conjecture une résolution polynomiale. Il s'agit des *jeux octaux* qui

3. On peut généraliser les règles du jeu d'échecs en prenant un plateau carré de taille $n \times n$ et les pièces habituelles en respectant les mêmes proportions que sur le 8×8. Le problème est alors de décider, pour une configuration quelconque, si l'un des joueurs a une stratégie gagnante.

4. `http://turing.plymouth.edu/~kgb1013/rulesetTable.php`

généralise à la fois le jeu de NIM et le jeu CRAM (DOMINEERING où les joueurs peuvent jouer les dominos dans n'importe quelle direction), joué sur une ligne.

Les jeux octaux se jouent avec plusieurs tas de jetons. Chacun son tour, les joueurs prennent des jetons dans un seul des tas et peuvent éventuellement séparer le tas en 2. Les règles de prises sont déterminées par le nom du jeu, qui est un code noté $\cdot\mathbf{d_1 d_2 d_3}\cdots$ où $\mathbf{d_i}$ est un entier compris entre 0 et 7. Chaque d_i étant compris entre 0 et 7, on peut écrire de manière unique $\mathbf{d_i} = x_0^i + 2x_1^i + 4x_2^i$ avec $x_j^i \in \{0,1\}$. La règle de prise est la suivante : on peut retirer i jetons dans l'un des tas et séparer le tas en j tas si et seulement si $x_j^i = 1$. Le joueur qui ne peut plus retirer de jetons a perdu.

Par exemple, $\mathbf{d_2} = 3$ indique qu'on peut enlever deux jetons d'un tas, sans le diviser en deux, tandis que $\mathbf{d_2} = 7$ indique que l'on peut enlever deux jetons en divisant éventuellement le tas en deux. Le jeu de NIM a pour code $\cdot\mathbf{3333}\cdots$ tandis que jouer à $\cdot\mathbf{07}$ revient à jouer à CRAM [5]. En effet, on peut considérer les jetons comme étant alignés. On ne prend alors que des jetons consécutifs et les trous délimitent les tas de jetons.

Pour résoudre complètement un jeu octal, il suffit de connaître sa valeur de Conway avec un seul tas de jetons. En effet, les jeux octaux sont des jeux impartiaux, leurs valeurs sont donc des nimbers. Ainsi, dès qu'il y a plusieurs tas, comme il s'agit en fait d'une somme de jeux d'un seul tas, on peut déduire la valeur du jeu en utilisant le théorème 2.2.17. On considère donc un jeu octal comme complètement résolu lorsque l'on connaît la séquence de ses nimbers $(u(n))_{n\in\mathbb{N}}$ pour un tas de taille n, appelée *séquence de Grundy*. Pour calculer $u(n)$ pour n fixé, il suffit de calculer les nimbers de toutes les options obtenues en jouant sur un tas de taille n et de prendre le plus petit nimber non présent (Théorème 2.2.16).

La séquence de Grundy de NIM est ainsi $*0, *1, *2, *3, *4, *5, \ldots$ que l'on notera $012345\cdots$ pour plus de lisibilité. D'autres jeux octaux ont des séquences particulièrement simples. C'est le cas du jeu $\cdot\mathbf{333}$ qui correspond au jeu populaire des allumettes : chacun son tour, les joueurs prennent 1,2 ou 3 allumettes et la personne qui prend la dernière allumette à gagner. Il est assez rapide de voir que la séquence de Grundy de $\cdot\mathbf{333}$ est $01230123\cdots$. Cette séquence est périodique, comme c'est le cas pour tous les jeux qui s'écrivent avec un nombre fini de **0** et de **3**, appelés *jeux de soustraction* (car on ne peut que soustraire des jetons d'un tas et jamais diviser les tas en deux). On peut démontrer la périodicité en remarquant qu'il y a un nombre borné d'options dans chaque position. D'autres jeux octaux ont des séquences d'apparence plus compliquée, comme celle de $\cdot\mathbf{07}$ qui commence

5. Lorsqu'il y a un nombre fini de $\mathbf{d_i}$ non nuls, l'écriture s'arrête au dernier $\mathbf{d_i}$ non nul.

par
$$0011203110332240522 33 \cdots .$$

Malgré son apparence aléatoire, elle est en fait périodique, de période 34. Pour démontrer qu'un tel jeu est périodique de période p à partir de l'indice i, il suffit de montrer que la périodicité est vraie jusqu'aux indices $2i + 2p + t$ où t est le plus grand nombre de jetons que l'on peut prendre [9]. De nombreux jeux octaux finis (c'est-à-dire avec un nombre fini de $\mathbf{d_i}$ non nuls) ont été étudiés et, de fait, tous ceux dont les calculs ont abouti ont une séquence de nimbers ultimement périodique, ce qui a mené à la conjecture suivante :

Conjecture 2.3.3 (Conjecture de Guy [9])**.** *La séquence de Grundy d'un jeu octal fini est ultimement périodique.*

Si cette conjecture est vraie, il suffirait alors, pour résoudre un jeu octal, de calculer le reste de n par la période pour savoir si la situation est gagnante ou perdante avec n jetons et cela donnerait aussi le coup à jouer. Cependant la période peut-être grande (pour $\cdot$**16** la période est de taille 149159 et la prépériode de taille 105351) et la conjecture loin d'être prouvée (l'étude de $\cdot$**06** est encore ouverte !).

Mentionnons pour finir des extensions des jeux octaux dans les graphes qui apportent de nouvelles questions de recherche. En effet, un jeu octal peut être vu comme un jeu sur un chemin où l'on retire des sommets suivant certaines règles. Il est ainsi naturel d'étudier des jeux similaires sur les graphes. Le jeu NODE KAYLES se joue sur un graphe simple non orienté. Chacun son tour, les joueurs choisissent un sommet et retirent du graphe ce sommet et tous ses voisins. Sur le chemin, cela revient à jouer à $\cdot$**137** : en effet, dans $\cdot$**137**, on peut retirer un sommet s'il est tout seul, ou bien deux sommets sur un bord (sans séparer le chemin) ou bien trois sommets adjacents n'importe où. Sans surprise, ce jeu est PSPACE-complet [120]. La réduction est moins simple que pour GEOGRAPHY. En effet, dans GEOGRAPHY, le nombre de coups jouables est le nombre de voisins et peut donc être très petit alors que dans NODE KAYLES, un joueur peut jouer sur n'importe quel sommet du graphe. Pour contrer cela, la réduction crée des sommets sur lesquels un joueur n'a jamais intérêt à jouer car il perdrait immédiatement. Notons que nous ne connaissons pas la complexité de NODE KAYLES sur les arbres.

Le jeu ARC KAYLES se joue sur un graphe où chacun son tour un joueur enlève une arête et les arêtes adjacentes. Ce jeu correspond à $\cdot$**07** sur les chemins. Sur une grille, cela correspond à placer des dominos sans superposition (comme à DOMINEERING mais en version impartiale : les joueurs n'ont pas d'orientation prédéfinie et peuvent jouer les dominos dans le sens qu'ils veulent). Ce jeu appelé CRAM n'est pas résolu sur les grilles dont les côtés sont tous les deux impairs (lorsqu'un côté est pair, un argument de symétrie donne une stratégie évidente pour l'un des deux joueurs). La complexité générale d'ARC KAYLES n'est pas non plus connue.

D'autres généralisations des jeux octaux et en particulier des jeux de soustraction sur les graphes commencent aussi à être étudiées.