Marthe BONAMY
Xavier GOAOC
Fredrik JOHANSSON
Jérôme LEROUX
Marni MISHNA
Irena PENEV
Sylvain SCHMITZ

Informatique Mathématique
Une photographie en 2020

Sébastien LABBÉ et Vincent PENELLE (éd.)

CNRS Éditions

CNRS Éditions, 2020, ISBN : 978-2-271-13201-7

Dépôt légal : mars 2020

Sommaire

Les auteurs


Marthe BONAMY est chargée de recherche au CNRS au Laboratoire Bordelais de Recherche en Informatique (LaBRI). au sein de l'équipe Combinatoire et Algorithmes et plus spécifiquement du groupe Graphes et Optimisation. **Irena PENEV** est Professeure adjointe au *Computer Science Institute* de *Charles University* à Prague, République tchèque. Leur chapitre « Nombre chromatique et sous-graphes induits » traite liens entre le nombre chromatique d'un graphe, la taille de son plus grand ensemble stable et celle de sa plus grande clique.


Jérôme LEROUX est directeur de recherche CNRS au LaBRI au sein de l'équipe Méthodes Formelles. **Sylvain SCHMITZ** est professeur d'informatique à l'Université de Paris et est membre de l'Institut de recherche en informatique fondamentale (IRIF). Leur chapitre « Accessibilité des systèmes d'addition de vecteurs » présente les dernières avancées sur la complexité algorithmique de ce problème.

Marni MISHNA est professeure de Mathématiques au Département de Mathematiques à l'Université Simon FRASER, Burnaby, Canada. Son chapitre « Combinatoire analytique » introduit les techniques de base de la combinatoire analytique et les séries génératrices multivariées.

Fredrik JOHANSSON est chercheur à INRIA Bordeaux et à l'Institut de Mathématiques de Bordeaux (IMB) au sein de l'équipe LFANT. Son chapitre « Calculer avec les nombres réels » s'intéresse aux problématiques liées à la représentation des nombres réels dans un ordinateur.

Xavier GOAOC est professeur à l'Université de Lorraine, au département d'informatique de l'École des Mines de Nancy et au Laboratoire lorrain de recherche en informatique et ses applications (LORIA), au sein de l'équipe INRIA Gamble. Son chapitre « Convexité combinatoire » introduit à la convexité combinatoire, ses applications algorithmiques et ses prolongements en combinatoire topologique.

Préface

Le présent ouvrage est le huitième de la série « Informatique Mathématique : une photographie ». Il rassemble des cours donnés lors de l'édition 2020 de l'École de Jeunes Chercheurs/Chercheuses en Informatique Mathématique, organisée à Bordeaux du 6 au 10 avril 2020. Au-delà d'être de simples supports de cours, ces livres ont vocation à constituer des photographies instantanées de notre domaine, qui illustrent sa diversité, sa vitalité et sa perpétuelle évolution. Ils constituent probablement la meilleure introduction à l'Informatique Mathématique que l'on puisse trouver. Au fil du temps, ils sont devenus la principale « vitrine » du GDR IM.

Les Écoles de Jeunes Chercheurs/Chercheuses en Informatique Mathématique sont organisées chaque année depuis la création en 2006 du GDR IM. Le but de ces écoles est de donner une formation complémentaire de haut niveau à un public constitué en général de doctorants ou de chercheurs ou enseignants chercheurs récemment recrutés. Cela peut constituer pour eux une mise à niveau dans certains domaines ou une véritable ouverture vers de nouvelles problématiques. Ceci est important car nos domaines évoluent et de nouvelles questions surgissent : qui travaillera exactement sur son sujet de thèse dans 10 ans ? De plus, il n'est pas rare qu'une idée surgisse par analogie : ce qui se passe dans un domaine proche peut enrichir le nôtre.

En leur montrant l'état de la recherche dans des domaines voisins de leur spécialité, l'école permet d'élargir la culture scientifique des doctorants et des jeunes chercheurs, et leur donne des outils qui leur permettront de mieux s'adapter à des environnements variés. Elle contribue ainsi à faciliter leur recrutement, leur insertion, leur mobilité et d'éventuelles reconversions. En leur donnant l'occasion de se rencontrer, de présenter leurs travaux et de confronter leurs idées, elle contribue aussi à créer et à souder une communauté de jeunes scientifiques autour des thèmes de l'informatique mathématique.

Nous remercions les auteurs d'avoir accepté avec enthousiasme de s'atteler à ce travail d'écriture, et pour l'ouvrage de grande qualité qui en résulte.

Les auteurs et le comité scientifique remercient très chaleureusement les relecteurs pour leur travail et leurs remarques sur les versions préliminaires de chaque chapitre : Simon ABÉLARD, Cedric CHAUVE, Célia CISTERNINO, Nathanaël EON, Paul GALLOT, Marie LEJEUNE, Théodore LOPEZ, Jean-François MARCKERT, Adeline MASSUIR, Pierre OHLMANN, Manon PHILIBERT, David RENAULT et Thibaut VERRON. Ils remercient également grandement Xavier CARUSO pour la traduction du chapitre écrit par Fredrik JOHANSSON.

À Bordeaux et Lyon, le 13 janvier 2020,

Guillaume THEYSSIER et Jean-Michel MULLER, co-directeurs du GDR IM,
Sébastien LABBÉ et Vincent PENELLE, coordinateurs de l'ouvrage.

Chapitre 1

Nombre chromatique et sous-graphes induits

Marthe BONAMY, Irena PENEV

Nous nous intéressons ici aux liens entre trois paramètres centraux en théorie des graphes : χ (nombre de couleurs nécessaires pour colorer les sommets de façon à ce que deux sommets adjacents reçoivent des couleurs distinctes), α (taille d'un plus grand stable, c.à.d. un ensemble de sommets deux à deux non-adjacents) et ω (taille d'une plus grande clique, c.à.d. ensemble de sommets deux à deux adjacents), ainsi qu'à l'impact de structures (sous-graphes) interdites sur leur comportement.

1.1 Introduction

1.1.1 Un peu d'histoire

En 1852, un botaniste, Francis GUTHRIE, s'aperçoit que pour colorer des cartes de façon à ce que des régions partageant une frontière reçoivent des couleurs distinctes, quatre couleurs semblent toujours suffire. Il écrit à son frère mathématicien pour partager cette conjecture, et lui demander s'il peut confirmer formellement. Le problème résiste, attire l'attention de nombreux combinatoriciens [1], et n'est prouvé qu'en 1976, avec assistance informatique [2, 3]. Cet énoncé est plus connu sous le nom de «Théorème des Quatre Couleurs».

1. Voir par exemple ici pour plus d'informations :
`https://www.enseignement.polytechnique.fr/profs/informatique/Georges.Gonthier/pi2000/pro/gonthier/`

Une carte peut être vue comme un graphe, dont les sommets sont les régions de la carte, et où deux régions sont adjacentes si elles ont une frontière commune (voir fig. 1.1). Une *bonne coloration* (ou simplement *coloration*) d'un graphe G est une fonction qui attribue une couleur à chaque sommet du graphe, de façon à ce que deux sommets adjacents reçoivent des couleurs distinctes. Autrement dit, une coloration est une fonction $c : V(G) \to \mathbb{N}^*$ telle que pour toute arête uv de G, on a $c(u) \neq c(v)$. Une *k-coloration* est une coloration utilisant au plus k couleurs différentes. Le Théorème des Quatre Couleurs peut alors s'énoncer ainsi : tout graphe issu d'une carte (graphe dit *planaire*) admet une 4-coloration. À noter que si aucune preuve purement humaine de ce fait n'est connue à ce jour, on connaît plusieurs preuves élémentaires du fait que les graphes planaires admettent tous une 5-coloration [7]. Inversement, il est NP-dur de décider si un graphe planaire est 3-colorable [20].

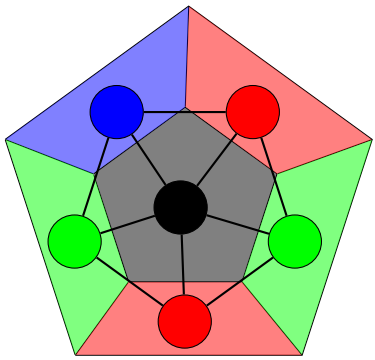


FIGURE 1.1 – *Un exemple du lien entre carte et graphe. Ici $\omega(G) = 3$, $\alpha(G) = 2$ et $\chi(G) = 4$, pour G étant le graphe représenté.*

Intéressons-nous de plus près à cette notion de nombre de couleurs nécessaires pour colorer un graphe G. Il s'agit du *nombre chromatique* de G, souvent dénoté $\chi(G)$. Pour démontrer une borne supérieure du nombre chromatique, il suffit d'exhiber une coloration appropriée [2]. En revanche, pour démontrer une borne inférieure, et justifier que le graphe n'admet pas de coloration avec moins de couleurs, il faut fournir des argument structurels souvent indirects. L'obstacle le plus évident à une k-coloration est la présence dans le graphe de $k + 1$ sommets deux à deux adjacents. Une telle structure est une *clique*, et la taille maximum d'une clique dans le graphe G est dénotée $\omega(G)$.

2. Ce qui est NP-dur en toute généralité.

Remarque 1.1.1. *Pour tout graphe G, on a $\omega(G) \leqslant \chi(G)$.*

Notons que dans les graphes planaires, si on peut trouver une clique de taille quatre, il est impossible d'obtenir une clique de taille cinq. Autrement dit, pour tout graphe planaire G, on a $\omega(G) \leqslant 4$. Ceci est un corollaire du Théorème des Quatre Couleurs, mais peut facilement être obtenu directement.

Un autre angle d'attaque pour une borne inférieure sur $\chi(G)$ consiste à observer que chaque couleur induit dans le graphe un ensemble de sommets deux à deux non-adjacents. C'est la notion de *stable*, qui fait directement écho à celle de clique. La taille maximum d'un stable dans le graphe G est dénotée $\alpha(G)$. Chaque couleur couvrant au plus $\alpha(G)$ sommets, on obtient la remarque suivante :

Remarque 1.1.2. *Pour tout graphe G, on a $\frac{|V(G)|}{\alpha(G)} \leqslant \chi(G)$.*

1.1.2 Toujours un gros stable ou une grosse clique ?

Rappelons que pour un graphe planaire G, on a la même borne supérieure pour $\omega(G)$ et pour $\chi(G)$: les deux valent au plus 4. S'agit-il là d'une tendance générale, que $\chi(G)$ et $\omega(G)$ se comportent de façon similaire ? Le graphe G de la fig. 1.1 est un exemple où $\chi(G) > \omega(G)$. Tout de même, on peut espérer que $\chi(G)$ soit borné par une fonction de $\omega(G)$. La remarque 1.1.2 nous donne $\alpha(G) \cdot \chi(G) \geqslant |V(G)|$ pour tout graphe G. Quelle est la plus grande fonction f telle que $\alpha(G) \cdot \omega(G) \geqslant f(|V(G)|)$ pour tout graphe G ? Si $f : x \mapsto x$ est peut-être un peu ambitieux, peut-on garantir $f : x \mapsto \sqrt{x}$?

Une borne logarithmique

Montrons par induction grossière que $\alpha(G) + \omega(G) \geqslant \log|V(G)|$ pour tout graphe G. Dénotons $g(n) = \min_{\text{graphe } G \text{ à } n \text{ sommets}}(\alpha(G) + \omega(G))$. Notons que $g(1) = 2$. Soit G un graphe à n sommets. Soit $u \in V(G)$, un sommet arbitraire de G. Soit G_1 le graphe induit par les voisins de u, et G_2 le graphe induit par les non-voisins de u. Notons que $\omega(G) \geqslant \omega(G_1) + 1$ et $\alpha(G) \geqslant \alpha(G_2) + 1$. On a donc $\alpha(G) + \omega(G) \geqslant \max(\alpha(G_1) + \omega(G_1) + 1, \alpha(G_2) + \omega(G_2) + 1)$. Tout sommet autre que u appartient soit à G_1 soit à G_2. L'un des deux graphes G_1 et G_2 contient nécessairement au moins $(n-1)/2$ sommets, d'où $g(n) \geqslant g(\lceil \frac{n-1}{2} \rceil) + 1$, et la conclusion. En faisant une analyse un peu plus fine, il est possible d'avoir une meilleure constante en facteur de $\log|V(G)|$. Lorsque l'on cherche à borner le produit et non la somme, on peut significativement améliorer l'analyse :

Exercice 1.1.3 (**). *Montrer l'existence d'une constante $c > 0$ telle que $\alpha(G) \cdot \omega(G) \geqslant c(\log |V(G)|)^2$ pour tout graphe G.*

Cependant, nous ne connaissons toujours pas la taille minimale d'un graphe contenant soit un stable soit une clique de taille 5 : nous savons seulement que le nombre de sommets est entre 43 et 48 [18, 1].

Exercice 1.1.4 (*). *Vérifier que tout graphe à au moins 6 sommets contient soit une clique soit un stable de taille 3, et qu'il existe un graphe à 5 ne contenant aucun des deux.*

Nous dirigeons le lecteur intéressé vers la notion de nombres de RAMSEY.

La méthode probabiliste

Les bornes présentées ci-dessus sont d'une grande naïveté tant dans leur valeur que dans les méthodes en jeu. Après tout, notre objectif initial était de démontrer quelque chose du type «$\alpha(G) \cdot \omega(G) \geqslant \sqrt{n}$».

Malheureusement, si cette borne est vérifiée par tout graphe auquel nous penserions spontanément, on peut aisément construire des contre-exemples via des arguments probabilistes.

Prenons un grand nombre de sommets, disons $n \geqslant 2^{40}$, et pour chaque paire indépendamment, plaçons une arête avec probabilité $1/2$. Quelle est la probabilité que le graphe G ainsi obtenu contienne un stable de taille k ? La probabilité qu'un ensemble de k sommets ne contienne aucune arête est $(\frac{1}{2})^{\binom{k}{2}}$, et le nombre de tels ensembles est $\binom{n}{k}$. Donc, la probabilité que le graphe contienne un stable de taille k est au plus $(\frac{1}{2})^{\binom{k}{2}} \cdot \binom{n}{k}$. Notons que cette probabilité est ridiculement faible lorsque $k = 3\log n$. Avec forte probabilité, nous avons donc $\alpha(G) \leqslant 3\log n$, et par symétrie $\omega(G) \leqslant 3\log n$. Ceci met un terme définitif à nos espoirs d'améliorer significativement les bornes naïves de la section précédente.

Cette méthode de construire un objet avec une part d'aléatoire est puissante. En tant qu'humains, nous sommes conditionnés à remarquer des motifs, et à en placer lorsque nous sommes amenés à créer. Un objet qui respecte certains motifs est souvent un objet qui se comporte bien vis-à-vis de propriétés de décompositions. La méthode probabiliste nous permet de contourner ces biais cognitifs. Elle peut être utilisée avec autant de succès pour démontrer des propriétés dans les graphes, par exemple pour obtenir une coloration avec peu de couleurs (par rapport au nombre maximum de voisins qu'un sommet peut avoir) dès lors que ω est constant. Nous dirigeons le lecteur intéressé vers l'excellent livre de MOLLOY et REED [27].

Petite digression avant de conclure cette section : l'idée d'éviter la structure est une idée clé ici, comme l'illustre la conjecture suivante (la notion de graphe sans H est définie formellement dans la section 1.1.3).

Conjecture 1.1.5 (ERDŐS-HAJNAL 1989 [17]). *Pour tout graphe H, il existe $\epsilon_H > 0$ tel que tout graphe G sans H satisfait $\alpha(G) \cdot \omega(G) \geqslant |V(G)|^{\epsilon_H}$.*

Cette conjecture reste grande ouverte, y compris pour des petits H, comme par exemple $H = C_5$, le cycle à cinq sommets.

Exercice 1.1.6 (**). *Montrer que la conjecture 1.1.5 est vraie pour H étant un stable de taille arbitraire.*

1.1.3 Quelques définitions

Nous complétons l'introduction avec quelques définitions qui nous seront nécessaires dans ce qui suit. Soit G un graphe. Pour un ensemble de sommets $S \subsetneqq V(G)$, on note $G \setminus S$ le graphe obtenu à partir de G en supprimant tous les sommets dans S ; si $S = \{v\}$, on écris parfois $G \setminus v$ au lieu de $G \setminus S$. Ensuite, pour un ensemble non-vide $X \subseteq V(G)$, on pose $G[X] = G \setminus (V(G) \setminus X)$. Tout graphe du type $G[X]$ pour un certain ensemble X est un *sous-graphe induit* de G. Pour $x_1, \ldots, x_t \in V(G)$, on écrit parfois $G[x_1, \ldots, x_t]$ au lieu de $G[\{x_1, \ldots, x_t\}]$.

Un graphe G est *connexe* si pour toute paire u, v de sommets, il existe une suite $w_1, \ldots, w_p$ de sommets (potentiellement $p = 1$) telle que $w_i w_{i+1} \in E(G)$ pour tout $1 \leqslant i \leqslant p-1$, ainsi que $w_1 = u$ et $w_p = v$. Une *composante connexe* d'un graphe G est un sous-graphe induit connexe maximal de G.

Deux graphes G_1, G_2 sont isomorphes s'il existe une bijection entre leurs sommets qui préserve l'adjacence. Par abus de langage tout au long de ce chapitre, nous disons qu'un graphe G_1 est le graphe G_2 si les deux sont isomorphes.

G est *sans H* si aucun sous-graphe induit de G n'est isomorphe à H. Inversement, G *contient H* si un de ses sous-graphes induits est isomorphe à H.

On dénote K_n le graphe complet à n sommets : le graphe à n sommets où toutes les arêtes possibles (il y en a $\binom{n}{2}$) sont présentes. L'étoile $K_{1,n}$ est le graphe à $n+1$ sommets et n arêtes, où un sommet est adjacent à tous les autres. Le chemin à n sommets P_n est le graphe à n sommets, que l'on peut considérer numérotés de 1 à n, tel que les arêtes sont précisément les couples de la forme $\{i, i+1\}$; les sommets numérotés 1 et n sont les *extrémités* du chemin. Pour $n \geqslant 3$, le cycle C_n à n sommets est le graphe obtenu à partir de P_n en ajoutant une arête entre les deux extrémités.

Si G est un graphe et $X, Y \subseteq V(G)$ sont disjoints, alors on dit que X est *complet* à Y dans G si tout sommet de X est adjacent à tous les sommets de Y, et on dit que X est *anti-complet* à Y dans G si aucun sommet de X n'est adjacent à un sommet de Y dans G. Par abus de notation, on dit qu'un sommet x est complet ou anti-complet à un ensemble $Y \subseteq V(G) \setminus \{x\}$ si $\{x\}$ l'est. Finalement, on dit que x est *mixte* sur Y dans G si x n'est ni complet ni anti-complet à Y dans G.

Le *complémentaire* d'un graphe G est le graphe $\overline{G}$ obtenu de G en échangeant les notions de sommets adjacents et non-adjacents. Formellement, le graphe $\overline{G}$ est défini par $V(\overline{G}) = V(G)$ et $E(\overline{G}) = \binom{V(G)}{2} \setminus E(G)$, où $\binom{V(G)}{2}$ est l'ensemble des paires de sommets.

Dans un graphe G, étant donné un sommet u, on dénote $N_G(u)$ l'ensemble des voisins de u dans G. On omet parfois l'indice lorsque le contexte ne laisse pas de doute quant au graphe concerné.

1.2 L'impact des petits cycles

Dans la section 1.1.2, nous avons construit[3] un graphe où $\chi(G) \geqslant 2^{\omega(G)/3}$. Que se passe-t-il lorsque $\omega(G) = 2$? Ou, plus généralement, lorsque le graphe ne contient pas de cycle de longueur k ou moins? (Rappelons qu'une clique de taille 3 est aussi un cycle de longueur 3, d'où cette généralisation).

La même méthode probabiliste évoquée dans la section 1.1.2 peut être utilisée pour construire, pour tous entiers g et k, un graphe où tout cycle a longueur au moins g, dont le nombre chromatique est au moins k. Il s'agit, en quelques mots, de mettre une arête entre deux sommets avec bien plus faible probabilité, de façon à pouvoir espérer obtenir très peu de cycles – cycles qui sont alors simplement éliminés du graphe. Avec le bon choix de probabilité, il reste suffisamment d'arêtes pour que le graphe ne puisse être coloré avec peu de couleurs. Soyons plus précis. La *maille* d'un graphe est la longueur du plus court de ses cycles (la maille d'une forêt est ∞). En utilisant la méthode probabiliste, ERDŐS [16] a démontré le théorème suivant.

Théorème 1.2.1. *[16] Pour tous entiers $g, k \geqslant 3$, il existe un graphe G ayant maille au moins g et nombre chromatique au moins k.*

Il existe donc des graphes ayant maille et nombre chromatique arbitrairement grands. Dans la section 1.2.1, nous présentons une famille

3. La vérification de cet énoncé est laissée en exercice au lecteur.

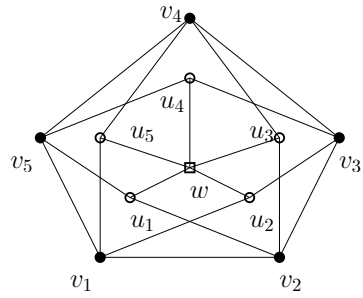


FIGURE *1.2 – Le graphe $M(C_5)$. Ce graphe est également appelé le* graphe de GRÖTZSCH.

concrète de graphes, sans triangle et pourtant de nombre chromatique arbitrairement grand.

1.2.1 Des graphes sans triangle et de nombre chromatique arbitrairement grand

Plusieurs constructions de tels graphes sont connues. Ici on donne la construction probablement la plus célèbre, celle de MYCIELSKI [28]. On considère d'autres constructions dans les exercices.

Étant donné un graphe G ayant l'ensemble des sommets $V(G) = \{v_1, \dots, v_n\}$, on construit le *Mycielskian* de G, dénoté par $M(G)$, de la manière suivante. L'ensemble des sommets de $M(G)$ est $V(M(G)) = \{v_1, \dots, v_n\} \cup \{u_1, \dots, u_n\} \cup \{w\}$. Ensuite, on a

— $M(G)[v_1, \dots, v_n] = G$;
— $\{u_1, \dots, u_n\}$ est un stable de $M(G)$;
— pour tout $i \in \{1, \dots, n\}$, u_i est non-adjacent à v_i dans $M(G)$;
— pour tous $i, j \in \{1, \dots, n\}$ avec $i \neq j$, u_i est adjacent à v_j dans $M(G)$ si et seulement si v_i est adjacent à v_j dans G ;
— pour tout $i \in \{1, \dots, n\}$, w est complet à $\{u_1, \dots, u_n\}$ et anti-complet à $\{v_1, \dots, v_n\}$ dans $M(G)$.

Pour un exemple, voir fig. 1.2.

Théorème 1.2.2. *[28] Tout graphe G vérifie $\omega(M(G)) = \max\{\omega(G), 2\}$ et $\chi(M(G)) = \chi(G) + 1$.*

Preuve. Soit G un graphe quelconque. On pose $V(G) = \{v_1, \dots, v_n\}$, et on utilise les notations de la définition du Mycielskian. Si $\omega(G) = 1$ (c'est-à-dire, si G est sans arêtes), alors il est clair que $\chi(G) = 1$ et $\omega(M(G)) = \chi(M(G)) = 2$: la preuve est complète. Désormais, on suppose que $\omega(G) \geqslant$

2, on pose $k := \chi(G)$. On démontre que $\omega(M(G)) = \omega(G)$ et $\chi(M(G)) = k+1$.

Nous démontrons d'abord que $\omega(M(G)) = \omega(G)$. Puisque G est sous-graphe induit de $M(G)$, il suffit de démontrer que $\omega(M(G)) \leqslant \omega(G)$. Soit K une clique de $M(G)$ de taille $\omega(M(G))$. Par construction, le voisinage de w dans $M(G)$ est un stable. Donc, si $w \in K$, alors $|K| \leqslant 2$, et on obtient $\omega(M(G)) = |K| \leqslant 2 \leqslant \omega(G)$. On suppose désormais que $w \notin K$. Puisque K est une clique et $\{u_1, \ldots, u_n\}$ est un stable, on a $|K \cap \{u_1, \ldots, u_n\}| \leqslant 1$. Si $K \cap \{u_1, \ldots, u_n\} = \emptyset$, alors K est une clique de G, et on a $\omega(M(G)) = |K| \leqslant \omega(G)$. Donc, on peut supposer que $|K \cap \{u_1, \ldots, u_n\}| = 1$, et l'on pose $K \cap \{u_1, \ldots, u_n\} = \{u_i\}$. Par construction, u_i est non-adjacent à v_i, et donc $v_i \notin K$. On déduit maintenant que $K^* := (K \setminus \{u_i\}) \cup \{v_i\}$ est une clique de G de même taille que K. Par conséquent, $\omega(M(G)) = |K| = |K^*| \leqslant \omega(G)$.

Nous démontrons maintenant que $\chi(M(G)) = k+1$. Il est clair que $M(G)$ est $(k+1)$-colorable : on colore d'abord $G = M(G)[v_1, \ldots, v_n]$ avec les couleurs $1, \ldots, k$, puis pour tout index $i \in \{1, \ldots, n\}$, on colore u_i par la même couleur que v_i, et finalement, on colore w par la couleur $k+1$. On vérifie facilement que ceci est une bonne coloration de $M(G)$.

Il reste à démontrer que $M(G)$ n'est pas k-colorable. Supposons par l'absurde qu'il existe une bonne coloration $c : V(M(G)) \to \{1, \ldots, k\}$ de $M(G)$. Par symétrie, on peut supposer que $c(w) = k$; puisque les voisins de w sont exactement $u_1, \ldots, u_n$, on voit que la couleur k n'est pas utilisée sur les sommets $u_1, \ldots, u_n$. Maintenant on définit $c' : V(G) \to \{1, \ldots, k-1\}$ en posant

$$c'(v_i) \quad = \quad \begin{cases} c(v_i) & \text{si} \quad c(v_i) \neq k \\ c(u_i) & \text{si} \quad c(v_i) = k \end{cases}$$

pour tout index $i \in \{1, \ldots, n\}$. On va démontrer que c' est une bonne coloration de G, contrairement au fait que G n'est pas $(k-1)$-colorable. Soient $i, j \in \{1, \ldots, n\}$ tels que $i \neq j$ et tels que v_i est adjacent à v_j ; il faut démontrer que $c'(v_i) \neq c'(v_j)$. Puisque c est une bonne coloration de $M(G)$, on voit que $c(v_i) \neq c(v_j)$. Par symétrie, on peut supposer que $c(v_i) \neq k$, et donc $c'(v_i) = c(v_i)$. Si $c(v_j) \neq k$, alors $c'(v_i) = c(v_i) \neq c(v_j) = c'(v_j)$, ce qu'il fallait démontrer. On suppose désormais que $c(v_j) = k$, et donc $c'(v_j) = c(u_j)$. Puisque v_i et v_j sont adjacents, v_i et u_j le sont aussi ; puisque c est une bonne coloration, il suit que $c(v_i) \neq c(u_j)$. Maintenant, on a $c'(v_i) = c(v_i) \neq c(u_j) = c'(v_j)$, ce qu'il fallait démontrer. Donc, c' est une bonne coloration de G, ce qui contredit le fait que G n'est pas $(k-1)$-colorable. ■

On construit maintenant la séquence $\{M_k\}_{k=2}^{\infty}$ des graphes de MYCIELSKI. On pose $M_2 := K_2$, et pour tout entier $k \geqslant 2$, on pose

$M_{k+1} = M(M_k)$. Nous remarquons que $M_3 = C_5$, et que M_4 est le graphe de Grötzsch (représenté dans la fig. 1.2).

Théorème 1.2.3. *[28] Pour tout entier $k \geqslant 2$, on a $\omega(M_k) = 2$ et $\chi(M_k) = k$. Donc, il existe des graphes sans triangle et de nombre chromatique arbitrairement grand.*

Preuve. Ceci est un corolaire immédiat du théorème 1.2.2 et de la construction des graphes $\{M_k\}_{k=2}^{\infty}$. ■

1.2.2 Exercices

Exercice 1.2.4 (*). *Démontrer que tout graphe sans triangle est sous-graphe induit d'un graphe de* MYCIELSKI. *Plus précisément, démontrer que pour tout graphe H sans triangle, il existe un entier $k \geqslant 2$ tel que H est isomorphe à un sous-graphe induit de M_k.*

Exercice 1.2.5 (**). *Dans cet exercice, nous considérons les graphes $\{Z_k\}_{k=1}^{\infty}$ construits par* ZYKOV *[38].*

Soit $Z_1 := K_1$. Ensuite, on suppose récursivement que les graphes $Z_1, \ldots, Z_k$ ont été construits, et l'on construit Z_{k+1} de manière suivante. D'abord, on construit l'union disjointe des graphes $Z_1, \ldots, Z_k$, et pour tout choix de sommets $v_1 \in V(Z_1), \ldots, v_k \in V(Z_k)$, on ajoute un sommet nouveau (correspondant à ce choix) et le relie aux sommets $v_1, \ldots, v_k$ (et à aucun autre sommet)[4]*. Le graphe résultant est Z_{k+1}.*

Démontrer que pour tout entier $k \geqslant 1$, le graphe Z_k est sans triangle et vérifie $\chi(Z_k) = k$.

Exercice 1.2.6 (**). *Dans cet exercice, nous considérons les graphes $\{G_k\}_{k=3}^{\infty}$ construits par Blanche* DESCARTES *[14].*

Soit G_3 un cycle impair quelconque de longueur au moins 5 (par exemple, $G_3 := C_7$). Ensuite, on suppose récursivement que le graphe G_k a été construit, et on construit G_{k+1} de la manière suivante. On pose $n_k := |V(G_k)|$. On construit l'union disjointe de $\binom{kn_k}{n_k}$ copies de G_k, on y ajoute kn_k sommets dits centraux *et deux-à-deux non-adjacents, on fixe une bijection entre les copies de G_k et les sous-ensembles des sommets centraux de taille n_k, et finalement, pour toute copie de G_k, on rajoute un couplage de taille n_k entre cette copie de G_k et l'ensemble des sommets centraux qui lui correspond (par notre bijection). Le graphe résultant est G_{k+1}.*

Démontrer que pour tout entier $k \geqslant 3$, G_k est sans triangle et vérifie $\chi(G_k) = k$.

4. Nous ajoutons donc $|V(Z_1)| \cdot \cdots \cdot |V(Z_k)|$ nouveaux sommets, et l'ensemble des sommets nouveaux est stable.

1.3 L'impact des cycles impairs et de leurs complémentaires

Dans la section 1.2, on a vu qu'interdire des cycles de longueur constante ne permettait pas de borner le nombre chromatique par une fonction de la taille d'une clique maximum. Cependant, les graphes sans cycle impair sont bipartis et donc trivialement 2-colorables. On s'intéresse ici à l'impact de l'interdiction de certaines parités de cycles, dans le graphe et dans son complémentaire.

1.3.1 Les graphes parfaits : $\chi = \omega$

Comme on a vu dans la section 1.2.1, la taille maximale d'une clique, borne inférieure triviale du nombre chromatique, est dans certains cas une borne assez mauvaise. En revanche, que peut-on dire de la structure des graphes dont le nombre chromatique est égal à cette borne inférieure triviale ? Malheureusement, on ne peut pas en dire grande chose : un graphe G vérifiant $\chi(G) = \omega(G)$ peut contenir n'importe quel graphe comme sous-graphe induit. En effet, si H est un graphe quelconque, et si G est l'union disjointe de H et du graphe complet à $\chi(H)$ sommets, alors G vérifie $\chi(G) = \omega(G)$.

Voici une définition plus féconde : un graphe G est *parfait* si tout sous-graphe induit H de G vérifie $\chi(H) = \omega(H)$. Un graphe est *imparfait* si il n'est pas parfait. Cela inspire la notion de classe de graphes «héréditaire» : une classe de graphes est *héréditaire* si tout sous-graphe induit d'un graphe de la classe appartient également à la classe. Évidemment, tout sous-graphe induit d'un graphe parfait est parfait, et donc la classe des graphes parfaits est héréditaire. Les graphes parfaits ont été introduits par BERGE [4, 5] dans les années 1960. La recherche autour des graphes parfaits a surtout été motivée par deux conjectures proposées par BERGE : la conjecture «faible» et la conjecture «forte» des graphes parfaits. Elles ont depuis été démontrées : la version faible en 1972 par LOVÁSZ [26], et la version forte en 2002 (publiée en 2006) par CHUDNOVSKY, ROBERTSON, SEYMOUR et THOMAS [10]. Nous citons les deux théorèmes ci-dessous.

Le théorème faible des graphes parfaits. *[26] Un graphe est parfait si et seulement si son complémentaire est parfait.*

Un *trou* est un cycle induit de longueur au moins 4. Un trou est *pair* ou *impair* selon la parité de sa longueur.

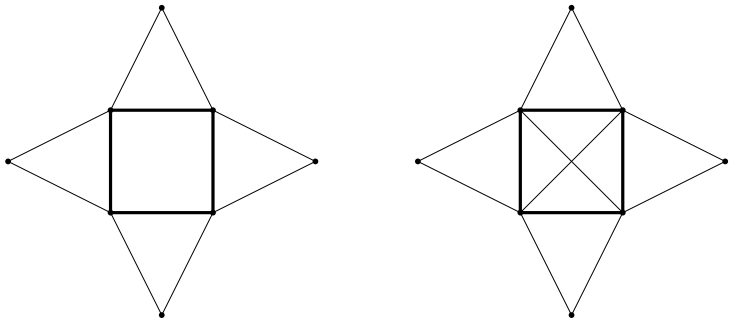
Le théorème fort des graphes parfaits. *[10] Un graphe est parfait si et seulement si ni lui ni son complémentaire ne contiennent de trou impair.*

Un graphe G est dit *de* BERGE si ni G ni son complémentaire $\overline{G}$ ne contiennent de trou impair. Donc, le théorème fort des graphes parfaits stipule qu'un graphe est parfait si et seulement si il est de BERGE. Évidemment, le complémentaire d'un graphe de BERGE est de BERGE ; le théorème fort implique immédiatement le théorème faible. Nous remarquons d'ailleurs que les graphes de BERGE sont reconnaissables en temps polynomial [8]. Par le théorème fort des graphes parfaits, les graphes parfaits le sont donc aussi.

Une direction du théorème fort des graphes parfaits est presque triviale : on vérifie facilement que les trous impairs et leurs complémentaires sont imparfaits (voir les exercices), donc tout graphe parfait est de BERGE. Or, la preuve de l'implication réciproque est d'une complexité immense (elle est longue de plus de 100 pages). Elle utilise la méthode dite «structurelle», et son ingrédient principal est un théorème de décomposition des graphes de BERGE qui déclare, essentiellement, que tout graphe de BERGE soit appartient à une classe «basique», soit admet une «décomposition» (nous reviendrons sur les notions de décompositions section 1.7). On démontre ensuite que les graphes appartenant aux classes basiques sont parfaits, ainsi qu'aucun contre-exemple minimum à la conjecture (c'est-à-dire, aucun graphe de BERGE et imparfait, ayant la plus petite taille parmi de tels graphes) n'admet de décomposition comme mentionnée dans le théorème de décomposition. Les détails sont hors de portée de ce chapitre ; le lecteur intéressé est renvoyé à la preuve du théorème fort des graphes parfaits [10] (pour un survol, voir [36]).

1.3.2 Le cas des graphes cordaux : tout cycle induit est un triangle

Ici, on considère un cas bien plus simple : celui des graphes «cordaux». Un graphe est dit *cordal* si il ne contient pas de trou (pour un exemple, voir fig. 1.3). Tout graphe cordal est de BERGE : le complémentaire de C_5 est isomorphe à C_5, et le complémentaire d'un cycle de longueur au moins six contient C_4 comme sous-graphe induit. Donc, le théorème fort des graphes parfaits implique immédiatement que les graphes cordaux sont parfaits. Ici, on donne une preuve directe (c'est-à-dire ne s'appuyant pas sur le théorème fort), datant des années 1960. Cette preuve utilise également l'approche structurelle : on démontre d'abord un théorème de décomposition déclarant que tout graphe cordal est «basique» ou bien admet une «décomposition» ; ensuite, on démontre que tous nos graphes basiques



graphe non-cordal graphe cordal

FIGURE 1.3 – *Un graphe non-cordal (à gauche) et un graphe cordal (à droite). Dans le graphe non-cordal, les arêtes d'un trou sont indiquées par des lignes en gras. Le cycle correspondant du graphe cordal n'est pas induit et donc n'est pas un trou.*

sont parfaits ; et finalement, on démontre qu'aucun graphe minimalement imparfait [5] n'admet notre décomposition.

On définit d'abord notre «décomposition». Un *ensemble d'articulation* d'un graphe G est un ensemble $C \subsetneqq V(G)$ tel que $G \setminus C$ n'est pas connexe ; une *clique d'articulation* de G est une clique de G qui est un ensemble d'articulation. (En particulier, si G n'est pas connexe, alors $\emptyset$ est une clique d'articulation de G.) On démontre maintenant un théorème de décomposition des graphes cordaux.

Théorème 1.3.1. *[15] Soit G un graphe cordal. Alors G est complet ou admet une clique d'articulation.*

Preuve. Nous supposons que G n'est pas complet, et nous démontrons qu'il admet une clique d'articulation. Évidemment, tout graphe non-complet a un ensemble d'articulation ; soit $C \subsetneqq V(G)$ un ensemble d'articulation minimal de G. Nous allons démontrer que C est une clique (et donc une clique d'articulation) de G.

Soient A et B les ensembles des sommets de deux composantes connexes distinctes de $G \setminus C$. Par la minimalité de C, tout sommet de C a un voisin dans A et un voisin dans B. Supposons par l'absurde que C n'est pas une clique, et soient $c_1, c_2 \in C$ distincts et non-adjacents. Les graphes $G[A \cup \{c_1, c_2\}]$ et $G[B \cup \{c_1, c_2\}]$ sont évidemment connexes ; soit

5. Un graphe G est *minimalement imparfait* si G est imparfait, mais tous les sous-graphes induits propres de G sont parfaits.

P_A un chemin induit de $G[A \cup \{c_1, c_2\}]$ entre c_1 et c_2, et soit P_B un chemin induit de $G[B \cup \{c_1, c_2\}]$ entre c_1 et c_2. Puisque c_1 et c_2 sont non-adjacents, et puisqu'il n'y a pas d'arêtes entre A et B, on voit que $G[V(P_A) \cup V(P_B)]$ est un trou de G, contrairement au fait que G est cordal. Donc, C est une clique, ce qu'il fallait démontrer. ■

Nous rappelons au lecteur qu'un graphe est dit *minimalement imparfait* si il est imparfait, mais tous ses sous-graphes induits propres sont parfaits.

Lemme 1.3.2. *[19] Aucun graphe minimalement imparfait ne contient de clique d'articulation.*

Preuve. Soit G un graphe tel que tout sous-graphe induit propre de G est parfait, et soit C une clique d'articulation de G. Il faut démontrer que G est parfait. Puisque tous les sous-graphes induits propres de G sont parfaits, il suffit de démontrer que $\chi(G) = \omega(G)$.

Soient $A_1, \ldots, A_t$ ($t \geqslant 2$) les ensembles des sommets des composantes connexes de $G \setminus C$. Pour tout $i \in \{1, \ldots, t\}$, soient $G_i = G[A_i \cup C]$, $\chi_i = \chi(G_i)$ et $\omega_i = \omega(G_i)$; par hypothèse, le graphe G_i est parfait, et donc $\chi_i = \omega_i$. Ensuite, pour tout $i \in \{1, \ldots, t\}$, soit $c_i : A_i \cup C \to \{1, \ldots, \chi_i\}$ une bonne coloration de G_i. Puisque C est une clique de G, pour tout $i \in \{1, \ldots, t\}$, la coloration c_i donne des couleurs différentes à tous les sommets de C. Quitte à permuter leurs couleurs, on peut supposer que les colorations $c_1, \cdots, c_t$ s'accordent sur les sommets de C (tout sommet u de C satisfait $c_1(u) = c_2(u) = \ldots = c_t(u)$). Maintenant $c := c_1 \cup \cdots \cup c_t$ est une bonne coloration de G, et évidemment, elle utilise seulement $\max\{\chi_1, \ldots, \chi_t\}$ couleurs. Donc, on a

$$\begin{array}{rcl} \chi(G) & \leqslant & \max\{\chi_1, \ldots, \chi_t\} \\ & = & \max\{\omega_1, \ldots, \omega_t\} \\ & \leqslant & \omega(G). \end{array}$$

Or, il est clair que $\chi(G) \geqslant \omega(G)$, et donc on a $\chi(G) = \omega(G)$. ■

Théorème 1.3.3. *[15] Les graphes cordaux sont parfaits.*

Preuve. Puisque la classe des graphes cordaux est héréditaire, il suffit de démontrer qu'aucun graphe cordal n'est minimalement imparfait. Évidemment, tout graphe complet est parfait, et par le lemme 1.3.2, aucun graphe minimalement imparfait n'admet de clique d'articulation. Donc, par le théorème 1.3.1, aucun graphe cordal n'est minimalement imparfait. ■

1.3.3 Exercices

Un *sommet simplicial* d'un graphe G est un sommet $v \in G$ tel que le voisinage de v dans G est une clique (éventuellement vide). Un *ordre d'élimination simplicial* de G est un ordre $v_1, \ldots, v_n$ des sommets de G tel que, pour tout $i \in \{1, \ldots, n\}$, v_i est simplicial dans le graphe $G[v_i, \ldots, v_n]$.

Exercice 1.3.4 (**). *Démontrer que tout graphe cordal contient un sommet simplicial. (Conseil : En utilisant le théorème 1.3.1, démontrer que tout graphe cordal non-complet contient deux sommets simpliciaux non-adjacents.)*

Exercice 1.3.5 (*). *Démontrer qu'un graphe est cordal si et seulement si il admet un ordre d'élimination simplicial. (Conseil : Exercice 1.3.4.)*

Exercice 1.3.6 (*). *Construire un algorithme de reconnaissance et de coloration des graphes cordaux en temps polynomial. (Conseil : Exercice 1.3.5.)*

Exercice 1.3.7 (**). *Démontrer que, si G est un graphe sans P_4 ayant au moins deux sommets, alors G ou son complémentaire $\overline{G}$ n'est pas connexe. Déduire que les graphes sans P_4 sont parfaits.*

1.4 Au-delà des graphes parfaits

Dans la section 1.3.1, nous avons considéré la classe (héréditaire) de graphes pour lesquels le nombre chromatique est toujours égal à la taille maximale d'une clique. C'est une propriété très forte, qui comme on l'a vu impose de grandes restrictions sur les sous-graphes induits possibles. On assouplit un peu cette propriété, comme suit.

1.4.1 Classes χ-bornées : généralisation des graphes parfaits

Au lieu de chercher l'égalité entre le nombre chromatique et sa borne inférieure triviale, on peut simplement chercher à ce que le premier soit borné quand le deuxième l'est. Plus formellement :

Définition 1.4.1. *Une classe $\mathcal{G}$ est* χ-bornée *s'il existe une fonction f telle que tout graphe $G \in \mathcal{G}$ satisfait $\chi(G) \leqslant f(\omega(G))$.*

En particulier, les graphes parfaits forment une classe χ-bornée, qui admet même la fonction identité comme fonction f telle qu'introduite dans la définition. Comme on l'a vu dans la section 1.2, les graphes sans petit cycle ne forment pas une classe χ-bornée. Nous remarquons que les graphes de grand nombre chromatique considérés dans la section 1.2 vérifient $\omega = 2$. Dans ce contexte, la conjecture suivante a souvent été proposée.

Conjecture 1.4.2 (ESPERET, non publiée [34, conjecture 12.1]). *Pour toute classe héréditaire $\mathcal{G}$, les deux énoncés suivants sont équivalents :*

(i) $\mathcal{G}$ est χ-bornée ;

(ii) il existe un entier $c > 0$ tel que tout graphe $G \in \mathcal{G}$ sans triangle est c-colorable.

Trivialement, (i) implique (ii), mais l'implication inverse est toujours ouverte.

1.4.2 L'impact des chemins et des étoiles

Comme premier exemple de classe χ-bornée contenant des graphes imparfaits, considérons la classe des graphes sans long chemin induit.

Théorème 1.4.3 (GYÁRFÁS [23]). *Pour tout entier $k \geqslant 2$, la classe des graphes sans P_k est χ-bornée par la fonction $f_k(n) = (k-1)^{n-1}$.*

Preuve. Puisque les graphes sans P_2 sont sans arêtes, le théorème est évidemment vrai pour $k = 2$. Soit donc $k \geqslant 3$ un entier quelconque. On va démontrer l'énoncé suivant, qui implique immédiatement le théorème : «tout graphe G soit vérifie $\chi(G) \leqslant (k-1)^{\omega(G)-1}$, soit contient un P_k induit». On fixe un graphe G quelconque, on pose $\omega := \omega(G)$, et l'on suppose par récursion que notre énoncé est vrai pour tout graphe où toute clique est de taille strictement inférieure à ω. Supposons que $\chi(G) \geqslant (k-1)^{\omega-1} + 1$; nous allons démontrer que G contient un P_k induit. Puisque $\chi(G) \geqslant 2$, on voit que $\omega \geqslant 2$.

On peut supposer que G est connexe (sinon, à la place de G, on considère la composante connexe de G ayant le plus grand nombre chromatique). Maintenant, soit x_1 un sommet quelconque de G. Alors

$$\chi(G \setminus x_1) \;\geqslant\; \chi(G) - 1 \;\geqslant\; (k-1)^{\omega-1} \;=\; (k-1)(k-1)^{\omega-2},$$

ce qui implique que G a une composante connexe ayant le nombre chromatique au moins $(k-1)(k-1)^{\omega-2}$. En outre, puisque G est connexe, x_1 a un voisin dans toute composante connexe de $G \setminus x_1$.

Maintenant, soit $m \in \{1, \ldots, k-1\}$ l'index maximal tel que G contient un chemin induit $x_1, \ldots, x_m$ et un sous-graphe induit H tels que :

(i) $x_1, \ldots, x_m \notin V(H)$,

(ii) $\{x_1, \ldots, x_{m-1}\}$ est anti-complet à $V(H)$, et x_m a un voisin dans $V(H)$,

(iii) H est connexe et vérifie $\chi(H) \geqslant (k-m)(k-1)^{\omega-2}$.

Pour tout voisin v de x_m dans $V(H)$, $x_1, \ldots, x_m, v$ est un chemin induit de G, et donc G contient un P_{m+1} induit. Donc, on peut supposer que $m \leqslant k-2$ (sinon, la preuve est complète).

On pose $N := N_G(x_m) \cap V(H)$. Par (ii), $N \neq \emptyset$. Ensuite, puisque x_m est complet à N, on voit que $\omega(G[N]) \leqslant \omega - 1$. Donc, si $\chi(G[N]) \geqslant (k-1)^{\omega-2}+1$, alors par récursion, $G[N]$ contient un P_k induit, et la preuve est complète. On suppose désormais que $\chi(G[N]) \leqslant (k-1)^{\omega-2}$, ce qui implique avec (iii) que

$$\chi(H \setminus N) \;\geqslant\; \chi(H) - \chi(G[N]) \;\geqslant\; (k-(m+1))(k-1)^{\omega-2}.$$

Soit H' la composante connexe de $H \setminus N$ ayant le nombre chromatique maximum [6]. Puisque H est connexe, il existe un sommet $x_{m+1} \in N$ ayant un voisin dans $V(H')$. Maintenant $x_1, \ldots, x_m, x_{m+1}$ et H' contredisent la maximalité de m. ■

Les graphes sans long chemin forment donc une classe χ-bornée. À l'opposé, on peut considérer les graphes sans grande étoile, i.e. les graphes où aucun sommet ne voit de grand stable dans son voisinage. On peut montrer assez simplement que de tels graphes forment également une classe χ-bornée, comme suit.

Exercice 1.4.4 (**). *Pour tout entier k, la classe des graphes sans étoile $K_{1,k}$ est χ-bornée.*

Rappelons en vue de l'exercice 1.4.4 que la notion de nombre de RAMSEY évoquée dans la section 1.1.2 stipule qu'un graphe sur lequel α comme ω ont une valeur bornée ne peut avoir qu'un petit nombre de sommets.

Puisqu'une classe est χ-bornée quand elle interdit un chemin, ou quand elle interdit une étoile, une généralisation naturelle passe par la notion d'arbre. Un arbre est un graphe connexe sans cycle ; un chemin et une étoile sont des cas particuliers d'arbres.

Conjecture 1.4.5 (GYÁRFÁS [22] SUMNER [35])**.** *Pour tout arbre T, la classe des graphes sans T est χ-bornée.*

Si les cas des étoiles ou des chemins admettent des preuves très élégantes et relativement élémentaires, la Conjecture 1.4.5 reste très largement ouverte.

6. Donc, $\chi(H') = \chi(H \setminus N) \geqslant (k-(m+1))(k-1)^{\omega-2}$.

1.5 L'impact des cycle longs ou des cycles impairs

Nous rappelons au lecteur qu'un graphe est dit *de* BERGE si ni lui ni son complémentaire ne contiennent de trou impair. Le théorème fort des graphes parfaits [10] (voir section 1.3.1) déclare qu'un graphe est parfait si et seulement si il est de BERGE ; donc, la classe des graphes de BERGE est χ-bornée par la fonction identité. Dans les années 1980, inspiré par ce qui était à l'époque toujours la conjecture forte des graphes parfaits, GYÁRFÁS [23] a proposé les trois conjectures suivantes.

Conjecture 1.5.1. *[23] La classe des graphes sans trou impair est χ-bornée.*

Conjecture 1.5.2. *[23] Pour tout entier $\ell \geqslant 4$, la classe des graphes sans trou de longueur au moins ℓ est χ-bornée.*

Conjecture 1.5.3. *[23] Pour tout entier $\ell \geqslant 4$, la classe des graphes sans trou impair de longueur au moins ℓ est χ-bornée.*

Évidemment, la troisième conjecture implique les deux précédentes, et en fait, les trois conjectures ont récemment été démontrées [33, 11, 12].

Les preuves des conjectures 1.5.1, 1.5.2 et 1.5.3 utilisent la «méthode de nivellement». L'idée principale est de prendre un graphe dont le nombre chromatique est grand par rapport à son nombre de clique (il est normalement facile de réduire le problème au cas où ce graphe est connexe), et de partitionner son ensemble des sommets selon la distance à un sommet fixe ; le «ℓ-ème niveau» est l'ensemble des sommets à distance ℓ de notre sommet fixe. Un de ces niveaux aura un grand nombre chromatique, et en manipulant ce niveau, ainsi que les chemins entre ce niveau et notre sommet fixe, on obtient un sous-graphe induit «interdit».

Les détails des preuves des conjectures 1.5.1, 1.5.2 et 1.5.3 sont hors de portée de ce chapitre. À leur place, on donne la preuve du théorème suivant de SCOTT.

Théorème 1.5.4. *[32] Pour tout entier $\ell \geqslant 4$, la classe des graphes sans trou impair et sans trou de longueur au moins ℓ est χ-bornée.*

Nous remarquons que le théorème 1.5.4 est un affaiblissement à la fois de la conjecture 1.5.1 et de la conjecture 1.5.2. Comme les preuves de ces deux conjectures, la preuve du théorème 1.5.4 utilise la méthode de nivellement, mais elle est plus courte et plus simple dans les détails.

On commence par quelques notations. D'abord, pour tout entier $\ell \geqslant 4$, nous dénoterons par $\mathcal{C}_\ell$ la classe de tous les graphes ne contenant ni de trou impair ni de trou de longueur au moins ℓ. (Donc, le théorème 1.5.4 déclare

que, pour tout entier $\ell \geqslant 4$, la classe $\mathcal{C}_\ell$ est χ-bornée.) Pour tout entier $n \geqslant 1$, on pose $[n] := \{1, \ldots, n\}$. Si G est un graphe et $V_0 \subseteq V(G)$, nous écrivons souvent $\chi(V_0)$ et $\omega(V_0)$ au lieu de $\chi(G[V_0])$ et $\omega(G[V_0])$, respectivement.

Pour un graphe G et des sommets $u, v \in V(G)$, la distance entre u et v dans G est dénotée par $d_G(u, v)$. (En particulier, si $u = v$, alors $d_G(u, v) = 0$.) Pour un graphe G, un sommet $x \in V(G)$, et un entier $d \geqslant 0$, la *boule de rayon d centrée en x* dans G est

$$B_G(x, d) = \{v \in V(G) \mid d_G(v, x) \leqslant d\},$$

et la *sphère de rayon d centrée en x* (ou bien le *d-ème niveau par rapport à x*) dans G est

$$S_G(x, d) = \{v \in V(G) \mid d_G(v, x) = d\},$$

Évidemment, si G est connexe et $x \in V(G)$, alors $V(G) = \bigcup_{d=0}^{\infty} S_G(x, d)$; en fait, puisque G est fini, il existe un entier $r \geqslant 0$ tel que les ensembles $S_G(x, r+1), S_G(x, r+2), S_G(x, r+3), \ldots$ sont vides, et donc $V(G) = \bigcup_{d=0}^{r} S_G(x, d)$. En outre, pour tout entier $d \geqslant 1$, tout sommet de $S_G(x, d)$ a un voisin dans $S_G(x, d-1)$, ce qui implique en particulier que tout sous-graphe de G induit par une boule centrée en x est connexe.

Lemme 1.5.5. *Soit G un graphe connexe et soit $x \in V(G)$. Alors il existe un entier $i \geqslant 0$ tel que $\chi(S_G(x, i)) \geqslant \frac{1}{2}\chi(G)$.*

Preuve. Exercice. ■

Pour un graphe G et un entier $\ell \geqslant 0$, le *nombre chromatique ℓ-local* de G est

$$\chi^{(\ell)}(G) = \max_{x \in V(G)} \chi(B_G(x, \ell)).$$

La preuve du théorème 1.5.4 est divisée en deux parties. Nous considérons d'abord le cas plus simple : celui des graphes ayant un «petit» nombre chromatique local (par rapport au nombre chromatique ; voir lemme 1.5.6). Après, nous considérons le cas plus compliqué : celui des graphes ayant un «grand» nombre chromatique local (par rapport à une constante fixe ; voir lemme 1.5.7). Finalement, nous démontrons le théorème 1.5.4 (le théorème principal de cette section) par récursion sur le nombre de clique, en utilisant les lemmes 1.5.6 et 1.5.7.

Lemme 1.5.6. *[32] Soit $\ell \geqslant 2$ un entier, et soit G un graphe tel que $\chi(G) > 2\chi^{(\ell)}(G)$. Alors G contient un trou de longueur au moins 2ℓ.*

Preuve. On peut supposer que G est connexe (sinon, à la place de G, on considère une composante connexe de G ayant le plus grand nombre chromatique). Soit $x \in V(G)$. Pour tout entier $j \geqslant 0$, on pose $S_j = S_G(x, j)$. Par le lemme 1.5.5, il existe un entier $i \geqslant 0$ tel que $\chi(S_i) \geqslant \frac{1}{2}\chi(G)$; donc, $\chi(S_i) > \chi^{(\ell)}(G)$, et par conséquent, $i > \ell$. Soit K une composante connexe de $G[S_i]$ telle que $\chi(K) > \chi^{(\ell)}(G)$.

Soit $v \in S_{i-1}$ un sommet quelconque ayant un voisin dans K. Puisque $\chi(K) > \chi^{(\ell)}(G)$, on voit que $V(K) \not\subseteq B_G(v, \ell)$; soit $y \in V(K) \setminus B_G(v, \ell)$, et soit $z \in S_{i-1}$ un voisin de y.

Maintenant, soit P_1 un chemin induit de $G[\{v, z\} \cup V(K)]$ entre v et z, et soit P_2 un chemin induit de $G[\{v, z\} \cup B_G(x, i-2)]$ entre v et z. Puisque z et y sont adjacents et $d_G(v, y) > \ell$, on voit que $d_G(v, z) \geqslant \ell$; donc, la longueur des chemins P_1 et P_2 est au moins ℓ. Puisqu'il n'y a pas d'arêtes entre $V(K) \subseteq S_i$ et $B_G(x, i-2)$, on déduit que $G[V(P_1) \cup V(P_2)]$ est un trou de longueur au moins 2ℓ. ■

Lemme 1.5.7. *[32] Soient $\ell, M \geqslant 2$ des entiers et soit G un graphe tel que $\chi^{(\ell)}(G) \geqslant M^{2^\ell}$. Alors au moins une de ces deux conditions est vérifiée :*

(i) G contient un trou impair de longueur au plus $2\ell + 1$;

(ii) G contient un sous-graphe induit H tel que $\omega(H) < \omega(G)$ et $\chi(H) > M$.

Preuve. On peut supposer que G est connexe (sinon, à la place de G, on considère une composante connexe de G ayant le plus grand nombre chromatique ℓ-local).

On pose $\omega = \omega(G)$. Puisque $\chi(G) \geqslant \chi^{(\ell)}(G) \geqslant M^{2^\ell} \geqslant 2$, on a $\omega \geqslant 2$. Par hypothèse, il existe un sommet $x \in V(G)$ tel que $\chi(B_G(x, \ell)) \geqslant M^{2^\ell}$. Afin de simplifier les notations, pour tout entier $j \geqslant 0$, on pose $S_j = S_G(x, j)$ (et donc, $S_j = S_{B_G(x,\ell)}(x, j)$). Par le lemme 1.5.5 (appliqué au graphe $B_G(x, \ell)$), il existe un index $j \in \{0, \ldots, \ell\}$ tel que

$$\chi(S_j) \geqslant \frac{1}{2}\chi(B_G(x, \ell)) \geqslant \frac{1}{2}M^{2^\ell} \geqslant M^{2^\ell - 1} > M^{2^{\ell-1}} \geqslant M^{2^{j-1}}.$$

Soit $i \in \{0, \ldots, \ell\}$ l'index minimal ayant la propriété que

$$\chi(S_i) > M^{2^{i-1}}.$$

Évidemment, $i \geqslant 1$[7]. Puisque x est adjacent à tous les sommets de S_1, on a $\omega(S_1) < \omega$. Donc, si $i = 1$, alors $H := G[S_i]$ vérifie (ii). On suppose désormais que $i \geqslant 2$.

7. En effet, $S_0 = \{x\}$, et donc $\chi(S_0) = 1 < \sqrt{M} = M^{2^{0-1}}$.

Notre objectif est de partitionner S_i de manière commode et d'utiliser cette partition soit pour trouver un court trou impair (comme dans (i)), soit pour trouver un sous-graphe induit de G vérifiant (ii). D'abord, nous partitionnons l'ensemble S_{i-1} en stables.

Par la minimalité de i, pour tout $j \in \{1, \dots, i-1\}$, on a $\chi(S_j) \leqslant M^{2^{j-1}}$, et l'on fixe une bonne coloration $\chi_j : S_j \to [M^{2^{j-1}}]$ de $G[S_j]$. Soit $C = \prod_{j=1}^{i-1} [M^{2^{j-1}}]$; alors $|C| = M^{2^{i-1}-1}$. Pour tout $c = (c_1, \dots, c_{i-1})$ dans C, un *c-chemin* est un chemin $x_1, \dots, x_{i-1}$ dans G tel que, pour tout $j \in \{1, \dots, i-1\}$, on a $x_j \in S_j$ et $\chi_j(x_j) = c_j$; un sommet $v \in S_{i-1}$ est *bon pour c* si il existe un c-chemin $x_1, \dots, x_{i-1}$ tel que $v = x_{i-1}$, sinon, v est *mauvais pour c*. Évidemment, pour tout $v \in S_{i-1}$, il existe un $c \in C$ tel que v est bon pour c.

Soit $<$ un ordre total strict quelconque sur C. Pour tout $c \in C$, on pose

$$X_c = \{v \in S_{i-1} \mid v \text{ est bon pour } c \text{ et mauvais pour tout } c' < c\}.$$

Par construction, $\{X_c\}_{c \in C}$ est une partition de S_{i-1} en stables [8] (potentiellement vides).

On construit maintenant une partition de S_i. Pour tout $c \in C$, soit Y_c l'ensemble de tous les sommets $v \in S_i$ vérifiant les deux conditions suivants :

— v a un voisin dans X_c ;

— pour tout $c' \in C$ tel que $c' < c$, v n'a pas de voisins dans $X_{c'}$.

Évidemment, $\{Y_c\}_{c \in C}$ est une partition de S_i (potentiellement, certains Y_c sont vides). Puisque $\chi(S_i) > M^{2^{i-1}}$ et $|C| = M^{2^{i-1}-1}$, il existe un $c = (c_1, \dots, c_{i-1})$ dans C tel que $\chi(Y_c) > M$. Si $\omega(Y_c) < \omega$, alors $H := G[Y_c]$ vérifie (ii). On suppose désormais que $\omega(Y_c) = \omega$. Soit K une clique de taille ω dans $G[Y_c]$.

Pour tout $j \in \{1, \dots, i-1\}$, soit V_j l'ensemble de tous les sommets $v \in S_j$ appartenant à un c-chemin (en particulier, $\chi_j(v) = c_j$). Alors V_{i-1} est exactement l'ensemble des sommets de S_{i-1} qui sont bons pour c (en particulier, V_{i-1} est un stable [9], et $X_c \subseteq V_{i-1}$), et donc par construction, tout sommet de Y_c a un voisin dans V_{i-1}. Puisque $\emptyset \neq K \subseteq Y_c$, on voit que $V_{i-1} \neq \emptyset$.

Maintenant, soit $u \in K$ tel que $|N_G(u) \cap V_{i-1}|$ est minimal, et soit $u' \in N_G(u) \cap V_{i-1}$. Alors il existe un sommet $w \in K$ non-adjacent à u' (sinon,

8. En effet, pour tout $c \in C$, χ_{i-1} colore tous les sommets de X_c par la même couleur, et χ_{i-1} est une bonne coloration de $G[S_{i-1}]$.

9. En effet, χ_{i-1} colore tous les sommets de V_{i-1} par la couleur c_{i-1}, et χ_{i-1} est une bonne coloration de $G[S_{i-1}]$.

$K \cup \{u'\}$ serait une clique de taille $\omega + 1$ dans G, une contradiction). Par la minimalité de $|N_G(u) \cap V_{i-1}|$, on voit qu'il existe un sommet $w' \in V_{i-1}$ adjacent à w et non-adjacent à u (en particulier, $u' \neq w'$, et puisque V_{i-1} est stable, u' et w' sont non-adjacents). Maintenant, soient $x, u_1, u_2, \ldots, u_{i-2}, u'$ et $x, w_1, w_2, \ldots, w_{i-2}, w'$ des chemins de G tels que $u_j, w_j \in V_j$ pour tout $j \in \{1, \ldots, i-2\}$, et soit $J := G[x, u_1, \ldots, u_{i-2}, u', w_1, \ldots, w_{i-2}, w']$; alors J est connexe et biparti, et les sommets u' et w' sont du même côté de la bipartition de J [10]. Soit P un chemin induit entre u' et w' dans J. Puisque J est biparti et u', w' sont du même côté de sa bipartition, la longueur de P est paire. En outre, $|V(J)| \leqslant 2i - 1 \leqslant 2\ell - 1$, et donc $|V(P)| \leqslant 2\ell - 1$. Maintenant, $G[V(P) \cup \{u, w\}]$ est un trou impair de longueur au plus $2\ell + 1$, et donc G vérifie (i). ■

On a maintenant tous les outils pour compléter la preuve du théorème 1.5.4.

Théorème 1.5.4. *[32] Pour tout entier $\ell \geqslant 4$, la classe des graphes sans trou impair et sans trou de longueur au moins ℓ est χ-bornée.*

Preuve. Soit $\ell \geqslant 4$ un entier. Il faut démontrer que la classe $\mathcal{C}_\ell$ est χ-bornée. On définit $f : \mathbb{N}^* \to \mathbb{N}^*$ récursivement de manière suivante :

— $f(1) := 1$;

— pour tout entier $n \geqslant 2$, $f(n) := 2f(n-1)^{2^\ell}$.

Nous allons démontrer que $\mathcal{C}_\ell$ est χ-bornée par f. Soit $G \in \mathcal{C}_\ell$ et $\omega = \omega(G)$; nous supposons par récursion que pour tout $G' \in \mathcal{C}_\ell$ tel que $\omega(G') < \omega$, on a $\chi(G') \leqslant f(\omega(G'))$. Il faut démontrer que $\chi(G) \leqslant f(\omega)$.

Supposons d'abord que $\omega \leqslant 2$. Alors G est sans triangle, et puisque G est également sans trou impair, G est biparti. Donc, $\chi(G) = \omega \leqslant f(\omega)$.

On suppose désormais que $\omega \geqslant 3$. Supposons par l'absurde que $\chi(G) > f(\omega)$; donc, $\chi(G) > 2f(\omega - 1)^{2^\ell}$. Par hypothèse, G n'a pas de trou de longueur au moins 2ℓ, et donc par le lemme 1.5.6, on a $\chi(G) \leqslant 2\chi^{(\ell)}(G)$, et l'on déduit que $\chi^{(\ell)}(G) > f(\omega - 1)^{2^\ell}$. Puisque G est sans trou impair, le lemme 1.5.7 implique que G a un sous-graphe induit, H, tel que $\omega(H) < \omega$ et $\chi(H) > f(\omega - 1)$. Mais ceci est impossible puisque, par récursion, $\chi(H) \leqslant f(\omega(H)) \leqslant f(\omega - 1)$. ■

Étant donné un graphe H, une *sous-division* de H, dénotée par H^*, est un graphe obtenu à partir de H en remplaçant chaque arête par un

10. En effet, si i est pair, alors les deux ensembles $\{x, u_2, w_2, u_4, w_4, \ldots, u_{i-2}, w_{i-2}\}$ et $\{u_1, w_1, u_3, w_3, \ldots, u_{i-3}, w_{i-3}, u', w'\}$ forment une bipartition de J, et si i est impair, alors $\{x, u_2, w_2, u_4, w_4, \ldots, u_{i-3}, w_{i-3}, u', w'\}$ $\{u_1, w_1, u_3, w_3, \ldots, u_{i-2}, w_{i-2}\}$) forment une bipartition de J. On remarque qu'il est possible que $u_j = w_j$ pour certaines valeurs de j.

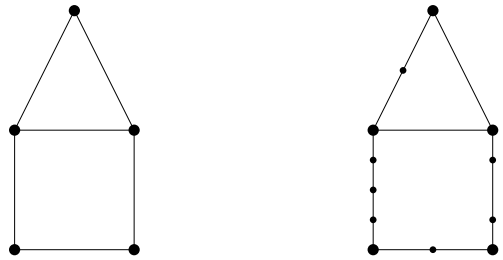


FIGURE *1.4 – Un graphe H et une sous-division H^* de H.*

chemin induit ayant au moins une arête (voir fig. 1.4 pour un exemple ; nous remarquons que H^* n'est pas unique, et que H lui-même est un H^*). Ensuite, pour tout graphe H, Forb(H) est la classe de tous les graphes ne contenant pas de H comme sous-graphe induit, et Forb$^*(H)$ est la classe de tous les graphes ne contenant pas de H^* comme sous-graphe induit (c'est-à-dire, ne contenant aucune sous-division de H comme sous-graphe induit) ; évidemment, Forb$^*(H) \subseteq$ Forb(H).

La méthode de nivellement (utilisée pour démontrer le théorème 1.5.4, ainsi que pour démontrer les conjectures 1.5.1, 1.5.2 et 1.5.3) a été introduite par SCOTT dans les années 1990 pour démontrer le théorème suivant.

Théorème 1.5.8. *[31] Pour tout arbre T, la classe Forb$^*(T)$ est χ-bornée.*

Le théorème 1.5.8 est un affaiblissement de la conjecture 1.4.5 que l'on peut reformuler comme suit.

Conjecture 1.5.9 (GYÁRFÁS [22] SUMNER [35])**.** *Pour tout arbre T, la classe Forb(T) est χ-bornée.*

La conjecture 1.5.9 est probablement la plus célèbre conjecture ouverte concernant les classes χ-bornées. En outre, inspiré par le théorème 1.5.8, SCOTT a conjecturé que, pour tout graphe H, la classe Forb$^*(H)$ est χ-bornée [31][11]. Or, cette conjecture est fausse : si K est un graphe non-planaire quelconque, et si H est obtenu en sous-divisant toutes les arêtes de K deux fois (c'est-à-dire, en remplaçant toutes les arêtes de K par des chemins induits à trois arêtes), alors H est un contrexemple à la conjecture [29].

Comme l'on a vu, la méthode de nivellement de SCOTT a été assez féconde en théorie des classes χ-bornées. L'avantage principal de cette

11. Nous remarquons que le théorème 1.5.4 implique que la classe Forb(C_5) est χ-bornée, et donc le graphe C_5 vérifie la conjecture de SCOTT.

méthode est le fait que les preuves qui l'utilisent se généralisent facilement, c'est-à-dire, la même preuve fonctionne souvent pour toute une série de classes [12]. Par contre, la faiblesse principale de la méthode est le fait qu'elle produit des bornes au moins exponentielles, qui sont donc potentiellement très loin d'être optimales. La raison pour ceci est le fait que (comme dans le cas du théorème 1.5.4) ces preuves procèdent par récursion sur le nombre de clique, en utilisant le lemme 1.5.5. Donc, quand on construit récursivement la borne f, à chaque étape, on multiplie au moins par 2 (c'est-à-dire, on a $f(\omega+1) \geqslant 2f(\omega)$ pour tout ω), et donc la fonction f ainsi obtenue est au moins exponentielle.

Dans la section 1.7, on discutera d'une méthode dont les avantages et les désavantages vont dans le sens inverse : les preuves qui l'utilisent sont assez difficiles à généraliser, mais elles produisent souvent de bonnes bornes (parfois optimales). Il s'agit de la méthode structurelle, dont on a déjà discuté dans le contexte des graphes parfaits (voir section 1.3.1). Dans la section 1.7, on discute de cette méthode dans le contexte des classes χ-bornées en général.

1.5.1 Exercices

Exercice 1.5.10 (*). *Démontrer le lemme 1.5.5.*

Exercice 1.5.11 (*). *En utilisant le théorème 1.5.8, démontrer que, pour toute forêt F, la classe $Forb(F)$ est χ-bornée.*

Exercice 1.5.12 (*). *Caractériser les graphes H vérifiant $Forb^*(H) = Forb(H)$.*

1.6 Le rêve des bornes polynomiales

Toutes les preuves évoquées ci-dessus (dans les sections 1.4 et 1.5) garantissent l'existence d'une fonction permettant de borner le nombre chromatique par rapport à la taille maximale d'une clique. Cependant, une analyse plus précise des preuves donne des fonctions assez peu attrayantes - tour simple d'exponentielle dans le cas des graphes sans long chemin ou sans étoile (voir section 1.4.2), tours plus vertigineuses dans le cas des preuves subséquentes (voir discussion en fin de section 1.5). De telles hauteurs sont-elles nécessaires ?

12. Par exemple, la preuve du théorème 1.5.4 établit que, pour tout entier $\ell \geqslant 4$, la classe $\mathcal{C}_\ell$ est χ-bornée. Nous remarquons que $\mathcal{C}_4 \subsetneqq \mathcal{C}_6 \subsetneqq \mathcal{C}_8 \subsetneqq \mathcal{C}_{10} \subsetneqq \dots$ (pourtant, $\mathcal{C}_{2i-1} = \mathcal{C}_{2i}$ pour tout entier $i \geqslant 3$).

1.6.1 Une conjecture folle

Un cadre formel pour la notion de classes χ-bornées par de petites fonctions passe par la notion de borne polynomiale.

Définition 1.6.1. *Une classe $\mathcal{G}$ est* polynomialement χ-bornée *s'il existe un polynôme P tel que tout graphe $G \in \mathcal{G}$ satisfait $\chi(G) \leqslant P(\omega(G))$.*

Peu de classes de graphes sont à ce jour connues pour être polynomialement χ-bornées, même si les graphes parfaits le sont trivialement. Pour mettre en lumière la faiblesse de nos outils en matière de bornes inférieures, et mobiliser les efforts dans cette direction, Louis ESPERET a proposé la conjecture suivante, parfois qualifiée de «défi» ou de «provocation» par des chercheurs de communauté :

Conjecture 1.6.2 (ESPERET [24])**.** *Toute classe de graphes χ-bornée est polynomialement χ-bornée.*

1.6.2 Le cas hantant des chemins

Pour souligner l'étendue de notre ignorance, contrastons la simplicité de la preuve de la section 1.4.2 avec ce qui est connu lorsque l'on cherche non pas simplement une borne, mais une borne polynomiale.

Notons qu'un graphe sans P_4 a une structure simple (voir exercice 1.3.7) qui implique, en particulier, que les graphes sans P_4 sont parfaits. La classe des graphes sans P_4 est donc polynomialement χ-bornée.

Considérons maintenant la classe des graphes sans P_5. Un graphe sans P_5 peut contenir un trou de longueur cinq ; tous les graphes de cette classe ne sont pas parfaits. Pour autant, plusieurs théorèmes structurels sont connus pour les graphes sans P_5 [6, 25]. Malgré tout, la question de si cette classe est polynomialement χ-bornée reste ouverte.

1.7 Les théorèmes de décomposition

Dans cette section, on revient à la méthode dont on a déjà discuté dans la section 1.3.1 : la méthode structurelle. Supposons que l'on veut démontrer qu'une classe héréditaire, $\mathcal{G}$, est χ-bornée. Comme dans le cas des graphes parfaits, on démontre d'abord un «théorème de décomposition» pour la classe $\mathcal{G}$ déclarant (en gros) que tout graphe $G \in \mathcal{G}$ est «basique» ou admet une «décomposition». Dans le cas le plus simple (si la décomposition qui apparait dans le théorème de décomposition est suffisamment commode), on fixe une fonction $f : \mathbb{N}^* \to \mathbb{N}^*$ bien choisie, on démontre que tout

graphe basique, H, vérifie $\chi(H) \leqslant f(\omega(H))$, et on démontre également que, pour tout graphe G admettant notre décomposition, si tout sous-graphe induit propre G' de G vérifie $\chi(G') \leqslant f(\omega(G'))$, alors G vérifie $\chi(G) \leqslant f(\omega(G))$. Ceci démontre que la classe $\mathcal{G}$ est χ-bornée par f.

L'avantage principal de la méthode structurelle dans le contexte des classes χ-bornées est le fait qu'elle produit souvent d'assez bonnes bornes (normalement polynomiales ou même linéaires, parfois optimales), ce qui n'est pas le cas avec d'autres méthodes qui ont été utilisées dans ce contexte. Son désavantage principal est le fait que les théorèmes de décomposition sont souvent très difficiles à généraliser. Par exemple, on vérifie facilement que tout graphe dans la classe $\text{Forb}(P_3)$ [13] est complet ou non-connexe [14]. Le théorème de décomposition pour la classe $\text{Forb}(P_4)$ est plus complexe (cf. Exercice 1.3.7), mais toujours relativement simple, et l'on peut l'utiliser pour démontrer que tout graphe sans P_4 est parfait, ce qui implique en particulier que la classe $\text{Forb}(P_4)$ est χ-bornée par la fonction identité. Par contre, il n'y a pas de théorème de décomposition comparablement fort pour la classe $\text{Forb}(P_5)$; cette classe est χ-bornée, mais la meilleure borne que l'on connait est exponentielle [21], ce qui est peut-être très loin de la borne optimale.

Un autre désavantage de la méthode structurelle est le fait que, pour la plupart des décompositions D que l'on rencontre dans des théorèmes de décomposition, ceci est normalement faux (et reste faux même si l'on impose quelques restrictions raisonnables) : «si G admet D, et si tout sous-graphe induit propre G' de G vérifie $\chi(G') \leqslant f(\omega(G'))$, alors G vérifie $\chi(G) \leqslant f(\omega(G))$». Dans de telles circonstances, il est parfois possible de s'en sortir en trouvant une propriété P qui «simule» la propriété d'être χ-bornée, et qui est «préservée» par la décomposition D (voir par exemple [30]). Pourtant, dans cette section, on ne considère que des exemples relativement simples, où de tels problèmes n'apparaissent pas. On considère deux classes : celle des graphes sans sous-divisions de la «patte» (voir section 1.7.1), et celle des graphes sans «configurations de TRUEMPER» (voir section 1.7.2). Nous remarquons finalement que, si H est un graphe tel que la classe $\text{Forb}(H)$ est χ-bornée par une fonction polynomiale, alors le graphe H vérifie la conjecture d'ERDŐS-HAJNAL [17] (voir les exercices), ce qui est une autre motivation pour trouver de bonnes bornes pour des classes χ-bornées.

13. Nous utilisons les notations introduites dans la section 1.5. Pour tout graphe H, $\text{Forb}(H)$ est la classe de tout les graphes ne contenant pas de H comme sous-graphe induit.

14. On déduit qu'un graphe G est sans P_3 si et seulement si il est l'union disjointe de graphes complets.

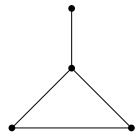


FIGURE *1.5 – La patte.*

1.7.1 Les graphes sans sous-division induite de la patte

La *patte* est le graphe à quatre sommets et quatre arêtes représenté dans la fig. 1.5. Dans cette section, nous démontrons un théorème de décomposition pour la classe Forb*(patte) [15], et nous l'utilisons pour démontrer que cette classe est χ-bornée. Un graphe G est *multiparti complet* si il existe une partition $(S_1, \dots, S_t)$ de $V(G)$ en stables non-vides tels que pour tous $i, j \in \{1, \dots, t\}$ tels que $i \neq j$, S_i est complet à S_j dans G ; dans de telles circonstances, on appelle $(S_1, \dots, S_t)$ une *multipartition* du graphe multiparti complet G.

Théorème 1.7.1. *[9] Soit G un graphe. Alors les deux énoncés suivants sont équivalents :*

(i) $G \in$ Forb*(patte) ;

(ii) toute composante connexe de G est un arbre, un cycle, ou un graphe multiparti complet.

Preuve. Il est facile de vérifier que (ii) implique (i). Nous supposons désormais que G vérifie (i), et nous démontrons que G vérifie (ii). Puisque Forb*(patte) est héréditaire, toutes les composantes connexes de G sont dans Forb*(patte) ; on peut donc supposer que G est connexe. On peut encore supposer que G contient un cycle induit : sinon, G est un arbre et la preuve est complète. Soit ℓ la maille (c'est-à-dire, la longueur du plus court cycle) de G.

Supposons d'abord que $\ell \geqslant 5$. Soit $c_1, c_2, \dots, c_\ell, c_1$ un cycle induit dans G. Si $V(G) = \{c_1, c_2, \dots, c_\ell\}$, alors G est un cycle, et la preuve est complète. Supposons donc que $\{c_1, c_2, \dots, c_\ell\} \subsetneqq V(G)$. Puisque G est connexe, il existe un sommet $v \in V(G) \setminus \{c_1, c_2, \dots, c_\ell\}$ ayant un voisin dans $\{c_1, c_2, \dots, c_\ell\}$. Par symétrie, on peut supposer que v est adjacent à c_1. Soit $i \in \{1, 2, \dots, \ell\}$ maximal tel que v est adjacent à c_i. Si $i = 1$, alors $G[v, c_1, c_2, \dots, c_\ell]$ est une patte*, une contradiction. Donc, $i \geqslant 2$, d'où

15. Ici, nous utilisons les notations introduites dans la section 1.5. En particulier, une *patte** est toute sous-division de la patte, et Forb*(patte) est la classe de tous les graphes ne contenant aucune patte* comme sous-graphe induit.

l'on déduit que $v, c_1, c_2, \ldots, c_i, v$ et $v, c_i, \ldots, c_\ell, c_1, v$ sont des cycles (pas forcément induits) de G de longueur $i+1$ et $\ell - i + 3$, respectivement. Puisque la maille de G est ℓ, on voit que $i + 1 \geqslant \ell$ et $\ell - i + 3 \geqslant \ell$, et donc $\ell - 1 \leqslant i \leqslant 3$, contrairement au fait que $\ell \geqslant 5$.

On suppose désormais que $3 \leqslant \ell \leqslant 4$; alors G contient un triangle ou un C_4 induit. Soit H un sous-graphe induit maximal de G ayant la propriété que H est un graphe multiparti complet contenant un cycle [16], et soit $(S_1, \ldots, S_t)$ une multipartition de H. Puisque H contient un cycle, on voit que $t \geqslant 2$. On peut supposer que $V(H) \subsetneqq V(G)$: sinon, G est multiparti complet, et la preuve est complète. Maintenant, puisque G est connexe, il existe un sommet $v \in V(G) \setminus V(H)$ ayant un voisin dans $V(H)$. Si v est complet à $V(H)$, alors $G[V(H) \cup \{v\}]$ est multiparti complet avec multipartition $(\{v\}, S_1, \ldots, S_t)$, contrairement à la maximalité de H. Supposons maintenant que v est mixte sur un des ensembles $S_1, \ldots, S_t$; par symétrie, on peut supposer que v est mixte sur S_1. Soient $s_1, s_1' \in S_1$ tels que v est adjacent à s_1 et non-adjacent à s_1'. Alors v est anti-complet à $S_2 \cup \cdots \cup S_t$: en effet, si v était adjacent à un sommet $s \in S_2 \cup \cdots \cup S_t$, alors $G[v, s, s_1, s_1']$ serait une patte, une contradiction. Puisque H contient un cycle et S_1 est un stable, on voit que $|S_2 \cup \cdots \cup S_t| \geqslant 2$; soient s, s' des sommets distincts de $S_2 \cup \cdots \cup S_t$. Si s, s' sont adjacents, alors $G[v, s_1, s, s']$ est une patte, et si s, s' sont non-adjacents, alors $G[v, s_1, s_1', s, s']$ est une patte*, une contradiction dans les deux cas.

On a donc démontré que pour tout $i \in \{1, \ldots, t\}$, v est soit complet, soit anti-complet à S_i. Puisque v a un voisin dans $V(H)$, mais n'est pas complet à $V(H)$, on peut supposer par symétrie que v est anti-complet à S_1 et complet à S_2. Si v est complet à $S_3 \cup \cdots \cup S_t$ [17], alors $G[V(H) \cup \{v\}]$ est multiparti complet avec multipartition $(S_1 \cup \{v\}, S_2, \ldots, S_t)$, contrairement à la maximalité de H. Donc, v a un non-voisin $s \in S_3 \cup \cdots \cup S_t$. Soient $s_1 \in S_1$ et $s_2 \in S_2$. Alors $G[v, s_1, s_2, s]$ est une patte, une contradiction. ■

Comme d'habitude, une fonction $f : \mathbb{N}^* \to \mathbb{N}^*$ est dite *croissante* si, pour tous $m, n \in \mathbb{N}^*$ tels que $m \leqslant n$, on a $f(m) \leqslant f(n)$.

Corollaire 1.7.2. *[9] La classe Forb*(patte) est χ-bornée par la fonction $f : \mathbb{N}^* \to \mathbb{N}^*$ définie par $f(2) = 3$ et $f(n) = n$ pour tout $n \neq 2$.*

Preuve. On remarque d'abord que, si H est un cycle, alors on a $\omega(G) = 2$ et $\chi(H) \leqslant 3$, et donc $\chi(H) \leqslant f(\omega(H))$. Ensuite, si H est un arbre ou un graphe multiparti complet, alors $\chi(H) = \omega(H) \leqslant f(\omega(H))$.

16. Un tel H existe parce que K_3 et C_4 sont des graphes multipartis complets contenant un cycle.

17. Si $t = 2$, alors $S_3 \cup \cdots \cup S_t = \emptyset$, et v est trivialement complet à $S_3 \cup \cdots \cup S_t$.

Maintenant, soit $G \in \text{Forb}^*(\text{patte})$, et soient $G_1, \ldots, G_t$ les composantes connexes de G. Par le théorème 1.7.1, pour tout $i \in \{1, \ldots, t\}$, G_i est un arbre, un cycle ou un graphe multiparti complet, et donc $\chi(G_i) \leqslant f(\omega(G_i))$. Puisque f est croissante, il suit que

$$\begin{aligned} \chi(G) &= \max\{\chi(G_1), \ldots, \chi(G_t)\} \\ &\leqslant \max\{f(\omega(G_1)), \ldots, f(\omega(G_t))\} \\ &= f(\max\{\omega(G_1), \ldots, \omega(G_t)\}) \\ &= f(\omega(G)), \end{aligned}$$

ce qu'il fallait démontrer. ■

Nous remarquons que la fonction f du corollaire 1.7.2 est en fait une borne optimale : pour $\omega = 2$, on considère les trous impairs, et pour $\omega \neq 2$, on considère les graphes complets.

1.7.2 Les graphes sans configurations de TRUEMPER

Les graphes *thêta*, *pyramide* et *prisme* sont représentés dans la fig. 1.6. On appelle *roue* tout graphe formé en ajoutant un sommet à un trou et en reliant ce sommet à au moins trois sommets de ce trou (voir fig. 1.7). Les *configurations de* TRUEMPER sont les thêtas, les pyramides, les prismes et les roues. Nous remarquons que toute configuration de TRUEMPER contient un trou, et donc les graphes cordaux sont sans configurations de TRUEMPER. En outre, tout pyramide contient un trou impair, et donc tout graphe sans trou impair est sans pyramide. En particulier, tout graphe de BERGE est sans pyramide, et en fait, les pyramides ont joué un rôle important dans l'algorithme de la reconnaissance des graphes de BERGE en temps polynomial [8]. Par contre, tout thêta ou prisme contient un trou pair, et donc tout graphe sans trou pair est sans thêta et sans prisme. Nous remarquons également que les roues ont joué un rôle important dans la preuve du théorème fort des graphes parfaits [10]. Pour un survol de classes définies en interdisant certaines (pas forcément toutes) configurations de TRUEMPER, nous renvoyons le lecteur vers [37].

Dans cette section, nous considérons la classe de tous les graphes ne contenant aucune configuration de TRUEMPER (comme sous-graphe induit). On appelle cette classe $\mathcal{T}$. D'abord, nous énonçons le théorème de décomposition pour la classe $\mathcal{T}$ démontré dans [13] ; la preuve est relativement longue, et nous l'omettons. Nous rappelons au lecteur qu'un *ensemble d'articulation* d'un graphe G est un ensemble $C \subsetneqq V(G)$ tel que $G \setminus C$ n'est pas connexe ; une *clique d'articulation* de G est une clique de G qui est un ensemble d'articulation. (En particulier, si G n'est pas connexe, alors $\emptyset$ est une clique d'articulation de G.)

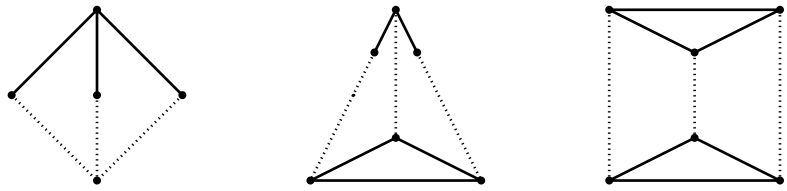


FIGURE 1.6 – *Les graphes thêta, pyramide et prisme. Les lignes pleines représentent des arêtes, et les lignes pointillées représentent des chemins ayant au moins une arête.*

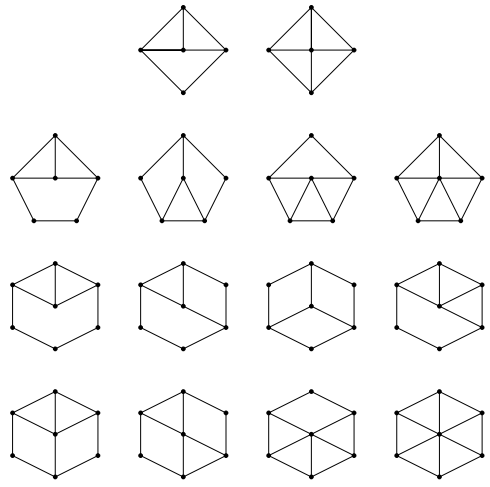


FIGURE 1.7 – *Quelques petites roues.*

Théorème 1.7.3. *[13] Soit $G \in \mathcal{T}$. Alors l'une de ces trois conditions est satisfaite :*

- *G est un graphe complet ;*
- *G est un cycle ;*
- *G admet une clique d'articulation.*

Le lemme suivant déclare concrètement que la propriété d'être χ-bornée par une fonction croissante fixe est «préservée» par les cliques d'articulation. Ce lemme généralise le lemme 1.3.2, qui déclare que la perfection est préservée par les cliques d'articulation.

Lemme 1.7.4. *Soit $\mathcal{G}$ une classe héréditaire et $f : \mathbb{N}^* \to \mathbb{N}^*$ une fonction croissante. Supposons que tout graphe $G \in \mathcal{G}$ vérifie $\chi(G) \leqslant f(\omega(G))$, ou bien admet une clique d'articulation. Alors $\mathcal{G}$ est χ-bornée par f.*

Preuve. Soit $G \in \mathcal{G}$. Nous supposons par récursion que tout graphe $G' \in \mathcal{G}$ ayant moins de $|V(G)|$ sommets vérifie $\chi(G') \leqslant f(\omega(G'))$. Il faut démontrer que $\chi(G) \leqslant f(\omega(G))$. On peut supposer que G admet une clique d'articulation (sinon, la preuve est complète par hypothèse). Soit donc C une clique d'articulation de G, et soient $A_1, \ldots, A_t$ ($t \geqslant 2$) les ensembles des sommets des composantes connexes de $G \setminus C$. Pour tout $i \in \{1, \ldots, t\}$, on pose $G_i = G[A_i \cup C]$; par récursion, on a $\chi(G_i) \leqslant f(\omega(G_i))$. Pour tout $i \in \{1, \ldots, t\}$, soit $c_i : A_i \cup C \to \{1, \ldots, f(\omega(G_i))\}$ une bonne coloration de G_i. Puisque C est une clique, pour tout $i \in \{1, \ldots, t\}$, c_i donne des couleurs différentes à tous les sommets de C. Donc, on peut supposer que $c_1 \restriction C = \cdots = c_t \restriction C$ (si nécessaire, nous permutons les couleurs). Maintenant $c := c_1 \cup \cdots \cup c_t$ est une bonne coloration de G, et évidemment, elle utilise seulement $\max\{f(\omega(G_1)), \ldots, f(\omega(G_t))\}$ couleurs. Puisque f est croissante, on déduit que

$$\begin{aligned} \chi(G) &\leqslant \max\{f(\omega(G_1)), \ldots, f(\omega(G_t))\} \\ &= f(\max\{\omega(G_1), \ldots, \omega(G_t)\}) \\ &\leqslant f(\omega(G)), \end{aligned}$$

ce qu'il fallait démontrer. ■

Corollaire 1.7.5. *La classe $\mathcal{T}$ est χ-bornée par la fonction $f : \mathbb{N}^* \to \mathbb{N}^*$ définie par $f(2) = 3$ et $f(n) = n$ pour tout $n \neq 2$.*

Preuve. Ceci est un corolaire immédiat du théorème 1.7.3 et du lemme 1.7.4. ■

La fonction f du corollaire 1.7.5 est en fait une borne optimale : pour $\omega = 2$, on considère les trous impairs, et pour $\omega \neq 2$, on considère les graphes complets.

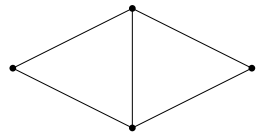


FIGURE *1.8 – Le diamant.*

1.7.3 Exercices

Le *diamant* est le graphe représenté dans la fig. 1.8. Un *sommet d'articulation* d'un graphe G est un sommet $v \in V(G)$ tel que $G \setminus v$ n'est pas connexe.

Exercice 1.7.6 (*). *En utilisant le théorème 1.7.3, démontrer que, pour tout graphe $G \in Forb^*(diamant)$, l'une de ces 4 conditions est satisfaite :*

- *— G est un cycle ;*
- *— G est un graphe complet ;*
- *— G a un sommet d'articulation ;*
- *— G n'est pas connexe.*

Démontrer ensuite que la classe $Forb^*(diamant)$ est χ-bornée. Quelle est la borne optimale ?

Exercice 1.7.7 (**). *Sans utiliser le théorème 1.7.3, démontrer le théorème de décomposition pour les graphes sans diamant énoncé dans l'exercice 1.7.6.*

Exercice 1.7.8 (*). *Soit H un graphe tel que la classe $Forb(H)$ est χ-bornée par une fonction polynomiale. Démontrer que H vérifie la conjecture d'*ERDŐS-HAJNAL*, c'est-à-dire, qu'il existe une constante $c > 0$ telle que tout graphe $G \in Forb(H)$ vérifie $\max\{\omega(H), \alpha(H)\} \geqslant |V(G)|^c$.*

Bibliographie

[1] V. ANGELTVEIT et B. D. MCKAY : $r(5,5) \leqslant 48$. *arXiv preprint arXiv :1703.08768*, 2017.

[2] K. APPEL et W. HAKEN : Every planar map is four colorable. *Bulletin of the American mathematical Society*, 82(5):711–712, 1976.

[3] K. APPEL, W. HAKEN et J. KOCH : Every planar map is four colorable. Part II : Reducibility. *Illinois Journal of Mathematics*, 21(3):491–567, 1977.

[4] C. BERGE : Les problemes de coloration en théorie des graphes. *Publications de l'Institut de statistique de l'Université de Paris*, 9:123–160, 1960.

[5] C. BERGE : Färbung von graphen, deren sämtliche bzw. deren ungerade kreise starr sind. *Wissenschaftliche Zeitschrift*, 10:114–115, 1961.

[6] E. CAMBY et O. SCHAUDT : A new characterization of P_k-free graphs. *Algorithmica*, 75(1):205–217, 2016.

[7] N. CHIBA, T. NISHIZEKI et N. SAITO : A linear 5-coloring algorithm of planar graphs. *Journal of Algorithms*, 2(4):317–327, 1981.

[8] M. CHUDNOVSKY, G. CORNUÉJOLS, X. LIU, P. SEYMOUR et K. VUŠKOVIĆ : Recognizing Berge graphs. *Combinatorica*, 25(2):143–186, 2005.

[9] M. CHUDNOVSKY, I. PENEV, A. D. SCOTT et N. TROTIGNON : Excluding induced subdivisions of the bull and related graphs. *Journal of Graph Theory*, 71(1):49–68, 2012.

[10] M. CHUDNOVSKY, N. ROBERTSON, P. SEYMOUR et R. THOMAS : The strong perfect graph theorem. *Annals of mathematics*, pp. 51–229, 2006.

[11] M. CHUDNOVSKY, A. D. SCOTT et P. SEYMOUR : Induced subgraphs of graphs with large chromatic number. III. Long holes. *Combinatorica*, 37(6):1057–1072, 2017.

[12] M. CHUDNOVSKY, A. D. SCOTT, P. SEYMOUR et S. SPIRKL : Induced subgraphs of graphs with large chromatic number. VIII. Long odd holes. *Journal of Combinatorial Theory, Series B*, 140:84–97, 2020.

[13] M. CONFORTI, G. CORNUÉJOLS, A. KAPOOR et K. VUŠKOVIĆ : Universally signable graphs. *Combinatorica*, 17(1):67–77, 1997.

[14] B. DESCARTES : Solution to advanced problem no. 4526. *American Mathematical Monthly*, 61:352, 1954.

[15] G. A. DIRAC : On rigid circuit graphs. *Actes de Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, vol. 25, pp. 71–76. Springer, 1961.

[16] P. ERDŐS : Graph theory and probability. *Canadian Journal of Mathematics*, 11:34–38, 1959.

[17] P. ERDŐS et A. HAJNAL : RAMSEY-type theorems. *Discrete Applied Mathematics*, 25(1-2):37–52, 1989.

[18] G. EXOO : A lower bound for r (5, 5). *Journal of graph theory*, 13(1):97–98, 1989.

[19] T. GALLAI : Graphen mit triangulierbaren ungeraden Vielecken. *A Magyar Tudományos Akadémia – Matematikai Kutató Intézetének Közleményei*, 7:3–36, 1962.

[20] M. R. GAREY et D. S. JOHNSON : *Computers and intractability*, vol. 29. W. H. Freeman New York, 2002.

[21] S. GRAVIER, C. T. HOÀNG et F. MAFFRAY : Coloring the hypergraph of maximal cliques of a graph with no long path. *Discrete mathematics*, 272(2-3):285–290, 2003.

[22] A. GYÁRFÁS : On RAMSEY covering-numbers. *Infinite and Finite Sets*, 2:801–816, 1975.

[23] A. GYÁRFÁS : Problems from the world surrounding perfect graphs. *Applicationes Mathematicae*, 19(3-4):413–441, 1987.

[24] T. KARTHICK et F. MAFFRAY : Vizing bound for the chromatic number on some graph classes. *Graphs and Combinatorics*, 32(4):1447–1460, 2016.

[25] D. LOKSHTANOV, M. VATSHELLE et Y. VILLANGER : Independent sand in p_5-free graphs in polynomial time. *Actes de Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 570–581. Society for Industrial and Applied Mathematics, 2014.

[26] L. LOVÁSZ : Normal hypergraphs and the perfect graph conjecture. *Discrete Mathematics*, 2(3):253–267, 1972.

[27] M. MOLLOY et B. REED : *Graph colouring and the probabilistic method*, vol. 23. Springer Science & Business Media, 2013.

[28] J. MYCIELSKI : Sur le coloriage des graphes. *Actes de Colloq. Math*, vol. 3, p. 9, 1955.

[29] A. PAWLIK, J. KOZIK, T. KRAWCZYK, M. LASOŃ, P. MICEK, W. T. TROTTER et B. WALCZAK : Triangle-free intersection graphs of line segments with large chromatic number. *Journal of Combinatorial Theory, Series B*, 105:6–10, 2014.

[30] I. PENEV : Amalgams and χ-boundedness. *Journal of Graph Theory*, 84(1):57–92, 2017.

[31] A. D. SCOTT : Induced trees in graphs of large chromatic number. *Journal of Graph Theory*, 24(4):297–311, 1997.

[32] A. D. SCOTT : Induced cycles and chromatic number. *Journal of Combinatorial Theory, Series B*, 76(2):150–154, 1999.

[33] A. D. SCOTT et P. SEYMOUR : Induced subgraphs of graphs with large chromatic number. I. Odd holes. *Journal of Combinatorial Theory, Series B*, 121:68–84, 2016.

[34] A. D. SCOTT et P. SEYMOUR : A survey of χ-boundedness. *arXiv preprint arXiv :1812.07500*, 2018.

[35] D. P. SUMNER : Subtrees of a graph and chromatic number. *The theory and applications of graphs*, pp. 557–576, 1981.

[36] N. TROTIGNON : Perfect graphs : a survey. *Topics in Chromatic Graph Theory*, pp. 137–160, 2015.

[37] K. VUŠKOVIĆ : The world of hereditary graph classes viewed through truemper configurations. *Surveys in Combinatorics 2013*, 409:265–325, 2013.

[38] A. A. ZYKOV : De certaines propriétés de complexes linéaires (en russe). *Matematicheskii sbornik*, 66(2):163–188, 1949.

Chapitre 2

Accessibilité des systèmes d'addition de vecteurs

Jérôme LEROUX
Sylvain SCHMITZ

Ce chapitre est consacré aux systèmes d'addition de vecteurs, un formalisme équivalent aux réseaux de PETRI *et employé pour raisonner sur des ressources discrètes : par exemple des processus en calcul parallèle ou distribué, des molécules dans des réactions chimiques, des organismes dans des processus biologiques, etc. De plus, de nombreuses questions algorithmiques en informatique théorique et mathématiques discrètes se ramènent à des questions sur les systèmes d'addition de vecteurs, et en particulier à leur problème d'accessibilité. Celui-ci est décidable, mais avec un coût algorithmique très élevé : il est (au moins) non-élémentaire et (au plus) ackermannien.*

Ce chapitre présente les derniers résultats connus à ce jour concernant les bornes de complexité pour le problème d'accessibilité dans les systèmes d'addition de vecteurs. C'est aussi l'occasion de donner un aperçu rapide de plusieurs outils mathématiques utilisés en informatique théorique tels que les systèmes d'équations linéaires ou les ordinaux, et de classes de complexité au-delà des classes habituelles comme P, NP *ou* EXP.

2.1 Introduction

Un *système d'addition de vecteurs avec états* (SAVE) est un automate fini pondéré par des vecteurs d'entiers, comme celui de la figure 2.1. La séman-

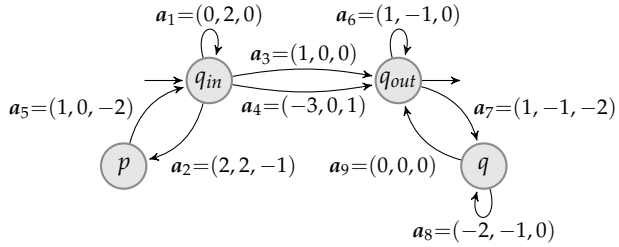


FIGURE *2.1 – Un système d'addition de vecteurs avec états.*

tique d'un tel système, en partant d'un vecteur d'entiers naturels initial, additionne composante par composante les vecteurs qui pondèrent les transitions empruntées, mais à chaque étape, les valeurs obtenues doivent être supérieures ou égales à zéro sur toutes les composantes. Par exemple, dans le système de la figure 2.1 et en partant d'un vecteur initial $(0,0,2)$, une exécution possible du système pourrait être

$$q_{in}(0,0,2) \xrightarrow{a_1} q_{in}(0,2,2) \xrightarrow{a_1} q_{in}(0,4,2) \xrightarrow{a_3} q_{out}(1,4,2) \xrightarrow{a_6} q_{out}(2,3,2) \xrightarrow{a_7} q(3,2,0) \xrightarrow{a_8} q(1,1,0) \xrightarrow{a_9} q_{out}(1,1,0)\ .$$

En revanche, $q(1,1,0) \xrightarrow{a_8} q(-1,0,0)$ n'est pas une étape d'exécution valide à cause de la valeur négative sur la première composante. L'exécution ci-dessus montre qu'il est possible d'atteindre $q_{out}(1,1,0)$ à partir de $q_{in}(0,0,2)$. Le *problème d'accessibilité* demande justement pour un SAVE où l'on a distingué un état initial q_{in} et un état final q_{out} et deux vecteurs d'entiers naturels $\boldsymbol{c}_{in}$ et $\boldsymbol{c}_{out}$ si l'on peut atteindre $q_{out}(\boldsymbol{c}_{out})$ à partir de $q_{in}(\boldsymbol{c}_{in})$.

2.1.1 Petit historique

Du fait de ses nombreuses applications (voir par exemple [32, sec. 5] pour un aperçu des problèmes de décision inter-réductibles), le problème d'accessibilité dans les systèmes d'addition de vecteurs a été l'objet d'une littérature abondante, dont la figure 2.2 donne un très bref aperçu.

Formalismes. L'étude des systèmes d'addition de vecteurs commence dans les années 1960 avec la définition des réseaux de PETRI dans la thèse de PETRI [27] et celle des systèmes d'addition de vecteurs (SAV) par KARP et MILLER en 1969 [11]. Les systèmes d'addition de vecteurs avec états (SAVE) comme celui de la figure 2.1 apparaissent dans les années 1970 notablement dans les travaux de GREIBACH [7] et d'HOPCROFT et PANSIOT [10]. Nous verrons les liens entre ces différents formalismes dans la section 2.2.

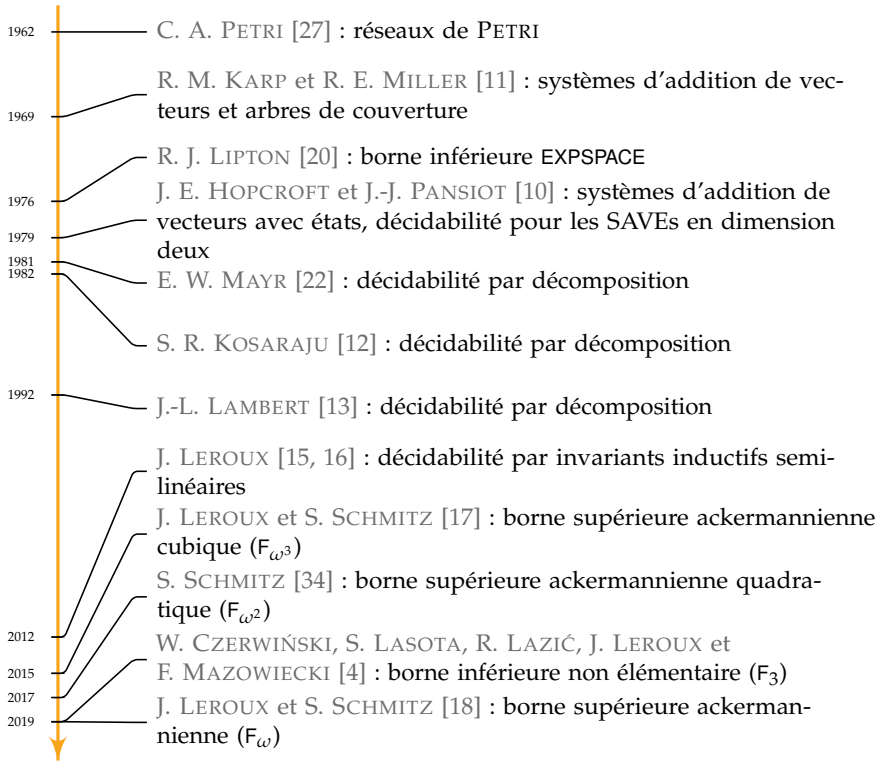


FIGURE 2.2 – *Aperçu historique sur l'accessibilité dans les systèmes d'addition de vecteurs.*

Décidabilité. Le problème d'accessibilité est alors considéré comme une question ouverte centrale [24, 9], pour laquelle la décidabilité n'est connue que dans des cas particuliers [36, 10]. La première preuve de décidabilité est due à MAYR en 1981 [22] et s'appuie sur un *algorithme par décomposition* qui sera simplifié à deux reprises par KOSARAJU en 1982 [12] et par LAMBERT en 1992 [13]. Cet algorithme et sa preuve de correction sont difficiles à présenter en quelques pages, aussi nous n'en donnerons que les idées principales dans la section 2.3, mais des présentations plus fouillées peuvent être trouvées dans [23, 30, 14].

Bornes supérieures. La sophistication de l'algorithme par décomposition rend son analyse assez ardue, et la première borne de complexité connue apparaît en 2015 dans [17]. Cette borne de complexité « ackermannienne cubique » est extrêmement élevée, et nécessite d'utiliser des classes de complexité inhabituelles (voir la figure 2.3) définies à partir de variantes de

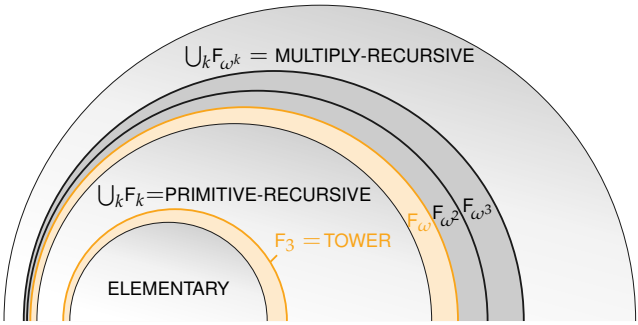


FIGURE 2.3 – *Classes de complexité non-élémentaires [33]. Actuellement, le problème d'accessibilité est connu comme étant* F_3*-difficile [4] et dans* F_ω *[18].*

la fonction d'ACKERMANN. Nous présentons succinctement ces classes de complexité ainsi que les principaux arguments de l'analyse de complexité plus récente de [18] dans la section 2.4.

Bornes inférieures. LIPTON [20] a démontré en 1976 que le *problème de couverture*, un problème qui se réduit à l'accessibilité, est difficile pour EXPSPACE. Pour ce problème de couverture, c'est en fait optimal [29], mais il a fallu attendre jusqu'en 2019 avant qu'une borne inférieure plus précise soit démontrée dans [4] : le problème d'accessibilité nécessite au moins une tour d'exponentielle dont la hauteur dépend de l'entrée du problème ; plus précisément, il est TOWER-difficile, aussi noté F_3 dans la figure 2.3. Cette construction est présentée dans la section 2.5.

2.2 Systèmes d'addition de vecteurs *et cætera*

2.2.1 Systèmes d'addition de vecteurs

Un système d'addition de vecteurs (SAV) [11] de dimension $d \in \mathbb{N}$ est un ensemble fini $\boldsymbol{A} \subseteq \mathbb{Z}^d$ de vecteurs appelés *actions*.

La sémantique opérationnelle d'un SAV est définie sur un ensemble de *configurations* dans $\mathbb{N}^d_\omega \stackrel{\text{def}}{=} (\mathbb{N} \uplus \omega)^d$, où ω dénote un élément plus grand que tous les entiers naturels. Une configuration dans $\mathbb{N}^d$ est dite *finie*. Nous associons à une action $\boldsymbol{a} \in \boldsymbol{A}$ la relation binaire $\xrightarrow{\boldsymbol{a}}$ sur les configurations définie par $\boldsymbol{x} \xrightarrow{\boldsymbol{a}} \boldsymbol{y}$ si $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{a}$, où l'addition est définie composante par composante avec la convention que $\omega + z = \omega$ pour tout $z \in \mathbb{Z}$. De manière cruciale, cette relation n'est définie qu'entre configurations dans $\mathbb{N}^d_\omega$; par

exemple, $(1,\omega) \xrightarrow{(-1,-5)} (0,\omega)$ mais $(1,\omega)$ n'est en relation avec aucun $\boldsymbol{y}$ de $\mathbb{N}_\omega^2$ par la relation $\xrightarrow{(-2,-5)}$.

Étant donné un mot fini $\sigma = \boldsymbol{a}_1 \dots \boldsymbol{a}_k \in \boldsymbol{A}^*$ d'actions, nous définissons aussi la relation binaire $\xrightarrow{\sigma}$ sur les configurations par $\boldsymbol{x} \xrightarrow{\sigma} \boldsymbol{y}$ s'il existe une suite $\boldsymbol{c}_0, \dots, \boldsymbol{c}_k$ de configurations telle que

$$\boldsymbol{x} = \boldsymbol{c}_0 \xrightarrow{\boldsymbol{a}_1} \boldsymbol{c}_1 \cdots \xrightarrow{\boldsymbol{a}_k} \boldsymbol{c}_k = \boldsymbol{y} \ .$$

On note aussi $\boldsymbol{x} \xrightarrow{*} \boldsymbol{y}$ s'il existe un mot d'actions σ tel que $\boldsymbol{x} \xrightarrow{\sigma} \boldsymbol{y}$.

Le problème de décision qui nous intéresse dans ce chapitre est le suivant.

Problème 2.2.1 (accessibilité des SAVs)**.**

entrée *Un SAV* $\boldsymbol{A} \subseteq \mathbb{Z}^d$ *dimension* d *et deux configurations finies* $\boldsymbol{c}_{in}, \boldsymbol{c}_{out} \in \mathbb{N}^d$.

question *Avons-nous* $\boldsymbol{c}_{in} \xrightarrow{*} \boldsymbol{c}_{out}$ *?*

Exemple 2.2.2. *Posons* $\boldsymbol{A}_{\text{ex}} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3, \boldsymbol{a}_4, \boldsymbol{a}_5, \boldsymbol{a}_6, \boldsymbol{a}_7, \boldsymbol{a}_8, \boldsymbol{a}_9\}$ *où les vecteurs* $\boldsymbol{a}_i$ *sont ceux indiqués dans la figure 2.1 :*

$$\begin{array}{lll} \boldsymbol{a}_1 = (0,2,0), & \boldsymbol{a}_2 = (2,2-1), & \boldsymbol{a}_3 = (1,0,0), \\ \boldsymbol{a}_4 = (-3,0,1), & \boldsymbol{a}_5 = (1,0,-2), & \boldsymbol{a}_6 = (1,-1,0), \\ \boldsymbol{a}_7 = (1,-1,-2), & \boldsymbol{a}_8 = (-2,-1,0), & \boldsymbol{a}_9 = (0,0,0). \end{array}$$

On a par exemple $(0,0,2) \xrightarrow{\boldsymbol{a}_1\boldsymbol{a}_1\boldsymbol{a}_3\boldsymbol{a}_6\boldsymbol{a}_7\boldsymbol{a}_8\boldsymbol{a}_9} (1,1,0)$*, mais aussi* $(0,0,2) \xrightarrow{\boldsymbol{a}_1\boldsymbol{a}_7} (1,1,0)$.

2.2.2 Systèmes d'addition de vecteurs avec états

Les systèmes d'addition de vecteurs ont une définition très simple, mais il est souvent plus pratique de travailler sur des systèmes dotés d'états de contrôle. Un *système d'addition de vecteurs avec états* (SAVE) [10] de dimension $d \in \mathbb{N}$ est un triplet $G = (Q, q_{in}, q_{out}, T)$ où Q est un ensemble fini non vide d'états, $q_{in} \in Q$ est l'état d'entrée, $q_{out} \in Q$ est l'état de sortie, et T est un ensemble fini de transitions dans $Q \times \mathbb{Z}^d \times Q$; $\boldsymbol{A} \stackrel{\text{def}}{=} \{\boldsymbol{a} \mid \exists p, q \in Q \,.\, (p, \boldsymbol{a}, q) \in T\}$ est l'ensemble d'actions associé.

Exemple 2.2.3. *La figure 2.1 représente le SAVE* $G_{\text{ex}} = (Q_{\text{ex}}, q_{in}, q_{out}, T_{\text{ex}})$ *de dimension* 3 *où* $Q_{\text{ex}} = \{q_{in}, q_{out}, p, q\}$ *et* $T_{\text{ex}} = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9\}$ *avec*

$$\begin{array}{lll} t_1 = (q_{in}, (0,2,0), q_{in}), & t_2 = (q_{in}, (2,2,-1), p), & t_3 = (q_{in}, (1,0,0), q_{out}), \\ t_4 = (q_{in}, (-3,0,1), q_{out}), & t_5 = (p, (1,0,-2), q_{in}), & t_6 = (q_{out}, (1,-1,0), q_{out}), \\ t_7 = (q_{out}, (1,-1,-2), q), & t_8 = (q, (-2,-1,0), q), & t_9 = (q, (0,0,0), q_{out}). \end{array}$$

Son ensemble d'actions associé est l'ensemble A_{ex} *de l'exemple 2.2.2.*

Un *état-configuration* d'un SAVE $G = (Q, q_{in}, q_{out}, T)$ est une paire $(q, \boldsymbol{x}) \in Q \times \mathbb{N}^d_\omega$, notée $q(\boldsymbol{x})$ dans la suite. Étant donnée une action $\boldsymbol{a}$, nous définissons la relation en un pas $\xrightarrow[G]{\boldsymbol{a}}$ sur les états-configurations par $p(\boldsymbol{x}) \xrightarrow[G]{\boldsymbol{a}} q(\boldsymbol{y})$ si $(p, \boldsymbol{a}, q) \in T$ et $\boldsymbol{x} \xrightarrow{\boldsymbol{a}} \boldsymbol{y}$. Par extension, pour un mot σ d'actions $\sigma = \boldsymbol{a}_1 \dots \boldsymbol{a}_k$, $p(\boldsymbol{x}) \xrightarrow[G]{\sigma} q(\boldsymbol{y})$ s'il existe une suite $q_0(\boldsymbol{c}_0), \dots, q_k(\boldsymbol{c}_k)$ d'états-configurations telle que

$$p(\boldsymbol{x}) = q_0(\boldsymbol{c}_0) \xrightarrow[G]{\boldsymbol{a}_1} q_1(\boldsymbol{c}_1) \cdots \xrightarrow[G]{\boldsymbol{a}_k} q_k(\boldsymbol{c}_k) = q(\boldsymbol{y}).$$

Ainsi, dans l'exemple 2.2.3, $q_{in}(0,0,2) \xrightarrow[G_{\text{ex}}]{\boldsymbol{a}_1\boldsymbol{a}_1\boldsymbol{a}_3\boldsymbol{a}_6\boldsymbol{a}_7\boldsymbol{a}_8\boldsymbol{a}_9} q_{out}(1,1,0)$. Finalement, nous écrivons $p(\boldsymbol{x}) \xrightarrow[G]{*} q(\boldsymbol{y})$ s'il existe $\sigma \in \boldsymbol{A}^*$ tel que $p(\boldsymbol{x}) \xrightarrow[G]{\sigma} q(\boldsymbol{y})$.

Accessibilité. Le problème de l'accessibilité pour les SAVs a son pendant pour les SAVEs.

Problème 2.2.4 (accessibilité des SAVEs)**.**

entrée *Un SAVE $G = (Q, q_{in}, q_{out}, T)$ de dimension d et deux configurations finies $\boldsymbol{c}_{in}, \boldsymbol{c}_{out} \in \mathbb{N}^d$.*

question *Avons-nous $q_{in}(\boldsymbol{c}_{in}) \xrightarrow[G]{*} q_{out}(\boldsymbol{c}_{out})$?*

Le problème d'accessibilité des SAVs se réduit au problème de l'accessibilité des SAVEs. En effet, étant donnés un SAV $\boldsymbol{A}$ et deux configurations finies $\boldsymbol{c}_{in}, \boldsymbol{c}_{out}$, il suffit de considérer le problème d'accessibilité des SAVEs avec l'entrée $(\{q\}, q, q, \{q\} \times \boldsymbol{A} \times \{q\})$ et les mêmes configurations $\boldsymbol{c}_{in}, \boldsymbol{c}_{out}$. Une réduction dans l'autre sens est possible en augmentant la dimension de trois et en codant les états dans ces nouvelles composantes [10] ; cette réduction ne sera pas abordée dans ce chapitre.

2.2.3 Programmes à compteurs

Pour montrer que les problèmes d'accessibilité des SAVs et des SAVEs nécessitent une tour d'exponentielles en temps et en espace, nous avons besoin de faire des transformations sur les systèmes. Pour simplifier la présentation, plutôt que de travailler directement sur les SAVEs, nous préférons introduire un langage de programmation impératif qui travaille sur des variables appelées *compteurs*, qui sont évaluées sur les entiers naturels.

Concrètement, un programme à compteurs est une suite de lignes d'instructions dont la sémantique est paramétrée par un entier natu-

rel B, et où chaque instruction est d'une des six formes suivantes :

$x \mathrel{+}= 1$	(incrémenter le compteur x)
$x \mathrel{-}= 1$	(décrémenter le compteur x)
goto L **or** L'	(branchement non déterministe à la ligne L ou L')
zero ? x	(continuer si le compteur x est égal à 0)
max ? x	(continuer si le compteur x est égal à B)
halt if $x_1, \ldots, x_\ell = 0$	(terminer si tous les compteurs $x_1, \ldots, x_\ell$ égalent 0)

Pour définir nos constructions et nos preuves, nous utiliserons deux types de compteurs :

1. Les compteurs *testés* sont bornés par un entier positif B et peuvent être testés par « **zero ?** x » et « **max ?** x » pour l'égalité aux valeurs extrêmes de leurs intervalles de définition, à savoir 0 et B ;
2. Les compteurs *non testés* sont non bornés et les instructions de test « **zero ?** x » et « **max ?** x » ne peuvent s'appliquer à eux – mais l'instruction **halt if** $x_1, \ldots, x_\ell = 0$ le peut.

Notons que les deux types de compteurs (testés ou non testés) ne sont pas déclarés explicitement : sans perte de généralité, un compteur x est considéré comme testé (et donc limité à des valeurs dans l'intervalle $\{0, \ldots, B\}$) si et seulement s'il apparaît dans une instruction « **zero ?** x » ou « **max ?** x » du programme. On utilisera l'abréviation « **halt** » lorsqu'aucun compteur n'est exigé d'être nul à la fin de l'exécution.

Remarquons que nous pourrions également étendre le modèle des SAVEs en permettant à des compteurs bornés par un entier B de pouvoir tester la valeur de ces compteurs à une valeur donnée sans modifier le pouvoir d'expressivité du modèle. En effet, il suffirait d'encoder la valeur de ces compteurs dans les états du SAVE. Cependant, un tel encodage énumératif implique une explosion du nombre d'états du modèle.

Branchement conditionnel. Pour illustrer comment le jeu d'instructions de base peut être utilisé pour définir des instructions de plus haut niveau, remarquons que l'addition « $x \mathrel{+}= m$ » et la soustraction « $x \mathrel{-}= m$ » d'un entier naturel m peuvent respectivement être écrites comme m incrémentations consécutives « $x \mathrel{+}= 1$ » et m décrémentations consécutives « $x \mathrel{-}= 1$ ». Nous pouvons aussi décrire les branchements conditionnels « **if** $x = 0$ **then goto** L **else** $x \mathrel{-}= 1$ » des machines de MINSKY, où L est un numéro de ligne, de la façon suivante :

1: **goto** 2 **or** 4
2: **zero ?** x
3: **goto** L
4: $x \mathrel{-}= 1$,

où « **goto** L » est un raccourci syntaxique pour le branchement déterministe « **goto** L **or** L ».

À bien noter que les compteurs (testés ou non) ne peuvent pas prendre de valeurs négatives. Dans l'exemple que nous venons juste d'introduire, cela explique pourquoi la décrémentation à la ligne 4 teste implicitement que le compteur est différent de zéro, ce qui correspond bien à la branche « **else** » du branchement conditionnel de la machine de MINSKY.

Deux autres remarques peuvent être utiles. Premièrement, notre notion de programmes à compteurs permet de présenter de manière commune à la fois les SAVEs et les machines de MINSKY travaillant sur des compteurs bornés, et la syntaxe exacte n'a pas vraiment d'importance ; elle s'inspire de la présentation d'ESPARZA [6] de la borne inférieure de LIPTON [20]. Deuxièmement, bien que la commande **halt if** $x_1, \ldots, x_\ell = 0$ puisse être exprimée par des tests à zéro suivis d'un simple **halt**, cette commande atomique nous permet de forcer des compteurs non testés à être nuls à la fin de l'exécution. Cela permet d'encoder à l'aide de compteurs non testés les composantes des configurations d'un SAVE, qui sont non bornées, mais prenant une valeur initiale fixée et devant atteindre une valeur particulière – correspondant aux configurations initiales et finales $\boldsymbol{c}_{in}, \boldsymbol{c}_{out} \in \mathbb{N}^d$ d'une instance du problème d'accessibilité.

Sémantique. Nous définissons maintenant brièvement la sémantique des programmes à compteurs. Une *B-exécution* d'un programme est une exécution pour laquelle tous les compteurs prennent des valeurs positives ou nulles, tous les compteurs testés ont une valeur d'au plus B, et la commande de test « **max?** x » est interprétée comme une égalité à B.

Une exécution est *maximale* si elle est *infinie*, ou si elle est *terminée* en exécutant avec succès la commande **halt** (qui est nécessairement la dernière du programme), ou si elle est bloquée par une instruction qui ne peut être effectuée : cela se produit si une décrémentation rend un compteur négatif, si une incrémentation d'un compteur testé dépasse la borne maximale des compteurs testés, si un compteur ne peut exécuter un test à zéro ou à la borne maximale, ou si le test à zéro final échoue, et on parle alors d'exécution *partielle*. De plus, à cause des branchements non déterministes, le même programme, à partir de la même configuration initiale peut avoir différents types de B-exécutions des trois catégories : terminée, partielle maximale, infinie. Nous nous intéressons principalement à la valeur des compteurs obtenue à la fin d'une exécution terminée.

Une exécution est dite *complète* si et seulement si elle est terminée et commence depuis la configuration initiale où tous les compteurs sont à zéro. Soient $x_1, \ldots, x_\ell$ certains (pas nécessairement tous) des compteurs

du programme. Nous dirons que la *relation B-calculée dans* $\mathsf{x}_1, \ldots, \mathsf{x}_\ell$ par le programme est l'ensemble des ℓ-uplets $\langle v_1, \ldots, v_\ell \rangle$ tels que le programme a une B-exécution complète dont la configuration finale admet la valeur v_i dans le compteur x_i pour tout $1 \leqslant i \leqslant \ell$.

Nous considérons des programmes dont la borne B peut recevoir plusieurs valeurs. Quand la borne B est claire, ou lorsqu'elle n'est pas importante parce que le programme ne comporte pas de compteur testé, nous écrirons simplement « exécution » et « relation calculée » plutôt que « B-exécution » et « relation B-calculée ».

Accessibilité. Le problème de l'accessibilité pour les programmes à compteurs est défini comme suit.

Problème 2.2.5 (accessibilité des programmes à compteurs)**.**

entrée *Un programme à compteur et une borne B.*

question *Existe-t-il une B-exécution complète ?*

Le problème de l'accessibilité prend en entrée la valeur B de la borne. On notera que la version existentielle du problème prenant en entrée un programme à compteur et demandant s'il existe une B-exécution complète pour une certaine borne B est indécidable car le problème de l'accessibilité pour les machines de MINSKY s'y réduit.

Le problème de l'accessibilité des programmes à compteur se réduit à celui des SAVE en encodant dans les états la valeur des compteurs bornées – avec un coût exponentiel en B. Réciproquement, le problème de l'accessibilité des SAVE se réduit à celui des programmes à compteurs en choisissant pour la borne B la valeur zéro, et en encodant chaque composante d'un SAVE par un compteur non testé.

Exemple 2.2.6. *Prenons comme exemple le programme suivant, où C est un entier naturel, et tous les compteurs sont non testés :*

1: $\mathsf{x}' \mathrel{+}= C$
2: **goto** 6 **or** 3
3: $\mathsf{x} \mathrel{+}= 1 \quad \mathsf{x}' \mathrel{-}= 1$
4: $\mathsf{y} \mathrel{+}= 2$
5: **goto** 2
6: **halt if** $\mathsf{x}' = 0$.

Il répète le bloc de trois commandes des lignes 3–4 un nombre de fois choisi de façon non déterministe (peut-être zéro fois) et finit par s'arrêter pourvu que le compteur x' soit nul. Le SAVE correspondant de dimension trois est illustré dans la figure 2.4, où les configurations sont vues comme des valuations de $(\mathsf{x}, \mathsf{y}, \mathsf{x}')$. Les transitions de q_2 à q_3 et q_6 correspondent au ***goto*** *de la ligne 2 et celle de q_5 à q_2*

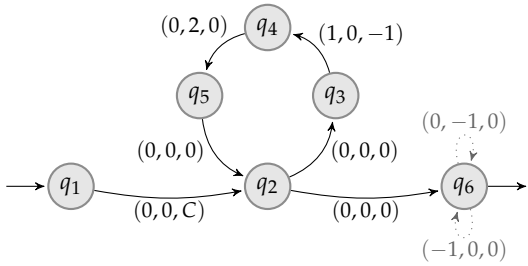


FIGURE *2.4 – Le SAVE correspondant au programme à compteurs de l'exemple 2.2.6.*

au ***goto*** *de la ligne 5. Les boucles finales en pointillés sur l'état q_6 correspondent au fait que* x *et* y *ne sont pas testés à zéro par l'instruction* **halt** *de la ligne 6 : il existe alors une exécution du SAVE de la figure 2.4 entre $q_1(0,0,0)$ et $q_6(0,0,0)$ si et seulement s'il existe une exécution complète du programme.*

En remplaçant les deux branchements par une syntaxe plus simple à lire, nous pouvons aussi écrire ce code comme suit, où l'instruction **loop** *itère le corps de boucle un nombre non déterministe de fois.*

```
1: x' += C
2: loop
3:     x += 1   x' -= 1
4:     y += 2
5: halt if x' = 0.
```

Il est facile de voir qu'il existe une seule exécution complète qui itère la boucle exactement C fois ; par ailleurs il n'y a pas d'exécution infinie à cause de la décrémentation $x' --$ 1. *Ainsi, la relation calculée dans* x, y *est le singleton contenant l'unique paire $\langle C, 2C\rangle$.*

Exemple 2.2.7. *Le SAVE de la figure 2.1 avec la configuration initiale $\boldsymbol{c}_{in} = (0,0,2)$ et la configuration finale $\boldsymbol{c}_{out} = (1,1,0)$ peut être implémenté comme le programme à compteur suivant, où tous les compteurs sont non testés ; on rappelle qu'un programme commence avec tous ses compteurs initialement à zéro.*

```
1: x3 += 2                                   // configuration c_in
2: loop
3:     loop
4:         x2 += 2                           // transition t1
5:     loop
6:         x1 += 2   x2 += 2   x3 -= 1       // transition t2
7:         x1 += 1   x3 -= 2                 // transition t5
8: goto 9 or 11
9: x1 += 1                                   // transition t3
10: goto 12
```

```
11: x1 −= 3   x3 += 1                          // transition t4
12: loop
13:     loop
14:         x1 += 1   x2 −= 1                  // transition t6
15:     loop
16:         x1 += 1   x2 −= 1   x3 −= 2        // transition t7
17:         loop
18:             x1 −= 2   x2 −= 1              // transition t8
19: x1 −= 1   x2 −= 1                          // configuration c_out
20: halt if x1, x2, x3 = 0
```

Dans la suite, utiliserons la commande **loop** à la place de la commande **goto**. Cette restriction ne change pas le problème de l'accessibilité des programmes à compteur car les commandes de branchement **goto** peuvent être simulées à l'aide de commandes **loop**. Cette restriction nous permettra de décrire plus facilement des transformations sur les programmes à compteurs.

Exemple 2.2.8. *Nous aurons besoin de raisonner sur des propriétés vérifiées par les valeurs des compteurs en certains points des programmes. Nous donnons un exemple de ce type de raisonnement qui sera utile plus tard dans la section 2.5 pour simuler des compteurs testés par des compteurs non testés. Pour cela, considérons un entier strictement positif B et supposons que les inégalités*

$$\mathsf{x} + \widehat{\mathsf{x}} \leqslant B \text{ et } \mathsf{d} \geqslant \mathsf{c} \cdot B \tag{2.1}$$

soient vérifiées au début d'une exécution du fragment de programme suivant :

```
loop
    x += 1   x̂ −= 1
    d −= 1
c −= 1
```

Supposons que cette exécution sorte de la boucle après K itérations. On peut voir que la propriété (2.1) *est nécessairement vraie à la sortie, vu que :*

— *la somme* $\mathsf{x} + \widehat{\mathsf{x}}$ *ne varie pas,*

— *nous avons* $K \leqslant B$ *du fait de la décrémentation* $\widehat{\mathsf{x}} -= 1$ *et*

— *les compteurs* d *et* c *ont été respectivement diminués par K et* 1.

En continuant sur cet exemple, si nous supposons de surcroît que, à la sortie, les valeurs des compteurs vérifient $\mathsf{d} = \mathsf{c} \cdot B$*, alors nous pouvons conclure que :*

— *nécessairement* $K = B$*,*

— *l'égalité* $\mathsf{d} = \mathsf{c} \cdot B$ *est aussi vérifiée à l'entrée, et*

— $\mathsf{x} = 0$ *et* $\widehat{\mathsf{x}} = B$ *à l'entrée, et leurs valeurs à la sortie sont échangées.*

Nous avons ainsi vu deux arguments simples, l'un basé sur la propagation en avant des propriétés des valeurs des compteurs d'exécutions de fragments de

programmes, et l'autre sur une propagation en arrière. Les deux propagations seront utiles pour établir une borne inférieure de complexité dans la section 2.5.

2.3 Algorithme de décomposition

Les algorithmes de décomposition pour résoudre l'accessibilité dans les systèmes d'addition de vecteurs procèdent tous selon le même schéma général. Ces algorithmes travaillent sur des systèmes mis sous une forme particulière : il s'agit essentiellement de séquences de systèmes séparés par des actions, que nous appellerons *séquences KLM* d'après MAYR, KOSARAJU et LAMBERT. À chaque étape,

— soit la séquence KLM courante ξ vérifie une condition (on dira ici qu'elle est *normale*) qui garantit l'existence d'une exécution,

— soit on décompose cette séquence ξ en un ensemble fini de nouvelles séquences $dec(\xi)$. Cette décomposition préserve l'ensemble des exécutions, mais chacune des nouvelles séquences $\xi' \in dec(\xi)$ est *plus petite* que ξ pour une certaine fonction de rang dans un ensemble bien fondé.

Autrement dit, ces algorithmes explorent un *arbre de décomposition* étiqueté par des séquences KLM, où un nœud non normal ξ a pour ensemble d'enfants $dec(\xi)$, à la recherche d'une feuille normale. Nous présentons ici les idées principales qui sous-tendent ces algorithmes, et illustrons leur fonctionnement sur l'exemple 2.2.3.

2.3.1 Séquences KLM

Chemins. Dans les algorithmes de décomposition, nous nous intéressons aux SAVEs plutôt qu'aux SAVs, car nous exploitons les propriétés de leurs graphes dirigés sous-jacents. Un *chemin* π dans un SAVE G d'un état p à un état q étiqueté par un mot $\boldsymbol{a}_1 \dots \boldsymbol{a}_k$ d'actions est un mot de transitions de G de la forme $(p_1, \boldsymbol{a}_1, q_1) \dots (p_k, \boldsymbol{a}_k, q_k)$ avec $p_0 = p$, $q_k = q$, et $q_j = p_{j+1}$ pour tout $1 \leqslant j < k$. Un tel chemin est dit *complet* si $p = q_{in}$ et $q = q_{out}$ sont les états d'entrée et de sortie du SAVE G. Notons que $p(\boldsymbol{x}) \xrightarrow[G]{\sigma} q(\boldsymbol{y})$ si et seulement s'il existe un chemin dans G de p à q étiqueté par σ tel que $\boldsymbol{x} \xrightarrow{\sigma} \boldsymbol{y}$. Le *déplacement* $\Delta(\pi) \in \mathbb{Z}^d$ d'un chemin $\pi = (p_1, \boldsymbol{a}_1, q_1) \dots (p_k, \boldsymbol{a}_k, q_k)$ est la somme $\Delta(\pi) \stackrel{\text{def}}{=} \sum_{1 \leqslant j \leqslant k} \boldsymbol{a}_j$.

Exemple 2.3.1. *L'exécution du système de la figure 2.1 présentée dans l'introduction et en utilisant les notations de l'exemple 2.2.3 correspond au chemin complet* $\pi_{\text{ex}} = t_1\, t_1\, t_3\, t_6\, t_7\, t_8\, t_9$ *étiqueté par* $\sigma_{\text{ex}} = \boldsymbol{a}_1\, \boldsymbol{a}_1\, \boldsymbol{a}_3\, \boldsymbol{a}_6\, \boldsymbol{a}_7\, \boldsymbol{a}_8\, \boldsymbol{a}_9$ *et de déplacement* $\Delta(\pi_{\text{ex}}) = (1, 1, -2)$.

Deux états p et q d'un SAVE sont dans la même composante fortement connexe s'il existe un chemin de p à q et un de q à p. Un SAVE $G = (Q, q_{in}, q_{out}, T)$ est dit *fortement connexe* si l'ensemble de ses états Q est une seule composante fortement connexe.

Exemple 2.3.2. *Dans l'exemple 2.2.3, les composantes fortement connexes sont* $\{q_{in}, p\}$ *et* $\{q, q_{out}\}$.

Séquences KLM. Fixons une dimension d et un ensemble d'actions $\boldsymbol{A} \subseteq \mathbb{Z}^d$ fini. Une *séquence KLM* ξ est une séquence finie

$$\xi = (\boldsymbol{x}_0 G_0 \boldsymbol{y}_0)\boldsymbol{a}_1(\boldsymbol{x}_1 G_1 \boldsymbol{y}_1)\dots\boldsymbol{a}_k(\boldsymbol{x}_k G_k \boldsymbol{y}_k) \tag{2.2}$$

où $\boldsymbol{x}_0, \boldsymbol{y}_0, \dots, \boldsymbol{x}_k, \boldsymbol{y}_k$ sont des configurations dans $\mathbb{N}^d_\omega$, $G_0, \dots, G_k$ sont des SAVEs de dimension d sur l'ensemble d'actions $\boldsymbol{A}$, et $\boldsymbol{a}_1, \dots, \boldsymbol{a}_k$ sont des actions issues de $\boldsymbol{A}$.

Une telle séquence représente intuitivement une généralisation du problème d'accessibilité, où chaque $\boldsymbol{x}_j$ (resp. $\boldsymbol{y}_j$) représente une configuration initiale (resp. finale) pour le SAVE G_j, et où l'on souhaite de plus « connecter » les exécutions entre G_j et G_{j+1} en effectuant l'action $\boldsymbol{a}_{j+1}$. Une séquence KLM est dite *fortement connexe* si tous ses SAVEs $G_0, \dots, G_k$ le sont.

Langage d'actions. On définit une relation $\sqsubseteq$ sur $\mathbb{N}_\omega \stackrel{\text{def}}{=} \mathbb{N} \uplus \{\omega\}$ par $x \sqsubseteq y$ si $y \in \{x, \omega\}$; cette relation est étendue composante par composante aux vecteurs dans $\mathbb{N}^d_\omega$. Par exemple, $(3, \omega, 5) \sqsubseteq (\omega, \omega, 5)$ et $\boldsymbol{x} \sqsubseteq \boldsymbol{\omega}$ pour tout $\boldsymbol{x} \in \mathbb{N}^d_\omega$ où $\boldsymbol{\omega}$ est le vecteur de valeur ω dans toutes ses composantes.

Afin de représenter l'ensemble des exécutions d'un SAVE entre configuration source et configuration cible, et plus généralement d'une séquence KLM, on définit le *langage d'actions* d'une séquence KLM ξ comme l'ensemble L_ξ des mots d'action $\sigma_0\boldsymbol{a}_1\sigma_1\dots\boldsymbol{a}_k\sigma_k \in \boldsymbol{A}^*$ tels que chaque σ_j soit l'étiquette d'un chemin complet du SAVE G_j et qu'il existe des configurations $\boldsymbol{m}_0, \boldsymbol{n}_0, \dots, \boldsymbol{m}_k, \boldsymbol{n}_k$ de $\mathbb{N}^d$ telles que

$$\boldsymbol{m}_0 \xrightarrow{\sigma_0} \boldsymbol{n}_0 \xrightarrow{\boldsymbol{a}_1} \cdots \boldsymbol{m}_k \xrightarrow{\sigma_k} \boldsymbol{n}_k \text{ où } \boldsymbol{m}_j \sqsubseteq \boldsymbol{x}_j \text{ et } \boldsymbol{n}_j \sqsubseteq \boldsymbol{y}_j \text{ pour tout } 0 \leqslant j \leqslant k. \tag{2.3}$$

Exemple 2.3.3. *Voici une séquence KLM basée sur l'exemple 2.2.3 :* $\xi_{\text{ex}} = ((0,0,2)G_{\text{ex}}(1,1,0))$. *Son langage d'actions* $L_{\xi_{\text{ex}}}$ *est*

$$\{\boldsymbol{a}_1^{2+3n}\boldsymbol{a}_3\,\boldsymbol{a}_6^{1+4n}\boldsymbol{a}_7\,\boldsymbol{a}_8^{1+2n}\boldsymbol{a}_9 \mid n \in \mathbb{N}\} \cup \{\boldsymbol{a}_1^{2+3n}\boldsymbol{a}_3\,\boldsymbol{a}_6^{4n}\boldsymbol{a}_7\,\boldsymbol{a}_8^{1+2n}\boldsymbol{a}_9\,\boldsymbol{a}_6 \mid n \in \mathbb{N}\}\,.$$

Remarquons que le problème d'accessibilité pour un SAVE G et deux configurations finies $\boldsymbol{c}_{in}$ et $\boldsymbol{c}_{out}$ dans $\mathbb{N}^d$ revient à tester que $L_\xi \neq \emptyset$ pour la séquence KLM $\xi = (\boldsymbol{c}_{in} G \boldsymbol{c}_{out})$. Dans ce cas, le langage d'actions est exactement l'ensemble des séquences $\sigma \in \boldsymbol{A}^*$ telles que $q_{in}(\boldsymbol{c}_{in}) \xrightarrow[G]{\sigma} q_{out}(\boldsymbol{c}_{out})$.

2.3.2 Fonction de rang

Un *cycle* sur un état q d'un SAVE est un chemin de q à q. Pour une transition t d'un SAVE G, nous considérons l'ensemble des déplacements $\Delta(\pi)$ des cycles π qui contiennent t : ces déplacements engendrent un espace vectoriel $\boldsymbol{V}_G(t) \subseteq \mathbb{Q}^d$. Cet espace vectoriel ne dépend en réalité que de la composante fortement connexe de t [18, lem. 3.2].

Le *rang* d'un SAVE G de dimension d est le $(d+1)$-uplet $rang(G) \stackrel{\text{def}}{=} (r_d, \ldots, r_0) \in \mathbb{N}^{d+1}$ où chaque r_i est le nombre de transitions t du SAVE telles que la dimension de $\boldsymbol{V}_G(t)$ vaut i. Le rang d'une séquence KLM ξ est la somme composante par composante des rangs de ses SAVEs : $rang(\xi) \stackrel{\text{def}}{=} \sum_{j=0}^{k} rang(G_j)$. Nous ordonnons les rangs lexicographiquement : $(r_d, \ldots, r_0) \leqslant_{lex} (s_d, \ldots, s_0)$ si les deux rangs sont égaux ou si le plus petit i tel que $r_i \neq s_i$ vérifie $r_i < s_i$.

L'ordre linéaire $(\mathbb{N}^{d+1}, <_{lex})$ est *bien fondé* : il n'existe pas de séquence strictement décroissante infinie de rangs. La décomposition $dec(\xi)$ d'une séquence KLM ξ sera telle que $\xi' <_{lex} \xi$ pour tout $\xi' \in dec(\xi)$, ce qui impliquera que les branches de l'arbre de décomposition seront finies. Notons en passant que, comme tout ordre linéaire bien fondé, $(\mathbb{N}^{d+1}, <_{lex})$ est isomorphe à un ordinal appelé son *type d'ordre*, qui est en l'occurrence ω^{d+1}.

Exemple 2.3.4. *Dans l'exemple 2.2.3,*

$$\boldsymbol{V}_G(t_3) = \boldsymbol{V}_G(t_4) = \{(0,0,0)\},$$
$$\boldsymbol{V}_G(t_1) = \boldsymbol{V}_G(t_2) = \boldsymbol{V}_G(t_5) = Vect((0,2,0),(3,2,\text{-}3)),$$
$$\boldsymbol{V}_G(t_6) = \boldsymbol{V}_G(t_7) = \boldsymbol{V}_G(t_8) = \boldsymbol{V}_G(t_9) = Vect((\text{-}2,\text{-}1,0),(1,\text{-}1,\text{-}2),(1,\text{-}1,0)).$$

Donc $rang(G_{\text{ex}}) = (4,3,0,2) = rang(\xi_{\text{ex}})$.

2.3.3 Composantes fortement connexes

Dans l'algorithme de décomposition, nous travaillons systématiquement avec des SAVEs fortement connexes. On peut toujours décomposer un graphe fini orienté en composantes fortement connexes, par exemple par l'algorithme de TARJAN [35]. Par suite, à partir d'une séquence KLM ξ qui ne serait pas déjà fortement connexe, on peut toujours construire un

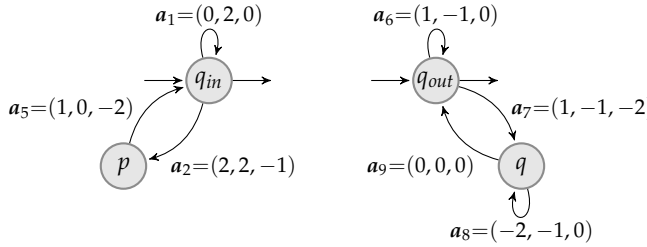


FIGURE 2.5 – *Les SAVEs fortement connexes G^1_{ex} (à gauche) et G^2_{ex} (à droite).*

ensemble fini Ξ de séquences KLM telles que $L_\xi = \bigcup_{\xi' \in \Xi} L_{\xi'}$ où tous les SAVEs sont fortement connexes. Comme le rang de toutes les transitions d'une composante fortement connexe est le même, $rang(\xi') <_{lex} rang(\xi)$ pour tout $\xi' \in \Xi$ [18, lem. 4.2].

Exemple 2.3.5. *Considérons à nouveau le SAVE G_{ex} de l'exemple 2.2.3 et la séquence KLM $\xi_{\text{ex}} = ((0,0,2)G_{\text{ex}}(1,1,0))$. La décomposition en composantes fortement connexes construit un ensemble $\Xi = \{\xi^1_{\text{ex}}, \xi^2_{\text{ex}}\}$ où*

$$\xi^1_{\text{ex}} \stackrel{\text{def}}{=} ((0,0,2)G^1_{\text{ex}}(\omega,\omega,\omega))\boldsymbol{a}_3((\omega,\omega,\omega)G^2_{\text{ex}}(1,1,0)),$$
$$\xi^2_{\text{ex}} \stackrel{\text{def}}{=} ((0,0,2)G^1_{\text{ex}}(\omega,\omega,\omega))\boldsymbol{a}_4((\omega,\omega,\omega)G^2_{\text{ex}}(1,1,0)),$$

où G^1_{ex} et G^2_{ex} sont donnés dans la figure 2.5.

2.3.4 Systèmes d'équations

Une des difficultés centrales du problème d'accessibilité (ou en termes de séquences KLM ξ, de savoir si $L_\xi \neq \emptyset$) est la contrainte que les configurations visitées par le système doivent toujours être des éléments de $\mathbb{N}^d$. De manière informelle, une façon de relâcher le problème est alors de permettre certaines configurations intermédiaires à être dans $\mathbb{Z}^d$. Par exemple, dans le système de la figure 2.1, l'exécution suivante pourrait être considérée, alors qu'elle n'est pas possible dans un SAVE :

$$q_{in}(0,0,2) \xrightarrow[\mathbb{Z}]{\boldsymbol{a}_2} p(2,2,1) \xrightarrow[\mathbb{Z}]{\boldsymbol{a}_5} q_{in}(3,2,-1) \xrightarrow[\mathbb{Z}]{\boldsymbol{a}_4} q_{out}(0,2,0) \xrightarrow[\mathbb{Z}]{\boldsymbol{a}_6} q_{out}(1,1,0).$$

Notons que dans cette exécution relâchée, le chemin complet étiqueté par $\boldsymbol{a}_2\boldsymbol{a}_5\boldsymbol{a}_4\boldsymbol{a}_6$ a pour déplacement $(1,1,-2)$: raisonner de manière relâchée revient à raisonner sur les déplacements associés à nos chemins.

Images de PARIKH. Quand on raisonne sur les déplacements $\Delta(\pi)$ de chemins π, seules les « multiplicités » des transitions dans π ont une influence sur $\Delta(\pi)$. *L'image de* PARIKH [26] d'un chemin π est justement la fonction $\phi \in \mathbb{N}^T$ qui associe à chaque transition $t \in T$ son nombre d'occurrences dans π. Pour une fonction $\phi \in \mathbb{N}^T$, son *déplacement* est $\Delta(\phi) \stackrel{\text{def}}{=} \sum_{t=(p,\boldsymbol{a},q)\in T} \phi(t) \cdot \boldsymbol{a}$; notons que si ϕ est l'image de PARIKH d'un chemin π, alors $\Delta(\pi) = \Delta(\phi)$.

Exemple 2.3.6. *Pour le chemin π_{ex} de l'exemple 2.3.1, $\phi_{\text{ex}} = (2,0,1,0,0,1,1,1,1)$ et $\Delta(\phi_{\text{ex}}) = (1,1,-2)$.*

Système d'équations de KIRCHHOFF. Plus généralement, si les SAVEs sur lesquels nous travaillons sont fortement connexes, l'existence d'un chemin complet dans un SAVE $G = (Q, q_{in}, q_{out}, T)$ est liée à l'existence d'une solution du *système de* KIRCHHOFF K_G de $|Q|$ équations linéaires, où les inconnues sont les multiplicités de chaque transition, qui force le nombre d'entrées dans chaque état à être égal à son nombre de sorties – en comptabilisant aussi l'entrée dans q_{in} et la sortie de q_{out}. Exprimé de manière vectorielle, $\phi \in \mathbb{N}^T$ est un *modèle* de K_G, noté $\phi \models K_G$, si

$$\mathbb{1}_{q_{out}} - \mathbb{1}_{q_{in}} = \sum_{t=(p,\boldsymbol{a},q)\in T} \phi(t)(\mathbb{1}_q - \mathbb{1}_p), \tag{2.4}$$

où $\mathbb{1}_q \colon Q \to \{0,1\}$ est la *fonction caractéristique* de $q \in Q$, définie par $\mathbb{1}_q(p) \stackrel{\text{def}}{=} 1$ si $p = q$ et par $\mathbb{1}_q(p) \stackrel{\text{def}}{=} 0$ sinon. Le lien entre existence d'un chemin complet et celle d'un modèle du système de KIRCHHOFF est le suivant.

Lemme 2.3.7 (EULER)**.** *Soit G un SAVE fortement connexe. Si $\phi \models K_G$ et $\phi(t) > 0$ pour tout $t \in T$, alors il existe un chemin complet π dans G tel que ϕ soit son image de* PARIKH.

Exemple 2.3.8. *Pour les SAVEs G^1_{ex} et G^2_{ex} de la figure 2.5, $K_{G^1_{\text{ex}}}$ est le système suivant avec une équation pour q_{in} et une pour p :*

$$\begin{cases} 0 = -\phi(t_1) + \phi(t_1) - \phi(t_2) + \phi(t_5) \\ 0 = \phi(t_2) - \phi(t_5) \end{cases}$$

Un modèle de $K_{G^1_{\text{ex}}}$ est ϕ^1_{ex} tel que $\phi^1_{\text{ex}}(t_1) = 0$ et $\phi^1_{\text{ex}}(t_2) = \phi^1_{\text{ex}}(t_5) = 1$. De même, $K_{G^2_{\text{ex}}}$ est le système suivant avec une équation pour q et une pour q_{out} :

$$\begin{cases} 0 = \phi(t_7) - \phi(t_8) + \phi(t_8) - \phi(t_9) \\ 0 = -\phi(t_6) + \phi(t_6) - \phi(t_7) + \phi(t_9). \end{cases}$$

Un modèle de $K_{G^2_{\text{ex}}}$ est ϕ^2_{ex} tel que $\phi^2_{\text{ex}}(t_6) = 1$ et $\phi^2_{\text{ex}}(t_7) = \phi^2_{\text{ex}}(t_8) = \phi^2_{\text{ex}}(t_9) = 0$.

Système d'équations caractéristique. Pour une séquence KLM fortement connexe $\xi = (\boldsymbol{x}_0 G_0 \boldsymbol{y}_0)\boldsymbol{a}_1 \cdots \boldsymbol{a}_k(\boldsymbol{x}_k G_k \boldsymbol{y}_k)$ où chaque SAVE G_j utilise un ensemble T_j de transitions, une *séquence caractéristique* $\boldsymbol{h} = (\boldsymbol{m}_0, \phi_0, \boldsymbol{n}_0) \cdots (\boldsymbol{m}_k, \phi_k, \boldsymbol{n}_k)$ est un vecteur d'entiers naturels de dimension $D_\xi \stackrel{\text{def}}{=} 2d(k+1) + \sum_{0 \leqslant j \leqslant k} |T_j|$, où $\boldsymbol{m}_j, \boldsymbol{n}_j \in \mathbb{N}^d$ et $\phi_j \in \mathbb{N}^{T_j}$ pour tout $0 \leqslant j \leqslant k$. Une séquence caractéristique $\boldsymbol{h}$ est un *modèle* du *système caractéristique* E_ξ de ξ si elle vérifie

1. $\boldsymbol{m}_j \sqsubseteq \boldsymbol{x}_j, \phi_j \models K_{G_j}, \Delta(\phi_j) = -\boldsymbol{m}_j + \boldsymbol{n}_j$ et $\boldsymbol{n}_j \sqsubseteq \boldsymbol{y}_j$ pour tout $0 \leqslant j \leqslant k$, et
2. $\boldsymbol{n}_{j-1} \xrightarrow{\boldsymbol{a}_j} \boldsymbol{m}_j$ pour tout $1 \leqslant j \leqslant k$.

Nous notons $\boldsymbol{h} \models E_\xi$ si $\boldsymbol{h}$ est un modèle de E_ξ. Une séquence KLM ξ est *satisfaisable* si E_ξ a un modèle, et *insatisfaisable* sinon. Soit $\boldsymbol{H}_\xi \stackrel{\text{def}}{=} \{\boldsymbol{h} \in \mathbb{N}^{D_\xi} \mid \boldsymbol{h} \models E_\xi\}$ l'ensemble des solutions du système caractéristique.

Puisque le système caractéristique capture une notion d'accessibilité relâchée, il est assez immédiat que, si ξ est insatisfaisable, alors L_ξ est vide. Plus précisément, il suffit d'observer que si $\sigma_0 \boldsymbol{a}_1 \sigma_1 \ldots \boldsymbol{a}_k \sigma_k \in L_\xi$ avec $\boldsymbol{m}_0, \boldsymbol{n}_0, \ldots, \boldsymbol{m}_k, \boldsymbol{n}_k$ vérifiant (2.3), alors $(\boldsymbol{m}_0 \phi_0 \boldsymbol{n}_0) \ldots (\boldsymbol{m}_k \phi_k \boldsymbol{n}_k)$ où ϕ_j est l'image de PARIKH de σ_j, est un modèle de E_ξ.

Exemple 2.3.9. *Pour la séquence KLM ξ^2_{ex} de l'exemple 2.3.5, le système caractéristique $E_{\xi^2_{\text{ex}}}$ est donné à gauche de la figure 2.6. Un modèle de $E_{\xi^2_{\text{ex}}}$ est la séquence caractéristique*

$$((0,0,2), \phi^1_{\text{ex}}, (3,2,-1))((0,2,0), \phi^2_{\text{ex}}, (1,1,0))$$

où ϕ^1_{ex} et ϕ^2_{ex} sont les modèles de $K_{G^1_{\text{ex}}}$ et $K_{G^2_{\text{ex}}}$ définis dans l'exemple 2.3.8.

Solutions de systèmes d'équations linéaires sur les entiers. Le système caractéristique E_ξ d'une séquence KLM ξ est un système d'équations linéaires $A \cdot \boldsymbol{h} = \boldsymbol{c}$ sur les entiers naturels. Quand les entiers sont représentés en binaire, l'existence d'un modèle est dans NP par des résultats classiques d'algèbre linéaire [37, 25] ; mieux encore, on sait décrire exactement l'ensemble des solutions du système.

Soit $D \in \mathbb{N}$ une dimension. Rappelons qu'un ensemble $\boldsymbol{S} \subseteq \mathbb{N}^D$ est *linéaire* s'il existe une base $\boldsymbol{b} \in \mathbb{N}^D$ et un ensemble fini $\boldsymbol{P} = \{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\} \subseteq \mathbb{N}^D$ de périodes tels que

$$\boldsymbol{S} = L(\boldsymbol{b}, \boldsymbol{P}) \stackrel{\text{def}}{=} \{\boldsymbol{b} + \lambda_1 \boldsymbol{p}_1 + \cdots + \lambda_n \boldsymbol{p}_n \mid \lambda_1, \ldots, \lambda_n \in \mathbb{N}\}.$$

Un ensemble *semi-linéaire* [26] est une union finie d'ensembles linéaires.

$$
\begin{cases}
0 = \boldsymbol{m}_0(1)\\
0 = \boldsymbol{m}_0(2)\\
2 = \boldsymbol{m}_0(3)\\
0 = -\phi_0(t_1) + \phi_0(t_1) - \phi_0(t_2) + \phi_0(t_5)\\
0 = \phi_0(t_2) - \phi_0(t_3)\\
0 = 2\phi_0(t_2) + \phi_0(t_5) + \boldsymbol{m}_0(1) - \boldsymbol{n}_0(1)\\
0 = 2\phi_0(t_1) + 2\phi_0(t_2) + \boldsymbol{m}_0(2) - \boldsymbol{n}_0(2)\\
0 = -\phi_0(t_2) - 2\phi_0(t_5) + \boldsymbol{m}_0(3) - \boldsymbol{n}_0(3)\\
-3 = \boldsymbol{m}_1(1) - \boldsymbol{n}_0(1)\\
0 = \boldsymbol{m}_1(2) - \boldsymbol{n}_0(2)\\
1 = \boldsymbol{m}_1(3) - \boldsymbol{n}_0(3)\\
0 = \phi_1(t_7) - \phi_1(t_8) + \phi_1(t_8) - \phi_1(t_9)\\
0 = -\phi_1(t_6) + \phi_1(t_6) - \phi_1(t_7) + \phi_1(t_9)\\
0 = \phi_1(t_6) + \phi_1(t_7) - 2\phi_1(t_8) + \boldsymbol{m}_1(1) - \boldsymbol{n}_1(1)\\
0 = -\phi_1(t_6) - \phi_1(t_7) - \phi_1(t_8) + \boldsymbol{m}_1(2) - \boldsymbol{n}_1(2)\\
0 = -2\phi_1(t_7) + \boldsymbol{m}_1(3) - \boldsymbol{n}_1(3)\\
1 = \boldsymbol{n}_1(1)\\
1 = \boldsymbol{n}_1(2)\\
0 = \boldsymbol{n}_1(3)
\end{cases}
\qquad
\begin{cases}
0 = \boldsymbol{m}_0(1)\\
0 = \boldsymbol{m}_0(2)\\
0 = \boldsymbol{m}_0(3)\\
0 = -\phi_0(t_1) + \phi_0(t_1) - \phi_0(t_2) + \phi_0(t_5)\\
0 = \phi_0(t_2) - \phi_0(t_3)\\
0 = 2\phi_0(t_2) + \phi_0(t_5) + \boldsymbol{m}_0(1) - \boldsymbol{n}_0(1)\\
0 = 2\phi_0(t_1) + 2\phi_0(t_2) + \boldsymbol{m}_0(2) - \boldsymbol{n}_0(2)\\
0 = -\phi_0(t_2) - 2\phi_0(t_5) + \boldsymbol{m}_0(3) - \boldsymbol{n}_0(3)\\
0 = \boldsymbol{m}_1(1) - \boldsymbol{n}_0(1)\\
0 = \boldsymbol{m}_1(2) - \boldsymbol{n}_0(2)\\
0 = \boldsymbol{m}_1(3) - \boldsymbol{n}_0(3)\\
0 = \phi_1(t_7) - \phi_1(t_8) + \phi_1(t_8) - \phi_1(t_9)\\
0 = -\phi_1(t_6) + \phi_1(t_6) - \phi_1(t_7) + \phi_1(t_9)\\
0 = \phi_1(t_6) + \phi_1(t_7) - 2\phi_1(t_8) + \boldsymbol{m}_1(1) - \boldsymbol{n}_1(1)\\
0 = -\phi_1(t_6) - \phi_1(t_7) - \phi_1(t_8) + \boldsymbol{m}_1(2) - \boldsymbol{n}_1(2)\\
0 = -2\phi_1(t_7) + \boldsymbol{m}_1(3) - \boldsymbol{n}_1(3)\\
0 = \boldsymbol{n}_1(1)\\
0 = \boldsymbol{n}_1(2)\\
0 = \boldsymbol{n}_1(3)
\end{cases}
$$

FIGURE 2.6 – *Le système caractéristique $E_{\xi^2_{\text{ex}}}$ (à gauche) et le système homogène $E^0_{\xi^2_{\text{ex}}}$ (à droite) de la séquence KLM ξ^2_{ex} de l'exemple 2.3.5.*

Lemme 2.3.10 ([28, 2]). *Soient $\boldsymbol{S}_0 = \{\boldsymbol{x} \in \mathbb{N}^D \mid A \cdot \boldsymbol{x} = \boldsymbol{0}\}$ et $\boldsymbol{S} = \{\boldsymbol{x} \in \mathbb{N}^D \mid A \cdot \boldsymbol{x} = \boldsymbol{c}\}$ des ensembles de solutions de systèmes d'équations linéaires. Alors on peut calculer (en temps exponentiel) des ensembles finis $\boldsymbol{B}, \boldsymbol{P} \subseteq \mathbb{N}^D$ tels que $\boldsymbol{S}_0 = L(\boldsymbol{0}, \boldsymbol{P})$ et $\boldsymbol{S} = \bigcup_{\boldsymbol{b} \in \boldsymbol{B}} L(\boldsymbol{b}, \boldsymbol{P})$.*

On déduit du lemme précédent que l'ensemble des modèles $\boldsymbol{H}_\xi$ du système caractéristique E_ξ est un ensemble semi-linéaire pour lequel on peut calculer un ensemble fini de bases $\boldsymbol{B}$ et un ensemble fini de périodes $\boldsymbol{P}$. En particulier, $\boldsymbol{P}$ est l'ensemble de périodes des solutions du *système homogène* E^0_ξ associé à E_ξ ; le système d'équations de droite de la figure 2.6 montre le système homogène $E^0_{\xi^2_{\text{ex}}}$ de la séquence KLM ξ^2_{ex} de l'exemple 2.3.5. Le lemme 2.3.10 permet par exemple de trouver des modèles $\boldsymbol{h} = (\boldsymbol{m}_0, \phi_0, \boldsymbol{n}_0) \cdots (\boldsymbol{m}_k, \phi_k, \boldsymbol{n}_k)$ dans lesquels $\phi_j(t) > 0$ pour tout $0 \leqslant j \leqslant k$ et $t \in T_j$ en inspectant le contenu de $\boldsymbol{B}$ et $\boldsymbol{P}$; l'intérêt est que le lemme 2.3.7 d'EULER s'applique à de tels modèles.

2.3.5 Séquences KLM normales

Revenons maintenant au problème d'accessibilité des SAVEs. Considérons pour simplifier nos explications une séquence KLM $\xi = (\boldsymbol{x}G\boldsymbol{y})$ avec un seul SAVE $G = (Q, q_{in}, q_{out}, T)$ fortement connexe et où $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{N}^d$; les raisonnements qui suivent se généralisent modulo une condition supplémentaire de « saturation » de la séquence KLM – voir [18] pour les détails.

Quand on dispose d'un modèle $\boldsymbol{h} = (\boldsymbol{m}, \phi, \boldsymbol{n})$ du système d'équations caractéristique E_ξ tel que $\phi(t) > 0$ pour tout $t \in T$, par le lemme 2.3.7 d'EULER, on sait qu'il existe un mot d'action $\sigma \in \boldsymbol{A}^*$ qui est un chemin complet du SAVE G d'image de PARIKH ϕ et donc telle que $\Delta(\phi) = -\boldsymbol{m} + \boldsymbol{n}$, que $\boldsymbol{m} \sqsubseteq \boldsymbol{x}$ et que $\boldsymbol{n} \sqsubseteq \boldsymbol{y}$. Tout ce qu'il manque au mot d'action σ pour être dans L_ξ est donc que $\boldsymbol{m} \xrightarrow{\sigma} \boldsymbol{n}$, parce que le mot d'action σ pourrait rendre certaines composantes négatives.

L'idée que nous allons affiner par la suite est qu'il existe un vecteur $\boldsymbol{c} \in \mathbb{N}^d$ tel que $\boldsymbol{m} + \boldsymbol{c} \xrightarrow{\sigma} \boldsymbol{n} + \boldsymbol{c}$: si on pouvait appliquer à $\boldsymbol{m}$ une translation par $+\boldsymbol{c}$ avant d'exécuter σ puis une translation par $-\boldsymbol{c}$ pour redescendre à $\boldsymbol{n}$, on pourrait construire un mot du langage L_ξ.

Pompage. Intuitivement, pour pouvoir effectuer de telles translations, nous allons chercher des boucles qui permettent de mettre des valeurs de plus en plus grandes depuis l'état-configuration $q_{in}(\boldsymbol{x})$ et des valeurs de plus en plus petites vers l'état-configuration $q_{out}(\boldsymbol{y})$. Notons que ce deuxième cas est symétrique : cela correspond à chercher des boucles qui mettent des valeurs de plus en plus grandes depuis $q_{out}(\boldsymbol{y})$ quand le sens des flèches du SAVE est inversé ; nous nous concentrerons donc sur le premier cas. Nous dirons qu'une séquence KLM où cela est possible est *pompable* et *rigide*, et nous allons maintenant définir plus formellement ce que cela signifie.

Une composante $i \in \{1, \ldots, d\}$ d'un SAVE $G = (Q, q_{in}, q_{out}, T)$ est dite *fixée* par G s'il existe une fonction $f_i \colon Q \to \mathbb{N}$ qui « fixe » une valeur finie pour la i^{e} composante dans le système : formellement, on demande que $f_i(q) = f_i(p) + \boldsymbol{a}(i)$ pour toutes les transitions $(p, \boldsymbol{a}, q)$ de T. Déterminer quelles composantes sont fixées et une fonction f_i correspondante peut être fait en temps polynomial.

Une séquence KLM $\xi = (\boldsymbol{x}G\boldsymbol{y})$ est *rigide* si, pour toute composante i fixée par G, il existe une fonction $g_i \colon Q \to \mathbb{N}$ telle que $g_i(q) = g_i(p) + \boldsymbol{a}(i)$ pour toutes les transitions $(p, \boldsymbol{a}, q)$ de T et telle que $g_i(q_{in}) \sqsubseteq \boldsymbol{x}(i)$ et $g_i(q_{out}) \sqsubseteq \boldsymbol{y}(i)$. Cela revient à vérifier que les composantes fixées par le SAVE G sont compatibles avec les contraintes $\boldsymbol{x}$ d'entrée et $\boldsymbol{y}$ de sortie de

la séquence KLM. À nouveau, cette condition peut être vérifiée en temps polynomial par propagation [18, lem. 4.9].

Pour une séquence KLM $\xi = (\boldsymbol{x}G\boldsymbol{y})$, l'*accélération en avant* $\mathrm{Facc}_G(\boldsymbol{x})$ et l'*accélération en arrière* $\mathrm{Bacc}_G(\boldsymbol{y})$ sont deux vecteurs dans $\mathbb{N}_\omega^d$ définis pour toute composante $1 \leqslant i \leqslant d$ comme suit :

$$\mathrm{Facc}_G(\boldsymbol{x})(i) \stackrel{\text{def}}{=} \begin{cases} \omega & \text{si } \exists \boldsymbol{x}' \geqslant \boldsymbol{x} \text{ tel que } \boldsymbol{x}'(i) > \boldsymbol{x}(i) \text{ et } q_{in}(\boldsymbol{x}) \xrightarrow[G]{*} q_{in}(\boldsymbol{x}') \\ \boldsymbol{x}(i) & \text{sinon} \end{cases}$$

$$\mathrm{Bacc}_G(\boldsymbol{y})(i) \stackrel{\text{def}}{=} \begin{cases} \omega & \text{si } \exists \boldsymbol{y}' \geqslant \boldsymbol{y} \text{ tel que } \boldsymbol{y}'(i) > \boldsymbol{y}(i) \text{ et } q_{out}(\boldsymbol{y}') \xrightarrow[G]{*} q_{out}(\boldsymbol{y}) \\ \boldsymbol{y}(i) & \text{sinon} \end{cases}$$

Par définition, $\mathrm{Facc}_G(\boldsymbol{x})(i) = \boldsymbol{x}(i)$ et $\mathrm{Bacc}_G(\boldsymbol{y})(i) = \boldsymbol{y}(i)$ pour toute composante i fixée par G. La séquence $(\boldsymbol{x}G\boldsymbol{y})$ est *pompable* si $\mathrm{Facc}_G(\boldsymbol{x})(i) = \omega$ et $\mathrm{Bacc}_G(\boldsymbol{y})(i) = \omega$ pour toute composante i non fixée par G. Les accélérations $\mathrm{Facc}_G(\boldsymbol{x})$ et $\mathrm{Bacc}_G(\boldsymbol{y})$ sont calculables *via* $2d$ appels à un oracle pour le problème de couverture [15, lem. 3.3], qui est lui-même soluble en espace exponentiel [29], et on sait donc vérifier en espace exponentiel si une séquence KLM est pompable. De plus, il existe une fonction g doublement exponentielle telle que, pour tout i, si $\mathrm{Facc}_G(\boldsymbol{x})(i) = \omega > \boldsymbol{x}(i)$ alors il existe $\sigma_{i,F}$ et $\boldsymbol{x}'$ tels que $q_{in}(\boldsymbol{x}) \xrightarrow[G]{\sigma_{i,F}} q_{in}(\boldsymbol{x}')$, $\boldsymbol{x}' \geqslant \boldsymbol{x}$, $\boldsymbol{x}'(i) > \boldsymbol{x}(i)$ et $|\sigma_{i,F}| \leqslant g(|\xi|)$ et pareillement pour $\mathrm{Bacc}_G(\boldsymbol{y})(i)$.

Proposition 2.3.11. *Si $\xi = (\boldsymbol{x}G\boldsymbol{y})$ est rigide, pompable et telle que $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{N}^d$, alors pour tout $1 \leqslant i \leqslant d$ il existe une fonction $f_i \colon Q \to \mathbb{N}_\omega$ telle que $f_i(q) = f_i(p) + \boldsymbol{a}(i)$ pour toute transition $t \in T$, $f_i(q_{in}) = \mathrm{Facc}_G(\boldsymbol{x})(i)$ et $f_i(q_{out}) = \mathrm{Bacc}_G(\boldsymbol{y})(i)$.*

Démonstration. Puisque ξ est pompable, pour toute composante i non fixée de G, $\mathrm{Facc}_G(\boldsymbol{x})(i) = \mathrm{Bacc}_G(\boldsymbol{y})(i) = \omega$ et on peut poser $f_i(q) \stackrel{\text{def}}{=} \omega$ pour tout état $q \in Q$. Puisque ξ est rigide, pour toute composante i fixée par G, il existe $g_i \colon Q \to \mathbb{N}$ telle que $g_i(q) = g_i(p) + \boldsymbol{a}(i)$ pour toutes les transitions $t \in T$ et telle que $g_i(q_{in}) \sqsubseteq \boldsymbol{x}(i) = \mathrm{Facc}_G(\boldsymbol{x})(i) \in \mathbb{N}$ et $g_i(q_{out}) \sqsubseteq \boldsymbol{y}(i) = \mathrm{Bacc}_G(\boldsymbol{y})(i) \in \mathbb{N}$: dans ce cas on peut poser $f_i \stackrel{\text{def}}{=} g_i$. ∎

Transitions bornées. Nous sommes presque en mesure de trouver des mots d'action dans le langage de notre séquence KLM fortement connexe, satisfaisable, pompable et rigide. Il nous reste deux problèmes que nous allons résoudre d'un coup.

— D'une part, pour pouvoir appliquer le lemme d'EULER, nous avons besoin de modèles de E_{ξ} tels que $\phi(t) > 0$ pour toutes les transitions $t \in T$.
— D'autre part, les effets des pompages avant $q_{in}(\boldsymbol{x}) \xrightarrow[G]{\sigma_{i,F}} q_{in}(\boldsymbol{x}')$ avec $\boldsymbol{x}' \geqslant \boldsymbol{x}$ et $\boldsymbol{x}'(i) > \boldsymbol{x}(i)$ et arrière $q_{out}(\boldsymbol{y}') \xrightarrow[G]{\sigma_{i,B}} q_{out}(\boldsymbol{y})$ avec $\boldsymbol{y}' \geqslant \boldsymbol{y}$ et $\boldsymbol{y}'(i) > \boldsymbol{y}(i)$ doivent pouvoir être compensés.

Nous dirons qu'une séquence KLM satisfaisable est *non bornée* s'il existe des modèles $(\boldsymbol{m}, \boldsymbol{\phi}, \boldsymbol{n})$ de E_{ξ} tels que $\boldsymbol{\phi}(t)$ soit arbitrairement grand pour toutes les transitions $t \in T$. Comme pour la saturation, par le lemme 2.3.10, cela peut se vérifier sur l'ensemble des périodes $\boldsymbol{P}$ telles que $\boldsymbol{H}_{\xi} = \bigcup_{\boldsymbol{b} \in \boldsymbol{B}} L(\boldsymbol{b}, \boldsymbol{P})$.

Plus précisément, puisque ξ est satisfaisable, on peut considérer $(\boldsymbol{m}, \boldsymbol{\phi}, \boldsymbol{n})$ un modèle de E_{ξ}. Rappelons que par le lemme 2.3.10, $\boldsymbol{P}$ est un ensemble de périodes tel que $L(\boldsymbol{0}, \boldsymbol{P})$ soit l'ensemble des modèles du système homogène E^0_{ξ}. La séquence ξ est donc non bornée si et seulement s'il existe un modèle $(\boldsymbol{0}, \boldsymbol{\phi}_0, \boldsymbol{0})$ du système homogène E^0_{ξ} tel que $\boldsymbol{\phi}_0(t) > 0$ pour toute transition $t \in T$ – en effet, $(\boldsymbol{m}, \boldsymbol{\phi} + r\boldsymbol{\phi}_0, \boldsymbol{n})$ est alors aussi un modèle de E_{ξ} pour tout $r \in \mathbb{N}$. Par le lemme 2.3.10, on sait de plus qu'il existe une fonction g doublement exponentielle telle que $\sum_{t \in T} \boldsymbol{\phi}(t) + \boldsymbol{\phi}_0(t) \leqslant g(|\xi|)$.

Exécutions d'une séquence KLM normale. On appelle une séquence KLM ξ fortement connexe, satisfaisable, rigide, pompable et non bornée une séquence *normale*. Cette condition garantit l'existence d'un mot dans le langage L_{ξ}.

Lemme 2.3.12 ([18, lem. 4.19])**.** *Si ξ est une séquence KLM normale, alors on peut calculer un mot d'action dans L_{ξ} de longueur élémentaire en la taille de ξ.*

Démonstration. Nous allons donner l'idée de cette preuve dans le cas particulier d'une séquence KLM $\xi = (\boldsymbol{x} G \boldsymbol{y})$ normale où $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{N}^d$. Soit I l'ensemble des composantes non fixées de G et $\|T\| \stackrel{\text{def}}{=} \max_{1 \leqslant i \leqslant d} \max_{\boldsymbol{a} \in \boldsymbol{A}} |\boldsymbol{a}(i)|$ la norme infinie des actions de G.

Résumons tout d'abord tout ce qui découle directement des hypothèses sur ξ. Comme ξ est satisfaisable et non bornée, il existe un modèle $(\boldsymbol{m}, \boldsymbol{\phi}, \boldsymbol{n})$ du système caractéristique E_{ξ} tel que $\boldsymbol{\phi}(t) > 0$ pour tout $t \in T$ et un modèle $(\boldsymbol{0}, \boldsymbol{\phi}_0, \boldsymbol{0})$ du système homogène E^0_{ξ} tel que $\boldsymbol{\phi}_0(t) > 0$ pour toutes les transitions $t \in T$. Comme ξ est pompable, pour toute composante $i \in I$, il existe un mot d'action $\sigma_{i,F}$ tel que $q_{in}(\boldsymbol{x}) \xrightarrow[G]{\sigma_{i,F}} q_{in}(\boldsymbol{x}')$, $\boldsymbol{x}' \geqslant \boldsymbol{x}$ et $\boldsymbol{x}'(i) > \boldsymbol{x}(i)$. En faisant la concaténation de ces mots d'action, on obtient un mot d'action u tel que $q_{in}(\boldsymbol{x}) \xrightarrow[G]{u} q_{in}(\boldsymbol{x}')$ pour un certain $\boldsymbol{x}' \geqslant \boldsymbol{x}$

tel que $\boldsymbol{x}'(i) > \boldsymbol{x}(i)$ pour toute composante $i \in I$. On construit aussi un mot d'action v de manière symétrique : $q_{out}(\boldsymbol{y}') \xrightarrow[G]{v} q_{out}(\boldsymbol{y})$ pour un certain $\boldsymbol{y}' \geqslant \boldsymbol{y}$ tel que $\boldsymbol{y}'(i) > \boldsymbol{y}(i)$ pour toute composante $i \in I$. Autrement dit, $\Delta(u)(i) > 0$ et $\Delta(v)(i) < 0$ pour tout $i \in I$ et $\Delta(u)(i) = \Delta(v)(i) = 0$ pour tout $i \notin I$. Enfin, il existe une fonction doublement exponentielle g telle que $\sum_{t\in T} \phi(t) + \phi_0(t) \leqslant g(|\xi|)$, $|u| \leqslant g(|\xi|)$ et $|v| \leqslant g(|\xi|)$.

Par le lemme 2.3.7 d'EULER, comme G est fortement connexe, il existe un mot d'action $\sigma \in \boldsymbol{A}^*$ qui est un chemin complet de q_{in} à q_{out} dans le SAVE G tel que $\Delta(\sigma) = \Delta(\phi) = -\boldsymbol{m} + \boldsymbol{n}$. Par la proposition 2.3.11,

$$q_{in}(\mathrm{Facc}_G(\boldsymbol{x})) \xrightarrow[G]{\sigma} q_{out}(\mathrm{Bacc}_G(\boldsymbol{y})) \ , \tag{2.5}$$

c'est-à-dire que seules les composantes dans I peuvent devenir négatives si on essaie d'exécuter σ entre $\boldsymbol{x}$ et $\boldsymbol{y}$. Il existe alors $s \geqslant |\sigma|\|T\|$ tel que

$$q_{in}(\boldsymbol{x}) \xrightarrow[G]{u^s} q_{in}(\boldsymbol{x} + s\Delta(u)) \tag{2.6}$$

et

$$q_{in}(\boldsymbol{x} - s\Delta(v)) \xrightarrow[G]{\sigma} q_{out}(\boldsymbol{y} - s\Delta(v)) \xrightarrow[G]{v^s} q_{out}(\boldsymbol{y}) \ . \tag{2.7}$$

On choisit pour cela $s \stackrel{\text{def}}{=} (2g(|\xi|) + 1)g(|\xi|)\|T\|$, ce qui convient puisque $|\sigma| \leqslant g(|\xi|)$.

Soit ψ_u l'image de PARIKH de u et ψ_v celle de v. On peut alors poser $r \stackrel{\text{def}}{=} 2g(|\xi|) + 1$, qui est tel que $r\phi_0(t) - \psi_u(t) - \psi_v(t) > 0$ pour toute transition $t \in T$ puisque $|u| \leqslant g(|\xi|)$ et $|v| \leqslant g(|\xi|)$. Posons alors $\psi \stackrel{\text{def}}{=} r\phi_0 - \psi_u - \psi_v$ et rappelons que $\Delta(\phi_0) = \mathbf{0}$. Alors comme u est un cycle sur q_{in} et v est un cycle sur q_{out}, ψ est un modèle d'une variante du système de KIRCHHOFF pour G où q_{in} sert à la fois d'état source et d'état cible. Comme G est fortement connexe, par le lemme 2.3.7 d'EULER il existe un cycle w qui visite l'état q_{in} et tel que $\Delta(w) = \Delta(\psi) = -\Delta(\psi_u) - \Delta(\psi_v) = -\Delta(u) - \Delta(v)$. Pour finir, comme $|w| \leqslant r \cdot g(|\xi|)$, on a alors $s \geqslant |w|\|T\|$ et donc

$$q_{in}(\boldsymbol{x} + s\Delta(u)) \xrightarrow[G]{w^s} q_{in}(\boldsymbol{x} - s\Delta(v)). \tag{2.8}$$

Pour conclure, d'après les équations (2.6) à (2.8), le mot d'action $u^s w^s \sigma v^s$ est un mot du langage L_ξ. ■

2.3.6 Décomposition

Que faire quand notre séquence KLM ξ n'est pas normale ? Comme déjà mentionné, dans ce cas on construit un ensemble fini $\mathrm{dec}(\xi)$ de nouvelles séquences KLM, tel que $L_\xi = \bigcup_{\xi' \in \mathrm{dec}(\xi)} L_{\xi'}$ et $rang(\xi') <_{lex} rang(\xi)$ pour tout $\xi' \in \mathrm{dec}(\xi)$. Complètement détailler cette décomposition serait beaucoup trop long pour ce chapitre. Les deux cas les plus intéressants sont

— si ξ n'est pas pompable, par exemple $\mathrm{Facc}_G(\boldsymbol{x})(i) < \omega$ bien que i soit non fixé : alors on peut décomposer le système en intégrant la valeur de la i^{e} composante jusqu'à une certaine constante dans les états de contrôle ;

— si ξ n'est pas non borné : on « déplie » alors toutes les transitions dont les valeurs dans les modèles de E_ξ sont bornées.

Le résultat est résumé dans le théorème suivant.

Théorème 2.3.13 ([18, lem. 4.17 et lem. 4.18])**.** *Soit ξ une séquence KLM non normale. Alors on peut calculer en temps élémentaire un ensemble fini* $\mathrm{dec}(\xi)$ *de séquences KLM telles que $L_\xi = \bigcup_{\xi' \in \mathrm{dec}(\xi)} L_{\xi'}$ et $rang(\xi') <_{lex} rang(\xi)$ pour toute séquence KLM $\xi' \in \mathrm{dec}(\xi)$.*

2.4 Borne supérieure de complexité

Nous présentons dans cette section les ingrédients de la preuve faite dans [18] que l'accessibilité des SAVEs peut être résolue en temps « ackermannien ». Pour cela, nous allons analyser la complexité de l'algorithme de décomposition de la section 2.3. Plus précisément, nous allons montrer comment borner explicitement la longueur des branches de l'arbre de décomposition. C'est en effet la source principale de complexité de l'algorithme, à partir de laquelle on pourra aussi borner la tailles des décompositions KLM normales que l'on obtient aux feuilles de l'arbre de décomposition, ainsi que la longueur des témoins d'accessibilité que l'on peut extraire de ces séquences normales.

Afin de borner la longueur des branches, nous allons raisonner sur la fonction de rang de la section 2.3.2 qui nous permet de prouver que ces branches sont toujours finies. Nous nous appuyons pour cela sur un « théorème de longueur » prouvé dans [31] qui donne des bornes sur la longueur de séquences strictement décroissantes d'ordinaux, et que nous présentons dans la section 2.4.2. Dans la section 2.4.4, il restera ensuite à localiser dans quelles classes de complexité nous amènent ces bornes, ce pour quoi nous rappelons en section 2.4.3 la définition des classes de

complexité « à croissance rapide » de [33]. Une introduction générale à ces techniques d'analyse de complexité peut être trouvée dans [34].

2.4.1 Ordinaux et notations ordinales

Un *ordinal* est un (représentant canonique d'une classe d'isomorphisme d')ordre linéaire bien fondé. Ainsi, les ordinaux finis n sont les ensembles $\{0, \ldots, n-1\}$ dotés de l'ordre habituel, tandis que l'ensemble des entiers naturels avec l'ordre habituel est l'ordinal ω ; la relation d'ordre $\leqslant$ coïncide avec la relation d'inclusion entre ordinaux et est bien fondée.

Forme normale de CANTOR. Les ordinaux peuvent être dotés d'une *arithmétique* : les opérations de somme directe $\alpha + \beta$, de produit direct $\alpha \cdot \beta$, et d'exponentiation α^β sont effet définies pour des ordinaux α et β. Ces opérations ne vérifient pas toutes les identités habituelles (par exemple, l'addition et la multiplication ne sont pas commutatives), mais se comportent néanmoins assez bien ; par exemple $\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma$, $\alpha^{\beta+\gamma} = \alpha^\beta \cdot \alpha^\gamma$, $\alpha \cdot 1 = \alpha$ et $\alpha^0 = 1$. Ces opérations coïncident avec les opérations usuelles sur les entiers si α et β sont des ordinaux finis ; par exemple, $3 = 1 + 1 + 1$.

Ces opérations permettent de représenter certains ordinaux comme des termes utilisant les opérations arithmétiques. Ainsi, il y a une bijection entre les ordinaux $\alpha < \omega^\omega$ et les termes de la forme $\omega^{k_1} + \cdots + \omega^{k_{n-1}} + \omega^{k_n}$ pour $n \geqslant 0$ tels que $k_1 \geqslant \cdots \geqslant k_{n-1} \geqslant k_n$ soient finis. Ce terme est appelé la *forme normale de* CANTOR de l'ordinal α. Si $n = 0$, l'ordinal associé est l'ordinal 0. Si $n > 0$ et $k_n = 0$, alors α s'écrit sous forme normale de CANTOR comme $\beta + 1$ pour $\beta = k_1 \geqslant \cdots \geqslant k_{n-1}$ et est appelé un ordinal *successeur*. Sinon et s'il n'est pas égal à 0, α est un ordinal *limite* et s'écrit sous forme normale de CANTOR comme $\beta + \omega^{k+1}$ où $\beta = \omega^{k_1} + \cdots + \omega^{k_{n-1}}$ et $k = k_n - 1$. À noter que ω^ω est lui aussi un ordinal limite.

Séquences fondamentales. Une *séquence fondamentale* pour un ordinal limite λ est une séquence strictement croissante $(\lambda(x))_{x<\omega}$ d'ordinaux $\lambda(x) < \lambda$ qui a λ pour limite supérieure. Dans le cas qui nous intéresse des ordinaux limites $\lambda \leqslant \omega^\omega$, il existe une définition standard des séquences fondamentales basée sur l'écriture de λ sous forme normale de CANTOR. Soit $\lambda = \omega^\omega$ et on pose alors

$$\lambda(x) = \omega^\omega(x) \stackrel{\text{def}}{=} \omega^{x+1}, \tag{2.9}$$

soit λ s'écrit sous forme normale de CANTOR comme $\beta + \omega^{k+1}$ pour un certain β et un certain k fini et alors

$$\lambda(x) = (\beta + \omega^{k+1})(x) \stackrel{\text{def}}{=} \beta + \omega^k \cdot (x+1). \tag{2.10}$$

Cet définition standard garantit en particulier que $0 < \lambda(x) < \lambda(y)$ pour tout $x < y$. Par exemple, $\omega(x) = x + 1$ et $(\omega^3 + \omega^3)(x) = \omega^3 + \omega^2 \cdot (x+1)$.

2.4.2 Séquences contrôlées

Dans cette section, il sera plus agréable d'exprimer la fonction de rang de la section 2.3.2 sur les séquences KLM en termes d'ordinaux. Si $\mathit{rang}(\xi) = (r_d, r_{d-1}, \ldots, r_0)$, alors on associera à la séquence KLM le *rang ordinal* $\alpha_\xi < \omega^{d+1}$ défini par

$$\alpha_\xi \stackrel{\text{def}}{=} \omega^d \cdot r_d + \omega^{d-1} \cdot r_{d-1} + \cdots + \omega^0 \cdot r_0. \tag{2.11}$$

L'équation (2.11) n'est réellement qu'une reformulation, car $\mathit{rang}(\xi) <_{lex} \mathit{rang}(\xi')$ si et seulement si $\alpha_\xi < \alpha_{\xi'}$. À une branche $\xi_0, \xi_1, \xi_2, \ldots$ de l'arbre de décomposition d'une séquence KLM ξ_0, nous associons ainsi une séquence strictement décroissante d'ordinaux

$$\alpha_{\xi_0} > \alpha_{\xi_1} > \alpha_{\xi_2} > \cdots \tag{2.12}$$

Par définition, les séquences strictement décroissantes d'ordinaux comme celle de l'équation (2.12) sont finies puisque l'ordre est bien fondé. Cependant, on ne peut généralement pas borner la longueur de ces séquences ; ainsi,

$$K+1 > K > K-1 > \cdots > 0 \quad \text{et} \quad \omega > K > K-1 > \cdots > 0 \tag{2.13}$$

sont deux exemples de séquences strictement décroissantes d'ordinaux de longueur $K+2$ pour tout K fini. Fort heureusement, une séquence strictement décroissante d'ordinaux de la forme (2.12) observée le long d'une branche de décomposition n'est pas arbitraire, car les ordinaux que nous y trouvons dépendent de la séquence KLM initiale ξ_0 ou résultent d'étapes de décomposition : on ne peut pas introduire une valeur K arbitraire comme dans les séquences de l'équation (2.13).

L'intuition précédente peut être formalisée par la notion de « séquence contrôlée ». Pour un ordinal $\alpha < \omega^\omega$ (ce qui est le cas des rangs ordinaux de l'équation (2.11)), on peut écrire $\alpha = \omega^n \cdot c_n + \cdots + \omega^0 \cdot c_0$ avec $n \geqslant 0$ et $0 \leqslant c_n, \ldots, c_0 < \omega$ des coefficients finis : c'est une écriture alternative de sa forme normale de CANTOR. On définit alors la *taille* de l'ordinal α comme

$$N\alpha \stackrel{\text{def}}{=} \max\{n, \max_{0 \leqslant i \leqslant n} c_i\}.$$

Ainsi, pour le rang ordinal α_ξ défini dans l'équation (2.11) pour une séquence KLM ξ de rang $rang(\xi) = (r_d, \ldots, r_0)$, on aura une taille

$$N\alpha_\xi = \max\{d, \max_{0 \leqslant i \leqslant d} r_i\}. \tag{2.14}$$

Soit maintenant n_0 un entier naturel dans $\mathbb{N}$ et $h\colon \mathbb{N} \to \mathbb{N}$ une fonction monotone croissante et inflationnaire, c'est-à-dire telle que $x \leqslant y$ implique $h(x) \leqslant h(y)$ et $x \leqslant h(x)$ pour tout x. Une séquence $\alpha_0, \alpha_1, \ldots$ d'ordinaux sous ω^ω est (n_0, h)-*contrôlée* si, pour tout indice $j \in \mathbb{N}$,

$$N\alpha_j \leqslant h^j(n_0), \tag{2.15}$$

c'est-à-dire si la taille du $(j+1)^{\text{e}}$ ordinal α_j de la séquence est bornée par la j^{e} itérée de la fonction h appliquée à n_0. En particulier, $N\alpha_0 \leqslant n_0$ pour le premier élément de la séquence.

Comme pour tout $n \in \mathbb{N}$, il n'existe qu'un nombre fini d'ordinaux $\alpha < \omega^\omega$ tels que $N\alpha \leqslant n$, par le lemme de KŐNIG la longueur des séquences strictement décroissantes contrôlées d'ordinaux sous ω^ω est bornée (voir par exemple [31]). On peut même donner une borne précise à l'aide de fonctions sous-récursives, dont nous allons maintenant rappeler les définitions.

Fonctions sous-récursives. La complexité dans le pire des cas d'algorithmes dont on a démontré la terminaison à l'aide d'une fonction de rangs dans des ordinaux peut être considérable. De manière à pouvoir exprimer de telles bornes de complexité, il est naturel d'employer des hiérarchies de fonctions sous-récursives, qui emploient une induction sur des termes ordinaux pour définir des fonctions qui croissent de plus en plus rapidement. Nous allons utiliser ici deux de ces hiérarchies, respectivement dues à HARDY et CICHOŃ [3]. Soit $h\colon \mathbb{N} \to \mathbb{N}$ une fonction. Relativement à une telle fonction h, la *hiérarchie de* HARDY $(h^\alpha)_{\alpha \leqslant \omega^\omega}$ et la *hiérarchie de* CICHOŃ $(h_\alpha)_{\alpha \leqslant \omega^\omega}$ sont deux familles de fonctions $h^\alpha\colon \mathbb{N} \to \mathbb{N}$ et $h_\alpha\colon \mathbb{N} \to \mathbb{N}$ définies par induction sur α par

$$\begin{aligned} h^0(x) &\stackrel{\text{def}}{=} x, & h_0(x) &\stackrel{\text{def}}{=} 0, \\ h^{\alpha+1}(x) &\stackrel{\text{def}}{=} h^\alpha\big(h(x)\big), & h_{\alpha+1}(x) &\stackrel{\text{def}}{=} 1 + h_\alpha\big(h(x)\big), \\ h^\lambda(x) &\stackrel{\text{def}}{=} h^{\lambda(x)}(x), & h_\lambda(x) &\stackrel{\text{def}}{=} h_{\lambda(x)}(x), \end{aligned}$$

où λ dénote un ordinal limite et $\lambda(x)$ le $(x+1)^{\text{e}}$ élément de sa séquence fondamentale comme défini dans les équations (2.9) et (2.10).

Les fonctions de HARDY h^α permettent de dénoter des itérées de la fonction de base h. Par exemple, la fonction h^k pour k fini est simplement

l'itérée k^{e} de h. Plus généralement, la fonction h^λ « diagonalise » à l'aide de la séquence fondamentale de λ pour exprimer des itérées qui dépendent de l'argument x. Ainsi, si on considère la hiérarchie de HARDY relativement à la *fonction successeur* $H(x) \stackrel{\text{def}}{=} x+1$, on peut voir que la première diagonalisation donne $H^\omega(x) = H^{x+1}(x) = 2x+1$. Le prochain ordinal limite est $\omega \cdot 2$, et là $H^{\omega\cdot 2}(x) = H^{\omega+\omega}(x) = H^{\omega+x+1}(x) = H^\omega(2x+1) = 4x+3$. Si on avance un peu, on trouvera une fonction de croissance exponentielle $H^{\omega^2}(x) = 2^{x+1}(x+1)-1$, puis une fonction non-élémentaire $H^{\omega^3}(x)$ proche d'une tour d'exponentielles de hauteur x, et une fonction non primitive-récursive $H^{\omega^\omega}(x)$ similaire à la fonction d'ACKERMANN.

Les fonctions de CICHOŃ h_α croissent à peu près à la même vitesse que les fonctions de HARDY. Si h est monotone croissante et inflationnaire, alors on peut vérifier par induction sur α que

$$h^\alpha(x) \geqslant h_\alpha(x) + x; \tag{2.16}$$

dans le cas où $h = H$ la fonction successeur, on a plus précisément $H^\alpha(x) = H_\alpha(x) + x$. Mais le principal intérêt des fonctions de CICHOŃ est qu'elles décrivent combien d'itérations sont effectuées par les fonctions de HARDY [3] :

$$h^\alpha(x) = h^{h_\alpha(x)}(x). \tag{2.17}$$

Théorème de longueur. Nous pouvons maintenant énoncer un « théorème de longueur » pour les séquences strictement décroissantes contrôlées d'ordinaux.

Théorème 2.4.1 ([31, thm. 3.3])**.** *Soit $n_0 \geqslant d+1$. La longueur maximale des séquences strictement décroissantes (n_0, h)-contrôlées d'ordinaux sous ω^{d+1} est $h_{\omega^{d+1}}(n_0)$.*

Si on combine le théorème 2.4.1 avec l'équation (2.17), l'image à retenir est celle de la figure 2.7 : la longueur d'une séquence (n_0, h)-contrôlée sous un ordinal α est bornée par $h_\alpha(n_0)$, tandis que la taille des ordinaux le long de cette séquence est bornée par $h^{h_\alpha(n_0)}(n_0) = h^\alpha(n_0)$.

2.4.3 Complexité à croissance rapide

Pour pouvoir exploiter des bornes exprimées à l'aide de fonctions de HARDY ou de CICHOŃ, nous allons utiliser des classes de complexité « à croissance rapide » définies dans [33]. Les *classes de complexité à croissance rapide* F_α sont indexées par des ordinaux α et sont définies à partir des fonctions de HARDY H^{ω^α} relatives à la fonction successeur $H(x) \stackrel{\text{def}}{=} x+1$.

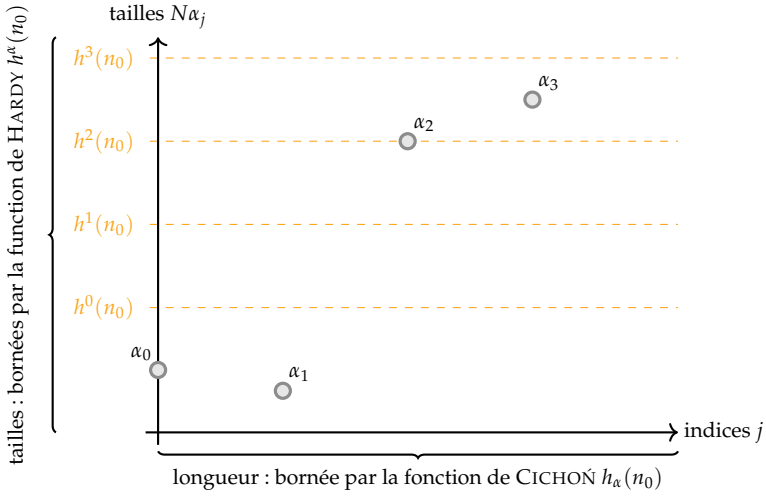


FIGURE 2.7 – *Schéma pour une séquence strictement décroissante d'ordinaux $\alpha_0 > \alpha_1 > \alpha_2 > \alpha_3 > \cdots$ sous α qui est (n_0, h)-contrôlée.*

On commence pour cela par définir

$$\mathscr{F}_{<\alpha} \stackrel{\text{def}}{=} \bigcup_{\beta<\omega^\alpha} \mathsf{FDTIME}\big(H^\beta(n)\big) \tag{2.18}$$

comme la classe de fonctions calculées par des machines de TURING déterministes en temps $O(H^\beta(n))$ pour un certain $\beta < \omega^\alpha$. On retrouve alors en particulier la classe des fonctions élémentaires comme $\mathscr{F}_{<3}$ et la classe des fonctions primitives-récursives comme $\mathscr{F}_{<\omega}$ [21, 38]. On définit ensuite pour $\alpha \geqslant 3$

$$\mathsf{F}_\alpha \stackrel{\text{def}}{=} \bigcup_{p\in\mathscr{F}_{<\alpha}} \mathsf{DTIME}\big(H^{\omega^\alpha}(p(n))\big) \tag{2.19}$$

comme la classe de problèmes de décision résolus par des machines de TURING déterministes en temps $O\big(H^{\omega^\alpha}(p(n))\big)$ pour une certaine fonction $p \in \mathscr{F}_{<\alpha}$. L'intuition derrière cette quantification sur p est que, tout comme par exemple $\mathsf{EXP} = \bigcup_{p\in\mathsf{poly}} \mathsf{DTIME}\big(2^{p(n)}\big)$ quantifie sur les fonctions polynomiales pour être une classe fermée par réductions polynomiales, la classe F_α est fermée par réductions issues de $\mathscr{F}_{<\alpha}$ [32, thms. 4.7 et 4.8].

Par exemple, $\mathsf{TOWER} \stackrel{\text{def}}{=} \mathsf{F}_3$ est la classe des problèmes qui peuvent être résolus en utilisant des ressources calculatoires (en temps ou en espace, ce qui est indifférent à ce niveau de complexité) bornées par une tour d'exponentielles de hauteur élémentaire dans la taille de l'entrée. L'union

$\bigcup_{k\in\mathbb{N}} \mathsf{F}_k$ est la classe des problèmes de décision primitifs-récursifs, et $\textsc{ackermann} \stackrel{\text{def}}{=} \mathsf{F}_\omega$ est la classe des problèmes qui peuvent être résolus en utilisant des ressources calculatoires bornées par la fonction d'ACKERMANN appliquée à une fonction primitive-récursive sur la taille de l'entrée. Voir la figure 2.3 dans l'introduction pour une vue schématique.

2.4.4 Borne supérieure de complexité

Considérons maintenant une branche $\xi_0, \xi_1, \dots$ de l'arbre de décomposition introduit dans la section 2.3. La séquence KLM initiale ξ_0 est de taille $n_0 \stackrel{\text{def}}{=} |\xi_0|$ bornée par la taille de l'entrée du problème d'accessibilité des SAVEs. Par le théorème 2.3.13, on sait que pour tout j, $\alpha_{\xi_{j+1}} < \alpha_{\xi_j}$ et que $|\xi_{j+1}| \leqslant e(|\xi_j|)$ pour une fonction élémentaire h que l'on peut supposer monotone croissante inflationnaire sans perte de généralité. La séquence de rangs ordinaux $\alpha_{\xi_0} > \alpha_{\xi_1} > \cdots$ est donc (n_0, h)-contrôlée. Par le théorème 2.4.1 et l'équation 2.16, la longueur de la branche est donc bornée par

$$L \stackrel{\text{def}}{=} h_{\omega^{d+1}}(n_0) \leqslant h^{\omega^{d+1}}(n_0). \tag{2.20}$$

Par l'équation (2.17), la taille des séquences KLM le long de cette branche est quant à elle bornée par

$$S \stackrel{\text{def}}{=} h^L(n_0) = h^{\omega^{d+1}}(n_0). \tag{2.21}$$

Le cardinal de $\mathrm{dec}(\xi_j)$ pour un nœud interne ξ_j est borné par $h(|\xi_j|) \leqslant S$, donc l'arbre de décomposition complet est de taille bornée par $S^L \leqslant S^S$. Le temps de calcul pour chaque nœud interne est borné par S, pour une complexité totale de construction de l'arbre en $e(S)$ pour une fonction élémentaire $e(x) \stackrel{\text{def}}{=} x \cdot x^x$.

Théorème 2.4.2. *Le problème d'accessibilité des SAVEs est dans* $\textsc{ackermann} = \mathsf{F}_\omega$.

Démonstration. Nous venons d'établir une borne supérieure de complexité pour l'algorithme de décomposition en $e\big(h^{\omega^{d+1}}(n)\big)$ pour deux fonctions élémentaires e et h. Par [33, thm. 4.2], $h^{\omega^{d+1}}(n) \leqslant H^{\omega^{d+4}}(f(n))$ pour une fonction élémentaire f, et par [33, lem. 4.6], $e(H^{\omega^{d+4}}(f(n))) \leqslant H^{\omega^{d+4}}(g(n))$ pour une fonction élémentaire g. On peut supposer $g(n) \geqslant d+3$ et on a alors $H^{\omega^{d+4}}(g(n)) \leqslant H^{\omega^{g(n)+1}}(g(n)) = H^{\omega^\omega}(g(n))$ avec $g \in \mathscr{F}_{<3} \subseteq \mathscr{F}_{<\omega}$.

Alternativement, au lieu de borner la complexité de l'algorithme par décomposition, on peut remarquer que l'équation (2.21) combinée

au lemme 2.3.12 implique que s'il existe un mot d'action $\sigma \in L_\xi$, alors sa longueur est bornée par $e(S)$ pour une fonction élémentaire e, ce qui est borné par $H^{\omega^{d+4}}(g(n))$ avec $g \in \mathscr{F}_{<3} \subseteq \mathscr{F}_{<\omega}$ par la même analyse que ci-dessus. Cela fournit un algorithme qui calcule $e(S)$ – ce qui peut être fait en temps élémentaire en S par [33, thm. 5.1] – et cherche un témoin $\sigma \in L_\xi$ de taille bornée par $e(S)$ de manière non-déterministe. ■

2.5 Borne inférieure de complexité

2.5.1 Un problème complet pour TOWER

Nous allons montrer que le problème de l'accessibilité des SAVEs est non élémentaire. Pour cela, nous fournissons une réduction en temps linéaire du problème canonique suivant. Écrivons $!^n$ pour l'itérée n^{e} de la fonction factorielle, de sorte que $a!^n = a\overbrace{!\cdots!}^{n}$. Le problème ci-dessous est complet pour la classe TOWER de tous les problèmes de décision qui peuvent se résoudre en temps ou en espace borné par une tour d'exponentielles dont la hauteur est une fonction élémentaire de la taille de l'entrée, à une réduction élémentaire près [33, sec. 2.3].

entrée Un programme à compteurs de taille n dont tous les compteurs sont testés.

question Existe-t'il une $3!^n$-exécution complète ?

2.5.2 Simuler des tests

Nous allons introduire une notion d'amplificateur pour un ratio, ainsi qu'un opérateur pour composer un tel amplificateur $\mathcal{A}$ avec un programme à compteurs $\mathcal{P}$. Sous l'hypothèse que le ratio R de l'amplificateur $\mathcal{A}$ soit égal à la borne B des compteurs testés par le programme, le résultat de la composition sera un programme $\mathcal{A} \triangleright \mathcal{P}$ équivalent pour lequel les compteurs testés deviennent des compteurs non testés (voir la proposition 2.5.2). Cela est obtenu en remplaçant les tests à zéro et les tests de valeur maximale du programme d'origine $\mathcal{P}$ par une simulation, laquelle utilise l'amplificateur $\mathcal{A}$ pour vérifier que les simulations sont correctes. Le prix à payer pour cette construction est l'introduction d'un compteur supplémentaire pour chaque compteur testé du programme d'origine.

Les amplificateurs peuvent avoir des compteurs testés. Un amplificateur dont les compteurs testés sont bornés par B et dont le ratio est un entier R plus grand que B, appelé un B-amplificateur par R, peut alors être

vu, en appliquant l'opérateur de composition, comme un moyen de transformer des programmes dont les compteurs testés sont bornés par R en des programmes équivalents dont les compteurs testés sont bornés par B. Dans le cas particulier d'un amplificateur sans compteur testé, le programme obtenu sera ainsi sans compteur testé.

Une autre particularité, qui sera la clé de la section 2.5.4, est que des amplificateurs de plus grand ratio sont obtenus par composition : en appliquant l'opérateur à un B-amplificateur par B' sur un B'-amplificateur par B'', nous obtenons un B-amplificateur par B''.

Amplificateur pour un ratio. Supposons que :

— B et R soient des entiers strictement positifs ;

— $\mathcal{A}$ soit un *B-amplificateur par R*, c'est-à-dire un programme à compteurs tel que la relation B-calculée dans trois compteurs distingués $\mathsf{b}, \mathsf{c}, \mathsf{d}$ soit

$$\{\langle b, c, d\rangle \ : \ b = R,\ c > 0,\ d = c \cdot b\};$$

— $\mathcal{P}$ soit un programme à compteurs.

Exemple 2.5.1. *Par exemple, quand R est suffisamment petit pour écrire explicitement R incrémentations explicitement, il est très facile de programmer un amplificateur par R (qui sera utilisé plus tard) :*

```
1: b += R                                  // mettre b à la valeur R
2: c += 1   d += R
3: loop
4:     c += 1   d += R
5: halt.
```

Observons que cet amplificateur n'a pas de compteur testé. Ainsi, pour tout entier positif B, c'est un B-amplificateur par R.

Opérateur de composition. Sous les hypothèses précédemment données, nous définissons maintenant une construction d'un programme $\mathcal{A} \triangleright \mathcal{P}$ qui B-calcule toutes les relations qui sont R-calculées par $\mathcal{P}$. L'idée est de transformer chaque compteur testé x de $\mathcal{P}$ en un compteur non testé à l'aide d'un nouveau compteur complémentaire $\widehat{\mathsf{x}}$, de sorte que l'invariant $\mathsf{x} + \widehat{\mathsf{x}} = R$ soit maintenu. Ainsi, chaque test à zéro de x peut être remplacé par une boucle qui incrémente R fois le compteur x puis décrémente R fois x, et chaque test à la valeur maximale est remplacé de manière similaire. Le compteur b fourni par $\mathcal{A}$ est utilisé pour initialiser chaque compteur complémentaire $\widehat{\mathsf{x}}$, alors que c et d sont utilisés pour être sûr que si d est nul à la fin de l'exécution alors toutes les boucles de simulation des tests

à zéro et à la valeur maximale ont bien été itérées R fois comme requis. Concrètement, le programme $\mathcal{A} \rhd \mathcal{P}$ est construit de la façon suivante :

1. les compteurs sont renommés si nécessaire de sorte qu'aucun compteur n'apparaisse à la fois dans $\mathcal{A}$ et $\mathcal{P}$;
2. en notant $x_1, \ldots, x_l$ les compteurs testés de $\mathcal{P}$, des nouveaux compteurs $\widehat{x}_1, \ldots, \widehat{x}_l$ sont introduits et le code suivant est inséré au début de $\mathcal{P}$:

   loop
   $\widehat{x}_1 \mathrel{+}= 1 \quad \cdots \quad \widehat{x}_l \mathrel{+}= 1$
   $b \mathrel{-}= 1 \quad d \mathrel{-}= 1$
   $c \mathrel{-}= 1$

   (Nous montrerons que les exécutions complètes doivent nécessairement itérer les boucles R fois, c'est-à-dire jusqu'à ce que le compteur b devienne nul) ;
3. chaque commande $x_i \mathrel{+}= 1$ de $\mathcal{P}$ est remplacée par deux commandes
   $x_i \mathrel{+}= 1 \quad \widehat{x}_i \mathrel{-}= 1$;
4. chaque commande $x_i \mathrel{-}= 1$ de $\mathcal{P}$ est remplacée par des commandes
   $x_i \mathrel{-}= 1 \quad \widehat{x}_i \mathrel{+}= 1$;
5. chaque commande **zero?** x_i de $\mathcal{P}$ est remplacée par le code suivant :

   loop
   $x_i \mathrel{+}= 1 \quad \widehat{x}_i \mathrel{-}= 1$
   $d \mathrel{-}= 1$
   $c \mathrel{-}= 1$
   loop
   $x_i \mathrel{-}= 1 \quad \widehat{x}_i \mathrel{+}= 1$
   $d \mathrel{-}= 1$
   $c \mathrel{-}= 1$

   (Nous montrerons que les exécutions complètes doivent nécessairement itérer les boucles R fois. Ainsi on teste que x_i est égal à 0 en vérifiant que $\widehat{x}_i$ est égal à R en transférant R de $\widehat{x}_i$ vers x_i et puis en transférant à nouveaux pour rétablir la valeur des compteurs) ;
6. chaque commande **max?** x_i de $\mathcal{P}$ est remplacée de manière analogue, c'est-à-dire par le même code que pour **zero?** x_i mais les incrémentations et les décrémentations de x_i et $\widehat{x}_i$ sont échangées ;
7. en notant $y_1, \ldots, y_m$ (respectivement, $z_1, \ldots, z_h$) les compteurs qui doivent être à zéro à la terminaison de $\mathcal{A}$ (respectivement, $\mathcal{P}$), le code de $\mathcal{A} \rhd \mathcal{P}$ est obtenu en concaténant le code de $\mathcal{A}$ avec le code de $\mathcal{P}$ modifié comme énoncé, tous deux sans leur commande **halt**, et en terminant par la commande suivante :

halt if $\mathsf{d}, \mathsf{y}_1, \ldots, \mathsf{y}_m, \mathsf{z}_1, \ldots, \mathsf{z}_h = 0$.

Remarquons que la simulation des tests à zéro des compteurs bornés par R en utilisant des transferts depuis et vers un compteur complémentaire est une technique bien connue déjà utilisée par LIPTON [20]. La nouveauté ici est que la vérification de toutes les simulations pendant une exécution est obtenue en diminuant convenablement les compteurs d et c dont le ratio est R, et en vérifiant seulement à la fin que d est bien nul.

Correction. La proposition suivante énonce que $\mathcal{A} \triangleright \mathcal{P}$ est correct : les relations B-calculées dans les compteurs de $\mathcal{P}$ sont les mêmes que celles R-calculées par $\mathcal{P}$. (Nous faisons ici un renommage implicite des compteurs dans l'étape (1) de la construction.) Dans une direction, la preuve s'obtient en remarquant que $\mathcal{A} \triangleright \mathcal{P}$ peut simuler précisément n'importe quelle R-exécution de $\mathcal{P}$. Dans l'autre direction, il suffit de remarquer que même si les boucles introduites dans les étapes (5) et (6) peuvent être itérées moins de R fois et ainsi valider un test par erreur, la façon dont les compteurs c et d sont initialisés par $\mathcal{A}$ puis utilisés dans la construction implique qu'une telle exécution ne peut pas se prolonger en une exécution complète. Informellement, dès qu'une boucle de simulation d'un des tests est itérée moins de R fois, l'égalité $\mathsf{d} = \mathsf{c} \cdot R$ devient une inégalité stricte $\mathsf{d} > \mathsf{c} \cdot R$ et reste ainsi sur tout le reste de l'exécution, empêchant *in fine* le compteur d d'atteindre zéro.

Proposition 2.5.2. *Pour toute valeur des compteurs de $\mathcal{P}$, elle apparaît à la fin d'une B-exécution complète de $\mathcal{A} \triangleright \mathcal{P}$ si et seulement si elle apparaît à la fin d'une R-exécution de $\mathcal{P}$.*

Démonstration. La direction « si » est aisée : d'une R-exécution complète de $\mathcal{P}$ avec un total de q tests à zéro et à la valeur maximale, on obtient une B-exécution de $\mathcal{A} \triangleright \mathcal{P}$ avec la même valeur finale des compteurs de $\mathcal{P}$ en

- exécutant $\mathcal{A}$ pour terminer avec $\mathsf{b} = R$, $\mathsf{c} = 2q + 1$, $\mathsf{d} = \mathsf{c} \cdot R$ et tous les compteurs $\mathsf{y}_1, \ldots, \mathsf{y}_m$ égaux à 0. Ces derniers ne seront pas modifié par le reste de l'exécution et satisfont ainsi la condition finale d'être à zéro (voir étape (7) de la construction),
- en itérant la boucle de l'étape (2) R fois pour initialiser les compteurs complémentaires $\widehat{\mathsf{x}}_i$ à R, ce qui soustrait R et 1 de d et c (respectivement) et décroît b à 0, et
- à chaque endroit où un test est effectué dans $\mathcal{P}$, en itérant les deux boucles des étapes (5) ou (6) (respectivement) R fois, ce qui retranche $2R$ et 2 de d et c (respectivement), ce qui fini par les faire aller à 0.

Pour la direction « seulement si », considérons une B-exécution complète de $\mathcal{A} \rhd \mathcal{P}$. Déduire de cette exécution une R-exécution complète de $\mathcal{P}$ avec les mêmes valeurs pour les compteurs de $\mathcal{P}$ est facile si l'on a montré que, pour chaque simulation d'une commande **zero?** x_i ou **max?** x_i par le code à l'étape (5) ou (6) de la construction, les valeurs de x_i au début et à la fin du code sont 0 ou R (respectivement).

D'abord, par l'étape (7) et le fait que les compteurs $\mathsf{y}_1, \ldots, \mathsf{y}_m$ ne sont pas utilisés par la partie du code après $\mathcal{A}$, nous savons que les valeurs de b, c et d qui ont étés fournis par $\mathcal{A}$ vérifient $\mathsf{b} = R$ et $\mathsf{d} = \mathsf{c} \cdot R$. Après le code de l'étape (2) nous avons donc $\mathsf{x}_i + \widehat{\mathsf{x}}_i \leqslant R$ pour tout i. En utilisant le raisonnement évoqué dans l'exemple 2.2.8 et en faisant une preuve en avant le long de l'exécution, nous déduisons que

$$\mathsf{x}_i + \widehat{\mathsf{x}}_i \leqslant R \text{ pour tout } i, \text{ et } \mathsf{d} \geqslant \mathsf{c} \cdot R$$

est un invariant maintenu pour le reste de l'exécution.

Maintenant, comme vu à l'étape (7), d doit être nul à la fin, et donc l'inégalité $\mathsf{d} \geqslant \mathsf{c} \cdot R$ est au final une égalité. Ainsi, c est nul aussi à la fin. Rappelons à nouveau le raisonnement appliqué dans l'exemple 2.2.8. En faisant une preuve en arrière le long de l'exécution, on déduit qu'en fait $\mathsf{d} = \mathsf{c} \cdot R$ a été maintenu et donc que, pour chaque simulation des commandes **zero?** x_i et **max?** x_i, chacune des deux boucles a été itérée exactement R fois, et donc les valeurs de x_i au début et à la fin de chaque simulation sont bien comme elles doivent l'être. Ainsi, la boucle introduite à l'étape (2) a été itérée R fois, et b est zéro au final. ■

2.5.3 Un amplificateur factoriel

Cette section est un peu technique. Elle montre l'existence d'un programme $\mathcal{F}$ appelé l'*amplificateur factoriel* et qui, pour chaque entier strictement positif k, est un k-amplificateur par $k!$. En composant $\mathcal{F}$ avec lui-même, nous aurons tous les ingrédients nécessaires pour montrer notre résultat principal de la section 2.5.4 : nous obtiendrons en effet un amplificateur par un ratio qui sera une tour d'exponentielles.

Un programme simple. En guise d'échauffement avant d'attaquer la présentation du programme principal et de sa preuve de correction, considérons le programme simplifié $\mathcal{E}$ spécifié dans l'algorithme 2.1. Deux macros sont utilisées pour aider la lisibilité. Nous donnons maintenant leurs contenus, en notant qu'elles utilisent de façon cachée un autre compteur i' :

$\mathsf{x} \mathrel{-}= \mathsf{i}$ **:** Pour retrancher la valeur du compteur i, nous utilisons un compteur auxiliaire i' dans lequel la valeur de i est transférée puis

restaurée. Au début du code, i' est supposé est nul, et la même propriété est garantie à la fin.

```
loop
    i −= 1    i′ += 1    x −= 1
zero? i
loop
    i′ −= 1    i += 1
zero? i′
```

$x' \mathrel{+}= i + 1$ **:** Cette macro est très similaire, mise à part l'incrémentation supplémentaire de x'.

```
x′ += 1
loop
    i −= 1    i′ += 1    x′ += 1
zero? i
loop
    i′ −= 1    i += 1
zero? i′
```

Algorithme 2.1 : Le programme à compteurs $\mathcal{E}$.

```
   //Compteurs non testés : x, y, x′
   //Compteurs testés : i, i′
 1: i += 1    x += 1    y += 1
 2: loop
 3:     x += 1    y += 1
 4: loop
 5:     loop
 6:         x −= i    x′ += i+1
 7:     loop
 8:         x′ −= 1    x += 1
 9:     i += 1
10: max? i
11: loop
12:     x −= i    y −= 1
13: halt if y = 0
```

Le programme $\mathcal{E}$ a des compteurs non testés x, x' et y, et les compteurs testés i et i'. En supposant que la borne des compteurs testés est l'entier positif k, le programme fait les opérations suivantes :

— il initialise x et y à un entier strictement positif a choisi de façon non déterministe, qui restera inchangé dans le compteur y jusqu'à la boucle finale ;

— à chaque itération de la boucle principale, il utilise le compteur x' pour essayer de multiplier le compteur x par la fraction $(i+1)/i$;

— par la boucle finale et le test final du compteur y à zéro, il termine si la valeur de x est au moins $a \cdot k$ (dans ce cas, la valeur sera exactement $a \cdot k$).

On peut se convaincre que, pour tout entier positif a, il existe une k-exécution complète de $\mathcal{E}$ qui initialise x et y à a, et multiplie x exactement par toutes les factions $(i+1)/i$ pour $i = 1, \ldots, k-1$, et vérifie au final que x est égal à $a \cdot k$.

Réciproquement, toute k-exécution complète de $\mathcal{E}$ est de cette forme. Comme indice, on peut remarquer que dès que la multiplication de x par la fraction $(i+1)/i$ est incomplète (soit parce que la première boucle interne ne fait pas décroître x vers zéro, soit parce que la seconde boucle interne ne fait pas décroître x′ à zéro), il sera impossible de réparer cette erreur avec le reste de l'exécution, car la valeur de x à la fin de la boucle principale sera nécessairement plus petite que $a \cdot k$ et donc, il sera impossible de compléter cette exécution. Nous remarquons aussi que cette propriété s'appuie sur le fait que toutes les fractions $(i+1)/i$ sont strictement plus grandes que 1.

L'amplificateur factoriel. La définition de $\mathcal{F}$ dans l'algorithme 2.2 est présentée dans un code de haut niveau : en plus des deux macros pour soustraire i et ajouter i + 1 présentées précédemment, nous utilisons une macro supplémentaire :

loop at most b **times** <*body*> **:** Pour exprimer cette construction, nous utilisons un compteur auxiliaire b′ vers lequel la valeur de b est transférée et rétablie. Sous l'hypothèse que b′ soit nul au départ, la boucle est effectivement itérée au plus b fois.

```
loop
   b −= 1    b′ += 1
loop
   b′ −= 1    b += 1
   <body>
```

Observons que les compteurs non testés du programme $\mathcal{F}$ sont b, b′, c, c′, d, d′, x et y, et que les compteurs testés sont i et i′ (le compteur i′ est caché dans les macros).

Correction de l'amplificateur factoriel. Avant de montrer que, pour tout entier strictement positif k, le programme $\mathcal{F}$ est un k-amplificateur par k! – ce qui est l'argument central de notre construction – donnons quelques intuitions :

— le compteur d est utilisé pour préserver la valeur de x à la fin de la boucle principale, vu que x est modifié par la boucle finale ;

Algorithme 2.2 : L'amplificateur factoriel $\mathcal{F}$.

```
    //Compteurs non testés : b, b′, c, c′, d, d′, x, y
    //Compteurs testés : i, i′
 1: i += 1   b += 1   c += 1   d += 1   x += 1   y += 1
 2: loop
 3:     c += 1   d += 1   x += 1   y += 1
 4: loop
 5:     loop
 6:         c −= i   c′ += 1
 7:         loop at most b times
 8:             d −= i   x −= i   d′ += i + 1
 9:     loop
10:         b −= 1   b′ += i + 1
11:     loop
12:         b′ −= 1   b += 1
13:     loop
14:         c′ −= 1   c += 1
15:         loop at most b times
16:             d′ −= 1   d += 1   x += 1
17:     i += 1
18: max? i
19: loop
20:     x −= i   y −= 1
21: halt if y = 0
```

— le compteur d′ joue le rôle d'un compteur auxiliaire à la fois pour d et x, ainsi il n'est pas nécessaire d'avoir un compteur x′ ;
— le compteur c est initialisé au même entier strictement positif a que d, x et y, alors que le compteur b est initialisé à 1 ;
— au début de n'importe quelle itération de la boucle principale d'une exécution complète, l'invariant $\mathsf{d} = \mathsf{c} \cdot \mathsf{b}$ sera vérifié, ainsi, la première itération de la boucle interne divisera c par i précisément ;
— afin que la dernière boucle interne transfère complètement d′ vers d et x, les deux boucles internes du milieu multiplieront nécessairement b par $\mathsf{i} + 1$ précisément ;
— à la fin de la boucle principale, d, c et b auront les valeurs $a \cdot k$, $a/(k-1)!$ et $k!$ (respectivement), et en particulier a est nécessairement divisible par $(k-1)!$.

Lemme 2.5.3. *Pour tout entier strictement positif k, le programme $\mathcal{F}$ est un k-amplificateur par $k!$, c'est-à-dire que la relation k-calculée dans les compteurs*

$\mathsf{b}, \mathsf{c}, \mathsf{d}$ *est*

$$\{\langle b, c, d\rangle \ : \ b = k!,\ c > 0,\ d = c \cdot b\}.$$

Démonstration. Nous considérons des k-exécutions de $\mathcal{F}$ dont les compteurs sont nuls initialement, et qui sont soit terminées soit bloquées par la commande **halt** parce que y n'est pas nul. En particulier, une telle exécution aura exécuté la boucle principale, pour chaque valeur $\mathsf{i} = 1, \ldots, k-1$. Ainsi, nous pouvons introduire les notations suivantes pour les valeurs des compteurs durant la i^{e} itération de la boucle, où v est l'un des compteurs b, b', c, c', d, d' :

$\bar{\mathsf{v}}_i$ **:** La valeur finale de v après les lignes 5–10 ;

v_i **:** La valeur finale de v après les lignes 11–16.

Il sera aussi pratique de noter v_0 pour la valeur de v au début de la première itération de la boucle principale. Ces notations sont naturellement relatives à l'exécution que l'on considère, qui pour des raisons de lisibilité n'est pas explicitée.

La preuve fonctionne pour tout entier strictement positif k et se décompose en deux parties, établissant les inclusions entre la relation k-calculée par $\mathcal{F}$ dans les compteurs $\mathsf{b}, \mathsf{c}, \mathsf{d}$ et la relation donnée dans l'énoncé du lemme.

La première inclusion, qui suppose que $b = k!$, $c > 0$ et $d = c \cdot b$ et montre que $\mathcal{F}$ a une k-exécution complète dont la valeur finale des compteurs $\mathsf{b}, \mathsf{c}, \mathsf{d}$ est exactement b, c, d, découle de la proposition suivante.

Proposition 2.5.4. *Pour tout a divisible par $(k-1)!$, le programme $\mathcal{F}$ a une k-exécution complète qui satisfait les égalités suivantes :*

$$\begin{array}{lll}
\mathsf{b}_0 = 1 & \mathsf{c}_0 = a & \mathsf{d}_0 = a \\
\mathsf{b}'_0 = 0 & \mathsf{c}'_0 = 0 & \mathsf{d}'_0 = 0 \\
\bar{\mathsf{b}}_i = 0 & \bar{\mathsf{c}}_i = 0 & \bar{\mathsf{d}}_i = 0 \\
\bar{\mathsf{b}}'_i = \mathsf{b}_{i-1} \cdot (i+1) & \bar{\mathsf{c}}'_i = \mathsf{c}_{i-1}/i & \bar{\mathsf{d}}'_i = \mathsf{d}_{i-1} \cdot (i+1)/i \\
\mathsf{b}_i = \bar{\mathsf{b}}'_i & \mathsf{c}_i = \bar{\mathsf{c}}'_i & \mathsf{d}_i = \bar{\mathsf{d}}'_i \\
\mathsf{b}'_i = 0 & \mathsf{c}'_i = 0 & \mathsf{d}'_i = 0
\end{array}$$

Preuve de la proposition 2.5.4. Une telle exécution peut être construite en itérant les boucles internes non-déterministes le nombre maximal de fois. C'est-à-dire, durant l'itération i de la boucle principale :

— la boucle à la ligne 5 est itérée c_{i-1}/i fois et chaque passe de la boucle à la ligne 7 est itérée b_{i-1} fois ;

— la boucle à la ligne 9 est itérée b_{i-1} fois ;

— la boucle à la ligne 11 est itérée $\bar{\mathsf{b}}_i$ fois ;

— la boucle à la ligne 13 est itérée $\bar{c}'_i$ fois et chaque passe de la boucle à la ligne 15 est itérée b_i fois.

La divisibilité de a par $(k-1)!$ assure que toutes les divisions introduites précédemment sont entières.

Pour voir qu'une telle exécution peut être complétée, observons que l'on peut déduire des égalités données de l'énoncé que

$$b_{k-1} = \prod_{i=1}^{k-1} (i+1) = k! \quad c_{k-1} = a \cdot \prod_{i=1}^{k-1} \frac{1}{i} = \frac{a}{(k-1)!} \quad d_{k-1} = a \cdot \prod_{i=1}^{k-1} \frac{i+1}{i} = a \cdot k.$$

En particulier, au début de la boucle finale (à la ligne 19), le compteur x est égal au compteur d et donc contient la valeur $a \cdot k$, et le compteur y a la valeur a. En itérant la boucle finale a fois, nous réduisons y (et x) à zéro comme cela est demandé. □

Pour obtenir b, c, d comme valeurs finales des compteurs b, c, d, nous appliquons la proposition 2.5.4 avec $a = c \cdot (k-1)!$.

Nous nous attaquons maintenant à l'inclusion réciproque, qui considère une k-exécution complète et montre que les valeurs finales b, c, d des compteurs b, c, d vérifient $b = k!$, $c > 0$ et $d = c \cdot b$.

Proposition 2.5.5. *Pour tout $i = 1, \ldots, k-1$, nous avons :*

— $\bar{d}_i + \bar{d}'_i \leqslant (d_{i-1} + d'_{i-1}) \cdot (i+1)/i$;

— $\bar{d}_i + \bar{d}'_i = (d_{i-1} + d'_{i-1}) \cdot (i+1)/i$ *si et seulement si* $\bar{d}_i = d'_{i-1} = 0$;

— $d_i + d'_i = \bar{d}_i + \bar{d}'_i$.

Preuve de la proposition 2.5.5. Simple calcul basé sur $(i+1)/i > 1$. □

Soit a la valeur des compteurs c, d, x et y au début de la boucle principale.

Proposition 2.5.6. *Les égalités de la proposition 2.5.4 sur les valeurs des compteurs* d *et* d′ *sont satisfaites.*

Preuve de la proposition 2.5.6. D'abord, rappelons qu'au début de la boucle finale (à la ligne 19) les compteurs x et d sont égaux, et par la proposition 2.5.5 leur valeur est au plus $a \cdot k$. Vu que le compteur y a la valeur a à ce point et que l'exécution est complète, il est nécessaire que la valeur de x à ce moment là soit exactement $a \cdot k$. À nouveau par la proposition 2.5.5, nous déduisons que pour tout $i = 1, \ldots, k-1$, nous avons effectivement :

$$\bar{d}_i = 0 \qquad \bar{d}'_i = d_{i-1} \cdot \frac{i+1}{i} \qquad d_i = \bar{d}'_i \qquad d'_i = 0. \qquad \square$$

Proposition 2.5.7. *L'entier a est divisible par* $(k-1)!$ *et les égalités de la proposition 2.5.4 pour les valeurs des compteurs* b, b', c *et* c' *sont satisfaites.*

Preuve de la proposition 2.5.7. Le fait que a soit divisible par $(k-1)!$ sera une conséquence immédiate des égalités que nous montrerons pour les compteurs c et c', vu qu'elles invoqueront une division de a par $(k-1)!$.

Pour le reste de la proposition, nous faisons une preuve par induction où l'hypothèse est que les égalités pour les valeurs de b, b', c et c' sont satisfaites pour tous les indices strictement plus petits que i. On déduit, en utilisant la proposition 2.5.6, que nous avons

$$\mathsf{d}_{i-1} = \mathsf{c}_{i-1} \cdot \mathsf{b}_{i-1}. \tag{2.22}$$

Considérons l'itération i de la boucle principale. Nous déduisons de la proposition 2.5.6 que les commandes à la ligne 8 doivent être exécutées d_{i-1}/i fois. Donc, les valeurs des compteurs b et b' restent inchangés jusqu'à la ligne 9. En utilisant l'équation (2.22) nous déduisons que les commandes à la ligne 6 doivent être exécutées c_{i-1}/i fois, et nous avons :

$$\bar{\mathsf{c}}_i = 0 \qquad \bar{\mathsf{c}}'_i = \mathsf{c}_{i-1}/i.$$

De plus, par la proposition 2.5.6, la commande à la ligne 16 doit être exécutée $\bar{\mathsf{d}}'_i = \mathsf{d}_{i-1} \cdot (i+1)/i$ fois. D'après ce que nous venons de montrer, ce nombre est égal à $\bar{\mathsf{c}}'_i \cdot \mathsf{b}_{i-1} \cdot (i+1)$, et on peut ainsi conclure :

$$\begin{array}{lcllcllcl} \bar{\mathsf{b}}_i & = & 0 & & \mathsf{b}_i & = & \bar{\mathsf{b}}'_i & \mathsf{c}_i & = & \bar{\mathsf{c}}'_i \\ \bar{\mathsf{b}}'_i & = & \mathsf{b}_{i-1} \cdot (i+1) & & \mathsf{b}'_i & = & 0 & \mathsf{c}'_i & = & 0. \end{array} \qquad \square$$

Comme dans la première partie, nous déduisons que les valeurs finales b, c, d des compteurs $\mathsf{b}, \mathsf{c}, \mathsf{d}$ sont $k!$, $a/(k-1)!$, $a \cdot k$, et en particulier $c \cdot b = a \cdot k!/(k-1)! = d$. ce qui conclut la preuve du lemme 2.5.3. ■

2.5.4 Borne inférieure de complexité

Comme déjà mentionné, afin de montrer que le problème de l'accessibilité des SAVEs est TOWER-dur, il suffit de montrer comment construire un amplificateur sans compteur testé, avec un ratio qui est une tour d'exponentielles. Toutes les pièces nécessaires ont été données dans les sections 2.5.2 et 2.5.3, et ici nous les assemblons pour obtenir une telle construction en temps linéaire (bien qu'une complexité élémentaire aurait suffi pour la borne inférieure TOWER).

Lemme 2.5.8. *Un amplificateur par* $3!^n$ *sans compteur testé est calculable en temps* $O(n)$.

Démonstration. Notons $\mathcal{A}$ l'amplificateur trivial par 3 (c.f. exemple 2.5.1). Le programme

$$\overbrace{((\mathcal{A} \rhd \mathcal{F}) \rhd \mathcal{F}) \rhd \cdots \mathcal{F}}^{n \text{ compositions}}$$

est un amplificateur par $3!^n$ sans compteur testé par la proposition 2.5.2 et le lemme 2.5.3, et il est calculable en temps $O(n)$ par définition de l'opérateur de composition (c.f. section 2.5.2). ■

Théorème 2.5.9. *Le problème de l'accessibilité des SAVEs est* TOWER*-difficile.*

Démonstration. Nous réduisons depuis le problème TOWER-complet de l'arrêt d'un programme à compteurs de taille n dont tous les compteurs sont bornés par $3!^n$. Soit $\mathcal{M}$ un tel programme, et soit $\mathcal{T}$ un amplificateur par $3!^n$ sans compteur testé ; un tel amplificateur est calculable en temps $O(n)$ d'après le lemme 2.5.8. Le programme $\mathcal{T} \rhd \mathcal{M}$ est alors sans compteur testé, et d'après la proposition 2.5.2 il a une exécution complète si et seulement si le programme $\mathcal{M}$ en a une. ■

2.6 Conclusion

La décidabilité du problème d'accessibilité des SAVs joue un rôle central dans de nombreux résultats en informatique théorique et sa complexité algorithmique impacte celle de quantité d'autres problèmes [32]. Malgré les avancées récentes, avec une borne inférieure $\mathsf{F}_3 = \mathsf{TOWER}$ [4] présentée dans la section 2.5 et une borne supérieure $\mathsf{F}_\omega = \mathsf{ACKERMANN}$ [18] présentée dans la section 2.4, cette complexité exacte n'est pas connue ; voir la figure 2.3 pour mesurer le gouffre qui sépare la borne inférieure de la borne supérieure.

Algorithme par décomposition. Il semble pour l'instant difficile d'améliorer la borne inférieure pour le problème de décision. En revanche, on peut argumenter que l'arbre de décomposition KLM d'un SAVE peut atteindre une taille ackermannienne [17]. Mieux, cette décomposition permet de calculer la clôture par le bas du langage d'un SAVE étiqueté [8], et peut donc servir à résoudre le problème d'inclusion entre les clôtures par le bas des langages de deux SAVEs étiquetés : comme ce problème est connu comme ACKERMANN-difficile [39], la borne supérieure de la section 2.4 montre qu'il est en fait ACKERMANN-complet. Autrement dit, il n'y a pas d'espoir d'améliorer la borne supérieure à l'aide d'un algorithme qui construit une décomposition KLM en entier.

Algorithme par invariants inductifs semi-linéaires. Développer de nouveaux algorithmes pour l'accessibilité des SAVs pourrait être le meilleur espoir pour améliorer les bornes supérieures. En l'occurrence, il existe déjà une alternative à l'algorithme par décomposition. LEROUX [16] a montré que, si la configuration cible $\boldsymbol{c}_{out}$ n'est pas accessible depuis la configuration source $\boldsymbol{c}_{in}$, alors il existe un invariant inductif I semi-linéaire qui contient $\boldsymbol{c}_{in}$ mais pas $\boldsymbol{c}_{out}$; ici, un *invariant inductif* est un ensemble fermé par application des actions du système. Cela mène à deux semi-algorithmes : l'un énumère les exécutions du système et s'arrête si on atteint la configuration cible, tandis que l'autre énumère les ensembles semi-linéaires et s'arrête s'il y en a un qui est un invariant inductif, contient $\boldsymbol{c}_{in}$ et ne contient pas $\boldsymbol{c}_{out}$. Cependant, la seule preuve constructive connue de l'existence de I, donnée dans [15], utilise la décomposition KLM, et on ne sait pas donner de meilleure borne de complexité pour cet algorithme.

Dimension fixée. Un axe de recherche que nous n'avons pas exploré dans ce chapitre consiste à étudier le problème d'accessibilité quand la dimension d est fixée. Quand $d = 2$, la relation d'accessibilité d'un SAVE est semi-linéaire [10, 19], et le problème d'accessibilité est PSPACE-complet si les vecteurs sont encodés en binaire [1] et NL-complet s'ils le sont en unaire [5].

Pour une dimension $d \geqslant 3$ fixée, il n'y a cependant pour l'instant guère d'espoir d'obtenir des résultats aussi précis ; en particulier, la relation d'accessibilité cesse d'être semi-linéaire [10]. À ce jour, tout ce que l'on sait est que le problème est dans F_{d+4} [18] et est $(d-13)$-EXPSPACE-difficile si $d \geqslant 14$ [4] et PSPACE-difficile sinon [1].

Bibliographie

[1] M. BLONDIN, A. FINKEL, S. GÖLLER, C. HAASE et P. MCKENZIE : Reachability in two-dimensional vector addition systems with states is PSPACE-complete. *Actes de LICS'15*, pp. 32–43. IEEE Press, 2015. doi:10.1109/LICS.2015.14.

[2] D. CHISTIKOV et C. HAASE : The taming of the semi-linear set. *Actes de ICALP'16*, vol. 55 de *Leibniz Int. Proc. Inf.*, pp. 128:1–128:13. LZI, 2016. doi:10.4230/LIPIcs.ICALP.2016.128.

[3] E. A. CICHOŃ et E. TAHHAN BITTAR : Ordinal recursive bounds for Higman's Theorem. *Theoretical Comput. Sci.*, 201(1–2):63–84, 1998. doi:10.1016/S0304-3975(97)00009-1.

[4] W. CZERWIŃSKI, S. LASOTA, R. LAZIĆ, J. LEROUX et F. MAZOWIECKI : The reachability problem for Petri nets is not elementary. *Actes de STOC'19*, pp. 24–33. ACM, 2019. doi:10.1145/3313276.3316369.

[5] M. ENGLERT, R. LAZIĆ et P. TOTZKE : Reachability in two-dimensional unary vector addition systems with states is NL-complete. *Actes de LICS'16*, pp. 477–484. ACM, 2016. doi:10.1145/2933575.2933577.

[6] J. ESPARZA : Decidability and complexity of Petri net problems — an introduction. *Dans Lectures on Petri Nets I: Basic Models*, vol. 1491 de *Lect. Notes Comput. Sci.*, pp. 374–428. Springer, 1998. doi:10.1007/3-540-65306-6_20.

[7] S. A. GREIBACH : Remarks on blind and partially blind one-way multicounter machines. *Theoretical Comput. Sci.*, 7(3):311–324, 1978. doi:10.1016/0304-3975(78)90020-8.

[8] P. HABERMEHL, R. MEYER et H. WIMMEL : The downward-closure of Petri net languages. *Actes de ICALP'10*, vol. 6199 de *Lect. Notes Comput. Sci.*, pp. 466–477. Springer, 2010. doi:10.1007/978-3-642-14162-1_39.

[9] M. H. T. HACK : *Decidability questions for Petri nets*. Thèse de doctorat, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 1976.

[10] J. E. HOPCROFT et J.-J. PANSIOT : On the reachability problem for 5-dimensional vector addition systems. *Theoretical Comput. Sci.*, 8:135–159, 1979. doi:10.1016/0304-3975(79)90041-0.

[11] R. M. KARP et R. E. MILLER : Parallel program schemata. *J. Comput. Syst. Sci.*, 3(2):147–195, 1969. doi:10.1016/S0022-0000(69)80011-5.

[12] S. R. KOSARAJU : Decidability of reachability in vector addition systems. *Actes de STOC'82*, pp. 267–281. ACM, 1982. doi:10.1145/800070.802201.

[13] J.-L. LAMBERT : A structure to decide reachability in Petri nets. *Theoretical Comput. Sci.*, 99(1):79–104, 1992. doi:10.1016/0304-3975(92)90173-D.

[14] S. LASOTA : VASS reachability in three steps. Preprint, 2018. arXiv:1812.11966 [cs.LO].

[15] J. LEROUX : The general vector addition system reachability problem by Presburger inductive invariants. *Logic. Meth. Comput. Sci.*, 6(3:22):1–25, 2010. doi:10.2168/LMCS-6(3:22)2010.

[16] J. LEROUX : Vector addition systems reachability problem (A simpler solution). *Actes de Turing-100*, vol. 10 de *EPiC Ser. Comput.*, pp. 214–228. EasyChair, 2012. doi:10.29007/bnx2.

[17] J. LEROUX et S. SCHMITZ : Demystifying reachability in vector addition systems. *Actes de LICS'15*, pp. 56–67. IEEE Press, 2015. doi:10.1109/LICS.2015.16.

[18] J. LEROUX et S. SCHMITZ : Reachability in vector addition systems is primitive-recursive in fixed dimension. *Actes de LICS'19*. IEEE, 2019. doi:10.1109/LICS.2019.8785796.

[19] J. LEROUX et G. SUTRE : On flatness for 2-dimensional vector addition systems with states. *Actes de CONCUR'04*, vol. 3170 de *Lect. Notes Comput. Sci.*, pp. 402–416. Springer, 2004. doi:10.1007/978-3-540-28644-8_26.

[20] R. LIPTON : The reachability problem requires exponential space. Rap. tech. 62, Yale University, 1976.

[21] M. LÖB et S. WAINER : Hierarchies of number-theoretic functions. I. *Arch. Math. Logic*, 13(1–2):39–51, 1970. doi:10.1007/BF01967649.

[22] E. W. MAYR : An algorithm for the general Petri net reachability problem. *Actes de STOC'81*, pp. 238–246. ACM, 1981. doi:10.1145/800076.802477.

[23] H. MÜLLER : The reachability problem for VAS. *Dans Advances in Petri Nets 1984*, vol. 188 de *Lect. Notes Comput. Sci.*, pp. 376–391. Springer, 1985. doi:10.1007/3-540-15204-0_21.

[24] B. O. NASH : Reachability problems in vector addition systems. *Amer. Math. Month.*, 80(3):292–295, 1973. doi:10.1080/00029890.1973.11993273.

[25] C. H. PAPADIMITRIOU : On the complexity of integer programming. *J. ACM*, 28(4):765–768, 1981. doi:10.1145/322276.322287.

[26] R. J. PARIKH : On context-free languages. *J. ACM*, 13(4):570–581, 1966. doi:10.1145/321356.321364.

[27] C. A. PETRI : *Kommunikation mit Automaten*. Thèse de doctorat, Universität Bonn, 1962. URL `http://edoc.sub.uni-hamburg.de/informatik/volltexte/2011/160/`.

[28] L. POTTIER : Minimal solutions of linear Diophantine systems: Bounds and algorithms. *Actes de RTA'91*, vol. 488 de *Lect. Notes Comput. Sci.*, pp. 162–173. Springer, 1991. doi:10.1007/3-540-53904-2_94.

[29] C. RACKOFF : The covering and boundedness problems for vector addition systems. *Theoretical Comput. Sci.*, 6(2):223–231, 1978. doi:10.1016/0304-3975(78)90036-1.

[30] C. REUTENAUER : *The mathematics of Petri nets*. Masson and Prentice, 1990.

[31] S. SCHMITZ : Complexity bounds for ordinal-based termination. *Actes de RP'14*, vol. 8762 de *Lect. Notes Comput. Sci.*, pp. 1–19. Springer, 2014. doi:10.1007/978-3-319-11439-2_1.

[32] S. SCHMITZ : Automata column: The complexity of reachability in vector addition systems. *ACM SIGLOG News*, 3(1):3–21, 2016. doi:10.1145/2893582.2893585.

[33] S. SCHMITZ : Complexity hierarchies beyond ELEMENTARY. *ACM T. Comput. Theory*, 8(1):1–36, 2016. doi:10.1145/2858784.

[34] S. SCHMITZ : *Algorithmic Complexity of Well-Quasi-Orders*. Mémoire d'habilitation, École Normale Supérieure Paris-Saclay, 2017. URL `http://tel.archives-ouvertes.fr/tel-01663266`.

[35] R. TARJAN : Depth-first search and linear graph algorithms. *SIAM J. Comput.*, 1(2):146–160, 1972. doi:10.1137/0201010.

[36] J. van LEEUWEN : A partial solution to the reachability-problem for vector-addition systems. *Actes de STOC'74*, pp. 303–309, 1974. doi:10.1145/800119.803908.

[37] J. von zur GATHEN et M. SIEVEKING : A bound on solutions of linear integer equalities and inequalities. *Proc. AMS*, 72(1):155–158, 1978. doi:10.2307/2042554.

[38] S. S. WAINER : Ordinal recursion, and a refinement of the extended Grzegorczyk hierarchy. *J. Symb. Log.*, 37(2):281–292, 1972. doi:10.2307/2272973.

[39] G. ZETZSCHE : The complexity of downward closure comparisons. *Actes de ICALP'16*, vol. 55 de *Leibniz Int. Proc. Inf.*, pp. 123:1–123:14. LZI, 2016. doi:10.4230/LIPIcs.ICALP.2016.123.

Chapitre 3

La combinatoire analytique

Marni MISHNA

On apprend les techniques de bases de la combinatoire analytique qui sert à comprendre le comportement des objets combinatoires de grande taille. On suit un chemin qui nous amène vers les séries génératrices multivariées.

3.1 Introduction

À quoi ressemble un arbre binaire aléatoire de taille 10000 ? Combien y a-t-il de cycles maximaux dans une permutation de 562 232 éléments en moyenne ? Quelle la loi de distribution de ce paramètre ? Comment estimer le nombre de marches qui ne quittent pas un cône convexe quand le nombre de pas tend vers l'infini ? En informatique, ces questions se posent quand on veut comprendre le comportement en moyenne d'un algorithme, ou faire un choix entre deux structures de données, ou bien développer des méthodes astucieuses. La combinatoire fournit une énorme boîte à outils pour répondre à ces questions. Les séries formelles – essentiellement les développements de TAYLOR – sont beaucoup utilisées en combinatoire. Elles nous permettent d'utiliser les mathématiques classique de manière systématique, guidés par quelques principes simples. Dans ce cours nous allons développer un calcul combinatoire – nous allons construire des ensembles d'objets avec quelques opérations ensemblistes bien connues (union, produit cartésien, sous-ensemble), et ensuite nous serons capable de faire des estimations très précises du nombre des objets d'une taille donnée, et comprendre le comportement de certains paramètres (c'est-à-dire, leur moyenne, déviation standard, loi de distribution). On ne va pas tout faire, bien sûr – c'est un sujet très développé. Dans ce chapitre nous ouvrons des

portes vers des ressources plus profondes, notamment le volume célèbre de FLAJOLET et SEDGEWICK [6], et ensuite une approche plus géométriques de PEMANTLE et WILSON [13]. Ces notes sont basées sur mon livre [11].

On commence par un résultat bien connu : l'approximation de STIRLING de la fonction factorielle :

$$n! \sim n^n e^{-n} \sqrt{2\pi n} \text{ quand } n \to \infty.$$

Déjà, pour n petit sa précision est forte :

	1	2	3	4	5	6	7	8	9	10
$n!$	1	2	6	24	120	720	5040	40320	362880	3628800
$\lceil n^n e^{-n} \sqrt{2\pi n} \rceil$	1	2	6	24	118	710	4981	39903	359537	3598696

Comment obtenir une telle d'approximation en commençant par une description combinatoire ?

3.2 Un formalisme combinatoire

3.2.1 Une classe combinatoire

Considérons un ensemble d'objets $\mathscr{C}$ équipé d'une application de taille de $\mathscr{C}$ à $\mathbb{N}$. Nous disons que $\mathscr{C}$ est *une classe combinatoire* si pour toute taille n le nombre d'objets de cette taille est fini. La taille d'un objet $\gamma \in \mathscr{C}$ est notée $|\gamma|$. Si plusieurs classes sont considérées, la classe est précisée par un indice, $|\gamma|_{\mathscr{C}}$. Les classes combinatoires sont donc des objets discrets, car la taille est un entier non négatif. Étant donnée une classe $\mathscr{C}$, la sous-classe d'objets de taille n est notée $\mathscr{C}_n$:

$$\mathscr{C}_n := \{\gamma \in \mathscr{C} \mid |\gamma| = n\}.$$

La cardinalité de cet ensemble est notée c_n et la séquence $c_0, c_1, c_2, \cdots = (c_n)_{n=0}^{\infty}$ est *la suite de dénombrement* de la classe combinatoire $\mathscr{C}$. L'objectif principal de la combinatoire analytique est, pour la classe $\mathscr{C}$, de comprendre $\mathscr{C}_n$ pour n grand, en particulier d'obtenir des formules pour c_n. Dans de nombreux cas, nous pouvons procéder systématiquement. Le mantra optimiste de FLAJOLET et SEDGEWICK [6] est

Si vous pouvez le spécifier, vous pouvez l'analyser.

Nous illustrons maintenant ces définitions et cette philosophie avec quelques exemples.

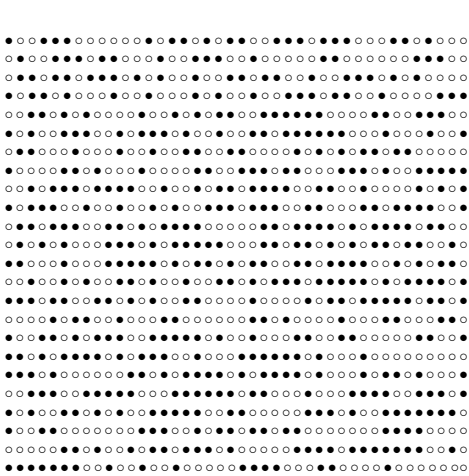


FIGURE 3.1 – *Un mot binaire arbitraire Un marche aléatoire dans le plan sur 10000 pas venant de l'ensemble* $\{(1,0),(0,1),(-1,0),(0,-1)\}$.

3.2.2 Exemple : les mots binaires

Soient $\Sigma = \{\circ, \bullet\}$, et $\mathscr{B}$ l'ensemble de tous les mots sur Σ. Puisque la cardinalité de Σ est égale à deux, on dit que $\mathscr{B}$ est un ensemble de *mots binaires*. Figure 3.1 donne un exemple de grande taille. Notons le mot de longueur 0 (le *mot vide*) par ϵ. On a alors :

$$\mathscr{B} := \{\epsilon, \circ, \bullet, \circ\circ, \circ\bullet, \circ\bullet\circ, \circ\bullet\bullet, \circ\circ\circ, \circ\circ\bullet, \bullet\bullet\circ, \bullet\bullet\bullet, \ldots\}.$$

Dans notre notation, $\mathscr{B}_2 = \{\circ\circ, \circ\bullet, \bullet\circ, \bullet\bullet\}$ et $b_2 = 4$. En général, $b_n = 2^n$. Attention ! La fonction $\mathscr{B} \to \mathbb{N}$ qui compte le nombre de $\circ$ dans un mot, n'est pas une fonction de taille qui donne un ensemble combinatoire– il y a un nombre infini de mot sans $\circ$, par exemple. C'est plutôt un exemple d'un paramètre. Nous allons aussi étudier les fonctions de paramètres combinatoires.

Réfléchir Comment estimer le nombre de mots binaire de taille $2n$ (pour n grand) qui évite le mot $\bullet\bullet\bullet$ comme facteur, avec la condition que le nombre de $\bullet$ est égal au nombre de $\circ$?

3.2.3 Les opérateurs ensemblistes

On manipule les classes combinatoires avec des opérateurs ensemblistes.

Addition L'*addition combinatoire* est une opération portant sur deux classes combinatoire. La somme de $\mathscr{A}$ et $\mathscr{B}$ est l'ensemble formé par leur union disjointe, équipé d'une fonction de taille héritée. Notons cette action par le symbole $+ : \mathscr{A} + \mathscr{B}$. Plus explicitement la fonction de taille est :

$$|\gamma|_{\mathscr{C}} = \begin{cases} |\gamma|_{\mathscr{A}} & \text{si } \gamma \in \mathscr{A} \\ |\gamma|_{\mathscr{B}} & \text{si } \gamma \in \mathscr{B}. \end{cases} \tag{3.1}$$

L'impact sur la suite de dénombrement est :

$$\mathscr{C} = \mathscr{A} + \mathscr{B} \implies c_n = a_n + b_n.$$

Produit Le *produit cartésien* de deux classes combinatoire est :

$$\mathscr{A} \times \mathscr{B} := \{(\alpha, \beta) : \alpha \in \mathscr{A}, \beta \in \mathscr{B}\}. \tag{3.2}$$

On pourrait, dans certaines situations, l'interpréter comme la concaténation des éléments : un objet de la classe $\mathscr{A}$ suivi par un objet de la classe $\mathscr{B}$. Souvent on utilise un raccourci : $\mathscr{C} = \mathscr{A}\mathscr{B}$. La taille d'un élément $(\alpha, \beta) \in \mathscr{A}\mathscr{B}$ et la somme des tailles des composants :

$$|(\alpha, \beta)| := |\alpha| + |\beta|.$$

Le nombre d'éléments de taille n est donc

$$\mathscr{C} = \mathscr{A} \times \mathscr{B} \implies c_n = \sum_{k=0}^{n} a_k b_{n-k}.$$

Le nombre c_n ne dépend pas du contenu des ensembles $\mathscr{A}$ et $\mathscr{B}$ – il dépend uniquement de leur suite de dénombrement. Cet opérateur est donc dit *admissible* dans notre contexte.

Exponentiation On définit l'exponentiation combinatoire d'une classe $\mathscr{A}$ comme suit :

$$\mathscr{A}^{\ell} := \{(\alpha_1, \ldots, \alpha_\ell) : \alpha_j \in \mathscr{A}\} \equiv \underbrace{\mathscr{A} \times \cdots \times \mathscr{A}}_{\ell \text{ fois}}.$$

$$\mathscr{C} = \mathscr{A}^{\ell} \implies c_n = \sum_{k_1 + \cdots + k_\ell = n} a_{k_1} \ldots a_{k_\ell}.$$

Cet opérateur est admissible. On définit $A^0 := \{\epsilon\}$ ou ϵ est un objet de taille 0.

Séquence On utilise une étoile pour noter un exposant arbitraire. Plus formellement, l'opérateur de *séquence* est défini par :

$$\mathscr{A}^* = \textsc{Seq}(\mathscr{A}) := \bigcup_{k\geqslant 0} \mathscr{A}^k.$$

Si $\mathscr{A}_0$ n'est pas vide, alors $\mathscr{A}^*$ n'est pas une classe combinatoire – il y a un nombre infini d'objets de taille zéro. Ainsi, pour appliquer cet opérateur, on doit vérifier la condition $\mathscr{A}_0 = \emptyset$. Un élément de $\mathscr{A}^\ell$ est une séquence de longueur ℓ d'objets de $\mathscr{A}$.

3.2.4 Les séries formelles

Maintenant nous codons une séquence par une série formelle. On pourrait identifier une série à une fonction, par exemple une séries de TAYLOR qui est utile pour estimations. Ici, on les utilise d'une manière plus formelle, mais on garde l'identification d'une fonction et sa développement en séries à zéro. Donc, on écrit $\frac{1}{1-x} = \sum_{k\geqslant 0} x^k$, même si ce n'est pas valable quand $|x| < 1$. L'ensemble de séries en x ayant pour coefficients des nombres rationnels est un anneau noté $\mathbb{Q}[[x]]$. La sommation et la multiplication sont définies par

$$\sum_n f_n x^n + \sum_n g_n x^n = \sum_n (f_n + g_n)\, x^n \quad \sum_n f_n x^n \sum_n g_n x^n = \sum_n \left(\sum_{k=0}^{n} f_k g_{n-k} \right) x^n.$$

L'élément $1 - x$ a un inverse multiplicatif dans cet anneau :

$$(1-x)\left(\sum_{n=0}^{\infty} x^n\right) = 1. \tag{3.3}$$

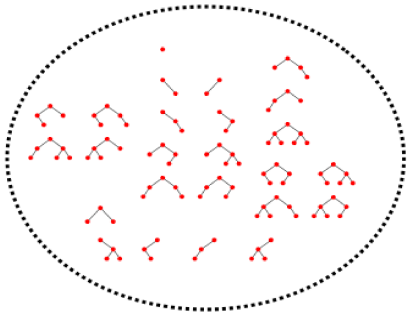
L'inverse est donc $\sum_{n=0}^{\infty} x^n$, et appelé la *série géométrique*. Le développement de TAYLOR de la fonction $F(x)$ autour du point x_0, quand il existe, est

$$\sum_{k\geqslant 0} \frac{1}{k!} \frac{d^k F}{dx^k}(x_0)(x - x_0)^k.$$

La série géométrique est le développement de TAYLOR de la fonction $F(x) = \frac{1}{1-x}$.

Il y a une correspondance entre les manipulations formelles des séries, et les opérations sur les fonctions. Par exemple, si $F(x) = \sum_{n=1} f_n x^n$ est une série convergente en x alors

$$\frac{d}{dx} F(x) = F'(x) := \sum_{n=0} f_{n+1} x^n \quad \text{et} \quad \int F(x)\, dx := \sum_{n=1} f_n \frac{x^{n+1}}{n+1}.$$



$$x + 2x^2 + 5x^3 + 14x^4$$

FIGURE 3.2 – *Une classe finie d'arbres binaires, et sa série génératrice.*

La composition de deux séries, notée $F(G(x)) := \sum_n f_n(G(x)^n)$ est bien définie si $F(x)$ est un polynôme ou $G(0) = 0$.

Une sommation $\sum_{k=-d}^{n} a_k x^k$ est un *polynôme de* LAURENT *en* x. Plus tard, on va aussi utiliser les séries de LAURENT, les sommations avec un nombre fini de puissance négatives : $\sum_{k=-d} a_k x^k$. En fait, y a nombreux types de séries qu'on utilise en combinatoire.

3.2.5 Séries génératrices

Étant donnée une classe combinatoire $\mathscr{C}$, sa *série génératrice (ordinaire)* $C(x)$ est la série formelle :

$$C(x) := \sum_{\gamma \in \mathscr{C}} x^{|\gamma|} \quad \Longrightarrow \quad C(x) = \sum_{n \geqslant 0} c_n x^n. \tag{3.4}$$

On appelle aussi $C(x)$ une *fonction génératrice*.

Mots Binaires Le nombre de mots binaires de taille n étant 2^n, sa séries génératrice est :

$$B(x) = \sum_{n=0}^{\infty} b_n x^n = \sum_{n=0}^{\infty} 2^n x^n = \frac{1}{1-2x}. \tag{3.5}$$

Pour l'instant il n'est pas clair laquelle des représentations de la suite de dénombrement est la plus utile.

3.2.6 Comment trouver un coefficient ? Première tentative

On utilise une notation un peu atypique pour indiquer un coefficient dans une série :

$$[x^n]\left(\sum_{k=0}^{\infty} f_k x^k\right) := f_n.$$

Par exemple, $[x^3]2x + 4x^2 + x^3 = 1$ et $[x^4]2x + 4x^2 + x^3 = 0$. La notation $[x^n]F(x)$ indique que l'on veut le coefficient de x^n dans le développement de TAYLOR de $F(x)$ à l'origine. (Ou bien, d'autres types de séries quand il n'y a pas un développement de TAYLOR). Alors,

$$[x^n]\frac{1}{1-2x} = 2^n \quad \text{et} \quad [x^k](1-x)^n = \binom{n}{k},$$

pour n entier.

Pour calculer $[x^n]F(x)$, on utilise quelques identités de base :

$$[x^n]\left(F(x) + G(x)\right) = \left([x^n]F(x)\right) + \left([x^n]G(x)\right)$$

$$[x^n]F(\rho x) = \rho^n [x^n]F(x).$$

On étend les coefficients binomiaux aux réels α par la formule $\binom{\alpha}{k} := \frac{\alpha(\alpha-1)(\alpha-2)\ldots(\alpha-k+1)}{k!}$. Le théorème de NEWTON sur les binomiaux généralisé stipule alors que pour $k \in \mathbb{N}$ et $\alpha \in \mathbb{R}$,

$$[x^k](1+x)^\alpha = \binom{\alpha}{k}. \tag{3.6}$$

Soit $\alpha \in \mathbb{R}^+$ et k entier positif. Alors,

$$\binom{-\alpha}{k} = \frac{(-\alpha)(-\alpha-1)\ldots(-\alpha-k+1)}{k!} = (-1)^k\binom{\alpha+k-1}{k}. \tag{3.7}$$

On en déduit que :

$$[x^n]\frac{1}{(1-mx)^\alpha} = m^n\binom{\alpha+n-1}{n} \tag{3.8}$$

pour des nombres réels m et α. Cela nous permet de calculer $[x^n]R(x)$ pour n'importe quelle fraction rationnelle $R(x)$. D'abord, on récrit $R(x)$ comme $\sum_{i,k}\frac{\alpha_{i,k}(x)}{(1-m_i x)^k}$ de sorte que chaque $\alpha_{i,k}(x)$ est un polynôme de degré au plus k. Ensuite, alors on peux appliquer l'Équation (3.8) à chaque terme.

Exemple Estimer $[x^n]R(x)$ avec

$$R(x) = \frac{486\,x^6 - 810\,x^5 + 540\,x^4 - 180\,x^3 + 14\,x^2 + 6\,x - 1}{(-1+3\,x)^5\,(-1+4\,x)^2}.$$

D'abord, on écrit [1]

$$R(x) = \frac{1}{(1-3x)^5} + \frac{2x}{(1-4x)^2}$$

Ensuite on applique Eq. (3.8) à chaque terme :

$$\begin{aligned}
[x^n]R(x) &= [x^n]\frac{1}{(1-3x)^5} + [x^n]\frac{2x}{(1-4x)^2} \\
&= 3^n\binom{4+n}{n} + 2\,4^{n-1}\binom{n}{n-1} \\
&= 3^n(n+4)(n+3)(n+2)(n+1) + 2n4^{n-1}.
\end{aligned}$$

Finalement, pour n grand, $[x^n]R(x) \sim \frac{4^n n}{2}$.

3.2.7 Blocs de constructions combinatoires

Nous allons maintenant voir comment construire des classes combinatoires. Les classes de base sont de deux types. Une contient un élément de taille 0 et l'autre un élément de taille 1.

Un *classe de type epsilon* contient un seul objet de taille 0 : $\mathscr{E} = \{\epsilon\}$, avec $|\epsilon| = 0$. Souvent on écrit ϵ pour représenter l'ensemble $\{\epsilon\}$. La série génératrice est $E(x) = 1$.

Une *classe atomique* contient un élément de taille un. Formellement on écrit $\mathscr{Z} = \{\circ\}$ où $|\circ| = 1$. Comme pour les classes de type epsilon, souvent dans les équations combinatoires on écrit l'élément au lieu de l'ensemble. La série génératrice est $Z(x) = x$. Les objets atomiques donnent la taille d'un objet combinatoire : les lettres dans un mot, les sommets dans un graphe, les pas dans une marche.

3.2.8 Opérateurs admissibles et séries génératrices

On a maintenant tout ce qu'il faut pour construire un dictionnaire entres opérateurs admissible et séries génératrices.

Théorème 3.2.1. *Soient $\mathscr{A}, \mathscr{B}$, et $\mathscr{C}$ des classes combinatoires. Les formules suivantes définissent la série génératrice de $\mathscr{C}$:*

1. En Maple : `convert(R(x), parfrac)`

$\mathscr{C}$	$C(x)$
Classe Atomique	x
Classe Epsilon	1
$\mathscr{A} + \mathscr{B}$	$A(x) + B(x)$
$\mathscr{A} \times \mathscr{B}$	$A(x)B(x)$
$\mathscr{A}^k$	$A(x)^k \quad (k \in \mathbb{N})$
$\mathscr{A}^*$	$\frac{1}{1-A(x)} \quad (si\ a_0 = 0)$

Exercice Démontrer ce théorème.

Exemple : Mots binaires $\mathscr{B} = \{\circ, \bullet\}^*$. On traduit la définition combinatoire en équation fonctionnelle pour le séries génératrice :

$$\{\circ, \bullet\} \mapsto 2x \qquad \{\circ, \bullet\}^* \mapsto \frac{1}{1-2x}.$$

Par le théorème 3.2.1, $B(x) = \frac{1}{1-2x}$. On pourrait faire ça autrement. Soit un mot binaire est vide, soit c'est un atome suivi par un mot binaire plus petit. Ainsi,

$$\mathscr{B} \equiv \{\epsilon\} + \{\circ, \bullet\} \times \mathscr{B}.$$

On déduit :

$$\begin{array}{ccccccc} \mathscr{B} & \equiv & \{\epsilon\} & + & \{\circ, \bullet\} & \times & \mathscr{B} \\ \downarrow & & \downarrow & & \downarrow & & \downarrow \\ B(x) & = & 1 & + & 2x & \cdot & B(x) \end{array}.$$

$$\implies B(x) = 1 + 2x\,B(x) \implies B(x) = \frac{1}{1-2x}.$$

Ces traductions sont vraies même quand la spécification d'une classe combinatoire est récursive. Une troisième formule combinatoire : on peut couper un mot d'une manière unique selon les occurrences des $\bullet$. Ainsi

$$\mathscr{B} \equiv \circ^* \times (\bullet \times (\circ)^*)^* \implies B(x) = \frac{1}{1-x}\frac{1}{1-\frac{x}{1-x}} = \frac{1}{1-2x}.$$

Cette décomposition sera utile quand on va regarder les mots qui évitent un motif.

Exercice Montrer que le langage des mots binaires sur l'alphabet $\{0,1\}$ qui représentent l'expansion binaire [2] des multiples entiers de 3 est

$$\mathscr{L} = (0 + (1(01^*0)^*1))^*.$$

Remarque :

$$\begin{aligned}\mathscr{L} &= \{ \text{ expansion binaires de } n \mid n \equiv 0 \mod 3\} \\ &= \{\epsilon, \overset{0}{0}, 00, 000, \ldots, \overset{3}{11}, 011, 0011, \ldots, \overset{6}{110}, 0110, 00110, \ldots, \\ &\quad \overset{9}{1001}, 01001, \overset{12}{1100}, 01100, \ldots, \overset{15}{1111}, 01111, \ldots\}.\end{aligned}$$

3.2.9 Les paramètres combinatoires

Un *paramètre* associé à une classe combinatoire $\mathscr{C}$ est une propriété quantitative des éléments de $\mathscr{C}$ qui se décrit par un application $\chi : \mathscr{C} \to \mathbb{N}$. On le note par une paire, $(\mathscr{C}, \chi)$. Sa série génératrice bivariée est

$$C(u,x) := \sum_{\gamma \in \mathscr{C}} u^{\chi(\gamma)} x^{|\gamma|}. \tag{3.9}$$

On peut réorganiser cette expression comme suit :

$$C(u,x) = \sum_{n \geqslant 0} \left(\sum_{k \geqslant 0} c_{k,n} u^k \right) x^n, \tag{3.10}$$

ou $c_{k,n}$ est le nombre d'objets de taille n pour lequel la valeur de χ est k. On dit que u marque le paramètre χ. Cette série est un élément de l'anneau $\mathbb{Q}[u][[x]]$, l'anneau des séries en x avec des coefficients qui sont des polynômes en u à coefficients dans $\mathbb{Q}$.

Exemple : Mots binaires évitant un motif Soit $\mathscr{L}$ le sous-ensemble de mots binaires sur l'alphabet $\{\circ, \bullet\}$ qui ne contient aucun $\circ\circ\circ$ comme facteur. Cet ensemble est spécifié par

$$\mathscr{L} \equiv (\epsilon + \circ + \circ\circ) \times (\bullet \times (\epsilon + \circ + \circ\circ))^*.$$

On considère le paramètre χ, donné par le nombre de $\circ$ dans un mot de $\mathscr{L}$. À titre d'exemple, $\chi(\circ \bullet \bullet \circ \bullet) = 2$. Les premiers termes de la séries génératrices bivariée de $(\mathscr{L}, \chi)$, sont :

$$L(u,x) = 1 + (u+1)\,x + \left(u^2 + 2u + 1\right) x^2 + \left(3u^2 + 3u + 1\right) x^3 + \ldots$$

2. $010101 \equiv 0 * 2^5 + 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 21$, par exemple qui est multiple de 3 donc $010101 \in \mathscr{L}$.

Pourrions-nous trouver une forme close pour $L(u,x)$? Sous certaines conditions, il y a un analogue multivarié du théorème 3.2.1, qui nous permet de trouver une expression pour une telle série génératrice de manière systématique. Pour le cas présent, on obtient

$$L(u,x) = (1+ux+u^2x^2)\left(\frac{1}{1-x(1+ux+u^2x^2)}\right). \tag{3.11}$$

Problème Montrer que $\frac{[x^n]\frac{\partial L}{\partial u}(1,x)}{[x^n]L(1,x)}$ est la valeur moyenne du paramètre χ parmi les éléments de $\mathcal{B}_n$. Obtenir une expression pour la variance utilisant les dérivés de la série génératrice.

Problème Trouver une série génératrice trivariée $B(u,v,x)$ qui marque le paramètre $\chi(w) = (|w|_\bullet, |w|_\circ) = (\#\circ \text{ dans } w, \#\bullet \text{ dans } w)$ et la taille pour $\mathscr{B}$.

3.2.10 La diagonale d'une série

Fixons une dimension d. On utilise des symboles gras pour désigner des vecteurs, par exemple

$$\mathbf{x} := x_1, \ldots, x_d.$$

On étend cette notation aux puissances de vecteurs réels par des vecteurs entiers :

$$\mathbf{x}^{\mathbf{n}} := x_1^{n_1} \ldots x_d^{n_d}$$

On note ainsi plus efficacement les séries formelles à d variables

$$F(\mathbf{x}) := \sum_{(n_1,\ldots,n_d)\in\mathbb{N}^d} f(n_1,\ldots,n_d)\, x_1^{n_1}\ldots x_d^{n_d} = \sum_{\mathbf{n}\in\mathbb{N}^d} f(\mathbf{n})\,\mathbf{x}^{\mathbf{n}}.$$

Les séries agissent comme un codage d'un tableau multidimensionnel. Plus formellement, nous pouvons identifier une série comme un élément de $K[[x_1]]...[[x_d]]$. Notez que ceci est un anneau plus grand que $K[x_1,...,x_{d-1}][[x_d]]$, l'anneau des séries de puissances de x_d à coefficients polynomiaux. On dit que $F(\mathbf{x})$ est *combinatoire* si $f(\mathbf{n}) > 0$ pour tout $\mathbf{n} \in \mathbb{N}^d$.

Le terme constant de $F(\mathbf{x})$ est défini à l'aide d'une application

$$\mathrm{CT} : K[[x_1]]\ldots[[x_d]] \to K \tag{3.12}$$

$$\sum_{\mathbf{n}\in\mathbb{N}^d} f(\mathbf{n})\,\mathbf{x}^{\mathbf{n}} \mapsto f(0,0,\ldots,0). \tag{3.13}$$

Nous désignons le terme constant par rapport à un sous-ensemble de variables utilisant des indices. Pour une fonction, s'il est clair de quelle série on parle, on peut appliquer CT à la fonction. Nous en dirons plus sur ce point important mais subtil dans un instant. La *diagonale centrale* est une application

$$\operatorname{Diag} : K[[x_1]][[x_2]] \dots [[x_d]] \to K[[x_d]] \tag{3.14}$$

$$\sum_{\mathbf{n}\in\mathbb{N}^d} f(\mathbf{n})\,\mathbf{x}^{\mathbf{n}} \mapsto \sum_{n\geqslant 0} f(n,n,\dots,n)\,x_d^n \tag{3.15}$$

Par défaut, nous utilisons la convention d'exprimer la série uniforme résultante comme une série dans la dernière variable.

$$\operatorname{Diag}(x^2yz + 3xyz + 7xyz^2 + 5x^2y^2z^2) = 3z + 5z^2.$$

Enfin, nous définissons la diagonale le long du rayon $\mathbf{r} = (r_1, r_2, \dots, r_d) \in \mathbb{N}^d$:

$$\operatorname{Diag}^{\mathbf{r}} F(\mathbf{x}) = \operatorname{Diag}^{\mathbf{r}} \sum_{\mathbf{n}\in\mathbb{N}^d} f(\mathbf{n})\,\mathbf{x}^{\mathbf{n}} := \sum_{n\geqslant 0} f(nr_1, nr_2, \dots, nr_d)\,x_d^n.$$

Si une fonction a une expansion de TAYLOR autour de l'origine, alors on définit le terme constant et la diagonale sont bien définis vis à vis de cette série. Nous pouvons également définir des diagonales et des termes constants pour des séries dans des plus grands anneaux, mais considérons d'abord un exemple.

Exemple 3.2.2 (Multinomiaux)**.** *La fonction rationnelle* $\frac{1}{1-x-y}$ *a une expansion de* TAYLOR *autour de l'origine facile à trouver, à partir de laquelle on déduit une expression exacte des coefficients de sa diagonale centrale :*

$$\operatorname{Diag} \frac{1}{1-x-y} = \operatorname{Diag} \sum_{n\geqslant 0} (x+y)^n \tag{3.16}$$

$$= \operatorname{Diag} \sum_{\ell\geqslant 0} \sum_{k\geqslant 0} \binom{\ell+k}{k} x^k y^\ell \tag{3.17}$$

$$= \sum_{n\geqslant 0} \binom{2n}{n} y^n. \tag{3.18}$$

En fait, on peut même déterminer une expression exacte pour la diagonale le long de n'importe quel rayon :

$$\operatorname{Diag}^{(r,s)} \frac{1}{1-x-y} = \sum_{n\geqslant 0} \binom{rn+sn}{rn} y^n.$$

Cet exemple se généralise naturellement à une dimension arbitraire, en utilisant les multinomiaux[3] *:*

$$\mathrm{Diag}^{\mathbf{r}} \frac{1}{1-(x_1+\cdots+x_d)} = \sum_{n\geqslant 0} \binom{n(r_1+\cdots+r_d)}{nr_1,\ldots,nr_d} x_d^n.$$

En général, on peut exprimer les diagonales de fonctions rationnelles à l'aide de coefficients binomiaux d'une façon systématique.

3.2.11 Une classe dérivée

Pour une classe et paramètre $(\mathscr{C},\chi)$, une sous-classe de la forme

$$\{\gamma\in\mathscr{C} \mid \gamma\in\mathscr{C}_n \implies \chi(\gamma)=K(n)\}$$

pour une valeur de paramètre $K(n)$ qui pourrait dépend sur n est dite *une classe dérivée*. Dans ce cas, on écrit la série génératrice en utilisant une diagonale, ou CT.

Exemple : les mots équilibrés Soit $\mathscr{L}$ est la classe combinatoire de mots binaires supérieurs à $\{\bullet,\circ\}$, de sorte qu'aucun mot ne possède une séquence de $\circ$ égale ou supérieure à 3 contigus. Par exemple, $\bullet\circ\bullet\circ\circ\circ\bullet\circ$ n'est pas un mot en $\mathscr{L}$. L'expression régulière suivante détermine $\mathscr{L}$, et on la traduit en équation fonctionnelle pour la série génératrice :

$$\begin{array}{ccccccccc}
\mathscr{L} & \equiv & (\epsilon+\circ+\circ\circ) & \times & (& \bullet & (\epsilon+\circ+\circ\circ))^* \\
\downarrow & & \downarrow & & & \downarrow & \downarrow & \\
L(x,y) & = & (1+x+x^2) & \cdot & \big(1-(& y & (1+x+x^2)\big)^{-1}
\end{array}$$

Soit $a(j,k)=$ # mots avec j $\circ$ et k $\bullet$. Alors, la série génératrice bi-variée où x marque $\circ$ et y marque $\bullet$ est

$$L(x,y)=\sum_{j,k} a(j,k)x^jy^k = \frac{1+x+x^2}{1-y(1+x+x^2)}.$$

Soit $\mathscr{L}_=$ la sous-langage équilibré de $\mathscr{L}$ ou chaque mot a le même nombre de $\circ$ et $\bullet$:

$$\mathscr{L}_= = \{\epsilon,\circ\bullet,\bullet\circ,\bullet\bullet\circ\circ,\bullet\circ\bullet\circ,\circ\bullet\circ\bullet,\bullet\circ\circ\bullet,\circ\circ\bullet\bullet,\ldots\}$$

3. $\binom{n_1+n_2+\cdots+n_d}{n_1,n_2,\ldots,n_d} := \frac{(n_1+n_2+\cdots+n_d)!}{n_1!n_2!\ldots n_d!}$

La série génératrice (la taille est la demi-longueur du mot) est exprimable avec une diagonale :

$$L_{=}(y) := \sum a(n,n)y^n = \operatorname{Diag} L(x,y)$$

Il y a beaucoup de classes combinatoires intéressantes qu'on pourrait définir d'un manière pareil : On défini une classe et un paramètre tel que le couple est facile à comprendre. Avec les diagonales, on trouve la série génératrice d'une classe dérivée. C'est un façon systématique d'étudier les marches dans un cône, par exemple.

3.3 Mini-mini-cours d'analyse complexe

Le mot analytique en combinatoire analytique vient des techniques d'analyse complexe qui sont très utile on combinatoire. Pour le lecteur qui connait peu dans ce sujet, il existe de nombreux références des concepts de bases. On indique des idées clés, mais nous ne pouvons pas tout rappeler ici.

3.3.1 Les singularités

Une fonction est dite *analytique au point* z_0 s'il existe un développement de TAYLOR autour de z_0 avec un rayon de convergence non nul. Ce n'est pas évident, mais c'est équivalent au fait d'être dérivable dans un *voisinage* de z_0 (pas seulement au point z_0). Quand une fonction n'est pas analytique en un point, on dit que le point est *une singularité*. Si une fonction est analytique pour chaque point $z \in \mathbb{C}$, on dit que c'est un fonction entière. La fonction exponentielle est entière car la sommation $\sum_{n\geqslant 0} \frac{z^n}{n!}$ est convergente pour chaque valeur $z \in \mathbb{C}$.

Les singularités des fonctions rationnelles sont appelées *pôles* et correspondent aux racine du dénominateur. Quand on approche un pôle, la valeur absolue de la fonction tend vers l'infini. Pour autant, si z est un pôle de F, alors il existe un entier positif tel que la limite $\lim_{z\to\rho}(z-\rho)^m F(z)$ est finie.

Un deuxième type de singularité qu'on voit en combinatoire vient par exemple des racines carrées. La valeur de la fonction en une telle singularité z_0 est finie mais pour autant la série n'est pas bien définie dans un voisinage de z_0 ; on dit qu'il y a une coupure au voisinage de z_0. À chaque côté c'est analytique, mais pas sur l'axe. Cela s'observe dans la figure 3.4, où l'on voit la partie imaginaire de $F(z)$ et on voit bien la coupure.

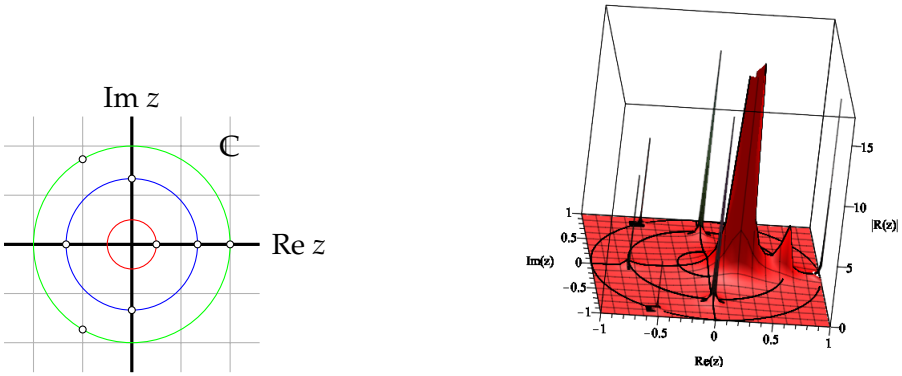


FIGURE 3.3 – *(à gauche) Les pôles de $R(z)$ indiqué dans le plan complexe. Il y a une singularité dominante à $1/4$. (à droite) Le valeur absolu de $R(z)$ pour ces points.*

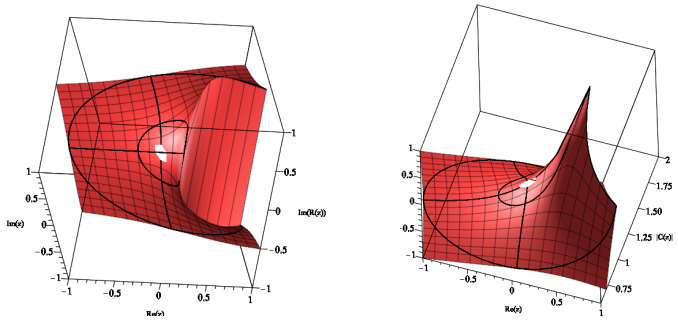


FIGURE 3.4 – *Deux façons de visualiser la fonction complexe $C(z) = \frac{\sqrt{1-4z}}{2z}$. (droit)* $\operatorname{Im} C(z)$ *(gauche)* $|C(z)|$.

Les *singularités dominantes de* $F(z)$ sont les singularités de module minimal. Cette valeur coïncide avec le rayon de convergence et donne le ratio de croissance exponentielle de la suite de coefficients (f_n) :

$$\text{RayondeConvergence}\left(\sum f_n z^n\right) = R \quad \Longrightarrow \quad \limsup_{n\to\infty} f_n^{1/n} = R^{-1}.$$

Si les coefficients d'une série sont tous positifs, on dit que la série est *combinatoire*. On a alors le résultat important suivant (lemme de PRINGSHEIM) : toute série combinatoire a une singularité dominante réelle et positive. En pratique, ce résultat nous aide à trouver les singularités dominantes et donc le ratio de croissance exponentielle.

3.3.2 Comment trouver un coefficient ? Deuxième tentative

L'analyse complexe classique des intégrales de contour nous offre une deuxième approche pour calculer et étudier les coefficients des séries. Étant donnée une fonction, on exprime le coefficient $[z^n]F(z)$ avec une intégrale. Nous avons des méthodes rapides pour évaluer et estimer ces intégrales. Un contour γ dans $\mathscr{C}$ est une courbe lisse. Un contour est *simple* il ne s'intersecte pas. Un contour peut être paramétré par $\gamma \equiv \gamma(t)$ pour $t \in [0,1]$. Si $\gamma(0) = \gamma(1)$, on dit que le contour est fermé. Une intégrale de contour est définie pour une fonction complexe, et peut s'exprimer via un paramétrage γ comme une intégrale sur un domaine réel via la formule :

$$\int_\gamma F(z)\,dz := \int_0^1 F(\gamma(t))\gamma'(t)\,dt.$$

On utilise ces intégrales pour exprimer les coefficients à l'aide du théorème suivant qui date du dix neuvième siècle.

Théorème 3.3.1 (Formule Intégrale de CAUCHY)**.** *Soit $F(z)$ une fonction analytique sur un ouvert simplement connexe Ω qui contient 0, et soit γ une courbe fermée simple à orientation positive dans Ω qui contourne 0 en l'évitant. Alors*

$$[z^n]F(z) = \frac{1}{2\pi i}\int_\gamma F(z)\frac{dz}{z^{n+1}}.$$

On peut alors estimer ces intégrales avec des méthodes classiques, ou mieux avec des méthodes d'analyse complexe. Un résultat analogue et essentiel décrit le comportement près des singularités – c'est le *calcul des résidus*. Ce sujet est trop vaste pour être expliqué dans ces notes mais nous donnons tout de même un petit aperçu pour illustrer le lien entre extraction

de coefficient et calcul des résidus au voisinage d'une singularité. Sous certaines conditions, on a

$$\int_\gamma F(z)dz = \sum_{s\in\text{sing}} \text{Res}_{z=s}\, F(z).$$

ou la somme est sur les singularités de $F(z)$ à l'intérieur de γ, et le résidu est liées aux expansions de LAURENT autour d'un pôle.

3.3.3 Singularités et l'asymptotique

FLAJOLET et SEDGEWICK ont énoncé les deux principes qui nous guident.

Premier principe d'asymptotique des coefficients La localisation des singularités d'une fonction analytique détermine le ratio de croissance exponentiel de ses coefficients de TAYLOR.

Deuxième principe d'asymptotiques des coefficients La nature des singularités détermine la manière dont le terme exponentiel dominant dans les coefficients est modulé par un facteur sous-exponentiel.

On voit ça explicitement avec les fonctions rationnelles.

Théorème 3.3.2 (Asymptotique pour les fonctions rationelles)**.** *Soit $F(z)$ une fonction rationnelle analytique en zéro avec des pôles aux points $\alpha_1, \ldots, \alpha_m$. Alors, il existe des polynômes $p_j(n), j = 1, \ldots, m$ tels que, pour n suffisamment grand,*

$$[z^n]f(z) = \sum_{j=1}^{m} p_j(n)\alpha_j^{-n}$$

où le degré de p_j est l'ordre du pôle en α_j moins un.

Un des grand théorèmes de la combinatoire analytique lie le comportement au voisinage d'un pôle au comportement asymptotique des coefficients. Voir [5] pour les détails.

Théorème 3.3.3 (Transfert)**.** *Supposons que $F(z)$ est analytique dans la région*

$$\Omega = \{z \mid |z| \leqslant 1 + \nu,\ |\arg(z-1)| \geqslant \phi\}$$

pour un $\nu > 0$ et $0 < \phi < \pi/2$, avec l'exception de $z = 1$. Supposons en plus que quand z approche 1 dans Ω,

$$F(z) = O(|1-z|^\alpha)$$

pour un α réel. Alors

$$[z^n]F(z) = O(n^{-\alpha-1}).$$

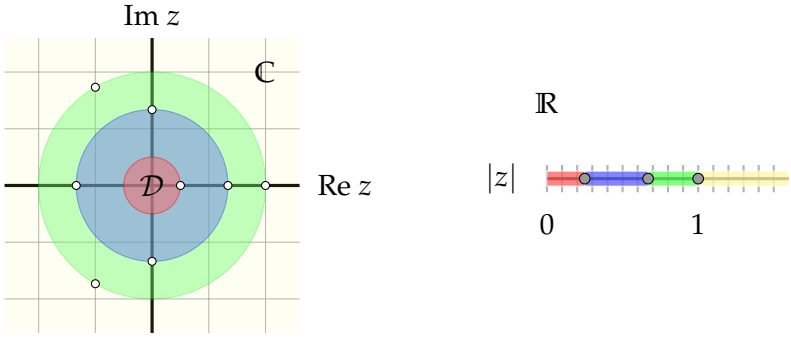


FIGURE 3.5 – *(À gauche) Le domaine de convergence de la série* TAYLOR *de* $\frac{1}{(1-z^3)(1-4z)^2(1-5z^4)}$ *(en rouge). (À droite) L'image du plan complexe sur* $z \mapsto |z|$. *Les points sur le même cercle (tore) ont la même image à droite.*

3.3.4 Analyse complexe multidimensionelle

Le cas multidimensionel est bien développé dans le livre de PEMANTLE et WILSON [13]. Nous commençons par un exemple simple pour montrer comment étendre ces idées pour étudier les sous-classes définies par le valeur d'un paramètre dérivé. Fixons maintenant la dimension d et considérons une fonction $F(z) : \mathbb{C}^d \to \mathbb{C}^d$ équipée d'un développement en série $\sum_{\mathbf{n}} f(\mathbf{n}) z^{\mathbf{n}}$ autour de l'origine. Dans la figure 3.5, on envoie $\mathbb{C}$ vers $\mathbb{R}$ avec $z \mapsto |z|$. Les singularités sur le même cercle sont donc envoyées au même point. L'image de droite est schématise les différents domaines de convergence : remarquez qu'il s'agit d'un ensemble de régions connectées disjointes. Dans la figure de droite, un point peut représenter plusieurs singularités dans la pré-image, ainsi que des points qui ne sont pas singuliers. Nous allons déterminer une image similaire pour les fonctions de plusieurs variables.

Une fonction peut avoir plusieurs développements en série, chacun valable dans un (unique) domaine donné pouvant être visualisé dans $\mathbb{R}^d$. En combinatoire, nous nous intéressons au domaine qui inclut l'origine.

D'abord, nous devrions être précis sur ce que cela signifie pour une série multivariée. Nous interprétons $\sum_{\mathbf{n}\in\mathbb{N}^d} f(\mathbf{n})\, \mathbf{z}^{\mathbf{n}}$ comme des sommes imbriquées

$$\sum_{n_1 \geqslant 0} \left(\dots \left(\sum_{n_d \geqslant 0} f(\mathbf{n})\, z_1^{n_1} \dots z_d^{n_d} \right) \dots \right).$$

En général, la valeur de la somme pourrait dépendre de l'ordre de sommation. Bien que dans le contexte combinatoire, les séries sont générées par une taille de marquage variable, ou bien contiennent tous des termes

entiers positifs, il est important de prendre conscience de cette subtilité. les termes d'une série absolument convergente, comme celles que l'on rencontre en combinatoire, peuvent être réorganisés de manière arbitraire sans changer la convergence ni la valeur de la somme. Le domaine de convergence d'une série formelle, désigné ici par $\mathcal{D}$, est un ensemble ouvert et connecté formé par l'intérieur de l'ensemble des points où la série converge de manière absolue. Comme dans le cas univarié, le domaine de convergence est multicirculaire. Le vocabulaire pertinent est le suivant.

Le *polydisque* d'un point z est le domaine

$$D(z) := \{z' \in \mathbb{C}^d \mid |z'_i| \leqslant |z_i|, 1 \leqslant i \leqslant d\}.$$

Le *tore associé* à un point est l'ensemble

$$T(z) := \{z' \in \mathbb{C}^d \mid |z'_i| = |z_i|, 1 \leqslant i \leqslant d\}.$$

Un domaine de convergence est *multicirculaire* : si un point $\mathbf{z} = (z_1, \ldots, z_d)$ se situe dans le domaine, alors le domaine contient également l'ensemble des points $T(\mathbf{z})$, par convergence absolue. L'ensemble des *singularités* d'une fonction rationnelle $F(z) = G(z)/H(z)$ est précisément l'ensemble

$$\mathcal{V} := \{\mathbf{z} \in \mathbb{C}^d \mid H(\mathbf{z}) = 0\}.$$

C'est une variété, appelée la variété singulière de la fonction rationnelle F. Nous nous intéressons aux singularités qui sont *les plus proches* de l'origine. On définit un point minimal comme un point de $\mathcal{V}$ à coordonnées non nulles sur la limite $\partial\mathcal{D}$ du domaine de convergence. Dit autrement, un point z de la variété $\mathcal{V}$ est *minimal* s'il n'y a pas de point $z' \in \mathcal{V}$ tel que $\left|z'_j\right| < |z_j|$ pour tout j de 1 à d. Un point minimal est dit strictement minimal s'il s'agit du seul point minimal sur son tore : $\mathcal{V} \cap T(\mathbf{z}) = \{\mathbf{z}\}$. Ces définitions étendent naturellement le cas univarié : si une fonction n'est pas entière, elle a une singularité sur sa frontière de convergence. Les singularités sur la frontière (un cercle) de convergence sont les points minimaux. S'il n'y a qu'une seule de ces singularités, alors c'est un point strictement minimal, et il contrôlera seul l'asymptotique dominante. Rappelons qu'une série est dite combinatoire si les coefficients sont tous positifs. Dans le cas $d = 1$, on a vu que cela impose l'existence d'une singularité dominante sur la droite réelle positive (lemme de PRINGSHEIM). Cela se généralise à des dimensions plus élevées.

Lemme 3.3.4. *Supposons que $F(\mathbf{z}) \in \mathbb{N}[[\mathbf{z}]]$ soit une série combinatoire. Le point $\rho \in \mathbb{C}^d$ est un point minimal de $F(z)$ si et seulement si le point $(|\rho_1|, \ldots, |\rho_d|) \in \mathbb{R}^d$ est un point minimal.*

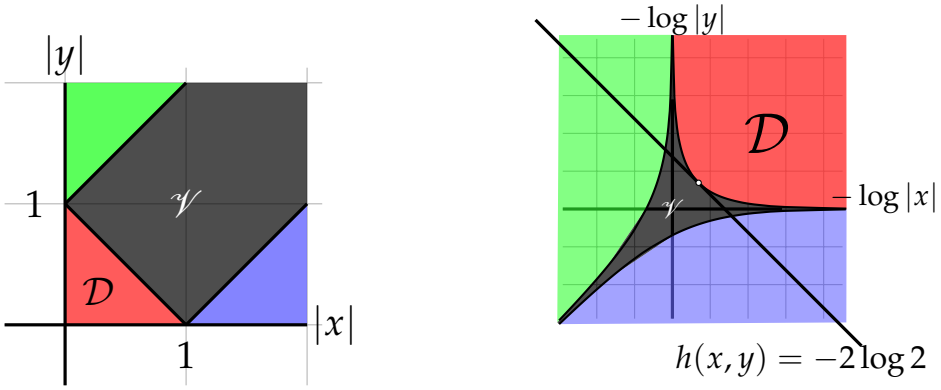


FIGURE 3.6 – *La variété singulière $\mathscr{V}$ (grise foncée) et le domaine de convergence $\mathcal{D}$ (rouge) de $\sum \binom{k+\ell}{k} x^k y^\ell$. Les zones bleu et vert correspondent aux domaines de convergence des autres développements en séries de $\frac{1}{1-x-y}$. (À gauche) L'image de $\mathbb{C}^2$ sous $(x, y) \mapsto (|x|, |y|)$. (À droite) L'image de $\mathbb{C}^2$ sous $(x, y) \mapsto (-\log|x|, -\log|y|)$. Remarque : Les points (x, y) satisfaisant $h(x, y) = -\log x - \log y = -2\log 2$ sont sur un ligne qui est tangente au domaine.*

Les points critiques contribuent chacun à la croissance asymptotique. Ceux-ci sont définis comme des points de la variété singulière $\mathscr{V} = \{\mathbf{z} \mid H(\mathbf{z}) = 0\}$ déterminés en résolvant un système d'équations polynomiales. Un point critique minimal ρ est lisse si

$$\nabla H(\mathbf{z}) \neq 0 \text{ pour tout } z \text{ dans un voisinage de } \rho.$$

Nous ne considérons que les points critiques lisses et strictement minimaux dans ce cours, le cas le plus facile.

3.3.5 Exemple

Illustrons toutes ces définitions avec un exemple. Soit

$$F(x, y) = \frac{1}{1 - x - y}.$$

On s'intéresse en particulier au développement en série autour de l'origine, donnée par

$$\sum_{\ell \geqslant 0} \sum_{k \geqslant 0} \binom{k+\ell}{\ell} x^k y^\ell.$$

Variété singulière $\mathscr{V}$ La variété singulière $\mathscr{V}$ de F est l'ensemble des points annihilant le polynôme $1 - x - y$:

$$\mathscr{V} = \{(x, 1-x) \mid x \in \mathbb{C}\}.$$

Points minimaux La figure 3.6 illustre l'image de $\mathbb{C}^2$ sous $(x, y) \mapsto (|x|, |y|)$. Considérons la limite sur le bord inférieur gauche de la région marquée $\mathscr{V}$. Ce sont des points (x, y) tels qu'il n'y a pas de point (x', y') dans $\mathscr{V}$ avec $|x'| < |x|$ et $|y'| < |y|$: il n'y a pas de point dans l'image de $\mathscr{V}$ à gauche et en dessous de ces points. Ainsi, les points appartenant à la pré-image de cette ligne de $\mathscr{V}$ sont minimaux. Ceux-ci satisfont $|1 - x| = 1 - |x|$, qui d'où $x \in \mathbb{R}_{\geqslant 0}$.

$$\text{Les points minimaux sont : } \quad \{(x, 1-x) \mid x \in \mathbb{R}_{\geqslant 0}\}.$$

Point strictement minimaux Si deux points minimaux $(x, 1-x)$ et $(x', 1-x')$ sont sur le même tore, alors $|x| = |x'| \implies x = x'$ puisqu'ils sont tous deux des nombres réels positifs. Ainsi, chaque point minimal est le seul sur son tore et **tous les points minimaux sont strictement minimaux.**

Domaine de convergence $\mathcal{D}$ Nous pouvons en déduire le domaine de convergence du développement en série à l'origine. Nous considérons les variables une à la fois. D'abord, considérons la somme intérieure pour un entier positif fixé ℓ. La sommation est une instance du théorème du binome généralisé :

$$\sum_{k \geqslant 0} \binom{\ell + k}{k} x^k = (1-x)^{-\ell-1}.$$

Il est clair que $|x| < 1$ lorsque cela converge. Comme les coefficients sont tous positifs et que l'expression est symétrique en x et y, l'ordre n'a pas d'importance et nous pouvons également conclure que $|y| < 1$.

Maintenant, si la série est convergente en (x, y) elle est convergente en $(|x|, |y|)$, nous considérons donc $\frac{1}{1-|x|-|y|}$. Cette série géométrique converge si et seulement si $|x| + |y| < 1$. Le domaine de convergence est donc

$$\mathcal{D} = \{(x, y) \in \mathbb{C}^2 \mid |x| + |y| < 1\}.$$

Frontière de convergence $\partial\mathcal{D}$ La frontière de ce domaine est $\partial\mathcal{D} = \{(x, y) \in \mathbb{C}^2 : |x| + |y| = 1\}$, et nous confirmons qu'il s'agit précisément de la contrainte des points minimaux. La figure 3.6 illustre différentes régions connectées après application $(x, y) \mapsto (|x|, |y|)$ c'est-à-dire les trois

régions distinctes de convergence et leurs limites. L'image de $\mathscr{V}$ sous cette application est la région en gris foncé.

3.4 La croissance exponentielle

La position des singularités d'une fonction analytique détermine l'ordre de croissance exponentiel de ses coefficients de TAYLOR.

Comme dans le cas univarié, les points minimaux déterminent les ratios/ordres/coefficients de la croissance exponentielle. Supposons que $\mathbf{z}$ est un point dans la domaine de convergence de la série $\sum_{\mathbf{n}} f(\mathbf{n})\mathbf{z}^{\mathbf{n}}$. Les séries géometriques convergentes étant absolument convergentes, la série

$$\sum_{\mathbf{n}} f(\mathbf{n})\ |z_1|^{n_1}\,|z_2|^{n_2}\dots|z_d|^{n_d}\ \text{ est convergente.}$$

Si une série est convergente, ses sous-séries le sont aussi :

$$\sum_{n\geqslant 0} f(n,n,\dots,n)\,|z_1|^{n}\,|z_2|^{n}\dots|z_d|^{n}\ \text{ est également convergente.}$$

La valeur de cette sommation est égale à $\sum f(n,n,\dots,n)\,|z_1z_2\dots z_d|^n$ et donc le point $|z_1z_2\dots z_d|$ est dans le domaine de convergence de la série univariée Diag $F(\mathbf{z})$. Nous avons ainsi lié la croissance exponentielle et le rayon de convergence :

$$\limsup_{n\to\infty} |f(n,n,\dots,n)|^{1/n} \leqslant |z_1z_2\dots z_d|^{-1} \quad \text{avec} (z_1,\dots,z_d)\in\overline{\mathcal{D}}$$

Nous trouvons

$$\limsup_{n\to\infty}\ |f(n,n,\dots,n)|^{1/n} \leqslant \inf_{(z_1,\dots,z_d)\in\partial\mathcal{D}} |z_1z_2\dots z_d|^{-1}.$$

C'est une limite - il se peut que la diagonale soit vide. Nous donnons des conditions pour lesquelles il y a égalité dans la section suivante. C'est un résultat très important. Pemantle et Wilson ajoutent la condition suivante : si dans un voisinage du rayon, la diagonale est définie, alors on a égalité dans l'inéquation précédente. Dans nos cas combinatoires, cette condition est fréquemment satisfaite et le supremum est atteint sur un point de la variété $\mathscr{V}$:

$$\rho = \sup_{\mathbf{z}\in\overline{\mathcal{D}}\cap\mathscr{V}} |z_1\dots z_d|. \tag{3.19}$$

3.4.1 Exemple

Nous revenons à notre exemple : $\frac{1}{1-x-y}$ avec $f_{k,\ell} = \binom{k+\ell}{k}$. Alors,

$$\limsup_{n\to\infty} f_{n,n}^{1/n} = \inf_{(x,y)\in\partial\mathcal{D}} |xy|^{-1} = \inf_{x\in\mathbb{R}} (x(1-x))^{-1} = 4.$$

Pour trouver la croissance exponentielle, comme la série est combinatoire nous pouvons restreindre notre attention sur les solutions réelles et positives. La valeur trouvée correspond à la limite $\lim_{n\to\infty} \binom{2n}{n}^{1/n} = 4$ pouvant être obtenue par l'approximation de Stirling. De même, sur un rayon quelconque, dirigé par r, s, on obtient

$$\limsup_{n\to\infty} f(rn, sn)^{1/n} = \inf_{(x,y)\in\partial\mathcal{D}} |x^r y^s|^{-1} = \inf_{x\in\mathbb{R}} (x^r (1-x)^s)^{-1}.$$

Ceci est minimisé à $x = \frac{r}{r+s}$ ce qui nous permet de déduire une expression pour croissance exponentielle :

$$\left(\left(\frac{r}{r+s} \right)^r \left(\frac{s}{r+s} \right)^s \right)^{-1}.$$

3.4.2 Stratégie

Maintenant, nous avons une stratégie : d'abord trouver les "singularités dominantes" parmi les points minimaux en considérant des points singuliers sur le domaine de la convergence. Quand la fonction rationnelle est combinatoire, et que nous nous intéressons à la croissance exponentielle, nous pouvons ne considérer que les solutions réelles. Nous trouvons le point qui minimise $\left|z_1^{r_1} \dots z_d^{r_d}\right|^{-1}$ parmi les solutions.

3.4.3 La fonction de hauteur

Nous cherchons donc à minimiser une fonction non linéaire. Il est plus facile de minimiser une fonction linéaire. On définit la fonction $h : \mathscr{V}^* \to \mathbb{R}$

$$h : (z_1, \dots, z_d) \mapsto -\log|z_1| - \dots - \log|z_d|. \tag{3.20}$$

Nous appelons cette fonction la *fonction de hauteur*. C'est une fonction réelle définie sur $\mathscr{V} \setminus \{\mathbf{z} : z_1 \dots z_d \neq 0\}$. L'application h est lisse et, par conséquent, elle est minimisée à ses points critiques, déterminés à la manière du calcul classique. Afin de travailler avec cette fonction et de visualiser son

interaction avec le domaine de convergence, nous regardons l'image de $\mathbb{C}^d$ dans $\mathbb{R}^d$ sous l'application $-$ relog :

$$\text{relog} : \mathbb{C}^d \to \mathbb{R}^d$$

$$-\text{relog} : \mathbf{z} \mapsto (-\log|z_1|, \ldots, -\log|z_d|).$$

Cela nous aide à analyser les différents dom aines de convergence. Dans ce système de coordonnées, $h(z) = c$ définit un hyperplan comme on le voit dans la figure 3.6(à droite). Pour minimiser $\left|z_1^{r_1} \ldots z_d^{r_d}\right|^{-1}$ sur $\partial\mathcal{D}$, on trouve le plus petit valeur de c pour que l'hyperplan $h(z) = c$ touche l'image de $\mathscr{V}$ dans l'image de $-$ relog. Chaque contact peut potentiellement représenter plusieurs points dans la pré-image de $-$ relog dans $\mathbb{C}^d$).

Si $H(\mathbf{z})$ est un polynôme de LAURENT, alors il s'avère que

$$\mathbb{R}^n \setminus \{-\text{relog}(\mathbf{z}) \mid \mathbf{z} \in \mathscr{V}\}$$

est une collection de régions convexes. Chaque ensemble convexe réel est en bijection avec une expansion de la série de LAURENT de la fonction rationnelle $\frac{1}{H(\mathbf{z})}$. Si $\frac{1}{H(\mathbf{z})}$ a une expansion en série, alors sous cette bijection elle correspond à la composante contenant $(r_1 n, r_2 n, ..., r_d n)$ pour n suffisamment grand. Dans la figure 3.6, nous voyons trois régions, chacune correspondant à un développement en série. La fonction $\frac{1}{1-x-y}$ a une développement de série de TAYLOR et elle correspond à la région marquée $\mathcal{D}$, comme nous l'avons indiquée. Dans ces notes, nous ne nous intéressons qu'aux développements de TAYLOR et nous ne dessinerons donc que la composante pertinente pour l'extraction de notre série (plus précisément, nous définissons une borne), mais il est utile de savoir que le processus décrit ici fonctionne pour 'autres extensions en séries. Les points critiques sont des solutions du système suivant, appelées les *équations des points critiques.*

$$H(\mathbf{z}) = 0, \quad r_1^{-1} z_1 \frac{\partial H(\mathbf{z})}{\partial z_1} = r_k^{-1} z_k \frac{\partial H(\mathbf{z})}{\partial z_k}, \quad k = 2, \ldots, d. \tag{3.21}$$

Dans ce qui suit, nous appelons les solutions les *points critiques*, même si elles sont définies géométriquement. La géométrie de la variété $\mathscr{V} = \{(z_1, \ldots, z_d) : H(z_1, \ldots, z_d) = 0\}$ aux points critiques décidera de la croissance sous-exponentielle.

Proposition 3.4.1. *Soit H un polynôme irréductible associé à la variété $\mathscr{V}$. Alors le point $\rho \in \mathscr{V}$ est un point critique pour $\mathbf{r} \in \mathbb{N}^d$ si et seulement si $\mathbf{r}$ peut être écrit sous la forme d'une combinaison linéaire du vecteur*

$$\left(z_1 \frac{\partial H}{\partial z_1}(\mathbf{z}), z_2 \frac{\partial H}{\partial z_2}(\mathbf{z}), \ldots, z_d \frac{\partial H}{\partial z_d}(\mathbf{z}) \right) \Bigg| \, \mathbf{z} = \rho.$$

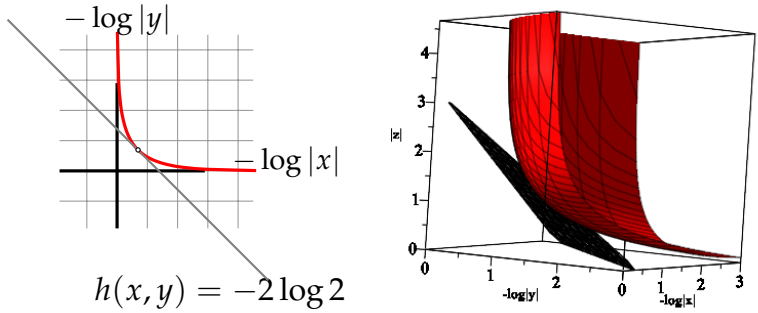


FIGURE 3.7 – *L'image sous* -relog *de la frontière de convergence pour les séries de* TAYLOR *de* $F(x,y) = \frac{1}{1-x-y}$ *(gauche) et* $F(x,y,z) = \frac{1}{1-x-y-z}$ *respectivement. On voit aussi les hyperplans tangent aux frontières :* $h(x,y) = 2\log 2$ *(gauche) et* $h(x,y,z) = 3\log 3$ *(droit).*

Rappel, un point critique est dit strictement minimal s'il se situe à la limite du domaine de convergence de la série. En outre, il est qualifié de minimum fini s'il n'y a qu'un nombre fini de points strictement minimaux. Un point critique est isolé s'il existe un voisinage de $\mathbb{C}^d$ où c'est le seul point critique. Dans la fonction génératrice que nous considérons dans ce texte, les points critiques isolés minimaux sont essentiels pour l'asymptotique. Ce sont des points singuliers isolés d'une expression rationnelle et il n'y a pas d'autre annulation du numérateur. C'est le cas générique le plus facile à étudier. Nous identifierons plus tard les points contributeurs. Ce sont les points qui contribuent réellement au terme dominant dans l'expansion asymptotique des coefficients.

3.4.4 Visualisation des points critiques

Qu'est ce que c'est, la fonction de hauteur ? Nous pouvons mieux le comprendre en regardant les régions de convergence sous cette transformation. Cela correspond à ce que nous avons déjà trouvé dans nos exemples. Par exemple, $1 - x - y = 0 \implies x + y = 1$ et

$$x\frac{\partial H(x,y)}{\partial x} = y\frac{\partial H(x,y)}{\partial y} \implies x = y \implies x = y = 1/2.$$

Le point critique est en $(1/2, 1/2)$. Nous pouvons le voir dans le domaine logarithmique de la convergence. La ligne $h(x,y) = 2\log(2)$ touche $\partial\mathcal{D}$ à $(-\log(2), -\log(2))$. La figure 3.7 illustre la géométrie dans le cas de $F(x,y,z) = \frac{1}{1-x-y-z}$. La fonction de hauteur h pour le rayon $\mathbf{r} = (1,1,1)$

définit un hyperplan $h(x, y, z) = 3 \log 3$ qui touche la limite de la convergence en un point unique $-\log 3 \cdot (1, 1, 1)$.

3.4.5 Exemple : Mots équilibrés

Rappelons que $\mathscr{L}$ est la classe combinatoire de mots binaires sur $\{\bullet, \circ\}$, de sorte qu'aucun mot ne possède le facteur $\circ\circ\circ$:

$$\mathscr{L} \equiv (\epsilon + \circ + \circ\circ) \cdot (\bullet \cdot (\epsilon + \circ + \circ))^*.$$

On dispose également de son sous-langage équilibré $\mathscr{L}_=$:

$$\mathscr{L}_= = \{\epsilon, \bullet\circ, \circ\bullet, \bullet\bullet\circ\circ, \bullet\circ\bullet\circ, \circ\bullet\circ\bullet, \circ\circ\bullet\bullet, \ldots\}$$

et de sa série génératrice :

$$L_=(y) = \mathrm{Diag} \frac{1 + x + x^2}{1 - y(1 + x + x^2)}.$$

Pour déterminer la croissance exponentielle des coefficients, nous trouvons d'abord la base de GRÖBNER [4] des équations de points critiques :

$$[x^2 - 1, 3y + x - 2].$$

Il existe deux solutions à cet ensemble d'équations $(1, 1/3)$ et $(-1, 1)$. Rappelons que si un point (x, y) se trouve sur la limite de convergence des séries de puissances, alors il en est de même pour $(|x|, |y|)$. Dans ce cas, nous voyons que, puisque (1, 1) n'est pas à la frontière de la convergence, ce n'est pas un point critique, mais $(1, 1/3)$ est un. En conséquence, $\limsup_{n\to\infty} a(n, n)^{1/n} = 3$.

Modulo des conditions de non dégénérescence, il existe des constantes complexes calculables C_k telles que pour tout entier positif N,

$$f(r_1\, n, \ldots, n\, r_d) = \left(\rho_1^{r_1} \ldots \rho_d^{r_d}\right)^{-n} \left[\sum_{k=0}^{N-1} C_k n^{-(d-1)/2-k} + O(n^{-(d-1)/2-N}).\right]$$

Nous considérons un problème classique avec une structure simple : $H(\mathbf{z}) = 1 - z_d P(z_1, \ldots, z_{d-1})$ pour certains polynômes P. Ceci nous permet de traiter un grand nombre de problèmes de mots et de marches dans un treillis, et les calculs ressemblent à ce que nous venons de faire.

4. C'est une manipulation des équations critiques qui pourrait fournir une équation en une seule variable, qu'on utilise pour trouver les points. Ici, on trouve notamment $x^2 - 1 = 0$.

3.5 Croissance sous-exponentielle

La nature des singularités détermine la manière dont le terme exponentiel dominant dans les coefficients est modulé par un facteur sous-exponentiel.

3.5.1 L'Approximation de STIRLING

L'approximation de STIRLING est une formule asymptotique pour la fonction factorielle :

$$n! \sim n^n e^{-n} \sqrt{2\pi n}.$$

Elle est très utile en combinatoire. Comment la retrouver avec nos méthodes ? On pourrait faire un estimation de la fonction Gamma, noté $\Gamma(x)$:

$$n! = \Gamma(n+1) = \int_0^\infty x^n e^{-x}\, dx.$$

Pour estimer l'intégrale, on trouve le point sur le contour d'intégration ou l'intégrant est maximisé. On récrit l'intégrale pour retirer la contribution dominante avec une erreur controllée. Plus précisément, on écrit $x^n\, e^{-x} = e^{n \log x - x}$, ce qui laisse voir que l'intégrant est maximisée quand $x = n$. On fait un changement de variables

$$\begin{aligned} n! &= \int_{-n}^\infty e^{n\log(n+u)-(n+u)}\, du && (3.22)\\ &= e^{n\log n - n} \int_{-n}^\infty e^{n\log\left(1+\frac{u}{n}\right)-u}\, du && (3.23)\\ &= e^{n\log n - n} \int_{-n}^\infty e^{-\frac{u^2}{2n}+O(u^3/n^2)}\, du && (3.24)\\ &\sim e^{n\log n - n} \frac{\sqrt{2n\pi}}{2} && (3.25) \end{aligned}$$

Dans l'équation (3.24) on utilise un développement limité de $\log(1+z)$:

$$\log(1+z) = z + z^2/2 + O(z^3),$$

et ensuite la formule bien connue : $\int_{-n}^\infty e^{-x^2/c}\, dx = \frac{\sqrt{c\pi}}{2}$. Pour terminee on aurait besoin de plus de justification pour vérifier l'erreur, ce que l'on ne fait pas ici. En résumé, il y a quelques techniques à retenir :

- L'écriture d'une intégrale en tant qu'intégrale gaussienne permet une transition rapide vers une expression en forme fermée ;
- Remplacer une fonction par le terme dominant de sa série de TAYLOR est utile pour faire des estimations contrôlées ;
- Des estimation sont optimale près des points critiques.

3.5.2 Intégrales de FOURIER-LAPLACE

On vient de regarder un exemple d'une intégrale de type FOURIER-LAPLACE. Ces intégrales sont de la forme

$$\int_{\mathcal{N}} A(\mathbf{t}) e^{-\lambda \phi(\mathbf{t})} dt_1 \dots dt_d,$$

où les fonctions A et ϕ sont analytiques sur le domaine d'intégration, et $\mathcal{N}$ est un voisinage dans $\mathbb{R}^d$. Dans le cas unidimensionnel, nous pouvons appliquer certaines des stratégies de l'échauffement la sous-section précédent : pour évaluer $\int A(z) e^{-\lambda \phi(z)}\, dz$, nous trouvons un point critique z_0 qui minimise $\phi(z)$ (pour maximiser l'intégrant), et ensuite, en supposant que ϕ soit lisse en z_0, nous approximons $\phi(z_0 + z)$ avec le terme principal de son expansion de TAYLOR en z_0. Puisque $\phi'(z0) = 0$, cette extension de série ressemble à

$$\phi(z) = \phi(z_0) + C_k (z - z_0)^k + \dots$$

pour certains C_k non nuls (très souvent $k = 2$). Nous n'entrons pas dans les détails, mais nous pouvons prouver le théorème suivant.

Proposition 3.5.1. *Soient ϕ et A des fonctions analytiques réelles. Supposons en outre que ϕ ait un minimum strict à z_0, $\phi(0) = 0$, $\phi(z) \sim C(z - z_0)^2$ et $A(z_0) \neq 0$. Alors,*

$$\int A(z)\, e^{-\lambda \phi(z)}\, dz \sim \sqrt{\frac{2\pi}{n\phi''(z_0)}} e^{-n\phi(z_0)}. \tag{3.26}$$

3.5.3 Intégrale de CAUCHY multidimensionnel

Généralisons d'abord la formule de CAUCHY.

Théorème 3.5.2. *Soit d un entier et notons $\mathbf{z} = (z_1, \dots, z_d)$. Supposons que $F(\mathbf{z}) \in \mathbb{Q}(\mathbf{z})$ est analytique à l'origine, avec le développement de* TAYLOR, *$F(\mathbf{z}) = \sum_{\mathbf{n} \in \mathbb{N}^d} f(\mathbf{n})\, \mathbf{z}^{\mathbf{n}}$. Alors pour $n \geqslant 0$,*

$$f(k_1, \dots, k_d) = \frac{1}{(2\pi i)^d} \int_T \frac{F(\mathbf{z})}{z_1^{k_1} \dots z_d^{k_d}} \cdot \frac{dz_1 \dots dz_d}{z_1 \dots z_d}, \tag{3.27}$$

où T est le tore $T(\epsilon) = T(\epsilon_1, \epsilon_2, \dots, \epsilon_d)$ a la propriété que chaque ϵ_j est suffisamment petit pour que $F(\mathbf{z})$ soit analytique à l'intérieur de $D(\epsilon)$ et continue à la frontière.

En gros, la preuve découle de la formule standard de CAUCHY par induction sur le nombre de variables.

3.5.4 Une formule pour les intégrales de FOURIER-LAPLACE

La prochaine étape consiste à généraliser la proposition 3.5.1. Prouver un tel théorème est hors de notre discussion, mais avec l'intuition que nous avons développée dans le cas univarié, le lecteur intéressé devrait pouvoir suivre les détails fournis dans PEMANTLE et WILSON. La version générale en plus grande dimension est la suivante. On peut revoir le théorème 7.7.5 dans [7]. Il utilise la matrice hessienne $\mathcal{H}$ de ϕ :

$$\mathcal{H} := \left[\frac{\partial^2}{\partial t_j \partial t_k} \phi(\mathbf{t}) \right]_{j,k}$$

Proposition 3.5.3. *Supposons que les fonctions $A(\mathbf{t})$ et $A(\mathbf{t})$ dans d variables sont lisses dans un voisinage $\mathcal{N} \subset \mathbb{R}^d$ de l'origine et que*

- *$\phi(\mathbf{0}) = 0$;*
- *la partie réelle de $\phi(\mathbf{t})$ est positive sur $\mathcal{N}$;*
- *ϕ a un point critique à $\mathbf{t} = \mathbf{0}$, c'est-à-dire que $(\nabla\phi)(\mathbf{0}) = \mathbf{0}$ et que l'origine est le seul point critique de ϕ dans $\mathcal{N}$;*
- *La matrice hessienne de ϕ à $\mathbf{t} = \mathbf{0}$ est non singulière.*

Ensuite, pour tout entier $M > 0$, il existe des constantes effectives $C_0, \ldots, C_M$ telles que

$$\int_{\mathcal{N}} A(\mathbf{t})\, e^{-n\phi(\mathbf{t})} d\mathbf{t} = \left(\frac{2\pi}{n}\right)^{d/2} \det(\mathcal{H})^{-1/2} \cdot \sum_{k=0}^{M} C_k n^{-k} + O\left(n^{-M-1}\right) \tag{3.28}$$

La constante C_0 est égale à $A(\mathbf{0})$.

De plus, si $A(\mathbf{t})$ disparaît à l'ordre de L à l'origine alors (au moins) les constantes $C_0, \ldots, C_{\lfloor \frac{L}{2} \rfloor}$ sont toutes nulles. Les constantes C_k sont données par une formule.

3.5.5 Comment utiliser cette formule ?

Supposons que A et B soient des polynômes non linéaires et non constants avec coefficients entiers. La fonction bivariée simple

$$F(x, y) = \frac{A(x)}{1 - yB(x)} = \sum f_{k\ell} x^k y^\ell$$

est une série combinatoire, car les $f_{k\ell}$ sont tous des entiers positifs. On retrouve souvent des fonctions sous cette forme pour les classes construites avec des séquences paramétrées par leur longueur. Ceci est courant dans les problèmes de marche et de mots. Nous notons que la variété singulière de F

est $\mathscr{V} = \{(x, \frac{1}{B(x)}) \mid x \in \mathbb{C}\}$. Comme il s'agit d'un problème combinatoire, on sait qu'il existe une solution réelle $x = \rho$ à l'équation $xB'(x) = 1$ (un point critique). On en déduit que

$$\begin{aligned}
[x^n y^n]A(x)y^n B(x)^n &= [x^n]A(x)B(x)^n \\
&= \frac{1}{2\pi i}\int_{|x|=\epsilon} \frac{G(x)S(x)^n}{x^{n+1}}\,dx \\
&= \frac{1}{2\pi i}\int_{|x|=\rho} \frac{A(x)B(x)^n}{x^{n+1}}\,dx \\
&= \frac{1}{2\pi i}\int_{t=-\pi}^{\pi} \frac{A(\rho e^{it})B(\rho e^{it})^n}{\rho^{n+1}e^{it(n+1)}} i\rho e^{it\rho}dt \\
&= \frac{\rho^{-n}B(\rho)^n}{2\pi}\int_{t=-\pi}^{\pi} A(\rho e^{it})\frac{B(\rho e^{it})^n}{B(\rho)^n}e^{-it(n+1)}dt \\
&= \frac{\rho^{-n}B(\rho)^n}{2\pi}\int_{t=-\pi}^{\pi} A(\rho e^{it})e^{-n\phi(t)}dt
\end{aligned}$$

avec $\phi(t) := \log\frac{B(\rho)}{B(\rho e^{it})} + it$. Nous reconnaissons une intégrale de type LAPLACE, et nous vérifions les hypothèses de la proposition 3.5.1 avec point minimum à 0. Nous vérifions que $\phi(0) = 0$, et $\phi'(0) = 0$ par construction parce que B a un point critique. La fonction $A(\rho e^{it})$ est clairement analytique puisque A est un polynôme. Le critère $A(\rho) \neq 0$ doit être vérifié au cas par cas. On en déduit la proposition la formule asymptotique du premier ordre :

$$[x^n y^n]F(x,y) \sim \left(\frac{B(\rho)}{\rho}\right)^n \frac{A(\rho)}{\sqrt{2\pi n\phi''(0)}}.$$

3.5.6 Exemple : mots binaires equilibrés

Maintenant on pourrait finir l'exemple qu'on a commencé. Le nombre de mot équilibrés sur $\{\circ, \bullet\}$ tel que aucun mot contient $\circ\circ\circ$ est

$$a(n,n) = [x^n y^n]\frac{1+x+x^2}{1-y(1+x+x^2)}.$$

Vu comme problème de diagonale, il y a un point critique à $\rho = (1, 1/3)$. Alors, $A(x) = B(x) = 1 + x + x^2$,

$$a(n,n) \sim 3\frac{3^n}{\sqrt{2\pi n}}.$$

Exercice Soit $\mathcal{L} = (0 + (1(01^*0)^*1))^*$. Donner une approximation pour le nombre de mots de longueur $2n$ dans le langage $\mathcal{L}_= = \{w \in \mathcal{L} \mid |w|_0 = |w|_1\}$.

3.5.7 Une formule explicite

On a une formule plus générale qui pourrait être démontrée avec le même idée de base. Pour l'appliquer on doit d'abord confirmer que le problème est bien defini. Pour la direction (r, s), soit $\rho(r, s)$ le point critique correspondant – cela définit une fonction. On a besoin que cette fonction soit continue, et en fait qu'elle soit lisse dans un voisinage de notre choix de (r, s).

Théorème 3.5.4. *Soit $F(x, y) = \frac{G(x,y)}{H(x,y)}$ une fonction méromorphe et supposons que la fonction ρ soit lisse dans un voisinage $\mathcal{N}$ de (r, s). Supposons aussi que $G(\rho(r, s)) \neq 0$. On définit*

$$Q : (r, s) \mapsto \left(-y^2 H_y^2 x H_x - y H_y x^2 H_x - x^2 y^2 \left(H_y^2 H_{xx} + H_x^2 H_y y - 2 H_x H_y H_{xy}\right)\right)(\rho(r, s)).$$

Si Q est non nulle dans un voisinage de (nr, ns) alors si $\rho = \rho(r, s)$,

$$f(nr, ns) \sim \frac{G(\rho)\left(x^{-r} y^{-s}\right)^n}{\sqrt{2\pi}} \sqrt{\frac{-\rho_2 H_y(\rho)}{ns Q(\rho)}}.$$

quand n tend vers l'infini.

Exercice Refaire l'exemple en haut avec cette formule.

3.6 Recherches dans ce domaine

Ceci est une domaine très actif avec des liens au sein de la théorie des nombres, et des fonctions elliptiques. Il reste beaucoup de problèmes ouverts et de nombreuses pistes de recherches.

Il y a des applications en combinatoire. Les problèmes de marches sont très intéressants pour ceux qui aiment l'aspect systèmatique. On peut commencer avec [4, 3] et [2]. Pour voir les techniques présentées ici, voir [9, 1]. Les mots sont beaucoup étudiés en combinatoire aussi [8]. Une référence classique pour la combinatoire est [17].

Quand il y a plus d'un seul point critique, ce n'est pas forcément plus compliqué, surtout quand il n'y a qu'un nombre fini de point minimaux et

que la variété reste lisse à ces points. Il existe aussi des stratégies quand le dénominateur factorise, ou quand la géométrie devient plus compliquée, voir [12, 15, 16].

Les sujets suivants font l'objet d'une recherche active. Déterminer quels points critiques contribuent aux asymptotiques dominantes revient à comparer les modules des composantes. MELCZER et SALVY [10] décrivent des méthodes numériques pour effectuer cette filtration dans le cas combinatoire. Remarquablement, ils sont capables de lier étroitement la complexité des calculs et des représentations de données. En général, il y a peu d'implémentations de ces techniques [14].

On peut aussi poser des questions sur la nature des séries génératrices. Il y a un lien fort entre la complexité des fonctions (est-elle algébrique ? est-ce qu'elle satisfait une équation différentielle ?) et la structure combinatoire. Les singularités sont très utiles pour classer les séries génératrices, et donc les classes combinatoires.

Remerciements

J'offre mes remerciements sincères aux membres de SPACE Tours (Institut Denis POISSON), PEC LaBRI (Bordeaux), Johannes KEPLER University Linz (Autriche), qui m'ont énormément aidé dans le développement de cette histoire. Je remercie Pierre OHLMAN et Cedric CHAUVE qui ont corrigé les fautes de français (ce n'était pas facile, je vous assure).

Bibliographie

[1] Y. BARYSHNIKOV, W. BRADY, A. BRESSLER et R. PEMANTLE : Two-dimensional quantum random walk. *J. Stat. Phys.*, 142(1):78–107, 2011.

[2] A. BOSTAN : *Calcul formel pour la combinatoire des marches*. Thèse de doctorat, Université Paris 13, 2017. Habilitation à diriger des recherches.

[3] A. BOSTAN et K. RASCHEL : Compter les excursions sur un échiquier. *Pour la Science*, (449):40 – 46, 2015.

[4] M. BOUSQUET-MÉLOU et M. MISHNA : Walks with small steps in the quarter plane. *Dans Algorithmic Probability and Combinatorics*, vol. 520 de *Contemp. Math.*, pp. 1–40. Amer. Math. Soc., 2010.

[5] P. FLAJOLET et A. ODLYZKO : Singularity analysis of generating functions. *SIAM J. Discrete Math.*, 3(2):216–240, 1990.

[6] P. FLAJOLET et R. SEDGEWICK : *Analytic combinatorics*. Cambridge University Press, Cambridge, 2009.

[7] L. HÖRMANDER : *The analysis of linear partial differential operators. I*, vol. 256 de *Grundlehren der Mathematischen Wissenschaften*. Springer-Verlag, Berlin,

second édn, 1990. doi:10.1007/978-3-642-61497-2. Distribution theory and Fourier analysis.

[8] M. LOTHAIRE : *Combinatorics on words*, vol. 17 de *Encyclopedia of Mathematics and its Applications*. Addison-Wesley Publishing Co., Reading, Mass., 1983.

[9] S. MELCZER et M. MISHNA : Asymptotic lattice path enumeration using diagonals. *Algorithmica*, 75(4):782–811, 2016.

[10] S. MELCZER et B. SALVY : Symbolic-numeric tools for analytic combinatorics in several variables. *Actes de ISSAC'16*, pp. 333–340. ACM, New York, 2016.

[11] M. MISHNA : *Analytic Combinatorics : A Multidimensional Approach*. CRC Press, 2019.

[12] R. PEMANTLE et M. WILSON : Asymptotics of multivariate sequences. II. Multiple points of the singular variety. *Combin. Probab. Comput.*, 13(4-5):735–761, 2004.

[13] R. PEMANTLE et M. WILSON : *Analytic combinatorics in several variables*, vol. 140 de *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2013.

[14] A. RAICHEV : amgf documentation – release 0.8. https ://github.com/araichev/amgf, 2012.

[15] A. RAICHEV et M. C. WILSON : Asymptotics of coefficients of multivariate generating functions : improvements for smooth points. *Electron. J. Combin.*, 15(1):Research Paper 89, 17, 2008.

[16] A. RAICHEV et M. C. WILSON : Asymptotics of coefficients of multivariate generating functions : improvements for multiple points. *Online J. Anal. Comb.*, (6):21, 2011.

[17] R. STANLEY : *Enumerative Combinatorics*, vol. 2. Cambridge University Press, 1999.

Chapitre 4

Calculer avec les nombres réels

Fredrik JOHANSSON
traduit de l'anglais par Xavier CARUSO

Le calcul sur machine avec des nombres réels pose des difficultés fondamentales, liées d'une part aux questions de calculabilité elles-mêmes et, d'autre part, aux problèmes pratiques d'efficacité et de suivi de précision. Dans ce chapitre, nous faisons le point sur les problématiques de ce domaine et sur les concepts fondamentaux qui ont été introduits pour les résoudre. Nous présentons également quelques outils (comme les différentes manières de représenter les nombres réels) qui permettent de surmonter en pratique les difficultés qui apparaissent.

4.1 Introduction

Les ordinateurs ont été inventés avec l'ambition de manipuler les nombres. Pourtant, bien souvent, ils semblent n'avoir qu'une faible maîtrise des nombres réels. L'exemple suivant, extrait d'une session de SageMath, illustre parfaitement ceci :

```
sage: sqrt(2.0)/2.0 == 1.0/sqrt(2.0)
False
sage: sqrt(2.0)/2.0; 1.0/sqrt(2.0)
0.707106781186548
0.707106781186547
```

Bien sûr, le test d'égalité $\sqrt{2}/2 = 1/\sqrt{2}$ a échoué car les calculs ont été menés avec des nombres réels approchés au lieu de nombres réels exacts. Dans certaines situations, une erreur sur la 15^{e} décimale, comme celle

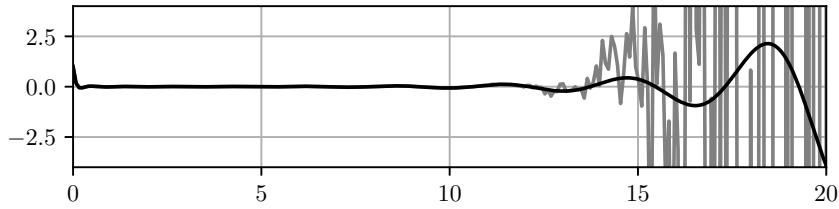


FIGURE 4.1 – *En noir : le graphe de la fonction hypergéométrique* ${}_1F_1(-50, 3, x)$ *sur l'intervalle* $[0, 20]$. *En gris : le graphe (erroné) de la même fonction calculé par* `scipy.special.hyp1f1`. *La différence entre les deux graphes provient des erreurs d'arrondis.*

observée ci-dessus, peut conduire *in fine* à des aberrations totales, comme illustré par la figure 4.1. Pourquoi autorisons-nous de tel crimes contre les mathématiques ? Plusieurs raisons acceptables peuvent être avancées :

1. les nombres réels exacts sont intrinsèquement des objets non calculables ; typiquement, un nombre réel tiré au hasard (comme 2,65323025327443 . . .) contient une quantité infinie d'information qui ne peut tenir dans la mémoire d'un ordinateur
2. si nous choisissons de nous limiter à un sous-ensemble « raisonnable » de nombres réels que l'on sait représenter par des structures finies, il existera toujours un certain nombre d'opérations impossibles à implémenter
3. et, même dans les cas où nous disposons d'algorithmes exacts pour les opérations les plus courantes, ceux-ci sont généralement coûteux et difficiles à implémenter.

Il y a donc un véritable besoin de considérer des nombres réels approchés, avec toutes les difficultés que cela implique.

L'objectif de ce texte est de présenter les difficultés inhérentes au calcul approché sur les nombres réels, sans toutefois entrer dans les rouages de l'arithmétique à virgule flottante (qui est le standard le plus répandu) ni dans ceux de l'implémentation. Nous verrons que certains problèmes sont intrinsèquement difficiles alors que d'autres peuvent être élégamment résolus avec les outils adéquats.

Ces outils, qui permettent de calculer avec des nombres réels approchés de façon plus fiable que l'arithmétique à virgule flottante classique, sont nombreux : il y a, par exemple, l'arithmétique multi-précision, l'arithmétique d'intervalles et différents types d'arithmétique paresseuse ou symbolique. Chacun d'entre eux a ses propres avantages et inconvénients.

Plusieurs d'entre eux sont disponibles à l'utilisation dans le logiciel SageMath. Nous encourageons la lectrice à tester par elle-même les nombreux exemples qui seront présentés dans la suite de ce texte. Pour aller plus loin, nous conseillons le livre *Calcul mathématique avec Sage* [1] dont plusieurs chapitres sont consacrés aux méthodes numériques.

4.2 Nombres algébriques

D'une manière générale, *calculer* avec une structure mathématique donnée S suppose que l'on dispose, d'une part, d'un moyen de représenter les éléments de S sous forme numérique, *i.e.*, par des suites de bits, et, d'autre part, d'algorithmes mettant en œuvre les opérations sur les éléments de S dans la représentation suscitée. Une structure S pour laquelle on dispose de ces prérequis est dite *calculable* ou *effective*. Dans la suite, nous emploierons le mot *effectif* pour ce concept et réservons l'utilisation du mot *calculable* pour une notion plus technique qui sera introduite plus tard.

Théorème 4.2.1. *L'anneau des entiers $\mathbb{Z}$ muni des opérations $\{+, -, \times\}$ et des prédicats de comparaison $\{=, \neq, \leqslant, <, \geqslant, >\}$ est effectif.*

La manière usuelle de représenter les entiers est d'utiliser des suites de chiffres (en base 10, ou plus traditionnellement pour les ordinateurs, en base 2 ou 2^{64}). Les algorithmes d'addition, de soustraction et de multiplication sont nombreux, les plus simples d'entre eux étant ceux des livres d'école où l'on effectue l'opération chiffre par chiffre en partant de la droite [1]. Bien entendu, beaucoup d'autres opérations sur les entiers, au delà de celles mentionnées dans le théorème 4.2.1, sont effectives : on peut citer la valeur absolue $|\cdot|$, le plus grand commun diviseur (pgcd), la factorisation en nombre premiers, *etc.*

Voici un corollaire simple du théorème 4.2.1 :

Théorème 4.2.2. *Le corps des nombres rationnels $\mathbb{Q}$ (muni des quatre opérations $\{+, -, \times, /\}$ et des prédicats de comparaison) est effectif.*

En transpirant davantage, il est aussi possible de démontrer un résultat analogue pour les nombres algébriques.

Théorème 4.2.3. *Le corps $\overline{\mathbb{Q}}$ formé des nombres complexes x qui sont solutions d'une équation algébrique de la forme $f(x) = 0$ avec $f \in \mathbb{Q}[x]$ (f n'étant pas le polynôme nul), est effectif pour les opérations de corps, la norme et la conjugaison complexe.*

1. Nous évoquerons brièvement des algorithmes plus rapides dans la partie 4.4.4 mais, pour ce qui concerne l'aspect effectif de $\mathbb{Z}$, ces considérations ne sont pas pertinentes.

À partir de là, on s'aperçoit que l'anneau des polynômes à coefficients dans $\overline{\mathbb{Q}}$ ou celui des matrices à coefficients dans $\overline{\mathbb{Q}}$ est aussi effectif.

Revenons un instant sur l'exemple de l'introduction. Il se trouve que SageMath propose une implémentation du corps $\overline{\mathbb{Q}}$ que nous pouvons utiliser à la place des nombres réels flottants. Voici le résultat que l'on obtient dans ce cas :

```
sage: x = QQbar(2)
sage: sqrt(x)/2 == 1/sqrt(x)
True
```

Soulignons bien que le corps $\overline{\mathbb{Q}}$ ne permet pas d'émuler parfaitement $\mathbb{R}$ ou $\mathbb{C}$, des constantes comme π ou e y étant inaccessibles. Il est cependant possible de faire déjà beaucoup de mathématiques à l'intérieur de $\overline{\mathbb{Q}}$; notamment, la majeure partie des problèmes calculatoires d'origine géométrique peut se traiter entièrement à l'aide de nombres algébriques.

L'inconvénient des nombres algébriques est le coût calculatoire qu'ils nous obligent à payer pour leur manipulation. L'évaluation de $1+1$ dans le corps `QQbar` de SageMath prend des milliers de cycles machine alors que tout processeur est *a priori* capable de réaliser plusieurs additions à un chiffre par cycle. Même en ignorant ces surcoûts qui ne jouent finalement que par un facteur constant, les algorithmes de calcul exact sur les nombres algébriques ont généralement une complexité asymptotique médiocre, même pour les tâches les plus simples (comme l'addition). Par exemple, ils nécessitent, la plupart du temps, de calculer un polynôme annulateur explicite de chaque nombre manipulé, polynômes qui sont souvent énormes.

Exemple 4.2.4. *Soit $x = \sqrt{2} + \sqrt{3} + \ldots + \sqrt{p_N}$ le nombre défini comme la somme des racines carrés de N premiers nombres premiers. Le polynôme minimal de x sur $\mathbb{Q}$ est de degré 2^N.*

Voici une expérience simple qui montre les limitations de `QQbar` :

```
sage: x = QQbar(sum(sqrt(nth_prime(n+1)) for n in range(6)))
sage: %time x - (x - 1) - 1 == 0
CPU times: user 8.98 s, sys: 14.4 ms, total: 9 s
Wall time: 9.17 s
True
```

Si l'on remplace `range(6)` par `range(7)`, le temps de calcul est multiplié par plus de 20 ! Dans ce cas particulier, il y aurait de bien meilleures façons de faire le calcul. Par exemple, nous aurions pu utiliser l'*anneau*

symbolique `SR` de SageMath à la place de `QQbar`. Toutefois, dans d'autres situations, `SR` pourrait être moins efficace, voire inutilisable [2].

Malgré l'existence de tels exemples, il ne faut pas exagérer les défauts de l'arithmétique exacte. Il serait trop facile de prendre un algorithme classique (comme, disons, l'élimination de GAUSS), de se rendre compte qu'il est peu performant dans le cas de l'arithmétique exacte, et de conclure hâtivement que le calcul exact ne peut conduire qu'à une impasse. Au contraire, il existe souvent des méthodes conçues spécialement pour l'arithmétique exacte qui, dans certaines situations, peuvent avoir d'excellentes performances, bien meilleures que celles de l'arithmétique approchée [3].

4.2.1 Logique et décidabilité

Il devient nettement plus difficile de mener des calculs algébriques lorsque interviennent, en même temps, des formules logiques avec quantificateurs. Dans ce contexte, plutôt que le mot *effectif*, nous utiliserons le terme *décidable* qui nous paraît plus approprié. Le célèbre 10e problème de HILBERT pose la question suivante sur les nombres entiers : existe-t-il un algorithme qui décide si une équation diophantienne donnée admet (au moins) une solution entière ? MATIYASEVICH donna une réponse négative à la question de HILBERT en 1970.

Théorème 4.2.5 (Corollaire du théorème M-R-D-P)**.** *Il existe un polynôme $f \in \mathbb{Z}[x_1, \ldots, x_n]$ pour lequel aucun algorithme ne peut décider si l'équation $f(x_1, \ldots, x_n) = 0$ a une solution $(x_1, \ldots, x_n) \in \mathbb{Z}^n$.*

L'ingrédient principal derrière la démonstration du théorème M-R-D-P est le suivant : les équations diophantiennes sont des objets mathématiques suffisamment généraux pour coder le comportement de machines de TURING arbitraires et ainsi rendre compte, d'une certaine manière, des limitations fondamentales de la théorie de la calculabilité à l'instar du problème de l'arrêt de TURING ou du théorème d'incomplétude de GÖDEL.

À l'opposé, lorsque l'on remplace $\mathbb{Z}$ ou $\mathbb{Q}$ par un corps algébriquement clos comme $\overline{\mathbb{Q}}$, de puissants résultats de décidabilité sont disponibles, de sorte que certains problèmes de décision deviennent plus simples.

2. L'exemple présenté ici est, en réalité, trivial à traiter dans le cas de l'anneau symbolique car les éléments qui apparaissent sont des sommes de radicaux simples qui s'éliminent directement. Mais, de manière générale, les nombres algébriques ne s'expriment pas à l'aide de radicaux ou peuvent avoir des expressions plus compliquées qui rendent leur manipulation plus délicate.

3. À ce sujet, on peut comparer `random_matrix(QQ, n, n).det()` (très optimisé) et `random_matrix(RR, n, n).det()` en SageMath pour différentes valeurs de n.

Par exemple, si $R = \overline{\mathbb{Q}} \cap \mathbb{R}$ désigne le corps des nombres algébriques réels, il existe un algorithme qui, étant donné un polynôme $f \in R[x_1, \ldots, x_n]$ décide si $f(x_1, \ldots, x_n) \geqslant 0$ pour tous $x_1, \ldots, x_n \in R$. Dans le même registre, un résultat plus fort est le théorème d'élimination des quantificateurs de TARSKI, démontré dans les années 1950, que nous énonçons brièvement ci-dessous.

Théorème 4.2.6 (TARSKI)**.** *Toute formule logique du premier ordre (i.e. formée à partir des opérations booléennes et des quantificateurs $\forall, \exists$) en n variables $x_1, \ldots, x_n \in R$ ne faisant intervenir que des égalités et inégalités polynômiales est décidable.*

Une conséquence du théorème de TARSKI est que les énoncés de la géométrie euclidienne sont décidables (une fois qu'ils ont été proprement formalisés).

L'algorithme originel de TARSKI réalisant l'élimination des quantificateurs avait une complexité lovecraftienne [4]. En 1975, COLLINS inventa la méthode de décomposition cylindrique algébrique (CAD), ce qui lui permit de résoudre le problème de l'élimination des quantificateurs en temps « seulement » doublement exponentiel (*i.e.* $2^{2^{O(n)}}$ où n est le nombre de variables) dans le pire cas. Les algorithmes de calcul exact en géométrie, tels que la CAD, sont aujourd'hui disponibles dans de nombreux logiciels de calcul et sont utilisés pour des applications variées comme la planification de mouvements de robots.

4.3 Nombres réels

De par leur définition, les nombres réels sont le lieu propice aux constructions par passage à la limite et, de ce fait, à la définition de constantes telles que π, de fonctions transcendantes comme l'exponentielle et à la mise en place de l'analyse avec le calcul différentiel (dérivation, intégration), la sommation des séries infinies, *etc.*

Une suite de CAUCHY $a_0, a_1, \ldots$ de nombres rationnels possède une limite $\lim_{n\to\infty} a_n \in \mathbb{R}$ et, de fait, $\mathbb{R}$ peut être défini formellement comme l'ensemble des classes d'équivalence de suite de CAUCHY de nombres rationnels. De manière à peine moins formelle, on peut aussi définir $\mathbb{R}$ comme l'ensemble des « nombres avec une infinité de chiffres après la virgule », à l'instar de 3,141 ... (ou plus exactement des classes d'équivalence de telles écritures car, par exemple, $0{,}999\ldots = 1{,}000\ldots$).

4. De l'horreur cosmique de l'inconnu des histoires de H. P. LOVECRAFT. Le terme technique plus approprié serait *complexité non élémentaire*.

Bien entendu, il n'est pas possible de stocker une suite infinie $a_0, a_1, \ldots$ de manière exhaustive sur un ordinateur et, en général, il est même impossible de la coder de manière implicite. L'indénombrabilité de $\mathbb{R}$, démontrée par CANTOR, implique que la plupart des nombres réels ne peuvent être uniquement déterminés par une quantité finie d'information. Par conséquent, le corps $\mathbb{R}$ n'est, à l'évidence, pas effectif. Dans tous les cas, calculer avec $\mathbb{R}$ signifie donc nécessairement calculer avec un sous-ensemble restreint et dénombrable de $\mathbb{R}$, par exemple $R = \overline{\mathbb{Q}} \cap \mathbb{R}$ ou éventuellement d'autres ensembles plus gros incluant des nombres transcendants comme $R(\pi, e, \log(2))$.

4.3.1 Nombres réels calculables

Une possibilité pour décrire un nombres réel de manière effective est de se donner un algorithme qui calcule une suite de CAUCHY le représentant. Formellement, ceci conduit à la définition suivante d'un *nombre réel calculable*.

Définition 4.3.1. *Un* nombre réel calculable *est un nombre réel x pour lequel il existe un programme (dans le sens d'une machine de* TURING*) qui, étant donnée une précision $p \in \mathbb{Z}$, renvoie un nombre rationnel $\widehat{x}$ tel que $|x - \widehat{x}| < 2^{-p}$.*

Plus généralement, on définit les fonctions calculables comme suit.

Définition 4.3.2. *Une* fonction calculable *(sur $\mathbb{R}$) est une fonction f pour laquelle il existe un programme qui, étant donnés une précision $p \in \mathbb{Z}$ et un programme calculant un nombre réel (calculable) x, renvoie un nombre rationnel $\widehat{y}$ tel que $|f(x) - \widehat{y}| < 2^{-p}$.*

Les définitions précédentes s'étendent sans difficulté aux nombres complexes, aux fonctions de plusieurs variables, *etc*. Il est à noter également que les nombres calculables peuvent être considérés comme des fonctions calculables n'acceptant aucun argument (une fonction de 0 variable). Enfin, il est facile de vérifier que la composée de deux fonctions calculables est calculable.

Exemple 4.3.3. *L'addition est une fonction calculable. En effet, étant donnés une précision $p \in \mathbb{Z}$ et des programmes calculant les nombres réels x et y, on peut obtenir des approximations de x et y à 2^{-p-1} près (en appelant les programmes suscités avec la précision $p + 1$) et ajouter ces approximations.*

Les nombres calculables forment un sous-ensemble dénombrable de $\mathbb{R}$. Les nombres algébriques et les fonctions algébriques sont toutes calculables,

mais il existe aussi des nombres et des fonctions calculables qui sont transcendantes. C'est le cas, par exemple, du nombre π étant donné qu'il peut être approché par les sommes partielles de la série $4\sum_{k=0}^{\infty}(-1)^n/(2n+1)$. De la même manière, la fonction exponentielle est calculable car elle peut être approchée par sa série de TAYLOR. Toutefois, il n'est pas vrai que toutes les fonctions simples soient calculables : nous verrons dans la partie 4.3.3 une restriction essentielle à la calculabilité.

4.3.2 Expressions symboliques

Une autre possibilité pour représenter les nombres réels consiste à utiliser des formules symboliques. Par exemple, en supposant que les entiers, les opérations arithmétiques et la constante π sont des symboles connus, nous pouvons former la quantité $\sqrt{2}+\frac{5}{3}\pi$ en l'encodant, au choix, comme une simple chaîne de caractères ou sous forme d'arbre comme suit : $(+,(\sqrt{\cdot},2),(\times,(/,5,3),\pi))$.

Les nombres réels pouvant être décrits de cette manière (à partir d'un langage fixé *a priori*) sont parfois appelés les *nombres réels symboliques* ou *nombres réels définissables* [5]. Les expressions symboliques sont trivialement effectives dans le sens où il est toujours possible de calculer la valeur qu'elles représentent en enchaînant les opérations décrites. Cette affirmation doit toutefois être prise avec des pincettes : de même que l'argent n'a de véritable valeur que lorsqu'il peut être échangé contre des biens ou des services, une expression symbolique ne représente un nombre réel calculable que si on peut l'interpréter comme une recette pour fabriquer un programme complexe à partir de programmes élémentaires qui représentent π, l'addition, *etc*.

4.3.3 Le test d'égalité

Les fonctions calculables ou un système suffisamment riche de formules symboliques (incluant, en particulier, les opérateurs logiques et les opérations usuelles de l'analyse comme $\lim_x f(x)$, $\int f(x)dx$) permettent, plus ou moins, d'exprimer tous les nombres réels qui apparaissent dans les problèmes concrets que l'on rencontre.

5. Ces notions sont toutefois informelles. En réalité, la formalisation correcte de la notion de réel définissable est un exercice délicat qui ouvre la porte à des questions subtiles; le lecteur pourra consulter `https://mathoverflow.net/questions/44102/is-the-analysis-as-taught-in-universities-in-fact-the-analysis-of-definable-numb/44129#44129` à ce propos.

Cependant, savoir exprimer un certain ensemble de nombres n'est pas synonyme de savoir le manipuler efficacement. Dans le cas des réels, un problème critique est le test d'égalité.

Problème 4.3.4 (Test d'égalité). *Étant donnés deux nombres réels, décider si $a = b$ ou, de manière équivalente, si $a - b = 0$.*

Dans l'idée de comparer des nombres réels, une idée naturelle est de comparer leurs approximations. Considérons par exemple le nombre $a = 8\int_0^\infty \cos(2x)\prod_{n=1}^\infty \cos\left(\frac{x}{n}\right) dx$. Voici une approximation numérique de a avec 15 décimales obtenue grâce à la bibliothèque `mpmath` de SageMath [6].

```
sage: from mpmath import mp
sage: print(8 * mp.quadosc(lambda x: mp.cos(2*x) * mp.nprod(
...       lambda n: mp.cos(x/n), [1,mp.inf]),
...           [0,mp.inf], omega=1))
3.14159265358979
```

Le résultat obtenu ressemble étrangement à π et, en effet, les 15 premières décimales de π sont exactement les mêmes que celles de a :

```
sage: print(mp.pi)
3.14159265358979
```

Il s'agissait toutefois d'un piège ! En fait, le nombre a n'est *pas* égal à π, mais l'est seulement à 10^{-41} près [21]. Cet exemple illustre que, de manière générale, il n'est pas possible de démontrer l'égalité de deux nombres réels en se contentant de comparer des approximations numériques. Certes, ceci est bien évident et ne surprendra aucunement notre lectrice aguerrie mais il est, malgré tout, extrêmement important de garder cette conclusion bien présente à l'esprit.

En contrepartie, il est toujours possible de démontrer que deux nombres calculables ne sont *pas* égaux en en calculant des approximations suffisamment fines. Typiquement, pour l'exemple précédent, on se serait rendu compte de la différence si on avait fait l'effort de poursuivre le calcul jusqu'à 50 chiffres après la virgule. Autrement dit, l'*inégalité* entre nombres réels calculables est un problème semi-décidable : l'algorithme qui consiste à comparer des approximations de plus en plus précises s'arrête lorsque les nombres réels sont différents mais boucle indéfiniment dans le cas contraire. Cependant, même dans le cas favorable de nombres réels

6. À cause des oscillations de la fonction dont on calcule l'intégrale, il est nécessaire d'utiliser la fonction spéciale `quadosc` pour avoir une valeur précise de a. Le calcul dans cet exemple particulier prend énormément de temps !

différents, il n'est pas possible de prédire *a priori* s'il sera nécessaire, pour conclure, de calculer 50 ou $10^{10^{50}}$ décimales.

A contrario, décider l'égalité de nombres entiers est un problème facile, lié au fait qu'il existe une représentation canonique des entiers, à savoir l'écriture en base 10 (ou 2, ou 2^{64}). En outre, les algorithmes classiques d'addition et de multiplication respectent cette représentation canonique. Ceci permet littéralement de *démontrer* par le calcul que $2 \times 6 = 3 \times 4 = 12$, par exemple. De même, les nombres algébriques admettent des représentations canoniques ; on en déduit que le problème 4.3.4 est effectif sur $\overline{\mathbb{Q}}$.

Pour les nombres calculables ou les expressions symboliques, il n'existe généralement pas de forme canonique qui permette de tester l'égalité. Comparer les valeurs de deux nombres calculables représentés par deux programmes se heurte rapidement au problème de l'arrêt de TURING. Pour les expressions symboliques, on peut certes implémenter des règles de simplification qui permettent de conclure dans *certains* cas (par exemple $\sqrt{2}/2 - 1/\sqrt{2} = 0$ ou $\sin(\pi) = 0$) ; cependant, il est illusoire de penser pouvoir résoudre le problème 4.3.4 en général, étant donné que l'expressivité du calcul symbolique est telle qu'elle permet *grosso modo* de coder n'importe quel énoncé mathématique.

Exemple 4.3.5. *L'hypothèse de* RIEMANN *est équivalente à l'égalité suivante :*

$$\frac{1}{\pi}\int_0^\infty \log\left(\left|\frac{\zeta\left(\frac{1}{2}+it\right)}{\zeta\left(\frac{1}{2}\right)}\right|\right)\frac{1}{t^2}\,dt \stackrel{?}{=} \frac{\pi}{8}+\frac{\gamma}{4}+\frac{\log(8\pi)}{4}-2 \qquad (4.1)$$

où $\zeta(s)$ est la fonction de RIEMANN *et $\gamma = \lim_{n\to\infty}\left[\left(\sum_{k=1}^{n}\frac{1}{k}\right)-\log(n)\right]$ est la constante d'*EULER [7].

Une démonstration (ou une infirmation) de l'égalité (4.1) entre nombres réels est donc équivalente à l'un des problèmes du millénaire, littéralement une question à un million de dollars.

Conséquences sur les fonctions calculables

Les problèmes de décidabilité que nous venons de discuter se posent plus généralement lorsque l'on cherche à construire des représentations paresseuses ou symboliques de nouveaux nombres réels. Par exemple :

— Étant donnée une description symbolique ou algorithmique d'une suite a_n, comment décider si la limite $x = \lim_{n\to\infty} a_n$ existe ? (En

7. `https://mathoverflow.net/q/279936`. Exercice : vérifier l'égalité (4.1) numériquement à l'aide de `mpmath`. (Indication : on pourra découper l'intégrale entre les zéros successifs de la fonction zêta de RIEMANN.)

général, cette question est indécidable, comme on le démontre en la réduisant au problème de l'arrêt.)

— Étant donné un nombre réel x, il est nécessaire de savoir si $x \neq 0$ avant de pouvoir dire que $1/x$ représente un nombre réel. Pareillement, étant donnée une fonction continue par morceaux comme

$$f(x) = \begin{cases} 1 & x \geqslant 0 \\ 0 & x < 0 \end{cases},$$

calculer $f(x)$ où x est donné par un programme demande de décider préalablement si x est positif ou négatif. Or, ceci est à nouveau indécidable étant donné que des approximations arbitrairement proches de x peuvent être de signe variable lorsque $x = 0$.

Le dernier exemple que nous avons donné met en lumière le principe suivant :

Théorème 4.3.6. *Les fonctions calculables sont toutes continues.*

À la lumière du théorème 4.3.6, examinons quelques exemples de fonctions calculables et non calculables. Pour commencer, la solution d'un système non singulier d'équations linéaires est calculable dès lors que les coefficients du système le sont.

Théorème 4.3.7. *La matrice A^{-1} est calculable si $A \in M_{n,n}(\mathbb{C})$ est inversible et calculable.*

L'algorithme d'élimination de Gauss fournit une solution effective au problème du calcul de l'inverse d'une matrice. En effet, bien qu'il fasse *a priori* intervenir des tests d'égalité à zéro, on se convainc facilement que dans le cas d'une matrice de rang maximal, il existe toujours n pivots dont on peut démontrer la non-nullité à l'aide d'approximations suffisamment précises. Par contre, on notera que calculer le rang d'une matrice ne fait pas partie des problèmes calculables ; par exemple, étant donnée la matrice

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \varepsilon & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

où ε est un nombre réel calculable qui représente 0, le mieux que l'on puisse décider est l'encadrement $1 \leqslant \operatorname{rang}(A) \leqslant 2$.

D'autres exemples de quantités calculables sont les racines de polynômes et les valeurs propres de matrices (on notera que ces deux quantités varient de façon continue avec les coefficients).

Théorème 4.3.8. *Si les coefficients d'un polynôme $f \in \mathbb{C}[x]$ sont calculables et que le coefficient dominant de f ne s'annule pas, alors il existe un entier p_0 tel que pour tout $p \geqslant p_0$, on puisse calculer une liste de disques deux à deux disjoints de rayon 2^{-p} ayant la propriété suivante : toute racine de f appartient à l'un de ces disques et, réciproquement, chacun de ces disques contient au moins une racine de f.*

La multiplicité d'une racine (ou, de même, la multiplicité d'une valeur propre) n'est pas une fonction calculable en général. Informellement, ceci est dû au fait que la détermination d'une multiplicité requiert des tests d'égalité. De manière plus formelle, on pourra remarquer que le nombre de racines distinctes de $f(x) = x(x + \varepsilon)$ est une fonction discontinue de ε, et conclure par le théorème 4.3.6. Le mieux que l'on puisse faire dans le cas de racines multiples est de démontrer (à l'aide du théorème de Rouché) qu'un disque suffisamment petit contient exactement m racines *comptées avec multiplicité*. De même, il est impossible de décider si un polynôme admet des racines réelles de multiplicité $m > 1$ puisque, à nouveau, le nombre de racines réelles de $f(x) = x^2 + \varepsilon$ est une fonction discontinue de ε.

Il existe de nombreux algorithmes classiques de calcul de racines ou de valeurs propres, ceux-ci venant toutefois rarement avec une preuve de correction. En pratique, les meilleures méthodes rigoureuses consistent généralement à calculer des racines approchées à l'aide d'heuristiques numériques puis de valider *a posteriori* le résultat du calcul par des méthodes spécifiques.

4.3.4 Décidabilité pour certains ensembles de nombres

Dans certaines circonstances particulières, on peut démontrer l'égalité de deux expressions par calcul direct. C'est le cas, par exemple, lorsque l'on se restreint aux nombres algébriques. En effet, étant donné un polynôme $f \in \mathbb{Z}[x]$, on peut écrire des bornes explicites (qui s'expriment en fonction du degré et de la taille des coefficients de f) sur la distance entre deux racines quelconques de f. Ainsi, étant donnés deux nombres algébriques α et β, il est possible de calculer *a priori* un nombre $\varepsilon > 0$ explicite pour lequel l'inégalité $|\alpha - \beta| < \varepsilon$ implique $\alpha = \beta$.

Au delà de $\overline{\mathbb{Q}}$, il est difficile d'exhiber d'autres sous-ensembles de $\mathbb{R}$ pour lesquels le test d'égalité est décidable ; au fil des années, il n'y a eu que peu de succès notables dans cette direction. Un cas prometteur, malgré tout, est celui des *nombres élémentaires*, définis comme les nombres qui peuvent être représentés par des expressions symboliques ne faisant intervenir que des nombres algébriques, des fonctions algébriques et des

fonctions élémentaires (exp, log, les fonctions trigonométriques et leurs inverses).

Théorème 4.3.9 (RICHARDSON et FITCH)**.** *L'égalité entre nombres élémentaires est décidable si la conjecture de* SCHANUEL *est vraie [22].*

La conjecture de SCHANUEL est un énoncé de transcendance qui prédit *grosso modo* qu'il n'existe pas de relations algébriques inattendues entre nombres élémentaires ; par exemple, elle implique que $\pi + e$ doit être transcendant. La conjecture de SCHANUEL est considérée par la communauté comme un problème difficile. Toutefois, RICHARDSON et FITCH ont réussi à écrire un semi-algorithme décidant l'égalité de deux nombres élémentaires ; celui-ci fonctionne systématiquement, à moins qu'il ne tombe, durant son exécution, sur un contre-exemple à la conjecture de SCHANUEL, auquel cas le programme boucle indéfiniment.

En lien étroit avec le théorème de RICHARDSON et FITCH, on notera que le problème de décision suivant concernant les fonctions élémentaires n'est pas décidable.

Théorème 4.3.10 (RICHARDSON)**.** *Il n'existe pas d'algorithme qui décide, en général, si une fonction $f(x)$ représentée par une formule symbolique ne faisant intervenir que des nombres rationnels, π, $\log(2)$, e^x et $\sin(x)$ et $|x|$ s'annule partout.*

L'un des candidats les plus prometteurs à être une extension transcendante effective de $\overline{\mathbb{Q}}$ est l'anneau des périodes [30]. Par définition, une *période* est un nombre complexe qui peut s'écrire sous la forme d'une intégrale $\int_A f$ où f est une fonction algébrique en n variables et A est un sous-ensemble de $\mathbb{R}^n$ défini par des inégalités algébriques (dans lesquelles toutes les constantes qui apparaissent sont aussi des nombres algébriques). Par exemple, $\pi = 4\int_0^1 \sqrt{1-x^2}dx$ et $\log(2) = \int_1^2 x^{-1}dx$ sont des périodes.

Il a été démontré que les périodes sont calculables dans le sens de la définition 4.3.1 [23]. La question de la décidabilité du test d'égalité reste toutefois un problème ouvert :

Conjecture 4.3.11 (KONTSEVICH-ZAGIER)**.** *L'égalité entre périodes est décidable. Concrètement, étant données deux intégrales $\int_A f$ et $\int_B g$ représentant la même période, il existe un procédé algorithmique qui permet de transformer la première en la seconde en lui appliquant une suite finie de transformations simples (changement de variables, formule de* STOKES*).*

4.4 Nombres réels approchés

Dans cette partie, nous abordons les principes de base du calcul numérique fiable, voire prouvé, sur machine. Nous discuterons également du coût des algorithmes mis en jeu.

L'idée fondamentale qui sous-tend tout le calcul numérique consiste à remplacer un nombre réel x (possiblement très difficile à décrire) par une approximation $\widehat{x} = x + \varepsilon$ qui est un nombre rationnel facile à manipuler. L'erreur ε, quant à elle, reste inconnue mais on saura généralement la borner ou, du moins, l'estimer. Il est souvent commode de séparer les sources d'erreurs en deux catégories :

— les erreurs d'arrondis, conséquence des opérations arithmétiques sous-jacentes qui s'effectuent à précision finie ;

— les erreurs de troncation ou de discrétisation qui apparaissent par exemple lorsque l'on remplace une somme infinie $\sum_{n=0}^{\infty} a_n$ par l'une de ses sommes partielles $\sum_{n=0}^{N} a_n$, ou lorsque l'on remplace la solution d'une équation différentielle par une approximation calculée par une méthode de discrétisation avec un pas $h > 0$.

L'étude de la manière dont les erreurs apparaissent et se propagent au fil des calculs est, bien entendu, un sujet de première importance mais nous ne l'évoquerons que brièvement dans la suite de ce cours. En particulier, nous n'étudierons pas les concepts de stabilité directe et stabilité inverse, ni la théorie du conditionnement, ni les détails des analyses de précision dans le cadre de l'arithmétique à virgule flottante. Le lecteur intéressé pourra trouver des discussions approfondies sur ces sujets dans n'importe quel ouvrage d'analyse numérique.

4.4.1 L'arithmétique à virgule flottante

Pour des raisons d'efficacité, la plupart des implémentations disponibles utilisent des approximations de la forme $\widehat{x} = a \cdot 2^b$ avec $a, b \in \mathbb{Z}$. Un nombre de cette forme est appelé un *nombre à virgule flottante binaire* (ou un *nombre dyadique*). Les entiers a et b sont respectivement appelés la *mantisse* et l'*exposant* de $\widehat{x}$. Si on se restreint aux a tels que $|a| < 2^p$, alors $\widehat{x}$ est appelé un *nombre à virgule flottante sur p bits* [8]. Pour certaines applications dans lesquelles les quantités considérées sont toutes du même ordre de grandeur,

8. Souvent, $\widehat{x}$ est plutôt choisi dans $\pm[0{,}5; 1[$, ou dans $[0{,}5; 1[$ et un bit supplémentaire est utilisé pour encoder le signe. Dans ce texte, nous omettrons volontairement ce type de considérations de même que les valeurs particulières *NaN* (Not a Number) et ∞ et les problèmes de dépassement de capacité (*overflow* et *underflow*).

l'*arithmétique à virgule fixe* (correspondant au cas où b est fixé, typiquement égal à $-p/2$) est parfois préférée.

Effectuer des opérations sur les nombres à virgule flottante implique généralement de faire des arrondis afin de préserver les conditions $a, b \in \mathbb{Z}$ et $|a| < 2^{-p}$. La règle d'or consiste à d'imposer que chaque arrondi n'introduise qu'une erreur relative $|x - \widehat{x}|/|x|$ de l'ordre de 2^{-p}. Pour un calcul impliquant de nombreuses opérations élémentaires, la précision p doit être ajustée en fonction de la précision finale souhaitée, en tenant compte des pertes de précision s'accumulant à chaque étape du calcul.

Les types *binary32* et *binary64* définis dans le standard IEEE 754 (qui correspondent respectivement à $p = 24$ et $p = 53$) sont aujourd'hui implémentés dans la majorité des processeurs et, dans la pratique, sont pratiquement devenus synonymes d'arithmétique à virgule flottante. Le calcul à plus haute précision, quant à lui, doit être implémenté au niveau logiciel. Les implémentations les plus répandues sont la *quadruple précision* ($p = 106$), émulée par des paires de *binary64*, et la *précision arbitraire* (où p n'est limité que par la mémoire disponible). L'inconvénient principal de l'arithmétique en précision arbitraire est sa lenteur ; en comparaison de l'arithmétique flottante qui existe nativement dans les processeurs, elle peut aller entre 10 et 1000 fois moins vite.

SageMath dispose de plusieurs fonctionnalités pour travailler avec des réels en précision arbitraire : le constructeur `RealField` (qui s'appuie sur la libraire MPFR), la libraire `mpmath` et le logiciel `Pari/GP`. Voici un exemple simple avec `mpmath` calculant $\pi = 2\int_{-1}^{1} \sqrt{1-x^2}dx$ (la ligne `mp.dps = 100` fixe la précision de calcul à 100 chiffres décimaux, ce qui correspond à $p = 336$ bits) :

```
sage: from mpmath import mp
sage: mp.dps = 100
sage: print(2 * mp.quad(lambda x: mp.sqrt(1-x**2), [-1,1]))
3.14159265358979323846264338327950288419716939937510
58209749445923078164062862089986280348253421170679
```

Faisons une remarque à propos de la pertinence d'un calcul à si haute précision. En réalité, les applications mathématiques pour lesquelles on a besoin d'une précision $p > 10^3$ ne sont pas anecdotiques. On peut citer, par exemple :

— le calcul des solutions en temps long de systèmes chaotiques,
— le calcul de termes éloignés d'une suite récurrente,
— l'évaluation de séries entières à signes alternés près du bord du disque de convergence,

— le calcul de n'importe quelle petite quantité qui, pour des raisons algorithmiques, doit être écrite comme la différence de deux grandes quantités,
— la démonstration d'inégalités entre deux nombres très proches,
— la reconnaissance (heuristique ou prouvée) de valeurs discrètes ou de formules exactes à partir de données numériques approchées.

Au delà des erreurs d'arrondis et de troncature, une troisième source d'erreurs en calcul scientifique est l'incertitude sur les entrées lorsque celles-ci proviennent de mesures physiques ou d'expériences statistiques. En réalité, il y a assez peu de contextes en physique ou en ingénierie où cela a du sens de donner un résultat avec plus de 7 chiffres significatifs (ou, au pire, disons 16 chiffres). Malgré cela, il y a beaucoup de situations où il est important de mener les calculs intermédiaires avec plus de précision en raison des erreurs numériques qui peuvent apparaître puis être amplifiées au gré des algorithmes utilisés. En général, au plus les données sont nombreuses et les calculs sont longs, au plus la précision numérique doit être augmentée [9].

4.4.2 Propagation des erreurs et arithmétique d'intervalles

On peut majorer (ou estimer) l'erreur totale commise lors d'un calcul complexe en majorant (ou en estimant) les erreurs engendrées par chaque opération élémentaire et en calculant des bornes (ou des estimations) sur la propagation de celles-ci lors des opérations futures. Au delà des algorithmes excessivement simples, ce travail est souvent fastidieux à faire à la main. Une solution alternative consiste à propager les erreurs automatiquement en utilisant l'arithmétique d'intervalles [14].

Le principe à la base de toute l'arithmétique d'intervalles est d'approcher un nombre réel x par un sous-ensemble X de $\mathbb{R}$ contenant x. Un tel ensemble X est parfois appelé une *inclusion*. Il doit être aisément représentable sur machine.

Définition 4.4.1. *Soif $f : A \to B$ une fonction. Une* fonction d'inclusion *de f est une fonction $F : \mathcal{P}(A) \to \mathcal{P}(B)$ qui envoie un sous-ensemble de A sur un sous-ensemble de B de façon à ce que $f(x) \in F(X)$ pour tout sous-ensemble X de*

9. Il y a cependant des exceptions notables à ce principe : par exemple, il a été observé que certains réseaux de neurones fonctionnent déjà bien avec un encodage de $\widehat{x}$ sur 16 bits (ce qui correspond peu ou prou à $p = 10$ ou $p = 8$), voire sur 8, 4, 2 et même 1 seul bit. La recherche est actuellement active sur les modèles à précision mixte (pas seulement dans le cadre des réseaux de neurones) où l'essentiel des calculs est conduit à faible précision, avec une augmentation possible ponctuelle de la précision pour certaines parties critiques [4].

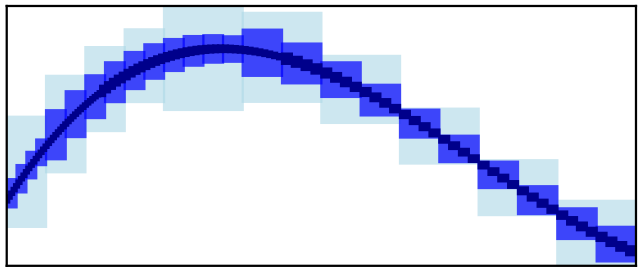


FIGURE 4.2 – *Illustration de la propriété de préservation de continuité d'une fonction d'inclusion d'une fonction à variable réelle : les intervalles $F(X)$ diminuent de taille avec X, tout en restant concentrés autour du graphe de la fonction $y = f(x)$. Dans cet exemple, on observe que les inclusions $F(X)$ ne sont pas les plus fines possibles.*

A et tout élément $x \in X$. En d'autres termes, on doit avoir $f(X) \subset F(X)$ pour tout $X \subset A$.

La règle de la définition précédente est parfois appelée le *principe d'inclusion*. Avant de poursuivre, notons que ce principe s'applique à des objets très généraux, et pas uniquement aux nombres réels : dans la définition 4.4.1, il n'est pas nécessaire de supposer que A et B sont des sous-ensembles de $\mathbb{R}$. Il est évident, par ailleurs, que les fonctions d'inclusion peuvent être composées dès lors que les fonctions sous-jacentes sont, elles-mêmes, composables. Notons encore qu'il n'est pas requis que les inclusions soient fines ; par exemple, la meilleure inclusion possible pour la fonction $\sin(x)$ sur $X = [0, \pi]$ est $[0, 1]$ mais une fonction d'inclusion qui retournerait $[-2, 2]$ serait également tout à fait acceptable. En pratique, il y aura souvent un compromis à trouver entre la qualité de la fonction d'inclusion et le coût calculatoire.

Une propriété raisonnable que l'on impose généralement sur les fonctions d'inclusion est la préservation de la continuité : si f est continue en un point $x \in \mathbb{R}$, on demande à ce que pour toute suite $(X_n)_{n \geqslant 0}$ d'intervalles contenant x telle que $\text{taille}(X_n) \to 0$ quand $n \to \infty$, on ait aussi $\text{taille}(F(X_n)) \to 0$ [10] (voir figure 4.2 pour une illustration de ce principe). Cette propriété résonne comme un analogue de la définition 4.3.2 dans le contexte de l'arithmétique d'intervalles. Elle est en outre souvent décisive pour démontrer la convergence d'algorithmes.

Il existe plusieurs possibilités pour représenter des nombres réels

10. Si l'arithmétique en virgule flottante est utilisée pour représenter les intervalles, cet axiome nécessite que la précision utilisée augmente rapidement avec n.

par des intervalles. L'une d'elles consiste à utiliser des intervalles $[a, b]$ où les extrémités a et b sont des nombres à virgule flottante et de représenter ceux-ci par le couple (a, b). C'est ainsi qu'est implémenté le corps `RealIntervalField` en SageMath, qui repose sur la bibliothèque MPFI. Une autre possibilité est d'utiliser des intervalles de la forme $[m \pm r] = [m - r, m + r]$ (appelés parfois des *boules arithmétiques*). Cette implémentation est disponible en SageMath *via* le constructeur `RealBallField` qui fait appel au logiciel Arb en coulisse. De manière générale, la première représentation paraît plus naturelle pour représenter des *subdivisions de l'espace* alors que les boules sont plus efficaces pour représenter des *nombres réels isolés*. Ceci dit, dans de nombreux cas, ces représentations sont simplement interchangeables.

Tester l'égalité de nombres réels représentés par des intervalles n'a, en général, pas de sens, hormis dans le cas très particulier où les intervalles en question sont tous réduits à un point. Il est cependant possible de tester si les intervalles sont disjoints (auquel cas, on peut conclure à l'inégalité des nombres réels correspondants) ou s'ils se chevauchent (auquel cas, l'égalité est possible). À noter que le résultat de la soustraction de deux intervalles qui représentent le même nombre réel est un intervalle contenant zéro :

```
sage: R = RealBallField(53)
sage: x = R(2); sqrt(x)/2 - 1/sqrt(x)
[+/- 4.45e-16]
sage: R = RealBallField(333)
sage: x = R(2); sqrt(x)/2 - 1/sqrt(x)
[+/- 1.72e-100]
```

Il est facile d'émuler les réels et les fonctions calculables (voir §4.3.1) par de l'arithmétique d'intervalles à précision arbitraire, simplement en écrivant une boucle qui effectue des évaluations de plus en plus précises jusqu'à atteindre la précision demandée :

```
sage: def f(p):
....:   prec = 10
....:   while True:
....:     R = RealBallField(prec)
....:     x = (R(163).sqrt()*R.pi()).exp()-640320**3-744
....:     print("prec = %s gave %s" % (prec, x))
....:     if x.accuracy() > p:   # relative accuracy
....:         break
....:     prec *= 2

sage: f(30)
prec = 10 gave [+/- 4.99e+16]
```

```
prec = 20 gave [+/- 6.83e+13]
prec = 40 gave [+/- 4.71e+7]
prec = 80 gave [+/- 4.76e-5]
prec = 160 gave [-7.499274028018143e-13 +/- 5.65e-29]
```

Remarquons qu'un tel programme peut boucler indéfiniment lorsque la fonction que l'on cherche a évaluer présente une discontinuité ou lorsque l'on souhaite atteindre une précision *relative* donnée pour un résultat exact égal à 0.

4.4.3 Sur la dépendance des erreurs

L'arithmétique d'intervalles assure un suivi rigoureux des erreurs mais, en contrepartie, ne le fait pas toujours de façon optimale. Pour illustrer cette affirmation, considérons le problème de calculer une somme $S_N = \sum_{k=1}^{N} x_k$, connaissant des approximations $\widehat{x}_k$ de x_k avec $\varepsilon_k = \widehat{x}_k - x_k \leqslant 2^{-p}$. Dans ces conditions, quelle est l'erreur maximale $|S_N - \sum_{k=1}^{N} \widehat{x}_k|$?

Dans le *pire cas*, la meilleure borne que l'on puisse donner est $N \cdot 2^{-p}$. Et, de fait, si nous effectuons le calcul dans le modèle de l'arithmétique d'intervalles, nous obtiendrons précisément cette borne (plus, éventuellement, des termes supplémentaires provenant d'erreurs d'arrondis si la somme des $\widehat{x}_k$ est elle-même calculée par de l'arithmétique approchée).

Toutefois, la borne du pire cas est parfois trop pessimiste. Typiquement, dans le cas où les erreurs ε_k sont réparties uniformément et indépendamment dans un intervalle centré en 0, le théorème de la limite centrale (ou, au choix, la théorie des marches aléatoires) nous apprend que l'erreur *moyenne* attendue est de l'ordre de $O(\sqrt{N}) \cdot 2^{-p}$. Ce type d'heuristique fournit des estimés qui sont souvent plus réalistes qu'une analyse dans le pire cas. Au contraire, l'arithmétique d'intervalles ne prend aucunement en compte l'indépendance des erreurs et calcule systématiquement l'estimation la plus pessimiste du pire cas [11].

Un cas extrême est celui où les erreurs sont entièrement corrélées et s'annulent. Par exemple, si $N = 2$ et $\varepsilon_1 = -\varepsilon_2$, on a $x_1 + x_2 = \widehat{x}_1 + \widehat{x}_2$ *exactement* mais l'arithmétique d'intervalles continue d'afficher la borne d'erreur pessimiste $2 \cdot 2^{-p}$. Quand ce phénomène se produit de façon répétée dans une boucle, on peut être amené à observer des bornes d'erreur qui grandissent *exponentiellement* vite. L'exemple le plus simple de ce phénomène apparaît lorsque l'on soustrait une quantité à elle même plusieurs fois d'affilée :

11. Exercice : tester quelques calculs avec, d'une part, l'arithmétique en virgule flottante et, d'autre part, l'arithmétique d'intervalles et observer la différence en pratique.

```
sage: x = RealBallField(53)(2).sqrt()
sage: x = x-x; print(x)
[+/- 4.45e-16]
sage: x = x-x; print(x)
[+/- 8.89e-16]
sage: x = x-x; print(x)
[+/- 1.78e-15]
...
[+/- 2.10e+6]
sage: x = x-x; print(x)
[+/- 4.20e+6]
```

Cette explosion exponentielle signifie, réciproquement, qu'il est nécessaire d'augmenter la précision initiale de manière exponentielle avec le nombre d'itérations si l'on désire atteindre une précision finale fixée à l'avance. Le calcul à haute précision est parfois inévitable, par exemple lorsque l'on itère des systèmes dynamiques chaotiques très sensibles aux conditions initiales, mais il peut aussi ne refléter aucune réalité intrinsèque. Ainsi, lorsqu'on est amené à utiliser l'arithmétique d'intervalles, il est souvent souhaitable de réécrire les formules évaluées et de reconcevoir les algorithmes utilisés afin de minimiser autant que possible les dépendances entre les erreurs.

Une situation classique où les problèmes de dépendance acquièrent une importance considérable est la résolution de systèmes linéaires. En effet, lorsque l'on exécute l'algorithme d'élimination de GAUSS en arithmétique d'intervalles, on observe généralement des bornes d'erreur qui grandissent exponentiellement vite avec la taille n de la matrice d'entrée ; on a ainsi besoin d'une précision d'au moins $O(n)$ chiffres significatifs. Le calcul à haute précision est, de fait, inévitable pour les matrices mal conditionnées. Par contre, pour les matrices bien conditionnées, l'arithmétique d'intervalles conduit à la même explosion de complexité alors que le même algorithme exécuté en arithmétique en virgule flottante est parfaitement stable.

À cause de cela, la meilleure façon de résoudre un système linéaire en arithmétique d'intervalles est de commencer par calculer une solution approchée en arithmétique à virgule flottante puis, dans un second temps, d'utiliser des méthodes de certification *a posteriori* pour obtenir des bornes d'erreur rigoureuses [12].

12. Cette observation selon laquelle la propagation directe des bornes d'erreur a tendance à surestimer exponentiellement vite les erreurs réelles (aussi bien pour l'algorithme d'élimination de GAUSS que pour d'autres algorithmes) est considérée comme l'un des

4.4.4 Analyse de complexité en arithmétique approchée

L'archétype des algorithmes rapides impliquant des nombres réels est la transformée de FOURIER rapide (FFT) qui met en œuvre une stratégie de type diviser pour régner pour calculer la transformée de FOURIER discrète (DFT) d'un vecteur de nombres complexes

$$X_k = \sum_{j=0}^{n-1} x_j e^{-2\pi ikj/n}, \quad k = 0, 1, \ldots, n-1$$

en $O(n \log n)$ opérations au lieu de $O(n^2)$ pour un algorithme naïf. L'une des principales applications de la FFT est la multiplication rapide des polynômes. On peut multiplier des polynômes de degré strictement inférieur à n à coefficients réels ou complexes en seulement $O(n \log n)$ opérations (au lieu de $O(n^2)$ pour un algorithme naïf) comme ceci : on évalue les polynômes d'entrée en $2n$ racines de l'unité, on multiplie ces évaluations point par point et, enfin, on interpole pour retrouver ainsi les $2n$ coefficients du polynômes produit. Les étapes d'évaluation multi-point et d'interpolation sont simplement des DFT.

Toutefois, il n'est généralement pas suffisant de compter les opérations arithmétiques (complexité algébrique) pour analyser l'efficacité des algorithmes numériques. En effet, ce décompte sommaire ne tient pas compte de la précision utilisée pour représenter les nombres et il peut très bien arriver qu'un algorithme effectuant moins d'opérations ait besoin en contrepartie d'augmenter sensiblement la précision de calcul, induisant finalement un coût total supérieur. Pour cette raison, les algorithmes « rapides » ne sont pas toujours les meilleurs. Souvent, il est plus réaliste de se placer dans le modèle de complexité binaire où l'on ne compte pas les opérations arithmétiques mais les opérations sur les bits.

Clairement, on peut additionner et soustraire des nombres entiers ou des nombres à virgule flottante de p bits en $O(p)$ opérations binaires. La multiplication, quant à elle, a un coût quasi-linéaire :

Théorème 4.4.2 (HARVEY–van der HOEVEN)**.** *Il est possible de multiplier deux entiers de p bits en $O(p \log p)$ opérations binaires.*

De façon surprenante, ce résultat n'a été démontré qu'en 2019 [27] [13]. L'idée fondamentale derrière la multiplication rapide des entiers est l'obser-

phénomènes les plus importants de l'analyse numérique. Elle a été la principale découverte de J. H. WILKINSON et a conduit au développement de l'analyse d'erreur par stabilité inverse dans les années 1960. WILKINSON a reçu le prix TURING en 1970.

13. En fait, au moment où j'écris ces notes, l'article de HARVEY et van der HOEVEN n'a pas été complètement arbitré. Croisons les doigts !

vation que l'entier 325 peut être vu comme l'évaluation en $x = 10$ du polynôme $3x^2 + 2x + 5$. À partir de là, avec un peu de travail, il est possible de ramener de la multiplication des entiers à celle des polynômes et d'utiliser des méthodes de type FFT. Le premier algorithme de multiplication rapide basé sur la FFT a été publié en 1971 par SCHÖNHAGE et STRASSEN [14] ; sa complexité était de $O(p \log p \log\log p)$ opérations binaires [26]. Le facteur parasite $\log\log p$ provient de ce que les opérations internes à l'algorithme de FFT n'ont pas, au moins en apparence, un coût de $O(1)$: en effet, celles-ci reposent elles-mêmes sur la multiplication d'entiers [15] et la précision totale dans ce procédé récursif complexe croît avec p.

L'algorithme de HARVEY et van der HOEVEN utilise plusieurs techniques ingénieuses pour effacer le facteur $\log\log p$. Parmi elles, on peut citer le passage d'une FFT unidimensionnelle à une FFT multidimensionnelle et l'utilisation d'un ré-échantillonage approché dans le but d'ajuster la taille des vecteurs. Toutes les étapes de l'algorithme doivent être analysées avec précaution afin de prendre en compte simultanément les erreurs d'approximation, les pertes de précision et, bien sûr, la complexité binaire.

En pratique, la différence entre $O(p \log p \log\log p)$ et $O(p \log p)$ n'est visible que pour des p de taille astronomique ; dans les implémentations, une mauvaise constante dans le $O(\cdot)$ importe souvent plus. Dans la bibliothèque GMP *bignum*, la FFT (précisément, l'algorithme de SCHÖNHAGE-STRASSEN) n'est utilisé que pour des nombres de plus de 100 000 chiffres décimaux. Pour les entiers de taille plus petite, il est plus rapide d'utiliser l'algorithme naïf (jusqu'à, à peu près, 1000 chiffres) ou l'algorithme de KARATSUBA et ses généralisations (de 1000 chiffres à 100 000 chiffres) [19]. Les calculs impliquant des nombres avec plusieurs dizaines de milliers de chiffres ne sont pas monnaie courante, mais ils peuvent apparaître, de temps en temps, en théorie algorithmique des nombres. Il y a également des situations où il est rentable de remplacer un grand nombre d'opérations sur des petits entiers par un nombre restreint d'opérations sur des entiers gigantesques.

Certaines autres opérations algébriques sur les entiers ou les nombres à virgule flottante reposent sur la multiplication rapide des entiers [3].

Théorème 4.4.3. *Il est possible de calculer le quotient $\lfloor a/b \rfloor$ d'un entier a de $2p$ bit par un entier b de p bits, ou la racine carrée $\lfloor \sqrt{a} \rfloor$ d'un entier a de $2p$ bits, ou*

14. En réalité, SCHÖNHAGE et STRASSEN ont publié deux algorithmes, le premier utilisant des nombres complexes et le second n'utilisant que de l'arithmétique exacte sur les entiers. C'est cette deuxième version qui est appelée couramment l'algorithme de SCHÖNHAGE-STRASSEN et à laquelle nous faisons référence dans la suite du texte.

15. Ces entiers sont plus petits, ce qui permet une approche récursive.

encore une approximation à 2^{-p} près de quotients de racines carrées de nombres à virgule flottante, pour un coût de $O(p \log p)$ opérations binaires.

L'idée derrière ce théorème est de reformuler la question en un problème de recherche de point fixe d'une équation algébrique que l'on résout finalement par une méthode de NEWTON en prenant garde à ce que les formules itératives obtenues ne fassent intervenir que des additions, des soustractions et des multiplications. Par exemple, pour calculer $1/b$, on peut résoudre l'équation $x - 1/b = 0$, ce qui conduit *in fine* à l'itération $x_{k+1} = 2x_k - bx_k^2$. À chaque étape, le nombre de décimales correctes est, plus ou moins, doublé. Ainsi, l'algorithme ne nécessite au total que $O(\log p)$ itérations pour une précision de p bits. Nous encourageons la lectrice à écrire proprement la démonstration du théorème 4.4.3 et, en particulier, à se convaincre que, malgré les $O(\log p)$ itérations du schéma de NEWTON, la complexité totale n'augmente pas jusqu'à $O(p \log^2 p)$.

Théorème 4.4.4. *Étant donné un nombre complexe en virgule flottante z, il est possible de calculer une approximation rationnelle de e^z et de* $\log(z)$ *(pour la détermination principale du* log*) à 2^{-p} près pour un coût de $O(p \log^2 p)$ opérations binaires.*

Ce résultat de complexité s'étend plus généralement à l'évaluation des fonctions élémentaires (c'est-à-dire des fonctions s'écrivant comme une composée d'opérations arithmétiques, d'exponentielles, de logarithmes, de fonctions trigonométriques et de leurs inverses), en dehors des points singuliers du domaine de définition des fonctions mises en jeu [16].

L'algorithme sous-jacent au théorème 4.4.4 s'appuie sur l'*itération de la moyenne arithmético-géométrique* :

$$a_{k+1}, b_{k+1} = \frac{a_k + b_k}{2}, \sqrt{a_k b_k}. \tag{4.2}$$

Pour n'importe quelles valeurs initiales a_0 et b_0 strictement positives, on peut démontrer que les suites (a_k) et (b_k) définies ci-dessus convergent vers une limite commune $a_\infty = b_\infty$ qui, pour a_0 et b_0 bien choisis, est reliée à $\log(z)$. À l'instar de la méthode de NEWTON, la convergence est quadratique, dans le sens où le nombre de chiffres corrects double à chaque itération ; ainsi, pour obtenir une précision de p bits, $O(\log p)$ itérations suffisent et la complexité annoncée en découle. Il est à noter que cette méthode, qui permet d'évaluer les fonctions élémentaires pour un coût de

16. Il est important de se rendre compte, malgré tout, que ce type de bornes de complexité n'est pas uniforme vis-à-vis de z. À cet effet, le lecteur intéressé pourra chercher à estimer comment le coût du calcul de e^z ou de $\tan(z)$ varie en fonction de z.

$O(p \log^2 p)$ opérations binaires, passe par des évaluations de fonctions non élémentaires (des intégrales elliptiques) qui sont elles-mêmes calculées par une variante de (4.2) adaptée aux nombres complexes.

De manière plus pédestre, on sait atteindre une complexité à peine moins bonne, à savoir $O(p \log^3 p)$, en utilisant uniquement des équations fonctionnelles classiques (du type $e^{x+y} = e^x e^y$) et des évaluations de séries de TAYLOR tronquées. Ces dernières doivent toutefois être réalisées avec précaution par un algorithme de type diviser pour régner appelé le *scindage binaire*.

L'idée du scindage binaire peut être illustrée simplement par le calcul de la factorielle : $N! = 1 \cdot 2 \cdot 3 \cdot \cdots \cdot N$. En effet, remarquons qu'un calcul itératif naïf a un coût de $N^{2+o(1)}$ opérations binaires, alors qu'une méthode de type diviser pour régner conduit à une complexité moindre, de l'ordre de $N^{1+o(1)}$. Cette technique s'étend au calcul de produits de matrices $M_N M_{N-1} \cdots M_0$ lorsque les coefficients des M_i sont des petits nombres rationnels. Or, il se trouve que la série de TAYLOR de e^x peut être décrite à l'aide de tels produits matriciels. Plus généralement, cette même méthode fonctionne pour l'évaluation des fonctions D-finies (c'est-à-dire des fonctions solutions d'une équation différentielle linéaire ordinaires à coefficients polynomiaux), ce qui inclut en particulier la fonction d'erreur $\mathrm{erf}(x)$ et les fonctions de BESSEL.

4.5 Dérivation et intégration

Dans la dernière partie de ce chapitre, nous nous proposons d'évoquer quelques autres questions sur les fonctions réelles et complexes, allant au-delà de la simple évaluation en un point donné. Plus précisément, nous nous focalisons sur deux opérations fondamentales de l'analyse : la dérivation et l'intégration.

La difficulté de chacun de ces problèmes dépend de la manière dont les fonctions sont représentées ainsi que de la forme du résultat attendue. Nous allons traiter trois cas : (1) les fonctions sont données par des expressions symboliques, (2) les fonctions sont données par des programmes en « boîte noire » et (3) les fonctions sont représentées par des approximations.

Par souci de simplicité, nous considèrerons uniquement le cas de fonctions d'une variable réelle ou complexe (en se tenant à l'écart de la boîte de Pandore qu'est la dérivation et l'intégration de fonctions à plusieurs variables).

4.5.1 Calcul symbolique

Supposons que nous soit donnée une fonction représentée par une expression symbolique (du type $f(x) = e^{x^2}/x$) qui puisse être évaluée pour une valeur numérique donnée de x en parcourant l'arbre qui encode l'expression symbolique et en appliquant successivement les opérations rencontrées (ici $x^2, e^x, /$).

Il est facile d'écrire une expression symbolique pour la dérivée (ici $f'(x) = e^{x^2}(2x^2 - 1)/x^2$) en utilisant les règles de dérivation de façon répétée. Cependant, l'expression que l'on obtient ainsi pour $f'(x)$ peut croître rapidement en taille. Pour évaluer $f'(x)$ en un point x donné, il est ainsi souvent plus efficace de parcourir l'arbre de f et d'évaluer simultanément les valeurs de $f(x)$ et $f'(x)$ en appliquant les opérations rencontrées à des séries tronquées de la forme $a + b\varepsilon + O(\varepsilon^2)$; ce faisant, le coefficient b coïncide avec la valeur de la dérivée. Cette méthode est connue sous le nom de *différentiation automatique* (AD) et peut être facilement généralisée au cas des dérivées supérieures.

Le calcul de primitives est nettement plus compliqué. En effet, sauf dans des cas très particuliers où l'on peut utiliser des « astuces » comme des intégrations par parties ou des changements de variables, il n'existe malheureusement pas de formules générales d'intégration. Typiquement, la fonction e^{x^2} n'admet pas de primitive qui s'exprime à l'aide de fonctions élémentaires.

Plus précisément, les problèmes que l'on peut résoudre sous forme symbolique dépendent fortement des symboles que l'on s'autorise : si on accepte la fonction $\operatorname{erf}(x)$, alors on peut représenter une primitive de e^{x^2}. Pareillement, si on introduit un symbole pour la fonction $\Gamma(x)$, il n'existe pas de forme close pour la fonction $\Gamma'(x)$ (sauf à introduire encore un nouveau symbole).

Intégration indéfinie

Il existe une méthode symbolique systématique pour calculer des intégrales indéfinies, connue sous le nom d'algorithme de RISCH, qui décide si une fonction élémentaire donnée en entrée admet une primitive qui est, elle-même, élémentaire et, le cas échéant, la détermine. Il existe même des généralisations de l'algorithme de RISCH permettant de traiter certains types de fonctions non élémentaires comme $\operatorname{erf}(x)$. Tous ces algorithmes fonctionnent sur le principe suivant : le problème d'intégration est traduit en un problème algébrique exprimé dans la théorie des corps différentiels.

L'algorithme de RISCH n'est, en réalité, pas un algorithme à proprement parler car il nécessite un test d'égalité pour les expressions symboliques. À vrai dire, les constantes qui interviennent dans l'expression à intégrer sont déjà problématiques : la fonction $f(x) = x + (b-a)e^{x^2}$ possède une primitive élémentaire si et seulement si $a = b$. En outre, l'algorithme de RISCH est extrêmement complexe et n'a jamais été totalement implémenté [17]. De plus, il est coûteux et ne fournit pas nécessairement des résultats sous forme simplifiée. Pour ces raisons, les logiciels de calcul formel traditionnels essaient souvent des heuristiques basées sur la reconnaissance de motifs avant de se lancer dans l'algorithme de RISCH en cas d'échec.

Intégration définie

De manière peut-être surprenante, le calcul symbolique d'une intégrale définie $\int_a^b f(x)dx$ est assez différent du calcul de primitives. Au moins deux raisons à cela peuvent être invoquées :

— Une intégrale définie entre deux points particuliers a et b peut avoir une expression simple, quand bien même l'intégrale indéfinie correspondante n'en aurait pas. Par exemple, $\int_0^\infty e^{-zx^2}dx = \frac{1}{2}(\pi/z)^{1/2}$ pour $z > 0$. Pour les calculer, les logiciels de calcul formel utilisent des méthodes variées qui dépassent le cadre de ce cours.

— La formule $\int_a^b f(x)dx = F(b) - F(a)$ (où F est une primitive de f) s'applique uniquement lorsque f est continue sur $[a, b]$. En général, l'intégration symbolique nécessite donc de localiser les points singuliers de f, ce qui constitue un problème annexe délicat en lui-même. Ce problème est d'ailleurs à l'origine de bugs fréquents dans les systèmes de calcul formel où, parfois même, l'intégrale $\int_a^b f(x)dx$ d'une fonction à valeurs réelles pouvait renvoyer un résultat aberrant comme $1{,}23456 + 3{,}14159i$ à cause d'un mauvais choix de point de branchement. Prendre le temps de comparer le résultat d'une intégration symbolique avec celui d'une méthode numérique est souvent une bonne idée !

Lorsque les méthodes d'intégration indéfinies s'appliquent, elles ont souvent un avantage décisif sur un calcul numérique : elles sont beaucoup moins sensibles aux comportements, possiblement irréguliers, de la fonction à intégrer. Par exemple, évaluer l'intégrale $\int_0^1 \sin(Nx)dx$ avec $N = 1$

17. Actuellement, le logiciel FriCAS se réclame d'avoir l'implémentation la « plus complète » de l'algorithme de RISCH. Le statut exact de cette implémentation est discuté sur `http://fricas-wiki.math.uni.wroc.pl/RischImplementationStatus`.

ou $N = 10^{10}$ revient à peu près au même si l'on utilise une méthode symbolique, là où les algorithmes numériques peinent pour les grandes valeurs de N à cause des oscillations rapides.

4.5.2 Fonctions calculables en boîte noire

Étant donnée une implémentation en boîte noire de $f(x)$ — c'est-à-dire un programme qui évalue numériquement $f(x)$ étant donné x — il existe de nombreuses méthodes numériques pour calculer des approximations de la dérivée de f ou de son intégrale. Les plus simples d'entre elles reposent sur les approximations classiques $\int_a^b f(x)dx \approx h \cdot \sum_n f(a+nh)$ et $f'(x) \approx (f(x+h) - f(x))/h$ pour un certain $h > 0$. Ces approximations convergent vers les valeurs exactes attendues lorsque h tend vers 0 si f est, respectivement, intégrable au sens de RIEMANN ou dérivable.

Afin de majorer l'erreur commise dûe à la discrétisation (ou, de manière équivalente, de choisir convenablement le pas h étant donnée une précision souhaitée 2^{-p}), il est nécessaire de connaître des informations supplémentaires sur la régularité de f comme, typiquement, une borne sur ses dérivées supérieures. *A contrario*, une telle borne ne peut être uniquement déduite des valeurs prises par f en un nombre fini de points isolés : l'intégrale $\int_a^b f(x)dx$ peut varier de façon arbitrairement grande alors que la fonction f ne subit qu'une perturbation très localisée (par exemple, par ajout d'une fonction en escaliers ou d'une fonction infiniment dérivable du type Ne^{-Nx^2}).

De même, à une précision 2^{-p} donnée, la fonction $f(x)$ peut rester indistinguable d'une de ses perturbations dont la dérivée, quant à elle, explose ; par exemple, $f(x) + \varepsilon \sin(x/\varepsilon^2)$ ou $f(x) + \varepsilon H(x)$ où $H(x)$ est une fonction en escaliers avec, disons, $H'(0) = \infty$. Un exemple encore plus pathologique est donné par la fonction de WEIERSTRASS $f(x) = \sum_{n=0}^{\infty} 2^{-n} \cos(3^n \pi x)$ qui est continue et calculable, mais dérivable en aucun point ; évidemment, il n'y a aucun moyen de déduire cette dernière propriété à partir d'un nombre fini de valeurs prises par f.

Étant donnée une implémentation en boîte noire d'une fonction d'inclusion (voir définition 4.4.1) de $f(x)$, il est généralement trivial d'obtenir des bornes rigoureuses sur l'intégrale $\int_a^b f(x)dx$ en utilisant une méthode de subdivision comme illustré par la figure 4.2 (page 131). Cette méthode ne nécessite d'ailleurs pas la continuité de f, elle fonctionne pareillement, par exemple, pour des fonctions continues par morceaux. En particulier, $g(a,b) = \int_a^b f(x)dx$ peut être calculable pour tous a et b (dans le sens de la définition 4.3.2) sans que $f(x)$ soit lui-même calculable pour tout x.

La dérivation, quant à elle, est un problème intrinsèquement mal posé : il est impossible d'évaluer rigoureusement $f'(x)$ à partir d'une implémentation en boîte noire d'une fonction d'inclusion de f, quand même bien on disposerait d'informations additionnelles de régularité.

Le cas des fonctions holomorphes mérite cependant qu'on s'y attarde. En effet, grâce à la formule de CAUCHY (qui permet de ramener le calcul de dérivées à celui d'une intégrale de contour), une implémentation en boîte noire d'une fonction d'inclusion complexe d'une fonction holomorphe f est suffisante pour le calcul de dérivées et d'intégrales de f avec des bornes d'erreur rigoureuses. De surcroît, des algorithmes rapides sont disponibles dans ce cas : alors qu'une méthode sommatoire basée sur une subdivision d'intervalles requiert généralement un nombre exponentiel (en p) d'évaluations pour obtenir un résultat correct à 2^{-p} près, on dispose d'algorithmes spécifiques aux fonctions holomorphes qui ne nécessitent que $O(p)$ évaluations. Ainsi si, par exemple, on sait évaluer $f(z)$ pour un coût de $p^{1+o(1)}$ opérations binaires, on peut évaluer son intégrale avec un complexité de seulement $p^{2+o(1)}$ opérations. Il est à noter que ces méthodes ne s'appliquent que « localement » lorsque le chemin d'intégration est de longueur finie et reste éloigné des singularités ; dans le cas contraire, la tâche est souvent plus difficile.

Exemple : une intégrale épineuse

Les exemples pathologiques pour les algorithmes d'intégration numérique sont nombreux. L'un d'entre eux est l'« intégrale épineuse » [18]

$$\int_0^1 \operatorname{sech}^2(10(x-0{,}2)) + \operatorname{sech}^4(100(x-0{,}4)) + \operatorname{sech}^6(1000(x-0{,}6))\ dx.$$

Voici le résultat que l'on obtient lorsqu'on l'évalue avec la fonction `numerical_integral` de SageMath (qui fait appel au code d'intégration numérique de la bibliothèque GSL) :

```
sage: numerical_integral(lambda x: sech(10*x-2)**2
...         + sech(100*x-40)**4 + sech(1000*x-600)**6, 0, 1)
(0.2097360688339336, 6.166358647858423e-14)
```

Le deuxième nombre affiché est supposé être une estimation de l'erreur commise. Or, bien que celle-ci suggère que les 13 premières décimales sont correctes, en réalité, seulement 2 le sont. La raison en est que la fonction à

18. *Spike integral* en anglais.

intégrer, représentée sur la figure 4.3, a trois pics ; or, les points d'évaluation choisis par la bibliothèque GSL ratent la contribution du pic le plus étroit.

Le code d'intégration numérique en arithmétique d'intervalles de Arb fournit, quant à lui, un résultat rigoureux. De plus, il permet de calculer l'« intégrale épineuse » à haute précision très facilement :

```
sage: f = lambda x, _: (10*x-2).sech()**2 +
...         (100*x-40).sech()**4 + (1000*x-600).sech()**6
sage: ComplexBallField(53).integral(f, 0, 1)
[0.21080273550055 +/- 4.44e-15]
sage: ComplexBallField(333).integral(f, 0, 1)
[0.21080273550054927737564325570572915436090918643678119034
78505058787206131281455002050586892615576 4 +/- 3.67e-99]
```

L'algorithme sous-jacent à ce calcul exploite l'hypothèse que la fonction à intégrer est holomorphe (en dehors des pôles) ; elle utilise l'arithmétique d'intervalles pour confiner de manière contrôlée le chemin d'intégration suffisamment loin des pôles et ainsi obtenir des bornes rigoureuses sur les erreurs commises par les algorithmes d'intégration numérique (ici, une méthode de quadrature de GAUSS avec subdivision), comme illustré par la figure 4.3 [28].

La qualité de l'approximation finale obtenue par l'arithmétique d'intervalles peut être très sensible à l'ordre des opérations effectuées ainsi qu'au choix des fonctions élémentaires appelées. En guise de preuve par l'exemple, le lecteur pourra essayer d'utiliser `cosh(x)**-n` à la place de `sech(x)**n` et remarquer que le calcul devient alors bien plus lent.

4.5.3 Approximants

Enfin, une possibilité pour représenter une fonction à variable réelle ou complexe est de l'approcher par une fonction plus simple, appelée un *approximant*. Parmi les approximants les plus classiques, on peut citer les fonctions polynomiales, les fonctions polynomiales par morceaux, les polynômes trigonométriques et les fractions rationnelles. Chacun des types de fonctions suscités a en outre la propriété agréable qu'ils permettent une dérivation et une intégration exacte, terme à terme.

Un candidat naturel d'approximant d'une fonction est l'une des troncations de sa série de TAYLOR au voisinage d'un point a. Ce choix aboutit à une approximation optimale locale au voisinage de a. Au moins localement, seulement $O(p)$ termes sont nécessaires pour obtenir des approximations correctes à 2^{-p} près ; la complexité binaire d'une opération typique utilisant

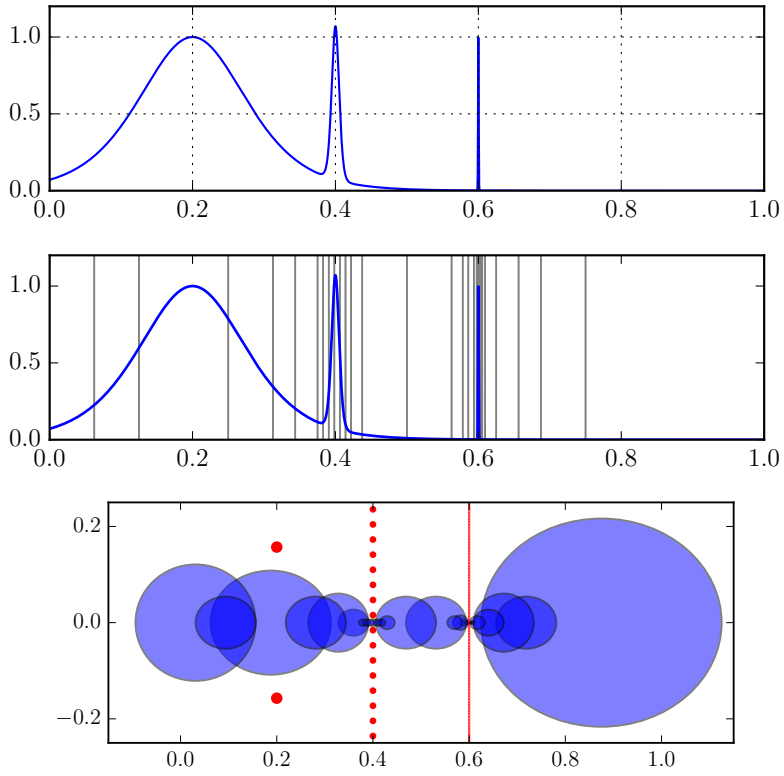


FIGURE 4.3 – *Haut : la « fonction épineuse » à intégrer. Milieu : les subdivisions choisies par le code d'intégration de Arb. Bas : les pôles de la « fonction épineuse » (points rouges) dans le plan complexe et les ellipses choisies par Arb pour confiner le chemin d'intégration loin des pôles et ainsi obtenir des bornes d'erreur rigoureuses.*

cette stratégie est donc de $p^{2+o(1)}$ étant donné que l'arithmétique polynomiale rapide basée sur la FFT a une complexité de $O(n \log n)$ opérations arithmétiques pour des polynômes de degré n.

Une série de TAYLOR tronquée accompagnée d'une borne sur l'erreur correspondante est appelée un *modèle de* TAYLOR [29]. Au delà de l'application évidente à la manipulation de fonctions, les modèles de TAYLOR peuvent être utilisées pour atténuer les problèmes de dépendance en arithmétique d'intervalles (voir §4.4.3) : plutôt que de modéliser une quantité à l'aide d'une constante avec terme d'erreur ε, on utilise une série de TAYLOR tronquée munie d'un terme d'erreur de la forme $C\varepsilon^N$; ce procédé peut rendre visibles certaines dépendances qui se compensent seulement à l'ordre ε^N.

Un autre choix naturel d'approximant est donné par les séries de TCHEBYCHEV tronquées. Ces dernières conduisent à de meilleures approximations uniformes. En outre, on dispose d'algorithmes rapides pour la manipulation de polynômes dans la base de TCHEBYCHEV, tout comme dans la base monomiale standard. Les développements de TCHEBYCHEV constituent le fondement de la bibliothèque *Chebfun* décrite par les auteurs comme « un analogue pour les fonctions de l'arithmétique en virgule flottante » [24]. Plus récemment, les développements de TCHEBYCHEV avec bornes d'erreur rigoureuses ont été considérées comme une alternative viable aux modèles de TAYLOR [25].

Bibliographie

[1] P. ZIMMERMANN et al. *Calcul mathématique avec Sage.* `http://sagebook.gforge.inria.fr/`, 2013.

[2] J. von zur GATHEN and J. GERHARD. *Modern Computer Algebra*. Cambridge University Press, 2013.

[3] R. P. BRENT and P. ZIMMERMANN. *Modern Computer Arithmetic*, Cambridge University Press, 2010. `http://www.loria.fr/~zimmerma/mca/mca-cup-0.5.7.pdf`.

[4] N. HIGHAM. The Rise of Multiprecision Computations, 2017. `https://www.maths.manchester.ac.uk/~higham/talks/samsi17.pdf`

[5] D. V. CHUDNOVSKY and G. V. CHUDNOVSKY. Computer algebra in the service of mathematical physics and number theory. *Computers in mathematics*, 125:109, 1990.

[6] D. H. BAILEY and J. M. BORWEIN. High-precision arithmetic in mathematical physics. *Mathematics*, 3(2):337–367, 2015.

[7] T. Y. CHOW. What is a closed-form number ? *The American Mathematical Monthly*, 106.5, 440–448 (1999).

[8] B. POONEN. Undecidable problems: a sampler. *Dans Interpreting Gödel : Critical Essays*, 211–241, 2014.

[9] S. M. RUMP. Verification methods: Rigorous results using floating-point arithmetic. *Acta Numerica*, 19:287–449, 2010.

[10] N. MÜLLER. The iRRAM: Exact arithmetic in C++. *Dans Computability and Complexity in Analysis*, pages 222–252. Springer, 2001. `http://irram.uni-trier.de`.

[11] N. REVOL and F. ROUILLIER. Motivations for an arbitrary precision interval arithmetic library and the MPFI library. *Reliable Computing*, 11(4):275–290, 2005. `http://perso.ens-lyon.fr/nathalie.revol/software.html`.

[12] M. SOFRONIOU and G. SPALETTA. Precise numerical computation. *Journal of Logic and Algebraic Programming*, 64(1):113–134, 2005.

[13] W. TUCKER. A rigorous ODE solver and Smale's 14th problem. *Foundations of Computational Mathematics*, 2(1):53–117, 2002.

[14] W. TUCKER. *Validated numerics*: *a short introduction to rigorous computations*. Princeton University Press, 2011.

[15] J. van der HOEVEN. Fast evaluation of holonomic functions. *Theoretical Computer Science*, 210:199–215, 1999.

[16] J. van der HOEVEN. Ball arithmetic. HAL preprint, 2009. `http://hal.archives-ouvertes.fr/hal-00432152/fr/`.

[17] J. van der HOEVEN, G. LECERF, B. MOURRAIN, P. TRÉBUCHET, J. BERTHOMIEU, D. N. DIATTA, and A. MANTZAFLARIS. Mathemagix: the quest of modularity and efficiency for symbolic and certified numeric computation ? *ACM Communications in Computer Algebra*, 45(3/4):186–188, January 2012. `http://mathemagix.org`.

[18] L. FOUSSE, G. HANROT, V. LEFÈVRE, P. PÉLISSIER, and P. ZIMMERMANN. MPFR: A multiple-precision binary floating-point library with correct rounding. *ACM Transactions on Mathematical Software*, 33(2):13, 2007.

[19] T. GRANLUND and the GMP development team. *GNU MP*: *The GNU Multiple Precision Arithmetic Library*, 6.1.2 edition, 2017.

[20] A. ENGE, M. GASTINEAU, P. THÉVENY, and P. ZIMMERMANN. MPC: a library for multiprecision complex arithmetic with exact rounding. `http://www.multiprecision.org/mpc/`, 2018.

[21] D. H. BAILEY, J. M. BORWEIN, V. KAPOOR and E. W. WEISSTEIN. Ten Problems in Experimental Mathematics. *The American Mathematical Monthly* 113:481–509, 2006.

[22] D. RICHARDSON and J. FITCH. The identity problem for elementary functions and constants *Actes de ISSAC'94*, ACM, 285–290, 1994.

[23] P. LAIREZ, M. MEZZAROBBA and M. SAFEY EL DIN. Computing the volume of compact semi-algebraic sets arXiv preprint arXiv:1904.11705, 2019.

[24] T. A. DRISCOLL, N. HALE, and L. N. TREFETHEN, editors. *Chebfun Guide*. Pafnuty Publications, Oxford, 2014.

[25] F. BRÉHARD, N. BRISEBARRE and M. JOLDES. Validated and numerically efficient Chebyshev spectral methods for linear ordinary differential equations. *ACM Transactions on Mathematical Software*, 44(4), 1–42, 2018.

[26] A. SCHÖNHAGE and V. STRASSEN. Schnelle Multiplikation grosser Zahlen. *Computing* 7, 281–292, 1971.

[27] D. HARVEY and J. van der HOEVEN. Integer multiplication in time $O(nlogn)$. HAL preprint, 2019. `https://hal.archives-ouvertes.fr/hal-02070778`.

[28] F. JOHANSSON. Numerical integration in arbitrary-precision ball arithmetic. *Mathematical Software – ICMS 2018*, 255–263, 2018.

[29] M. BERZ and K. MAKINO. Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. *Reliable Computing* 4, 361–369, 1998.

[30] M. KONTSEVICH and D. ZAGIER. Periods. Dans B. ENGQUIST et W. SCHMID (eds.), *Mathematics unlimited–2001 and beyond*, Berlin, New York : Springer-Verlag, 771–808, 2001.

Chapitre 5

Convexité combinatoire

Xavier GOAOC

Ce chapitre introduit à la convexité combinatoire, ses applications algorithmiques et ses prolongements en combinatoire topologique.

5.1 Introduction

Les structures combinatoires définies à partir de la géométrie exhibent souvent des propriétés assez particulières. Considérons par exemple les *graphe d'intervalles*, qui ont pour sommets une famille d'intervalles de $\mathbb{R}$ et pour arêtes les paires d'intervalles qui se coupent. Ces graphes ont la propriété remarquable qu'une densité positive d'arêtes assure l'existence d'une clique de taille linéaire. Plus précisément :

> Pour tout $\alpha \in (0, 1]$, tout graphe d'intervalle à n sommets ayant au moins $\alpha\binom{n}{2}$ arêtes a une clique de taille au moins $\frac{\alpha}{2}n$.

L'objectif de ce cours est d'introduire à des outils permettant d'étudier des propriétés similaire en dimension supérieure, par exemple les intersections de triangles du plan, de tétraèdres de l'espace, ... Nous établirons par exemple le résultat suivant, appelé *théorème de* HELLY *fractionnaire* :

> Pour tout $\alpha > 0$ et $d \geqslant 1$, il existe une constante $\beta = \beta(\alpha, d) > 0$ telle que pour toute famille $\mathcal{F}$ de n convexes de $\mathbb{R}^d$, si au moins $\alpha\binom{n}{d+1}$ sous-ensembles de $\mathcal{F}$ de taille $d+1$ sont d'intersection non vide, alors au moins βn éléments de $\mathcal{F}$ sont d'intersection non vide.

Ce cours est construit en deux temps.

Première partie

La section 5.2 détaille quelques résultats élémentaires mais fort utiles de géométrie convexe : il s'agit des théorèmes de CARATHÉODORY, RADON et HELLY. Une fois ces préliminaires géométriques établis, nous les utilisons pour analyser des questions géométriques *en dimension arbitraire* par des raisonnements essentiellement combinatoires. La section 5.3 détaille des exemples en analyse de profondeur géométrique, la section 5.4 donne une première preuve du théorème de HELLY fractionnaire ci-dessus (avec $\beta = \frac{\alpha}{d+1}$) et la section 5.5 présente deux applications en analyse d'algorithmes pour l'optimisation. L'objectif de cette première partie est d'illustrer comment les prolongements des théorèmes de CARATHÉODORY, RADON et HELLY esquissent une théorie «combinatoire» de la convexité.

Seconde partie

Cette convexité combinatoire entretient des liens étroits et relativement inattendus avec diverses questions en topologie de basse dimension, en combinatoire topologique et en théorie extrémale des hypergraphes. Nous examinerons cela dans une seconde partie en revisitant les résultats de convexité via trois structures voisines. Les *complexes simpliciaux géométriques* nous permettent, en section 5.6, de déduire le du théorème de BORSUK-ULAM des généralisations topologiques des théorèmes de RADON et HELLY. Les *complexes simpliciaux abstraits* nous permettent, en section 5.7, de déduire du *théorème de la borne supérieure* une version optimale du théorème de HELLY fractionnaire, où $\beta = 1 - (1-\alpha)^{1/(d+1)}$. Nous esquissons enfin en section 5.8 quelques liens entre convexité combinatoire et théorie extrémale des *hypergraphes* autour de questions de dimension de Vapnik-Chervonenkis et de motifs exclus.

Notations

Pour tout entier $n \geqslant 1$, on note $[n] \stackrel{\text{déf}}{=} \{1, 2, \ldots, n\}$. Pour tout ensemble fini X on note $|X|$ le cardinal de X, 2^X l'ensemble des parties de X, et $\binom{X}{k}$ l'ensemble des sous-ensembles de X de cardinal k. Pour toute famille $\mathcal{F}$ d'ensembles on note $\cap\mathcal{F}$ et $\cup\mathcal{F}$, respectivement, l'intersection et l'union des éléments de $\mathcal{F}$:

$$\cap\mathcal{F} \stackrel{\text{déf}}{=} \bigcap_{A\in\mathcal{F}} A \qquad \text{et} \qquad \cup\mathcal{F} \stackrel{\text{déf}}{=} \bigcup_{A\in\mathcal{F}} A. \tag{5.1}$$

5.2 Quelques bases en convexité

Commençons par reprendre les notions élémentaires de convexité.

5.2.1 Points et combinaisons linéaires

Nous travaillons dans $\mathbb{R}^d$, vu comme espace euclidien [1] de dimension d. Les *points* sont les d-uplets de nombres réels $\mathbf{p} = (p_1, p_2, \ldots, p_d)$. On note $\mathbf{x}, \mathbf{y}, \mathbf{z} \ldots$ les points et $\overrightarrow{x}, \overrightarrow{y}, \overrightarrow{z} \ldots$ les vecteurs. On note $\cdot$ le produit scalaire et $\| \ \|_2$ la norme euclidienne.

L'espace euclidien étant avant tout un espace vectoriel, nous pouvons faire des *combinaisons linéaires* de points. Étant donnés deux points $\mathbf{p} = (p_1, p_2, \ldots, p_d)$ et $\mathbf{q} = (q_1, q_2, \ldots, q_d)$ et deux réels α, β, appelés *poids*, on note

$$\alpha\mathbf{p} + \beta\mathbf{q} \stackrel{\text{déf}}{=} (\alpha p_1 + \beta q_1, \alpha p_2 + \beta q_2, \ldots, \alpha p_d + \beta q_d). \tag{5.2}$$

Les combinaisons linéaires de trois points ou plus sont définies de manière analogue. Une combinaison linéaire est *non-triviale* si au moins un de ses poids est non nul.

5.2.2 Indépendance affine et position générique

Une combinaison linéaire est *affine* si la somme de ses poids vaut 1. L'*enveloppe affine* d'une famille de points est l'ensemble de leurs combinaisons affines. Une famille $\mathcal{F}$ de points est *affinement indépendante* si pour tout $\mathbf{p} \in \mathcal{F}$, $\mathbf{p}$ n'appartient pas à l'enveloppe affine de $\mathcal{F} \setminus \{\mathbf{p}\}$. Un *sous-espace affine de dimension k* est l'enveloppe affine de $k+1$ points affinement indépendants. Dans $\mathbb{R}^d$, un sous-espace affine de dimension $d-1$ est appelé un *hyperplan*. Un ensemble de points de $\mathbb{R}^d$ est *en position générique* si chacun de ses sous-ensembles d'au plus $d+1$ points est affinement indépendant.

Exercice 1 Soit $P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$ un ensemble de points de $\mathbb{R}^d$. Montrer que l'indépendance affine de P est équivalente à chacune des conditions suivantes :

(i) Les vecteurs $\mathbf{p}_1 - \mathbf{p}_n, \mathbf{p}_2 - \mathbf{p}_n, \ldots, \mathbf{p}_{n-1} - \mathbf{p}_n$ sont linéairement indépendants.
(ii) $|P| \leqslant d+1$ et P est contenu dans un ensemble de $d+1$ points affinement indépendants.

1. La convexité est une notion affine mais nous prendrons quelques raccourcis euclidiens, voir par exemple les preuves du théorème de séparation et du théorème de CARATHÉODORY coloré.

Exercice 2 La *courbe des moments*, est le graphe de la fonction

$$\gamma : \left\{ \begin{array}{ccc} \mathbb{R} & \to & \mathbb{R}^d \\ t & \mapsto & (t, t^2, \ldots, t^d) \end{array} \right. \tag{5.3}$$

Montrer que la courbe des moments est en position générique.

Exercice 3 Montrer que pour tout hyperplan h de $\mathbb{R}^d$ il existe des réels $a_1, a_2, \ldots, a_d$ non tous nuls et un réel b tels que

$$h = \{(p_1, p_2, \ldots, p_d) \in \mathbb{R}^d : a_1 p_1 + a_2 p_2 + \ldots + a_d p_d = b\}. \tag{5.4}$$

5.2.3 Convexes et enveloppe convexe

On appelle *combinaison convexe* une combinaison linéaire dont les poids sont tous positifs ou nuls et sont de somme égale à 1. Un *segment* est l'ensemble des combinaisons convexes de deux points, ses *extrémités*. Un sous-ensemble $C \subseteq \mathbb{R}^d$ est *convexe* si tout segment à extrémités dans C est lui-même contenu dans C. Un *convexe* de $\mathbb{R}^d$ est un sous-ensemble convexe de $\mathbb{R}^d$.

FIGURE 5.1 – *Dans $\mathbb{R}^2$, un ensemble convexe (à gauche) et trois ensembles non-convexes. Pour chaque ensemble non-convexe, un exemple de segment violant la condition de convexité est donné.*

L'*enveloppe convexe* d'un sous-ensemble $E \subseteq \mathbb{R}^d$, notée $\mathrm{conv}(E)$, est l'intersection de tous les ensembles convexes $C \subseteq \mathbb{R}^d$ qui contiennent E. Comme une intersection *quelconque* d'ensembles convexes est convexe, $\mathrm{conv}(E)$ est convexe pour tout $E \subseteq \mathbb{R}^d$.

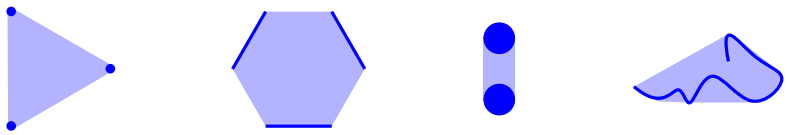


FIGURE 5.2 – *Exemples de sous-ensembles du plan (en sombre) et de leurs enveloppes convexes (union des parties claires et sombres).*

Exercice 4 Montrer que l'enveloppe convexe d'un ensemble E peut aussi se définir comme l'ensemble des combinaisons convexes de sous-ensembles finis de E :

$$\begin{aligned}\mathrm{conv}(E) = \Big\{ &\alpha_1 \mathbf{p_1} + \alpha_2 \mathbf{p_2} + \ldots + \alpha_n \mathbf{p_n} : n \in \mathbb{N}^* \,; \\ &\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_n} \in E \,; \alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{R}^+ \,; \sum_{i=1}^{n} \alpha_i = 1 \Big\}.\end{aligned} \tag{5.5}$$

(Indications : procéder par récurrence sur n pour $\subseteq$ et remarquer, pour $\supseteq$, que le terme de droite est un convexe contenant E.)

5.2.4 Simplexes et lemmes de CARATHÉODORY et RADON

Un *simplexe* de $\mathbb{R}^d$ est l'enveloppe convexe de points de $\mathbb{R}^d$ affinement indépendants ; ces points sont les *sommets* du simplexe. La *dimension* d'un simplexe est son nombre de sommets moins 1. On abrège «simplexe de dimension k» en *k-simplexe*.

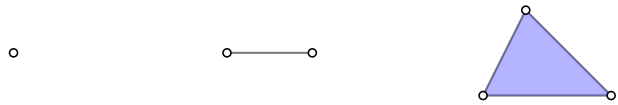


FIGURE 5.3 – *Des simplexes de $\mathbb{R}^2$ de dimension 0 (un point), 1 (un segment) et 2 (un triangle).*

Nous allons avoir recours à deux propriétés élémentaires des simplexes. La première a été observée par Constantin CARATHÉODORY vers 1905.

Lemme 5.2.1. *Pour tout $X \subseteq \mathbb{R}^d$, l'enveloppe convexe de X est l'union des simplexes à sommets dans X.*

Démonstration. Associons à tout point $\mathbf{a}$ de $\mathbb{R}^d$ le vecteur $\overrightarrow{a} = (a_1, a_2, \ldots, a_d, 1)$ de $\mathbb{R}^{d+1}$. D'après l'équation (5.5), il existe $n \geqslant 2$, $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n$ dans X et $\alpha_1, \alpha_2, \ldots, \alpha_n \in [0, 1]$ tels que

$$\mathbf{p} = \sum_{i=1}^{n} \alpha_i \mathbf{p}_i \quad \text{et} \quad 1 = \sum_{i=1}^{n} \alpha_i, \qquad \text{c'est-à-dire} \qquad \overrightarrow{p} = \sum_{i=1}^{n} \alpha_i \overrightarrow{p_i}. \tag{5.6}$$

Montrons que si les $\mathbf{p}_i$ sont affinement dépendants, alors une écriture de la forme (5.6) avec un $n-1$ termes existe. Il suffira alors de considérer une écriture de la forme (5.6) dans laquelle n est minimal pour conclure.

Si les $\mathbf{p}_i$ sont affinement dépendants, alors les $\overrightarrow{p_i}$ sont linéairement dépendants (les deux propriétés sont en fait équivalentes). Il existe alors des réels $\beta_1, \beta_2, \ldots, \beta_n$ non tous nuls tels que

$$\beta_1 \overrightarrow{p_1} + \beta_2 \overrightarrow{p_2} + \ldots + \beta_n \overrightarrow{p_n} = \overrightarrow{0}. \tag{5.7}$$

Pour tout $\lambda \in \mathbb{R}$, on a donc

$$\overrightarrow{p} = \sum_{i=1}^{n} (\alpha_i + \lambda\beta_i)\, \overrightarrow{p_i}. \tag{5.8}$$

Les vecteurs $\overrightarrow{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ et $\overrightarrow{\beta} = (\beta_1, \beta_2, \ldots, \beta_n)$ ne sont pas colinéaires puisque la somme de leurs coordonnées vaut, respectivement, 1 et 0. Quitte à changer $\overrightarrow{\beta}$ en $-\overrightarrow{\beta}$, on peut supposer qu'il existe $i \in [n]$ tel que $\beta_i < 0$. Pour $\lambda = \min\{-\alpha_i/\beta_i : \beta_i < 0\}$, dans l'équation (5.8), tous les coefficients sont positifs et au moins l'un d'entre eux est nul. ■

La seconde propriété qui nous intéresse ici a été observée par Johann RADON vers 1921. Définissons deux simplexes comme *indépendants* s'ils n'ont aucun sommet commun.

Lemme 5.2.2. *Si $X \subset \mathbb{R}^d$ est affinement dépendant, alors il existe deux simplexes indépendants à sommets dans X qui se coupent.*

FIGURE 5.4 – *Le lemme de* RADON *dans le plan : 4 points non alignés déterminent deux segments qui se coupent ou un triangle qui contient le dernier point.*

Démonstration. Associons à tout point $\mathbf{a}$ de $\mathbb{R}^d$ le vecteur $\overrightarrow{a} = (a_1, a_2, \ldots, a_d, 1)$ de $\mathbb{R}^{d+1}$ et notons $\overrightarrow{X} \stackrel{\text{déf}}{=} \{\overrightarrow{a} : a \in X\}$. Puisque X est affinement dépendant, $\overrightarrow{X}$ est linéairement dépendant et il existe une combinaison linéaire non-triviale

$$\alpha_1 \overrightarrow{p_1} + \alpha_2 \overrightarrow{p_2} + \ldots + \alpha_k \overrightarrow{p_k} = \overrightarrow{0}. \tag{5.9}$$

où les $\mathbf{p}_i$ sont dans X. Choisissons-en une où k est minimal. Notons $P \stackrel{\text{déf}}{=} \{i \in [k] : \alpha_i \geqslant 0\}$ et remarquons que P n'est ni vide ni égal à $[k]$. L'examen de la dernière coordonnée révèle que les α_i somment à 0. Notons

$$\alpha \stackrel{\text{déf}}{=} \sum_{i \in P} \alpha_i = \sum_{i \in [k] \setminus P} (-\alpha_i),$$

et remarquons que

$$\sum_{i\in P}\frac{\alpha_i}{\alpha}\mathbf{p}_i = \sum_{i\in[k]\setminus P}\frac{-\alpha_i}{\alpha}\mathbf{p}_i \qquad \text{et} \qquad \sum_{i\in P}\frac{\alpha_i}{\alpha} = 1 = \sum_{i\in[k]\setminus P}\frac{-\alpha_i}{\alpha}.$$

Le point $\mathbf{p} \stackrel{\text{déf}}{=} \sum_{i\in P}\frac{\alpha_i}{\alpha}\mathbf{p}_i = \sum_{i\in[k]\setminus P}\frac{-\alpha_i}{\alpha}\mathbf{p}_i$ est donc à la fois dans l'enveloppe convexe de $A = \{\mathbf{p}_i : i \in P\}$ et dans l'enveloppe convexe de $B = \{\mathbf{p}_i : i \in [k] \setminus P\}$. Ces enveloppes convexes se coupent donc et la minimalité de k assure que chacune de ces enveloppes convexes est un simplexe. ■

Exercice 5 Un sous-ensemble de $\mathbb{R}^d$ est *compact* s'il est fermé et borné ; un résultat classique est que l'image d'un compact par une application continue est compacte. Prouver que l'enveloppe convexe d'un sous-ensemble compact de $\mathbb{R}^d$ est compacte.

Exercice 6 Prouver la «réciproque» suivante du lemme de RADON. Si P et Q sont deux ensembles de points de $\mathbb{R}^d$ tels que $P \cup Q$ est affinement indépendant, alors $\text{conv}(P) \cap \text{conv}(Q) = \text{conv}(P \cap Q)$.

Exercice 7 Prouver que pour tous entiers $r \geqslant 2$ et $d \geqslant 2$ il existe un entier $t = t(r, d)$ tel que pour tout ensemble X d'au moins t points en position générique dans $\mathbb{R}^d$, il existe t simplexes indépendants d'intersection non vide et à sommets dans X.

5.2.5 Demi-espaces et séparation

Un *demi-espace fermé* (resp. *ouvert*) est l'ensemble des points dont les coordonnées satisfont une inegalité linéaire large (resp. stricte). Ainsi, un hyperplan peut être vu comme le bord de deux demi-espaces fermés et de deux demi-espaces ouverts.

Exercice 8 Montrer que tout hyperplan, tout demi-espace ouvert et tout demi-espace fermé est convexe.

Soient A et B deux convexes disjoints de $\mathbb{R}^d$. Un hyperplan h *sépare* A et B si les deux demi-espaces fermés bordés par h contiennent chacun au moins un [2] des convexes. Un hyperplan h *sépare strictement* A et B si les deux demi-espaces ouverts bordés par h contiennent chacun soit A, soit B. Le théorème classique suivant nous sera utile.

2. En particulier, pour cette définition, h sépare A et B s'ils sont tous les deux contenus dans h.

Théorème 5.2.3. *Toute paire de convexes disjoints de $\mathbb{R}^d$ peut être séparée. Toute paire de convexes disjoints de $\mathbb{R}^d$ dont au moins l'un est compact peut être séparée strictement.*

Idée de preuve. Soient A et B deux convexes disjoints de $\mathbb{R}^d$. Supposons tout d'abord A et B compacts. Il existe alors deux points $\mathbf{a} \in A$ et $\mathbf{b} \in B$ tels que la distance entre A et B est $d_2(A, B) = \|\mathbf{ab}\|_2 > 0$. Remarquons que l'hyperplan h perpendiculaire au segment d'extrémités $\mathbf{a}$ et $\mathbf{b}$ en son milieu est disjoint de A et de B. L'hyperplan h sépare donc strictement A et B. Si seul A est compact, définissons B' comme l'intersection de B avec une boule euclidienne fermée, choisie suffisamment grande pour que $d_2(A, B) = d_2(A, B')$. Il existe alors deux point $\mathbf{a} \in A$ et $\mathbf{b} \in B'$ tels que $\|\mathbf{ab}\|_2 = d_2(A, B') = d_2(A, B)$ et on est ramené au cas précédent.

Dans le cas où ni A ni B n'est compact, on utilise le fait suivant : pour tout convexe $C \subseteq \mathbb{R}^d$, il existe une suite $\{C_i\}_{i\in\mathbb{N}}$ telle que chaque C_i est convexe et compact, et $\cup_{i\in\mathbb{N}} C_i = C$.[3] Ainsi, on approche A et B par deux suites $\{A_i\}_{i\in\mathbb{N}}$ et $\{B_i\}_{i\in\mathbb{N}}$ de convexes compacts et on note h_i un hyperplan séparant strictement A_i et B_i. Cet ensemble $\{h_i\}_{i\in\mathbb{N}}$ d'hyperplans admet un point d'accumulation h^*, qui sépare A et B. ■

Exercice 9 Formaliser complètement la preuve du théorème de séparation. *(Indication : pour l'argument final de compacité, on pourra représenter un hyperplan h_i par un vecteur $\overrightarrow{a_i}$ et un réel b_i tels que $h_i = \{\mathbf{x} : \mathbf{x} \cdot \overrightarrow{a_i} = b_i\}$ et $(\overrightarrow{a_i}; b_i)$ soit de norme 1.)*

Exercice 10 Un ensemble E de points est *en position convexe* si aucun d'entre eux n'appartient à l'enveloppe convexe des autres, c'est-à-dire

$$\forall \mathbf{p} \in E, \quad \mathbf{p} \notin \mathrm{conv}(E \setminus \{\mathbf{p}\}). \tag{5.10}$$

Montrer que toute famille finie de points du cercle unité dans $\mathbb{R}^2$ est en position convexe.[4]

3. En effet, fixons $\mathbf{p} \in C$ et pour $\alpha \in \mathbb{R}$ notons αC l'image de C par une homothétie de centre $\mathbf{p}$ et de rapport α. Définissons maintenant C_i comme l'intersection de la clôture de $\left(1 - \frac{1}{n}\right) C$ et de la boule fermée de centre $\mathbf{p}$ et de rayon n.

4. Le théorème d'Erdös-Szekeres [7, §3] énonce que pour tous entiers $d \geqslant 2$ et $k \geqslant 3$ il existe un entier $N_{k,d}$ tel que tout ensemble de $N_{k,d}$ points de $\mathbb{R}^d$ en position générique contient k points en position convexe. Ainsi, non seulement les ensembles de points en position convexe peuvent être arbitrairement grands, mais ils sont de plus inévitables au sens de la théorie de Ramsey.

5.2.6 Théorème de HELLY

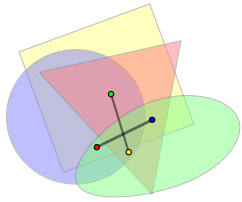
Si A et B sont deux sous-espaces vectoriels de $\mathbb{R}^d$ non inclus l'un dans l'autre, alors $A \cap B$ est un espace vectoriel de dimension strictement inférieure à A et B. Qu'en est-il dans le cadre convexe ? D'une part, la classe des sous-ensembles convexes de $\mathbb{R}^d$ est elle aussi close par intersection. D'autre part, si cette opération ne fait pas, en général, diminuer la dimension, la propriété suivante, plus faible, demeure.

Théorème 5.2.4. *Toute famille finie de convexes de $\mathbb{R}^d$ d'intersection vide contient une sous-famille d'intersection vide et de taille au plus $d+1$.*

Cette propriété a été découverte par Eduard HELLY vers 1913. Elle est souvent donnée dans sa forme contraposée, à savoir : une famille finie de convexes de $\mathbb{R}^d$ est d'intersection non vide si et seulement si toute sous-famille d'au plus $d+1$ d'entre eux est d'intersection non vide. Nous allons en donner trois preuves.

Exercice 11 Soient P et Q deux ensembles finis de points de $\mathbb{R}^2$. Montrer que P et Q sont séparables si et seulement si l'on peut séparer tous les sous-ensembles $P' \subset P$ et $Q' \subset Q$ tels que $|P'| + |Q'| \leqslant 4$. Généraliser ce résultat en dimension d.

RADON $\Rightarrow$ HELLY



Cette preuve est sans doute la plus élégante et a motivé le lemme de RADON. Soit $\mathcal{F}$ une famille finie d'intersection vide et dont toutes les sous-familles propres sont d'intersection non vide. Pour tout $A \in \mathcal{F}$, choisissons un point $\mathbf{p}_A$ dans $\cap(\mathcal{F} \setminus \{A\})$. Notons P l'ensemble des points choisis. Si $\mathcal{F}$ compte $d+2$ éléments ou plus, P est affinement dépendant et se décompose en $P = P_1 \cup P_2$, où P_1 et P_2 sont deux parties disjointes, non vides, et telles que $\mathrm{conv}(P_1) \cap \mathrm{conv}(P_2)$ est non vide. Choisissons $\mathbf{p} \in \mathrm{conv}(P_1) \cap \mathrm{conv}(P_2)$. Remarquons que pour tout $A \in \mathcal{F}$, $\mathbf{p}_A$ est le seul

point de P qui n'est pas contenu dans A. Ainsi,

$$\begin{aligned}\mathbf{p} \in \operatorname{conv}(P_1) \cap \operatorname{conv}(P_2) &\subseteq \big(\cap_{A\colon \mathbf{p}_A \notin P_1} A\big) \cap \big(\cap_{A\colon \mathbf{p}_A \notin P_2} A\big) \\ &= \big(\cap_{A\colon \mathbf{p}_A \in P_2} A\big) \cap \big(\cap_{A\colon \mathbf{p}_A \in P_1} A\big) = \cap\mathcal{F},\end{aligned}$$

et $\cap\mathcal{F}$ est non vide.

Séparation $\Rightarrow$ HELLY

Soit $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$ une famille de convexes compacts de $\mathbb{R}^d$ telle que $\cap\mathcal{F}$ est vide et pour tout $A \in \mathcal{F}$, $\cap(\mathcal{F} \setminus \{A\})$ est non vide. Il nous suffit de prouver que $n \leqslant d+1$. Nous procédons par récurrence sur la dimension d. L'initialisation pour $d = 1$ est immédiate. Notons $B \stackrel{\text{déf}}{=} \cap_{i=1}^{n-1} A_i$. Puisque A_n et B sont disjoint et compacts, il existe un hyperplan h qui les sépare strictement. Notons $B_j \stackrel{\text{déf}}{=} A_j \cap h$ pour $j \leqslant n-1$ et $\mathcal{F}' = \{B_1, B_2, \ldots, B_{n-1}\}$. Comme $B \cap h$ est vide, $\mathcal{F}'$ est d'intersection vide. Comme $\cap(\mathcal{F} \setminus \{A_j, A_n\})$ coupe A_n et B pour tout $j \leqslant n-1$, toute sous-famille de $\mathcal{F}'$ est d'intersection non vide. Ainsi, $\mathcal{F}'$ est une famille minimale d'intersection vide de convexes compacts de h. Comme h est affinement équivalent à $\mathbb{R}^{d-1}$, l'hypothèse de récurrence assure que $n-1 \leqslant (d-1)+1$.

La relaxation de l'hypothèse que les A_i sont compacts peut se faire, comme dans la preuve du théorème de séparation (5.2.3) en remarquant que pour tout convexe $C \subseteq \mathbb{R}^d$ il existe une suite $C_1 \subseteq C_2 \subseteq \ldots \subseteq C_n \subseteq \ldots$ de convexes compacts tels que $C = \cup_{i \geqslant 1} C_i$. Ainsi, on peut partir d'une famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$ telle que toute sous-famille d'au plus $d+1$ ensembles est d'intersection non vide. Pour $\mathcal{G} \subseteq \mathcal{F}$ avec $|\mathcal{G}| \leqslant d+1$, on fixe $\mathbf{p}_\mathcal{G} \in \cap\mathcal{G}$. On fixe aussi pour chaque ensemble A_i une famille $\{A_i^j\}_{j \in \mathbb{N}}$ de convexes compacts emboîtés tels que $A_i = \cup_{j \in \mathbb{N}} A_i^j$. Soit n_0 le plus petit entier tel que

$$\forall \mathcal{G} \subseteq \mathcal{F} \text{ t. q. } |\mathcal{G}| \leqslant d+1, \quad p_\mathcal{G} \in \bigcap_{i\colon A_i \in \mathcal{G}} A_i^{n_0}. \tag{5.11}$$

L'énoncé dans le cas compact assure que la famille $\{A_i^{n_0}\}_{i \leqslant n}$ est d'intersection vide. Il en va de même pour $\mathcal{F}$ par inclusion.

CARATHÉODORY $\Rightarrow$ HELLY

Le demi-espace *polaire* à un point $\mathbf{a} = (a_1, a_2, \ldots, a_d) \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ est défini comme

$$\widehat{\mathbf{a}} \stackrel{\text{déf}}{=} \{\mathbf{p} \in \mathbb{R}^d \colon a_1 p_1 + a_2 p_2 + \ldots + a_d p_d \geqslant 1\}. \tag{5.12}$$

Pour une famille P de points de $\mathbb{R}^d$, on note $\widehat{P} \stackrel{\text{déf}}{=} \{\widehat{p}\colon p \in P\}$. On laisse en exercice le fait de montrer que pour toute famille finie P de points de $\mathbb{R}^d$, $\mathbf{0} \in \text{conv}(P)$ si et seulement si $\cap\widehat{P}$ est vide. Cette propriété permet de déduire le théorème de HELLY dans le cas particulier de demi-espaces fermés de $\mathbb{R}^d$ du théorème de CARATHÉODORY. [5] Cela fonctionne en fait dans les deux sens : le théorème de HELLY pour les demi-espaces fermés est équivalent, par polarité, au théorème de CARATHÉODORY.

Le cas général du théorème de HELLY peut se déduire comme suit du cas particulier où les objets sont des demi-espaces fermés. Partant d'une famille $\mathcal{F}$ de convexes de $\mathbb{R}^d$, on commence par fixer pour chaque $\mathcal{G} \subseteq \mathcal{F}$ d'intersection non vide un point $\mathbf{p}_\mathcal{G}$. Pour chaque élément $A \in \mathcal{F}$, posons $A' \stackrel{\text{déf}}{=} \text{conv}\,(\{\mathbf{p}_\mathcal{G}\colon A \in \mathcal{G}\})$. Remarquons que le théorème de HELLY est vrai pour $\mathcal{F}$ si et seulement si il est vrai pour la famille $\{A'\colon A \in \mathcal{F}\}$. On conclut par le théorème de WEYL-MINKOWSKI qui énonce que les intersections *bornées* d'un nombre fini de demi-espace fermés sont exactement les enveloppes convexes d'un nombre fini de points.

5.3 Application aux profondeurs géométriques

Examinons une première application des outils de convexité combinatoire à l'analyse d'analogues en dimension quelconque de la notion de médiane.

5.3.1 Définitions

Nous allons nous intéresser ici à la mesure de la *profondeur* d'un point $\mathbf{x}$ relativement à un sous-ensemble (fini) P de $\mathbb{R}^d$. On examine plus précisément deux notions.

Profondeur de demi-espace

Notons $\mathcal{H}_d$ l'ensemble des demi-espaces fermés de $\mathbb{R}^d$ et posons, pour tout point $\mathbf{x}$ de $\mathbb{R}^d$,

$$\text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) \stackrel{\text{déf}}{=} \min_{h \in \mathcal{H}_d\colon \mathbf{x} \in h} |h \cap P|. \tag{5.13}$$

5. Dans le détail, soit $\mathcal{F} = \{\widehat{\mathbf{a}}_1, \widehat{\mathbf{a}}_2, \ldots, \widehat{\mathbf{a}}_n\}$ un ensemble de demi-espaces fermés, et $P = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$ où $\mathbf{a}_i$ est le point polaire de $\widehat{\mathbf{a}}_i$. Si $\cap\mathcal{F}$ est vide, alors $\mathbf{0} \in \text{conv}(P)$ et donc, par le théorème de CARATHÉODORY, il existe $Q \subseteq P$ tel que $\mathbf{0} \in \text{conv}(Q)$ et $|Q| \leqslant d+1$. La sous-famille $\mathcal{G} \stackrel{\text{déf}}{=} \{\widehat{\mathbf{a}}\colon \mathbf{a} \in Q\}$ de $\mathcal{F}$ est de cardinal au plus $d+1$ (comme Q) et d'intersection vide (car $\mathbf{0} \in \text{conv}(Q)$).

Autrement dit, la *profondeur de demi-espace* $\mathrm{prof}_{\mathcal{H}_d}(\mathbf{x}, P)$ est le nombre minimum de points de P contenus dans tout demi-espace contenant $\mathbf{x}$. Cette notion est aussi appelée *profondeur de* TUKEY. Pour $d = 1$, cette profondeur est au plus $\lfloor \frac{|P|}{2} \rfloor$ et cette borne est atteinte par toute médiane.

Profondeur simpliciale

Notons $S(P)$ l'ensemble des simplexes à sommets dans P et posons, pour tout point $\mathbf{x}$ de $\mathbb{R}^d$,

$$\mathrm{prof}_S(\mathbf{x}, P) \stackrel{\text{déf}}{=} |\{\sigma \in S(P) \colon \mathbf{x} \in \sigma\}| \,. \tag{5.14}$$

Autrement dit, la *profondeur simpliciale* $\mathrm{prof}_S(\mathbf{x}, P)$ est le nombre de simplexes à sommets dans P contenant $\mathbf{x}$. Pour $d = 1$, cette profondeur est au plus $\lceil \frac{|P|}{2} \rceil \lfloor \frac{|P|}{2} \rfloor$ et cette borne est atteinte pour toute médiane.

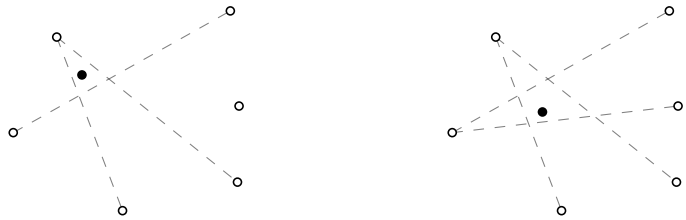


FIGURE 5.5 – *Un ensemble P de six points (en blanc) et deux points (en noir) de profondeurs de demi-espace* 1 *(à gauche) et* 2 *(à droite) et de profondeurs simpliciale* 5 *(à gauche) et* 6 *(à droite).*

5.3.2 Théorème du point central

Richard RADO a établi en 1946 une borne inférieure sur la profondeur de demi-espace maximale pour *tout* ensemble fini de points de $\mathbb{R}^d$.

Théorème 5.3.1. *Pour tout ensemble fini $P \subseteq \mathbb{R}^d$ il existe un point $\mathbf{c} \in \mathbb{R}^d$ tel que $\mathrm{prof}_{\mathcal{H}_d}(\mathbf{c}, P) \geqslant \frac{1}{d+1}|P|$.*

Démonstration. Notons $Q_1, Q_2, \ldots, Q_k$ les éléments de $\{P \cap h \colon h \in \mathcal{H}_d\}$ de taille strictement plus grande que $\left(1 - \frac{1}{d+1}\right)|P|$. Autrement dit, les Q_i sont les sous-ensemble de P de grande taille que l'on peut obtenir comme intersection de P par un demi-espace. Notons $\mathcal{F} = \{\mathrm{conv}(Q_i) \colon i = 1, 2, \ldots, k\}$. Remarquons que toute sous-famille de $d + 1$ éléments de $\mathcal{F}$ est d'intersection non vide, ne serait-ce que sur P. Ainsi, le théorème de HELLY, dans sa forme contraposée, implique que $\cap\mathcal{F}$ est non vide. Soit $\mathbf{c}$ un point de

cette intersection. Observons maintenant que si un demi-espace contient strictement moins que $\frac{1}{d+1}|P|$ points de P, alors son complémentaire est un des Q_i et contient par conséquent le point $\mathbf{c}$. ■

Un point $\mathbf{c}$ tel que $\text{prof}_{\mathcal{H}_d}(\mathbf{p}, P) \geqslant \frac{1}{d+1}|P|$ est appelé *point central* de P. Le théorème 5.3.1 est appelé *théorème du point central*.

Le théorème du point central peut aussi s'énoncer sur les lois de probabilités sur $\mathbb{R}^d$ pour la σ-algèbre des Boréliens par densité des mesures empiriques : toute loi μ admet un point $\mathbf{c} \in \mathbb{R}^d$ tel que tout demi-espace (ouvert ou fermé) contenant $\mathbf{c}$ est de μ-mesure au moins $\frac{1}{d+1}$. Dans le cas où μ est la loi uniforme sur un convexe K de $\mathbb{R}^d$, Branko GRÜNBAUM a prouvé que tout demi-espace contenant le centre de gravité de K a μ-mesure au moins $\left(\frac{d}{d+1}\right)^d$. Ce ratio est bien meilleur que celui garanti par le théorème du point central, puisqu'il décroît vers $\frac{1}{e}$ quand d augmente.

Exercice 12 Proposer une ou plusieurs constructions montrant que la constante $\frac{1}{d+1}$ dans le théorème 5.3.1 est optimale et que le point dont l'existence est établie au théorème 5.3.1 peut être unique.

5.3.3 Profondeur de demi-espace et simplexes indépendants

Nous attaquons maintenant la preuve que tout ensemble fini $P \subset \mathbb{R}^d$ admet un point de grande profondeur simpliciale. Notre première étape consiste à relier la profondeur de demi-espace d'un point $\mathbf{x}$ relativement à P au nombre de simplexes *indépendants* à sommets dans P qui couvrent $\mathbf{x}$.

Corollaire 5.3.2. *Pour tout ensemble fini $P \subset \mathbb{R}^d$ en position générique et tout point $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{x}$ est contenu dans au moins $\lceil \frac{1}{d+1} \text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) \rceil$ simplexes indépendants à sommets dans P.*

Démonstration. Supposons $\text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) \geqslant d+1$ sans quoi l'énoncé est trivialement vrai. Nous produisons cette famille de simplexes indépendants en appliquant de manière répétée le théorème de CARATHÉODORY.

Notons $P_1 \stackrel{\text{déf}}{=} P$. Puisque $\text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) > 0$, il n'existe pas d'hyperplan séparant strictement $\mathbf{x}$ de $\text{conv}(P)$. Cela implique que $\mathbf{x} \in \text{conv}(P)$. Tant que $\text{conv}(P_i)$ contient $\mathbf{x}$, et c'est le cas pour $i = 1$, on choisit un simplexe de sommets $Q_i \subseteq P_i$ tel que $\mathbf{x} \in \text{conv}(Q_i)$, et on pose $P_{i+1} \stackrel{\text{déf}}{=} P_i \setminus Q_i$. Notons k l'index du dernier ensemble Q_i ainsi construit. Il s'agit maintenant de minorer k.

Soit H un hyperplan séparant strictement $\mathbf{x}$ et $\text{conv}(P_{k+1})$. Notons D le demi-espace fermé borné par H contenant $\mathbf{x}$. Comme D contient $\mathbf{x}$, on a $|D \cap P_1| \geqslant \text{prof}_{\mathcal{H}_d}(\mathbf{x}, P_1) = \text{prof}_{\mathcal{H}_d}(\mathbf{x}, P)$. Comme H sépare strictement $\mathbf{x}$ de $\text{conv}(P_{k+1})$, on a $|D \cap P_{k+1}| = 0$. Ainsi,

$$\text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) \leqslant |D \cap P_1| = \sum_{i=1}^{k} |D \cap Q_i| \leqslant \sum_{i=1}^{k} |Q_i| \leqslant k(d+1)$$

et k est donc au moins $\lceil \frac{1}{d+1} \text{prof}_{\mathcal{H}_d}(\mathbf{x}, P) \rceil$. ■

Ainsi, d'après ce corollaire, tout point central de P est contenu dans un nombre linéaire de simplexes indépendants à sommets dans P.

5.3.4 Théorème de CARATHÉODORY coloré

Nous établissons maintenant une variante, dite *colorée*, du théorème de CARATHÉODORY découverte par Imre BÁRÁNY en 1982. Il est souvent résumé ainsi : si dans un ensemble fini de points de $\mathbb{R}^d$ en position générique, colorés par $d+1$ couleurs, chaque couleur contient l'origine dans son enveloppe convexe, alors il existe un simplexe arc-en-ciel (*i.e.* ayant un point de chaque couleur) qui contient l'origine dans son enveloppe convexe. Plus formellement :

Théorème 5.3.3. *Pour tous ensembles finis $E_1, E_2, \ldots, E_{d+1} \subseteq \mathbb{R}^d$ et tout point $\mathbf{p} \in conv(E_1) \cap conv(E_2) \cap \ldots \cap conv(E_{d+1})$, il existe $\mathbf{p}_1 \in E_1$, $\mathbf{p}_2 \in E_2$, …, $\mathbf{p}_{d+1} \in E_{d+1}$ tels que $\mathbf{p} \in conv(\{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{d+1}\})$.*

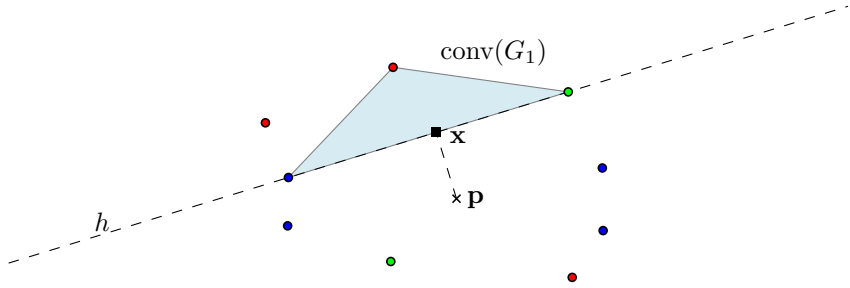
Démonstration. Notons $E = E_1 \cup E_2 \cup \ldots \cup E_{d+1}$ et

$$X = \{G \subseteq E : |G| = d+1 \text{ et } \forall i \in [d+1], |G \cap E_i| = 1\} \tag{5.15}$$

Prenons $G_1 \in X$ tel que la distance $d_2(\mathbf{p}, \text{conv}(G_1))$ de $\mathbf{p}$ à $\text{conv}(G_1)$ soit minimale. Nous prouvons que $\mathbf{p} \in \text{conv}(G_1)$ par contraposition en montrant que si cette distance est non nulle, il existe un autre $G_2 \in X$ tel que $d_2(\mathbf{p}, \text{conv}(G_2)) < d_2(\mathbf{p}, \text{conv}(G_1))$.

Pour cela, introduisons les objets suivants. Soit $\mathbf{x}$ le point de $\text{conv}(G_1)$ le plus proche de $\mathbf{p}$. Soit h l'hyperplan perpendiculaire en $\mathbf{x}$ au segment d'extrémités $\mathbf{p}$ et $\mathbf{x}$. Soit h^- le demi-espace ouvert bordé par h contenant $\mathbf{p}$ et h^+ son complémentaire. Remarquons d'une part que l'unicité de $\mathbf{x}$ découle de la convexité de $\text{conv}(G_1)$, et d'autre part que $\text{conv}(G_1)$ est contenu dans h^+ (nous avons utilisé une idée similaire dans la preuve du Théorème 5.2.3).

Par le théorème de CARATHÉODORY dans h, il existe un sous-ensemble $T \subseteq G_1 \cap h$ d'au plus d points tel que $\mathbf{x} \in \text{conv}(T)$. Soit i tel que $T \cap E_i = \emptyset$.



Remarquons que E_i doit rencontrer h^- : en effet, si $E_i \subset h^+$ alors $\mathbf{p} \in \text{conv}(E_i) \subset h^+$ or $\mathbf{p} \in h^-$. Il existe donc un point $\mathbf{y} \in E_i \cap h^-$. Remarquons que $T \cup \{y\}$ peut s'étendre en un élément G_2 de X. Au voisinage de $\mathbf{x}$, le segment $\mathbf{xp}$ est contenu dans $\text{conv}(G_2)$ et l'on a donc la contradiction souhaitée $d_2(\mathbf{p}, \text{conv}(G_2)) < d_2(\mathbf{p}, \text{conv}(G_1))$. ■

Théorème de HELLY coloré

Via l'équivalence par polarité entre théorèmes de CARATHÉODORY et de HELLY observée dans la troisième preuve de la Section 5.2.6, le théorème de CARATHÉODORY coloré peut se reformuler ainsi :

Théorème 5.3.4. *Soient $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_{d+1}$ des familles finies de convexes de $\mathbb{R}^d$. Si pour tous $A_1 \in \mathcal{F}_1$, $A_2 \in \mathcal{F}_2$, $\ldots$, $A_{d+1} \in \mathcal{F}_{d+1}$ l'intersection $A_1 \cap A_2 \cap \ldots \cap A_{d+1}$ est non vide, alors il existe $1 \leqslant i \leqslant d+1$ tel que $\cap \mathcal{F}_i$ est non vide.*

L'exercice 13 propose une preuve directe de ce résultat et la section 5.8 en donne une application.

5.3.5 Lemme de sélection

Nous arrivons au résultat annoncé que tout ensemble fini de points admet un point de grande profondeur simpliciale :

Théorème 5.3.5. *Pour tout entier $d \geqslant 2$, il existe une constante $c_d > 0$ telle que pour tout ensemble fini $P \subset \mathbb{R}^d$ en position générique, il existe $\mathbf{x} \in \mathbb{R}^d$ tel que $prof_S(\mathbf{x}, P) \geqslant c_d \binom{|P|}{d+1}$.*

Démonstration. Soit $\mathbf{x}$ un point central de P. Examinons sa profondeur simpliciale. Le Corollaire 5.3.2 assure qu'il existe $k = \lfloor \frac{|P|}{(d+1)^2} \rfloor$ simplexes indépendants à sommets dans P qui contiennent $\mathbf{x}$. Notons leurs ensembles de sommets $Q_1, Q_2, \ldots, Q_k$. Pour chaque choix d'indices $1 \leqslant i_1 < i_2 < \ldots < i_{d+1} \leqslant k$, le Théorème de CARATHÉODORY coloré appliqué aux ensembles

$Q_{i_1}, Q_{i_2}, \ldots, Q_{i_{d+1}}$ assure qu'il existe un sous-ensemble $Q_{i_1,i_2,\ldots,i_{d+1}} \subseteq P$ tel que

$$|Q_{i_1,i_2,\ldots,i_{d+1}} \cap Q_{i_1}| = |Q_{i_1,i_2,\ldots,i_{d+1}} \cap Q_{i_2}| = \ldots = |Q_{i_1,i_2,\ldots,i_{d+1}} \cap Q_{i_{d+1}}| = 1$$

et $\mathbf{x} \in \text{conv}(Q_{i_1,i_2,\ldots,i_{d+1}})$. Comme les Q_i sont deux à deux disjoints, les $Q_{i_1,i_2,\ldots,i_{d+1}}$ sont deux à deux distincts. Ainsi, $\mathbf{c}$ est contenu dans au moins $\binom{k}{d+1} \geqslant \frac{(d+1)!}{(d+1)^{2d+2}}\binom{n}{d+1}$ simplexes distincts de P. ■

Le théorème 5.3.5 est appelé *lemme de sélection*. Il est remarquable que la constante c_d soit indépendante non seulement de P mais aussi de sa taille. Ce phénomène a été découvert en 1984 par Endre BOROS et Zoltan FÜREDI ; ces derniers ont montré que $c_2 \geqslant \frac{1}{27}$. La valeur optimale de c_d vaut $\frac{2}{9}$ pour $d = 2$ et est inconnue pour tout $d \geqslant 3$. L'énoncé s'étend aux ensembles de points en position non-générique si l'on relaxe la notion de simplexe en «enveloppe convexe d'au plus $d+1$ points, pas nécessairement affinement indépendants».

5.4 Théorème de HELLY fractionnaire

Nous allons maintenant nous intéresser à une version fractionnaire du théorème de HELLY :

Théorème 5.4.1. *Pour tout $\alpha > 0$ et $d \geqslant 1$ il existe $\beta > 0$ tel que pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$, si une fraction α des sous-ensembles de $\mathcal{F}$ de taille $d+1$ sont d'intersection non vide, alors une fraction β de $\mathcal{F}$ est d'intersection non vide.*

Cette propriété a été découverte par Meir KATCHALSKI et A. LIU en 1979. Nous allons donner ici une preuve, élémentaire, que l'on peut prendre $\beta \geqslant \alpha/(d+1)$. Nous verrons en section 5.7.6 une preuve que l'on peut en fait prendre $\beta \geqslant 1 - (1-\alpha)^{\frac{1}{d+1}}$, ce qui s'avère optimal (la section 5.7.5 présente un exemple qui atteint cette borne).

5.4.1 Point maximum d'une intersection de convexes

Notons $\prec$ l'ordre lexicographique sur $\mathbb{R}^d$:

$$(x_1, x_2, \ldots, x_d) \prec (y_1, y_2, \ldots, y_d) \quad \Leftrightarrow \quad \begin{array}{l}\text{il existe } i \in [d] \text{ tel que } x_i < y_i \text{ et} \\ x_j = y_j \text{ pour tout } j < i.\end{array}$$

On note $\mathbf{x} \preceq \mathbf{y}$ le fait que $\mathbf{x} \prec \mathbf{y}$ ou $\mathbf{x} = \mathbf{y}$.

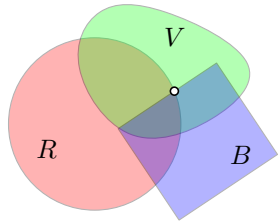


FIGURE 5.6 – *Trois convexes (R, V et B) dans le plan et le point maximum pour $\prec$ dans $R \cap V \cap B$. Noter que ce point est aussi maximum pour $\prec$ dans $R \cap B$.*

Pour tout point $\mathbf{p} \in \mathbb{R}^d$, posons $h_{\mathbf{p}} \stackrel{\text{déf}}{=} \{\mathbf{x} \in \mathbb{R}^d : \mathbf{p} \prec \mathbf{x}\}$ et remarquons que c'est un ensemble convexe (compris entre un demi-espace ouvert et sa clôture). Pour toute famille A de compacts convexes de $\mathbb{R}^d$ d'intersection non vide, notons $\max A$ le point de $\cap A$ maximal pour $\prec$.

Lemme 5.4.2. *Pour toute famille finie A de compacts convexes de $\mathbb{R}^d$ d'intersection non vide, il existe $B \subseteq A$ telle que $|B| \leqslant d$ et $\max A = \max B$.*

Démonstration. Pour toute famille C d'ensembles de $\mathbb{R}^d$ et tout point $\mathbf{q} \in \mathbb{R}^d$, $\max C \preceq \mathbf{q}$ si et seulement si $\cap C$ est disjoint de $h_{\mathbf{q}}$. Ainsi, $\max C$ est le point $\mathbf{q}$ minimal pour $\prec$ parmi ceux tels que $\cap C$ est disjoint de $h_{\mathbf{q}}$.

Revenons à notre famille A de convexes. Notons $\mathcal{F} = A \cup \{h_{\max A}\}$. Remarquons que $\mathcal{F}$ est une famille de convexes d'intersection vide. Par le théorème de HELLY, il existe une sous-famille $\mathcal{G} \subseteq \mathcal{F}$ de cardinal au plus $d+1$ telle que $\cap \mathcal{G}$ est vide. On ne peut pas avoir $\mathcal{G} \subseteq A$ puisque $\cap A$ est non vide. Par conséquent, $\mathcal{G}$ est de la forme $B \cup \{h_{\max A}\}$ où $B \subseteq A$.

Au final, on a $\max B \leqslant \max A$ car $\cap B$ ne coupe pas $h_{\max A}$, et $\max B \geqslant \max A$ car $\cap A \subseteq \cap B$. ∎

5.4.2 HELLY fractionnaire avec $\beta \geqslant \frac{\alpha}{d+1}$

Nous pouvons maintenant donner une preuve du théorème de HELLY fractionnaire dans le cas où les ensembles sont compacts.

Preuve du théorème 5.4.1 (cas compact). Chacun des $\alpha\binom{|\mathcal{F}|}{d+1}$ sous-ensembles A de $\mathcal{F}$ d'intersection non vide contient un sous-ensemble $\phi(A) \subseteq A$ de taille d tel que $\max A = \max \phi(A)$. Par le principe des tiroirs, il existe un sous-ensemble B de $\mathcal{F}$ de taille d tel que

$$\left|\left\{A \in \binom{\mathcal{F}}{d+1} : \phi(A) = B\right\}\right| \geqslant \frac{\alpha\binom{|\mathcal{F}|}{d+1}}{\binom{|\mathcal{F}|}{d}} = \frac{\alpha}{d+1}\left(|\mathcal{F}| - d\right).$$

Chacun des ensembles A tel que $\phi(A) = B$ consiste en l'ensemble B augmenté d'un autre élément ; notons ce dernier X_A. Si $\phi(A) = \phi(A')$ alors X_A et $X_{A'}$ sont distincts. Le point max B est donc contenu dans au moins $d + \frac{\alpha}{d+1}(|\mathcal{F}| - d) \geqslant \frac{\alpha}{d+1}|\mathcal{F}|$ éléments de $\mathcal{F}$. ■

Exercice 13 Utiliser le Lemme 5.4.2 pour prouver le théorème de HELLY coloré (Théorème 5.3.4) pour les familles d'ensembles compacts convexes.

5.5 Applications à la programmation linéaire

Notre deuxième application des outils de convexité combinatoire est algorithmique : nous allons voir qu'ils permettent d'analyser l'efficacité des méthodes d'échantillonnage pour la programmation linéaire.

5.5.1 Sous-programme d'un programme linéaire

Rappelons [6] qu'un problème linéaire de la forme

$$(PL) \qquad \begin{array}{ll} \min & \mathbf{c} \cdot \mathbf{x} \\ \text{t.q.} & A\mathbf{x} \leqslant \mathbf{b} \end{array} \tag{5.16}$$

met en jeu un vecteur $\mathbf{x}$ d'indéterminées, chacune à valeur dans $\mathbb{R}$, et demande de déterminer le point $\mathbf{x}$ qui minimise le produit scalaire $\mathbf{c} \cdot \mathbf{x}$ sous la contrainte que $A\mathbf{x} \leqslant \mathbf{b}$. Ici, A est une matrice, $\mathbf{b}$ un vecteur, et une inégalité entre points doit être vraie composante par composante. Chaque composante de l'inégalité $A\mathbf{x} \leqslant \mathbf{b}$ est appelée une *contrainte*.

On considère tout au long de la section 5.5 le programme linéaire (5.16) et on le suppose *faisable*. Autrement dit, l'ensemble des points $\mathbf{x}$ satisfaisant $A\mathbf{x} \leqslant \mathbf{b}$ est non vide. On suppose de plus $\mathbf{c}$ *générique* au sens où le minimum de $\mathbf{c} \cdot \mathbf{x}$ sur cet ensemble est atteint en un unique point.

Dans le reste de cette section 5.5, nous allons nous intéresser aux programmes linéaires définis par des sous-ensembles de contraintes de (5.16). Pour cela, on note $D_i \stackrel{\text{déf}}{=} \{\mathbf{x} : (A\mathbf{x})_i \leqslant b_i\}$ le demi-espace défini par la $i$ème ligne de A. On note $\mathcal{F} = \{D_1, D_2, \ldots, D_m\}$ l'ensemble des contraintes. Ainsi, notre programme linéaire revient à minimiser $\mathbf{c} \cdot \mathbf{x}$ sur $\cap\mathcal{F}$.

6. On suppose une certaine familiarité avec la programmation linéaire. Pour une introduction efficace à ce sujet, nous renvoyons par exemple au livre de MATOUŠEK et GÄRTNER [10].

Pour tout $R \subseteq \mathcal{F}$ on note :

$$\mathrm{val}(R) \stackrel{\text{déf}}{=} \min_{\cap R} \mathbf{c} \cdot \mathbf{x}. \tag{5.17}$$

Autrement dit, $\mathrm{val}(R)$ est la valeur du programme linéaire obtenu si on ne prend en compte que les contraintes de R, sans changer la fonction à minimiser. Remarquons que pour tous sous-ensembles A, B de $\mathcal{F}$,

$$A \subseteq B \quad \Rightarrow \quad \cap A \supseteq \cap B \quad \Rightarrow \quad \mathrm{val}(A) \leqslant \mathrm{val}(B). \tag{5.18}$$

La fonction val est donc croissante pour l'inclusion.

5.5.2 Test approximatif (mais rapide) de la valeur

Examinons une conséquence du théorème de HELLY fractionnaire. On définit qu'un programme linéaire est *ε-loin* d'avoir valeur $\leqslant v$ si il faut supprimer au moins une fraction ε de ses contraintes pour que sa valeur devienne $\leqslant v$. Autrement dit, $\mathcal{F}$ est *ε-loin* d'avoir valeur $\leqslant v$ si tout sous-ensemble $\mathcal{G} \subseteq \mathcal{F}$ tel que $\mathrm{val}(\mathcal{G}) \leqslant v$ est de taille au plus $(1-\varepsilon)|\mathcal{F}|$. Peut-on distinguer rapidement entre les programmes linéaires de valeur $\leqslant v$ et ceux qui sont ε-loin d'avoir valeur $\leqslant v$?

Ce type de question est étudié par le domaine appelé *property testing*, que l'on traduit ici par *test de propriété* : on souhaite savoir si l'objet $\mathcal{F}$ a la propriété d'avoir valeur au plus v. Un *ε-testeur* est un algorithme probabiliste qui a un accès direct aux éléments de la donnée (ici, il peut donc choisir une contrainte aléatoirement en temps constant) et qui doit répondre comme suit :

- Si la donnée d'entrée a la propriété, elle doit être acceptée par l'algorithme.
- Si la donnée d'entrée est ε-loin d'avoir la propriété, elle doit être rejetée par l'algorithme avec probabilité au moins $2/3$.

Soulignons que la probabilité d'erreur est uniquement liée à l'aléa interne de l'algorithme et doit être bornée pour *toute* donnée d'entrée. Le testeur peut librement accepter ou refuser les entrées qui n'ont pas la propriété mais sont proches de l'avoir.

Proposition 5.5.1. *L'algorithme 5.1 est un ε-testeur pour la propriété qu'un programme linéaire soit de valeur au plus v.*

Algorithme 5.1 : ε-testeur pour la propriété «avoir valeur $\leqslant v$».

1. Pour $i = 1$ à $\frac{-\log 3}{\log\left(1-\varepsilon^{d+1}\right)}$
2. Choisir $R \in \binom{\mathcal{F}}{d+1}$ aléatoire uniformément
3. Si $\mathrm{val}(R) > v$, refuser
4. accepter

Démonstration. Si $\mathcal{F}$ est de valeur $\leqslant v$ alors tout sous-programme de $\mathcal{F}$ est de valeur $\leqslant v$ et l'algorithme ne peut qu'accepter. Supposons $\mathcal{F}$ de valeur $> v$ et ε-loin d'être de valeur $\leqslant v$. Notons α la proportion des $R \in \binom{\mathcal{F}}{d+1}$ tels que $\mathrm{val}(R) \leqslant v$ et notons $k \stackrel{\text{déf}}{=} \frac{-\log 3}{\log\left(1-\varepsilon^{d+1}\right)}$ le nombre d'échantillons examinés par l'algorithme 5.1. La probabilité que $\mathcal{F}$ soit acceptée par erreur est au plus α^k.

Notons $h \stackrel{\text{déf}}{=} \{\mathbf{x} \colon \mathbf{v} \cdot \mathbf{x} \leqslant v\}$. Pour tout $S \subseteq \mathcal{F}$, on a

$$\mathrm{val}(S) \leqslant v \quad \Leftrightarrow \quad (\cap S) \cap h \neq \emptyset.$$

Notons $\phi(D) = D \cap h$. On a ainsi que $\mathrm{val}(S) \leqslant v$ si et seulement si $\cap\phi(S)$ est non vide. Ainsi, α égale la proportion des sous-ensembles de taille $d+1$ de $\phi(\mathcal{F})$ qui sont d'intersection non vide.

D'après le théorème de HELLY fractionnaire fort (5.7.7), il existe un sous-ensemble $\phi(\mathcal{G}) \subseteq \phi(\mathcal{F})$ de taille $\left(1 - (1-\alpha)^{\frac{1}{d+1}}\right)|\mathcal{F}|$ qui est d'intersection non vide. Ainsi, $\mathrm{val}(\mathcal{G}) \leqslant r$ et, par conséquent, on doit avoir

$$1 - (1-\alpha)^{\frac{1}{d+1}} = \frac{|\mathcal{G}|}{|\mathcal{F}|} \leqslant 1 - \varepsilon.$$

On a donc $\alpha \leqslant 1 - \varepsilon^{d+1}$ et $\alpha^k \leqslant \frac{1}{3}$. ∎

Naturellement, un tel testeur n'est intéressant que quand la dimension d est très petite devant le nombre de contraintes.

Exercice 14 Proposer un testeur pour la propriété qu'un ensemble de points de $\mathbb{R}^d$ soit contenu dans une boule de rayon r. On dira qu'un ensemble P de points de $\mathbb{R}^d$ est ε-loin d'être contenu dans une boule de rayon r si toute boule de rayon r contient au plus $(1-\varepsilon)|P|$ points de P.

Exercice 15 Proposer un testeur pour la propriété que deux ensembles de points P et Q dans $\mathbb{R}^d$ soient séparables par un hyperplan. On dira que des ensembles P et Q de points de $\mathbb{R}^d$ sont ε-loin d'être séparables si tous sous-ensembles $P' \subseteq P$ et $Q' \subseteq Q$ séparables par un hyperplan sont de taille $|P' \cup Q'| \leqslant (1-\varepsilon)|P \cup Q|$. *(Indication : revoir l'exercice 11.)*

5.5.3 Contraintes extrêmes d'un sous-programme

À tout ensemble de contraintes $R \subseteq \mathcal{F}$ on associe l'ensemble

$$X(R) \stackrel{\text{déf}}{=} \{h \in R \colon \operatorname{val}(R \setminus \{h\}) \neq \operatorname{val}(R)\}. \tag{5.19}$$

C'est le sous-ensemble des contraintes de R dont on est sûr qu'elles ne sont pas inutiles ; elles sont dites *extrêmes* dans R. Le théorème de HELLY permet de borner le nombre de contraintes extrêmes.

Proposition 5.5.2. *Pour tout $R \subseteq \mathcal{F}$, $|X(R)| \leqslant d$.*

Démonstration. Le programme linéaire étant faisable, $\cap R$ est non vide. Pour tout $t \in \mathbb{R}$, notons $h_t \stackrel{\text{déf}}{=} \{\mathbf{x} \colon \mathbf{c} \cdot \mathbf{x} \leqslant t\}$. Remarquons que pour tout $S \subseteq \mathcal{F}$

$$\operatorname{val}(S) \leqslant t \quad \Leftrightarrow \quad h_t \cap (\cap S) \neq \emptyset. \tag{5.20}$$

Ainsi, par le théorème de HELLY, pour tout t tel que $\operatorname{val}(R) > t$, il existe $d+1$ éléments de $R \cup \{h_t\}$ d'intersection vide. Comme $\cap R$ est non vide, h_t fait partie de ces $d+1$ éléments ; notons S_t l'ensemble des d autres contraintes.

Pour tous réels $t < t'$, on a $h_t \subseteq h_{t'}$ et donc $h_t \cap (\cap S_{t'})$ est vide. L'ensemble $\left\{S_{\operatorname{val}(R)-\frac{1}{k}}\right\}_{k \in \mathbb{N}^*}$ prend un ensemble fini de valeurs distinctes. Au moins une de ces valeurs est répétée une infinité de fois ; notons-la S. D'une part, cette construction assure que $h_t \cap (\cap S) = \emptyset$ pour tout $t < \operatorname{val}(R)$; cela implique que $\operatorname{val}(S) \geqslant \operatorname{val}(R)$. D'autre part, l'inclusion $\cap S \supset \cap R$ assure que $\operatorname{val}(S) \leqslant \operatorname{val}(R)$.

Ainsi, $\operatorname{val}(S) = \operatorname{val}(R)$. Comme val est croissante pour l'inclusion (5.18), pour tout $h \in R \setminus S$ on a $\operatorname{val}(S) \leqslant \operatorname{val}(R \setminus \{h\}) \leqslant \operatorname{val}(R)$ et donc $\operatorname{val}(R \setminus \{h\}) = \operatorname{val}(R)$. Ainsi, seules les contraintes de S peuvent être extrêmes pour R. ■

Tant que notre programme linéaire contient plus de d contraintes, l'une d'elles n'est pas extrême et peut être effacée. On obtient ainsi :

Corollaire 5.5.3. *Tout programme linéaire faisable à d variables admet un sous-programme à d contraintes qui a même valeur.*

Exercice 16 Donner un exemple de programme linéaire (en dimension $d = 2$) pour lequel $X(\mathcal{F})$ est vide.

Exercice 17 Donner un analogue du corollaire 5.5.3 pour les programmes linéaires infaisables.

5.5.4 Contraintes violées par un sous-programme

Pour analyser le comportement de sous-programmes d'un programme linéaire $\mathcal{F}$, il est utile d'associer à tout sous-ensemble $R \subseteq \mathcal{F}$ l'ensemble

$$V(R) \stackrel{\text{déf}}{=} \{h \in \mathcal{F} \setminus R \colon \text{val}(R \cup \{h\}) \neq \text{val}(R)\} \tag{5.21}$$

On l'appelle *ensemble des contraintes violées* par R. Géométriquement, si $\mathbf{x}^*$ est le [7] point réalisant le minimum de $\mathbf{c} \cdot \mathbf{x}$ sur $\cap R$, alors $V(R)$ est l'ensemble des contraintes qui ne contiennent pas $\mathbf{x}^*$. Ajouter les contraintes de $\mathcal{F} \setminus V(R)$ une à une à R ne change donc pas l'optimum et l'on a

$$\text{val}(R) = \text{val}\left(R \cup (\mathcal{F} \setminus V(R))\right). \tag{5.22}$$

En particulier, si $V(R)$ est vide alors $\text{val}(R) = \text{val}(\mathcal{F})$ et les contraintes de $\mathcal{F} \setminus R$ sont inutiles dans notre programme linéaire de départ.

Contrairement aux ensembles de contraintes extrêmes, les ensembles de contraintes violées peuvent être arbitrairement grands. En revanche, en *moyenne* les tailles des ensembles de contraintes extrêmes et de contraintes violées sont liées. Notons S_r un sous-ensemble $\mathcal{F}$ choisi uniformément parmi tous les sous-ensembles de $\mathcal{F}$ de taille r.

Lemme 5.5.4. *Pour tout $r \geqslant 1$ on a*

$$\mathbb{E}\left[V(S_r)\right] = \frac{|\mathcal{F}| - r}{r+1}\, \mathbb{E}\left[X(S_{r+1})\right].$$

Noter que les expériences aléatoires à gauche et à droite sont différentes : l'une porte sur S_r et l'autre sur S_{r+1}. Ce lemme appelé *sampling lemma*, a été identifié par Berndt GÄRTNER et Emo WELZL en 2000.

7. Ici, on se sert de l'hypothèse que $\mathbf{c}$ est générique.

Preuve du lemme 5.5.4. On a

$$\begin{aligned}\binom{|\mathcal{F}|}{r}\mathbb{E}\left[V(S_r)\right] &= \sum_{A\in\binom{\mathcal{F}}{r}}\sum_{h\in\mathcal{F}\setminus A}\mathbb{1}_{h\in V(A)}\\ &= \sum_{A\in\binom{\mathcal{F}}{r}}\sum_{h\in\mathcal{F}\setminus A}\mathbb{1}_{h\in X(A\cup\{h\})}\\ &= \sum_{B\in\binom{\mathcal{F}}{r+1}}\sum_{h\in B}\mathbb{1}_{h\in X(B)} = \binom{|\mathcal{F}|}{r+1}\mathbb{E}\left[X(S_{r+1})\right].\end{aligned}$$

L'identité annoncée s'en déduit. ■

Exercice 18 Montrer que pour tous sous-ensembles R, S de $\mathcal{F}$, si $R \subseteq S$ et si S et $V(R)$ sont disjoints, alors $V(S) = V(R)$.

5.5.5 Algorithme de repondération itérée

Le corollaire 5.5.3 énonce que quand m est sensiblement plus grand que d, la plupart des contraintes sont inutiles. Nous allons maintenant examiner un algorithme dit de *repondération itérée* (en anglais : *iterative reweighting*) pour simplifier efficacement un programme linéaire. Il a été proposé en 1995 par Kenneth CLARKSON.

L'algorithme

L'algorithme de CLARKSON produit des échantillons R_1, R_2, … de taille r (paramètre à déterminer) de l'ensemble $\mathcal{F}$ des contraintes. L'échantillon R_1 est choisi uniformément. Une fois R_i choisi, on calcule $V(R_i)$. Si cet ensemble est vide, l'algorithme termine. Sinon, on biaise la distribution sur $\mathcal{F}$ de sorte à favoriser la sélection des contraintes violées par R_i, et on prend l'échantillon suivant.

Pour formaliser cela, on associe à chaque contrainte $D_i \in \mathcal{F}$ un *poids* w_i, initialisé à 1. Pour $S \subseteq \mathcal{F}$, notons $w(S)$ la somme des poids des contraintes de S. La *loi induite* par les poids $w_\bullet$ sur les sous-ensembles de taille r de $\mathcal{F}$ assigne à tout $A \in \binom{\mathcal{F}}{r}$ la probabilité

$$\Pr\left[R = A\right] = \frac{w(A)}{\sum_{A'\in\binom{\mathcal{F}}{r}} w(A')}. \tag{5.23}$$

Voici l'algorithme en détail :

Algorithme 5.2 : Repondération itérée de paramètres r et α.

1. Choisir un sous-ensemble aléatoire R de $\mathcal{F}$ de taille r selon la loi de probabilité donnée par les poids $w_\bullet$.
2. Si $V(R) = \emptyset$ retourner R
3. Si $w(V(R)) > \alpha w(\mathcal{F})$ retourner en 1.
4. Sinon, doubler w_i pour chaque $D_i \in V(R)$ et retourner en 1.

Théorème 5.5.5. *Pour $r = 4d^2$ et $\alpha = \frac{1}{2d}$, la probabilité que l'algorithme* (5.2) *nécessite plus de $4d \log \frac{m}{d} + k$ échantillonnages est au plus $e^{-\frac{k^2}{16d \log \frac{m}{d} + 4k}}$.*

Vue d'ensemble de l'analyse

Choisissons un sous-ensemble $\mathcal{G} \subseteq \mathcal{F}$ de taille $d+1$ tel que $\mathrm{val}(\mathcal{G}) = \mathrm{val}(\mathcal{F})$. Son existence est garantie par la Proposition 5.5.2. Appelons *repondération* un passage par l'étape 4 de l'algorithme. La preuve commence par noter le fait suivant :

Fait 1. *Chaque étape de repondération double le poids d'au moins une des contraintes de $\mathcal{G}$.*

Comparons les poids $w(\mathcal{F})$ et $w(\mathcal{G})$ après la $k$ème repondération. D'une part, l'étape 3 garantit que $w(\mathcal{F})$ est *au plus* $m(1+\alpha)^k$. D'autre part, $w(\mathcal{G})$ est *au moins* $d2^{\frac{k}{d}}$. En effet, le nombre de fois où une contrainte de $\mathcal{G}$ a vu son poids doubler est au moins k, et l'effet de ces doublements est minimisé quand ils se répartissent équitablement entre les contraintes de $\mathcal{G}$.

L'inclusion $\mathcal{G} \subseteq \mathcal{F}$ assure que $w(\mathcal{G})$ ne peut dépasser $w(\mathcal{F})$. Pourtant, en fixant $\alpha = \frac{2}{d}$, on a $1 + \alpha < 2^{\frac{1}{d}}$ et $w(\mathcal{G})$ croît plus rapidement que $w(\mathcal{F})$. Cela garantit que le nombre de repondérations effectuées par l'algorithme est borné, en l'occurrence par

$$\ell^* \leqslant \frac{\log_2 \frac{m}{d}}{\frac{1}{d} - \log_2(1+\alpha)} \leqslant 2d \log_2 \frac{m}{d}. \tag{5.24}$$

Il reste à contrôler la probabilité que l'examen d'un échantillon donne lieu à une repondération, c'est à dire la probabilité de ne pas retourner en 1 à l'étape 3.

Pour analyser la loi de $w(V(R))$, une première étape consiste à raffiner le lemme 5.5.4 :

Fait 2. *Soit S_r un sous-ensemble aléatoire de $\mathcal{F}$ de taille r choisi selon la loi de probabilité* (5.23) *induite par les poids $\{w_i\}$.*

$$\mathbb{E}\left[w(V(S_r))\right] \leqslant \frac{|\mathcal{F}|-r}{r}\mathbb{E}\left[w(X(S_{r+1}))\right]. \tag{5.25}$$

Avec la proposition 5.5.2, ceci garantit que

$$\mathbb{E}\left[w(V(R))\right] \leqslant \frac{|\mathcal{F}|-r}{r}d. \tag{5.26}$$

Par l'inégalité de MARKOV, on a donc

$$\Pr\left[w(V(R)) > \alpha w(\mathcal{F})\right] \leqslant \left(\frac{|\mathcal{F}|-r}{w(\mathcal{F})}\right)\frac{d}{r\alpha}. \tag{5.27}$$

Choisir $r = 4d^2$ assure que chaque échantillonnage donne lieu à une repondération avec probabilité au moins $\frac{1}{2}$ (et, vraisemblablement, beaucoup plus). Notons $\ell(s)$ le nombre de repondérations provoquées par s échantillonnages. Les tirages étant indépendants, l'inégalité de CHERNOFF s'applique :

$$\Pr\left[\ell(s) < \frac{s}{2} - t\right] \leqslant e^{-\frac{t^2}{s}} \tag{5.28}$$

En prenant $s = 4d\log\frac{m}{d} + k$ et $\frac{s}{2} - t = 2d\log\frac{m}{d}$ on obtient

$$\Pr\left[\ell\left(4d\log\frac{m}{d}+k\right) < 2d\log\frac{m}{d}\right] \leqslant e^{-\frac{k^2}{16d\log\frac{m}{d}+4k}}. \tag{5.29}$$

Dans le détail...

Preuve du fait 1. La monotonie (5.18) de la fonction val pour l'inclusion assure que

$$\mathrm{val}(\mathcal{F}) = \mathrm{val}(\mathcal{G}) \leqslant \mathrm{val}(R\cup\mathcal{G}) \leqslant \mathrm{val}(\mathcal{F})$$

et donc $\mathrm{val}(R\cup\mathcal{G}) = \mathrm{val}(\mathcal{F})$. Par ailleurs, si $V(R)\cap\mathcal{G}$ est vide, alors la monotonie et l'égalité (5.22) assure que

$$\mathrm{val}(R) \leqslant \mathrm{val}(R\cup\mathcal{G}) \leqslant \mathrm{val}\left(R\cup(\mathcal{F}\setminus V(R))\right) = \mathrm{val}(R)$$

et donc $\mathrm{val}(R\cup\mathcal{G}) = \mathrm{val}(R)$. Ainsi, si $V(R)\cap\mathcal{G}$ est vide, alors $\mathrm{val}(R) = \mathrm{val}(\mathcal{F})$ et la monotonie de val assure que $V(R)$ est vide. ■

Preuve du fait 2. La somme des poids de tous les sous-ensembles de taille r de $\mathcal{F}$ s'écrit

$$\sum_{A\in\binom{\mathcal{F}}{r}} w(A) = \sum_{h\in\mathcal{F}}\binom{|\mathcal{F}|-1}{r-1}w(h) = \binom{|\mathcal{F}|-1}{r-1}w(\mathcal{F}).$$

On a donc

$$\begin{aligned}\mathbb{E}\left[w(V(S_r))\right] &= \sum_{A\in\binom{\mathcal{F}}{r}} \frac{w(A)}{\binom{|\mathcal{F}|-1}{r-1}w(\mathcal{F})} w(V(A)) \\ &= \sum_{A\in\binom{\mathcal{F}}{r}} \sum_{h\in\mathcal{F}\setminus A} \frac{w(A)}{\binom{|\mathcal{F}|-1}{r-1}w(\mathcal{F})} \left(w(h)\mathbb{1}_{h\in V(A)}\right) \\ &= \sum_{B\in\binom{\mathcal{F}}{r+1}} \sum_{h\in B} \frac{w(B\setminus\{h\})}{\binom{|\mathcal{F}|-1}{r-1}w(\mathcal{F})} \left(w(h)\mathbb{1}_{h\in X(B)}\right)\end{aligned}$$

En utilisant que $\binom{n-1}{r-1} = \binom{n-1}{r}\frac{r}{n-r}$ on obtient

$$\begin{aligned}\mathbb{E}\left[w(V(R))\right] =& \frac{|\mathcal{F}|-r}{r} \sum_{B\in\binom{\mathcal{F}}{r+1}} \sum_{h\in B} \frac{w(B\setminus\{h\})}{\binom{|\mathcal{F}|-1}{r}w(\mathcal{F})} \left(w(h)\mathbb{1}_{h\in X(B)}\right) \\ \leqslant& \frac{|\mathcal{F}|-r}{r} \sum_{B\in\binom{\mathcal{F}}{r+1}} \frac{w(B)}{\binom{|\mathcal{F}|-1}{r}w(\mathcal{F})} w(X(B)) \\ =& \frac{|\mathcal{F}|-r}{r}\mathbb{E}\left[w(X(S_{r+1}))\right]\end{aligned}$$

■

5.5.6 Généralisation à d'autres problèmes d'optimisation

L'analyse que nous venons de présenter se généralise au-delà du cadre convexe. Soit $\mathcal{F}$ une famille d'ensembles (dans $\mathbb{R}^d$, des entiers, des listes de restaurants favoris, ...) telle que $\cap\mathcal{F} = \emptyset$. On définit le *nombre de* HELLY de $\mathcal{F}$ comme la taille de la plus grande sous-famille de $\mathcal{F}$ d'intersection vide et minimale pour cette propriété. Le nombre de HELLY s'avère borné dans de nombreuses situations, par exemple :

- Un sous-ensemble de $\mathbb{Z}^d$ est $\mathbb{Z}$-*convexe* si c'est l'intersection de $\mathbb{Z}^d$ avec un convexe de $\mathbb{R}^d$. Toute famille de $\mathbb{Z}$-convexes a nombre de HELLY au plus 2^d.
- Soit C une courbe convexe du plan. Si $\mathcal{F}$ est une famille finie de copies de C par des translations et des homothéties, alors le nombre de HELLY de $\mathcal{F}$ est au plus 4.
- Soit C un polygone simple du plan et ∂C son bord. Pour tout point $\mathbf{x} \in \partial C$, notons $V_{\mathbf{x}}$ l'ensemble des points de C visibles depuis $\mathbf{x}$. Notons $\mathcal{F} = \{V_{\mathbf{x}} : \mathbf{x} \in \partial C\}$. Le nombre de HELLY de $\mathcal{F}$ est au plus 3.

Au cours des 100 dernières années, ce sont des centaines de théorèmes «à la HELLY» de ce type qui ont été découverts.

Un examen attentif révèle que les applications que nous avons détaillées pour la programmation linéaire se généralisent à *tout* problème de calcul de

$$\min_{\mathbf{x}\in D_1\cap D_2\cap\ldots\cap D_n} f(\mathbf{x})$$

où $f\colon \mathbb{R}^d \to \mathbb{R}$ et $D_i \subseteq \mathbb{R}^d$, à la condition que les familles

$$\mathcal{F}_t = \{D_i \cap f^{-1}((-\infty, t)) : 1 \leqslant i \leqslant n\}$$

aient nombre de HELLY (fractionnaire) borné uniformément en t. En particulier on ne suppose ni convexité, ni continuité de f, ni structure particulière des D_i : la seule chose qui compte [8] est que les nombres de HELLY soient bornés.

Exercice 19 Adapter l'algorithme de repondération itérée pour calculer le plus petit cercle contenant n points de $\mathbb{R}^2$ donnés en entrée. Adapter l'analyse de complexité à ce cadre. Ce problème (le calcul du cercle englobant minimum) peut-il se modéliser par un programme linéaire ?

Transition

Nous abordons maintenant la seconde partie de ce cours, où nous glissons de la convexité à la topologie, la combinatoire topologique et la combinatoire extrémale. Nous organisons cette seconde partie en trois sections traitant chacune d'une structure combinatorio-géométrique.

5.6 Complexes simpliciaux géométriques

La première structure que nous examinons est le complexe simplicial géométrique. Nous allons voir que cet objet permet de généraliser en dimension quelconque la notion de «graphe planaire» largement étudiée en théorie géométrique des graphes. Nous introduisons au passage le théorème de BORSUK-ULAM, résultat fondamental en topologie (algébrique) qui trouve de multiples applications en combinatoire et en géométrie. Nous en déduisons notamment des généralisations topologiques du lemme de RADON et du théorème de HELLY.

8. Il faut bien entendu pouvoir les manipuler en pratique... Les seules opérations requises par les méthodes présentées sont de pouvoir résoudre des sous-problèmes de taille constante et de pouvoir tester les violations.

5.6.1 Définition et terminologie

Rappelons qu'un simplexe de $\mathbb{R}^d$ est l'enveloppe convexe de points de $\mathbb{R}^d$ affinement indépendants, ses sommets, et que la dimension d'un simplexe est son nombre de sommets moins 1. Une *face* d'un simplexe σ est l'enveloppe convexe d'un sous-ensemble des sommets de σ. Toute face d'un simplexe est donc elle aussi un simplexe.

Un *complexe simplicial géométrique* de $\mathbb{R}^d$ est une famille $\mathcal{K}$ de simplexes de $\mathbb{R}^d$ qui vérifie :

$$\forall A, B \in \mathcal{K}, \quad A \cap B \in \mathcal{K}, \tag{5.30}$$

et

$$\forall A \in \mathcal{K}, \quad \forall B \text{ face de } A, \quad B \in \mathcal{K}. \tag{5.31}$$

Autrement dit, un complexe simplicial est un ensemble de simplexes qui est stable par les opérateurs «intersection» et «prendre une face».

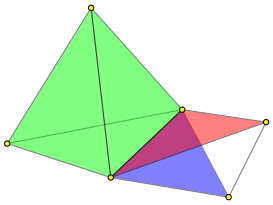


FIGURE 5.7 – *Un exemple de complexe simplicial géométrique comportant 1 tétraèdre, 6 triangles, 11 arêtes et 6 sommets. (La face rouge et la face bleue ne sont pas coplanaires.)*

Prenons un exemple. Soit $P = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_k\}$ un ensemble de points affinement indépendants de $\mathbb{R}^d$. Le fait que $\text{conv}(P \cap Q) = \text{conv}(P) \cap \text{conv}(Q)$ si $P \cup Q$ est affinement indépendant (*c.f.* exercice 6) assure que l'ensemble $\mathcal{K}$ des faces de $\text{conv}(P)$ est un complexe simplicial géométrique.

Soit $\mathcal{K}$ un complexe simplicial géométrique de $\mathbb{R}^d$. Un élément de $\mathcal{K}$ est appelé une *face* de $\mathcal{K}$. Une face de $\mathcal{K}$ de dimension 0 est appelée *sommet* de $\mathcal{K}$. La *dimension* de $\mathcal{K}$ est la dimension maximale d'une de ses faces. Le *polyèdre* de $\mathcal{K}$, noté $\|\mathcal{K}\|$, est le sous-ensemble de $\mathbb{R}^d$ formé par l'union de ses faces :

$$\|\mathcal{K}\| \stackrel{\text{déf}}{=} \bigcup_{\sigma \in \mathcal{K}} \sigma. \tag{5.32}$$

(On peut limiter l'union aux faces maximales pour l'inclusion.)

FIGURE 5.8 – *Deux complexes simpliciaux géométriques de même polyèdre. Celui de droite comporte 3 tétraèdres, 12 triangles, 17 arêtes et 8 sommets. (Les faces rouges et les faces bleues ne sont pas coplanaires.)*

5.6.2 Reformulation d'énoncés de convexité combinatoire

Notons Δ_n le simplexe de dimension n obtenu comme l'enveloppe convexe dans $\mathbb{R}^n$ de $(0,\dots,0)$, $(1,0,\dots,0)$, $(0,1,0,\dots,0)$, …, $(0,\dots,0,1)$. Notons $\Delta_n^{(k)}$ le complexe simplicial géométrique composé de l'ensemble des faces de dimension au plus k de Δ_n. Ainsi, le polyèdre de $\Delta_{d+1}^{(d)}$ est le bord du simplexe Δ_{d+1}.

Rappelons que deux simplexes sont indépendants s'ils n'ont aucun sommet commun. Revoici le lemme de RADON :

Théorème 5.6.1 (5.2.2 reformulé)**.** *Pour tout $d \geqslant 1$ et pour toute application linéaire $f : \|\Delta_{d+1}^{(d)}\| \to \mathbb{R}^d$, il existe un point de $\mathbb{R}^d$ appartenant aux images de 2 faces indépendantes de $\|\Delta_{d+1}^{(d)}\|$.*

De même, revoici le lemme de sélection :

Théorème 5.6.2 (5.3.5 reformulé)**.** *Pour tout entier $d \geqslant 2$, il existe une constante $c_d > 0$ telle que pour toute application linéaire $f : \|\Delta_{n-1}^{(d)}\| \to \mathbb{R}^d$, il existe un point de $\mathbb{R}^d$ appartenant aux images de $c_d\binom{n}{d+1}$ faces indépendantes de $\|\Delta_{n-1}^{(d)}\|$.*

Ces reformulations font apparaître une question naturelle : les énoncés ci-dessus admettent-ils des *généralisations topologiques* au sens où l'on peut remplacer «application linéaire» par «application continue» ?

La réponse s'avère positive dans les deux cas : un lemme de Radon topologique a été établi par E.G BAJMÓCZY et Imre BÁRÁNY en 1979 et un lemme de sélection topologique a été établi par Mikhail GROMOV en 2010. Nous allons détailler le premier (théorème 5.6.7).

5.6.3 Théorème de BORSUK-ULAM

Un outil fondamental pour étudier les applications continues d'un complexe simplicial géométrique dans $\mathbb{R}^d$ est le théorème suivant, dû à Karol BORSUK et Stanislaw ULAM. On note $\mathbb{S}^d \stackrel{\text{déf}}{=} \{\mathbf{x} \in \mathbb{R}^{d+1} : \|\mathbf{x}\|_2 = 1\}$ le bord de la boule unité de $\mathbb{R}^{d+1}$; c'est une sphère de dimension d. Une application $f : \mathbb{S}^d \to \mathbb{R}^k$ est *antipodale* si $f(-\mathbf{x}) = -f(\mathbf{x})$ pour tout $\mathbf{x} \in \mathbb{S}^d$.

Théorème 5.6.3. *Pour tous entiers $1 \leqslant k < d$, il n'existe pas d'application continue antipodale de $\mathbb{S}^d$ dans $\mathbb{S}^k$.*

Nous ne prouvons pas ce résultat, et renvoyons le lecteur intéressé à l'ouvrage de MATOUŠEK [8] dédié à ce théorème et ses applications en (géométrie) combinatoire. Signalons toutefois que cet énoncé a un analogue combinatoire, appelé *lemme de* TUCKER, et est équivalent aux formulations suivantes :

(a) Pour toute fonction continue $f : \mathbb{S}^d \to \mathbb{R}^d$ il existe un point $\mathbf{x} \in \mathbb{S}^d$ tel que $f(\mathbf{x}) = f(-\mathbf{x})$.

(b) Pour toute fonction continue et antipodale $f : \mathbb{S}^d \to \mathbb{R}^d$ il existe un point $\mathbf{x} \in \mathbb{S}^d$ tel que $f(\mathbf{x}) = \mathbf{0}$.

5.6.4 Premières applications de BORSUK-ULAM

Pour tout point $\mathbf{v}$ de $\mathbb{R}^d$ et tout réel $\lambda \in \mathbb{R}$, notons $\mathbf{x} \cdot \mathbf{v} = \lambda$ l'hyperplan formé par les points $\mathbf{x} \in \mathbb{R}^d$ satisfaisant cette équation. Étant donnée une fonction $f : \mathbb{S}^{d-1} \to \mathbb{R}$, notons H_f la famille des hyperplans $\mathbf{x} \cdot \mathbf{v} = f(\mathbf{v})$ pour tout $\mathbf{v} \in \mathbb{S}^{d-1}$. Lorsque la fonction f est continue et antipodale, la famille H_f est appelée un *système d'hyperplans*. Informellement, c'est une famille d'hyperplans qui contient, pour toute direction, un unique hyperplan de normale cette direction, et tel que l'hyperplan dépend continûment de la direction.

Le théorème de BORSUK-ULAM a la conséquence suivante, découverte par Vladimir DOL'NIKOV, sur les systèmes d'hyperplans.

Lemme 5.6.4. *Toute famille de d systèmes d'hyperplans de $\mathbb{R}^d$ a au moins un hyperplan commun.*

Démonstration. Notons $f_1, f_2, \ldots, f_d$ les fonctions continues et antipodales de $\mathbb{S}^d$ dans $\mathbb{R}$ qui définissent chacun des systèmes d'hyperplans. Introduisons la fonction $\phi = (f_1 - f_d, f_2 - f_d, \ldots, f_{d-1} - f_d)$. Remarquons que ϕ est une application continue et antipodale de $\mathbb{S}^{d-1}$ dans $\mathbb{R}^{d-1}$. D'après la variante (b) du théorème de BORSUK-ULAM, il existe une direction $\mathbf{v} \in \mathbb{S}^{d-1}$

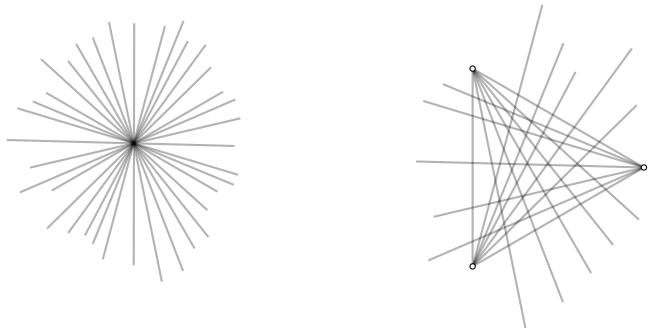


FIGURE 5.9 – *Exemples de systèmes de droites dans* $\mathbb{R}^2$ *: (gauche) les droites passant par un point, (droite) les droites passant par un sommet d'un triangle et coupant le côté opposé.*

telle que $\phi(\mathbf{v}) = \mathbf{0}$. Ainsi, $f_1(\mathbf{v}) = f_2(\mathbf{v}) = \ldots = f_d(\mathbf{v})$ et chacun des systèmes d'hyperplan contient $\mathbf{x} \cdot \mathbf{v} = f_d(\mathbf{v})$. ■

Une première conséquence du lemme de DOL'NIKOV est le célèbre *théorème du sandiwch au jambon*.

Théorème 5.6.5. *Soient* $\mu_1, \mu_2, \ldots, \mu_d$ *des mesures finies de* $\mathbb{R}^d$ *telles que* $\mu_i(h) = 0$ *pour tout hyperplan* h *et tout* $i \in [d]$*. Il existe un hyperplan* h *tel que*

$$\forall i \in [d], \quad \mu_i\left(h^+\right) = \frac{1}{2}\mu_i(\mathbb{R}^d), \tag{5.33}$$

où h^+ *est l'un des espaces fermés bordé par* h*.*

Démonstration. Fixons $i \in [d]$ et $\mathbf{v} \in \mathbb{S}^{d-1}$. Lorsque l'on fait varier t de $-\infty$ à $+\infty$, la μ_i-mesure du demi-espace $\mathbf{x} \cdot \mathbf{v} \leqslant t$ varie continûment de 0 à $\mu_i(\mathbb{R}^d)$. D'après le théorème des valeurs intermédiaires, il existe $t(i, \mathbf{v})$ tel que l'hyperplan $\mathbf{x} \cdot \mathbf{v} = t(i, \mathbf{v})$ satisfasse la condition (5.33) pour i. Posons $f_i(\mathbf{v}) \overset{\text{déf}}{=} t(i, \mathbf{v})$. Chaque fonction f_i est continue et antipodale. Le lemme 5.6.4 appliqué aux systèmes d'hyperplans $H_{f_1}, H_{f_2}, \ldots, H_{f_d}$ donne le résultat annoncé. ■

La seconde application que nous examinons porte sur les nombres chromatiques de graphes. Le *graphe de* KNESER de paramètre n, k a pour sommets les sous-ensembles de taille k de $[n]$ et pour arêtes les paires de sous-ensembles disjoints. L'énoncé suivant a été conjecturé par Martin KNESER en 1955 et prouvé en 1978 par Lázló LOVÁSZ. La preuve qui suit est due à Vladimir DOL'NIKOV.

Théorème 5.6.6. *Pour tous entiers $n \geqslant 2k \geqslant 0$, dans toute coloration des éléments de $\binom{[n]}{k}$ par $n-2k+1$ couleurs, il existe deux éléments disjoints (comme sous-ensembles de $[n]$) et de même couleur.*

Démonstration. Notons $d \stackrel{\text{déf}}{=} n-2k+1$ et Notons $\binom{[n]}{k} = \{X_1, X_2, \ldots, X_m\}$ où $m = \binom{n}{k}$. supposons que l'énoncé soit faux, c'est-à-dire qu'il existe une coloration $c : \binom{[n]}{k} \to [d]$ tels que pour tous $X_i, X_j \in \binom{[n]}{k}$, $c(X_i) = c(X_j) \Rightarrow X_i \cap X_j \neq \emptyset$.

Considérons un ensemble $P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$ de points de $\mathbb{R}^d$ en position générique. Notons $A_i \stackrel{\text{déf}}{=} \text{conv}\{\mathbf{p}_x : x \in X_i\}$ et appelons $c(X_i)$ la *couleur* de A_i. Ainsi, chaque A_i est un simplexe de dimension $k-1$ et si deux A_i ont même couleur, ils doivent avoir au moins un sommet commun. Notons $\mathcal{F} = \{A_1, A_2, \ldots, A_m\}$.

Fixons $\mathbf{u} \in \mathbb{S}^{d-1}$ et une couleur $\alpha \in [d]$. Considérons les projections orthogonales sur la droite $\mathbb{R}\mathbf{u}$ des A_i de couleur α. Les intervalles obtenus se coupent deux à deux, donc d'après le théorème de HELLY, ils se coupent tous. L'intersection de ces intervalles est donc un intervalle borné et non vide ; soit $f_\alpha(\mathbf{u})$ tel que $f_\alpha(\mathbf{u})\mathbf{u}$ soit le milieu de cet intervalle.

Chaque fonction f_α est continue et antipodale, et définit donc un système d'hyperplans. De plus, chaque hyperplan $\mathbf{x} \cdot \mathbf{v} = f_\alpha(\mathbf{v})$ coupe tous les A_i de couleur α. Le lemme 5.6.4 garantit donc qu'il existe un hyperplan h qui coupe *tous* les A_i pour $1 \leqslant i \leqslant m$. Chacun des demi-espaces ouverts bordé par h contient moins de k points de P, sans quoi il contiendrait un A_i qui éviterait donc h. L'hyperplan h de $\mathbb{R}^d$ contient donc au moins $n-2k = d+1$ points, ce qui contredit l'hypothèse que P est en position générale. ■

5.6.5 Théorèmes de RADON et HELLY topologiques

Examinons maintenant de quelle manière le théorème de BORSUK-ULAM permet de généraliser le lemme de RADON.

Théorème 5.6.7. *Pour $d \geqslant 1$ et toute application continue $f : \|\Delta^{(d)}_{d+1}\| \to \mathbb{R}^d$, il existe un point de $\mathbb{R}^d$ couvert par les images de deux faces indépendantes.*

Remarquons que $\|\Delta^{(d)}_{d+1}\|$ est le bord d'un simplexe de dimension $d+1$ et est donc homéomorphe à $\mathbb{S}^d$. Le théorème 5.6.7 est donc proche de la variante (a) du théorème de BORSUK-ULAM, mais ajoute la contrainte que les faces supportant les points de même image soient indépendantes.

Idée de la preuve. Notons S l'ensemble des sommets de Δ_{d+1}, c'est-à-dire

$$S \overset{\text{déf}}{=} \{(0,\dots,0),(1,0,\dots,0),(0,1,0,\dots,0),\dots,(0,\dots,0,1)\} \subset \mathbb{R}^{d+1}.$$

Notons $P \overset{\text{déf}}{=} \operatorname{conv}(S)$, $Q \overset{\text{déf}}{=} P - P = \{\mathbf{x}-\mathbf{y} : \mathbf{x},\mathbf{y} \in P\}$ et ∂X le bord de X. Définissons ensuite

$$h : \left\{ \begin{array}{ccl} Q & \to & P \\ \mathbf{x} & \mapsto & \max_{\prec}\{\mathbf{y} \in P : \exists \mathbf{z} \in P,\ \mathbf{x} = \mathbf{y} - \mathbf{z}\} \end{array} \right.$$

où $\prec$ désigne l'ordre lexicographique sur $\mathbb{R}^{d+1}$ (*cf.* section 5.4.1).

Remarquons que Q est symmétrique par rapport à l'origine et que si $\mathbf{x} = \mathbf{y} - \mathbf{z}$ avec $\mathbf{y} = h(\mathbf{x})$, alors $\mathbf{z} = h(-\mathbf{x})$. Une analyse un peu technique mais élémentaire [2, Preuve du théorème 2′] révèle en outre que h est continue, que $h(\partial Q) = \partial P$ et que si $h(\mathbf{x})$ est un point extrême de P dans une direction $\overrightarrow{u}$, alors $h(-\mathbf{x})$ est un point extrême de P dans la direction $-\overrightarrow{u}$. Par conséquent, h définit une application continue de ∂Q dans ∂P telle que pour tout $\mathbf{x} \in \partial Q$, les faces de P minimales contenant $h(\mathbf{x})$ et $h(-\mathbf{x})$ sont indépendantes.

Puisque Q est un convexe symmétrique, l'application $g : \mathbb{S}^d \to \partial Q$ définie par $g(\overrightarrow{u}) \overset{\text{déf}}{=} \mathbb{R}^+ \overrightarrow{u} \cap \partial Q$ est continue et antipodale. En la composant par h, on obtient une application $\mathbb{S}^d \to \partial P = \|\Delta_{d+1}^{(d)}\|$ telle que pour tout $\mathbf{x} \in \mathbb{S}^d$ les faces de P minimales contenant $g(\mathbf{x})$ et $g(-\mathbf{x})$ sont indépendantes. L'énoncé se déduit donc de la variante (a) du théorème de BORSUK-ULAM appliquée à $f \circ g$. ■

La preuve, dans le cas convexe, du théorème de HELLY à partir du lemme de RADON (*cf.* section 5.2.6) s'adapte au cadre topologique avec quelques notions d'homotopie.

Notons $\mathbb{B}^d$ la boule unité fermée de $\mathbb{R}^d$, c'est à dire de bord $\mathbb{S}^{d-1}$. Un ouvert U de $\mathbb{R}^d$ est *contractile* si toute fonction continue à valeurs dans U est homotope à une application constante ; en particulier, pour toute application continue $f : \mathbb{S}^k \to U$ avec $0 \leqslant k \leqslant d-1$, il existe une application continue $g : \mathbb{B}^{k+1} \to U$ dont la restriction à $\mathbb{S}^k$ égale f. Pour $k = 0$, cela signifie que U est connexe par arc et, plus généralement, on peut envisager la contractilité comme le fait de ne pas avoir de «trous». Une *bonne couverture* est une famille finie d'ouverts telle que l'intersection de toute sous-famille est vide ou contractile. (En particulier, chaque membre de la famille est contractile.) Les familles d'ouverts convexes sont autant d'exemples de bonnes couvertures.

Théorème 5.6.8. *Toute bonne couverture de $\mathbb{R}^d$ d'intersection vide contient une sous-famille d'intersection vide et de taille au plus $d+1$.*

Démonstration. Soit $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$ une bonne couverture de $\mathbb{R}^d$. Notons $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ les sommets du simplexe Δ_{n-1}. Pour $I \subseteq [n]$, notons σ_I la face $\mathrm{conv}\{\mathbf{x}_i : i \in I\}$ de Δ_{n-1}.

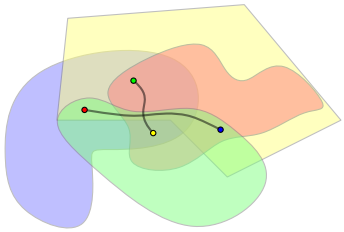
Supposons $n \geqslant d+2$ et que toutes les sous-familles propres de $\mathcal{F}$ sont d'intersection non vide. On construit une application $f : \|\Delta_{n-1}^{(d)}\| \to \mathbb{R}^d$ contrainte par $\mathcal{F}$ au sens suivant :

$$\forall I \subset [n], |I| \leqslant d+1, \quad f(\|\sigma_I\|) \subseteq \cap_{j \notin I} A_j. \tag{5.34}$$

Puisque $n \geqslant d+2$, le complexe simplicial géométrique $\Delta_{n-1}^{(d)}$ contient une face σ de dimension $d+1$. Le lemme de RADON topologique (théorème 5.6.7) assure que σ a deux faces indépendantes, notons les σ_I et σ_J, telles que $f(\sigma_I)$ et $f(\sigma_J)$ se coupent. Fixons $\mathbf{p} \in f(\sigma_I) \cap f(\sigma_J)$. En utilisant la condition (5.34), on obtient

$$\mathbf{p} \in \left(\cap_{j \notin I} A_j\right) \cap \left(\cap_{i \notin J} A_i\right) = \cap_{i \notin I \cap J} A_i$$

et comme σ_I et σ_J sont indépendants, $I \cap J = \emptyset$ et $\mathbf{p} \in \cap\mathcal{F}$. Ainsi, $\cap\mathcal{F} \neq \emptyset$ et le résultat s'en suit.



Il reste à construire l'application f contrainte par $\mathcal{F}$. On construit par récurrence sur $k = 0, 1, \ldots, d$, une application continue $f_k : \|\Delta_{n-1}^{(k)}\| \to \mathbb{R}^d$ telle que

$$\forall I \subset [n], |I| \leqslant k+1, \quad f_k(\|\sigma_I\|) \subseteq \cap_{j \notin I} A_j.$$

Pour l'initialisation, il suffit pour chaque $A_i \in \mathcal{F}$ de choisir un point $\mathbf{p}_i$ dans $\cap_{j \neq i} A_j$, non vide par hypothèse, et de définir $f_0(\mathbf{x}_i) \overset{\text{déf}}{=} \mathbf{p}_i$. Pour l'induction, supposons f_{k-1} construite. Pour tout $\mathbf{x} \in \|\Delta_{n-1}^{(k-1)}\|$ on fixe $f_k(\mathbf{x}) \overset{\text{déf}}{=} f_{k-1}(\mathbf{x})$.

On considère ensuite une face σ_I pour $I \in \binom{[n]}{k+1}$ et on étend f_k sur $\|\sigma_I\|$; ces extensions sont indépendantes les unes des autres. Pour étendre f_k à $\|\sigma_I\|$, notons $\partial\sigma_I$ l'ensemble des faces de σ_I autre que σ_I elle même. Remarquons que $f(\|\partial\sigma_I\|)$ est contenu dans $\cap_{j\notin I} A_j$; cela se vérifie séparément pour chaque face de σ_I. Comme $\|\partial\sigma_I\|$ est homéomorphe à $\mathbb{S}^{k-1}$ et que $\cap_{j\notin I} A_j$ est contractile, on peut étendre $f_k : \|\partial\sigma\| \to \cap_{j\notin I} A_j$ en une application continue $f_k : \|\sigma\| \to \cap_{j\notin I} A_j$. ■

Plongeabilité de complexes simpliciaux

Un complexe simplicial géométrique $\mathcal{K}$ est *plongeable* dans $\mathbb{R}^d$ s'il existe une injection continue $\|\mathcal{K}\| \to \mathbb{R}^d$. Pour les complexes simpliciaux de dimension 1 et $d = 2$, on retrouve la notion classique de *planarité* d'un graphe. Contrairement au cas des graphes, il existe des complexes simpliciaux qui peuvent se plonger continûment mais pas linéairement.

Le cadre intéressant pour étudier la plongeabilité d'un point de vue algorithmique est intermédiaire entre linéaire et topologique. Un complexe simplicial géométrique $\mathcal{K}$ est *plongeable linéairement par morceaux* dans $\mathbb{R}^d$ s'il existe une injection continue et linéaire par morceau $\|\mathcal{K}\| \to \mathbb{R}^d$. Cette notion est distincte des deux précédentes : il existe des complexes simpliciaux géométriques plongeables mais pas plongeables linéairement par morceaux, et d'autres plongeables linéairement par morceaux mais pas linéairement.

Notons $\text{PLONG}_{k\to d}$ le problème algorithmique de décider un complexe simplicial $\mathcal{K}$ de dimension k donné est plongeable dans $\mathbb{R}^d$. Si $\text{PLONG}_{1\to 2}$, le test de planarité de graphe, a une solution en temps linéaire, la question de savoir si $\text{PLONG}_{2\to 3}$ est dans NP est à ce jour ouverte ; on sait seulement que ce problème est décidable et NP-difficile, deux résultats non-triviaux. Certaines variantes de ce problème, par exemple $\text{PLONG}_{4\to 5}$, sont indécidables.

Si l'étude de la plongeabilité est, en général, plus délicate que l'étude de la planarité, voici deux exemples de résultats qui se généralisent :

- Le complexe $\Delta^{(d)}_{2d+2}$ à $2d + 3$ sommets dont tout sous-ensemble de taille $\leqslant d + 1$ forme une face, n'est pas plongeable dans $\mathbb{R}^{2d}$. Cette généralisation de la non-planarité de K_5 a été découverte par Egbert Van KAMPEN et A. FLORES en 1932-1933.
- Une conjecture de Branko GRÜNBAUM datant des années 1970 annonce que si un complexe simplicial géométrique est plongeable

dans $\mathbb{R}^{2d}$ alors son nombre de faces de dimension d est au plus $d+2$ fois son nombre de faces de dimension $d-1$. Cela généralise le fait classique qu'un graphe planaire à n sommets a au plus $3n$ arêtes ; Karim ADIPRASITO en a annoncé une preuve fin 2018.

5.7 Complexes simpliciaux abstraits

La seconde structure à laquelle nous nous intéressons est le complexe simplicial abstrait. Elle est au complexe simplicial géométrique ce que le graphe abstrait – une paire d'ensembles (V, E) où $E \subseteq \binom{V}{2}$ – est au graphe plongé. Nous commençons par clarifier les relations entre complexes simpliciaux géométriques et abstraits. Nous examinons ensuite les f-vecteurs, qui sont des paramètres purement combinatoires des complexes simpliciaux, et établissons le théorème de KRUSKAL-KATONA. Nous étudions enfin une classe de complexes simpliciaux abstraits, dits *friables*, qui apparaissent naturellement en géométrie ; nous prouvons pour cette classe un *théorème de borne supérieure* par des arguments d'inspiration topologique (l'homotopie discrète de WHITEHEAD). Nous concluons en déduisant du théorème de la borne supérieure une version forte du théorème de HELLY fractionnaire.

5.7.1 Définition et terminologie

Un *complexe simplicial abstrait* est une famille finie $\mathcal{C}$ d'ensembles finis qui vérifie :

$$\forall A \in \mathcal{C}, \quad \forall B \subseteq A, \quad B \in \mathcal{C}. \tag{5.35}$$

Autrement dit, $\mathcal{C}$ est stable par l'opérateur «prendre une sous-partie». Nous supposerons sans perte de généralité que $\mathcal{C}$ est une famille de sous-ensembles de $[n]$, pour un entier n.

Précisons quelques termes dont l'origine, géométrique, se précisera en Section 5.7.2. Un élément σ d'un complexe simplicial abstrait $\mathcal{C}$ est appelé une *face* de $\mathcal{C}$. Si $\tau \subseteq \sigma \in \mathcal{C}$, on appelle τ une *face* de σ et σ une *coface* de τ. La *dimension* d'une face σ de $\mathcal{C}$ égale le cardinal de σ moins 1, et est notée $\dim \sigma$. La *dimension* d'un complexe simplicial est la dimension maximale d'une de ses faces. Les faces de dimension 0 et 1 d'un complexe simplicial sont appelées, respectivement, ses *sommets* et ses *arêtes*.

Deux complexes simpliciaux $\mathcal{C}$ et $\mathcal{C}'$ sont *isomorphes* s'il existe une bijection $f : \mathcal{C} \to \mathcal{C}'$ qui commute avec l'inclusion :

$$\forall A, B \in \mathcal{C}, \quad A \subseteq B \Leftrightarrow f(A) \subseteq f(B). \tag{5.36}$$

On peut remarquer qu'un isomorphisme de complexes simpliciaux abstraits est déterminé par sa restriction aux seuls sommets, commute avec les opérateurs d'union et d'intersection, préserve les dimension, les faces, les cofaces, ... Dans ce qui suit, nous n'examinerons que des propriétés de complexes simpliciaux qui sont invariantes par isomorphisme. C'est pour cette raison que nous supposons que $\mathcal{C}$ est un sous-ensemble de $2^{[n]}$ pour un entier n donné.

Soit $G = (V, E)$ un graphe simple, sans boucle et non orienté. L'ensemble

$$\mathcal{C}_G \stackrel{\text{déf}}{=} \{\{v\} : v \in V\} \cup \{\{u, v\} : \{u, v\} \in E\} \cup \{\emptyset\}. \tag{5.37}$$

est un complexe simplicial contenant la même information que $G : V$ et E en sont, respectivement, les ensembles d'éléments de cardinal 1 et 2. De même, l'ensemble des cliques de G est un complexe simplicial (tout comme l'ensemble des parties indépendantes de G).

Exercice 20 Montrer que pour tout complexe simplicial $\mathcal{C}$ de dimension 1, il existe un graphe G tel que $\mathcal{C}_G$, défini en (5.37), est isomorphe à $\mathcal{C}$.

5.7.2 Abstraits VS géométriques

Soit $\mathcal{K}$ un complexe simplicial géométrique de $\mathbb{R}^d$. Pour tout simplexe $\sigma \in \mathcal{K}$, notons $V(\sigma)$ l'ensemble de ses sommets. Notons $V = \cup_{\sigma \in \mathcal{K}} V(\sigma)$ l'ensemble des points de $\mathbb{R}^d$ qui sont sommet d'au moins un simplexe de $\mathcal{K}$. La stabilité de $\mathcal{K}$ par l'opération «prendre une face» assure que

$$\mathcal{C}_{\mathcal{K}} \stackrel{\text{déf}}{=} \{V(\sigma) : \sigma \in \mathcal{K}\} \cup \{\emptyset\} \tag{5.38}$$

est un complexe simplicial *abstrait* ; on appelle cette «famille des ensembles de sommets des simplexes de $\mathcal{K}$» le complexe simplicial abstrait *réalisé par* $\mathcal{K}$. Réciproquement, une *réalisation géométrique* d'un complexe simplicial abstrait $\mathcal{C}$ est tout complexe simplicial géométrique $\mathcal{K}$ tel que $\mathcal{C}_{\mathcal{K}}$ est isomorphe à $\mathcal{C}$.

Proposition 5.7.1. *Tout complexe simplicial abstrait admet une réalisation géométrique.*

Démonstration. Soit $\mathcal{C}$ un complexe simplicial abstrait. Notons δ le cardinal maximal d'un élément de $\mathcal{C}$. Soit $V = \bigcup \mathcal{C}$ et notons $V = \{v_1, v_2, \ldots, v_n\}$. La condition (5.35) assure que tout élément de $\mathcal{C}$ est contenu dans V.

Soit $P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$ un ensemble de points de $\mathbb{R}^{2\delta-1}$ tel que tout sous-ensemble de 2δ points de P soit affinement indépendant. Introduisons

$$f : \left\{ \begin{array}{ccc} 2^V & \to & 2^P \\ A & \mapsto & \{\mathbf{p}_i : v_i \in A\} \end{array} \right. \qquad \text{et} \qquad \mathcal{K} = \{\text{conv}(f(A)) : A \in \mathcal{C}\}.$$

Vérifions que $\mathcal{K}$ est un complexe simplicial géométrique. Puisque $\mathcal{C}$ est stable par l'opérateur «prendre une sous-partie», $\mathcal{K}$ est stable par l'opérateur «prendre une face». Soit $\text{conv}(f(A))$ et $\text{conv}(f(B))$ deux éléments de $\mathcal{K}$. On a

$$|f(A) \cup f(B)| = |f(A \cup B)| = |A \cup B| \leqslant 2\delta,$$

et l'ensemble $f(A) \cup f(B)$ est affinement indépendant. On a donc (comme vu à l'exercice 6) que

$$\text{conv}\,(f(A)) \cap \text{conv}(f(B)) = \text{conv}(f(A) \cap f(B)) = \text{conv}(f(A \cap B)).$$

Comme $A \in \mathcal{C}$, on doit avoir $A \cap B \in \mathcal{C}$ et donc $\text{conv}(f(A \cap B)) \in \mathcal{K}$. L'intersection de deux simplexes de $\mathcal{K}$ est un simplexe de $\mathcal{K}$, et $\mathcal{K}$ est donc un complexe simplicial géométrique. La vérification qu'il réalise $\mathcal{C}$ est laissée en exercice. ■

Pour un exemple explicite d'ensemble P arbitrairement grand en «position générique», voir l'exercice 2.

Certaines propriétés des complexes simpliciaux géométriques ne dépendent que du complexe simplicial abstrait qu'ils réalisent. C'est le cas, par exemple, du problème de plongeabilité (topologique, linéaire par morceaux ou linéaire). Un autre exemple, que nous ne développons pas, est la caractéristique d'EULER-POINCARÉ : la somme alternée des nombres de BETTI d'un complexe simplicial géométrique $\mathcal{K}$ égale la somme alternée de son nombre de faces de dimension i pour $i = 0, 1, \ldots$ qui ne dépend, elle, que du complexe simplicial abstrait réalisé par $\mathcal{K}$.

5.7.3 f-vecteur et théorème de KRUSKAL-KATONA

Soit $\mathcal{C}$ un complexe simplicial abstrait. On note $f_i(\mathcal{C})$ le nombre d'éléments de $\mathcal{C}$ de cardinal $i+1$, c'est-à-dire le nombre de faces de dimension i de $\mathcal{C}$. Le *f-vecteur* d'un complexe simplicial $\mathcal{C}$ de dimension d est le vecteur $(f_0(\mathcal{C}), f_1(\mathcal{C}), \ldots, f_d(\mathcal{C}))$.

Exercice 21 Montrer que pour tout complexe simplicial abstrait $\mathcal{C}$ à n sommets l'on a $(k+1)f_k(\mathcal{C}) \leqslant (n-k)f_{k-1}(\mathcal{C})$. En déduire que la fonction $k \mapsto \frac{1}{\binom{n}{k}} f_{k-1}(\mathcal{C})$ est décroissante (au sens large) pour $0 \leqslant k \leqslant n$. *(Indication : on pourra s'inspirer de la preuve du sampling lemma* (5.5.4)*)*

La propriété d'hérédité 5.35 implique que si $f_k(\mathcal{C})$ est grand, alors $f_{k-1}(\mathcal{C})$ ne peut être trop petit. Ce qui va nous intéresser ici c'est de quantifier cette intuition. Pour tout réel x, définissons $\binom{x}{0} \stackrel{\text{déf}}{=} 1$ et posons, pour tout entier $k \geqslant 1$,

$$\binom{x}{k} \stackrel{\text{déf}}{=} \frac{x(x-1)(x-2)\dots(x-k+1)}{k!}. \tag{5.39}$$

Ainsi, $x \mapsto \binom{x}{k}$ est un polynôme de degré k en x qui interpole les coefficients binomiaux.

Théorème 5.7.2. *Soit $\mathcal{C}$ un complexe simplicial abstrait. Pour tous réel $x \geqslant 0$ et tout entier $k \geqslant 0$, si $f_k(\mathcal{C}) \geqslant \binom{x}{k}$ alors $f_{k-1}(\mathcal{C}) \geqslant \binom{x}{k-1}$.*

Ce résultat a été établi par Joseph KRUSKAL et Gyula KATONA, et annoncé quelques années auparavant par Marcel-Paul SCHÜTZENBERGER. La formulation ci-dessus est due à Lázló LOVÁSZ et la preuve que nous en esquissons est due à Peter FRANKL [4].

Démonstration. Supposons que $V(\mathcal{C}) = \cup\mathcal{C} = [n]$. Notons $\mathcal{F}$ l'ensemble des faces de dimension k de $\mathcal{C}$; soulignons que $|B| = k+1$ pour tout $B \in \mathcal{F}$. Introduisons l'*ombre* $\partial\mathcal{F}$ de $\mathcal{F}$ par

$$\partial\mathcal{F} \stackrel{\text{déf}}{=} \{A : |A| = k \text{ et } \exists B \in \mathcal{F}, A \subseteq B\}.$$

Remarquons que $\partial\mathcal{F}$ est contenu dans $\mathcal{C}$ par hérédité. Nous allons montrer que si $|\mathcal{F}| \geqslant \binom{x}{k}$ alors $|\partial\mathcal{F}| \geqslant \binom{x}{k-1}$.

Nous allons normaliser $\mathcal{F}$ de sorte que si un élément $B \in \mathcal{F}$ ne contient pas 1, alors remplacer n'importe quel sommet de B par 1 produit un autre élément de $\mathcal{F}$. Pour cela, on introduit pour tout $i \in [n]$ et pour tout ensemble $\mathcal{H} \subseteq 2^{[n]}$ l'opérateur

$$T_i^{(\mathcal{H})} : \begin{cases} \mathcal{H} & \to & 2^{[n]} \\ e & \mapsto & \begin{cases} e \setminus \{i\} \cup \{1\} \text{ si } i \in e, 1 \notin e, \text{ et } e \setminus \{i\} \cup \{1\} \notin \mathcal{H} \\ e \text{ sinon} \end{cases} \end{cases}$$

et on note $S_i\mathcal{F} = \{T_i^{(\mathcal{F})}(e) : e \in \mathcal{F}\}$. On considère alors la suite

$$\begin{gathered}\mathcal{F}_1 = \mathcal{F}, \quad \mathcal{F}_2 = S_2\mathcal{F}_1, \quad \mathcal{F}_3 = S_3\mathcal{F}_2, \quad \ldots, \quad \mathcal{F}_n = S_n\mathcal{F}_{n-1}, \\ \mathcal{F}_{n+1} = S_2\mathcal{F}_n, \quad \mathcal{F}_{n+2} = S_3\mathcal{F}_{n+1}, \quad \ldots\end{gathered} \tag{5.40}$$

Remarquons que si $\mathcal{F}_{j+1} \neq \mathcal{F}_j$, alors $\mathcal{F}_{j+1}$ contient strictement plus d'éléments contenant 1 que $\mathcal{F}_j$. Il existe donc un entier s tel que l'on ait $\mathcal{F}_j = \mathcal{F}_s$ pour tout $j \geqslant s$. Notons $\mathcal{F}^* \stackrel{\text{déf}}{=} \mathcal{F}_s$.

La preuve de FRANKL repose sur trois observations. Tout d'abord, les opérateurs $S_\bullet$ ne changent pas le nombre d'éléments, aussi $|\mathcal{F}^*| = |\mathcal{F}|$. Ensuite, l'opérateur $S_\bullet$ préserve les ombres au sens où

$$\partial\left(S_i(\mathcal{F})\right) \subseteq S_i(\partial\mathcal{F}).$$

Cette propriété s'obtient par une analyse de cas élémentaire [4] que l'on omet ici. Enfin, le fait que $\mathcal{F}^*$ soit invariant par tout $S_\bullet$ assure que

$$\forall A \in \partial\mathcal{F}^*, \quad 1 \notin A \Rightarrow A \cup \{1\} \in \mathcal{F}^*. \tag{5.41}$$

ce qui aide à prouver l'inégalité souhaitée. Posons pour cela $\mathcal{F}' \stackrel{\text{déf}}{=} \{e \setminus \{1\} : e \in \mathcal{F}^*, 1 \in e\}$. On a pour tout $A \in \partial\mathcal{F}^*$,

$$1 \notin A \Leftrightarrow A \in \mathcal{F}' \qquad \text{et} \qquad 1 \in A \Leftrightarrow A \setminus \{1\} \in \partial\mathcal{F}'.$$

Ainsi, $|\partial\mathcal{F}^*| = |\mathcal{F}'| + |\partial\mathcal{F}'|$. Par ailleurs, les éléments de $\mathcal{F}^*$ non comptés par $\mathcal{F}'$ sont précisément

$$\mathcal{F}'' \stackrel{\text{déf}}{=} \{e : e \in \mathcal{F}^*, 1 \notin e\},$$

et $|\mathcal{F}^*| = |\mathcal{F}'| + |\mathcal{F}''|$. Enfin, remarquons que $|\mathcal{F}'| \geqslant |\partial\mathcal{F}''|$ puisque

$$A \in \partial\mathcal{F}'' \Rightarrow A \cup \{1\} \in \mathcal{F} \Rightarrow A \in \mathcal{F}'$$

Nous avons ainsi une formulation récursive :

$$|\mathcal{F}^*| = |\mathcal{F}'| + |\mathcal{F}''| \quad \text{et} \quad |\partial\mathcal{F}^*| \geqslant |\mathcal{F}'| + |\partial\mathcal{F}'|. \tag{5.42}$$

On peut conclure par une double récurrence, d'abord sur k puis sur $|\mathcal{F}|$, en remarquant que $\binom{x-1}{k-1} + \binom{x-1}{k-2} = \binom{x}{k-1}$. ■

Application : fonctions de seuils dans les graphes aléatoires

Signalons, sans entrer dans le détail, une application du théorème de KRUSKAL-KATONA en théorie des graphes aléatoires : toute propriété monotone non triviale admet une fonction de seuil dans le modèle de graphes aléatoires d'ERDÖS-RENYI.

Commençons par trois définitions. Notons $G(n,m)$ un graphe aléatoire choisi uniformément parmi les graphes à n sommets et m arêtes. Une *propriété monotone de graphes* est une famille de graphes stable par ajout d'arête : ajouter une arête à un graphe de la famille produit un graphe lui-aussi dans la famille. (Par exemple, l'ensemble des graphes connexes est une propriété monotone.) Une fonction $m^* = m^*(n)$ est un *seuil* pour une propriété monotone de graphe $\mathcal{P}$ si pour toute fonction $m(n)$, on a

$$\begin{aligned} m(n)/m^*(n) \to_{n\to\infty} 0 \quad &\Rightarrow \quad \Pr\left[G(n,m(n)) \in \mathcal{P}\right] \to_{n\to\infty} 0, \\ m(n)/m^*(n) \to_{n\to\infty} \infty \quad &\Rightarrow \quad \Pr\left[G(n,m(n)) \in \mathcal{P}\right] \to_{n\to\infty} 1. \end{aligned}$$

Pour toute propriété monotone de graphes $\mathcal{P}$, la probabilité que $G(n,m)$ soit dans $\mathcal{P}$ augmente avec m. Lorsque cette propriété admet une fonction de seuil, la transition tend pour $n \to \infty$ vers un passage brutal de 0 à 1. Belà BOLLOBÀS et Andrew THOMASON [3] ont découvert que *toute* propriété monotone de graphe admet une fonction seuil.

Restreignons nous, sans perte de généralité, aux graphes de sommets $[n]$. La preuve de BOLLOBÀS et THOMASON observe que pour toute propriété monotone $\mathcal{P}$, les graphes à n sommets qui *ne* sont *pas* dans $\mathcal{P}$ forment un complexe simplicial abstrait $\mathcal{C}$. Il suffit en effet d'identifier chaque graphe au sous-ensemble de $\binom{[n]}{2}$ des paires de sommets qui *ne* forment *pas* une arête. Ainsi, $f_k(\mathcal{C})$ égale le nombre de graphes à sommets dans $[n]$ ayant $m = \binom{n}{2} - k - 1$ arêtes, et le théorème de KRUSKAL-KATONA permet d'analyser comment ce nombre varie quand m croît.

5.7.4 Friabilité et théorème de la borne supérieure

Soit $\mathcal{C}$ un complexe simplicial abstrait. Une face σ de $\mathcal{C}$ est dite *friable* (*collapsable* en anglais) dans $\mathcal{C}$ s'il existe une unique face τ maximale pour l'inclusion dans $\mathcal{C}$ et contenant σ *strictement*. Un *effritement élémentaire* de $\mathcal{C}$ consiste en la suppression de $\mathcal{C}$ de toutes les cofaces d'une face σ friable dans $\mathcal{C}$, y-compris σ. On note cela $\mathcal{C} \searrow \mathcal{C}'$. Cette notion, introduite par Alfred WHITEHEAD, est une analogie discrète de l'homotopie (*cf.* figure 5.10).

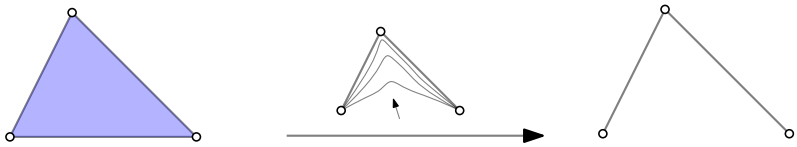


FIGURE 5.10 – *L'effritement supprimant le triangle τ et l'arête horizontale σ est l'analogue discret d'un rétract par déformation de τ sur $\partial\tau \setminus \sigma$.*

Un *d-effritement* de $\mathcal{C}$ est un complexe simplicial abstrait $\mathcal{C}'$ obtenu par une suite d'effritements élémentaires partant de $\mathcal{C}$,

$$\mathcal{C} = \mathcal{C}_0 \searrow \mathcal{C}_1 \searrow \ldots \searrow \mathcal{C}_t = \mathcal{C}',$$

et qui satisfait deux propriétés : (i) lors de chaque effritement élémentaire $\mathcal{C}_i \searrow \mathcal{C}_{i+1}$, la face minimale supprimée est de dimension égale à $d-1$, et (ii) $\mathcal{C}'$ est de dimension au plus $d-1$.

Exercice 22 Montrer que $\mathcal{C} \stackrel{\text{déf}}{=} 2^{[n]}$ admet un d-effritement pour tout $d \leqslant n$.

Le théorème de KRUSKAL-KATONA permet de *minorer* le nombre de k-faces d'un complexe simplicial en fonction de son nombre de k'-faces pour $k' > k$. Pour les complexes admettant un d-effritement, on peut obtenir une *majoration* du même type :

Théorème 5.7.3. *Pour tous entiers n, d et r, pour tout complexe simplicial $\mathcal{C}$ à n sommets admettant un d-effritement, on a*

$$f_{d+r}(\mathcal{C}) = 0 \quad \Rightarrow \quad \forall d \leqslant k \leqslant d+r-1, \quad f_k(\mathcal{C}) \leqslant \sum_{i=0}^{d} \binom{n-r}{i} \binom{r}{k-i+1}$$

Cet énoncé, appelé *théorème de la borne supérieure*, a été conjecturé par Jürgen ECKHOFF pour les nerfs de convexes, objets sur lesquels nous revenons en section 5.7.6. Il a été établi indépendamment par Gil KALAI et Jurgen ECKHOFF en 1984. La preuve qui suit est due à Noga ALON et Gil KALAI.

Cette preuve commence par un comptage purement combinatoire basé sur un argument d'algèbre linéaire (voir [1, 9] pour d'autres applications de l'algèbre linéaire en combinatoire).

Lemme 5.7.4. *Soient $1 \leqslant s \leqslant m \leqslant n$ et t des entiers. Soient $A_1, A_2, \ldots, A_t$ et $B_1, B_2, \ldots, B_t$ des sous-ensembles de $[n]$. Si*

$$\begin{aligned} \forall i \in [t], \quad A_i \subseteq B_i, \quad |A_i| \leqslant s \quad et \quad |B_i| \geqslant m, \\ \forall 1 \leqslant i < j \leqslant t, \qquad\qquad A_i \not\subset B_j, \end{aligned}$$

alors $t \leqslant \binom{n-m+s}{s}$.

Démonstration. Soient $\overrightarrow{v_1}, \overrightarrow{v_2}, \ldots, \overrightarrow{v_n}$ des vecteurs de $V = \mathbb{R}^{n-m+s}$ en position générique, c'est-à-dire tels que toute sous-famille de $n-m+s$ soit linéairement indépendante. Associons à A_i un élément y_i de l'algèbre extérieure ΛV de V munie du produit extérieur usuel $\wedge$:

$$y_i \overset{\text{déf}}{=} \Lambda_{j \in A_j} \overrightarrow{v_j}.$$

On peut supposer que pour tout $i \in [t]$ on a $|A_i| = s$, quitte à compléter A_i par des éléments de $B_i \setminus A_i$. Ainsi, $y_i \in \Lambda^s V$. Montrons que $\{y_i\}_{i\in[t]}$ est libre dans cet espace ; comme il est de dimension $\binom{n-m+s}{s}$, le résultat s'en suit. Pour cela, associons à chaque B_i un élément $z_i \in \Lambda V$ défini par

$$z_i \overset{\text{déf}}{=} \Lambda_{j \in [n]\setminus B_i} \overrightarrow{v_j}.$$

Les vecteurs $\overrightarrow{v_j}$ étant en position générique, un produit extérieur de k d'entre eux, sans répétition, est nul si et seulement si $k > n-m+s$. Les hypothèses du lemme se traduisent donc en

$$\forall i \in [t], \quad y_i \wedge z_i \neq 0 \qquad \text{et} \qquad \forall i < j \in [t], \quad y_i \wedge z_j = 0.$$

Considérons des réels $c_1, c_2, \ldots, c_t$ tels que

$$\sum_{i\in[t]} c_i y_i = 0.$$

Supposons qu'il existe un c_j non nul et considérons $k \overset{\text{déf}}{=} \max\{j : j \in [t] \text{ et } c_j \neq 0\}$. On a alors

$$0 = \left(\sum_{i\in[t]} c_i y_i\right) \wedge z_k = \left(\sum_{i\in[k]} c_i y_i\right) \wedge z_k = c_k \underbrace{y_k \wedge z_k}_{\neq 0}$$

et l'on aboutit à la contradiction que $c_k = 0$. Donc tous les c_j doivent être nuls et la famille $\{y_i\}_{i\in[t]}$ est libre. ■

Preuve du théorème 5.7.3. Fixons n, d et r et considérons un complexe simplicial abstrait $\mathcal{C}$ à n sommets qui admet un d-effritement

$$\mathcal{C} = \mathcal{C}_0 \searrow \mathcal{C}_1 \searrow \ldots \searrow \mathcal{C}_t = \mathcal{C}'$$

et tel que $f_{d+r}(\mathcal{C}) = 0$. Dans chaque effritement élémentaire $\mathcal{C}_i \searrow \mathcal{C}_{i+1}$ la face minimale supprimée a taille d ; notons h_ℓ le nombre de ceux dans

lesquels la face maximale a taille $d+\ell$. Dans un tel effritement élémentaire, le nombre de faces de dimension k qui sont effacées est $\binom{\ell}{k+1-d}$ pour tout $d-1 \leqslant k \leqslant d+\ell-1$. On a donc

$$\forall d \leqslant k \leqslant d+r, \quad f_k(\mathcal{C}) = \sum_{\ell=k+1-d}^{r} h_\ell \binom{\ell}{k+1-d}.$$

Posons $\widetilde{h}_\ell \stackrel{\text{déf}}{=} \sum_{j=\ell}^{r} h_i$ pour $\ell \leqslant r$ et $\widetilde{h}_{r+1} = 0$. On a pour tous $d \leqslant k \leqslant d+r$,

$$f_k(\mathcal{C}) = \sum_{\ell=k+1-d}^{r} \left(\widetilde{h}_\ell - \widetilde{h}_{\ell+1}\right) \binom{\ell}{k+1-d} = \sum_{\ell=k+1-d}^{r} \widetilde{h}_\ell \binom{\ell-1}{k-d}. \quad (5.43)$$

Pour $i < j$, la face maximale effacée dans $\mathcal{C}_j \searrow \mathcal{C}_{j+1}$ ne peut contenir la face minimale effacée dans $\mathcal{C}_i \searrow \mathcal{C}_{i+1}$. Ainsi, le lemme 5.7.4 assure que le nombre d'étapes d'effritement dans lesquelles la face minimale enlevée a $\leqslant s$ sommets et la face maximale enlevée a $\geqslant m$ sommets est au plus $\binom{n-m+s}{s}$. Pour $s = d$ et $m = d+\ell$ on trouve

$$\widetilde{h}_\ell \leqslant \binom{n-\ell}{d}.$$

En injectant cette majoration dans l'identité (5.43) on trouve que pour tous $d \leqslant k \leqslant d+r$,

$$f_k(\mathcal{C}) \leqslant \sum_{\ell=k+1-d}^{r} \binom{n-\ell}{d} \binom{\ell-1}{k-d} = \sum_{i=0}^{d} \binom{n-r}{i} \binom{r}{k+1-i},$$

La dernière réécriture est laissée en exercice (*Indication : utiliser l'identité de Chu-Vandermonde*.). ■

5.7.5 Nerfs de convexes

Pour analyser les motifs d'intersection dans une famille $\mathcal{F}$ d'ensembles, il est pratique d'introduire le complexe simplicial

$$\mathcal{N}(\mathcal{F}) \stackrel{\text{déf}}{=} \{\mathcal{G} \subseteq \mathcal{F} \colon \cap \mathcal{G} \neq \emptyset\} \cup \{\emptyset\}. \quad (5.44)$$

appelé *nerf* de $\mathcal{F}$. De manière équivalente, si $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$, on peut définir $\mathcal{N}(\mathcal{F})$ comme $\{I \subseteq [n] \colon \cap_{i\in I} A_i \neq \emptyset\} \cup \{\emptyset\}$; ces deux définitions produisent des complexes simpliciaux abstraits isomorphes.

Exercice 23 Prouver que pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^2$ il existe une famille finie $\mathcal{F}'$ de polygones convexes telle que $\mathcal{N}(\mathcal{F}) = \mathcal{N}(\mathcal{F}')$. Généraliser cette propriété en dimension d.

Exercice 24 Construire une famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^2$ telle qu'il n'existe aucune famille finie $\mathcal{F}'$ de disques telle que $\mathcal{N}(\mathcal{F}) = \mathcal{N}(\mathcal{F}')$.

Dans cette section, nous prouvons le résultat suivant :

Théorème 5.7.5. *Pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$, $\mathcal{N}(\mathcal{F})$ admet un d-effritement.*

Ainsi, le nerf de toute famille de convexes doit satisfaire le théorème de la borne supérieure (5.7.3). Le théorème 5.7.5 a été établi par Gerd WEGNER en 1975. Sa preuve utilise une idée que l'on a déjà vue au lemme 5.4.2 : le point maximal pour l'ordre lexicographique dans l'intersection de $d+1$ convexes de $\mathbb{R}^d$ est déjà maximal dans l'intersection de d d'entre eux.

Il est utile de commencer par formuler une notion voisine de l'effritement. Un *d-effritement élémentaire* de $\mathcal{C}$ est un complexe simplicial abstrait $\mathcal{C}'$ obtenu en supprimant de $\mathcal{C}$ un simplexe σ de dimension au plus $d-1$ ainsi que toutes ses cofaces ; on le note $\mathcal{C} \searrow_d \mathcal{C}'$. Un complexe simplicial abstrait $\mathcal{C}$ est *d-friable* s'il existe une suite de d-effritements élémentaires : $\mathcal{C} = \mathcal{C}_0 \searrow_d \mathcal{C}_1 \searrow_d \ldots \searrow_d \mathcal{C}_k = \emptyset$.

Lemme 5.7.6. *Un complexe simplicial abstrait est d-friable si et seulement s'il admet un d-effritement.*

Idée de preuve. La preuve se fait en deux étapes. La première étape transforme la séquence $\mathcal{C} = \mathcal{C}_0 \searrow_d \mathcal{C}_1 \searrow_d \ldots \searrow_d \mathcal{C}_k = \emptyset$ en une succession de deux types d'opérations : des effritements élémentaires (au sens de WHITEHEAD) et des suppressions de faces maximales de dimension au plus d. La seconde étape réordonne ces opérations de manière à faire tous les effritements élémentaires avant la première suppression. Pour plus de détails, voir [11, lemme 1]. ■

Nous pouvons maintenant prouver le théorème de WEGNER en établissant que le nerf de toute famille de convexes est d-friable.

Preuve du théorème 5.7.5. Soit $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$ une famille de convexes de $\mathbb{R}^d$. On peut supposer, sans perte de généralité, que chaque A_i est l'enveloppe convexe d'un ensemble fini de points et que les A_i sont en position générique au sens où l'intersection des bords de $d+1$ d'entre eux est vide (*cf.* exercice 23).

Notons $\prec$ l'ordre lexicographique sur $\mathbb{R}^d$ (*cf.* section 5.4.1). Pour tout point $\mathbf{p} \in \mathbb{R}^d$ posons $h_{\mathbf{p}} \stackrel{\text{déf}}{=} \{\mathbf{q} \in \mathbb{R}^d : \mathbf{p} \preceq \mathbf{q}\}$. Chaque ensemble $h_{\mathbf{p}}$ est

convexe. On définit

$$\mathcal{N}_{\mathbf{p}}(\mathcal{F}) \stackrel{\text{déf}}{=} \{\mathcal{G} \subseteq \mathcal{F} \colon\ (\cap\mathcal{G}) \cap h_{\mathbf{p}} \neq \emptyset\}$$

et on note $\mathcal{N}(\mathcal{F}) = \mathcal{N}_0 \supsetneq \mathcal{N}_1 \supsetneq \ldots \supsetneq \mathcal{N}_t = \emptyset$ la suite des valeurs prises par $\mathcal{N}_{\mathbf{p}}(\mathcal{F})$ quand $\mathbf{p}$ parcourt $\mathbb{R}^d$ dans l'ordre lexicographique.

Pour tout point $\mathbf{x} \in \mathbb{R}^d$, considérons l'ensemble $M_{\mathbf{x}} = \{\sigma : \sigma \in \mathcal{N}_{\mathbf{x}}(\mathcal{F}) \text{ et } \max_{\prec} \cap\sigma = \mathbf{x}\}$. Notre hypothèse de position générique assure que quand $M_{\mathbf{x}} \neq \emptyset$, exactement un de ses élément est minimal pour l'inclusion ; notons le $\sigma_{\mathbf{x}}$. Le lemme 5.4.2 assure que $\dim \sigma_{\mathbf{x}} \leqslant d-1$. Remarquons de plus que $M_{\mathbf{x}}$ contient tout $\sigma \in \mathcal{N}(\mathcal{F})$ qui contient lui-même $\sigma_{\mathbf{x}}$. Autrement dit, $M_{\mathbf{x}}$ est l'ensemble des cofaces de $\sigma_{\mathbf{x}}$ dans $\mathcal{N}_{\mathbf{x}}(\mathcal{F})$.

Les éléments de $\mathcal{F}$ étant compacts, pour $0 \leqslant i \leqslant t-1$ il existe un point $\mathbf{x}_i \in \mathbb{R}^d$ tel que $\mathcal{N}_{i+1} \setminus \mathcal{N}_i = M_{\mathbf{x}_i}$. On a donc bien $\mathcal{N}_i \searrow_d \mathcal{N}_{i+1}$ et $\mathcal{N}(\mathcal{F})$ est d-friable. ■

Un exemple atteignant la borne du théorème 5.7.3

Fixons des paramètres $n > r > d$ et considérons une famille $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$ où $\{A_1, A_2, \ldots, A_{n-r}\}$ est un ensemble d'hyperplans de $\mathbb{R}^d$ en position générale et où $A_i = \mathbb{R}^d$ pour $i > n-r$. Remarquons que $f_{d+r+1}(\mathcal{N}(\mathcal{F})) = 0$ puisque toute sous-famille de $\mathcal{F}$ de taille $d+r+1$ doit contenir $d+1$ hyperplans. Par ailleurs, on a pour tous k entre d et r,

$$f_k(\mathcal{N}(\mathcal{F})) = \sum_{i=0}^{d} \binom{n-r}{i}\binom{r}{k+1-i}$$

en sommant les nombres de sous-ensembles formés de i hyperplans et $k+1-i$ copies de $\mathbb{R}^d$. Le théorème 5.7.5 assure que $\mathcal{N}(\mathcal{F})$ admet un d-effritement, aussi cet exemple montre que la borne du théorème 5.7.3 peut être atteinte et, qui plus est, par un nerf d'une famille de convexes de $\mathbb{R}^d$. Nous nous servirons de cela dans la preuve du théorème 5.7.7.

5.7.6 Théorème de HELLY fractionnaire (fort)

Nous pouvons finalement combiner les théorèmes 5.7.3 et 5.7.5 pour obtenir une version forte du théorème de HELLY fractionnaire.

Théorème 5.7.7. *Pour tout $\alpha > 0$ et $d \geqslant 1$, pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$, si une fraction α des sous-ensembles de $\mathcal{F}$ de taille $d+1$ sont d'intersection non vide, alors une fraction $1-(1-\alpha)^{\frac{1}{d+1}}$ de $\mathcal{F}$ est d'intersection non vide.*

Démonstration. Notons $\mathcal{R}_{d,n}$ l'ensemble des complexes simpliciaux à sommets dans $[n]$ réalisables comme nerf d'une famille de n convexes de $\mathbb{R}^d$. Pour $\rho \in [0,1]$ et $n > k > d$ on définit

$$\alpha(\rho,d,k,n) \overset{\text{déf}}{=} \frac{1}{\binom{n}{k}} \max_{\substack{\mathcal{C}\in\mathcal{R}_{d,n}\\ f_{\lfloor \rho n\rfloor}(\mathcal{C})=0}} f_{k-1}(\mathcal{C}),$$

de sorte que pour prouver l'énoncé il suffit de montrer que

$$\forall n > k > d \quad \text{et} \quad \rho \in [0,1], \qquad \alpha(\rho,d,k,n) \geqslant 1-(1-\rho)^{d+1}.$$

Notons $\mathcal{E}_{d,n}$ l'ensemble des complexes simpliciaux à sommets dans $[n]$ qui admettent un d-effritement. On a

$$\max_{\substack{\mathcal{C}\in\mathcal{R}_{d,n}\\ f_{\lfloor \rho n\rfloor}(\mathcal{C})=0}} f_{k-1}(\mathcal{C}) \leqslant \max_{\substack{\mathcal{C}\in\mathcal{E}_{d,n}\\ f_{\lfloor \rho n\rfloor}(\mathcal{C})=0}} f_{k-1}(\mathcal{C}) \leqslant \sum_{i=0}^{d} \binom{n+d-\lfloor \rho n\rfloor}{i}\binom{\lfloor \rho n\rfloor - d}{k-i}.$$

En effet, la première inégalité vient de l'inclusion $\mathcal{R}_{d,n} \subseteq \mathcal{E}_{d,n}$ donnée par le théorème de WEGNER (5.7.5) et la seconde du théorème 5.7.3. La fin de la section 5.7.5 donne un exemple de $\mathcal{C} \in \mathcal{R}_{n,d}$ qui atteint cette borne (prendre $r = \lfloor \rho n\rfloor - d$), aussi les deux inégalités sont des égalités. On a donc

$$\alpha(\rho,d,k,n) = \frac{1}{\binom{n}{k}} \sum_{i=0}^{d} \binom{n+d-\lfloor \rho n\rfloor}{i}\binom{\lfloor \rho n\rfloor - d}{k-i}.$$

On souhaite un majorant de $\alpha(\rho,d,k,n)$ pour ρ et d fixés et uniforme en k et n. Considérons une famille $\mathcal{F}$ de n convexes de $\mathbb{R}^d$ et notons $\mathcal{F}'$ la famille de $2n$ convexes de $\mathbb{R}^d$ obtenue en prenant deux copies de chaque élément de $\mathcal{F}$. Une comparaison de $\mathcal{N}(\mathcal{F})$ et de $\mathcal{N}(\mathcal{F}')$ révèle [6, §4] que

$$\frac{1}{\binom{2n}{k}} f_{k-1}\left(\mathcal{N}(\mathcal{F}')\right) \geqslant \frac{1}{\binom{n}{k}} f_{k-1}\left(\mathcal{N}(\mathcal{F})\right),$$

d'où $\alpha(\rho,d,k,n) \leqslant \alpha(\rho,d,k,2n)$. Par conséquent,

$$\alpha(\rho,d,k) \overset{\text{déf}}{=} \sup_{n>k} \alpha(\rho,d,k,n) = \lim_{n\to\infty} \alpha(\rho,d,k,n) = \sum_{i=0}^{d} \binom{k}{i} \rho^{k-i}(1-\rho)^i.$$

Comme vu à l'exercice 21, pour tout complexe simplicial abstrait $\mathcal{C}$ la fonction $k \mapsto \frac{1}{\binom{n}{k}} f_{k-1}(\mathcal{C})$ est décroissante (au sens large) pour $0 \leqslant k \leqslant n$.

La fonction $k \mapsto \alpha(\rho, d, k, n)$ est donc elle aussi décroissante au sens large (pour ρ, d, n fixés). Ainsi, $\sup_{k>d} \alpha(\rho, d, k) = \alpha(\rho, d, d+1)$ et

$$\sup_{n>k>d} \alpha(\rho, d, k, n) = \sum_{i=0}^{d} \binom{d+1}{i} \rho^{d+1-i}(1-\rho)^i = 1 - (1-\rho)^{d+1},$$

ce qui conclut la preuve. ■

5.8 Hypergraphes

La dernière structure à laquelle nous nous intéressons est l'hypergraphe. Un hypergraphe de sommets V est tout simplement une sous-famille de 2^V, c'est à dire un ensemble de sous-ensembles de V. Ainsi, tout complexe simplicial abstrait est un hypergraphe mais l'inverse est faux, puisqu'un hypergraphe ne satisfait pas nécessairement la condition (5.35). Cette section conclut ce chapître en esquissant des liens entre la convexité combinatoire et deux thèmes classiques en théorie des hypergraphes : la *dimension de* VAPNIK-CHERVONENKIS et l'*exclusion de motifs*.

5.8.1 Dimension de VAPNIK-CHERVONENKIS

Soit V un ensemble fini et $H \subseteq 2^V$ un hypergraphe de sommets V. La *trace* $H_{|U}$ d'un hypergraphe H sur un sous-ensemble $U \subseteq V$ est l'hypergraphe de sommets U défini par

$$H_{|U} = \{e \cap U : e \in H\}. \tag{5.45}$$

La *dimension VC* d'un hypergraphe H est la taille du plus grand sous-ensemble $U \subseteq V$ tel que $H_{|U} = 2^U$. Cette dimension a été proposée par Vladimir VAPNIK et Alexey CHERVONENKIS notamment en raison de la propriété suivante.

Lemme 5.8.1. *Si H est un hypergraphe de sommets $[n]$ et dimension VC d, alors*

$$|H| \leqslant \sum_{i=0}^{d} \binom{n}{i} = O(n^d).$$

Ce lemme a été découvert indépendamment par VAPNIK et CHERVONENKIS, par Norbert SAUER et par Saharon SHELAH au début des années 1970.

Une première preuve procède par récurrence en posant $H' \stackrel{\text{déf}}{=} H_{|[n-1]}$ et $H'' \stackrel{\text{déf}}{=} \{e \in H' : e \in H \text{ et } e \cup \{n\} \in H\}$, puis en remarquant que

$|H| = |H'| + |H''|$, que H' est de dimension VC au plus d et que H'' est de dimension VC au plus $d-1$. Une autre preuve procède par «compression», de manière analogue à la preuve du théorème de KRUSKAL-KATONA présentée en section 5.7.3. Une troisième preuve, encore, utilise un argument d'algèbre linéaire du type de celui utilisé dans la preuve du lemme 5.7.4.

Dimension VC et lemme de RADON

Considérons une famille finie $\mathcal{F}$ de sous-ensembles de $\mathbb{R}^d$. La *dimension VC de* $\mathcal{F}$ est le maximum, pour un sous-ensemble fini P, de la dimension VC de l'hypergraphe

$$\mathcal{H}_{\mathcal{F}}(P) \stackrel{\text{déf}}{=} \{P \cap A \colon A \in \mathcal{F}\}.$$

(De manière équivalente, c'est la dimension VC de l'hypergraphe infini $\mathcal{H}_{\mathcal{F}}(\mathbb{R}^d)$, c'est à dire de $\mathcal{F}$ vu comme ensemble d'arêtes sur l'ensemble infini de sommets $\mathbb{R}^d$.) Voici une conséquence du lemme de RADON.

Proposition 5.8.2. *La dimension VC de toute famille de demi-espaces de* $\mathbb{R}^d$ *est au plus* $d+1$.

Démonstration. Supposons qu'une famille $\mathcal{F} = \{A_1, A_2, \ldots, A_n\}$ de demi-espaces de $\mathbb{R}^d$ ait dimension VC $\delta \geqslant d+2$. Il existe donc un ensemble S de δ points de $\mathbb{R}^d$ telle que pour tout $T \subseteq S$, il existe $i(T) \in [n]$ tel que $T = S \cap A_{i(T)}$. Le lemme de RADON assure que S contient deux ensembles disjoints T_1, T_2 tels que $\mathrm{conv}(T_1)$ et $\mathrm{conv}(T_2)$ se coupent, disons en un point $\mathbf{p}$. D'une part, le demi-espace $A_{i(T_1)}$ contient T_1 et donc $\mathbf{p}$. D'autre part, le complémentaire de $A_{i(T_1)}$ contient T_2 et donc $\mathbf{p}$, contradiction. ■

Exercice 25 Étant donné un polynôme Q en d variables, notons $\{Q \geqslant 0\}$ le sous-ensemble de $\mathbb{R}^d$ des points en lesquels Q prends une valeur positive. Fixons un entier k et considérons une famille finie $\mathcal{P}$ de polynôme d-variés à coefficient réels et de degré total au plus k. Utiliser le lemme de RADON pour borner la dimension VC de la famille $\mathcal{F} \stackrel{\text{déf}}{=} \{\{Q \geqslant 0\} \colon Q \in \mathcal{P}\}$. *Indication : on pourra utiliser l'application de Veronese*

$$(x_1, x_2, \ldots, x_d) \mapsto (x_1, x_2, \ldots, x_d, x_1^2, x_1x_2, \ldots, x_d^k). \tag{5.46}$$

Dimension VC duale et théorème de HELLY fractionnaire

Considérons une famille finie $\mathcal{F}$ de sous-ensembles de $\mathbb{R}^d$. Le *diagramme de Venn* $V(\mathcal{F})$ de $\mathcal{F}$ est

$$V(\mathcal{F}) \stackrel{\text{déf}}{=} \{\mathcal{G} \subseteq \mathcal{F} \colon (\cap\mathcal{G}) \setminus (\cup(\mathcal{F} \setminus \mathcal{G})) \neq \emptyset\}.$$

C'est un hypergraphe de sommets $\mathcal{F}$, dual de $\mathcal{H}_{\mathcal{F}}(\mathbb{R}^2)$ au sens usuel (qui revient à transposer la matrice d'incidence entre sommets et arêtes). La *dimension VC duale* de $\mathcal{F}$ est la dimension VC de $V(\mathcal{F})$. Autrement dit, c'est le cardinal de la plus grande sous-famille $\mathcal{G} \subseteq \mathcal{F}$ telle que $V(\mathcal{G}) = 2^{\mathcal{G}}$. Le lemme de SAUER-SHELAH assure que pour toute famille $\mathcal{F}$ de taille n et dimension VC duale δ, on a

$$|V(\mathcal{F})| \leqslant \sum_{i=0}^{\delta} \binom{n}{i} = O(n^{\delta}). \tag{5.47}$$

Voici une variante du théorème de Helly fractionnaire (5.4.1 ou 5.7.7) :

Théorème 5.8.3. *Pour tout $\alpha > 0$, $d \geqslant 1$ et $k \geqslant 2$ il existe $\beta > 0$ tel que pour toute famille finie $\mathcal{F}$ de sous-ensembles de $\mathbb{R}^d$ de dimension VC duale $k-1$, si une fraction α des sous-ensembles de $\mathcal{F}$ de taille k sont d'intersection non vide, alors une fraction β de $\mathcal{F}$ est d'intersection non vide.*

Ce résultat a été découvert par Jiří MATOUŠEK en 2004. Sa preuve est un simple double-comptage.

Preuve du théorème 5.8.3. Fixons α, d et k. Notons $g_d(m) \stackrel{\text{déf}}{=} \sum_{i=0}^{k-1} \binom{m}{i}$ et fixons $m > k$ tel que $g_d(m) < \frac{\alpha}{4}\binom{m}{k}$. Fixons $\beta = \frac{1}{2m}$ et supposons qu'il existe une famille $\mathcal{F}$ de taille $n \geqslant m/\beta$ et de dimension VC duale $k-1$ telle que toute sous-famille de $\mathcal{F}$ de taille au moins βn est d'intersection vide. [9] Notons $\mathcal{F} \stackrel{\text{déf}}{=} \{A_1, A_2, \ldots, A_n\}$. Définissons maintenant

$$\mathcal{E} \stackrel{\text{déf}}{=} \left\{ (I, J) \in \binom{[n]}{k} \times \binom{[n]}{m} : I \subseteq J \right\}.$$

Définissons une paire $(I, J) \in \mathcal{E}$ comme *bonne* s'il existe un point $\mathbf{p} \in \mathbb{R}^d$ qui appartient à tous les A_i pour $i \in I$ et à aucun A_j pour $j \in J \setminus I$.

Choisissons I uniformément au hasard dans $\binom{[n]}{k}$. Par hypothèse, l'ensemble $R = \cap_{i \in I} A_i$ est non vide avec probabilité au moins α ; supposons qu'il l'est et fixons un point arbitraire $\mathbf{x} \in R$. Choisissons maintenant les $m-k$ éléments de J uniformément dans $[n] \setminus I$. Puisque $\mathcal{F}$ contredit l'énoncé, $\mathbf{x}$ est contenu dans moins de βn éléments de $\mathcal{F}$ et la probabilité qu'aucun des A_j pour $j \in J \setminus I$ ne contienne $\mathbf{x}$ est au moins

$$\frac{\binom{\lceil (1-\beta)n \rceil}{m-k}}{\binom{n-k}{m-k}} \geqslant \prod_{i=0}^{m-k-1} \frac{(1-\beta)n - i}{n-i} \geqslant \left(\frac{(1-\beta)n - m}{n-m} \right)^m \geqslant \frac{1}{4}.$$

9. La minoration de n se fait sans perte de généralité : si les seules familles pour lesquelles $\beta = \frac{1}{2m}$ ne convient pas sont de taille au plus N, alors $\beta \stackrel{\text{déf}}{=} 1/N$ convient.

Ainsi, $p \geqslant \alpha/4$.

Maintenant, pour $I \subseteq [n]$ notons $\mathcal{F}_I \stackrel{\text{déf}}{=} \{A_i : i \in I\}$. Si on fixe $J \in \binom{[n]}{m}$, le nombre de $I \in \binom{J}{k}$ tels que $(I, J) \in \mathcal{E}$ est au plus $|V(\mathcal{F}_J)|$. Comme $\mathcal{F}_J$ est de dimension VC au plus $k-1$, on a $|V(\mathcal{F}_J)| \leqslant g_d(m)$. En choisissant J uniformément dans $\binom{[n]}{m}$, puis I uniformément dans $\binom{J}{k}$, on trouve

$$p\binom{m}{k} = \Pr\left[(I, J) \text{ bonne}\right]\binom{m}{k} \leqslant \mathbb{E}\left[|V(\mathcal{F}_J)|\right] \leqslant g_d(m)$$

Comme $p \geqslant \alpha/4$, on a $\frac{\alpha}{4}\binom{m}{k} \leqslant g_d(m)$ ce qui contredit le choix de m. ■

5.8.2 Cliques et motifs exclus dans les hypergraphes uniformes

Un hypergraphe est *k-uniforme* si toutes ses arêtes ont cardinal k. Étant donnée une famille $\mathcal{F}$ de convexes de $\mathbb{R}^d$, définissons

$$\mathcal{E}_k(\mathcal{F}) \stackrel{\text{déf}}{=} \left\{\mathcal{G} \in \binom{\mathcal{F}}{k} : \cap\,\mathcal{G} \neq \emptyset\right\}.$$

Ainsi, $\mathcal{E}_k(\mathcal{F})$ est un hypergraphe k-uniforme de sommets $\mathcal{F}$, qui représente les k-uplets de $\mathcal{F}$ d'intersection non vide. Nous allons maintenant voir comment certains résultats de convexité combinatoire dans $\mathbb{R}^d$ peuvent se reformuler en termes des hypergraphes $\mathcal{E}_{d+1}(\mathcal{F})$, de manière analogue à la section 5.6.2.

Une *clique* dans un hypergraphe k-uniforme $\mathcal{H}$ de sommets V est un sous-ensemble $U \subseteq V$ tel que $\binom{U}{k} \subseteq \mathcal{H}$. Revoici le théorème de HELLY fractionnaire faible :

Théorème 5.8.4 (5.4.1 reformulé)**.** *Pour tout $\alpha > 0$ et $d \geqslant 1$ il existe $\beta > 0$ tel que pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$, si $|\mathcal{E}_{d+1}(\mathcal{F})| \geqslant \alpha\binom{|\mathcal{F}|}{d+1}$ alors $\mathcal{E}_{d+1}(\mathcal{F})$ admet une clique de taille au moins $\beta|\mathcal{F}|$.*

Dans un hypergraphe k-uniforme $\mathcal{H}$ de sommets V, un *m-uplet complet d'arêtes manquantes* est un m-uplet $(\tau_1, \tau_2, \ldots, \tau_m)$ d'éléments de $\binom{V}{k} \setminus \mathcal{H}$ tels que : (i) les τ_i sont deux à deux disjoints, (ii) tout ensemble $\{t_1, t_2, \ldots, t_m\}$ obtenu en prenant un sommet t_i dans chaque non-arête τ_i est une clique dans $\mathcal{H}$. Revoici le théorème de HELLY coloré :

Théorème 5.8.5 (5.3.4 reformulé)**.** *Pour toute famille finie $\mathcal{F}$ de convexes de $\mathbb{R}^d$, $\mathcal{E}_{d+1}(\mathcal{F})$ ne contient pas de $(d+1)$-uplet complet d'arête manquante.*

Le résultat suivant énonce que dans ces reformulations, le théorème de HELLY coloré «implique» le théorème de HELLY fractionnaire.

Théorème 5.8.6. *Pour tout $m \geqslant k \geqslant 2$ et $\alpha \in (0,1)$, il existe $\beta > 0$ tel que tout hypergraphe k-uniforme à n sommet ayant au moins $\alpha\binom{n}{m}$ cliques de taille m et aucun m-uplet complet d'arêtes manquantes contient une clique de taille βn.*

Remarquer que le théorème 5.8.6 s'applique à tout hypergraphe k-uniforme, pas seulement ceux représentant les intersections de convexes. Ce théorème a été découvert par Andreas HOLMSEN en 2019, et nous renvoyons à son manuscrit pour la preuve [5].

Bibliographie

[1] L. BABAI et P. FRANKL : *Linear Algebra Methods in Combinatorics : With Applications to Geometry and Computer Science*. Department of Computer Science, univ. of Chicag, 1992.

[2] E. G. BAJMÓCZY et I. BÁRÁNY : On a common generalization of Borsuk's and Radon's theorem. *Acta Math. Acad. Sci. Hungar.*, 34(3-4):347–350, 1979.

[3] B. BOLLOBÁS et A. G. THOMASON : Threshold functions. *Combinatorica*, 7(1):35–38, 1987.

[4] P. FRANKL : A new short proof for the kruskal-katona theorem. *Discrete Mathematics*, 48(2-3):327–329, 1984.

[5] A. F. HOLMSEN : Large cliques in hypergraphs with forbidden substructures, 2019. ArXiv :1903.00245.

[6] G. KALAI : Intersection patterns of convex sets. *Israel Journal of Mathematics*, 48(2-3):161–174, juin 1984.

[7] J. MATOUŠEK : *Lectures on Discrete Geometry*, vol. 212 de *Grad. Texts in Math.* Springer-Verlag, New York, 2002.

[8] J. MATOUŠEK : *Using the Borsuk-Ulam theorem : lectures on topological methods in combinatorics and geometry*. Universitext. Springer-Verlag, Berlin, 2003. Written in cooperation with A. Björner and G. M. Ziegler.

[9] J. MATOUŠEK : *Thirty-three miniatures : Mathematical and Algorithmic applications of Linear Algebra*, vol. 53 de *Student Mathematical Library*. Amer. Math. Soc., 2010.

[10] J. MATOUSEK et B. GÄRTNER : *Understanding and using linear programming*. Springer Science & Business Media, 2007.

[11] G. WEGNER : d-collapsing and nerves of families of convex sets. *Archiv der Mathematik*, 26(1):317–321, 1975.

Table des matières