Plans
et
développements
pour l'Agrégation de mathématiques

Option D informatique

Gaëtan Douéneau-Tabot
gaetan.doueneau@ens-paris-saclay.fr
École Normale Supérieure Paris-Saclay

Juillet 2018

INTRODUCTION

Préparer les oraux de l'agrégation, c'est sans doute très difficile. Certes, des sources foisonnent de recueils de développements (plus ou moins prémâchés), mais il reste à les goûter, les comprendre, les apprendre, les apprécier, les sentir, les faire tenir en 15 minutes… C'est un travail tellement important que les 15min d'un développement bien préparé deviennent *incompréhensibles* par un élève de licence pris au hasard ! Une solution possible pour se motiver à travailler, c'est d'écrire soi-même un recueil en imaginant son public. C'est ce que j'ai essayé de faire ici entre avril et juillet 2018, hélas il est *truffé de typos*[1].

Préparer des plans et des développements. Pour être bien prêt le jour de l'oral, on aura soin d'écrire tous ses développements en étant sûr que le moindre détail est compris. Enfin, il sera de bon goût de les essayer au moins une fois au tableau. Réussir son développement est *essentiel*, puisque c'est le moment de l'oral qui ressemble le plus à un cours devant une classe[2].

Dans ce recueil, chaque développement est précédé d'un *Avis* et encadré de ses *Prérequis* et *Postrequis*[3]. Les *Prérequis* contiennent les indispensables à savoir (démontrer) pour aborder ce développement. J'ai essayé de recenser dans les *Postrequis* des thèmes qui pourraient tomber en questions, ou au moins un peu de matière utile pour y répondre. Ce dernier élément est en général absent des recueils de développements, mais néanmoins essentiel pour avoir du recul face au jury ! En ce qui concerne la structure des développements, certains ne peuvent pas être traités en détail en 15 minutes. Dans ce cas, la section *Développement* contient les informations importantes exposées sur le tableau, mais certains résultats auxiliaires ou admis sont indiqués au fur et à mesure dans les notes en bas de page. Contrairement aux *Prérequis*, il s'agira là de concepts que l'on ne peut pas vraiment extraire hors du développement.

Mais ce n'est pas tout ! Il est bon de prévoir des "plans de plans"[4] et de les apprendre. Il s'agit surtout d'avoir une idée des résultats essentiels, de respecter un ordre de démonstration cohérent, et de savoir comment on justifie les développements présentés. Enfin, le rapport du jury est une mine d'informations sur ce qu'il faut mettre (ou ne pas mettre) dans chacune des leçons.

L'option info. D'aucuns diront que l'option informatique n'est pas plus difficile que les autres. Je pense qu'il n'en est rien. D'une part, les leçons de maths réservées aux informaticiens forment un ensemble dense dans le programme, ce qui fait qu'il ne peuvent pas vraiment faire

1. N'hésitez pas à me les signaler par mail, vous recevrez une surprise en échange.
2. L'agreg est un concours de recrutement d'enseignants, et pas un classement annuel des (jeunes) mathématiciens.
3. On pensera (ou pas) aux triplets de Hoare.
4. Aussi appelés "trames de plans" ou "métaplans" par les adeptes d'un contre-contre-mouvement.

l'impasse sur certaines parties de celui-ci (sauf peut-être la théorie des représentations, l'analyse complexe et les équations diophantiennes). D'autre part, le programme spécifique de l'informatique est *totalement disjoint* du programme de maths, alors que les autres options éclairent un domaine mathématique précis (probabilités, numérique, algèbre). A cet égard, on finira par voir l'agrégation option informatique comme une *double agrégation*, délicate à aborder pour qui n'a fait que peu de maths *ou* peu d'informatique auparavant.

Attendu que l'informatique est plus récente, il semblerait en outre que les concepts soient moins standards. Ainsi, il n'existe pas vraiment de corpus de développements pertinents en info, et ils nécessitent un travail accru car les questions peuvent pas être différentes selon la manière dont on présente la chose. Dans la même veine, le jury serait plus subjectif en info, apprécierait peu les recasages indus, et ne donnerait en général pas d'exercices[1].

Face à cette option assez bancale, une solution de bon goût serait la création d'une agrégation d'informatique, mais c'est un autre débat. En attendant, il faut travailler un peu plus.

Remerciements. Toute ma gratitude va d'abord vers les préparateurs de l'ENS qui nous ont tant appris, tant motivés et tant conseillés tout au long de cette année 2017-2018. J'ai toutes les peines du monde à concevoir l'étendue de mon ignorance avant le début de cette préparation. Je remercie également tous mes camarades de cette année, et tout particulièrement Émilie & Aliaume, avec qui j'ai partagé les pleurs et les joies de l'option info.

Peroraison. Du reste, bon courage ! Et si vous êtes perdus, n'oubliez pas qu'il y a toujours des gens *moins meilleurs* que vous, par exemple le "vous" d'hier (qui ignorait encore tout ce que vous savez aujourd'hui). Mais ça ne vous interdit pas de continuer à travailler.

> Prier ! le moment est mauvais.
> Assurez d'abord le succès ;
> Vous rendrez grâce au ciel après.
>
> *Benvenuto Cellini*, Hector Berlioz, Léon de Wailly, Auguste Barbier

1. Le lecteur aura remarqué que ce paragraphe est au conditionnel. Il ne s'agit là que d'une impression pour laquelle je ne mettrais pas ma main à couper. La question de l'informatique reste assez opaque, puisque très peu de gens s'y présentent (le rapport ne donne même pas les chiffres), et d'après mes préparateurs en 2018 : "L'option D, elle existe, mais personne n'en parle jamais dans les réunions, pas même le jury". D'après mes observations, il y aurait fort peu d'informaticiens présents à l'oral…

TABLE DES MATIÈRES

TABLE DES FIGURES

Première partie

Développements

CHAPITRE 1

DÉVELOPPEMENTS DE MATHÉMATIQUES

Ce bruit n'est rien, sur mon honneur !
C'est le gai carnaval qui dehors parle en maître.
Laissez-le sous votre fenêtre
Agiter son grelot moqueur

Benvenuto Cellini, Hector Berlioz, Léon de Wailly, Auguste Barbier

AA Théorème de Frobenius-Zolotarev & application.

Leçons possibles : 104 105 106 120 121 123 152
Adapté depuis : [BMP05],

Elegance : ★★★★★

Avis.

Ce développement complète une discussion sur le groupe linéaire des corps finis (un riche exemple de groupe fini), ou sur les liens entre signature et déterminant[1], ou encore sur le déterminant *per se* (voir le groupe dérivé de GL). Les aspects arithmétiques sont aussi à garder en mémoire. Les diagrammes commutatifs effrayent sans raison le peuple, mais ils sont un formidable outil pour savoir ce qu'il faut prouver : il suffit d'y "lire" les propriétés universelles à appliquer.

Prérequis.

1. Groupes linéaires et transvections [Per98], mais contrairement à lui, on travaillera avec des matrices. Notre objectif est de déterminer le groupe dérivé de $\mathrm{GL}_n(\mathbb{K})$.

   Définition AA.1. *Une* matrice de transvection *est une matrice de la forme* $T_{i,j}(\lambda) = I_n + \lambda E_{i,j}$ *avec* $i \neq j$ *et* $\lambda \in \mathbb{K}^*$. *Une* matrice de dilatation *est une* $D_i(\lambda) = diag(1, \dots, 1, \lambda, 1, \dots, 1)$ *avec* $\lambda \in \mathbb{K}^*$ *en position* i.

   Proposition AA.2. *Les matrices de transvection engendrent* $\mathrm{SL}_n(\mathbb{K})$.

   Multiplier à droite (resp. à gauche) par une matrice de transvection $T_{i,j}(\lambda)$ revient à faire l'opération élémentaire $C_i \leftarrow C_i + \lambda C_j$ (resp. $L_i \leftarrow L_i + \lambda L_j$).

   > *Preuve.* Remarquons d'abord que les transvections sont de déterminant 1.
   > Ensuite prenons $A \in \mathrm{SL}_n(\mathbb{K})$, en multipliant par des transvections à gauche (pivot de Gauss) on peut ramener A à une matrice triangulaire supérieure (permuter des lignes est possible avec des transvections, à un signe près). En remontant le système (transvections à droite) on rend A diagonale.
   > Donc il reste à montrer que si $A = \mathrm{diag}(a_1, \dots, a_n)$ avec $\prod_i a_i = 1$, alors A est produit de transvections. Il suffit en fait de le faire pour $A = \begin{pmatrix} a & 0 \\ 0 & \frac{1}{a} \end{pmatrix} \in \mathrm{M}_2(\mathbb{R})$.
   > On se convainc alors que c'est possible. □

   Remarque AA.3. *Les matrices de transvection et de dilatation engendrent* $\mathrm{GL}_n(\mathbb{K})$. *On peut le prouver en refaisant le pivot de Gauss dans toute sa généralité, ou exhiber le produit semi-direct* $\mathrm{GL}_n(\mathbb{K}) = \mathrm{SL}_n(\mathbb{K}) \rtimes \mathbb{K}^*$ *et utiliser AA.2.*

   Proposition AA.4. *Les matrices de transvection sont toutes conjuguées dans* $\mathrm{GL}_n(\mathbb{K})$.

   > *Preuve.* Il suffit d'exhiber le changement de base pour aller de $T_{i,j}(\lambda)$ à $T_{n-1,n}(1)$, qui permute les vecteurs et normalise le λ.
   >
   > □

   Remarque AA.5. *Si* $n \geq 3$, *elles sont mêmes conjuguées dans* $\mathrm{SL}_n(\mathbb{K})$.

   Proposition AA.6. *En caractéristique différente de* 2, *on a toujours* $\mathcal{D}(\mathrm{GL}_n(\mathbb{K})) = \mathrm{SL}_n(\mathbb{K})$.

   > *Preuve.* On traite à part le cas du groupe abélien $\mathrm{GL}_1(\mathbb{K})$. Soit dans la suite $n > 1$, les commutateurs sont de déterminant 1, et donc $\mathcal{D}(\mathrm{GL}_n(\mathbb{K})) \subseteq \mathrm{SL}_n(\mathbb{K})$.
   > Comme $\mathrm{car}(\mathbb{K}) \neq 2$, $T_{i,j}(\lambda)^2 = (I_n + E_{i,j}(\lambda))^2 = I_n + 2E_{i,j}(\lambda) = T_{i,j}(2\lambda)$ est encore une matrice de transvection. Donc par conjugaison AA.4, il existe $P \in \mathrm{GL}_n(\mathbb{K})$ telle que $T_{i,j}(\lambda)^2 = PT_{i,j}(\lambda)P^{-1}$ d'où $T_{i,j}(\lambda) = PT_{i,j}(\lambda)P^{-1}T_{i,j}(\lambda)^{-1}$ et c'est un

1. Mais un lien inattendu, pas celui de la définition du déterminant.

commutateur. Ainsi toutes les matrices de transvection sont dans le groupe dérivé, et par engendrement AA.2 on a $\mathrm{SL}_n(\mathbb{K}) \subseteq \mathcal{D}(\mathrm{GL}_n(\mathbb{K}))$.

□

Remarque AA.7. *Il aurait suffit de montrer qu'une matrice de transvection (une seule) était dans* $\mathrm{GL}_n(\mathbb{K})$*. En effet, le groupe dérivé est distingué*[1] *et les matrices de transvections sont conjuguées dans* $\mathrm{GL}_n(\mathbb{K})$ *AA.4, donc si on en a une on les a toutes.*

Remarque AA.8. *Si on veut faire du zèle, on établit en raffinant les arguments précédents :*
— $\mathcal{D}(\mathrm{GL}_n(\mathbb{K})) = \mathrm{SL}_n(\mathbb{K})$ *sauf si* $\mathbb{K} = \mathbb{F}_2$ *et* $n = 2$, *.*
— $\mathcal{D}(\mathrm{SL}_n(\mathbb{K})) = \mathrm{SL}_n(\mathbb{K})$ *sauf si* $\mathbb{K} = \mathbb{F}_2, \mathbb{F}_3$.

2. Groupes multiplicatifs des corps finis.

Proposition AA.9. *Le groupe multiplicatif d'un corps (commutatif) fini est cyclique.*

Preuve. Notons $n = |\mathbb{K}^*|$ et écrivons les facteurs premiers $n = \prod_{i=1}^{k} p_i^{\alpha_i}$.
Montrons maintenant qu'il existe un élément d'ordre $p_i^{\alpha_i}$. En regardant les racines d'un polynôme, il existe x_i tel que $(x_i)^{n/p_i} \neq 1$. Soit $a_i = x_i^{n/p_i^{\alpha_i}}$, cet élément est d'ordre divisant $p_i^{\alpha_i}$. Or $a_i^{p_i^{\alpha_i - 1}} = x_i^{n/p_i} \neq 1$. Donc l'ordre est exactement $p_i^{\alpha_i}$.
Par suite, $\prod_i a_i$ est d'ordre n (car les ordres sont premiers 2 à 2).

□

3. Symbole de Legendre. Dans cette partie p est un nombre premier impair.

Définition AA.10. *On définit le* symbole de Legendre $\left(\frac{\cdot}{p}\right) : \mathbb{F}_p^* \to \{\pm 1\}$ *tel que* $\left(\frac{x}{p}\right) = 1$ *ssi* x *est un carré dans* $\mathbb{F}_p^*$.

Proposition AA.11. *Le symbole de Legendre vérifie* $\left(\frac{x}{p}\right) = x^{\frac{p-1}{2}}$ *et c'est donc un morphisme de groupes entre* $\mathbb{F}_p^*$ *et* $\{\pm 1\}$.

Idée de preuve. S'obtient en dénombrant l'image de $x \mapsto x^2$..

□

Proposition AA.12. *Le symbole est l'unique morphisme de groupes non-trivial entre* $\mathbb{F}_p^*$ *et* $\{\pm 1\}$.

Preuve. On a vu que c'était un morphisme de groupes, en outre il n'est pas trivial car x un générateur de $\mathbb{F}_p^*$ (cyclique[2]) ne peut pas vérifier $x^{\frac{p-1}{2}} = 1$. En fait, un morphisme est uniquement déterminé par l'image de x qui vaut soit -1 (donc c'est Legendre) ou 1 (morphisme trivial).

□

Développement.

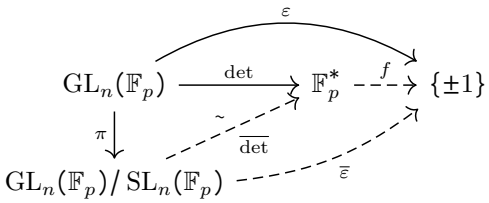
Théorème AA.13 (Frobenius-Zolotarev). *Soit* p *premier impair. Si* $u \in \mathrm{GL}_n(\mathbb{F}_p)$, *sa signature vérifie* $\varepsilon(u) = \left(\frac{\det(u)}{p}\right)$ *(symbole de Legendre).*

Preuve. 1. Existence (et unicité) d'une factorisation par le déterminant.

Lemme AA.14. *Il existe un morphisme*[3] *de groupes* $f : \mathbb{F}_p^* \to \{\pm 1\}$ *tel que pour tout* $u \in \mathrm{GL}_n(\mathbb{F}_p)$, $\varepsilon(u) = f \circ \det(u)$.

1. Et même caractéristique.
2. Noter qu'ici on n'utilise pas la structure générale du groupe multiplicatif d'un corps fini, mais seulement celle de $(\mathbb{Z}/p\mathbb{Z})^*$.
3. On peut montrer qu'il est unique si on en ressent le besoin, mais cela ne sert pas après. Plus exactement (avec les notations de la preuve), si on dispose de f' telle que $f' \circ \det = \varepsilon$, alors $f' \circ \overline{\det} \circ \pi = f' \circ \det = \varepsilon$ donc $f' \circ \overline{\det} = \overline{\varepsilon}$ par unicité dans la factorisation de ε. Et donc $f' = \overline{\varepsilon} \circ \overline{\det}^{-1} = f$.

Preuve. Puisque signature et déterminant sont des morphismes de groupes, on dessine le diagramme commutatif suivant :



On justifie la construction :
— π est la projection canonique vers le quotient ;
— le déterminant $\det$ se factorise en $\overline{\det}$ tel que $\overline{\det} \circ \pi = \det$ (et $\overline{\det}$ est un isomorphisme) puisque son noyau est exactement $\mathrm{SL}_n(\mathbb{F}_p)$ et qu'il est surjectif ;
— la signature ε se factorise en $\overline{\varepsilon}$ tel que $\overline{\varepsilon} \circ \pi = \varepsilon$. En effet, $\{\pm 1\}$ est un groupe abélien donc les commutateurs de $\mathrm{GL}_n(F_p)$ sont dans le noyau de ε. Par AA.6 on a donc $\mathrm{SL}_n(\mathbb{F}_p) = \mathcal{D}(\mathrm{GL}_n(\mathbb{F}_p)) \subseteq \mathrm{Ker}(\varepsilon)$ [1] ;
— en posant $f = \overline{\varepsilon} \circ \overline{\det}^{-1}$, on vérifie sur le diagramme que $f \circ \det = \overline{\varepsilon} \circ \overline{\det}^{-1} \circ \det = \overline{\varepsilon} \circ \pi = \varepsilon$.

□

2. La signature ε n'est pas triviale. En effet, $\mathbb{F}_{p^n} \simeq (\mathbb{F}_p)^n$ en tant qu'espaces vectoriels. Soit g est un générateur de $\mathbb{F}_{p^n}^*$ [2], alors $u : x \mapsto gx$ est un automorphisme $\mathbb{F}_p$-linéaire de $\mathbb{F}_{p^n}$. De plus sa signature vaut $(-1)^{p^n} = -1$ car il se décompose en tant que permutation en un point fixe (0) et un cycle de taille $p^n - 1$ (sur $\mathbb{F}_{p^n}^*$).
3. Donc $\overline{\varepsilon}$ est un morphisme non-trivial : c'est le symbole de Legendre d'après AA.12.

□

Application AA.15. *L'automorphisme de Frobenius F sur $\mathbb{F}_{p^n}$, vu comme élément de $\mathrm{GL}_n(\mathbb{F}_p)$ est cyclique, de déterminant $\det(F) = (-1)^{n+1}$ et de signature $\varepsilon(F) = (-1)^{(n+1)\frac{p-1}{2}}$.*

Preuve. Le polynôme minimal π_F de F est $P := X^n - 1$. En effet pour $x \in \mathbb{F}_{p^n}$, $P(F)(x) = (x^p)^n - x = x^{p^n} - x = 0$ (Lagrange). En outre, tout polynôme Q qui annule F est au moins de degré n, car $Q(F)(x)$ [3] possède p^n racines donc est au moins de degré p^n.
Comme π_F est de degré n, il est donc égal au polynôme caractéristique χ_F [4] et F est cyclique. D'autre part, $\det(F) = (-1)^n \chi_F(0) = (-1)^{n+1}$. D'après AA.13 juste prouvé, on a donc $\varepsilon(F) = (-1)^{(n+1)\frac{p-1}{2}}$.

□

Postrequis.

1. Et si $p = 2$? Deux cas se présentent.
Si $n \neq 2$, on a toujours $\mathcal{D}(\mathrm{GL}_n(\mathbb{F}_2)) = \mathrm{SL}_n(\mathbb{F}_2)$ et donc le morphisme f existe encore avec $f \circ \det = \varepsilon$. Mais puisque $\mathbb{F}_2^*$ est réduit à un élément, f est nécessairement trivial et la signature aussi : tous les automorphismes sont des permutations paires !
Lorsque $n = 2$, $\mathbb{F}_2^2 = \{(0,0),(0,1),(1,0),(1,1)\}$. Son groupe linéaire s'injecte dans les permutations de $\{(0,1),(1,0),(1,1)\}$ et il lui est en fait égal (on le vérifie à la main ou par cardinalité) donc isomorphe à $\mathfrak{S}_3$. Il y a alors 3 isomorphismes pairs (3-cycles et identité) et 3 impairs (transpositions).

1. Cette propriété est parfois appelée "Propriété universelle du groupe dérivé", L. Devilliers m'a suggéré qu'elle n'était pas assez standard pour ne pas être redémontrée.
2. Qui existe puisque $\mathbb{F}_{p^n}^*$ est cyclique, voir AA.9.
3. Vu comme vrai polynôme sur $\mathbb{F}_{p^n}$.
4. On peut y voir Cayley-Hamilton.

2. Application au calcul de $\left(\frac{2}{p}\right)$.

Proposition AA.16 (loi complémentaire). $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$.

Preuve. Dans $\mathbb{F}_p$, l'application $s : x \mapsto 2x$ est linéaire de déterminant 2. Sa signature $\varepsilon(s)$ est donc $\left(\frac{2}{p}\right)$ par AA.13. Calculons maintenant $\varepsilon(s)$ par les inversions.
Soit $P := \lfloor p/2 \rfloor = \frac{p-1}{2}$. En prenant un système de représentants $\{0, \dots, p-1\}$, on a $s(x) = 2x$ si $0 \leq x \leq P$; $2x - p$ si $P + 1 \leq x \leq p - 1$. Une inversion de s sur $i < j$ ne peut intervenir que si i et j ne sont pas du même côté de P.
Le nombre d'inversions $N(s)$ est $\mathrm{Card}\left(\biguplus_{i=0}^{P}\{P + 1 \leq j \leq p - 1 \mid s(j) < s(i)\}\right)$
$= \mathrm{Card}\left(\biguplus_{i=0}^{P}\{P + 1 \leq j \leq i + P\}\right) = \sum_{i=0}^{P} i = \frac{P(P+1)}{2} = \frac{p^2+1-2p+2p-2}{8} = \frac{p^2-1}{8}$.
D'où $\varepsilon(s) = \left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$. □

3. Application à la loi de réciprocité quadratique.

AB Théorème de Brauer.

Leçons possibles : 104 105 108
Pas de référence.

Elegance : ★★★★☆

Avis.

Un développement qui contient presque toutes les manipulations spécifiques aux permutations. D'un point de vue actions du groupe symétrique, on fait le lien entre deux d'entre elles. Dans la leçon de parties génératrices, on insistera bien entendu sur l'intérêt de la décomposition en cycles à supports disjoints. On pourra aussi évoquer (sans le développer) ce résultat dans les Déterminants, comme application du calcul de celui de Smith.

Prérequis.

1. Déterminants.

   Proposition AB.1 (déterminant de Smith, [Gou09a]). $\det((i \wedge j)_{1\leq i,j\leq n}) = \prod_{i=1}^{n} \varphi(i)$.

   Preuve. Notons $A = (i \wedge j)_{1\leq i,j\leq n}$. On sait que $\forall m \geq 0$, $m = \sum_{k|m} \varphi(k)$ donc ici $i \wedge j = \sum_{k|i\wedge j} \varphi(k) = \sum_{k|i \text{ et } k|j} \varphi(k)$.
   On considère les matrices $D = \text{diag}(\varphi(1), \dots, \varphi(n))$ et $B = (\chi_{i|j})_{1\leq i,j\leq n}$ qui est triangulaire supérieure avec des 1 sur la diagonale. On vérifie (c'est un calcul) que $A = {}^tBDB$ et enfin $\det(A) = 1 \times \det(D) \times 1 = \prod_{i=1}^{n} \varphi(i)$. □

2. Propriétés du groupe symétrique.

   Proposition AB.2. *Deux permutations sont conjuguées (dans $\mathfrak{S}_n$) si et seulement si elles ont même type (multiensemble des longueurs de leurs cycles à supports disjoints).*

   Remarque AB.3. *L'existence de la décomposition en cycles à supports disjoints de σ s'obtient en regardant les orbites de l'action du groupe $\langle\sigma\rangle$ sur l'ensemble.*

Développement.

Théorème AB.4. *Soit $\mathbb{K}$ un corps. Deux permutations $\sigma_1, \sigma_2 \in \mathfrak{S}_n$ sont conjuguées (dans $\mathfrak{S}_n$) si et seulement si les matrices de permutations associées P_{σ_1} et P_{σ_2} le sont dans $\text{GL}_n(\mathbb{K})$.*

Preuve. Pour le sens direct, si $\sigma_1 = \tau\sigma_2\tau^{-1}$ alors $P_{\sigma_1} = P_\tau P_{\sigma_2} P_{\tau^{-1}} = P_\tau P_{\sigma_2} {P_\tau}^{-1}$.

On suppose maintenant que $P_{\sigma_1} = AP_{\sigma_2}A^{-1}$ et on va montrer qu'alors σ_1 et σ_2 sont de même type. Notons $n(\sigma)$ le nombre de cycles dans la décomposition de σ, et $d_k(\sigma)$ le nombre de cycles de longueur k. On veut donc montrer que $\forall 1 \leq k \leq n$, $d_k(\sigma_1) = d_k(\sigma_2)$.

Lemme AB.5. *Si V_σ est le sous-espace des vecteurs stables*[1] *par P_σ, alors $n(\sigma) = \dim(V_\sigma)$.*

Preuve. En effet, un vecteur $v = (v_1, \dots, v_n)$ est stable si et seulement si $\forall i, v_i = v_{\sigma(i)}$[2]. Cela signifie que si i et j sont dans la même orbite alors $v_i = v_j$. Les vecteurs $(\chi_{i\in c})_{1\leq i\leq n}$ pour chaque cycle c de la décomposition de σ forment une base de V_σ, de même cardinal que le nombre de cycles. □

Puisque P_{σ_1} et P_{σ_2} sont conjuguées, alors $P_{\sigma_1^m}$ et $P_{\sigma_2^m}$ aussi. Donc $n(\sigma_1^m) = n(\sigma_2^m)$.

Lemme AB.6 (Eclatement des cycles). *Si $\psi \in \mathfrak{S}_n$ est un k-cycle alors $\forall m \geq 1$, ψ^m est le produit de $k \wedge m$ cycles à supports disjoints de taille $\frac{k}{k\wedge m}$.*

1. Vecteurs propres associés à la valeur propre 1.
2. Puisque la matrice de permutation permute les coordonnées.

Preuve. On peut sans perte de généralité supposer $\psi = (1, \dots, k)$. Notons $c_m(x)$ pour $1 \leq x \leq k$ le cycle de la décomposition de ψ^m contenant x.
Alors $c_m^i = \mathrm{Id}$ ssi $c_m^i(x) = x$ ssi $(1, \dots, k)^{mi}(x) = x$ ssi $\psi^{mi} = \mathrm{Id}$[1]. Ceci est équivalent à dire que $k | mi$ soit $\frac{k}{k \wedge m} | i$[2].
Donc l'ordre de c_m est $\frac{k}{k \wedge m}$. Par suite la décomposition se compose de $k \wedge m$ cycles d'ordre $\frac{k}{k \wedge m}$ chacun.

□

Etant donné l'éclatement de chaque cycle sur son support, on conclut que le nombre d'orbites de σ^m est $n(\sigma^m) = \sum_{k=1}^{m} (k \wedge m) d_k(\sigma)$.

Enfin, on a donc $\sum_{k=1}^{m} (k \wedge m) d_k(\sigma_1) = \sum_{k=1}^{m} (k \wedge m) d_k(\sigma_2)$ pour tout $m \geq 0$. La matrice des $(k \wedge m)_{1 \leq k, m \leq n}$ étant inversible, on conclut que $d_k(\sigma_1) = d_k(\sigma_2)$ et ainsi les permutations sont conjuguées car de même type.

□

Postrequis.

1. Dans le cas où le corps est $\mathbb{C}$[3], on peut faire plus rapide.

   Lemme AB.7. *La matrice de permutation associée au cycle $(1, \dots, n)$ est la matrice compagnon du polynôme $X^n - 1$.*

   Preuve. Il suffit de l'écrire : des 1 sur la sur-diagonale plus un 1 dans le coin.

   □

   Avec les notations précédentes, le polynôme caractéristique $\chi(\sigma)$ est $\prod_k (X^k - 1)^{d_k(\sigma)}$[4].
   Etant donné que c'est un invariant de similitude, on en déduit que si P_{σ_1} et P_{σ_2} sont semblables, alors $\prod_k (X^k - 1)^{d_k(\sigma_1)} = \prod_k (X^k - 1)^{d_k(\sigma_2)}$.
   En évaluant la multiplicité d'une racine de l'unité $\xi = \exp\left(\frac{2\,\mathrm{i}\,\pi}{k}\right)$ dans ce polynôme, on conclut que $\sum_{k|m} d_k(\sigma_1) = \sum_{k|m} d_k(\sigma_2)$. On trie alors le résultat par une récurrence.

1. En effet un cycle est l'identité si et seulement si il a un point fixe dans son support.
2. On factorise pour obtenir $\frac{k}{k \wedge m} | \frac{m}{k \wedge m} i$ puis par le lemme de Gauss, on déduit $\frac{k}{k \wedge m} | i$.
3. A des racines de l'unité suffit.
4. En découpant par blocs.

AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.

Leçons possibles : 108 182 183
Adapté depuis : [Per98], 2

Elegance : ★★★★☆

Avis.

Le chapitre VII. Quaternions de [Per98] est très bien fait. On ne vise pas ici d'aller très loin dans la théorie, mais simplement de voir comment on pourrait paramétrer $\mathrm{SO}_3(\mathbb{R})$ a l'aide d'un corps bien choisi, comme on l'a fait avec $\mathbb{C}$ pour $\mathrm{SO}_2(\mathbb{R}) \simeq \mathbb{U}$. Attention, comme le groupe n'est plus commutatif, il y a peu de chances que le "corps" le soit... C'est aussi un joli usage des générateurs de $\mathrm{SO}_3(\mathbb{R})$ pour montrer la surjectivité d'un morphisme.

Prérequis.

1. Quaternions : existence et premières propriétés.

   Proposition AC.1. *Le centre de $\mathbb{H}$ est $\mathbb{R}$.*

2. Générateurs de $\mathrm{SO}_3(\mathbb{R})$.

   Définition AC.2. *Une* réflexion hyperplane *une symétrie orthogonale par rapport à un hyperplan (parallèlement à une droite).*

   Ces réflexions *renversent une droite* en *préservant un plan*. Dans une certaine base orthonormée, leur matrice est de la forme $\mathrm{diag}(1, \dots, 1, -1)$ et sont donc de $\det = -1$. En particulier, les symétries axiales orthogonales sont des réflexions hyperplanes.

   Proposition AC.3. *Les réflexions orthogonales engendrent $\mathrm{O}_2(\mathbb{R})$.*

   Proposition AC.4. *Les réflexions hyperplanes engendrent $\mathrm{O}_n(\mathbb{R})$.*

   > *Idée de preuve.* L'idée est de réduire la matrice (réduction des endomorphismes normaux), puis de traiter indépendamment les sous-espace stables de dimension 2 qui sont apparus.
   >
   > □

   Définition AC.5. *Une* retournement *une symétrie orthogonale par rapport à un sous-espace de codimension 2 (parallèlement à une plan).*

   Ici on *renverse un plan*. Dans une certaine base orthonormée, leur matrice est de la forme $\mathrm{diag}(1, \dots, 1, -1, -1)$ et sont donc de $\det = 1$, donc dans $\mathrm{SO}_n(\mathbb{R})$.

Développement.

Lemme AC.6. *$\mathrm{SO}_3(\mathbb{R})$ est engendré par les retournements.*

> *Preuve.* Si $u \in \mathrm{SO}_3(\mathbb{R})$, il est le produit de réflexions par AC.4 en nombre pair (déterminant). Or en dimension 3, l'opposé d'une réflexion est un retournement (voir la matrice). En ajoutant des $-$ qui se simplifient par parité on obtient le résultat.
>
> □

Soit G la sphère des quaternions de norme 1, c'est un sous-groupe de $\mathbb{H}^*$.

Théorème AC.7. *Il existe un isomorphisme de groupes $G/\{\pm 1\} \simeq \mathrm{SO}_3(\mathbb{R})$.*

> *Preuve.* 1. On fait agir G sur $\mathbb{H}$ par conjugaison : pour $q \in G$, soit $S_q : q' \mapsto qq'q^{-1}$. Alors S_q est bijective d'inverse $S_{q^{-1}} = S_{\overline{q}}$ et $\mathbb{R}$-linéaire. En outre, $S : G \to \mathrm{GL}_4(\mathbb{R}), q \mapsto S_q$ est un morphisme de groupes.

2. En fait, $S_q \in \mathrm{O}(N) = \mathrm{O}_4(\mathbb{R})$ car il préserve la norme [1]. Or $\mathbb{R}$ est stable par S_q, donc son orthogonal $P := \mathrm{Vect}(\mathrm{i}, \mathrm{j}, \mathrm{k})$ des quaternions purs est aussi stable.
   En notant $s_q := S_q|_P$, on obtient un morphisme de groupes $s : G \to \mathrm{O}_3(\mathbb{R}), q \mapsto s_q$.
3. $\mathrm{Ker}(s) = \{q \in G \mid \forall q' \in P, qq' = q'q\} = \{q \in G \mid \forall q' \in \mathbb{H}, qq' = q'q\}$ car $\mathbb{R}$ est central. C'est donc $\mathbb{R} \cap G = \{+1, -1\}$ par AC.1.
4. L'application $\det \circ s : G \to \mathbb{R}$ est polynomiale en les coordonnées donc continue (pour les topologies induites). Or G est connexe (par arcs) donc $\det \circ s(G)$ aussi et c'est $\{+1\}$. Donc $s(G) \subseteq \mathrm{SO}_3(\mathbb{R})$.
5. Si $p \in P \cap G$, $s_p(p) = pp\overline{p} = pN(p)^2 = p$. Donc s_p est une rotation d'axe $\mathrm{Vect}(p)$. En outre $\overline{p} = -p$ soit $p^2 = -1$ et donc $(s_p)^2 = (s_{p^2}) = s_{-1} = \mathrm{id}$. Comme il est involutif, s_p est le retournement d'axe $\mathrm{Vect}(p)$. Puisque les retournements sont générateurs par AC.6 il vient $s(G) = \mathrm{SO}_3(\mathbb{R})$.

On conclut en factorisant par le noyau que $G/\{\pm 1\} \simeq \mathrm{SO}_3(\mathbb{R})$.

□

Postrequis.

1. On en déduit que $\mathrm{SO}_3(\mathbb{R})$ est connexe (dans la preuve, on a utilisé seulement le fait que $\mathrm{SO}_3(\mathbb{R})$ et $\mathrm{O}_3^-(\mathbb{R})$ sont dans deux composantes différentes).
2. De l'intérêt de la représentation par les quaternions.
   En animation 3D, on va chercher à représenter des rotations. Utiliser un quaternion plutôt qu'une matrice 3×3 de $\mathrm{SO}_3(\mathbb{R})$ possède plusieurs avantages :
   (a) 4 nombres en mémoire au lieu de 9 ;
   (b) on peut toujours calculer sans difficultés le produit de deux rotations, ou l'image d'un vecteur par une rotation ;
   (c) on peut très facilement extraire l'axe (c'est sa partie "imaginaire" [2]) et l'angle de la rotation ;
   (d) on peut gérer les erreurs d'arrondi. En effet, après plusieurs approximations, une matrice de $\mathrm{SO}_3(\mathbb{R})$ arrondie n'est plus nécessairement orthogonale, il faut alors la re-orthogonaliser et c'est compliqué. Alors que re-orthogonaliser un quaternion, c'est tout facile (on le renormalise) !
3. Générateurs de $\mathrm{SO}_n(\mathbb{R})$, cas général.
   Proposition AC.8. *Pour $n \geq 3$, $\mathrm{SO}_n(\mathbb{R})$ est engendré par les retournements.*

   Preuve. On procède de manière similaire au cas $n = 3$ ci-haut. En général, on a encore que tout élément de $\mathrm{SO}_n(\mathbb{R})$ s'obtient comme produit d'un nombre pair de réflexions hyperplanes. On va trouver un sous-espace de dimension 3 sur lequel on peut prendre l'opposé de deux réflexions et obtenir des retournements.
   Soient s_1, s_2 deux réflexions hyperplanes d'axes D_1 et D_2. S'ils sont égaux, les deux réflexions sont égales et $s_1 \circ s_2$ est l'identité. Sinon, on pose $F := \mathrm{Vect}(x_1, x_2)$ de dimension 2, alors sur $F^\perp$ les réflexions s_1 et s_2 sont l'identité. Donc si H est un hyperplan de F, H est stable par s_1 et s_2, et donc $H^\perp$ aussi.
   En outre $\mathrm{Vect}(D_1, D_2) \subseteq H^\perp$ et ce dernier est de dimension 3. Donc $s_1|_{H^\perp}$ et $s_2|_{H^\perp}$ sont des réflexions et leurs opposés sont des retournements. On conclut en les ré-étendant sur H par l'identité.

   □
4. D'autres isomorphismes de quaternions.

1. $N(qq'q^{-1}) = 1 \times N(q) \times 1$.
2. En effet si $q = a + p$ avec $p \in P$ alors $S_q(p) = (a+p)p\overline{(a+p)} = a^2p + ap\overline{p} + ap^2 + pp\overline{p} = (a^2 + N(p)^2)p + ap\overline{p} + ap^2$ or $a^2 + N(p)^2 = 1$ et $\overline{p} = -p$ d'où on trouve $S_q(p) = p$ fixe.

Théorème AC.9 ([Per98]). *On a les isomorphismes suivants :*

(i) $G \times G/(1,1) \sim (-1,-1) \simeq \mathrm{SO}_4(\mathbb{R})$ *;*

(ii) $\mathrm{PO}_4(\mathbb{R}) \simeq \mathrm{SO}_3(\mathbb{R}) \times \mathrm{SO}_3(\mathbb{R})$ [1] *;*

(iii) $\mathrm{SU}_2(\mathbb{C}) \simeq G$.

1. Et il n'est donc pas simple.

AD Ordre moyen de l'indicatrice d'Euler.

Leçons possibles : 120 121 190 224 223 230
Adapté depuis : [FGN01a],

Elegance : ★★★☆☆

Avis.

Le FGN détermine la "probabilité" que deux entiers soient premiers entre eux, mais c'est plus parlant - notamment dans les leçons d'arithmétique - de prétendre étudier la moyenne asymptotique de $\varphi(n)$ (sans que ce soit très différent). Toute la philosophie autour de la fonction de Möbius (mais aussi du crible de Poincaré) rentre à merveille dans une partie adaptée de la leçon de combinatoire. Ce développement est aussi intéressant dans quelques leçons d'analyse, bien qu'il semble excessif de le présenter en probabilités.

Prérequis.

1. Fonction de Möbius, définition et propriétés.

   Définition AD.1 (fonction de Möbius). *Soit $\mu : \mathbb{N}^* \to \{-1, 0, 1\}$ telle que $\mu(n) = 0$ si n est divisible par un carré > 1 ; $(-1)^k$ sinon où k est le nombre de facteurs premiers de n.*

   Remarque AD.2. *$\mu(n) = 0$ ssi il existe un nombre premier p tel que $v_p(n) \geq 2$.*

   Remarque AD.3 (multiplicativité). *$\mu(mn) = \mu(m)\mu(n)$.*

   Lemme AD.4. *Si $n > 1$, $\sum_{d|n} \mu(d) = 0$.*[1]

   > *Preuve.* Soit P l'ensemble des facteurs premiers distincts de n. Alors les seuls termes non-nuls de la somme sont $\mu(d)$ pour $d = \prod_{p\in D} p$ pour $D \subseteq P$. Il vient donc $\sum_{d|n} \mu(d) = \sum_{D\subseteq P}(-1)^{|D|} = \sum_{k=0}^{|P|} \binom{|P|}{k}(-1)^k = 0$ en réindexant selon le cardinal de D.
   >
   > □

   Corollaire AD.5 (inversion de Möbius). *Soient $f, g : \mathbb{N}^* \to \mathbb{N}$ telles que $g(n) = \sum_{d|n} f(d)$, alors $f(n) = \sum_{d|n} \mu(d)g(\frac{n}{d})$.*

   > *Preuve.* En effet, $\sum_{d|n} \mu(d)g(\frac{n}{d}) = \sum_{d|n} \mu(\frac{n}{d})g(d) = \sum_{d|n} \mu(\frac{n}{d}) \sum_{k|d} f(k)$
   > $= \sum_{k|d|n} \mu(\frac{n}{d})f(k) = \sum_{k|n} f(k) \sum_{k|d|n} \mu(\frac{n}{d}) = \sum_{k|n} f(k) \sum_{p|\frac{n}{k}} \mu(p)$.
   > En vertu de AD.4, seul le terme en $k = n$ est non-nul, ce qui conclut.
   >
   > □

   Application AD.6. *$\varphi(n) = \sum_{d|n} \mu(d)\frac{d}{n}$*[2].

2. Formule du crible de Poincaré.

   Proposition AD.7 (crible). *Si $(U_i)_{1\leq i\leq n}$ est une famille de sous-ensembles finis d'un ensemble E, alors*

   $$Card\left(\bigcup_{1\leq i\leq k} U_i\right) = \sum_{\substack{I\neq\emptyset \\ I\subseteq[\![1,k]\!]}} (-1)^{|I|+1} Card\left(\bigcap_{i\in I} U_i\right).$$

   > *Preuve.* Par récurrence et c'est peu gratifiant.
   >
   > □

3. Un peu de théorie des famille sommables.

1. Ce lemme pourra éventuellement être intégré au développement.
2. Voir la Variante des Postrequis.

Développement.

Notons $A_n = \{(a,b) \in [\![1,n]\!]^2 \mid a \wedge b = 1\}$ et $r_n := \frac{1}{n^2}\text{Card}(A_n)$

Théorème AD.8. $r_n \to \frac{6}{\pi^2}$

Preuve. 1. Expression de A_n.

Notons que $A_n = [\![1,n]\!]^2 \setminus \bigcup_{1\le i\le k} U_i$ où $p_1,\dots,p_k$ sont les nombres premiers inférieurs à n et $U_i = \{(a,b) \in [\![1,n]\!]^2 \mid p_i|a \text{ et } p_i|b\}$ [1]

Donc $\text{Card}(A_n) = n^2 - \text{Card}\left(\bigcup_{1\le i\le k} U_i\right)$ puis d'après le crible AD.7 :

$$= n^2 - \sum_{\substack{I\neq\emptyset \\ I\subseteq[\![1,k]\!]}} (-1)^{|I|+1}\text{Card}\left(\bigcap_{i\in I} U_i\right)$$

Or $\text{Card}\left(\bigcap_{i\in I} U_i\right) = \left\lfloor \frac{n}{\prod_{i\in I} p_i} \right\rfloor^2$ puisque c'est le nombre de couples d'éléments tous deux divisibles par chaque p_i donc par leur produit (Euclide).

On conclut que $\text{Card}(A_n) = n^2 + \sum_{\substack{I\neq\emptyset \\ I\subseteq[\![1,k]\!]}} (-1)^{|I|}\left\lfloor \frac{n}{\prod_{i\in I} p_i} \right\rfloor^2$

$$= \sum_{d=1}^{n} \mu(d)\left\lfloor \frac{n}{d} \right\rfloor^2 \text{ en reconnaissant les termes [2].}$$

2. Simplification de la limite [3].

Il vient donc $r_n = \frac{1}{n^2}\sum_{d=1}^n \mu(d)\left\lfloor \frac{n}{d} \right\rfloor^2$.

Soit $\varepsilon_n = \left|r_n - \frac{1}{n^2}\sum_{d=1}^n \frac{\mu(d)}{d^2}\right| = \left|\sum_{d=1}^n \mu(d)\left(\frac{1}{n^2}\left\lfloor \frac{n}{d} \right\rfloor^2 - \frac{1}{d^2}\right)\right| \le \sum_{d=1}^n \left|\frac{1}{n^2}\left\lfloor \frac{n}{d} \right\rfloor^2 - \frac{1}{d^2}\right|$.

Or $\frac{n}{d} - 1 < \left\lfloor \frac{n}{d} \right\rfloor \le \frac{n}{d}$ donc $\frac{1}{n^2} - \frac{2}{nd} + \frac{1}{d^2} < \frac{1}{n^2}\left\lfloor \frac{n}{d} \right\rfloor^2 \le \frac{1}{d^2}$.

Donc $\varepsilon_n \le \sum_{d=1}^n \left(\frac{2}{dn} + \frac{1}{n^2}\right) = \mathcal{O}\left(\frac{\ln(n)}{n} + \frac{1}{n}\right) \to 0$.

Enfin $\lim_{n\to+\infty} r_n = \sum_{d=1}^{+\infty} \frac{\mu(d)}{d^2}$ (et la limite existe bien).

3. Calcul de la limite par un produit de Dirichlet.

Par sommabilité il vient $\sum_{d=1}^{+\infty} \frac{\mu(d)}{d^2} \times \sum_{n=1}^{+\infty} \frac{1}{n^2} = \sum_{d,n\ge1} \frac{\mu(d)}{d^2n^2} = \sum_{p=1}^{+\infty}\sum_{d|p} \frac{\mu(d)}{p^2}$

$$= \sum_{p=1}^{+\infty} \frac{1}{p^2}\sum_{d|p} \mu(d) = 1 \text{ par la pré-inversion AD.4.}$$

Donc $r_n \to \sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$ [4].

□

Corollaire AD.9. $\sum_{k=1}^n \varphi(k) \sim \frac{3}{\pi^2}n^2$.

Preuve. Rappelons que $\varphi(k) = \text{Card}(\{1 \le b \le k \mid b \wedge k = 1\})$. En outre $A_n = \{(1,1)\} \uplus \{(a,b) \in [\![1,n]\!]^2 \mid a < b \text{ et } a \wedge b = 1\} \uplus \{(a,b) \in [\![1,n]\!]^2 \mid a > b \text{ et } a \wedge b = 1\}$.

Donc $\text{Card}(A_n) = -1 + 2\sum_{k=1}^n \varphi(k)$ [5] ce qui donne $\sum_{k=1}^n \varphi(k) \sim \frac{3}{\pi^2}n^2$ par AD.8.

□

1. Puisque diviser a et b, c'est diviser $a \wedge b$.
2. Dans la première somme, les termes valent 0 dès que $\prod_{i\in I} p_i$ dépasse n. On retrouve alors tous les termes où d est sans facteur carré de la seconde somme (donc non-nuls), le cas $d = 1$ venant du n^2.
3. Intuitivement, on veut remplacer $\left\lfloor \frac{n}{d} \right\rfloor^2$ par son équivalent $\frac{n^2}{d^2}$, ce qu'on justifie proprement.
4. Résultat non trivial, on peut citer les séries de Fourier pour l'obtenir.
5. Il faudrait compter deux fois $(1,1)$, d'où le -1.

Postrequis.

1. Variante pour obtenir directement AD.9.
   $\sum_{k=1}^{n} \varphi(k) = \sum_{k=1}^{n} \sum_{d|k} \mu(d)\frac{n}{d} = \sum_{d|k\leq n} \mu(d)\frac{k}{d}$ par AD.6
   $= \sum_{d=1}^{n} \mu(d) \sum_{dp\leq n} p = \sum_{d=1}^{n} \mu(d)\lfloor \frac{n}{d} \rfloor (\lfloor \frac{n}{d} \rfloor + 1)/2.$
   On reconnaît alors, au $/2$ près, une formule sensiblement identique à ce que l'on avait pour Card(A_n). Pourquoi alors ne pas faire cela ?
2. Une autre expression de $\varphi(n)$.
   Proposition AD.10. *Si* $n = \prod p_i^{\alpha_i}$ *(facteurs premiers), alors* $\varphi(n) = n\prod(1 - \frac{1}{p_i})$.

   Idée de preuve. On montre d'abord la multiplicativité de l'indicateur d'Euler : si $m \wedge n = 1$ alors $\varphi(mn) = \varphi(m)\varphi(n)$ (avec le théorème chinois). Ensuite, on évalue sa valeur sur les puissances de nombres premiers, et on conclut.

   □
3. On a obtenu un équivalent de le moyenne, mais cela ne donne pas de résultat sur φ ! Et pour cause, cette fonction est par trop irrégulière.
   Proposition AD.11 ([QZ13], ex 3 p 8 et ex 3 p 32)**.** $\limsup \frac{\varphi(n)}{n} = 1$ *et* $\liminf \frac{\varphi(n)}{n} = 0$.

   Preuve. Notons d'abord que $0 \leq \varphi(n) \leq n$.
   Ensuite si n est premier alors $\frac{\varphi(n)}{n} = \frac{n-1}{n} \to 1$
   D'autre part si $n = p_1 \dots p_m$ avec $(p_i)_{i\geq 0}$ la suite des nombres premier alors $\frac{\varphi(n)}{n} = \prod_{i=1}^{m}(1 - \frac{1}{p_i})$. Son inverse est $\prod_{i=1}^{m} \frac{1}{1-\frac{1}{p_i}} = \prod_{i=1}^{m} \sum_{k=0}^{+\infty} \frac{1}{p_i^k}$.
   En développant le produit, on retrouve plein de termes de la série harmonique (décompositions de facteurs premiers) qui diverge.

   □
4. D'autres ordres moyens : τ nombre de diviseurs, σ somme des diviseurs,...
5. D'une manière générale, on a plus ou moins manipulé des séries de Dirichlet. Voir [QZ13].

AE Groupes d'isométries du cube et du tétraèdre. (NR)

Leçons possibles : 183

Elegance : ★★★☆☆

Développement non-rédigé...

AF Algorithme de Berlekamp.

Leçons possibles : 123 141 151 162
Adapté depuis : [BMP05], p 244

Elegance : ★★★★☆

Avis.

Ce n'est pas si dur. On prendra soin de s'inspirer à la fois de [BMP05] (qui en fait un peu trop) et de [Dem08] (un peu trop concis).
Développer cet algorithme dans la leçon 162 est une preuve de mauvaise foi, mais on insistera sur le fait que le pivot de Gauss permet de faire tout ce dont on rêve sur les systèmes linéaires... et que des problèmes inattendus s'y réduisent !

Prérequis.

1. Théorème chinois.
2. Algorithme de Berlekamp.

Algorithme 1 : Algorithme de Berlekamp

Entrée : $\mathbb{F}_{p^n}$ *corps fini de cardinal* $q = p^n$, $P \in \mathbb{F}_q[X]$ *non constant.*
Sortie : *L'irréductibilité de* P *ou un facteur non-trivial.*
si $P \wedge P' \neq P, 1$ **alors**
  retourner $P \wedge P'$
fin
si $P \wedge P' = P$ **alors**
  retourner Q *tel que* $Q^p = P$
fin
si $P \wedge P' = 1$ **alors**
  Soit $S : \mathbb{F}_q[X]/(P) \to \mathbb{F}_q[X]/(P), Q \mod P \mapsto Q^q \mod P$;
  si *dim*(*Ker*)$(S - Id) = 1$ **alors**
    retourner *Irréductible*
  sinon
  fin
  Trouver V tel que $V \mod P \in \mathrm{Ker}(S)$ non constant ;
  Trouver $\alpha \in \mathbb{F}_q$ tel que $P \wedge (V - \alpha) \neq P, 1$:
  retourner $P \wedge (V - \alpha)$
fin

Développement.

Théorème AF.1. *L'algorithme de Berlekamp permet effectivement*[1] *de déterminer l'irréductibilité de* P *sur* $\mathbb{F}_q$ *et donne un facteur non-trivial si possible.*

Preuve. On peut calculer les PCGD à l'aide de l'algorithme d'Euclide.

Si $P \wedge P' \neq P, 1$. Alors $P \wedge P'$ est un facteur non-trivial de P (qui n'est pas irréductible).

Si $P \wedge P' = P$. Alors $P' = 0$ puisque $\deg(P') < \deg(P)$. Cela n'est possible que si P s'écrit $\sum_{k=1}^{n} a_{pk} X^{pk} = \left(\sum_{k=1}^{n} \sqrt[p]{a_{pk}} X^k\right)^p$ (le morphisme de Frobenius permet la factorisation, et sa surjectivité assure l'existence des racines p-ièmes).

1. Comprendre "algorithmiquement", ou encore "the procedure is effective".

On pose donc $Q := \sum_{k=1}^{n} \sqrt[p]{a_{pk}} X^k = \sum_{k=1}^{n} (a_{pk})^{p^{n-1}} X^k$ [1] calculable.

Si $P \wedge P' = 1$. Si P a un facteur irréductible carré Q^2, alors $Q|P$ et $Q|P'$ [2] donc $Q|P \wedge P'$. Ce n'est pas possible ici. Donc $P = \prod_{i=1}^{r} P_i$ tous distincts.

1. L'application $\varphi : \mathbb{F}_q[X]/(P) \to \prod_i \mathbb{F}_q[X]/(P_i), Q \mod P \mapsto (Q \mod P_i)_i$ est un isomorphisme de $\mathbb{F}_q$-algèbres, d'après le théorème chinois.
   Par irréductibilité, chaque membre droit du produit est un corps.
   L'application [3] $S : Q \mod P \mapsto Q^q \mod P$ de $\mathbb{F}_q[X]/(P)$ est linéaire [4]. Via l'isomorphisme φ elle devient $\overline{S} : (Q \mod P_i)_i \mapsto (Q^q \mod P_i)_i$ [5].
   Or dans chaque $\mathbb{F}_q[X]/(P_i)$ surcorps de $\mathbb{F}_q$, $x^q = x$ si et seulement si $x \in \mathbb{F}_q$. Donc $\mathrm{Ker}(\overline{S} - \mathrm{Id}) = \prod_i \mathbb{F}_q$ et $\dim(\mathrm{Ker})(S - Id) = r$ (qui peut être calculé par le pivot de Gauss).
2. Si $r > 1$, on peut trouver V tel que $V \mod P \in \mathrm{Ker}(S)$ non constant (Gauss)
   Notons $\alpha_i := V \mod P_i \in \mathbb{F}_q$ [6]. Alors $P \wedge (V - \alpha) = \prod_{i \mid \alpha_i = \alpha} P_i$ [7].
   Donc $P = \prod_\alpha P \wedge (V - \alpha) = \prod_\alpha (\prod_{i \mid \alpha_i = \alpha} P_i)$.
   Puisque $V \mod P$ n'est pas constant, on ne peut pas avoir $P \wedge (V - \alpha) = P$ et donc les facteurs ne sont pas tous triviaux.

□

Postrequis.

1. Attention ! On savait *déjà* que l'on pouvait décider si un polynôme était irréductible : il suffisait de tester la divisibilité par tous polynômes de degré plus petit. L'intérêt de Berlekamp est qu'il est *a priori* plus efficace, mais ce n'est pas si clair car la dernière étape (essayer tous les α) peut être très long. Voir aussi la riante probabiliste de [Dem08].
2. Algorithme de Rabin.
   Eu égard à AG.2 qui va venir, on aurait une autre solution algorithmique pour savoir si P de degré n est irréductible (ou déterminer un facteur non-trivial) :
   — vérifier que $P | X^{q^n} - X$;
   — vérifier que pour tout $d < n$, $P \wedge (X^{q^d} - X) = 1$.
   Cela peut être vu comme un raffinement de la méthode débile proposée au point 1. Allez savoir maintenant qui est plus rapide, de lui ou de Berlekamp ? !

1. Puisque $x^q = x$ (Lagrange), donc $x^{p^{n-1}} = \sqrt[p]{x}$.
2. On fait le calcul.
3. Il est inutile de fondre en explications quant au fait que cette application est bien définie. Elle l'est ! Parce que l'écriture avec les modulos est celle d'un *anneau* quotient, où l'on peut faire des puissances sans se soucier des représentants.
4. Lagrange et Frobenius
5. Là encore c'est évident.
6. Puisque $P \in \mathrm{Ker}(S)$.
7. Admis faute de temps. Il suffit de regarder quand $P_i | V - \alpha \iff V = \alpha \mod P_i \iff \alpha = \alpha_i$.

AG Dénombrement des polynômes irréductibles sur $\mathbb{F}_q$.

Leçons possibles : 141
Pas de référence.

Elegance : ★★☆☆☆

Avis.

On le mentionnera dans la leçon de Dénombrement sans le proposer en développement, puisqu'il est trop proche de AD Ordre moyen de l'indicatrice d'Euler.. Ce serait aussi un bon développement dans les corps finis, puisqu'il manipule leur structure avec précision. Attention [Rom17] ne fait pas la même preuve : il construit les corps finis ainsi (alors que je le fais avant).

Prérequis.

Proposition AG.1. *Les polynômes irréductibles sur un corps fini sont séparables.*

Preuve. Comme $P \wedge P'$ est un diviseur non-trivial de P, c'est 1 ou P par irréductibilité. Supposons que ce soit P, alors il existe Q tel que $Q^p = P$ (cf AF Algorithme de Berlekamp.). Cela contredit l'irréductibilité.
Donc $P \wedge P' = 1$, et P est à racines simples dans une extension de décomposition. □

Développement.

Soit $\mathbb{F}_q$ un corps fini, on note I_n l'ensemble des irréductibles unitaires de degré n sur $\mathbb{F}_q$.

Proposition AG.2. $X^{q^n} - X = \prod_{d|n} \prod_{P \in I_n} P$

Preuve. Les racines (simples) de $X^{q^n} - X$ sont exactement les éléments de $\mathbb{F}_{q^n}$. En particulier il n'a pas de facteurs carrés.

1. Soit P un polynôme irréductible de degré d divisant P. Alors P a une[1] racine x dans $\mathbb{F}_{q^n}$ et $\mathbb{F}_q(x)$ est une extension de rupture de P, donc $[\mathbb{F}_q : \mathbb{F}_q(x)] = d$. Or $n = [\mathbb{F}_q : \mathbb{F}_{q^n}] = [\mathbb{F}_q : \mathbb{F}_q(x)][\mathbb{F}_q(x) : \mathbb{F}_{q^n}]$ donc $d|n$.
2. Réciproquement, si $d|n$ et $P \in I_d$, soit x une racine de P dans une extension qui décompose P et $X^{q^n} - X$. Alors $\mathbb{F}_q(x)$ est une extension de degré d c'est $\mathbb{F}_{q^d} \subseteq \mathbb{F}_{q^n}$. Donc toute racine de P est aussi racine de $X^{q^n} - X$ donc $P | X^{q^n} - X$ (car P est à racines simples par AG.1).

□

En passant au degré il vient $q^n = \sum_{d|n} d|I_d|$.

Lemme AG.3. *Pour $n > 1$, $\sum_{d|n} \mu(d) = 0$*

Preuve. Soit P l'ensemble des facteurs premiers distincts de n. Alors les seuls termes non-nuls de la somme sont $\mu(d)$ pour $d = \prod_{p \in D} p$ pour $D \subseteq P$. Il vient donc $\sum_{d|n} \mu(d) = \sum_{D \subseteq P} (-1)^{|D|} = \sum_{k=0}^{|P|} \binom{|P|}{k} (-1)^k = 0$ en réindexant selon le cardinal de D. □

Corollaire AG.4. *Si $f, g : \mathbb{N}^* \to \mathbb{N}$ telles que $g(n) = \sum_{d|n} f(d)$, alors $f(n) = \sum_{d|n} \mu(\frac{n}{d}) g(d)$.*

Preuve. En effet, $\sum_{d|n} \mu(\frac{n}{d}) g(d) = \sum_{d|n} \mu(\frac{n}{d}) \sum_{k|d} f(k)$
$= \sum_{k|d|n} \mu(\frac{n}{d}) f(k) = \sum_{k|n} f(k) \sum_{k|d|n} \mu(\frac{n}{d}) = \sum_{k|n} f(k) \sum_{p|\frac{n}{k}} \mu(p)$.
En vertu de AG.3, seul le terme en $k = n$ est non-nul, ce qui conclut. □

Corollaire AG.5. $|I_n| = \frac{1}{n} \sum_{d|n} \mu(\frac{n}{d}) q^d$.

1. En fait au moins 2.

Postrequis.

1. De l'existence de polynômes irréductibles de tous degrés.

   Proposition AG.6. *$\forall n \geq 1$, $|I_n| \geq 1$ et $I_n \sim \frac{q^n}{n}$.*

   Idée de preuve. On procède par récurrence pour la première assertion.

   □

   Donc il existe des polynômes irréductibles de tout degré, notamment sur $\mathbb{F}_p$. On peut alors *toujours* construire $\mathbb{F}_{p^n}$ comme corps de rupture $\mathbb{F}_p/(P)$ en choisissant P irréductible de degré n. A partir de là, on peut faire *sans effort*[1] des calculs dans $\mathbb{F}_{p^n} \simeq \mathbb{F}_p/(P)$.

   Comment trouver un tel P (puisqu'il en existe) ? On peut décomposer $X^{p^n} - X$ en facteurs premiers unitaires (à l'aide de AF Algorithme de Berlekamp.) et on obtiendra parmis eux tous les P irréductibles de degré n (par AG.2).

1. Avec des divisions euclidiennes de temps à autre.

AH Translatées d'une fonction dérivable.

Leçons possibles : 151 159 221 228
Adapté depuis : [FGN01a], ex 6.28. p 300

Elegance : ★★★★★

Avis.

Un très beau développement pour les questions de dimension finie et de dualité. Un peu plus discutable dans la leçon sur les équations différentielles linéaires, mais il faut bien recaser.

Prérequis.

1. Dualité en dimension finie. On utilise essentiellement la proposition suivante.

   Proposition AH.1. *Soit E un espace vectoriel de dimension finie et E^* son espace dual. Si $X \subseteq E^*$ alors* $\mathrm{Vect}(X) = (X^\circ)^\perp$.

Développement.

Théorème AH.2. *Le sous-espace vectoriel de $\mathcal{C}(\mathbb{R})$ engendré par les translatées $(f_a)_{a\in\mathbb{R}}$ d'une fonction dérivable f est de dimension finie si et seulement si f est solution d'une équation différentielle homogène à coefficients constants (EDLHC).*

Preuve. — Si f est solution d'une EDLHC $\sum_{k=0}^{n} a_k y^{(k)} = 0$, il est clair que toutes les translatées f_a le sont également. L'espace vectoriel qu'ils engendrent est contenu dans l'espace des solutions, de dimension finie $n+1$.

— Si les translatées engendrent un espace F de dimension finie $n \geq 0$, il existe une base de translatées $f_{a_1}, \dots, f_{a_n}$. Comme elles sont dérivables, tout élément de F l'est également (combinaison linéaire). En outre, F est stable par translation [1].
Soit $g \in F$, montrons que $g' \in F$. Pour ce faire on considère $g_a \in F$ sa translatée qui s'écrit $g_a = \sum_{k=1}^{n} \lambda_i(a) f_{a_i}$ et on montre que les λ_i sont dérivables.

Lemme AH.3. *Si $f_1, \dots, f_n$ est une famille libre de $\mathcal{C}(\mathbb{R})$, il existe $x_1, \dots, x_n \in \mathbb{R}$ tels que la matrice $(f_i(x_j))_{1\leq i,j\leq n}$ soit inversible [2].*

Preuve. Soit $G = \mathrm{Vect}(f_1, \dots, f_n)$ de dimension n.
G^* est engendré par $X := \{(e_x)_{x\in\mathbb{R}}\}$ l'ensemble des formes linéaires d'évaluation en $x \in \mathbb{R}$. En effet $X^\circ = \{f \in G \mid \forall x, e_x(f) = f(x) = 0\} = \{0\}$, d'où $\mathrm{Vect}(X) = (X^\circ)^\perp = \{0\}^\perp = G^*$.
On peut donc extraire une base $e_{x_1}, \dots, e_{x_n}$ de G^*. Alors $x_1, \dots, x_n$ convient pour le lemme. Sinon, les lignes de la matrice sont liées et il existe $\lambda_1, \dots, \lambda_n$ tels que $\forall 1 \leq j \leq n, \sum_{i=1}^{n} \lambda_i f_i(x_j) = 0$. Puisque $e_{x_1}, \dots, e_{x_n}$ est une base duale cela signifierait $\sum_{i=1}^{n} \lambda_i f_i = 0$. □

Pour les $x_1, \dots, x_n$ obtenus à partir de $f_{a_1}, \dots, f_{a_n}$, on écrit $g_a(x_j) = \sum_{k=1}^{n} \lambda_i(a) f_{a_i}(x_j)$.
Matriciellement cela donne $\begin{pmatrix} g(a+x_1) \\ \vdots \\ g(a+x_n) \end{pmatrix} = \begin{pmatrix} g_a(x_1) \\ \vdots \\ g_a(x_n) \end{pmatrix} = M \begin{pmatrix} \lambda_1(a) \\ \vdots \\ \lambda_n(a) \end{pmatrix}$ avec M une matrice (constante) inversible par AH.3.
Donc $\begin{pmatrix} \lambda_1(a) \\ \vdots \\ \lambda_n(a) \end{pmatrix} = M^{-1} \begin{pmatrix} g(a+x_1) \\ \vdots \\ g(a+x_n) \end{pmatrix}$ et donc les λ_i sont dérivables.

1. On écrit la combinaison linéaire.
2. La réciproque est vraie et immédiate.

Maintenant, $g(a+x) = \sum_{i=1}^{n} \lambda_i(a) f_{a_i}(x)$ et en dérivant par rapport à a puis en évaluant en $a = 0$, dans il vient $g' = \sum_{i=1}^{n} \lambda_i'(0) f_{a_i} \in F$.
Donc tout élément de F est $\mathcal{C}^\infty$ et ses dérivées sont dans F, en particulier c'est le cas de f. L'espace engendré par ses dérivées successives est donc de dimension finie, on peut trouver une EDLHC dont f est solution.

□

Postrequis.

1. Sous-algèbres de dimension finie.

   Proposition AH.4 ([FGN01a], p 310). *L'unique sous-algèbre non-triviale de dimension finie de $\mathcal{C}^0(\mathbb{R})$ est celle des fonctions constantes (de dimension 1).*

   Preuve. Si $(f^n)_{n\geq 0}$ est une famille liée, alors il existe un polynôme $P \neq 0$ qui s'annule sur l'image de f, ce qui n'est possible que si f est constante (puisque c'est un intervalle non-vide).

   □

2. En fait, la condition f dérivable n'est pas nécessaire !

   Théorème AH.5 ([BMP05], p 144.). *Le sous-espace vectoriel de $\mathcal{C}(\mathbb{R})$ engendré par les translatées $(f_a)_{a\in\mathbb{R}}$ d'une fonction* continue *f est de dimension finie si et seulement si f est solution d'une équation différentielle homogène à coefficients constants (EDLHC).*

   La preuve de [BMP05] est assez étrangement rédigée, mais l'idée est là. Il s'agit essentiellement d'approcher f par une fonction régulière en convolant. En remarquant que les λ_i sont inchangés par cette opération, on conclut qu'ils sont $\mathcal{C}^\infty$.
   Attention, cette version forte est un peu plus longue.

AI Réduction de Frobenius.

Leçons possibles : 150 153 159
Adapté depuis : [Gou09a], 2

Elegance : ★★★★★

Avis.

Prérequis.

1. Quelques résultats sur les polynômes d'endomorphismes.

Théorème AI.1 (lemme des noyaux, [Gou09a]). *Soit $f \in \mathcal{L}(E)$ et $P = \prod_{i=1}^{r} P_i$ avec les P_i deux-à-deux premiers entre eux, alors $Ker(P(f)) = \bigoplus_{i=1}^{r} Ker(P_i(f))$.*

Preuve. On traite le cas où $r = 2$, le reste se faisant facilement par récurrence.
L'inclusion $\mathrm{Ker}(P(f)) \supseteq \mathrm{Ker}(P_1(f)) + \mathrm{Ker}(P_2(f))$ est claire.
Puisque $P_1 \wedge P_2 = 1$, il existe (Bezout), U_1, U_2 avec $U_1P_1 + U_2P_2 = 1$, soit pour tout $x \in E$, $U_1(f) \circ P_1(f)(x) + U_2(f) \circ P_2(f)(x) = x$.
— Si $x \in \mathrm{Ker}(P_1(f)) \cap \mathrm{Ker}(P_2(f))$, alors nécessairement $x = 0$.
— De plus, si $x \in \mathrm{Ker}(P(f))$, alors $x_1 := U_2(f) \circ P_2(f)(x) \in \mathrm{Ker}(P_1(f))$, $x_2 := U_1(f) \circ P_1(f)(x) \in \mathrm{Ker}(P_2(f))$, et enfin $x_1 + x_2 = x$.

□

Lemme AI.2 ([Gou09a]). *Soit $f \in \mathcal{L}(E)$, il existe $x \in E$ tel que $\pi_{f,x} = \pi_f$.*

Preuve. Notons d'abord que $\pi_{f,x} = \pi_f$.

(a) On écrit $\pi_f = \prod_{i=1}^{r} P_i^{\alpha_i}$ la décomposition en produit d'irréductibles. En vertu du lemme des noyaux AI.1, il vient $E = \bigoplus_{i=1}^{r} \mathrm{Ker}(P_i^{\alpha_i}(f))$ (tous stables).

(b) Pour $x \in \mathrm{Ker}(P_i^{\alpha_i}(f))$ on a $\pi_x = P_i^{m_x}$ pour $m_x \leq \alpha_i$ (par irréductibilité).
Si on suppose que $\forall x, m_x \leq \alpha_i - 1$, alors $\mathrm{Ker}(P_i^{\alpha_i}(f)) = \mathrm{Ker}(P_i^{\alpha_i - 1}(f))$.
En remplaçant dans la décomposition précédente, $E = \bigoplus_{j \neq i} \mathrm{Ker}(P_j^{\alpha_j}(f)) \oplus \mathrm{Ker}(P_i^{\alpha_i - 1})(f) = \mathrm{Ker}((\prod_{j \neq i} P_j^{\alpha_j}) P_i^{\alpha_i - 1})(f)$ par une nouvelle application de AI.1. Ce qui contredit la minimalité de π_f.
Donc il existe $x_i \in \mathrm{Ker}(P_i^{\alpha_i}(f))$ tel que $\pi_{f,x} = P_i^{\alpha_i}$.

(c) On montre que si $\langle x \rangle_f \cap \langle y \rangle_f = \{0\}$, alors $\pi_{f,x+y} = \pi_{f,x} \vee \pi_{f,y}$. En outre $\langle x + y \rangle_f \subseteq \langle x \rangle_f \oplus \langle y \rangle_f$ puisque ce dernier contient $x + y$ et est stable par f.
En additionnant de proche en proche les x_i obtenus ci-haut, on peut maintenir l'invariant de somme directe et donc $\pi_{f, \sum_{i=1}^{r} x_i} = \prod_{i=1}^{r} P_i^{\alpha_i} = \pi_f$.

□

Développement.

Théorème AI.3 (réduction de Frobenius). *Soit $f \in \mathcal{L}(E)$, il existe une décomposition $E = \bigoplus_{i=1}^{r} F_i$ telle que : les F_i sont stables par f et mêmes cycliques ; et si P_i est le polynôme minimal de l'endomorphisme induit sur F_i, alors $P_{i+1} | P_i$.*
De plus la suite de polynômes $P_1, \ldots, P_r$ ne dépend pas de la décomposition choisie.

Preuve. Notons d'abord que nécessairement $P_1 = \pi_f$ [1].

Existence. On va procéder par récurrence sur la dimension n de l'espace.
En vertu de AI.2 il existe $x \in E$ tel que $\pi_{f,x} = \pi_f$. Notons $F := \langle x \rangle_f$ et $k := \deg(\pi_f)$, alors $e_1 := x, \ldots, e_k := f^{k-1}(x)$ est une base de F par cyclicité. On la complète en une base $e_1, \ldots, e_n$ de l'espace E.

1. $P_1(f) = 0$ (stabilité) donc $\pi_f | P_1$. De plus il existe (cyclicité) $x \in F_1$ tel que $\pi_{f,x} = P_1$ et donc $P_1 | \pi_f$.

Soit $\Gamma := \{e_k^* \circ P(f) \mid P \in \mathbb{K}[X]\}$ sous-espace vectoriel de F^* et $G := \Gamma^\circ$. Alors G est stable par f. De plus $G \oplus F = E$, en effet :

— $G \cap F = \{0\}$. En effet, si $y = \sum_{p=0}^{m} a_p f^p(x) \in G \cap F$ avec $m \leq k-1$, alors $e_k^* \circ f^{k-1-m}(y) = a_m = 0$. Donc $y = 0$.

— $\dim(G) = n - \dim(\Gamma) = n - \dim(F)$ puisque $\dim(\Gamma) = k = \dim(F)$. En effet l'application $\varphi : \mathbb{K}[X]/(\pi_f) \to \Gamma, P \mod \pi_f \mapsto e_k^* \circ P(f)$ est bien définie et surjective par construction. En outre elle est injective, puisque si $P = \sum_{p=0}^{m} a_p f^p \in \mathrm{Ker}(\varphi)$ avec $m \leq k-1$, alors $e_k^*(P(f)(f^{k-1-m}(x))) = a_p = 0$, d'où $P = 0$.

On pose donc $F_1 = F$ et on applique le procédé par récurrence sur G. Le polynôme minimal de l'endomorphisme induit divise bien π_f, puisque G est stable.

Unicité. Supposons disposer d'autres espaces $G_1, \ldots, G_s$ de polynômes $Q_1, \ldots, Q_s$ différents mais vérifiant les conditions. Soit $j \leq r, s$ le premier indice tel que $Q_j \neq P_j$ [1].
Alors $P_j(f)(E) = P_j(f)(F_1) \oplus \cdots \oplus P_j(f)(F_{j-1})$ [2].
Et $P_j(f)(E) = P_j(f)(G_1) \oplus \cdots \oplus P_j(f)(G_{j-1}) \oplus P_j(f)(G_j) \oplus \cdots \oplus P_j(f)(G_s)$.
Or pour $i \leq j-1$, F_i et G_i ont même dimension, et les endomorphismes induits $f|_{F_i}$ et $f|_{G_i}$ sur ces deux espaces sont semblables [3]. Donc $\dim(P_f(F_i)) = \dim(P_f(G_i))$. En concluant avec les dimensions, $P_j(f)(G_j) = \{0\}$, donc $Q_j | P_j$. Par symétrie $P_j | Q_j$, donc $P_j = Q_j$ ce qui est absurde.

□

Postrequis.

1. Classes de similitude.

   Définition AI.4. *On appelle* invariants de similitude *de f les polynômes $P_1, \ldots, P_r$ précédents.*

   Corollaire AI.5. *f et g sont semblables ssi elles ont mêmes invariants de similitude.*

   Preuve. La preuve précédente est clairement "invariante par similitude" (on pourrait raisonner à changement de base près). Réciproquement, si f et g ont mêmes invariants, elles sont représentées par la matrice (concaténation des compagnons des P_i) dans des bases bien choisies.

   □

2. De la calculabilité des invariants de Frobenius.

   Théorème AI.6. *Modulo calculabilité des opérations usuelles sur le corps de base [4], les invariants de similitude d'une matrice A donnée sont calculables.*

   En fait, ce sont les *invariants de Smith* (ou facteur invariants) non-inversibles ($\neq 1$) de la matrice $A - XI_n$ vivant le $\mathbb{K}[X]$-module correspondant. Ces derniers sont calculables attendu que $\mathbb{K}[X]$ est euclidien (le stathme est le degré) et que l'on peut effectivement faire des divisions euclidiennes. Voir [BMP05], 6.77 p 286 pour l'algorithme des facteurs invariants et 6.102 p 301 pour l'application à Frobenius.

   Attention, ne pas annoncer ce résultat si on ne sait pas ce qu'il y a derrière (des modules).

1. Ce n'est pas 1, puisque $P_1 = Q_1 = \pi_f$, mais néanmoins un tel indice existe toujours, pour des raisons de degrés.
2. La somme reste directe par stabilité, et les espaces après F_j sont annulés par divisibilité.
3. Ils sont cycliques de même polynôme minimal, donc dans certaines bases leurs matrices sont les mêmes.
4. C'est le cas dans les corps finis, dans $\mathbb{C}$ et tous ses sous-corps...

AJ Suite de polygones.

Leçons possibles : 152 181 182
Pas de référence.

Elegance : ★★☆☆☆

Avis.

Un développement qui illustre bien l'utilisation des nombres complexes pour transformer des problèmes "géométriques" (un dessin est d'ailleurs très convaincant) en problèmes de calcul pur. En plus les calculs sont plutôt faciles ici. Du reste, ce n'est pas un développement très valorisant... L'interprétation "géométrique" est la suivante : à partir d'un polygone quelconque du plan, on trace le polygone dont les sommets sont les milieux des côtés du précédents. On répète l'opération. La suite ainsi obtenue converge vers l'isobarycentre des points de départ.

Prérequis.

1. Des barycentres.
2. Déterminant de Vandermonde.

Développement.

Lemme AJ.1 (déterminant circulant, [Gou09a]). *Soit* $a_1, \dots, a_n \in \mathbb{C}$, $P = \sum_{i=1}^n a_i X^{i-1}$ *et* $\omega = e^{\frac{2i\pi}{n}}$.
Si $\mathcal{C}_n(a_1, a_2, \dots, a_n) := \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ a_n & a_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_2 \\ a_2 & \dots & a_n & a_1 \end{pmatrix}$, *alors* $\det(\mathcal{C}_n(a_1, a_2, \dots, a_n)) = \prod_{i=1}^n P(\omega^{i-1})$.

Preuve. Notons $\Omega := (\omega^{(i-1)(j-1)})_{1\le i,j\le n}$ et $C := \mathcal{C}_n(a_1, a_2, \dots, a_n)$.
Alors $\det(\Omega) \neq 0$ car c'est un Vandermonde.
De plus $C\Omega = \left(P\left(\omega^{(j-1)}\right)\omega^{(i-1)}\omega^{(j-1)}\right)_{1\le i,j\le n}$ [1].
En factorisant chaque colonne par $P\left(\omega^{(j-1)}\right)$, il vient $\det(C\Omega) = \prod_{i=1}^n P(\omega^{i-1})\det(\Omega)$ d'où $\det(C) = \prod_{i=1}^n P(\omega^{i-1})$. □

Théorème AJ.2. *Soit* $Z_0 = (z_1^0, \dots, z_n^0)$ *un* n*-uplet de points du plan complexe et pour* $k \ge 0$, $Z_{k+1} = (z_1^{k+1}, \dots, z_n^{k+1}) := (\frac{z_1^k + z_2^k}{2}, \dots, \frac{z_n^k + z_1^k}{2})$. *Cette suite converge vers l'isobarycentre de* Z_0.

Preuve. On écrit $Z_{k+1} = AZ_k$ où $A = z_i^k$ est une matrice circulante.

1. Alors $Z_k = A^k Z_0$ et pour étudier les puissances, on étudie la diagonalisabilité de A. Notons $\chi_A := \det(\lambda I_n - A) = \det\left(\mathcal{C}_n\left(\lambda - \frac{1}{2}, -\frac{1}{2}, 0, \dots, 0\right)\right)$ et par AJ.1 c'est $\prod_{i=1}^n P(\omega^i)$ avec $P = \lambda - \frac{1}{2} - \frac{1}{2}X$. Donc $\det(\lambda I_n - A) = \prod_{i=1}^n \left(\lambda - \frac{1+\omega^i}{2}\right)$ et A est diagonalisable [2].
2. Donc $A = PDP^{-1}$ avec $D = \mathrm{diag}\left(1, \dots, \frac{1+\omega^{n-1}}{2}\right)$
et $A^k = PD^kP^{-1} \to P\mathrm{diag}(1, 0, \dots, 0)P^{-1} =: A^\infty$ [3] et donc $Z_k \to Z_\infty := A^\infty Z_0$.

1. C'est un calcul simple, mais qui fait bien comprendre le côté cyclique de la chose :
$(C\Omega)_{i,j} = \sum_{k=1}^n (C)_{i,k}(\Omega)_{k,j} = \sum_{k=1}^n (C)_{i,k}\left(\omega^{j-1}\right)^{k-1} = \sum_{k=1}^n a_{\sigma(k)}\left(\omega^{j-1}\right)^{k-1}$
où σ est le n-cycle $k \mapsto k + (-i+1)$ (aux modulos près).
$= \sum_{k=1}^n a_k \left(\omega^{j-1}\right)^{\sigma^{-1}(k)-1}$ en réindexant la somme. Or $\sigma^{-1} : k \mapsto k + i - 1$ et on peut faire disparaître les modulos car ω^{j-1} est une racine n-ième de l'unité. On retrouve donc bien $P(\omega^{j-1})\omega^{(j-1)(i-1)}$.
2. Car elle a n valeurs propres distinctes.
3. Puisque toutes les valeurs diagonales, sauf le 1, sont de module < 1.

3. En passant à la limite dans $Z_{k+1} = AZ_k$ il vient $Z_\infty = AZ_\infty$ or le sous-espace propre associé à 1 est de dimension 1 et c'est Vect$(1,\dots,1)$. Donc $Z_\infty = (g,\dots,g)$ or l'isobarycentre de Z_k est préservé par la formule de récurrence et passe à la limite [1] et g est l'isobarycentre de Z_0.

□

Postrequis.

1. Bien sûr, on interprète ce résultat comme le joli dessin.
2. Cela fonctionne encore si on remplace la relation $z_i^{k+1} = \frac{z_i^k + z_{i+1}^k}{2}$ par n'importe quelle combinaison strictement convexe de z_i^k et z_{i+1}^k.
3. Si renormalise les points à chaque étape, cela converge vers un polygone régulier.

1. On remarque que si $g = \frac{1}{n}\sum_{i=1}^n z_i^k$, alors $g = \frac{1}{n}\sum_{i=1}^n z_i^{k+1}$ aussi, et on peut passer à la limite dans cette formule continue.

AK Théorème de Householder & méthodes iteratives.

Leçons possibles : 157 162 208 226 233
Adapté depuis : [Cia90], p 18, p 96

Elegance : ★★★☆☆

Avis.

J'ai un peu rechigné à me mettre aux méthodes numériques, sous (le faux ?) prétexte que c'était inélégant. Mais jury aime ça, alors pourquoi s'en priver ? En plus, c'est assez facile et cela se recase un peu partout [1].

Prérequis.

Développement.

Théorème AK.1 (Householder). *Le rayon spectral $\rho(A)$ d'une matrice $A \in \mathrm{M}_n(\mathbb{C})$ est l'infimum de ses normes subordonnées (et même matricielles).*

Preuve. 1. Soit $|||\cdot|||$ une norme matricielle (c'est le cas des normes subordonnées), montrons que $|||A||| \geq \rho(A)$. Soit x vecteur propre associé à la valeur propre λ de module $\rho(A)$. Alors $|||A||| \times |||x^t x||| \geq |||Ax^t x||| = |||\lambda x^t x||| = \rho(A)|||x^t x|||$. En simplifiant comme x n'est pas nul [2], on a $|||A||| \geq \rho(A)$.

2. Soit $\varepsilon > 0$. Il existe une matrice T triangulaire supérieure telle que $A = PTP^{-1}$.
En notant pour $\delta > 0$, $D_\delta := \mathrm{diag}(1, \delta, \delta^2, \dots, \delta^{n-1})$, il vient :

$$D_\delta^{-1} T D_\delta = \begin{pmatrix} t_{1,1} & & \delta(*) \\ & \ddots & \\ (0) & & t_{n,n} \end{pmatrix}$$

En effet $(D_\delta^{-1} T D_\delta)_{i,j} = \frac{\delta^{j-1}}{\delta^{i-1}} t_{i,j}$ [3].
On pose alors $\|x\| = \|D_\delta^{-1} P^{-1} x\|_\infty$ et soit $|||\cdot|||$ la norme subordonnée associée [4].
Alors $|||A||| = \sup_{x\neq 0} \frac{||Ax||}{||x||} = \sup_{x\neq 0} \frac{||D_\delta^{-1}P^{-1}Ax||_\infty}{||D_\delta^{-1}P^{-1}x||_\infty} = \sup_{x\neq 0} \frac{||D_\delta^{-1}P^{-1}APD_\delta||_\infty}{||x||_\infty}$
$= |||D_\delta^{-1} T D_\delta|||_\infty \leq \rho(A) + \varepsilon$ pour δ assez petit [5].

□

Méthode itérative. Pour résoudre $Ax = b$, on écrit $A = M - N$ avec M inversible.
Alors $Ax = b$ ssi $(M - N)x = b$ ssi $x = M^{-1}(b + Nx)$.
Donc on pose la suite x_0 arbitraire et $x_{k+1} := M^{-1}(b + Nx_k)$.

Proposition AK.2. *La méthode itérative converge pour tout x_0 ssi $\rho(M^{-1}N) < 1$.*

Preuve. Soit u la solution de $Ax = b$ et $e_k = x_k - u$ l'erreur comise.
Alors $e_{k+1} = x_{k+1} - u = M^{-1}(Nx_k + b) - M^{-1}(Nu + b) = M^{-1}Ne_k$.

1. Si $\rho(M^{-1}N) < 1$. Alors il existe une norme subordonnée telle que $|||M^{-1}N||| < 1$, donc $(M^{-1}N)^k \to 0$ donc $e_k \to 0$.
2. Si $\rho(M^{-1}N) \geq 1$ soit λ une valeur propre avec $|\lambda| \geq 1$ et e un vecteur propre associé. On pose $e := x_0 - u$. Alors $e_k = \lambda^k e$.

□

1. Je suis d'ailleurs assez convaincu - tout comme des gens de diverses options - que l'option Calcul Scientifique est la "meilleure" pour réussir l'agrégation.
2. Alors ${}^t xx$ est non-nulle.
3. On commence par calculer $(TD_\delta)_{i,j} = \delta^{j-1} t_{i,j}$.
4. Elle dépend de δ, donc (à terme) de ε ET de A.
5. Rappelons que la norme infinie est le sup des sommes des coefficients sur une ligne.

Postrequis.

1. L'erreur décroit exponentiellement (linéaire).
2. L'intérêt des méthodes itératives : inverser une matrice, cela peut être très difficile (n^3 avec le pivot de Gauss, c'est beaucoup trop pour n grand). Ici, si l'on trouve M "facile à inverser", cela simplifie grandement le problème !

   Attention : on n'inversera pas explicitement M (pas plus qu'on ne calculera $M^{-1}N$), on va se contenter de résoudre un système (dont les coefficients sont ceux de M) à chaque étape.
3. Quels sont les paradigmes pour choisir une bonne M ? [Cia90], p 97.

   Notons $A = D - I - S$ avec $D := (\chi_{i=j}a_{i,j})_{i,j}$ partie diagonale, $I := -(\chi_{i>j}a_{i,j})_{i,j}$ opposé de la partie inférieure, $S := (\chi_{i<j}a_{i,j})_{i,j}$ opposé de la partie supérieure.

   Méthode de Jacobi. $M = D$, $N = E + F$ (si M inversible). Alors $M^{-1}N = I_n - D^{-1}A$ (matrice de Jacobi). On résout la méthode en retenant l'ancien x_k (en entier) et le nouveau x_{k+1}, et en calculant une à une les coordonnées. Une itération se fait en $\mathcal{O}(n^2)$.

   Attention on résout par un calcul "à la main", mais sans inverser la matrice.

   Méthode de Gauss-Seidel. $M = D-E$, N = F. On a ici à résoudre un système triangulaire, ce qui se fait en cascade. En fait, c'est la même chose que la méthode de Jacobi, sauf que les calculs se font avec les nouvelles valeurs de x_{k+1} (ce qui fait qu'on peut le calculer "en place" sans mémoriser ailleurs tout x_k).

   Méthode de la relaxation. $M = \frac{1}{\omega}D - E$, $N = \frac{1-\omega}{\omega}D + F$ avec ω paramètre de relaxation. C'est encore un système triangulaire.

   Pour accélérer la convergence de la méthode, on cherchera ω minimisant $\rho(M^{-1}N)$.
4. Quand a-t-on convergence ?

   Théorème AK.3 (Ostrowski-Reich, [Cia90], Th. 5.3-2, p 103). *Si A est hermitienne définie positive, alors dans la méthode de la relaxation $\rho(M^{-1}N) < 1$ pour $0 < \omega < 2$ (donc elle converge et Gauss-Seidel aussi).*

   Idée de preuve.

   Lemme AK.4. *Si $M^* + N$ est définie positive, alors $\rho(M^{-1}N) < 1$.*

   Ensuite, ici $M^* + N = \frac{2-\omega}{\omega}D$ [1]. Or on montre que D est définie positive. Enfin $M^* + N$ est définie positive dès que $\omega \in]0, 2[$.

   □

   Proposition AK.5 ([Cia90], Th. 5.3-3, p 104). *$\rho(M^{-1}N) \geq |\omega - 1|$ dans la relaxation. En particulier, il n'y a convergence possible que pour $\omega \in]0, 2[$.*

   Preuve. $\det(M^{-1}N) = \frac{\det\left(\frac{1-\omega}{\omega}D+F\right)}{\det\left(\frac{1}{\omega}D-E\right)} = (1-\omega)^n$ [2].

   Or $\rho(M^{-1}N) \geq \left|\det(M^{-1}N)^{1/n}\right| = |1-\omega|$ [3]

   □

   En pratique les problèmes "de la vraie vie" (type EDP) produisent des matrices symétriques définies positives et c'est donc bien. On estime un paramètre de relaxation sympathique, puis on résout le système efficacement sans trop se poser de questions éthiques. Cependant des résultats très fins sur l'étude de la relaxation existent.
5. Estimation du rayon spectral. [Cia90], p 22.

   Théorème AK.6. *Si $|||\cdot|||$ est une norme matricielle, $\lim_{k\to+\infty} |||A^k|||^{1/k} = \rho(A)$.*

1. En utilisant $D = D^*$ et $E^* = F$.
2. Ce sont des déterminants de matrices diagonales.
3. Le max des valeurs propres est au moins aussi grand que leur moyenne.

Preuve. On a vu que $\rho(A^k) \leq |||A^k|||$. Or $\rho(A)^k = \rho(A^k)$ [1]. Donc $\rho(A) \leq |||A^k|||^{1/k}$.
D'autre part, soit $\varepsilon > 0$. Alors la matrice $A_\varepsilon := \frac{A}{\rho(A)+\varepsilon}$ est de rayon spectral < 1 donc $A_\varepsilon^k \to 0$. Donc il existe k_0 tel que $\forall k \geq n_0$, $\frac{|||A^k|||}{(\rho(A)+\varepsilon)^k} = |||A_\varepsilon^k||| \leq 1$.
Donc pour k assez grand, $|||A^k|||^{1/k} \leq \rho(A) + \varepsilon$.

□

6. Le rayon spectral n'est PAS une norme (considérer une nilpotente pour mettre en défaut le fait qu'elle soit définie, ou la somme de deux nilpotentes bien choisies pour mettre en défaut l'inégalité triangulaire).

1. On peut obtenir ce résultat en trigonalisant et en prenant les puissances.

AL Décomposition de Dunford effective.

Leçons possibles : 153 157 226 233
Adapté depuis : [Rom17], p 606

Elegance : ★★★☆☆

Avis.

Comme dans toute preuve d'algorithme qui se respecte[1], il faut ici être attentif à vérifier les invariants dans le bon ordre, sans quoi la preuve est subtilement fausse. La décomposition de Dunford est un bel outil pour obtenir des résultats théoriques sur les exponentielles de matrices, mais il semblerait malheureusement que l'effectivité soit inutile en pratique[2].

Prérequis.

1. Inversion et nilpotence.

Lemme AL.1. *Si M est inversible alors M^{-1} est un polynôme en M.*

Preuve. M est annulée par son polynôme minimal $\pi_M = \sum_{k=0}^{p} a_k X^k$, autrement dit $\sum_{k=0}^{p} a_k M^k = 0$. Or $a_0 \neq 0$, sinon en factorisant par M et en simplifiant (inversibilité), on pourrait trouver un polynôme annulateur de degré plus petit.
Il vient donc $I_n = \sum_{k=1}^{p} \frac{-a_k}{a_0} M^k = M\left(\sum_{k=0}^{p-1} \frac{-a_{k+1}}{a_0} M^k\right)$.
Enfin, $M^{-1} = \sum_{k=0}^{p-1} \frac{-a_{k+1}}{a_0} M^k$.

□

Lemme AL.2. *Si N est nilpotente et M inversible commutent alors $M+N$ est inversible.*

Preuve. Il suffit de montrer que $M+N$ est inversible. On pose en pensant aux séries géométriques $Z = M^{-1}\sum_{k=0}^{+\infty}(-NM^{-1})^k = M^{-1}\sum_{k=0}^{+\infty} N^k (M^{-1})^k$ par commutativité de N et M^{-1} et qui est une somme finie par nilpotence. On vérifie que $Z(M+N) = \sum_{k=0}^{+\infty}(-NM^{-1})^k + \sum_{k=0}^{+\infty}(-NM^{-1})^{k+1} = I$.

□

Lemme AL.3. *Si N_1 et N_2 sont nilpotentes et commutent alors N_1+N_2 est nilpotente.*

Preuve. Il suffit d'écrire $(N_1+N_2)^{2n}$ avec le binôme.

□

Développement.

Théorème AL.4. *Soit $\mathbb{K}$ un corps de caractéristique nulle et $U \in \mathrm{M}_n(\mathbb{K})$ dont le polynôme minimal π_U est scindé. Alors il existe un (unique) couple $D, N \in \mathrm{M}_n(\mathbb{K})$ tel que :*
- *$U = D + N$;*
- *D diagonalisable ;*
- *N nilpotente ;*
- *D et N commutent.*

De plus D et N sont des polynômes en U qui peuvent être calculés effectivement.

Preuve. Si $\pi_U = \prod_{i=1}^{n}(X-\lambda_i)^{\alpha_i}$ note Q le polynôme $\prod_{i=1}^{n}(X-\lambda_i) = \frac{P}{P\wedge P'}$.
Pour récupérer une matrice diagonalisable, on cherche à annuler Q et pour ce faire on applique une "méthode de Newton" en définissant la suite $(U_k)_{k\geq 0}$ avec $U_0 := U$ et $U_{k+1} := U_k - Q(U_k)Q'(U_k)^{-1}$.

Lemme AL.5. *Pour tout $k \geq 0$ on a*

1. On est vraiment entrain de réaliser une preuve de correction d'algorithme, en utilisant des invariants.
2. Après avoir demandé çà et là, je n'ai pas trouvé d'algorithme qui l'exploite pour faire un truc intelligent.

1. *U_k polynôme en U ;*
2. *$Q'(U_k)$ est inversible ;*
3. *$Q(U_k) \in \left(Q(U)^{2^k}\right)$ (idéal de $\mathbb{K}[U]$).*

Preuve. On procède par récurrence sur $k \geq 0$.
— Initialisation avec $k = 0$.

1. $U_0 = U$ est un polynôme en U.
2. Puisque $Q \wedge Q' = 1$, l'identité de Bezout donne $A, B \in \mathbb{K}[X]$ tels que $I_n = AQ(u) + BQ'(u)$ donc $B(U)Q'(U) = I_n - AQ(U)$. Or $Q(U)$ est nilpotente (d'indice $\max \alpha_i$) donc $AQ(U)$ l'est également [1] et $I_n - AQ(U)$ est inversible par commutation AL.3. Donc $Q'(U)$ est inversible également.
3. $Q(U_0) = Q(U) \in \left(Q(U)^1\right) = \left(Q(U)^{2^0}\right)$.

— Hérédité, vers $k+1$.

1. Puisque $Q'(U_k)$ est inversible, alors $Q'(U_k)^{-1}$ est un polynôme en $Q'(U_k)$. Donc par 1. c'est un polynôme en U, et U_{k+1} aussi.
2. $Q'(U_{k+1}) - Q'(U_k) = (U_{k+1} - U_k)S(U_{k+1}, U_k)$ où $S \in \mathbb{K}[X, Y]$ en factorisant les termes non-constants par $U_{k+1} - U_k$ [2]. Il vient donc $Q'(U_{k+1}) - Q'(U_k) = Q(U_k)Q'(U_k)^{-1}S(U_{k+1}, U_k) \in \left(Q(U)^{2^k}\right)$ [3] idéal de matrices nilpotentes commutant avec $Q'(U_k)$ inversible. Donc $Q'(U_{k+1})$ est inversible par AL.3.
3. $Q(U_{k+1}) = Q(U_k) + (U_{k+1} - U_k)Q'(U_k) + (U_{k+1} - U_k)^2 T(U_{k+1}, U_k)$ pour un certain $T \in \mathbb{K}[X, Y]$ (formule de Taylor).
   Donc $Q(U_{k+1}) = (U_{k+1} - U_k)^2 T(U_{k+1}, U_k)$
   $= Q(U_k)^2 Q'(U_k)^{-2} T(U_{k+1}, U_k) \in \left(Q(U)^{2^k \times 2}\right)$ par 3.

□

Enfin, dès que $k = \lceil \log_2(n) \rceil$, $Q(U)^{2^k} = 0$ par nilpotence et donc $Q(U_k) = 0$ [4] et donc $D := U_k$ est diagonalisable [5]. De plus si $N := U - U_k = \sum_{i=0}^{k-1} U_i - U_{i+1}$ alors cette matrice est nilpotente [6]. Donc $U = D + N$, diagonalisable + nilpotente, et ce sont bien des polynômes en U (qui commutent).

□

Postrequis.

1. La preuve standard de la décomposition de Dunford.
   Pour l'unicité, si $U = D' + N'$ pour deux autres matrices, alors $D - D' = N - N'$. Puisque D' commute avec U, alors aussi avec D qui est un polynôme donc $D - D'$ est diagonalisable. De manière similaire $N - N'$ est nilpotente et cette matrice est donc nulle.
2. Aller vers la décomposition de Jordan.
3. En déduire la décomposition de Dunford multiplicative.

1. Puisque tout commute donc $(AQ(U))^r = (A(U))^r \circ (Q(U))^r$.
2. Plus exactement, si $Q'(X) = \sum_{k=0}^{n} a_k X_k$ alors $Q'(X) - Q'(Y) = \sum_{k=1}^{n} a_k (X^k - Y^k)$ et pour $k > 0$ on a $X^k - Y^k = (X - Y)(\sum_{i=0}^{k} X^i Y^{k-i})$. On factorise ensuite la somme par $X - Y$.
3. Puisque $Q(U_k) \in \left(Q(U)^{2^k}\right)$ par 3. et les autres termes sont des polynômes en U.
4. Car il est dans l'idéal $\left(Q(U)^{2^k}\right) = (0)$.
5. Car annulé par Q scindé à racines simples.
6. Chaque terme $U_i - U_{i+1}$ est dans un idéal de nilpotentes, et ls commutent tous en tant que polynômes en U.

AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.

Leçons possibles : 106 150 170 181 203
Pas de référence.

Elegance : ★★★★★

Prérequis.

1. Carathéodory et cie.

   Théorème AM.1 (Carathéodory). *Soit E un espace vectoriel de dimension finie n et $X \subseteq E$. Alors $\mathrm{Conv}(X)$ est l'ensemble des barycentres à coefficients positifs de $n+1$ points de X.*

   Corollaire AM.2. *Soit E un espace vectoriel de dimension finie n et $X \subseteq E$ une partie compacte. Alors $\mathrm{Conv}(X)$ est compacte.*

2. Matrices symétriques définies positives, produits scalaires.

   Lemme AM.3. *L'ensemble $\mathcal{S}_n^{++} = \{P^tP \mid P \in \mathrm{GL}_n(\mathbb{R})\}$ des matrices symétriques définies positives est définie de manière équivalente comme $\{S \in \mathcal{S}_n \mid \forall x \in \mathbb{R}^n \setminus \{0\}, {}^txSx > 0\}$.*

   Preuve. Il s'agit de l'orbite de l'identité pour l'action par congruence. Les formes bilinéaires ainsi définies sont clairement définies positives. Et réciproquement, en examinant les signatures possibles, on remarque que les seules formes définies positives sont nécessairement de signature $(n, 0)$ [1].

   □

   Corollaire AM.4. *$\mathcal{S}_n^{++}$ est convexe.*

Développement.

Lemme AM.5 (point fixe de Kakutani [2]). *Soit E un espace euclidien, H un sous-groupe compact de $\mathrm{GL}(E)$ et K un compact convexe de E stable par (chaque élément de) H.*
Alors il existe un point fixe $a \in K$ commun à tous les $h \in H$.

Preuve. 1. Posons $N(x) = \sup_{h\in H} ||h(x)||_2$. C'est une norme sur E.

2. Si $N(x) + N(y) = N(x+y)$. Alors $N(x+y) = ||h(x+y)||_2$ pour un certain $h \in H$ (atteint par compacité). De plus $N(x) + N(y) \leq ||h(x)||_2 + ||h(y)||_2$ donc $||h(x+y)||_2 \geq ||h(x)||_2 + ||h(y)||_2$ et on a égalité dans l'inégalité triangulaire pour $||\cdot||_2$. Cette norme étant euclidienne, on en déduit [3] que $h(x)$ et $h(y)$ sont positivement liés, et donc x et y également.
3. Soit maintenant $a \in K$ qui minimise N (il existe par compacité et continuité). S'il existe un autre b tel que $N(b)$ soit minimale, alors par convexité de K, $\frac{N(a+b)}{2} = N(\frac{a+b}{2}) \geq N(a) = \frac{N(a)+N(b)}{2}$. Donc on a égalité dans l'inégalité triangulaire et a et b sont positivement liés, donc égaux car de même norme. Donc a est unique.
4. Enfin pour $g \in H$, $N(g(a)) = \sup_{h\in H} N(h \circ g(a)) = \sup_{h\in H} N(h(a)) = N(a)$. Comme K est stable par $g(a) \in K$ donc $a = g(a)$. C'est un point fixe.

□

Corollaire AM.6. *Si G sous-groupe compact de $\mathrm{GL}_n(\mathbb{R})$, alors G est conjugué à un sous-groupe de $\mathrm{O}_n(\mathbb{R})$.*

Preuve. 1. Posons $\Phi : G \to \mathrm{GL}(\mathrm{S}_n), g \mapsto (s \mapsto {}^tgsg)$ qui est un morphisme de groupes continu, donc $\Phi(G)$ est un sous-groupe compact de $\mathrm{GL}(\mathrm{S}_n)$ (euclidien).

1. Qui n'aime pas la classification des formes bilinéaires/quadratiques peut aussi utiliser la diagonalisation des endomorphismes normaux en base orthonormée, ce qui est quelque part un résultat plus puissant.
2. Attention, il y a pléthore de théorèmes du point fixe de Kakutani, alors on oubliera d'insister sur le nom.
3. C'est le cas d'égalité dans Cauchy-Schwartz.

2. $P = \{{}^t gg \mid g \in G\} \subseteq \mathrm{S}_n^{++}$ est une partie compacte de S_n. Donc son enveloppe convexe K est encore compacte par Carathéodory AM.2. Comme S_n^{++} est convexe on a $K \subseteq \mathrm{S}_n^{++}$. De plus P est clairement stable par $\Phi(G)$ donc K également (les combinaisons convexes sont stables par $\Phi(G)$).
3. En appliquant Kakutani AM.5, il existe $s \in K \subseteq S_n^{++}$ telle que $\forall g \in G, {}^t gsg = s$. Or par AM.3 $s = {}^t pp$ pour un certain $p \in \mathrm{GL}_n(\mathbb{R})$.
   Donc ${}^t g {}^t ppg = {}^t pp$ soit ${}^t(pgp^{-1})(pgp^{-1}) = I_n$, et enfin $pGp^{-1} \subseteq \mathrm{O}_n(\mathbb{R})$.

□

Postrequis.

1. C'est facile d'être un sous-groupe !

   Proposition AM.7. *Soit G une sous-partie compacte multiplicative de $\mathrm{GL}_n(\mathbb{R})$. Alors G est un sous-groupe compact.*

2. $\mathrm{O}_n(\mathbb{R})$ est un sous-groupe compact maximal (pour l'inclusion) de $\mathrm{GL}_n(\mathbb{R})$. Cela n'utilise pas le résultat qu'on a montré, mais juste la décomposition polaire.
3. Dans le cas d'un sous-groupe fini, $\langle x, y\rangle := \frac{1}{|G|}\sum_{g\in G}(gx, gy)$ est un produit scalaire G-invariant. On prend la matrice symétrique définie positives associée, elle joue directement le rôle de S pour la fin de la preuve. En fait, cette technique de sommation fonctionnerait aussi si G est compact, mais il faut d'abord donner vie à la mesure de Haar et c'est délicat.

AN Formule sommatoire de Poisson. (NR)

Leçons possibles : 246 250

Elegance : ★★★☆☆

Développement non-rédigé...

AO Théorème de Sarkovski.

Leçons possibles : 223 228
Adapté depuis : [FGN01b], p 92

Elegance : ★★★★★

Avis.

Un peu long, on a envie de passer rapidement sur les études de cas pénibles (ce qui est tout à fait possible). Nonobstant le peu d'esthétique de certains passages de la preuve, c'est sans doute la plus belle et la plus surprenante application du TVI qui soit.

Prérequis.

1. Le gentil théorème de point fixe.

   Proposition AO.1. *Soit $f : I \to \mathbb{R}$ une fonction continue et I un segment.*

   (a) si $f(I) \subseteq I$, alors f possède un point fixe ;

   (b) si $f(I) \supseteq I$, alors f possède un point fixe.

   Preuve. Soit $I =: [a, b]$.

   (a) $f(a) \in [a, b]$ donc $f(a) \geq a$. De même $f(b) \leq b$. En appliquant le TVI à $\varphi : x \mapsto f(x) - x$ on conclut que f possède un point fixe.

   (b) L'argument est dual. Soit $\alpha := f^{-1}(a)$ et $\beta := f^{-1}(b)$. Alors $\alpha \geq f(\alpha) = a$ et $\beta \leq f(\beta) = b$. On conclut de même.

   □

2. Systèmes dynamiques.

   Définition AO.2 (cycles). *On dit que $f : I \to I$ au k -cycle s'il existe un point $x \in I$ tel que $f^k(x) = x$ et $\forall 1 \leq i < k$, $f^i(x) \neq x$.*

Développement.

Théorème AO.3 (Sarkovski). *Si $f : I \to I$ possède un 3-cycle, alors elle possède un k-cycle pour tout $k \geq 1$.*

Preuve. On commence par établir une "réciproque" de la préservation des segments.

Lemme AO.4. *Soit $K \subseteq f(I)$ un segment, alors il existe un segment $L \subseteq I$ tel que $f(L) = K$.*

Preuve. Notons $I = [a, b]$, on suppose $a < b$. Soient a', b' tels que $f(a') = a$ et $f(b') = b$. On suppose $a < b$ (l'autre cas est symétrique).
Soit $B = \{x \in [a', b'] \mid f(x) = b\}$ c'est un compact non-vide. Soit $\beta = \min B$, alors $\forall x \in [a', \beta[, f(x) < b$ en appliquant le TVI [1].
Soit $A = \{x \in [a', \beta] \mid f(x) = a\}$ c'est un compact non-vide. Soit $\alpha = \max A$, alors on conclut de manière similaire (avec le TVI) que $f([\alpha, \beta]) = [a, b]$.

□

On notera désormais $I \to I'$ si $f(I) \supseteq I'$.

Lemme AO.5. *Si $I_0 \to I_1 \to \cdots \to I_{n-1} \to I_n = I_0$ (tous segments), alors il existe x point fixe de f^n tel que $\forall 0 \leq j \leq n$, $f^j(x) \in I_j$.*

1. Puisque $f(a') = a$, s'il existe $b'' < \beta$ tel que $f(b'') \geq b$, alors en vertu du TVI il existe $a < \beta' < b''$ tel que $f(\beta') = b$ ce qui est absurde.

Preuve. On va construire par récurrence sur $0 \leq j \leq n$ une suite de segments $I_0 = J_0 \supseteq \cdots \supseteq J_j$ tel que $\forall i \leq j, f^i(J_i) = I_i$.
— $j = 0$. On prend $J_0 = I_0$ et c'est évident.
— $j \to j+1$. On sait que $f^j(J_j) = I_j$, donc $f^{j+1}(J_j) = f(f^j(J_j)) \supseteq I_{j+1}$. Par AO.4 appliqué à f^{j+1}, il existe donc $J_{j+1} \subseteq J_j$ tel que $f^{j+1}(J_{j+1}) = I_{j+1}$.
On a enfin, $f^n(J_n) = I_n = I_0 \supseteq J_n$. Donc f^n admet un point fixe $x \in J_n$. Comme $x \in J_j$ pour tout j, $f^j(x) \in I_j$.

□

Passons à la preuve du théorème. Soit a un point de période 3, $b := f(a)$, $c := f(b)$ (ils sont tous distincts). On suppose $a < b < c$ (les autres cas sont similaires), soit $I_0 := [a, b]$ et $I_1 := [b, c]$. Alors $I_0 \to I_1$, $I_1 \to I_0$ et $I_1 \to I_1$ [1].
Soit $n \geq 1$, on écrit $I_0 \to I_1 \to \cdots \to I_1 \to I_0$ (de taille $n+1$). Par AO.5 on a x tel que $f^n(x) = x$ et $f^j(x) \in I_j$ pour tout j. Si la période n'est pas n, alors $f^j(x) = x$ pour un $0 < j < n$. Donc $x \in I_0 \cap I_1$, soit $x = b$ ce qui n'est pas possible si $n \neq 3$ [2]. Donc x est un point de période n.

□

Postrequis.

1. L'ordre de Sarkovski.
2. Un exemple de fonction chaotique : la fonction tente.

1. $f(a) = b$ et $f(b) = c$. Donc $I_1 = [b, c] \subseteq f([a, b]) = f(I_0)$ par le TVI.
$f(b) = c$ et $f(c) = a$. Donc $I_0, I_1 \subseteq f([b, c]) = f(I_1)$ par le TVI again.
2. Ici il faut regarder les cas pour voir que b n'est pas "au bon endroit au bon moment". Pour $n = 1, 2$ c'est clair et si $n \geq 4$, alors $f^2(x) = a \in I_1$ c'est absurde.

AP Réduction lisse des formes quadratiques et lemme de Morse.

Adapté depuis : [Rou14], ex 66 p 209, ex 109 p 330
Leçons possibles : 170 218 214 215 219

Elegance : ★★★★☆

Avis.

Un développement qui témoigne d'une bonne maîtrise des théorèmes d'inversion et des concepts de différentielles (secondes). Une petite technicité vient de la formule de Taylor que l'on utilise, compacte mais un peu indigeste à prouver (et délicate à apprendre). Pour obtenir le lemme de Morse, on cherche à réduire régulièrement les formes quadratiques le long d'un chemin. Plutôt que de faire cela, on réduit *tout un voisinage* de la forme qui nous intéresse.
Le danger - caché - de ce développement est qu'il a *énormément* d'applications *assez loin* du programme de l'agreg : la théorie de Morse (voir un aperçu dans les Postrequis). Il est bon de ne pas attirer les questions à ce sujet, à moins d'être un spécialiste de géométrie différentielle.

Prérequis.

1. Classification des formes quadratiques, cas réel.
2. Formule de Taylor avec reste intégrale (pour une fonction de plusieurs variables).

Développement.

Lemme AP.1. *Si $A_0 \in \mathrm{S}_n$ est inversible, il existe un voisinage V de A_0 dans S_n et une application de classe $\mathcal{C}^1$, $V \to \mathrm{GL}_n(\mathbb{R}), A \mapsto M$ telle que $A = {}^tMA_0M$ pour tout $A \in V$* [1].

Preuve. 1. L'application $\varphi : \mathrm{M}_n(\mathbb{R}) \to \mathrm{S}_n, M \mapsto {}^tMA_0M$ est de classe $\mathcal{C}^1$ car polynomiale. L'idée est d'essayer de l'inverser localement près de I_n.
$d\varphi(I_n).H = {}^tHA_0 + A_0H = {}^t(A_0H) + A_0H$ [2] donc son noyau est exactement $A_0^{-1}\mathrm{A}_n$ et elle est surjective pour des raisons de rang/dimension.

2. Or $\mathrm{M}_n(\mathbb{R}) = A_0^{-1}\mathrm{A}_n \oplus A_0^{-1}\mathrm{S}_n$ [3].
Donc $d\varphi(I_n)|_{A_0^{-1}\mathrm{S}_n}$ est bijective [4], et c'est $d\varphi|_{A_0^{-1}\mathrm{S}_n}(I_n)$ puisque $I_n \in A_0^{-1}\mathrm{S}_n$.
En vertu du théorème d'inversion locale, il existe un voisinage W de I_n dans $A_0^{-1}\mathrm{S}_n$ tel que $\varphi|_{A_0^{-1}\mathrm{S}_n}$ soit un $\mathcal{C}^1$-difféomorphisme entre W et $\varphi(W) =: V$.
Enfin, poser $M := \varphi|_{A_0^{-1}\mathrm{S}_n}(A)$ convient pour trouver $A = {}^tMA_0M$.

□

Théorème AP.2 (lemme de Morse). *Soit $f : U \to \mathbb{R}$ une fonction de classe $\mathcal{C}^3$ sur un ouvert de $\mathbb{R}^n$ telle que $f(0) = 0$ et $df(0) = 0$ et $d^2f(0)$ non dégénérée de signature $(p, n-p)$* [5]. *Alors il existe un $\mathcal{C}^1$-difféomorphisme $\varphi : x \mapsto u$ local entre deux voisinages de 0 tel que $f(x) = u_1^2 + \cdots + u_p^2 - u_{p+1}^2 - \cdots - u_n^2$.*

Preuve. On écrit en 0 la formule de Taylor à l'ordre 1 avec reste intégrale [6].
$f(x) = f(0) + df(0).x + {}^txQ(x)x$ où $Q(x) := \displaystyle\int_0^1 (1-t)d^2f(tx)\mathrm{d}t$.
$Q(x)$ est (matriciellement) une matrice symétrique $\mathcal{C}^1$ en x [7]. On peut appliquer AP.1 à

1. Une matrice suffisamment proche d'une matrice symétrique lui est congruente, et le changement de base peut être fait de manière $\mathcal{C}^1$.
2. C'est la différentielle d'une application bilinéaire.
3. On multiplie juste par A_0^{-1} dans le très classique $\mathrm{A}_n \oplus \mathrm{S}_n$.
4. Restriction à un supplémentaire du noyau.
5. 0 est un point critique non dégénéré.
6. Pour le moment, f de classe $\mathcal{C}^2$ est suffisant.
7. Ici on a besoin du caractère $\mathcal{C}^3$.

$Q(x)$ pour x assez proche de 0, puisque $Q(0) = d^2 f(0)$ et en composant par l'application obtenue il vient : $Q(x) = {}^t M(x) d^2 f(0) M(x)$.
Donc $Q(x) = {}^t M(x)^t A I_{p,n-p} A M(x)$ pour une certaine matrice $A \in \mathrm{GL}_n(\mathbb{R})$ (signature).
Enfin l'application $\varphi : x \mapsto AM(x)x$ est un $\mathcal{C}^1$-difféomorphisme local en 0. En effet, sa différentielle en 0 est $AM(0) = A$ inversible et on applique l'inversion locale. φ donne exactement la forme voulue. □

Postrequis.

1. Restreindre les hypothèses.
   Si $f(0) \neq 0$ ou $Df(0) \neq 0$, on peut appliquer le lemme à $g : x \mapsto f(x) - f(0) - Df(0).x$. Il vient alors $f(x) = f(0) + Df(0).x + u_1^2 + \cdots + u_p^2 - u_{p+1}^2 - \cdots - u_n^2$ ("Taylor sans reste").
2. Les différences avec le théorème de rang constant.
   Théorème AP.3 (rang constant, [Rou14], p 231). *Soit $f : U \subseteq \mathbb{R}^n \to \mathbb{R}^p$ une application de classe $\mathcal{C}_1$, avec U est un voisinage de 0 et $f(0) = 0$. On suppose $Df(x)$ de rang constant r sur U.*
   Alors il existe un changement de coordonnées locales $\varphi : x \mapsto X$ au départ (au voisinage de 0 dans $\mathbb{R}^n$), et un autre $\psi : y \mapsto Y$ à l'arrivée (au voisinage de 0 dans $\mathbb{R}^p$) tels que :
   $$\psi \circ f \circ \varphi^{-1} : (X_1, \ldots, X_n) \mapsto (X_1, \ldots, X_r, 0, \ldots, 0).$$
   Ici on doit connaître le comportement de $Df(x)$ sur *tout un ouvert*. Cette hypothèse ne s'applique pas aux fonctions qu'on traite dans lemme de Morse, puisque la différentielle en 0 est nulle, d'où $r = 0$ mais c'est absurde car alors on aurait la fonction nulle.
   Attention aussi, le lemme de Morse ne traite que $p = 1$.
3. Les points critiques non-dégénérés sont isolés.
   Proposition AP.4. *Un point critique non-dégénéré d'une fonction $\mathcal{C}^3$ est isolé.*
   > *Preuve.* On reprend les hypothèses et notations de AP.2.
   > On avait $f \circ \varphi^{-1}(u_1, \ldots, u_n) = u_1^2 + \cdots + u_p^2 - u_{p+1}^2 - \cdots - u_n^2$.
   > Donc $Df(x) \circ D\varphi^{-1}(u) = 2\,\mathrm{diag}(u_1, \ldots, u_p, -u_{p+1}, \ldots, -u_n)$ qui n'est pas nul en dehors de 0. Puisque $D\varphi^{-1}(u)$ est inversible, on conclut que $Df(x)$ est non-dégénérée sauf en 0.
   □
4. Application à l'étude locale d'une fonction $\mathbb{R}^2 \to \mathbb{R}$: minimum local, maximum local, point selle (et alors on a des points doubles sur les courbes de niveau). Voir [Rou14].
5. Un brin de théorie de Morse. Voici (en très abrégé) un plan pour classifier les variétés $\mathcal{C}^\infty$ compactes connexes orientées de dimension 2 (aka surfaces) à homéomorphisme près.
   (a) Prendre une telle variété V.
   (b) Définir une fonction de Morse comme une fonction $V \to \mathbb{R}$, de classe $\mathcal{C}^\infty$ et dont tous les points critiques sont non-dégénérés.
   (c) Montrer qu'il existe effectivement f fonction de Morse sur V (c'est très long, on passe par le théorème de Sard & le plongement de Whitney).
   (d) Remarquer que (par compacité et AP.4), il n'y a qu'un nombre fini de points critiques non-dégénérés.
   (e) L'idée est de "plonger" la variété dans $\mathbb{R}^3$ comme $\{x, f(x)\}$ (et donc la voir comme une vraie surface). Plus précisément, on étudie les morceaux $\{x \mid f(x) \leq a\}$ pour a croissant. C'est une variété à bord qui reste homéomorphe, sauf quand on franchit un (ou des) points critiques. Selon la signature $(++, --, +-)$ de la Hessienne, en utilisant le point 4, on peut préciser les modifications apportées lors du passage d'un point critique (minimum, maximum ou selle).

(f) Conclure que ces surfaces sont classées par leur genre g [1] : ce sont la sphère, et les tores collés les uns aux autres par le côté (la bouée pour g enfants). Le groupe fondamental (ici $\mathbb{Z}^g$) est un invariant topologique total...

1. "Nombre de trous" pour les profanes.

AQ Théorème de Hadamard-Lévy.

Leçons possibles : 203 214 215 220
Adapté depuis : [QZ02],

Elegance : ★★★★☆

Avis.

Le théorème d'inversion locale donne une condition simple pour *inverser localement* une application (et ce de manière régulière). En ajoutant une hypothèse d'injectivité, on peut *inverser globalement*, i.e. sur toute l'image. La question est de trouver ici une condition élégante pour savoir quand cette image est $\mathbb{R}^n$ tout entier, et donc obtenir un théorème d'*inversion très globale*. La preuve, très élégante, fait appel à des champs de vecteurs, plus précisément à l'étude qualitative d'un flot bien choisi.
Deux remarques essentielles quant à la référence :
— je propose une factorisation de la preuve de [QZ02], qui est un peu trop longue ;
— dans les versions ≥ 4 du livre [QZ13], la preuve proposée est très différente.

Prérequis.

1. Flot d'un champ de vecteurs.
   Définition AQ.1. *Un* champ de vecteurs *sur $U \subseteq \mathbb{R}^n$ ouvert est une application $X : U \to \mathbb{R}^n$.*
   Remarque AQ.2. *On peut aussi définir un champ sur une (sous-)variété, il est alors à valeurs dans l'espace tangent.*
   Le système autonome associé à X est alors $y' = X(y)$. Quand il existe, on définit son *flot* $\varphi(t,x)$. Le théorème de Cauchy-Lipschitz assure l'existence locale dans le cas $\mathcal{C}_1$. Avec le lemme de Gronwall, on peut montrer qu'il y a *dépendance continue en la condition initiale*.

Développement.

Théorème AQ.3. *Soit $f \in \mathcal{C}^2(\mathbb{R}^n)$ propre et telle que $df(x)$ est inversible pour tout x. Alors f est un $\mathcal{C}^1$-difféomorphisme de $\mathbb{R}^n$ sur $\mathbb{R}^n$.*

Preuve. En vertu du TIG, il suffit de montrer que f est bijective [1]. On se place sans perte de généralités [2] en 0 pour montrer que le cardinal de $S := f^{-1}(\{0\})$ est 1.

Soit le champ de vecteurs $X : \mathbb{R}^n \to \mathbb{R}^n, z \mapsto -df(z)^{-1}(f(z))$ et $\varphi(t,x)$ le flot associé. Il existe localement par Cauchy-Lipschitz [3].

1. **Les flots sont globaux à droite.**
   Soit x fixé et $[0, T^*[$ intervalle maximal de définition de $\varphi(t,x)$.
   Soit $g_x : [0,T^*[\to \mathbb{R}^n, t \mapsto f \circ \varphi(t,x)$, elle est $\mathcal{C}^1$. Alors $g_x'(t) = df(\varphi(t,x)).\frac{\partial \varphi}{\partial t}(t,x)$ $= -f(\varphi(t,x)) = -g_x(t)$ et $g_x(0) = f(\varphi(0,x)) = f(x)$.
   Donc $g_x(t) = e^{-t} f(x)$ est à valeurs dans un compact, donc [4] $\varphi(\cdot, x)$ également.
   Donc $T^* = +\infty$ pour ne pas contredire le théorème des bouts.
2. **Soit $y \in S$, alors c'est un point critique asymptotiquement stable.**
   En vertu du TIL, f est un $\mathcal{C}^1$-difféomorphisme entre un voisinage U_y de y et $B(0,\varepsilon)$ (d'inverse notée $f^{-1}|_{B(0,\varepsilon)}$). Soient t_0, x tel que $\varphi(t_0,x) \in U_y$. Alors $\forall t \geq t_0$, $\varphi(t,x) \in U_y$. En effet $\{t \geq t_0 \mid \varphi(t,x) \in U_y\}$ est un ouvert-fermé de $[t_0, +\infty[$ car :

1. Si f est injective, ce sera un $\mathcal{C}^1$-difféomorphisme vers son image. Si f est surjective, cette image sera $\mathbb{R}^n$ tout entier.
2. Pour traiter un autre point a, on considère $x \mapsto f(x) - a$ qui vérifie encore les hypothèses du résultat.
3. $x \mapsto df(x)$ est de classe $\mathcal{C}^1$ (car f est $\mathcal{C}^2$! !), donc $x \mapsto df(x)^{-1}$ aussi (composée avec l'inverse), mais $x \mapsto f(x)$ l'est aussi, d'où $x \mapsto -df(x)^{-1}(f(x))$ encore. En particulier elle est localement Lipschitzienne.
4. Par propreté de f.

— c'est $\varphi(\cdot, x)^{-1}(U_y) \cap [t_k, +\infty[$ est ouvert ;
— c'est $\{t \geq t_k \mid \varphi(t,x) = f^{-1}|_{B(0,\varepsilon)}(g_x(t))\}$ [1] fermé.
Donc $\varphi(t,x) = f^{-1}|_{B(0,\varepsilon)}(g_x(t)) \to f^{-1}|_{B(0,\varepsilon)}(0) = y$.

3. Soit $A(y) := \{x \mid \varphi(\cdot, x) \to y\}$ le bassin d'attraction de $y \in S$.
Alors $\mathbb{R}^n = \biguplus_{y\in S} A(y)$ union d'ouverts disjoints non-vides. En effet :
— $A(y)$ contient U_y donc il est non-vide ;
— $A(y) = \bigcup_{t\geq 0} \varphi(t,\cdot)^{-1}(U_y)$ [2] donc c'est un ouvert (continuité en x) ;
— si x fixé $\varphi(\cdot, x)$ est à valeurs dans un compact. Donc il existe $t_k \to +\infty$ avec $\varphi(t_k, x) \to y$, d'où $g_x(t_k) \to f(y) = 0$ par continuité. Donc $y \in S$ et $\varphi(t,x) \to y$ par stabilité asymptotique.
Donc S est un singleton par connexité.

□

Postrequis.

1. En fait, comme f est $\mathcal{C}^2$, c'est même un $\mathcal{C}^2$-difféomorphisme.
2. Réciproquement, si f est un $\mathcal{C}^1$-difféomorphisme de $\mathbb{R}^n$ sur $\mathbb{R}^n$, alors :
— f est propre, puisque l'image réciproque $f^{-1}(K)$ est en fait l'image par f^{-1} une application continue ;
— en tout point $df(x)$ est inversible (différentier $f \circ f^{-1}(x) = x$).
3. Caractérisation simple des fonctions propres.

Proposition AQ.4. *$f : \mathbb{R}^n \to \mathbb{R}^n$ continue est propre si et seulement si $||f(x)|| \to +\infty$ pour $||x|| \to +\infty$ (f coercive).*

> *Preuve.* Par continuité l'image réciproque d'un compact (fermé) est fermée, le lot de l'affaire est de montrer que sa bornitude est équivalente à notre condition.
> Or, $||f(x)|| \to +\infty$ ssi pour tout $M \geq 0$ il existe A tel que $||x|| \geq A$ implique $||f(x)|| \geq M$. Ce qui est équivalent à dire que $f^{-1}(B(0,M))$ est bornée. □

4. Le "vrai" théorème de Hadamard-Lévy se passe de l'hypothèse $\mathcal{C}^2$ et s'énonce comme suit.

Théorème AQ.5 (Hadamard-Lévy). *$f \in \mathcal{C}^1(\mathbb{R}^n)$ est un difféomorphisme local de $\mathbb{R}^n$ sur $\mathbb{R}^n$ si et seulement si f est propre et telle que $df(x)$ est inversible pour tout x.*

On avait utilisé crucialement l'hypothèse $\mathcal{C}^2$ pour appliquer Cauchy-Lipschitz. Le théorème de Cauchy-Peano-Arzela démontre qu'au mois un flot existe localement avec la continuité seule du champ. On montre "à la main" le reste des choses utilisées ci-haut.

1. $f^{-1}|_{B(0,\varepsilon)}(g(t))$ est bien défini car g décroît et on a pris une boule. En outre, si $\varphi(t,x) = f^{-1}|_{B(0,\varepsilon)}(g_x(t))$ alors $\varphi(t,x) \in U_y$ car c'est l'image de $f^{-1}|_{B(0,\varepsilon)}$. Réciproquement, si $\varphi(t,x) \in U_y$, comme f est un difféomorphisme de U_y vers $B(0,\varepsilon)$, il est clair que $f^{-1} \circ f(\varphi(t,x)) = \varphi(t,x)$.
2. Si $\varphi(\cdot, x) \to y$, alors à partir d'un certain temps on est dans U_y. Réciproquement on a montré (attraction) qu'en entrant dans U_y on finit par tendre vers y.

AR Gradient à pas optimal. (NR)

Leçons possibles : 219 229

Elegance : ★★☆☆☆

Développement non-rédigé...

AS Théorème de Liapounov. (NR)

Leçons possibles : 220 221
Adapté depuis : [Rou14], ex 46 p 139

Elegance : ★★★★☆

Développement non-rédigé...

AT Théorème de Banach-Steinhaus & séries de Fourier divergentes.

Leçons possibles : 208 246
Adapté depuis : [Rud09], 5.11 p 130

Elegance : ★★★★☆

Prérequis.

1. Espaces de Baire et cie.

   Lemme AT.1 (Baire). *Dans un espace de Banach, une intersection dénombrable d'ouverts denses est dense (et en particulier non vide).*

   Preuve. Soit E un espace de Banach et $(U_n)_{n\geq 0}$ une suite d'ouverts denses. On prend $x \in E$, $\varepsilon > 0$ et on cherche $y \in \bigcap_n U_n$ tel que $\mathrm{d}(y,x) < \varepsilon$.
   Il existe $x_0 \in U_0$ et $r_0 > 0$ (par ouverture et densité de U_0) tels que $\mathrm{B}(x_0,r_0) \subseteq \overline{\mathrm{B}(x,\varepsilon/2)}$ (en particulier $\mathrm{d}(x,x_0) < \varepsilon/2$). On peut aisément construire par récurrence des suites $(x_n)_{n\geq 0} \in E^{\mathbb{N}}$ et $(r_n)_{n\geq 0} \in \mathbb{R}^{\mathbb{N}}$ telles que $\mathrm{B}(x_{n+1},r_{n+1}) \subseteq U_n \cap \overline{\mathrm{B}(x_n,r_n/2)}$.
   La suite $(x_n)_{n\geq 0}$ est de Cauchy (car $r_{n+1} \leq r_n/2$), donc elle converge vers une limite y. En outre, $y \in \overline{\mathrm{B}(x_n,r_n/2)} \subseteq U_n$ (par fermeture) pour tout $n \geq 0$ donc $y \in \bigcap_n U_n$. Enfin, $y \in \overline{\mathrm{B}(x,\varepsilon/2)} \subseteq \mathrm{B}(x,\varepsilon)$ et donc $\mathrm{d}(x,y) < \varepsilon$.
   □

   Remarque AT.2. *Cela reste vrai dans un ouvert d'un espace de Banach (adapter la preuve).*

2. Noyau de Dirichlet.

Développement.

Théorème AT.3 (Banach-Steinhaus). *Soit E un espace de Banach, F un espace vectoriel normé et H une famille d'opérateurs (linéaires continus) de E vers F. Alors on a l'alternative suivante :*
- $\sup_{h\in H} |||h||| < +\infty$ *et* $\forall x \in E$, $\sup_{h\in H} ||h(x)||_E < +\infty$.
- $\sup_{h\in H} |||h||| = +\infty$ *et il existe une intersection dénombrable d'ouverts denses X telle que* $\forall x \in X$, $\sup_{h\in H} ||h(x)||_E = +\infty$.

Preuve. Posons $U_n = \{x \in E \mid \exists h \in H, ||h(x)||_F > n\}$ pour $n \geq 0$. Ces ensembles sont ouverts comme union d'ouverts (comme les éléments de H sont continus).
- soit chaque U_n est dense dans E, auquel cas leur intersection X l'est aussi en vertu du lemme de Baire. De plus, pour tout $x \in X$, $\sup_{h\in H} ||h(x)||_F = +\infty$ et donc comme il existe un tel x on a $\sup_{h\in H} |||h||| = +\infty$.
- soit il existe un U_n non-dense dans E. Prenons $x \in E$ et $\varepsilon > 0$ tels que $\forall y \in \mathrm{B}(x,\varepsilon)$, $h \in H$, $\sup_{h\in H} ||h(y)||_F \leq n$. Si $z \in \mathrm{B}(0,\varepsilon)$ et $h \in H$ on a $h(z) = h(z+x) + h(-x)$ donc $\sup_{h\in H} ||h(z)||_F \leq n + ||h(x)||_F$. On conclut que $\sup_{h\in H} |||h||| < +\infty$ et donc également $\forall x \in E$, $\sup_{h\in H} ||h(x)|| < +\infty$.

□

Corollaire AT.4. *Il existe $f \in \mathcal{C}^0(\mathbb{T})$ telle que* $\sup_N S_N(f)(0) = +\infty$.

Preuve. Soit E l'espace $\mathcal{C}^0(\mathbb{T})$ muni de la norme de la convergence uniforme, c'est un Banach. On étudie la famille $l_n : E \to \mathcal{C}, f \mapsto S_n(f)(0)$.

1. Ce sont des opérateurs linéaires continus. La linéarité est claire. De plus, pour tout $n \geq 0$, $l_n(f) = \frac{1}{2\pi}\int_0^{2\pi} f(t)D_n(0-t)\mathrm{d}t = \frac{1}{2\pi}\int_0^{2\pi} f(t)D_n(t)\mathrm{d}t$ par parité.
   Donc $|l_n(f)| \leq ||f||_\infty ||D_n||_1$ ce qui justifie la continuité.

2. On a en fait $|||l_n||| = ||D_n||_1$. En effet, soit $g : \mathbb{T} \to \mathbb{R}$ telle que $g(x) = 1$ si $D_n(x) \geq 0$ et -1 sinon. Il existe une suite $(f_k)_{k\geq 0}$de fonctions de $\mathcal{C}^0(\mathbb{T})$, telles que $||f_k||_\infty \leq 1$ et convergeant simplement vers g [1] pour $k \geq 1$. En vertu du théorème de convergence dominée, $\lim_k l_n(f_k) = \frac{1}{2\pi}\int_0^{2\pi} f(t)|D_n|(t)\mathrm{d}t = \frac{1}{2\pi}\int_0^{2\pi} g(t)D_n(t)\mathrm{d}t = ||D_n||_1$.
3. $|D_n|_1 \to +\infty$. En effet $D_n(t) = \frac{\sin((n+\frac{1}{2})t)}{\sin(\frac{t}{2})}$ [2] et donc $||D_n||_1 \geq \frac{1}{\pi}\int_0^\pi \frac{|\sin((n+\frac{1}{2})t)|}{|\sin(\frac{t}{2})|}$. Or comme $|t| \geq |\sin(t)|$ il vient $||D_n||_1 \geq \frac{2}{\pi}\int_0^\pi \frac{|\sin((n+\frac{1}{2})t)|}{|t|}\mathrm{d}t = \int_0^{(n+\frac{1}{2})\pi} \frac{|\sin(t)|}{|t|}\mathrm{d}t$ et ce dernier terme tend vers l'infini (logarithmiquement d'ailleurs) [3].

En appliquant le Théorème AT.3 de Banach-Steinhaus, on à l'existence de fonctions continues dont la série de Fourier est non bornée en 0.

□

Postrequis.

1. On peut exhiber une telle fonction (contre exemple de Paul du Bois-Reymond) mais ce n'est pas trivial.
2. En fait, Banach-Steinhaus prouve même qu'il existe un G_δ-dense de telles fonctions. Quitte à translater, on sait que pour tout $t \in \mathbb{T}$ il existe un G_δ-dense de fonctions continues dont la série de Fourier est non bornée en t. En prenant l'intersection de ces ensembles sur tous les rationnels (c'est toujours une intersection dénombrable), il vient que l'ensemble des fonctions continues dont la série de Fourier n'est pas bornée sur tout rationnel est dense dans les fonctions continues du tore. Et c'est fâcheux !

1. Il s'agit d'approcher le créneau au niveau des discontinuités par une pente de plus en plus grande. Prendre par exemple $f_k = \frac{D_n}{D_n + 1/k}$.
2. C'est un calcul de Sup... On remarque que $D_n(t) = \sum_{k=-n}^n \exp(\mathrm{i}\,kt)$ est une somme géométrique d'où $D_n(t) = \exp(-\mathrm{i}\,nt)\frac{1-\exp(\mathrm{i}(2n+1)t)}{\exp(\mathrm{i}\,t)} = 1 \times \frac{\sin((n+\frac{1}{2})t)}{\sin(\frac{t}{2})}$ en prenant l'angle moitié.
3. C'est classique : il suffit de remarquer que $\int_0^{n\pi} \frac{|\sin(t)|}{t}\mathrm{d}t = \sum_{k=1}^n \int_{(k-1)\pi}^{k\pi} \frac{|\sin(t)|}{t}\mathrm{d}t \geq \sum_{k=1}^n \frac{1}{k}\int_{(k-1)\pi}^{k\pi} |\sin(t)|\mathrm{d}t \geq \sum_{k=1}^n \frac{1}{k} \times \frac{2}{\pi} \sim \frac{2}{\pi}\ln(n)$. Et on peut faire pareil avec une majoration.

AU Formule d'inversion de Fourier dans $\mathcal{S}(\mathbb{R})$. (NR)

Leçons possibles : 239 250

Elegance : ★★☆☆☆

Développement non-rédigé...

AV Méthode de Laplace et formule de Stirling.

Leçons possibles : 218 224 239
Adapté depuis : [Rou14], ex 113 p 349

Elegance : ★★☆☆☆

Avis.

Voir ZQ ou PGCD ?

Prérequis.

1. Intégrale de Gauss

   Proposition AV.1. $\int_{-\infty}^{+\infty} e^{-u^2}\,du = \sqrt{\pi}$.

Développement.

Théorème AV.2. *On considère une intégrale de la forme* $F(t) = \int_a^b f(x)e^{-t\varphi(x)}dx$ *sous les hypothèses :*

— $[a,b[\subseteq \mathbb{R}$;
— φ *de classe* $\mathcal{C}^2$, $\varphi' > 0$ *sur* $]a,b[$, $\varphi'(a) = 0$, $\varphi''(a) > 0$.
— f *continue en* a
— $f(x)e^{-t\varphi(x)}$ *intégrable pour* $t \geq t_0$.

Alors $F(t) \sim e^{-t\varphi(a)} f(a)\sqrt{\frac{2\pi}{\varphi''(a)t}}$.

Preuve. On supposera (à changement $t - t_0$ près) que $t_0 = 0$.

1. **Cas particulier.** On se place d'abord dans le cas où $a = 0$ et $\varphi(x) = x^2$.
   Alors soit $\alpha > 0$, $\sqrt{t}\int_0^\alpha \mathrm{e}^{-tx^2} f(x)\mathrm{d}x = \int_0^{\alpha\sqrt{t}} \mathrm{e}^{-u^2} f(\frac{u}{\sqrt{t}})\mathrm{d}u$ en posant $u = \sqrt{t}x$
   $= \int_0^{+\infty} \mathrm{e}^{-u^2} f(\frac{u}{\sqrt{t}})\chi_{u\leq\sqrt{t}\alpha}$ ce qui est borné par $M\mathrm{e}^{-u^2}$ pour t assez grand [1].
   Le théorème de convergence dominée et AV.1 permettent de conclure $\to \frac{\sqrt{\pi}}{2}f(0)$.
   D'autre part, $\left|\int_\alpha^b \mathrm{e}^{-tx^2} f(x)\mathrm{d}x\right| \leq \mathrm{e}^{-t\alpha^2}\int_a^b |f(x)|\mathrm{d}x = \mathrm{o}\left(\frac{1}{\sqrt{t}}\right)$.
   Donc $F(t) \sim \frac{\sqrt{\pi}f(0)}{2\sqrt{t}}$
2. **Cas général.** On reprend les hypothèses du départ.
   On pose a $\psi : x \mapsto \sqrt{\varphi(x) - \varphi(a)}$ qui est de classe $\mathcal{C}_1$ [2] sur $[a,b[$, puisque $\psi'(x) = \frac{\varphi'(x)}{2\sqrt{\varphi(x)-\varphi(a)}} \to \sqrt{\frac{\varphi''(a)}{2}}$ par un développement de Taylor-Young [3]. C'est même un $\mathcal{C}_1$ difféomorphisme entre $[a,b[$ et un certain intervalle $[0,c[$.
   Donc $\varphi(x) = \varphi(a) + \psi(x)^2$ et $F(t) = \mathrm{e}^{-t\varphi(a)}\int_a^b f(x)\mathrm{e}^{-t\psi(x)^2}\mathrm{d}x$
   $= \mathrm{e}^{-t\varphi(a)}\int_0^c f(\psi^{-1}(y))\psi^{-1\prime}(y)\mathrm{e}^{-ty^2}\mathrm{d}y \sim \mathrm{e}^{-t\varphi(a)}\frac{\sqrt{\pi}f(\psi^{-1}(0))\psi^{-1\prime}(0)}{2\sqrt{t}}$ par le cas 1.
   $\sim \mathrm{e}^{-t\varphi(a)} f(a)\sqrt{\frac{\pi}{2\varphi''(a)t}}$.

□

Application AV.3. $\Gamma(t) \sim t^t e^{-t}\sqrt{2\pi t}$.

1. car f est continue en 0, donc bornée au voisinage de 0
2. Elle est bien définie par croissance, et il n'y a aucun problème sur $]a,b[$.
3. Comme $\varphi'(x) = \varphi'(a) + (x-a)\varphi''(a) + \mathrm{o}(x-a)$ et $\varphi(x) = \varphi(a) + (x-a)\varphi'(a) + \frac{(x-a)^2}{2}\varphi''(a) + \mathrm{o}((x-a)^2)$

Preuve. $\Gamma(t) = \int_0^{+\infty} x^t \mathrm{e}^{-x} \mathrm{d}x$.
On pose $x = t(u+1)$, alors $\Gamma(t) = \int_{-1}^{+\infty} \mathrm{e}^{-t(u+1)}(t(u+1))^t t\mathrm{d}u = t^{t+1} \int_{-1}^{+\infty} \mathrm{e}^{-t\varphi(u)}\mathrm{d}u$
avec $\varphi(u) = -\ln(u+1) + u + 1$.
En découpant en deux morceaux autour de 0 et en appliquant la méthode précédente[1] ($\varphi(0) = 1, \varphi'(0) = 0$ et $\varphi''(0) = 1$) et avec $f = 1$. On obtient deux morceaux $t^{t+1}\mathrm{e}^{-t}\sqrt{\frac{\pi}{2t}}$, ce qui en sommant fournit bien $t^t \mathrm{e}^{-t}\sqrt{2\pi t}$.

□

Postrequis.

1. Le cas où φ' ne s'annule pas. Alors φ est un $\mathcal{C}^1$-difféomorphisme.
2. Cela fonctionnerait aussi si φ possède un nombre fini de minima locaux isolés. En effet, on peut appliquer le résultat séparément au voisinage de chacun d'entre eux, puis sommer les développement limités (comme on l'a fait pour Stirling).

1. u et $-u$

AW Nombres de Catalan.

Leçons possibles : 190 243
Adapté depuis : [FGN01a], ex 1.5 p 12

Elegance : ★★★☆☆

Avis.

Un développement qui sent bon l'informatique de prépa.
A la différence de ce qu'on fera ici, [FGN01a] considère directement le problème de connaître les produits possibles pour une suite de n facteurs, ce qui je trouve rend la discussion combinatoire moins claire. On retrouvera néanmoins cet aspect dans le point 4 des Postrequis.

Prérequis.

1. Mots bien parenthésés.
   On dira qu'un mot sur $\{(,)\}$ est correctement parenthésé[1] si :
   — il a autant de parenthèses ouvrantes que fermantes ;
   — aucun préfixe n'a plus de fermantes que d'ouvrantes.

Développement.

On note P_n les mots bien parenthésés sur $2n$ parenthèses. Soit $C_n = \mathrm{Card}(P_n)$.

Théorème AW.1. $C_n = \frac{1}{n+1}\binom{2n}{n}$

Preuve. 1. **Raisonnement combinatoire.**
On a $C_0 = 1$ (mot vide), $C_1 = 1$ (mot "()")[2].
Soit $n \geq 1$ et $w_1, \dots, w_{2n} \in P_n$. Si $1 \leq i \leq n$ est le plus petit indice tel que $w_1 \dots w_2 i \in P_i$. Alors $w_2, \dots, w_{2i-2} \in P_{i-1}$[3] et $w_{2i+1}, \dots, w_{2n} \in P_{n-i}$.
Les cas étant dijsoints[4], on conclut que $C_n = \sum_{i=0}^{n-1} C_i C_{n-1-i}$[5].

2. **Séries entières : analyse.**
Soit $G(x) = \sum_{n\geq 0} C_n x^n$ la série génératrice. On suppose que son rayon de convergence R est > 0, et on prend $0 < x < R$.
D'où $G(x) = 1 + \sum_{n=1}^{+\infty} \left(\sum_{i=0}^{n-1} C_i C_{n-1-i}\right) x^n = 1 + x \sum_{n=0}^{+\infty} \left(\sum_{i=0}^{n} C_i C_{n-i}\right) x^n$
$= 1 + xG(x)^2$ en reconnaissant un produit de Cauchy.
En résolvant $xy^2 - y + 1$ on trouve $G(x) = \frac{1+\varepsilon(x)\sqrt{1-4x}}{2x}$. Puisque $G(0) = 1$ et que G est continue, il faut $\varepsilon(x) = -1$ constante.

3. **Séries entières : synthèse.**
La fonction $g : x \mapsto \frac{1-\sqrt{1-4x}}{2}$ est développable en série entière avec un rayon de convergence $\frac{1}{4}$ et vérifie $g(0) = 0$, donc $f : x \mapsto g(x)/x$ est développable de même.
Or $f(1) = 1$ et $f(x) = 1 + xf(x)^2$. Donc en reprenant les calculs précédents, on trouve (par récurrence) que les coefficients de f sont bien les C_n[6].
Donc $R = \frac{1}{4}$ et $G = f$, l'analyse était légitime.

1. Il reste à se convaincre intuitivement que cela fait bien ce qu'on pense, cf point 1 des Postrequis.
2. En fait le cas $n = 1$ est inutile, mais rassurant car la vacuité fait peur.
3. Car nécessairement $w_1, \dots, w_i$ commence par (et finit par).
4. La disjonction se faisant sur i le *plus petit* indice.
5. Cette formule est fausse pour $n = 0$.
6. Attention, le fait que $f(x)$ soit développable ne suffisait pas : peut-être était-on tombé sur cette fonction par hasard, sans vrai rapport avec les C_n !

4. **Calcul des coefficients**
   $g'(x) = \frac{1}{\sqrt{1-4x}} = (1-4x)^{-1/2}$.
   Donc pour $|x| < \frac{1}{4}$, on a $g'(x) = \sum_{n=0}^{+\infty} a_n x^n$
   avec $a_n = (-\frac{1}{2})(-\frac{3}{2})\dots(-\frac{2n-1}{2})\frac{(-4)^n}{n!} = \frac{(2n)!}{2^n 2^n n!}\frac{4^n}{n!} = \binom{2n}{n}$.
   En intégrant (g) puis en décalant les coefficients (f), on identifie dans le développement de f pour trouver $C_n = \frac{1}{n+1}\binom{2n}{n}$.

□

Postrequis.

1. Intuitivement, on dire qu'un mot bien parenthésé peut se réécrire vers ε par simplifications successives de "()". Cela fournit une définition équivalente.
2. Etude asymptotique débile.

   Proposition AW.2. $C_n \sim \frac{4^n}{n\sqrt{\pi n}}$.

   Preuve. C'est un calcul depuis la formule de Stirling.

   □

   En appliquant le critère de Cauchy, on retrouve que le rayon de convergence est $\frac{1}{4}$.
3. En fait, on peut passer par les séries formelles plutôt.
4. L'étude des mots bien parenthésés revient à celle des arbres binaires (pour décrire des expressions que l'on parenthèse).
5. Langages de Dyck (pour l'info).
   Soit $k \geq 1$ et $\Sigma_{2k} := \{(_1,)_1, (_2,)_2, \dots, (_k,)_k\}$ un alphabet de k paires de parenthèses.
   On définit $D_{2k} \subseteq \Sigma_{2k}^*$ le langage des mots bien parenthésés. Attention ici, il faut vérifier que les parenthèses ouvrantes et fermantes sont du même type, ce qui se fait très bien avec la réécriture par exemple.
   Ce sont des langages algébriques mais pas rationnels. En fait, ce sont les langages algébriques "génériques" : tout langage algébrique est image morphique de l'intersection d'un langage rationnel ("automate") et d'un langage de Dyck ("à pile").
6. Dans la même veine, on a les *mots de Motzkin*.

AX Processus de Galton-Watson.

Leçons possibles : 229 230 243 260 264
Pas de référence.

Elegance : ★★★☆☆

Avis.

Un développement extrêmement classique, mais qui se recase bien en analyse comme en probabilités (attention à ne pa abuser tout de même : 223, 226 et 228 semblent un peu exagérés.). Comme il est plutôt long, on s'attardera particulièrement à l'oral sur l'argument qui est en lien avec la leçon, quitte à passer plus vite sur le reste.
Le modèle a été introduit au XVIIIème pour étudier la propagation d'un patronyme [1] par la descendance mâle (d'où un modèle asexué). En partant d'hypothèses *a priori* très fortes, on arrive néanmoins à une disjonction de cas non-triviale et pas trop irréaliste.

Prérequis.

1. Variables aléatoires discrètes.

   Proposition AX.1 ([Ouv07], p 137). *Soit X une variable aléatoire à valeurs dans $\mathbb{N}$ et G_X sa série génératrice. Alors $\mathbb{E}[X] = \lim_{s\to 1^-} \frac{G_X(1)-G_X(s)}{1-s}$ (qu'elle soit finie ou pas).*

   Preuve. Pour tout $s \in [0,1[$, $G_X(1) - G_X(s) = \sum_{n=0}^{+\infty} \mathbb{P}(X=n)(1-s^n)$.
   Or $1 - s^n = (1-s)\sum_{k=0}^{n-1} s^k$, d'où $T(s) := \frac{G_X(1)-G_X(s)}{1-s} = \sum_{n=0}^{+\infty} \mathbb{P}(X=n)\sum_{k=0}^{n-1} s^k$ (on peut factoriser car la convergence est absolue).
   — Si $\mathbb{E}[X] = \sum_{n=0}^{+\infty} n\mathbb{P}(X=n) < +\infty$, alors la convergence de $T(s)$ est normale car $\sum_{k=0}^{n-1} s^k \leq n$. En permutant la limite $s \to 1^-$, il vient $T(s) \to \mathbb{E}[X]$.
   — Si $\mathbb{E}[X] = +\infty$, prenons $A > 0$. Alors il existe N_0 tel que pour $N \geq N_0$, $\sum_{n=0}^{N} n\mathbb{P}(X=n) \geq 2A$. Il existe aussi $\delta > 0$ tel que pour $s \in [1-\delta, 1[$, $\left|\sum_{n=0}^{N_0} \mathbb{P}(X=n)\sum_{k=0}^{n-1} s^k - \sum_{n=0}^{N_0} n\mathbb{P}(X=n)\right| \leq A$.
   Donc pour $s \in [1-\delta,1[$, $T(s) \geq \sum_{n=0}^{N_0} \mathbb{P}(X=n)\sum_{k=0}^{n-1} s^k \geq A$.
   Enfin, $T(s) \to +\infty$.

   □

2. Convexité et cie.

   Proposition AX.2. *Soit $f : I \to \mathbb{R}$ deux fois dérivable sur $\mathrm{Int}(I)$, telle que $f'' > 0$. Alors f possède au plus deux points fixes.*

   Preuve. Si $\alpha < \beta < \gamma \in I$ sont des points fixes, alors la fonction deux fois dérivable $\varphi :\mapsto f(x) - x$ s'annule en trois points. Par le lemme de Rolle (itéré), $\varphi'' = f''$ s'annule au moins une fois entre α et γ.

   □

Développement.

Objectif : modéliser l'évolution d'une population.
Formalisation : $(Z_n)_{n\geq 0}$ suite de VA dans $\mathbb{N}$ définie avec $Z_0 = 1$ et $Z_{n+1} = \sum_{k=1}^{Z_n} X_{n,k}$, où les $(X_{n,k})_{n,k}$ [2] VAIID de même loi que X [3]. On suppose $0 < \mathbb{P}(X=0) < 1$ et soit $m := \mathbb{E}[X]$.

On étudie l'événement $A_n := (Z_n = 0)$ [4].

1. De noble extraction.
2. $X_{n,k}$ est le (potentiel) nombre d'enfants du k-ième individu de la génération n. Je ne crois pas qu'introduire juste X_n soit suffisant pour les questions d'indépendance qui suivent.
3. X sert à simplifier les notations.
4. Extinction de la population.

Théorème AX.3. *Si $m \leq 1$, alors $\mathbb{P}(\bigcup_{n\geq 0} A_n) = 1$. Si $m > 1$, $0 < \mathbb{P}(\bigcup_{n\geq 0} A_n) < 1$.*

Preuve. 1. **Détermination de la série génératrice G_{Z_n}.**
Soit G_{Z_n} la série génératrice de Z_n, alors
$G_{Z_{n+1}(s)} = \mathbb{E}[s^{Z_{n+1}}] = \mathbb{E}[\sum_{k=0}^{+\infty} \chi_{Z_n=k} s^{\sum_{p=1}^{k} X_{n,p}}]$
$= \sum_{k=0}^{+\infty} \mathbb{E}[\chi_{Z_n=k}] \prod_{p=1}^{k} \mathbb{E}[s^{X_{n,p}}]$ par Fubini-Tonelli et indépendance
$= \sum_{k=0}^{+\infty} \mathbb{P}(Z_n = k)(\mathbb{E}[s^X])^k$ par ID
$= G_{Z_n}(G_X(s))$.
Par récurrence immédiate il vient $G_{Z_n} = G_X^n (= G_X \circ \cdots \circ G_X)$.

2. **Propriétés analytiques de G_X.**
   (a) En dérivant une fois, G_X est croissante sur $[0,1]$.
   (b) En dérivant deux fois, G_X est soit une droite qui n'est pas l'identité, soit strictement convexe sur $[0,1]$. Dans tous les cas, par AX.2, elle possède au plus un point fixe dans $\alpha \in]0,1[$ [1].

3. **Limite et point fixe.**
La suite $(A_n)_{n\geq 0}$ est croissante, donc $(\mathbb{P}(A_n))_{n\geq 0}$ aussi.
Soit $r := \mathbb{P}(\cup_{n\geq 0} A_n) = \lim_{n\to+\infty} \mathbb{P}(A_n)$ (qui existe). Or $\mathbb{P}(A_n) = G_{Z_n}(0) = G_X^n(0)$ donc r est un point fixe de G_X (continuité [2]), et c'est le plus petit (croissance [3]).

4. **Etude de cas.**
On sait par AX.1 que $m = \lim_{x\to 1^-} \frac{1-G_X(x)}{1-x}$.
   (a) Si $m > 1$, il existe donc $x < 1$ tel que $1 - G_X(x) > 1 - x$ donc $G_X(x) < x$. Comme $G_X(0) > 0$, par le TVI il existe $\alpha \in]0,1[$ point fixe de G_X. Donc $r = \alpha$.
   (b) Si $m < 1$, la courbe de G_X est au-dessus de la demi-tangente en 1. Le seul point fixe est est 1 et $r = 1$.
   (c) Si $m = 1$ et qu'il existe $\alpha \in]0,1[$ avec $G_X(\alpha) = \alpha$, alors $G_X(s) = s$ pour $s \in [\alpha, 1]$, car la courbe est entre sa corde et sa tangente. Par identification $G_X(s) = s$ sur tout $[0,1]$ ce qui est absurde. Le seul point fixe est 1 et $r = 1$.

□

Postrequis.

1. Cas non-traités. Si $\mathbb{P}(X = 0) = 1$ (stérilité), alors $Z_n = 0$ pour $n \geq 1$ et c'est fini. Si $\mathbb{P}(X = 0) = 0$, $Z_{n+1} \geq Z_n$ et $Z_n > 0$ pour tout n.
2. Espérance de Z_n.
   Proposition AX.4. $\mathbb{E}[Z_n] = m^n$ *(si $m < +\infty$).*

   Preuve. Par récurrence immédiate en calculant la dérivée des fonctions génératrices G_{Z_n} en 1.

   □
3. Analyse plus fine du (a). On suppose $m > 1$, (cas *sur-critique*), alors il y a extinction avec probabilité $0 < r < 1$. En fait, on peut montrer que lorsqu'il n'y a pas d'extinction, la population tend vers $+\infty$ p.s.
4. Analyse plus fine de (b) et (c). On supposera en outre que $\mathrm{Var}[X] < +\infty$. Soit τ le temps d'extinction $\min\{n \geq 0 \mid Z_n = 0\}$, on cherche un équivalent de $\mathbb{P}(\tau > n)$.
   Remarque AX.5. $\mathbb{P}(\tau > n) = \mathbb{P}(Z_n > 0) = 1 - \mathbb{P}(Z_n = 0) = 1 - G_X^n(0)$.

1. Cette remarque est inutile.
2. $G(r) = G(\lim_n G_X^n(0)) = \lim_n G_X^{n+1}(0) = r$
3. Si $G(\beta) = \beta$, alors $\forall n \geq 0$, $\beta = G_x^n(\beta) \geq G_x^n(0)$ donc $\beta \geq r$ en passant à la limite.

(b) $m = 1$ (cas *critique*). On a "lentement" extinction presque sûre.
Proposition AX.6. *Si* $m = 1$, *alors* $\mathbb{P}(\tau > n) \sim \frac{2}{n \operatorname{Var}[X]}$.

(c) $m < 1$ (cas *sous-critique*). On a "rapidement" extinction presque sûre.
Proposition AX.7. *Si* $m < 1$, *alors il existe* $C > 0$ *tel que* $\mathbb{P}(\tau > n) \sim Cm^n$.

AY Nombres normaux.

Leçons possibles : 260 264
Adapté depuis : [QZ13], p 550

Elegance : ★★☆☆☆

Avis.

Facile et pas très ambitieux. Mais loin de moi l'idée d'être ambitieux en probabilités...
Néanmoins, le développement montre que l'on comprend la notion d'indépendance et les manipulations de base sur les événements. C'est en outre un exemple de *méthode probabiliste* : pour montrer (de manière non-constructive) l'existence de certains objets, on montre que la probabilité d'en tirer un est non-nulle.

Prérequis.

1. Loi forte des grands nombres, version faible.

   Proposition AY.1. *Soit $(X_k)_{k\geq 1}$ une suite de VAIID de même loi que X et telle que $\mathbb{E}[|X|^4] < +\infty$, alors $\frac{\sum_{k=1}^n X_k}{n} \to \mathbb{E}[X]$.*

   C'est le cas en particulier des VA à valeurs dans un ensemble *fini*, et ce sera toujours le cas ici. Citons pour mémoire la "vraie" loi forte des grands nombres (qui utilise AY.1).

   Théorème AY.2 (LFGN). *Soit $(X_k)_{k\geq 1}$ une suite de VAIID de même loi que X et telle que $\mathbb{E}[|X|] < +\infty$, alors $\frac{\sum_{k=1}^n X_k}{n} \to \mathbb{E}[X]$.*

Développement.

Tout $x \in [0,1[$ admet un unique *développement propre en base* $r \geq 2$, noté $x = \sum_{k=1}^{+\infty} \frac{\varepsilon_r(x)}{r^k}$ où $\varepsilon_r(k)$ n'est pas stationnaire à $r-1$ [1]. Notons $\mathbb{P}$ la mesure de Lebesgue sur $[0,1[$.

Proposition AY.3. *Les variables aléatoires $\varepsilon_r(x)$ sont IID.*

Preuve. Elles sont mesurables [2]. Soit $a_1, \dots, a_k \in \{0, \dots, r-1\}^k$ alors
$\{x \mid \varepsilon_1(x) = a_1, \dots, \varepsilon_k(x) = a_k\} = \left[\sum_{i=1}^k \frac{a_i}{r_i}, \sum_{i=1}^k \frac{a_i}{r_i} + \frac{1}{r^k}\right[$
$\subseteq$ en encadrant le reste dans la somme, $\supseteq$ par unicité du développement.
Donc en sommant [3], il vient pour $i_1 < \dots < i_k$ que $\mathbb{P}(\varepsilon_{i_1}(x) = a_1, \dots, \varepsilon_{i_k}(x) = a_k) = \frac{1}{r^k}$.
En particulier $\mathbb{P}(\varepsilon_i(x) = a) = \frac{1}{r}$ et on a indépendance [4].
□

Le nombre x est dit *normal en base* r si pour tout $(a_1, \dots a_k)$, la fréquence d'apparition de cette séquence dans le développement tend vers $\frac{1}{r^k}$.

Théorème AY.4 (Borel). *Presque tout $x \in [0,1[$ est normal (dans toute base).*

Preuve. Il suffit de montrer que c'est vrai pour une base $r \geq 2$ fixée [5].

1. Normalité simple : $k = 1$ [6]. Il faut montrer une convergence presque sûre.
   Soit a fixé et $X_i := \chi_{\varepsilon_i(x)=a}$. Les X_i sont IID.
   Donc $\frac{\sum_{i=1}^n X_i}{n} \to^{\text{p.s.}} \frac{1}{r}$ par la LFGN. C'est ce qu'on voulait.

1. C'est la même histoire que $0.99999\dots = 1$.
2. Etant à valeurs dans un espace fini, il suffit de voir que les préimages des singletons sont clairement des boréliens.
3. Sommer sur toutes les suites de longueur i_k dont les termes $i_1, \dots, i_k$ sont fixés.
4. Sur les produits de singletons, ce qui suffit.
5. Car on obtiendra ainsi une intersection dénombrable d'événements certains.
6. Ce cas n'est pas utile autrement que pour des raisons pédagogiques (et pour rallonger la preuve).

2. Cas général. Soit $a_1, \dots, a_k$ une séquence fixée.
   Soit Y_i l'indicatrice de $\varepsilon_i(x) = a_1, \dots, \varepsilon_{i+k-1}(x) = a_k$.
   Alors les $(Y_{ik+j})_{i\geq 0}$ avec $j \in \{1, \dots, k\}$ sont indépendantes.
   On termine le raisonnement dans le cas $k = 2$[1].
   Alors $\frac{\sum_{i=1}^{2n} Y_i}{2n} = \frac{1}{2}\left(\frac{\sum_{i=1}^{n} Y_{2i}}{n} + \frac{\sum_{i=1}^{n} Y_{2i-1}}{n}\right)$ et chaque terme converge p.s. vers $\frac{1}{r^2}$ (LFGN).
   Donc $\frac{\sum_{i=1}^{2n} Y_i}{2n} \to^{\text{p.s.}} \frac{1}{r^2}$ et de même $\frac{\sum_{i=1}^{2n+1} Y_i}{2n+1} \to^{\text{p.s.}} \frac{1}{r^2}$.
   Donc $\frac{\sum_{i=1}^{n} Y_i}{n} \to^{\text{p.s.}} \frac{1}{r^2}$.

Postrequis.

1. Des exemples de nombres normaux.
   Notons que les rationnels n'ont aucune chance d'être normaux (dans aucune base), puisque leur écriture est périodique à partir d'un certain rang (*ultimement* périodique). On peut exhiber des nombres normaux dans une base donnée (ce n'est pas trivial), mais il ne me semble pas que l'on dispose d'un nombre normal en toute base.

□

1. Pour éviter d'alourdir la preuve.

CHAPITRE 2

DÉVELOPPEMENTS D'INFORMATIQUE

Te reverrai-je encor,
Heure trop fugitive,
Où mon âme au bonheur allait enfin s'ouvrir !

La damnation de Faust, Hector Berlioz

BA Complexité du tri rapide aléatoire.

Leçons possibles : 902 903 926
Pas de référence.

Elegance : ★★☆☆☆

Avis.

Il ne faut pas confondre la complexité d'un *algorithme aléatoire* avec la complexité moyenne d'un *algorithme déterministe* sur un ensemble de données. Dans le premier cas, on évalue la complexité espérée de l'exécution sur une entrée, la moyenne se faisant sur les exécutions possibles de l'algorithme. Dans le second cas, il faut munir l'ensemble de données d'une mesure de probabilité et prendre l'espérance dessus. Pour le tri rapide, les deux preuves se ressemblent beaucoup et produisent un $\mathcal{O}(n\log(n))$. On s'intéresse à la première, également faite dans [CLRS02] mais avec une rigueur insatisfaisante. Comme ce n'est pas assez long, on ajoute une étude de la complexité des pires et meilleurs aléas (à taille de donnée fixée).

Prérequis.

1. Algorithme du tri rapide aléatoire.
2. Espérances conditionnelles. On reste dans le cas discret et tout se manipule donc comme des probabilités conditionnelles (discrètes).

Développement.

Théorème BA.1. *Etant donnée une permutation de $\{1,\dots,n\}$ en entrée, l'algorithme du tri rapide aléatoire s'exécute au pire en $\mathcal{O}(n^2)$, au mieux en $\mathcal{O}(n\ln(n))$ et en temps espéré $\mathcal{O}(n\ln(n))$.*

Preuve. **Complexité au pire.** Soit $P(n)$ la complexité au pire sur les entrées de taille n.
$P(0) = 0$ et pour $n \geq 1$, $P(n) \leq n + \max_{1\leq q\leq n} P(q-1) + P(n-q-1)$.
On montre par récurrence que $P(n) \leq Cn^2 = \mathcal{O}(n^2)$.
En effet, $q \mapsto c(q-1)^2 + (n-q-1)^2$ atteint son maximum au bord en $q = 1$ ou n [1] et alors $P(n) \leq C(n-1)^2 + n = Cn^2 - Cn + 1 \leq Cn^2$ pour $C \geq 2$.

Complexité au mieux. Soit $M(n)$ la meilleure complexité sur les entrées de taille n.
Alors $M(n) \geq \min_{1\leq q\leq n} M(q-1) + M(n-q-1) + n$. On montrerait de même par récurrence que $M(n) = \Omega(n\log(n))$.

Complexité moyenne 1. On considère une permutation $\sigma \in \mathfrak{S}_n$.
Soit X^σ la VA du nombre de comparaisons effectuées par l'algorithme sur l'entrée σ, et $X^\sigma_{i,j}$ la VA des comparaisons entre les nombres i et j dans cette exécution.
Alors $\mathbb{E}[X^\sigma] = \mathbb{E}\left[\sum_{i=1}^n \sum_{j=i+1}^n X^\sigma_{i,j}\right] = \sum_{i=1}^n \sum_{j=i+1}^n \mathbb{E}\left[X^\sigma_{i,j}\right]$.

2. Pour $i < j$, $\mathbb{E}[X^\sigma_{i,j}] = \sum_{k=1}^n \mathbb{P}(\text{pivot} = k)\mathbb{E}[X^\sigma_{i,j} \mid \text{pivot} = k]$ [2].
avec $\mathbb{P}(\text{pivot} = k) = \frac{1}{n}$ [3] ;
et $\mathbb{E}[X^\sigma_{i,j} \mid \text{pivot} = k] =$
— 0 si $i < k < j$ [4] ;
— 1 si $i = k$ ou $j = k$ [5] ;

1. On dira "par convexité" pour convaincre les esprits réticents.
2. C'est la formule des probabilités totales, pour les espérances conditionnelles.
3. Car l'algorithme choisit uniformément son pivot.
4. Les deux valeurs sont seront dans deux sous-tableaux séparés.
5. C'est le pivot, il est comparé une fois avec tout le monde, puis plus jamais.

— $\mathbb{E}[X_{i,j}^{\sigma'}]$ pour une certaine $\sigma' \in \mathfrak{S}_{k-1}$ si $i < j < k$ [1] ;
— $\mathbb{E}[X_{i-k,j-k}^{\sigma''}]$ pour une certaine $\sigma'' \in \mathfrak{S}_{n-k}$ si $k < i < j$.

3. Par récurrence montrons pour $1 \leq i < j \leq n$ que $\mathbb{E}[X_{i,j}^{\sigma}] = \frac{2}{j-i+1}$ [2].
   $n = 0, 1$ c'est vrai par vacuité.
   $n \geq 1$, Par le calcul précédent $\mathbb{E}[X^{\sigma}{}_{i,j}] = \frac{1}{n}(1 + 1 + \mathbb{E}[X_{i,j}^{\sigma'}] + \mathbb{E}[X_{i-k,j-k}^{\sigma''}])$
   $= \frac{1}{n}(2 + (n-j)\frac{2}{j-i+1} + (i-1)\frac{2}{j-i+1}) = \frac{2}{n} \times \frac{j-i+1+n-j+i-1}{j-i+1} = \frac{2}{j-i+1}$.
4. Enfin $\mathbb{E}[X^{\sigma}] = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{2}{j-i+1} \leq \sum_{i=1}^{n} K \log(n) = \mathcal{O}(n \log(n))$.
   C'est même un $\theta(n \log(n))$ puisqu'on ne peut pas faire mieux.

□

Postrequis.

1. Il est facile de trouver un exemple d'exécution qui se fait effectivement en $\mathcal{O}(n^2)$.
2. Et si on ne donne pas une permutation en entrée.
   Lorsqu'il peut y avoir plusieurs fois le même élément, l'analyse précédente tombe en défaut. Il est facile de donner un exemple (suite constante) où la complexité est $\theta(n^2)$ dans tous les cas. La solution pour éviter le problème, c'est faire une partition qui trie renvoie 2 pointeurs séparant 3 blocs intérieur-égal-supérieur.

1. Les deux sont dans le même sous-tableau, on exécute récursivement (indépendamment) l'algorithme dessus. Attention, σ' est une permutation *quelconque* sur laquelle on n'a aucun contrôle.
2. C'est indépendant de σ !

BB Tri bitonique.

Leçons possibles : 902 903
Adapté depuis : [CLRS02], p 689

Elegance : ★★★★★

Avis.

Le chapitre de [CLRS02] consacré aux réseaux de tri est plutôt bien écrit. Les exemples pédagogiques n'y manquent pas. Malheureusement, ce chapitre a disparu des versions plus récentes comme [CLRS10]! Il reste donc à l'apprendre par cœur... Les réseaux de tri ne sont pas un passage obligé dans la leçon 903, mais ils constituent une jolie extension des notions étudiées. Les questions de parallélisation sous-jacentes sont très intéressantes, la notion de profondeur étant essentielle à cet égard.
On va réaliser un tri fusion avec des réseaux de comparaison. *A priori*, c'est un paradigme "Diviser-pour-régner" assez standard. Sauf que la re-combinaison des deux sous-problèmes est non-triviale avec ce modèle : elle requiert finalement un second diviser-pour-régner ! Bien expliqué, le développement peut être magnifique à l'oral.

Prérequis.

1. Réseaux de comparaison, réseaux de tri, [CLRS02], p 682.
   Définition : comparateur, réseau de comparaison (syntaxe et sémantique par induction), profondeur d'un fil, profondeur d'un réseau, réseau de tri.
2. Le fabuleux lemme du 0-1, [CLRS02], p 686.

   Théorème BB.1 (lemme du 0-1)**.** *Si un réseau de comparaison trie correctement toutes les séquences de n bits, il trie correctement toutes les séquences d'entiers (naturels).*

   > *Preuve.* On remarque que les réseaux de comparaison préservent la croissance.
   > **Lemme BB.2.** *Soit $f : \mathbb{N} \to \mathbb{N}$ une fonction croissante, et $\mathcal{R}$ un réseau de comparaison sort $b_1, \dots, b_n$ (permutation) sur l'entrée $a_1, \dots, a_n$.*
   > *Alors $\mathcal{R}$ sort $f(b_1), \dots, f(b_n)$ sur $f(a_1), \dots, f(a_n)$.*
   >
   > > *Preuve.* On a $\min(f(x), f(y)) = f(\min(x), \min(y))$[1] et donc f "passe aux comparateurs". Ensuite on conclut par induction.
   > >
   > > □
   >
   > Supposons qu'un réseau $\mathcal{R}$ trie correctement toutes les séquences de n bits, mais qu'il existe une séquence $a_1, \dots, a_n$ d'entiers non-triés. Dans la sortie $b_1, \dots, b_n$ il existe $i < j$ avec $b_i > b_j$.
   > Considérons la fonction croissante $f := \chi_{x \geq b_i}$, alors $f(b_i) = 1$ et $f(b_j) = 0$ donc la séquence $f(b_1), \dots f(b_i) = 1, \dots, f(b_j) = 0, \dots, f(b_n)$ est non-triée. Pourtant par BB.2 c'est la sortie de $\mathcal{R}$ sur une séquence de bits, ce qui est impossible.
   >
   > □

3. Le tri fusion.
   Idée : séparer la liste en deux, puis la recombiner en une liste triée. La recombinaison est facile si l'on peut manipuler des listes et faire des boucles.

Développement.

Théorème BB.3. *Pour les entiers $n \geq 0$ et puissances de 2, on peut construire une famille de réseaux de tri de profondeur $\theta(\log^2(n))$.*

1. Pas la peine d'en faire tout un plat comme dans la preuve de [CLRS02].

Preuve. Dans toute la preuve n est une puissance de 2, et en vertu de BB.1, on considèrera maintenant seulement des séquences de $0-1$.

1. **Fusion et séquences bitoniques**

   (a) Idée : utiliser le tri fusion (cf Figure 2.1). Question : comment fusionner ?

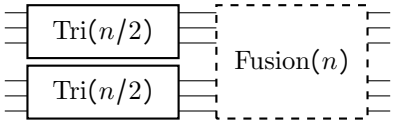


FIGURE 2.1 – Principe d'un réseau de tri par fusion

   (b) Une séquence est dite *bitonique* si elle est de la forme $0^*1^*0^*$ ou $1^*0^*1^*$.
   Notons que si $a_1, \dots, a_{n/2}$ et $a_{n/2+1}, \dots, a_n$ sont triées, alors leur concaténation en renversant la deuxième $a_1, \dots, a_{n/2}, a_n, \dots, a_{n/2+1}$ est bitonique [1]. En effet, les deux séquences sont de la forme 0^*1^* et en les concaténant il vient $0^*1^*0^*$.

2. **Construction d'une trieuse bitonique**

   (a) Notons S_n le *séparateur* à n entrées.
   Sur l'entrée $a_1, \dots, a_n$ il compare a_i et $a_{n/2+i}$ pour tout $1 \leq i \geq n/2$ (cf Figure 2.2).

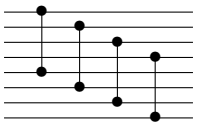


FIGURE 2.2 – Le séparateur S_8

   Notons la sortie $b_1, \dots, b_{n/2}, b_{n/2+1}, \dots b_n$.
   Proposition BB.4. *Si $a_1, \dots, a_n$ est bitonique, alors après S_n ; les sorties $b_1, \dots, b_{n/2}$ et $b_{n/2}+1, \dots b_n$ sont bitoniques, et l'une au moins est constante et bien placée* [2].

   Preuve. Superposons les deux moitiés $\begin{pmatrix} a_1 \\ a_{n/2+1} \end{pmatrix} \cdots \begin{pmatrix} a_{n/2} \\ a_n \end{pmatrix}$.
   Alors $b_1, \dots, b_{n/2}$ est le min des colonnes, et $b_{n/2}+1, \dots b_n$ le max.
   Le reste est une analyse de cas en fonction de la structure de $a_1, \dots a_n$ (que l'on peut supposer dans $0^*1^*0^*$).
   □

   (b) On construit donc par récurrence un bloc TriBit(n).
   — pour $n = 1$ c'est un fil ;
   — pour $n \geq 2$, TriBit(n) est construit en Figure 2.3.

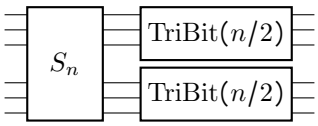


FIGURE 2.3 – Tri des séquences bitoniques

1. On a donc réduit le problème de la fusion au problème du tri d'une séquence bitonique.
2. Constante et bien placée : comprendre que si elle est constante= 0 elle se retrouve en bas, et constante= 1 en haut.

On montre par récurrence sur $n \geq 1$ que $\mathrm{TriBit}(n)$ trie correctement toutes les séquences bitoniques.

3. **Construction d'une trieuse**
   On construit par récurrence un bloc $\mathrm{Tri}(n)$.
   — pour $n = 1$ c'est un fil ;
   — pour $n \geq 2$, $\mathrm{Tri}(n)$ est construit en Figure 2.4 (attention à l'inversion des fils).

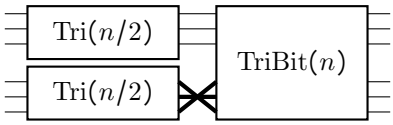


FIGURE 2.4 – Réseau de tri bitonique

   On montre par récurrence sur $n \geq 1$ que $\mathrm{Tri}(n)$ est un réseau de tri.

4. **Profondeur des réseaux**
   On a $\mathrm{d}(S_{2^k}) = 1$ et $\mathrm{d}(\mathrm{TriBit}(2^k)) = \mathrm{d}(S_{2^k}) + \mathrm{d}(\mathrm{TriBit}(2^{k-1}))$ [1]. D'où $\mathrm{d}(\mathrm{Tribit}(2^k)) = k$.
   De même il vient $\mathrm{d}(\mathrm{Tri}(2^k)) = \sum_{i=0}^{k} i = \theta(k^2)$.
   Donc $\mathrm{d}(\mathrm{Tri}(n)) = \theta(\log^2(n))$.

□

Postrequis.

1. En général, une séquence bitonique est une séquence croissante puis décroissante, à permutation circulaire près. Si on se limite aux $0-1$, c'est équivalent à notre définition.
2. On peut déduire un réseau depuis le tri insertion, ou le tri bulle (c'est le même).
   Il est de profondeur n, donc fatalement moins bon.
3. Il existe d'autres réseaux de tri de profondeur $\mathcal{O}(\log^2(n))$, comme le tri *pair-impair de Batcher* qui repose encore sur un tri fusion, mais propose un fusionneur différent.
4. Il existe même des réseaux de tri de profondeur $\mathcal{O}(\log(n))$ [AKS83]. Néanmoins la construction est beaucoup plus compliquée que les précédentes, et les constantes du $\mathcal{O}$ le rendent finalement moins intéressant en pratique.
5. Complexité des problèmes associés.

   Proposition BB.5. *Tester si un réseau de comparaison est un réseau de tri est* coNP *complet.*

   Pour le vérifier, il suffit d'essayer pour les séquences de $0-1$. Autrement dit, l'algorithme coNP doit en deviner une qui invalide le tri.

1. Si on voulait être précis, il faudrait voir que tous les fils ont la même profondeur.

BC Arbres AVL.

Leçons possibles : 901 921 926
Pas de référence.

Elegance : ★★★★☆

Avis.

Les développements de ce type ont l'avantage de pouvoir être justifiés par des dessins. On peut même presque atteindre une présentation "pédagogique". Joli et sympathique, celui-ci ne mange pas de pain une fois que l'on a compris le principe des rotations. C'est aussi un exemple de structure de donnée dont la complexité est non-triviale. Enfin on illustre l'intérêt et les difficultés d'un passage du linéaire à l'arborecent.

Prérequis.

1. Arbres binaires de recherche (ABR)

   Définition BC.1. *On appelle ABR un arbre binaire (fini) dont les feuilles sont étiquetées par des entiers et tel que* $\mathrm{e}(x.\mathrm{FilsGauche}) \leq \mathrm{e}(x) \leq \mathrm{e}(x.\mathrm{FilsDroit})$ *(quand ces nœuds existent).*

   On peut implémenter récursivement les opérations de recherche et d'insertion dans un ABR. Voir l'Algorithme 2 pour l'insertion d'une clef k dans A. Malheureusement, celles-ci se font dans le pire cas en temps $\mathcal{O}(n)$ (cas d'un arbre "liste").

Algorithme 2 : Insertion dans un ABR

```
Fonction InsertionABR(A, k) :
    si A = vide alors
        retourner Feuille(k)
    sinon
        si A.racine > k alors
            A.gauche ← InsertionABR(A.gauche,k)
        sinon
            A.droite ← InsertionABR(A.droite,k)
        fin
    fin
fin
```

Développement.

On appelle un nœud d'un arbre binaire est dit k-*équilibré* si la hauteur de ses fils droits et gauche diffère d'au plus k. Un AVL est un arbre binaire dont tout nœud est 1-équilibré.

Lemme BC.2. *Un AVL de hauteur* h *possède au moins* F_h [1] *feuilles.*

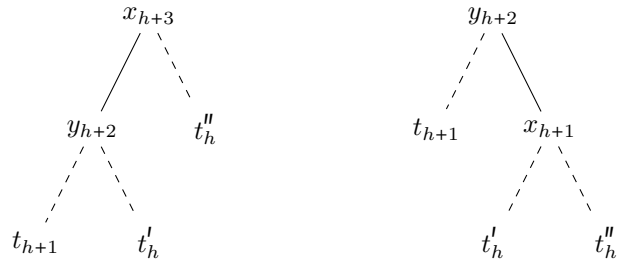
Preuve. On procède par récurrence sur h.
$h = 0, 1$. Un arbre de hauteur $h = 0$ est vide, de hauteur $h = 1$ a 1 feuille (racine).
$h \geq 2$. Un tel arbre a un racine et deux sous-arbres dont l'un est de hauteur $h + 1$ et l'autre de hauteur $h + 1$ ou h puisqu'il est 1-équilibré. Par hypothèse, le premier a au moins F_{h+1} feuilles et le second au moins $\min(F_h, F_{h+1}) = F_h$.
Donc l'arbre a au moins $F_h + F_{h+1} = F_{h+2}$ feuilles.

□

1. h-ième nombre de Fibonacci, défini par la récurrence $F_0 = 0, F_1 = 1$ et $F_{h+2} = F_{h+1} + F_h$.

Or $F_h \sim [(1+\sqrt{5})/2]^h$ [1]. Donc la hauteur h d'un AVL est $\mathcal{O}(\log(n))$ où n est le nombre de feuilles (et *a foritori* de noeuds). C'est donc un bon arbre pour faire des requêtes d'ABR. Comment maintenir l'équilibre ?

Question : Si dans un arbre tout nœud est 1-équilibré, sauf peut-être la racine qui est 2-équilibrée, peut-on le rendre 1-équilibré partout ? On étudie un exemple de cas possible en Figure 2.5 [2], la solution étant de faire une *rotation* autour d'un sommet.



(a) Arbre T_1 dont la racine n'est pas 1-équilibrée.

(b) Arbre T_1 après RD(y) : il est 1-équilibré partout.

FIGURE 2.5 – Un exemple de correction par une rotation.

Les autres cas se traitent de manière similaire avec 1 ou 2 rotations [3].
En outre, ces opérations :
- se font en temps constant ;
- préservent la propriété d'être un *ABR* ;
- conservent ou font diminuer la hauteur de 1.

On en déduit le schéma l'Algorithme 3.

Algorithme 3 : Insertion dans un AVL

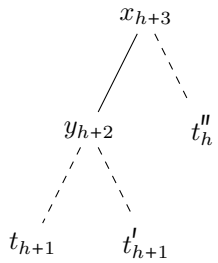
```
Fonction InsertionAVL(A, k) :
    si A = vide alors
        retourner Feuille(k)
    sinon
        si A.racine > k alors
            A.gauche ← InsertionAVL(A.gauche,k)
        sinon
            A.droite ← InsertionAVL(A.droite,k)
        fin
        Corriger l'équilibrage par des rotations.
    fin
fin
```

Il s'exécute au pire en $\mathcal{O}(h)$ soit $\mathcal{O}(\log(n))$ (attention à maintenir pour chaque nœud maintenir un champ pour sa hauteur [4]). On justifie sa correction avec l'invariant suivant : "InsertionAVL(A,k) insère la valeur k dans l'AVL, et retourne un AVL dont la hauteur a au pire augmenté de 1".
En faisant de même pour les autres opérations, on en déduit le résultat suivant.

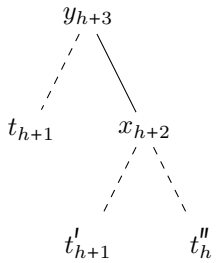
Théorème BC.3. *Dans un AVL, on peut insérer, supprimer et chercher en $\mathcal{O}(\log(n))$.*

1. On peut résoudre explicitement une récurrence linéaire d'ordre 2, merci.
2. la hauteur est exprimée en indice, et les t pointillés sont des sous-arbres (potentiellement vides)
3. On traite les autres cas intéressants en Figure 2.6 (ceux qui restent sont vraiment symétriques).
4. Ou plus efficacement, la valeur du déséquilibre entre fils droit et gauche suffirait.

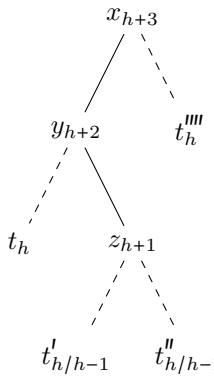
Postrequis.



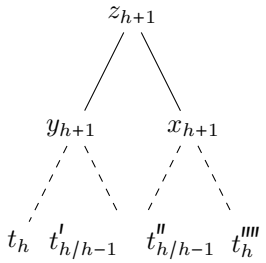
(a) Arbre T_2.



(b) Arbre T_2 après RD(y) : il est 1-équilibré partout.



(c) Arbre T_3.



(d) Arbre T_2 après RD(x) et RG(y) : il est 1-équilibré partout.

FIGURE 2.6 – Autres corrections par les rotations.

1. Après une insertion, il y aura au plus une rotation (double) à faire (regarder les cas).
2. Et pour les suppressions ? On peut avoir à faire jusqu'à $\Omega(\log(n))$ rotations à faire. Pour le voir, prendre un arbre le plus déséquilibré, nommément celui de Fibonacci (qui apparaît dans la preuve de BC.2). Une insertion au "mauvais endroit" oblige à tout tourner.
3. Application : le *tri arborescent*.
   En utilisant cette structure de données, on obtient un algorithme de tri en $\mathcal{O}(n\log(n))$: on insère toutes les clefs dans un AVL en $n\mathcal{O}(\log(n))$ puis on parcourt l'arbre en gauche-droite pour obtenir les clefs croissantes en $\mathcal{O}(n)$.
   Cette procédure est un peu duale du *tri pas tas binaire* (on peut construire le tas en $\mathcal{O}(n)$, puis l'obtention du tableau trié est en $n\mathcal{O}(\log(n))$), mais elle ne peut pas être implémentée en place dans un tableau. La manipulation des pointeurs dans un AVL la rend donc moins efficace, bien que partageant la même complexité asymptotique.
4. L'intérêt des arbres de recherche est surtout historique et pédagogique. La structure de données *dictionnaire* est de nos jours implémentée efficacement par des tables de hachage Il serait donc naïf de prétendre utiliser en pratique les ABR. Néanmoins, il existe moult variantes des AVL permettant de maintenir des ABR en $\mathcal{O}(n\log(n))$ foisonnent, on pourra citer les arbres rouge-noir (mais leur construction est encore plus lourde), les b-arbres (motivés par des questions d'accès disque), les arbres splay (étude de complexité amortie).

BD Hachage parfait.

Leçons possibles : 901 921
Adapté depuis : [CLRS10], p 258

Elegance : ★★☆☆☆

Avis.

Un développement simple mais efficace. On peut aussi en parler dans la 926 comme illustration de la complexité des algorithmes probabilistes, ou pour discuter de complexité en espace (par rapport au hachage naïf qui serait trop gros).

Prérequis.

1. Philosophie du hachage.
   On veut stocker n clefs parmi un ensemble de très grande taille U. On utilisera un ensemble de taille m pour les stocker (comme $0, \dots, m-1$). Objectif : accéder aux clefs que l'on a, en ajouter, en supprimer. Et ce le plus efficacement possible.
   L'idée est de se donner une fonction de hachage $h : U \to \{0, \dots, m-1\}$, facilement calculable, en priant ne pas avoir trop de collisions sur les données. Il suffit ensuite de travailler avec un tableau $\{0, \dots, m-1\}$, choisi beaucoup plus petit que $|U|$.
2. Familles universelles.
   Définition BD.1. *Une famille $\mathcal{H}$ de fonctions de hachage est dite (1-)universelle si pour toute paire $x \neq y$ de clefs, $\{h \in \mathcal{H} \mid h(x) = h(y)\} \leq \frac{|\mathcal{H}|}{m}$.*
   Proposition BD.2 ([CLRS10], Th. 11.5 p 248)**.** *Soit $p > m$, la famille $\{x \mapsto (ax+b) \mod p \mod m \mid 0 < a < p, 0 \leq b < p\}$ est universelle.*

   > *Preuve.* En utilisant la bijectivité de $x \mapsto ax+b$ dans $\mathbb{Z}/p\mathbb{Z}$ et une propriété d'uniformité, on montre que si $x \neq y$, leurs images sont toujours différentes et réparties uniformément dans $\mathbb{Z}/p\mathbb{Z}$ quand a, b varient. On a donc ramené le dénombrement sur les fonctions de hachage en dénombrement sur les valeurs...
   > Il suffit de compter le nombre de cas où $r = s \mod m$ quand $r \neq s$ sont choisis dans $\mathbb{Z}/p\mathbb{Z}$. Etant donnée une valeur de r, il y a de l'ordre de p/m valeurs de s qui rentrent en collision. On a donc de l'ordre de p^2/m couples qui font des collisions, d'où le $\frac{1}{m}$. On peut donner des majorations exactes, je ne le fais pas.
   > □

Développement.

On considère une famille universelle $\mathcal{H}$ dans laquelle on tire uniformément.

Proposition BD.3. *On prend $m = n^2$. Soit X la variable aléatoire qui compte le nombre de collisions, alors $\mathbb{P}(X \geq 1) \leq \frac{1}{2}$.*

> *Preuve.* Il y a $\binom{n}{2} = \frac{n(n-1)}{2}$ paires de clefs susceptibles d'entrer en collision. Chacune a une probabilité au plus $\frac{1}{m}$ d'arriver (par universalité). Si X la variable aléatoire qui compte le nombre de collisions, par linéarité $\mathbb{E}[X] \leq \frac{n(n-1)}{2m} \leq \frac{n-1}{2n} < \frac{1}{2}$.
> Enfin $\mathbb{P}(X \geq 1) \leq \mathbb{E}[X] < \frac{1}{2}$ par l'inégalité de Markov.
> □

Proposition BD.4. *On prend $m = n$. Alors si N_i est la VA du nombre de collisions sur i ($1 \leq i \leq m$), Alors $\mathbb{P}(\sum_{i=1}^{m} N_i^2 \geq 4n) \leq \frac{1}{2}$.*

Preuve. Notons que $\sum_{i=1}^{m} N_i = n$ constante. D'autre part $\sum_{i=1}^{m} \binom{N_i}{2} = X$ (car c'est le nombre total de collisions).
Or $N_i^2 = N_i + 2\binom{N_i}{2}$, donc $\mathbb{E}[\sum_i N_i^2] = \mathbb{E}[\sum_i N_i] + 2\mathbb{E}[\sum_i \binom{N_i}{2}] = n + 2\mathbb{E}[X]$.
On a vu que $\mathbb{E}[X] \leq \frac{n(n-1)}{2m} \leq \frac{n-1}{2} < \frac{n}{2}$.
Donc $\mathbb{E}[\sum_i n_i^2] \leq 2n$. Enfin, en appliquant l'inégalité de Markov, $\mathbb{P}(\sum_i n_i^2 \geq 4n) \leq \frac{1}{2}$. □

Pour le hachage parfait, on va créer deux tables.

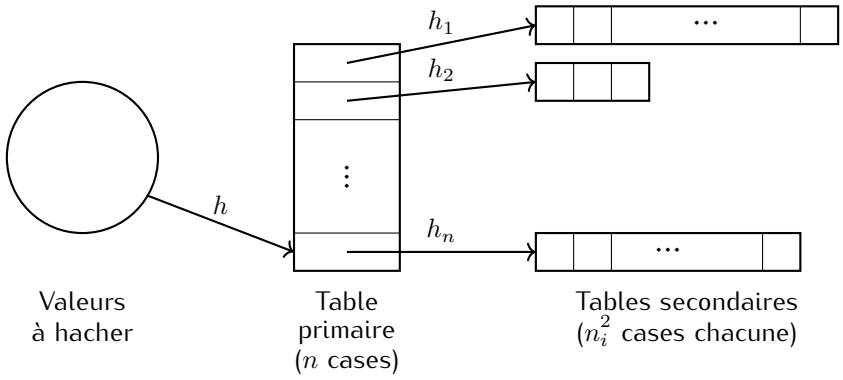


FIGURE 2.7 – Principe du hachage parfait.

Il faut donc :

1. tirer une fonction de hachage primaire et vérifier que $\sum_{i=1}^{n} n_i^2 \leq 4n$.
   Temps $\mathcal{O}(n)$, espace $\mathcal{O}(n)$ [1] et nombre d'itérations espéré ≤ 2 [2].
2. pour chaque i, tirer une fonction de hachage auxiliaire et vérifier l'absence de collision.
   Temps $\mathcal{O}(n_i)$ et espace $\mathcal{O}(n_i)$ [3] et nombre d'itérations espéré ≤ 2.

Au total la construction se fait en temps espéré $\mathcal{O}(n)$ et en espace $\mathcal{O}(n)$.
Une fois la table construite, les requêtes se font en $\mathcal{O}(1)$ déterministe.

Postrequis.

1. Il existe d'autres familles universelles, plus ou moins ésotériques.
2. Outre le hachage universel (avec rehashing), l'*adressage ouvert*, assez différent, est très utilisé en pratique. Voir [CLRS10] et la technique de *double hachage*.

1. On crée un tableau de taille n, on calcule les n_i (parcours des clefs). Enfin on calcule $\sum n_i^2$ et on regarde si $\leq 4n$.
2. Puisque la probabilité d'échecs successifs suit une loi géométrique de paramètre $\leq \frac{1}{2}$, donc d'espérance 2 fixée.
3. Il suffit de vérifier si les valeurs primairement hachées vers n_i entrent en collision par la fonction de hachage auxiliaire. Cela se fait en $\mathcal{O}(n_i)$ pour peu que l'on ait pré-calculé la liste des collisions en n_i (intégré à la première étape).

BE Distance d'édition.

Leçons possibles : 906 907
Pas de référence.

Elegance : ★★☆☆☆

Avis.

Présenter uniquement l'équation de récurrence sur la distance, et l'algorithme qui en découle, paraît peu intéressant. Ici on définit la distance d'édition de manière "naturelle" en utilisant un système de réécriture (les "fautes"), puis on justifie proprement l'équation de récurrence à l'aide d'un autre modèle : les alignements. L'aspect programmation dynamique n'était pas évident au départ, et c'est cette non-trivialité que l'on cherche à faire ressortir !

Prérequis.

1. Distance d'édition, définition propre.

   Définition BE.1. *On définit le système de réécriture* $\to$ *sur* Σ^* *par* $w \to w'$ *si et seulement si* $w = uav$ *et* $w' = ubv$ *avec* $u, v \in \Sigma^*$ *et* $a, b \in \Sigma \cup \{\varepsilon\}$.

   Notons que ce système est symétrique. En outre, le graphe associé est connexe.

   Proposition BE.2. *Pour* $w, w' \in \Sigma^*$, *on a toujours* $w \to^* w'$.

   Preuve. Il suffit de remarquer que $w \to^* \varepsilon$ et $\varepsilon \to^* w'$. □

   Définition BE.3 (Distance d'édition). *Si* $w, w' \in \Sigma^*$, *leur distance d'édition est* $d(w, w') := \min\{n \geq 0 \mid w \to^n w'\}$.

   L'intuition est donc la suivante : les mots "à distance au plus n" sont tous les mots que l'on peut atteindre en faisant au plus n fautes élémentaires.

   Remarque BE.4. *A priori, c'est compliqué à calculer...*

Développement.

Définition BE.5. *On appelle* alignement de u et v *la donnée de deux mots* $\overline{u}, \overline{v} \in \Sigma \uplus \{\square\}$ *tels que* $|\overline{u}| = |\overline{v}|$ *et si* μ *est le morphisme effaçant les* $\square$, $\mu(\overline{u}) = u$ *et* $\mu(\overline{v}) = v$.

L'écart d'un alignement est le nombre de lettres qu'il faut modifier entre $\overline{u}$ et $\overline{v}$ [1]

Exemple BE.6. $\binom{s}{\square}\binom{o}{\square}\binom{r}{r}\binom{\square}{o}\binom{t}{t}\binom{i}{i}\binom{e}{s}$ *d'écart* 4.

Définition BE.7. $d_A(u, v)$ *est l'écart du plus petit alignement de* u *et* v.

Proposition BE.8. $d_A = d$.

Preuve. On montre d'abord par récurrence sur n que si$u \to^n v$, il existe un alignement d'écart $\leq n$ [2]. Le cas $n = 0$ est clair. Sinon la dernière réécriture $w \to v$. Que ce soit une insertion, une suppression ou une modification, on peut traduire en modifiant en conséquence l'alignement, et cela n'augmente son écart que d'au plus 1.
Réciproquement on montrerait par récurrence sur $|\overline{u}|$ que si on a un alignement $\overline{u}, \overline{v}$ d'écart n on peut construire une réécriture d'écart $\leq n$.

□

On a "ordonné" les dérivations, ce qui permet d'obtenir une équation de récurrence.

Proposition BE.9. *Pour* $u, v \in \Sigma^*$, $a, b \in \Sigma$, *on a :*

1. Elles matérialisent les suppressions/insertions/modifications.
2. On ne cherche pas le =, puisqu'on est susceptible de modifier plusieurs fois une même lettre.

— $d(\varepsilon, u) = |u|$ *et* $d(u, \varepsilon) = |u|$;
— $d(ua, vb) = \min\{d(ua, v) + 1, d(u, va) + 1, d(u, v) + \delta_{a=b}\}$.

Preuve. 1. C'est clair puisque $\to$ ne peut ajouter qu'une lettre à la fois.

2. Il vient que $d(ua, vb) \leq \min\{d(ua, v), d(u, va), d(u, v) + \delta_{a=b}\}$ en remarquant qu'on peut écrire des réécritures de taille $d(ua, v) + 1$, $d(u, va) + 1$ ou $d(u, v) + \delta_{a=b}$.
Soit maintenant un alignement optimal entre ua et vb. Supposons que la dernière paire est une modification $\binom{a}{b}$, alors le reste est nécessairement un alignement d'écart optimal pour u et v[1]. Les autres cas se traitent de même.

□

On en déduit l'Algorithme 4 exploitant la programmation dynamique pour trouver $d(u, v)$ en temps $\mathcal{O}(|u||v|)$. Les sous-problèmes sont les distances pour entre préfixes de u et de v.

Algorithme 4 : Calcul de la distance d'édition.

Fonction *DistanceEdition*(u, v) *:*
  $d \leftarrow$ Tableau$(|u| + 1, |v| + 1)$
  `# Initialisation`
  pour $0 \leq i \leq |u|$ **faire**
    $d[i, 0] \leftarrow u$
  fin
  pour $0 \leq j \leq |v|$ **faire**
    $d[0, j] \leftarrow j$
  fin
  pour $1 \leq i \leq |u|$ **faire**
    pour $1 \leq j \leq |v|$ **faire**
      si $u[i] = v[j]$ **alors**
        $d[i, j] \leftarrow \min(d[i-1, j] + 1, d[i, j-1] + 1, d[i-1, j-1])$
      sinon
        $d[i, j] \leftarrow \min(d[i-1, j] + 1, d[i, j-1] + 1, d[i-1, j-1] + 1)$
      fin
    fin
  fin
fin

Postrequis.

1. Trouver un alignement. On peut laisser des marqueurs dans le tableau, afin de déterminer un alignement optimal (il peut y en avoir plusieurs différents). C'est classique en programmation dynamique, il suffit de montrer "par où on est passé dans le tableau".
2. Variante. On peut attribuer un poids différent à chacune des opérations, voire aux lettres. Cela donnera une autre équation de récurrence mais un calcul similaire
3. Complexité spatiale. Ici on utilise $\mathcal{O}(|u||v|)$. On peut en fait se restreindre à $\mathcal{O}(|u|)$ (ou $\mathcal{O}(|v|)$, au choix) en ne retenant que la dernière ligne du tableau. Si u est fixé, on peut même en déduire un automate qui reconnaît les mots à distance k.
4. String matching with errors. [CR94], p 266.
   Etant donné un texte t et un motif m, on peut chercher $s(t, m) := \min\{d(w, m) \mid w \text{ facteur de } t\}$.
   Proposition BE.10. *On peut résoudre ce problème en* $\mathcal{O}(|t||m|)$.

1. Sinon, on pourrait faire mieux au total.

Preuve. Notons $S(t, m) = \min\{d(w, m) \mid w \text{ suffixe de } t\}$. Alors S vérifie la relation de récurrence BE.9. Attention, on a $S(t, \varepsilon) = 0$ pour l'initialisation. Ensuite, il suffit de prendre $\min\{S(t', m) \mid t' \text{ suffixe de } t\}$.

□

On peut même déterminer les facteurs qui sont à distance ≤ k de m.
En fait, si k est fixé, on peut faire mieux !

Proposition BE.11. *On peut réaliser chercher les facteurs à distance k de m en $\mathcal{O}(k|t|)$.*

L'algorithme dédié repose sur un calcul parcimonieux du tableau de S.

BF Automate des bordures.

Leçons possibles : 907
Pas de référence.

Elegance : ★★★★☆

Avis.

A évoquer bien entendu dans la leçon d'automates, mais il y avait aussi des développements un peu moins arides pour cette leçon, alors j'ai fait le choix de ne pas le développer. Attention, BF.4 sur les chevauchements n'est absolument pas trivial, et c'est vraiment le résultat central, alors il serait dommage de le mal prouver ou de l'admettre.
L'automate des bordures est je crois essentiel, dans la mesure où ses états indiquent *exactement* l'information utile à retenir pour la recherche de motif. Il s'agit d'un *chevauchement*.

Prérequis.

1. Quelques notions de combinatoire des mots.

   On notera parfois $u \sqsubseteq v$ pour "u préfixe de v" et $u \sqsupseteq v$ pour "u suffixe de v" [1].

   Définition BF.1 (Chevauchement). *Si $u, v \in \Sigma^*$, un chevauchement entre u et v est un suffixe de u qui est préfixe de v. On note $\mathrm{ch}(u,v) \in \Sigma^*$ le plus long chevauchement entre u et v.*

   Exemple BF.2. $\mathrm{ch}(aabab, abcaa) = ab$.

   Définition BF.3 (Bordure). *Si $u \in \Sigma^+$, une bordure de u est un suffixe de u qui est aussi un préfixe. On note $\mathrm{Bord}(u) \in \Sigma^*$ la plus grande bordure propre de u (autrement dit $\neq u$).*

Développement.

Lemme BF.4. $\mathrm{ch}(ta, m) = \mathrm{ch}(t,m)a$ *si* $\mathrm{ch}(t,m)a \sqsubseteq m$ *;* $\mathrm{Bord}(\mathrm{ch}(t,m)a)$ *sinon.*

Preuve. 1. Montrons d'abord que si w est un chevauchement entre ta et m, c'est une bordure de $\mathrm{ch}(u,v)a$. Si $w = \varepsilon$ c'est évident. Sinon $w = va$, voir Figure 2.8

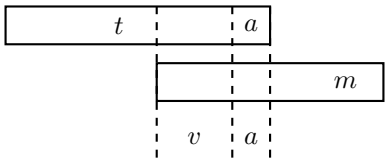


Figure 2.8 – Chevauchement $w \neq \varepsilon$ entre ta et m.

Donc v chevauchement entre t et m. Donc $v \sqsupseteq \mathrm{ch}(t,m)$ et $v \sqsubseteq \mathrm{ch}(t,m)$ [2].

(a) Soit $v = \mathrm{ch}(t,m)$, alors $w = va = \mathrm{ch}(t,m)a$ qui est une bordure de lui-même.

(b) Soit v est un préfixe strict de $\mathrm{ch}(t,m)$, lui-même préfixe de m. Or la lettre suivant v dans m est un a, donc $w = va \sqsubseteq \mathrm{ch}(t,m) \sqsubseteq \mathrm{ch}(t,m)a$. En outre $va \sqsupseteq \mathrm{ch}(t,m)a$ [3]. Donc c'est une bordure de $\mathrm{ch}(t,m)a$.

2. Si $\mathrm{ch}(t,m)a \sqsubseteq m$. On aussi $\mathrm{ch}(t,m)a \sqsupseteq ta$, donc $\mathrm{ch}(t,m)a$ un chevauchement entre ta et m. Par 1 c'est le plus grand possible.
3. Sinon $\mathrm{Bord}(\mathrm{ch}(t,m)a)$ est un chevauchement entre ta et m, voir Figure 2.9.

1. Attention, ce n'est pas du tout symétrique : on ne peut pas renverser les notations.
2. Puisque c'est le plus long chevauchement, et qu'ils sont ordonnés pour l'ordre suffixe et l'ordre préfixe.
3. Evident, c'est la "monotonie à droite" des suffixes.

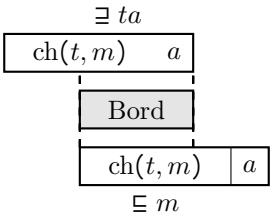


FIGURE 2.9 – $\mathrm{Bord}(\mathrm{ch}(t,m)a)$ est un chevauchement entre ta et m.

En vertu de 1 et 2 c'est le plus grand.

□

Fixons-nous un motif $m \in \Sigma^*$ à reconnaître, de longueur $|m|$.
On veut construire un algorithme qui "retient" $\mathrm{ch}(t,m)$ pour éviter de relire les lettres.

Définition BF.5. *Soit l'automate* $\mathcal{A} = (Q, q_0, q_f, \delta)$ *avec :*
- $Q = \mathrm{Pref}(m)$ *(*$|Q| = |m| + 1$*);*
- $q_0 = \varepsilon$ *;*
- $q_f = m$ *;*
- $\delta(ua, a) = ua$ *si* $ua \sqsubseteq m$, $\mathrm{Bord}(ua)$ *sinon ;*

Proposition BF.6. *L'automate* $\mathcal{A}$ *reconnaît* $\Sigma^* m$.

Preuve. Montrons par récurrence sur $|t|$ que $\delta(q_0, t) = \mathrm{ch}(t, m)$.

Initialisation $t = \varepsilon$ c'est évident car $q_0 = \varepsilon$.

Hérédité $t \to ta$. On suppose que $\delta(q_0, t) = \mathrm{ch}(t, m) =: u$.
Alors $\delta(q_0, ta) = \delta(u, a) = ch(ta, m)$ par BF.4.

□

Il suffit donc dans la lecture de t, de marquer les positions où $\mathcal{A}$ atteint q_f en lisant le texte.

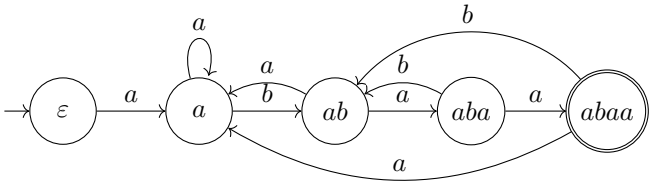


FIGURE 2.10 – L'automate associé au motif $abaa$.

Postrequis.

1. En fait, l'automate $\mathcal{A}$ est l'automate minimal de $\Sigma^* m$. Ce n'est pas très dur à prouver, il suffit de supposer que l'on a strictement moins de $|m| + 1$ états et de remarquer qu'alors deux préfixes conduisent au même état.
2. Complexité 1 : la foire aux automates.
   Une fois l'automate construit, on peut faire de la recherche de motif en $\mathcal{O}(|t|)$.
   En revanche, l'automate prend un espace $\mathcal{O}(|\Sigma||m|)$. Pour le calculer on peut :
   (a) calculer naïvement les bordures en $\mathcal{O}(|\Sigma||m|^2)$;
   (b) remarquer que l'automate non-déterministe $\mathcal{B}$ (évident à construire) pour $\Sigma^* m$ est co-déterministe. Donc son déterminisé par la méthode des parties est l'automate minimal (théorème de Brzozowski), autrement dit $\det(\mathcal{B}) = \mathcal{A}$. La déterminisation de $\mathcal{B}$ peut

sembler une mauvaise idée, mais rappelons que sa complexité est $\mathcal{O}(|\det(\mathcal{B})|)$ [1], ce qui heureusement ici est petit ! Modulo implémentation, on retrouve à peu près la même complexité que pour la méthode naïve ;

(c) faire une construction *en ligne* de l'automate (méthode d'Aho-Corasick) pour un total de $\mathcal{O}(|\Sigma||m|)$. La technique se généralise subtilement à des ensembles de mots ;

(d) calculer intelligemment les bordures en $\mathcal{O}(|\Sigma||m|)$.

3. Complexité 2 : algorithmes MP et KMP.
   Si Σ est un gros alphabet, les complexités précédentes sont tout de même trop importantes. L'idée de MP et KMP est de calculer une approximation des bordures en $\mathcal{O}(|m|)$. Celle-ci nous amène parfois à tester plusieurs fois la même lettre pour savoir où aller dans l'automate, mais la lecture reste en $\mathcal{O}(|m| + |t|)$.

4. Complexité 3 : l'automate de Simon.
   On peut démontrer qu'il y a au plus $2|m|$ transitions "arrières", i.e. qui ne suivent pas le motif vers l'avant du dessin, mais ne reviennent pas non plus vers ε. Ce sont les seules transitions vraiment intéressantes ! Décrire l'automate élagué à ces transitions prend donc (avec listes d'adjacence) un espace $\mathcal{O}(|m|)$. On peut effectivement le construire en temps $\mathcal{O}(|m|)$ en utilisant Aho-Corasick. Attention, à cause des listes, la lecture d'un caractère se fait (au pire) en $\mathcal{O}(|\Sigma|)$ maintenant ! Néanmoins, on peut montrer que cet automate fait toujours *moins de comparaisons* que KMP.

1. En ne construisant que les états accessibles.

BG Correction de l'algorithme de Dijkstra.

Leçons possibles : 925 927
Pas de référence.

Elegance : ★★★★☆

Avis.

Un grand classique dans les graphes et les preuves d'algorithmes. Il faut être bien clair sur les trois invariants[1], sinon c'est délicat de s'en sortir. [CLRS10] fait des trucs bizarres.

Prérequis.

1. Dijkstra sur $G = (S, A)$ pondéré par $w : A \to \mathbb{R}_+$.

Algorithme 5 : L'algorithme de Dijkstra

Fonction *Dijsktra*(S, A, w, s) *:*
  $d \leftarrow$ tableau indexé par S initialisé à $+\infty$
  $d[s] \leftarrow 0$
  $F \leftarrow$ InsererNouvelleFile(S, d) `#objets sommets de priorités dans` d
  $E \leftarrow \varnothing$
  tant que $F \neq \varnothing$ **faire**
    $u \leftarrow$ ExtraireMin(F)
    $E \leftarrow E \uplus \{u\}$
    pour $(u, v) \in A$ **faire**
      $d[v] \leftarrow \min(d[v], d[u] + w(u, v))$
      DiminuerPriorite(v,F)
    fin
  fin
  retourner *d*
fin

Développement.

On rappelle les notations[2] :
— $G = (S, A)$ un graphe orienté ;
— $w : A \to \mathbb{R}_+$ fonction de pondération positive ;
— s sommet de départ ;
— E est l'ensemble des sommets déjà vus ;
— F est la file des sommets qui restent à traiter.

Si $Z \subseteq W$, on note $\delta_Z(s, v)$ l'infimum des poids des chemins de s à v dont l'avant-dernier (s'il existe) sommet est dans Z.

Théorème BG.1. *L'algorithme de Dijkstra répond correctement au problème des plus courts chemins à origine unique : à la fin on a* $d[v] = \delta(s, v)$ *pour tout* $v \in S$.

Preuve.

Proposition BG.2. *Les assertions suivantes sont des invariants pour* tant que *:*

1. $F \uplus E = S$;

1. C'est sans doute le nombre minimal.
2. Idéalement, l'algorithme est dans le plan/annexes et on rappelle rapidement les notations au tableau.

2. $\forall v \in E$, $d[v] = \delta(s, v)$;
3. $\forall v \in F$, $d[v] = \delta_E(s, v)$.

Preuve. **Initialisation.** Avant l'entrée dans la boucle.

1. $E = \varnothing$ et $F = S$.
2. vrai par vacuité ($E = \varnothing$).
3. si $v = s$, alors $\delta_\varnothing(s, s) = 0$; si $v \neq s$, $\delta_\varnothing(s, v) = +\infty$ [1].

Hérédité Supposons que $1, 2$ et 3 sont vrais. On fait un autre tour de boucle, où l'on suppose que $s \in E$ déjà traité et donc $u \neq s$ [2].

1. $F \to F \setminus \{u\}$ et $E = E \uplus \{u\}$ donc c'est préservé.
2. Au début de la boucle, on a $d[u] = \delta_E(s, u) \geq \delta(s, u)$. Supposons que $d[u] > \delta(s, u)$, alors il existe un chemin allant de s à u. Puisque $s \in E$ et $u \notin E$, il existe $y \neq s$ le premier sommet sortant de E dans ce chemin. Alors $d[y] = \delta_E(s, y) \leq \delta(s, u) < d[u]$ (par positivité).
   Donc y aurait dû être dépilé avant u, absurde. Donc $d[u] = \delta(s, u)$.
   Ensuite, on remarque que les valeurs $d[v]$ pour $v \in E$ ne sont pas modifiées par la boucle "pour" (puisqu'elles sont déjà optimales).
3. Il suffit de noter que $\delta_{E \uplus \{u\}}(s, v) = \min(\delta_E(s, v), \delta(s, u) + w(u, v))$ (on sépare les chemins selon que l'avant-dernier sommet est u ou non).

□

Enfin, l'algorithme termine puisque $|F|$ décroit d'une unité à chaque boucle. A la fin on a $E = S$ et les distances sont donc celles que l'on veut.

□

Postrequis.

1. Variante : on pourrait être tenté de remplacer la définition de δ_E par "l'infimum sur les chemins qui sont restent E sauf au dernier sommet". Cela permet de prouver la conservation de 2 sans problème, mais la conservation de 3 est plus lourde. En effet, on ne peut plus se contenter d'une disjonction sur l'avant-dernier sommet).
2. Etude de la complexité. On fait $\mathcal{O}(|S|)$ enfilements au départ et $\mathcal{O}(|A|)$ diminutions de poids. L'implémentation de la file avec un tas binaire donne du $\mathcal{O}((|S| + |A|)\log(|S|))$. Avec un tas de Fibonacci (inventé historiquement pour ce problème, et qui optimise subtilement les diminutions de poids en $\mathcal{O}(1)$ amorti) on a $\mathcal{O}(|A| + |S|\log(|S|))$.
3. Et si les poids sont négatifs ? Bellman-Ford en $\mathcal{O}(|S||A|)$ permet de détecter la présence d'un éventuel cycle de poids négatif.
4. Si on veut aller exactement entre deux sommets et que l'on dispose d'une sous-estimation de la distance (en pratique le "vol d'oiseau") ?
   On peut faire A^* qui peut éviter d'explorer tout le graphe !
5. Aspects pratiques.
   Comment gérer les infinis ? Soit faire un type spécial (lourd), soit mettre la somme de tous les poids des arêtes plus 1.
   Et les débordements ? Si les entiers sont trop grands, les calculs seront fait modulo la puissance de 2 utilisée pour les représenter en mémoire. Cela limite les valeurs possibles pour une vraie exécution de l'algorithme.

1. Il n'existe aucun chemin vérifiant les conditions.
2. le cas = s arrive au début, il se traite similairement

BH Grammaire $\mathrm{LL}(1)$ et table d'analyse.

Leçons possibles : 923
Pas de référence.

Elegance : ★☆☆☆☆

Avis.

Un peu pénible, mais assez attendu dans la 923. En outre, il démontre que le candidat sait manipuler l'analyse syntaxique, à condition de bien comprendre les algorithmes de First et Follow.

Prérequis.

1. First et Follow.
   Soit G une grammaire algébrique de non-terminaux $\mathcal{N}$, d'axiome S et de terminaux Σ.

   Définition BH.1. *Pour $w \in \Sigma^*$, soit $\mathrm{First}_k(w) := w$ si $|w| < k$; $w[1]\dots w[k]$ sinon. Pour $\alpha \in (\mathcal{N} \cup \Sigma)^*$, on définit $\mathrm{First}_k(\alpha) := \{\mathrm{First}_k(w) \mid w \in \mathcal{L}_G(\alpha)\}$.*

   On peut calculer $\mathrm{First}_k(X)$ pour $X \in \mathcal{N}$ par saturation (c'est polynomial).

   Définition BH.2. *La grammaire G est dite $\mathrm{LL}(k)$ si pour tout calcul $S \to^w X\delta$ de l'automate des expansions-vérifications avec $X \in \mathcal{N}$, et pour toutes règles distinctes $X \to \alpha$ et $X \to \beta$, on a : $\mathrm{First}_k(\alpha\delta) \cap \mathrm{First}_k(\alpha\delta) = \varnothing$.*

   Mais cette définition utilise les *calculs possibles* de l'automate, ce qui n'est pas clairement calculable. Pour faciliter le travail, on sur-approxime $\mathrm{First}_k(\alpha\delta)$ par $\mathrm{Follow}_k(\alpha\delta)$.

Développement.

1. Soit la grammaire G_0 d'axiome E et de règles [1]
   — $E \to T + E | T$
   — $T \to F * T | F$
   — $T \to (E) | a$
   On ne peut pas lever le non-déterminisme en lisant k lettres. En effet, si E est sur la pile, on ne peut pas choisir entre les mots $E \to_g E{+}T \to_g^* a{*}\cdots{*}a{+}T$ et $E \to_g T \to_g^* a{*}\cdots{*}a$.
2. On construit donc une nouvelle grammaire G d'axiome E et de règles [2] :
   — $E \to TE'$
   — $E' \to +TE' | \varepsilon$
   — $T \to FT'$
   — $T' \to *FT' | \varepsilon$
   — $F \to (E) | a$
   Elle reconnaît le même langage que la précédente. En effet montrerait par récurrence que si $E \to_d^* T + \cdots + T + E$ dans G_0 alors $E \to_d^* T + \cdots + TE'$ dans G, et ainsi de suite.
3. Montrons que G est LL(1) et calculons la 1-table d'analyse associée.
   (a) Calcul de First_1 par saturation.

	E	E'	T	T'	F
1		ε		ε	$(, a$
2		ε	$(, a$	$\varepsilon, *$	$(, a$
3	$(, a$	$\varepsilon, +$	$(, a$	$\varepsilon, *$	$(, a$
4	$(, a$	$\varepsilon, +$	$(, a$	$\varepsilon, *$	$(, a$

1. On remarquera qu'elle engendre les expressions "parenthésées au minimum" sur $+, *$. Et alors E sont les expressions, T les termes de sommes, F les facteurs de produits.
2. L'intuition est de repousser à plus tard le choix de si un terme est le dernier.

(b) Calcul de Follow_1 par saturation.

	E	E′	T	T′	F
1	ε				
2	ε	ε	$\varepsilon, +$		
3	ε	ε	$\varepsilon, +$	$\varepsilon, +$	$\varepsilon, *$
4	$\varepsilon,)$	ε	$\varepsilon, +$	$\varepsilon, +$	$\varepsilon, *, +$
8	$\varepsilon,)$	$\varepsilon,)$	$\varepsilon, +,)$	$\varepsilon, +,)$	$\varepsilon, *, +,)$

(c) Table d'analyse.

	E	E′	T	T′	F
a	TE'		FT'		a
(	TE'		FT'		(E)
)		ε		ε	
+		$+TE'$		ε	
*				$*FT'$	
ε		ε		ε	

BI Décidabilité de l'arithmétique de Presburger.

Leçons possibles : 909 914 924
Adapté depuis : [Car08], p 178

Elegance : ★★★★☆

Avis.

Une jolie application de la *notion* d'automate pour exhiber une procédure de décision non-triviale. Pour montrer la décidabilité d'une théorie, on montre que l'on peut *décrire par un automate* les valuations qui satisfont chaque formule (construction inductive).
La preuve étant interminable si on tente de tout détailler, il faudra rester autant que possible à "haut niveau" et parler davantage de propriétés de clôture que de constructions d'automates. Attention cependant, soyez prêt à répondre au jury s'il demande effectivement de construire l'automate associé à une formule (simple, je l'espère).

Prérequis.

1. Automates. Propriétés de clôture usuelles et effectivité.
2. Logique du premier ordre. Avoir une idée de l'arithmétique de Presburger.
3. Codage (à mettre dans le plan pour alléger le développement).
   Si $n \in \mathbb{N}$ notons $\mathrm{bin}(n) \in \{0,1\}^*$ l'*unique* représentation binaire de n avec bit de poids fort à droite qui termine par un symbole 1[1]. Si $n_1 \dots n_m \in \mathbb{N}$, on désigne par $[n_1, \dots n_m]$ un codage de $\mathrm{bin}(n_1) \dots \mathrm{bin}(n_m)$ sur l'alphabet $\{0,1\}^m$, en rajoutant des 0 dans les trous[2]. Ce codage est *unique* étant donné un tuple.

Développement.

Théorème BI.1 (Presburger). *Le problème suivant est décidable.*
Entrée : φ *formule close du premier ordre sur la signature* $+, 1, =$.
Question : Est-ce que $\langle \mathbb{N}, +, = \rangle \models \varphi$ *?*[3]

Preuve.

Exemple BI.2. $[3,8] = \begin{pmatrix}1\\0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix}\begin{pmatrix}0\\0\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}$.

Le langage $\mathrm{VAL}_m := \{[n_1, \dots, n_m] \mid n_i \geq 0\}$ est en bijection avec l'ensemble des valuations dans $\mathbb{N}$ sur m variables. On peut construire un automate pour VAL_m.
Si $\varphi(x_1 \dots x_m)$ est une formule[4], on note $\mathcal{L}_m(\varphi) := \{[n_1 \dots n_m] \mid \mathcal{M} \models \varphi(n_1 \dots n_m)\}$.

Lemme BI.3. *Si* $\varphi(x_1 \dots x_m)$ *est plate*[5] *et sans quantificateurs, on peut construire un automate reconnaissant* $\mathcal{L}_m(\varphi)$.

Preuve. On procède par induction structurelle sur φ.
— Si $\varphi = \top$, alors $\mathcal{L}_m(\varphi) = \mathrm{VAL}_m$ et c'est bon ;
— Si $\varphi = (x_i = x_j)$, alors $\mathcal{L}_m(\varphi) = \{0^m + 1^m\}^* \cap \mathrm{VAL}_m$;

1. On posera aussi - plus ou moins implicitement - $\mathrm{bin}(0) = \varepsilon$
2. Si l'on souhaite s'engager dans une définition formelle, on peut s'en sortir comme suit.
   Si $n_1, n_2 \geq 0$, on définit $[n_1, n_2] \in (\{0,1\}^2)^*$ comme étant le mot de longueur $\max(|\mathrm{bin}(n_1)|, |\mathrm{bin}(n_2)|)$ vérifiant $[n_1,n_2][i] = (\mathrm{bin}(n_1)[i], \mathrm{bin}(n_2)[i])$ si $i < |w_1|$ et $i < |w_2|$; $[n_1,n_2][i] = (\mathrm{bin}(n_1)[i], 0)$ si $|w_2| \leq i < |w_1|$; $[n_1,n_2][i] = (0, \mathrm{bin}(n_2)[i]))$ si $|w_1| \leq i < |w_2|$.
   Ce mot code juste l'écriture de $\mathrm{bin}(n_1)$ et $\mathrm{bin}(n_2)$ l'un en-dessous de l'autre, en bouchant les trous par des 0. La définition se généralise à des m-tuples de mots.
3. dans $\mathcal{M}$ les symboles $+, =$ sont interprétés comme on s'y attend.
4. On suppose dans cette écriture que $\mathrm{VL}(\varphi) \subseteq \{x_1 \dots x_m\}$, et que φ est sur la signature qui va bien.
5. Cela signifie que ses sous-termes sont de la forme $x + y = z$, $x = y$ ou $x = 1$ et rien d'autre.

— Si $\varphi = (x_i + x_j = x_k)$.
On suppose pour simplifier $i = 1, j = 2, k = 3$ et $n = 3$. Il suffit d'intersecter le langage de l'automate de la Figure 2.11 avec VAL_3.

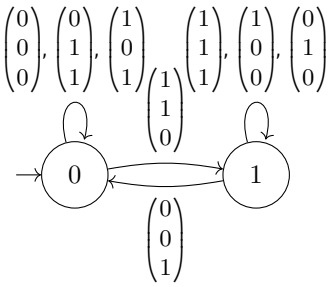


Figure 2.11 – Automate de l'addition.

— Si $\varphi = \varphi_1 \wedge \varphi_2$, alors $\mathcal{L}_m(\varphi) = \mathcal{L}_m(\varphi_1) \cap \mathcal{L}_m(\varphi_2)$;
— Si $\varphi = \neg\varphi_1$, alors $\mathcal{L}_m(\varphi) = \mathrm{VAL}_m \setminus \mathcal{L}_m(\varphi_1)$.

□

Lemme BI.4. *A partir d'un automate pour $\mathcal{L}_m(\varphi(x_1 \dots x_m))$, on peut construire un automate reconnaissant $\mathcal{L}_{m-1}(Qx_m\varphi(x_1 \dots x_m))$ pour $Q \in \{\exists, \forall\}$.*

Preuve. Deux cas se présentent :
— si $Q = \exists$ alors $\mathcal{L}_{m-1}(\exists x_m \varphi) = \{[n_1, \dots, n_{m-1}] \mid \exists n_m\ [n_1 \dots n_m] \in \mathcal{L}_m(\varphi)\}$
$= \pi_{m-1}(\mathcal{L}_m(\varphi))C^{-1} \cap \mathrm{VAL}_{m-1}$ où
  — π_{m-1} la projection sur les $m-1$ premières coordonnées ;
  — $C = (\{0\}^{m-1})^*$ sert à enlever les zéros en trop ;
— si $Q = \forall$ alors $Q = \neg\exists\neg$ ce qui peut être traité par ce que l'on a fait avant.

□

On en déduit la procédure de décision sur une formule φ formule close :

1. Mettre en φ en forme plate et prénexe, i.e. sous la forme $Q_1x_1 \dots Q_mx_m\psi(x_1 \dots x_m)$ où ψ est plate et sans quantificateurs.
2. Utiliser la construction du Lemme BI.3 sur ψ.
3. Itérer la construction du Lemme BI.4 pour éliminer un à un les quantificateurs.
4. Tester si le langage final est vide, auquel cas $\mathcal{M} \not\models \varphi$; sinon $\mathcal{M} \models \varphi$ [1].

□

Postrequis.

1. Pertinence du résultat.
   Dans la théorie donnée, on peut définir 0 (c'est l'élément neutre de l'addition), $x \leq y$ ($\exists z$, $x + z = y$) et enfin le successeur et 1 (en utilisant $\leq$). On peut donc faire les opérations "usuelles" avec $+$.
2. Etude de la complexité.
   (a) La mise en forme plate prénexe est polynomiale. Ensuite, la construction de l'étape 2 est exponentielle en la taille de la formule (si on a descendu les négations à la base). L'étape 3 crée au pire une tour d'exponentielles de taille le nombre de complémentations requises, soit $2 \times \mathrm{nb}(\forall)$. Plus intelligemment, on peut remarquer qu'il suffit d'en faire lors d'une alternation de quantificateurs.

1. Lors de la dernière application du Lemme BI.4, on projette sur le langage à un seul élément tuple vide (). Cela correspond aux valuations sur zéro variables, et on regarde juste si la valuation vide satisfait ou non la formule.

(b) Mais on peut faire beaucoup mieux, une borne triplement exponentielle seulement, voir comme son nom l'indique l'article [Opp78].

3. D'autres théories décidables, ou pas.

(a) La théorie de $\langle \mathbb{N}, +\times, = \rangle$ (Church-Gödel) est indécidable, et même pas récursivement énumérable. Elle n'est même pas dans la hiérarchie arithmétique.

(b) On peut faire un peu la même chose pour décider la théorie de $\langle \mathbb{N}, \times, = \rangle$ (Skolem), avec des automates d'arbres et des décompositions en facteurs premiers (assez joli).

(c) De manière plus générale, utiliser des automates est une méthode assez générique pour obtenir la décidabilité de la théorie du premier ordre d'une structure. Voir la notion de *structure automatique* et ses dérivés (c'est un thème de recherche actuel, comme en témoigne - sans prétention - [DT18]).

BJ Machines de Turing et langages rationnels.

Leçons possibles : 909 913
Adapté depuis : [Car08], p 189

Elegance : ★★★★★

Avis.

Un résultat assez surprenant, puisqu'il montre que "toute" l'expressivité d'une machine de Turing réside dans sa capacité à écrire sur son entrée ! Enfin, c'est un exemple assez rare d'équivalence entre modèles de calculs qui ne soit pas totalement effective.

Prérequis.

Proposition BJ.1 (caractérisation algébrique). *Un langage $L \subseteq \Sigma^*$ est régulier ssi il existe une congruence du monoïde Σ^* d'indice fini qui sature L.*

Développement.

Théorème BJ.2. *Une machine de Turing à une bande qui n'écrit pas sur son entrée accepte un langage rationnel.*

Preuve. Soit $\mathcal{M} = (Q, q_0, \delta, \Sigma, \Gamma, q_f)$ une telle machine reconnaissant un langage L. On supposera qu'elle retourne au début de la bande (sur \$) pour accepter.
On définit pour $w \in \Sigma_+$ [1] :
— $\lambda^{\rightarrow}(w) := \{(q, q') \in Q^2 \mid$ il existe un calcul de $\mathcal{M}$ dans w, partant à gauche dans l'état q et finissant par sortir à droite dans $q'\}$;
— $\lambda^{\leftarrow}(w)$, $\lambda^{\curvearrowright}(w)$, $\lambda^{\curvearrowleft}(w) := \{(q, q') \in Q^2 \mid \}$ de manière similaire (faire des dessins).
Soit $\sim$ la relation sur Σ^* telle que $\varepsilon \sim \varepsilon$ et pour $w, w' \in \Sigma_+$, $w \sim w'$ ssi $\lambda^X(w) = \lambda^X(w')$ pour tout $X \in \{\rightarrow, \leftarrow, \curvearrowright, \curvearrowleft\}$ [2].

Lemme BJ.3. *$\sim$ est une congruence [3] d'indice fini sur Σ^*.*

Preuve. C'est clairement une relation d'équivalence d'indice $\leq 2^{4|Q|^2} + 1$ [4].
Pour la congruence, montrons que si $u \sim v$ et $u' \sim v'$ alors $uu' \sim vv'$. On suppose les mots différents de ε et on traite un cas en faisant un dessin.

□

Lemme BJ.4. *$\sim$ sature L.*

Preuve. Supposons $w \sim w'$ et $w \in L$, montrons $w' \in L$.
Si $w = w' = \varepsilon$ c'est évident. Sinon on fait encore un dessin.

□

On a donc une congruence d'indice fini qui sature L, par BJ.1 ce langage est régulier. □

Postrequis.

1. Ce résultat est outrancièrement faux pour les machines à plusieurs bandes, puisqu'il suffirait de recopier l'entrée sur une bande de travail et de faire ce que l'on veut avec.
2. Inversement, une machine à une bande qui n'écrit pas sur son entrée ne peut donc pas la recopier ailleurs sur sa bande [5] (sans quoi elle pourrait travailler avec).

1. Cela n'aurait pas clairement de sens pour $w = \varepsilon$.
2. Intuitivement, w et w' sont indiscernables par $\mathcal{M}$.
3. De monoïde.
4. $2^{|Q|^2} \times 2^{|Q|^2} \times 2^{|Q|^2} \times 2^{|Q|^2} + 1$ (pour ε).
5. Essayer de s'en convaincre est troublant, ça ne marche pas, ahhh si, ahhh non en fait.

3. Questions d'effectivité. Peut-on construire un automate équivalent ?

   En fait, presque... L'ensemble des états est constructible (indicé par $\{\varepsilon\} \uplus \mathcal{P}((Q \times Q)^4)$) et on peut calculer la fonction de transition. En effet, il suffit de déterminer comment on peut "prolonger" les calculs quand on ajoute une lettre à une classe.

   Mais alors ? C'est gagné ? Non ! On n'a aucune chance de pouvoir déterminer quels états sont finaux, sans quoi on pourrait décider le problème du mot, et ce n'est pas possible même pour ces machines particulières (voir BJ.5 ci-dessous).

   Proposition BJ.5. *Savoir si une machine de Turing à une bande qui n'écrit par sur son entrée accepte un mot donné est indécidable*[1].

   > *Preuve.* On réduit le problème de l'arrêt d'une machine $\mathcal{M}$ sur w. La machine construite simule $\mathcal{M}$ sur w en utilisant la bande après son entrée, quelle que soit celle-ci, et accepte selon le résultat de la simulation. □

   Ce qui illustre également l'importance de la représentation en entrée pour que les problèmes usuels sur les langages rationnels soient décidables. Avec une représentation par automate, tout se passe bien. Avec une machine de Turing, c'est mal.

4. Un cas particulier intéressant : les *automates boustrophédons*. On va essaye de sauver les meubles pour l'effectivité en restreignant le modèle de départ.

   Un boustrophédon est un automate fini qui peut lire son entrée en déplaçant une tête de lecture à gauche ou à droite. En d'autre termes, c'est une machine de Turing qui n'écrit pas et qui ne sort pas de son entrée (on peut placer des symboles $\vdash$ et $\dashv$ pour d'aider à repérer les bords). Il reconnaît donc, et quelque part *a fortiori*, un langage rationnel.

   Dans le Développement, les classes de congruence acceptantes étaient celles qui avaient un calcul acceptant. On ne pouvait pas les déterminer, parce que le calcul pouvait *sortir* de l'entrée, sans qu'on sache ce qu'il s'y passe. Désormais on peut *calculer* les classes acceptantes, et donc les états finaux d'un automate fini équivalent au Boustrophédon.

   A noter qu'un Boustrophédon déterministe peut être exponentiellement plus succinct qu'un automate fini déterministe. Il y a des problèmes ouverts dans les questions de *tradeoff* entre Boustrophédons, alternants et non-déterministes.

5. On peut imposer *syntaxiquement* aux machines de ne pas modifier leur entrée (par exemple en utilisant deux alphabets disjoints d'entrée et de travail). En revanche :

   Proposition BJ.6. *Etant donné un mot $v \neq \varepsilon$ fixé*[2]*, savoir si une machine de Turing à une bande écrit sur w lorsque c'est son entrée est indécidable.*

   > *Preuve.* On reprend la réduction de la Proposition BJ.5. La machine construite simule $\mathcal{M}$ sur w en utilisant la bande après son entrée si c'est v. Si la simulation accepte, la machine part écrire un coup sur son entrée v.
   >
   > □

6. Quid des machines de Turing à une bande qui n'écrivent *que* sur l'entrée ?[3] Elles reconnaissent des langages de type 1 dans la hiérarchie de Chomsky, aussi engendrés par les grammaires contextuelles.

   Exemple BJ.7. *Le langage $\{a^n b^n c^n \mid n \geq 0\}$ n'est pas de type 2 (pas algébrique) mais néanmoins de type 1. Donner une machine de Turing pour ce langage est d'ailleurs un exercice pertinent (s'il en faut) dans la leçon 913.*

   On peut décider l'arrêt d'un automate linéairement borné, mais pas son arrêt sur toute entrée (voir [Car08], p 174).

1. Si tant est que ce problème en soit vraiment un. En vertu de la Proposition BJ.6 le langage d'entrée est lui-même indécidable, ce que l'on évite parfois dans la définition d'un problème. Enfin...
2. Ce n'est pas une donnée du problème.
3. Parfois appelées automates linéairement bornés.

BK Théorèmes de hiérarchie en espace et en temps.

Leçons possibles : 913 915
Pas de référence.

Elegance : ★★★☆☆

Avis.

L'idée est encore de faire un argument diagonal. Seulement, la construction est un peu plus fine, à cause de constantes dont il faut clarifier les dépendances. Le contexte n'est pas évident non plus : il faut être clair sur les simulations de machines (ce qui est assez désagréable). Le résultat annoncé p 233 de [Car08] pour la hiérarchie en temps est très douteux, et la preuve proposée est incomplète (on ne voit pas où l'hypothèse intervient).

Prérequis.

1. Fonctions propres.

   Définition BK.1. *Une fonction f est dite propre en temps si $f(n) \geq n$ et s'il existe une machine de Turing qui retourne $f(n)$ sur l'entrée 1^n en temps $\mathcal{O}(f(n))$.*

   Définition BK.2. *Une fonction f est dite propre en espace si $f(n) \geq \log(n)$ et s'il existe une machine de Turing qui retourne $f(n)$ sur l'entrée 1^n en espace $\mathcal{O}(f(n))$.*

   Les fonctions usuelles sont propres. Attention, on ne définira les classes de complexité DTIME, NTIME, DSPACE, NSPACE que pour de telles fonctions.

2. Simulation.

   Théorème BK.3 ([AB09], Th. 1.9 p 23)**.** *Il existe un codage des machines de Turing $\mathcal{M} \mapsto \overline{\mathcal{M}}$ et une machine universelle $\mathcal{U}$ qui sur l'entrée $\overline{\mathcal{M}}, x$ simule $\mathcal{M}$ sur x. Si T est le temps de calcul de $\mathcal{M}$ sur x et E l'espace utilisé, la simulation se fait en temps $\alpha_M T^2$ et espace $\beta_M E$ où α_M, β_M sont des constantes dépendant de $\mathcal{M}$ mais pas de x.*

   Schéma de preuve. La machine $\mathcal{U}$ aura deux bandes de travail.

   Bande 1 : bande d'état. Pour maintenir l'état courant de $\mathcal{M}$. Connaître l'état suivant se fait en temps constant à $\mathcal{M}$ fixé (en parcourant $\overline{\mathcal{M}}$ sur l'entrée).

   Bande 2 : simulation des bandes de travail. Pour simuler k bandes en une seule, on utilise des k-uplets de lettres. Il faut aussi ajouter des marqueurs pour les positions des têtes de lecture. Cela dit, la simulation d'une étape de calcul prend un temps $\mathcal{O}(T)$, car il faut peut-être parcourir toute la bande [1]. Ensuite, on encode ce nouvel alphabet (dépendant de $\mathcal{M}$) dans l'alphabet fixé de $\mathcal{U}$, ce qui augmente les complexités d'un produit constant.

   On trouve bien une simulation en temps $\mathcal{O}(T^2)$ et espace $\mathcal{O}(E)$.

   □

   Remarque BK.4. *En fait, une construction plus fine permet d'obtenir une borne $\mathcal{O}(T\log(T))$. Cela permet à terme d'affaiblir les hypothèses du théorème de hiérarchie en temps.*

Développement.

Théorème BK.5. *Si f, g sont propres en temps et $f^2 = \circ(g)$, alors* DTIME$(f) \subsetneq$ DTIME(g).

Preuve. Définissons une machine $\mathcal{V}$ qui sur l'entrée $\overline{\mathcal{M}}, x$ de taille n fait :

1. Initialiser un compteur à la valeur $g(n)$.

1. Il faut modifier les k têtes, qui peuvent être très loin.

2. Exécuter $\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x])$ en décrémentant le compteur à chaque pas.
3. Si $\mathcal{U}$ n'a pas terminé son calcul avant la fin, rejeter. Sinon, renvoyer $\neg\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x])$.

Il vient que $\mathcal{V}$ calcule en temps $\mathcal{O}(g)$ [1]. Soit $L \in \mathsf{DTIME}(g)$ le langage reconnu.
Supposons que L est reconnu par une machine $\mathcal{M}$ qui travaille en $\leq Cf$. Alors sa simulation par $\mathcal{U}$ prend $\leq \alpha_M C^2 f^2$. Or comme $f^2 = \circ(g)$, à partir d'un certain rang $g(n) \geq \alpha_M C^2 f(n)^2$ et les simulations terminent dans $\mathcal{V}$.
Donc pour x assez long, $\mathcal{V}(\overline{\mathcal{M}}, x) = \neg\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x]) = \neg\mathcal{M}(\overline{\mathcal{M}}, x)$. Or puisqu'ils reconnaissent L, $\mathcal{M}(\overline{\mathcal{M}}, x) = \mathcal{V}(\overline{\mathcal{M}}, x)$. C'est absurde.

□

Théorème BK.6. *Si f, g sont propres en espace et $f = \circ(g)$, alors* $\mathsf{DSPACE}(f) \subsetneq \mathsf{DSPACE}(g)$.

Preuve. Définissons une machine $\mathcal{V}$ qui sur l'entrée $\overline{\mathcal{M}}, x$ de taille n fait :

1. Marquer les bandes à la case $g(n)$.
2. Exécuter $\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x])$ [2] (la machine universelle construite en BK.3) en vérifiant qu'on ne sort pas après les marques.
3. Si $\mathcal{U}$ sort, rejeter. Sinon, renvoyer $\neg\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x])$.

Il vient que $\mathcal{V}$ calcule en espace $\mathcal{O}(g)$ [3]. Soit $L \in \mathsf{DSPACE}(g)$ le langage reconnu.
Supposons que L est reconnu par une machine $\mathcal{M}$ qui travaille en espace $\leq Cf$. Sans perte de généralité, on peut supposer que $\mathcal{M}$ termine sur toute entrée [4]. Alors sa simulation par $\mathcal{U}$ prend un espace $\leq \beta_M Cf$. Or comme $f = \circ(g)$, à partir d'un certain rang $g(n) \geq \beta_M Cf(n)$ et les simulations terminent dans $\mathcal{V}$.
Donc pour x assez long, $\mathcal{V}(\overline{\mathcal{M}}, x) = \neg\mathcal{U}(\overline{\mathcal{M}}, [\overline{\mathcal{M}}, x]) = \neg\mathcal{M}(\overline{\mathcal{M}}, x)$. Or puisqu'elles reconnaissent toutes deux L, $\mathcal{M}(\overline{\mathcal{M}}, x) = \mathcal{V}(\overline{\mathcal{M}}, x)$. C'est absurde.

□

Postrequis.

1. Théorème de hiérarchie en espace non-déterministe.

   Théorème BK.7. *Si f, g sont propres en espace et $f = \circ(g)$, alors* $\mathsf{NSPACE}(f) \subsetneq \mathsf{NSPACE}(g)$.

   Schéma de preuve. On suit la même idée que pour BK.6. La simulation se fait de manière non-déterministe, mais on ne peut plus inverser les réponses ! Alors que faire ? On n'inverse pas ! Si le calcul termine, on répond $\mathcal{U}(\mathcal{M}, [\langle\mathcal{M}\rangle, x])$.
   Maintenant, il reste à "extérioriser" la négation.
   Supposons ensuite que $L \in \mathsf{NSPACE}(f)$. Alors en vertu du théorème d'Immerman-Szelepcsényi, son complémentaire $\overline{L}$ est aussi accepté par une machine $\mathcal{M}$ en espace non-déterministe $\mathcal{O}(f)$. On a donc $\mathcal{M}(\langle\mathcal{M}\rangle, x) = \neg\mathcal{V}(\langle\mathcal{M}\rangle, x)$.
   D'autre part pour x assez long (et toujours pour les mêmes raisons), $\mathcal{V}(\langle\mathcal{M}\rangle, x) = \mathcal{U}(\langle\mathcal{M}\rangle, [\langle\mathcal{M}\rangle, x]) = \mathcal{M}(\langle\mathcal{M}\rangle, x)$.

   □

2. Théorème de hiérarchie en temps non-déterministe.

   Théorème BK.8 ([AB09], Th. 3.2 p 62)**.** *Si f, g sont propres en temps et $f(n+1) = \circ(g(n))$, alors* $\mathsf{NTIME}(f) \subsetneq \mathsf{NTIME}(g)$.

   Non, le $f(n+1)$ n'est pas une typo, mais il témoigne de la subtilité de la preuve. On utilise toujours une méthode de diagonalisation, mais explorer toutes les branches n'est pas possible pour renvoyer la négation. La technique s'appelle *diagonalisation lente*.

3. Des conséquences sur les classes usuelles

1. Pour 1., l'argument est que g est propre en temps.
2. L'argument essentiel est de simuler $\mathcal{M}$ sur elle-même. Le x n'est là que pour "ajouter de la longueur".
3. Pour 1., l'argument est que g est propre en espace.
4. Quitte à ajouter un compteur de taille approximativement $f(n)2^{f(n)}$, donc codé sur $f(n)$ bits (prendre le log).

Application BK.9. $\mathrm{L} \subsetneq$ PSPACE.

Application BK.10. NL $\subsetneq$ PSPACE

Preuve. On peut soit utiliser la hiérarchie en espace non-déterministe pour obtenir NL $\subsetneq$ NPSPACE puis Savitch avec NPSPACE = PSPACE, soit utiliser Savitch pour NL $\subseteq$ DSPACE($\log^2$) et utiliser la hiérarchie en espace déterministe. □

Application BK.11. P $\subsetneq$ EXP

Application BK.12. NP $\subsetneq$ NEXP

BL Théorème de Berman (langage unaires).

Leçons possibles : 906 915 928
Pas de référence.

Elegance : ★★★★★

Avis.

C'est finalement beaucoup plus un développement de programmation dynamique que de NP-complétude ou de complexité. Enfin, on peut dire qu'on "manipule" des réductions polynomiales et que c'est bon pour la santé.

Prérequis.

1. Notion de réduction, de réduction polynomiale.
   La taille de la sortie ne peut jamais excéder le temps de calcul.
2. Problème SAT.
   Une idée de comment on peut l'encoder.

Développement.

Théorème BL.1. *S'il existe un langage* NP*-dur unaire, alors* P = NP.

Preuve. On note I l'ensemble des instances du problème SAT dans un codage fixé [1]. On suppose que pour toute φ formule φ, $|\varphi(x/0)| \leq |\varphi|$ et $|\varphi(x/1)| \leq |\varphi|$ [2].
Soit $L \subseteq 0^*$ un langage NP-dur, alors il existe une réduction $f : I \to 0^*$ calculable en temps polynomial et telle que $f(\varphi) \in L$ si et seulement si φ est satisfiable. Notons P un majorant polynomial du temps de calcul que l'on peut supposer *croissant*, alors $|f(\varphi)| \leq P(|\varphi|)$.

L'idée est de résoudre la satisfiabilité de φ avec de la programmation dynamique. On va écrire l'Algorithme 6, inspiré de la Figure 2.12 et du gain de temps par mémoïzation.

(a) Termes de la suite de Fibonacci.

(b) Satisfiabilité d'une formule propositionnelle.

Figure 2.12 – Arbres d'appels récursifs (sans mémoïzer).

On justifie que :
— l'algorithme termine sur toute entrée sans erreur (décroissance des codages) ;
— l'algorithme est correct. On montrerait par induction que $\mathrm{SatRec}(\varphi)$ répond correctement au problème de satisfiabilité et préserve l'invariant que si $T[m] \neq -1$, alors la satisfiabilité des formules d'image 1^m est décrite dans $T[m]$.

1. On ne fera pas de différence de notation entre une formule et son codage.
2. Un tel codage existe et donne bien la NP-complétude.

Algorithme 6 : Satisfiabilité d'une formule

```
Fonction Satisfiable(φ) :
    T ← tableau de −1 de taille P(|φ|)
    Fonction SATrec(ψ) :
        0^n ← f(ψ)
        si T[n] = −1 alors
            si ψ sans variables alors
                T[n] ← évaluation de ψ
            sinon
                T[n] ← max(SATrec(ψ(x/0)), SATrec(ψ(x/1))).
            fin
            retourner T[n]
        fin
        retourner SATrec(φ)
    fin
fin
```

— le calcul se fait en temps polynomial :
  — le calcul d'un passage est en $\mathcal{O}(P(|\varphi|))$;
  — l'arbre des appels récursifs peut être étiqueté par les $f(\psi)$, il y en a un nombre polynomial et deux noeuds n'ont jamais la même étiquette.

Ayant trouvé un moyen de résoudre polynomialement un problème NP-complet (SAT), on conclut que P = NP.

□

Postrequis.

1. Sans mémoïzation, c'est perdu car exponentiel.
2. Et si on faisait un bottom-up ? On ne saurait pas comment remonter car tout est déterminé par f qui n'est pas inversible vraisemblablement.
3. A priori rien n'interdit d'être NP-dur et indécidable, hein.
4. Pour des raisons de cardinalité (ou de temps de calcul), f ne peut pas être injective.
5. Si P = NP, alors il se passe des trucs incroyables comme :
   — EXP = NEXP (padding) ;
   — la hiérarchie polynomiale s'effondre (dès le départ)...
6. En fait, EXP = NEXP si et ssi tout langage unaire de NP est aussi dans P. C'est encore du padding, voir [AB09].

BM 2SAT en temps linéaire.

Leçons possibles : 916 925 928
Adapté depuis : [APT79], article

Elegance : ★★★★★

Avis.

La reference est un article, mais elle donne les idées essentielles. Le recasage est clair dans la 916 et dans la 925 (on vérifie la correction d'un algo de graphe dans la preuve), un peu moins sur 928 (mais le jury dit lui-même que l'on pourra présenter des problèmes NP-complets qui deviennent P sous des hypothèses plus fortes).

Prérequis.

1. Algorithme de Kosaraju pour le calcul des CFC. Soit $G = (S, A)$.

Algorithme 7 : L'algorithme de Kosaraju

Fonction *Kosaraju(G) :*
  $l \leftarrow$ ListeVide
  tant que S *non entièrement visité* **faire**
    Trouver s non visité.
    Parcours(G, s), en ajoutant dans l l'historique des dépilements.
  fin
  Inverser l.
  tant que S *non entièrement visité* **faire**
    Trouver le premier s de l non visité.
    Parcours($^tG, s$).
    Ecrire les sommets trouvés comme une nouvelle CFC.
  fin
fin

Théorème BM.1. *L'algorithme de Kosaraju est correct.*

Développement.

Théorème BM.2. *Le problème 2SAT est résoluble en temps linéaire (sur une machine RAM).*

Preuve. On suppose que la formule a exactement 2 littéraux par clause.
Soit $\varphi := \bigwedge_{i=1}^{n}(l_i \vee l_i')$ une instance de 2SAT.
Elle est équivalente à $\bigwedge_{i=1}^{n}(\neg l_i \to l_i') \wedge (\neg l_i' \to l_i)$ [1].

Suivons cette idée en considérant le *graphe d'implication* $G := \{S, A\}$ associé :
— $V := \{l_1, \dots, l_n, \neg l_1, \dots, \neg l_n\}$ [2] ;
— pour toute $l_i \wedge l_i'$ clause, $(\neg l_i, l_i') \in A$ et $(\neg l_i', l_i) \in A$.
Remarquons que par construction, si $l \to l'$ alors $\neg l' \to \neg l$.

Lemme BM.3. *φ est insatisfiable ssi il existe $l \in S$ tel que l et $\neg l$ soient dans la même composante fortement connexe (CFC) de G.*

Preuve. $\Leftarrow$ Supposons que l_i et $\neg l_i$ soient dans la même CFC et que $\mathcal{I} \vDash \varphi$.
Alors par construction, $\mathcal{I} \vDash l_i$ ssi $\mathcal{I} \vDash \neg l_i$ ce qui est absurde.

1. Cette écriture est volontairement redondante.
2. On identifie les littéraux identiques, de même que les doubles négations qui pourraient apparaître.

⇒ Supposons que l_i et $\neg l_i$ ne sont jamais dans la même CFC. On construit itérativement un marquage partiel $m : S \to \{0, 1\}$ tel que :

(i) si l est marqué, $\neg l$ aussi et $m(\neg l) = \neg m(l)$;

(ii) si $m(l) = 1$, pour toute $l \to l'$ on a $m(l) = 1$;

(iii) si si $m(l') = 0$, pour toute $l \to l'$ on a $m(l) = 0$ [1].

Construction itérative. Sélectionnons $\mathcal{C}$ une CFC minimale [2] parmi celles qui ne sont pas marquées, et soit $\neg\mathcal{C} := \{\neg l \mid l \in \mathcal{C}\}$. Alors :

— $\neg\mathcal{C}$ est une CFC, en effet $l \to^* l' \to^* l$ ssi $\neg l \leftarrow^* \neg l' \leftarrow^* \neg l$ [3] ;

— $\neg\mathcal{C}$ est maximale parmi les composantes non-marquées [4] ;

— $\mathcal{C}$ et $\neg\mathcal{C}$ sont disjointes [5].

On marque les sommets de $\mathcal{C}$ par 1 et ceux de $\neg\mathcal{C}$ par 0.

Préservation des invariants. (i) reste vrai par définition de $\mathcal{C}$ et $\neg\mathcal{C}$;

(ii) si $l \in \mathcal{C}$ alors $m(l) = 1$. Si $l \to l'$ deux cas se présentent. Soit l' était marqué avant, et donc $m(l') = 1$. En effet si c'était 0, alors par (iii) l aurait dû être marqué avant. Soit l' n'était pas marqué. Par minimalité de $\mathcal{C}$ on a $l' \in \mathcal{C}$ et donc $m(l') = 1$.

(iii) se vérifie symétriquement.

Résultat final. A partir de m, on déduit une affectation $\mathcal{I}$ telle que $\mathcal{I} \vDash l$ ssi $m(l) = 1$. Les invariants donnent $\mathcal{I} \vDash \varphi$.

□

On en déduit une procédure pour résoudre le problème :

1. Construire le graphe G représenté par listes d'adjacence.
2. Déterminer les composantes connexes à l'aide de l'Algorithme ?? de Kosaraju.
3. Vérifier la propriété du lemme BM.3.

□

Postrequis.

1. On peut même obtenir une affectation. En effet, l'algorithme de Kosaraju fournit un tri topologique des CFC, on peut donc les évaluer au fur et à mesure.
2. Théorème de Cook.

   Théorème BM.4. SAT *est* NP*-complet.*

   Corollaire BM.5. 3SAT *est* NP*-complet.*
3. Complétude de 2SAT

   Théorème BM.6 ([Car08])**.** 2SAT *est* NL*-complet.*

   Preuve. Montrons qu'il est dans NL. La réduction présentée en développement peut être effectuée en espace logarithmique. Elle réduit au problème de la non-accessibilité de $\neg l_i$ depuis l_i pour tout i, qui est dans $co - \mathsf{NL} = \mathsf{NL}$.
   Pour la complétude, on co-réduit le problème de l'accessibilité. Soit $G = (S, A)$ et s, t une instance. On construit une formule $\varphi := s \wedge \left(\bigwedge_{(u,v)\in E} u \to v\right) \wedge \neg t$ qui code la relation d'accessibilité. Si $\mathcal{I} \vDash \varphi$, alors $\mathcal{I}(u) = 1$ pour tout sommet accessible

1. En y regardant de plus près, cette condition est redondante, mais on la rend explicite par pédagogie.
2. Pour l'ordre induit sur les CFC par la relation d'accessibilité. Noter que grâce aux invariants (ii) et (iii), chaque CFC est marquée entièrement ou pas du tout.
3. Ce qui est équivalent à dire que l et l' sont dans la même CFC
4. Si $l \to^* l'$ avec $l' \in \neg\mathcal{C}$, $l \notin \neg\mathcal{C}$ non-marquée, alors $\neg l' \to^* \neg l$ ce qui contredit la minimalité de $\mathcal{C}$.
5. Car non confondues.

depuis s. D'autre part $\mathcal{I}(t) = 0$, donc t n'est pas accessible. Réciproquement si t n'est pas accessible, on peut trouver une interprétation qui satisfait la formule en valuant t à 0 (et les sommets accessibles à 1).

□

4. Le résultat original est un peu plus fort.
   Le problème QBF (validité d'une formule booléenne quantifiée) est connu pour être PSPACE-complet. On appelle problème $2\mathrm{QBF}$ sa restriction au cas où les formules sont des disjonction de 2-clauses, le tout en forme normale prénexe.

   Exemple BM.7. $\forall p \exists q \forall s\,[(p \wedge q) \vee (\neg q \wedge \neg s)]$ *est une instance de* $2\mathrm{QBF}$.

   Théorème BM.8 ([APT79]). *On peut résoudre* $2\mathrm{QBF}$ *en temps linéaire.*

BN Complétude de la déduction naturelle (Henkin).

Leçons possibles : 918 924
Adapté depuis : [Dow08], p 51

Elegance : ★★★★★

Avis.

On propose la preuve dite "de Henkin" pour le théorème de complétude. Elle est (à mon humble avis) la preuve de complétude la plus accessible et la plus rapide. Attention cela reste quand même long à développer. Se placer dans le cadre d'un langage fini ou dénombrable est une hypothèse raisonnable pour éviter les récurrences transfinies, les filtres, ou autres joyeusetés.

Prérequis.

1. Deux lemmes techniques (mais pas tant que ça).
   Si l'on peut prouver G à partir de F, on peut prouver $F \to G$.

   Lemme BN.1 (lemme de déduction). *Si $\mathcal{T} \cup \{F\}$ démontre G, alors $\mathcal{T}$ démontre $F \to G$.*

   > *Preuve.* Par induction sur la preuve.
   >
   > □

   Si on peut prouver $\varphi(c)$ sans avoir fait d'hypothèses sur c, on peut montrer $\forall x \varphi(x)$.

   Lemme BN.2 (lemme de généralisation). *Si c n'apparaît pas dans $\mathcal{T}$ ni $\varphi(x)$ et que $\mathcal{T}$ démontre $\varphi(c/x)$, alors $\mathcal{T}$ démontre $\forall y \varphi(y/x)$.*

   > *Preuve.* La preuve de $\varphi(c/x)$ se transforme en une preuve de $\varphi(y/x)$ en remplaçant tous les c par des y (variable fraîche). A justifier par induction.
   > Par $\forall$-intro, on obtient une preuve de $\forall y \varphi(y/x)$.
   >
   > □

2. Diverses formulations de la complétude.

   Lemme BN.3 (Complétude, conditions équivalentes). *Sont équivalentes :*

   (a) pour toute $\mathcal{T}$ et A, si $\mathcal{T}$ démontre A alors A est valide dans tous les modèles de $\mathcal{T}$;

   (b) pour toute $\mathcal{T}$ et A, si A non démontrable dans $\mathcal{T}$, alors il existe un modèle de $\mathcal{T}$ qui n'est pas un modèle de A ;

   (c) pour toute $\mathcal{T}$ cohérente, $\mathcal{T}$ a un modèle.

   > *Preuve.* (a) et (b) sont clairement équivalentes. En prenant $A = \bot$, (a) $\Rightarrow$(c) devient clair. Enfin si on a (c), et que A n'est pas démontrable dans $\mathcal{T}$, alors $\mathcal{T} \cup \{\neg A\}$ est cohérente (voir BN.1) donc elle a un modèle. Ce n'est pas un modèle de A. □

3. Témoins de Henkin.

   Définition BN.4. *On dit qu'une théorie $\mathcal{T}$ possède des témoins de Henkin, si lorsque $\exists x \varphi(x)$ est démontrable, il existe un terme t tel que $\varphi(t)$ soit démontrable.*

Développement.

Dans la preuve les langages sont supposés finis ou dénombrables.

Lemme BN.5 (Complétion de Henkin). *Soit $\mathcal{T}$ cohérente sur un langage $\mathcal{L}$ il existe $\mathcal{L}' \supseteq \mathcal{L}$ et $\mathcal{U} \supseteq \mathcal{T}$ une théorie sur $\mathcal{L}'$ qui est cohérente, complète et possède des témoins de Henkin.*

Preuve. Soit H un ensemble dénombrable de constantes fraîches.
On pose $\mathcal{L}' := \mathcal{L} \uplus H$ dénombrable. Soit $A_0, \dots, A_n, \dots$ une énumération des formules closes sur ce langage.

Soit $\mathcal{T}_0 := \mathcal{T}$, on construit par récurrence une suite croissante de théories cohérentes $\mathcal{T}_n$. Notons $G_n := A_n$ si A_n démontrable dans $\mathcal{T}_n$, et $G_n := \neg A_n$ sinon.
— si G_n n'est pas de la forme $\exists x\varphi(x)$, on pose $\mathcal{T}_{n+1} := \mathcal{T}_n \cup \{G_n\}$;
— sinon on pose $\mathcal{T}_{n+1} := \mathcal{T}_n \cup \{G_n, \varphi(c/x)\}$ où $c \in H$ n'apparaît pas dans $\mathcal{T}_n$ ni $\varphi(x)$.

Alors si $\mathcal{T}_n$ est cohérente $\mathcal{T}_{n+1}$ aussi.

1. $\mathcal{T}_n \cup \{G_n\}$ est cohérente. C'est clair si $G_n = A_n$. Dans l'autre cas supposons $\mathcal{T}_n \cup \{\neg A_n\}$ incohérente, alors par BN.1 $\mathcal{T}_n$ démontre A_n ce qui n'est pas possible.
2. Si $G_n = \exists x\varphi(x)$. Supposons que $\mathcal{T}_n \cup \{G_n, \varphi(c/x)\}$ est incohérente, alors $\mathcal{T}_n \cup \{\exists x\varphi(x)\}$ démontre $\neg\varphi(c/x)$ par BN.1, donc $\forall y\varphi(y)$ par BN.2. On en déduit une contradiction.

Par compacité [1], la théorie $\mathcal{U} := \bigcup_{n\geq 0} \mathcal{T}_n$ est cohérente. Elle contient $\mathcal{T}$. Il est clair qu'elle est complète et qu'elle a des témoins de Henkin.

□

Théorème BN.6 (Complétude). *Si $\mathcal{T}$ est cohérente, $\mathcal{T}$ a un modèle.*

Preuve. Si $\mathcal{U}$ sur $\mathcal{L}'$ le complété de $\mathcal{T}$ par BN.5, montrons $\mathcal{U}$ a un modèle.
Considérons le modèle $\mathcal{M}$ sur $\mathcal{L}'$ dont les éléments sont les termes et les interprétations :
— fonctions $f(t_1, \dots, t_n) = f(t_1, \dots, t_n)$;
— le prédicats $P(t_1, \dots, t_n)$ vrai ssi démontrable dans $\mathcal{U}$.
On montre alors par induction structurelle que si A close est démontrable (axiome) dans $\mathcal{U}$, on a $\mathcal{M} \vDash A$. C'est évident pour les formules atomiques. Si $\exists x\varphi(x)$ est démontrable, alors grâce aux témoins il existe c tel que $\varphi(c/x)$ démontrable. Par induction $\mathcal{M} \vDash \varphi(c/x)$ donc $\mathcal{M} \vDash \exists x\varphi(x)$ [2].
Donc $\mathcal{M}$ modèle de $\mathcal{U}$. Par restriction du langage, $\mathcal{M}$ est également un modèle de $\mathcal{T}$.

□

Postrequis.

1. **Attention.** Cette preuve produit *toujours* un modèle dénombrable…
   Mais alors, que dire de $\exists x \forall y, x = y$ qui décrit les modèles de cardinal 1 ? Et bien non ! Elle décrirait les modèles de cardinal 1 si on forçait l'interprétation de = comme la "vraie égalité". Mais rien n'y oblige a priori.
   Le forcer, c'est se placer en logique égalitaire.

   Proposition BN.7. *Soit $\mathcal{T}$ est une théorie sur $\mathcal{L}$ contenant =, et $\mathcal{E}$ la théorie de l'égalité associée. Si $\mathcal{T} \cup \mathcal{E}$ est cohérente, alors $\mathcal{T}$ a un modèle égalitaire.*

   Preuve. On peut trouver un modèle de $\mathcal{T} \cup \mathcal{E}$ par complétude. L'interprétation de = dans ce modèle définit une relation d'équivalence. En fait, c'est une congruence (eu égard aux autres constructeurs du langage). En passant au quotient, on obtient encore un modèle de $\mathcal{T}(\cup\mathcal{E})$, mais dans lequel = est la vraie égalité.

   □

2. Cas général (si le langage n'est pas forcément dénombrable). On peut utiliser le lemme de Zorn en regardant les sur-théories cohérentes (sur des extensions du langage de même cardinalité que $\mathcal{L}$) qui est un "ensemble" inductif. Un élément maximal est alors une théorie cohérente, nécessairement complète et qui possède nécessairement des témoins de Henkin.

1. Juste le fait qu'il existerait une preuve finie de l'incohérence.
2. Pour les autres cas, il suffit de regarder $\vee$ et $\neg$. Ce n'est pas totalement trivial et exploite la complétude de $\mathcal{U}$.

3. La porte de la théorie des modèles : compacité et Löwenheim-Skolem.

   Théorème BN.8 (Compacité de la logique du premier ordre)**.** *Si $\mathcal{T}$ n'a pas de modèle, elle possède une sous-théorie finie qui n'a pas de modèle.*

   > *Preuve.* Par complétude, si $\mathcal{T}$ n'a pas modèle, elle est incohérente. Donc (puisqu'une preuve n'utilise qu'un nombre fini d'axiomes) elle a une sous-théorie incohérente. Celle-ci n'a pas de modèle (correction).
   >
   > □

   Théorème BN.9 (Löwenheim-Skolem descendant)**.** *Si $\mathcal{T}$ a un modèle, elle a un modèle dénombrable si son langage est fini ou dénombrable ; de même cardinalité que son langage sinon.*

   > *Preuve.* Si $\mathcal{T}$ a un modèle, elle est cohérente. La preuve du théorème des complétude permet alors de produire un modèle de cardinalité dénombrable (dans notre cas), ou de sinon de même cardinalité que le langage (non traité ici). □

   Théorème BN.10 (Löwenheim-Skolem ascendant)**.** *Si $\mathcal{T}$ a un modèle infini, elle a un modèle de toute cardinalité supérieure à celui de son langage.*

4. On verrait comment montrer la complétude dans le cas propositionnel : c'est la même preuve sans les témoins de Henkin (et c'est donc plus facile).

5. Peut-on simplifier la preuve ? Se débarrasser des témoins de Henkin ? Non !
   En effet, si $\mathcal{L} = \{P, c\}$ et $\mathcal{T} = \{\exists x P(x), \neg P(c)\}$, on aura du mal à trouver un modèle de $\mathcal{T}$ donc le seul élément soit la constante c. Il faut donc en ajouter un jour.

BO Complétude de la résolution propositionnelle.

Leçons possibles : 916 918
Pas de référence.

Elegance : ★★★☆☆

Avis.

Gentil et sympathique, il ne pose pas de difficulté particulière. Dans la 916, on présentera comme application la compacité du calcul propositionnel (voir le point 1 des Postrequis). Dans la 918, on déduira la complétude de la résolution au premier ordre, via le théorème de Herbrand et un résultat de relèvement (voir le point 2 des Postrequis).

Prérequis.

1. Voir les clauses comme des ensembles et non comme des formules.
2. Résolution.
3. Lemme de König.

Théorème BO.1.

Preuve. Pour faire du style, on notera que cette preuve utilise l'axiome du choix dépendant...

□

Développement.

On se place sur un ensemble de variables propositionnelles $\mathcal{P}$ dénombrable.

Théorème BO.2 (Complétude réfutationnelle de la résolution). *Si S est un ensemble de clauses insatisfiable, alors $S \vdash^R \bot$.*

Preuve. On considère $S^* := \{C \text{ clause} \mid S \vdash^R C\}$ qui est insatisfiable.
Notons $P_1, \dots, P_n, \dots$ une énumération de $\mathcal{P}$. On appelle interprétation partielle une fonction $\nu : \{P_1, \dots, P_n\} \to \{0,1\}$ et on dit que $\mathcal{I} \vDash C$ s'il existe une extension [1] $\mathcal{I}$ de ν à $\mathcal{P}$ telle que $\mathcal{I}(C) = 1$.
Considérons l'arbre binaire infini défini comme suit :
— nœuds : interprétations partielles ν ;
— les fils de ν (définie sur $\{P_1, \dots, P_n\}$) sont $\nu + [P_{n+1} \mapsto 0]$ et $\nu + [P_{n+1} \mapsto 1]$;
On appelle *nœud d'échec* une ν telle qu'il existe une clause $C \in S^*$ avec $\nu \nvDash C$. Notons qu'alors tous ses successeurs sont des nœuds d'échec.
Notons A [2] l'arbre précédent, élagué aux premiers nœuds d'échec inclus)

Lemme BO.3. *A est fini.*

Preuve. Supposons le contraire. En vertu du lemme de König BO.1 il a une branche infini. Soit $\mathcal{I}$ l'interprétation (totale) déduite cette branche. On pour $C \in S^*$, $\mathcal{I}(C) = 1$. En effet sinon il existe $n \geq 0$ tel que $\mathcal{I}|_{\{P_1,\dots,P_n\}}(C) = 0$ (car la clause est finie de variables), et ce serait un nœud d'échec.

□

DESSIN

Lemme BO.4. *A n'a aucun nœud de succès.*

Preuve. Supposons que l'arbre sémantique contienne au moins un nœud de succès. Soit ν un tel nœud de profondeur maximale, il a nécessairement deux fils d'échec [3]. Il existe donc $C_1, C_2 \in S^*$ et $P \in \mathcal{P}$ telles que $\nu \vDash C_1$ et $\nu \vDash C_2$

1. Au sens des fonctions partielles.
2. arbre sémantique
3. Par maximalité.

mais $\nu + [P \to 1] \not\models C_1$ et $\nu + [P \to 0] \not\models C_2$.
Alors $C_1 = \neg P \vee C_1'$ [1] et de même $C_2 = P \vee C_2'$ où P n'apparaît pas dans C_1', C_2'.
De plus, $\nu \not\models C_1'$ et $\nu \not\models C_2'$ [2] donc $\nu \not\models C_1' \vee C_2'$.
Or $C_1, C_2 \vdash^R C_1' \vee C_2'$ par résolution.
Donc $C_1' \vee C_2' \in S^*$ et ν était un nœud d'échec.

□

A était donc réduit à sa racine, donc une clause est invalidée par l'interprétation vide. C'est nécessairement la clause vide $\bot$. Donc $\bot \in S^*$.

□

Postrequis.

1. Compacité propositionnelle.

   Corollaire BO.5 (Compacité). *Si un ensemble de formules propositionnelles S est insatisfiable, il possède un sous-ensemble fini insatisfiable.*

   Preuve. Tout ensemble de formules est équivalent à une conjonction infinie de clauses (il suffit de mettre chaque formule en CNF, puis de les "découper"). Donc on peut supposer que S est de cette forme. Comme il est insatisfiable et par complétude réfutationnelle, $S \vdash \bot$ en résolution. Donc $S_0 \vdash \bot$ pour un $S_0 \subseteq S$ fini. Par correction, S_0 est incohérente.

   □

   De là, on en déduit toutes les applications dont on a toujours rêvé :
   — comme tous les graphes planaires finis sont 4-coloriables, alors tous les graphes planaires infinis aussi [3] ;
   — comme tout ensemble fini peut être muni d'un ordre total, tout ensemble infini aussi [4] ;
   — et en général, tout ce qu'on peut faire passer de *fini* à *infini*.

2. Application en logique du premier ordre.
   Le terme "formule" désignera ici par défaut une formule du premier ordre (signature Σ).

   (a) La résolution au premier ordre.
   On appellera désormais *clause* une conjonction de littéraux du premier ordre. Elle n'a donc pas de quantificateurs, et son interprétation se fait avec des $\forall$ implicites.

   Proposition BO.6 (Mise en forme clausale). *Soit F un ensemble de formules closes, il existe un ensemble S de clauses tel que F satisfiable ssi S satisfiable* [5]*. La construction est effective.*

   Idée de preuve. La transformation se décompose en 4 étapes.

   Mise en forme prénexe. Transformer chaque formule en une formule de la forme $Q_1 x_1, \dots Q_n x_n \varphi(x_1, \dots, x_n)$ avec φ sans quantificateurs.

   CNF Mettre chaque φ en forme CNF (comme une formule propositionnelle).

   Skolémisation Skolémiser la formule obtenue. On retire les $\exists$ en ajoutant des symboles de fonctions marquant les dépendances.

   Cas des $\forall$ Retirer les $\forall$ (tout simplement).

   □

1. Car si P n'apparaît pas, comme $\nu \models C_1$, l'extension ne change rien. Et s'il apparaît positivement c'est vrai.
2. Sans quoi il n'y aurait pas d'échec.
3. J'ai toujours rêvé de 4-colorier un graphe *infini*...
4. On notera ici l'inutilité de ce résultat, puisque la version infinie non-dénombrable de la compacité utilise l'axiome du choix. Celui-ci est équivalent au théorème de Zermelo : tout ensemble possède un bon ordre. Donc *a fortiori* un ordre total. Et pour le cas dénombrable, on savait déjà mettre un ordre total sur $\mathbb{N}$...
5. Attention, F et S ne sont pas équivalents (d'ailleurs ils ne sont pas sur la même signature).

On a donc "normalisé" les formules du calcul propositionnel. On définit alors les règles de la résolution au premier ordre, sur les formes clausales.

(b) Correction de la résolution.

Rappelons que $S \vDash \varphi$ signifie que tout modèle de S est aussi un modèle de φ.

Théorème BO.7 (Correction de la résolution au premier ordre). *Si $S \vdash^R \varphi$ alors $S \vDash \varphi$.*

> *Idée de preuve.* On vérifie que c'est vrai, par induction sur la longueur de la preuve par résolution. Il faut utiliser le fait que le MGU unifie les deux termes, mais on n'a même pas besoin ici que ce soit le plus général. □

(c) Herbrand : descendre l'insatisfiabilité du premier ordre au propositionnel.

Soit D_Σ l'ensemble des termes clos sur Σ. Un *modèle de Herbrand* est un modèle dont le domaine est D_Σ et où les fonctions sont interprétées comme constructeurs de termes. Notons qu'on peut définir librement les valeurs de vérité des prédicats de la *base de Herbrand* $\mathcal{B} := \{P(t_1, \dots, t_n) \mid P \text{ prédicat et } t_1, \dots, t_n \in D_\Sigma\}$.

Enfin, notons $\varphi\Sigma = \{\varphi\sigma \mid \sigma \text{ substitution close}\}$ et $S\Sigma = \bigcup_{\varphi \in S} \varphi\sigma$.

Théorème BO.8 (Herbrand). *Soit S ensemble de clauses, alors sont équivalentes :*

(i) S a un modèle ;

(ii) S a un modèle de Herbrand ;

(iii) $S\Sigma$ est propositionnellement satisfiable.

> *Idée de preuve.* L'équivalence entre (i) et (ii) est claire en remarquant que la validité de S dans un modèle de Herbrand donné est équivalente à la validité propositionnelle de $S\Sigma$ dans l'interprétation de $\mathcal{B}(\Sigma)$ correspondante.
> (ii) $\Rightarrow$ (i) est évident.
> Montrons que (i) $\Rightarrow$ (iii) par contre apposée. Si $S\Sigma$ n'est pas satisfiable, alors il existerait une preuve propositionnelle de $\bot$ depuis $S\Sigma$, donc une preuve au premier ordre de $\bot$ depuis S (avec des instanciations par $\forall$-elim)[1].
>
> □

(d) Relèvement : remonter les preuves du propositionnel au premier ordre.

Théorème BO.9 (Relèvement). *Soit S un ensemble de clauses, si $S\Sigma \vdash^R C'$ (propositionnelle), alors il existe C telle que $C' \in C\Sigma$ et $S \vdash^{R_1} C$.*

> *Idée de preuve.* On procède par induction sur la structure de la preuve. Le seul cas est celui de la règle de résolution, qui donne lieu au lemme suivant. C'est à l'intérieur que l'on utilise la maximalité du MGU et la règle de factorisation.
>
> **Lemme BO.10.** *Soient C_1, C_2 des clauses (au premier ordre) et $C_1' \in C_1\Sigma$, $C_2' \in C_2\Sigma$. Si C_1', C_2' se résolvent (en une étape) en C', alors il existe une clause C telle que $C' \in C\Sigma$ et $C_1, C_2 \vdash^{R_1} C$.*
>
> □

(e) Complétude.

Théorème BO.11 (Complétude réfutationnelle de la résolution au premier ordre). *Si S est un ensemble insatisfiable de clauses, alors $S \vdash^{R_1} \bot$.*

> *Preuve.* Par Herbrand BO.8 $S\Sigma$ est également insatisfiable (propositionnellement). En vertu de la complétude réfutationnelle de R (résolution propositionnelle), il vient $S\Sigma \vdash^R \bot$. Par le relèvement BO.9, il vient $S \vdash^{R_1} C$ où $\bot \in C\Sigma$, et nécessairement $C = \bot$.
>
> □

1. On prend ici un sytème de preuve comme la déduction naturelle, et pas la résolution. En conséquence qui fait que l'on n'a pas besoin du lemme de relèvement (qui arrive après).

BP Problèmes indécidables sur les grammaires algébriques.

Leçons possibles : 914 923
Adapté depuis : [Car08], p 166

Elegance : ★★★★☆

Avis.

Attention, le [Car08] oublie que ε n'est pas une solution de PCP (donc il aurait toujours une solution, donc il serait décidable... ahhh !), ce qui fait que la preuve est inexacte. On enlève un de ses résultats pour le remplacer par un autre (ce qui évite une technicité).
Les idées illustrées dans ce développement. Analyse lexicale : que les grammaires algébriques *dans leur généralité* sont un outil trop puissant pour avoir de bonnes propriétés. A comparer aux automates pour lesquels tout était décidable. En outre, l'ambiguïté est un problème insoluble, ce qui motive l'emploi d'analyseurs LL ou LR. Décidabilité : on illustre l'intérêt de PCP pour montrer l'indécidabilité de problèmes sur les langages. C'est beaucoup plus agréable que de passer par des machines de Turing ! Enfin on montre comment "emboiter" les réductions pour déduire l'indécidabilité du problème 2 de celle du 1.

Prérequis.

1. Problème de correspondance de Post[1] (PCP).
2. Un peu de grammaires.

   Proposition BP.1. *Etant donnée G grammaire sur Σ, $\mathcal{L}(G) = \Sigma^*$ est indécidable.*

   Preuve. On réduit PCP comme dans le Développement à suivre[2], mais on donne des grammaires pour les complémentaires $\overline{\mathcal{L}(G)}$ et $\overline{\mathcal{L}(G)}$. L'union de ces deux langages est universelle ssi $\mathcal{L}(G) \cap \mathcal{L}(G') = \varnothing$ (passer au complémentaire). □

   Proposition BP.2. *$L_0 := \{a^n b^n \mid n \geq 0\}$ est algébrique mais pas rationnel.*

Développement.

Proposition BP.3. *Les problèmes suivants sont indécidables :*

1. *G, G' grammaires non-ambigües, est-ce que $\mathcal{L}(G) \cap \mathcal{L}(G') = \varnothing$?*
2. *G grammaire, est-ce que G est ambigüe ?*
3. *G grammaire, est-ce que $\mathcal{L}(G)$ est rationnel ?*

Preuve. 1. On réduit PCP. Soit $u_1, \dots, u_n$ et $v_1, \dots, v_n \in \Sigma^*$ une instance de PCP.
Construisons une grammaire G :
— symboles non-terminaux : S axiome ;
— symboles terminaux : $\Sigma' := \Sigma \uplus \{1, \dots, n\}$;
— règles : $S \to u_i S i | u_i i$ (pour $1 \leq i \leq n$).
Alors G est non-ambigüe. En outre $\mathcal{L}(G) = \{u_{i_1} \dots u_{i_k} i_k \dots i_1 | k \geq 1, 1 \leq i_j \leq n\}$.
De même soit G' non-ambigüe telle que $\mathcal{L}(G') = \{v_{i_1} \dots v_{i_k} i_k \dots i_1 | k \geq 1, 1 \leq i_j \leq n\}$.
Alors $\mathcal{L}(G) \cap \mathcal{L}(G') = \varnothing$ ssi PCP a une solution. $\Leftarrow$ est évident. $\Rightarrow$. Si w est dans l'intersection, il s'écrit $u_{i_1} \dots u_{i_k} i_k \dots i_1 = v_{j_1} \dots v_{j_{k'}} j_{k'} \dots j_k =$. Donc $i_l = j_l$ et $k = k'$, enfin $u_{i_1} \dots u_{i_k} = v_{i_1} \dots v_{i_k}$ est une solution.

1. De Emil Post. Ne pas confondre avec *La Poste*, ni avec le *problème du postier chinois* (*chinese postman problem* en algorithmique des graphes), et encore moins avec le *théorème* PCP (très dur, classes de complexité randomisées).
2. On ne peut pas utiliser ce résultat en "boite noire", puisque le complémentaire n'est pas algébrique en général.

2. On réduit le problème 1. Soient G et G' de telles grammaires de non terminants Γ, Γ' et d'axiomes respectifs S, S'. On construit la grammaire G_0 :
   — non-terminaux : $N_0 := \Gamma \uplus \Gamma' \uplus \{S_0\}$ (avec S_0 axiome) ;
   — règles : celles de G et G' + $S_0 \to S|S'$.
   Alors G_0 est ambigüe ssi $\mathcal{L}(G) \cap \mathcal{L}(G') \neq \varnothing$. $\Leftarrow$ évident. $\Rightarrow$. Si w est un mot ayant deux dérivations, alors l'une commence par $S_0 \to S$ et l'autre part $S_0 \to S'$ (utilise la non-ambiguïté de G et G'). Ensuite il vient que $w \in \mathcal{L}(G) \cap \mathcal{L}(G')$.
3. On réduit l'universalité BP.1. Soit G une grammaire de terminants Σ.
   Soit $L_0 = \{a^n b^n \mid n \geq 0\}$ qui est algébrique non rationnel. Pour $\#$ nouveau symbole, le langage $L := L_0 \# \Sigma^* \cup \{a, b\}^* \# \mathcal{L}(G)$ est algébrique.
   L est rationnel ssi $\mathcal{L}(G) = \Sigma^*$. $\Leftarrow$ alors $L = \{a, b\}^* \# \Sigma^*$ rationnel. $\Rightarrow$. Par l'absurde, si L est rationnel et $w \notin \mathcal{L}(G)$, alors $L \cap \{a, b\}^* \# w = L_0 \# w$ qui devrait être rationnel (stabilité par intersection), mais ne l'est pas.

□

Postrequis.

1. La technique employée en 3 est assez générique pour peu que la classe de langages vérifie certaines propriétés de clôture (on parle de *full trio*). On peut employer l'idée pour montrer l'indécidabilité de l'*ambiguïté inhérente* d'un langage algébrique.
2. Une bonne référence pour clore le sujet. [Hoo].

BQ Complétude des règles de Hoare.

Leçons possibles : 927 930
Adapté depuis : [Win93], p 107

Elegance : ★★★☆☆

Avis.

On se place dans la logique de Hoare au premier ordre, pour parler du langage IMP. On admettra le fait que la logique puisse exprimer les plus faibles préconditions, et on se concentrera sur la complétude en utilisant les règles de preuve. La complétude de la logique de Hoare fait faire des "preuves d'algorithmes" avec des invariants de boucle, c'est une belle illustration dans la 927.
Quelques remarques en vrac sur Hoare :

- la règle pre/post est *essentielle* à la complétude, mais malheureusement elle n'est pas "calculable" (il s'agit de déterminer une implication au premier ordre) ;
- les variables logiques libres servent à parler de valeurs qui resteront constantes dans le programme (par exemple pour continuer à parler de la valeur initiale d'une variable), on pourrait aussi ajouter des variables au programme sur lesquelles on ne fait rien ;
- la correction est toujours *partielle*, il n'est pas question de terminaison ici.

Prérequis.

1. Logique de Hoare.
   Syntaxe des formules. Triplets de Hoare.
   Interprétation : environnement (variables de programme) + valuation (variables logiques).
   Règles de Hoare : Skip, Affectation, Séquence, Conditions, While, Pre/Post.
2. Correction.
   Lemme BQ.1. $\sigma \vDash^{\varphi} [e/x]$ *ssi* $[e/x]\sigma \vDash^{\varphi}$.
   Théorème BQ.2 (Correction partielle, [Win93], p). *Si* $\vdash^{H} \{\varphi\}c\{\psi\}$ *est démontrable, alors pour toute valuation* I *et environnement* σ*, on a* $\sigma \vDash^{I} \{\varphi\}c\{\psi\}$.
3. Plus faibles préconditions.
   Définition BQ.3. *On dit qu'une formule* φ *est* une plus faible précondition *de* ψ *pour* c *si pour toute* I, $\{\sigma \mid \sigma \vDash^{I} \varphi\} = \{\sigma \mid [c]\sigma \vDash^{I} \psi\}$.
   Théorème BQ.4 (Expressivité, [Win93] p 103). *Pour tous* ψ *et* c*, il existe* une plus faible précondition *notée* $\mathrm{wp}(c, \psi)$ *et cette dernière est effectivement calculable.*

   > *Vague idée de preuve.* On procède par induction sur c. Tous les cas sont assez clairs (on les revoit d'ailleurs dans le développement) sauf le while. Ce dernier requiert une gödelisation des fonctions, c'est *assez pénible*. Donc on admettra ce théorème.
   >
   > □

4. Un lemme d'équivalence.
   Lemme BQ.5. *Soit* $c :=$ $\mathrm{while}\, e\, \mathrm{do}\, c_0$ *alors* c *et* $\mathrm{if}\, e\, \mathrm{then}\, c_0; c\, \mathrm{else}\, \mathrm{skip}$ *ont même sémantique.*

Développement.

Lemme BQ.6. $\{\mathrm{wp}(c, \varphi)\}c\{\varphi\}$ *est démontrable.*

> *Preuve.* On procède par induction sur c.
>
> **Cas** $c = \mathrm{skip}$. Alors $\sigma \vDash^{I} \mathrm{wp}(c, \varphi)$ ssi $[\mathrm{skip}]\sigma \vDash^{I} \varphi$ ssi $\sigma \vDash^{I} \varphi$.
> Donc $\mathrm{wp}(c, \varphi) \iff \varphi$ et en particulier $\mathrm{wp}(c, \varphi) \Rightarrow \varphi$.
> Or $\vdash^{H} \{\varphi\}\,\mathrm{skip}\{\varphi\}$ d'où $\vdash^{H} \{\mathrm{wp}(c, \varphi)\}\,\mathrm{skip}\{\varphi\}$ par pré-renforcement.

Cas $c = x := e$. Alors $\sigma \vDash^I \mathrm{wp}(c, \varphi)$ ssi $[x := e]\sigma \vDash^I \varphi$ ssi $\sigma \vDash^I \varphi[e/x]$.
Donc $\mathrm{wp}(c, \varphi) \iff \varphi[e/x]$ et en particulier $\mathrm{wp}(c, \varphi) \Rightarrow \varphi[e/x]$.
Or $\vdash^H \{\varphi[e/x]\}x := e\{\varphi\}$ et on conclut par pré-renforcement.

Cas $c = c_1; c_2$. Alors $\sigma \vDash^I \mathrm{wp}(c_1; c_2, \varphi)$ ssi $[c_1; c_2]\sigma \vDash^I \varphi$ ssi $[c_2]([c_1]\sigma) \vDash^I \varphi[e/x]$
ssi $[c_1]\sigma \vDash^I \mathrm{wp}(c_2, \varphi)$ ssi $\sigma \vDash^I \mathrm{wp}(c_1, \mathrm{wp}(c_2, \varphi))$.
Donc $\mathrm{wp}(c, \varphi) \iff \mathrm{wp}(c_1, \mathrm{wp}(c_2, \varphi))$.
Mais par HR, $\vdash^H \{\mathrm{wp}(c_1, \mathrm{wp}(c_2, \varphi))\}c_1\{\mathrm{wp}(c_2, \varphi)\}$ et $\vdash^H \{\mathrm{wp}(c_2, \varphi)\}c_2\{\varphi\}$.
On conclut en utilisant la règle de la séquence et un pré-renforcement.

Cas $c = \mathrm{if}\, e \,\mathrm{then}\, c_1 \,\mathrm{else}\, c_2$. Admis (pas de difficulté autre qu'une disjonction de cas).

Cas $c = \mathrm{while}\, e \,\mathrm{do}\, c_0$. Montrons deux faits :

(i) $\vdash^H \{\mathrm{wp}(c, \varphi) \wedge e \neq 0\}c_0\{\mathrm{wp}(c, \varphi)\}$ [1].
Supposons que $\sigma \vDash^I \mathrm{wp}(c, \varphi) \wedge e \neq 0$. Alors $[e]\sigma \neq 0$ et $[c]\sigma \vDash^I \varphi$.
Or $[c] = [\mathrm{if}\, e \,\mathrm{then}\, c_0; c \,\mathrm{else}\, \mathrm{skip}]$ par BQ.5.
Donc $[c]\sigma = [c_0; c]\sigma = [c]([c_0]\sigma) \vDash^I \varphi$.
Donc $[c_0]\sigma \vDash^I \mathrm{wp}(c, \varphi)$ et on conclut par HR et pré-renforcement.

(ii) $\mathrm{wp}(c, \varphi) \wedge e = 0 \Rightarrow \varphi$.
Supposons que $\sigma \vDash^I \mathrm{wp}(c, \varphi) \wedge e = 0$. Alors $[e]\sigma = 0$ et $[c]\sigma \vDash^I \varphi$.
En utilisant l'équivalence précédente, $[c]\sigma = [\mathrm{skip}]\sigma = \sigma \vDash^I \varphi$.

Enfin, on applique la règle while à (i) et il vient : $\vdash^H \{\mathrm{wp}(c, \varphi)\}c\{\mathrm{wp}(c, \varphi) \wedge e \neq 0\}$.
Par (ii) et post-affaiblissement [2], $\vdash^H \{\mathrm{wp}(c, \varphi)\}c\{\varphi\}$

□

Théorème BQ.7. *Si pour toute I, σ, $\sigma \vDash^I \{\varphi\}c\{\psi\}$, alors $\vdash^H \{\varphi\}c\{\psi\}$.*

Preuve. Si $\sigma \vDash^I \varphi$ alors $[c]\sigma \vDash^I \psi$. Donc $\sigma \vDash^I \mathrm{wp}(c, \psi)$. Donc $\varphi \Rightarrow \mathrm{wp}(c, \psi)$.
Or $\vdash^H \mathrm{wp}(c, \psi) \Rightarrow \varphi$ donc $\vdash^H \varphi \Rightarrow \psi$ aussi par pré-renforcement.

□

Postrequis.

1. En pratique.
   On pourrait calculer les $\mathrm{wp}(c, \psi)$ pour vérifier la validité d'un triplet. Mais c'est lourd, et on préférera *annoter à la main* le programme (avec des invariantes devant les while, mais aussi pour le séquence) et tenter de prouver ce qui reste au premier ordre.

1. Autrement dit c'est un invariant de boucle !
2. C'est enfin ici qu'on l'utilise !

BR Adéquation dénotationnel/grands pas. (NR)

Leçons possibles : 930
Adapté depuis : [Win93], p 65

Elegance : ★★☆☆☆

Développement non-rédigé...

Deuxième partie

Plans de plans, leçon par leçon

CHAPITRE 3

LEÇONS D'ALGÈBRE

Ouvre-nous, noir Pluton,
Les portes du Ténare !
Fais retentir, Caron,
Ta funèbre fanfare !

Les Troyens, Hector Berlioz

104 Groupes finis. Exemples et applications.

Développements choisis

— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★☆☆
— AB Théorème de Brauer.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Dans cette leçon il faut savoir manipuler correctement les éléments de différentes structures usuelles ($\mathbb{Z}/n\mathbb{Z}$, $\mathfrak{S}_n$, etc.) comme, par exemple, en proposer un générateur ou une famille de générateurs, savoir calculer un produit de deux permutations, savoir décomposer une permutation en produit de cycles à supports disjoints. Il est important que la notion d'ordre d'un élément soit mentionnée et comprise dans des cas simples. Le théorème de structure des groupes abéliens finis doit être connu. Les exemples doivent figurer en bonne place dans cette leçon. Les groupes d'automorphismes fournissent des exemples très naturels. On peut aussi étudier les groupes de symétries $\mathfrak{A}_4$, $\mathfrak{S}_n 4$, $\mathfrak{A}_5$ et relier sur ces exemples géométrie et algèbre, les représentations ayant ici toute leur place ; il est utile de connaître les groupes diédraux. S'ils le désirent, les candidats peuvent ensuite mettre en avant les spécificités de groupes comme le groupe quaternionique, les sous-groupes finis de $\mathrm{SU}(2)$ ou les groupes $\mathrm{GL}_n(\mathbb{F}_q)$.

Avis.

Énormément de choses à dire donc il faut faire des choix ! On gagnera du temps en ne rappelant pas les notions générales de théorie des groupes (i.e. pas spécifiques aux groupes finis) comme les définitions et les propriétés de quotients. L'étude des groupes cycliques conduit naturellement aux $\mathbb{Z}/n\mathbb{Z}$ et plus généralement à l'étude des groupes abéliens, qui peut être menée sans trop de prérequis. Cela dit, on ne pourra pas faire l'impasse sur les techniques de base sur les groupes finis : l'ordre des éléments, la combinatoire des actions, et les théorèmes de Sylow.

Plan.

1. Sous-groupes et groupes abéliens

1.1 Généralités

Ordre et théorème de Lagrange (sans mentionner explicitement l'action).
Groupe fini ssi nombre fini de sous-groupes. Exposant. Si tous les éléments sont d'ordre 2.

1.2 Groupes abéliens

Etude de $\mathbb{Z}/n\mathbb{Z}$ comme groupe. Chinois. Ordre des éléments, sous-groupes. Automorphismes et étude de $(\mathbb{Z}/n\mathbb{Z})^*$ [Per98].
Structure des groupes abéliens (de type) finis.

2. Outils pour l'étude des groupes finis

2.1 Actions de groupes

Formule des classes. Exemples débiles. Actions par translation et conjugaison.
Application aux p-groupes : centre, groupes d'ordre p^2, existence de sous-groupe distingué de tout cardinal. Application au petit facteur premier du cardinal. Application à Wedderburn.
Fixateur, formule de Burnside.

2.2 Théorèmes de Sylow

Théorèmes. Remarques sur la manière de le prouver (Wiedlandt vs GL). Application aux groupes d'ordre pq [Per98]. Application à la classification des groupes d'ordre ≤ 8.

3. Le groupe symétrique

Théorème de Cayley.
Décomposition, $\mathfrak{A}_n$, simplicité, automorphismes. Petits cas 1,2,3,4.
Matrices de permutations. Injection des groupes finis dans GL. AB Théorème de Brauer.Injection des groupes finis. Burnside. Sous-groupes finis de SO2, SO3. Evoquer les groupes diédraux.

4. Groupes finis et corps finis

4.1 Autour du groupe multiplicatif

Cyclicité. Symbole de Legendre et interprétation. Application au théorème des deux carrés. Loi de réciprocité quadratique.

4.2 Groupe linéaire des corps finis

Baby isomorphismes exceptionnels. Dénombrements, théorèmes de Sylow. Groupes dérivés. AA Théorème de Frobenius-Zolotarev & application.Application aux symboles de Legendre.

105 Groupe des permutations d'un ensemble fini. Applications.

Développements choisis

— AB Théorème de Brauer.
Pertinence : ★★★★★
— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Parmi les attendus, il faut savoir relier la leçon avec les notions d'orbites et d'actions de groupes. Il faut aussi savoir décomposer une permutation en cycles à supports disjoints, tant sur le plan théorique (preuve du théorème de décomposition), que pratique (sur un exemple). Il est important de savoir déterminer les classes de conjugaisons du groupe symétrique par la décomposition en cycles, d'être capable de donner des systèmes de générateurs. L'existence du morphisme signature est un résultat non trivial mais ne peut pas constituer, à elle seule, l'objet d'un développement. Les applications sont nombreuses, il est très naturel de parler du déterminant, des polynômes symétriques ou des fonctions symétriques des racines d'un polynôme. S'ils le désirent, les candidats peuvent aller plus loin en s'intéressant aux automorphismes du groupe symétrique, à des problèmes de dénombrement, aux représentations des groupes des permutations ou encore aux permutations aléatoires.

Avis.

L'étude du groupe des permutations peut être motivée par le théorème de Cayley. Ne pas oublier de traiter les petits cas. Rien n'oblige à parler de représentations. Dans le plan ci-dessous, on cherche d'abord à comprendre le groupe symétrique "par lui-même" et avec des techniques de théorie de groupes. On voit ensuite ce qu'il peut apporter (et ce qui lui apportent) des actions sur divers objets. Enfin, on l'utilise pour construire le déterminant. Déterminant et groupe symétrique sont finalement reliés de manière inattendue dans le cas des corps finis.

Plan.

1. Permutations d'un ensemble fini

1.1 Groupe symétrique

Groupe des permutations d'un ensemble. Cas d'isomorphisme. Groupe $\mathfrak{S}_n$. Cardinal. Action sur l'ensemble $\{1, \dots, n\}$. Théorème de Cayley.

1.2 Cycles et transpositions

Orbite d'un élément. Décomposition en cycles à support disjoints. Lien avec l'ordre. Classes de conjugaison. Centre.
Engendrement par les transpositions.

2. Groupe symétrique et alterné

Signature, caractère morphique. Groupe alterné $\mathfrak{A}_n$. Engendrement les 3-cycles. Groupes dérivés. Simplicité de $\mathfrak{A}_n$. Sous-groupes distingués de $\mathfrak{S}_n$. Sous-groupes d'indice n de $\mathfrak{S}_n$.
Autormorphismes de $\mathfrak{S}_n$.
Etude des petits cas :
— $\mathfrak{S}_2 \simeq \mathbb{Z}/2\mathbb{Z}$
— $\mathfrak{S}_3 \simeq D_6 \simeq \mathrm{GL}_2(\mathbb{F}_2)$.

— $\mathfrak{S}_4$ cardinal 24. $\mathfrak{A}_4$ cardinal 12, dérivé V_4 et $\simeq V_4 \rtimes \mathbb{Z}/3\mathbb{Z}$. Sous-groupes distingués de $\mathfrak{S}_4$: $\mathfrak{A}_4$, V_4 et triviaux.

3. Le groupe symétrique en action

3.1 Polynômes symétriques

Algèbre des polynômes symétriques. Polynômes symétriques élémentaires. Relations coefficients-racines. Théorème de Warning. Effectivité. Applications : méthode de Lagrange pour résoudre les équations de degré 3 et 4. [Gou09a].

3.2 Matrices de permutations

Matrices de permutation. Lien avec les opérations élémentaires. AB Théorème de Brauer.

3.3 Action sur les polyèdres réguliers

Action sur les polyèdres réguliers. Interprétation géométrique des Sylows. Sous-groupes finis de SO_3.

4. Groupe symétrique et déterminants

Construction du déterminant. Déterminants des matrices de permutations. Calcul effectif par les opérations élémentaires.
Groupe linéaire d'un espace vectoriel vu comme ensemble de permutation. AA Théorème de Frobenius-Zolotarev & application.Applications aux calculs de symboles de Legendre.

106 Groupe linéaire d'un espace vectoriel de dimension finie E, sous-groupes de GL(E). Applications.

Développements choisis

— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★★★
— AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon ne doit pas se résumer à un catalogue de résultats épars sur $\mathrm{GL}(E)$. Il est important de savoir faire correspondre les sous-groupes du groupe linéaire avec les stabilisateurs de certaines actions naturelles (sur des formes quadratiques, symplectiques, sur des drapeaux, sur une décomposition en somme directe, etc.). On doit présenter des systèmes de générateurs, étudier la topologie et préciser pourquoi le choix du corps de base est important. Les liens avec le pivot de Gauss sont à détailler. Il faut aussi savoir réaliser $\mathfrak{S}_n$ dans $\mathrm{GL}(n, \mathbb{K})$ et faire le lien entre signature et déterminant. S'ils le désirent, les candidats peuvent aller plus loin en remarquant que la théorie des représentations permet d'illustrer l'importance de $\mathrm{GL}_n(\mathbb{C})$ et de son sous-groupe unitaire.

Avis.

Là encore, une leçon beaucoup trop riche. On pourra ignorer les questions suggérées ci-haut de "formes symplectiques" [1] ou de "drapeaux" si on le souhaite. Dans ce plan on utilisera librement des résultats de réduction des endomorphismes (normaux, par exemple).

Plan.

1. Groupes linéaire et spécial linéaire

Groupe spécial linéaire. Produit semi-direct.

1.1 Opérations élémentaires, générateurs

Matrices de transvections, dilatations. Opérations élémentaires à gauche. Pivot de Gauss. H2G2 Applications : orbites sous l'action à droite, à gauche. Engendrement de SL par les transvections.

1.2 Centres et groupes dérivés

Conjugaison des transvections. Groupe dérivé de GL en caractéristique $\neq 2$.

1.3 Actions par similitude et par congruence

2. Groupes finis et corps finis

Matrices de permutations. Injection de tout groupe fini. AB Théorème de Brauer..
Dénombrement de GL, théorèmes de Sylow. CG
Plongement de $\mathrm{GL}_n(\mathbb{F}_q)$ dans $\mathfrak{S}_{q^n}$. Signature d'un automorphisme.
AA Théorème de Frobenius-Zolotarev & application. Application aux lois complémentaires.

1. Mais qu'est-ce que c'est ?

3. Les cas $\mathbb{R}$ et $\mathbb{C}$

3.1 Groupe orthogonal et compacité.

Décomposition QR.
Décomposition polaire. Application : maximalité du groupe orthogonal.
AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.

3.2 Exponentielle.

Exponentielle, surjectivité vers $\mathrm{GL}_n(\mathbb{C})$ ou les carrés de $\mathrm{GL}_n(\mathbb{R})$. Inversibilité locale en 0, application sous-groupes de GL.
Homéomorphisme exponentiel $\mathcal{S}_n \simeq \mathcal{S}_n^{++}$. Application à $\mathrm{GL}_n(\mathbb{R}) \simeq \mathrm{O}_n(\mathbb{R}) \times \mathcal{S}_n^{++}$
Décomposition polaire II.

4. Groupes SO_2 et SO_3

Etude de SO_2, commutativité. Sous-groupes, groupes diédraux.
Etude de SO_3. AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions. Remarque sur leur usage pratique. Sous-groupes finis.

108 Exemples de parties génératrices d'un groupe. Applications.

Développements choisis

— AB Théorème de Brauer.
Pertinence : ★★☆☆☆
— AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.
Pertinence : ★★★★☆

Rapport 2018 du jury.

C'est une leçon qui doit être illustrée par des exemples très variés qui peuvent être en relation avec les groupes de permutations, les groupes linéaires ou leurs sous-groupes ; les groupes $\mathbb{Z}/n\mathbb{Z}$, fournissent aussi des exemples intéressants. La connaissance de parties génératrices s'avère très utile dans l'analyse des morphismes de groupes ou pour montrer la connexité de certains groupes. Tout comme dans la leçon 106, la présentation du pivot de Gauss et de ses applications est envisageable.

Avis.

Une notion utile pour l'étude d'un groupe : obtenir des décompositions (groupes abéliens, cycles dans $\mathfrak{S}_n$), des informations sur les morphismes (surjectivité)... Les exemples essentiels à traiter : groupes abéliens, groupes symétriques. Il y a beaucoup à dire sur GL et ses sous-groupes. Les développements illustrent l'intérêt d'étudier les parties génératrices, sur des exemples concrets.

Plan.

1. Notion de partie génératrice

Sous-groupe engendré. Ordre d'un élément. Lagrange. Exemples. Groupe dérivé. Propriété universelle du groupe dérivé.

2. Groupes abéliens

2.1 Groupes monogènes et cycliques

$\mathbb{Z}/n\mathbb{Z}$. Générateurs. Indicatrice d'Euler.
Un morphisme est déterminé par l'image d'un générateur. Applications : automorphismes de $\mathbb{Z}/n\mathbb{Z}$, morphismes $\mathbb{F}_p$ vers $\{\pm 1\}$ (trivial et symbole de Legendre).
[Per98], p 24.

2.2 Groupes abéliens de type fini

Ordre et exposant d'un groupe. Element d'ordre maximal.
Pseudo-base d'un groupe abélien. Classification.

3. Parties génératrices de groupes finis

3.1 Le groupe symétrique

Engendrement par les cycles. Décomposition en cycles à supports disjoints et conjugaison. Application à AB Théorème de Brauer.. Engendrement par les transpositions. Engendrement de $\mathfrak{A}_n$, simplicité, automorphismes intérieurs.

3.2 Le groupe diédral

Engendrement par les rotations et les symétries.

4. Groupe linéaire et déterminant

4.1 Groupe linéaire et déterminant

Définition de SL, des transvections, des dilatations. Opérations élémentaires, pivot de Gauss. Engendrement de SL_n par les transvections. Engendrement de GL_n. Conjugaison des transvections dans GL et dans SL. Application : $\mathcal{D}(\mathrm{GL}_n) = \mathrm{SL}_n$ hors caractéristique 2. Application à AA Théorème de Frobenius-Zolotarev & application..
[Per98], p 96.

4.2 Groupe orthogonal

Réflexions et retournement (orthogonaux). Engendrement de $\mathrm{O}_2(\mathbb{R})$ par les réflexions planes. Engendrement de $\mathrm{O}_n(\mathbb{R})$ par les réflexions. AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.. Remarque sur l'intérêt pratique des quaternions. Engendrement de $\mathrm{SO}_n(\mathbb{R})$ par les retournements. Conjugaison des retournements dans SO_n. Application : groupes dérivés de SO_n et O_n.
[Per98], p 142.

120 Anneaux $\mathbb{Z}/n\mathbb{Z}$ Applications.

Développements choisis

— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★★★☆
— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★★★

Rapport 2018 du jury.

Dans cette leçon, l'entier n n'est pas forcément un nombre premier. Il serait bon de connaître les idéaux de $\mathbb{Z}/n\mathbb{Z}$ et, plus généralement, les morphismes de groupes de $\mathbb{Z}/n\mathbb{Z}$ dans $\mathbb{Z}/m\mathbb{Z}$. Il est nécessaire de bien maîtriser le théorème chinois et sa réciproque. S'ils le désirent, les candidats peuvent poursuivre en donnant une généralisation du théorème chinois lorsque deux éléments ne sont pas premiers entre eux, ceci en faisant apparaître le PGCD et le PPCM de ces éléments. Il faut bien sûr savoir appliquer le théorème chinois à l'étude du groupe des inversibles et, ainsi, retrouver la multiplicativité de l'indicatrice d'Euler. Toujours dans le cadre du théorème chinois, il est bon de distinguer clairement les propriétés de groupes additifs et d'anneaux, de connaître les automorphismes, les nilpotents et les idempotents. Enfin, il est indispensable de présenter quelques applications arithmétiques des propriétés des anneaux $\mathbb{Z}/n\mathbb{Z}$, telles que l'étude de quelques équations diophantiennes bien choisies. De même, les applications cryptographiques telles que l'algorithme RSA sont naturelles dans cette leçon. S'ils le désirent, les candidats peuvent aller plus loin en s'intéressant au calcul effectif des racines carrées dans $\mathbb{Z}/n\mathbb{Z}$.

Avis.

Il faut absolument maîtriser le théorème chinois et la résolution de systèmes de congruences. L'indicateur d'Euler est un outil puissant, dont les petites propriétés valent le détour. Les applications en cryptographie sont inévitables : le test de Fermat, le test de Miller, et surtout RSA. Les corps finis ne sont pas hors sujet, à condition de rester au maximum dans $\mathbb{F}_p$ avec p premier.

Plan.

1. Construction et généralités

Idéaux de $\mathbb{Z}$. Sous-groupes. Anneaux $\mathbb{Z}/n\mathbb{Z}$. Remarque : calcul modulo n.
Elements inversibles, générateurs du groupe. Structure
Exemple débile d'utilisation des réductions modulo n.
Théorème Chinois pour les anneaux (transport sur les groupes, transport sur les inversibles).
Application : résolution de système de congruences.

2. Etude des inversibles

2.1 Indicateur d'Euler

Définition. $\varphi(p) = p - 1$ ssi p premier. Théorèmes d'Euler et de Fermat.
Multiplicativité. Application : expression en fonction des facteurs premiers.
$n = \sum_{d|n} \varphi(d)$. Fonction de Möbius.
AD Ordre moyen de l'indicatrice d'Euler.
$\overline{\lim}\varphi(n)/n = 1$, $\underline{\lim}\varphi(n)/n = 0$.
[Rom17]

2.2 Le groupe $(\mathbb{Z}/n\mathbb{Z})^*$

Aut$(\mathbb{Z}/n\mathbb{Z}) \simeq (\mathbb{Z}/n\mathbb{Z})^*$.
Structure de $(\mathbb{Z}/n\mathbb{Z})^*$. Application aux groupes d'ordre pq.
[Per98]

3. Cryptographie et nombres premiers

3.1 Critères de primalité

Test de Fermat. Nombres de Carmichaël (561).
Test de Miller-Rabin. Evoquer son efficacité (preuve admise).

3.2 Cryptosystème RSA

Principe du chiffrement à clef publique.
Principe de RSA.

4. Corps finis et $\mathbb{Z}/p\mathbb{Z}$

4.1 Construction des corps finis

Pour $n \geq 2$, $\mathbb{Z}/n\mathbb{Z}$ corps ssi $\mathbb{Z}/n\mathbb{Z}$ intègre ssi n premier.
Caractéristique d'un corps.
Existence des corps finis $\mathbb{F}_q$ et inclusions.
Cyclicité du groupe multiplicatif.

4.2 Symbole de Legendre

Définition : Legendre. Caractère morphique non-trivial.
Cas où -1 est un carré. Application aux premiers de la forme $4m + 1$.

4.3 Algèbre linéaire sur $\mathbb{F}_p$

Dénombrement de $\mathrm{GL}_n(\mathbb{F}_p)$. Application à Sylow.
$\mathcal{D}(\mathrm{GL}_n(\mathbb{F}_p)) = \mathrm{SL}_n(\mathbb{F}_p)$.
AA Théorème de Frobenius-Zolotarev & application.
Application à la loi complémentaire. Application à la loi de réciprocité quadratique.

121 Nombres premiers. Applications.

Développements choisis

— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★★★☆
— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Le sujet de cette leçon est très vaste. Aussi les choix devront être clairement motivés. La réduction modulo p n'est pas hors-sujet et constitue un outil puissant pour résoudre des problèmes arithmétiques simples. La répartition des nombres premiers est un résultat historique important qu'il faudrait citer. Sa démonstration n'est bien sûr pas exigible au niveau de l'agrégation. Quelques résultats sur les corps finis et leur géométrie sont les bienvenus, ainsi que des applications en cryptographie.

Avis.

On met les mêmes idées que dans la 120, à condition de les orienter différemment.

Plan.

1. Généralités en arithmétique

1.1 Divisibilité et nombres premiers

Premiers infinis. Décomposition en facteurs premiers. Caractérisation des PCGD, PPCM et écriture des nombres premiers entre eux. Indicatrice d'Euler et écriture. Crible.
Théorème des nombres premiers (de la Vallée Poussin). Version faible faisable. Progression arithmétique de Dirichlet. Version faible faisable.

1.2 Fonctions arithmétiques

Définition. $\varphi(p) = p - 1$ ssi p premier. Théorèmes d'Euler et de Fermat.
Multiplicativité. Application : expression en fonction des facteurs premiers.
$n = \sum_{d|n} \varphi(d)$. Fonction de Möbius.
AD Ordre moyen de l'indicatrice d'Euler.
$\overline{\lim}\varphi(n)/n = 1$, $\underline{\lim}\varphi(n)/n = 0$.
[Rom17]

2. Cryptographie

2.1 Critères de primalité

Test de Fermat. Nombres de Carmichaël (561).
Test de Miller-Rabin. Evoquer son efficacité (preuve admise).

2.2 Cryptosystème RSA

Principe du chiffrement à clef publique.
Principe de RSA.

3. Corps finis

3.1 Construction des corps finis

Pour $n \geq 2$, $\mathbb{Z}/n\mathbb{Z}$ corps ssi $\mathbb{Z}/n\mathbb{Z}$ intègre ssi n premier.
Caractéristique d'un corps.
Existence des corps finis $\mathbb{F}_q$ et inclusions.
Cyclicité du groupe multiplicatif.

3.2 Symbole de Legendre

Définition : Legendre. Caractère morphique non-trivial.
Cas où -1 est un carré. Application aux premiers de la forme $4m + 1$.

3.3 Algèbre linéaire sur $\mathbb{F}_p$

Dénombrement de $\mathrm{GL}_n(\mathbb{F}_p)$. Application à Sylow.
$\mathcal{D}(\mathrm{GL}_n(\mathbb{F}_p)) = \mathrm{SL}_n(\mathbb{F}_p)$.
AA Théorème de Frobenius-Zolotarev & application.
Application à la loi complémentaire. Application à la loi de réciprocité quadratique.

123 Corps finis. Applications.

Développements choisis

— AF Algorithme de Berlekamp.
Pertinence : ★★★★★
— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★★★

Rapport 2018 du jury.

Une construction des corps finis doit être connue et une bonne maîtrise des calculs dans les corps finis est indispensable. Les injections des divers $\mathbb{F}_q$ doivent être connues et les applications des corps finis (y compris pour $\mathbb{F}_q$ avec q non premier !) ne doivent pas être oubliées, par exemple l'étude de polynômes à coefficients entiers et de leur irréductibilité peut figurer dans cette leçon. La structure du groupe multiplicatif doit aussi être connue. Le calcul des degrés des extensions et le théorème de la base téléscopique sont incontournables. L'étude des carrés dans un corps fini et la résolution d'équations de degré 2 sont envisageables. S'ils le désirent, les candidats peuvent aller plus loin en détaillant des codes correcteurs ou en étudiant l'irréductiblilité des polynômes à coefficients dans un corps fini

Avis.

Une leçon centrale, dont le travail se recase partout ailleurs.

Plan.

1. Existence des corps finis

1.1 Construction de corps

Intégrité et corps des $\mathbb{Z}/n\mathbb{Z}$. Caractéristique d'un corps, sous-corps premier. Techniques vectorielles et Wedderburn. Corps de rupture pour construire les $\mathbb{F}_q$.

1.2 Premières propriétés

Cyclicité, élément primitif. Etude des automorphismes, Frobenius et ses amis. Sous-corps les uns des autres, exemple de $\mathbb{F}_{64}$.

2. Polynômes et arithmétique

2.1 Carrés dans $\mathbb{F}_q$

Etude des résidus quadratiques. Interprétation en termes d'irréductibilité. Théorème des deux carrés. Symbole de Legendre. Multiplicativité. Loi de réciprocité quadratique.

2.2 Polynômes irréductibles

Décomposition = rupture. Séparabilité des polynômes irréductibles.
AG Dénombrement des polynômes irréductibles sur $\mathbb{F}_q$.
Existence d'irréductibles de tout degré. Clôture algébrique. Réduction modulo p.
AF Algorithme de Berlekamp.

3. Algèbre linéaire et bilinéaire sur les corps finis

Baby isomorphismes exceptionnels. Dénombrements, théorèmes de Sylow. Groupes dérivés. AA Théorème de Frobenius-Zolotarev & application.Application aux calculs de Legendre. Classification des formes quadratiques.

141 Polynômes irréductibles à une indéterminée. Corps de rupture. Exemples et applications.

Développements choisis

— AF Algorithme de Berlekamp.
Pertinence : ★★★★★
— AG Dénombrement des polynômes irréductibles sur $\mathbb{F}_q$.
Pertinence : ★★★★☆

Rapport 2018 du jury.

La présentation du bagage théorique permettant de définir corps de rupture, corps de décomposition, ainsi que des illustrations dans différents types de corps (réel, rationnel, corps finis) sont inévitables. Les corps finis peuvent être illustrés par des exemples de polynômes irréductibles de degré 2, 3, 4 sur $\mathbb{F}_2$ ou $\mathbb{F}_3$. Il est nécessaire de présenter des critères d'irréductibilité de polynômes et des polynômes minimaux de quelques nombres algébriques. Il faut savoir qu'il existe des corps algébriquement clos de caractéristique nulle autres que $\mathbb{C}$; il est bon de savoir montrer que l'ensemble des nombres algébriques sur le corps $\mathbb{Q}$ des rationnels est un corps algébriquement clos. Le théorème de la base téléscopique, ainsi que les utilisations arithmétiques (utilisation de la divisibilité) que l'on peut en faire dans l'étude de l'irréductibilité des polynômes, est incontournable.

Avis.

On commence par discuter les polynômes irréductibles dans le cas général et de manière un peu abstraite : la seule motivation est de comprendre les polynômes les plus simples. Ensuite, cette théorie aride s'enrichit avec la notion d'extension de corps et la possibilité de "rompre" les polynômes (cf applications en algèbre linéaire). La dernière partie est une application naturelle à la construction et l'étude des corps finis.

Plan.

1. Polynômes irréductibles sur un anneau commutatif

1.1 Généralités

Propriété universelle de l'algèbre des polynômes. Irréductibilité. Irréductibles de petit degré. Structure de $\mathbb{A}[X]$: noetherien ssi $\mathbb{A}$ est noetherien [Per98], factoriel si $\mathbb{A}$ factoriel (Gauss, preuve dans la sous-partie suivante), euclidien si $\mathbb{A}$ corps, principal si et seulement si $\mathbb{A}$ corps. Remarques sur le cas principal.

1.2 Critères d'irréductibilité

Contenu dans le cas factoriel. Irréductibles de $\mathbb{A}[X]$ en fonction de ceux de Frac($\mathbb{A}$)$[X]$. Preuve de Gauss. Eisenstein. Exemples sur $\mathbb{Z}[X]$. Réduction modulo p. Exemples et contre-exemples $X^4 + 1$.

2. Rupture et décomposition

2.1 Extensions de corps

Base télescopique. Element algébrique, transcendant. Clôture algébrique. Exemples. Applications : trigonalisation et Cayley-Hamilton.

2.2 Des résultats plus fins : rupture et décomposition

Définitions. Existence et unicité. Questions de degré. Application : irréductibilité des cyclotomiques. Exemples. Element primitif en caractéristique nulle.

3. Les corps finis

3.1 Construction des corps finis

Cardinaux des corps finis. Existence et unicité des corps de décomposition de $X^{p^n} - X$. Cyclicité. Automorphismes des corps finis. Existence de polynômes irréductibles de tout degré. Résidus quadratiques et irréductibilité dans $\mathbb{F}_q$ de $X^2 - a$ en fonction de q. Symbole de Legendre. Théorème des deux carrés.

3.2 Irréductibilité et calculs effectifs

Rupture = décomposition. Séparabilité des polynômes irréductibles.
AG Dénombrement des polynômes irréductibles sur $\mathbb{F}_q$.
AF Algorithme de Berlekamp.
Insister sur l'effectivité.

150 Exemples d'actions de groupes sur les espaces de matrices.

Développements choisis

— AI Réduction de Frobenius.
Pertinence : ★★★★★
— AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Pertinence : ★★★★★

Rapport 2018 du jury.

Dans cette leçon il faut présenter différentes actions (congruence, similitude, équivalence, ...) et dans chaque cas on pourra dégager d'une part des invariants (rang, matrices échelonnées réduites...) et d'autre part des algorithmes, comme le pivot de Gauss, méritent aussi d'être présentés dans cette leçon. Si l'on veut aborder un aspect plus théorique, il est pertinent de faire apparaître à travers ces actions quelques décompositions célèbres ; on peut décrire les orbites lorsque la topologie s'y prête. S'ils le désirent, les candidats peuvent travailler sur des corps finis et utiliser le dénombrement dans ce contexte.

Avis.

Le plan est une liste d'actions : Steinitz, conjugaison, congruence. Pour chacune d'entre elles, il faut parler de formes normales, d'invariants et de topologie. On n'oubliera pas de mentionner le lien de chacune d'entre elles avec les objets de l'algèbre linéaire et bilinéaire (endomorphismes, formes quadratiques...) et de proposer quelques algorithmes.
Un dilemme est de savoir s'il faut ou non de matrices sur les anneaux commutatifs, d'invariants de Smith, et du calcul effectif des invariants de similitude. On fait le pari que non.

Plan.

1. Actions de Steinitz

1.1 Produit à droite et à gauche

Formes normales : échelonnement réduit en lignes, en colonnes. Invariants : image et noyau.
Pivot de Gauss, transvections. Engendrement de SL et de GL. Applications aux groupes dérivés.

1.2 Action par équivalence

Formes normales : I_r. Invariant : rang.
Topologie des orbites. Ordres de dégénérescence.

2. Action par conjugaison

2.1 Classification : invariants de Frobenius

Position du problème. Endomorphismes cycliques.
AI Réduction de Frobenius.
Invariants de similitude.
Cas de la dimension $2, 3$: classification par π et χ.

2.2 Réduction de Jordan

Réduction des diagonalisables (par Frobenius).
Réduction des nilpotents (par Frobenius).
Réduction de Jordan.

2.3 Topologie

Adhérences des orbites : diagonalisable, nilpotente...

3. Action par congruence

3.1 Action sur les matrices symétriques

Action par congruence. Réduction de Gauss symétrique.
Classification des formes quadratiques sur $\mathbb{C}$, $\mathbb{R}$ et $\mathbb{F}_q$ (car $\neq 2$).
Topologie ? AP Réduction lisse des formes quadratiques et lemme de Morse.

3.2 Actions du groupe orthogonal

Stabilisateur de S_n. Compacité.
Décomposition polaire.
AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Remarque : action sur les endomorphismes normaux.

151 Dimension d'un espace vectoriel (dimension finie). Rang. Exemples et applications.

Développements choisis

— AF Algorithme de Berlekamp.
Pertinence : ★★★★☆
— AH Translatées d'une fonction dérivable.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Dans cette leçon, il est important de présenter les résultats fondateurs de la théorie des espaces vectoriels de dimension finie en ayant une idée de leurs preuves. Ces théorèmes semblent simples car ils ont été très souvent pratiqués, mais leur preuve demande un soin particulier. Il est important de savoir justifier pourquoi un sous-espace vectoriel d'un espace vectoriel de dimension finie est aussi de dimension finie. Le pivot de Gauss ainsi que les diverses notions et caractérisations du rang trouvent leur place dans cette leçon. Les applications sont nombreuses, on peut par exemple évoquer l'existence de polynômes annulateurs ou alors décomposer les isométries en produits de réflexions. S'ils le désirent, les candidats peuvent déterminer des degrés d'extensions dans la théorie des corps ou s'intéresser aux nombres algébriques. Dans un autre registre, il est pertinent d'évoquer la méthode des moindre carrés dans cette leçon, par exemple en faisant ressortir la condition de rang maximal pour garantir l'unicité de la solution et s'orienter vers les techniques de décomposition en valeurs singulières pour le cas général. On peut alors naturellement explorer l'approximation d'une matrice par une suite de matrices de faible rang.

Avis.

Il faut être bien clair sur les résultats de base.

Plan.

1. Espaces vectoriels de dimension finie

1.1 Familles génératrices, bases, libres

Définitions par application linéaire. Dénombrements dans les corps finis. Théorème de la base incomplète. Lemme d'échange (Steinitz). Définition de la dimension. Similitude avec les matroïdes.

1.2 Sous-espaces vectoriels

Décroissance de la dimension. Paradigme de preuve par récurrence. Grassmann et crible faux de Poincaré. Supplémentaire commun. Un zeste de réduction.

1.3 Dualité en dimension finie

AH Translatées d'une fonction dérivable.

1.4 Aspects topologiques

Equivalence des normes en dimension finie. Géométrie des boules. Continuité automatique. Riesz.[Rou14] Carathéodory et contre-exemple en dim infinie δ_n/n.

2. Applications linéaires et rang

2.1 Théorème du rang

Définitions. Théorème du rang. Surjectivité, injectivité. Liens avec les déterminants.

2.2 Aspects algorithmiques

Pivot de Gauss et calcul du rang. Opérations élémentaires. Echelonnement réduit et application pour déterminer si même noyau/même image. AF Algorithme de Berlekamp.

2.3 Calcul différentiel

Théorème de rang constant [Rou14]. Submersion, immersion, semi-continuité du rang. Exemples de sous-variétés.

3. Extensions de corps

3.1 Les techniques vectorielles

Finitude et cardinalité. Application à Wedderburn. Base télescopique. Sur-algèbres de $\mathbb{R}$: $\mathbb{C}$ ou $\mathbb{H}$ [Per98].

3.2 Algébricité, transcendance

Définitions. Clôture algébrique de $\mathbb{Q}$. Polynômes irréductibles, rupture, décomposition. Corps finis. Berlekamp et sa variante probabiliste.

152 Déterminant. Exemples et applications.

Développements choisis

— AA Théorème de Frobenius-Zolotarev & application.
Pertinence : ★★★☆☆
— AJ Suite de polygones.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Dans cette leçon, il faut commencer par définir correctement le déterminant. Il est possible d'entamer la leçon en disant que le sous-espace des formes n-linéaires alternées sur un espace de dimension n est de dimension 1 et, dans ce cas, il est essentiel de savoir le montrer. Le plan doit être cohérent ; si le déterminant n'est défini que sur $\mathbb{R}$ ou $\mathbb{C}$, il est délicat de définir $\det(A - XI_n)$ avec A une matrice carrée. L'interprétation du déterminant comme volume est essentielle. On peut rappeler son rôle dans les formules de changement de variables, par exemple pour des transformations de variables aléatoires. Le calcul explicite est important, mais le jury ne peut se contenter d'un déterminant de Vandermonde ou d'un déterminant circulant. Les opérations élémentaires permettant de calculer des déterminants, avec des illustrations sur des exemples, doivent être présentées. Il est bienvenu d'illustrer la continuité du déterminant par une application, ainsi que son caractère polynomial. Pour les utilisations des propriétés topologiques, on n'omettra pas de préciser le corps de base sur lequel on se place. S'ils le désirent, les candidats peuvent s'intéresser aux calculs de déterminants sur $\mathbb{Z}$ avec des méthodes multimodulaires. Le résultant et les applications simples à l'intersection ensembliste de deux courbes algébriques planes peuvent aussi trouver leur place dans cette leçon pour des candidats ayant une pratique de ces notions.

Avis.

Le déterminant est un outil puissant, tant en algèbre linéaire (c'est ce qu'on fait ressortir dès le début) qu'en analyse. On passera bien sûr par une partie de calculs, qui sera l'occasion de présenter les déterminants célèbres... Il sera important de remarquer que calculer un déterminant donne *existence et unicité* d'une solution, sans avoir à la calculer explicitement !

Plan.

1. Généralités

1.1 Définitions

Forme p-linéaires alternées. Remarque sur l'algèbre extérieure. Déterminant d'une famille de vecteurs dans une base, n-linéarité alternée. Déterminant d'un endomorphisme et d'une matrice. Indépendance de la base et propriétés débiles.

1.2 Déterminant et inversibilité

Solutions d'un système linéaire. Inverse d'une matrice et comatrice. Rouché-Fontené. Systèmes de Cramer. Remarques sur l'effectivité. Inversion de matrices dans $\mathbb{Z}$. Polynôme caractéristique et Cayley-Hamilton.

2. Calculs de déterminants

2.1 Techniques génériques

Opérations sur les lignes et les colonnes. Lien avec les matrices de permutations. Pivot de Gauss. Développement sur ligne et colonnes. Déterminant par blocs.

2.2 Déterminants célèbres

Déterminant de Vandermonde. Interprétation comme problème d'interpolation. Déterminant de Cauchy. Interprétation comme problème d'interpolation.
Déterminant circulant. AJ Suite de polygones.
Déterminants de Smith. Application : AB Théorème de Brauer.

3. Déterminant et groupe linéaire

Définition de SL. Produit semi-direct. Déterminant des matrices de permutations, opérations élémentaires et engendrement de SL par des transvections. Groupes dérivés.
AA Théorème de Frobenius-Zolotarev & application. Application.

4. Un outil en analyse

Continuité du déterminant, GL est ouvert. Différentiabilité du déterminant. Exp det et trace. Application au Wronskien. Jacobien. Changement de variable. Théorème de Brouwer par Milnor (gonnord).
Rappels sur Hilbert. Gram. Application à Müntz. Application à la minimisation.

153 Polynômes d'endomorphisme en dimension finie. Réduction d'un endomorphisme en dimension finie. Applications.

Développements choisis

— AL Décomposition de Dunford effective.
Pertinence : ★★★★★
— AI Réduction de Frobenius.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon ne doit pas être un catalogue de résultats autour de la réduction qui est ici un moyen pour démontrer des théorèmes ; les polynômes d'endomorphismes doivent y occuper une place importante. Il faut consacrer une courte partie de la leçon à l'algèbre $\mathbb{K}[u]$ et connaître sa dimension sans hésitation. Il est ensuite possible de s'intéresser aux propriétés globales de cette algèbre. Les liens entre réduction d'un endomorphisme u et la structure de l'algèbre $\mathbb{K}[u]$ sont importants, tout comme ceux entre les idempotents et la décomposition en somme de sous-espaces caractéristiques. Il faut bien préciser que, dans la réduction de Dunford, les composantes sont des polynômes en l'endomorphisme, et en connaître les conséquences théoriques et pratiques. L'aspect applications est trop souvent négligé. Il est possible, par exemple, de mener l'analyse spectrale de matrices stochastiques. On attend d'un candidat qu'il soit en mesure, pour une matrice simple de justifier la diagonalisabilité et de déterminer un polynôme annulateur (voire minimal). Il est bien sûr important de ne pas faire de confusion entre diverses notions de multiplicité pour une valeur propre λ donnée (algébrique ou géométrique). Enfin, calculer A^k ne nécessite pas, en général, de réduire A (la donnée d'un polynôme annulateur de A suffit souvent).

Plan.

1. Polynômes d'endomorphismes

1.1 Algèbre $\mathbb{K}[u]$ et polynôme minimal

Propriété universelle. Polynôme minimal (qui en général n'est pas irréductible). Degré et dimension de $\mathbb{K}[u]$, majoration par n^2.
Cas de la dimension nulle.
L'inverse est un polynôme. Calcul pratique des puissances d'une matrice.
Exemples de polynômes minimaux : homothéties, symétries, projecteurs, nilpotents.

1.2 Polynôme caractéristique

Définition pour une matrice. Invariance par changement de base.
Petits cas : dimension 2. Lien avec la trace, le déterminant.
Valeurs propres et racines.
Endomorphismes cycliques, caractéristique = minimal. Polynôme conducteur. Matrices compagnons. Théorème de Cayley-Hamilton.

2. Application à l'étude des endomorphismes

Lemme des noyaux

2.1 En scindant les polynômes

Sous-espaces caractéristiques. Trigonalisabilté, diagonalisabilité.
Matrices de permutations. Décomposition de Dunford. Effectivité et se limites.

2.2 Invariants de similitude

Polynôme conducteur. Lemme du polynôme conducteur. Al Réduction de Frobenius. Invariants de similitude. Calculabilité.
Caractérisation en fonction des polynômes minimal et caractéristique en dimension 2 et 3.
Application à la structure du commutant. Bicommutant $\mathcal{C}(\mathcal{C}(u)) = \mathbb{K}[u]$.
Application à la réduction des endomorphismes nilpotents. Réduction de Jordan.

3. Cas réel ou complexe

3.1 Topologie des classes de similitude

Topologie des classes de similitude. Densité et applications.
Théorème de Kronecker. [Gou09a]

3.2 Exponentielle matricielle

Définition. Convergence. C'est un polynôme en A.
L'exponentielle matricielle. Liens avec la trace et le déterminant.
Surjectivité vers $\mathrm{GL}_n(\mathbb{C})$.
[Rom17]

3.3 Méthodes numériques

Disques de Gershgörin. Méthode de la puissance, évoquer Housholder.

157 Endomorphismes trigonalisables. Endomorphismes nilpotents.

Développements choisis

— AL Décomposition de Dunford effective.
Pertinence : ★★★★★
— AK Théorème de Householder & méthodes iteratives.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

L'utilisation des noyaux itérés est fondamentale dans cette leçon, par exemple pour déterminer si deux matrices nilpotentes sont semblables. Il est intéressant de présenter des conditions suffisantes de trigonalisation simultanée ; l'étude des endomorphismes cycliques a toute sa place dans cette leçon. L'étude des nilpotents en dimension 2 débouche naturellement sur des problèmes de quadriques et l'étude sur un corps fini donne lieu à de jolis problèmes de dénombrement. S'ils le désirent, les candidats peuvent aussi présenter la décomposition de Frobenius.

Avis.

Le plan est banal : trigonalisables d'une part, nilpotents de l'autre.

Plan.

1. Endomorphismes trigonalisables

1.1 Généralités

Polynôme irréductible, caractéristique.
Lemme de noyaux.
Trigonalisation simultanée.
Liens avec les valeurs propres.

1.2 Co-trigonalisation

1.3 Trigonalisation et rayon spectral

Rayon spectral. Multiplicativité.
AK Théorème de Householder & méthodes iteratives.

2. Endomorphismes nilpotents

Définition, caractérisations.
Lien avec la trace.
Propriétés de stabilité.
Théorème : Jordan pour les nilpotents. Tableaux de Young.
Dénombrement sur un corps fini.

3. Applications à la réduction

3.1 Décomposition de Dunford

Théorème : Dunford. Application aux EDO.
AL Décomposition de Dunford effective.

3.2 Décomposition de Jordan

Théorème : Jordan (puis pour les trigonalisables).
Evoquer le lien avec Frobenius.

159 Formes linéaires et dualité en dimension finie. Exemples et applications.

Développements choisis

— AI Réduction de Frobenius.
Pertinence : ★★★★☆
— AH Translatées d'une fonction dérivable.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Il est important de bien placer la thématique de la dualité dans cette leçon ; celle-ci permet de mettre en évidence des correspondances entre un morphisme et son morphisme transposé, entre un sous-espace et son orthogonal (canonique), entre les noyaux et les images ou entre les sommes et les intersections. Bon nombre de résultats d'algèbre linéaire se voient dédoublés par cette correspondance. Les liens entre base duale et fonctions de coordonnées doivent être parfaitement connus. Savoir calculer la dimension d'une intersection d'hyperplans via la dualité est important dans cette leçon. L'utilisation des opérations élémentaires sur les lignes et les colonnes permet facilement d'obtenir les équations d'un sous-espace vectoriel ou d'exhiber une base d'une intersection d'hyperplans. Cette leçon peut être traitée sous différents aspects : géométrique, algébrique, topologique ou analytique. Il faut que les développements proposés soient en lien direct avec la leçon. Enfin rappeler que la différentielle d'une fonction à valeurs réelles est une forme linéaire semble incontournable. Il est possible d'illustrer la leçon avec un point de vue probabiliste, en rappelant que la loi d'un vecteur aléatoire X est déterminée par les lois unidimensionnelles de $X \cdot u$ pour tout vecteur u

Avis.

On suit dans les deux premières parties la présentation de [Rom17] (ou [Gou09a]). Certaines de ses preuves sont matricielles et obscures, mais on s'en contentera. Peut-être serait-il possible de faire une présentation plus "catégorielle" en présentant plus tôt la transposée, avec de beaux diagrammes, à réfléchir. Saupoudrer le tout d'exemples choisis (les exercices des livres).

Plan.

1. Formes linéaires

1.1 Formes linéaires

Forme linéaire. Espace dual. Base duale et dimension. Bidual et isomorphisme canonique. Base antéduale (récupérée par cet isomorphisme).
Représentation de Riesz (cas euclidien). Application à l'isomorphisme $A \mapsto (M \mapsto \mathrm{tr}(AM))$.

1.2 Hyperplans

Définition d'un hyperplan. Caractérisation comme espace de codimension 1.
Lemme de factorisation. Application à $\bigcap_{i=1}^{m} \mathrm{Ker}(\varphi_i) \subseteq \mathrm{Ker}(\varphi)$ implique $\varphi \in \mathrm{Vect}(\varphi_1, \dots, \varphi_m)$ [1].
[Rom17]

2. Orthogonalité et transposition

2.1 Orthogonalité

Définitions : orthogonaux $X^{\perp}$ et X°.

1. Partie algébrique des multiplicateurs de Lagrange.

Propriétés 1 : orthogonal de 0, E, E^* ; inclusions ; au Vect.
Propriétés 2 : dimensions, double orthogonal, somme et inclusion.
Application : équation d'un sous-espace de dimension $p \leq n$. Parler du pivot.
[Rom17]
Remarque : orthogonal au sens du produit scalaire (identification de E et E^*).

2.2 Transposition

Définition. Propriétés [Rom17]. Aspects matriciels. Lien avec l'adjoint.

3. Applications de la dualité

3.1 En algèbre

Espaces cycliques. Lemme du polynôme conducteur. AI Réduction de Frobenius. Invariants de similitude d'un endomorphisme.
Formes quadratiques et orthogonalité, dimensions. [CG13], p 173.

3.2 En analyse

Différentielle d'une application $\mathbb{R}^n \to \mathbb{R}$. Gradient. Sous-variétés définies implicitement (système). Intersection des hyperplans et espace tangent. Théorème des extrema liés. [Rou14]
AH Translatées d'une fonction dérivable.

162 Systèmes d'équations linéaires ; opérations élémentaires, aspects algorithmiques et conséquences théoriques.

Développements choisis

— AF Algorithme de Berlekamp.
Pertinence : ★☆☆☆☆
— AK Théorème de Householder & méthodes iteratives.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Dans cette leçon, les techniques liées au simple pivot de Gauss constituent l'essentiel des attendus. Il est impératif de présenter la notion de système échelonné, avec une définition précise et correcte, et de situer l'ensemble dans le contexte de l'algèbre linéaire (sans oublier la dualité). Un point de vue opératoire doit accompagner l'étude théorique et l'intérêt pratique (algorithmique) des méthodes présentées doit être expliqué y compris sur des exemples simples où l'on attend parfois une résolution explicite. S'ils le désirent, les candidats peuvent aussi présenter les relations de dépendance linéaire sur les colonnes d'une matrice échelonnée qui permettent de décrire simplement les orbites de l'action à gauche de $\mathrm{GL}(n,\mathbb{K})$ sur $\mathrm{M}_n(\mathbb{K})$ donnée par $(P,A)\mapsto PA$. De même, des discussions sur la résolution de systèmes sur $\mathbb{Z}$ et la forme normale de Hermite peuvent trouver leur place dans cette leçon. Enfin, il est possible de présenter les décompostions LU et de Choleski , en évaluant le coût de ces méthodes ou encore d'étudier la résolution de l'équation normale associée aux problèmes des moindres carrés et la détermination de la solution de norme minimale par la méthode de décomposition en valeurs singulières.

Avis.

On insistera sur les méthodes numériques indirectes et itératives.

Plan.

1. Théorie des systèmes linéaires

1.1 Généralités

Système d'équations. Matrices. Homogénéité. Inversion.
Intersection d'hyperplans. Equations d'un sous-espace (vectoriel, affine) de dimension $p \leq n$.

1.2 Théorie de Cramer

Déterminant. Système de Cramer et formules associées. Remarque sur la non-mise en pratique.
Théorie générale de Cramer. Fontené-Rouché.

2. Opérations élémentaires, pivot de Gauss

2.1 Opérations élémentaires

Transvections, dilatations, permutations.
Orbites de l'action de $\mathrm{GL}_n(\mathbb{K})$ droite, à gauche. Systèmes échelonnés réduits en lignes/colonnes.
Pivot de Gauss. Action de Steinitz. Matrices équivalentes.

2.2 Mise en place pratique

Pratique du pivot. Complexité. Application au calcul du rang, du déterminant.
AF Algorithme de Berlekamp.

2.3 Conséquences théoriques : générateurs

Engendrement de SL par les transvections. Conjugaison.
Groupe dérivé. AA Théorème de Frobenius-Zolotarev & application.

3. Méthodes directes basées sur une décomposition, méthodes indirectes

3.1 En utilisant une décomposition

Motivation : systèmes triangulaires, systèmes orthogonaux.
Décomposition LU. Décomposition QR. Décomposition de Cholesky.

3.2 Des méthodes itératives

AK Théorème de Householder & méthodes iteratives. Conditions suffisantes de convergente de la relaxation.
Evoquer le gradient de la fonctionnelle quadratique.

170 Formes quadratiques sur un espace vectoriel de dimension finie. Orthogonalité, isotropie. Applications.

Développements choisis

— AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Pertinence : ★★☆☆☆
— AP Réduction lisse des formes quadratiques et lemme de Morse.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Il faut tout d'abord noter que l'intitulé implique implicitement que le candidat ne doit pas se contenter de travailler sur $\mathbb{R}$. Le candidat pourra parler de la classification des formes quadratiques sur le corps des complexes et sur les corps finis. L'algorithme de Gauss doit être énoncé et pouvoir être mis en œuvre sur une forme quadratique simple. Les notions d'isotropie et de cône isotrope sont un aspect important de cette leçon. On pourra rattacher cette notion à la géométrie différentielle.

Avis.

On se placera toujours encaractéristique différente de 2.

Plan.

1. Formes quadratiques

1.1 Formes bilinéaires, formes quadratiques

Forme quadratique associée à une forme bilinéaire. Forme bilinéaire symétrique associée à une forme quadratique. Identités de polarisation. Exemples

1.2 Représentations matricielle & polynomiale

Matrice d'une forme bilinéaire. Action par congruence.
Polynômes homogènes.

1.3 Noyau et rang

Définitions matricielles du noyau, du rang. Interprétations en formes linéaires. Dégénérescence.

2. Orthogonalité, isotropie

2.1 Généralités

Orthogonalité. Orthogonal d'une partie. Dimension. Isotropie (totale, cône).

2.2 Bases orthogonales, réduction de Gauss symétrique

Définition. Existence d'une base orthogonale. Version polynomiale et .

2.3 Espaces hyperboliques

3. Classification des formes quadratiques

3.1 Réduction

Cas complexe. Signature et cas réel. Discriminant et cas des corps finis.
Application : loi de réciprocité quadratique.

3.2 Application en analyse réelle

Hessienne symétrique (Schwartz). Extrema.
AP Réduction lisse des formes quadratiques et lemme de Morse. Petits dessins dans le cas $n = 2$.

4. Étude des groupes orthogonaux

4.1 Engendrement, cas général

4.2 Le cas réel

Cas de compacité de $\mathcal{O}(p, q)$. AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.

181 Barycentres dans un espace affine réel de dimension finie, convexité. Applications.

Développements choisis

— AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Pertinence : ★★★★☆
— AJ Suite de polygones.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Dans cette leçon, la notion de coordonnées barycentriques est incontournable ; des illustrations dans le triangle (coordonnées barycentriques de certains points remarquables) sont envisageables. Il est important de parler d'enveloppe convexe, de points extrémaux, ainsi que des applications qui en résultent. S'ils le désirent, les candidats peuvent aller plus loin en présentant le lemme de Farkas, le théorème de séparation de Hahn-Banach, ou les théorèmes de Helly et de Caratheodory.

Avis.

Le plan Barycentres/Convexité semble inévitable. On peut enrichir la dernière partie en parlant de Hahn-Banach, des points extrémaux, de Krein-Millman, l'appliquer aux polyèdres (parler de sommets, de drapeaux).

Plan.

1. Barycentres

1.1 Généralités

Fonction vectorielle de Leibniz. Bijectivité. Barycentre d'un système de points pondérés.
Homogénéité, associativité, commutativité.
Application : construction d'un barycentre à la main dans le plan (points 2 par 2).
Isobarycentre. Cas du triangle pondéré.

Sous-espace affine $\iff$ stable par barycentre (SEA engendré).
Application affine $\iff$ préserve les barycentres. Préserve aussi les SEA engendrés.

1.2 Coordonnées barycentriques

Repère affine, définitions équivalentes : barycentres, base vectorielle.
Coordonnées barycentriques d'un point dans un repère (proportionnalité, normalisation).
Exemples.

1.3 Barycentres dans le plan

Coordonnées barycentriques des points remarquables du triangle.
Théorèmes dans le plan : Menelaus, Ceva.
AJ Suite de polygones.

2. Convexité

2.1 Définitions

Partie convexe, définitions équivalentes.

Exemples : ensembles convexes de $\mathbb{R}$, boules (et "réciproque" ?). Intersection. Parties étoilées. Inégalité des accroissements finis.
Préservation de la convexité par les applications affines.

2.2 Enveloppe convexe

Définitions. Gauss-Luca. Enveloppe convexe d'un borné, d'un ouvert, d'un fermé.
Théorème de Carathéodory. Enveloppe convexe d'un compact (dim finie). Contre-exemple en dimension infinie : dans $\ell^2(\mathbb{N})$ avec $\{0\} \cup \{\chi_n \mid n \geq 0\}$ compacte, mais son enveloppe convexe n'est pas fermée.
AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Théorème de Helly.

2.3 Points extrémaux

Points extrémaux d'un convexe.
Hahn-Banach géométrique. Hyperplan d'appui. BERGER
Application à Krein-Millman. Isométries des polyèdres.

182 Applications des nombres complexes à la géométrie.

Développements choisis

— AJ Suite de polygones.
Pertinence : ★★★★☆
— AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Cette leçon ne doit pas rester au niveau de la classe de Terminale. L'étude des inversions est tout à fait appropriée, en particulier la possibilité de ramener un cercle à une droite et inversement ; la formule de Ptolémée illustre bien l'utilisation de cet outil. On peut parler des suites définies par récurrence par une homographie et leur lien avec la réduction dans $\mathrm{SL}_2(\mathbb{C})$. S'ils le désirent, les candidats peuvent aussi étudier l'exponentielle complexe et les homographies de la sphère de Riemann. La réalisation du groupe SU_2 dans le corps des quaternions et ses applications peuvent trouver leur place dans la leçon. Il est possible de présenter les similitudes, les homographies et le birapport.

Avis.

La Grande Idée des nombres complexes appliqués en géométrie plane : ils permettent de réduire tous les problèmes géométriques à des questions de calcul pur sur les nombres [1]. Certes ce n'est pas toujours très agréable à faire, mais cette méthode est néanmoins générique.
Après une présentation des concepts usuels, on établit (avec difficulté croissante) des expressions complexes pour certaines transformations du plan. On ne mentionnera pas explicitement la notion de droite projective, pas plus qu'on ne parle d'homographies.
Enfin, on regarde ce que les quaternions peuvent apporter à la géométrie de l'espace. Ils sont justifiés à la fois comme extension de $\mathbb{C}$ et par leur lien avec $\mathrm{SU}_2(\mathbb{C})$.

Plan.

1. Géométrie plane

1.1 Le plan complexe

Bijection en $\mathbb{C}$ et $\mathbb{R}^2$. Produit scalaire.
Critère d'alignement de trois points,
Exponentielle complexe. Nombre π. Module, argument. Formules trigonométriques.
Angle de deux vecteurs. Critère d'alignement.

1.2 Utilisation calculatoire des nombres complexes

Théorèmes de Morley (intersection des trisectrices). Théorème de Napoléon. Point de Fermat. Gauss-Luca. AJ Suite de polygones.

2. Quelques transformations du plan

2.1 Transformations affines du plan

— Groupe $\mathrm{SO}_2(\mathbb{R})$. Isomorphisme à $\mathbb{U}$. Forme analytique $z \mapsto \mathrm{e}^{\mathrm{i}\theta} z$.
— Groupe $\mathrm{O}_2(\mathbb{R})$. Suite exacte $\mathrm{SO}_2 \to \mathrm{O}_2 \to_{\det} \{\pm 1\}$. Relèvement $\mathrm{SO}_2(\mathbb{R}) \rtimes \mathbb{Z}/2\mathbb{Z}$ par le conjugué $\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$. Forme analytique $z \mapsto \mathrm{e}^{\mathrm{i}\theta} z$ ou $z \mapsto \mathrm{e}^{\mathrm{i}\theta} \overline{z}$.

1. Et celles-ci, on peut les décider, cf décidabilité de la théorie des corps algébriquement clos…

— Groupe $\mathrm{Is}(\mathbb{R}^2)$. Suite exacte $\mathrm{T} \to \mathrm{Is}(\mathbb{R}^2) \to_{\text{partie lin.}} \mathrm{O}_2(\mathbb{R})$ et relèvement $\mathrm{T} \rtimes \mathrm{O}_2(\mathbb{R})$ centré en 0. Forme analytique $z \mapsto \mathrm{e}^{\mathrm{i}\,\theta} z + b$ ou $z \mapsto \mathrm{e}^{\mathrm{i}\,\theta} \overline{z} + b$. Cas de $\mathrm{Is}^+(\mathbb{R}^2)$ avec $z \mapsto \mathrm{e}^{\mathrm{i}\,\theta} z + b$.
— Groupe des similitudes. Forme analytique $z \mapsto az + b$ ou $z \mapsto a\overline{z} + b$ (multiplier par une isométrie dans les précédents). Elements caractéristiques : rapport, angle, centre.

Similitudes d'un polygone régulier. Cas des triangles équilatéraux. EIDEN Application au point de Fermat.

2.2 Inversions

Inversion de rayon R et de centre 0 : formulation géométrique $\overline{OM} \times \overline{Oi(M)} = R^2$. Formulation analytique $z \mapsto \frac{R^2}{\overline{z}}$ (ne pas oublier la barre). Composition.
Transformations des droites et des cercles :

— droite passant par $0 \to$ invariante ;
— cercle ne passant par par $0 \to$ cercle ne passant par par 0 ;
— cercle passant par $0 \leftrightarrow$ droite ne passant par par 0.

Application au théorème de Ptolémée. EIDEN

3. Quaternions et géométrie de l'espace

Algèbre $\mathbb{H}$ des quaternions. Conjugué, norme. Propriété de corps gauche. Théorème de Frobenius. Centre de $\mathbb{H}$. Aspects matriciels. Isomorphisme $G \simeq \mathrm{SU}_2(\mathbb{C})$. [Per98]
Générateurs de $\mathrm{O}_n(\mathbb{R})$. AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.Remarque sur l'utilisation en pratique.

183 Utilisation des groupes en géométrie.

Développements choisis

— AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.
Pertinence : ★★★★☆
— AE Groupes d'isométries du cube et du tétraèdre. (NR)
Pertinence : ★★★☆☆

Rapport 2018 du jury.

C'est une leçon dans laquelle on s'attend à trouver des utilisations variées. On s'attend à ce que soient définis différents groupes de transformations (isométries, déplacements, similitudes, translations) et à voir résolus des problèmes géométriques par des méthodes consistant à composer des transformations. De plus, les actions de groupes sur la géométrie permettent aussi de dégager des invariants essentiels (angle, birapport, excentricité d'une conique). Les groupes d'isométries d'une figure sont incontournables.

Avis.

Une leçon assez subtile, puisqu'il ne s'agit pas de "groupes et géométrie" [1].

Plan.

1. Géométrie affine

1.1 Espaces affines

1.2 Groupe affine

2. Etude en dimension 2 et 3

2.1 Isométries du plan et nombres complexes

Nombres complexes.
Applications.

2.2 Dimension 3

Classification des rotations, des isométries en dimension 3.
AC $\mathrm{SO}_3(\mathbb{R})$ et les quaternions.. Applications pratiques.

3. Isométries préservant une partie

3.1 Généralités

3.2 Polygones convexes réguliers

Parler des groupes diédraux.

3.3 Solides platoniciens

AE Groupes d'isométries du cube et du tétraèdre. (NR)

1. On pensera à [CG13]

190 Méthodes combinatoires, problèmes de dénombrement.

Développements choisis

— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★★★★
— AW Nombres de Catalan.
Pertinence : ★★★★★

Rapport 2018 du jury.

Il est nécessaire de dégager clairement différentes méthodes de dénombrement et de les illustrer d'exemples significatifs. De nombreux domaines de mathématiques sont concernés par des problèmes de dénombrement, cet aspect varié du thème de la leçon doit être mis en avant. L'utilisation de séries génératrices est un outil puissant pour le calcul de certains cardinaux. De plus, il est naturel de calculer des cardinaux classiques et certaines probabilités. Il est important de connaître l'interprétation ensembliste de la somme des coefficients binomiaux et ne pas se contenter d'une justification par le binôme de Newton. L'introduction des corps finis (même en se limitant aux cardinaux premiers) permet de créer un lien avec l'algèbre linéaire. Les actions de groupes peuvent également conduire à des résultats remarquables. S'ils le désirent, les candidats peuvent aussi présenter des applications de la formule d'inversion de Möbius ou de la formule de Burnside. Des candidats ayant un bagage probabiliste pourront explorer le champ des permutations aléatoires, en présentant des algorithmes pour générer la loi uniforme sur le groupe symétrique $\mathfrak{S}_n$ et analyser certaines propriétés de cette loi uniforme (points fixes, cycles, limite $n \to +\infty$).

Avis.

Beaucoup d'idées à développer. On commencera par de la "combinatoire de sup" avec les notions usuelles sur les cardinaux. Les séries génératrices constituent un formidable outil de calcul dans ces circonstances. Dans la partie 3, les techniques de dénombrement s'orientent naturellement vers des résultats plus puissants, mais aussi plus spécifiques, lorsqu'on dénombre des ensembles enrichis d'une structure : groupes, corps, espaces vectoriels. Ce qui permet (entre autres) de les mieux comprendre. Ensuite, on s'intéresse à des questions liées aux anneaux $\mathbb{Z}/n\mathbb{Z}$, avec l'indicatrice d'Euler et le symbole de Legendre. On effleure les "vraies" applications pratiques en algorithmique des nombres (sans les mentionner explicitement).

Plan.

1. Ensembles finis, cardinaux, bijections

1.1 Ensembles finis

Définition. Cardinal de l'union, de l'intersection, du produit, des fonctions de E dans F.Crible de Poincaré. Application aux nombre d'entiers qui ne sont pas divisibles par un ensemble de facteurs donnés.

1.2 Combinaisons, arrangements

Coefficients binomiaux (définition combinatoire). Formule de Pascal. Expression factorielle.
Binôme de Newton. Formule de Vandermonde.
Nombre d'arrangements (injections).

2. Utilisation de séries génératrices

Mots bien parenthésés. AW Nombres de Catalan. Nombres de Bell.
[FGN01a], p 10.

3. Dénombrement d'objets algébriques finis

3.1 Techniques de théorie des groupes

Formule des classes. Théorème de Lagrange. Application aux carrés de $\mathbb{F}_q$.
Cas plus simple où le groupe agit transitivement.
Action par conjugaison dans un p-groupe. Centre des p-groupes. Groupes d'ordre p, p^2.
Formule de Burnside. Permutations aléatoires (nb moyens de PF).

3.2 Corps et espaces vectoriels finis

Cardinal de $\mathrm{GL}_n(\mathbb{F}_q)$, de $\mathrm{SL}_n(\mathbb{F}_q)$. Application : cardinal du cône nilpotent.
Les matrices triangulaires unitaires sont un p-sylow de $\mathrm{GL}_n(\mathbb{F}_p)$.
Application : tout p-groupe possède un p-Sylow (théorème de Sylow 1).

4. Questions arithmétiques

4.1 Autour de l'indicatrice d'Euler

Indicatrice d'Euler. $\varphi(n) = \prod(1 - \frac{1}{p_i})$. $n = \sum_{d|n} \varphi(d)$. Application à la cyclicité des groupes multiplicatifs de corps.
Fonction de Möbius. Inversion.
Application à $\varphi(n) = \sum_{d|n} d\mu\left(\frac{n}{d}\right)$.
Application à AD Ordre moyen de l'indicatrice d'Euler.
Application au dénombrement des polynômes irréductibles sur $\mathbb{F}_q$.

4.2 Autour du symbole de Legendre

Définition. Caractère morphique (et unicité du morphisme).
Nombre de solutions de $ax^2 = 1$.
AA Théorème de Frobenius-Zolotarev & application.
Loi de réciprocité quadratique et lois complémentaires.

CHAPITRE 4

LEÇONS D'ANALYSE

Alors l'enfer se tut.
L'affreux bouillonnement de ces grands lacs de flammes,
Les grincements de dents de ses tourmenteurs d'âmes,
Se firent seuls entendre ; et dans ses profondeurs,
Un mystère d'horreur s'accomplit.

La damnation de Faust, Hector Berlioz

203 Utilisation de la notion de compacité.

Développements choisis

— AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.
Pertinence : ★★★☆☆
— AQ Théorème de Hadamard-Lévy.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Il est important de ne pas concentrer la leçon sur la compacité en général (confusion entre utilisation de la notion compacité et notion de compacité), et de se concentrer en priorité sur le cadre métrique. Néanmoins, on attend des candidats d'avoir une vision synthétique de la compacité. Des exemples d'applications comme le théorème de Heine et le théorème de Rolle doivent y figurer et leur démonstration être connue. Par ailleurs, le candidat doit savoir quand la boule unité d'un espace vectoriel normé est compacte. Des exemples significatifs d'utilisation comme le théorème de Stone-Weierstrass (version qui utilise pertinemment la compacité), des théorèmes de point fixe, voire l'étude qualitative d'équations différentielles, sont tout-à fait envisageables. Le rôle de la compacité pour des problèmes d'existence d'extrema mériterait d'être davantage étudié. On peut penser comme application à la diagonalisation des matrices symétriques à coefficients réels. Pour aller plus loin, les familles normales de fonctions holomorphes fournissent des exemples fondamentaux d'utilisation de la compacité. Les opérateurs auto-adjoints compacts sur l'espace de Hilbert relèvent également de cette leçon, et on pourra développer l'analyse de leurs propriétés spectrales.

Avis.

Une leçon intéressante, sur laquelle on peut faire de belles choses sans être un expert en analyse fonctionnelle. Le jury attend une présentation synthétique de la compacité, mais surtout des applications très convaincantes. On dira naïvement que "pour faire des théorèmes, il faut des hypothèses" [1], la compacité est une hypothèse particulièrement pertinente et utile, c'est ce qu'il faut savoir mettre en valeur. On se restreindra au cadre des espaces métriques.

Plan.

1. Généralités

Définition : espace métrique compact (Borel-Lebesgue), sous-partie compacte.
Fermeture, espace précompact.
Théorème : Bolzano-Weierstrass. Précompact + complet.
Théorème : Tychonoff (dénombrable). Application : compacité du calcul propositionnel.

2. Compacité et fonctions continues

2.1 Fonctions continues sur un compact

Propriété : image d'un compact par une application continue. Corollaire : homéomorphismes.
Propriétés : bornée et atteint ses bornes.
Théorème : Rolle, accroissements finis.
Fonctions coercives.
Théorème : Heine.
Application : gradient à pas optimal.

1. De l'avis d'A. Leclaire

2.2 Familles de fonctions continues

Equicontinuité. Convergence uniforme. Ascoli.
Théorème : Stone-Weierstrass (réél, complexe).

3. Utilisations de la compacité en analyse

3.1 Pour la dimension

Propriété : compacts en dimension finie = fermés bornés.
Equivalence des normes.
Théorème de Riesz.

3.2 Pour la recherche de points fixe

Propriété : une fonction strictement 1-lip sur un compact admet un point fixe.
Théorème : Brouwer (sur la boule). Généralisation.
Théorème : Schauder.
AM Sous-groupes compacts de $\mathrm{GL}_n(\mathbb{R})$.

3.3 Aux équations différentielles

Théorème : Cauchy-Peano-Arzela.
Théorème des bouts.
AQ Théorème de Hadamard-Lévy.

208 Espaces vectoriels normés, applications linéaires continues. Exemples.

Développements choisis

— AK Théorème de Householder & méthodes iteratives.
Pertinence : ★★★★★
— AT Théorème de Banach-Steinhaus & séries de Fourier divergentes.
Pertinence : ★★★★★

Rapport 2018 du jury.

Une telle leçon doit bien sûr contenir beaucoup d'illustrations et d'exemples, notamment avec quelques calculs élémentaires de normes subordonnées (notion qui met en difficulté un trop grand nombre de candidats). Le lien avec la convergence des suites du type $X_{n+1} = AX_n$ doit être connu. Lors du choix de ceux-ci (le jury n'attend pas une liste encyclopédique), le candidat veillera à ne pas mentionner des exemples pour lesquels il n'a aucune idée de leur pertinence et à ne pas se lancer dans des développements trop sophistiqués. La justification de la compacité de la boule unité en dimension finie doit être maîtrisée. Il faut savoir énoncer le théorème de Riesz sur la compacité de la boule unité fermée d'un espace vectoriel normé. Le théorème d'équivalence des normes en dimension finie, ou le caractère fermé de tout sous-espace de dimension finie d'un espace normé, sont des résultats fondamentaux à propos desquels les candidats doivent se garder des cercles vicieux. A contrario, des exemples d'espaces vectoriels normés de dimension infinie ont leur place dans cette leçon et il faut connaître quelques exemples de normes usuelles non équivalentes, notamment sur des espaces de suites ou des espaces de fonctions et également d'applications linéaires qui ne sont pas continues.

Avis.

Il est important de bien poser le cadre par des définitions. On étudie ensuite le cas de la dimension finie (essentiellement l'équivalence des normes et Riesz), puis on s'intéresse à des espaces naturels en analyse fonctionnelle : les Banach, et les Hilbert. Ne pas hésiter à mettre de nombreux exemples et contre-exemples (ne serait-ce qu'entre dimension finie et infinie).

Plan.

1. Généralités

1.1 Espaces vectoriels normés

Définition : norme, distance, équivalence de normes.
Liens avec la topologie.

1.2 Applications linéaires continues

Propriété : caractérisations de la continuité.
Proposition-définition : norme subordonnée.

1.3 Spécificités de la dimension finie

Equivalence des normes. Continuité automatique en dimension finie.
Théorème de Riesz.
Normes subordonnées aux matrices. Rayon spectral.
AK Théorème de Householder & méthodes iteratives.

2. Espaces de Banach

2.1 Généralités

Définition : espace de Banach.
Propriété : convergence normale des séries.
Théorème : point fixe de Picard. Application à Cauchy-Lipschitz.

2.2 Théorie de Baire et applications

Lemme de baire.
AT Théorème de Banach-Steinhaus & séries de Fourier divergentes.
Théorème : application ouverte. Isomorphisme.
Théorème : graphe fermé [1].

3. Espaces de Hilbert

Produit scalaire. Norme associée. Identité du parallélogramme.
Définition : espace de Hilbert. Exemples.
Projection sur un convexe fermé.
Base hilbertienne.
Théorème de Riesz-Fréchet.

1. On en voit une application dans les Hilbert.

214 Théorème d'inversion locale, théorème des fonctions implicites. Exemples et applications en analyse et en géométrie.

Développements choisis

— AQ Théorème de Hadamard-Lévy.
Pertinence : ★★★★★
— AP Réduction lisse des formes quadratiques et lemme de Morse.
Pertinence : ★★★★★

Rapport 2018 du jury.

Il s'agit d'une leçon qui exige une bonne maîtrise du calcul différentiel. Même si le candidat ne propose pas ces thèmes en développement, on est en droit d'attendre de lui des idées de démonstration des deux théorèmes fondamentaux qui donnent son intitulé à la leçon. Il est indispensable de savoir mettre en pratique le théorème des fonctions implicites au moins dans le cas de deux variables réelles. On attend des applications en géométrie différentielle notamment dans la formulation des multiplicateurs de Lagrange. Plusieurs inégalités classiques de l'analyse peuvent se démontrer avec ce point de vue : Hölder, Carleman, Hadamard,... En ce qui concerne la preuve du théorème des extrema liés, la présentation de la preuve par raisonnement « sous-matriciel » est souvent obscure ; on privilégiera si possible une présentation géométrique s'appuyant sur l'espace tangent. Pour aller plus loin, l'introduction des sous-variétés est naturelle dans cette leçon. Il s'agit aussi d'agrémenter cette leçon d'exemples et d'applications en géométrie, sur les courbes et les surfaces.

Avis.

Comme dans la leçon suivante, on se place dans $\mathbb{R}^n$ et pas dans un espace de Banach quelconque. Attention celle-ci est moins générale que la 215, donc a priori plus délicate.

1. Fonctions inverses

1.1 Inversion locale

Difféomorphisme (local). Inversion locale. Exemples.
Application : Brouwer.
Exemples en algèbre : autour de l'exponentielle.

1.2 Inversion globale

Inversion globale. AQ Théorème de Hadamard-Lévy..

1.3 Changement de coordonnées

Changement local de coordonnées. Changement de variables. Brouwer.
Immersion, submersion. Formes normales locales. Théorème de rang constant. AP Réduction lisse des formes quadratiques et lemme de Morse.

2. Fonctions implicites

Enoncé. Exemples. Equivalence avec le TIL.

3. Sous-variétés

Théorème de caractérisation des sous-variétés. Immersions. Exemples avec des groupes de Lie.
Espace tangent. Exemples. Extrema liés et applications pratiques.

215 Applications différentiables sur un ouvert de $\mathbb{R}^n$. Exemples et applications.

Développements choisis

— AQ Théorème de Hadamard-Lévy.
Pertinence : ★★★★★
— AP Réduction lisse des formes quadratiques et lemme de Morse.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon requiert une bonne maîtrise de la notion de différentielle première et de son lien avec les dérivées partielles, mais aussi de ce qui les distingue. On doit pouvoir mettre en pratique le théorème de différentiation composée pour calculer des dérivées partielles de fonctions composées dans des situations simples (par exemple le laplacien en coordonnées polaires). La différentiation à l'ordre 2 est attendue, notamment pour les applications classiques quant à l'existence d'extrema locaux. On peut aussi faire figurer dans cette leçon la différentielle d'applications issues de l'algèbre linéaire (ou multilinéaire). La méthode du gradient pour la minimisation de la fonctionnelle $\frac{1}{2}(Ax,x) - (b,x)$est une matrice symétrique définie positive, conduit à des calculs de différentielles qui doivent être acquis par tout candidat. Pour aller plus loin, l'exponentielle matricielle est une ouverture pertinente. D'autres thèmes issus de la leçon 214 trouvent aussi leur place ici.

Avis.

On se place sur $\mathbb{R}^n$ et ce n'est pas plus mal. Ne pas oublier de donner des exemples et des contre-exemples.

1. Différentiabilité

1.1 Application différentiable

Définition. Exemple du cas réel, d'une application linéaire, bilinéaire. Différentiable implique continu. Différentielle de la somme, du produit, de la composition. Classe $\mathcal{C}^1$.
[Gou09b], p 303 [Rou14], p 45

1.2 Dérivée selon un vecteur

Définitions. Liens avec la différentiabilité. Matrice jacobienne. Gradient.
[Gou09b], p 304

1.3 Inégalité de la moyenne

Egalité des accroissements finis dans $\mathbb{R}$, inégalité. Attention aux hypothèses de convexité, connexité.
[Rou14], p 103

2. Inversion locale et fonctions implicites

Théorème d'inversion locale. Théorème d'inversion globale. Application propre. AQ Théorème de Hadamard-Lévy..
Changement de coordonnées locales. Immersion, submersion. Formes normales, théorème de rang constant.
Formule de changement de variables. Application au lemme de non-rétraction $\mathcal{C}^1$, puis Brouwer $\mathcal{C}^1$, Brouwer tout court. Généralisation à un ouvert quelconque.
Fonctions implicites. Equivalence avec le TIL. Exemples.

3. Différentielles d'ordre supérieur

Définitions. Formules de Taylor. Théorème de Schwarz.
Formules de Taylor : Young, reste intégrale, Taylor-Lagrange. AP Réduction lisse des formes quadratiques et lemme de Morse..

4. Recherche d'extrema

Extrema libres. Points critiques. Convexité. Gradient à pas optimal. Application à la résolution de systèmes linéaires. Extrema liés.

218 Applications des formules de Taylor.

Développements choisis

— AV Méthode de Laplace et formule de Stirling.
Pertinence : ★★★☆☆
— AP Réduction lisse des formes quadratiques et lemme de Morse.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Il faut connaître les formules de Taylor et certains développements très classiques et surtout être capable de faire la différence entre les formules et de maîtriser leurs champs d'application. En général, le développement de Taylor d'une fonction comprend un terme de reste qu'il est crucial de savoir analyser. Le candidat doit pouvoir justifier les différentes formules de Taylor proposées ainsi que leur intérêt. Le jury s'inquiète des trop nombreux candidats qui ne savent pas expliquer clairement ce que signifient les notations o ou O qu'ils utilisent. De plus la différence entre l'existence d'un développement limité à l'ordre deux et l'existence de dérivée seconde doit être connue. On peut aussi montrer comment les formules de Taylor permettent d'établir le caractère développable en série entière (ou analytique) d'une fonction dont on contrôle les dérivées successives. Pour aller plus loin, on peut mentionner des applications en algèbre bilinéaire (lemme de Morse), en géométrie (étude locale au voisinage des points stationnaires pour les courbes et des points critiques pour la recherche d'extrema) et, même si c'est plus anecdotique, en probabilités (théorème central limite). On peut aussi penser à la méthode de Laplace, du col, de la phase stationnaire ou aux inégalités contrôlant les dérivées intermédiaires lorsque f et sa dérivée n-ième sont bornées, ou encore à l'analyse de méthodes d'intégration numérique ou l'étude de consistance de l'approximation de $\frac{\partial^2}{\partial x^2}$ par différences finies. On soignera particulièrement le choix des développements.

Avis.

L'énoncé des formules de Taylor est le centre de cette leçon. On n'oubliera pas le cas multidimensionnel, ni le lien avec les séries entières.

1. Formules de Taylor pour la variable réelle

1.1 Formules de Taylor

Enoncés.

1.2 Liens avec la régularité

Exemples et contre-exemples.

2. Applications en analyse

2.1 Liens avec les séries entières

Exemples et contre-exemples.
Utilisation du reste pour majorer.

2.2 Obtention de développements limités

AV Méthode de Laplace et formule de Stirling.
Développements que l'on itère, etc.

2.3 Résolution d'équations

Méthode de Newton.

3. Formules de Taylor des fonctions de plusieurs variables

3.1 Enoncés

Accroissements finis.
Théorèmes : Taylor-Lagrange, reste intégral, Taylor-Young.

3.2 Recherche d'extrema

Théorème : conditions nécessaires d'extrema.
AP Réduction lisse des formes quadratiques et lemme de Morse..
Lemme de Hadamard, lemme de division.
[Rou14], p 282.

219 Extremums : existence, caractérisation, recherche. Exemples et applications.

Développements choisis

— AP Réduction lisse des formes quadratiques et lemme de Morse.
Pertinence : ★★★★☆
— AR Gradient à pas optimal. (NR)
Pertinence : ★★★★★

Rapport 2018 du jury.

Comme souvent en analyse, il est tout à fait opportun d'illustrer dans cette leçon un exemple ou un raisonnement à l'aide d'un dessin. Il faut savoir faire la distinction entre propriétés locales (caractérisation d'un extremum) et globales (existence par compacité, par exemple). Dans le cas important des fonctions convexes, un minimum local est également global. Les applications de la minimisation des fonctions convexes sont nombreuses et elles peuvent illustrer cette leçon. L'étude des algorithmes de recherche d'extremums y a toute sa place : méthode du gradient et analyse de sa convergence, méthode à pas optimal,... Le cas particulier des fonctionnelles sur $\mathbb{R}^n$ de la forme $\frac{1}{2}(Ax,x)-(b-x)$ où A est une matrice symétrique définie positive, ne devrait pas poser de difficultés (la coercivité de la fonctionnelle pose problème à de nombreux candidats). Les problèmes de minimisation sous contrainte amènent à faire le lien avec les extrema liés et la notion de multiplicateur de Lagrange. À ce sujet, une preuve géométrique des extrema liés sera fortement valorisée par rapport à une preuve algébrique, formelle et souvent mal maîtrisée. Enfin, la question de la résolution de l'équation d'Euler-Lagrange peut donner l'opportunité de mentionner la méthode de Newton. Les candidats pourraient aussi être amenés à évoquer les problèmes de type moindres carrés, ou, dans un autre registre, le principe du maximum et ses applications.

Avis.

La recherche d'extrema est toute naturelle dans les problèmes d'optimisation. Donner des *conditions nécessaires d'extrema* permet de limiter leur recherche, du moins quand on on procède par un calcul manuel. On donnera également des conditions nécessaires à la convergence des méthodes numériques.

Plan.

1. Extrema : généralités

1.1 Définitions

Définition : maximum local, global.
Exemple débile.

1.2 Existence et compacité

Propriétés : image d'un compact.
Application : équivalence des normes en dimension finie.
Propriété : fonctions coercives.
Application : distance à un SEV.

2. Localisation en calcul différentiel

2.1 Extrema libres

Propriété : extrema $\Rightarrow$ critique (sur un ouvert).
Application : Rolle, accroissements finis.
Lien avec la différentielle seconde.
AP Réduction lisse des formes quadratiques et lemme de Morse.
Cas de la dimension 2.

2.2 Extrema liés

Théorème : extrema liés.
Application : caractérisation de $\mathrm{SO}_n(\mathbb{R})$.
Point de Fermat.

3. Convexité et extrema globaux

3.1 Convexité

Caractérisation des extrema.

3.2 Application en analyse Hilbertienne

Proposition : Proposition sur un convexe fermé.

4. Optimisation numérique

4.1 Méthodes de gradient

Idée générale.
AR Gradient à pas optimal. (NR)
Application : résolution de systèmes linéaires.

4.2 Méthode de Newton

Idée : annuler la dérivée.

220 Equations différentielles $X' = f(t, X)$. Exemple d'étude des solutions en dimension 1 et 2.

Développements choisis

— AQ Théorème de Hadamard-Lévy.
— AS Théorème de Liapounov. (NR)
Pertinence : ★★★★★

Rapport 2018 du jury.

C'est l'occasion de rappeler une nouvelle fois que le jury s'alarme des nombreux défauts de maîtrise du théorème de Cauchy-Lipschitz. Il est regrettable de voir des candidats ne connaître qu'un énoncé pour les fonctions globalement lipschitziennes ou, plus grave, mélanger les conditions sur la variable de temps et d'état. La notion de solution maximale et le théorème de sortie de tout compact sont nécessaires. Bien évidemment, le jury attend des exemples d'équations différentielles non linéaires. Le lemme de Grönwall semble trouver toute sa place dans cette leçon mais est trop rarement énoncé. L'utilisation du théorème de Cauchy-Lipschitz doit pouvoir être mise en œuvre sur des exemples concrets. Les études qualitatives doivent être préparées et soignées. Pour les équations autonomes, la notion de point d'équilibre permet des illustrations de bon goût comme par exemple les petites oscillations du pendule. Trop peu de candidats pensent à tracer et discuter des portraits de phase alors que le sujet y invite clairement. Il est possible d'évoquer les problématiques de l'approximation numérique dans cette leçon en présentant le point de vue du schéma d'Euler . On peut aller jusqu'à aborder la notion de problèmes raides et la conception de schémas implicites pour autant que le candidat ait une maîtrise convenable de ces questions.

Avis.

Le plan est canonique : généralités (structure, existence de solutions), recherche de solutions, étude qualitative.

Plan.

1. Equations différentielles

1.1 Généralités

Définition : EDO, problème de Cauchy, solution.
Liens entre système et équation.

1.2 Existence et unicité des solutions

Théorème : Cauchy-Lipschitz. Remarque : Cauchy-Peano (et contre-exemple à l'unicité).
Lemme de Gronwall.
Théorème : sortie de tout compact (cas simple).
Conséquences : globalité (notamment le cas linéaire).

2. Résolution d'équations particulières

2.1 Ordre 1, ordre 2

2.2 Equations à coefficients constants

3. Etude qualitative des systèmes autonomes

3.1 Généralités

Définition : champ de vecteurs, flot.
Propriétés : existence, continuité du flot.
Définition : trajectoire, absence de coupe.
Définition : point critique, stabilité.
AQ Théorème de Hadamard-Lévy.

3.2 Systèmes linéaires à coefficients constants

Exemples de dimension 1 et 2.
AS Théorème de Liapounov. (NR).

3.3 D'autres exemples

Système proie-prédateur de Lotka-Volterra.
Petites oscillations du pendule.

221 Equations différentielles linéaires. Systèmes d'équations différentielles linéaires. Exemples et applications.

Développements choisis

— AH Translatées d'une fonction dérivable.
— AS Théorème de Liapounov. (NR)
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Le jury attend d'un candidat qu'il sache déterminer rigoureusement la dimension de l'espace vectoriel des solutions. Le cas des systèmes à cœfficients constants fait appel à la réduction des matrices qui doit être connue et pratiquée. Le jury attend qu'un candidat puisse mettre en œuvre la méthode de variation des constantes pour résoudre une équation différentielle linéaire d'ordre 2 simple (à coefficients constants par exemple) avec second membre. L'utilisation des exponentielles de matrices a toute sa place ici. Les problématiques de stabilité des solutions et le lien avec l'analyse spectrale devraient être exploitées. Le théorème de Cauchy-Lipschitz linéaire constitue un exemple de développement pertinent pour cette leçon. Les résultats autour du comportement des solutions, ou de leurs zéros, de certaines équations linéaires d'ordre 2 (Sturm, Hill-Mathieu,...) sont aussi d'autres possibilités.

Avis.

Ne pas confondre *linéaire* et *linéaire à coefficients constants* : le premier cas est assez général, et le second très restreint. Le plan est canonique : généralités (structure, existence de solutions), recherche de solutions, étude qualitative.

1. Généralités

1.1 Définitions

Définition : équations différentielles linéaires, ordre, liens avec les systèmes.

1.2 Existence et unicité des solutions

Théorème : sortie de tout compact.
Théorème : Cauchy-Lipschitz linéaire.
AH Translatées d'une fonction dérivable.
Structure de l'espace des solutions.

2. Résolution des équations différentielles linéaires

2.1 A coefficients constants

Equations homogène, exponentielle matricielle. Croissance avec Dunford.

2.2 Wronskien

Variation de la constante, Wronskien.

2.3 Autres méthodes

Développements en série entière, équations non-linéaires qui se ramènet

3. Etude qualitative

3.1 Résultats généraux

Flot, continuité.
AS Théorème de Liapounov. (NR)

3.2 Exemples en dimension 1 et 2.

Les dessins.

223 Suites numériques. Convergence, valeur d'adhérence. Exemples et applications.

Développements choisis

— AO Théorème de Sarkovski.
Pertinence : ★★★★☆
— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★☆☆☆

Rapport 2018 du jury.

Cette leçon permet souvent aux candidats de s'exprimer. Il ne faut pas négliger les suites de nombres complexes. Le théorème de Bolzano-Weierstrass doit être cité et le candidat doit être capable d'en donner une démonstration. On attend des candidats qu'ils parlent des limites inférieure et supérieure d'une suite réelle bornée, et qu'ils en maîtrisent le concept. Les procédés de sommation peuvent être éventuellement évoqués mais le théorème de Cesàro doit être mentionné et sa preuve maîtrisée par tout candidat à l'agrégation. Les résultats autour des sous-groupes additifs de R permettent d'exhiber des suites denses remarquables et l'ensemble constitue un joli thème. Des thèmes de la leçon 226 peuvent également se retrouver dans cette leçon. Pour aller plus loin, un développement autour de l'équirépartition est tout à fait envisageable. La méthode de Newton peut aussi illustrer la notion de vitesse de convergence.

Avis.

Plan.

1. Convergence

1.1 Généralités

Définition.
Exemple : suites arithmétiques, géométriques.
Théorème : convergence au sens de Césaro.
Théorèmes de convergence de suite débiles.

1.2 Aspects topologiques

Fermeture, ouverture, continuité.

2. Valeurs d'adhérence

Valeurs d'adhérences.
Limites supérieures et inférieures. Exemple : $\varphi(n)/n$.
Parler de suites équiréparties.
Compacité. Bolzano-Weierstrass.

3. Convergence des suites récurrentes

3.1 Suites récurrentes

Suites réelles $u_{n+1} = f(u_n)$.
AO Théorème de Sarkovski.

3.2 Méthode de Newton

Méthode de Newton dans le cas réel. Application à $\sqrt{a}$.

4. Suites et séries

Suites adjacentes et critère de Leibniz.
Motivation : le formalisme des séries pour comprendre les suites.
AD Ordre moyen de l'indicatrice d'Euler.

224 Exemples de développements asymptotiques de suites et de fonctions. (NR)

Développements choisis

— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★★★☆
— AV Méthode de Laplace et formule de Stirling.
Pertinence : ★★★★☆

Leçon non-rédigée...

Rapport 2018 du jury.

Cette leçon doit permettre aux candidats d'exprimer leur savoir-faire sur les techniques d'analyse élémentaire que ce soit sur les suites, les séries ou les intégrales. On peut par exemple établir un développement asymptotique à quelques termes des sommes partielles de la série harmonique, ou bien la formule de Stirling que ce soit dans sa version factorielle ou pour la fonction Γ. On peut également s'intéresser aux comportements autour des singularités de fonctions spéciales célèbres. Du côté de l'intégration, on peut évaluer la vitesse de divergence de l'intégrale de la valeur absolue du sinus cardinal, avec des applications pour les séries de Fourier, voire présenter la méthode de Laplace. Par ailleurs, le thème de la leçon permet l'étude de suites récurrentes (autres que le poncif $u_{n+1} = \sin(u_n)$, plus généralement de suites ou de fonctions définies implicitement, ou encore des études asymptotiques de solutions d'équations différentielles (sans résolution explicite).

226 Suites vectorielles et réelles définies par une relation de récurrence $u_{n+1} = f(u_n)$. Exemples. Applications à la résolution approchée d'équations.

Développements choisis

— AL Décomposition de Dunford effective.
Pertinence : ★★★★★
— AK Théorème de Householder & méthodes iteratives.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Citer au moins un théorème de point fixe dans cette leçon est pertinent. Le jury attend d'autres exemples que la sempiternelle suite récurrente $u_{n+1} = \sin(u_n)$(dont il est souhaitable de savoir expliquer les techniques sous-jacentes). La notion de points attractifs ou répulsifs peut illustrer cette leçon. L'étude des suites linéaires récurrentes d'ordre p est souvent mal connu, notamment le lien avec l'aspect vectoriel, d'ailleurs ce dernier point est trop souvent négligé. Le comportement des suites vectorielles définies par une relation linéaire $X_{n+1} = AX_n$ fournit pourtant un matériel d'étude conséquent. La formulation de cette leçon invite résolument à évoquer les problématiques de convergence d'algorithmes (notamment savoir estimer la vitesse) d'approximation de solutions de problèmes linéaires et non linéaires : dichotomie, méthode de Newton (avec sa généralisation au moins dans $\mathbb{R}^2$), algorithme du gradient, méthode de la puissance, méthodes itératives de résolution de systèmes linéaires, schéma d'Euler,...

Avis.

Plan.

1. Suites récurrentes, définitions

1.1 Généralités

Définitions. Ordre.

1.2 Résolution dans le cas des coefficients constants

Ordre 1, 2 et cas général.

2. Convergence et points fixes

2.1 Théorème de Picard et applications

Cauchy-Lipschitz essentiellement.
Contre-exemples : nécessité des hypothèses.

2.2 Classification dans le cas réel

Point répulsif, attractif, super attractif.
Exemples de cas où on ne peut pas conclure.
AO Théorème de Sarkovski.

3. Application à la résolution approchée d'équations

3.1 Méthodes itératives de résolution des systèmes linéaires

Motivation. Technique générale.
AK Théorème de Householder & méthodes iteratives.

3.2 Méthode de Newton

Méthode de Newton.
Application : AL Décomposition de Dunford effective..

228 Continuité et dérivabilité des fonctions d'une variable réelle. Exemples et applications. (NR)

Développements choisis

— AH Translatées d'une fonction dérivable.
Pertinence : ★★★★☆
— AO Théorème de Sarkovski.
Pertinence : ★★★★★

Leçon non-rédigée...

Rapport 2018 du jury.

Cette leçon permet des exposés de niveaux très variés. Les théorèmes de base doivent être maîtrisés et illustrés par des exemples intéressants, par exemple le théorème des valeurs intermédiaires pour la dérivée. Le jury s'attend évidemment à ce que le candidat connaisse et puisse calculer la dérivée des fonctions usuelles. Les candidats doivent disposer d'un exemple de fonction dérivable de la variable réelle qui ne soit pas continûment dérivable. La stabilité par passage à la limite des notions de continuité et de dérivabilité doit être comprise par les candidats. De façon plus fine, on peut s'intéresser aux fonctions continues nulle part dérivables. Pour aller plus loin, la dérivabilité presque partout des fonctions lipschitziennes ou des fonctions monotones relève de cette leçon. L'étude de la dérivée au sens des distributions de $x \in [a,b] \mapsto \int_a^x f(t)\mathrm{d}t$ pour une fonction intégrable $f \in L^1([a,b])$ est un résultat intéressant qui peut trouver sa place dans cette leçon.

229 Fonctions monotones. Fonctions convexes. Exemples et applications. (NR)

Développements choisis

— AR Gradient à pas optimal. (NR)
Pertinence : ★★★★★
— AX Processus de Galton-Watson.
Pertinence : ★★★☆☆

Leçon non-rédigée...

Rapport 2018 du jury.

L'énoncé et la connaissance de la preuve de l'existence de limites à gauche et à droite pour les fonctions monotones sont attendues. Ainsi on doit parler des propriétés de continuité et de dérivabilité à gauche et à droite des fonctions convexes de la variable réelle. Il est souhaitable d'illustrer la présentation de la convexité par des dessins clairs. On notera que la monotonie concerne les fonctions réelles d'une seule variable réelle, mais que la convexité concerne également les fonctions définies sur une partie convexe de $\mathbb{R}^n$, qui fournissent de beaux exemples d'utilisation. L'étude de la fonctionnelle quadratique ou la minimisation de $||Ax - b||^2$ pourront illustrer agréablement cette leçon. Pour aller plus loin, la dérivabilité presque partout des fonctions monotones est un résultat remarquable (dont la preuve peut être éventuellement admise). L'espace vectoriel engendré par les fonctions monotones (les fonctions à variation bornée) relève de cette leçon. Enfin, la dérivation au sens des distributions fournit les caractérisations les plus générales de la monotonie et de la convexité ; les candidats maîtrisant ces notions peuvent s'aventurer utilement dans cette direction.

230 Séries de nombres réels ou complexes. Comportement des restes ou des sommes partielles des séries numériques. Exemples. (NR)

Développements choisis

— AD Ordre moyen de l'indicatrice d'Euler.
Pertinence : ★★★☆☆
— AX Processus de Galton-Watson.
Pertinence : ★★★★☆

Leçon non-rédigée...

Rapport 2018 du jury.

De nombreux candidats commencent leur plan par une longue exposition des conditions classiques assurant la convergence ou la divergence des séries numériques. Sans être hors sujet, cette exposition ne doit pas former l'essentiel de la matière de la leçon. Un thème important de la leçon est en effet le comportement asymptotique des restes et sommes partielles (équivalents, développements asymptotiques — par exemple pour certaines suites récurrentes — cas des séries de Riemann, comparaison séries et intégrales,...). Le manque d'exemples est à déplorer. On peut aussi s'intéresser à certaines sommes particulières, que ce soit pour exhiber des nombres irrationnels (voire transcendants), ou mettre en valeur des techniques de calculs non triviales (par exemple en faisant appel aux séries de Fourier ou aux séries entières). Enfin le jury apprécie que le théorème des séries alternées (avec sa version sur le contrôle du reste) soit maîtrisé, mais il rappelle aussi que la transformation d'Abel trouve toute sa place dans cette leçon.

233 Méthodes itératives en analyse numérique matricielle.

Développements choisis

— AL Décomposition de Dunford effective.
Pertinence : ★★★★☆
— AK Théorème de Householder & méthodes iteratives.
Pertinence : ★★★★★

Rapport 2018 du jury.

Dans cette leçon de synthèse, les notions de norme matricielle et de rayon spectral sont centrales, en lien avec le conditionnement et avec la convergence des méthodes itératives ; elles doivent être développées. Le résultat général de convergence, relié au théorème du point fixe de Banach, doit être enrichi de considérations sur la vitesse de convergence. Le jury invite les candidats à étudier diverses méthodes issues de contextes variés : résolution de systèmes linéaires, optimisation de fonctionnelles quadratiques (du type $\frac{1}{2}(Ax,x)-(b,x)$), recherche de valeurs propres,... Parmi les points intéressants à développer, on peut citer les méthodes de type Jacobi pour la résolution de systèmes linéaires, les méthodes de gradient dans le cadre quadratique, les méthodes de puissance pour la recherche de valeurs propres. Les candidats pourront également envisager les schémas numériques pour les équations différentielles ou aux dérivées partielles linéaires.

Avis.

Une leçon finalement facile et intéressante.

Plan.

1. Outils théoriques pour l'étude numérique

Normes subordonnées, normes matricielles. Rayon spectral. AK Théorème de Householder & méthodes iteratives.(partie 1).

2. Méthodes itératives de résolution de systèmes linéaires

Motivations : complexité du pivot de Gauss, stabilité. Inconvénients des autres méthodes directes ?

2.1 A partir d'une décomposition régulière

Motivation autour des théorèmes de point fixe. AK Théorème de Householder & méthodes iteratives.(partie 2). Application aux méthodes itératives. Jacobi (cas de convergence : diagonale strictement dominante), Gauss-Seidel, relaxation (cas de convergence : symétrique définie positive, $\omega \in]0,2[$.

2.2 Minimisation convexe.

Méthode du gradient à pas optimal. Application dans le cas particulier.

3. Recherche d'une décomposition

AL Décomposition de Dunford effective. ? QR ?

4. Recherche d'éléments propres

Gerchgörin. Méthode de la puissance.

236 Illustrer par des exemples quelques méthodes de calcul d'intégrales de fonctions d'une ou plusieurs variables (I).

Impasse Stratégique...

Rapport 2018 du jury.

Cette leçon doit être très riche en exemples, que ce soit l'intégrale $\int_0^{+\infty} \frac{\sin(t)}{t} \mathrm{d}t$ ou bien d'autres encore. Il est tout à fait pertinent de commencer par les différentes techniques élémentaires (intégration par parties, changement de variables, décomposition en éléments simples, intégrale à paramètres,...). On peut également présenter des utilisations du théorème des résidus, ainsi que des exemples faisant intervenir les intégrales multiples comme le calcul de l'intégrale d'une gaussienne. Le calcul du volume de la boule unité de $\mathbb{R}^n$ ne doit pas poser de problèmes insurmontables. Le calcul de la transformation de Fourier d'une gaussienne a sa place dans cette leçon. On peut aussi penser à l'utilisation du théorème d'inversion de Fourier ou du théorème de Plancherel. Certains éléments de la leçon précédente, comme par exemple l'utilisation des théorèmes de convergence monotone, de convergence dominée et/ou de Fubini, sont aussi des outils ermettant le calcul de certaines intégrales. Enfin, il est aussi possible d'évoquer les méthodes de calcul approché d'intégrales (méthodes des rectangles, méthode de Monte-Carlo, etc.).

Avis.

Technique et atroce ! Qui de nos jours peut se vanter de connaître les règles de Bioche et de pouvoir les appliquer sans hésitation ?

239 Fonctions définies par une intégrale dépendant d'un paramètre. Exemples et applications. (NR)

Développements choisis

— AV Méthode de Laplace et formule de Stirling.
Pertinence : ★★★★★
— AU Formule d'inversion de Fourier dans $\mathcal{S}(\mathbb{R})$. (NR)

Leçon non-rédigée...

Rapport 2018 du jury.

Souvent les candidats incluent les théorèmes de régularité (version segment — a minima — mais aussi version « convergence dominée ») ce qui est pertinent. Cette leçon peut être enrichie par des études et méthodes de comportements asymptotiques. Les propriétés de la fonction Γ d'Euler fournissent un développement standard (on pourra y inclure le comportement asymptotique, voire son prolongement analytique). Les différentes transformations classiques (Fourier, Laplace,...) relèvent aussi naturellement de cette leçon. On peut en donner des applications pour obtenir la valeur d'intégrales classiques (celle de l'intégrale de Dirichlet par exemple). Le théorème d'holomorphie sous le signe intégrale est trop peu souvent cité. Pour aller encore plus loin, on peut par exemple développer les propriétés des transformations mentionnées (notamment la transformée de Fourier, par exemple en s'attardant sur le lien entre régularité de la fonction et décroissance de sa transformée de Fourier), ainsi que de la convolution.

243 Convergence des séries entières, propriétés de la somme. Exemples et applications.

Développements choisis

— AX Processus de Galton-Watson.
Pertinence : ★★★★★
— AW Nombres de Catalan.
Pertinence : ★★★★★

Rapport 2018 du jury.

Les candidats évoquent souvent des critères (Cauchy, D'Alembert) permettant d'estimer le rayon de convergence mais oublient souvent la formule de Cauchy-Hadamard. Le jury attend bien sûr que le candidat puisse donner des arguments justifiant qu'une série entière en 0 dont le rayon de convergence est R est développable en série entière en un point z_0 intérieur au disque de convergence et de minorer le rayon de convergence de cette série. Sans tomber dans un catalogue excessif, on peut indiquer les formules de développement de fonctions usuelles importantes $(\exp, \log, 1/(1-z), \sin$. S'agissant d'exemples fondamentaux et classiques, le jury attend que le candidat puisse les donner sans consulter ses notes. En ce qui concerne la fonction exponentielle, le candidat doit avoir réfléchi au point de vue adopté sur sa définition et donc sur l'articulation entre l'obtention du développement en série entière et les propriétés de la fonction. À ce propos, les résultats sur l'existence du développement en série entière pour les fonctions dont on contrôle toutes les dérivées successives sur un voisinage de 0 sont souvent méconnus. Le comportement de la série entière dans le disque de convergence vis à vis des différents modes de convergence (convergence absolue, convergence uniforme, convergence normale) doit être maîtrisé. Le théorème d'Abel (radial ou sectoriel) trouve toute sa place mais doit être agrémenté d'exercices pertinents. Réciproquement, les théorèmes taubériens offrent aussi de jolis développements. On pourra aller plus loin en abordant quelques propriétés importantes liées à l'analyticité de la somme d'une série entière.

Avis.

On s'intéresse à des séries de fonctions bien particulières : les séries entières. Mais la restriction à ce cas de figure n'est pas si grossière qu'il y paraît. D'abord on réfléchira à des propriétés formelles sur la sommation : rayon de convergence et stabilité par les opérations usuelles. On s'intéressera ensuite à la régularité de la somme : d'abord dans le cas réel (qui nous permettra d'étudier les DSE usuels), ensuite dans le cas complexe (holomorphie, et étude du comportement au bord du disque de convergence).

Plan.

1. Séries entières, rayon de convergence

Définition d'une série entière. Lemme d'Abel. Rayon de convergence, comportement dans le disque, hors du disque, sur le bord (exemples du n^2, $1/n^2$ et $(-1)^n/n$)
Règles de d'Alembert et de Cauchy-Hadamard.
Fonction somme d'une série entière. Somme, produit (de Cauchy), inverse.
[Gou09b], p 236.

2. Régularité de la somme, cas réel

Dérivabilité, calcul des dérivées. Liens avec le développement de Taylor de la fonction ($\frac{f^{(n)}(0)}{n!}$ et contre-exemple du e^{-1/x^2}). Parité/imparité.

Méthodes de majoration. Obtention des développements classiques (en annexe). Lien avec les équations différentielles.

3. Régularité sur le disque

Holomorphie, analycité. Formules de Cauchy et de Parseval. Majoration de Cauchy. Application à Liouville.
Théorème d'Abel (angulaire). Application à $\sum \frac{\sin(nt)}{n} = \frac{\pi - t}{2}$ et à $\sum \frac{(-1)^{n-1}}{n} = \ln(2)$.
Théorèmes taubériens faibles (cas $o(1/n^2)$ et ≥ 0).

4. Séries génératrices

Série génératrice associée à une suite.

4.1 Applications en combinatoire

Mots bien parenthésés. AW Nombres de Catalan. Nombres de Bell.
[FGN01a], p 10.

4.2 Applications en probabilités

Définition. Rayon de convergence. Caractérisation de la loi. Lien avec les moments d'ordre 1. Somme de variables indépendantes. Application à Galton-Watson.
[Ouv07], p 137.

246 Séries de Fourier. Exemples et applications. (NR)

Développements choisis

— AT Théorème de Banach-Steinhaus & séries de Fourier divergentes.
Pertinence : ★★★★☆
— AN Formule sommatoire de Poisson. (NR)
Pertinence : ★★★★☆

Leçon non-rédigée...

Rapport 2018 du jury.

Les différents résultats autour de la convergence ($_2$, Féjer, Dirichlet, ...) doivent être connus. On prendra garde au sens de la notation $\sum_{n\in\mathbb{Z}}$ (qu'il peut être plus prudent d'éviter en général). Il faut avoir les idées claires sur la notion de fonctions de classe $\mathcal{C}_1$ par morceaux (elles ne sont pas forcément continues). Dans le cas d'une fonction continue et $\mathcal{C}_1$ par morceaux on peut conclure sur la convergence normale de la série Fourier sans utiliser le théorème de Dirichlet. Il est classique d'obtenir des sommes de séries remarquables comme conséquence de ces théorèmes. On peut aussi s'intéresser à la formule de Poisson et à ses conséquences. L'existence d'exemples de séries de Fourier divergentes, associées à des fonctions continues (qu'ils soient explicites ou obtenus par des techniques d'analyse fonctionnelle) peuvent aussi compléter le contenu. Il est souhaitable que cette leçon ne se réduise pas à un cours abstrait sur les cœfficients de Fourier. La résolution d'équations aux dérivées partielles (par exemple l'équation de la chaleur ou l'équation des ondes avec une estimation de la vitesse de convergence) peuvent illustrer de manière pertinente cette leçon, mais on peut penser à bien d'autres applications (inégalité isopérimétrique, comportements remarquables des fonctions à spectre lacunaire,...).

250 Transformation de Fourier. Applications. (NR)

Développements choisis

— AU Formule d'inversion de Fourier dans $\mathcal{S}(\mathbb{R})$. (NR)
— AN Formule sommatoire de Poisson. (NR)

Leçon non-rédigée...

Rapport 2018 du jury.

Cette leçon offre de multiples facettes. Les candidats peuvent adopter différents points de vue : L_1, L_2 et/ou distributions. L'aspect « séries de Fourier» n'est toutefois pas dans l'esprit de cette leçon ; il ne s'agit pas de faire de l'analyse de Fourier sur n'importe quel groupe localement compact mais sur $\mathbb{R}$ ou $\mathbb{R}^d$. La leçon nécessite une bonne maîtrise de questions de base telle que la définition du produit de convolution de deux fonctions de L_1. En ce qui concerne la transformation de Fourier, elle ne doit pas se limiter à une analyse algébrique de la transformation de Fourier. C'est bien une leçon d'analyse, qui nécessite une étude soigneuse des hypothèses, des définitions et de la nature des objets manipulés. Le lien entre la régularité de la fonction et la décroissance de sa transformée de Fourier doit être fait, même sous des hypothèses qui ne sont pas minimales. Les candidats doivent savoir montrer le lemme de Riemann-Lebesgue pour une fonction intégrable. La formule d'inversion de Fourier pour une fonction L_1 dont la transformée de Fourier est aussi L_1 est attendue ainsi que l'extension de la transformée de Fourier à l'espace L_2 par Fourier-Plancherel. Des exemples explicites de calcul de transformations de Fourier, classiques comme la gaussienne ou $\frac{1}{1+x^2}$, paraissent nécessaires. Pour aller plus loin, la transformation de Fourier des distributions tempérées ainsi que la convolution dans le cadre des distributions tempérées peuvent être abordées. Rappelons une fois de plus que les attentes du jury sur ces questions restent modestes, au niveau de ce qu'un cours de première année de master sur le sujet peut contenir. Le fait que la transformée de Fourier envoie $\mathcal{S}(\mathbb{R}^d)$ dans lui même avec de bonnes estimations des semi-normes doit alors être compris et la formule d'inversion de Fourier maîtrisée dans ce cadre. Des exemples de calcul de transformée de Fourier peuvent être données dans des contextes liés à la théorie des distributions comme par exemple la transformée de Fourier de la valeur principale. La résolution de certaines équations aux dérivées partielles telle que, par exemple, l'équation de la chaleur sur R, peut être abordée, avec une discussion sur les propriétés qualitatives des solutions. Dans un autre registre, il est aussi possible d'orienter la leçon vers l'étude de propriétés de fonctions caractéristiques de variables aléatoires.

Avis.

Vu la longueur du rapport, le jury l'aime de tout cœur.

260 Espérance, variance, et moments d'une variable aléatoire.

Développements choisis

— AX Processus de Galton-Watson.
Pertinence : ★★★☆☆
— AY Nombres normaux.
Pertinence : ★★★★★

Rapport 2018 du jury.

Le jury attend des candidats qu'ils donnent la définition des moments centrés, qu'ils rappellent les implications d'existence de moments (décroissance des L_p). Le candidat peut citer — mais doit surtout savoir retrouver rapidement — les espérances et variances de lois usuelles, notamment Bernoulli, binomiale, géométrique, Poisson, exponentielle, normale. La variance de la somme de variables aléatoires indépendantes suscite souvent des hésitations. Les inégalités classiques (de Markov, de Bienaymé-Chebyshev, de Jensen et de Cauchy-Schwarz) pourront être données, ainsi que les théorèmes de convergence (lois des grands nombres et théorème central limite). La notion de fonction génératrice des moments pourra être présentée ainsi que les liens entre moments et fonction caractéristique. Pour aller plus loin, le comportement des moyennes empiriques pour une suite de variables aléatoires indépendantes et identiquement distribuées n'admettant pas d'espérance pourra être étudié. Pour les candidats suffisamment à l'aise avec ce sujet, l'espérance conditionnelle pourra aussi être abordée.

1. Espérance d'une variable aléatoire

2. Moments d'une variable aléatoire

3. Fonction caractéristique et série génératrice

3.1 Le cas discret : les séries génératrices

Définition. Formules. Caractérisation de la loi. Exemples classiques. Lien avec les moments. Somme.
AX Processus de Galton-Watson.

3.2 Fonctions génératrices

4. Théorèmes limites

AY Nombres normaux.

264 Variables aléatoires discrètes. Exemples et applications.

Développements choisis

— AX Processus de Galton-Watson.
Pertinence : ★★★★★
— AY Nombres normaux.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le jury attend des candidats qu'ils rappellent la définition d'une variable aléatoire discrète et que des lois usuelles soient présentées, en lien avec des exemples classiques de modélisation. Le lien entre variables aléatoires de Bernoulli, binomiale et de Poisson doit être discuté. Il peut être d'ailleurs intéressant de mettre en avant le rôle central joué par les variables aléatoires de Bernoulli. Les techniques spécifiques aux variables discrètes, notamment à valeurs entières, devront être mises en évidence, comme par exemple la caractérisation de la convergence en loi, la notion de fonction génératrice. Pour aller plus loin, le processus de Galton-Watson peut se traiter intégralement à l'aide des fonctions génératrices et cette voie a été choisie par plusieurs candidats : cela donne un développement de très bon niveau pour ceux qui savent justifier les étapes délicates. Pour aller beaucoup plus loin, les candidats pourront étudier les marches aléatoires, les chaînes de Markov à espaces d'états finis ou dénombrables, les sommes ou séries de variables aléatoires indépendantes.

1. Variables aléatoires discrètes, généralités

1.1 Définitions

1.2 Exemples

2. Moments d'une variable aléatoire discrète

3. Fonctions génératrices

Définition. Formules. Caractérisation de la loi. Exemples classiques. Lien avec les moments. Somme. Application à Galton-Watson.

4. Suite de variables aléatoires discrètes

Poisson.
Loi des grands nombres dans des cas particuliers.
AY Nombres normaux.

CHAPITRE 5

LEÇONS D'INFORMATIQUE

Tu dois fonder un empire puissant,
Dans l'avenir, dominateur du monde,
Où la mort des héros t'attend.

Les Troyens, Hector Berlioz

901 Structures de données. Exemples et applications.

Développements choisis

— BC Arbres AVL.
Pertinence : ★★★★★
— BD Hachage parfait.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le mot algorithme ne figure pas dans l'intitulé de cette leçon, même si l'utilisation des structures de données est évidemment fortement liée à des questions algorithmiques. La leçon doit donc être orientée plutôt sur la question du choix d'une structure de données. Le jury attend du candidat qu'il présente différents types abstraits de structures de données en donnant quelques exemples de leur usage avant de s'intéresser au choix de la structure concrète. Les notions de complexité des opérations usuelles sur la structure de données sont bien sûr essentielles dans cette leçon. Le candidat ne peut se limiter à des structures linéaires simples comme des tableaux ou des listes, mais doit présenter également quelques structures plus complexes, reposant par exemple sur des implantations à l'aide d'arbres.

Avis.

Pourquoi utiliser des structures de données ? Pourquoi créer/utiliser des bibliothèques d'implémentations *efficaces* des structures usuelles ? Quelques éléments de réponse :

— Elles peuvent être réutilisées dans *tous* les programmes qui en ont besoin (factorisation).
— Ce découpage permet de rendre le code d'un programme plus lisible.
— Spécifier formellement ce que l'on attend d'un type rend le code plus modulaire : on peut remplacer une implémentation par une autre "qui fait la même chose". Important en génie logiciel [1], en gestion *collaborative* de projets.
— Se concentrer sur des problèmes précis permet de fournir des implémentations efficaces. On donnera en outre des garanties théoriques sur leurs complexités, ce qui facilitera l'étude des algorithmes qui l'utilisent. La complexité des suites d'opérations, ou *complexité amortie* prend tout sens dans ce cadre.

Le plan suivra une liste de types abstraits (par difficulté croissante, soit dans la conception de l'implémentation, soit dans l'analyse). Cela peut paraître idiot, mais c'est probablement la manière la plus pédagogique d'introduire le sujet [2]. Ne pas hésiter à faire des dessins et à mentionner des algorithmes qui utilisent les structures que l'on présente. On implémentera essentiellement des structures linéaires ou arborescentes. D'aucuns évoquent un type abstrait "graphe" implémenté par matrices ou listes. C'est étrange ! Il faudrait savoir à quoi peut servir un tel type, et quelles sont les opérations que l'on veut faire dessus [3].
Ce plan pourrait être complété par une mention des structures de données pour l'algorithmique du texte (automates, suffix trees, suffix tables, dwags...) mais la place fait défaut.

Plan.

1. Introduction : l'exemple de la pile

Définition d'un type abstrait.
Opérations sur le type pile. Remarque : on peut parfois axiomatiser (pas trop formel).
Implémentation par une liste chaînée. Application au DFS d'un graphe.

1. Qu'est-ce que c'est ?
2. Suivant la maxime : "un plan trop travaillé devient étrange pour qui ne l'a pas écrit".
3. Je m'explique. On utilise des implémentations arborescentes, non pas parce les données ont un structure d'arbre, mais parce que l'arbre est une implémentation pratique. On ne peut donc pas décemment motiver l'utilisation d'un graphe par le fait que certains problèmes se posent sur les graphes...

Complexité amortie. Etude du dépilage de k élément, amorti en $\mathcal{O}(1)$.

2. Quelques structures de base

2.1 File

Opérations. Implémentation par liste chaînée. Application au BFS d'un graphe. Implémentation avec 2 piles. Complexité amortie constante.

2.2 Pile

Opérations : accès/ajout/suppression partout. Implémentation avec une liste doublement chaînée. Implémentation par tableaux dynamiques (modification $\times 2$ ou $/2$ à $1/4$). Complexité amortie constante.

2.3 Arborescences

Avec des pointeurs.

3. File de priorité

Opérations. Implémentation par un tableau trié. Complexité $\mathcal{O}(n)$.
Utilisation d'un tas binaire. Complexité $\mathcal{O}(\log(n))$. Implémentation efficace dans un tableau.
Applications : plus courts chemins avec Dijsktra en $\mathcal{O}((|A| + |S|)\log(|S|))$, arbre couvrant avec Prim en $\mathcal{O}((|A| + |S|)\log(|S|))$.
(Remarque : autres structures de tas comme binomial ou Fibonacci - ce dernier réduit Dijsktra et Prim à $\mathcal{O}(|A| + |S|\log(|S|))$).

4. Dictionnaires

Opérations (ensemble de clefs associées à des objets).
Adressage direct, défaut en espace. Liste chaînée, défaut en temps.

4.1 Implémentations par ABR

Arbre binaire de recherche. Complexité $\mathcal{O}(h)$.
Arbre AVL. BC Arbres AVL. Complexité $\mathcal{O}(\log(n))$.

4.2 Tables de hachage

Fonction de hachage. Famille universelle $ax + b$ dans $\mathbb{Z}/p\mathbb{Z}$. Hachage universel.
Cas où l'ensemble des clefs est constant. BD Hachage parfait.

5. Gestion de partitions : union-find

Opérations : gestion d'une relation d'équivalence, tester l'équivalence, faire l'union des classes.
Implémentation : forêt avec union par rangs et compression de chemins. Complexité amortie en $\log^*(n)$.
Applications : arbre couvrant avec Kruskal, équivalence d'automates.

902 Diviser pour régner. Exemples et applications.

Développements choisis

— BA Complexité du tri rapide aléatoire.
Pertinence : ★★★☆☆
— BB Tri bitonique.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon permet au candidat de proposer différents algorithmes utilisant le paradigme diviser pour régner. Le jury attend du candidat que ces exemples soient variés et touchent des domaines différents. Un calcul de complexité ne peut se limiter au cas où la taille du problème est une puissance exacte de 2, ni à une application directe d'un théorème très général recopié approximativement d'un ouvrage de la bibliothèque de l'agrégation.

Avis.

Se prête bien assez à un plan théorie/applications. Ce n'est pas très pédagogique, mais comme il est agréable d'avoir déjà tous les outils à sa disposition pour traiter des exemples riches et variés ! La présentation du paradigme est conduite au travers de deux exemples : la recherche dichotomique et le tri fusion. L'étude de complexité du second motive le développement de méthodes génériques, ce que l'on présente sans plus attendre. Le reste n'est qu'exemples.

1. Paradigme diviser-pour-régner

1.1 Généralités

Paradigme Diviser pour régner : Diviser-Conquérir-Régner. Régner est souvent le plus dur.
Recherche dichotomique. Présenter l'algorithme. Etude triviale de la complexité.
Tri fusion. Présenter l'algorithme. Etude "à la main" de complexité.

1.2 Etude générique de complexité

Récurrences de partitions. Exemple du tri fusion.
Méthode : pondre une formule et l'injecter dedans.
Méthode : master théorème.
Méthode : arbre des appels récursifs.

2. Application aux algorithmes de tri

Tri rapide. Principe. Complexité naïve. Complexité moyenne sur les données. BA Complexité du tri rapide aléatoire.Utilisation pratique
Calcul du k-ième élément. Médian des médians. Complexité. Application au tri rapide en $\mathcal{O}(n\log(n))$ dans le pire des cas.
Réseaux de comparaisons. Réseaux de tri. Exemple du réseau de tri par insertion/bulle. Lemme du $0-1$. Notion de profondeur d'un circuit. BB Tri bitonique.

3. Algorithmes de calcul

Exponentiation rapide.
Produit d'entiers. Karatsuba.
Produits de matrices.
Produit de polynômes. FFT. Remarque : utilisé pour calculer les coefficients de Fourier.

4. Algorithmes géométriques dans le plan

Enveloppe convexe. Quickhull (attention le Diviser n'est pas trivial). Complexité : $\mathcal{O}(n^2)$.
Plus proches voisins. Algorithme standard

903 Exemples d'algorithmes de tri. Correction et complexité.

Développements choisis

— BA Complexité du tri rapide aléatoire.
Pertinence : ★★★★★
— BB Tri bitonique.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Sur un thème aussi classique, le jury attend des candidats la plus grande précision et la plus grande rigueur. Ainsi, sur l'exemple du tri rapide, il est attendu du candidat qu'il sache décrire avec soin l'algorithme de partition et en prouver la correction en exhibant un invariant adapté. L'évaluation des complexités dans le cas le pire et en moyenne devra être menée avec rigueur : si on utilise le langage des probabilités, il importe que le candidat sache sur quel espace probabilisé il travaille. On attend également du candidat qu'il évoque la question du tri en place, des tris stables, ainsi que la représentation en machine des collections triées. Le jury ne manquera pas de demander au candidat des applications non triviales du tri.

Avis.

Pourquoi étudier les algorithmes de tri ? D'une part, ils constituent une brique de base de nombreux programmes. Mais d'un point de vue plus pédagogique, leur variété offre *en germe* un aperçu de nombreuses questions algorithmiques : des algorithmes naïfs issus d'un traitement manuel, le paradigme diviser-pour-régner, utilisation de structures de données, les algorithmes probabilistes, l'étude de leurs correction et de leur complexité... Sur ce dernier point, on peut même obtenir quelques bornes inférieures (ce qui est assez rare).

1. Généralités sur les tris

1.1 Position du problème

Problème du tri. Représentation des données par listes et tableaux. Question de la mesure de complexité. Question des opérations autorisées (exemple du tri de crêpes). Tri en place. Tri stable. Applications non-triviales : gloutons...

1.2 Quelques algorithmes naïfs

Tri par insertion. Complexité : temporelle $\mathcal{O}(n^2)$ dans le pire, spatiale $\mathcal{O}(1)$. Propriétés : stable, en place, en ligne, utilisable "à la main".
Tri par sélection. Complexité : temporelle $\mathcal{O}(n^2)$ dans tous les cas, spatiale $\mathcal{O}(1)$. Propriétés : en place, instable mais peut-être stabilisé.
Tri bulle. Complexité : temporelle $\mathcal{O}(n^2)$ dans tous les cas (sans optimisation), spatiale $\mathcal{O}(1)$. Propriétés : en place, stable. Variante du cocktail qui va dans les 2 sens.

2. Tris par comparaison optimaux.

Borne inférieure de $\mathcal{O}(n\log(n))$ sur les tris par comparaison.

2.1 Diviser-pour-régner

Tri fusion. Algorithme. Complexité. Propriétés : pratique sur les listes, pas en place sur les tableaux, stable.

Tri rapide. Fonction de partition. Complexité naïve. Complexité moyenne sur les données[1]. Complexité espérée de l'algorithme aléatoire. Propriétés : en place, instable, mais pratique.Pivot = médiane permet de garantir l'équilibre. On peut la calculer en temps linéaire par l'algorithme du médian des médians. Ce n'est pas fait en pratique.

2.2 Utilisation de structures de données

Tri par tas. Structure abstraite de tas binaire. Complexité que l'on peut obtenir. Implémentation avec un tableau. Propriétés : en place, stable.
Tri arborescent. Opérations sur les ABR. Complexité naïve. Complexité moyenne sur les données. Utilisation d'AVL pour être optimal. Propriétés : pas en place.

3. D'autres techniques de tri

3.1 Par l'histogramme des valeurs

Tri comptage. Valeurs entre 1 et N. Complexité : temporelle $n + N$, spatiale N ; donc efficace quand N n'est pas trop grand. Propriétés : instable.

3.2 En utilisant la structure des clefs

Tri par base. k taille des clefs. Complexité : temporelle $\mathcal{O}(nk)$, spatiale. Propriétés : stable, en place. Exemples binaire. Aspect historique des cartes perforées.

3.3 Réseaux de tri [CLRS02]

Réseaux de comparaisons. Réseaux de tri. Exemple du réseau de tri par insertion/bulle. Lemme du $0-1$. Notion de profondeur d'un circuit. Tri bitonique de profondeur $\mathcal{O}(\log^2(n))$.

1. Attention, ce développement est délicat.

906 Programmation dynamique. Exemples et applications.

Développements choisis

— BE Distance d'édition.
Pertinence : ★★★★☆
— BL Théorème de Berman (langage unaires).
Pertinence : ★★★★☆

Rapport 2018 du jury.

Même s'il s'agit d'une leçon d'exemples et d'applications, le jury attend des candidats qu'ils présentent les idées générales de la programmation dynamique et en particulier qu'ils aient compris le caractère générique de la technique de mémoïsation. Le jury appréciera que les exemples choisis par le candidat couvrent des domaines variés, et ne se limitent pas au calcul de la longueur de la plus grande sous-séquence commune à deux chaînes de caractères. Le jury ne manquera pas d'interroger plus particulièrement le candidat sur la question de la correction des algorithmes proposés et sur la question de leur complexité en espace.

Avis.

Il *faut* parler de mémoïzation, de sous-problèmes. Faire la différence entre diviser-pour-régner (éviter le recouvrement) et la programmation dynamique (partage). Expliquer comment reconstruire les solutions.
L'étude de la complexité est en général plus facile que pour les algorithmes "Diviser pour régner".

Plan.

Parenthésage optimal.

1. Principe général

1.1 Sous-problèmes et optimalité

Exemple de départ. Plus court chemin avec droite-bas dans une matrice $n \times n$. Complexité naïve : $\mathcal{O}(2^n)$. Mais $\mathcal{O}(n^2)$ en stockant les minimums.
Principe général. Diviser pour régner : chercher une solution à un problème d'optimisation. 1.Généraliser le problème à des sous-problèmes. 2. Comprendre les relations entre problèmes. 3. Les résoudre dans un ordre de dépendance.
Exemple. Parenthésage optimal de matrices. Nombres de Catalan.

1.2 Récursif ou itératif

Question : comment traiter les sous-problèmes ? De haut en bas ? De bas en haut ? Ordre total ?
Exemple. Suite de Fibonacci. Itératif : tableau. Récursif : sans/avec tableau. Complexité
Paradigmes. Bottom/Up (il faut avoir déterminé un ordre total sur les sous-problèmes) et Top/Down (plus "à la volée"). Mémoïzation.

1.3 Reconstruction de la solution

Marquage dans le tableau.

2. Algorithmes de graphes

Plus courts chemins. Algorithme de Bellman-Ford. Algorithme de Floyd-Warshall. Algorithme BMC. Algorithme de MacNaughton-Yamada.
Problème des longs lexèmes. Algorithme de recherche du plus long lexème.

3. Problèmes de mots

Problème du mot dans une grammaire. Forme Méthode naïve : $\mathcal{O}(|G|^{|w|})$. Algorithme CYK en $\mathcal{O}(|w|^3)$
Distance d'édition. Définition par réécriture. BE Distance d'édition. Réduction d'espace.Extensions.
Plus longue sous-séquence commune. Algorithme.

4. Algorithmes pseudo-poly et NP-complétude

Problème du sac à dos. Différence entre pseudo-polynomial et polynomial.
BL Théorème de Berman (langage unaires). Remarque sur itératif ou récursif.

907 Algorithmique du texte. Exemples et applications.

Développements choisis

— BE Distance d'édition.
Pertinence : ★★★★☆
— BF Automate des bordures.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon devrait permettre au candidat de présenter une grande variété d'algorithmes et de paradigmes de programmation, et ne devrait pas se limiter au seul problème de la recherche d'un motif dans un texte, surtout si le candidat ne sait présenter que la méthode naïve. De même, des structures de données plus riches que les tableaux de caractères peuvent montrer leur utilité dans certains algorithmes, qu'il s'agisse d'automates ou d'arbres par exemple. Cependant, cette leçon ne doit pas être confondue avec la 909,"Langages rationnels et Automates finis. Exemples et applications.". La compression de texte peut faire partie de cette leçon si les algorithmes présentés contiennent effectivement des opérations comme les comparaisons de chaînes : la compression LZW, par exemple, est plus pertinente dans cette leçon que la compression de Huffman.

Avis.

La référence est [CR94], mais il faut l'avoir lu avant. Dans cette leçon, on doit manipuler la structure d'un texte, étudier les bordures, faire des preuves de "matching" qui exploitent les comparaisons. La recherche de motif est le sujet le plus naturel. Il me semble [1] que l'automate associé à un motif donne une bonne intuition de ce que font MP et KMP par la suite (ces deux algorithmes étant inévitables). On parle ensuite rapidement de Boyer-Moore.
Le reste du plan relève davantage de "morceaux choisis". La distance d'édition est pertinente en ce qu'elle permet des recherches approchées et des alignements. Enfin, les *suffix trees* constituent une excellente structure de donnée, largement non-triviale, pour résoudre nombre de problèmes. Le développement sur le codage de Huffman est hors-sujet.

Plan.

Hypothèses : alphabet Σ fixé.
Evoquer la pertinence du modèle (utf8, etc).

1. Recherche de motif

Objectif : chercher les occurrence d'un motif m dans un texte t.

1.1 Algorithme naïf

Algorithme : recherche naïve.
Complexité : espace $\mathcal{O}(1)$, temps $\mathcal{O}(|m||t|)$. La borne est atteinte.
Observation : on ne retient pas ce que l'on a déjà lu.

1.2 Automate associé à un motif

Idée : états= plus grands préfixes du motif.
Bordure : calcul.
BF Automate des bordures.

1. C'est sans doute un biais, eu égard aux sujets que j'aime bien.

1.3 MP et KMP

Idée : ne pas construire entièrement l'automate pour éviter le $|\Sigma|$.
Algorithme MP : revenir à l'état Bord(u) si la dernière ne correspond pas, puis relire a.
Complexité : a priori pas en $\mathcal{O}(|t|)$ mais en fait si (on ne peut pas "trop" relire des lettres).
Algorithme KMP : utiliser aussi l'information de l'échec sur la dernière lettre.
Remarque : pré-traitement possible aussi en temps $\mathcal{O}(|m|)$.

1.4 Boyer-Moore

Idée : améliorer la complexité au meilleur cas, lire à l'envers.
Exemple :
Remarque : on peut améliorer la complexité dans le pire des cas.

2. Mesure de distance et recherche approchée

2.1 Distance d'édition

Définition : réécriture → des fautes. Définition
BE Distance d'édition.

3. Arbre des suffixes et applications

Objectif : A partir d'un texte t, avoir une structure de taille $\mathcal{O}(|t|)$, construite en temps $\mathcal{O}(|t|)$, qui permet de rechercher un motif m en $\mathcal{O}(|m|)$. Σ n'apparaît pas dans les complexités.
Définition : *suffix trie* de t. Exemple.
Propriété : recherche du motif m en $\mathcal{O}(|m|)$.
Algo : construction en ligne en temps $\mathcal{O}(|t|^2)$ (avec les suffix links).
Définition : *suffix tree* de t. Exemple. Taille $\mathcal{O}(|t|)$.
Algo (Ukkonen) : construction en ligne du suffix tree en $\mathcal{O}(|t|)$ (subtil).
Application : plus long facteur commun.
[CR94], p 79.

909 Langages rationnels et automates finis. Exemples et applications.

Développements choisis

— BI Décidabilité de l'arithmétique de Presburger.
Pertinence : ★★★★☆
— BJ Machines de Turing et langages rationnels.
Pertinence : ★★★★★

Rapport 2018 du jury.

Pour cette leçon très classique, il importe de ne pas oublier de donner exemples et applications, ainsi que le demande l'intitulé. Une approche algorithmique doit être privilégiée dans la présentation des résultats classiques (déterminisation, théorème de Kleene, etc.) qui pourra utilement être illustrée par des exemples. Le jury pourra naturellement poser des questions telles que : connaissez-vous un algorithme pour décider de l'égalité des langages reconnus par deux automates ? quelle est sa complexité ? Des applications dans le domaine de l'analyse lexicale et de la compilation entrent naturellement dans le cadre de cette leçon.

Avis.

Attention, les automates sont introduits dans le volet "algorithmique", il n'est donc pas question de ne faire *que* de la logique ou de l'algèbre (même si c'est le plus joli). On ne manquera pas de parler d'algorithmique du texte et d'analyse lexicale. Presburger utilise crucialement l'effectivité des propriétés de clôture. Pour rester dans les aspects pénibles, il faudra être capable de donner "à la main" un automate qui reconnaît tel ou tel langage.

Plan.

1. Automates finis

Définition : automate (par défaut ND et incomplet). Calcul. Langage reconnu. Exemple.
Définition : déterminisme, complétude. Déterminisation (exponentiel). Complétion (linéaire). Remarque sur la possibilité des ε-transitions.
Lemme(s) de pompage. Exemples de langages non-rationnels.
Utilisation en pratique d'un automate. BF Automate des bordures.

2. Langages reconnaissables

2.1 Autour des propriétés de clôture

Clôture par les opérations booléennes. Parler de l'automate produit. Clôture par morphisme et morphisme inverse. Clôture par étoile. Effectivité de toutes les transformations.
BI Décidabilité de l'arithmétique de Presburger.

2.2 Expressions rationnelles

Définitions : syntaxe, sémantique. Philosophie : utile pour *spécifier* des propriétés.
Théorème de Kleene ($\Rightarrow$ par clôture (peu efficace) ou Glushkov ($\mathcal{O}(||E||)$ états, $\mathcal{O}(||E||^2)$ transitions), $\Leftarrow$ par Mac Naughton-Yamada).
Application : analyse lexicale (simuler l'automate, chercher le lexème le plus long).

3. Aspects algébriques

3.1 Congruences et monoïdes

Congruences d'indice fini. Reconnaissable par monoïde/congruence.
BJ Machines de Turing et langages rationnels. Remarque : automates boustrophéons.

3.2 Automate minimal

Automate résiduel. Congruence minimale de Nerode. Calcul par Moore. Objet canonique.
Remarque : résolution de l'équivalence sans minimiser, par bisimulation.

912 Fonctions récursives primitives et non primitives. Exemples. (I)

Impasse Stratégique...

Rapport 2018 du jury.

Il s'agit de présenter un modèle de calcul : les fonctions récursives. Il est important de faire le lien avec d'autres modèles de calcul, par exemple les machines de Turing. En revanche, la leçon ne peut pas se limiter à l'aspect "modèle de calcul" et doit traiter des spécificités de l'approche. Le candidat doit motiver l'intérêt de ces classes de fonctions sur les entiers et pourra aborder la hiérarchie des fonctions récursives primitives. Enfin, la variété des exemples proposés sera appréciée.

Avis.

Les fonctions récursives sont le "parent mal-aimé de la complexité" [1]. Pourquoi diable ? D'une part les exemples les plus simples sont délicats à décrire, d'autre part les théorèmes de décidabilité sont compliqués à concevoir dans ce cadre [2], et enfin la notion de complexité n'apparaît pas naturellement. D'ailleurs, est-ce que ce ne serait pas... un peu comme de λ-calcul non-typé ? On pourra donc arguer que le jury n'oserait donc pas faire un couplage 912/929 et faire l'impasse sur les deux.
En dépit le peu de passion qu'on leur accorde, les fonctions récursives ne sont pas totalement inutiles. Leur construction *par induction* les rend dans certains cadres bien plus faciles à manipuler que les machines de Turing. Par exemple pour montrer que toutes les fonctions calculables sont définissables dans l'arithmétique [3]. Attention, la "hiérarchie des fonctions récursives primitives", c'est un peu technique et il faut trouver une référence qui le fait...

1. Pour reprendre une expression de D. Madore.
2. Peut-être comprenons-nous mieux les calculs par une "vraie" machine ?
3. La réciproque est loin d'être vraie, hélas. Les fonctions définissables dans l'arithmétique ne sont même pas un langage de la hiérarchie arithmétique...

913 Machines de Turing. Applications.

Développements choisis

— BJ Machines de Turing et langages rationnels.
Pertinence : ★★★★☆
— BK Théorèmes de hiérarchie en espace et en temps.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Il s'agit de présenter un modèle de calcul. Le candidat doit expliquer l'intérêt de disposer d'un modèle formel de calcul et discuter le choix des machines de Turing. La leçon ne peut se réduire à la leçon 914 ou à la leçon 915, même si, bien sûr, la complexité et l'indécidabilité sont des exemples d'applications. Plusieurs développements peuvent être communs avec une des leçons 914, 915, mais il est apprécié qu'un développement spécifique soit proposé, comme le lien avec d'autres modèles de calcul, ou le lien entre diverses variantes des machines de Turing.

Avis.

Le développement spécifique est BJ Machines de Turing et langages rationnels., qui permet d'illustrer la notions de calcul pour obtenir un résultat assez original. Canoniquement on fera un plan : machines/applications en décidabilité/applications en complexité.

Plan.

1. Machines de Turing

1.1 En mode acceptant

Définitions : machine (par défaut ND), configuration, calcul. Langage accepté. Langages récursivement énumérables (c'est la définition naturelle d'acceptation).

1.2 Variantes et robustesse

Non-déterminisme, machines à plusieurs bandes, bandes bi-infinies, réduction de l'alphabet, réduction du nombre d'états (certaines développées, certaines en remarque).

1.3 En mode calculant

Entrée/sortie. Possibilité d'utiliser des bandes spécifiques.
Fonction (partielle) calculable. Graphe récursivement énumérable.

1.4 Lien avec d'autres modèles de calcul

Equivalence avec les fonctions récursives. Remarque sur la thèse de Church.
Plus expressive que les automates (à pile).
BJ Machines de Turing et langages rationnels..

2. Calculabilité et décidabilité

2.1 Langages récursivement énumérables

Liens avec l'existence d'un énumérateur.
Existence par cardinalité. Exemple du langage diagonal[1].
Remarque sur les similitudes programme/langage.

1. Qui est en fait la preuve canonique que les cardinaux sont différents...

2.2 Langages décidables

Définition. Propriétés de clôture. RE+cRE $\Rightarrow$ décidable.
Exemples.

2.3 Réductions : principe et exemples

Réduction : fonction calculable totale. Propagation de la décidabilité/indécidabilité.
Théorème de Rice. Philosophie : on ne peut pas vérifier les programmes.
PCP. Motivation : plus facile à encoder (modèle de calcul "plus simple").
Problèmes indécidables sur les langages.

3. Complexité en temps et en espace

3.1 Définition de la complexité

Définition : Complexité d'un calcul (espace/temps borné pour toutes les branches).
Speedup en espace, en temps (attention aux bornes).
Changement d'alphabet de travail, passage à une bande. Remarque : palindromes.
Fonction propre (motivée par les restrictions précédentes).
Définitions : DTIME, NTIME, DSPACE, NSPACE. Classes duales. Croissance.

3.2 Exemples de classes.

P, NP, PSPACE
Complétude, problèmes naturels.

3.3 Simulation et théorèmes et de hiérarchie

Théorème de simulation : temps $\alpha_M T^2$, espace $\beta_M E$.
BK Théorèmes de hiérarchie en espace et en temps.
Remarque sur les théorèmes non-déterministes.
Applications aux classes usuelles.

914 Décidabilité et indécidabilité. Exemples.

Développements choisis

— BP Problèmes indécidables sur les grammaires algébriques.
Pertinence : ★★★★☆
— BI Décidabilité de l'arithmétique de Presburger.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le programme de l'option offre de très nombreuses possibilités d'exemples. Si les exemples classiques de problèmes sur les machines de Turing figurent naturellement dans la leçon, le jury apprécie des exemples issus d'autres parties du programme : théorie des langages, logique,... Le jury portera une attention particulière à une formalisation propre des réductions, qui sont parfois très approximatives.

Avis.

La motivation est toute bête : on veut chercher des algorithmes pour résoudre des problèmes, mais on montre que dans certains cas ce n'est pas possible. Et donc inutile de chercher... A l'oral, on dira qu'il faut formaliser tout cela : d'une part la notion de modèle de calcul (les machines de Turing, mais ce n'est pas le sujet), d'autre part les problèmes (vivant dans le monde réel, pratique, et matériel, modèle ambiant de la théorie des ensembles ?) vers des langages.
On partira donc de résultats "théoriques" autour de ce modèle (langages reconnus, problème de l'arrêt) et d'outils (réduction). On utilisera ensuite cet outil pour étudier la décidabilité (avec Rice, qui dit que la vérification de programmes est impossible) et dériver peu à peu vers des résultats pratiques. Attention, il faudra toujours comparer décidabilité et l'indécidabilité, sans pencher trop lourdement d'un côté.
Si l'on est très fort, on pourra introduire en quatrième partie la hiérarchie arithmétique, parler de son non-effondrement et y situer quelques problèmes connus (vacuité Π_1-complet, finitude Σ_2-complet, universalité Π_2-complet...[1]). C'est joli et cela cache une théorie profonde, hélas un peu oubliée par la "jeune" génération[2], sous prétexte qu'elle manque d'applications pratiques[3].

1. Calculabilité

1.1 Langages reconnus par les machines de Turing

Définitions : décidable, récursivement énumérable. Exemples débiles. Propriétés de clôture (booléenne pour D, union et intersection pour RE, RE et coRE $\Rightarrow$ D).
Notion de problème. Problème de l'arrêt borné (décidable) et de l'arrêt.

1.2 Fonctions calculables et réductions

Fonctions calculables. Calculable $\iff$ graphe RE.
Equivalence avec les autres modèles de calcul. Thèse de Church.
Réduction. Propagation de la décidabilité ou de l'indécidabilité.
Remarque : permet aussi de propager RE.

1. A ce propos. Je suis à la recherche d'une preuve du fait suivant (annoncé comme vrai par Kozen) : le problème de décider si le langage d'une machine est décidable/régulier est Σ_3-complet. Repas offert pour toute preuve correcte.
2. D'après certains, les gens qui comprenaient cette théorie mangent désormais les pissenlits par la racine.
3. Les problèmes indécidables par les machines à oracle le problème de l'arrêt, par exemple, sont délicats à utiliser.

2. Etudier l'indécidabilité par la réduction

2.1 Autour des machines de Turing

Théorème de Rice. Exemples de conséquences.
Problème PCP. Indécidabilité.

2.2 En langages formels

Machines linéairement bornées. Décidabilité de l'acceptation mais pas de la vacuité.
Indécidabilité sur les grammaires (linéaires). BP Problèmes indécidables sur les grammaires algébriques. Décidabilité : vacuité, problème du mot, intersection avec un rationnel.
Langages rationnels : tout est décidable d'après un automate. Attention cependant au modèle donné en entrée (pas une machine de Turing...).

3. Décidabilité des théories logiques

3.1 Quelques théories décidables

Définitions (se placer en logique égalitaire) : théorie décidable, théorie complète.
Toute théorie complète aux axiomes D est décidable.
Elimination des quantificateurs sur les ordres denses.
BI Décidabilité de l'arithmétique de Presburger.($\langle \mathbb{N}, =, 1, + \rangle$).
Décidabilité de l'arithmétique de Skolem (avec $\langle \mathbb{N}, =, 1, \times \rangle$).

3.2 L'arithmétique

Arithmétique de Peano (axiomes en annexe). Modèle $\langle \mathbb{N}, =, +, \times \rangle$.
Lemme de représentation dans Peano et dans $\langle \mathbb{N}, =, +, \times \rangle$.
Indécidabilité de Peano et de la théorie du $\mathbb{N}$-modèle.
Conséquences : incomplétude de Peano (et de ses extensions dont les axiomes sont D), $\langle \mathbb{N}, =, +, \times \rangle$ n'est pas axiomatisable par des axiomes décidables.

915 Classes de complexité. Exemples.

Développements choisis

— BK Théorèmes de hiérarchie en espace et en temps.
Pertinence : ★★★★★
— BL Théorème de Berman (langage unaires).
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le jury attend que le candidat aborde à la fois la complexité en temps et en espace. Il faut naturellement exhiber des exemples de problèmes appartenant aux classes de complexité introduites, et montrer les relations d'inclusion existantes entre ces classes, en abordant le caractère strict ou non de ces inclusions. Le jury s'attend à ce que les notions de réduction polynomiale, de problème complet pour une classe, de robustesse d'une classe vis à vis des modèles de calcul soient abordées. Parler de décidabilité dans cette leçon serait hors sujet.

Avis.

La référence fiable est [AB09]. On pourra aussi s'inspirer des résultats [Car08], mais il faut faire attention aux preuves et aux hypothèses des théorèmes...
On commence par définir les notions de complexité liées aux machines de Turing. Ensuite, on se concentre (pour des raisons pédagogiques) sur deux classes particulières : P et NP, qui ne requièrent pas d'autre théorie. Cela nous permet de parler de réductions et de complétude.

1. Modèle de calcul et notion de complexité

1.1 Complexité des machines de Turing

Rappel : machines de Turing à plusieurs bandes (entrée/travail/sortie).
Définition : Complexité d'un calcul (espace temps borné pour toutes les branches).

1.2 Robustesse des notions

Speedup en espace, en temps (attention aux bornes).
Changement d'alphabet de travail, passage à une bande. Remarque : palindromes.
Simulation du temps ND f en temps $2^{\mathcal{O}(f)}$.

1.3 Classes de complexité

Fonction propre (motivée par les restrictions précédentes).
Définitions : DTIME, NTIME, DSPACE, NSPACE. Classes duales. Croissance.

2. Temps polynomial

Classes P, NP. Exemples de problèmes.
Réduction polynomiale. Transitivité. NP-complétude.
Théorème de Cook : NP-complétude de SAT, de 3SAT.
Problèmes NP-complets : chemin hamlitonien, k-clique, subset sum.
BL Théorème de Berman (langage unaires).

3. D'autres classes

3.1 Espace polynomial

Définition PSPACE. NP ⊆ PSPACE.
Savitch : NSPACE($f(n)$)) ⊆ DSPACE($f(n)^2$) pour $f(n) \geq n$. Conséquence PSPACE = NPSPACE.
Problèmes PSPACE-complets. QSAT. Stratégie dans un jeu.
PSPACE ⊆ EXP.

3.2 Espace logarithmique

Classes et NL. Réductions en espace logarithmique. Transitivité.
Problèmes P complets. Hornsat. Clôture transitive.
Problèmes NL complets. Accessibilité. 2SAT est co-NL-complet.
Immerman-Szelepcsényi : non-accessibilité est dans NL. Conséquence NL = co – NL.

4. Simulation et théorèmes de hiérarchie

Théorème de simulation : temps $\alpha_M T^2$, espace $\beta_M E$.
BK Théorèmes de hiérarchie en espace et en temps.
Remarque sur les théorèmes non-déterministes.
Applications aux classes usuelles.

916 Formules du calcul propositionnel : représentation, formes normales, satisfiabilité. Applications.

Développements choisis

— BM 2SAT en temps linéaire.
Pertinence : ★★★★★
— BO Complétude de la résolution propositionnelle.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le jury attend des candidats qu'ils abordent les questions de la complexité de la satisfiabilité. Pour autant, les applications ne sauraient se réduire à la réduction de problèmes NP-complets à SAT. Une partie significative du plan doit être consacrée à la représentation des formules et à leurs formes normales.

Avis.

On propose un plan en trois parties, inspiré du titre : formules/formes normales, représentations/satisfiabilité, aspects pratiques. Le début de la présentation se trouvera dans [CL03]. La résolution intervient comme utilisation des formes CNF, et sert à obtenir la compacité. Les ROBDD sont un très bel exemple de forme normale & de représentation, qui a l'intérêt d'être *canonique*, mais aussi d'être utile en pratique.

Plan.

1. Formules du calcul propositionnel

1.1 Syntaxe

Plus petit ensemble contenant $p \in \mathcal{P}$ stable par $\wedge, \vee, \rightarrow, \iff, \neg$.
Union des $\mathcal{F}_n$, formules de hauteur n. Méthode : preuve par induction sur la hauteur.
Représentation sous forme d'arbre, représentation parenthésée.
Possibilité de substituer.

1.2 Sémantique

Interprétation (des variables). Extension aux formules.
Indépendance des variables absentes. Table de vérité.
Exemples de modélisation. Résolution de problèmes.
Formules satisfiables, tautologies, antilogies. Externalisation des équivalences.
Exemples de formules équivalentes.

2. Formes normales

2.1 Fonction booléennes, systèmes de connecteurs

Fonctions booléennes. Représentations par les formules.
Systèmes complets : or, xor.
Connecteur if then else des BDD.

2.2 Formes normales conjonctive et disjonctive

Construction. Trouver une formule CNF/DNF équivalente.
Trouver une formule CNF équisatisfiable.
Définition : résolvante. Etre clair sur les duplications.
Correction. Complétude.
Conséquence : compacité de la logique propositionnelle.
Remarque : utile pour la complétude de résolution pour le calcul des prédicats.

2.3 Diagrammes binaires de décision

BDD : DAG étiqueté par les variables et $\top, \bot$, avec degré sortant 2 sauf $\top$ et $\bot$ de degré 0.
Formule associée à un BDD, connecteurs.
OBDD : BDD dont les variables sont ordonnées.
ROBDD : OBDD tel que 1. deux arêtes sortantes ne pointent jamais vers le même sous graphe 2. n'a pas deux sous-graphes isomorphes.
Théorème : deux formules sont équivalentes ssi leurs ROBDD sont syntaxiquement les mêmes.
Construction des ROBDD : calcul récursif avec hash consing. On regarde la variable qui apparaît en tête pour maintenir l'ordre.

3. Problèmes de satisfiabilité

3.1 Cas général, NP-complétude

Théorème de Cook-Levin : NP-complétude de SAT, de 3SAT.
Remarque : la difficulté de la satisfiabilité dépend du modèle.
Applications : d'autres réductions clique, ham...

3.2 Quelques cas plus simples

BM 2SAT en temps linéaire.NL-complétude.
Clauses de Horn. P-complétude.

3.3 SAT solveurs et applications.

Principe de DPLL : backtrack + (supprimer les clauses vraies + propagation unitaire + littéraux purs).
Applications : réductions dans l'autre sens (hampath, ...).

918 Systèmes formels de preuve en logique du premier ordre. Exemples.

Développements choisis

— BN Complétude de la déduction naturelle (Henkin).
Pertinence : ★★★★★
— BO Complétude de la résolution propositionnelle.
Pertinence : ★★☆☆☆

Rapport 2018 du jury.

Le jury attend du candidat qu'il présente au moins la déduction naturelle ou un calcul de séquents et qu'il soit capable de développer des preuves dans ce système sur des exemples classiques simples. La présentation des liens entre syntaxe et sémantique, en développant en particulier les questions de correction et complétude, et de l'apport des systèmes de preuves pour l'automatisation des preuves est également attendue. Le jury appréciera naturellement si des candidats présentent des notions plus élaborées comme la stratégie d'élimination des coupures mais est bien conscient que la maîtrise de leurs subtilités va au-delà du programme.

Avis.

A part faire une liste de systèmes de preuves, je ne vois pas comment organiser le plan. En échange de cette lâcheté intellectuelle, on réfléchira à l'intérêt de chaque système de preuve.
La déduction naturelle est la plus... hum... naturelle, dans la mesure où elle s'apparente aux preuves[1] que l'on fait tous les jours. Ses axiomes font assez clairement ressortir les notions logiques que l'on voudrait utiliser : tiers-exclus, généralisation, *ex falso*[2],etc.
Le calcul des séquents de Gentzen va symétriser la déduction naturelle en redéfinissant la notion de séquent. En montrant qu'il est équivalent à la déduction naturelle, on récupère les théorèmes de correction et de complétude. On regarde ensuite ses spécificités, conséquences de l'élimination des coupures : la normalisation des preuves. La propriété de la sous-formule est utile pour limiter les recherches de preuve !
En résolution, on ne cherche plus à prouver qu'une formule est conséquence d'une théorie, mais qu'on obtient une contradiction en ajoutant sa négation à la théorie. On peut alors avoir beaucoup moins de règles, et surtout lever une forme de non-déterminisme : celui du choix des termes à introduire (grâce aux MGU). Ce système est utilisé pour automatiser les preuves (cf Prolog).
On pourrait aussi insister sur la logique intuitionniste, ou parler des systèmes à la Hilbert (qui ont au moins un intérêt philosophique et historique).

Plan.

1. Déduction naturelle

1.1 Séquents, validité et démonstration

Notion de séquent (avec une seule formule à droite). Sémantique.
Règles de la déduction naturelle.
Remarque : plus ou moins de sucre syntaxique, règles admissibles.
Exemple de preuve.
Théorème : correction.

1. Hélas informelles.
2. A ce sujet, je conseille *vivement* au lecteur la découverte de Falso sur http://inutile.club/estatis/falso.

1.2 Correction et complétude

Lemme de déduction, lemme de généralisation.
Equivalence des notions de complétude.
Témoins de Henkin, théorie complète.
BN Complétude de la déduction naturelle (Henkin).

2. Calcul des séquents

2.1 Séquents, validité et démonstration

Notion de séquent : nouvelle définition (plein de formules à droite). Sémantique.
Règles du calcul des séquents.

2.2 Equivalence avec la déduction naturelle

Preuves d'équivalence.

2.3 Elimination des coupures et conséquences

Théorème : élimination des coupures.
Normalisation des preuves.
Propriété de la sous-formule. Application.

3. Résolution

3.1 Prérequis : clauses et unification

Clauses. Mise en forme clausale de toute théorie.
Unification. Existence des MGU. Algorithme de Robinson.

3.2 Résolution au premier ordre

Règles de résolution : résolution, factorisation.
Exemple de preuve.
Théorème : correction.

3.3 Théorème de Herbrand et complétude

Modèle de Herbrand, base de Herbrand.
Théorème : S a un modèle ssi S a un modèle de Herbrand ssi $S\Sigma$ est propositionnellement sat.
Règles de la résolution propositionnelle. Théorème de relèvement.
BO Complétude de la résolution propositionnelle.
Corollaire : complétude de la résolution du premier.

921 Algorithmes de recherche et structures de données associées.

Développements choisis

— BC Arbres AVL.
Pertinence : ★★★★★
— BD Hachage parfait.
Pertinence : ★★★★★

Rapport 2018 du jury.

Le sujet de la leçon concerne les algorithmes de recherche : les structures de données proposées doivent répondre à une problématique liée aux algorithmes, et la leçon ne peut donc être structurée sur la base d'un catalogue de structures de données. La recherche d'une clé dans un dictionnaire sera ainsi par exemple l'occasion de définir la structure de données abstraite « dictionnaire », et d'en proposer plusieurs implantations concrètes. De la même façon, on peut évoquer la recherche d'un mot dans un lexique : les arbres préfixes (ou digital tries) peuvent alors être présentés. Mais on peut aussi s'intéresser à des domaines plus variés, comme la recherche d'un point dans un nuage (et les quad-trees), et bien d'autres encore.

Avis.

Le cœur de cette leçon est la construction efficace d'une structure de données *dictionnaire*, avec une table de hachage. Pour y arriver, on passera par des structures plus naïves. On pourra mentionner la recherche d'un mot dans un texte (mais ce n'est pas essentiel), ici avec deux approches duales : un automate (pré-calcul sur le motif), et un suffix tree (pré-calcul sur le texte).

Plan.

1. Recherche d'un élément donné

Idée : recherche d'un élément dans un ensemble dynamique.
Type abstrait dictionnaire.

1.1 Listes et tableaux

Liste. Problème : linéaire.
Tableau trié, recherche dichotomique. Problème : pas dynamique.

1.2 Arbre binaire de recherche

Définition : ABR. Opérations en $\mathcal{O}(h)$.
BC Arbres AVL.

1.3 Tables de hachage

Idée générale du hachage. Problèmes des collisions.
Familles universelles, hachage universel.
Cas où l'ensemble est statique.
BD Hachage parfait.

2. Recherche d'un élément d'ordre donné

2.1 Sélection en temps linéaire

Selection du k-ième élément dans un tableau non-trié.

2.2 Recherche du minimum

Objectif : trouver le min en temps constant.
Structure de tas binaire. Implémentation dans un tableau.
Application : tri par tas.

3. Cas particulier : recherche de motif dans un texte

Hypothèse : alphabet Σ fixé (qui n'apparaîtra pas dans les complexités).

3.1 Automates : recherche d'un motif donné

Propriété : Il existe un automate $|m| + 1$ états reconnaissant $\Sigma^* m$ (avec les bordures).
Propriété : recherche dans un texte t en $\mathcal{O}(|t|)$.
Algo : On peut construire cet automate en temps $\mathcal{O}(|m|)$.

3.2 Suffix trees : recherche dans un texte donné

Définition : *suffix trie* de t. Exemple.
Propriété : recherche du motif m en $\mathcal{O}(|m|)$.
Algo : construction en ligne en temps $\mathcal{O}(|t|^2)$ (avec les suffix links).
Définition : *suffix tree* de t. Exemple. Taille $\mathcal{O}(|t|)$.
Algo (Ukkonen) : construction en ligne du suffix tree en $\mathcal{O}(|t|)$ (subtil).

923 Analyse lexicale et syntaxique. Exemples.

Développements choisis

— BP Problèmes indécidables sur les grammaires algébriques.
Pertinence : ★★★★☆
— BH Grammaire LL(1) et table d'analyse.
Pertinence : ★★★★★

Rapport 2018 du jury.

Cette leçon ne doit pas être confondue avec la 909, qui s'intéresse aux seuls langages rationnels, ni avec la 907, sur l'algorithmique du texte. Si les notions d'automates finis, de langages rationnels et de grammaires algébriques sont au coeur de cette leçon, l'accent doit être mis sur leur utilisation comme outils pour les analyses lexicale et syntaxique. Il s'agit donc d'insister sur la différence entre langages rationnels et algébriques, sans perdre de vue l'aspect applicatif : on pensera bien sûr à la compilation. On pourra s'intéresser à la transition entre analyse lexicale et analyse syntaxique, et on pourra présenter les outils associés classiques, sur un exemple simple. Les notions d'ambiguïté et l'aspect algorithmique doivent être développés. La présentation d'un type particulier de grammaire algébrique pour laquelle on sait décrire un algorithme d'analyse syntaxique efficace sera ainsi appréciée. Le programme 2018 permet de nouveaux développements pour cette leçon avec une ouverture sur des aspects élémentaires d'analyse sémantique.

Avis.

Attention, cette leçon n'est pas l'antique "910 Langages algébriques. Exemples et applications." ni la "911 Automates à pile. Exemples et applications.". Il faut donc sortir du cadre théorique des langages formels pour s'intéresser *effectivement* à la compilation. L'analyse lexicale permet de parler d'automates et d'expressions rationnelles. On ne négligera pas le problème des "lexèmes les plus longs" qui se résout efficacement par programmation dynamique. Ensuite, on motivera l'emploi de grammaires par des constructions naturelles de programmation, comme les "if/then" imbriqués. Après avoir essayé quelques moyens d'obtenir des arbres de syntaxe, on se restreint au cas des grammaires (fortement) LL pour lesquelles on donne un algorithme efficace.
On pourrait parler d'analyse LR ou SLR, mais la place manque. Enfin, on s'est tourné vers des questions de compilation, mais d'autres applications sont possibles : analyse des langues naturelles, analyse de données génétiques en biologie...

Plan.

1. Chaîne de compilation

Objectif : traduire un code d'un langage source (haut niveau) vers un langage cible (bas niveau), en préservant sa sémantique.
texte → LEXER → PARSER → arbre de syntaxe abstraite → PARTIE AVANT → binaire
Exemples : C et gcc, détail sur une expression arithmétique.
Remarque : différence avec un interpréteur (Python).

2. Analyse lexicale

2.1 Idée générale

Notion de lexème. Unité lexicale : constructeur et valeur. Lexème : instance d'une unité lexicale. Motif : description concise des lexèmes. Exemple : unité lexicale = nombre, motif = suite de $0, \dots, 9$, exemple de lexème = 2.
Objectif. Résoudre le problème d'analyse lexicale. Entrée : texte d'un programme + unité lexicales. Sortie : suite d'unités lexicales découpant le texte. Remarque : priorités sur les règles.

2.2 Implémentation par automate fini

Fondements théoriques. Idée : associer E_i rationnelle à motif m_i.
Théorème de Kleene. Parler de la complexité (déterminisation).
Construction de l'analyseur.

1. construire un automate $\mathcal{A}_i$ pour chaque E_i ;
2. faire $\mathcal{A}$ l'union des $\mathcal{A}_i$;
3. déterminiser $\mathcal{A}$.

Méthode d'analyse naïve. Simuler l'automate sur le texte w, en mémorisant le dernier état final croisé et la position dans le texte. Attention, on cherche le plus long lexème possible, ce qui fait parfois revenir en arrière. Complexité : $\mathcal{O}(|w|^2)$ au pire. Atteint avec a, a^*b sur $w := a^n$.
Méthode d'analyse par programmation dynamique. Sur l'entrée w, calculer $T(q,i) = (q', j)$ où $u := w[i] \dots w[j]$ est le plus long facteur commençant en $w[i]$ tel que $q \to^u q'$. Peut se calculer récursivement par programmation dynamique (en partant de $i = |w|$). Complexité : $\mathcal{O}(|Q||w|)$.
Remarque : il n'était plus nécessaire d'avoir déterminisé l'automate !

3. Analyse syntaxique

Objectif : produire un AST à partir des lexèmes.

3.1 Grammaires algébriques

Définitions. Grammaire. Dérivation, arbre de dérivation. Motivation "if/then" non-rationnel.
Problème du mot. Algorithme CYK. Complexité $\mathcal{O}(|w|^3)$. Générique mais trop long.
Ambiguïté. Définitions. BP Problèmes indécidables sur les grammaires algébriques.

3.2 Automates à pile

Définitions. Automate à pile. Acceptation par pile vide. Remarque : autres modes d'acceptation.
Construction d'un automate. Grammaire → automate par les expansions/vérification. Dérivations gauches. Simulation effective, problème du non-déterminisme.
Remarque : réciproquement, on a automate → grammaire.

3.3 Analyse descendante

Table prédictive. Idée : regarder des lettres à l'avance pour savoir quelle expansion faire. Définition : fonction $Z \times \Sigma^{\leq k} \to$ ensemble des regles . Analyseur utilisant cette table.
First et follow. Définition de First_k. Calcul effectif par saturation. Définition de Follow_k. Calcul effectif par saturation.
Analyse LL. Grammaire fortement LL, grammaire LL. Non-ambiguïté.
BH Grammaire LL(1) et table d'analyse.

924 Théories et modèles en logique du premier ordre. Exemples.

Développements choisis

— BI Décidabilité de l'arithmétique de Presburger.
Pertinence : ★★★★☆
— BN Complétude de la déduction naturelle (Henkin).
Pertinence : ★★★★☆

Rapport 2018 du jury.

Le jury s'attend à ce que la leçon soit abordée dans l'esprit de l'option informatique, en insistant plus sur la décidabilité/indécidabilité des théories du premier ordre que sur la théorie des modèles. Il est attendu que le candidat donne au moins un exemple de théorie décidable (respectivement complète) et un exemple de théorie indécidable. Si le jury peut s'attendre à ce que le candidat connaisse l'existence du théorème d'incomplétude, il ne s'attend pas à ce que le candidat en maîtrise la démonstration.

Avis.

Le plan ne doit pas se limiter à Théories/Modèles, mais peut contenir ces deux parties. Ici, on supposera connues la syntaxe et la sémantique de la logique du premier ordre.
On commence par étudier uniquement des aspects syntaxiques. On rappelle la notion de démontrabilité, puis on l'utilise pour définir quelques propriétés des théories logiques. On obtient rapidement des premiers résultats de décidabilité : par la "recherche de preuve" et par l'élimination des quantificateurs. Les modèles nous ouvrent ensuite la sémantique des théories. Essentiels : correction et complétude. Quelques conséquences "divertissantes" en théorie des modèles. Enfin, on met en pratique les résultats précédents dans le cadre de l'arithmétique, à la fois assez restreint et assez riche.

Plan.

Prérequis : syntaxe des formules, interprétation d'une formule dans une structure.

1. Théories : syntaxe et démontrabilité

1.1 Démontrabilité en logique du premier ordre

Séquents (syntaxe). Déduction naturelle (règles en annexe).
Remarque : sucre syntaxique, règles supplémentaires admissibles.

1.2 Propriétés des théories

Définition : théorie, axiomes.
Exemples : théorie de l'égalité, théorie des groupes.
Définition : théorie complète, théorie cohérente, théorie décidable. Exemples.
Propriété : complète+axiomes décidables $\Rightarrow$ décidable.

1.3 L'élimination des quantificateurs

Définition : éliminer les quantificateurs $\exists$. Effectivité.
Propriété : décidable pour les formules sans variables $\Rightarrow$ décidable.
Questions de complétude.

2. Modèles : aspects sémantiques

2.1 Correction et complétude

Définition : structure modèle d'une théorie. Définition : théorie associée à un modèle (complète).
Interprétation égalitaire vs non-égalitaire.
Exemple : modèle de la théorie des groupes, de la théorie de l'égalité.
Théorème : correction.
Lemme : équivalence des trois notions de complétude. Témoins de Henkin.
Théorème : BN Complétude de la déduction naturelle (Henkin). Variante en logique égalitaire.
Exemples de théories cohérentes.

2.2 Applications en théorie des modèles

Propriété : complétude $\iff$ tous les modèles valident les mêmes formules.
Théorème : Löwenheim-Skolem. Application : paradoxe de Skolem.
Compacité. Applications : de Brujin Erdös, 4 couleurs infini...

3. Exemple détaillé : l'arithmétique

3.1 Cas général : indécidabilité et incomplétude

Axiomes de Peano. Modèle standard $\langle \mathbb{N}, +, \times, = \rangle$.
Lemme de représentation des fonctions récursives : $f(n_1, \ldots, n_m) = q$ démontrable $\iff$ $A[q, n_1, \ldots, n_m]$ démontrable dans PA $\iff$ $A[q, n_1, \ldots, n_m]$ valide dans $\mathbb{N}$.
Corollaire : indécidabilité. Corollaire : incomplétude.
Corollaire : incomplétude de toute sur-théorie raisonnable.

3.2 Sous-théories complètes et décidables

Arithmétique de Presburger $\langle \mathbb{N}, +, = \rangle$. BI Décidabilité de l'arithmétique de Presburger.
Arithmétique de Skolem $\langle \mathbb{N}, \times, = \rangle$. Décidabilité.

925 Graphes : représentations et algorithmes.

Développements choisis

— BG Correction de l'algorithme de Dijkstra.
Pertinence : ★★★★★
— BM 2SAT en temps linéaire.
Pertinence : ★★★★☆

Rapport 2018 du jury.

Cette leçon offre une grande liberté de choix au candidat, qui peut décider de présenter des algorithmes sur des problèmes variés : connexité, diamètre, arbre couvrant, flot maximal, plus court chemin, cycle eulérien, etc. mais aussi des problèmes plus difficiles, comme la couverture de sommets ou la recherche d'un cycle hamiltonien, pour lesquels il pourra proposer des algorithmes d'approximation ou des heuristiques usuelles. Une preuve de correction des algorithmes proposés sera évidemment appréciée. Il est attendu que diverses représentations des graphes soient présentées et comparées, en particulier en termes de complexité.

Avis.

Les graphes interviennent naturellement dans nombre de problèmes pratiques, et c'est essentiellement pour y répondre que l'on cherche des algorithmes efficaces. D'un point de vue plus pédagogique, ces algorithmes (en général courts) permettent d'illustrer auprès des étudiants l'intérêt de prouver la correction, d'étudier la complexité, et enfin de bâtir des types abstraits de données. Il n'est pas nécessaire de mentionner d'autres représentations que *matrice d'adjacence* et *liste d'adjacence*, car la différence entre les deux est assez riche. On comparera par exemple Dijkstra, qui utilise très bien les listes, et Floyd-Warshall, qui travaille efficacement sur les matrices. Cela dit, on pourra dire à l'oral quelques mots de graphes donnés *implicitement* par une représentation très compacte, comme le graphe associé à un jeu.
Le [CLRS10] contient tout ce qu'il faut mettre dans le plan.
Parler de *classes de complexité* est absolument hors sujet[1]. Les développements montrant la difficulté d'un problème de graphes comme la NP-complétude du voyageur de commerce ou de la k-clique n'ont rien à faire là ! Toutefois, on pourra développer un algorithme d'approximation de ces problèmes puisque la motivation de cette leçon est *pratique*.

Plan.

1. Graphes : structure et représentations

Définition : graphe. Dessin. Relation d'accessibilité. Cycles, composantes (fortement) connexes.
Représentation par matrice d'adjacence ou liste d'adjacence.
Comparaisons des coûts en espace/temps pour les opérations de base (présence d'une arête…).

2. Parcours et composantes fortement connexes

Parcours. En largeur, en profondeur. Complexité : $\mathcal{O}(|S| + |E|)$.

Algorithme de Kosaraju. Pour déterminer les CFC. Une série de parcours en profondeur, puis une autre série dans le graphe transposé, en sélectionnant les sommets dans l'ordre inverse de l'ordre de dépilement. Complexité : $\mathcal{O}(|S| + |E|)$.
Application : BM 2SAT en temps linéaire.

1. De l'avis de P. Gastin.

3. Plus courts chemins

Graphe pondéré par $w : A \to \mathbb{R}_+$.

3.1 A origine unique

Bellman-Ford. Déterminer itérativement les plus courts chemins d'au plus k étapes. Permet aussi de détecter les cycles de poids négatif. Complexité : $\mathcal{O}(|A||S|)$ avec listes.

Dijsktra. On suppose $w : A \to \mathbb{R}_+$. On prend le voisin le plus proche. BG Correction de l'algorithme de Dijkstra.. Complexité : $\mathcal{O}(|A|\log(|S|))$ avec un tas binaire, $\mathcal{O}(|A| + |S|\log(|S|))$ avec Fibonacci. Remarque : l'algorithme A^* peut améliorer Dijkstra, sous réserve d'avoir une sous-approximation de la distance.

3.2 Entre toutes paires de sommets

Motivation : itérer Dijsktra en $\mathcal{O}(|A||S| + S^2\log(|S|)) = \mathcal{O}(|S|^2)$, mais impossible si on a des poids négatifs. Itérer Bellman-Ford en $\mathcal{O}(|A||S|^2) = \mathcal{O}(|S|^4)$? Pas terrible.

Floyd-Warshall. Calculer itérativement la matrice des $d^k[i,j]$ (plus courte distance entre i et j passant par les sommets $\leq k$). Complexité : $\mathcal{O}(|S|^3)$ avec une matrice d'adjacence.

4. Arbres couvrants de poids minimal

Propriétés d'un arbre : sous-graphe connexe minimal, ou sans cycle maximal. $|A| = |S| - 1$. Algorithme générique d'ajout d'arêtes.

Kruskal. Trier les arêtes par poids croissant puis les ajouter dans l'ordre (si possible). Complexité : $\mathcal{O}(|A|\log(|S|))$ avec une structure Union-Find.

Prim. Ajouter la plus petite arête accessibles qui ne crée pas de cycle. Complexité : $\mathcal{O}(|A|\log(|S|))$ avec un tas binaire et $\mathcal{O}(|A| + |S|\log(|S|))$ avec Fibonacci.

5. Problèmes de flot

Capacité (avec poids positifs). Flot sur un graphe complet (avec arcs retour). Flot maximal.

Ford-Fuelkerson. Chercher un chemin augmentant et augmenter. Termine si les capacités sont entières, mais peut être très lent.

926 Analyse des algorithmes, complexité. Exemples.

Développements choisis

— BA Complexité du tri rapide aléatoire.
Pertinence : ★★★★☆
— BC Arbres AVL.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Il s'agit ici d'une leçon d'exemples. Le candidat prendra soin de proposer l'analyse d'algorithmes portant sur des domaines variés, avec des méthodes d'analyse également variées : approche combinatoire ou probabiliste, analyse en moyenne ou dans le cas le pire. Si la complexité en temps est centrale dans la leçon, la complexité en espace ne doit pas être négligée. La notion de complexité amortie a également toute sa place dans cette leçon, sur un exemple bien choisi, comme union find (ce n'est qu'un exemple).

Avis.

L'Union-Find (avec compression de chemins et union par rangs) est un bon exemple, mais même la borne $\log^*$ sur sa complexité est trop longue pour tenir en développement.

Plan.

1. Algorithmes et complexité

1.1 Généralités

Définition : complexité temporelle, spatiale d'une exécution.
Remarque : aspect informel, dépend du modèle de calcul.
Complexité dans le meilleur des cas, dans le pire.
Intérêt de l'analyse asymptotique.

1.2 Premiers exemples

Complexité d'une boucle. Floyd-Warshall.
Complexité de tris débiles. On compte les comparaisons.
Complexités spatiales, quelques tableaux.

2. Techniques de calcul

2.1 Equations de récurrences

Récurrences linéaire, Fibonacci.
Récurrences de partition, master théorème. Application au tri fusion.
Selection en temps linéaire.
Remarque : compromis temps/espace en Programmation Dynamique (mémoïzation ou pas).

2.2 Complexité amortie

Technique d'étude, potentiel, etc.
Incrémentation d'un compteur binaire en amorti $\mathcal{O}(1)$.
Tableaux dynamiques.

2.3 Complexité moyenne et analyse probabiliste

Différence entre les deux notions.
Complexité du tri rapide en moyenne.
BA Complexité du tri rapide aléatoire.

3. Complexité et structures de données

3.1 Propriétés de la structure

Idée : maintenir une forme à la structure permet de tenir la complexité.
Arbres binaires de recherche. Cas linéaire. BC Arbres AVL.
Remarque : les tables de hachage sont meilleures (hachage parfait et probas).
Union-Find, borne $\log^*$.

3.2 Application aux algorithmes

Motivation algorithmique : garantir de bonnes complexités sur les opérations d'une structure, c'est garantir une bonne complexité sur les algorithmes qui l'utilisent.
La complexité amortie prend tout son sens, puisqu'on fait des suite d'opérations.
Dijkstra : différence entre $\mathcal{O}(|A|\log(|S|))$ ou $\mathcal{O}(|A| + |S|\log(|S|))$ selon le tas utilisé.
Kruskal : utiliser l'Union-Find pour avoir $\mathcal{O}(|A|\log(|S|))$.

927 Exemples de preuve d'algorithme : correction, terminaison.

Développements choisis

— BG Correction de l'algorithme de Dijkstra.
Pertinence : ★★★★★
— BQ Complétude des règles de Hoare.
Pertinence : ★★★☆☆

Rapport 2018 du jury.

Le jury attend du candidat qu'il traite des exemples d'algorithmes récursifs et des exemples d'algorithmes itératifs. En particulier, le candidat doit présenter des exemples mettant en évidence l'intérêt de la notion d'invariant pour la correction partielle et celle de variant pour la terminaison des segments itératifs. Une formalisation comme la logique de Hoare pourra utilement être introduite dans cette leçon, à condition toutefois que le candidat en maîtrise le langage. Des exemples non triviaux de correction d'algorithmes seront proposés. Un exemple de raisonnement type pour prouver la correction des algorithmes gloutons pourra éventuellement faire l'objet d'un développement.

Plan.

1. Terminaison

Motivation : boucles tantque, définitions récursives.
Théorème : indécidabilité de l'arrêt (formulation soft, avec des algos).
Définition : relation bien fondée. Exemples.

Méthode : terminaison en itératif. du variant de boucle : dépend de l'état courant, à valeurs dans un ensemble bien fondé, qui décroit strictement.
Exemple : Euclide.

Méthode : terminaison en récursif.
Exemples : Euclide, diviser-pour-régner, Ackermann.

Remarque : dépend en fait de la sémantique (appel par nom, par valeurs, vrais entiers...)

2. Correction

Correction totale, partielle.

2.1 Preuves de correction partielle

Méthode : correction en itératif. Invariant de boucle.
Exemples : Euclide, partition du tri rapide, BG Correction de l'algorithme de Dijkstra..

Méthode : correction en récursif. Induction bien fondée.
Exemple : binôme de Newton, diviser pour régner.

2.2 Un exemple détaillé : correction générique des gloutons

Matroïde : hérédité, échange. Propriété d'augmentation. Bases.
Exemples : forêts, familles libres, tâches.
Algorithme du glouton générique : indépendant de poids maximal. Correction.

3. Vers une formalisation des preuves : logique de Hoare

Syntaxe/Sémantique de IMP : langage, commandes.
Triplet de Hoare. Validité.
Règles de preuve de Hoare : skip, affectation, séquence, if, while, pre/post.
Théorème de correction.
Plus faible pré-condition (sémantique). Théorème (admis) : la syntaxe permet d'exprimer les wp.
BQ Complétude des règles de Hoare.
Remarque : en pratique on annote les programmes plutôt que de calculer des wp.

928 Problèmes NP-complets : exemples et réduction.

Développements choisis

— BL Théorème de Berman (langage unaires).
Pertinence : ★★★★☆
— BM 2SAT en temps linéaire.
Pertinence : ★★☆☆☆

Rapport 2018 du jury.

L'objectif ne doit pas être de dresser un catalogue le plus exhaustif possible ; en revanche, pour chaque exemple, il est attendu que le candidat puisse au moins expliquer clairement le problème considéré, et indiquer de quel autre problème une réduction permet de prouver sa NP-complétude. Les exemples de réduction polynomiale seront autant que possible choisis dans des domaines variés : graphes, arithmétique, logique, etc. Un exemple de problème NP-complet dans sa généralité qui devient P si on contraint davantage les hypothèses pourra être présenté, ou encore un algorithme P approximant un problème NP-complet. Si les dessins sont les bienvenus lors du développement, le jury attend une définition claire et concise de la fonction associant, à toute instance du premier problème, une instance du second ainsi que la preuve rigoureuse que cette fonction permet la réduction choisie.

Avis.

Comme suggéré par le rapport du jury, on regarde les aspects pratiques/positifs des notions de NP-complétude. D'abord la possibilité de contraindre davantage les hypothèses d'un problème pour le rendre plus facile, ensuite la possibilité d'accepter une réponse "approchée". Enfin, on s'intéresse brièvement à la philosophie des SAT-solveurs, qui exploite les réductions dans un sens *inverse* de ce que l'on avait fait avant !

Plan.

1. Classe NP et réductions polynomiales

Rappel : classes P et NP. Inclusion. Réduction polynomiale, stabilité, transitivité.
Problèmes de décision. Remarque : passage de décision au comptage, astuce du $\leq n$.
Equivalence NP/certificat polynomial.

2. Exemples de problèmes NP-complets

2.1 Problème SAT

Problème SAT. NP-complétude. NP-complétude de 3SAT.
Remarque : dépend de comment sont données les formules (DNF, BDD c'est linéaire).
Remarque : utile en pratique pour éviter les machines de Turing.
BL Théorème de Berman (langage unaires).

2.2 Problèmes de graphes

Clique. Couverture de sommet. Chemin Ham. Voyageur de commerce.

2.3 Algorithmes de calcul

Parler de pseudo-polynomial.

3. La NP-complétude en pratique

3.1 Changer d'objectifs pour simplifier le travail

Contraindre un peu plus les hypothèses. BM 2SAT en temps linéaire.

Algorithmes d'approximation. Couverture de sommets. Chemin hamiltonien (cas euclidien). Inapproximabilité du voyageur de commerce.

3.2 Optimisation de SAT

SAT-solveurs. DPLL : backtrack + (clauses vraies + propagation unitaire + littéraux purs).
Applications : réductions dans l'autre sens (hampath, ...) !

929 Lambda-calcul pur comme modèle de calcul. Exemples. (I)

Impasse Stratégique...

Rapport 2018 du jury.

Il s'agit de présenter un modèle de calcul : le lambda-calcul pur. Il est important de faire le lien avec au moins un autre modèle de calcul, par exemple les machines de Turing ou les fonctions récursives. Néanmoins, la leçon doit traiter des spécificités du lambda-calcul. Ainsi le candidat doit motiver l'intérêt du lambda-calcul pur sur les entiers et pourra aborder la façon dont il permet de définir et d'utiliser des types de données (booléens, couples, listes, arbres).

Avis.

Une leçon technique et austère, d'autant qu'on demande à rester dans le cadre d'un calcul non-typé. Ce qui manque surtout, c'est une référence sur le sujet (les livres sont en général assez vieux, et assez durs). Pour éviter de se perdre, on pourra insister sur la "construction" : prendre le temps de définir les termes, la substitution, la réduction... Pour l'aspect "modèle de calcul", on parlera bien sûr de l'équivalence avec les fonctions récursives, mais aussi des spécificités de l'approche notamment avec les stratégies de réduction, et c'est délicat.

930 Sémantique des langages de programmation. Exemples.

Développements choisis

— BQ Complétude des règles de Hoare.
Pertinence : ★★★☆☆
— BR Adéquation dénotationnel/grands pas. (NR)
Pertinence : ★★★★★

Rapport 2018 du jury.

L'objectif est de formaliser ce qu'est un programme : introduction des sémantiques opérationelle et dénotationelle, dans le but de pouvoir faire des preuves de programmes, des preuves d'équivalence, des preuves de correction de traduction. Ces notions sont typiquement introduites sur un langage de programmation (impératif) jouet. On peut tout à fait se limiter à un langage qui ne nécessite pas l'introduction des CPOs et des théorèmes de point fixe généraux. En revanche, on s'attend ici à ce que les liens entre sémantique opérationelle et dénotationelle soient étudiés (toujours dans le cas d'un langage jouet). Il est aussi important que la leçon présente des exemples d'utilisation des notions introduites, comme des preuves d'équivalence de programmes ou des preuves de correction de programmes.

Avis.

Une leçon toute neuve. Piocher dans les chapitres 2, 4, 5, 6 et 7 de [Win93].
On restera dans le cas d'un langage simple : IMP (qui ne permet pas d'écrire des fonctions, et ce n'est pas plus mal). L'objectif, comme indiqué dans le rapport, est d'arriver à faire des preuves d'équivalence (ce qui peut servir en optimisation de code, par exemple) ainsi que des preuves de correction (avec la logique de Hoare, riche en conséquences). Bien entendu, il va falloir formaliser la notion de "programme" pour y parvenir.
On commence par la syntaxe. Donner des sémantiques aux expressions arithmétiques permet déjà de voir les différences entre petits pas, grands pas et dénotationelle. On notera d'ailleurs que le sens d'évaluation pourrait avoir une influence. Ensuite, les définitions opérationnelles s'étendent naturellement aux commandes. La sémantique dénotationelle requiert un peu plus de travail.

Plan.

1. Première approche

1.1 Le langage IMP

Syntaxe : Expressions et commandes.
Exemple : Euclide.
Environnement pour les variables.
Remarque sur l'expressivité, l'absence de fonctions.

1.2 Sémantique des expressions

Sémantique à petits pas (réécriture avec des contextes). A grands pas.
Sémantique dénotationelle.
Théorème : équivalence des 3.

2. Sémantiques opérationnelles

2.1 Sémantique à petits pas

Sémantique à petits pas. [Win93], p 24.
Exemple. Remarque sur les choix gauche/droite. Terminaison.

2.2 Sémantique à grands pas

Sémantique à grands pas. [Win93], p 20.
Exemple.
Correction et adéquation des deux sémantiques.

3. Sémantique dénotationelle

Sémantique dénotationelle (de manière *fonctionnelle* et pas relationnelle). [Win93], p 58
Exemples. Correction (par rapport aux grands pas).
BR Adéquation dénotationnel/grands pas. (NR)

4. Application aux preuves de programmes

4.1 Equivalence, optimisation de code

Variables inutiles.
Débouclage de boucle, parité.

4.2 Logique de Hoare

Triplet de Hoare. Validité.
Règles de preuve de Hoare : skip, affectation, séquence, if, while, pre/post.
Théorème de correction.
Plus faible pré-condition (sémantique). Théorème (admis) : la syntaxe permet d'exprimer les wp.
BQ Complétude des règles de Hoare.
Remarque : en pratique on annote les programmes plutôt que de calculer des wp.

Troisième partie

Divers

1 Statistiques.

Nombre de développements : 43 (25 maths - 18 info[1])
Nombre de leçons d'algèbre : 20 (20 couvertes).
Nombre de leçons d'analyse : 22 (21 couvertes).
Nombre de leçons d'informatique : 21 (19 couvertes).

1. C'est assez peu.

2 Des avis sur les références.

> Beaucoup de gens sont ainsi faits. De ce qu'ils ont bien voulu convenir un jour que vous n'êtes pas sans quelque valeur, vous êtes tenus pour cela de les admirer à jamais, sans restriction, dans tout ce qu'il leur plaira de faire… ou de défaire ; sous peine d'être traité d'*ingrat*.
>
> *Mémoires*, Hector Berlioz.

Opinion personnelle sur certains livres qui m'ont plu… ou énervé.

— [Rou14], aka Rouvière.
Pertinence : ★★★★★
Tout simplement excellent et complet ! Et dire que je n'aimais pas le calcul différentiel, mais ça c'était avant. Petit bémol : on voudrait parfois plus de preuves du cours.

— [CLRS02], aka Cormen. INFO
Pertinence : ★★★★☆
La référence complète pour les questions algorithmiques. Attention cependant, ce livre est écrit pour des gens qui n'ont pas fait beaucoup de mathématiques, ce qui peut le rendre désagréable quand on a déjà du recul (allez, je vais montrer des propriétés évidentes sur les racines de l'unité). Paradoxalement, certaines preuves (d'algorithmes, de probabilités…) sont faites sans grande rigueur.

— [Rom17], aka Rombaldi.
Pertinence : ★★★★☆
Tout frais, tout neuf (en 2018). Les chapitres sont les noms de certaines des leçons d'algèbre, ce qui donne une base intéressante pour les plans. On voudrait parfois le voir aller plus loin, mais le livre est déjà gros.

— [FGN01a], aka les Francinou (ou FGN).
Pertinence : ★★★★☆
Bien connus de certains élèves de prépa qui croient naïvement que ce sont tous de vrais oraux X-ENS [1], alors qu'ils auraient dû s'intituler "Développements d'agrégation avec les outils de classe prépa" ou "Bréviaire du colleur de MP***", mais les ventes en auraient pâti. Très utiles pour trouver des développements, qui sont en général déjà calibrés et parfaitement clairs. Les preuves parfois tendent à oublier la théorie générale (au programme de l'agrégation) pour se concentrer sur un cas particulier avec des astuces (esprit prépa), au point de "sortir les choses de leur contexte" [2]. Se renseigner donc un peu sur ce contexte pour éviter des questions méchantes du jury (dont les auteurs ont fait/font partie).

— [CG13], aka H2G2.
Pertinence : ★★★★☆
Ecrit avec beaucoup d'humour et de décontraction, sa lecture fait passer un bon moment. Il y a plein de bonnes idées à mettre ça et là, mais il va parfois beaucoup trop loin pour l'agreg (notamment presque tout le tome second). La nouvelle version "Nouvelles histoires hédonistes", où les exercices sont corrigés, semble se tourner d'avantage vers l'objectif agrégation.

— [CR94], aka Crochemore. INFO
Pertinence : ★★★★☆
Incontournable dans l'algorithmique du texte. Malheureusement, il est écrit en anglais et pas toujours très clair. Pas d'erreurs cependant, mais il reste que les preuves mériteraient d'être *beaucoup* plus détaillées.

— [QZ13], aka Zuily-Quéffélec.
Pertinence : ★★★☆☆
Attention les prérequis en analyse sont non-négligeables, et certaines parties (vers la fin) sont assez loin du niveau de l'agreg. De bonnes idées, mais les preuves sont rédigées de

1. Et donc que les livres sont *indispensables* pour ces concours. La bonne blague.
2. Selon l'expression de C. Picaronny.

manière… étrange[1]. Plus agréable que [BMP05], cela dit.

— [BMP05], aka Objectif Agrégation.
Pertinence : ★☆☆☆☆
Un ouvrage qui se vendait “pour les agrégatifs” et qui partait de la bonne intention de réunir une mine de développements. Le résultat est extrêmement décevant, parce que les auteurs ont cédé à une curieuse envie de faire dans le détail inutile, ce qui rend la majorité des preuves incompréhensible. Les idées principales n’apparaissent pas de prime abord, parce que cachées au milieu de remarques évidentes. Il faut intensément retravailler tout ce qu’on prend dedans, et relire d’autres versions des démonstrations.

1. Mettons que cela ne cadre pas avec la manière dont je conçois la réalité du monde.

3 Ce qu'il m'est arrivé à l'oral.

J'ai été reçu à l'agrégation en 2018.

En maths

J'avais le choix entre 157 et 218. Petite note : les couplages de maths pour les informaticiens tendent à être de la forme algèbre-analyse, et la 157 tombe particulièrement souvent. Même si la 157 permettait davantage d'envolées lyriques (encore que...), je préférais mes développements dans la 218. J'ai donc présenté un plan assez banal (Aspects locaux, Aspects globaux, Topologie des fonctions) et suis passé sur AH Translatées d'une fonction dérivable.

Note. Je voulais mettre une annexe avec des tableaux, en 4ème page de mon plan. Malheureusement, l'apparitrice[1] a *déchiré* ma feuille en se fendant d'un "seules les figures géométriques sont autorisées en annexes, je vous l'avais bien dit hahaha"[2]. Devant cette mésinterprétation du terme *figure*, j'ai laissé faire. Mais par la suite, une discussion avec le président du jury (par ailleurs très sympathique) m'a confirmé que cela n'aurait pas dû avoir lieu[3].

En informatique

Le couplage était 921/925. Il me semblait maîtriser davantage le sujet des graphes, donc j'ai pris la 925 en donnant le plan et les motivations décrites ci-haut. Le jury a explicitement apprécié que je ne discute pas de problèmes NP-complets sur les graphes, car *ce n'est pas le sujet*. J'ai présenté BG Correction de l'algorithme de Dijkstra. Chose troublante, le jury m'a interrompu dans mon développement en me demandant d'expliquer vaguement ce que faisait Dijkstra.

Questions : (pas d'exercices en info)

1. Vous l'avez dit, mais où intervient la positivité des poids ?
2. Re-prouvez l'algorithme en prenant l'invariant "standard" (voir dans les Postrequis).
3. Comment fournir un algorithme pour calculer le chemin le plus probable (si possible) quand on considère non-plus la somme des poids, mais le produit des poids (dans $[0, 1]$, avec des probabilités). J'ai proposé de passer au log et d'utiliser Bellman-Ford, mais en fait il y a un moyen intelligent de le faire "à la Dijkstra"[4].
4. Justifiez les complexités annoncées pour cet algorithme. Comment gérer un tas binaire ? Un tas de Fibonacci ? Leurs complexités ? J'ai essentiellement dit que Fibonacci était dur.
5. Comment faire pour aller plus vite ? Parlez de A^*. Et les algorithmes qui utilisent la dualité départ-arrivée (en lançant deux recherches simultanées).
6. Expliquez en quoi votre autre développement exploite les CFC. J'ai donc fait une version abrégée du second développement.
7. Détaillez comment faire une 2-approximation du voyageur de commerce (dans le cas d'une distance), avec les arbres couvrants de poids minimal.
8. Connaissez-vous des classes de graphes sur lesquelles les problèmes NP-durs (que je n'avais pas mentionnés) deviennent plus faciles ? J'ai parlé du cas d'un poids qui est une distance, ou des graphes planaires, en précisant qu'en général les problèmes restaient aussi difficiles dans ce cas. Ils attendaient que je parle des graphes de *tree-* et *clique-width* bornée et des algorithmes FPT associés, mais je n'ai pas osé m'aventurer dedans.
9. Parlez d'Union-Find. Implémentation ? Complexité ? Comment l'utilise-t-on dans Kruskal ?

1. Appariteur = en charge de l'organisation et de la surveillance des épreuves. Ce n'est pas un membre du jury.
2. J. Goubault-Larrecq me dira par la suite "Il n'est rien de plus pénible que les gens qui se croient investis de la vérité, alors qu'ils ont tort."
3. Donc vous avez le droit de vous battre si cela vous arrive.
4. Connu sous le nom d'algorithme de Knuth.

4 Florilège de preuves fausses

> Aussi le bon Cherubini qui avait voulu déjà me faire avaler tant de couleuvres, dut se résigner à recevoir de ma main un boa constrictor qu'il ne digéra jamais.
>
> *Mémoires*, Hector Berlioz.

Avertissement. Les preuves de cette section sont (un peu) fausses, mais pas trop.

Un Banach à base dénombrable

On se place sur l'espace $\mathbb{T} := \mathbb{R}/2\pi\mathbb{Z}$ pour considérer $\mathrm{L}^2(\mathbb{T})$ l'espace des fonctions 2π-périodiques de carré intégrable sur une période. On sait (c'est classique autour des séries de Fourier) que la famille $(x \mapsto \mathrm{e}^{inx})_{n\in\mathbb{Z}}$ est une base de l'espace de Hilbert $\mathrm{L}^2(\mathbb{T})$.
On a donc un espace de Banach à base dénombrable. On se souviendra en parallèle du lemme suivant, qui se déduit de la théorie de Baire.

Lemme 4.1. *Un espace vectoriel réel à base dénombrable n'est pas complet.*

> *Preuve.* Soit E un espace de Banach de base dénombrable $(e_i)_{i\in\mathbb{N}}$. Alors $\forall n \geq 0$ $E_n :=$ Vect $((e_i)_{0\leq i\leq n})$ est un sous-espace de dimension finie, donc fermé et d'intérieur vide. Donc $E = \bigcup_{n\geq 0} E_n$ devrait être (Lemme de Baire) également d'intérieur vide, ce qui est manifestement faux. □

Formes quadratiques linéaires

Dans cette section E est un espace vectoriel sur un corps de caractéristique autre que 2. On se propose de démontrer le résultat suivant, bien utile dans la pratique, et qui permet de comprendre comment la théorie des formes quadratiques généralise les aspects linéaires.

Proposition 4.2. *Toute forme linéaire sur E est une forme quadratique.*

> *Preuve.* Soit $l : E \to \mathbb{R}$ une forme linéaire. Pour montrer qu'elle est quadratique, il est naturel de chercher la forme bilinéaire associée en utilisant une identité de polarisation. On pose donc $b(x, y) = \frac{l(x+y)-l(x)-l(y)}{2}$. Cette forme est symétrique et clairement bilinéaire. On conclut donc que l est une forme quadratique.
>
> □

Corps commutatifs

Dans cette section, on donne une preuve expéditive du célèbre théorème de Wedderburn.

Théorème 4.3 (Wedderburn). *Tout corps fini est commutatif.*

On commence par un lemme bien utile dans l'étude des corps finis, voir par exemple [Per98] pour une preuve détaillée.

Lemme 4.4. *Le groupe multiplicatif d'un corps fini est cyclique.*

Passons maintenant à la preuve du théorème. Considérons un corps fini $\mathbb{F}$, alors son sous-groupe $\mathbb{F}^*$ est cyclique en vertu du Lemme 4.4, donc commutatif. Ainsi $\forall x, y \in \mathbb{F}^*$ on a $xy = yx$. Puisque $\mathbb{F} = \mathbb{F}^* \uplus \{0\}$ et que 0 est absorbant, on conclut $\forall x, y \in \mathbb{F}$ que $xy = xy$.

Un nouveau résultat en complexité théorique

Abandonnons un peu les mathématiques pour parler de classes de complexité. On rappelle que P est la classe des problèmes résolubles par une machine de Turing déterministe en temps polynomial, et NL en espace logarithmique sur une machine non-déterministe. Nous montrons à présent le résultat suivant.

Théorème 4.5. P = NL.

Preuve. On sait déjà que NL ⊆ P, il reste à montrer l'inclusion inverse. On rappelle pour ce faire le problème suivant dit de *clôture d'une loi binaire* :
Entrée : Un magma $(G, \times)$ fini par sa table, une partie $H \subseteq G$ et un élément $g \in G$.
Question : Est-ce que $g \in \langle H \rangle$ le sous-magma engendré par H ?
Le problème de clôture est connu pour être P-complet (pour les réductions L). Ce n'est pas là qu'est l'erreur. Montrons que ce problème est en fait dans NL.
On donne l'algorithme suivant :

1. Choisir de manière non-déterministe un élément $h \in H$.
2. $c \leftarrow 0$
3. Tant que $c \leq |G|$:
   (a) si $h = g$ alors Accepter ;
   (b) $c \leftarrow c + 1$;
   (c) choisir de manière non-déterministe un élément $h' \in H$;
   (d) $h \leftarrow h \times h'$.
4. Rejeter.

Il est clair qu'il tourne en espace logarithmique et résout bien le problème (le compteur servant à faire terminer toutes les branches, il peut être retiré si on n'exige pas cela). Ceci conclut que P ⊆ NL.

□

5 Développements rédigés mais inusités

CA Confluence de la β-réduction.

Leçons possibles : 929

Elegance : ★☆☆☆☆

Avis.

Technique, technique, technique ! C'est un peu le lot des développements de cette leçon !

Prérequis.

Définition 5.1 (β-réduction). *La β-réduction est la plus petite relation sur les λ-termes vérifiant les propriétés suivantes :*
- *si $u \to_\beta u'$ alors $\lambda x.u \to_\beta \lambda x.u'$;*
- *si $u \to_\beta u'$ et alors $vu \to_\beta vu'$ et $u'v \to_\beta u'v$;*
- *$(\lambda x.u)v \to_\beta u[v/x]$.*

Remarque 5.2. *Les substitutions se font modulo α-renommage... pas de captures.*

Remarque 5.3. *Définir "plus petite relation" est équivalent à dire "en utilisant un nombre fini de fois les règles données" (arbre de preuve).*

Développement.

Théorème 5.4 (Church, Rosser). *La β-réduction est confluente.*

Définition 5.5 (Réduction parallèle). *On définit la réduction parallèle $\Rightarrow$ comme la plus petite relation sur les λ-termes vérifiant les propriétés suivantes :*
- *$x \Rightarrow x$ si x est une variable ;*
- *si $u \Rightarrow u'$ alors $\lambda x.u \Rightarrow \lambda x.u'$;*
- *si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $uv \Rightarrow u'v'$;*
- *si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $(\lambda x.u)v \Rightarrow u'[v'/x]$.*

Intuitivement cette réduction autorise à contracter en une seule étape plusieurs redex à l'intérieur du terme. Plus précisément on a le lemme suivant.

Lemme 5.6. $\to_\beta \subseteq \Rightarrow \subseteq \to_\beta^*$

Preuve. 1. $\to_\beta \subseteq \Rightarrow$ est clair par restriction des règles.

2. On commence par montrer que la clôture $\to_\beta^*$ passe au contexte.

Lemme 5.7. *Si $u \to_\beta^* u'$ alors $\lambda x.u \to_\beta^* \lambda x.u'$ et $vu \to_\beta^* vu'$ et $uv \to_\beta^* u'v$.*

Preuve. On montre par induction sur n que si $u \to_\beta^n u'$ alors ... (simple). □

Maintenant montrons par induction sur la structure de t que si $t \Rightarrow t'$ alors $t \to_\beta^* t'$
- si $t = x$ c'est vrai car $t' = x$;
- si $t = \lambda x.u$ alors $t' = \lambda x.u'$ où $u \Rightarrow u'$. Par hypothèse d'induction il vient $u \to_\beta^* u'$ et on conclut que $\lambda x.u \to_\beta^* \lambda x.u'$ par le Lemme 5.7 ;
- si $t = uv$ et que la dernière règle appliquée était $uv \Rightarrow u'v'$, alors l'induction fournit $u \to_\beta^* u'$ et $v \to_\beta^* v'$. Par le Lemme 5.7 il vient $uv \to_\beta^* u'v \to_\beta^* u'v'$;

— si $t = (\lambda x.u)v$ et que la règle était $t \Rightarrow u'[v'/x]$, alors $u \to_\beta^* u'$ et $v \to_\beta^* v'$ par hypothèse d'induction. Donc $(\lambda x.u)v \to_\beta^* (\lambda x.u')v \to_\beta^* (\lambda x.u')v' \to_\beta u'[v'/x]$ modulo utilisation du Lemme 5.7 pour les deux premières étapes.

□

On en déduit que $\Rightarrow^* = \to_\beta^*$. Il suffit donc de montrer la confluence de $\Rightarrow$ pour obtenir ce que l'on veut. En fait on va prouver que $\Rightarrow$ est fortement confluente.

Lemme 5.8 (Substitution). *Si $t \Rightarrow t'$ et $s \Rightarrow s'$ alors $t[s/x] \Rightarrow t'[s'/x]$*

Preuve.

On montre le résultat par induction sur la structure de t.

— si t est une variable c'est clair ;
— si $t = \lambda y.u$, alors $t' = \lambda y.u'$ avec $u \Rightarrow u'$. Par induction on a $u[s/x] \Rightarrow u'[s'/x]$ et donc $\lambda y.(u[s/x]) \Rightarrow \lambda y.(u'[s'/x])$. Comme y est liée on peut supposer que $y \neq x$ (α-équivalence) d'où $\lambda y.(u[s/x]) = (\lambda y.u)[s/x]$ et $\lambda y.(u[s'/x]) = (\lambda y.u')[s'/x]$;
— si $t = uv$ et la dernière règle était $uv \Rightarrow u'v'$, alors $u[s/x] \Rightarrow u'[s'/x]$ et $v[s/x] \Rightarrow v'[s'/x]$ par induction. D'où $(u[s/x])(v[s/x]) \Rightarrow (u'[s'/x])(v'[s'/x])$ ce qui conclut en factorisant les substitutions ;
— si $t = (\lambda y.u)v$ et la règle était $t \Rightarrow u'[v'/y]$ alors $u[s/x] \Rightarrow u'[s'/x]$ et $v[s/x] \Rightarrow v'[s'/x]$ par induction. Donc $(\lambda y.(u[s/x]))v[s/x] \Rightarrow (u'[s'/x][v'[s'/x]/y])$.
D'une part on peut supposer que $y \neq x$ dans $(\lambda y.(u[s/x]))v[s/x]$ (par α-équivalence car y liée), et alors $(\lambda y.(u[s/x]))v[s/x] = ((\lambda y.u)v)[s/x]$ en factorisant.
Pour l'autre terme, on a $(u'[s'/x][v'[s'/x]/y]) = (u'[v'/y])[s'/x]$ dès que y n'est pas libre dans s'. C'est effectivement le cas car y n'est pas libre dans s (pour substituer $t[s/x]$ sans capture) et $s \Rightarrow s'$ préserve cette propriété.

□

Il ne reste plus qu'à conclure la preuve du théorème. On va montrer par induction sur t que si $t \Rightarrow t'$ et $t \Rightarrow t''$ alors il existe s tel que $t' \Rightarrow s$ et $t'' \Rightarrow s$.

— si t est une variable c'est immédiat ;
— si $t = \lambda x u$ alors $t' = \lambda x.u'$ et $t'' = \lambda x.u''$. Par induction il existe s_u tel que $u' \Rightarrow s_u$ et $u'' \Rightarrow s_u$, et alors $\lambda x.t' \Rightarrow \lambda x.s_u$ et $\lambda x.t'' \Rightarrow \lambda x.s$;
— si $t = uv$ et les règles donnent $t' = u'v'$ et $t'' = u''v''$ et par induction il existe s_u et s_v tel que $u' \Rightarrow s_u$ et $u'' \Rightarrow s_u$ et $v' \Rightarrow s_v$ et $v'' \Rightarrow s_v$. D'où $t' \Rightarrow s_u s_v$ et $t'' \Rightarrow s_u s_v$;
— si $t = (\lambda x.u)v$ il reste deux cas non traités (le troisième est symétrique) :
  — $t' = u'[v'/x]$ et $t'' = u''[v''/x]$. Par induction il existe s_u et s_v tels que $u' \Rightarrow s_u$ et $u'' \Rightarrow s_u$ et $v' \Rightarrow s_v$ et $v'' \Rightarrow s_v$. Alors en vertu du Lemme 5.8, $u'[v'/x] \Rightarrow s_u[s_v/x]$ et $u''[v''/x] \Rightarrow s_u[s_v/x]$, ce qui conclut.
  — $t' = u'[v'/x]$ et $t'' = (\lambda x.u'')v''$. Par induction il existe s_u et s_v tels que $u' \Rightarrow s_u$ et $u'' \Rightarrow s_u$ et $v' \Rightarrow s_v$ et $v'' \Rightarrow s_v$. Alors en vertu du Lemme 5.8 $u'[v'/x] \Rightarrow s_u[s_v/x]$. D'autre part $(\lambda x.u'')v'' \Rightarrow s_u[s_v/x]$, ce qui conclut.

Postrequis.

1. Il serait bon à l'oral d'admettre le Lemme 5.6 (qui n'est pas trivial quand même), ainsi que le Lemme 5.8 peut-être en partie.
2. Liens avec la standardisation et le codage des entiers de Church.
3. Et si on essayait de montrer que $\to_\beta$ est fortement confluente ? C'est faux ! En effet soit $t := ((\lambda x.(xx))((\lambda y.y)z)$ alors $t \to_\beta ((\lambda x.(xx))z)$ et $t \to_\beta ((\lambda y.y)z)((\lambda y.y)z)$ mais on ne peut pas les faire confluer en une seule étape.
4. Et si on essayait de montrer que $\to_\beta$ est localement confluente et termine (pour appliquer Newman) ? C'est faux elle ne termine pas ! Considérer $\Omega := (\lambda x.(xx))(\lambda x.(xx))$.
5. La $\beta\eta$-réduction est également confluente.
6. Soit $\to$ une relation de réécriture, et $\equiv$ la plus petite relation d'équivalence qui la contient. On dit que $\to$ vérifie la propriété de Church-Rosser lorsque $t \equiv t'$ implique l'existence de s tel que $t \to s$ et $t' \to s$. Cette propriété est équivalente à la confluence.

CB Théorème de Stone-Weierstrass.

Leçons possibles : 203
Pas de référence.

Elegance : ★★★★★

Avis.

Une preuve charmante. Son nom ? Je me souviens qu'il est doux et sonore.

Prérequis.

1. Approximation de la valeur absolue.

   Lemme 5.9. *Dans $\mathcal{C}([-1,1],\mathbb{R})$, il existe une suite de polynômes convergeant uniformément vers la valeur absolue.*

   > *Idée de preuve.* On peut soit utiliser le théorème d'approximation de Weierstrass (Stone-Weierstrass dans le cas $\mathcal{C}([a,b],\mathbb{R})$, démontré par exemple avec les polynômes de Bernstein), soit expliciter une suite convergeant ponctuellement vers la valeur absolue et passer à l'uniformité par un théorème de Dini.

   □

Développement.

Dans la suite X est un compact contenant au moins 2 éléments.

Lemme 5.10. *Soit H une partie de $\mathcal{C}(X,\mathbb{R})$ telle que :*
— $\forall x \neq y$, $\forall a,b \in \mathbb{R}$, $\exists h \in H$, $h(x) = a$ *et* $h(y) = b$;
— *H est stable par* $\sup$ *et* $\inf$ *(finis)*[1].
Alors H est dense dans $\mathcal{C}(X,\mathbb{R})$.

> *Preuve.* Soit $f \in \mathcal{C}(X,\mathbb{R})$ et $\varepsilon > 0$.
>
> 1. Soit $x \in X$ fixé. $\forall y \neq x$, $\exists h_y \in H$ telle que $h_y(x) = f(x)$ et $h_y(y) = f(y)$.
>   On note $\mathcal{O}_y = \{z \mid h_y(z) > f(z) - \varepsilon\}$ c'est un ouvert contenant x et y.
>   Donc $X = \bigcup_{y \in X \setminus \{x\}} \mathcal{O}_y$.
>   On peut extraire un sous-recouvrement fini $X = \bigcup_{y_1,\dots,y_n} \mathcal{O}_{y_i}$.
>   Soit $g_x = \sup h_{y_i}$, alors $g_x(x) = f(x)$ et $\forall z \in X$, $g_x(z) > f(z) - \varepsilon$.
> 2. On note $\mathcal{O}_y = \{z \mid g_x(z) < f(z) + \varepsilon\}$ c'est un ouvert contenant x.
>   Donc $X = \bigcup_{x \in X} \mathcal{O}_x$ et on extrait $X = \bigcup_{x_1,\dots,x_m} \mathcal{O}_{x_i}$.
>   Si $g = \inf g_i$, alors $\forall z, f(z) - \varepsilon < g(z) < f(z) + \varepsilon$.
>
> Donc H est dense dans $\mathcal{C}(X,\mathbb{R})$.

□

Théorème 5.11 (Stone-Weierstrass). *Soit H une sous-algèbre de $\mathcal{C}(X,\mathbb{R})$ séparante et contenant les constantes. Alors H est dense dans $\mathcal{C}(X,\mathbb{R})$.*

> *Preuve.* On montre que $\overline{H}$ vérifie les hypothèses de 5.10.
>
> 1. Soient $x,y \in X$ et h tel que $h(x) \neq h(y)$. Alors soit $g : t \mapsto a + \frac{h(x)-h(t)}{h(x)-(y)}(b-a)$. Alors $g \in H$[2] et $g(x) = a$ et $g(y) = b$.

1. On dit que cette partie est *réticulée*.
2. Puisque c'est une algèbre contenant les constantes

2. Sur $[-1,1]$, il existe par 5.9 une suite de polynômes $(P_n)_{n\geq 0}$ convergeant uniformément vers $x \mapsto |x|$.

   Si $f \in \overline{H} \setminus \{0\}$, alors $P_n\left(\frac{f}{||f||_\infty}\right) \to \frac{|f|}{||f||_\infty}$ uniformément.

   Donc $||f||_\infty P_n\left(\frac{f}{||f||_\infty}\right) \to |f|$ uniformément. Donc $|f| \in \overline{H}$ [1].

   Donc H est stable par sup et inf [2].

Donc $\overline{H}$ est dense dans $\mathcal{C}(X,\mathbb{R})$ [3], donc H aussi. □

1. $||f||_\infty P_n\left(\frac{f}{||f||_\infty}\right) \in H$ par algèbre.
2. Puisque $\sup(f,g) = \frac{f+g+|f-g|}{2}$ et $\inf = \frac{f+g+|f-g|}{2}$.
3. Il y a même égalité $\overline{H} = \mathcal{C}(X,\mathbb{R})$.

BIBLIOGRAPHIE

[AB09] Sanjeev Arora and Boaz Barak. *Computational complexity : a modern approach*. Cambridge University Press, 2009.

[AKS83] Miklós Ajtai, János Komlós, and Endre Szemerédi. Sorting in $c\log(n)$ parallel steps. *Combinatorica*, 3(1) :1–19, 1983.

[APT79] Bengt Aspvall, Michael F Plass, and Robert Endre Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3) :121–123, 1979.

[BMP05] Vincent Beck, Jérôme Malick, and Gabriel Peyré. Objectif agrégation. 2005.

[Car08] Olivier Carton. *Langages formels, calculabilité et complexité*. Vuibert, 2008.

[CG13] Philippe Caldero and Jérôme Germoni. *Histoires hédonistes de groupes et de géométries*. Calvage & Mounet, 2013.

[Cia90] Philippe G. Ciarlet. Introduction à l'analyse numérique matricielle et à l'optimisation. 1990.

[CL03] René Cori and Daniel Lascar. Logique mathématique, tome 1 : Calcul propositionnel, algèbre de boole, calcul des prédicats, coll. *Sciences Sup, Dunod*, 2003.

[CLRS02] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction à l'algorithmique : Cours et exercices (seconde édition)*. Dunod, 2002.

[CLRS10] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction à l'algorithmique : Cours et exercices (troisième édition)*. Dunod, 2010.

[CR94] Maxime Crochemore and Wojciech Rytter. *Text algorithms*. Maxime Crochemore, 1994.

[Dem08] Michel Demazure. Cours d'algèbre : primalité, divisibilité, codes. 2008.

[Dow08] Gilles Dowek. *Les démonstrations et les algorithmes*. Éditions de l'École polytechnique, 2008.

[DT18] Gaëtan Douéneau-Tabot. On the complexity of infinite advice strings. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, 2018.

[FGN01a] Serge Francinou, Hervé Gianella, and Serge Nicolas. Exercices de mathématiques : Oraux x-ens algèbre 1. *Cassini, Paris*, 2001.

[FGN01b] Serge Francinou, Hervé Gianella, and Serge Nicolas. Exercices de mathématiques : Oraux x-ens analyse 1. *Cassini, Paris*, 2001.

[Gou09a] Xavier Gourdon. Algèbre : mathématiques pour MP*. *Les Maths en tête. Ellipses, Paris*, 2009.

[Gou09b] Xavier Gourdon. Analyse : mathématiques pour MP*. *Les Maths en tête. Ellipses, Paris*, 2009.

[Hoo] Hendrik Jan Hoogeboom. Undecidable problems for context-free grammars.

[Opp78] Derek C Oppen. A $2^{2^{2^{pn}}}$ upper bound on the complexity of presburger arithmetic. *Journal of Computer and System Sciences*, 16(3) :323–332, 1978.

[Ouv07] Jean-Yves Ouvrard. Probabilités. tome i, licence-capes. 2007.

[Per98] Daniel Perrin. Cours d'algèbre maths agreg, 1998.

[QZ02] Hervé Queffélec and Claude Zuily. Eléments d'analyse : agrégation de mathématiques. 2002.

[QZ13] Hervé Queffélec and Claude Zuily. Analyse pour l'agrégation : cours et exercices corrigés. 2013.

[Rom17] Jean-Etienne Rombaldi. *Mathématiques pour l'Agrégation : Algèbre & géométrie*. De Boeck Superieur, 2017.

[Rou14] François Rouvière. Petit guide de calcul différentiel : à l'usage de la licence et de l'agrégation (quatrième édition). 2014.

[Rud09] Walter Rudin. Analyse réelle et complexe, 2009.

[Win93] Glynn Winskel. *The formal semantics of programming languages : an introduction*. MIT press, 1993.