Annales des Concours

MP
Mathématiques · Informatique
2017

Sous la coordination de

Guillaume Batog
professeur en CPGE
ancien élève de l'École Normale Supérieure (Cachan)

Julien Dumont
professeur en CPGE
ancien élève de l'École Normale Supérieure (Cachan)

Vincent Puyhaubert
professeur en CPGE
ancien élève de l'École Normale Supérieure (Cachan)

Par

Virgile Andreani
ENS Ulm

William Aufort
ENS Lyon

Guillaume Batog
professeur en CPGE

Josselin Giet
ENS Ulm

Martin Guy
ENS Lyon

Alban Levy
ENS Cachan

Florian Metzger
docteur en mathématiques

Robin Michaud
ENS Lyon

Matthias Moreno Ray
professeur en CPGE

Rémi Pellerin
ENS Lyon

Sophie Rainero
professeur en CPGE

Cyril Ravat
professeur en CPGE

Yvon Vignaud
professeur en CPGE

Sommaire

Mines-Ponts

Polytechnique-ENS

Formulaires

Sommaire thématique de mathématiques

2015 – 2017

	Structures algébriques et arithmétique	Polynômes	Algèbre linéaire générale	Réduction des endomorphismes	Produit scalaire et espaces euclidiens	Topologie des espaces vectoriels normés	Suites et séries numériques	Suites et séries de fonctions	Séries entières	Analyse réelle	Intégration	Équations différentielles	Fonctions de plusieurs variables	Dénombrement et probabilités
e3a PSI Maths A		•	••	••		•	•				••			•••
e3a PSI Maths B		•	••	•	•	•			•		•			
CCP MP Maths 1							•	•••	•		••		•	••
CCP MP Maths 2	•	•	•	•••	••									
CCP PC Maths		•	••	•		•		•	•		••			•••
CCP PSI Maths			•	••	•	•	•		•		•	••		
Centrale MP Maths 1			•	••							•	•		
Centrale MP Maths 2							•	•	•		••			••
Centrale PC Maths 1		•	••	•					••		•	•		••
Centrale PC Maths 2	•	•	•				•	•		•	•			•
Centrale PSI Maths 1			•	•		•	•			•				•••
Centrale PSI Maths 2		•	••							•	••	•	•	•
Mines MP Maths 1			•	•	•			•				•		••
Mines MP Maths 2					•	••	•	•			•			•
Mines PC Maths 1			•	•			••		•	•				•••
Mines PC Maths 2		•	•				•		•	•	•	•		•
Mines PSI Maths 1			•	•			••		•	•				•••
Mines PSI Maths 2			•••	•										
X/ENS MP Maths A		••	••	••		••								
X MP Maths B								••		•••	••		•	•
X/ENS PC Maths			•	••		•				•			•	•

SESSION 2017 **MPMA102**

CONCOURS COMMUNS POLYTECHNIQUES

EPREUVE SPECIFIQUE - FILIERE MP

MATHEMATIQUES 1

Mardi 2 mai : 14 h - 18 h

N.B. : le candidat attachera la plus grande importance à la clarté, à la précision et à la concision de la rédaction. Si un candidat est amené à repérer ce qui peut lui sembler être une erreur d'énoncé, il le signalera sur sa copie et devra poursuivre sa composition en expliquant les raisons des initiatives qu'il a été amené à prendre.

Les calculatrices sont interdites

Le sujet est composé de deux exercices et d'un problème tous indépendants.

EXERCICE 1

On définit deux fonctions :
- la fonction f de $\mathbb{R}^2$ dans $\mathbb{R}$ par $f(x, y) = \sin(x^2 - y^2)$,
- la fonction g de $\mathbb{R}^2$ dans $\mathbb{R}^2$ par $g(x, y) = (x + y, x - y)$.

Q1. Justifier que les fonctions f et g sont différentiables en tout vecteur $(x, y) \in \mathbb{R}^2$ et écrire la matrice jacobienne de f puis de g en (x, y).

Q2. Pour $(x, y) \in \mathbb{R}^2$, déterminer l'image d'un vecteur $(u, v) \in \mathbb{R}^2$ par l'application linéaire $d(f \circ g)((x, y))$ en utilisant les deux méthodes suivantes :

1. en calculant $f \circ g$;
2. en utilisant le produit de deux matrices jacobiennes.

EXERCICE 2

On admet que $\displaystyle\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$ et on pose $A = \mathbb{N}^* \times \mathbb{N}^*$.

Q3. Démontrer que la famille $\displaystyle\left(\frac{1}{p^2\, q^2}\right)_{(p,q)\in A}$ est sommable et calculer sa somme.

Q4. Démontrer que la famille $\displaystyle\left(\frac{1}{p^2 + q^2}\right)_{(p,q)\in A}$ n'est pas sommable.

PROBLÈME
Séries trigonométriques

Il est utile en physique, notamment pour étudier des spectres d'énergie ou pour décomposer un signal périodique en harmoniques, de pouvoir écrire une fonction périodique en somme d'une série de fonctions trigonométriques.

Nous allons nous intéresser à l'aspect mathématique de cette décomposition pour les fonctions de période 2π.

Dans ce qui suit, on appelle "série trigonométrique" une série de fonctions du type :

$$\sum [a_n \cos(nx) + b_n \sin(nx)] \text{ où } (a_n) \text{ et } (b_n) \text{ sont deux suites de réels.}$$

Dans la première partie, on étudie quelques exemples. Dans la deuxième partie, on s'intéresse plus particulièrement aux séries trigonométriques qui convergent normalement sur $\mathbb{R}$.

On notera $C_{2\pi}$ l'espace vectoriel des fonctions continues et 2π-périodiques de $\mathbb{R}$ dans $\mathbb{R}$.

Pour une fonction f élément de $C_{2\pi}$, on notera, pour tout entier naturel n :

$$\alpha_n(f) = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(nx)\,\mathrm{d}x \qquad \text{et} \qquad \beta_n(f) = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(nx)\,\mathrm{d}x\ .$$

Partie I - Exemples

Q5. Démontrer que la série trigonométrique $\sum\left[\frac{1}{2^n}\cos(nx)+\frac{1}{3^n}\sin(nx)\right]$ converge normalement sur $\mathbb{R}$. Pour tout entier $p \geq 2$, déterminer la somme de la série $\sum_{n\geq 0}\left(\frac{e^{ix}}{p}\right)^n$ puis en déduire la valeur de $\sum_{n=0}^{+\infty}\left[\frac{1}{2^n}\cos(nx)+\frac{1}{3^n}\sin(nx)\right]$ (il n'est pas utile de réduire au même dénominateur).

Q6. Écrire la fonction $\varphi : x \mapsto \exp(\cos x)\ \cos(\sin x)$ comme la somme d'une série trigonométrique. On pourra écrire la fonction $x \mapsto \exp(e^{ix})$ comme la somme d'une série de fonctions.

Q7. Donner un exemple de suite (a_n) de limite nulle, telle que la série trigonométrique $\sum a_n\ \cos(nx)$ ne converge pas simplement sur $\mathbb{R}$.

Q8. On admet que la série trigonométrique $\sum_{n\geq 1}\frac{1}{\sqrt{n}}\sin(nx)$ converge simplement sur $\mathbb{R}$. Converge-t-elle normalement sur $\mathbb{R}$?

Partie II - Propriétés

Une condition suffisante

Q9. Démontrer que si les séries $\sum a_n$ et $\sum b_n$ sont absolument convergentes, alors la série trigonométrique $\sum [a_n\cos(nx)+b_n\sin(nx)]$ converge normalement sur $\mathbb{R}$.

Une condition nécessaire

Q10. Soient a et b deux réels quelconques.

Démontrer que le maximun sur $\mathbb{R}$ de la fonction $x \mapsto |a\cos x + b\sin x|$ est $\sqrt{a^2+b^2}$.

Q11. Démontrer que si la série trigonométrique $\sum [a_n\cos(nx)+b_n\sin(nx)]$ converge normalement sur $\mathbb{R}$, alors les suites (a_n) et (b_n) convergent vers 0 et les séries $\sum a_n$ et $\sum b_n$ sont absolument convergentes.

Autres propriétés

Q12. On note f la somme d'une série trigonométrique $\sum [a_n\cos(nx)+b_n\sin(nx)]$ qui converge normalement sur $\mathbb{R}$. Justifier que $f \in C_{2\pi}$.

Q13. Calculer $\int_{-\pi}^{\pi}\cos^2(nx)\mathrm{d}x$ pour $n \neq 0$ et donner la valeur de $\int_{-\pi}^{\pi}\sin(kx)\cos(nx)\mathrm{d}x$ pour $k \neq n$.

Q14. On note f la somme d'une série trigonométrique $\sum [a_n\cos(nx)+b_n\sin(nx)]$ qui converge normalement sur $\mathbb{R}$: pour tout réel x, $f(x)=\sum_{k=0}^{+\infty}[a_k\cos(kx)+b_k\sin(kx)]$. Démontrer que pour tout

CCP Maths 1 MP 2017 — Corrigé

Ce corrigé est proposé par Robin Michaud (ENS Lyon) ; il a été relu par Florian Metzger (docteur en mathématiques) et Sophie Rainero (professeur en CPGE).

L'épreuve se compose de deux exercices et d'un problème, tous trois indépendants. Le premier exercice est calculatoire : il comporte deux questions de calcul différentiel très proches du cours. Dans le deuxième exercice, on étudie la sommabilité des familles

$$\left(\frac{1}{p^2q^2}\right)_{(p,q)\in(\mathbb{N}^*)^2} \qquad \text{et} \qquad \left(\frac{1}{p^2+q^2}\right)_{(p,q)\in(\mathbb{N}^*)^2}$$

Dans le problème, qui constitue l'essentiel du sujet, on s'intéresse aux séries dites trigonométriques, c'est-à-dire les séries de la forme

$$\sum \left(a_n \cos(nx) + b_n \sin(nx)\right)$$

Un des buts du problème est d'établir le lien entre une fonction f qui s'exprime comme une somme trigonométrique et ses coefficients de Fourier, à savoir

$$\alpha_n(f) = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(nx)\,\mathrm{d}x \quad \text{et} \quad \beta_n(f) = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(nx)\,\mathrm{d}x$$

- La première partie du problème est consacrée à quelques exemples de séries trigonométriques : en les étudiant, on met en évidence les différences entre convergence simple et convergence normale.
- Dans une seconde partie, on prouve que la convergence absolue des séries $\sum a_n$ et $\sum b_n$ est une condition nécessaire et suffisante à la convergence normale de la série $\sum (a_n \cos(nx) + b_n \sin(nx))$. Puis on montre qu'une fonction f continue et 2π-périodique vérifie pour tout $x \in \mathbb{R}$

  $$f(x) = \frac{\alpha_0(f)}{2} + \sum_{n=1}^{+\infty} \alpha_n(f)\cos(nx) + \beta_n(f)\sin(nx)$$

  lorsque cette série converge normalement. Cette série est appelée *développement en série de Fourier* de la fonction f. On utilise ces résultats pour calculer les sommes

  $$\sum_{n=1}^{+\infty} \frac{1}{n^2} \qquad \text{et} \qquad \sum_{n=1}^{+\infty} \frac{(-1)^n}{n^2}$$

  ainsi que l'intégrale $$\int_0^1 \frac{\ln(1+x)}{x}\,\mathrm{d}x$$

  Enfin, on trouve un critère de dérivabilité pour les séries trigonométriques et on l'applique au calcul d'une somme.

Ce sujet est abordable et comporte un grand nombre de questions proches du cours. De la rigueur est néanmoins nécessaire pour ne pas passer à côté des difficultés, notamment dans la justification des interversions de limites et d'intégrales. Cela constitue une bonne révision du cours sur les séries de fonctions et un aperçu intéressant sur les séries de Fourier, qui ne figurent plus au programme de MP.

Indications

Exercice 1

1 La fonction f est une composée de fonctions différentiables et g est linéaire.

2.1 Calculer les dérivées partielles de $f \circ g$.

2.2 Utiliser la formule donnant la matrice jacobienne de $f \circ g$.

Exercice 2

3 Après avoir justifié la sommabilité, calculer la somme demandée en utilisant la formule pour la somme des séries doubles positives sommables

$$\sum_{(p,q)\in(\mathbb{N}^*)^2} u_{p,q} = \sum_{p=1}^{+\infty}\Big(\sum_{q=1}^{+\infty} u_{p,q}\Big)$$

4 Écrire $\mathrm{A} = \bigcup\limits_{n=2}^{+\infty} \mathrm{I}_n$ où $\mathrm{I}_n = \{\,(p,q)\in \mathrm{A}\,|\, p+q=n\}$ et sommer par paquets.

Problème

5 Penser aux formules d'Euler qui établissent un lien entre les fonctions trigonométriques et la fonction exponentielle.

6 Remarquer que pour tout $x \in \mathbb{R}$

$$\exp(\mathrm{e}^{\,\mathrm{i}x}) = \sum_{n=0}^{+\infty} \frac{\mathrm{e}^{\,\mathrm{i}nx}}{n!} \qquad \text{et} \qquad \exp(\cos x)\cos(\sin x) = \mathrm{Re}\,\left(\exp(\mathrm{e}^{\,\mathrm{i}x})\right)$$

7 Regarder ce qui se passe en $x = 0$.

8 Utiliser le fait que le maximum de la fonction sinus sur $\mathbb{R}$ est 1.

9 Majorer le terme $|a_n \cos(nx) + b_n \sin(nx)|$ à l'aide de $|a_n|$ et $|b_n|$.

10 À l'aide d'une formule trigonométrique, chercher le maximum sur $\mathbb{R}$ de la fonction

$$x \longmapsto \frac{a}{\sqrt{a^2+b^2}}\cos x + \frac{b}{\sqrt{a^2+b^2}}\sin x$$

11 Utiliser la question 10 ainsi que la définition de la convergence normale pour obtenir une majoration de $|a_n|$ et $|b_n|$ par des suites qui sont les termes généraux de séries convergentes.

12 Appliquer le théorème de continuité d'une limite uniforme de fonctions continues.

13 La première intégrale se calcule à l'aide de l'égalité

$$\forall x \in \mathbb{R} \qquad \cos^2 x = \frac{1+\cos(2x)}{2}$$

et la seconde en remarquant que l'intégrande est une fonction impaire.

14 Utiliser un théorème d'interversion d'une série et d'une intégrale, puis appliquer la question précédente et l'égalité donnée par l'énoncé à chacune des intégrales.

15 Appliquer la question 14 à la série de fonctions $\sum u_n(x)$.

16 On pourra montrer que $\alpha_n(f-g) = \alpha_n(f) - \alpha_n(g)$ (idem avec β_n).

17 Étudier la parité de la fonction $x \longmapsto f(x)\sin x$.

18 Deux intégrations par parties successives suffisent à calculer α_n, tandis que la question 17 donne la valeur de β_n.

19 Évaluer la fonction f définie par l'énoncé à la question 18 d'une part, et son développement en série trigonométrique calculée à la même question d'autre part, en des points bien choisis.

20 Montrer que la fonction $x \longmapsto \ln(1+x)/x$ est prolongeable par continuité en 0. Pour le calcul de l'intégrale, utiliser le développement en série entière du logarithme et appliquer un théorème d'intégration terme à terme.

21 La question 18 fournit un contre-exemple. Pour trouver une condition nécessaire et suffisante pour que la somme de la série trigonométrique

$$\sum \left(a_n \cos(nx) + b_n \sin(nx)\right)$$

soit une fonction dérivable, penser au théorème du cours qui permet de dériver une somme de fonctions dérivables.

22 Utiliser la série trigonométrique $\sum \sin(nx)/3^n$ et la dérivée de sa somme.

Exercice 1

1 La fonction f est la composée de la fonction $(x,y) \longmapsto x^2 - y^2$ qui est différentiable sur $\mathbb{R}^2$ car polynomiale, avec la fonction $\theta \longmapsto \sin\theta$ qui est différentiable sur $\mathbb{R}$. Ainsi, f est différentiable en tout point de $\mathbb{R}^2$ comme composée de fonctions différentiables. Soit $(x,y) \in \mathbb{R}^2$. On note $\text{Jac}(f)(x,y)$ la matrice jacobienne de f en ce point. Comme $f : \mathbb{R}^2 \to \mathbb{R}$, cette matrice est de taille $(1,2)$. Les dérivées partielles de f en (x,y) valent

$$\frac{\partial f}{\partial x}(x,y) = 2x\cos(x^2-y^2) \quad \text{et} \quad \frac{\partial f}{\partial y}(x,y) = -2y\cos(x^2-y^2)$$

si bien que la matrice jacobienne de f est

$$\boxed{\forall (x,y) \in \mathbb{R}^2 \qquad \text{Jac}(f)(x,y) = \begin{pmatrix} 2x\cos(x^2-y^2) & -2y\cos(x^2-y^2) \end{pmatrix}}$$

La fonction g est une application linéaire de $\mathbb{R}^2$ dans $\mathbb{R}^2$, elle est donc différentiable en tout point de $\mathbb{R}^2$ et sa matrice jacobienne (de taille $(2,2)$) est la matrice de g dans la base canonique :

$$\boxed{\forall (x,y) \in \mathbb{R}^2 \qquad \text{Jac}(g)(x,y) = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}}$$

> On rappelle que la matrice jacobienne d'une application $h : \mathbb{R}^n \to \mathbb{R}^p$ est une matrice de $\mathcal{M}_{p,n}(\mathbb{R})$. Si $h = (h_1, \dots, h_p)$ avec $h_i : \mathbb{R}^n \to \mathbb{R}$ pour tout $i \in [\![\, 1\,;p\,]\!]$, alors la matrice jacobienne de h est
>
> $$\text{Jac}(h) = \begin{pmatrix} \dfrac{\partial h_1}{\partial x_1} & \cdots & \dfrac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial h_p}{\partial x_1} & \cdots & \dfrac{\partial h_p}{\partial x_n} \end{pmatrix}$$

2.1 Soit $(x,y) \in \mathbb{R}^2$. D'après le cours l'image par l'application $d(f\circ g)(x,y)$ d'un point $(u,v) \in \mathbb{R}^2$ est donnée par

$$\Big(d(f\circ g)(x,y)\Big)(u,v) = \frac{\partial (f\circ g)}{\partial x}(x,y)\cdot u + \frac{\partial (f\circ g)}{\partial y}(x,y)\cdot v$$

On calcule en premier lieu

$$f\circ g(x,y) = \sin((x+y)^2 - (x-y)^2) = \sin(4xy)$$

car $(x+y)^2 - (x-y)^2 = 2x \cdot 2y = 4xy$. Puis, on en déduit les dérivées partielles de cette fonction

$$\frac{\partial (f\circ g)}{\partial x}(x,y) = 4y\cos(4xy) \quad \text{et} \quad \frac{\partial (f\circ g)}{\partial y}(x,y) = 4x\cos(4xy)$$

Par conséquent,

$$\boxed{\begin{array}{l} \text{Pour tout } (x,y) \in \mathbb{R}^2 \text{ et tout } (u,v) \in \mathbb{R}^2 \text{ on a} \\ \Big(d(f\circ g)(x,y)\Big)(u,v) = 4y\cos(4xy)u + 4x\cos(4xy)v \end{array}}$$

CCP Maths 2 MP 2017 — Corrigé

Ce corrigé est proposé par Guillaume Batog (professeur en CPGE) ; il a été relu par Quentin Guilmant (ENS Lyon) et Benjamin Monmege (enseignant-chercheur à l'université).

Le sujet est constitué d'un seul problème qui traite de matrices stochastiques, c'est-à-dire de matrices carrées à coefficients réels positifs dont la somme vaut 1 à chaque ligne. Ces matrices interviennent en probabilités pour représenter des chaînes de Markov, qui sont des processus à temps discret dans lesquels le passage d'un état à un autre est régi par des probabilités indépendantes des états précédents. Ces objets sont étudiés en spécialité maths en TS et les résultats qui y sont admis trouveront un éclairage fort intéressant dans ce problème.

- La première partie, indépendante du reste, donne un exemple de chaîne de Markov à deux états modélisée par une matrice stochastique A de taille 2. On diagonalise A pour calculer la limite de la suite de matrices $(\mathrm{A}^n)_{n\in\mathbb{N}}$, ce qui fournit la probabilité limite de se trouver dans l'un ou l'autre des deux états. C'est le seul endroit du problème où les probabilités apparaissent.
- La deuxième partie s'intéresse au spectre complexe d'une matrice stochastique. Elle nécessite de bien savoir manipuler vecteurs propres, normes et inégalités. Le seul résultat admis du problème concerne la dimension du sous-espace propre pour la valeur propre 1.
- La troisième partie établit la convergence de la suite de matrices $(\mathrm{A}^n)_{n\in\mathbb{N}}$ pour toute matrice stochastique A strictement positive (aucun coefficient n'est nul). Elle fait intervenir des matrices nilpotentes et des matrices définies par blocs.
- La quatrième partie décrit un processus limite pour obtenir une probabilité invariante par une matrice stochastique A strictement positive, c'est-à-dire un vecteur ligne μ représentant une distribution de probabilités et vérifiant $\mu\mathrm{A} = \mu$. Ce vecteur est unique. Un brin de topologie apparaît au cours des questions.
- La cinquième partie met en œuvre l'étude théorique précédente pour calculer avec Python une valeur approchée de la probabilité invariante d'une matrice stochastique strictement positive. Les matrices sont représentées par des listes de listes. Les questions de programmation se ramènent à utiliser les algorithmes de base du programme d'informatique commune de première année.

Ce problème est excellemment bien calibré pour le concours CCP MP, les questions sont bien ciblées, ce qui permet à n'importe quel étudiant sérieux d'avancer à n'importe quel endroit du sujet. Aucune question n'est insurmontable. Toutefois, il faut faire attention à ne pas survoler les questions, se contenter d'affirmer des propriétés, ou mener des calculs sans fournir les arguments attendus, au risque de perdre énormément de points !

Indications

Partie I

1 Employer la formule des probabilités totales pour calculer $\mathrm{P}(\mathrm{X}_1 = 1)$ avec le système complet d'événements $\{\mathrm{X}_0 = 1\}$ et $\{\mathrm{X}_0 = 2\}$. De même pour $\mathrm{P}(\mathrm{X}_1 = 2)$.

2 Même technique qu'à la question 1 pour les rangs n et $n+1$.

4 Exprimer l'événement $\{\mathrm{T} = k\}$ en fonction de $\mathrm{X}_0, \mathrm{X}_1, \ldots, \mathrm{X}_{k-1}$.

6 Invoquer une propriété sur les applications linéaires.

7 La continuité des fonctions de la question 6 doit justifier des passages à la limite.

Partie II

9 Majorer chaque composante de $\mathrm{A}x$ avec l'inégalité triangulaire sur les sommes.

10 Utiliser la question 9 avec un vecteur propre pour la valeur propre λ.

11 Normaliser un vecteur propre.

12 Pour le vecteur propre x de la question 11, isoler le terme $a_{i,i}\, x_i$ de la somme définissant la i^{e} composante de $\mathrm{A}x$.

14 En appliquant l'inégalité triangulaire renversée sur le membre de gauche de l'inégalité de la question, établir que $|\lambda| \geqslant a_{i,p} > 0$.

15 Montrer que le rang de $\mathrm{A} - \mathrm{I}_p$ vaut $p - 1$.

Partie III

18 Établir que le polynôme minimal de N vaut X^q avec $q \leqslant p$.

19 Rappelons la croissance comparées $n^\alpha = \mathrm{o}(q^n)$ pour $\alpha > 0$ et $q > 1$.

20 Appliquer la formule du binôme de Newton et remarquer que la somme ne comporte que p termes.

21 Calculer la limite de chacun des blocs diagonaux de A élevés à la puissance n.

Partie IV

22 Construire l'ensemble des vecteurs stochastiques de $\mathbb{R}^p$ (et non de $\mathbb{R}^n$) comme l'image réciproque d'un fermé par une fonction continue.

23 L'existence de μ_∞ est justifiée comme à la question 7.

24 Calculer les composantes de $\mu\,\mathrm{A}$ puis les sommer.

25 Montrer que μ_∞ est stochastique à l'aide des questions 24 puis 22.

27 Se ramener à la question 15 grâce à l'invariance du rang par transposée.

28 Considérer deux probabilités invariantes et utiliser les questions 26 et 27.

Partie V

29 Les indices d'un tableau de longueur n sont compris entre 0 et $n - 1$.

31 Adapter l'algorithme de recherche d'un maximum vu en cours.

32 Essayer avec deux boucles `for` imbriquées : la première pour parcourir les composantes du vecteur résultat, la seconde pour calculer le produit ligne $\times$ colonne à stocker dans la composante concernée.

33 Utiliser une boucle `while` qui maintient l'invariant suivant : les variables `muAvant` et `muApres` contiennent respectivement μ_{k-1} et μ_k avant le k^{e} tour de boucle.

I. Un exemple de chaîne de Markov

1 L'énoncé fournit les probabilités conditionnelles suivantes :

$$\mathrm{P}(\mathrm{X}_1 = 2 \,|\, \mathrm{X}_0 = 1) = \frac{1}{2} \qquad \text{et} \qquad \mathrm{P}(\mathrm{X}_1 = 1 \,|\, \mathrm{X}_0 = 2) = \frac{1}{4}$$

Puisque $\mathrm{P}(\cdot \,|\, \mathrm{X}_0 = 1)$ et $\mathrm{P}(\cdot \,|\, \mathrm{X}_0 = 2)$ sont des probabilités sur $(\Omega, \mathcal{F})$ et que les événements $\{\mathrm{X}_1 = 1\}$ et $\{\mathrm{X}_1 = 2\}$ sont complémentaires l'un de l'autre, on en déduit que

$$\mathrm{P}(\mathrm{X}_1 = 1 \,|\, \mathrm{X}_0 = 1) = 1 - \frac{1}{2} = \frac{1}{2} \qquad \text{et} \qquad \mathrm{P}(\mathrm{X}_1 = 2 \,|\, \mathrm{X}_0 = 2) = 1 - \frac{1}{4} = \frac{3}{4}$$

D'après la formule des probabilités totales avec le système complet $\big(\{\mathrm{X}_0 = i\}\big)_{i=1,2}$,

$$\begin{aligned} \mathrm{P}(\mathrm{X}_1 = 1) &= \mathrm{P}(\mathrm{X}_1 = 1 \,|\, \mathrm{X}_0 = 1)\,\mathrm{P}(\mathrm{X}_0 = 1) + \mathrm{P}(\mathrm{X}_1 = 1 \,|\, \mathrm{X}_0 = 2)\,\mathrm{P}(\mathrm{X}_0 = 2) \\ &= \frac{1}{2} \times \frac{1}{2} + \frac{1}{4} \times \frac{1}{2} = \frac{3}{8} \end{aligned}$$

$$\text{et} \quad \begin{aligned} \mathrm{P}(\mathrm{X}_1 = 2) &= \mathrm{P}(\mathrm{X}_1 = 2 \,|\, \mathrm{X}_0 = 1)\,\mathrm{P}(\mathrm{X}_0 = 1) + \mathrm{P}(\mathrm{X}_1 = 2 \,|\, \mathrm{X}_0 = 2)\,\mathrm{P}(\mathrm{X}_0 = 2) \\ &= \frac{1}{2} \times \frac{1}{2} + \frac{3}{4} \times \frac{1}{2} = \frac{5}{8} \end{aligned}$$

> Remarquons également que $\mathrm{P}(\mathrm{X}_1 = 2) = 1 - \mathrm{P}(\mathrm{X}_1 = 1) = 1 - \dfrac{3}{8} = \dfrac{5}{8}$.

Conclusion : La loi de X_1 est donnée par

x	1	2
$\mathrm{P}(\mathrm{X}_1 = x)$	3/8	5/8

.

2 Soit $n \in \mathbb{N}$. Les événements $\{\mathrm{X}_n = 1\}$ et $\{\mathrm{X}_n = 2\}$ forment un système complet d'événements. D'après la formule des probabilités totales,

$$\forall\, j \in \{1, 2\} \qquad \begin{aligned} \mathrm{P}(\mathrm{X}_{n+1} = j) = \mathrm{P}(\mathrm{X}_{n+1} = j \,|\, \mathrm{X}_n = 1)\,\mathrm{P}(\mathrm{X}_n = 1) \\ + \mathrm{P}(\mathrm{X}_{n+1} = j \,|\, \mathrm{X}_n = 2)\,\mathrm{P}(\mathrm{X}_n = 2) \end{aligned}$$

D'après l'énoncé, les probabilités conditionnelles de la formule précédente ne dépendent pas de n. Posons alors, avec les données de l'énoncé,

$$a_{1,2} = \mathrm{P}(\mathrm{X}_{n+1} = 2 \,|\, \mathrm{X}_n = 1) = \frac{1}{2} \quad \text{et} \quad a_{2,1} = \mathrm{P}(\mathrm{X}_{n+1} = 1 \,|\, \mathrm{X}_n = 2) = \frac{1}{4}$$

$$\text{puis} \quad a_{1,1} = \mathrm{P}(\mathrm{X}_{n+1} = 1 \,|\, \mathrm{X}_n = 1) = \frac{1}{2} \quad \text{et} \quad a_{2,2} = \mathrm{P}(\mathrm{X}_{n+1} = 2 \,|\, \mathrm{X}_n = 2) = \frac{3}{4}$$

La formule des probabilités totales se réécrit

$$\forall\, j \in \{1, 2\} \qquad (\mu_{n+1})_j = a_{1,j}\,(\mu_n)_1 + a_{2,j}\,(\mu_n)_2$$

d'où

$$\big((\mu_{n+1})_1 \quad (\mu_{n+1})_2\big) = \big((\mu_n)_1 \quad (\mu_n)_2\big) \times \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}$$

Finalement,

$$\boxed{\forall\, n \in \mathbb{N} \qquad \mu_{n+1} = \mu_n \times \mathrm{A} \quad \text{avec} \quad \mathrm{A} = \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & 3/4 \end{pmatrix}}$$

> Remarquons que la matrice A est stochastique. En effet, la ligne i correspond à la loi conditionnelle de X_{n+1} sachant l'événement $\{\mathrm{X}_n = i\}$.

3 D'après la relation de récurrence géométrique de la question 2,

$$\mu_5 = \mu_0\,\mathrm{A}^5 = \begin{pmatrix} 1/2 & 1/2 \end{pmatrix} \times \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & 3/4 \end{pmatrix}^5 \simeq \begin{pmatrix} 0{,}33 & 0{,}67 \end{pmatrix}$$

d'où La loi de X_5 est donnée par

x	1	2
$\mathrm{P}(\mathrm{X}_5 = x)$	$\simeq 0{,}33$	$\simeq 0{,}67$

.

4 Écrivons $$\{\mathrm{T} = 1\} = \{\mathrm{X}_0 = 2\} \cap \{\mathrm{X}_1 = 1\}$$

D'après la formule des probabilités conditionnelles,

$$\begin{aligned} \mathrm{P}(\mathrm{T} = 1) &= \mathrm{P}(\mathrm{X}_0 = 2, \mathrm{X}_1 = 1) \\ &= \mathrm{P}(\mathrm{X}_0 = 2)\,\mathrm{P}(\mathrm{X}_1 = 1 \mid \mathrm{X}_0 = 2) \end{aligned}$$

$$\boxed{\mathrm{P}(\mathrm{T} = 1) = \frac{1}{2} \times \frac{1}{4} = \frac{1}{8}}$$

Pour $k \geqslant 2$, $$\{\mathrm{T} = k\} = \Big(\bigcap_{i=0}^{k-1} \{\mathrm{X}_i = 2\} \Big) \cap \{\mathrm{X}_k = 1\}$$

D'après la formule des probabilités composées,

$$\begin{aligned} \mathrm{P}(\mathrm{T} = k) &= \mathrm{P}(\mathrm{X}_0 = 2)\,\mathrm{P}(\mathrm{X}_1 = 2 \mid \mathrm{X}_0 = 2)\,\mathrm{P}(\mathrm{X}_2 = 2 \mid \mathrm{X}_0 = 2, \mathrm{X}_1 = 2) \cdots \\ &= \mathrm{P}(\mathrm{X}_0 = 2) \times \prod_{i=1}^{k-1} \mathrm{P}(\mathrm{X}_i = 2 \mid \mathrm{X}_0 = \cdots = \mathrm{X}_{i-1} = 2) \\ &\qquad\qquad \times\ \mathrm{P}(\mathrm{X}_k = 1 \mid \mathrm{X}_0 = \cdots = \mathrm{X}_{k-1} = 2) \end{aligned}$$

Or, l'état de la particule au rang $i+1$ ne dépend que de son état au rang $i \in \mathbb{N}$ donc

$$\begin{aligned} \mathrm{P}(\mathrm{T} = k) &= \mathrm{P}(\mathrm{X}_0 = 2) \Big(\prod_{i=1}^{k-1} \mathrm{P}(\mathrm{X}_i = 2 \mid \mathrm{X}_{i-1} = 2) \Big) \mathrm{P}(\mathrm{X}_k = 1 \mid \mathrm{X}_{k-1} = 2) \\ &= \frac{1}{2} \cdot \Big(\prod_{i=1}^{k-1} \frac{3}{4} \Big) \cdot \frac{1}{4} = \frac{1}{2} \cdot \frac{3^{k-1}}{4^{k-1}} \cdot \frac{1}{4} \end{aligned}$$

soit $$\boxed{\forall\, k \geqslant 2 \qquad \mathrm{P}(\mathrm{T} = k) = \frac{1}{6} \cdot \Big(\frac{3}{4}\Big)^k} \qquad \text{(valable pour } k = 1)$$

> Attention à ne pas invoquer une loi géométrique dans cette question, alors même qu'il s'agit de déterminer la loi du rang d'apparition d'un premier succès, ici « être dans l'état 1 ». En effet, l'expérience aléatoire ne se présente pas comme la répétition d'épreuves de Bernoulli *indépendantes*.

5 Calculons le polynôme caractéristique χ_A de A :

$$\begin{aligned} \chi_\mathrm{A} = \det(\mathrm{X}\,\mathrm{I}_2 - \mathrm{A}) &= \begin{vmatrix} \mathrm{X} - 1/2 & -1/2 \\ -1/4 & \mathrm{X} - 3/4 \end{vmatrix} \underset{\mathrm{C}_1 \leftarrow \mathrm{C}_1 + \mathrm{C}_2}{=} \begin{vmatrix} \mathrm{X} - 1 & -1/2 \\ \mathrm{X} - 1 & \mathrm{X} - 3/4 \end{vmatrix} \\ &= (\mathrm{X} - 1) \begin{vmatrix} 1 & -1/2 \\ 1 & \mathrm{X} - 3/4 \end{vmatrix} = (\mathrm{X} - 1)\,(\mathrm{X} - 3/4 + 1/2) \\ \chi_\mathrm{A} &= (\mathrm{X} - 1)(\mathrm{X} - 1/4) \end{aligned}$$

Ainsi, 1 et 1/4 sont les deux valeurs propres de la matrice A de taille 2, elles sont nécessairement simples d'où

$$\boxed{\text{La matrice A est diagonalisable.}}$$

Centrale Maths 1 MP 2017 — Corrigé

Ce corrigé est proposé par Yvon Vignaud (professeur en CPGE) ; il a été relu par Sélim Cornet (ENS Paris-Saclay) et Vincent Puyhaubert (professeur en CPGE).

Ce problème porte sur les matrices définies positives et sur les propriétés des matrices dont la partie symétrique est définie positive. Les parties II et III sont indépendantes mais elles exploitent intensivement les résultats de la partie I sur les matrices symétriques positives.

La résolution de ce problème utilise de nombreuses parties du programme de deuxième année : calcul matriciel par blocs, réduction des matrices, automorphismes orthogonaux des espaces euclidiens, exponentielles de matrices, systèmes différentiels linéaires, intégration de fonctions à valeurs vectorielles (matricielles en l'occurrence).

- Dans la première partie, après quelques résultats classiques sur les matrices symétriques et antisymétriques, on démontre dans la sous-partie I.B l'existence et l'unicité de la racine carrée d'une matrice définie positive. À partir d'un encadrement des valeurs propres réelles par les valeurs propres extrêmes de la partie symétrique, on obtient ensuite une minoration du déterminant par le déterminant de la partie symétrique lorsque celle-ci est définie positive. On y établit enfin une caractérisation, parmi les matrices symétriques, de celles qui sont les parties symétriques de matrices orthogonales.
- La deuxième partie introduit la notion de matrice F-singulière avec F un sous-espace vectoriel de $\mathcal{M}_{n,1}(\mathbb{R})$; ce concept généralise la notion de matrice non inversible, qui correspond au cas particulier $\mathrm{F} = \mathcal{M}_{n,1}(\mathbb{R})$. Le résultat principal est une caractérisation de la F-singularité reposant sur la non inversibilité d'une certaine matrice, construite par blocs à partir d'une base de $\mathrm{F}^{\perp}$. Les sous-parties II.A et II.C se restreignent aux cas où la dimension de F vaut respectivement $n-1$ et $n-2$, ces résultats étant ensuite étendus à toute dimension dans la sous-partie II.E ; les parties II.B et II.D traitent un exemple concret pour illustrer les résultats respectifs de II.A. et II.C.
- La troisième partie étudie les matrices dites positivement stables, c'est-à-dire celles dont les valeurs propres ont des parties réelles strictement positives. Dans la sous-partie III.A, on montre que si deux matrices positivement stables commutent alors leur somme reste positivement stable ; un contre-exemple dans le cas non commutatif est explicitement demandé. Le résultat principal de la sous-partie III.B est le fait que pour A positivement stable et $t \to +\infty$, la norme de $\exp(-t\mathrm{A})$ tend exponentiellement vite vers 0. Enfin, à l'aide de ce résultat, on parvient dans la sous-partie III.C à montrer que si A est positivement stable alors il existe une unique matrice B telle que $\mathrm{A}^{\mathrm{T}}\mathrm{B} + \mathrm{BA} = \mathrm{I}_n$ et que cette matrice B est nécessairement définie positive.

En conclusion, il s'agit d'un sujet long et plutôt difficile, compte tenu de la maîtrise nécessaire d'une grande partie du programme. Les exemples et contre-exemples explicites étudiés permettent de mieux s'approprier les concepts algébriques introduits par ce sujet ; en outre, sa résolution complète met bien en valeur l'importance des propriétés de symétrie et de positivité en mathématiques, et plus particulièrement ici dans un cadre matriciel.

Indications

Partie I

- I.A.2 Penser aux propriétés du projeté orthogonal sur un sous-espace vectoriel.
- I.B.1 Appliquer le théorème spectral à la matrice A_s et procéder par double implication.
- I.B.2 Montrer que pour tout vecteur colonne X on a $X^T A_s X = X^T AX$ et appliquer ensuite cette identité à un vecteur propre X de A.
- I.B.3.a Pour l'existence, considérer la matrice diagonale dont les coefficients diagonaux sont les racines carrées des valeurs propres de A_s ; pour l'unicité, montrer que toute matrice qui convient commute avec A_s et raisonner sur les endomorphismes induits sur les sous-espaces propres de A_s.
- I.B.3.b Considérer la matrice $Q = B^{-1} A_a B^{-1}$ avec la matrice B fournie par la question I.B.3.a.
- I.B.3.c Montrer que $-Q^T Q \in \mathcal{S}_n^+(\mathbb{R})$ et en déduire que les valeurs propres de Q sont imaginaires pures. Trigonaliser alors Q pour exprimer $\det(I_n + Q)$ sous la forme d'un produit.
- I.B.4 Vérifier que $A \left(A^{-1}\right)_s A^T = A_s$.
- I.C.1 Appliquer l'inégalité de Cauchy-Schwarz à X et AX où X est un vecteur propre de A et exploiter à nouveau l'égalité $X^T A_s X = X^T AX$ vue à la question I.B.2.
- I.C.2 Rappeler les différentes formes des éléments de $\mathcal{O}_2(\mathbb{R})$ et calculer leurs parties symétriques.
- I.C.3.a Diagonaliser S et regrouper deux par deux ses valeurs propres distinctes de 1 et -1 pour obtenir une écriture diagonale par blocs.
- I.C.3.b Réduire la matrice orthogonale A dans une base orthonormée afin d'obtenir une matrice diagonale par blocs semblable à S.

Partie II

- II.A.3 Appliquer la question II.A.2 et considérer le vecteur colonne $Y = \begin{pmatrix} X \\ -\lambda \end{pmatrix}$.

  Porter un soin particulier à justifier que les vecteurs utilisés dans la preuve sont non nuls.
- II.A.6 Appliquer une nouvelle fois l'identité vue à la question I.B.2 à un élément non nul du noyau de $\left(A^{-1}\right)_s$.
- II.A.7 Exploiter le résultat de la question I.B.4.
- II.B.3 Calculer A^{-1} puis sa partie symétrique afin de trouver un vecteur du noyau de cette dernière.
- II.C.4 Remarquer que $N^T A^{-1} N$ est une matrice de taille 2.
- II.C.5 Pour le sens direct, considérer $P' = (A^{-1})^T P$.
- II.C.6 Exprimer $Y^T AZ$ et $Z^T AY$ en fonction de $Y^T A_s Z$ et $Y^T A_a Z$ pour tous vecteurs Y et Z.
- II.C.7 Prendre la matrice B donnée par la question I.B.3 telle que $B^2 = A_s$ et minorer l'expression de la question II.C.6 grâce à l'inégalité de Cauchy-Schwarz.

II.E.1 Il s'agit de définir une matrice N' telle que la nullité du déterminant de $(N')^T AN'$ implique la F-singularité de A. Généraliser les idées des parties II.A et II.C en utilisant une base de l'orthogonal de F.

II.E.4 Exprimer le déterminant en regroupant les valeurs propres complexes non réelles par paires conjuguées.

II.E.5 Justifier qu'avec la matrice N' définie à la question II.E.1, le résultat de cette question est en réalité une équivalence, puis utiliser tout ce qui précède.

Partie III

III.A.1 Distinguer deux cas suivant la nature des valeurs propres de A (réelles ou complexes non réelles conjuguées).

III.A.2.a Proposer un contre-exemple à l'aide de matrices triangulaires.

III.A.2.b Considérer l'endomorphisme induit par A sur $E_\lambda(A+B)$.

III.A.3.a Penser encore une fois à l'identité vue à la question I.B.2 afin d'exploiter la positivité de A_s.

III.B.1 Suivre l'indication afin d'exprimer u sous forme intégrale

$$u(t) = u(0)e^{-\lambda t} + \int_0^t e^{-\lambda(t-s)} v(s)\,ds$$

III.B.2 À l'aide de la question III.B.1, montrer par récurrence forte (descendante) que les fonctions u_n, puis $u_{n-1}, \ldots, u_2$ et u_1 sont bornées.

III.C.1 Montrer que les endomorphismes $M \longmapsto MA$ et $M \longmapsto A^T M$ sont positivement stables.

III.C.2.b Calculer $(BA)_s$ pour minorer $\det(BA)$ à l'aide du résultat de la question I.B.3. Regrouper les valeurs propres complexes conjuguées de χ_A pour montrer que $\det(A) > 0$.

III.C.3.a Montrer que

$$\forall\, M \in \mathcal{M}_n(\mathbb{C}) \qquad (\exp M)^T = \exp(M^T)$$

III.C.3.b Dériver $C : t \mapsto A^T W(t) + W(t)A + V(t)$.

III.C.3.c Choisir α comme à la question III.B.3 et montrer que les coefficients de la matrice $V(t)$ sont dominés par $e^{-\alpha t}$ lorsque $t \to +\infty$. Pour montrer que $B \in \mathcal{S}_n^{++}(\mathbb{R})$, montrer par exemple que $X^T BX \geqslant X^T W(1)X$.

I. Résultats préliminaires

I.A.1 Le préambule du sujet rappelle que toute matrice $\mathrm{A} \in \mathcal{M}_n(\mathbb{R})$ peut s'écrire comme ceci : $\mathrm{A} = \mathrm{A}_s + \mathrm{A}_a$ où A_s est symétrique et A_a antisymétrique. On en déduit que $\mathcal{M}_n(\mathbb{R}) \subset \mathcal{S}_n(\mathbb{R}) + \mathcal{A}_n(\mathbb{R})$ et, l'inclusion réciproque étant immédiate,

$$\mathcal{M}_n(\mathbb{R}) = \mathcal{S}_n(\mathbb{R}) + \mathcal{A}_n(\mathbb{R})$$

Pour montrer que ces sous-espaces sont supplémentaires orthogonaux, il suffit désormais de montrer qu'ils sont orthogonaux. Soient $\mathrm{S} = (s_{ij})_{1\leqslant i,j\leqslant n} \in \mathcal{S}_n(\mathbb{R})$ et $\mathrm{A} = (a_{ij})_{1\leqslant i,j\leqslant n} \in \mathcal{A}_n(\mathbb{R})$. Calculons leur produit scalaire

$$\langle \mathrm{S}, \mathrm{A}\rangle = \mathrm{Tr}\,\left(\mathrm{S}^{\mathrm{T}}\mathrm{A}\right) = \sum_{i=1}^{n}\sum_{j=1}^{n} s_{ij}a_{ij}$$

Grâce aux relations $s_{ij} = s_{ji}$ et $a_{ij} = -a_{ji}$ valides pour tous $1 \leqslant i, j \leqslant n$, l'interversion de la double somme donne

$$\langle \mathrm{S}, \mathrm{A}\rangle = \sum_{j=1}^{n}\sum_{i=1}^{n} s_{ij}a_{ij} = -\sum_{j=1}^{n}\sum_{i=1}^{n} s_{ji}a_{ji} = -\langle \mathrm{S}, \mathrm{A}\rangle$$

d'où $\langle \mathrm{S}, \mathrm{A}\rangle = 0$. Ainsi, les sous-espaces vectoriels $\mathcal{S}_n(\mathbb{R})$ et $\mathcal{A}_n(\mathbb{R})$ sont orthogonaux et

$$\boxed{\mathcal{M}_n(\mathbb{R}) = \mathcal{S}_n(\mathbb{R}) \overset{\perp}{\oplus} \mathcal{A}_n(\mathbb{R})}$$

Considérons maintenant la base canonique $(\mathrm{E}_{ij})_{1\leqslant i,j\leqslant n}$ de $\mathcal{M}_n(\mathbb{R})$ et remarquons que la famille $\mathcal{S} = (\mathrm{E}_{ij} + \mathrm{E}_{ji})_{1\leqslant i\leqslant j\leqslant n}$ est une base de $\mathcal{S}_n(\mathbb{R})$ de sorte que

$$\dim \mathcal{S}_n(\mathbb{R}) = \mathrm{Card}\ \mathcal{S} = \sum_{j=1}^{n} j = \frac{n(n+1)}{2}$$

Comme $\mathcal{A}_n(\mathbb{R})$ est un supplémentaire de $\mathcal{S}_n(\mathbb{R})$ dans $\mathcal{M}_n(\mathbb{R})$,

$$\dim \mathcal{A}_n(\mathbb{R}) = \dim \mathcal{M}_n(\mathbb{R}) - \dim \mathcal{S}_n(\mathbb{R}) = n^2 - \frac{n(n+1)}{2} = \frac{n(n-1)}{2}$$

$$\boxed{\dim \mathcal{S}_n(\mathbb{R}) = n(n+1)/2 \quad \text{et} \quad \dim \mathcal{A}_n(\mathbb{R}) = n(n-1)/2}$$

La base canonique est une base orthonormée de $\mathcal{M}_n(\mathbb{R})$ pour le produit scalaire $\langle\cdot,\cdot\rangle$. Par bilinéarité du produit scalaire, il en découle que la famille $(\mathrm{E}_{ij}+\mathrm{E}_{ji})_{1\leqslant i\leqslant j\leqslant n}$ est constituée de matrices non nulles deux à deux orthogonales ; ceci prouve d'une autre façon la liberté de cette famille, qui se trouve donc être une base orthogonale (non orthonormée) de $\mathcal{S}_n(\mathbb{R})$.

La diagonalisation de l'endomorphisme de transposition $\mathcal{T} : \mathrm{A} \mapsto \mathrm{A}^{\mathrm{T}}$ fournit un autre point de vue sur les résultats de la question I.A.1.

L'application $\mathcal{T}$ est une symétrie vectorielle de $\mathcal{M}_n(\mathbb{R})$ puisque c'est une application linéaire de $\mathcal{M}_n(\mathbb{R})$ dans lui-même qui vérifie $\mathcal{T}\circ\mathcal{T} = \mathrm{id}$. En effet,

$$\forall\, \mathrm{A} \in \mathcal{M}_n(\mathbb{R}) \qquad (\mathcal{T}\circ\mathcal{T})\,(\mathrm{A}) = (\mathrm{A}^{\mathrm{T}})^{\mathrm{T}} = \mathrm{A}$$

En tant que symétrie vectorielle, $\mathcal{T}$ est diagonalisable et ses espaces propres

$$\mathcal{S}_n(\mathbb{R}) = \mathrm{Ker}\ (\mathcal{T} - \mathrm{id}) = \mathrm{E}_1(\mathcal{T}) \quad \text{et} \quad \mathcal{A}_n(\mathbb{R}) = \mathrm{Ker}\ (\mathcal{T} + \mathrm{id}) = \mathrm{E}_{-1}(\mathcal{T})$$

sont supplémentaires dans $\mathcal{M}_n(\mathbb{R})$. De plus, en notant $\mathrm{A} = (a_{ij})_{1\leqslant i,j\leqslant n}$ et $\mathrm{B} = (b_{ij})_{1\leqslant i,j\leqslant n}$ deux matrices arbitraires,

$$\langle \mathcal{T}(\mathrm{A}), \mathrm{B}\rangle = \sum_{1\leqslant i,j\leqslant n} a_{ij}b_{ij} = \sum_{1\leqslant i,j\leqslant n} b_{ij}a_{ij} = \langle \mathrm{A}, \mathcal{T}(\mathrm{B})\rangle$$

Centrale Maths 2 MP 2017 — Corrigé

Ce corrigé est proposé par Alban Levy (docteur en mathématiques) ; il a été relu par Yvon Vignaud (professeur en CPGE) et Florian Metzger (docteur en mathématiques).

Ce problème d'analyse propose l'étude de la représentation de la loi d'une variable aléatoire comme loi d'une somme de variables aléatoires discrètes indépendantes. En particulier, l'égalité des fonctions génératrices de deux variables aléatoires discrètes étant équivalente à l'égalité de leurs lois, l'étude des distributions de probabilité est souvent ramenée à une étude des fonctions génératrices. La première partie est largement indépendante des deux autres.

- Dans la première partie, la décomposition étudiée est une somme de deux variables à valeurs entières. Quelques lois classiques y sont vues, comme les lois binomiale et uniforme. Les polynômes et fonctions génératrices y jouent un rôle important.
- Dans la deuxième partie, on étudie quelques exemples de variables aléatoires à valeurs dans $\mathbb{N}$ ayant la propriété d'être infiniment divisibles, c'est-à-dire s'écrivant comme une somme de m variables aléatoires mutuellement indépendantes et de même loi, pour $m \in \mathbb{N}^*$ quelconque. Les variables aléatoires proposées sont bornées ou de Poisson.
- La troisième partie étudie la cas général des variables infiniment divisibles. Le résultat principal est la caractérisation de ces variables de deux façons.

Il s'agit d'un sujet ambitieux qui navigue astucieusement entre algèbre et analyse, utilisant comme ligne directrice les lois de probabilités classiques mais incluant de nombreux raisonnements autour des polynômes, séries numériques et séries entières. Outre une bonne maîtrise des probabilités au programme de MP, il faut donc être à l'aise avec une importante partie du programme d'analyse. On trouvera par exemple une équation différentielle, des produits infinis ou un polynôme cyclotomique. On démontre aussi le lemme de Borel-Cantelli. Bien que les trois derniers ne soient pas au programme, il est utile de les avoir déjà étudiés.

La résolution de ce sujet est gratifiante car elle tisse des liens entre des concepts mathématiques habituellement éloignés, requérant parfois des raisonnements purement algébriques.

Indications

Partie I

I.A.1 Calculer les dérivées n-ièmes de G_X en 0 pour exprimer la loi de X.

I.A.2 Exprimer G_X comme une espérance pour utiliser l'indépendance de Y et Z.

I.A.3 Séparer les cas $n = 1$ (par l'absurde) et $n \geqslant 2$ (somme de binomiales).

I.A.4.a Supposer que le degré de U vaut 1 ou 2 et déterminer les coefficients de U.

I.A.4.b Voir que $A(T)/4$ est la fonction génératrice du carré d'une variable aléatoire de loi bien connue.

I.B.1.a Utiliser l'existence et l'unicité de la division euclidienne.

I.B.1.b Obtenir la loi du couple (Q, R) par unicité d'une division euclidienne puis sommer pour obtenir les lois marginales.

I.B.1.c Montrer l'indépendance de Q et R en tant que variables aléatoires.

I.B.2.a Raisonner par l'absurde.

I.B.2.b Décomposer U en produit de polynômes de degré 1.

I.B.2.c Écrire la dérivée r-ième de UV en 0 de deux façons.

I.B.2.d Procéder par récurrence sur k allant de 1 à r, en utilisant la dérivée $(k+1)$-ième de UV en 0 pour montrer que les propriétés $u_i \in \{0, 1\}$ et $v_i \in \{0, 1\}$ sont vraies pour $i \in [\![1\,; k + 1]\!]$.

I.B.2.e Regarder $(UV)(1)$ droit dans les yeux : un produit d'entiers se cache dans ce terme.

Partie II

II.A.1 Utiliser le résultat admis dans le préambule pour avoir m copies de X/m.

II.A.2.a Établir une première inégalité entre $\mathbb{P}(X_1 > M/n)^n$ et $\mathbb{P}(X > M)$ et une autre entre $\mathbb{P}(|X_1| > M/n)$ et $\mathbb{P}(X_1 > M/n) + \mathbb{P}(X_1 < -M/n)$.

II.A.2.b Majorer $\mathbb{V}(X_1)$ par $\mathbb{E}[(X_1)^2]$.

II.A.3 Montrer que $\mathbb{V}(X) = 0$.

II.B.1 Se ramener à une variable aléatoire bornée.

II.B.2 Se référer à la question où a été établi le résultat sur la fonction génératrice.

II.B.3 Sommer m variables aléatoires de lois judicieusement choisies.

II.B.4 Décomposer chacun des r termes X_i en m variables aléatoire indépendantes, et s'assurer que ces rm variables aléatoires peuvent être choisies mutuellement indépendantes.

II.C.1.a Partir de $A = (A \cap B) \cup (A \cap \overline{B})$ et aller vers une majoration de $\mathbb{P}(A \cap B)$ par $\mathbb{P}(A)$ et $\mathbb{P}(B)$.

II.C.1.b Utiliser les définitions des fonctions génératrices puis majorer en utilisant la question II.C.1.a.

II.C.2.a Majorer $\mathbb{P}(Z_n)$ par le reste d'une série convergente faisant intervenir les U_i.

II.C.2.b Relier l'ensemble $\{i \in \mathbb{N}^* | U_i \neq 0\}$ aux variables Z_m pour $m \in \mathbb{N}$.

II.C.2.c Utiliser les questions II.C.2.b et II.C.1.b et relier l'événement $(S \neq S_n)$ à Z_{n+1}.

II.C.3.a Noter que $\mathbb{P}(X_i \neq 0) = 1 - e^{-\lambda_i}$ puis utiliser une comparaison de séries.

II.C.3.b Étudier la limite de la suite de fonctions génératrices $G_{X_1+\cdots+X_n}$ puis conclure par unicité de la limite.

II.C.3.c Vérifier que les hypothèses de la question II.C.2 s'appliquent puis utiliser le résultat de la question II.B.4.

Partie III

III.A.1 Procéder par récurrence sur k pour montrer l'existence et l'unicité de la famille $(\lambda_i)_{i=1,\ldots,k}$.

III.A.2 Majorer certaines probabilités en jeu par $\mathbb{P}(X \geqslant 1)$.

III.A.3 Procéder par récurrence sur $k \in \mathbb{N}^*$.

III.A.4.a Invoquer la règle de d'Alembert.

III.A.5 Utiliser un produit de Cauchy puis le théorème de Cauchy-Lipschitz.

III.B.1 Se servir de la formule construisant $\lambda_1, \ldots, \lambda_k$.

III.B.2 Prouver la convergence normale de la série définissant H_X sur $[-1\,;1]$.

III.B.3 Montrer l'égalité des fonctions génératrices de X et de la somme de $\sum iX_i$.

III.C.1.a Il suffit que tous les $X_{n,i}$ pour $n \in \mathbb{N}^*$ et $i \in [\![1\,;n]\!]$ soient négatifs pour que X le soit.

III.C.1.b Calculer la valeur de $\mathbb{P}(X_{n,1} = 0)$.

III.C.1.c Si la variable X_1 possède une valeur non entière et que les autres variables X_j pour $j \geqslant 2$ sont nulles, alors la variable X n'est pas entière.

III.C.2.a On connaît $\mathbb{P}(X_{n,1} = 0)$.

III.C.3.b Identifier les coefficients des deux séries entières définissant nH_n et H_X.

III.C.4 Utiliser les questions III.C.2.a et b.

III.C.5.a Rechercher trois implications dans les questions précédentes.

III.C.5.b Se servir de $X - m_X$ où m_X est le minimum presque sûr de X.

III.C.5.c Calculer explicitement la famille $(\lambda_k)_{k\in\mathbb{N}^*}$.

I. Variables aléatoires entières décomposables

I.A.1 Si $\mathrm{X} \sim \mathrm{X}'$, alors pour tout $n \in \mathbb{N}$, $\mathbb{P}(\mathrm{X} = n) = \mathbb{P}(\mathrm{X}' = n)$ d'où l'égalité de chaque terme général des séries entières définissant, pour tout $t \in \mathbb{R}$, $\mathrm{G_X}(t)$ et $\mathrm{G_{X'}}(t)$. Ainsi, leurs sommes sont égales sur leur domaine de définition.

Réciproquement, supposons que $\mathrm{G_X} = \mathrm{G_{X'}}$. Les coefficients de la série entière définissant $\mathrm{G_X}$ étant des probabilités, donc compris entre 0 et 1, par comparaison de séries à termes positifs, cette série a un rayon de convergence supérieur ou égal à celui de la série $\sum t^n$, c'est-à-dire 1. La fonction $\mathrm{G_X}$ est de plus de classe $\mathscr{C}^\infty$ au voisinage de 0 et

$$\forall n \in \mathbb{N} \qquad \mathrm{G_X}^{(n)}(0) = n!\,\mathbb{P}(\mathrm{X} = n)$$

et de même pour $\mathrm{G_{X'}}$. L'égalité de $\mathrm{G_X}$ et de $\mathrm{G_{X'}}$ entraînant l'égalité de leurs dérivées en zéro, on en déduit que

$$\boxed{\mathrm{X} \sim \mathrm{X}' \iff \mathrm{G_X} = \mathrm{G_{X'}}}$$

I.A.2 Fixons $t \in \mathbb{R}$ tel que la somme définissant $\mathrm{G_X}(t)$ soit convergente, ce qui inclut l'intervalle $]-1\,;1\,[$ d'après la question I.A.1. Réécrivons alors $\mathrm{G_X}(t)$ comme une espérance grâce au théorème de transfert :

$$\mathrm{G_X}(t) = \sum_{n=0}^{+\infty} \mathbb{P}(\mathrm{X} = n)t^n = \mathbb{E}[t^{\mathrm{X}}]$$

La décomposition $\mathrm{X} \sim \mathrm{Y} + \mathrm{Z}$ donne de plus l'égalité $\mathbb{E}[t^{\mathrm{X}}] = \mathbb{E}[t^{\mathrm{Y+Z}}] = \mathbb{E}[t^{\mathrm{Y}}\, t^{\mathrm{Z}}]$. Enfin, l'indépendance de Y et Z entraînant celle de t^{Y} et t^{Z}, on peut écrire

$$\mathbb{E}[t^{\mathrm{Y}}\, t^{\mathrm{Z}}] = \mathbb{E}[t^{\mathrm{Y}}]\,\mathbb{E}[t^{\mathrm{Z}}] = \mathrm{G_Y}(t)\,\mathrm{G_Z}(t)$$

en reconnaissant l'expression de $\mathrm{G_Y}$ et $\mathrm{G_Z}$. Ainsi,

> Pour Y et Z indépendants, l'égalité $\mathrm{G_{Y+Z}} = \mathrm{G_Y}\,\mathrm{G_Z}$ est vraie sur le domaine de définition commun à $\mathrm{G_Y}$ et $\mathrm{G_Z}$. En particulier, $\mathrm{G_X} = \mathrm{G_Y}\,\mathrm{G_Z}$ sur $]-1\,;1\,[$.

I.A.3 • Si $n = 1$, $\mathrm{X} \sim \mathcal{B}(n,p)$ suit une loi de Bernoulli de paramètre p. Montrons par l'absurde que X n'est pas décomposable. Supposons donc que $\mathrm{X} \sim \mathrm{Y} + \mathrm{Z}$ avec Y et Z indépendants, à valeurs dans $\mathbb{N}$ et presque sûrement non constantes. Comme

$$(\mathrm{Y} \geqslant 1) \cap (\mathrm{Z} \geqslant 1) \subset (\mathrm{Y} + \mathrm{Z} \geqslant 2)$$

et
$$\mathbb{P}(\mathrm{Y} \geqslant 1, \mathrm{Z} \geqslant 1) = \mathbb{P}(\mathrm{Y} \geqslant 1)\,\mathbb{P}(\mathrm{Z} \geqslant 1)$$

par indépendance de Y et Z, on a

$$\mathbb{P}(\mathrm{X} \geqslant 2) = \mathbb{P}(\mathrm{Y} + \mathrm{Z} \geqslant 2) \geqslant \mathbb{P}(\mathrm{Y} \geqslant 1, \mathrm{Z} \geqslant 1) = \mathbb{P}(\mathrm{Y} \geqslant 1)\,\mathbb{P}(\mathrm{Z} \geqslant 1)$$

Comme Y est à valeurs dans $\mathbb{N}$ et non constante presque sûrement, $\mathbb{P}(\mathrm{Y} = 0) < 1$. Il s'ensuit par additivité de $\mathbb{P}$ que $\mathbb{P}(\mathrm{Y} \geqslant 1) = \mathbb{P}(\mathrm{Y} \in \mathbb{N}) - \mathbb{P}(\mathrm{Y} = 0) > 0$, et de même $\mathbb{P}(\mathrm{Z} \geqslant 1) > 0$. Or, $\mathbb{P}(\mathrm{X} \geqslant 2) = 0$ car X suit une loi de Bernoulli. On obtient alors la contradiction suivante montrant finalement que X n'est pas décomposable :

$$0 = \mathbb{P}(\mathrm{X} \geqslant 2) \geqslant \mathbb{P}(\mathrm{Y} \geqslant 1)\,\mathbb{P}(\mathrm{Z} \geqslant 1) > 0$$

• Si $n \geqslant 2$, X suit une loi binomiale $\mathcal{B}(n,p)$. Le résultat donné par l'énoncé garantit l'existence de n variables aléatoires indépendantes $\mathrm{X}_1, \ldots, \mathrm{X}_n$ suivant une loi de Bernoulli de paramètre $p \in\,]\,0\,;1\,[$. Vérifions que $\mathrm{Y} = \mathrm{X}_1$ et $\mathrm{Z} = \mathrm{X}_2 + \cdots + \mathrm{X}_n$ forment une décomposition de X. Leur indépendance est garantie par construction

Centrale Informatique commune MP 2017 — Corrigé

Ce corrigé est proposé par Cyril Ravat (professeur en CPGE) ; il a été relu par Julien Dumont (professeur en CPGE) et Jean-Julien Fleck (professeur en CPGE).

Ce sujet d'informatique a pour contexte la mission martienne *Mars Exploration Rovers*. Celle-ci conduit à une étude en trois parties des aspects robotiques de l'exploration, principalement le déplacement du système sur une carte : calculs de distances parcourues et optimisation du chemin reliant tous les points suivant deux approches. De nombreuses questions font intervenir des tableaux `numpy`, donc l'utilisation des fonctions associées, alors que des listes de listes suffiraient la plupart du temps. Un catalogue de ces fonctions est fourni en fin de sujet, avec des explications claires et des exemples.

- La première partie est composée de trois sous-parties indépendantes et de difficultés inégales : la première permet de réaliser deux fonctions élémentaires de génération de carte et de calcul de distance ; la deuxième traite très brièvement de traitement d'image ; la troisième étudie le fonctionnement de la base de données des analyses à effectuer. On regrette la difficulté importante de la première question, qui a dû décourager nombre de candidats, du moins ceux qui ont correctement lu le sujet. Comparativement, les deux questions sur le traitement d'image sont d'une simplicité déconcertante. Enfin, on note que les requêtes SQL demandent l'utilisation d'une syntaxe (`IS NULL`) conceptuellement complexe et certainement peu enseignée.
- La deuxième partie aborde le problème très classique dit « du voyageur de commerce » : il faut optimiser le passage du robot par tous les points d'une liste une seule et unique fois. On met en place ici quelques fonctions élémentaires puis on réalise l'algorithme glouton du plus proche voisin. L'ensemble est équilibré et suffisamment progressif en termes de difficulté.
- La dernière partie continue le travail précédent en proposant pour le même problème l'étude d'un autre type de solution, lui aussi classique : un algorithme génétique. Les questions sont bien détaillées et se suivent parfaitement ; y répondre donne le plaisir de construire petit à petit un algorithme complexe et utilisable sur un cas concret.

Ce sujet laisse au final deux impressions : celle d'un début mal maîtrisé et peu progressif, suivie dans les deux dernières parties d'un sujet classique permettant aux candidats de s'exprimer correctement. Il contient quelques questions de langage SQL mais n'aborde pas la partie ingénierie numérique du programme et ne demande aucune démonstration théorique.

Indications

Partie I

I.A.1.a Le plus simple est de continuer à générer des couples *tant que* le résultat n'en contient pas `n` et d'enregistrer ces couples uniquement s'ils ne sont pas déjà présents dans le résultat, à l'aide de l'opérateur `not in`.

I.A.2 Ce genre de matrices est symétrique, on évite de doubler les calculs. La distance entre deux points se calcule à l'aide du théorème de Pythagore.

I.B Il existe une fonction permettant d'obtenir les dimensions d'un tableau `numpy`.

I.C.1 Pour tester si un champ est égal à `NULL`, ce n'est pas l'opérateur `=` mais l'opérateur `IS` qui est nécessaire : `champ IS NULL` ou `champ IS NOT NULL`.

I.C.3 Un seul résultat par exploration, pour plusieurs explorations, car il s'agit d'une requête d'agrégation. Les champs à calculer et la sélection des explorations terminées se font sur deux tables, une jointure est donc nécessaire.

I.C.4 L'énoncé est incomplet, il ne dit pas comment sont stockés les entiers dans la base. On peut imaginer qu'ils sont positifs et enregistrés sur 64 bits.

I.C.5 Il faut réaliser une jointure. Toutes les tables décrites dans l'énoncé ne sont pas utiles. Les champs `EX_NUM` et `TY_NUM` sont présents et équivalents dans trois tables.

Partie II

II.A.2 Générer une liste de `n` valeurs booléennes permet de savoir rapidement si un point a déjà été visité, sans utiliser la syntaxe `in`.

II.C.1 Il s'agit d'une recherche classique de minimum, mais uniquement sur les points encore disponibles à la `i`e itération. Comme en II.A.2, une liste de valeurs booléennes peut aider. Faire attention à l'initialisation du critère minimal recherché.

II.C.3 Les points donnés sont alignés, faire un dessin au brouillon.

Partie III

III.A Pour générer une liste aléatoire d'entiers, le plus simple est d'utiliser la fonction de permutation donnée en fin d'énoncé, sur la liste complète.

III.B Pour trier les chemins, la méthode `sort` fonctionne immédiatement et il n'est pas utile de la coder à la main. Attention, il faut supprimer directement des éléments de la liste `p`, il est interdit d'écrire `p = p[0:len(p)//2]` par exemple.

III.C.1 La fonction `random.sample` permet d'obtenir plusieurs valeurs distinctes prises aléatoirement au sein d'une liste.

III.C.2 Penser à utiliser les fonctions `muter_chemin` et `longueur_chemin` réalisées aux questions précédentes.

III.D.1 On peut concaténer deux listes avec l'opérateur `+`.

III.E.1 Attention à la syntaxe des fonctions précédentes, notamment de celles qui ne renvoient pas de valeur.

III.E.2 Où est le meilleur chemin dans une génération ? Est-il modifié ?

III.E.3 Classiquement, une série converge si sa valeur n'évolue plus beaucoup.

I. Création d'une exploration et gestion des points d'intérêt

> La syntaxe des définitions de fonctions choisie pour cet énoncé est une possibilité offerte par Python pour rendre le code plus explicite (les « annotations »). Bien que peu d'élèves l'aient déjà vu, son emploi n'a pas dû être un problème.

I.A.1.a Dans la fonction `générer_PI`, on peut choisir de manipuler une liste et d'utiliser la méthode `append` pour ajouter chaque point avant une transformation finale en tableau `numpy`, puisque celle-ci est imposée par l'énoncé :

```
def générer_PI(n:int, cmax:int) -> np.ndarray:
    PI = []
    while len(PI) < n:
        x = random.randrange(0,cmax+1)
        y = random.randrange(0,cmax+1)
        if [x,y] not in PI:
            PI.append([x,y])
    return np.array(PI)
```

> La difficulté de cette question réside en grande partie dans le besoin de ne garder que des points distincts. La réflexion pour obtenir une liste de n points aléatoire est très courte, mais celle pour éliminer les doublons l'est beaucoup moins, notamment parce qu'un grand nombre de possibilités existent. Si l'on ajoute à cela le retour sous forme de tableau `numpy` exigé, difficulté supplémentaire pour beaucoup de candidats, alors que la liste de listes est suffisante, cette question est plutôt difficile pour un début d'épreuve.
>
> Il est aussi possible d'utiliser un tableau `numpy` depuis le début de l'algorithme, en l'initialisant à un tableau de `n` zéros puis en le remplissant ligne par ligne, en vérifiant que chaque couple tiré au sort n'est pas déjà présent :
>
> ```
> def générer_PI(n:int, cmax:int) -> np.ndarray:
>     PI = np.zeros((n,2),int) # n lignes, 2 colonnes
>     i = 0
>     while i < n:
>         x = random.randrange(0,cmax+1)
>         y = random.randrange(0,cmax+1)
>         if [x,y] not in PI[0:i]:
>             PI[i] = [x,y]; i = i+1
>     return PI
> ```
>
> Malheureusement, malgré ce qu'indique l'annexe de l'énoncé, la construction en `[x,y] in PI[0:i]` ne donne pas le résultat prévu avec un tableau `numpy` :
>
> ```
> >>> PI = np.array([[1,2],[3,4]])
> >>> [1,2] in PI # Normal
> True
> >>> [1,5] in PI # Pas normal !
> True
> ```
>
> ce qui n'est pas vraiment le résultat escompté... À n'en pas douter, la fonction proposée plus haut devrait être acceptée par les correcteurs, mais ce code ne fonctionne pas en pratique. Une des possibilités pour résoudre ce problème, très au-delà de ce que l'on peut attendre d'un candidat, est
>
> ```
> any(np.equal([x,y],PI[0:i]).all(1))
> ```

qui répond correctement sur notre exemple :

```
>>> any(np.equal([1,2],PI).all(1))
True
>>> any(np.equal([1,5],PI).all(1))
False
```

I.A.1.b Les deux nombres doivent **être positifs** et aussi permettre d'obtenir `n` points distincts sur les `cmax+1`, c'est-à-dire

$$\boxed{\texttt{n} \leqslant (\texttt{cmax+1})^2}$$

Les coordonnées ainsi que `cmax` étant notées en millimètres, il est peu probable que `n` s'approche de cette valeur limite. On peut néanmoins noter dans ce cas que l'algorithme donné à la question précédente devient très inefficace. Il convient alors de le remplacer par un algorithme procédant par élimination d'un nombre complémentaire de points, choisis aléatoirement.

I.A.2 La fonction demandée doit calculer pour chaque couple de points (`i`,`j`) la distance, à l'aide du théorème de Pythagore. Le résultat est donc symétrique et il faut recopier chaque valeur du triangle inférieur gauche vers le triangle supérieur droit. On n'oublie pas d'ajouter la position actuelle en fin de ligne et de colonne.

```
def calculer_distances(PI:np.ndarray) -> np.ndarray:
    n = len(PI)
    pos = position_robot()
    distances = np.zeros((n+1,n+1))
    for i in range(n):
        for j in range(i):
            distances[i,j] = math.sqrt((PI[i,0]-PI[j,0])**2 \
                                  + (PI[i,1]-PI[j,1])**2)
            distances[j,i] = distances[i,j]
        distances[i,n] = math.sqrt((PI[i,0]-pos[0])**2 \
                              + (PI[i,1]-pos[1])**2)
        distances[n,i] = distances[i,n]
    return distances
```

On peut éviter une des racines en ajoutant la position actuelle directement à la liste des points d'intérêt avant calcul, mais il faut pour cela être un peu familier de la fonction `np.concatenate`, qui nécessite des listes ou des tableaux `numpy` en argument. Les crochets ici sont essentiels, ce qui n'est pas du tout évident...

```
def calculer_distances(PI:np.ndarray) -> np.ndarray:
    points = np.concatenate((PI,[position_robot()]))
    n = len(points)
    distances = np.zeros((n,n))
    for i in range(n):
        for j in range(i):
            distances[i,j] = math.sqrt(                        \
                                (points[i,0]-points[j,0])**2 \
                              + (points[i,1]-points[j,1])**2)
            distances[j,i] = distances[i,j]
    return distances
```

Centrale Informatique optionnelle MP 2017 Corrigé

Ce corrigé est proposé par William Aufort (ENS Lyon) ; il a été relu par Martin Guy (ENS Lyon) et Benjamin Monmege (enseignant-chercheur à l'université).

Cette épreuve est consacrée aux aspects théorique et algorithmique des mots synchronisants associés à une machine. Une machine se comporte comme un automate fini, hormis le fait qu'on ignore les états initial et finals. La notion de mot synchronisant est importante en théorie des automates et apparaît la première fois dans un problème naturel : comment réinitialiser un système ayant un nombre fini d'états sans savoir dans quel état il se trouve ?

Le sujet comporte quatre parties, dont les trois premières sont indépendantes.

- La première partie étudie quelques propriétés générales des mots synchronisants, notamment le fait que l'ensemble des mots synchronisants d'une machine forme un langage reconnaissable. Il y est question à la fois d'étude d'exemples, de questions d'implémentation plutôt faciles, et de questions utilisant des raisonnements proches de ceux sur les automates.
- La deuxième partie, purement algorithmique, a pour objectif l'implémentation d'un algorithme qui détermine l'ensemble des sommets s d'un graphe d'automate accessibles à partir d'un ensemble d'états E donné, ainsi qu'un moyen d'obtenir un plus court chemin de E à s. Tout d'abord, on implémente une structure de file à l'aide d'un tableau circulaire. Puis on étudie un algorithme dont le principe est de parcourir en largeur le graphe, en mettant à jour des structures supplémentaires permettant de retrouver les plus courts chemins. Cette partie comporte beaucoup de questions d'implémentation et de questions relatives à la correction, la terminaison et la complexité des algorithmes.
- Dans la troisième partie, on construit à partir d'une formule booléenne quelconque une machine dont on montre qu'elle admet un mot synchronisant d'une certaine longueur si et seulement si la formule est satisfiable. Cette partie mêle donc des raisonnements de théorie des automates et de logique.
- Enfin, la quatrième partie permet de résoudre le problème de l'existence d'un mot synchronisant en utilisant la partie II. Elle contient des questions plus avancées sur les automates et des questions d'implémentation.

Le sujet est très bien écrit et permet de manipuler toutes les notions du nouveau programme d'informatique optionnelle (automates, graphes, algorithmique et calcul propositionnel). Cela en fait un sujet très intéressant pour la préparation aux épreuves d'informatique.

Indications

I.B Raisonner sur la parité de l'état $q.m$.

I.E Il suffit de vérifier que l'état $q.m$ est le même pour tout état q.

I.F Si u est synchronisant, considérer deux « calculs » partant d'états différents de la machine sur l'entrée u.

I.G.1 Reformuler la propriété « admettre un mot synchronisant » en utilisant $\widehat{\delta}$.

I.G.2 Construire un automate qui reconnaît LS(M) à partir de la machine des parties, en utilisant la propriété établie à la question I.G.1.

I.G.3 « Simplifier » l'automate construit avant d'en déterminer le langage.

I.H On peut détecter qu'un chemin passe par q_0 dans une machine où q_0 est un état puits.

II.A Attention à la gestion des indices de début et de fin de file.

II.B Montrer qu'un sommet est inséré au plus une fois dans la file.

II.C La complexité de l'algorithme dépend du nombre de sommets insérés dans la file et du nombre d'arcs parcourus.

II.D Montrer le résultat par récurrence sur le nombre de tours de boucle effectués.

II.E.1 Observer que quand l'algorithme change le contenu de D[s], cela provient de la découverte d'un chemin de E à s d'une certaine longueur.

II.E.2 Raisonner par récurrence sur la valeur de d_s. On pourra raisonner par l'absurde et obtenir une contradiction en utilisant le prédécesseur de s dans un plus court chemin de E à s, ainsi que le sommet défilé juste avant l'insertion de s dans la file.

II.G Utiliser la question II.E.2 pour construire le mot de longueur minimale de façon récursive.

III.A Suivre la définition de la machine donnée au début de la partie. Dans une telle machine, une transition depuis l'état $q_{i,j}$ correspond à la recherche de la variable x_j dans la clause i.

III.C Remarquer que les états accessibles à partir de $q_{i,j}$ en une transition sont $q_{i,j+1}$ (s'il existe) et f, puis généraliser cette remarque.

III.D Utiliser le lien entre l'appartenance d'un littéral dans une clause et les transitions de la machine.

III.E Raisonner comme à la question III.D. Penser à compléter le mot synchronisant de la machine pour obtenir une distribution de vérité totale.

IV.A.1 Montrer que cette suite est décroissante et calculer $|\mathrm{Q}.u_r|$.

IV.A.2 Pour l'implication réciproque, la propriété dit qu'on peut synchroniser deux états quelconques. On peut chercher ensuite à synchroniser trois états, puis formuler une récurrence pour synchroniser tous les états.

IV.C Les fonctions `nb_to_set` et `set_to_nb` permettent de passer entre les représentations de $\overline{q}$ par un entier et par un ensemble d'états.

IV.D Attention au nombre d'états du graphe retourné $\widetilde{\mathrm{G}}_{\mathrm{R}}$.

IV.E Reformuler la caractérisation de la question IV.A.2 en terme d'accessibilité dans le graphe $\widetilde{\mathrm{G}}_{\mathrm{R}}$.

IV.F Implémenter la méthode proposée à la question IV.E.

I. Considérations générales

I.A Soit M une machine ayant un seul état, que l'on note q. Pour tout mot m de Σ^*, on a nécessairement $q.m = q$, donc m est un mot synchronisant. Par conséquent,

$$\boxed{\text{Tout mot est synchronisant pour une machine à un état.}}$$

> Il faut bien comprendre qu'un mot m est synchronisant si, en partant de n'importe quel état de la machine et en lisant m, on tombe toujours sur le même état.

I.B Remarquons qu'une transition dans la machine M_1 change la parité de l'état courant. Pour tout $m \in \Sigma^*$, $1.m$ et $2.m$ ont alors des parités opposées. Or, si m est un mot synchronisant, $1.m = 2.m$, ce qui est impossible. Finalement,

$$\boxed{\mathrm{M}_1 \text{ n'admet aucun mot synchronisant.}}$$

I.C Considérons le mot de trois lettres bcb. On a

$$1.bcb = 1.cb = 4.b = \delta(4, b) = 4$$

De même,

$$\begin{aligned} 2.bcb &= 1.cb = 4 \\ 3.bcb &= 4.cb = 3.b = \delta(4, b) = 4 \\ 4.bcb &= 4.cb = 4 \end{aligned}$$

ce qui prouve que $\boxed{bcb \text{ est un mot synchronisant pour } \mathrm{M}_2.}$

I.D Pour implémenter la fonction `delta_etoile`, on suit la définition récursive de δ^* rappelée dans l'énoncé : si le mot est vide, on renvoie l'état en entrée, sinon on fait une étape de calcul avec δ, et le reste du calcul est traité par appel récursif.

```
let rec delta_etoile machine etat mot = match mot with
  | [] -> etat
  | x::u -> delta_etoile machine (machine.delta etat x) u;;
```

I.E On utilise directement la définition d'un mot synchronisant : on veut vérifier, étant donné un mot m, s'il existe un état q_0 tel que pour tout état q, $q.m = q_0$. On trouve cet état q_0 en partant d'un état quelconque de la machine, par exemple 0, et en lisant m, soit $q_0 = 0.m$. Il reste à vérifier la propriété pour tous les autres états à l'aide d'une fonction récursive.

```
let est_synchronisant machine mot =
  let etat_zero = delta_etoile machine 0 mot in
  let rec verifie_etat machine mot etat =
    if etat = 0 then true
    else if (delta_etoile machine etat mot <> etat_zero) then false
    else verifie_etat machine mot (etat-1)
  in verifie_etat machine mot (machine.n_etats-1);;
```

On pourrait aussi utiliser une boucle `while` décroissante indicée par l'état à vérifier à l'aide d'une référence. On utilise une autre référence pour détecter si un état ne vérifie pas la propriété, auquel cas on peut sortir de la boucle directement. La boucle `while` simule alors la fonction auxiliaire récursive précédente.

```
let est_synchronisant_imperatif machine mot =
  let etat_zero = delta_etoile machine 0 mot
  and etat = ref (machine.n_etats-1)
  and continue = ref true in
  while (!continue && !etat > 0) do
    if (delta_etoile machine !etat mot <> etat_zero)
    then continue := false
    else etat := !etat-1
  done;
  !continue;;
```

I.F Soit M une machine ayant un mot synchronisant $u = u_1 \cdots u_n$. Soient q_0 et q'_0 deux états de M distincts (c'est possible car M a au moins deux états). Notons

$$\forall i \in \{1, \ldots, n\} \qquad q_i = q_0.(u_1 \cdots u_i) \qquad \text{et} \qquad q'_i = q'_0.(u_1 \cdots u_i)$$

les états successivement rencontrés en lisant le mot u à partir des états q_0 et q'_0 respectivement. Soit i le plus grand entier inférieur ou égal à n tel que $q_i \neq q'_i$. Comme u est synchronisant, $q_n = q_0.u = q'_0.u = q'_n$. De plus, $q_0 \neq q'_0$. Ceci prouve que i est bien défini. On pose enfin $q = q_i$, $q' = q'_i$ et $x = u_{i+1}$. Par définition $q \neq q'$, et $q.x = q_i.u_{i+1} = q_{i+1} = q'_{i+1} = q'.x$. Ainsi,

Une condition nécessaire d'existence d'un mot synchronisant est donnée par

$$\boxed{\exists\, x \in \Sigma \quad \exists\, q \in \mathrm{Q} \quad \exists\, q' \in \mathrm{Q} \qquad q' \neq q \quad \text{et} \quad q.x = q'.x}$$

De manière informelle, ce résultat dit que si une machine admet un mot synchronisant, alors il existe une lettre qui synchronise deux états différents de la machine. La question IV.A.2 montre un résultat similaire mais sous forme de condition nécessaire et suffisante.

I.G.1 De la même manière que pour δ, on peut étendre la fonction de transition $\widehat{\delta}$ en une fonction de $\widehat{\mathrm{Q}} \times \Sigma^*$, que l'on note $\widehat{\delta}^*$. M admet un mot synchronisant si et seulement s'il existe un mot $u \in \Sigma^*$ et un état $q_0 \in \mathrm{Q}$ tel que pour tout état $q \in \mathrm{Q}$, $q.u = q_0$, ce qui équivaut à dire que, dans la machine $\widehat{\mathrm{M}}$, $\widehat{\delta}^*(\mathrm{Q}, u) = \{q_0\}$. Autrement dit,

M admet un mot synchronisant si et seulement si il existe dans $\widehat{\mathrm{M}}$ un état singleton $\{q_0\}$ accessible depuis l'état Q.

I.G.2 Pour prouver qu'un langage est reconnaissable, une méthode générale consiste à exhiber un automate fini qui reconnaît ce langage. Cette méthode semble particulièrement intéressante ici car la notion de machine est très proche de celle d'automate.

À une machine M, on associe l'automate déterministe $\mathcal{A}_{\mathrm{M}} = (\widehat{\mathrm{Q}}, \Sigma, \mathrm{I}, \mathrm{F}, \widehat{\delta})$ où $\mathrm{I} = \mathrm{Q}$ est l'état initial de l'automate et $\mathrm{F} = \{\{q\}, q \in \mathrm{Q}\}$, constitué des singletons, est l'ensemble des états finals de l'automate. Un mot $u \in \Sigma^*$ est reconnu par l'automate $\mathcal{A}_{\mathrm{M}}$

Mines Maths 1 MP 2017 — Corrigé

Ce corrigé est proposé par Rémi Pellerin (ENS Lyon) ; il a été relu par Hervé Diet (professeur agrégé) et Antoine Sihrener (professeur en CPGE).

Ce sujet propose l'étude de deux endomorphismes u et v sur des espaces fonctionnels. Chaque partie s'appuie sur les précédentes. La difficulté est croissante au sein d'une partie, mais pas d'une partie à l'autre.

- La première partie établit quelques résultats préliminaires qui seront utiles tout au long du sujet. Elle demande également de justifier que les objets introduits sont bien définis et se termine sur une rapide étude des intégrales de Wallis. Il vaut mieux ne pas se tromper à la question 5 tant son résultat sera réutilisé par la suite.
- La deuxième partie a pour but l'étude de la continuité de ces endomorphismes pour certaines normes en dimension infinie. Un résultat de densité, fondamental pour la suite, y est prouvé.
- La troisième partie démontre l'inversibilité des endomorphismes considérés. Elle se termine par deux applications qui font appel au calcul intégral et à quelques formules de trigonométrie circulaire et hyperbolique en bordure du programme.
- La quatrième partie propose une étude des éléments propres de u et v et se termine par un peu de topologie.

Ce problème de longueur raisonnable permet de bien revoir les particularités de la dimension infinie. On y trouve des normes non équivalentes, de la réduction en dimension infinie et des applications dont la continuité dépend de la norme considérée. Il permet de réviser les espaces vectoriels normés, la topologie, les séries entières, les théorèmes de permutation somme/intégrale et les intégrales à paramètres. Dans l'ensemble, le sujet n'est ni très difficile ni très long, mais il exige d'avoir les idées claires sur les fondamentaux. Cependant, beaucoup de questions sont ouvertes et certains résultats utiles pour la suite ne sont pas donnés. Une lecture intégrale du sujet avant de commencer à écrire était (comme toujours) utile car certaines réponses pouvaient être déduites d'autres questions.

Indications

Partie A

1 Attention à bien comprendre ce que signifie « au voisinage de zéro » !

2 Montrer que $u(f)$ est de classe $\mathscr{C}^n$ pour tout $n \in \mathbb{N}$.
Remarquer que $v(f)(x) = f(0) + x\,\dfrac{\pi}{2}\,u(f')$. Attention à ne pas oublier de vérifier que les intégrandes sont bien définies sur l'intervalle d'intégration.

4 Faire une intégration par parties en partant de $\sin^{n+2}(t) = (1 - \cos^2(t))\sin^n(t)$.

5 Commencer par établir que $\mathrm{W}_{n+1} \sim \mathrm{W}_n$.

Partie B

6 Penser à un théorème du programme sur la continuité des applications linéaires sur des espaces vectoriels normés.

7 Considérer $f(x) = (x/a)^n$.

9 Penser au théorème d'approximation de Stone-Weierstrass.

Partie C

11 Montrer que v est injective.

12 Montrer que u est continue de $(\mathcal{E}, \mathrm{N})$ dans $(\mathcal{E}, \mathrm{N})$.

13 La fonction Argsh n'est pas au programme. On donne les propriétés utiles à la résolution de cette question en remarque. On notera en particulier que

$$\mathrm{Argsh}\,'(x) = 1/\sqrt{1+x^2}$$

14 Commencer par établir certaines implications puis utiliser que $u \circ v = v \circ u = \mathrm{id}_{\mathcal{E}}$.

Partie D

16 Penser à la permutation série intégrale.

17 Montrer par l'absurde que $(m_n)_{n\in\mathbb{N}}$ est nulle à partir d'un certain rang.

18 Raisonner par analyse-synthèse.

19 Montrer qu'une suite de valeurs propres de v qui converge est stationnaire.

A. Préliminaires

$\boxed{1}$ Montrons que $\mathcal{P}$ est un sous-espace vectoriel de $\mathcal{E}$. Considérons la fonction suivante entre les espaces vectoriels $\mathbb{C}[\mathrm{X}]$ et $\mathcal{E}$:

$$\phi\colon \begin{cases} \mathbb{C}[\mathrm{X}] \longrightarrow \mathcal{E} \\ \ \mathrm{P} \quad \longmapsto (x \mapsto \mathrm{P}(x)) \end{cases}$$

Vérifions que ϕ est linéaire. Soient $\mathrm{P}, \mathrm{Q} \in \mathbb{C}[\mathrm{X}]$ et $\lambda \in \mathbb{C}$, alors

$$\begin{aligned} \phi(\lambda\mathrm{P} + \mathrm{Q}) &= (x \mapsto (\lambda\mathrm{P} + \mathrm{Q})(x)) \\ &= \lambda\,(x \mapsto \mathrm{P}(x)) + (x \mapsto \mathrm{Q}(x)) \\ \phi(\lambda\mathrm{P} + \mathrm{Q}) &= \lambda\phi(\mathrm{P}) + \phi(\mathrm{Q}) \end{aligned}$$

De plus, $\mathrm{Im}\ \phi = \mathcal{P}$ donc

$$\boxed{\mathcal{P} \text{ est un sous-espace vectoriel de } \mathcal{E}.}$$

> Un morphisme de ce type est souvent appelé « morphisme d'évaluation ».

Montrons que $\mathcal{D}$ est un sous-espace vectoriel de $\mathcal{E}$. Par définition, $\mathcal{D} \subseteq \mathcal{E}$. En outre, $\mathcal{D}$ n'est pas vide car $0_{\mathbb{R}^{\mathrm{I}}} \in \mathcal{D}$. Soient $\lambda \in \mathbb{C}$, $f \in \mathcal{D}$ et $g \in \mathcal{D}$. Les applications f et g sont développables en séries entières au voisinage de zéro. Par définition, il existe un intervalle de la forme $[\,-\varepsilon\,;\varepsilon\,]$ avec $\varepsilon > 0$, un intervalle de la forme $[\,-\alpha\,;\alpha\,]$ avec $\alpha > 0$ et deux suites $(a_n)_{n\in\mathbb{N}} \in \mathbb{C}^{\mathbb{N}}$ et $(b_n)_{n\in\mathbb{N}} \in \mathbb{C}^{\mathbb{N}}$ telles que

$$\forall x \in [\,-\varepsilon\,;\varepsilon\,] \qquad f(x) = \sum_{n=0}^{+\infty} a_n x^n$$

et

$$\forall x \in [\,-\alpha\,;\alpha\,] \qquad g(x) = \sum_{n=0}^{+\infty} b_n x^n$$

En considérant $\mu = \mathrm{Min}\,(\varepsilon, \alpha)$, on a

$$\forall x \in [\,-\mu\,;\mu\,] \qquad \begin{aligned} (\lambda f + g)(x) &= \lambda f(x) + g(x) \\ &= \sum_{n=0}^{+\infty} \lambda a_n x^n + \sum_{n=0}^{+\infty} b_n x^n \\ (\lambda f + g)(x) &= \sum_{n=0}^{+\infty} (\lambda a_n + b_n)\, x^n \end{aligned}$$

par linéarité du développement en série entière. Ceci prouve que $\lambda f + g$ admet un développement en série entière sur un voisinage de zéro (car $\mu > 0$) et par conséquent $\lambda f + g \in \mathcal{D}$. Finalement,

$$\boxed{\mathcal{D} \text{ est un sous-espace vectoriel de } \mathcal{E}.}$$

$\boxed{2}$ Vérifions d'abord que u et v sont bien définies sur $\mathcal{E}$. Soit $f \in \mathcal{E}$. Il s'agit tout d'abord de vérifier que pour tout $t \in [\,0\,;\pi/2\,]$, et pour tout $x \in \mathrm{I}$, la quantité $x\sin^n(t)$ reste bien dans I. C'est le cas car

$$\forall x \in [\,-a\,;a\,] \quad \forall t \in \left[\,0\,;\frac{\pi}{2}\,\right] \qquad -a \leqslant -a\sin(t) \leqslant x\sin(t) \leqslant a\sin^n(t) \leqslant a$$

Ainsi, les intégrandes sont bien définies sur $[\,0\,;\pi/2\,]$. De plus, celles-ci sont continues sur le segment $[\,0\,;\pi/2\,]$ ce qui prouve que les intégrales existent bien. Par conséquent, pour tout $f \in \mathcal{E}$, $u(f)$ et $v(f)$ sont bien définies. On vérifie également que u et v sont bien linéaires par linéarité de la dérivation, de l'intégrale et de l'évaluation en 0.

Montrons désormais que u et v sont à valeurs dans $\mathcal{E}$. Pour ce faire, il suffit d'établir que pour tout $f \in \mathcal{E}$, $u(f)$ et $v(f)$ sont de classe $\mathscr{C}^\infty$. Posons

$$g(x,t) = f(x\sin(t))$$

Soit $n \in \mathbb{N}$, montrons que $u(f)$ est de classe $\mathscr{C}^n$.

- Pour tout $t \in \mathrm{I}$, l'application $g(\cdot,t)$ est de classe $\mathscr{C}^\infty$ sur I par les théorèmes généraux. On vérifie que
$$\forall k \in \mathbb{N} \qquad \frac{\partial^k g}{\partial x^k}(x,t) = \sin^k(t) f^{(k)}(x\sin(t))$$
- $\forall (x,t) \in \mathrm{I} \times [\,0\,;\pi/2\,] \qquad \left|\dfrac{\partial^n g}{\partial x^n}(x,t)\right| = \left|\sin^n(t) f^{(n)}(x\sin^n(t))\right| \leqslant \|f^{(n)}\|_\infty$
- L'application $\big(t \mapsto \|f^{(n)}\|_\infty\big)$ existe bien par continuité de $f^{(n)}$ sur le compact I. Elle est indépendante de x et intégrable sur I.
- Pour tout $k \in [\![\,0\,;n-1\,]\!]$, tout $x \in \mathrm{I}$, l'application $\dfrac{\partial^k g}{\partial x^k}(x,\cdot)$ est intégrable et continue par morceaux sur $[\,0\,;\pi/2\,]$ (car la domination précédente est valable pour tout n).

> La dernière hypothèse de continuité par morceaux du théorème n'est en fait pas nécessaire. En effet, la théorie de l'intégration de Lebesgue permet de s'en affranchir. Ce n'est cependant pas le cas de la théorie de Riemann en vigueur au programme. Attention à ne pas l'oublier !

Ainsi, la fonction $u(f)$ est de classe $\mathscr{C}^n$ sur I et sa dérivée n^{e} est

$$u(f)^{(n)}(x) = \frac{2}{\pi}\int_0^{\frac{\pi}{2}} \sin^n(t) f^{(n)}(x\sin(t))\,\mathrm{d}t$$

Montrons maintenant que $v(f)$ est elle aussi de classe $\mathscr{C}^n$ pour tout $n \in \mathbb{N}$. Commençons par remarquer que

$$v(f)(x) = f(0) + x\,\frac{\pi}{2}\,u(f')$$

Comme $f' \in \mathcal{E}$, $u(f')$ est de classe $\mathscr{C}^n$ d'après ce qui précède. Il vient que $v(f)$ est de classe $\mathscr{C}^n$ d'après les théorèmes généraux. Ainsi, $u(f)$ et $v(f)$ sont de classe $\mathscr{C}^\infty$. De plus, u et v sont linéaires donc

$$\boxed{u \text{ et } v \text{ sont des endomorphismes de } \mathcal{E}.}$$

3 Soit $f \in \mathcal{P}$. Montrons que $u(f)$ et $v(f)$ appartiennent à $\mathcal{P}$. Par définition de f, il existe $\mathrm{N} \in \mathbb{N}$ et $(a_0,\cdots,a_n) \in \mathbb{C}^{n+1}$ tels que

$$\forall x \in \mathrm{I} \qquad f(x) = \sum_{n=0}^{\mathrm{N}} a_n x^n$$

Calculons d'abord, pour tout $x \in \mathrm{I}$,

$$\begin{aligned} u(f)(x) &= \frac{2}{\pi}\int_0^{\frac{\pi}{2}} \sum_{n=0}^{\mathrm{N}} a_n x^n \sin^n(t)\,\mathrm{d}t \\ &= \sum_{n=0}^{\mathrm{N}} \frac{2}{\pi}\,a_n \left(\int_0^{\frac{\pi}{2}} \sin^n(t)\,\mathrm{d}t\right) x^n \\ u(f)(x) &= \sum_{n=0}^{\mathrm{N}} \frac{2}{\pi}\,a_n \mathrm{W}_n x^n \end{aligned}$$

Mines Maths 2 MP 2017 — Corrigé

Ce corrigé est proposé par Sophie Rainero (professeur en CPGE) ; il a été relu par Robin Michaud (ENS Lyon) et Benoit Chevalier (ENS Ulm).

Ce sujet étudie des sous-groupes compacts du groupe linéaire $\mathrm{GL}(\mathrm{E})$ d'un espace vectoriel euclidien E. Il se compose de cinq parties, les trois premières étant indépendantes.

- Dans la partie **A**, on établit des résultats préliminaires très classiques sur les matrices symétriques réelles.
- La partie **B** est constituée de trois questions permettant d'obtenir des résultats préliminaires sans lien entre eux : la compacité de l'enveloppe convexe d'une partie compacte, une propriété des endomorphismes qui conservent l'orthogonalité et enfin la compacité du groupe $\mathrm{O}_n(\mathbb{R})$ des matrices orthogonales de taille n.
- Dans la partie **C**, on établit un résultat important sur les parties compactes de E : de tout recouvrement d'un compact de E par des parties ouvertes de E, on peut extraire un sous-recouvrement fini.
- La partie **D** s'intéresse à un théorème du point fixe, le théorème de Markov-Kakutani : soient G un sous-groupe compact de $\mathrm{GL}(\mathrm{E})$ et K un compact convexe de E stable par tous les éléments de G, il existe alors a appartenant à K qui est point fixe de tous les éléments de G.
- Dans la partie **E**, on établit que $\mathrm{O}_n(\mathbb{R})$ est un sous-groupe compact maximal du groupe $\mathrm{GL}_n(\mathbb{R})$ des matrices réelles inversibles de taille n, c'est-à-dire que si K est un sous-groupe compact de $\mathrm{GL}_n(\mathbb{R})$ qui contient $\mathrm{O}_n(\mathbb{R})$, alors K est égal à $\mathrm{O}_n(\mathbb{R})$.

Ce sujet aborde plusieurs parties importantes du programme de MP : la topologie, en particulier la convexité et la compacité, et l'algèbre euclidienne, notamment les matrices symétriques réelles et les matrices orthogonales. D'une difficulté et d'une longueur raisonnables, il est bien guidé : de nombreuses questions comportent des indications ou sont découpées de façon à orienter le candidat. Globalement, c'est un sujet très intéressant qui constitue un joli problème de révision en topologie et en algèbre.

Indications

Partie A

1 Démontrer l'équivalence par double implication et utiliser le théorème spectral pour le sens réciproque.

2 Utiliser à nouveau le théorème spectral.

Partie B

4 Démontrer que ϕ est continu et que $\mathscr{H} \times \mathrm{K}^{n+1}$ est compact.

6 Se servir du fait qu'en dimension finie, les parties compactes sont les parties fermées et bornées.

Partie C

7 Procéder par l'absurde.

8 Lorsque K est non vide, procéder par l'absurde et construire par récurrence une suite d'éléments de K qui vérifie les hypothèses de la question 7.

9 Suivre l'indication de l'énoncé puis faire appel à la question 8.

10 Introduire les complémentaires de F_i pour $i \in \mathrm{I}$ afin d'appliquer le résultat de la question 9.

Partie D

12 Pour le premier point, se servir du fait que G est un sous-groupe de GL(E). Pour le second, utiliser l'indication de l'énoncé avec le vecteur $z = x + y$ pour le sens direct.

15 Le premier point de la question 12 aide à démontrer la première égalité. Pour la seconde, se servir de la première et utiliser deux fois l'inégalité triangulaire.

16 Appliquer le résultat de la question 15 et le second point de la question 12 lorsque a est non nul.

17 Distinguer encore deux cas selon que a est nul ou pas.

18 Caractériser l'existence d'un point fixe commun à tous les éléments de G à l'aide des ensembles $\mathrm{F}_u = \mathrm{Ker}\,(u - \mathrm{Id}_{\mathrm{E}}) \cap \mathrm{K}$, pour $u \in \mathrm{G}$, puis appliquer la question 10.

Partie E

19 Utiliser la caractérisation des sous-groupes et écrire H comme l'image d'un compact par une application continue.

20 Pour Δ, appliquer la question 2. Pour K, se servir des questions 3, 4 et 19.

21 Appliquer la question 18 en vérifiant soigneusement toutes les hypothèses nécessaires à l'aide des questions 19 et 20. Pour l'existence de N, se souvenir de la question 2.

22 Que peut-on dire d'un endomorphisme qui est à la fois une symétrie et un endomorphisme orthogonal ? Pour montrer que g préserve l'orthogonalité de deux vecteurs x et y, introduire l'hyperplan $\mathrm{P} = \mathrm{Vect}\,(x)^{\perp}$ lorsque x est non nul. Appliquer enfin le résultat de la question 5.

A. Préliminaires sur les matrices symétriques

1 Soit S appartenant à $S_n(\mathbb{R})$.

Supposons dans un premier temps que S est définie positive. Soient λ une valeur propre de S et $X \in \mathscr{M}_{n,1}(\mathbb{R})$ un vecteur colonne propre associé, que l'on note $X = {}^t(x_1, \ldots, x_n)$. Alors X est non nul donc, par hypothèse, ${}^t X\,SX > 0$. Ainsi,

$$\lambda \sum_{i=1}^{n} {x_i}^2 = \lambda\, {}^t X\, X = {}^t X(\lambda X) = {}^t X\, SX > 0$$

Or, comme X est non nul, ${x_1}^2 + \cdots + {x_n}^2$ est une somme de termes positifs non tous nuls donc elle est strictement positive. Par conséquent, $\lambda > 0$. Finalement, si S est une matrice symétrique définie positive, alors ses valeurs propres sont des réels strictement positifs.

Supposons maintenant que $\mathrm{Sp}(S) \subset \mathbb{R}_+^*$. La matrice S étant symétrique réelle, le cours assure qu'il existe une matrice $P \in O_n(\mathbb{R})$ et une matrice diagonale réelle $D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, dont les coefficients diagonaux sont les valeurs propres de S, telles que $S = PDP^{-1}$. Soit $X = {}^t(x_1, \ldots, x_n)$ une matrice colonne réelle non nulle. Comme P est orthogonale, $P^{-1} = {}^t P$ et

$${}^t X\, SX = {}^t X(PDP^{-1})X = {}^t X(PD\,{}^t P)X$$

d'où
$${}^t X\, SX = {}^t({}^t P\, X)D({}^t P\, X)$$

En notant $Y = {}^t P\, X = {}^t(y_1, \ldots, y_n)$, on obtient alors

$${}^t X\, SX = {}^t Y\, DY = \sum_{i=1}^{n} \lambda_i\, {y_i}^2$$

Or, pour tout $i \in [\![1\,; n]\!]$, $\lambda_i > 0$ et ${y_i}^2 \geqslant 0$. De plus, Y est non nulle car ${}^t P$ est inversible et X est non nulle, donc il s'agit d'une somme de termes positifs dont au moins un des termes est strictement positif. Par conséquent, ${}^t X\, SX > 0$. On a démontré que la matrice S est définie positive. Finalement,

$$\boxed{\text{S est définie positive} \iff \mathrm{Sp}(S) \subset \mathbb{R}_+^*}$$

> Cette caractérisation des matrices symétriques définies positives est une question très classique, que l'on retrouve dans de nombreux sujets, sous forme matricielle comme ici, mais aussi pour des endomorphismes symétriques.

2 Soit $S \in S_n^{++}(\mathbb{R})$. Comme S est symétrique réelle, il existe une matrice $P \in O_n(\mathbb{R})$ et une matrice diagonale réelle $D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, dont les coefficients diagonaux sont des réels strictement positifs (puisque ce sont les valeurs propres de S), telles que $S = PDP^{-1}$. Posons alors

$$\Delta = \mathrm{diag}(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}) \qquad \text{puis} \qquad R = P\Delta P^{-1}$$

La matrice Δ est diagonale à coefficients diagonaux tous strictement positifs, c'est donc une matrice inversible et par suite la matrice R est inversible, la matrice P étant aussi inversible. De plus,

$${}^t R\, R = {}^t(P\Delta P^{-1})(P\Delta P^{-1}) = {}^t P^{-1}\, {}^t\Delta\, {}^t P\, P\Delta P^{-1} = P\Delta^2 P^{-1}$$

car P est orthogonale et Δ est diagonale. Par conséquent,

$${}^t R\, R = PDP^{-1} = S$$

Ainsi, $\boxed{\text{Pour tout } S \in S_n^{++}(\mathbb{R}), \text{ il existe } R \in GL_n(\mathbb{R}) \text{ telle que } S = {}^t R\, R.}$

> Comme la première question, cette décomposition d'une matrice symétrique définie positive sous la forme d'un produit d'une matrice inversible et de sa transposée est extrêmement classique. Notons au passage que
>
> $$ {}^t\mathrm{R} = {}^t(\mathrm{P}\Delta\mathrm{P}^{-1}) = {}^t\mathrm{P}^{-1}\, {}^t\Delta\, {}^t\mathrm{P} = \mathrm{P}\Delta\mathrm{P}^{-1} = \mathrm{R} $$
>
> On a donc construit une matrice symétrique.

Soit maintenant R une matrice réelle inversible de taille n, posons $\mathrm{S} = {}^t\mathrm{R}\,\mathrm{R}$ et vérifions que $\mathrm{S} \in \mathrm{S}_n^{++}(\mathbb{R})$. Déjà,

$$ {}^t\mathrm{S} = {}^t({}^t\mathrm{R}\,\mathrm{R}) = {}^t\mathrm{R}\,\mathrm{R} = \mathrm{S} $$

donc S est symétrique. De plus, pour tout vecteur colonne non nul $\mathrm{X} \in \mathscr{M}_{n,1}(\mathbb{R})$,

$$ {}^t\mathrm{X}\,\mathrm{S}\mathrm{X} = {}^t\mathrm{X}\, {}^t\mathrm{R}\,\mathrm{R}\mathrm{X} = {}^t(\mathrm{R}\mathrm{X})(\mathrm{R}\mathrm{X}) = \sum_{i=1}^{n} {y_i}^2 $$

où l'on a noté $\mathrm{RX} = {}^t(y_1, \ldots, y_n)$. En outre, la matrice R est inversible et X est non nul donc RX est non nul, d'où

$$ {}^t\mathrm{X}\,\mathrm{S}\mathrm{X} = \sum_{i=1}^{n} {y_i}^2 > 0 $$

On conclut que $\boxed{\text{Pour tout } \mathrm{R} \in \mathrm{GL}_n(\mathbb{R}),\ {}^t\mathrm{R}\,\mathrm{R} \in \mathrm{S}_n^{++}(\mathbb{R}).}$

$\boxed{3}$ Démontrons que l'ensemble $\mathrm{S}_n^{++}(\mathbb{R})$ est une partie convexe de $\mathscr{M}_n(\mathbb{R})$. Soient A et B dans $\mathrm{S}_n^{++}(\mathbb{R})$ et $t \in [\,0\,;1\,]$. On pose $\mathrm{C} = t\,\mathrm{A} + (1-t)\,\mathrm{B}$, montrons que C est symétrique définie positive. Notons déjà que si $t = 0$ ou $t = 1$, $\mathrm{C} = \mathrm{B}$ ou $\mathrm{C} = \mathrm{A}$ donc le résultat est immédiat. Supposons alors $t \in \,]\,0\,;1\,[$. La matrice C est symétrique en tant que combinaison linéaire de deux matrices symétriques. De plus, pour toute matrice colonne non nulle $\mathrm{X} \in \mathscr{M}_{n,1}(\mathbb{R})$,

$$ {}^t\mathrm{X}\,\mathrm{C}\mathrm{X} = {}^t\mathrm{X}(t\,\mathrm{A} + (1-t)\,\mathrm{B})\mathrm{X} = t\ {}^t\mathrm{X}\,\mathrm{A}\mathrm{X} + (1-t)\ {}^t\mathrm{X}\,\mathrm{B}\mathrm{X} $$

Or, comme $t \in \,]\,0\,;1\,[$, t et $1-t$ sont strictement positifs, de même que ${}^t\mathrm{X}\,\mathrm{A}\mathrm{X}$ et ${}^t\mathrm{X}\,\mathrm{B}\mathrm{X}$ puisque A et B sont définies positives. Donc, par produit et somme de réels strictement positifs,

$$ {}^t\mathrm{X}\,\mathrm{C}\mathrm{X} > 0 $$

On a ainsi prouvé que C appartient à $\mathrm{S}_n^{++}(\mathbb{R})$. Finalement, pour toutes matrices A et B dans $\mathrm{S}_n^{++}(\mathbb{R})$, le segment $[\,\mathrm{A}\,;\mathrm{B}\,]$, égal à $\{t\,\mathrm{A} + (1-t)\,\mathrm{B} \mid t \in [\,0\,;1\,]\}$, est inclus dans $\mathrm{S}_n^{++}(\mathbb{R})$. Par conséquent,

$$ \boxed{\text{L'ensemble } \mathrm{S}_n^{++}(\mathbb{R}) \text{ est convexe.}} $$

> La matrice C construite ci-dessus est symétrique définie positive et donc en particulier elle est inversible. On a ainsi obtenu une partie convexe de $\mathrm{GL}_n(\mathbb{R})$, bien que celui-ci ne soit pas convexe : par exemple, $\mathrm{I}_n \in \mathrm{GL}_n(\mathbb{R})$ et $-\mathrm{I}_n \in \mathrm{GL}_n(\mathbb{R})$, mais
>
> $$ \frac{1}{2}\mathrm{I}_n + \left(1 - \frac{1}{2}\right)(-\mathrm{I}_n) = 0_{\mathscr{M}_n(\mathbb{R})} \notin \mathrm{GL}_n(\mathbb{R}) $$

Mines Informatique commune MP 2017 — Corrigé

Ce corrigé est proposé par Virgile Andreani (ENS Ulm) ; il a été relu par Cyril Ravat (Professeur en CPGE) et Julien Dumont (Professeur en CPGE).

Ce sujet aborde la modélisation d'un croisement routier, représenté par deux files de voitures qui s'intersectent.

- Dans la première partie, il pose les bases de la modélisation avec le cas d'une file unique.
- Dans la deuxième, il invite à programmer la dynamique de la file en présence ou non d'un obstacle sur la voie, à l'aide d'une opération élémentaire, fournie par une fonction déjà écrite, qui consiste à faire avancer toute la file d'un coup.
- Dans une troisième partie qui ne comporte que deux questions, on s'intéresse enfin au cas du croisement, en simulant conjointement deux files couplées par une case commune.
- La quatrième partie, courte elle aussi, prépare la suivante en introduisant la notion de transition entre configurations non immédiatement successives.
- L'avant-dernière partie, la moins facile de l'épreuve, est consacrée au codage d'un algorithme de recherche en largeur sur les états du système ce qui permet de déterminer si une configuration est atteignable depuis une autre.
- Enfin, l'épreuve se termine par trois questions utilisant les bases de données.

La difficulté de ce problème est progressive, avec de nombreuses questions très accessibles en début et milieu d'épreuve, suivies d'autres plus délicates en fin d'épreuve, notamment dans la cinquième partie. Il est accessible dès la première année (sauf la question 18). L'ensemble du programme est couvert à l'exception de la simulation numérique.

Indications

Partie II

9 La fonction `avancer` peut servir d'opération élémentaire dans la fonction demandée, il faut s'en servir. On peut faire appel à la concaténation de listes au moyen de l'opérateur `+`.

10 Noter que la case m est inoccupée. Que cela implique-t-il sur les voitures à gauche de m ?

11 Étant donné que les voitures immédiatement à gauche de m sont bloquées, où sont les voitures qui ont la possibilité d'avancer ?

Partie III

12 Bien réfléchir à l'ordre dans lequel appeler les trois fonctions précédemment définies `avancer_debut`, `avancer_debut_bloque` et `avancer_fin`.

Partie V

17 Noter que la liste est triée : il suffit donc de comparer chaque valeur à la précédente pour déterminer si elle est nouvelle ou non.

20 Chercher dans le code où apparaissent les variables en question et les opérations dans lesquelles elles sont impliquées pour déterminer leur type.

21 Quel est le principal critère de choix d'un algorithme ?

24 Démontrer que le nombre d'éléments uniques contenus dans la liste `espace` est croissant et borné.

I. Préliminaires

1 Soit une case est vide, soit elle contient une voiture. On peut donc encoder ces deux états au moyen d'une variable binaire qui vaut `True` si une voiture est présente à cet endroit et `False` sinon. Pour représenter n cases, on utilise une liste de n de ces booléens.

2 On commence par créer une liste vide, puis on remplit les cases qui doivent l'être :

```
A = 11*[False]
A[0] = True
A[2] = True
A[3] = True
A[10] = True
```

> Il vaut mieux éviter ici de construire la liste directement avec `A = [True, False, True, True, etc.]` pour réduire les risques d'erreur.

3 Vu la manière dont on a encodé les états des cases, la fonction `occupe` ne renvoie rien d'autre que la valeur booléenne de la case demandée :

```
def occupe(L, i):
    return L[i]
```

> Il faut prendre soin à partir d'ici d'utiliser uniquement cette fonction pour vérifier l'état des cases et éviter d'indexer directement les listes. En effet, si on décidait un jour de changer de représentation pour les états des cases, il suffirait d'adapter cette fonction une fois au lieu de chercher dans le code tous les endroits où l'on indexe une liste.

4 Chaque case pouvant être dans deux états différents, et ce de manière indépendante des autres cases,

Le nombre de files différentes est le produit du nombre d'états possibles pour chacune des cases soit 2^n

.

5 Deux listes sont égales si et seulement si elles ont la même longueur et chacune de leurs cases égales deux à deux. Par conséquent :

```
def egal(L1, L2):
    if len(L1) != len(L2):
        return False
    for i in range(len(L1)):
        if L1[i] != L2[i]:
            return False
    return True
```

> On fait le test de longueur en premier pour s'assurer que les indices seront valides dans la boucle. À l'intérieur de celle-ci, on peut retourner `False` dès la première paire de valeurs différentes, inutile de poursuivre.
>
> Python permet de comparer directement deux listes avec l'opérateur `==`, mais écrire `L1 == L2` ici annule l'intérêt de la question et ne fait pas apparaître explicitement la complexité.

6 Si les listes sont de longueur différente, il n'y a pas de boucle et la fonction s'achève en temps constant. Le pire cas correspond à la situation où l'on itère la boucle un maximum de fois, c'est-à-dire quand les listes sont de même taille et que tous leurs éléments sont égaux deux à deux (sauf éventuellement les deux derniers). Dans ce cas,

La complexité est linéaire en la longueur commune des listes

7 Comme il était demandé à la question 5, cette fonction retourne un booléen.

II. Déplacement de voitures dans la file

8 Cette évolution consiste à faire avancer la file deux fois, d'abord sans ajouter de voiture au début, puis en en ajoutant une. On obtient donc `[True, False, True, False, True, True, False, False, False, False, False]`.

9 Cette étape partielle consiste à faire avancer la deuxième partie de la liste et à laisser la première inchangée. On peut réutiliser la fonction `avancer` pour la deuxième partie et recoller le résultat avec la première partie de la liste au moyen de l'opérateur de concaténation `+`.

```
def avancer_fin(L, m):
    return L[:m] + avancer(L[m:], False)
```

> Pour cette question comme pour les suivantes, l'énoncé incite à réutiliser la fonction `avancer`, qu'il définit mais n'utilise pas ailleurs. Une réponse acceptable sans utiliser cette fonction aurait pu être
>
> ```
> def avancer_fin(L, m):
>     return L[:m] + [False] + L[m:-1]
> ```
>
> ou encore, sans concaténation :
>
> ```
> def avancer_fin(L, m):
>     L1 = L[:m]
>     L1.append(False)
>     for i in range(m, len(L)-1):
>         L1.append(L[i])
>     return L1
> ```

10 La case m étant inoccupée, les voitures à gauche de m ne sont pas bloquées et peuvent toutes avancer. On peut donc là aussi réutiliser la fonction `avancer` sur la première partie de la liste et recoller le résultat avec la seconde partie.

```
def avancer_debut(L, b, m):
    return avancer(L[:m+1], b) + L[m+1:]
```

11 Les voitures bloquées immédiatement à la gauche de m ne bougent pas, de la même manière que les voitures à la droite de m. Toutes les voitures à la droite de la première case inoccupée à la gauche de m (si elle existe) voient donc leur position inchangée par `avancer_debut_bloque`, seules les voitures à la gauche de cette case

Mines Informatique optionnelle MP 2017 — Corrigé

Ce corrigé est proposé par Martin Guy (ENS Lyon) ; il a été relu par William Aufort (ENS Lyon) et Benjamin Monmege (enseignant-chercheur à l'université).

L'épreuve se compose de deux problèmes indépendants. Le premier porte sur les langages et les automates ; le second est consacré à l'algorithmique.

- Le premier problème se concentre sur les langages sur un alphabet unaire et sur les automates finis. Après l'observation et la construction de différents automates, on prouve que les langages rationnels unaires infinis contiennent nécessairement $\left\{a^{\alpha k+\beta} \mid k \in \mathbb{N}\right\}$ pour un certain α et un certain β. On en dérive enfin une propriété permettant de prouver que certains langages unaires ne sont pas rationnels. Ce premier problème permet de vérifier la maîtrise des bases sur les automates ainsi que de prouver des propriétés sur certains langages.
- Le second problème, plus conséquent, présente deux algorithmes permettant de calculer la puissance n d'un élément a (pouvant être un entier, un réel ou une matrice) avec le moins de multiplications possible. Il peut se séparer en plusieurs parties. Tout d'abord, les premières questions permettent de s'approprier la notion de suite pour l'obtention de la puissance n à partir d'exemples, puis se concentrent sur l'algorithme d'exponentiation rapide. Les questions suivantes présentent un second algorithme basé sur la décomposition binaire de la puissance n. Au passage, on montre qu'un algorithme optimal d'élévation à la puissance n effectue nécessairement entre $\lceil \log(n) \rceil$ et $2 \times \lfloor \log(n) \rfloor$ multiplications. On constate ensuite que les deux algorithmes étudiés ne sont pas optimaux puisqu'il existe des exemples pour lesquels ils ne renvoient pas la suite optimale de multiplications à effectuer. Les dernières questions se consacrent donc à la recherche d'une suite optimale via l'exploration exhaustive de l'ensemble des suites. Ce problème permet de vérifier les acquis en algorithmique mais aussi en programmation car tous les algorithmes étudiés doivent être implémentés.

Ce sujet comporte quelques questions pointues et il est assez long, mais plusieurs questions peuvent se faire rapidement, notamment celles qui demandent d'omettre la justification.

Indications

Partie I

1 Se concentrer sur les conditions pour que $\{\alpha k + \beta \mid k \in \mathbb{N}\}$ soit fini.

2 Trouver « à la main » quelques mots reconnus par l'automate A_1 et observer un motif.

3 Séparer l'automate en deux parties.

7 Commencer par trouver un automate non déterministe semblable à celui de la question 3 reconnaissant $\mathrm{L}(2,3) \cup \mathrm{L}(5,2)$.

8 Observer qu'il suffit de supprimer des états finals à l'automate déterministe obtenu à la question 7.

9 Utiliser le déterminisme de l'automate pour construire le chemin $q_0, \ldots, q_s$ puis le fait que L est infini pour justifier la transition entre q_s et q_r.

10 Utiliser l'hypothèse que L est un langage rationnel pour en tirer un automate qui le reconnaît puis utiliser la question 9 pour affirmer qu'il existe un état final dans le circuit $q_r, \ldots, q_s$.

11 Prouver par l'absurde. Se ramener à la question précédente en commençant par montrer que L est infini.

Partie II

13 Montrer ce résultat par une récurrence forte sur n. Pour la seconde partie de la question, considérer n de la forme 2^k.

16 Prouver le résultat par récurrence forte sur n. Pour la seconde partie, considérer n de la forme $2^k - 1$.

18 Remarquer que la décomposition en binaire de $n/2$ se déduit aisément de celle de n pour en déduire un algorithme récursif.

19 Utiliser l'écriture binaire inverse $(c_0, \ldots, c_k)$ de n. Utiliser une fonction auxiliaire récursive qui va traiter la décomposition binaire inverse de la puissance, en distinguant deux cas selon la valeur du coefficient courant de la décomposition.

20 Prouver par récurrence sur k en utilisant la formule donnée et en observant que pour obtenir la puissance 2×3^k il suffit de calculer $3^k + 3^k$. Pour la seconde partie de la question, décomposer 27 avec la formule à la manière de la preuve.

21 Se servir de l'exemple précédent $(n = 27)$ pour trouver l'algorithme.

22 Utiliser l'identité $5^k = 2 \times 2 \times 5^{k-1} + 5^{k-1}$.

23 S'inspirer de la suite obtenue à la question précédente pour $n = 125$.

24 Il faut tester toutes les paires d'indices.

25 Utiliser un tableau pour stocker les puissances intermédiaires calculées.

26 Il faut réussir à énumérer toutes les listes en respectant deux conditions : la liste doit être triée par ordre croissant et chaque élément est somme d'exactement deux éléments d'indice plus petit. Visualiser l'énumération exhaustive comme un arbre.

27 Séparer en plusieurs fonctions auxiliaires codant différentes étapes de l'algorithme.

I. Langages et automates

$\boxed{1}$ Soit $(\alpha, \beta) \in \mathbb{N}^2$ fixé. Distinguons deux cas. Si $\alpha \neq 0$, la suite $(\alpha k + \beta)_{k\in\mathbb{N}}$ est strictement croissante (et donc non majorée car c'est une suite d'entiers). Par conséquent, $\mathrm{L}(\alpha, \beta)$ contient une infinité d'éléments. Sinon, $\alpha = 0$ et on a $\mathrm{L}(0, \beta) = \{a^\beta\}$. Ainsi,

$$\boxed{\mathrm{L}(\alpha, \beta) \text{ est fini si et seulement si } \alpha = 0. \text{ Dans ce cas, on a Card } \mathrm{L}(0, \beta) = 1.}$$

$\boxed{2}$ En déroulant les étapes de l'automate, on s'intéresse aux 3 premiers mots reconnus par l'automate pour découvrir le motif :

Mot	**Longueur**	**Décomposition**
aa	2	$2 = 2 + 0 \times 4$
$aaaaaa$	6	$6 = 2 + 1 \times 4$
$aaaaaaaaaa$	10	$10 = 2 + 2 \times 4$

En décomposant la longueur de chaque mot sous la forme $\beta_1 + k\alpha_1$ on trouve les valeurs recherchées. Ainsi,

$$\boxed{\text{L'automate } \mathrm{A}_1 \text{ reconnaît le langage } \mathrm{L}(4, 2).}$$

> Pour aller plus vite, il suffit de regarder le nombre minimum de a à lire pour aller de l'état initial jusqu'à l'état final, ici 2, ce qui donne la valeur de β_1. Pour α_1, la longueur du circuit depuis l'état final donne la valeur recherchée. Cette approche est motivée par la forme $\alpha k + \beta$ de l'ensemble des exposants.

$\boxed{3}$ L'automate A_2 est composé de deux automates de la forme de A_1 : la partie haute (composée des états $\{0, 1, 2, 3\}$) et la partie basse (composée de $\{0, 4, 5, 6\}$). On raisonne alors comme précédemment sur chacune des parties.

> L'automate A_2 est un automate non déterministe. La lecture de la première lettre permet soit d'aller à l'état 1, soit d'aller à l'état 4. Il suffit alors de raisonner séparément sur chaque partie.

Ainsi, la partie haute reconnaît le langage $\mathrm{L}(3, 2)$ et la partie basse reconnaît le langage $\mathrm{L}(2, 3)$. Le langage L_2 reconnu par A_2 est l'union de ces deux langages donc

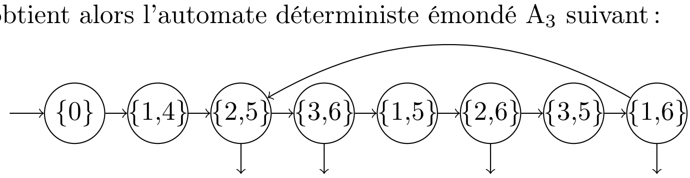
$$\boxed{\mathrm{L}_2 = \mathrm{L}(3, 2) \cup \mathrm{L}(2, 3)}$$

$\boxed{4}$ Pour déterminiser, on construit successivement un automate en partant de l'état initial et en créant des nouveaux états suivant l'ensemble des états accessibles via une transition de l'automate depuis l'état en cours (autrement dit, les successeurs de l'état courant en lisant a).

Par exemple, les successeurs de l'état 0 sont les états 1 et 4. On crée donc un état $\{1, 4\}$. Ensuite, on fait de même avec l'état $\{1, 4\}$: on regarde les successeurs à partir de ces états. On s'arrête lorsqu'on ne crée pas de nouvel état.

État	**Successeurs**
0	$\{1, 4\}$
$\{1, 4\}$	$\{2, 5\}$
$\{2, 5\}$	$\{3, 6\}$
$\{3, 6\}$	$\{1, 5\}$
$\{1, 5\}$	$\{2, 6\}$
$\{2, 6\}$	$\{3, 5\}$
$\{3, 5\}$	$\{1, 6\}$
$\{1, 6\}$	$\{2, 5\}$

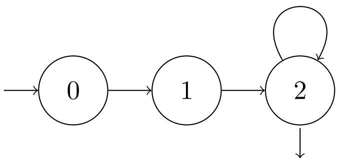
Les états finals de A_3 sont les parties d'états de A_2 qui contiennent un état final de A_2. On obtient alors l'automate déterministe émondé A_3 suivant :



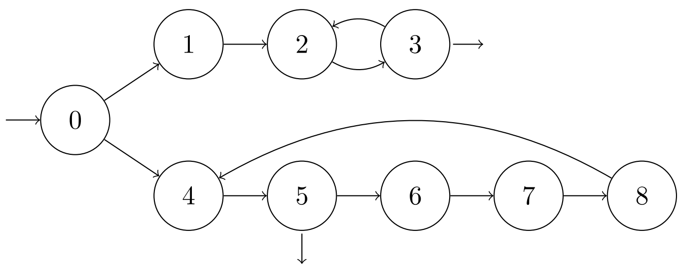
5 On remarque que la forme de l'automate A_3 est similaire à celle de A_1 à ceci près qu'il y a 4 états finals, chacun étant associé à un β_i différent. Comme précédemment, on compte la longueur du chemin depuis l'état initial jusqu'à chacun des états finals pour trouver la valeur β_i associée. Enfin, α_4 est encore une fois la longueur du circuit en partant d'un état final (elle est la même quel que soit l'état final).

$$\boxed{\text{Le langage reconnu par } A_3 \text{ est } L_3 = L(6,2) \cup L(6,3) \cup L(6,5) \cup L(6,7).}$$

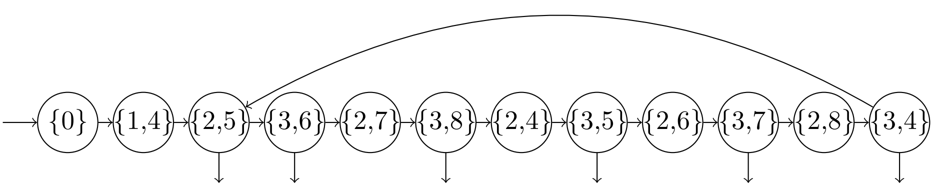
6 Un exemple d'automate de la forme F reconnaissant $L(1,2)$ est



7 En combinant un automate pour $L(2,3)$ en partie haute et un automate pour $L(5,2)$ en partie basse, on obtient l'automate suivant pour le langage $L(2,3) \cup L(5,2)$:

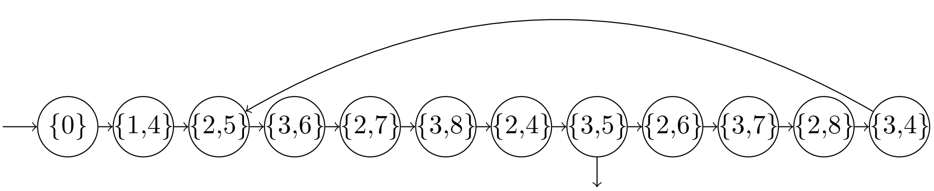


Suivant le cheminement de la question 4, déterminisons cet automate pour en déduire l'automate suivant de la forme F :



> Afin de gagner du temps lors de l'épreuve et déterminiser rapidement le premier automate, il n'est pas la peine de faire le tableau. Il suffit de remarquer que l'on ne crée à chaque fois qu'un seul état et lire parallèlement la partie haute et la partie basse de l'automate.

8 L'automate A_4 suivant est un automate de la forme F reconnaissant le langage $L(2,3) \cup L(5,2)$:



X/ENS Maths A MP 2017 — Corrigé

Ce corrigé est proposé par Florian Metzger (docteur en mathématiques) ; il a été relu par Rémi Pellerin (ENS Lyon) et Sophie Rainero (professeur en CPGE).

Ce sujet traite des formes symplectiques sur un espace vectoriel réel E de dimension finie n paire, c'est-à-dire des formes bilinéaires antisymétriques non dégénérées sur E. Le but est d'en montrer plusieurs propriétés, d'étudier les endomorphismes symétriques pour ces formes symplectiques ainsi que de montrer l'existence de structures complexes qui sont dites domptées par une certaine forme symplectique ω. Une telle structure désigne un automorphisme J de E vérifiant $\mathrm{J}^2 = -\operatorname{Id}_{\mathrm{E}}$ tel que $\omega(x, \mathrm{J}(x)) > 0$ pour tout vecteur $x \neq 0$. La partie II peut être résolue indépendamment des autres.

- Dans la première partie sont établis des résultats élémentaires sur les formes symplectiques ayant pour but de montrer qu'étant donné une forme symplectique ω, il existe une base de E dans laquelle sa matrice écrite par blocs est
$$\mathrm{J}_n = \begin{pmatrix} 0_{n/2} & -\mathrm{I}_{n/2} \\ \mathrm{I}_{n/2} & 0_{n/2} \end{pmatrix}$$
Il en résulte l'existence d'une structure complexe domptée par ω.
- La deuxième partie permet d'obtenir deux résultats sur les polynômes qui seront utiles dans le reste du problème. Le premier est qu'il existe pour tout $d \geqslant 1$ une fonction sur $\mathbb{R}_d$ à valeurs réelles, polynomiale par morceaux, qui est non nulle pour tout polynôme à racines complexes simples. Le second résultat est que $f^{-1}(\mathbb{R}^*)$ est dense dans $\mathbb{R}^d$ pour toute fonction polynomiale réelle non nulle $f : \mathbb{R}^d \to \mathbb{R}$.
- Dans la troisième partie, deux formes symplectiques ω et ω_1 sont fixées et l'on introduit les endomorphismes symétriques pour ω. On étudie leur réduction sous certaines hypothèses portant sur leur polynôme caractéristique. On montre en particulier que l'espace E peut être décomposé en somme directe de sous-espaces stables par un endomorphisme u symétrique pour ω, sur lesquels ω et ω_1 restent symplectiques. Si χ_u est à racines complexes au plus doubles, on peut alors trouver une base dans laquelle $\operatorname{Mat}(\omega)$ est diagonale avec des blocs de J_2 et J_4 et $\operatorname{Mat}(\omega_1)$ est diagonale par blocs qui sont des multiples de J_2 ou des $r\mathrm{Q}_\theta$ avec $r > 0$, $\theta \in \mathbb{R} \smallsetminus \pi\mathbb{Z}$ et
$$\mathrm{Q}_\theta = \begin{pmatrix} 0 & -\mathrm{R}_{-\theta} \\ \mathrm{R}_\theta & 0 \end{pmatrix} \qquad \text{où} \qquad \mathrm{R}_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$
- La dernière partie a pour but l'étude des liens d'implication entre l'existence d'une structure complexe domptée à la fois par ω et ω_1, et le fait que le segment $\{(1-t)\omega + t\omega_1 \mid t \in [\,0\,;1\,]\}$ est inclus dans l'ensemble des formes symplectiques sur E.

Un grand nombre de questions de ce sujet sont des résultats classiques sur les formes bilinéaires, qui ne sont plus au programme depuis 2014. Quelques questions sont abordables ; les autres sont d'un niveau assez soutenu, certaines étant même ardues car elles nécessitent du recul sur le sujet, une très bonne maîtrise des outils d'algèbre linéaire et de polynômes, mais aussi beaucoup d'initiative et de technicité.

Indications

Partie I

1 Introduire une base de E et considérer les formes linéaires qui sont nulles sur la base sauf sur un de ses vecteurs en lequel elles prennent la valeur 1.

3.a Raisonner par analyse-synthèse en choisissant des X et Y adéquats pour trouver les coefficients de la matrice M.

3.b Utiliser l'antisymétrie d'une forme ω sur la base canonique et la question 3.a.

3.c Avec la question 3.b, expliciter la forme des matrices de A(E) quand $\dim \mathrm{E} = 2$.

3.d On pourra montrer successivement $(\mathcal{E}_1) \Longrightarrow (\mathcal{E}_2) \Longrightarrow (\mathcal{E}_3) \Longrightarrow (\mathcal{E}_1)$ et utiliser les questions 1, 3.a ainsi que le morphisme φ_ω.

5 Réécrire ω_0 à l'aide du produit scalaire canonique sur $\mathbb{R}^n$ pour la bilinéarité. Les caractères antisymétrique et symplectique se prouvent à l'aide des conditions obtenues en 3.b et 3.d.

7.a En dimension finie, tout sous-espace vectoriel admet un supplémentaire.

7.d On pourra se servir du théorème du rang et de la question 7.a.

7.e Pour la somme directe, faire référence aux questions 7.b et 7.d. Regarder ensuite ce que donne la matrice de ω dans une base adaptée à la somme directe.

8 Prouver qu'il existe un plan de E sur lequel ω est symplectique et appliquer les résultats des questions 7.d et 7.e pour obtenir l'hérédité.

9 Réindexer les vecteurs d'une base $\widetilde{\mathcal{B}}$ donnée par la question 8.

Partie II

10 Remarquer que l'isomorphie de $\mathrm{L}_{\mathrm{P},\mathrm{Q}}$ est équivalente à son injectivité. Regarder ensuite son noyau quand $\operatorname{pgcd}(\mathrm{P},\mathrm{Q}) = 1$. Réciproquement, si $\mathrm{L}_{\mathrm{P},\mathrm{Q}}$ est un isomorphisme, alors il est surjectif. Or $1 \in \mathbb{R}_{p+q-1}[\mathrm{X}]$.

11 Les racines complexes d'un polynôme réel P sont simples si et seulement si $\operatorname{pgcd}(\mathrm{P},\mathrm{P}') = 1$. On pourra donc considérer l'application $\mathrm{L}_{\mathrm{P},\mathrm{Q}}$ pour P et Q bien choisis. Attention cependant aux conditions de degrés pour pouvoir définir $\mathrm{L}_{\mathrm{P},\mathrm{Q}}$: on pourra donner une définition de r « par morceaux ». Quelle application polynomiale en les coefficients peut être utilisée pour décréter si une application linéaire est un isomorphisme ?

12 On pourra établir que $f^{-1}(\{0\})$ est d'intérieur vide. Pour cela on raisonnera par l'absurde et on montrera qu'une fonction polynomiale s'annulant sur un produit cartésien d'intervalles ouverts non vides est nécessairement nulle.

Partie III

13 Remarquer que pour tout $x \in \mathrm{E}$ fixé, $\omega_1(x,\cdot) \in \mathrm{E}^*$ et que ω est symplectique. Pour la bijectivité, on pourra s'intéresser au noyau de u.

14.a Utiliser la formulation matricielle de $\omega(x,y)$ donnée à la question 3.a. Que dire si A et B sont deux matrices telles que ${}^t\mathrm{X}\,\mathrm{A}\mathrm{Y} = {}^t\mathrm{X}\,\mathrm{B}\mathrm{Y}$ pour tout couple de vecteurs colonnes $(\mathrm{X},\mathrm{Y}) \in (\mathcal{M}_{n,1}(\mathbb{R}))^2$?

14.b En utilisant la question 14.a, chercher la matrice U par blocs sous la forme

$$\mathrm{U} = \begin{pmatrix} \mathrm{N} & \mathrm{M} \\ \mathrm{R} & \mathrm{S} \end{pmatrix}$$

pour $\mathrm{N},\mathrm{M},\mathrm{R},\mathrm{S}$ matrices de taille $(2,2)$.

14.c Le théorème de Cayley-Hamilton pourra être appliqué en explicitant le polynôme caractéristique de matrices de taille 2.

16 Interpréter Z_1, Z_2 comme parties réelle et imaginaire de Z, idem pour Y. Montrer que la famille $(Z_1, Z_2, Y_1, -Y_2)$ est libre dans $\mathbb{R}^4$. Il pourra être utile de considérer $U\overline{Z}$.

17 Montrer que si V, W sont vecteurs propres de U pour $\lambda \notin \mathbb{R}$, alors ${}^t\overline{V}\, J_2 W = 0$ et l'appliquer pour Y, Z en identifiant les parties réelles et imaginaires.

18 Écrire les nouveaux vecteurs y_1, y_2 obtenus pour ξY si $\xi = \xi_1 + i\xi_2 \in \mathbb{C}^*$. Remarquer que les conditions demandées par l'énoncé se réécrivent sous la forme d'un système linéaire avec une matrice dont l'inversibilité peut être étudiée à l'aide de la question 17.

19 Utiliser la question 18 pour trouver une base dans laquelle la matrice de ω est J_4. Que dire de celle de u dans la même base ?

20 Utiliser le lemme des noyaux ainsi que le fait que u commute avec tout endomorphisme qui est un polynôme en u.

21 Choisir une base de E dans laquelle la matrice de ω est J_n. Chercher, comme à la question 14.a, une relation vérifiée par U, la matrice de u dans cette base. Il pourra également être judicieux d'utiliser une relation de Bézout entre P_j et P_k pour écrire $x \in F_k$ et $y \in F_j$ faisant intervenir des polynômes d'endomorphismes en $P_j(u)$ et $P_k(u)$. Ce faisant, évaluer $\omega(x, y)$ en utilisant la forme matricielle de la question 3.a.

22 Se servir du critère obtenu à la question 7.b ainsi que des résultats des questions 20 et 21.

23 C'est une question de synthèse de plusieurs questions précédentes. Considérer une décomposition en facteurs premiers de χ_u donnée par ses racines complexes. Utiliser ensuite les sous-espaces F_j associés par la question 20. On pourra montrer que si $\chi_u = (X - \lambda)^m Q$ avec $Q(\lambda) \neq 0$, alors $\dim \operatorname{Ker} (u - \lambda \operatorname{Id}_E)^m = m$.

Partie IV

24 Construire une structure complexe domptée par $\omega_{|F_j \times F_j}$ et $\omega_{1|F_j \times F_j}$ sur chaque sous-espace F_j d'une décomposition en somme directe comme à la question 23. Remarquer que les restrictions à des sous-espaces de dimensions 2 et 4 ont été étudiées aux questions 3.c et 19. L'hypothèse $(\mathcal{F}_2)$ pourra être utilisée pour montrer que les restrictions de ω et ω_1 à des espaces F_j de dimension 2 sont strictement positivement liées.

25 Commencer par raisonner en reformulant $\mathcal{S}$ à l'aide de relations matricielles. On pourra ensuite remarquer que

$$D = \left\{ U \in \mathcal{M}_n(\mathbb{R}) \mid {}^tU\, J_n = J_n U \right\}$$

est un sous-espace vectoriel de $\mathcal{M}_n(\mathbb{R})$. Penser aux questions 11 et 12 pour construire, à l'aide d'un d bien choisi, une application polynomiale de $\mathbb{R}^d$ dans $\mathbb{R}$ dont l'image réciproque permet de décrire l'ensemble dont on souhaite montrer la densité. On pourra enfin remarquer que les applications suivantes

$$\mathcal{X} \colon \begin{cases} D \longrightarrow \mathbb{R} \\ U \longmapsto r({\chi_U}') \end{cases} \qquad \text{et} \qquad \delta \colon \begin{cases} D \longrightarrow \mathbb{R} \\ U \longmapsto \det U \end{cases}$$

sont polynomiales et que deux ouverts denses ont une intersection dense.

26 L'énoncé semble indiquer une piste qui ne permet pas d'aboutir. On pourra quand même expliciter la stratégie suggérée et quelles en sont les limites.

I. Bases symplectiques

$\boxed{1}$ Soit $\mathscr{B} = (e_1, \ldots, e_n)$ une base de E. Exhibons une famille libre et génératrice de E^* à l'aide de $\mathscr{B}$. Il suffit de définir les formes linéaires sur $\mathscr{B}$. Soit donc, pour tout $i \in [\![1\,; n]\!]$, la forme linéaire e_i^* vérifiant

$$\forall (i,j) \in [\![1\,; n]\!]^2 \qquad e_i^*(e_j) = \delta_{i,j}$$

où l'on rappelle que $\delta_{i,j}$ est le symbole de Kronecker défini par

$$\delta_{i,j} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

Ainsi e_i^* est l'application qui donne la i^{e} coordonnée d'un vecteur dans la base $\mathscr{B}$. Établissons que $\mathscr{B}^* = (e_1^*, \ldots, e_n^*)$ est une base de E^*.

Montrons que $\mathscr{B}^*$ est génératrice de E^*. Soient $\varphi \in \mathrm{E}^*$ et $x \in \mathrm{E}$. Écrivons

$$x = \sum_{i=1}^{n} e_i^*(x) e_i$$

Par linéarité de φ, $$\varphi(x) = \varphi\Big(\sum_{i=1}^{n} e_i^*(x) e_i\Big) = \sum_{i=1}^{n} e_i^*(x)\varphi(e_i)$$

ce qui fournit $$\varphi = \sum_{i=1}^{n} \varphi(e_i) e_i^*$$

et donc $\mathscr{B}^*$ est génératrice.

Montrons que la famille $\mathscr{B}^*$ est libre dans E^*. Fixons $\lambda_1, \ldots, \lambda_n$ des réels tels que

$$\sum_{i=1}^{n} \lambda_i e_i^* = 0$$

L'évaluation de la forme linéaire du membre de gauche de cette inégalité en tout e_j de la base $\mathscr{B}$ fournit pour tout $j \in [\![1\,; n]\!]$

$$0 = \sum_{i=1}^{n} \lambda_i e_i^*(e_j) = \sum_{i=1}^{n} \lambda_i \delta_{i,j} = \lambda_j$$

Cela prouve la liberté de $\mathscr{B}^*$.

Finalement, $\mathscr{B}^*$ est une base de E^* et par suite

$$\boxed{\dim \mathrm{E}^* = \dim \mathrm{E} = n}$$

> La base $\mathscr{B}^*$ s'appelle *base duale de la base* $\mathscr{B}$. Réciproquement, si une base B' de E^* est donnée, il existe une unique base B de E dont la duale est B'. La base B est appelée *base antéduale de* B'.

$\boxed{2}$ Soient $\omega \in \mathrm{A(E)}$ et $x \in \mathrm{E}$. Comme ω est antisymétrique, la définition de A(E) assure que $\omega(x,y) = -\omega(y,x)$ pour tout $(x,y) \in \mathrm{E}^2$ donc en particulier, pour $x = y$,

$$\omega(x,x) = -\omega(x,x)$$

soit $2\omega(x,x) = 0$. Comme 2 est inversible dans $\mathbb{R}$, l'égalité précédente assure que

$$\boxed{\omega(x,x) = 0}$$

> Le résultat que l'on vient de démontrer s'appelle caractère *alterné* de la forme bilinéaire. On définit la caractéristique d'un corps k (notée $\mathrm{car}(k)$) comme l'unique entier $p \in \mathbb{N}$ tel que $\mathrm{Ker}\ \psi = p\mathbb{Z}$ où ψ est le morphisme $n \mapsto n \cdot 1_k$

X Maths B MP 2017 — Corrigé

Ce corrigé est proposé par Matthias Moreno Ray (professeur en CPGE) ; il a été relu par Émilie Liboz (professeur en CPGE) et Guillaume Batog (professeur en CPGE). L'auteur remercie Denis Choimet pour son aide précieuse.

Le problème étudie des inégalités fonctionnelles issues de la théorie des lois de probabilités continues. Une loi continue sur $\mathbb{R}$ est définie à partir d'une mesure $m : \mathbb{R} \to \mathbb{R}$ continue, positive, intégrable et d'intégrale valant 1 sur $\mathbb{R}$. Le problème met en valeur le rôle de la mesure gaussienne $\mu(x) = \mathrm{e}^{-x^2}/\sqrt{\pi}$ pour $x \in \mathbb{R}$, qui correspond à la loi normale centrée d'écart-type $1/\sqrt{2}$. Les inégalités à démontrer portent sur la variance $\mathrm{Var}_m(f)$ et l'entropie $\mathrm{Ent}_m(f)$ d'une fonction $f : \mathbb{R} \to \mathbb{R}$ continue relativement à une mesure m. Ces quantités sont définies en cas d'intégrabilité par

$$\mathrm{Var}_m(f) = \mathrm{E}_m(f^2) - \big(\mathrm{E}_m(f)\big)^2 \qquad \text{et} \qquad \mathrm{Ent}_m(f) = \mathrm{E}_m(h \circ f^2) - h\big(\mathrm{E}_m(f^2)\big)$$

avec $$\forall\, x > 0 \quad h(x) = x \ln x \qquad \text{et} \qquad \mathrm{E}_m(g) = \int_{-\infty}^{+\infty} g(x)m(x)\,\mathrm{d}x$$

pour g continue intégrable sur $\mathbb{R}$.

- La partie préliminaire établit des liens entre l'intégrabilité d'une fonction sur $\mathbb{R}$ et l'existence d'une variance ou d'une entropie. On y caractérise également les fonctions d'entropie nulle dans le cas d'une mesure qui ne s'annule pas. Les questions sont proches de démonstrations du cours.
- La partie I étudie la mesure μ. On introduit un opérateur différentiel L et un opérateur intégral Φ qui présentent des propriétés d'invariance vis-à-vis de μ (caractère symétrique de L, invariance de la moyenne par Φ). Certaines questions sont techniques et utilisent plusieurs théorèmes relatifs aux intégrales à paramètres.
- La partie II prolonge la précédente en établissant que, pour toute fonction $f \in \mathscr{C}_b^2$ (de classe $\mathscr{C}^2$ sur $\mathbb{R}$ à dérivées bornées sur $\mathbb{R}$), on a $\mathrm{Ent}_\mu(f) \leqslant \mathrm{E}_\mu({f'}^2)$.
- La partie III s'intéresse à l'inverse aux mesures m vérifiant $\mathrm{Ent}_m(f) \leqslant \mathrm{E}_m({f'}^2)$ pour $f \in \mathscr{C}_b^1$. D'une part, on montre que $\mathrm{Var}_m(f) \leqslant \mathrm{E}_m({f'}^2)/2$. D'autre part, on établit que $x \longmapsto \mathrm{e}^{\alpha x^2}\, m(x)$ est intégrable sur $\mathbb{R}$ pour tout $\alpha < 1$. Si cette fonction est de plus bornée, cela implique que m est dominée par la mesure gaussienne μ au voisinage de l'infini.
- La partie IV est indépendante et se termine par une majoration de la probabilité, pour la mesure μ, qu'un réel soit à distance au moins t d'une union finie d'intervalles donnés.

Ce problème couvre plusieurs points du programme d'analyse : les intégrales à paramètre, les intégrales généralisées et plusieurs inégalités fondamentales de l'analyse (convexité, accroissements finis, Cauchy-Schwarz, Taylor reste intégrale). Quelques questions portent sur l'étude de fonctions auxiliaires à deux variables.

Il constitue un bon test des capacités de rédaction et de persévérance, certaines questions demandant de lier de nombreux résultats intermédiaires entre eux. Bien que des formules-étapes non fournies par l'énoncé soient vitales pour résoudre les questions de synthèse, suffisamment de résultats sont donnés pour que l'on puisse poursuivre l'étude du problème sans l'avoir entièrement résolu.

Indications

Préliminaires

1 Pour l'intégrabilité de $f\,m$, utiliser la majoration de $|u|$ par u^2+1 valable pour tout réel u. Pour la deuxième partie de la question, calculer la variance de la fonction $f - \displaystyle\int f(x)\,m(x)\,\mathrm{d}x$.

2a Montrer que, pour tout réel u positif, $u-1 \leqslant h(u)$ et évaluer cette inégalité en $f(x)^2$.

2c Dans le cas où $a>0$, utiliser l'indication de l'énoncé. Dans le cas où $a=0$, montrer que $\mathrm{Ent}_m(f)=0$.

2d Montrer que $\mathrm{Ent}_m(f)=0$ si, et seulement si, f est constante.

Partie I

3b Remplacer $\mathrm{L}h_2$ par l'expression obtenue à la question 3a. Effectuer ensuite une intégration par parties.

4 Utiliser le théorème de continuité sous le signe intégrale.

5a Utiliser le théorème de dérivation sous le signe intégrale pour montrer l'existence et obtenir l'expression des dérivées partielles. Utiliser ensuite le théorème de continuité sous le signe intégrale pour justifier que ces dérivées partielles sont continues sur $\mathbb{R}^2$.

5b Déduire la formule demandée de l'expression de $\partial_x\Phi_f$ obtenue à la question 5a.

5c Calculer d'une part $\mathrm{L}\Phi_f(t,x)$ grâce aux résultats des questions 5a et 5b. Transformer d'autre part l'expression de $\partial_t\Phi_f$ obtenue à la question 5a en intégrant par parties (primitiver $y \longmapsto y\,\mu(y)$ et dériver $y \longmapsto f'(x\,\cos(t)+y\,\sin(t))$).

5d Montrer que $t \longmapsto \displaystyle\int \Phi_f(t,x)\,\mu(x)\,\mathrm{d}x$ est dérivable sur $\mathbb{R}$ et de dérivée nulle.

Partie II

6 Utiliser le théorème de continuité sous le signe intégrale.

7a Justifier que J est de classe $\mathscr{C}^1$ grâce au théorème de dérivation sous le signe intégrale. Simplifier ensuite l'expression de J' grâce à la question 3b.

7b Reconnaître une inégalité de Cauchy-Schwarz.

7c Exprimer l'accroissement de J entre 0 et $\pi/2$ sous forme d'une intégrale et le majorer grâce aux questions 7b et 5d.

8 Suivre l'indication de l'énoncé. Utiliser le théorème de convergence dominée pour calculer les limites lorsque $\delta \to 0$.

Partie III

10a Composer f par une fonction affine pour se ramener à $\displaystyle\int f(x)\,m(x)dx = 0$ et $\mathrm{Var}_m(f)=1$.

10b Appliquer à la fonction h la formule de Taylor avec reste intégrale à l'ordre 4. Montrer que le reste intégrale est positif. Évaluer l'inégalité obtenue en $\varepsilon f(y)$ avec y réel et ε voisin de zéro.

11a Utiliser l'inégalité (1) de l'énoncé avec $g(x) = \mathrm{e}^{\lambda\,f(x)/2}$.

11b Calculer la dérivée de $\ln \mathrm{H}(\lambda)/\lambda$ et majorer avec la question 11a. Intégrer l'inégalité obtenue sur le segment $[\,\varepsilon\,;\lambda\,]$ avec $\varepsilon > 0$. Montrer que $\ln \mathrm{H}(\varepsilon)/\varepsilon$ tend vers $\mathrm{H}'(0)$ lorsque $\varepsilon \to 0^+$.

12 Suivre l'indication de l'énoncé. Utiliser le théorème de convergence dominée sur $\mathbb{R}$ pour justifier que

$$\int f_n(x)\, m(x)\, \mathrm{d}x \xrightarrow[n\to\infty]{} \int x\, m(x)\, \mathrm{d}x$$

Montrer que $\forall\, \mathrm{A} > 0 \quad \displaystyle\int_{-\mathrm{A}}^{\mathrm{A}} \mathrm{e}^{\lambda f_n(x)}\, m(x)\, \mathrm{d}x \xrightarrow[n\to\infty]{} \int_{-\mathrm{A}}^{\mathrm{A}} \mathrm{e}^{\lambda f(x)}\, m(x)\, \mathrm{d}x$

Utiliser enfin que $\displaystyle\int_{-\mathrm{A}}^{\mathrm{A}} \mathrm{e}^{\lambda f_n(x)}\, m(x)\, \mathrm{d}x \leqslant \int \mathrm{e}^{\lambda f_n(x)}\, m(x)\, \mathrm{d}x$

et l'inégalité (3) de l'énoncé.

13a Prendre $\lambda = 2(a - \mathrm{M})/\mathrm{C}$ et utiliser la question 12.

13b Justifier que la fonction $\varphi : x \longmapsto \displaystyle\int_x^{+\infty} m(t)\, \mathrm{d}t$ est une primitive de $-m$ sur $\mathbb{R}$. Utiliser φ pour effectuer une intégration par parties dans l'intégrale $\displaystyle\int_0^{\mathrm{A}} \mathrm{e}^{\alpha x^2}\, m(x)\, \mathrm{d}x$. Majorer les termes obtenus pour justifier l'intégrabilité de la fonction sur $\mathbb{R}_+$. Appliquer ensuite ce résultat à la mesure $\widetilde{m}(x) = m(-x)$.

Partie IV

14a Montrer que la fonction $x \longmapsto \displaystyle\int_{-\infty}^{x} p(t)\, \mathrm{d}t$ est de classe $\mathscr{C}^1$ et strictement croissante. Normaliser cette fonction pour obtenir une bijection entre $\mathbb{R}$ et $]\,0\,;1\,[$.

14b Réaliser le changement de variable indiqué par l'énoncé. Minorer l'intégrale avec l'hypothèse (4) puis avec une inégalité de type $a + b \geqslant 2\sqrt{ab}$.

15b Poser $p(x) = \mathrm{e}^{d(x,\mathrm{A})/2 - x^2}$, $q(x) = \mathbb{1}_{\mathrm{A}}(x)\mathrm{e}^{-x^2}$ et $r(x) = \mathrm{e}^{-x^2}$ et utiliser la question 15a.

16a Si I est un intervalle, alors I_t est un intervalle, pour t réel.

Préliminaires

1 Soit $x \in \mathbb{R}$. Si $|f(x)| \geqslant 1$ alors $|f(x)| \leqslant f(x)^2$. On a donc dans tous les cas

$$|f(x)| \leqslant f(x)^2 + 1 \qquad \text{d'où} \qquad |f(x)|\, m(x) \leqslant f(x)^2\, m(x) + m(x)$$

Supposons que f admette une variance relativement à m. Dans ce cas, la fonction $f^2\, m$ est intégrable sur $\mathbb{R}$. Par définition d'une mesure, la fonction m est intégrable sur $\mathbb{R}$. L'inégalité précédente prouve donc que

$$\boxed{\text{Si } f \text{ admet une variance relativement à } m, \text{ alors } fm \text{ est intégrable sur } \mathbb{R}.}$$

Notons $\mathrm{I} = \displaystyle\int f(x)\, m(x)\, \mathrm{d}x$. Pour $x \in \mathbb{R}$, on a

$$(f(x) - \mathrm{I})^2\, m(x) = f(x)^2\, m(x) - 2\mathrm{I} f(x) m(x) + \mathrm{I}^2\, m(x)$$

ce qui prouve que $x \longmapsto (f(x) - \mathrm{I})^2\, m(x)$ est intégrable sur $\mathbb{R}$ en tant que combinaison linéaire de fonctions intégrables. Il vient en outre, par linéarité de l'intégrale,

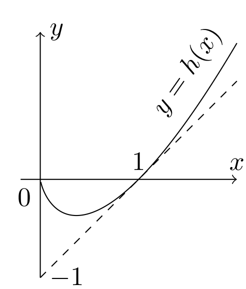
$$\begin{aligned}\int (f(x) - \mathrm{I})^2\, m(x)\, \mathrm{d}x &= \int f(x)^2\, m(x)\, \mathrm{d}x - 2\mathrm{I} \int f(x) m(x)\, \mathrm{d}x + \mathrm{I}^2 \int m(x)\, \mathrm{d}x \\ &= \int f(x)^2\, m(x)\, \mathrm{d}x - 2\mathrm{I}^2 + \mathrm{I}^2 \qquad \left(\text{car} \int m(x)\, \mathrm{d}x = 1\right)\end{aligned}$$

$$\int (f(x) - \mathrm{I})^2\, m(x)\, \mathrm{d}x = \int f(x)^2\, m(x)\, \mathrm{d}x - \mathrm{I}^2 = \mathrm{Var}_m(f)$$

Comme la fonction $x \longmapsto (f(x) - \mathrm{I})^2\, m(x)$ est positive sur $\mathbb{R}$, l'intégrale de gauche est positive, ce qui prouve que

$$\boxed{\mathrm{Var}_m(f) \geqslant 0}$$

> Comme f est continue et que $(f - \mathrm{I})^2$ est de signe constant, on peut observer que, dans le cas où m ne s'annule pas, la variance de f est nulle si et seulement si $(f - \mathrm{I})^2$ est nulle, ce qui équivaut à f constante.
>
> L'intégrabilité de $f\, m$ s'obtient aussi avec l'inégalité de Cauchy-Schwarz appliquée aux fonctions $|f|\, \sqrt{m}$ et $\sqrt{m}$ dans l'espace préhilbertien
>
> $$\mathrm{E} = \left\{ f \in \mathscr{C}^0(\mathbb{R}, \mathbb{R}) \mid f^2 \text{ est intégrable sur } \mathbb{R} \right\}$$
>
> muni du produit scalaire $(f \mid g) = \displaystyle\int f(x) g(x)\, \mathrm{d}x$. Ce produit scalaire est bien défini sur $\mathrm{E} \times \mathrm{E}$ car
>
> $$\forall x \in \mathbb{R} \qquad |f(x) g(x)| \leqslant (f(x)^2 + g(x)^2)/2$$
>
> L'inégalité de Cauchy-Schwarz appliquée à $f\, \sqrt{m}$ et $\sqrt{m}$ montre ensuite la positivité de la variance.



2a Pour comparer f^2 et $h(f^2)$, étudions la fonction h. Elle est de classe $\mathscr{C}^2$ sur $]\,0\,; +\infty\,[$, et pour $t > 0$, on a $h''(t) = 1/t$ qui est strictement positif. La fonction h est donc convexe. Sa tangente au point d'abscisse 1 a pour équation $y = t - 1$ d'où

$$\forall t > 0 \qquad h(t) \geqslant t - 1$$

Cette inégalité est par ailleurs également vraie lorsque $t = 0$.

X/ENS Informatique B (MP/PC) 2017 — Corrigé

Ce corrigé est proposé par Josselin Giet (ENS Ulm) ; il a été relu par Virgile Andreani (ENS Ulm) et Vincent Puyhaubert (professeur en CPGE).

Ce sujet présente une solution optimisée à la recherche d'intersection d'ensembles de points à coordonnées entières.

- La première partie permet de définir le problème et d'en donner une solution naïve et guidée, dans le langage Python. Comme toute résolution naïve, elle permet de constater la nécessité de trouver une optimisation.
- La deuxième partie relie cette problématique à l'utilisation d'une base de données. Elle permet d'écrire des exemples de requêtes SQL afin de résoudre le problème.
- La troisième partie introduit la notion de codage de Lebesgue d'un point à coordonnées entières. Il s'agit d'une autre manière de représenter un point en entrelaçant l'écriture binaire de ses coordonnées.
- La quatrième partie introduit la notion d'ordre lexicographique appliquée aux codages de Lebesgue de points ainsi qu'une interprétation visuelle du codage de Lebesgue sous forme de « chemin » pour trouver la position du point.
- La dernière partie définit le codage de Lebesgue compacté d'un ensemble de points et utilise ce procédé pour résoudre le problème du calcul de l'intersection d'ensembles de points de manière optimisée.

Cet énoncé permet d'aborder un problème d'algorithmique original et intéressant. Les remarques tout au long de l'énoncé ainsi que les questions « exemple » permettent de bien comprendre les nouvelles notions introduites. La remarque en début d'énoncé, à propos des opérations sur les listes, interdit d'utiliser un grand nombre d'opérations qui sont rapides à écrire en Python, comme la compréhension de liste.

Indications

Partie I

2 Utiliser la fonction de la question précédente.

Partie II

4 Faire une jointure entre les tables `POINTS` et `MEMBRE`.

5 Utiliser l'opérateur `INTERSECT`.

6 Faire une jointure dont un des arguments est une autre requête complète adéquate.

Partie III

8 On peut remarquer que chaque élément de la liste peut être calculé indépendamment des autres.

Partie IV

9 Lire la remarque sur l'ordre lexicographique, ce qui permet de mieux comprendre la définition.

11 Il y a une coquille dans cette question : Il ne faut pas lire le mot « compacte ».

Partie V

13 Faire attention au cas `k == 0`.

14 Utiliser la fonction de la question précédente.

15 Expliciter ce que signifie une inclusion stricte de deux ensembles au niveau de leur codage de Lebesgue.

16 Utiliser l'ordre lexicographique pour parcourir une seule fois les deux AQL en parallèle.

I. Une solution naïve en Python

1 L'algorithme consiste à regarder tous les éléments de `q`, un par un, et à tester si l'un d'entre eux est égal à `p`.

```
def membre(p,q):
    for i in range(len(q)):
        q_i = q[i]
        if p[0] == q_i[0] and p[1] == q_i[1]:
            return True
    return False
```

> Cet algorithme s'arrête quand on a trouvé un élément dans la liste égal à `p` (ce qui correspond au `return True` ligne 5) ou quand on a fini d'énumérer les éléments de `q` (ce qui correspond au `return False` ligne 6).
>
> Le test de la ligne 4 peut s'écrire de nombreuses manières. Par exemple, en remarquant que `q` est en réalité une matrice de taille `len(p)`$\times$`2`, on peut enlever la ligne 4 et remplacer `q_i[0]` (*resp.* `q_i[1]`) par `q[i][0]` (*resp.* `q[i][1]`), ce qui sera toujours le cas dans la suite.
>
> Enfin, il est possible d'écrire une version purement itérative de cet algorithme (*i.e.* en ne faisant pas de `return` dans une boucle `for` ou `while`). Il faut savoir le faire, car de nombreux langages ne permettent pas de renvoyer le résultat en plein milieu d'une boucle `for`. Pour cela, on utilise une boucle `while` dont le test dépend du nombre d'éléments de `q` testés, ainsi que d'une variable qui dit si un élément de `q` égal à `p` a été trouvé.
>
> ```
> def membre(p,q):
>     res , i = False , 0
>     while (i < len(q)) and not res:
>         if p[0] == q[i][0] and p[1] = q[i][1]:
>             res = True
>         else
>             i += 1
>     return res
> ```

2 En suivant l'algorithme donné dans l'énoncé, on obtient la fonction ci-dessous.

```
def intersection(p,q):
    res = []
    for i in range(len(p)):
        if membre(p[i],q):
            res.append(p[i])
    return res
```

3 Notons $|p|$ (*resp.* $|q|$) la taille de `p` (*resp.* de `q`). À chaque passage dans la boucle `for` de la fonction `membre`, on fait deux comparaisons. Ainsi, la complexité d'un appel de la fonction `membre` est $2 \cdot |q|$. Or, on fait $|p|$ appels à la fonction `membre` dans la fonction `intersection`. Par conséquent,

$$\boxed{\text{La complexité de l'algorithme est } \mathcal{O}(|p| \cdot |q|)}$$

II. Une solution naïve en SQL

4 Cette requête peut s'écrire en utilisant une jointure.

```
SELECT idensemble
FROM POINTS JOIN MEMBRE ON id = idpoint
WHERE x = a AND y = b
```

> Cette jointure peut aussi s'écrire comme un simple produit cartésien en déplaçant la condition de jointure à la condition de sélection.
>
> ```
> SELECT idensemble
> FROM POINTS JOIN MEMBRE
> WHERE id = idpoint AND x = a AND y = b
> ```
>
> Il faut toutefois noter que cette manière de procéder, bien que parfaitement correcte, est fortement déconseillée, car elle construit toute la table alors que la jointure sélectionne les entrées puis fait les tests de la ligne `WHERE`. Dès lors, dans le premier cas, la requête est généralement plus rapide. Dans la suite, nous utiliserons systématiquement une jointure sur une clé primaire avant de faire des tests.

5 Cette requête peut se faire au moyen d'une intersection.

```
(   SELECT idpoint
    FROM MEMBRE
    WHERE idensemble = i)
INTERSECT
(   SELECT idpoint
    FROM MEMBRE
    WHERE idensemble = j)
```

> Cette requête n'est pas la seule réponse possible. On peut aussi bien faire une jointure sur la table `MEMBRE`, comme dans l'exemple suivant :
>
> ```
> SELECT t1.idpoint
> FROM MEMBRE AS t1 JOIN MEMBRE AS t2
>     ON t1.idpoint = t2.idpoint
> WHERE t1.idensemble = i AND t2.idensemble = j
> ```
>
> Cette requête crée, en effet, une table relationnelle contenant
>
> `(t1.idpoint, t1.ensemble, t2.idpoint, t2.ensemble)`
>
> mais comme `t1.idpoint = t2.idpoint`, chaque entrée de cette table correspond à un point et à un couple d'ensembles qui contiennent ce point (ce couple peut être deux fois le même ensemble et on retrouve deux fois la même entrée en échangeant `t1.idensemble` et `t2.idensemble`).
>
> On peut écrire la requête plus lisiblement que dans l'exemple précédent en utilisant l'opérateur `IN` (qui n'est pas explicitement au programme).
>
> ```
> SELECT x,y
> FROM POINTS JOIN MEMBRE ON id = idpoint
> WHERE idensemble = i AND idpoint IN
>     (SELECT idpoint
>      FROM MEMBRE
>      WHERE idensemble = j)
> ```