

Partiel de M.A.O. Calcul Formel Durée : 3 heures

Mai 2020
Master 1 M.F., Orsay

Suite à l'épidémie de Covid-19, cette épreuve a lieu à distance. Les modalités sont donc différentes de celles d'un partiel habituel, mais le sujet et le barême sont conçus comme si il s'agissait d'un partiel dans des conditions normales.

Tous les documents sont autorisés (y compris le poly, les vidéos, les TP, ..., et évidemment le livre "Calcul mathématique avec Sage"). Il est interdit de communiquer. Au plus tard le dimanche 17 mai à 20 heures vous devez m'envoyer un mail avec votre copie (scannée ou prise en photo). Vérifiez bien que l'image soit nette, et facilement lisible. Vous devez joindre à ce mail le fichier Notebook qui contient les calculs que vous aurez effectués en Sage. Cette feuille de calcul sera corrigée en même temps que votre copie ; il est donc conseillé de commenter raisonnablement vos programmes, et de veiller à leur lisibilité. Vous pouvez bien sûr créer des feuilles de calcul auxiliaires, qui ne seront pas corrigées et peuvent vous servir de brouillon électronique.

Ce partiel est une épreuve de mathématiques. Avant d'utiliser l'ordinateur, il convient de se demander s'il n'est pas possible d'obtenir par un raisonnement simple ce qui pourrait être obtenu par un calcul plus complexe fait par ordinateur. A titre indicatif, dans les questions signalées par le signe ($\star$), on s'attend à une réponse utilisant de façon essentielle l'ordinateur ; pour les autres, on s'attend davantage à une réponse argumentée, utilisant éventuellement l'ordinateur mais de façon plus marginale.

Les modalités d'utilisation de Sage sont les suivantes. Pour ce partiel, vous pouvez utiliser toutes les fonctions implémentées dans Sage (notamment pour calculer des pgcd ou des sommes, produits ou puissances, y compris de polynômes ou de matrices). Il est inutile de recopier sur votre copie les lignes de code Sage que vous avez utilisées, puisque vous m'enverrez votre code par email. En revanche vous *devez* inscrire sur votre copie le résultat du calcul, et toutes les explications utiles (notamment toute justification mathématique de la façon dont vous avez fait votre programme). Vous devez aussi, le cas échéant, faire le lien dans votre copie entre le résultat brut donné par Sage et l'interprétation mathématique qui vous est demandée.

En tout état de cause n'hésitez pas à me poser des questions par email si les consignes ne sont pas claires, ou si vous n'êtes pas sûr de bien comprendre une des questions.

Dans tout le problème, on note $\mathbb{K}$ un corps sur lequel les opérations arithmétiques (addition, soustraction, multiplication, division) sont implémentées. Lorsqu'on évoquera le coût d'un

algorithme, il s'agira toujours de son coût en nombre d'opérations arithmétiques dans $\mathbb{K}$. On ne cherchera *jamais* à calculer exactement le nombre d'opérations arithmétiques utilisées : la réponse attendue est toujours de la forme $O(\ldots)$ où les points de suspension désignent une fonction simple du (ou des) paramètre(s) du problème. Calculer la constante implicite dans le symbole O ne fait pas partie de la question posée.

Les trois parties de ce sujet sont, dans une large mesure, indépendantes ; elles sont de difficulté variable, mais au sein de chaque partie la difficulté est plutôt croissante. Les questions qui utilisent les parties précédentes sont indiquées explicitement au début de chaque partie.

Pour tout $d \in \mathbb{N}$, on note $\mathbb{K}_d[X]$ l'ensemble des polynômes de $\mathbb{K}[X]$ de degré inférieur ou égal à d.

Partie 1 : Une équation polynomiale

Dans cette partie on se donne un entier $d \in \mathbb{N}^*$ et un polynôme $A \in \mathbb{K}[X]$ tels que $\deg A < 2d$. On cherche deux polynômes $P, Q \in \mathbb{K}[X]$ tels que

$$\deg P \leq d, \qquad \deg Q < d \qquad \text{et} \qquad P(X)A(X) \equiv Q(X) \bmod X^{2d}. \tag{1}$$

1. Soit $P \in \mathbb{K}_d[X]$; notons $P(X)A(X) = \sum_{i=0}^{3d-1} \lambda_i X^i$. Démontrer qu'il existe un polynôme Q tel que (1) soit vérifiée si, et seulement si, $\lambda_i = 0$ pour tout $i \in \{d, d+1, \ldots, 2d-1\}$. Quand c'est le cas, que vaut Q en fonction des λ_i ?
2. Déduire de la question précédente une façon de tester, pour un polynôme $P \in \mathbb{K}_d[X]$ donné, si il existe un polynôme Q tel que (1) soit vérifiée. Quel algorithme vu en cours utilise-t-on pour effectuer ce test efficacement ? Quel est le coût de ce test en opérations arithmétiques dans $\mathbb{K}$?
3. ($\star$) Programmer l'algorithme de la question 1 en Sage. Précisément, en se restreignant au cas où $\mathbb{K} = \mathbb{Q}$, programmer une fonction en Sage qui prend en entrée un entier `d` et deux polynômes `P` et `A`, et renvoie une liste `[Bool, Q]`. Lorsqu'on applique cette fonction à des polynômes `P` et `A` tels que $\deg P \leq d$ et $\deg A < 2d$, `Bool` doit valoir `True` si il existe un polynôme Q tel que (1) soit vérifiée, et `False` sinon. Dans le cas où `Bool` vaut `True`, le polynôme `Q` renvoyé doit vérifier (1). *On ne demande pas de programmer la multiplication des polynômes : on utilisera sans restriction les fonctions implémentées en Sage.*
4. ($\star$) Appliquer la fonction précédente à $d = 3$, $P(X) = X^3 + 2X^2 - X - 1$ et $A(X) = 11X^5 - 3X^4 + 4X^3 + X + 2$. Ecrire sur la copie ce qu'elle renvoie. Même question avec $P(X) = X^3$ et les mêmes valeurs de d et A.
5. En utilisant la question 1, montrer que la recherche des couples (P, Q) vérifiant (1) se ramène à la résolution d'un système linéaire de d équations à $d+1$ inconnues. Rappeler le nom de l'algorithme permettant alors de trouver ces couples (P, Q), et le coût de ce calcul en opérations arithmétiques dans $\mathbb{K}$.

6. Démontrer qu'il existe toujours un couple (P, Q), autre que $(0, 0)$, qui vérifie (1).

Partie 2 : Récurrences linéaires

Excepté la question 15, cette partie est indépendante de la partie précédente (même si il est fortement conseillé d'aborder la partie 1 avant la partie 2).

Soit $(u_n)_{n\in\mathbb{N}}$ une suite d'éléments de $\mathbb{K}$, et $d \in \mathbb{N}$. Etant donné $P = a_d X^d + \ldots + a_1 X + a_0 \in \mathbb{K}_d[X]$, on dit que $(u_n)_{n\in\mathbb{N}}$ *est annulée par* P *à l'ordre* d si on a :

$$a_0 u_{n+d} + a_1 u_{n+d-1} + \ldots + a_d u_n = \sum_{i=0}^{d} a_i u_{n+d-i} = 0 \text{ pour tout } n \in \mathbb{N}.$$

Lorsqu'il existe $P \in \mathbb{K}_d[X]$ non nul qui annule $(u_n)_{n\in\mathbb{N}}$ à l'ordre d, on dit que la suite $(u_n)_{n\in\mathbb{N}}$ *satisfait à une récurrence linéaire d'ordre* d.

Fixons une suite $(u_n)_{n\in\mathbb{N}}$ d'éléments de $\mathbb{K}$ et un entier $d \in \mathbb{N}$. Pour tout entier $t \in \mathbb{N}^*$, on note U_t le polynôme de degré au plus $t-1$ suivant :

$$U_t(X) = \sum_{i=0}^{t-1} u_i X^i.$$

On considère l'ensemble $\mathcal{E}$ des couples $(P, Q) \in \mathbb{K}[X]^2$ tels que :

$$\deg P \leq d, \qquad \deg Q < d \qquad \text{et} \qquad P(X)U_t(X) \equiv Q(X) \bmod X^t \text{ pour tout } t \in \mathbb{N}^*.$$

7. Dans cette question seulement, on considère la suite $(u_n)_{n\in\mathbb{N}}$ définie par $u_0 = 1$ et $u_n = 0$ pour tout $n \geq 1$. Démontrer que cette suite satisfait à une récurrence linéaire d'ordre 1, mais pas à une récurrence linéaire d'ordre 0.
8. Quelles sont les suites qui satisfont à une récurrence linéaire d'ordre 0 ? Même question avec l'ordre 1.
9. Soient $P \in \mathbb{K}_d[X]$, $j \in \mathbb{N}$ et $t \in \mathbb{N}^*$. Exprimer le coefficient de X^j dans le polynôme $P(X)U_t(X)$, et constater que lorsque $j < t$ ce coefficient ne dépend que de j et pas de t.
10. Soit $P \in \mathbb{K}_d[X]$ non nul. Déduire de la question précédente que les propriétés suivantes sont équivalentes :
    (*i*) La suite $(u_n)_{n\in\mathbb{N}}$ est annulée par P à l'ordre d.
    (*ii*) Pour tous $j, t \in \mathbb{N}$ tels que $d \leq j < t$, le coefficient de X^j dans le polynôme $P(X)U_t(X)$ est nul.
    (*iii*) Il existe $Q \in \mathbb{K}[X]$ tel que $(P, Q) \in \mathcal{E}$.
11. Soit $P_1, Q_1, P_2, Q_2 \in \mathbb{K}[X]$ tels que $(P_1, Q_1) \in \mathcal{E}$ et $(P_2, Q_2) \in \mathcal{E}$. Démontrer que $P_1 Q_2 = P_2 Q_1$.

12. Soit $(P, Q) \in \mathcal{E}$ avec $P \neq 0$; notons D le pgcd de P et Q. En faisant apparaître si nécessaire la multiplicité de 0 comme racine de D, démontrer que $(P/D, Q/D) \in \mathcal{E}$.
13. On suppose $\mathcal{E} \neq \{(0,0)\}$. En utilisant les questions précédentes, démontrer les propriétés suivantes :
    (*i*) Il existe un couple $(P_0, Q_0) \in \mathcal{E}$ tel que P_0 soit unitaire et $\text{pgcd}(P_0, Q_0) = 1$.
    (*ii*) Les éléments de $\mathcal{E}$ sont exactement les couples $(P, Q) \in \mathbb{K}_d[X] \times \mathbb{K}_{d-1}[X]$ de la forme $P = P_0 S$ et $Q = Q_0 S$ avec $S \in \mathbb{K}[X]$.
    (*iii*) Le couple $(P_0, Q_0) \in \mathcal{E}$ de l'assertion (*i*) est unique.
14. Dans cette question on suppose que $(u_n)_{n\in\mathbb{N}}$ satisfait à une récurrence linéaire d'ordre d, et on note $(P_0, Q_0) \in \mathcal{E}$ le couple déterminé à la question précédente. Soient $P \in \mathbb{K}_d[X] \setminus \{0\}$ et $Q \in \mathbb{K}_{d-1}[X]$ tels que $P(X)U_{2d}(X) \equiv Q(X) \bmod X^{2d}$. En reprenant les résultats ou les preuves de certaines des questions précédentes, démontrer qu'en posant $D = \text{pgcd}(P, Q)$ on a $P_0 = P/D$ et $Q_0 = Q/D$.
15. Supposons que $(u_n)_{n\in\mathbb{N}}$ satisfait à une récurrence linéaire d'ordre d. En combinant la question précédente aux résultats de la partie 1, proposer un algorithme qui calcule le couple (P_0, Q_0) déterminé à la question 13. Quel est le coût de cet algorithme en opérations arithmétiques dans $\mathbb{K}$?

Partie 3 : Application matricielle

Jusqu'à la question 19 incluse, cette partie est indépendante des deux parties précédentes.

Soit $d \in \mathbb{N}^*$, et $M \in M_d(\mathbb{K})$. On fixe deux matrices colonnes $X, Y \in M_{d,1}(\mathbb{K})$. Pour tout entier $n \in \mathbb{N}$ on considère la matrice M^n et le scalaire $u_n = {}^tXM^nY \in \mathbb{K}$. Dans cette partie les complexités dépendront des deux paramètres d et n, dont on imagine qu'ils sont grands tous les deux.

16. Expliquer brièvement pourquoi le coût du calcul du produit de deux matrices de taille d à coefficients dans $\mathbb{K}$ est $O(d^3)$ opérations arithmétiques dans $\mathbb{K}$.
17. Dans cette question seulement, on s'intéresse à *une seule* valeur donnée de n, et on suppose qu'elle est extrêmement grande par rapport à d. Dans ce cas, comment calculer efficacement M^n puis u_n ? Quel est le coût du calcul de u_n, en nombre d'opérations arithmétiques dans $\mathbb{K}$?
18. On se donne maintenant un entier $n \geq 1$, sur lequel on ne fait plus d'hypothèse particulière. Expliquer comment on peut calculer efficacement les n premiers termes de la suite (u_n). Quel est le coût de ce calcul, en nombre d'opérations arithmétiques dans $\mathbb{K}$?
19. ($\star$) Dans cette question seulement on pose $\mathbb{K} = \mathbb{F}_{37}$, $d = 3$, et on note M, X et Y les matrices suivantes (dans lesquelles les coefficients entiers sont évidemment vus modulo 37) :

$$M = \begin{bmatrix} 2 & 3 & 14 \\ 5 & 26 & 11 \\ 4 & 13 & 8 \end{bmatrix}, \qquad X = \begin{bmatrix} 1 \\ 3 \\ 7 \end{bmatrix} \qquad \text{et} \qquad Y = \begin{bmatrix} 28 \\ 19 \\ 10 \end{bmatrix}.$$

Ecrire une fonction Sage qui prend en entrée un entier n et renvoie la liste formée par u_0, u_1, …, u_n. Combien vaut u_{100} ? *Dans cette question on a le droit d'utiliser toutes les fonctions implémentées en Sage.*

20. Démontrer que la suite (u_n) satisfait à une récurrence linéaire d'ordre d (au sens défini au début de la partie 2).
21. Notons μ_M le polynôme minimal de la matrice M, et (P_0, Q_0) le couple construit à la question 13 (puisque (u_n) satisfait à une récurrence linéaire d'ordre d). Démontrer que $X^{\deg P_0} P_0(1/X)$ divise μ_M.

Dans toute la fin du problème on suppose (pour simplifier) que la matrice $M \in M_d(\mathbb{K})$ est inversible.

22. Si $\mathbb{K}$ est infini, on peut démontrer qu'en choisissant au hasard les vecteurs colonnes X et Y on aura avec probabilité 1 l'égalité $\mu_M(X) = cX^{\deg P_0} P_0(1/X)$ pour un certain $c \in \mathbb{K}^*$. Donner un argument intuitif qui corrobore ce résultat. *Cette question est volontairement floue. Une preuve de ce résultat n'est absolument pas attendue. Cette question est vue comme un bonus : ne pas y répondre, ou mal y répondre, ne sera pas pénalisé.*
23. En admettant que $\mu_M(X) = cX^{\deg P_0} P_0(1/X)$ pour un certain $c \in \mathbb{K}^*$, s'appuyer sur les questions précédentes pour proposer un algorithme qui calcule le polynôme minimal d'une matrice $M \in M_d(\mathbb{K})$. Déterminer le coût de cet algorithme, en nombre d'opérations arithmétiques dans $\mathbb{K}$. Comparer ce coût à celui de l'algorithme naïf qui consiste à chercher des relations linéaires entre les M^k, $0 \leq k \leq d$.
24. ($\star$) Mettre en œuvre l'algorithme de la question précédente sur l'exemple de la question 19. *Pour faire de l'algèbre linéaire en Sage il est conseillé, si besoin, d'utiliser les méthodes* `MatrixSpace`, `VectorSpace`, `right_kernel` *et* `basis`. En admettant que $\mu_M(X) = cX^{\deg P_0} P_0(1/X)$ pour un $c \in \mathbb{K}^*$, en déduire le polynôme minimal de M.
25. On suppose maintenant que la plupart des coefficients de la matrice M sont nuls. Précisément, on note r le nombre de coefficients non nuls de M. On suppose que $r \geq d$, et que r est beaucoup plus petit que d^2 (par exemple r pourrait être de l'ordre de grandeur de d ou de $d \log d$, mais on ne fera pas d'hypothèse restrictive de ce type et on calculera les complexités en fonction de r et de d). L'algorithme de la question 23 sera alors beaucoup plus efficace : calculer le produit de deux éléments de $\mathbb{K}$ dont l'un au moins vaut 0 est immédiat, et ne doit pas être compté comme une opération arithmétique dans $\mathbb{K}$. De même, additionner un terme nul ne compte pas non plus. Enfin, on admet qu'il existe un algorithme permettant, pour A et d donnés, de trouver en $O(d^2)$ opérations arithmétiques dans $\mathbb{K}$ un couple (P, Q) vérifiant la relation (1) de la partie 1. Déterminer alors, en fonction de r et d, le coût en nombre d'opérations arithmétiques dans $\mathbb{K}$ de l'algorithme de la question 23.
26. Sachant que d'autres algorithmes existent qui permettent de calculer le polynôme minimal de M en $O(d^3)$ opérations arithmétiques dans $\mathbb{K}$, que peut-on dire de l'intérêt de l'algorithme développé dans ce sujet ?