

# COURS DE CALCUL FORMEL EN M1 : TP 5

## UN GROS DÉTERMINANT

Soit  $A_n$  la matrice  $n \times n$  dont les coefficients sont tous nuls sauf les nombres premiers 2,3,5,7,... sur la diagonale et le nombre 1 dans toutes les positions  $a_{ij}$  où  $|i - j| = 1, 2, 4, 8, \dots$ . L'objectif de ce TP<sup>1</sup> est le calcul du déterminant de  $A_n$ .

### 1. APPROCHE DIRECTE

- (1) Écrire une procédure prenant en entrée  $n$  et le type des coefficients souhaité (`integer` ou `float`) et renvoyant la matrice  $A_n$ . La matrice doit être créée en un nombre d'opérations *linéaire* en  $n$ .
- (2) Estimer empiriquement la complexité de la procédure `LinearAlgebra[Determinant]` sur ces matrices en comparant les temps de calculs pour  $|A_k|$  et  $|A_{2k}|$  pour des valeurs de  $k$  bien choisies en fonction du type des coefficients.

### 2. LA MÉTHODE DE WIEDEMANN

La méthode de Wiedemann calcule le polynôme minimal de matrices pour lesquelles le produit matrice-vecteur est moindre que quadratique, à l'aide d'un approximant de Padé. Lorsque (comme c'est fréquemment le cas) le polynôme minimal est aussi le polynôme caractéristique, le déterminant est obtenu comme terme constant au signe près.

Pour ce calcul, la méthode commence par tirer aléatoirement deux vecteurs  $v$  et  $w$ , puis calcule la suite de scalaires  $u_i := wA^i v$  pour  $i = 0, \dots, 2n$ . Ce calcul n'utilise que des produits scalaires et des multiplication de  $A$  par des vecteurs. Si le polynôme minimal est  $\chi(A) = t^n + c_1 t^{n-1} + \dots + c_n$ , alors la suite  $u_i$  vérifie la récurrence

$$u_{i+n} + c_1 u_{i+n-1} + \dots + c_n u_i = 0, \quad i \geq 0.$$

Dans un second temps, la méthode reconstruit cette récurrence à partir des  $2n + 1$  premières valeurs de  $u_i$  par un approximant de Padé de type  $(n - 1, n)$  de la série  $S_n = u_0 + \dots + u_{2n} z^{2n} + O(z^{2n+1})$ .

En toute généralité, pour de mauvais choix de  $v$  et  $w$ , il est possible (mais peu probable) d'obtenir un diviseur de  $\chi(A)$  et non  $\chi(A)$  lui-même. Dans ce cas, il suffit de reprendre le calcul avec un autre choix de  $(v, w)$ . Le résultat est correct si le degré obtenu est  $n$ .

- (3) Écrire une procédure qui prend en entrée  $A$  et une variable  $z$  et calcule la série tronquée  $S_n$  correspondante. (On obtient un vecteur aléatoire par `LinearAlgebra[RandomVector]`).
- (4) Estimer empiriquement la complexité de ce calcul.
- (5) Calculer le déterminant de la matrice  $A_{10}$  par la méthode de Wiedemann sur la matrice  $A_{10}$ , d'abord en flottants, puis en entiers.

### 3. CALCULS EXACTS

Pour des raisons de stabilité numérique, la méthode de Wiedemann est réservée au calcul exact (soit sur des entiers, soit sur des entiers modulaires).

- (6) Écrire une procédure qui prend en entrée les  $2n$  premiers coefficients d'une série tronquée ainsi qu'un entier  $n$  et renvoie le dénominateur d'un approximant de Padé de type  $(n - 1, n)$  calculé par l'algorithme d'Euclide étendu. Dans un premier temps on ne se préoccupera pas du type des coefficients.
- (7) Comparer cette implantation à `numapprox[pade]` pour calculer  $|A_{10}|$ .

---

<sup>1</sup>Il s'agit d'une variante d'un des problèmes étudiés dans le livre *The SIAM 100-Digit Challenge*, SIAM Press, 2004.

#### 4. CALCULS MODULAIRES EXACTS

Pour éviter de calculer des entiers intermédiaires trop gros, il est plus efficace de calculer le déterminant modulo plusieurs nombres premiers et de le reconstruire par le théorème des restes chinois. Il faut pour cela disposer d'une borne facile à calculer sur ce déterminant, et on pourra prendre le produit  $B$  des éléments diagonaux. Il suffit alors de disposer de nombres premiers  $(p_1, \dots, p_k)$  tels que  $p_1 \cdots p_k \geq B$ . Pour rendre les calculs efficaces, ils doivent être assez gros (pour diminuer  $k$ ) mais pas trop (pour que les calculs tiennent dans un mot mémoire). Typiquement on prend  $p_1$  le plus grand nombre premier inférieur à  $2^{32}$  et  $p_2, p_3, \dots$  les précédents.

- (8) Récrire la procédure de calcul d'approximants de Padé pour des coefficients modulaires en utilisant les opérations fournies par le package `modp1`.
- (9) Estimer empiriquement la complexité de ce calcul d'approximants de Padé.
- (10) Écrire la procédure générale de calcul de déterminant modulaire, en calculant la série sur les entiers, puis les images modulaires du déterminant et en reconstruisant ce déterminant (par `chrem`); estimer sa complexité empirique, et conclure.