TD 3 : Multiplication de polynômes

Exercice 1. *Résolution de récurrence*

Soit T définie (sur les entiers) par $T(1) = T_1 \geq 0$ et $T(n) \leq aT(\lceil n/b \rceil) + O(n)$. On suppose T croissante et $b > 1$.

1. Borner, pour $k \geq 1$, $T(n)$ en fonction de $T(n/b^k)$ (pour n multiple de b^k).
2. Montrer que si $a > b$, $T(n) = O(n^{\log_b a})$.
3. Montrer que si $a = b$, $T(n) = O(n \log n)$.
4. Montrer que si $a < b$, $T(n) = O(n)$.

Exercice 2. *Algorithme Toom-3*

L'objectif de l'exercice est de voir une généralisation de l'algorithme de Karatsuba.

1. Montrer qu'on peut adapter les formules d'interpolation de Lagrange au cas où l'un des évalués est remplacé par la valeur du coefficient dominant du polynôme (« évaluation en ∞ »).
2. Montrer qu'on peut multiplier deux polynômes de degré 2 avec 5 multiplications, en utilisant un schéma « évaluation – interpolation » sur les 5 points 0, 1, -1, 2 et ∞.
3. Généraliser le résultat précédent à des polynômes de degré quelconque d, et analyser la complexité de l'algorithme obtenu.
   Indication : si $P = P_0 + X^{n/3}P_1 + X^{2n/3}P_2$, définir $P^\star(X,Y) = P_0(X) + YP_1(X) + Y^2P_2(X)$ et appliquer l'algorithme précédent à $P^\star$. On pourra supposer que $n = d + 1$ est une puissance de 3.

Exercice 3. *Variante de la FFT*

L'algorithme de transformée de Fourier rapide admet deux variantes, appelées *décimation en temps* et *décimation en fréquence*. L'objectif de cet exercice est de voir la seconde variante, non présentée en cours. L'idée de base est la suivante : pour calculer la transformée de Fourier discrète d'un polynôme P sur une racine ω de taille $n = 2^k$, on coupe P sous la forme $P_b(X) + X^{n/2}P_h(X)$. On calcule la transformée de Fourier discrète des deux polynômes $P_b + P_h$ et $P_b - P_h$ sur la racine ω^2. À partir de ces deux résultats, on construit la transformée de Fourier discrète de P.

✎ Compléter l'algorithme esquissé et l'écrire en détail. Analyser sa complexité.