TD 4 – Transformée de Fourier rapide

Margot Catinaud margot.catinaud@lmf.cnrs.fr
Valentin Dardilhac valentin.dardilhac87@gmail.com

I Opérations sur les polynômes

On considère deux ensembles $X, Y \subset \mathbb{N}$, contenant chacun $n \in \mathbb{N}$ entiers compris entre 0 et $10n$. On souhaite calculer la somme cartésienne $\Gamma = X \oplus Y$ de X et Y définie par

$$\Gamma \overset{\text{def}}{=} \Big\{ x + y \mid x \in X, y \in Y \Big\}.$$

On veut trouver les éléments de Γ et le nombre de fois que chaque élément de Γ est obtenu comme somme d'éléments de X et Y.

Question 1 *Montrer que l'on peut résoudre ce problème en temps quasi-linéaire* $\Theta(n \log n)$.

Question 2 *Déduire une représentation par valeurs de* $\widetilde{P}(X) \overset{def}{=} \sum_{i=0}^{n-1} a_{n-1-i} X^i$ *à partir d'une représentation par valeurs de* $P(X) \overset{def}{=} \sum_{i=0}^{n-1} a_i X^i$*, en supposant qu'aucun des points n'est* 0.

Question 3 *Étant donnée une liste* $(z_i)_{i=0}^{n-1}$ *de valeurs complexes dans* $\mathbb{C}$ *(avec répétitions possibles), montrer comment trouver les coefficients d'un polynôme* $P \in \mathbb{C}[X]$ *de degré borné par* n *qui s'annule uniquement sur les points donnés. La procédure trouvée devra s'exécuter en temps* $\Theta(n \log^2 n)$.

On suppose disposer d'un algorithme efficace calculant la division euclidienne de deux polynômes. Plus précisément, étant donné $P, Q \in \mathbb{C}[X]$ deux polynômes de degrés inférieurs ou égaux à $n-1$, on sait calculer le reste de la division euclidienne de P par Q, noté $P \mod Q$, en temps $\Theta(n \log n)$.

Question 4 *En remarquant très judicieusement que, pour tout* $z \in \mathbb{C}$, $P(z) = P \mod (X - z)$*, montrer que l'on peut calculer* $\big(P(z_i)\big)_{i=0}^{n-1}$ *en temps* $\Theta(n \log^2 n)$.

II Algorithme FFT itératif, parallèle

On cherche à donner une version itérative de l'algorithme récursif FFT (pour l'image des racines de l'unité par un polynôme) vu en cours.

Question 5 *Dessiner l'arbre des appels récursifs de l'algorithme* FFT *pour un polynôme* P *de degré* 7*, avec* $P = (a_i)_{i=0}^{7}$.

Question 6 *Écrire un algorithme* Reordonne *prenant en entrée une liste de* n *coefficients (représentant un polynôme de degré* $n-1$*) et qui renvoie cette liste triée en fonction de l'ordre d'apparition des feuilles de l'arbre des appels récursifs de* FFT *sur* P*. L'algorithme doit s'exécuter en temps* $\Theta(n)$.

Par exemple, pour les coefficients $P \overset{def}{=} (a_i)_{i=0}^{7}$ *de la question précédente, le résultat de* Reordonne(P) *doit être* $(a_0, a_4, a_2, a_6, a_1, a_5, a_3, a_7)$.

Question 7 *En déduire un algorithme itératif* $\mathsf{FFT}_{\mathsf{it}}$*. Montrer que sa complexité est la même que la version récursive.*

Question 8 *Montrer qu'en utilisant* $n/2$ *processeurs en parallèle, le calcul peut s'effectuer en temps logarithmique* $\Theta(\log n)$ *(une fois que la liste des coefficients a été réordonnée).*

III Calcul des n premières dérivées d'un polynôme en un point

Étant donnée la représentation d'un polynôme $P \in \mathbb{C}[X]$ par ses coefficients $(a_i)_{i=0}^{n-1}$ et un point $x_0 \in \mathbb{C}$, on souhaite déterminer $P^{(i)}(x_0)$ la dérivée i-ème de P en x_0, pour tout $i \in [\![0; n-1]\!]$.

Question 9 *Connaissant les coefficients $(b_i)_{i=0}^{n-1}$ tels que*

$$P(X) = \sum_{i=0}^{n-1} b_i(X - x_0),$$

montrer comment calculer les dérivées successives $P^{(i)}(x_0)$ pour tout $i \in [\![0; n-1]\!]$ en temps linéaire $\Theta(n)$.

Question 10 *On suppose connu l'ensemble $\left(P(x_0 + \omega^i)\right)_{i=0}^{n-1}$ et ω la racine primitive n-ème de l'unité. Expliquer comment trouver la séquence $(b_i)_{i=0}^{n-1}$ de la question précédente en temps quasi-linéaire $\Theta(n \log n)$.*

Question 11 *Démontrer que l'on a :*

$$\forall\, i \in [\![0; n-1]\!],\ P(x_0 + \omega^i) = \sum_{l=0}^{n-1} \frac{\omega^{il}}{l!} \left(\sum_{j=l}^{n-1} f(j) g(j-l) \right),$$

où, pour tout $l \in [\![0; n-1]\!]$ et $j \in [\![l; n-1]\!]$, $f(j) \overset{def}{=} a_j j!$ et $g(l) = \frac{(x_0)^l}{l!}$.

Question 12 *Expliquer comment évaluer $P(x_0 + \omega^i)$ pour $i \in [\![0; n-1]\!]$ en temps quasi-linéaire $\Theta(n \log n)$. On pourra appliquer plusieurs fois l'algorithme* `FFT` *et essayer d'exprimer $P(x_0 + \omega^i)$ à l'aide d'un produit de polynômes.*

Question 13 *Que peut-on en conclure ?*

IV Exemple d'application de la transformée de Fourier rapide – Filtrage

Un dispositif physique tel qu'un microphone, un oscilloscope, etc., est utilisé pour acquérir une suite de réels $(x_i)_{i=0}^{n}$ "assez longue". Lorsque le dispositif n'est pas de très bonne qualité et que les échantillons contiennent du "bruit", une mesure rudimentaire pour lutter contre ce phénomène consiste à appliquer un lissage gaussien, c'est-à-dire à remplacer chaque échantillon par une moyenne pondérée de ses voisins :

$$y_i \overset{\text{def}}{=} \frac{1}{Z} \sum_{j=-l}^{l} x_{i+j} e^{-j^2}$$

où l est un paramètre de largeur et Z un facteur de normalisation choisit convenablement. On remarque qu'il y a un problème pour les valeurs limites (*i.e.* lorsque $i - l < 0$ ou $i + l > n$). Une manière de résoudre ce problème est de calculer y_i uniquement pour $i \in [\![l; n-l]\!]$.

Devoir maison 4 *Montrer comment calculer la suite $(y_i)_{i=l}^{n-l}$ en temps quasi-linéaire $\Theta(n \log n)$.*