

TRANSFORMÉES DE FOURIER DISCRÈTES ET SÉRIES DE FOURIER

PARTIE FACULTATIVE

1. Comparaison des transformations de Fourier discrètes et rapides

Question 1

Montrer que, si l'on pose,

$$X = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} \text{ et } Y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \end{pmatrix},$$

alors les transformées de Fourier discrète directe et inverse sont définies par

$$Y = \text{tfd}_N(X) \iff Y = AX, \quad (1a)$$

et

$$X = \text{tfdinv}_N(Y) \iff X = \frac{1}{N} BY, \quad (1b)$$

où les matrices A et B de $\mathcal{M}_n(\mathbb{C})$ sont définies par

$$\forall (k, l) \in \{1, \dots, N\}, \quad A_{k,l} = e^{-\frac{2i(k-1)(l-1)\pi}{N}} \text{ et } B_{k,l} = e^{\frac{2i(k-1)(l-1)\pi}{N}}. \quad (2)$$

Question 2

Quel est le lien entre les matrices A et B ?

Pour utiliser les produits de vecteurs de matlab, afin d'éviter les boucles qui sont lentes, on utilisera les formules (1) en remarquant que (2) est équivalente à

$$A = \exp\left(-\frac{2i\pi}{N} \text{t}UU\right) \text{ et } B = \exp\left(\frac{2i\pi}{N} \text{t}UU\right),$$

où

$$U = (0\ 1\ 2\ \dots\ N - 1).$$

Question 3

Programmer deux fonctions matlab **fftman**(X) et **ifftman**(Y) qui aux vecteurs X et Y associent respectivement les vecteurs $tfd_N(X)$ et $tfdinv_N(Y)$. On comparera ces deux fonctions aux fonctions déjà programmées de matlab **fft** et **ifft**, en terme de nombres d'opérations et de temps de calcul en utilisant les instructions *tic*, *toc* et *flops*.

2. Écriture des algorithmes de passage des valeurs ponctuelles d'une fonction à ses coefficients de Fourier pour des valeurs paires de N .

L'inconvénient de la méthode proposée dans le TP est de ne s'appliquer qu'à des vecteurs de taille N impair. Cependant, les transformées de Fourier sont plus rapides si on les applique à des vecteurs de taille $N = 2^p$, où $p \in \mathbb{N}$. Nous montrons dans cette section comment modifier les deux algorithmes du TP déjà vu avec une taille N de vecteur paire.

Soient $n \in \mathbb{N}$, $N = 2n$ et $(\alpha_l)_{0 \leq l \leq N-1}$. On cherche f sous la forme

$$\forall t \in [0, 2\pi[, \quad f(t) = \sum_{k=-n}^n c_k e^{ikt},$$

telle que

$$\forall l \in \{0, \dots, N - 1\}, \quad f\left(\frac{2\pi l}{N}\right) = \alpha_l.$$

Le nombre des constantes réelles α_l est égal à N . On peut montrer que l'on peut choisir $c_n = c_{-n}$ réel, que c_0 est réel et que, pour tout $k \in \{1, \dots, n - 1\}$, $c_{-k} = \overline{c_k}$. Ainsi, il y a bien $N = 2n$ réels définissant la fonction f .

On admet que l'algorithme de calcul définissant les complexes c_k à partir des valeurs α_l est :

$$(\alpha_l)_{0 \leq l \leq N-1} \longrightarrow (d_k)_{0 \leq k \leq N-1} = tfdinv_N\left((\alpha_l)_{0 \leq l \leq N-1}\right) \longrightarrow \begin{cases} c_0 = \text{Re}(d_0), \\ c_n = \text{Re}(d_n), \\ \forall k \in \{1, \dots, n - 1\}, \quad c_k = \overline{d_k}. \end{cases} \quad (3)$$

Question 4

À partir de l'algorithme (3), écrire une fonction matlab qui, à un vecteur de N réels, contenant les valeurs $(\alpha_l)_{0 \leq l \leq N-1}$, renvoie un vecteur de $n + 1$ complexes, correspondant aux valeurs $(c_k)_{0 \leq k \leq n}$, avec $N = 2n + 1$ ou $N = 2n$.

On admet que l'algorithme de calcul définissant les valeurs α_l à partir des complexes c_k est :

$$(c_k)_{0 \leq k \leq n} \text{ (où } c_0, c_n \in \mathbb{R}) \longrightarrow \begin{cases} d_0 = c_0, \\ d_n = c_n, \\ \forall k \in \{1, \dots, n-1\}, & d_k = \overline{c_k}, \\ \forall k \in \{n+1, \dots, 2n-1\}, & d_k = c_{2n-k}, \end{cases} \\ \longrightarrow (\alpha_l)_{0 \leq l \leq N-1} = \text{tfd}_N \left((d_k)_{0 \leq k \leq N-1} \right). \quad (4)$$

Question 5

À partir de l'algorithme (4), écrire une fonction matlab qui, à un un vecteur de $n+1$ complexes, correspondant aux valeurs $(c_k)_{0 \leq k \leq n}$, renvoie un vecteur de N réels, contenant les valeurs $(\alpha_l)_{0 \leq l \leq N-1}$ avec $N = 2n$ ou $N = 2n + 1$.