

TRANSFORMÉE DE FOURIER DISCRÈTE

G. BAUDOIN et J.-F. BERCHER

École Supérieure d'Ingénieurs en Électrotechnique et Électronique

Novembre 2001 – version 0.1

CHAPTER I

Table des matières

I	Table des matières	3
I	Transformée de Fourier discrète: TFD et TFR	5
1	Transformée de Fourier Discrète: TFD	5
1.1	Définition de la TFD	5
1.2	Inversion de la TFD	6
1.3	Lien entre la transformée de Fourier et la TFD	6
1.4	Comparaison entre la transformée de Fourier et la TFD	8
1.5	Fenêtres de pondération	9
1.5.1	Fenêtres rectangulaires, triangulaires et paraboliques	10
1.5.2	Fenêtres Fenêtres détruisant par addition algébrique, les lobes secondaires de la fenêtre rectangulaire	11
1.5.3	Autres fenêtres: Gauss, Kaiser, Dolph-Chebyshev	12
1.6	Problèmes de visualisation de la TFD	14
1.7	Propriétés de la TFD et convolution circulaire	14
1.7.1	Théorème de Parseval	14
1.7.2	Théorème de la convolution discrète	15
1.7.3	Théorème du retard circulaire	16
2	Transformée de Fourier Rapide TFR, Fast Fourier transform FFT	17
2.1	FFT avec entrelacement temporel	17
2.2	FFT avec entrelacement fréquentiel	20
2.3	Bit reversal	22
2.4	Formulation matricielle de l'algorithme de Cooley-Tukey	22
2.5	Autres algorithmes de FFT	25
2.6	Utilisation de la FFT pour la convolution rapide	25
2.7	Calcul de convolution par section d'une des suites	26
	Exercices et problèmes	28

CHAPTER I

TRANSFORMÉE DE FOURIER DISCRÈTE: TFD ET TFR

LORSQU'ON désire calculer la transformée de Fourier d'une fonction $x(t)$ à l'aide d'un ordinateur, ce dernier n'ayant qu'un nombre fini de mots de taille finie, on est amené à:

- discrétiser la fonction temporelle,
- tronquer la fonction temporelle,
- discrétiser la fonction fréquentielle.

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt$$

En approchant l'intégrale par une somme d'aires de rectangles de durée T_e et en limitant la durée d'intégration à l'intervalle $[0, (N-1)T_e]$, on obtient:

$$X(f) \approx T_e \sum_{n=0}^{(N-1)} x(nT_e)e^{-j2\pi fnT_e}$$

Ce qui donne pour les valeurs de fréquences $f_k = kf_e/N$:

$$X(f_k) \approx T_e \sum_{n=0}^{(N-1)} x(nT_e)e^{-j2\pi \frac{nk}{N} f_e T_e} \approx T_e \sum_{n=0}^{(N-1)} x(nT_e)e^{-j2\pi \frac{nk}{N}}$$

Ce n'est pas une approximation sophistiquée de $X(f)$, mais elle est très utilisée en pratique sous le nom de TFD car il existe un algorithme de calcul efficace appelé FFT (Fast Fourier Transform) ou TFR (Transformée de Fourier rapide).

La TFD est par ailleurs utilisée, lorsque l'on travaille avec des suites numériques sans lien avec un signal physique, pour définir une représentation de la suite sur une base de fonctions fréquentielles.

1 Transformée de Fourier Discrète: TFD

1.1 Définition de la TFD

On appelle transformée de Fourier discrète d'une suite de N termes $x(0), x(1), \dots, x(N-1)$, la suite de N termes $X(0), X(1), \dots, X(N-1)$, définis par

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi \frac{nk}{N}}$$

En pratique, les N termes $x(n)$ peuvent être N échantillons d'un signal analogique échantillonné:

$x_n = x(nT_e)$, et les N termes $X(k)$ correspondre à une approximation (à un facteur multiplicatif T_e près) de la transformée de Fourier de ce signal aux N points de fréquence $f_k = kf_e/N$, avec k entre 0 et $N-1$, c'est à dire f entre 0 et f_e .

1.2 Inversion de la TFD

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi \frac{nk}{N}}$$

En effet, calculons:

$$\begin{aligned} A &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi \frac{nk}{N}} = \frac{1}{N} \sum_{k=0}^{N-1} \left(\sum_{i=0}^{N-1} x(i) e^{-j2\pi \frac{ik}{N}} \right) e^{j2\pi \frac{nk}{N}} \\ A &= \frac{1}{N} \sum_{i=0}^{N-1} x(i) \left(\sum_{k=0}^{N-1} e^{j2\pi \frac{(n-i)k}{N}} \right) \\ \text{si } i \neq n & \sum_{k=0}^{N-1} e^{j2\pi \frac{(n-i)k}{N}} = \frac{1 - e^{j2\pi (n-i)}}{1 - e^{j2\pi \frac{n-i}{N}}} = 0 \\ \text{si } i = n & \sum_{k=0}^{N-1} e^{j2\pi \frac{(n-i)k}{N}} = \sum_{k=0}^{N-1} 1 = N \\ A &= \frac{1}{N} \sum_{i=0}^{N-1} x(i) \left(\sum_{k=0}^{N-1} e^{j2\pi \frac{(n-i)k}{N}} \right) = \frac{1}{N} x(n) N \\ A &= x(n) \quad \text{c.q.f.d.} \end{aligned}$$

1.3 Lien entre la transformée de Fourier et la TFD

Soit $x(t)$ un signal analogique continu.

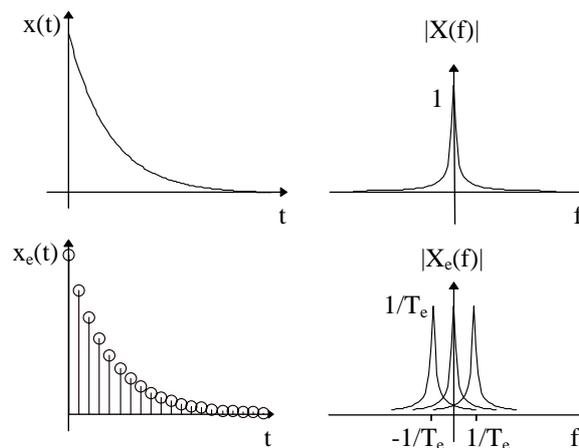
1. On échantillonne $x(t)$ à $f_e = 1/T_e$.

$$x(t) \rightarrow x_e(t) = \sum_{n=-\infty}^{+\infty} x(nT_e) \delta(t - nT_e) = x(t)P(t)$$

où $P(t)$ est la « fonction peigne »:

$$P(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) \stackrel{TF}{\Leftrightarrow} P(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} \delta\left(f - \frac{n}{T_e}\right)$$

L'échantillonnage rend le spectre périodique et peut entraîner un phénomène de « recouvrement de spectre » ou *aliasing*.



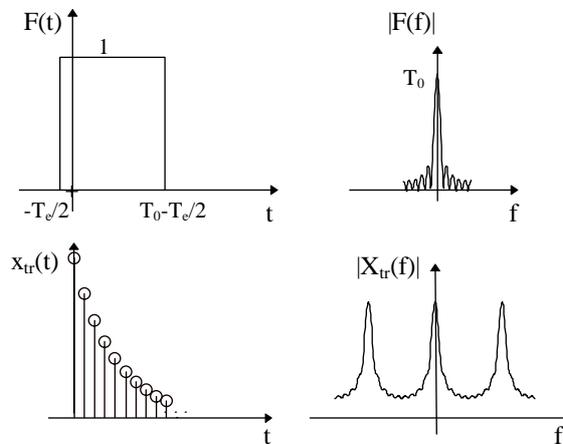
2. On tronque la suite $x_e(nT_e)$ en ne conservant qu'un nombre fini N de termes pour obtenir le signal $x_{tr}(t)$ formé des échantillons: $x(0) \dots x((N-1)T_e)$:

$$\begin{aligned} x_{tr}(t) &= x_e(t)F(t) = \sum_{n=0}^{N-1} x(nT_e) \delta(t - nT_e) \\ x_{tr}(t) &= x(t)P(t)F(t) \end{aligned}$$

où $F(t)$ est une fonction fenêtre de durée NT_e

$$F(t) = \begin{cases} 1 & \text{si } t \in [-\frac{T_e}{2}, T_0 - \frac{T_e}{2}] \\ 0 & \text{sinon} \end{cases}$$

où $T_0 = NT_e$.



La convolution avec un sinus cardinal introduit des ondulations sur le spectre. Elles sont appelés « ripples » en anglais.

$$X_{tr}(f) = \sum_{n=0}^{N-1} x(nT_e)e^{-j2\pi fnT_e}$$

3. On échantillonne $X_{tr}(f)$ à $1/T_0$

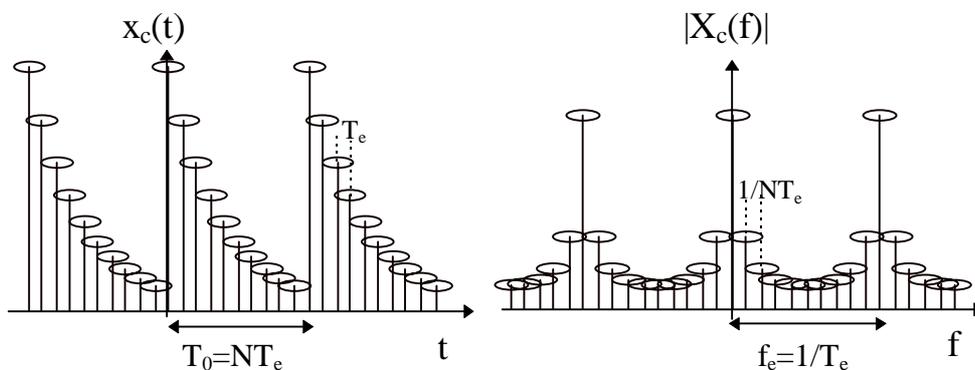
On obtient alors N valeurs différentes espacées de $1/T_0$ entre 0 et $1/T_e$, car $T_0 = NT_e$. Cette dernière opération rend périodique la « fonction » dans le temps. Appelons $x_c(t)$ la fonction résultante.

$$X_c(f) = X_{tr}(f) \sum_{n=-\infty}^{+\infty} \delta\left(f - \frac{n}{T_0}\right) = \sum_{n=-\infty}^{+\infty} X_{tr}\left(\frac{n}{T_0}\right) \delta\left(f - \frac{n}{T_0}\right)$$

$$X_c(f) = \sum_{n=-\infty}^{+\infty} \left(\sum_{k=0}^{N-1} x(kT_e)e^{-j2\pi \frac{nk}{N}} \right) \delta\left(f - \frac{n}{T_0}\right)$$

$$x_c(t) = T_0 \sum_{n=-\infty}^{+\infty} x_{tr}(t - nT_0)$$

$x_c(t)$ et $X_c(f)$ sont deux distributions échantillonnées reliées par la transformation de Fourier.



On obtient donc une correspondance entre N points dans le domaine temporel $x(nT_e)$ et N points dans le domaine fréquentiel $X_c(n/T_0)$, pour n entre 0 et $N - 1$. De plus:

$$x_c(nT_e) = T_0 x(nT_e) \quad \text{pour } n \in [0, N - 1]$$

$$X_c\left(\frac{k}{T_0}\right) = \sum_{n=0}^{N-1} x(nT_e)e^{-j2\pi \frac{nk}{N}}$$

c'est-à-dire que la suite $X_c(k) = X_c(k/T_0)$ est précisément la TFD de la suite $x(n) = x(nT_e)$.

1.4 Comparaison entre la transformée de Fourier et la TFD

Soit un signal $x(t)$ et sa transformée de Fourier $X(f)$.

à la suite $x(nT_e)$ pour $n \in [0, (N-1)]$ correspond la suite TFD $X(k)$ pour $k \in [0, (N-1)]$ avec:

$$X(k) = \sum_{n=0}^{N-1} x(nT_e) e^{-j2\pi \frac{nk}{N}}$$

Quel est le lien entre $X(f)$ et $X(k)$ pour $k \in [0, (N-1)]$?

On pose $T_0 = NT_e$.

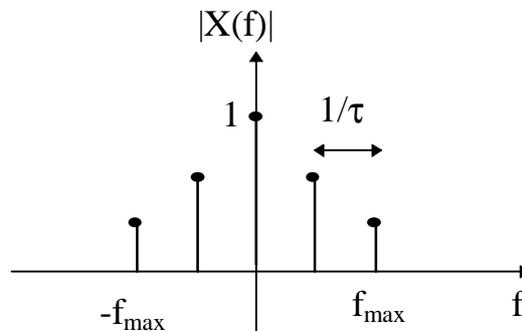
Dans le cas où:

- $x(t)$ est périodique de période τ ,
- $x(t)$ est à bande limitée $[-f_{max}, f_{max}]$,
- la largeur de la fenêtre $F(t)$ est égale à un multiple de τ , $T_0 = k\tau$,
- et la fréquence d'échantillonnage $f_e = 1/T_e$ est supérieure à $2 f_{max}$.

Il y a coïncidence à un facteur multiplicatif près entre $X(k)$ et $X(f = k/T_0)$:

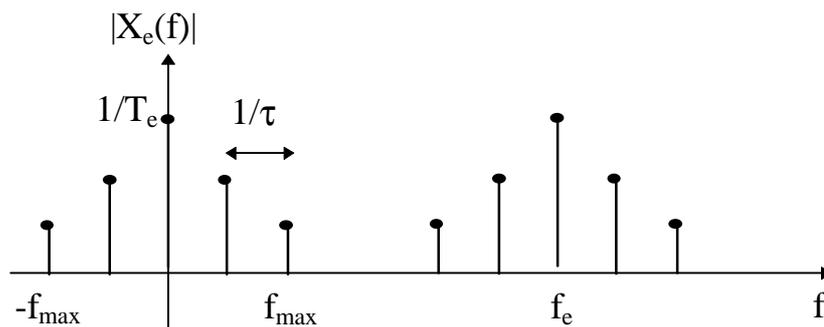
$$X(k) = \frac{T_0}{T_e} X\left(\frac{k}{T_0}\right) = NX\left(\frac{k}{T_0}\right)$$

En effet, $x(t)$ étant périodique de période τ a un spectre formé de raies distantes de $1/\tau$. De plus, ce spectre est limité à f_{max} .

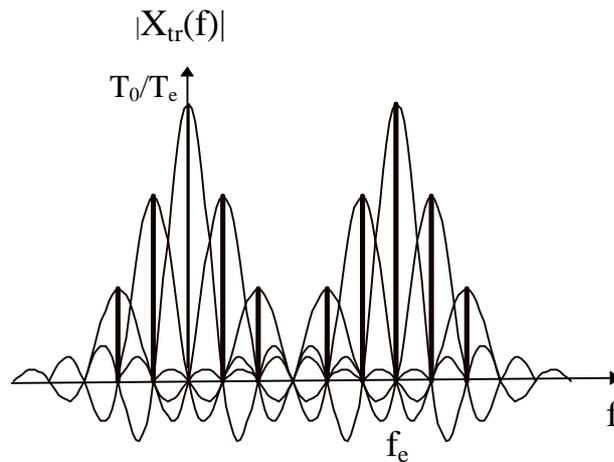


Les trois opérations qui conduisent à la suite $X(k)$ auront les conséquences suivantes:

1. L'échantillonnage de $x(t)$ à f_e rend périodique le spectre et le multiplie par $1/T_e$.



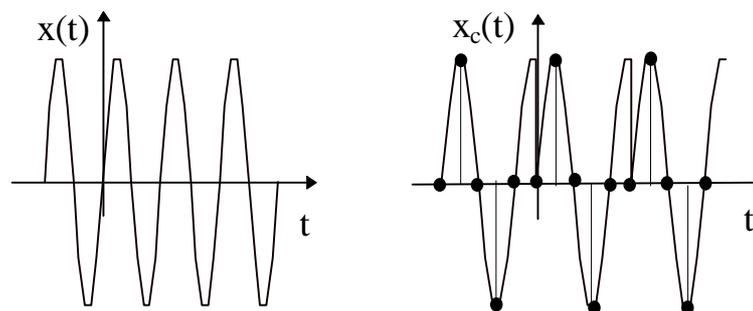
2. La troncation de $x_e(t)$ par une fenêtre de largeur T_0 a pour effet de convoluer le spectre avec un sinus cardinal qui s'annule tous les $1/T_0$ avec $T_0 = k\tau$.



3. L'échantillonnage du spectre à la fréquence $1/T_0$ a pour effet de ne conserver que des valeurs où X_r et X coïncident au facteur $T_0/T_e = N$ près. C'est le seul cas où il y a identité entre la TFD et la TF au facteur N près, aux N points de calcul k/T_0 avec $k \in [0, (N-1)]$.

Dans tous les autres cas, la TFD diffère de la TF aux points k/T_0 . L'erreur est introduite:

- par recouvrement de spectre si $X(f)$ n'est pas à support limité, erreur que l'on minimise en augmentant f_e .
- par les ondulations dues à la troncature par la fonction fenêtre si $x(t)$ n'est pas périodique ou à durée limitée: erreur que l'on peut chercher à atténuer en choisissant une fenêtre autre que la fenêtre rectangulaire (fenêtre de Hanning par exemple) et en augmentant autant que possible la largeur de la fenêtre.
- pour les deux premières raisons à la fois si $x(t)$ n'est ni à durée limitée ni à bande limitée
- même si $x(t)$ est périodique et à bande limitée, on introduit une erreur si la fenêtre de troncature n'a pas une durée égale à un multiple de la période car la troncature introduit alors de fortes discontinuités (voir la figure suivante).



1.5 Fenêtres de pondération

De nombreuses fenêtres de pondération ont été proposées pour l'analyse spectrale. Ces fenêtres sont utilisées pour limiter (tronquer) la durée temporelle du signal $x(t)$ à analyser. En notant $x(t)$ le signal, $F(t)$ la fenêtre, et $x_{tr}(t)$ le signal tronqué, on obtient la relation suivante:

$$x_{tr}(t) = x(t)F(t)$$

et dans le domaine fréquentiel:

$$X_{tr}(f) = X(f) \star F(f)$$

Pour une même durée temporelle NT_e , on compare les différentes fenêtres essentiellement par leurs propriétés fréquentielles. Idéalement, on aimerait que la troncation du signal en temps ne modifie pas son contenu fréquentiel, c'est-à-dire que $X(f) = X_r(f)$, ce qui suppose que $F(f) = \delta(f=0)$. En pratique, ce n'est pas possible et les fenêtres $F(f)$ présentent un lobe principal de largeur non nulle centré autour de la fréquence nulle et en général des lobes secondaires de hauteur non nulle. On peut caractériser une fenêtre par des paramètres tels que:

- La largeur du lobe principal, mesurée à 3 dB d'atténuation par rapport à l'amplitude en $f = 0$, ou bien à mi-hauteur.
- La hauteur maximale des lobes secondaires (quand ils existent).

Ces paramètres influencent respectivement la résolution et la dynamique de l'analyse spectrale.

La résolution est la capacité à distinguer 2 fréquences proches. La dynamique est la capacité à mesurer des composantes fréquentielles d'amplitudes très différentes sans que la plus forte ne masque la plus faible.

De manière générale, la largeur du lobe principal est inversement proportionnelle à la durée temporelle de la fenêtre.

1.5.1 Fenêtres rectangulaires, triangulaires et paraboliques

- Fenêtre rectangulaire $F_r(t)$

Une fenêtre rectangulaire $F_r(t)$ centrée de durée NT_e s'écrit en temps et en fréquence:

$$\begin{aligned} F_r(t) &= \text{rect}_{NT_e}(t) \\ F_r(f) &= \frac{\sin 2\pi f N T_e / 2}{2\pi f N \frac{T_e}{2}} \end{aligned}$$

La transformée de Fourier de la fenêtre rectangulaire, présente des lobes secondaires importants qui ne décroissent qu'en $\frac{1}{f}$. Le lobe secondaire le plus haut n'est qu'à -13 dB en dessous de l'amplitude en $f = 0$.

- Fenêtre triangulaire $F_t(t)$ et fenêtre parabolique $F_p(t)$

La fenêtre triangulaire $F_t(t)$ (appelée aussi fenêtre de Bartlett) et la fenêtre parabolique $F_p(t)$ se déduisent simplement de la fenêtre rectangulaire par élévation au carré ou au cube de $F_r(f)$. Les transformées de Fourier ainsi obtenues décroissent donc en $\frac{1}{f^2}$ et en $\frac{1}{f^3}$. elles sont respectivement égales à:

$$F_t(\omega) = \left(\frac{\sin N\omega \frac{T_e}{4}}{N\omega \frac{T_e}{4}} \right)^2 \quad \text{ou à} \quad F_p(\omega) = \left(\frac{\sin E\omega \frac{T_e}{6}}{N\omega \frac{T_e}{6}} \right)^3$$

De plus, l'amplitude maximum des lobes secondaires est à -26 db en dessous du lobe principal dans le cas de $F_t(\omega)$, et à -39 db dans le cas de $F_p(\omega)$. Par contre, le lobe principal est, dans les 2 cas, plus large que pour la fenêtre rectangulaire.

Déterminons les expressions temporelles de $F_t(\omega)$ et de $F_p(\omega)$:

$$F_t(\omega) = F_r(\omega) \times F_r(\omega)$$

donc

$$F_t(t) = F_r(t) * F_r(t) = 1 - \left(\frac{2|t|}{NT_e} \right)$$

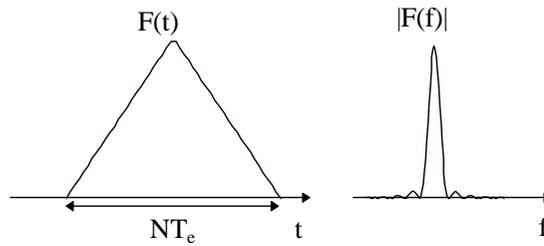
et

$$F_p(\omega) = F_t(\omega) \times F_r(\omega)$$

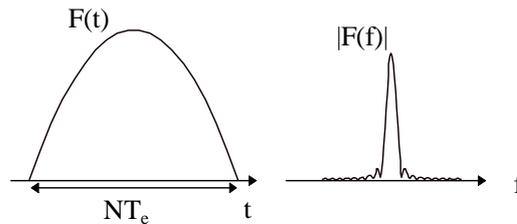
donc

$$F_p(t) = F_t(t) * F_r(t) = 1 - \left(\frac{t}{NT_e} \right)^2$$

La figure suivante représente la fenêtre triangulaire.



La figure suivante représente la fenêtre parabolique.



1.5.2 Fenêtres détruisant par addition algébrique, les lobes secondaires de la fenêtre rectangulaire

D'autres fenêtres intéressantes s'obtiennent en détruisant les lobes secondaires de la fenêtre rectangulaire, par addition algébrique. On peut citer dans cette catégorie la fenêtre cosinusoidale, les fenêtres de Hanning, de Hamming, de Blackman.

- Fenêtre cosinusoidale $F_c(t)$

L'expression algébrique de la fenêtre cosinusoidale est:

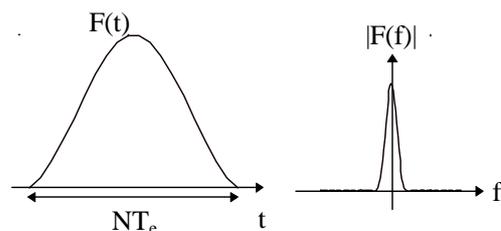
$$F_c(f) = \frac{1}{2} \left(F_R \left(f - \frac{1}{2NT_e} \right) + F_R \left(f + \frac{1}{NT_e} \right) \right)$$

$$F_c(t) = F_r(t) \cos \left(\pi \frac{t}{NT_e} \right)$$

Les lobes secondaires de $F_c(f)$ sont plus faibles que ceux de $F_r(f)$; ainsi l'amplitude maximum de ces lobes est 34 db en dessous de l'amplitude en $f = 0$ et leur décroissance est en $1/f^2$. Par contre, le lobe principal est plus large.

- Fenêtre de Hanning F_H

Nous avons vu précédemment comment, par une combinaison algébrique de deux fonctions déduites de $F_r(f)$ par des décalages en fréquence, on pouvait diminuer l'amplitude des lobes secondaires mais en augmentant la bande de transition. On peut encore diminuer l'amplitude des lobes secondaires en augmentant le nombre de fonctions combinées algébriquement. C'est le cas pour la fenêtre de Hanning (voir la figure suivante).



L'expression algébrique de la fenêtre de Hanning est (en notant $T_0 = NT_e$):

$$F_H(f) = \frac{1}{2} F_R(f) + \frac{1}{4} F_R \left(f - \frac{1}{T_0} \right) + \frac{1}{4} F_R \left(f + \frac{1}{T_0} \right)$$

$$F_H(t) = \frac{1}{2} \left(1 + \cos \left(2\pi \frac{t}{T_0} \right) \right) \quad \text{pour } t \in [-T_0/2, T_0/2]$$

$$F_H(t) = 0 \quad \text{ailleurs.}$$

L'amplitude maximum des lobes secondaires est alors égale à -44 db (en dessous du lobe principal); ils décroissent en $\frac{1}{f^3}$. Le lobe principal est presque 2 fois plus large que pour la fenêtre rectangulaire.

- Fenêtre de Hamming F_{hm}

On peut améliorer les résultats obtenus par la fenêtre précédente en modifiant les pondérations de $F(f)$, $F_r(f - 1/NT_e)$ et $F_r(f + 1/NT_e)$:

$$F_{hm}(f) = 0.56 F_r(f) + 0.22 [F_r(f - 1/NT_e) + F_r(f + 1/NT_e)]$$

Dans ce cas, la décroissance des lobes secondaires est toujours en $1/f^3$ mais l'amplitude maximum de ces lobes est à -60 db sous le lobe principal.

L'expression temporelle $F_{hm}(t)$ de la fenêtre de Hamming s'écrit:

$$F_{hm}(t) = 0,56 + 0,44 \times \cos(2\pi t/NT_e) \quad -\frac{NT_e}{2} < t < \frac{NT_e}{2}$$

$$= 0 \quad \text{ailleurs}$$

- Fenêtre de Blackman $F_B(t)$

Pour diminuer encore l'amplitude des lobes secondaires, on peut combiner:

$F_r(f)$, $F_r(f - 1/NT_e)$, $F_r(f - 2/NT_e)$, $F_r(f + 2/NT_e)$, $F_r(f + 1/NT_e)$ selon la loi suivante:

$$F_B(f) = 0,42 F_r(f) + 0,25 [F_r(f - 1/NT_e) + F_r(f + 1/NT_e)]$$

$$+ 0,08 [F_r(f - 2/NT_e) + F_r(f + 2/NT_e)]$$

La décroissance des lobes secondaires est en $\frac{1}{f^5}$; l'amplitude maximum des lobes secondaires est -87 db en dessous du lobe principal; Le lobe principal est 2 fois plus large que pour la fenêtre rectangulaire. L'expression temporelle de la fenêtre est:

$$F_B(t) = 0,42 + 0,5 \cos(2\pi t/NT_e) + 0,08 \cos(4\pi t/NT_e) \quad -\frac{NT_e}{2} < t < \frac{NT_e}{2}$$

$$\text{ou } 0 \quad \text{ailleurs}$$

1.5.3 Autres fenêtres: Gauss, Kaiser, Dolph-Chebychev

- Fenêtre de Gauss $F_g(t)$

Pour supprimer totalement les lobes de la transformée de Fourier de la fenêtre rectangulaire, on peut utiliser une fenêtre telle que sa transformée de Fourier $F_g(f)$ soit une exponentielle

$$F_g(\omega) = \exp \left[-\frac{(N\omega T_e/2)^2}{4k} \right]$$

Le paramètre k est lié à la dispersion σ de l'exponentielle en prenant comme variable réduite $u = N\omega T_e/2$. σ est alors égal à $\sqrt{2k}$. On pourra donc régler la dispersion de $F_g(f)$, c'est-à-dire la largeur de la bande de transition en faisant varier k .

Cherchons l'expression temporelle de cette fonction: $F_g(t) = \exp[-4k(t/NT_e)^2]$, à une constante près puisque la transformée de Fourier conserve la loi gaussienne:

$$\left[\exp\left(-\frac{t^2}{2}\right) \longrightarrow \exp\left(\frac{-\omega^2}{2}\right) \right]$$

La fenêtre gaussienne a donc pour expression:

$$F_g(t) = \exp[-4k(t/MT)^2] \text{ si } |t| < NT_e/2$$

$$F_g(t) = 0 \text{ si } |t| > NT_e/2$$

Le paramètre k permet de réaliser un compromis entre l'ondulation en bande atténuée et la largeur de la bande de transition, ce que ne permettaient pas de faire les fenêtres décrites précédemment.

- Fenêtre de Kaiser $F_K(t)$

Cette fenêtre est une des plus efficaces: sa transformée de Fourier $F_K(f)$ a pour expression

$$F_K(V) = \frac{2}{I_0(\pi V_a)} \frac{\sin \left[\pi \sqrt{V^2 - V_a^2} \right]}{\pi \sqrt{V^2 - V_a^2}}$$

où $V = fNT_r$, $V_a = f_a NT_e$, $\sqrt{V^2 - V_a^2}$ peut être complexe et $I_0(x)$ est la fonction de Bessel modifiée de première espèce.

Cette fonction dépend d'un paramètre V_a qui permet de diminuer l'amplitude des lobes secondaires mais qui augmente la largeur du lobe principal. Dans la plupart des applications, une valeur de V_a comprise entre $4/\pi$ et $9/\pi$ conviendra.

$$4/\pi < V_a < 9/\pi$$

L'expression temporelle de la fenêtre de Kaiser, transformée de Fourier inverse de $F_K(f)$ est

$$F_K(t) = \frac{I_0 \left[\pi V_a \sqrt{1 - (2t/NT_e)^2} \right]}{I_0(\pi V_a)} \text{ si } |t| < NT_e/2$$

$$F_K(t) = 0 \text{ ailleurs}$$

Comparaison avec la fenêtre de Hamming:

Pour obtenir une bande de transition égale à celle de la fenêtre de Hamming, il suffit d'imposer que le premier zéro de la fonction $F_K(f)$ corresponde à $V = 2$, c'est-à-dire d'imposer que $V_a = \sqrt{3}$ puisque le premier zéro est solution de l'équation $V^2 - V_a^2 = 1$. Dans ces conditions, la fenêtre de Kaiser a 99,8 % de son énergie dans le lobe principal alors que la fenêtre de Hamming n'a que 96,3 % de son énergie dans ce lobe ; par conséquent, l'ondulation en bande atténuée sera meilleure dans le cas de la fenêtre de Kaiser.

Comparaison avec la fenêtre de Blackman:

On peut obtenir la même largeur de lobe principal avec la fenêtre de Kaiser, en positionnant le premier zéro à $V = 3$, c'est-à-dire en faisant $V_a = 2\sqrt{2}$. A nouveau, l'ondulation en bande atténuée est inférieure dans le cas de la fenêtre de Kaiser.

- Fenêtre de Dolph-Chebyshev F_D

La fenêtre de Doph-Tchebychev est celle qui est réalise le meilleur compromis largeur du lobe principal, hauteur des lobes secondaires.

L'expression mathématique de cette fenêtre exprimée dans le domaine fréquentiel est:

$$F_D(f) = \frac{\cos [P \cos^{-1}(\lambda \cos \pi f T_e)]}{ch [P \arg ch(\lambda)]}$$

où $P = N-1$, N impair.

Le paramètre λ permet de régler l'amplitude des ondulations.

les fonctions $\cos(x)$ et $\cos^{-1}(x)$ sont des fonctions complexes.

L'amplitude maximum δ des ondulations est liée au paramètre λ par la relation

$$\delta = \frac{1}{\text{ch}[P \arg \text{ch}(\lambda)]}$$

L'expression temporelle de cette fenêtre n'a pas une forme simple ; la meilleure façon de l'obtenir étant de calculer la transformée de Fourier inverse de $F_D(f)$ en utilisant la transformée de Fourier discrète.

Dans tous les cas, à nombre N de coefficients constant, on devra réaliser un compromis entre l'amplitude des ondulations et la largeur du lobe principal. Il faudra choisir entre une moins grande dispersion ou une meilleure résolution.

1.6 Problèmes de visualisation de la TFD

Quand on lui présente un graphe discret, l'œil réalise une interpolation entre les points du dessin. Cette interpolation est d'autant plus réussie que les points sont rapprochés.

Pour observer à la fois un « beau » signal $x(n)$ et un « beau » spectre, on ne peut visualiser le même « buffer ».

Pour le graphe temporel, on a intérêt à ce que $T_e \ll T_0$ mais si $T_e \ll T_0$ alors $1/T_e$ sera trop grand par rapport à $1/T_0$ et l'œil aura du mal à interpoler le graphe fréquentiel

. Pour améliorer le graphe fréquentiel on peut rajouter des 0 à la suite $x(n)$, on ne change rien au résultat mais on augmente T_0/T_e .

On appelle cette opération « zero-padding » en anglais.

Exemple: soit $x(t)$ dont la fréquence de Shannon ($2f_{max}$) est 128 Hz = f_{SH}

1. pour visualiser en « temps », on échantillonne une durée T à 1024 Hz = $8 f_{SH}$
2. pour visualiser le spectre, on échantillonne à 128 Hz une durée égale à $8 T$ en rajoutant des 0 à la suite $x(n)$.

On aura alors le même nombre de points sur une période de signal et de spectre et les deux graphes seront bien visualisés.

1.7 Propriétés de la TFD et convolution circulaire

Elles sont analogues à celles de la Transformée de Fourier, mais il faut prendre en compte une notion de périodicité des séquences.

1.7.1 Théorème de Parseval

Le théorème de Parseval, sous sa forme discrète, s'écrit:

$$\sum_{n=0}^{N-1} |x(nT_e)|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2$$

Démonstration:

$$\begin{aligned} \sum_{n=0}^{N-1} |x(nT_e)|^2 &= \frac{1}{N^2} \sum_{n=0}^{N-1} \left(\sum_{k=0}^{N-1} X_k e^{2j\pi \frac{nk}{N}} \sum_{l=0}^{N-1} X_l^* e^{2j\pi \frac{-nl}{N}} \right) \\ \sum_{n=0}^{N-1} |x(nT_e)|^2 &= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \left[X_k X_l^* \left(\sum_{n=0}^{N-1} e^{2j\pi \frac{n(k-l)}{N}} \right) \right] \end{aligned}$$

Or:

$$\text{Si } k \neq l \quad \sum_{n=0}^{N-1} e^{2j\pi \frac{n(k-l)}{N}} = \frac{1 - e^{2j\pi(k-l)}}{e^{2j\pi(k-l)/N}} = 0$$

$$\text{Et si } k = l \quad \sum_{n=0}^{N-1} e^{2j\pi \frac{n(k-l)}{N}} = N$$

On en déduit:

$$\sum_{n=0}^{N-1} |x(nT_e)|^2 = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \left[X_k X_l^* \left(\sum_{n=0}^{N-1} e^{2j\pi \frac{n(k-l)}{N}} \right) \right] = \frac{1}{N} \sum_{k=0}^{N-1} |X_k|^2$$

1.7.2 Théorème de la convolution discrète

Avant de présenter les résultats concernant la convolution discrète, on a besoin de définir les notions de convolution circulaire et de convolution linéaire.

Convolution circulaire:

Soit 2 suites périodiques $(x(n))$ et $(y(n))$ de période N . La convolution circulaire de ces 2 suites donne la suite $(z(n))$ de période N définie par:

$$(z(n)) = (x(n)) \otimes (y(n)) = \left(\sum_{i=0}^{N-1} x(i)y(\|n-i\|) \right) \quad \text{pour } n \in [0, N-1],$$

où la notation $\|n-i\|$ signifie $(n-i)$ modulo N .

Convolution linéaire:

La convolution linéaire d'une période de $(x(n))$ et d'une période de $(y(n))$ conduit quant-à-elle à une suite $(u(n))$ de longueur $2N-1$ valant:

$$(u(n)) = (x(n)) * (y(n)) = \left(\sum_{i=0}^{N-1} x(i)y(n-i) \right) \quad \text{pour } n \in [0, 2N-1]$$

EXEMPLE:

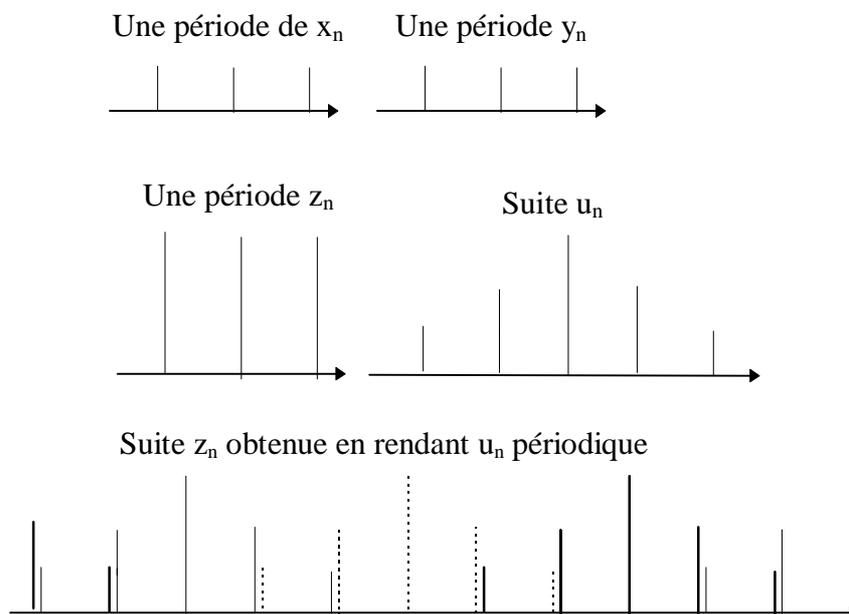
Soit les suites $(x(n))$ et $(y(n))$ périodiques de période $N = 3$, telles que

$$x(n) = y(n) = 1 \quad \text{pour } n \text{ entre } 0 \text{ et } 2.$$

La suite $(z(n))$, convolution circulaire des suites $(x(n))$ et $(y(n))$ est périodique de période $N = 3$, et vaut $z(n) = 3$ pour n entre 0 et 2. La suite $(u(n))$, convolution linéaire des suites $(x(n))$ et $(y(n))$ est de durée $2N-1 = 5$, et vaut:

$$z(0) = 1, z(1) = 2, z(2) = 3, z(3) = 2, z(4) = 1.$$

La suite $(z(n))$ peut s'obtenir en répétant périodiquement la suite $(u(n))$ avec la période N .



Théorème de la convolution discrète circulaire La TFD de la suite $(z(n))$ convolution circulaire de 2 suites périodique $(x(n))$ et $(y(n))$ de période N , est le produit des TFD des suites $(x(n))$ et $(y(n))$:

$$z(n) = x(n) \otimes y(n) \Rightarrow TFD(z(n)) = TFD(x(n))TFD(y(n)) \quad (I.1)$$

où le symbole \otimes représente la convolution circulaire.

Réciproquement, la suite $p(n)$ produit des suites $x(n)$ et $y(n)$, a pour TFD une suite $P(k)$ qui est la convolution circulaire des suites $X(k)$ et $Y(k)$:

$$p(n) = x(n)y(n) \Rightarrow P(k) = X(k) \otimes Y(k)$$

avec $P(k) = TFD(p(n)), X(k) = TFD(x(n)), Y(k) = TFD(y(n))$.

Démonstration de (I.1)

$$\begin{aligned} (z(n)) &= (x(n)) \otimes (y(n)) = \left(\sum_{i=0}^{N-1} x(i)y(\|n-i\|) \right) \text{ pour } n \in [0, N-1] \\ Z(n) &= \sum_{i=0}^{N-1} z(i)e^{-j2\pi\frac{ni}{N}} = \sum_{i=0}^{N-1} \left(\sum_{k=0}^{N-1} x(k)y(\|i-k\|) \right) e^{-j2\pi\frac{ni}{N}} \\ Z(n) &= \sum_{i=0}^{N-1} \left(\sum_{k=0}^{N-1} x(k)e^{-j2\pi\frac{nk}{N}} y(\|i-k\|)e^{-j2\pi\frac{n(i-k)}{N}} \right) = \sum_{i=0}^{N-1} \left(\sum_{k=0}^{N-1} x(k)e^{-j2\pi\frac{nk}{N}} y(\|i-k\|)e^{-j2\pi\frac{n\|i-k\|}{N}} \right) \\ Z(n) &= \left(\sum_{k=0}^{N-1} x(k)e^{-j2\pi\frac{nk}{N}} \right) \left(\sum_{k=0}^{N-1} y(k)e^{-j2\pi\frac{nk}{N}} \right) = X(n)Y(n) \end{aligned}$$

Relation entre convolution discrète linéaire et convolution continue

Soit $x(t)$ et $y(t)$ de durées finies. La convolution de $x(t)$ avec $y(t)$ s'écrit $u(t)$:

$$u(t) = \int_{-\infty}^{+\infty} x(\tau)y(t-\tau)d\tau$$

Échantillons $x(t)$ et $y(t)$ à $f_e = 1/T_e$. On obtient alors P échantillons pour x et Q échantillons pour y .

On peut approcher l'intégrale $u(t)$ par la méthode de l'intégration rectangulaire (de pas T). La suite v_n ainsi obtenue (au terme multiplicatif T_e près) correspond à une convolution discrète linéaire (et non circulaire) des suites $x(n)$ et $y(n)$.

$$v(n) = \sum_{k=0}^{P+Q-1} x(k)y(n-k)$$

1.7.3 Théorème du retard circulaire

Soit la suite $x(n)$ périodique, de période N .

Soit la suite $y(n)$ obtenue en retardant $x(n)$ de k_0 échantillons. La suite $y(n)$ est périodique de période N et sa TFD $Y(k)$ se déduit de celle de $x(n)$ par:

$$Y(k) = X(k)e^{-j2\pi k k_0 / N}$$

où $Y(k)$ et $X(k)$ sont les TFD de $x(n)$ et $y(n)$.

2 Transformée de Fourier Rapide TFR, Fast Fourier transform FFT

La Transformée de Fourier Rapide (notée par la suite FFT) est simplement une TFD calculée selon un algorithme permettant de réduire le nombre d'opérations et, en particulier, le nombre de multiplications à effectuer. Il faut noter cependant, que la réduction du nombre d'opérations arithmétiques à effectuer, n'est pas synonyme de réduction du temps d'exécution. Tout dépend de l'architecture du processeur qui exécute le traitement.

Pour calculer une TFD, on doit calculer N valeurs $X(k)$:

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi \frac{nk}{N}}$$

et ceci pour $k \in [0, N-1]$.

Si on effectue le calcul directement sans algorithme efficace, on doit effectuer:

$$\begin{cases} N^2 & \text{multiplications complexes} \\ N(N-1) & \text{additions complexes} \end{cases}$$

Il existe différents algorithmes de FFT. Le plus connu est sûrement celui de Cooley-Tukey (appelé aussi à entrelacement temporel ou à « *decimation in time* ») qui réduit à

$$\frac{N}{2} \log_2(N) \quad \text{le nombre de multiplications.}$$

Il existe deux versions de l'algorithme:

- FFT avec entrelacement temporel,
- FFT avec entrelacement fréquentiel.

L'algorithme nécessite que N soit une puissance de 2. Le principe de l'algorithme consiste à décomposer le calcul de la TFD d'ordre $N = 2^l$ en l étapes successives.

2.1 FFT avec entrelacement temporel

Illustrons tout d'abord la méthode par un exemple pour $N = 4$.

Les données sont notées $x(n)$ et la suite TFD $X(n)$.

La notation w représente $e^{-j2\pi/N}$, c'est-à-dire $e^{-j2\pi/4}$. On peut remarquer que $w^N = 1$ et $w^{N/2} = -1$.

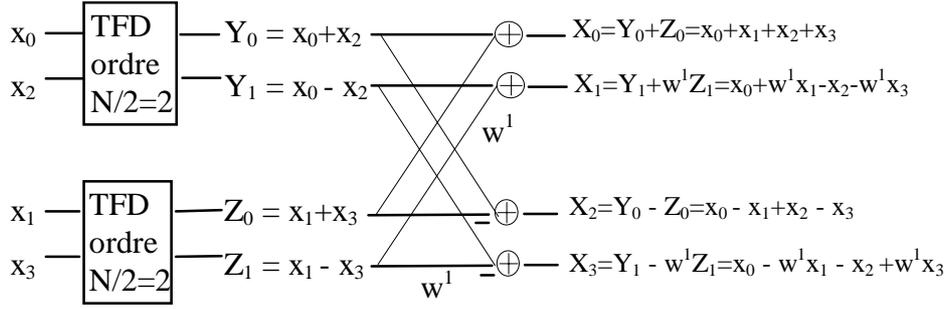
Pour $N = 4$, $w^4 = 1$ et $w^2 = -1$.

la suite TFD s'écrit:

$$\begin{aligned} X(0) &= x(0) + x(1) + x(2) + x(3) = (x(0) + x(2)) + (x(1) + x(3)) \\ X(1) &= x(0) + w^1 x(1) + w^2 x(2) + w^3 x(3) = (x(0) - x(2)) + w^1 (x(1) - x(3)) \\ X(2) &= x(0) + w^2 x(1) + w^4 x(2) + w^6 x(3) = (x(0) + x(2)) - (x(1) + x(3)) \\ X(3) &= x(0) + w^3 x(1) + w^6 x(2) + w^9 x(3) = (x(0) - x(2)) - w^1 (x(1) - x(3)) \end{aligned}$$

Les données $(x(0), x(1), \dots, x(N-1))$ sont regroupées en 2 paquets: un paquet formé des données d'indices pairs $(x(0), x(2), \dots, x(N-2))$ et un paquet formé des données d'indices impairs $(x(1), x(3), \dots, x(N-1))$. Soit pour $N = 4$, un paquet $(x(0), x(2))$ et un paquet $(x(1), x(3))$.

Puis sur chaque paquet on effectue une DFT d'ordre $N/2$ et on combine les résultats de ces 2 DFT pour obtenir celle d'ordre N . Ce qui donne, toujours pour $N = 4$:



Pour obtenir les 4 valeurs $X(k)$, il suffit donc de calculer 2 DFT d'ordre $N/2 = 2$ et de combiner les résultats 2 à 2 à l'aide d'une addition et d'une multiplication au maximum, pour chaque valeur $X(k)$. Cette étape est appelée étage de « papillons », pour des raisons évidentes liées à la forme du schéma de calcul. Ce résultat se généralise à toute valeur de N multiple de 2. En effet:

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{nk}{N}} \\
 X(k) &= \sum_{i=0}^{N/2-1} x(2i) e^{-j2\pi \frac{2ik}{N}} + \sum_{i=0}^{N/2-1} x(2i+1) e^{-j2\pi \frac{2(i+1)k}{N}} \\
 X(k) &= \sum_{i=0}^{N/2-1} x(2i) e^{-j2\pi \frac{ik}{N/2}} + e^{-j2\pi \frac{k}{N}} \sum_{i=0}^{N/2-1} x(2i+1) e^{-j2\pi \frac{ik}{N/2}} \\
 X(k) &= \sum_{i=0}^{N/2-1} y(i) e^{-j2\pi \frac{ik}{N/2}} + w^k \sum_{i=0}^{N/2-1} z(i) e^{-j2\pi \frac{ik}{N/2}}
 \end{aligned}$$

On note $y(i) = x(2i)$ et $z(i) = x(2i+1)$, pour $i \in [0, (N/2 - 1)]$. On remarque que les 2 termes de la somme donnant $X(k)$ se déduisent directement des 2 TFD d'ordre $N/2$ des suites $y(i)$ et $z(i)$ de $N/2$ points. On note ces TFD $Y(k)$ et $Z(k)$.

Ainsi pour $k \leq N/2 - 1$, les 2 termes de la somme se déduisent des termes de rang k de $Y(k)$ et $Z(k)$:

$$X(k) = \sum_{i=0}^{N/2-1} y(i) e^{-j2\pi \frac{ik}{N/2}} + w^k \sum_{i=0}^{N/2-1} z(i) e^{-j2\pi \frac{ik}{N/2}} = Y(k) + w^k Z(k)$$

Pour $k \in [N/2, (N-1)]$, on peut écrire $k = k' + N/2$, avec $k' \in [0, (N/2 - 1)]$. De plus, comme quelque soit i entier $e^{-j2\pi i} = 1$, on peut déduire $X(k)$ des termes de rang $k - N/2$ des 2 TFD $Y(k)$ et $Z(k)$:

$$\begin{aligned}
 X(k) &= \sum_{i=0}^{N/2-1} y(i) e^{-j2\pi \frac{ik}{N/2}} + w^k \sum_{i=0}^{N/2-1} z(i) e^{-j2\pi \frac{ik}{N/2}} \\
 X(k) &= \sum_{i=0}^{N/2-1} y(i) e^{-j2\pi \frac{i(k+N/2)}{N/2}} + w^k \sum_{i=0}^{N/2-1} z(i) e^{-j2\pi \frac{i(k+N/2)}{N/2}} \\
 X(k) &= \sum_{i=0}^{N/2-1} y(i) e^{-j2\pi \frac{ik}{N/2}} + w^k \sum_{i=0}^{N/2-1} z(i) e^{-j2\pi \frac{ik}{N/2}} \\
 X(k) &= Y(k - N/2) + w^k Z(k - N/2)
 \end{aligned}$$

En conclusion, pour tout N multiple de 2, on peut calculer chaque terme $X(k)$ de la TFD d'ordre N en combinant, à l'aide d'au plus 1 multiplication et 1 addition, 2 termes des TFD d'ordre $N/2$ des 2 suites $y(i)$ et $z(i)$ de longueur $N/2$, formées respectivement des termes d'indices pairs et des termes d'indices impairs de la suite $x(n)$. En notant $Y(k)$ et $Z(k)$ les TFD d'ordre $N/2$ de ces suites on peut écrire:

$$\text{Pour } k \in \left[0, \frac{N}{2} - 1\right] \quad X(k) = Y(k) + w^k Z(k)$$

$$\text{Pour } k \in \left[\frac{N}{2}, N-1 \right] \quad X(k) = Y(k - N/2) + w^k Z(k - N/2)$$

On appelle « papillon », l'étape de calcul consistant à calculer 2 points de la TFD d'indices distants de $N/2$, par exemple $X(k)$ et $X(k + N/2)$ avec $k \in [0, N/2 - 1]$. Le calcul de ce couple de valeurs de la TFD d'ordre N utilise le couple de valeurs $Y(k)$ et $Z(k)$ des TFD d'ordre $N/2$:

$$\text{Pour } k \in \left[0, \frac{N}{2} - 1 \right] \quad \begin{cases} X(k) = Y(k) + w^k Z(k) \\ X(k + N/2) = Y_k + w^{k+N/2} Z(k) = Y(k) - w^k Z(k) \end{cases}$$

Chaque papillon nécessite 1 multiplication et 2 additions ou soustractions.

Ainsi tout TFD d'ordre N multiple de 2, peut se calculer à l'aide de 2 TFD d'ordre $N/2$ et d'un étage de $N/2$ papillons.

La complexité de calcul, pour la TFD d'ordre N est donc égale à celle de 2 TFD d'ordre $N/2$ plus celle de $N/2$ papillons. Si on suppose que les TFD d'ordre $N/2$ sont calculées directement (sans algorithme efficace), on peut dire que:

Le calcul d'une TFD d'ordre N pair, avec cet algorithme, demande:

$$\begin{aligned} \text{Le calcul de 2 TFD d'ordre } N/2: & \quad \begin{cases} 2 \left(\frac{N}{2}\right)^2 = \frac{N^2}{2} \text{ multiplications complexes} \\ 2 \frac{N}{2} \left(\frac{N}{2} - 1\right) = N \left(\frac{N}{2} - 1\right) \text{ additions complexes} \end{cases} \\ \text{Le calcul de } N/2 \text{ papillons:} & \quad \begin{cases} \frac{N}{2} \text{ multiplications complexes} \\ N \text{ additions/soustractions} \end{cases} \end{aligned}$$

Soit un total de

$$\begin{cases} \frac{N^2}{2} + \frac{N}{2} \text{ multiplications complexes} \\ \frac{N^2}{2} \text{ additions complexes} \end{cases}$$

au lieu de:

$$\begin{cases} N^2 \text{ multiplications complexes} \\ N(N-1) \text{ additions complexes} \end{cases}$$

pour le calcul direct.

Ainsi pour $N = 4$, on a besoin de 10 multiplications et de 8 additions/soustractions complexes au lieu de 16 multiplications et de 12 additions/soustractions complexes.

Si $N/2$ est un multiple de 2, on peut réitérer la méthode pour le calcul des 2 TFD d'ordre $N/2$. Chaque TFD d'ordre $N/2$ est alors calculée à l'aide de 2 TFD d'ordre $N/4$ et de $N/4$ papillons, ce qui donne au total 4 TFD d'ordre $N/4$ plus 2 étages de $N/2$ papillons.

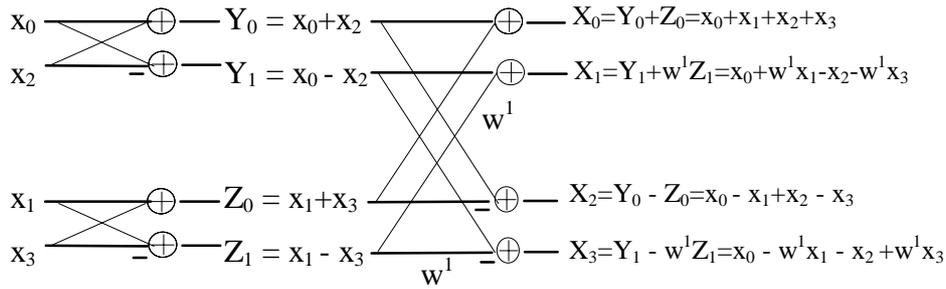
D'une manière plus générale si N est une puissance de 2, $N = 2^l$, on peut réitérer la méthode l fois et calculer la TFD d'ordre N à l'aide de l étages de $N/2$ papillons, avec $l = \log_2(N)$. La complexité de calcul d'une TFD d'ordre N devient alors celle de l étages de $N/2$ papillons, soit:

$$\begin{cases} l \frac{N}{2} = \log_2(N) \frac{N}{2} \text{ multiplications complexes} \\ l N = \log_2(N) N \text{ additions complexes} \end{cases}$$

Cet algorithme est l'algorithme de FFT avec entrelacement temporel (base 2) de Cooley-Tukey.

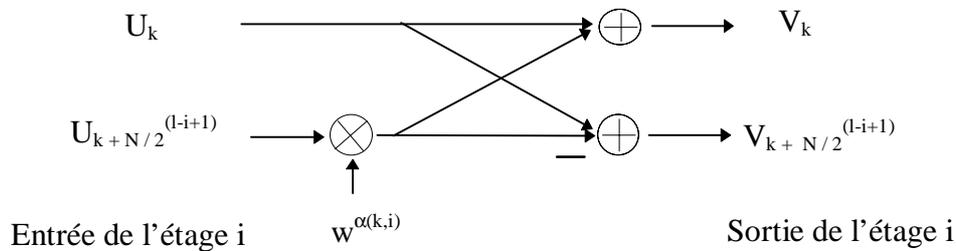
Ainsi pour $N = 1024 = 2^{10}$, le calcul direct demande: 1024x1024 multiplications et 1024x1023 additions, alors que le calcul avec l'algorithme de FFT demande: 10x512 multiplications et 10x1024 additions. Dans ce cas, l'algorithme divise environ par 200 le nombre d'opérations à effectuer. L'efficacité de la FFT augmente avec N .

Pour $N = 4$, le schéma complet de l'algorithme est le suivant:



On remarque sur ce schéma que les données $x(n)$ en entrée sont désordonnées, alors que celles de sortie $X(k)$ sont dans l'ordre naturel. De ce fait cet algorithme de FFT s'appelle FFT avec entrelacement temporel. On verra par la suite qu'il existe un algorithme symétrique appelé FFT avec entrelacement fréquentiel.

Pour l'algorithme de FFT en base 2 avec entrelacement temporel, un papillon élémentaire, à l'étage i (en numérotant de 1 à $l = \log_2(N)$), a la forme suivante:



A l'étape i , les indices des termes associés dans un papillon sont séparés de N_i , N_i étant la taille des DFT intervenant à l'étape i , c'est à dire $N_i = 2^{i-1} = 2^l / 2^{(l-i+1)} = N / 2^{(l-i+1)}$.

Le terme $w^{\alpha(i,k)}$ vaut:

$$w^{\alpha(i,k)} = e^{-j2\pi \frac{k}{2N_i}} = e^{-j2\pi \frac{k}{N} \frac{N}{2^i}} = w^{k2^{l-i}}$$

2.2 FFT avec entrelacement fréquentiel

Cet algorithme est symétrique du précédent. Les données temporelles $x(n)$ restent dans l'ordre naturel, mais les résultats $X(k)$ sont désordonnés.

Le principe consiste encore à décomposer le calcul de la TFD d'ordre $N = 2^l$ en l étapes successives. Mais le regroupement de données se fait différemment.

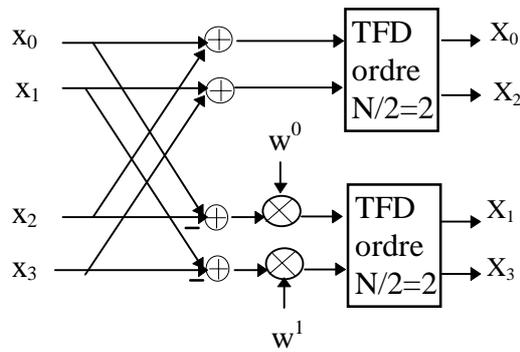
Illustrons la méthode par un exemple pour $N = 4$.

Les données fréquentielles $(X(0), X(1), \dots, X(N-1))$ sont regroupées en 2 paquets: un paquet formé des données d'indices pairs $(X(0), X(2), \dots, X(N-2))$ et un paquet formé des données d'indices impairs $(X(1), X(3), \dots, X(N-1))$. Soit pour $N=4$, un paquet $(X(0), X(2))$ et un paquet $(X(1), X(3))$.

Pour $N = 4$, on peut écrire:

$$\begin{aligned}
 X(0) &= x(0) + x(1) + x(2) + x(3) = (x(0) + x(2)) + (x(1) + x(3)) \\
 X(2) &= x(0) + w^2 x(1) + w^4 x(2) + w^6 x(3) = (x(0) + x(2)) - (x(1) + x(3)) \\
 X(1) &= x(0) + w^1 x(1) + w^2 x(2) + w^3 x(3) = (x(0) - x(2)) + [w^1 (x(1) - x(3))] \\
 X(3) &= x(0) + w^3 x(1) + w^6 x(2) + w^9 x(3) = (x(0) - x(2)) - [w^1 (x(1) - x(3))]
 \end{aligned}$$

Pour obtenir chaque paquet de résultats fréquentiels, on effectue une DFT d'ordre $N/2$ sur des données résultant d'une étape de papillons sur les données $x(n)$.



On a donc un étage de 2 papillons suivi d'un étage de 2 DFT d'ordre $N/2 = 2$.

Ce résultat se généralise à toute valeur de N multiple de 2. En effet:

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{nk}{N}} \\
 X(2i) &= \sum_{n=0}^{N/2-1} x(n) e^{-j2\pi \frac{n2i}{N}} + \sum_{n=N/2}^{N-1} x(n) e^{-j2\pi \frac{n2i}{N}} \\
 X(2i) &= \sum_{n=0}^{N/2-1} x(n) e^{-j2\pi \frac{ni}{N/2}} + \sum_{m=0}^{N/2-1} x(m+N/2) e^{-j2\pi \frac{mi}{N/2}} \\
 X(2i) &= \sum_{n=0}^{N/2-1} (x(n) + x(n+N/2)) e^{-j2\pi \frac{ni}{N/2}}
 \end{aligned}$$

Ainsi les $N/2$ termes $X(k)$ de rang pair sont égaux aux termes de la TFD d'ordre $N/2$ de la suite de $N/2$ valeurs $(x(n) + x(n+N/2))$, avec n entre 0 et $N/2 - 1$.

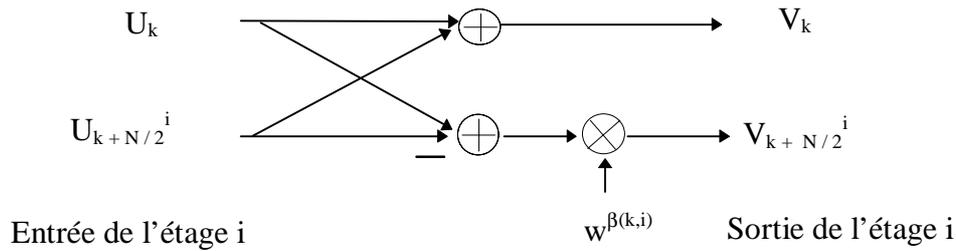
De même pour les termes $X(k)$ de rang impair:

$$\begin{aligned}
 X(k) &= \sum_n x(n) e^{-j2\pi \frac{nk}{N}} \\
 X(2i+1) &= \sum_{n=0}^{N/2-1} x(n) e^{-j2\pi \frac{n(2i+1)}{N}} + \sum_{n=N/2}^{N-1} x(n) e^{-j2\pi \frac{n(2i+1)}{N}} \\
 X(2i+1) &= \sum_{n=0}^{N/2-1} x(n) e^{-j2\pi \frac{ni}{N/2}} e^{-j2\pi \frac{ni}{N/2}} - \sum_{m=0}^{N/2-1} x(m+N/2) e^{-j2\pi \frac{m}{N}} e^{-j2\pi \frac{mi}{N/2}} \\
 X(2i+1) &= \sum_{n=0}^{N/2-1} w^n (x(n) - x(n+N/2)) e^{-j2\pi \frac{ni}{N/2}}
 \end{aligned}$$

les $N/2$ termes $X(k)$ de rang impair sont égaux aux termes de la TFD d'ordre $N/2$ de la suite de $N/2$ valeurs $w^n(x(n) - x(n+N/2))$, avec n entre 0 et $N/2 - 1$.

D'une manière générale si N est une puissance de 2: $N = 2^l$, on peut répéter la méthode l fois et calculer la TFD d'ordre N à l'aide de l étages de $N/2$ papillons., avec $l = \log_2(N)$. La complexité de calcul d'une FFT avec entrelacement fréquentiel est identique à celle de la FFT avec entrelacement temporel.

Pour l'algorithme de FFT en base 2 avec entrelacement fréquentiel, un papillon élémentaire, à l'étage i (en numérotant de 1 à $l = \log_2(N)$), a la forme suivante:



A l'étape i , Les indices des termes associés dans un papillon sont séparés de N_i , N_i étant la taille des DFT intervenant à l'étape i , c'est à dire $N_i = N/2^i$. Et le terme $w^{\beta(i,k)}$ vaut:

$$w^{\beta(i,k)} = e^{-j2\pi \frac{k}{2N_i}} = e^{-j2\pi \frac{k2^i}{2N}} = w^{k2^{i-1}}$$

2.3 Bit reversal

On remarque que dans les 2 cas précédents: FFT avec entrelacement temporel et FFT avec entrelacement fréquentiel, l'ordre entrelacé est obtenu à partir de l'ordre naturel en appliquant une technique dite du « bit reversal ».

Cette technique consiste à écrire en binaire l'indice dans l'ordre naturel puis à retourner l'ordre des bits pour obtenir la représentation binaire de l'indice correspondant dans l'ordre entrelacé.

Par exemple pour $N = 4$:

indices ordre naturel	représentation binaire	représentation retournée	indices ordre entrelacé
0	00	00	0
1	01	10	2
2	10	01	1
3	11	11	3

2.4 Formulation matricielle de l'algorithme de Cooley-Tukey

On utilise les mêmes notations que précédemment:

$$N = 2^l$$

$w = e^{-j\frac{2\pi}{N}}$. On utilise le fait que $w^N = 1$ et que $w^{N/2} = -1$

On illustre l'algorithme pour le cas $N = 8$, $l = 3$. Et on présente l'algorithme avec entrelacement fréquentiel.

- On calcule la TFD de la suite $x(n)$ avec $n \in [0, 7]$
- On note la suite transformée $X(n)$ avec $n \in [0, 7]$

$$X(n) = \sum_{k=0}^7 x(k) e^{-j2\pi \frac{nk}{N}} = \sum_{k=0}^7 x(k) w^{nk}$$

On décompose les indices n et k (qui sont compris entre 0 et 7) en base 2:

$$\begin{aligned} n &= n_2 2^2 + n_1 2^1 + n_0 2^0 \\ k &= k_2 2^2 + k_1 2^1 + k_0 2^0 \end{aligned}$$

On effectue le produit nk en développant selon k , (Il suffirait de développer selon n pour obtenir l'algorithme avec entrelacement temporel):

$$\begin{aligned} w^{nk} &= w^{n k_2 2^2} w^{n k_1 2^1} w^{n k_0} \\ w^{nk} &= w^{(n_2 2^2 + n_1 2^1 + n_0) k_2 2^2} w^{(n_2 2^2 + n_1 2^1 + n_0) k_1 2^1} w^{(n_2 2^2 + n_1 2^1 + n_0) k_0} \\ w^8 &= 1 \Rightarrow \\ w^{nk} &= w^{n_0 k_2 2^2} w^{(n_1 2^1 + n_0) k_1 2^1} w^{(n_2 2^2 + n_1 2^1 + n_0) k_0} \end{aligned}$$

$$X(n) = \sum_{k=0}^7 x(k)w^{nk}$$

$$X(n) = \sum_{k_0=0}^1 \left[\sum_{k_1=0}^1 \left(\sum_{k_2=0}^1 x(k_2, k_1, k_1)w^{n_0 k_2 2^2} \right) w^{(2n_1+n_0)k_1 2^1} \right] w^{(4n_2+2n_1+n_0)k_0}$$

On a ainsi décomposé la sommation unique sur k en $3 = \log_2(8)$ sommations sur k_0, k_1, k_2 .

Si on effectue la sommation sur k_2 , on obtient un terme:

$$x_1(n_0, k_1, k_0) = \sum_{k_2=0}^1 x(k_2, k_1, k_0)w^{n_0 k_2 2^2}$$

$$x_1(n_0, k_1, k_0) = x(0, k_1, k_0) + x(1, k_1, k_0)w^{4n_0}$$

Si l'on considère l'ensemble des valeurs x_1 que l'on peut obtenir en donnant aux indices n_0, k_1, k_0 toutes les valeurs possibles, on obtient 8 valeurs $x_1(n)$ avec $n \in [0, 7]$ qui vérifient les relations:

$$x_1(i) = x(i) + w^0 x(i+4) \quad \text{pour } i \leq 3$$

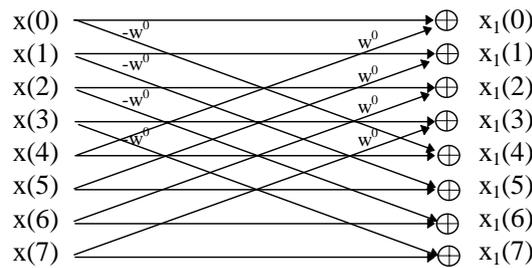
$$x_1(i+4) = x(i) - w^0 x(i+4)$$

avec $i = n_0 2^2 + k_1 2 + k_0$

On peut regrouper les valeurs de x_1 2 par 2 pour former des paires duales que l'on calcule à partir de 2 noeuds de même indice de l'étape précédente. Les indices de 2 noeuds duaux étant espacés de $4 = N/2$. L'ensemble des relations précédentes peut s'écrire sous la forme matricielle:

$$\begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \\ x_1(4) \\ x_1(5) \\ x_1(6) \\ x_1(7) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & w^0 \\ 1 & 0 & 0 & 0 & -w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -w^0 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

Ces relations peuvent aussi se représenter sous la forme graphique de « papillons »:



La deuxième étape est la sommation sur k_1 :

$$x_2(n_0, n_1, k_0) = \sum_{k_1=0}^1 x_1(n_0, k_1, k_0)w^{(2n_1+n_0)2k_1}$$

$$x_2(n_0, n_1, k_0) = x_1(n_0, 0, k_0) + x_1(n_0, 1, k_0)w^{4n_1+2n_0}$$

avec $w^{4n_1} = (-1)^{n_1}$

En donnant aux indices n_0, n_1, k_0 toutes les valeurs possibles on obtient les relations duales:

$$x_2(i) = x_1(i) + w^{2n_0} x_1(i+2) \quad \text{pour } i \leq 3$$

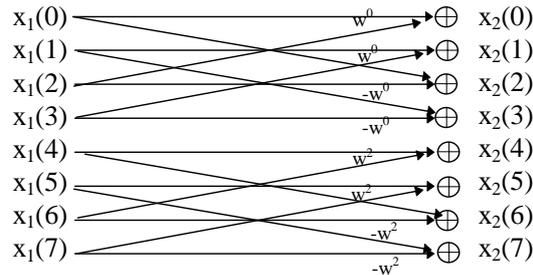
$$x_2(i+2) = x_1(i) - w^{2n_0} x_1(i+2)$$

avec $i = 4n_0 + k_0$ c'est à dire $n_1 = 0$

Les valeurs de x_2 se calculent 2 par 2 à partir des 2 valeurs de x_1 de mêmes indices de l'étape précédente. Les noeuds duaux étant espacés de $N/2^2 = 2$. Les relations précédentes correspondent à la forme matricielle:

$$\begin{bmatrix} x_2(0) \\ x_2(1) \\ x_2(2) \\ x_2(3) \\ x_2(4) \\ x_2(5) \\ x_2(6) \\ x_2(7) \end{bmatrix} = \begin{bmatrix} 1 & 0 & w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & w^0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -w^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & w^2 \\ 0 & 0 & 0 & 0 & 0 & 1 & -w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -w^2 \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \\ x_1(4) \\ x_1(5) \\ x_1(6) \\ x_1(7) \end{bmatrix}$$

Ces relations peuvent aussi se représenter sous la forme graphique de papillons:



L'exposant de w dans les relations précédentes est obtenu en divisant logiquement i par 2 et en retournant le résultat.

Enfin la troisième étape est la sommation sur k_0 qui conduit à la relation:

$$\begin{aligned} x_3(n_0, n_1, n_2) &= \sum_{k_0=0}^1 x_2(n_0, n_1, k_0) w^{(4n_2+2n_1+n_0)k_0} \\ x_3(n_0, n_1, n_2) &= x_2(n_0, n_1, 0) + x_2(n_0, n_1, 1) w^{(4n_2+2n_1+n_0)} \\ \text{avec } w^{4n_2} &= (-1)^{n_2} \end{aligned}$$

Ce qui conduit aux relations duales:

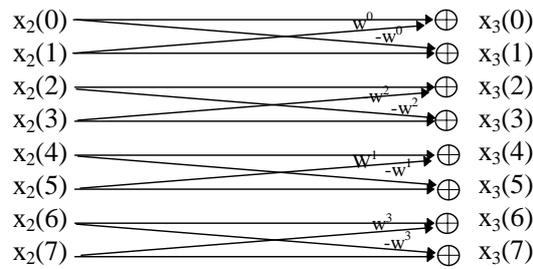
$$\begin{aligned} x_3(i) &= x_2(i) + w^{2n_1+n_0} x_2(i+1) \\ x_3(i+1) &= x_2(i) - w^{2n_1+n_0} x_2(i+1) \text{ pour } i \text{ pair} \\ \text{avec } i &= 4n_0 + 2n_1 \end{aligned}$$

Les valeurs de x_3 se calculent 2 par 2 à partir des 2 valeurs de x_2 de mêmes indices de l'étape précédente. Les noeuds duaux étant espacés de $N/2^3 = 1$. L'exposant de w s'obtient en divisant logiquement i par 1 et en retournant le résultat.

Les relations précédentes correspondent à la forme matricielle:

$$\begin{bmatrix} x_3(0) \\ x_3(1) \\ x_3(2) \\ x_3(3) \\ x_3(4) \\ x_3(5) \\ x_3(6) \\ x_3(7) \end{bmatrix} = \begin{bmatrix} 1 & w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & w^3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -w^3 \end{bmatrix} \begin{bmatrix} x_2(0) \\ x_2(1) \\ x_2(2) \\ x_2(3) \\ x_2(4) \\ x_2(5) \\ x_2(6) \\ x_2(7) \end{bmatrix}$$

Relations qui se représentent graphiquement par:



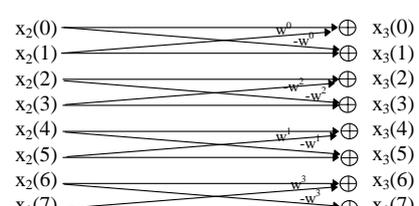
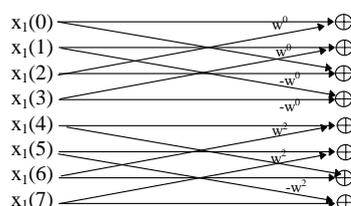
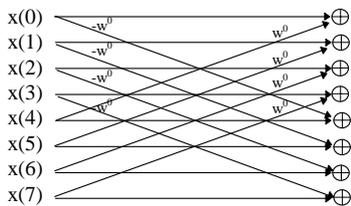
Cette dernière suite de valeurs est égale à la suite cherchée X(n) mais dans le désordre:

$$X(n_2, n_1, n_0) = x_3(n_0, n_1, n_2)$$

En résumé:

$$\begin{bmatrix} X(0) \\ X(4) \\ X(2) \\ X(6) \\ X(1) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} x_3(0) \\ x_3(1) \\ x_3(2) \\ x_3(3) \\ x_3(4) \\ x_3(5) \\ x_3(6) \\ x_3(7) \end{bmatrix} = \begin{bmatrix} 1 & w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & w^3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -w^3 \end{bmatrix} \times$$

$$\begin{bmatrix} 1 & 0 & w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & w^0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -w^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & w^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & -w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -w^2 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & w^0 \\ 1 & 0 & 0 & 0 & -w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -w^0 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$



La méthode illustrée pour $N = 8$ se généralise pour tout N puissance de 2.

2.5 Autres algorithmes de FFT

Il existe de nombreux autres algorithmes de FFT, qui s'appliquent par exemple au cas où N n'est pas une puissance de 2.

Lorsque N est une puissance de 4, on peut appliquer un algorithme de FFT en base 4, plus efficace que l'algorithme en base 2. Le nombre d'opérations nécessaires étant alors de:

$$\begin{cases} \log_4(N) \frac{3N}{4} \text{ multiplications complexes} \\ \log_4(N) N \text{ additions complexes} \end{cases}$$

2.6 Utilisation de la FFT pour la convolution rapide

Soit la suite $(u(n))$ convolution de la suite $(x(n))$, de durée P, et de la suite $(y(n))$, de durée Q.:

$$u(n) = \sum_{i=0}^{P-1} x(i)y(n-i)$$

la suite $(u(n))$ a pour durée $P + Q - 1$.

On peut calculer directement les $(P+Q-1)$ valeurs de la suite $(u(n))$. Ce calcul nécessite $P(Q+1)$ multiplications pour $(Q \geq P)$. Si P et Q sont grands, il peut être efficace de faire ce calcul par FFT.

Pour cela, on complète par des zéros les suites $(x(n))$ et $(y(n))$ de façon qu'elles aient toutes les deux N points, où N est une puissance de 2 telle que $N \geq P + Q - 1$.

Appelons $(x_l(n))$ et $(y_l(n))$ les suites prolongées par des zéros.

La suite $(z_l(n)) = (x_l(n)) \otimes (y_l(n))$ coïncide avec la suite $(u(n))$ sur ses $P + Q - 1$ premiers points.

Pour calculer la suite $(u(n))$, il suffit donc de:

1. compléter les suites $(x(n))$ et $(y(n))$ par des zéros pour qu'elles aient N points, avec $N \geq P + Q - 1$.
2. Calculer par FFT les suites $X_l(k)$ et $Y_l(k)$: DFT des suites $(x_l(n))$ et $(y_l(n))$.
3. Calculer la suite $Z_l(k)$ produit des suites $X_l(k)$ et $Y_l(k)$: $Z_l(k) = X_l(k) \times Y_l(k)$
4. Calculer la suite $z_l(n)$ par FFT inverse de $Z_l(k)$
5. Identifier $(u(n))$ avec les $P + Q - 1$ premiers points de $z_l(n)$.

Le nombre d'opérations à effectuer est de:

3 FFT (directes ou inverses) de N points + N multiplications complexes, c'est à dire:

$$\begin{cases} \frac{3}{2}N \log_2(N) + N \text{ multiplications complexes} \\ N \log_2(N) \text{ additions complexes} \end{cases}$$

Ce qu'il faut comparer au calcul direct: $P(Q+1)$ multiplications et additions réelles.

Exemple numérique: $P = Q = 500$

On choisirait $N = 1024$.

Le calcul direct demanderait 500×501 multiplications et additions réelles.

Le calcul par FFT demanderait environ $1500 \times 10 + 1000$ soit 16000 multiplications complexes. Une multiplication complexe demandant 4 multiplications réelles, on constate que le gain de calcul par FFT serait de:

$$\frac{500 \times 500}{16000 \times 4} \approx 4$$

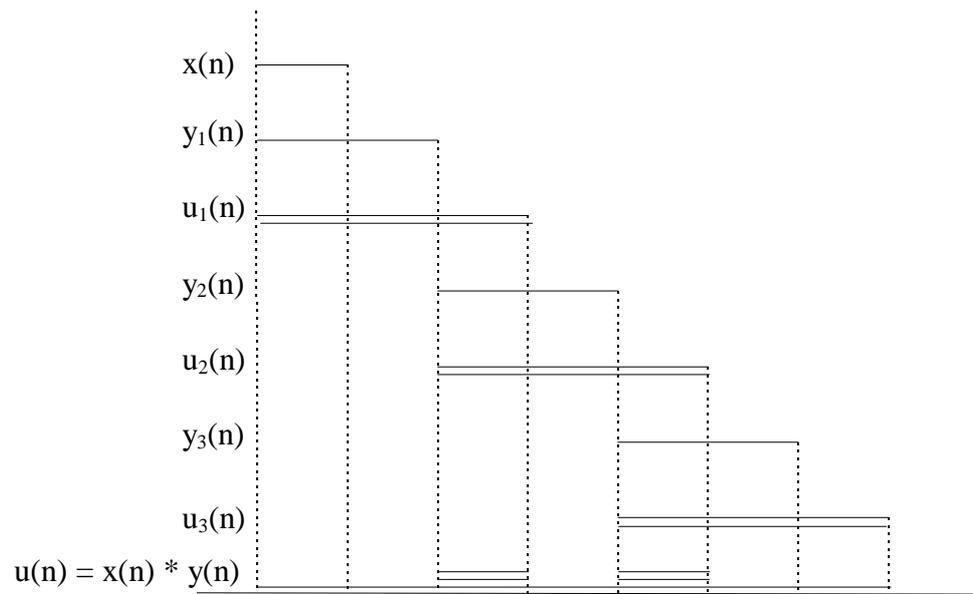
2.7 Calcul de convolution par section d'une des suites

Souvent l'une des suites est beaucoup plus courte que l'autre. Le calcul peut alors se faire en sectionnant la suite la plus longue en sous-suites.

Appelons $(x(n))$ la suite courte de longueur P et $(y(n))$ la suite longue de longueur Q .

$(y(n))$ est coupée en suites de longueur $M > P$.

La convolution totale de $(x(n))$ et $(y(n))$ est obtenue en sommant correctement les convolutions de $(x(n))$ avec les sous-suites de $(y(n))$. Ces petites convolutions sont calculées par exemple par FFT. Le résultat coïncide sur $P + M - 1$ premiers points avec les bonnes valeurs. Les petites convolutions se superposent 2 à 2 sur $(P - 1)$ points.



Cette méthode permet de limiter la taille de la mémoire nécessaire.

EXERCICES ET PROBLÈMES

Exercice 1 : Calculer la transformée de Fourier discrète de la suite (x_n) formée de $N = 8$ points ($n \in [0,7]$), obtenue en échantillonnant à la fréquence $f_e = 16$ Hz le signal $x(t)$:

$$x(t) = 2 \sin(8\pi t) + 8 \cos(4\pi t)$$

Exercice 2 : Calculer la TFD de la suite x_n suivante:

x_n est formée de $N = 24$ points obtenus en échantillonnant le signal $x(t) = 3 \sin(8\pi t) + 4 \cos(6\pi t)$ à la fréquence $f_e = 24$ Hz.

Exercice 3 : Comparer le résultat de la convolution linéaire et de la convolution circulaire des 2 suites x_n et y_n suivantes:

$$\begin{cases} x_n = 1 & \text{pour } 0 \leq n \leq 3 \\ x_n = 0 & \text{pour } n \notin [0,3] \end{cases}$$

$$\begin{cases} y_n = 2 & \text{pour } 0 \leq n \leq 3 \\ y_n = 0 & \text{pour } n \notin [0,3] \end{cases}$$

On appellera z_n le résultat de la convolution linéaire:

$$z_n = \sum_{k=0}^3 x_k y_{n-k}$$

et t_n le résultat de la convolution circulaire:

$$t_n = \sum_{k=0}^3 x_k y_{\|n-k\|}$$

où $\|n-k\|$ signifie $n-k$ modulo $N = 4$.

Exercice 4 : Pour $N = 4$, la TFD de la suite: x_0, x_1, x_2, x_3 s'écrit: X_0, X_1, X_2, X_3 avec:

$$X_k = \sum_{n=0}^3 x_n e^{-j2\pi \frac{nk}{4}}$$

1) Ecrire la matrice de passage du vecteur $\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$ au vecteur $\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix}$. On notera $w = e^{-2j\frac{\pi}{4}}$.

2) Quel est le nombre de multiplications et d'additions à effectuer pour effectuer le calcul de la TFD sans algorithme particulier?

3) Chercher à décomposer le calcul en 2 étapes pour diminuer le nombre d'opérations globales. Par exemple, commencer par grouper les indices pairs et les indices impairs séparément.

plain

plain