

Résumé de LEPL1106

compilation du 12 avril 2023

Thomas Debelle

Juin 2023

Table des matières

I	Signaux	4
1	Les signaux	5
1.1	Définition	5
1.2	Signaux élémentaires	5
1.2.1	Signaux exponentiels	5
1.2.2	Signaux sinusoïdales	5
1.2.3	Signaux amortis	6
1.2.4	L'impulsion (temps discret)	6
1.2.5	L'échelon (temps discret)	6
1.2.6	L'impulsion (temps continu)	6
1.3	Opération sur les signaux	7
1.4	Convolution (temps discret)	7
1.4.1	Propriétés	8
1.5	Convolution (temps continu)	8
1.5.1	Propriétés	8
2	Fourier	9
2.1	La représentation de Fourier	9
2.1.1	Signaux continus	9
2.1.2	Signaux discrets	11
2.2	La transformée de Fourier	11
2.2.1	Calcul de la transformée	11
2.2.2	Calcul discret	12
2.2.3	Astuce	12
2.3	Propriétés de Fourier	12
2.3.1	Dualité	12
2.3.2	Linéarité	13
2.3.3	Translation	13
2.3.4	Modulation	13
2.3.5	Différentiation	13
2.3.6	Multiplication par un monôme	13
2.3.7	Intégration	14
2.3.8	Dilatation	14
2.3.9	Renversement	14
2.4	Transformée usuelle	14
2.4.1	Train d'impulsions (peigne) de Dirac	15
2.4.2	Fonction échelon	16
2.4.3	Fonction fenêtre	16
2.4.4	Fonction signe	16
2.5	Tableau de Check	17

3	L'échantillonnage	18
3.1	Lien entre le continu et le discret	18
3.1.1	L'échantillonnage	18
3.2	Repli spectral	19
3.2.1	Bien échantillonner	19
3.2.2	Reconstruire un signal	20
3.2.3	Généralisation de Shanon	20
3.3	Sur- et Sous-échantillonnage	20
4	Discrete Fourier Transform (DFT)	21
4.1	Définition	21
4.2	Transformée Rapide	22
II	Systèmes	23
5	Système LIT	24
5.1	LIT	24
5.2	Réponse impulsionnelle	24
5.3	Type de système	25
5.4	Modélisation et représentation des systèmes	25
5.4.1	Inconvénient	25
5.4.2	Représentation	25
5.4.3	Équation différentielle entrée-sortie	26
5.4.4	Schéma Bloc	27
5.4.5	Représentation d'état	28
5.4.6	Passage de représentation	28
5.4.7	Temps discret	28
5.4.8	Résumé	29
5.4.9	Existence des systèmes LIT	29
6	Filtrage et Bode	30
6.1	Filtre	30
6.1.1	Type de filtre	30
6.1.2	Filtres non-idéaux	31
6.1.3	Filtre analogique	32
6.2	Caractérisation d'un filtre	33
6.2.1	Argument	34

Préface

Bonjour à toi !

Cette synthèse recueille toutes les informations importantes données au cours, pendant les séances de tp et est améliorée grâce au note du Syllabus. Elle ne remplace pas le cours donc écoutez bien les conseils et potentielles astuces que les professeurs peuvent vous donner. Notre synthèse est plus une aide qui, on l'espère, vous sera à toutes et tous utile.

Elle a été réalisée par toutes les personnes que tu vois mentionnées. Si jamais cette synthèse a une faute, manque de précision, typo ou n'est pas à jour par rapport à la matière actuelle ou bien que tu veux simplement contribuer en y apportant tes connaissances ? Rien de plus simple ! Améliore la en te rendant [ici](#) où tu trouveras toutes les infos pour mettre ce document à jour. (*en plus tu auras ton nom en gros ici et sur la page du github*)

Nous espérons que cette synthèse te sera utile d'une quelconque manière ! Bonne lecture et bonne étude.

Première partie

Signaux

Chapitre 1

Les signaux

1.1 Définition

Définition

Un signal est une fonction de une ou plusieurs variables (continues ou discrètes) qui correspondent à de l'information ou à un phénomène physique.

Continues ou discrets ?

Un signal est dit continu si il est défini sur un espace de temps continu. On note ce signal $x(t)$. Et il est dit discret si il est défini sur un espace discret de temps. On note ce signal $x[t]$.

Manipulation des signaux

Pour le cas discrets ou continu, nous pouvons réaliser les opérations suivantes.

- Combinaison linéaire $\rightarrow \alpha_1 x_1(t) + \alpha_2 x_2(t)$
- Multiplication $\rightarrow x_1[t]x_2[t]$
- Dilatation $\rightarrow x[n/a], a > \mathbb{R}$
- Translation $\rightarrow x(t - t_0), t_0 \in \mathbb{R}$
- Renversement $\rightarrow x(-t)$
- Différentiation (que pour le cas continu) $\rightarrow \frac{d^n x(t)}{dt^n}$
- Intégration (que pour le cas continu) $\rightarrow \int x(t)dt$

1.2 Signaux élémentaires

1.2.1 Signaux exponentiels

Pour les signaux continus nous avons :

$$x(t) = Be^{at} \quad (1.1)$$

Et pour les signaux discrets nous avons :

$$x[n] = Br^n \rightarrow 0 < r < 1 \quad (1.2)$$

1.2.2 Signaux sinusoïdales

Pour les signaux continus nous avons :

$$x(t) = A \cos(\omega t + \phi) \quad (1.3)$$

Et pour les signaux discrets nous avons :

$$x[n] = A \cos(\Omega n + \Phi) \quad (1.4)$$

Il a une période de $\Omega N = 2\pi m$

1.2.3 Signaux amortis

Pour les signaux continus et avec $\alpha > 0$:

$$x(t) = A e^{-\alpha t} \cos(\omega t + \phi) \quad (1.5)$$

Et pour les signaux discrets :

$$x[n] = B r^n \cos(\Omega n + \Phi) \quad (1.6)$$

1.2.4 L'impulsion (temps discret)

Comme son nom l'indique, ce signal se représente sous la forme d'une impulsion. Par sa définition, cela nous force à avoir un signal discret !

$$\begin{cases} \delta[n] = 1 \rightarrow n = 0 \\ \delta[n] = 0 \rightarrow \forall n \neq 0 \end{cases} \quad (1.7)$$

On peut réaliser des impulsions décaler en écrivant $\delta[n-x]$ avec x représentant la valeur du décalage.

1.2.5 L'échelon (temps discret)

Ce type de signal élémentaire est encore plus trivial puisqu'il se résume à :

$$\begin{cases} 1 \rightarrow n \geq 0 \\ 0 \rightarrow n < 0 \end{cases} \quad (1.8)$$

On peut aussi voir l'échelon comme une somme infinie d'impulsion comme $\sum_{k \geq 0}^{\infty} \delta[n-k]$.

1.2.6 L'impulsion (temps continu)

$$\begin{cases} \delta(t) = 0 \text{ si } t \neq 0 \\ \delta(0) = (+\infty) \\ \int_{-a}^a \delta(s) ds = 1 \rightarrow \forall a > 0 \end{cases} \quad (1.9)$$

À noter que la dernière ligne nous crée une propriété bien spécifique. En effet, la valeur de l'impulsion est limitée par les bornes a puisqu'on impose une intégrale égale à 1.

Lien entre impulsion et échelon

$\delta(t) = u'(t)$ donc l'impulsion est une sorte de dérivé de l'échelon. ceci nous permet également d'obtenir cette formule :

$$\int_{-\infty}^{\infty} x(s) \delta(s) ds = x(0) \quad (1.10)$$

On prouve cela de manière *peu rigoureuse* en remarquant que : pour $s \neq 0$, on a $\delta(s) = 0$ donc $x(s)\delta(s) = 0 = x(0)\delta(s)$ finalement on a :

$$\int_{-\infty}^{\infty} x(s) \delta(s) ds = \int_{-\infty}^{\infty} x(0) \delta(s) ds = x(0) \int_{-\infty}^{\infty} \delta(s) ds = x(0) \quad (1.11)$$

Décomposition en impulsions

On peut décomposer tout signal en impulsion comme ci-dessous :

$$\begin{cases} \text{temps discret} \Rightarrow x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \\ \text{temps continu} \Rightarrow x(t) = \int_{-\infty}^{\infty} x(s)\delta(t-s)ds \end{cases} \quad (1.12)$$

1.3 Opération sur les signaux

Voici un tableau résumant les différentes opérations possibles sur les signaux :

	Temps discret	Temps continu
Combinaison linéaire	$\alpha_1 x_1[n] + \alpha_2 x_2[n]$	$\alpha_1 x_1(t) + \alpha_2 x_2(t)$
Multiplication	$x_1[n]x_2[n]$	$x_1(t)x_2(t)$
Différentiation		$d^n x(t)/dt^n$
Intégration		$\int_{-\infty}^t x(\tau)d\tau$
Convolution	$x_1[n] * x_2[n]$	$x_1(t) * x_2(t)$
Dilatation	$x[n/a]$ (arrondi n/a)	$x(t/a)a > 0$
Translation	$x[n - n_0], n_0 \in \mathbb{Z}$	$x(t - t_0), t_0 \in \mathbb{R}$
Renversement	$x[-n]$	$x(-t)$

Une chose importante à voir dans ces formules est que x est un *signal*, $x[k]$ la *valeur* de ce signal en k et on a \mathcal{H} qui est un *système* prenant et donnant des signaux.

1.4 Convolution (temps discret)

La convolution est un nouvel opérateur qui nous sera très utile. Son signe est "*" et la formule qui définit cette opération est :

$$f[n] * g[n] := \sum_{k=-\infty}^{\infty} f[k]g[n-k] \quad (1.13)$$

La méthode pour trouver le résultat d'une convolution de manière **graphique**¹ est :

1. Il faut "*renverser*" une des fonctions. C'est-à-dire faire $f[n] \rightarrow f[-n]$.
2. On décale une des fonctions le plus à droite. (on prend le k_0 où après, tous les résultats de $f[k]g[n-k]$ valent 0)
3. Puis multiplier chaque point entre eux et les sommer.
4. On met le résultat sur un *graphe* au point k .
5. On décale notre fonction d'un point vers la gauche et on répète le processus.

Pour trouver de manière **calculatoire**, on applique simplement la formule 1.13.

Appliquer analytiquement

Pour résoudre facilement (mais de manière calculatoire) une convolution. On applique une méthode très simple :

1. On renverse un des signaux
2. On décompose nos signaux $x[n]$ et $y[n]$ en δ .
3. On fait une sorte de distribution tel que :

$$x[n] = \delta_1[n-a] + \dots + \delta_n[n-x] \quad x[n] = \delta_\alpha[n-a] + \dots + \delta_\eta[n-y]$$

$$\delta_1[n-a](\delta_\alpha[n-a] + \dots + \delta_\eta[n-y]) + \dots + \delta_n[n-x](\delta_\alpha[n-a] + \dots + \delta_\eta[n-y])$$

1. [Gif](#) pour mieux visualiser

1.4.1 Propriétés

Commutativité	$(f * g)[n] = (g * f)[n]$
Associativité	$(f * (g * h))[n] = ((f * g) * h)[n]$
Distributivité	$(f * (g + h))[n] = (f * g + f * h)[n]$

On également ces propriétés :

Élément neutre	$f[n] * \delta[n] = f[n]$
Décalage	$f[n] * \delta[n - n_0] = f[n - n_0], n_0 \in \mathbb{Z}$

1.5 Convolution (temps continu)

Si on a 2 signaux $f(t)$ et $g(t)$ leur convolution est donnée par (ressemble très fort à 1.4) :

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (1.14)$$

Pour calculer de manière **graphique** il faut suivre ces étapes :

1. Il faut "*renverser*" une des fonctions. C'est-à-dire faire $f(t) \rightarrow f(-t)$.
2. On décale une des fonctions le plus à droite. (on prend le τ_0 où après, tous les résultats de $f(t)g(t - \tau_0)$ valent 0)
3. Puis multiplier chaque point entre eux et on les intègre. (on prend la surface sous la courbe).
4. On met le résultat sur un *graphe* au pont τ .
5. On décale notre fonction d'une distance (pas trop loin mais pas trop proche pour ainsi avoir une nuée de points) vers la gauche et on répète le processus.

En somme, nous avons une méthode très proche du temps discret si ce n'est l'utilisation d'intégrale allant de $-\infty$ à ∞ .

1.5.1 Propriétés

On retrouve les mêmes propriétés que en temps discret.

Commutativité	$(f * g)(t) = (g * f)(t)$
Associativité	$(f * (g * h))(t) = ((f * g) * h)(t)$
Distributivité	$(f * (g + h))(t) = (f * g + f * h)(t)$

On également ces propriétés :

Élément neutre	$f(t) * \delta(t) = f(t)$
Décalage	$f(t) * \delta(t - t_0) = f(t - t_0), t_0 \in \mathbb{R}$

Chapitre 2

Fourier

2.1 La représentation de Fourier

2.1.1 Signaux continus

Stocker un signal efficacement

Un signal est composé de sinus et de cosinus à des amplitudes, phases et fréquences différentes. La manière la plus efficace pour stocker un signal est d'avoir pour chaque fréquence *multiple* de la fréquence de base ω_0 son amplitude. (on verra que en effet, on peut faire cette supposition que chaque fréquence sont des multiples de fréquences) Cela ressemble donc à ça :

$$x(t) = 1\cos(1\omega_0 t) + 2\sin(2\omega_0 t) + \frac{1}{2}\cos(2\omega_0 t) + \frac{1}{2}\sin(3\omega_0 t) + \frac{4}{5}\cos(4\omega_0 t) \quad (2.1)$$

$$\cos = [1, \frac{1}{2}, 0, \frac{4}{5}] \quad (2.2)$$

$$\sin = [0, 2, \frac{1}{2}] \quad (2.3)$$

Périodicité

On sait que $x(t)$ est **périodique** et d'énergie finie. (Périodicité de $\omega_0 = \frac{1}{T_0}$) On peut décomposer notre signal en **Série de Fourier trigonométrique** :

$$x(t) = a_0 + \sum_{k=1}^{+\infty} (2a_k \cos(k\omega_0 t) + 2b_k \sin(k\omega_0 t)) \quad (2.4)$$

$$a_k = \frac{1}{T_0} \int_0^{T_0} x(t) \cos(k\omega_0 t) dt \quad (2.5)$$

$$b_k = \frac{1}{T_0} \int_0^{T_0} x(t) \sin(k\omega_0 t) dt \quad (2.6)$$

De plus, on peut toujours décomposer un signal *périodique* en une partie **impair** et **pair**. (en Fourier, il suffit de prendre la partie *cos* donc paire et *sin* donc impaire)

Chose importante à remarquer, on va souvent tracer des graphes avec un axe y de type $2a[n]$. Ce 2 est un des prémisses de Fourier complexe.

Signaux carrés

Un signal carré est exprimé comme ci-dessous avec une période de 2π :

$$x(t) = \begin{cases} 1 & \text{si } 0 \leq t < \pi \\ -1 & \text{sinon} \end{cases} \quad (2.7)$$

Si on fait la série de Fourier, on peut se rendre compte que ce signal est composé d'une *infinité* de sinus de tel sorte que :

$$x(t) = \sum_0^{+\infty} \frac{1}{2n+1} \sin((2n+1)t) \quad (2.8)$$

Fourier Complexes

On voit que cette série est bien plus simple et est une révolution pour calculer *Fourier* :

$$x(t) = \sum_{k=-\infty}^{+\infty} X_k e^{jk\omega_0 t} \quad (2.9)$$

$$X_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{-jk\omega_0 t} dt \quad (2.10)$$

Donc on prend le conjugué pour calculer X_k . C'est ici que le coefficient 2 fait du sens, on se rappelle les formules d'Euler.

$$\cos(t) = \frac{e^{jt} + e^{-jt}}{2} \quad \sin(t) = \frac{e^{jt} - e^{-jt}}{2j} \quad (2.11)$$

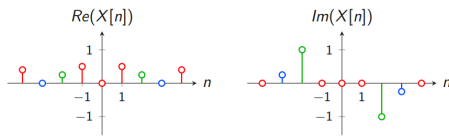
Donc le lien avec la série en Réelle est : $X_0 = a_0$, $X_k = X_{-k}^* = a_k - jb_k$. On trouve cela en injectant dans l'équation 2.10 la formule d'Euler.

Représentation de Fourier

Pour faire la représentation de Fourier, on va commencer par prendre l'équation 2.1 et la transformer :

$$x(t) = 2\left(\frac{1}{2}\cos(1\omega_0 t) + \frac{1}{4}\cos(2\omega_0 t) + \frac{4}{10}\cos(4\omega_0 t)\right) + 2\left(1\sin(2\omega_0 t) + \frac{1}{4}\sin(3\omega_0 t)\right) \quad (2.12)$$

Ensuite on injecte la formule d'Euler dans l'équation 2.12 ce qui donne :

$$x(t) = \frac{4}{10}e^{-4j\omega_0 t} + \frac{j}{4}e^{-3j\omega_0 t} + \left(\frac{1}{4} + j\right)e^{-2j\omega_0 t} + \frac{1}{2}e^{-j\omega_0 t} + \frac{1}{2}e^{j\omega_0 t} + \left(\frac{1}{4} - j\right)e^{2j\omega_0 t} - \frac{j}{4}e^{3j\omega_0 t} + \frac{4}{10}e^{4j\omega_0 t}$$


Notre signal $x(t)$ est bien défini de manière unique via ses coefficients !

Base orthogonale

Les exponentielles complexes de Fourier sont orthogonales dans l'espace de Hilbert \mathcal{H}_T équipé avec le produit scalaire :

$$\langle x, y \rangle = \frac{1}{T_0} \int_0^{T_0} x(t) y^*(t) dt \quad (2.13)$$

$$\{e^{jk\omega_0 t}\}, k \in \mathbb{Z} \rightarrow \text{est } \perp \quad (2.14)$$

2.1.2 Signaux discrets

La périodicité d'un signal change puisqu'on ne peut avoir que des valeurs \mathbb{N} . En effet, on dit qu'un signal est de période N si $x[n+N] = x[n]$. De plus, N est en $[rad]$ et pas $[\frac{rad}{s}]$ car la **pulsation fondamentale** est $\Omega_0 = \frac{2\pi}{N}$

Calcul de la série de Fourier

$$x[n] = \sum_{k=0}^{N-1} X[k] e^{jk\Omega_0 n} \quad (2.15)$$

$$X[k] = \frac{1}{N} \sum_{m=0}^{N-1} x[m] e^{-jk\Omega_0 m} \quad (2.16)$$

On a seulement besoin de N coefficient grâce à la **périodicité**. En effet, $e^{j(N+k)\Omega_0 n} = e^{jk\Omega_0 n}$

2.2 La transformée de Fourier

Cela s'applique sur les signaux *non-périodiques* et *continus*.
Il faut donc voir le signal comme ayant un $\omega = 0 = \frac{1}{T_0} = \frac{1}{\infty}$ et y appliquer la série de Fourier !

2.2.1 Calcul de la transformée

On dit que le **spectre** $X(j\omega)$ du signal $x(t)$ est sa **transformée de Fourier**.

$$X(j\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (2.17)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\omega) e^{j\omega t} d\omega \quad (2.18)$$

On appelle l'équation 2.18, la transformée de Fourier **inverse**.

Pour arriver à ce résultat, nous avons réalisés quelques modifications à la série de Fourier :

$$\omega_k = k\omega_0 = \frac{2\pi k}{T_0} \Rightarrow \omega_{k+1} - \omega_k = \Delta\omega = \frac{2\pi}{T_0} \quad (2.19)$$

$$\int_0^T \approx \int_{-\frac{T}{2}}^{\frac{T}{2}} \quad \text{Pour "couvrir" tout le domaine} \quad (2.20)$$

$$x(t) = \sum_{k=-\infty}^{\infty} X_k e^{j\omega_0 k t} \quad (2.21)$$

$$= \sum_{k=-\infty}^{\infty} \left\{ \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t) e^{-jk\omega_0 t} dt \right\} e^{jk\omega_0 t} \quad (2.22)$$

$$= \sum_{k=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t) e^{-jk\omega_0 t} dt \quad e^{jk\omega_0 t} \Delta\omega \quad (2.23)$$

On remarque qu'on a 2 parties intéressantes dans l'équation 2.23. On a une partie qui évolue dans le temps, c'est notre *signal continu*, c'est notre **spectre**. L'autre partie dépend de la fréquence et c'est la *reconstruction du signal*.

La transformée de $x(t) = e^{-2|t|}$ est $X(j\omega) = \frac{4}{4 + \omega^2}$.

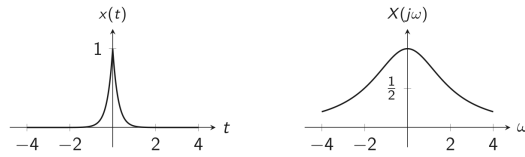


FIGURE 2.1 – à gauche, un signal pair à droite, le spectre

2.2.2 Calcul discret

On a un signal $x[n]$ qui s'exprime de manière unique selon ses fréquences :

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\Omega}) e^{j\Omega n} d\Omega \quad (2.24)$$

$$X(e^{j\Omega}) = \sum_{m=-\infty}^{\infty} x[m] e^{-j\Omega m} \quad (2.25)$$

L'équation donne la [transformée](#) de Fourier.

2.2.3 Astuce

Ne jamais oublier la forme complexe des fonctions sinus et cosinus et repasser en forme complexe. Une fonction qui revient souvent est le sinus *cardinal* qui peut être transformé comme suit :

$$\text{sinc}(u) = \frac{\sin(u)}{u}$$

2.3 Propriétés de Fourier

2.3.1 Dualité

$$x(t) \longleftrightarrow X(j\omega) \quad (2.26)$$

$$X(j\omega) \longleftrightarrow 2\pi x(-\omega) \quad (2.27)$$

Donc on échange $t \rightarrow -\omega$ et $\omega \rightarrow t$ et on échange les fonctions.

Démonstration

$$\begin{aligned} X(j\omega) &= \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \\ x(t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\omega) e^{j\omega t} d\omega \\ x(-t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\omega) e^{-j\omega t} d\omega \\ 2\pi x(-t) &= \int_{-\infty}^{\infty} X(j\omega) e^{-j\omega t} d\omega \\ 2\pi x(-\omega) &= \int_{-\infty}^{\infty} X(jt) e^{-j\omega t} dt \end{aligned}$$

2.3.2 Linéarité

En effet,

$$\begin{aligned} x_k(t) &\longleftrightarrow X_k(j\omega) \\ \sum_k x_k(t) &\longleftrightarrow \sum_k X_k(j\omega) \end{aligned}$$

2.3.3 Translation

On a :

$$\begin{aligned} x(t) &\longleftrightarrow X(j\omega) \\ x(t - t_0) &\longleftrightarrow X(j\omega)e^{-j\omega t_0} \end{aligned}$$

Il faut bien retenir qu'une translation en temporelle amène à une exponentielle en phasorielle. Combiner à la propriété de dualité venant de 2.3.1 est très puissant et utile.

On doit aussi noter que cela ne change **pas** l'amplitude du spectre mais change sa **phase**.

2.3.4 Modulation

C'est quand on combine la translation et la dualité :

$$\begin{aligned} x(t) &\longleftrightarrow X(j\omega) \\ e^{j\omega_0 t} x(t) &\longleftrightarrow X(j(\omega - \omega_0)) \end{aligned}$$

Cette modulation va elle modifier l'amplitude du signal car déplace son "espace" phasorielle revient à déplacer ses fréquences et en ajoute.

$$\begin{aligned} x(t) &\longleftrightarrow X(j\omega) \\ \cos(\omega_0 t)x(t) &\longleftrightarrow \frac{1}{2}(X(j(\omega - \omega_0)) + X(j(\omega + \omega_0))) \end{aligned}$$

On reconnait clairement les formules d'Euler.

2.3.5 Différentiation

Dériver un signal à **énergie finie** revient à **amplifier** ses **hautes** fréquences.

$$\begin{aligned} x(t) &\longleftrightarrow X(j\omega) \\ \frac{d^k x}{dt^k}(t) &\longleftrightarrow (j\omega)^k X(j\omega) \end{aligned}$$

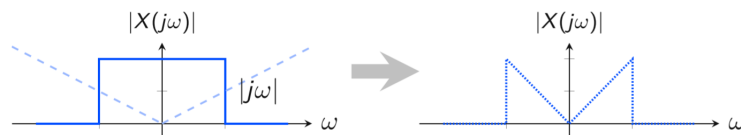


FIGURE 2.2 – Exemple de transformation de Fourier

2.3.6 Multiplication par un monôme

Par dualité, faire ceci :

$$\begin{aligned} x(t) &\longleftrightarrow X(j\omega) \\ t^n x(t) &\longleftrightarrow j^n \frac{d^n X}{d\omega^n}(j\omega) \end{aligned}$$

2.3.7 Intégration

Donc intégrer, augmente les basses fréquences :

$$x(t) \longleftrightarrow X(j\omega)$$
$$\sum_{-\infty}^t x(t)dt \longleftrightarrow \frac{X(j\omega)}{j\omega} + \pi X(0)\delta(\omega)$$

La partie en rouge est dû au fait qu'un résultat d'une intégration à toujours une constante d'intégration (le fameux $+C$)

2.3.8 Dilatation

Quand on *dilate* un signal, on *comprime* son espace spectre.

$$x(t) \longleftrightarrow X(j\omega)$$
$$x(t/a) \longleftrightarrow |a|X(aj\omega)$$

Un truc utile est de penser au vidéo au ralenti où on *dilate* le temps et on *comprime* son spectre donc on entend plus les basses fréquences.

Parseval

On **conserve** l'énergie du signal grâce à son spectre. donc :

$$x(t) \longleftrightarrow X(j\omega)$$
$$\int_{-\infty}^{\infty} |x(t)|^2 dt \longleftrightarrow \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(j\omega)|^2 d\omega$$

2.3.9 Renversement

Renverser un signal revient à renverser son spectre.

$$x(t) \longleftrightarrow X(j\omega)$$
$$x(-t) \longleftrightarrow X(-j\omega)$$

C'est une dilatation par $a = -1$.

Complexe conjugué

Le spectre d'un signal conjugué est le conjugué renversé du spectre du signal.

$$x(t) \longleftrightarrow X(j\omega)$$
$$x^*(t) \longleftrightarrow X^*(-j\omega)$$

2.4 Transformée usuelle

On utilisera des propriétés vu à la section 2.3.

Delta de Dirac

C'est plutôt simple, il faut appliquer simplement la formule :

$$\begin{cases} \delta(t) = 0 \text{ si } t \neq 0 \\ \int_a^{-a} \delta(t) dt = 1 \forall a > 0 \\ \int_{-\infty}^{\infty} x(t) \delta(t) dt = x(0) \end{cases} \quad (2.28)$$

Via toutes ces propriétés on voit facilement que :

$$X(j\omega) = \int_{-\infty}^{\infty} \delta(t) e^{-j\omega t} dt = 1(e^0) = 1 \quad (2.29)$$

Delta de Kronecker

Même idée que à 2.4 si ce n'est qu'on est en discret. Le résultat reste le même.

2.4.1 Train d'impulsions (peigne) de Dirac

On a donc une fonction qui ressemble à cela : (d'où le nom)

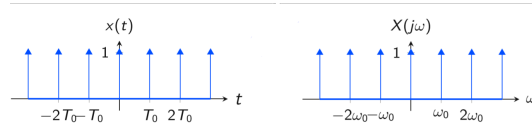


FIGURE 2.3 – peigne de Dirac (avec un petit spoiler)

formellement un peigne de Dirac est :

$$x(t) = \sum_{k \in \mathbb{Z}} \delta(t - kT_0) \quad (2.30)$$

et son $X(j\omega)$ est :

$$x(t) \longleftrightarrow X(j\omega) = \omega_0 x_{\omega_0}(\omega) \text{ avec } \omega_0 = \frac{2\pi}{T_0} \quad (2.31)$$

Preuve

$$\begin{aligned} X(j\omega) &= \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \\ &= \sum_{k=-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(t - kT_0) e^{-j\omega t} dt \\ &= \sum_{k=-\infty}^{\infty} e^{-j\omega kT_0} \text{ car on a une énergie finie} \\ \mathcal{F}(\delta(t - kT_0)) &= e^{-j\omega kT_0} \end{aligned}$$

Ensuite, on va refaire une série de Fourier :

$$\begin{aligned} x(t) &= \sum_{l=-\infty}^{\infty} X_l e^{j\omega_0 l t} & X_l &= \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t) e^{-j\omega_0 l t} dt \\ &= \frac{1}{T_0} \sum_{l=-\infty}^{\infty} e^{j\omega_0 l t} & &= \frac{1}{T_0} \int \delta(t) e^{-j\omega_0 l t} dt = \frac{1}{T_0} \end{aligned}$$

Ensuite, on utilise la dualité :

$$\begin{aligned}
 x(t) &\longleftrightarrow X(j\omega) \\
 X(jt) &\longleftrightarrow 2\pi x(-\omega) \\
 \mathcal{F}(e^{j\omega_0 l t}) &= 2\pi \delta(\omega - \omega_0 l) \\
 x(t) &= \frac{1}{T_0} \sum_{l=-\infty}^{\infty} 2\pi \delta(\omega - l\omega_0) = \omega_0 \sum (\delta(\omega - l\omega_0) = \omega_0 x(\omega)
 \end{aligned}$$

2.4.2 Fonction échelon

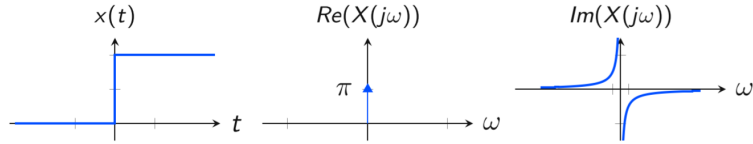


FIGURE 2.4 – Transformée de la fonction échelon

Sa transformée de Fourier est :

$$x(t) = u(t) \longleftrightarrow X(j\omega) = \pi\delta(t) + \frac{1}{j\omega} \quad (2.32)$$

Preuve

On utilise l'intégration car on sait que :

$$\begin{aligned}
 u(t) &= \int_0^{\infty} \delta(t) dt \\
 \mathcal{F}(\delta(t)) &= 1 \\
 u(j\omega) &= \frac{1}{j\omega} + \pi\delta(\omega)
 \end{aligned}$$

2.4.3 Fonction fenêtre

Cette fonction, correspond à 2 impulsions. tel que :

$$\Pi = u(t+1) - u(t-1) \quad (2.33)$$

Ces fonctions sont des filtres *passes-bandes* et à pour transformé de Fourier :

$$x(t) = \Pi(t) \longleftrightarrow X(j\omega) = 2\text{sinc}(t) = 2\frac{\sin(\omega)}{\omega}$$

Preuve

On utilise l'*addition* de 2 échelons *translatés*.

2.4.4 Fonction signe

La fonction signe est :

$$\text{sign}(t) = \begin{cases} -1 & t < 0 \\ 0 & t = 0 \\ 1 & t > 0 \end{cases} \quad (2.34)$$

et à pour transformée de Fourier :

$$x(t) = \text{sign}(t) \longleftrightarrow X(j\omega) = \frac{2}{j\omega}$$

Cela se prouve avec la dilatation et la translation de la fonction échelon.

2.5 Tableau de Check

Domaine temporel	Domaine spectral
$x(t) = u(t)$	$X(j\omega) = \frac{1}{j\omega} + \pi\delta(\omega)$
$x(t) = \delta(t)$	$X(j\omega) = 1$
$x(t) = 1$	$X(j\omega) = 2\pi\delta(\omega)$
$x(t) = \begin{cases} 1 & \text{si } t \leq T_0 \\ 0 & \text{sinon} \end{cases}$	$X(j\omega) = \frac{2T_0 \sin(\omega T_0)}{\omega T_0}$
$x(t) = \frac{1}{t\pi} \sin(Wt)$	$X(j\omega) = \begin{cases} 1 & \text{si } \omega \leq W \\ 0 & \text{sinon} \end{cases}$
$x(t) = e^{-at}u(t), \Re(a) > 0$	$X(j\omega) = \frac{1}{a+j\omega}$
$x(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT)$	$X(j\omega) = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta(\omega - k\frac{2\pi}{T})$

Transformées de signaux discrets

Domaine temporel	Domaine spectral
$x[n] = u[n]$	$X(e^{j\Omega}) = \frac{1}{1-e^{-j\Omega}} + \pi \sum_{p \in \mathbb{Z}} \delta(\Omega - 2\pi p)$
$x[n] = \delta[n]$	$X(e^{j\Omega}) = 1$
$x[n] = \begin{cases} 1 & \text{si } n \leq M \\ 0 & \text{sinon} \end{cases}$	$X(e^{j\Omega}) = \frac{\sin(\Omega(M+1/2))}{\sin(\Omega/2)}$
$x[n] = \frac{1}{n\pi} \sin(Wn), 0 < W \leq \pi$	$X(e^{j\Omega}) = \begin{cases} 1 & \text{si } \Omega \leq W \\ 0 & \text{sinon} \end{cases}$
$x[n] = \alpha^n u[n], \alpha < 1$	$X(e^{j\Omega}) = \frac{1}{1-\alpha e^{-j\Omega}}$

Chapitre 3

L'échantillonnage

L'échantillonnage est quelque chose qui est partout autour de nous. C'est le fait de transformer un signal continu (*son, vidéo, ...*) en un signal discret. (afin de le stocker sur un pc, ...) C'est donc un processus vital à l'ère de l'informatique.

3.1 Lien entre le continu et le discret

Donc, pour sauvegarder un signal en discret, il faut *ponctuellement* le sauvegarder. On appelle cela [l'échantillonnage](#)

3.1.1 L'échantillonnage

On relève le signal discret toutes les T_e secondes. Notre signal discret correspond à $x[n]$. Donc la transformation continu \rightarrow est *trivial*.

Une fois qu'on a notre signal *discret*, on peu le retransformer en un signal discret ! En effet, on peut lier $x(t)$ à un peigne/-train de Dirac.

On remarque que $x(nT_e) = x[n]$.

Pas d'échantillonnage : T_e

Fréquence d'échantillonnage : $f_e = \frac{1}{T_e}$

$$x[n] = x(nT_e)$$

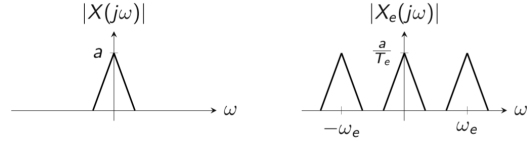
$$x_e(t) = x(t) \sum_{n \in \mathbb{Z}} \delta(t - nT_e) = \sum_{n \in \mathbb{Z}} x(nT_e) \delta(t - nT_e)$$

Transformé de Fourier via continu

Pour trouver la transformée de Fourier de notre $x_e(t)$ on réalise ceci :

$$\begin{aligned} x_e(t) &= x(t) \sum_{n \in \mathbb{Z}} \delta(t - nT_e) \\ X_e(j\omega) &= \frac{1}{2\pi} \left[X(j\omega) * \frac{2\pi}{T_e} \sum_{k \in \mathbb{Z}} \delta(\omega - \frac{2\pi}{T_e} k) \right] \quad \frac{2\pi}{T_e} = \omega_e \\ &= \frac{1}{T_e} \left[X * \sum_{k \in \mathbb{Z}} \delta(-\omega_e k) \right](j\omega) \\ &= \frac{1}{T_e} \sum_{k \in \mathbb{Z}} X(j(\omega - k\omega_e)) \end{aligned}$$

Donc notre ω_e représente la période. On observe également une différence entre $|X(j\omega)|$ et $|X_e(j\omega)|$. En effet, on peut remarquer une répétition tous les $\omega_e = 2\pi f_e$. Cette contrainte va être très importante pour le taux d'échantillonnage et pour le repli spectral (3.2)



Transformé de Fourier via discret

$$\begin{aligned} x_e(t) &= \sum_{n \in \mathbb{Z}} x[n] \delta(t - nT_e) \\ X_e(j\omega) &= \sum_{n \in \mathbb{Z}} x[n] e^{-j\omega n T_e} \\ &= X(e^{j\Omega})|_{\Omega=\omega T_e} \end{aligned}$$

On retrouve également ce phénomène de *répétition* pouvant mener à un enchevêtrement.

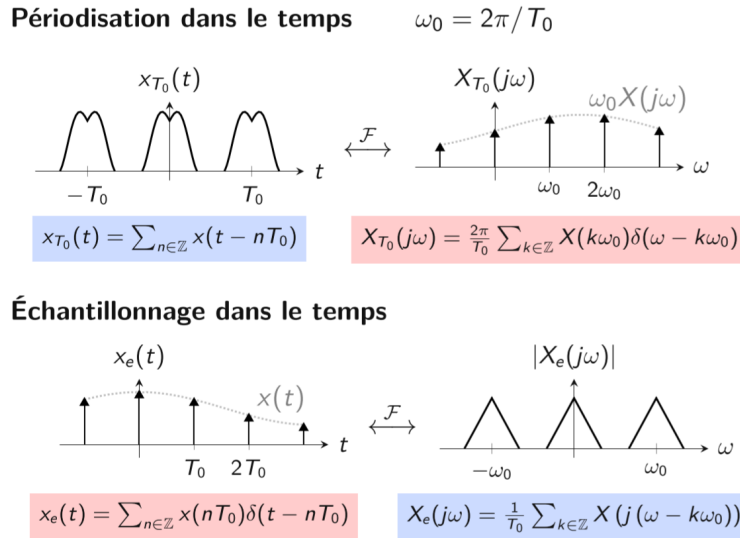


FIGURE 3.1 – Dualité de Fourier

3.2 Repli spectral

Comme notre transformé de Fourier se répète tous les ω_e , cela indique que le pas de temps *d'échantillonnage* est primordial. Un pas trop *petit* et les répétitions se **superposent**. On appelle cela le **repli spectral** (*aliasing*).

On peut penser aux roues d'une voiture, qui si le nombre de *FPS* n'est pas assez élevé pour le nombre de rotation par seconde, nous donnera l'impression qu'elle tourne à l'envers.

3.2.1 Bien échantillonner

Avec l'idée de la *voiture*, on peut déjà poser le fait qu'un bon échantillonnage est $2T_e \leq T$. Ainsi on verra la rotation.

Théorème de Shannon

Une fonction (à bande limitée) qui ne contient pas de fréquences supérieures à f_{max} est complètement déterminée par ses échantillons pour autant que :

$$f_e \geq 2f_{max} \longleftrightarrow \omega_e \geq 2\omega_{max}$$

3.2.2 Reconstruire un signal

Si le théorème de Shanon (3.2.1) est bien respecté, on peut donc reconstituer un signal en *continu* en filtrant son *spectre discret*.

$$X(j\omega) = X_e(j\omega)T_e \Pi\left(\frac{\omega}{\omega_c}\right) \quad (\omega_{max} \leq \omega_c \leq \omega_e - \omega_{max})$$

Il existe également une formule s'intitulant la [formule de Shanon](#). On rappelle que $x[n]$ correspond au signal échantillonné à un interval T_e :

$$\sum_{n \in \mathbb{Z}} x[n] \text{sinc}\left(\frac{t - nT_e}{T_e}\right)$$

L'utilité du sinus [cardinal](#) est qu'il interpole les points $(t, x(nT_e))$. Il existe d'autres fonctions d'interpolations qui donnent des résultats différents.

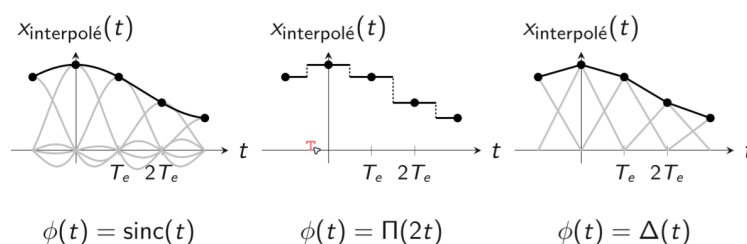


FIGURE 3.2 – Différentes interpolations

3.2.3 Généralisation de Shanon

Donc de manière plus général, on peut reconstruire tous signaux tant qu'il n'y a pas de recouvrement spectral. Par exemple des signaux plus complexes possédant **plusieurs** fréquences et qui sont de formes variées. On doit simplement respecter la formule de Shanon et bien lier la fréquence positive et sa fréquence négative. (*dualité fréquentielle*)

3.3 Sur- et Sous-échantillonnage

	Variation du nombre d'échantillon sur une <i>même</i> période	Variation de la durée de prise de donnée
Sous-échantillonnage	On a une même largeur de bande mais un rapprochement des spectres	On a une sorte de dilatation des spectres en phasorielles donc plus facile d'avoir de l' <i>aliasing</i> même un même espacement entre les répétitions.
Sur-échantillonnage	Si on rajoute des données inutiles (ex : des 0) on a un rétrécissement des bandes mais un même espacement	On a une même bande des spectres et un rapprochement des spectres entre eux.

Chapitre 4

Discrete Fourier Transform (DFT)

On est motivé à trouver la *DFT* car sur les PC, on manipule que des signaux discrets. Donc on veut passer de signal continu à un signal discret, le manipuler et pouvoir le restituer en continu. De plus, un signal ne peut être infini.

Candidat :

1. Transformé de Fourier : signal x est continu et sur un support infini.
2. Série de Fourier : signal x est discrète et sur un support infini.
3. Transformé de Fourier en temps discret : est continue.
4. Série de Fourier discrète : on a besoin d'un signal discret périodique et nous renvoie un signal périodique

Donc de ce qu'on a vu, rien ne fonctionne, on doit introduire une nouvelle *transformée*, la **DFT** ou *transformée de Fourier discrète*.

4.1 Définition

Depuis une suite M -périodique $x[n]$ on construit une suite $X_{DFT}[k]$ qui est aussi M -périodique et appelé la *transformée de Fourier discrète* ou DFT de $x[n]$:

$$X_{DFT}[k] = \sum_{n=0}^{M-1} x[n] e^{-2\pi j k n / M} \quad (4.1)$$

On peut aussi représenter cela via une matrice de Vandermonde :

$$\begin{bmatrix} X_{DFT}[0] \\ X_{DFT}[1] \\ \vdots \\ X_{DFT}[M-1] \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-2\pi j \frac{1}{M}} & \dots & e^{-2\pi j \frac{M-1}{M}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-2\pi j \frac{M-1}{M}} & \dots & e^{-2\pi j \frac{(M-1)^2}{M}} \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[M-1] \end{bmatrix} \quad (4.2)$$

non-périodique

Si $x[n]$ est **non-périodique** on va définir sa version périodisé.

$$\begin{aligned} x_M[n] &= \sum_{m \in \mathbb{Z}} x[n + mM] \\ X(e^{j\Omega_k}) &= \sum_{n \in \mathbb{Z}} x[n] e^{-2\pi j k n / M} \\ &= X_{M,DFT}[k] \end{aligned}$$

Et donc notre DTFT de $x[n]$ évaluée en $\Omega_k = 2\pi k / M$ **TODO démonstration**

En pratique

Ils existent 3 grandes familles de signaux qui nous intéressent pour la *DFT* :

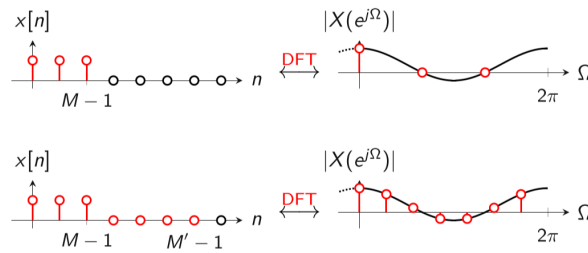
1. Signaux périodiques de période M
2. Signaux à support fini dans $\{0, \dots, M-1\}$
3. Signaux à support infini.

Donc pour le point 3, on doit *tronquer* notre signal pour le périodiser.

Zero padding

C'est le principe du *ré-échantillonnage* de la DTFT. On prend un signal $x[n]$ qui a un support fini et non-périodique.

On va choisir une taille de périodisation M' tel que $M' > M$ en rajoutant des zéros. On évalue donc la DTFT de $x[n]$ en d'autres fréquences.



Propriétés de la DFT

Opération	$x[n]$	$X_{DFT}[k]$
Combinaison linéaire	$\sum_i \alpha_i x_i[n]$	$\sum_i \alpha_i X_{DFT,i}[k]; \alpha_i \in \mathbb{R}$
Renversement	$x[-n]$	$X_{DFT}[-k]$
Complexe conjugué	$x[n]^*$	$X_{DFT}[-k]^*$
Translation	$x[n - n_0]; n_0 \in \mathbb{Z}$	$e^{-jk \frac{2\pi n_0}{M}} X_{DFT}[k]$
Modulation	$e^{jk \frac{2\pi k_0}{M}} x[n]; k_0 \in \mathbb{Z}$	$X_{DFT}[k - k_0]$
Convolution cyclique	$\sum_{m=0}^{M-1} x_1[m] x_2[n - m]$	$(X_{DFT,1} \cdot X_{DFT,2})[k]$
Multiplication	$(x_1 \cdot x_2)[n]$	$\frac{1}{M} \sum_{m=0}^{M-1} X_1[m] X_2[k - m]$
Parseval	$\sum_{n=0}^{M-1} x[n] ^2$	$\frac{1}{M} \sum_{m=0}^{M-1} X_{DFT}[k] ^2$

4.2 Transformée Rapide

On doit en moyenne réaliser $8M^2$ opérations (avec M la portée finie du signal)

En 1965, Cooley et Tukey ont proposé FFT ou *Fast Fourier Transform* qui est basé sur la [décimation temporelle](#)

Pour simplifier, on factorise M et on divise la somme de la *DFT* en sous-sommes qui correspondent à des DFT de plus petites tailles.

Deuxième partie

Systemes

Chapitre 5

Système LIT

Un système est une entité qui prend *un ou plusieurs signaux* en entrée et produit *de nouveaux signaux* en sortie.

Exemple : $H\{x[n]\} = x[n] + x[n-1]$. Un système est par exemple : *une radio, une caméra, une voiture, ...*

5.1 LIT

Un système *Linéaire et Indépendant du Temps* ou **LIT** est, comme son nom l'indique, linéaire donc :

$$\mathcal{H}\{a_1x_1 + \dots + a_Nx_N\} = a_1\mathcal{H}\{x_1\} + \dots + a_N\mathcal{H}\{x_N\} \quad (5.1)$$

Et un système est *invariant temporelle* donc :

$$\begin{cases} \mathcal{H} \text{ est invariant si } \forall t_0 \in \mathbb{N} \\ \mathcal{H}\{x\}[n] = y[n] \Rightarrow \mathcal{H}\{x[n - n_0]\} = y[n - n_0] \end{cases} \quad (5.2)$$

On remarque qu'on peut ré-écrire tous signaux via une somme d'impulsions. De plus, si \mathcal{H} est linéaire alors :

$$\mathcal{H}\{x\} = \mathcal{H}\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\} = \sum_{k=-\infty}^{\infty} x[k]\mathcal{H}\{\delta[n-k]\} \quad (5.3)$$

Si \mathcal{H} est *invariant* au temps et qu'on pose $h := \mathcal{H}\{\delta\}$

$$\mathcal{H}\{\delta[n-k]\} = \mathcal{H}\{\delta\}[n-k] = h[n-k]$$

$$\mathcal{H}\{x\} = \sum_{k=-\infty}^{\infty} x[k]h[n-k] =: x * h$$

A noter que "*" fait référence à la convolution, sujet abordé à la section 1.4.

Il est important de noter que toutes ces propriétés et caractéristiques des systèmes *LIT* en **temps discret** sont également valables et ont un équivalent en **temps continu**.

5.2 Réponse impulsionnelle

La réponse impulsionnelle d'un système *LIT* \mathcal{H} est la réaction du système à une *impulsion d'entrée* ($\delta[0]$) on le note h .

Pour **tout** système *LIT*, le signal de sortie est le résultat de la convolution entre *le signal d'entrée* et *la réponse impulsionnelle*. Cela est d'autant plus utile grâce à la propriété d'un tel système et sa linéarité.

$$y[n] = x[n] * h[n] \quad (5.4)$$

$$y(t) = x(t) * h(t) \quad (5.5)$$

Réponse indicielle

C'est comme la *réponse impulsionnelle* mais quand on fait passer un **échelon** dans notre système.

Réponse fréquentielle

Pour établir cette réponse, on applique une transformée de Fourier à la réponse impulsionnelle **OU** on utilise la propriété de convolution et multiplication :

$$H(j\omega) = \mathcal{F}(h(t))$$

$$H(j\omega) = \frac{Y(j\omega)}{X(j\omega)}$$

Pour plus d'informations sur les réponses fréquentielles, il faut aller voir 6.2.

5.3 Type de système

Système sans mémoire	Si la sortie du système à <i>un temps donné</i> ne dépend que de l'entrée à cet instant .
Système causal	Le système ne dépend pas de ce qui se passe dans le <i>futur</i>
Système stable ou (<i>BIBO</i>)	Entrée <i>bornée</i> donne une sortie <i>bornée</i>
Système inversible	On sait <i>retrouver</i> l'entrée en ayant la <i>sortie</i> .

5.4 Modélisation et représentation des systèmes

Comme vu précédemment, la réponse impulsionnelle est le résultat du système étant perturbé par une impulsion.

5.4.1 Inconvénient

1. Fonction de taille *infinie* et représentation donc *peu simple*.
2. La modélisation d'un système ne mène généralement pas à une réponse impulsionnelle.
3. On doit connaître l'entrée depuis $-\infty$.

5.4.2 Représentation

Il existe 3 grandes façons de représenter un système. Tout d'abord la méthode *équation différentielle d'entrée-sortie*. Pour la suite des exemples, j'utiliserai un circuit *RLC* comme montré ci-contre.

équation différentielle d'entrée-sortie est une somme des dérivées comme montré dans l'équation 5.6. C'est plutôt facile de trouver les équations mais on fait face à un problème, l'opération *dérivée* n'existe pas dans le monde réel, il faut un opérateur intégrateur.

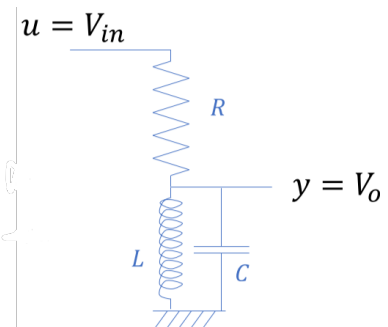


FIGURE 5.1 – Circuit RLC

Ensuite, nous avons la *représentation d'état* qui utilise des matrices pour former les équations différentielles comme nous voyons à l'équation 5.7

La dernière représentation type est le *schéma bloc* qui est visuel et qui utilise lui des blocs intégrateurs au lieu de dérivé comme montré à la figure 5.2.

$$\ddot{y} + \frac{1}{CR}\dot{y} + \frac{1}{LC}y = \frac{1}{CR}\dot{u} \quad (5.6)$$

$$\frac{d}{dt} \begin{pmatrix} V_0 \\ I_L \end{pmatrix} = \begin{pmatrix} -1/RC & -1/C \\ 1/L & 0 \end{pmatrix} \begin{pmatrix} V_0 \\ I_L \end{pmatrix} + \begin{pmatrix} 1/RC \\ 0 \end{pmatrix} u \quad (5.7)$$

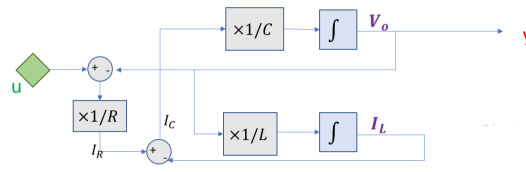


FIGURE 5.2 – Représentation *bloc* du système à la figure 5.1

5.4.3 Équation différentielle entrée-sortie

La forme générale de ces équations est de ce type :

$$\sum_{k=0}^N a_k \left(\frac{d^k}{dt^k} \right) y(t) = \sum_{k=0}^M b_k \left(\frac{d^k}{dt^k} \right) u(t) \quad (5.8)$$

Quelque chose à bien remarquer est que ces équations ne modélise *qu'une partie* d'un système *LIT*. Par exemple, on ne peut pas représenter un *délai* ce qui également rare dans la réalité.

Une chose à remarquer est que l'opérateur *dérivé* peut se "*démultiplier*" et possède une *associativité* et *commutativité*. Ainsi, on peut avoir une représentation dite "*polynomiale*" comme ci-dessus qui est une autre écriture de l'équation 5.8.

$$p\left(\frac{d}{dt}\right)y(t) = q\left(\frac{d}{dt}\right)u(t) \quad (5.9)$$

Puis après, pour résoudre ce genre d'équation, on utilise des méthodes classiques vu au cours d'Analyse donc, solution homogène et particulière ...

Réponse libre et forcée

Une réponse *libre* est la solution de l'équation homogène, donc quand $u(t) = 0$ à l'équation 5.9 et on garde les **mêmes conditions initiales**. En somme c'est la représentation de l'impact des *CI*.

Une réponse *forcée* est l'équation 5.9 mais avec les *conditions initiales* nulles. Donc on s'intéresse à l'impact de l'entrée sans les conditions initiales. La somme de la réponse *libre* et *forcée* nous donne la réponse générale.

Stabilité

On peut avoir une intuition sur la stabilité de notre système en posant $y_H(t)$ qui équivaut à :

$$y_H(t) = \sum_i \alpha_i e^{r_i t} \rightarrow r_i \text{ correspond aux racines de } p(z) \text{ de 5.9} \quad (5.10)$$

Si la partie réelle de $r_i < 0 \forall i$ alors on a une exponentielle décroissante donc **stable**. On appelle ce genre de système *BIBO stable* ou *Bounded Input Bounded Output*.

En revanche, si $r_i > 0 \exists i$ donc on a au moins une exponentielle croissante créant une *instabilité*.

Si $r_i = 0 \exists i$ on dit qu'on a une *stabilité marginale* ou *instabilité*. Cela dépendra de la **multiplicité** et de $te^{r_i t}$.

Linéarité de l'entrée

Avec cette représentation polynomiale, on peut facilement voir qu'on a une linéarité de l'entrée nous permettant de simplifier différent calcul.

Avantages et Inconvénients

2 représentations qui sont 5.8 et 5.9. Les avantages :

- Représentation compacte.
- Conditions initiales claires.
- Facile de la transformer dans d'autres représentations.

Les désavantages :

- On peut perdre la représentation physique.

5.4.4 Schéma Bloc

Comme son nom l'indique, le schéma bloc utilise des "*blocs*" pour représenter notre système. Ci-dessus on peut voir les composants de base composant ce type de schéma

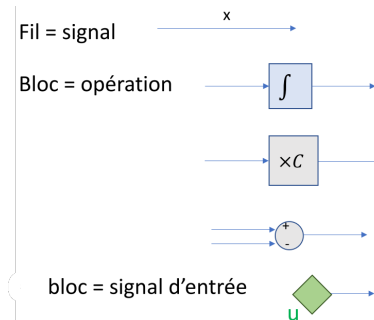


FIGURE 5.3 – Liste reprenant les blocs de base

Dans le cadre des systèmes *LIT*, on se restreint souvent à l'addition, multiplication et intégration. On ne réalise que des opérations *linéaires*.

Avantages et Inconvénients

Les avantages :

- Représentation très intuitive.
- Proche de l'implémentation réelle du système.
- On peut plus facilement réfléchir sur le *design* de notre système.
- Très modulaire.

Les désavantages :

- Lien moins clair avec la solution.

De plus, on peut voir comme un *avantage* ou *inconvénient* le fait de pouvoir voir l'évolution des signaux entre chaque bloc plutôt qu'une sorte de boîte noire "*entrée-sortie*". De plus, on peut réaliser de bien des manières des circuits.

5.4.5 Représentation d'état

Dans ce type de représentation, on a **état** x qui est un *vecteur* comportant toutes les infos internes de notre système. On **entrée** u qui est un *signal* extérieur affectant le système. Finalement on a une **sortie** y qui est un signal qu'on peut accéder depuis l'extérieur.

$$\begin{cases} \text{évolution : } \frac{d}{dt}x(t) = Ax(t) + Bu(t) \\ \text{sortie : } Cx(t) + Du(t) \end{cases} \quad (5.11)$$

Solution

La solution *homogène* de $e^{\lambda_i t}$ avec λ_i étant les valeurs propres de l'équation. Si la partie réelle des racines est négative pour tout λ_i .

État non-unique

Avec cette représentation, on peut facilement modifier les vecteurs et signaux pour rendre les équations plus simples. Cela n'a aucun impacte et rend les équations plus logiques pour un certain sens "*réel*".

$$\begin{cases} z = Tx \Rightarrow \frac{d}{dt}z = T \frac{d}{dt}x = TAT^{-1}z(t) + TBu(t) \\ y = Cx(t) + Bu(t) = CT^{-1}z(t) + Du(t) \end{cases} \quad (5.12)$$

Si la matrice A est diagonalisable, on peut réaliser un **découplage** et obtenir ainsi un mode dit *découplé*. On peut également faire des *blocs de Jordan* pour diagonaliser le tout :

$$\frac{d}{dt}z_i = \lambda_i z_i + \tilde{B}_i u \quad (5.13)$$

5.4.6 Passage de représentation

5.4.7 Temps discret

Pour passer du temps continu au temps discret, il faut transformer $\frac{d}{dt}$ en l'opérateur de *décalage* D :

$$Dx[n] = x[n-1] \quad (5.14)$$

Ce qui nous permet d'établir l'équation de *différence*

$$\begin{cases} \sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k u[n-k] \\ p(D)y = q(D)u \end{cases} \quad (5.15)$$

Ce qui nous donne pour solution homogène :

$$y_H[N] = \sum_i c_i r_i^N \quad (5.16)$$

On a une décroissance donc *stabilité* si $|r_i| < 1$ et une croissance si $|r_i| > 1$

De plus, on remplace le bloc *intégrateur* du temps discret en bloc D^{-1} . La ré-écriture de l'équation 5.12 en temps discret :

$$\begin{cases} x[n+1] = Ax[n] + Bu[n] \\ y[n] = Cx[n] + Du[n] \end{cases} \quad (5.17)$$

On approxime un système en temps *continu* en temps *discret* ssi :

$$A_d \approx A_c \Delta t \text{ si } \Delta t \text{ petit} \quad (5.18)$$

5.4.8 Résumé

1. Réponse impulsionnelle, universelle mais peu maniable \Rightarrow On voit que l'entrée et sortie et **Pas de CI**
2. Équation différentielle entrée-sortie \Rightarrow On voit que l'entrée et sortie et on a des **CI**
3. Représentation d'état (matrice) \Rightarrow On voit l'intérieur et on a des **CI**
4. Schéma Bloc (très concret) \Rightarrow On voit l'intérieur et on a des **CI**

5.4.9 Existence des systèmes LIT

Une forme usuelle des systèmes LIT dans la vraie vie est de type $\dot{x}(t) = f(x(t), u(t), t)$.

Invariance temporelle : tout système fait face à l'usure mais on estime que sur la période d'observation, l'usure est minime et peut être ignorée.

Linéarité : aucun système n'est pas linéaire. Cela peut être dû à des *imperfections* ou si on augmente *énormément* l'entrée ce qui change le comportement du système.

Chapitre 6

Filtrage et Bode

6.1 Filtre

On se souvient, que convoluer 2 signaux revient à multiplier leur spectre. De plus, dans un système **LIT** le résultat d'un signal x est : $y = h * x$. Donc, la plupart du temps quand un programme informatique fait un système LIT, il va faire la *transformée de Fourier* pour simplement multiplier et pas avoir des sommes de multiplications.

Donc en Résumé :

$$\begin{aligned} Y(t) = H(t) * X(t) &\longleftrightarrow Y(j\omega) = H(j\omega)X(j\omega) \\ &= |H(j\omega)||X(j\omega)|e^{j(\arg\{H(j\omega)\} + \arg\{X(j\omega)\})} \end{aligned}$$

Donc, il paraît clair qu'un système modifie le contenu fréquentiel de notre signal X .

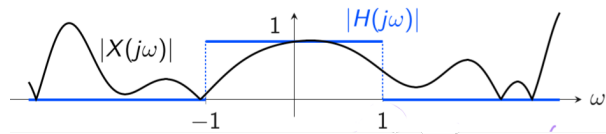


FIGURE 6.1 – Exemple de système LIT

Le fait de modifier le *contenu fréquentiel* est appelé le **filtrage**.

6.1.1 Type de filtre

Il existe 3 grands types de filtres :

1. **Les filtres passe-bas** : ils vont atténuer et annuler les *hautes-fréquences* pour ne laisser passer que les *basses*.
2. **Les filtres passe-haut** : ils vont atténuer et annuler les *basses fréquences* et ne laisser passer que les *hautes*.
3. **Les filtres passe-bande** : ils ne vont laisser passer qu'une partie précise de fréquence. On peut voir cela comme la différence entre un filtre passe-bas d'envergure A et un filtre passe-bas d'envergure B avec $A > B$. La différence $A - B$ est appelé la **largeur de bande**.

C'est grâce à cela que la radio fonctionne ou que les filtres photo fonctionnent.

Les circuits RC

Parmi les exemples typiques, on peut s'intéresser au circuit RC (cf : [LELEC1370](#)). En effet, on peut modéliser les équations d'un tel circuit comme suit :

$$x(t) = RC \frac{du}{dt} + u(t) \longleftrightarrow$$

$$X(j\omega) = RCj\omega Y(j\omega) + Y(j\omega) = Y(j\omega)(j\omega RC + 1)$$

$$\frac{Y(j\omega)}{X(j\omega)} = H(j\omega) = \frac{1}{1 + j\omega RC}$$

$$\text{Si } \omega_c = \frac{1}{RC}; |H(j\omega)| = \frac{1}{\sqrt{1 + j(\omega/\omega_c)^2}}$$

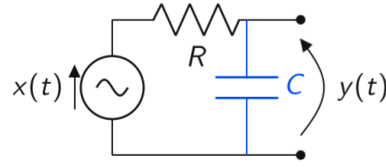


FIGURE 6.2 – RC classique

Ici, notre ω_c représente la **fréquence de coupure**.

C'est donc la valeur à partir de laquelle, on aura un changement notable de résultat.

Une chose à remarquer est que le déphasage lié à la fréquence angulaire augmente jusqu'à $\frac{\pi}{2}$. Peu perceptible au début, cela ne change rien fondamentalement.

6.1.2 Filtres non-idéaux

Un filtre parfait serait un filtre qui soit : nulle partout où on ne veut pas la fréquence et unitaire partout où on la veut.

Mais comme vu au point 2.4.3, un tel signal en phasorial correspond à :

$$H(j\omega) \longleftrightarrow h(t) = \frac{\sin(\omega_c t)}{\pi t}$$

Le gros soucis avec cela, c'est qu'on a une réponse actuelle du système qui dépend de quelque chose se passant dans le *futur* donc pas vraiment physique. Le système n'est pas *causal*. On peut aussi noter qu'on aura un déphasage nul. En réalité, on cherche à avoir un déphasage linéaire car c'est simplement une modulation et est plus facilement corrigable.

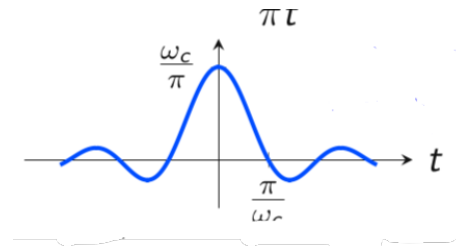


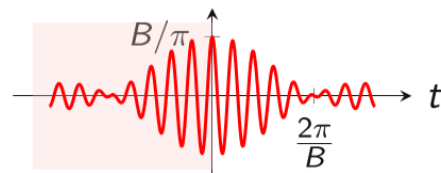
FIGURE 6.3 – Sinus cardinal

Passe-bande non-idéal

Il en va de même pour le filtre passe-bande qui est composé de deux fonctions fenêtres comme suit :

$$|H(j\omega)| = \Pi\left(\frac{\omega + \omega_0}{B/2}\right) + \Pi\left(\frac{\omega - \omega_0}{B/2}\right)$$

$$h(t) = \frac{\sin(Bt/2)}{\pi t} 2\cos(\omega_0 t)$$



Comme au point précédent, c'est la non causalité et donc le manque de sens physique qui rend ce type de filtres idéaux impossibles.

6.1.3 Filtre analogique

Ce sont des filtres composés d'éléments **passifs** et/ou **actifs**.

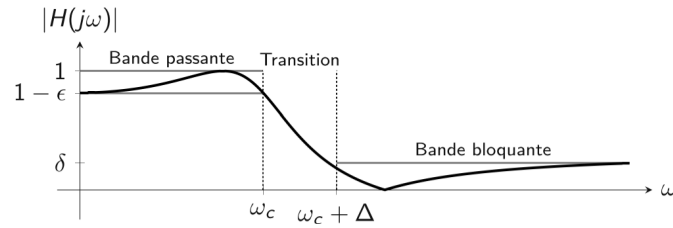


FIGURE 6.4 – Réponse impulsionnelle en fonction de la fréquence

Un filtre *réel* a un $|H(j\omega)|$ qui ressemblent plus à la figure 6.4. Le **gabarit** précise les *tolérances* qu'on accepte quand on design un filtre.

1. **Fluctuations bande passante** : on détermine un ϵ qui définit la largeur de bande passante.
2. **Zone de transition** : correspond au Δ qui indique la largeur de cette zone.
3. **Fluctuations bande bloquante** : la largeur de cette bande est définit via δ .

Les filtres passe-bas

Nom	Réponse impulsionnelle	Avantages & Inconvénients
Butterworth	$ H(j\omega) ^2 = \frac{1}{1+(\omega/\omega_c)^{2n}}$	On a un filtre avec une <i>fluctuation de bande passante</i> minime. Une <i>zone de transition</i> plus ou moins élevé et une <i>bande bloquante</i> qui tend vers 0.
Tchebychev Direct	$ H(j\omega) ^2 = \frac{1}{1+(\gamma T_n^2(\omega/\omega_c))}$	γ est le facteur d'ondulation et T_n^2 est une fonction oscillante à amplitude <i>constante</i> (polynôme de Tchebychev de première espèce d'ordre n). On a un filtre avec une <i>fluctuation de bande passante</i> oscillante mais borné. Une <i>zone de transition</i> rapide et une <i>bande bloquante</i> qui tend vers 0 et plus vite que Butterworth .
Tchebychev Inverse	$ H(j\omega) ^2 = \frac{\sqrt{\gamma} T_n(\omega_c/\omega)}{1+(\gamma T_n^2(\omega/\omega_c))}$	γ est le facteur d'ondulation et T_n^2 est une fonction oscillante à amplitude <i>constante</i> (polynôme de Tchebychev de première espèce d'ordre n). On a un filtre avec une <i>fluctuation de bande passante</i> plate. Une <i>zone de transition</i> rapide et une <i>bande bloquante</i> oscille de manière bornée (l'inverse de Tchebychev).
Elliptiques/Cauer	$ H(j\omega) ^2 = \frac{1}{1+\gamma R_n^2(\omega/\omega_n \zeta)}$	γ est le facteur d'ondulation , ζ est le facteur de sélectivité et R_n^2 est une fonction rationnelle elliptique . On a donc un filtre où en modifiant les paramètres γ et ζ , on peut obtenir un filtre spécifique.

Les filtres passe-haut

Nom	Réponse impulsionnelle	Avantages & Inconvénients
Butterworth	$ H(j\omega) ^2 = \frac{1}{1+(\omega/\omega_c)^{2n}}$	Est très similaire à son homologue passe-bas. Donc on a des bandes très plate et une zone de transition assez rapide dépendant de son degré.

Les filtres passe-bande

Pour obtenir un filtre passe-bande, on combine 2 filtres passe-bas. On doit réaliser le changement de variable suivant :

$$j\omega \rightarrow \frac{\omega_0^2 - \omega^2}{Bj\omega}$$

Où B est la largeur du filtre souhaitée et ω_0 est la fréquence à laquelle on va centrer nos deux bandes (une *positive* et *négative*).

Les filtres à encoche / notch

Ce sont des filtres qui ne font passer qu'une seule fréquence *spécifique*.

$$H(j\omega) = \frac{(j\omega + j\omega_0)(j\omega - j\omega_0)}{(j\omega + \omega_0(\cos(\theta) + j\sin(\theta)))(j\omega + \omega_0(\cos(\theta) - j\sin(\theta)))} \quad (6.1)$$

Les filtres passe-tout

Il existe également une famille de filtre qui fait *tout passer*. Son utilité? *déphaser* de $+/ - \pi$ un signal ce qui peut être fort utile!

$$H(j\omega) = \frac{(j\omega - 2)}{(j\omega + 2)} \quad (6.2)$$

6.2 Caractérisation d'un filtre

Pour se faire, on prend son équation de réponse impulsionnelle et on trace son diagramme de *Bode*. (ne pas oublier que Bode c'est *la norme et l'argument*)

Pour se faire, on utilise une échelle *logarithmique*. On doit également *factoriser* notre réponse impulsionnelle comme suit :

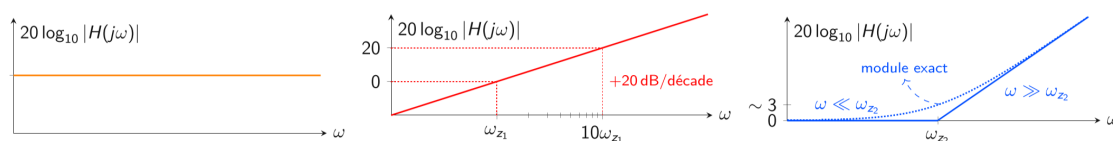
$$|H(j\omega)| = K \frac{|1 + \frac{j\omega}{\omega_{z1}}| |\frac{j\omega}{\omega_{z2}}| |(\frac{j\omega}{\omega_{z3}})^2 + 2\xi \frac{j\omega}{\omega_{z3}} + 1|}{|1 + \frac{j\omega}{\omega_{p1}}| |\frac{j\omega}{\omega_{p2}}| |(\frac{j\omega}{\omega_{p3}})^2 + 2\xi \frac{j\omega}{\omega_{p3}} + 1|} \quad (6.3)$$

$$20\log(|H(j\omega)|) \quad (6.4)$$

Grâce aux propriétés des logarithmes, on peut séparer et additionner les multiplications et divisions.

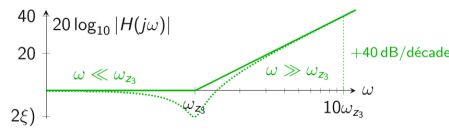
Ainsi, on trace morceau par morceau notre fonction et on va sommer ses différentes parties.

$$20\log|K| + 20\log\left|\frac{j\omega}{\omega_{z1}}\right| + 20\log\left|1 + \frac{j\omega}{\omega_{z2}}\right|$$



et pour la version au *carré* on a ceci. Il faut faire attention aux **asymptotes** qui peuvent se former !

$$20 \log \left(\left| \left(\frac{j\omega}{\omega_{z_3}} \right)^2 + 2\xi \frac{j\omega}{\omega_{z_3}} + 1 \right| \right)$$



Après cela, on additionne/soustrait simplement nos courbes ensembles et on obtient notre *première partie* du diagramme de [Bode](#).

6.2.1 Argument

Pour l'argument, le processus est le même si ce n'est qu'on ne met pas l'argument dans un logarithme mais est représenté sur une échelle logarithmique. On trace nos différents arguments et on les somme.