



Projet Signal (Matlab)

Détection de transitoires sur des signaux d'alimentation électrique d'avion

1 Introduction

Sur un avion, l'usure des gaines d'isolation des câbles d'alimentation électriques (figure 1) peut engendrer un phénomène de conduction entre phases. Cela provoque l'apparition d'arcs qui se propagent le long des câbles (phénomène d'"arc tracking"). Ce phénomène est précédé de faibles perturbations des signaux électriques (transitoires, cf figure 2) qui doivent être détectés le plus tôt possible afin de changer les circuits avant que les équipements ne soient endommagés.

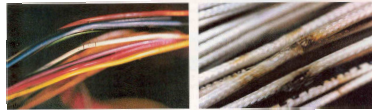


FIGURE 1 – Usure des gaines d'isolation

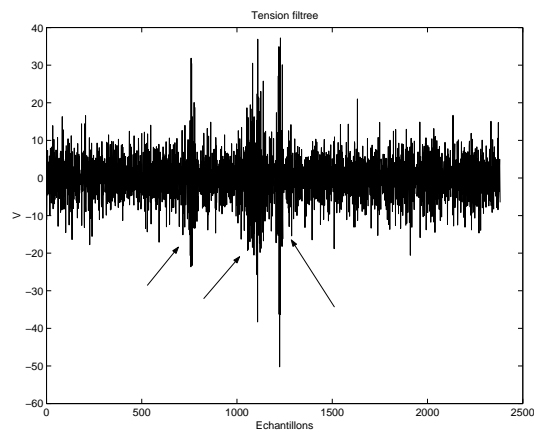


FIGURE 2 – transitoire visible sur un signal de tension filtré

Ce projet concerne la détection des transitoires sur des signaux de tension. Le réseau électrique de l'avion sera assimilé au réseau d'alimentation terrestre à 50 Hz. Le phénomène qui nous intéresse affecte les signaux à des fréquences supérieures à 500 Hz. C'est pourquoi on applique aux signaux un filtrage passe-haut afin de mettre en évidence les transitoires, qui sont très peu énergétiques par rapport au signal à 50 Hz. On procède alors à la détection de transitoires sur les signaux filtrés : pour chaque période de $\frac{1}{50} = 20$ ms, le détecteur donne une décision binaire : 1 si un transitoire est détecté, 0 sinon. La chaîne de traitement est illustrée en figure 3.

On commencera par générer un signal synthétique à partir d'hypothèses sur les signaux réels. On travaillera par la suite sur ce signal, ainsi que sur les signaux de tension réels fournis dans les fichiers

transitoire.mat (avec transitoires) et *sig_ref.mat* (sans transitoire). L'analyse spectrale et temporelle des signaux nous permettra de nous orienter vers des méthodes de détection appropriées. On étudiera ici deux détecteurs, l'un temporel, et l'autre fréquentiel.

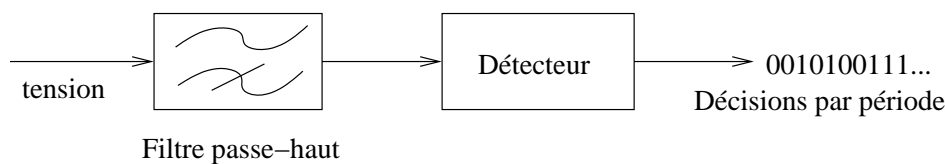


FIGURE 3 – chaîne de traitement

2 Modélisation d'un signal synthétique avec transitoire

Le réseau électrique de l'avion est alternatif, à la fréquence $f_0 = 50$ Hz. La tension varie entre -220 V et $+220$ V. La fréquence d'échantillonnage de la sonde est de $f_e = 2500$ éch/s.

1. Générer un signal sinusoïdal d'amplitude 220 (V) à la fréquence f_0 échantillonné à la fréquence f_e sur 50 périodes.
2. Ajouter à cette sinusoïde pure un bruit blanc gaussien de moyenne nulle et de variance $v_0 = 1$.
3. On modélise un transitoire par un saut de variance : ajouter à la 20^{ème} période du signal un bruit gaussien de variance $10v_0$.

3 Analyse spectrale

Déterminer le spectre du signal synthétique et du signal réel donné dans *transitoire.mat*, à l'aide des méthodes suivantes (voir annexe 1) :

1. Périodogramme
Calculer le périodogramme du signal en utilisant les fenêtres rectangulaire et de Hamming.
2. Corrélogramme
Déterminer les estimations biaisée et non biaisée de la fonction d'autocorrélation du signal (fonction *xcorr.m*). En déduire une estimation par corrélogramme de la densité spectrale de puissance du signal et comparer les résultats avec ceux de 1.

NB : on utilisera la fonction *load.m* pour récupérer les données du fichier.

4 Filtrage des signaux

La fréquence fondamentale (raie en f_0 dans le spectre) et ses éventuelles harmoniques masquent les transitoires qui sont beaucoup moins énergétiques. Afin de les détecter plus facilement, on supprime ces composantes à l'aide d'un filtre passe-haut. On trouvera en annexe 2 quelques rappels sur le filtrage analogique et numérique.

1. Calculer la réponse impulsionnelle (b_0, b_1, \dots, b_{2N}) d'un filtre non récursif passe-haut idéal, de fréquence de coupure $f_c = 500$ Hz, échantillonnée à la fréquence f_e et tronquée sur $2N + 1 = 61$ coefficients. On utilisera la méthode de la fenêtre présentée en annexe.
2. Synthétiser ce filtre, et observer sa réponse à l'aide de la fonction *freqz.m*.

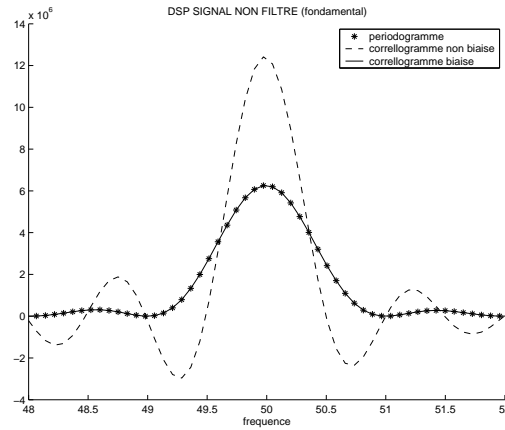


FIGURE 4 – Analyse spectrale d’un signal sinusoïdal par différentes méthodes

3. Procéder au filtrage en utilisant la fonction *filter.m*. Que peut-on dire des N premiers points du signal filtré à l’aide de cette fonction ? Afin d’accentuer l’atténuation dans la bande coupée, on filtrera le signal deux fois consécutives. On filtrera le signal synthétique, puis le signal réel (fichier *transitoire.mat*).
4. Comparer (dans les domaines temporel et fréquentiel) les signaux avant et après filtrage.

5 Un détecteur temporel : le détecteur d’énergie

On fait les hypothèses suivantes sur la nature du signal filtré en présence/absence de transitoire :

$$\begin{aligned} x_n &\sim N(0, \sigma_0^2) \text{ sous hypothèse } H_0 \text{ (pas de transitoire)} \\ x_n &\sim N(0, \sigma_1^2) \text{ sous hypothèse } H_1 \text{ (transitoire)} \end{aligned} \quad (1)$$

1. On dispose d’un signal de référence donné dans *sig_ref.mat*, dont on sait qu’il ne comporte aucun transitoire (hypothèse H_0). Observer la distribution statistique de ce signal, après filtrage passe-haut. On pourra utiliser la fonction *histfit.m* qui trace l’histogramme d’un vecteur de points ainsi que la gaussienne approchant au mieux leur distribution. Cela est-il cohérent avec (1) ? Estimer la variance σ_0^2 du signal de référence filtré (fonction *var.m*).
2. Le détecteur d’énergie est le détecteur de Neyman Pearson sous les hypothèses gaussiennes exprimées ci-dessus. Calculer le rapport de vraisemblance de la v.a. (x_n) et montrer que le détecteur de Neyman Pearson peut s’exprimer ainsi :

$$H_0 \text{ rejetée si } T(x) = \sum_n x_n^2 > \gamma$$

où le seuil de détection γ dépend de la probabilité de fausse alarme P_{FA} souhaitée et de la valeur de σ_0 . On remarquera que σ_1 n’intervient pas dans le calcul du seuil. Cependant, la probabilité de détection des transitoires sera d’autant plus grande que σ_1 est élevé.

3. Quelle est la loi de la statistique de test $T(X)$ sous l’hypothèse H_0 ?
On rappelle que si (x_1, x_2, \dots, x_L) sont L variables aléatoires gaussiennes indépendantes, de moyenne nulle et de variance σ^2 , alors $\frac{1}{\sigma^2} \sum_{n=1}^L x_n^2$ est distribuée selon une loi du χ^2 à L degrés de liberté.
4. Calculer le seuil de détection γ pour une probabilité de fausse alarme fixée $P_{FA} = 10^{-15}$ en utilisant la valeur de σ_0^2 connue a priori (signal synthétique) ou estimée sur le signal de référence.
Remarque : pour inverser la loi du χ^2 on pourra utiliser la fonction *chi2inv.m* (plus d’info : `help chi2inv`)

5. Appliquer l'algorithme de détection sur chaque période du signal modélisé et du signal réel.
6. Sur le signal *transitoire.mat*, procéder à l'analyse spectrale du signal filtré, sur des périodes où l'algorithme détecte. Que peut-t-on en déduire sur les transitoires ? le détecteur d'énergie utilise-t-il les propriétés fréquentielles des transitoires ?
7. Sous-échantillonner d'un facteur 2 une période du signal (filtré) où un transitoire est détecté. Observer le spectre du signal sous-échantillonné. Que remarque-t-on ? Expliquer.

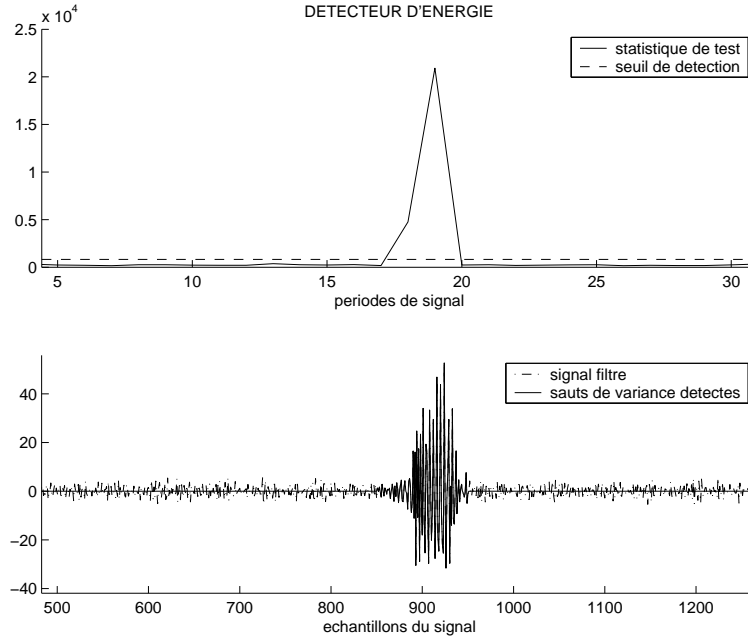


FIGURE 5 – Detection d'un transitoire

6 Un détecteur fréquentiel : le détecteur de Nuttall

Le détecteur de Nuttall utilise les propriétés spectrales du transitoire à détecter. Le principe de ce détecteur est présenté dans [1], partie I. On étudiera ici le détecteur original non normalisé.

1. Montrer que lorsque $\nu = 1$, le détecteur de Nuttall est exactement équivalent au détecteur d'énergie.
2. Montrer que la loi de la statistique de test de Nuttall sous hypothèse H_0 peut être approximée par une loi normale. Les paramètres σ_0 et μ_0 de cette loi seront estimés à l'aide du signal de référence (*mean.m* pour estimer la moyenne d'un signal, *var.m* pour la variance).
3. Montrer qu'on peut exprimer la relation entre μ_0 , σ_0 , la probabilité de fausse alarme P_{FA} et le seuil de détection de Nuttall γ_{nut} par

$$P_{FA} = Q_{N(\mu_0, \sigma_0)}(\gamma)$$

où

$$Q_{N(\mu_0, \sigma_0)}(u) = \int_u^{+\infty} \frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{(t-\mu_0)^2}{2\sigma_0^2}} dt$$

4. Implanter le détecteur de Nuttall avec $\nu = 2$ et pour une probabilité de fausse-alarme $P_{FA} = 10^{-15}$.

Pour le calcul du seuil de détection, on pourra utiliser la fonction *norminv.m* qui inverse la

fonction de répartition $f_{\mu,\sigma}$ (ou cdf : cumulative density function) de la loi normale. On rappelle

que si $x \sim N(\mu, \sigma)$, alors $f_{\mu,\sigma}(a) = P(x < a) = \int_{-\infty}^a \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx$

5. Comparer les résultats de détection à ceux obtenus avec le détecteur d'énergie.

Référence :

- [1] Z.Whang and P.Willett, "All-Purpose and Plug-In Power-Law Detectors for Transient Signals", *IEEE Transactions on Signal Processing*, vol. 49, n^o11, nov. 2001.

ANNEXE 1 - Périodogramme / Corrélogramme

La densité spectrale de puissance (DSP) d'un signal $x(t)$ à énergie finie est définie par

$$s(f) = TF[K_x(\tau)] = |X(f)|^2$$

où $X(f)$ est la transformée de Fourier de $x(t)$ et $K_x(\tau)$ sa fonction d'autocorrélation. Il en découle deux méthodes d'estimation de la DSP appelées **périodogramme** et **corrélogramme**.

Périodogramme : Lorsqu'on estime la transformée de Fourier avec l'algorithme de FFT rapide de Matlab, on montre qu'un estimateur satisfaisant de la DSP du signal $x(t)$ appelé périodogramme est défini par

$$\frac{1}{N} |TFD[x(n)]|^2$$

où $x(n)$ est obtenu par échantillonnage de $x(t)$.

Corrélogramme : L'estimation de la DSP par corrélogramme comporte deux étapes :

1. Estimation de la fonction d'autocorrélation (`xcorr.m`) qui produit $\hat{K}_x(n)$
2. Transformée de Fourier discrète de $(\hat{K}_x(n) f(n))$, où $f(n)$ est une fenêtre de pondération et $\hat{K}_x(n)$ est l'estimation biaisée ou non biaisée de la fonction d'autocorrélation.

– *Remarque 1* : il est important de noter que lorsque $\hat{K}_x(n)$ est l'estimateur biaisé de la fonction d'autocorrélation de $x(t)$, le corrélogramme coïncide exactement avec le périodogramme.

– *Remarque 2* : *Estimateurs spectraux moyennés*

On montre que la variance des estimateurs de la DSP (corrélogramme et périodogramme) ne dépend pas de la durée du signal. Ces estimateurs ne sont donc pas convergents. Afin de réduire la variance des estimateurs, il est habituel de diviser le signal observé en plusieurs tranches et à moyenner les estimateurs obtenus sur chaque tranche. La variance des estimateurs moyennés est alors inversement proportionnelle au nombre de tranches, ce qui réduit la variance.

– *Remarque 3* : *Implantation numérique*

Si le signal numérique $x(n)$ possède N_s points, la fonction `xcorr` calcule la fonction d'autocorrélation $\hat{K}_x(n)$ pour $n = -(N_s - 1), \dots, -1, 0, 1, \dots, (N_s - 1)$ (on a donc $2N_s - 1$ points). On peut "padder" cette autocorrélation par des zéros afin d'avoir une représentation plus précise de la DSP. L'algorithme de transformée de Fourier discrète de Matlab nécessite une symétrisation de la fonction d'autocorrélation de la façon suivante :

– Points d'autocorrélation $\hat{K}_x(0), \hat{K}_x(1), \dots, \hat{K}_x(N_s - 1)$

– N_z zéros

– zéro central

– N_z zéros

– Points d'autocorrélation renversés $\hat{K}_x(N_s - 1), \hat{K}_x(N_s - 2), \dots, \hat{K}_x(1)$

Cette procédure de symétrisation est illustrée sur la figure 6 pour $N_s = 4$ et $N_z = 4$

ANNEXE 2 - Du filtrage analogique au filtrage numérique

Filtrage Analogique :

Les opérations de filtrage consistent à "éliminer" ou "mettre en évidence" certaine(s) partie(s) de la bande de fréquence occupée par un signal donné. Dans le domaine continu, la transformée de Fourier d'un signal $x(t)$ est définie par $X(f) = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi ft} dt$. Filtrer un signal consiste à sélectionner une bande de fréquence d'intérêt et la mettre en évidence par rapport au reste des composantes fréquentielles du signal. Dans le domaine fréquentiel (Fourier) l'opération "naturelle" de filtrage correspond donc à une multiplication du type : $X_{\text{filtre}}(f) = X(f) H(f)$, où $X(f) = TF\{x(t)\}$ est

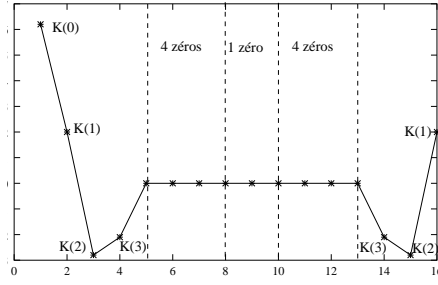


FIGURE 6 – Symétrisation de la fonction d'autocorrélation

la transformée de Fourier du signal temporel et $H(f)$ est la **fonction de transfert** du filtre qui correspond au gabarit fréquentiel pour l'opération de filtrage désirée.

L'opération de multiplication dans le domaine fréquentiel trouve son équivalent dans la convolution dans le domaine temporel (et inversement). Ainsi une opération de filtrage temporel analogique est : $x_{\text{filtre}}(t) = x(t) * h(t)$, où $h(t)$ désigne la réponse impulsionnelle du filtre et correspond à : $h(t) = TF^{-1}\{H(f)\}$ ou encore $H(f) = TF\{h(t)\}$.

Filtrage Numérique :

Si les opérations de filtrage analogique présentées précédemment sont physiquement réalisées par des circuits électroniques (du Hardware), mettant en jeu des circuits (R, L, C), le traitement numérique mis en oeuvre par l'intermédiaire de μP et de programmation informatique connaît depuis plusieurs années un essor grandissant, offrant des possibilités beaucoup plus vastes (en termes de complexité, taille, évolutivité, reprogrammabilité).

Le signal numérique $x(n)$, $n \in \{1, \dots, N\}$ est obtenu par échantillonnage du signal continu qui représente le signal analogique associé $x(t)$. En toute rigueur, si T_e est la période d'échantillonnage, le signal numérique obtenu en sortie du convertisseur analogique-numérique est $x(n) = x(nT_e)$. Il est donc possible comme dans le cas de signaux continus de définir les opérations de filtrage dans le domaine discret. Dans cet espace de représentation, la transformée en Z est le pendant de la transformée de Fourier pour le continu :

$$X(z) = \sum_{k=-\infty}^{+\infty} x(k) z^{-k}$$

Une des propriétés essentielle de la transformée en Z (notée TZ) est le théorème du retard :

$$X(z) z^{-m} = TZ[x(n)] z^{-m} = TZ[x(n-m)]$$

Ainsi un filtre dans le domaine discret se définit par une réponse impulsionnelle, une suite de points $h(n)$, $n \in \mathbb{N}$, dont la transformée en Z désigne la fonction de transfert du filtre :

$$H(z) = \sum_{k=-\infty}^{+\infty} h(k) z^{-k}$$

Remarque : En pratique on n'utilise que la transformée en Z dite unilatérale, $X(z) = \sum_{k=0}^{+\infty} x(k) z^{-k}$ qui suppose le signal x causal ($x(n)=0$ pour $n<0$). De la même manière, le principe de causalité ("l'effet ne peut précéder la cause") fait que la réponse impulsionnelle de tout système physique est obligatoirement nulle pour $t<0$. Donc pour tout système causal on a : $h(n)=0$ pour $n<0$. C'est pourquoi nous ne noterons plus les indices de la TZ.

L'opération de filtrage dans le domaine fréquentiel, en Z , tout comme pour le cas continu est un produit de l'entrée par la fonction de transfert du filtre :

$$X_{\text{filtre}}(z) = H(z) X(z)$$

Ainsi, en utilisant le théorème du retard on montre que l'opération de filtrage peut encore s'écrire dans le domaine temporel discret :

$$X_{\text{filtre}}(z) = H(z) X(z) = \left[\sum h(k) z^{-k} \right] X(z) = \sum h(k) X(z) z^{-k}$$

En utilisant le théorème du retard et la définition de la TZ, cette expression devient dans le domaine temporel :

$$x_{\text{filtre}}(n) = \sum_k h(k) TZ^{-1} [X(z) z^{-k}] = \sum_k h(k) x(n-k)$$

qui n'est autre que l'expression de la formule de convolution discrète. C'est l'opération de filtrage qui est concrètement implantée dans un ordinateur : la sortie à un instant donné n est une combinaison linéaire des échantillons d'entrée aux instants inférieurs à n .

Cependant on vise généralement des opérations de filtrage dites temps réel (calcul de l'échantillon de sortie pendant l'intervalle de temps entre 2 échantillons d'entrée); or la charge de calcul de cette opération générale croît avec le temps (i.e., les indices sont tels que k varie entre 0 et l'infini dans l'équation précédente), plus on a d'échantillons passés plus le calcul d'un échantillon nécessite d'opérations élémentaires. Afin de limiter cette charge de calcul, on considère une classe de filtres linéaires ayant une charge calculatoire constante au cours du temps qui est la classe des filtres récursifs qui possèdent une expression générale de la forme :

$$x_{\text{filtre}}(n) = - \sum_{k=1}^M a_k x_{\text{filtre}}(n-k) + \sum_{k=0}^N b_k x(n-k) \quad (2)$$

Rappel : c'est cette équation temporelle qui est concrètement implantée dans le ordinateur. Dans ce cas la fonction de transfert associée est une fraction rationnelle en Z :

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 + \sum_{k=1}^M a_k z^{-k}}$$

Pour définir un tel filtre numérique, il suffit donc de déterminer les coefficients (a_k) et (b_k) intervenant dans $H(z)$ (sous Matlab, voir la commande filter).

Filtrage non récursif :

Lorsque les coefficients (a_k) sont nuls, le filtre est dit non récursif. Dans ce cas, l'opération de filtrage issue de (2) est une simple convolution discrète :

$$x_{\text{filtre}}(n) = \sum_{k=0}^N b_k x(n-k)$$

Par analogie avec l'opération de filtrage analogique, on appelle alors la suite (b_k) , $k = 1, \dots, N$ la **réponse impulsionnelle du filtre**.

Synthèse d'un filtre non récursif par la méthode de la fenêtre :

Il existe une méthode simple de synthèse d'un filtre numérique non récursif : la méthode de la fenêtre consiste à calculer les coefficients (b_k) , $k = 1, \dots, N$ du filtre souhaité en échantillonnant et tronquant la réponse impulsionnelle d'un filtre analogique idéal.

Par exemple, pour synthétiser un filtre *passé-bas* sur $2N+1$ coefficients par cette méthode, on procède de la manière suivante :

1. Expression du filtre analogique idéal :

$$\begin{aligned} H(f) &= 1 \text{ pour } f \in [-f_c, f_c] \\ &= 0 \text{ ailleurs} \end{aligned}$$

2. Calcul de la réponse impulsionnelle du filtre analogique par transformée de Fourier inverse :

$$h(t) = TF^{-1} \{H(f)\} = \frac{\sin(2\pi f_c t)}{\pi t}$$

3. Échantillonnage et troncature de $h(t)$: $b_0 = h\left(-\frac{N}{f_e}\right)$, ..., $b_N = h(0)$, ..., $b_{2N} = h\left(\frac{N}{f_e}\right)$