TP 2 de communications numériques

M. Ellouze et R. Tajan

1 Objectifs et évaluation

Le but de ce TP est de mettre en place, à l'aide du logiciel un filtrage de mise en forme et un filtrage adapté de signaux ainsi qu'à tracer les périodogrammes de Welch les signaux filtrés. Dans cette séance, vous écrirez un script interactif de Matlab (notebook) ainsi que des fonctions Matlab permettant

1. de mettre en œuvre un filtrage de mise en forme pour lequel l'utilisateur pourra choisir
   — la nature du filtre utilisé,
   — Les paramètres du filtre,
2. d'afficher le périodogramme de Welch pour les signaux filtrés
3. de mettre en œuvre un filtrage adapté.

Vos codes et votre script interactif sont à rendre en fin de séance sur l'interface Thor `https://thor.enseirb-matmeca.fr/ruby/`.
Pour ce deuxième TP, les fonctions du premier TP vous sont fournies en .p . Vous pouvez faire appel à ces fonctions si jamais vous n'avez pas réussi à développer toutes les fonctions du premier TP.

2 Filtrage de mise en forme

L'objectif de cette partie est de réaliser le filtrage de mise en forme d'un signal modulé. Dans toute la suite, on notera par :
— $T_s = 1\mu s$ le temps symbole
— $f_e = \frac{1}{T_e} = 10MHz$ la fréquence d'échantillonnage
— $F_{se} = \frac{T_s}{T_e}$ le facteur de sur-échantillonage permettant d'adapter le rythme du signal présenté en entrée du filtre de mise en forme
— $g(t)$ le filtre de mise en forme

Écrire dans la fonction `shaping` qui aura le prototype suivant :

Listing 1 – Fonction **shaping.m**.

```
function sl = shaping(s, Fse, g)
% s : (vecteur complexe) vecteur de symboles
% Fse : (int) Facteur de sur-échantillonnage (nombre d'échantillons par symboles)
% g : (vecteur réel) vecteur de réponse impulsionnelle de g(t)
```

Cette fonction construit le **signal complexe en bande de base** (nommé $s_l(t)$ dans le cours) une à partir des symboles modulés contenus dans le vecteur `s`. L'expression mathématique de $s_l(t)$ à temps continu est la suivante :

$$s_l(t) = \sum_{m\in\mathbf{Z}} S_m g(t - mT_s) \tag{1}$$

On souhaite synthétiser une version échantillonnée à la fréquence F_e de $s_l(t)$. F_e sera choisie de telle sorte que $\frac{T_s}{T_e} = F_{se}$, F_{se} étant une valeur entière donnée par l'utilisateur. Le filtre de mise en forme $g(t)$ devra donc lui aussi être échantillonné à la fréquence F_e. Le vecteur retourné est nommé `sl` correspondra aux échantillons du signal $s_l(t)$ de l'équation (1) échantillonné à la fréquence F_e.

Dans votre notebook : Faites appel à la fonction `shaping` pour générer un signal correspondant à l'envoi de 10 symboles BPSK ($S_n \in \{\pm 1\}$) Vous considèrerez le filtre de mise en forme $g(t)$ défini par :

$$g(t) = \begin{cases} 1 & \text{si} \quad 0 \leq t < T_s \\ 0 & \text{sinon} \end{cases} \tag{2}$$

Afficher le signal $s_l(t)$ et annoter votre graphique pour montrer l'effet du "retard de groupe" pour chaque symbole.

Même question pour un filtre en racine de cosinus sur-élevé de roll-off $\alpha = 0.5$ et de temps de propagation de groupe $T_g = 4T_s$. Dans ses fonctions, Matlab considère le "span" du filtre qui est défini comme $2T_g$.

Commenter les deux courbes en les comparant l'une à l'autre.

3 Densité spectrale de puissance

On s'intéresse dans cette partie à l'étude de la densité spectrale de puissance du signal $s_l(t)$. Pour cette étude, on supposera que les symboles sont indépendants et uniformément distribués dans une **constellation** $\{s_0, \ldots, s_{M-1}\}$ centrée ($m_a = \frac{1}{M}\sum_{m=0}^{M-1} s_m = 0$). Sous cette hypothèse, la DSP théorique d'un signal de communications numériques peut être calculée à l'aide de la formule suivante (cas particulier de la formule de Bennett)

$$\Gamma_{s_l}(f) = \frac{\sigma_a^2}{T_s}\left|G(f)\right|^2 \tag{3}$$

où σ_a^2 est la variance des symboles

$$\sigma_a^2 = \frac{1}{M}\sum_{m=0}^{M-1} |s_m|^2 \tag{4}$$

T_s est le temps symbole et $G(f)$ la réponse en fréquence du filtre de mise en forme. Le calcul des DSP expérimentales $\hat{\Gamma}_{sl}(f)$ se fera ici en utilisant la méthode du périodogramme de Welch avec les paramètres suivants

1. on utilisera une fenêtre rectangulaire de taille $N_{fft} = 512$ échantillons (`ones(1,Nfft)`),
2. il ne doit y avoir aucun échantillon en commun entre deux fenêtres successives.

Vous pouvez utiliser la méthode `pwelch` de Matlab à condition de bien lire sa documentation.

Dans votre notebook : Comparer la DSP obtenue avec la DSP théorique $\Gamma_{sl}(f)$ pour les deux filtres définis dans la section 2. Afin de tracer ces DSP, vous génèrerez des signaux QPSK pour $N_s = 5000$ symboles.

Vous commenterez ces courbes en dressant un bilan comparatif. Ce bilan inclura la comparaison des points suivants :

- **bande passante** (obtenue en mesurant la largeur du lobe principal)
- **efficacité spectrale** : obtenue par la formule suivante $\eta = \frac{D_b}{B}$ où D_b est le débit binaire et B la bande passante.

4 Récepteur

Écrire la fonction `recepteur_optimal` qui implémentera un récepteur optimal. Un récepteur est dit optimal s'il vérifie les conditions suivantes :

1. son filtre est adapté au filtre d'émission (et filtre du canal s'il y en a un)
2. la cascade filtre de mise en forme - filtre adapté vérifie le critère de Nyquist.

La fonction `recepteur_optimal` aura le prototype suivant :

Listing 2 – Fonction **recepteur_optimal**.

```
function r_n = recepteur_optimal(y_l, Fse, g)
% Arguments :
% y_l : (vecteur complexe) signal reçu (sur-échantillonné)
% Fse : (int) facteur de sur-échantillonnage
% g : (vecteur réel) Réponse impulsionnelle du filtre de
    mise en forme
% Retour :
% r_n : (vecteur complexe) signal après filtrage adapté et
    échantillonnage à T_s
```

Cette fonction réalisera les actions suivantes sur le signal en sortie de canal y_l (ici $y_l = s_l$) :

- filtrer le signal $y_l(t)$ par le filtre adapté (ce qui donne $r_l(t)$),
- échantillonner le signal $r_l(t)$ avec un pas T_s pour récupérer les N_s symboles,

Dans votre notebook : Montrez qu'il n'y a pas d'interférences entre symboles en affichant la constellation de r_n après l'envoi de $N_s = 5000$ symboles QPSK. Commentez le diagramme de constellation obtenu.

Astuce : vous pouvez tester votre chaîne de communications numériques en vérifiant que les bits estimés en sortie du bloc "Association Symbole$\rightarrow$bits" sont bien les mêmes que les bits émis.

5 Bonus

1. Maintenant que vous avez implémenté tous les blocs, ajouter un bloc à l'émission réalisant la mise sur fréquence porteuse

$$s(t) = Re(s_l(t)e^{j2\pi f_p t}) \tag{5}$$

avec $f_p = 250kHz$. **Dans votre notebook affichez le spectre du signal transposé en fréquence et commentez-le.**

2. Implémentez la partie réception, **dans votre notebook illustrez son bon fonctionnement.**

6 Contacts

— Malek Ellouze - malek.ellouze@ims-bordeaux.fr
— Romain Tajan - romain.tajan@ims-bordeaux.fr