

TP de communications numériques

module TS218 – année 2018/2019

Romain Tajan (romain.tajan@ims-bordeaux.fr)

1 Objectifs et évaluation

L'objectif des TP de communications numériques est de simuler à l'aide du logiciel Matlab une chaîne de transmission en bande de base dont la structure repose sur le standard DVB-S (Digital Video Broadcasting - Satellite). L'intérêt de cette chaîne est double :

- elle pourra être interfacée avec des radios logicielles
- elle pourra permettre de recevoir des signaux réellement transmis.

Pour cela, cette chaîne devra être robuste aux erreurs mais aussi aux problèmes de synchronisation.

Les TP se font en **binôme ou monôme** et l'évaluation porte sur une note de rapport et une note de travail continu (en séances).

Concernant le rapport, il ne doit pas excéder **15 pages**, vous devez fournir un document scientifique et technique, qui doit présenter votre travail, vos choix techniques et dans lequel tous les résultats obtenus doivent être interprétés et commentés. Les codes Matlab doivent également être transmis à votre enseignant. Ils doivent pouvoir être compris rapidement. Cela passe par l'utilisation de commentaires. Les commentaires doivent permettre de répondre au moins à la question : que fait la ligne de code ? Une attention particulière doit être portée à la lisibilité du programme. Les rapports doivent être au format NOM1_NOM2.pdf et les codes au format NOM1_NOM2.zip. Ces deux fichiers doivent être transmis à votre encadrant par Email au **plus tard deux semaines après la dernière séance de TP, sachant que la date exacte vous sera fixée par votre encadrant.**

Remarque : chaque jour de retard dans la remise des livrables sera sanctionné de 2 points en moins sur la note finale du TP.

2 Communications numériques en bande de base

Dans cette première partie vous allez vous intéresser au cas des communications numériques en bande de base émettant des symboles QPSK L'architecture bande de base à considérer est présentée sur la Figure 1.

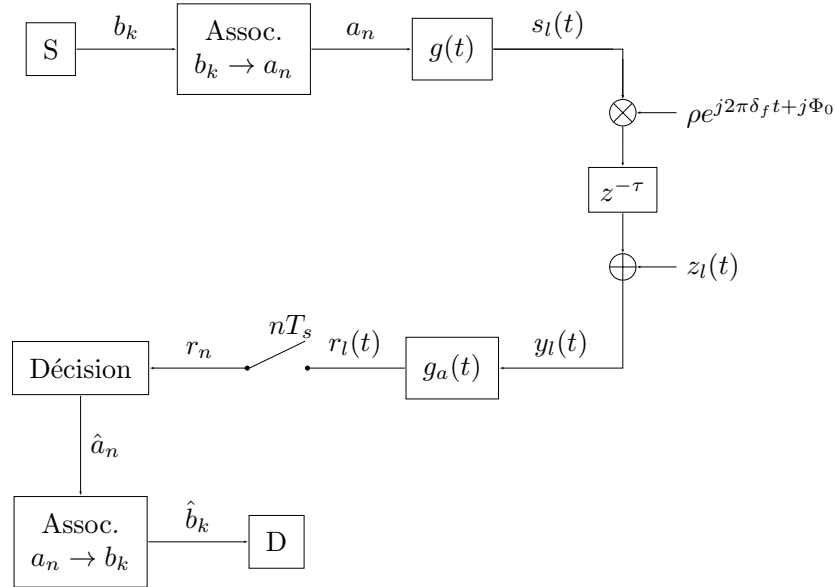


FIGURE 1 – Architecture bande de base sans synchronisation

À travers la simulation Matlab, vous vérifierez entre autre une partie de vos résultats théoriques du cours et évalueriez les performances de la chaîne de communications numériques en terme de probabilité d'erreur binaire.

Paramètres de simulation :

- Fréquence d'échantillonnage : $f_e = \frac{1}{T_e} = 1\text{MHz}$,
- Le débit symbole $D_s = 250\text{kSymboles/s}$,
- Le filtre de mise en forme est un filtre en racine de cosinus sur-élevé de roll-off $\alpha = 0.35$ et de temps de propagation de groupe $T_g = 8T_s$.
- La séquence de bits est issue de la lecture d'un fichier de transport MPEG-2, le scrambleur est fourni, vous trouverez un travail initial à l'adresse <https://github.com/rtajan/TS218>
- $F_{se} = \frac{T_s}{T_e}$ représente le facteur de sur-échantillonnage, c'est-à-dire le nombre d'échantillons simulés par symbole envoyé,
- La modulation est une modulation QPSK vérifiant le mapping de Gray.

3 Simulations avec défauts

La chaîne de réception finale, incluant synchronisations temporelle et fréquentielle est donnée en figure 2.

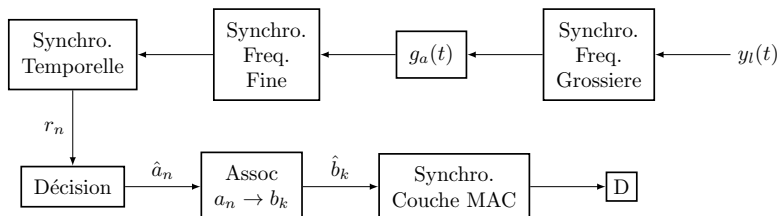


FIGURE 2 – Chaîne de réception incluant les blocs de synchronisations.

3.1 Évolution de l'impact d'une désynchronisation temporelle

Le but de cette partie est d'étudier l'impact d'une désynchronisation temporelle sur le signal reçu. Tout paramètre constant par ailleurs, nous allons donc faire varier τ et observer les conséquences de cette variation.

3.1.1 Simulation sans synchronisation

Tâches à effectuer :

1. Donner le modèle du signal reçu après échantillonnage au temps symbole (modèle de r_n faisant apparaître les symboles transmis a_n , le retard τ , le filtre global $v(t)$ et le bruit en sortie de filtre adapté w_l).
2. Tracer les constellations de a_n et de r_n si $\frac{E_b}{N_0} = 100dB$, pour $\tau = 0.1T_s$, $\tau = 0.3T_s$ et $\tau = 0.55T_s$, commentez votre résultat.
3. Pour les valeurs de τ suivantes : $\tau = 0.1T_s$, $\tau = 0.33T_s$, $\tau = 0.35T_s$ et $\tau = 0.4T_s$ tracer l'évolution du TEB en fonction de rapport $\frac{E_b}{N_0}$ en dB, lorsque ce dernier varie de 0dB à 30dB. Vous afficherez la probabilité d'erreur binaire théorique dans le cas QPSK. N'affichez que des valeurs de TEB/ P_b supérieure à 10^{-6} . Expliquer votre résultat.
4. Pour chacune des valeurs de τ précédente, relever la perte de sensibilité pour $P_b = 10^{-3}$, la tracer en fonction de τ . Conclure cette partie.

3.1.2 Mise en place de la synchronisation temporelle

Avant de mettre en place cette synchronisation, il convient de modifier votre filtre adapté de sorte qu'il renvoie 4 échantillons par symboles. Dans cette partie, la sortie du filtre adapté est appelée $r_l(t)$ ses échantillons seront contenus dans le vecteur matlab `r1` échantillonné à la fréquence F_e . Pour réaliser la synchronisation temporelle, vous utiliserez l'algorithme suivant :

1. Initialiser $\tau = 0$ pour tout nouveau point de SNR,
Au moment de la synchronisation.
2. Créer le vecteur `r1e = [zeros(Fse,1); r1; zeros(Fse,1)]`
3. Créer le vecteur contenant les instants d'échantillonnage suivant :
`te = Fse + (1:Fse:(1+(Ns-1)*Fse)) + int_tau` où `int_tau = floor(tau)`

4. On utilisera une interpolation linéaire pour calculer $r_l(t)$ aux instants d'échantillonnage non-entiers :

$$r_n = rle(te)*(1-frac_tau) + rle(te + 1)*frac_tau$$
 où $frac_tau = \tau - \lfloor \tau \rfloor$
5. Décider les symboles contenus dans r_n :

$$r_nd = 0.707*(sign(real(r_n)) + 1i * sign(imag(r_n)))$$
6. Calculer l'erreur après décision :

$$err = r_n - r_nd$$
7. Calculer la dérivée de $r_l(t)$ aux instants d'échantillonnage non-entiers :

$$drl = 0.5*(1 - frac_tau)*(rle(te+1)-rle(te-1)) + 0.5*frac_tau*(rle(te+2)-rle(te))$$
8. mettre à jour τ :

$$tau = tau - mu * real(err'*drl/Ns);$$

Pour la suite du TP on utilisera $\mu = 10$.

Tâches à effectuer :

1. Pour $\tau = 0.55T_s$, tracer les constellations des symboles a_n et r_n pour $\frac{E_b}{N_0} = 100dB$, comparer ce résultat à celui obtenu sans synchronisation.
2. Pour les valeurs de τ suivantes : $\tau = 0.1T_s$, $\tau = 0.33T_s$, $\tau = 0.35T_s$ et $\tau = 0.4T_s$ tracer l'évolution du TEB en fonction de rapport $\frac{E_b}{N_0}$ en dB, lorsque ce dernier varie de 0dB à 30dB. Vous afficherez la probabilité d'erreur binaire théorique dans le cas QPSK. N'affichez que des valeurs de TEB/ P_b supérieure à 10^{-6} .
3. Pour chacune des valeurs de τ précédente, relever la perte de sensibilité pour $P_b = 10^{-3}$, la tracer en fonction de τ . Conclure.
4. Comparer les débits de réceptions avec et sans synchronisation temporelle, vous mettrez le paramètre `ComputationDelay` du bloc `stat_erreur` égal à 6 fois le nombre de bit par trame. Cela permet d'éviter de calculer le TEB sur la phase transitoire de l'algorithme.

3.2 Étude de l'impact d'un déphasage ou d'une désynchronisation fréquentielle

Le but de cette partie est d'étudier l'impact d'une désynchronisation fréquentielle sur le signal reçu. Tout paramètre constant par ailleurs, nous allons donc faire varier Φ_0 et δ_f et observer les conséquences de cette variation.

3.3 Évolution du TEB en fonction de $\frac{E_b}{N_0}$ en présence de déphasage ou de désynchronisation fréquentielle

3.3.1 Impact d'un déphasage

Dans cette partie, $\tau = 0$ et $\delta_f = 0$ par défaut. Tâches à effectuer :

1. Tracer les constellations de a_n et de r_n si $\frac{E_b}{N_0} = 100dB$, pour $\Phi_0 = 10^\circ$, $\Phi_0 = 30^\circ$ et $\Phi_0 = 50^\circ$, commentez votre résultat.
2. Pour les mêmes valeurs de Φ_0 , tracer l'évolution du TEB en fonction de rapport $\frac{E_b}{N_0}$ en dB, lorsque ce dernier varie de 0dB à 30dB. Vous afficherez la probabilité d'erreur binaire théorique dans le cas QPSK. N'affichez que des valeurs de TEB/ P_b supérieure à 10^{-6} . Expliquer votre résultat.
3. Pour chacune des valeurs de Φ_0 précédente, relever la perte de sensibilité pour $P_b = 10^{-3}$, la tracer en fonction de τ . Conclure cette partie.

3.3.2 Impact d'une désynchronisation fréquentielle

Dans cette partie, $\tau = 0$ et $\Phi_0 = 0$ par défaut. Tâches à effectuer :

1. Pour $\frac{E_b}{N_0} = 100dB$ et $\delta_f = 100kHz$, tracer le périodogramme de Welch du signal en sortie du canal. Superposer à cette courbe la réponse en fréquence du filtre de réception ($|G_a(f)|^2$). Commenter l'intérêt d'une synchronisation grossière avant filtrage adapté.
2. Tracer les constellations de a_n et de r_n si $\frac{E_b}{N_0} = 100dB$, pour $\delta_f = 1Hz$, $\delta_f = 1kHz$ et $\delta_f = 100kHz$, commentez votre résultat.
3. Pour les valeurs de δ_f de la question précédente, tracer l'évolution du TEB en fonction de rapport $\frac{E_b}{N_0}$ en dB, lorsque ce dernier varie de 0dB à 30dB. Vous afficherez la probabilité d'erreur binaire théorique dans le cas QPSK. N'affichez que des valeurs de TEB/ P_b supérieure à 10^{-6} . Expliquer votre résultat.

3.3.3 Mise en place des blocs de synchronisation fréquentielle

On considère d'abord les paramètres suivants :

- $\rho = 1$ le canal n'amplifie ni n'atténue le signal,
- $\delta_f = 11021Hz$,
- $\Phi_0 = 60^\circ$,
- $\tau = 0$, il n'y a pas de retard entre le signal émis et le signal reçu,
- $\frac{E_b}{N_0} = 100dB$.

Tâches à effectuer :

1. La synchronisation fréquentielle grossière sera effectuée en calculant la position du maximum d'un périodogramme moyenné (périodogramme de Welch - cf cours de M. Vallet) obtenu avec la commande suivante `pwelch(y_1.^4, hanning(Nfft), 0, Nfft, Fe, 'centered')` où $Nfft = 1024$.
2. Mettre en place la synchronisation fine à l'aide d'une boucle à remodulation.
3. Tracer la courbe du TEB en fonction de $\frac{E_b}{N_0}$. Comment expliquez-vous ce phénomène? Proposer une solution.

4 Transmission à base d'USRP

5 Contacts

— Romain Tajan - romain.tajan@ims-bordeaux.fr