



Travaux Pratiques

Notion de fréquence normalisée et paramètres de simulation

Par défaut, la fréquence d'échantillonnage (sampling) F_s du logiciel MATLAB est de « un » ($F_s=1$). Ainsi, les quantités représentant des fréquences (ou des débits) devront être normalisées par rapport à la fréquence d'échantillonnage utilisée. Dans toute la suite, il ne sera fait référence qu'aux fréquences normalisées. De même, la période d'échantillonnage T_s du logiciel MATLAB est de « un » ($T_s=1$). Les quantités représentant des temps, comme les périodes, seront normalisées par rapport à la période d'échantillonnage utilisée.

| | Valeur réelle | Valeur normalisée |
|-----------------------------|---------------|-------------------|
| Fréquence d'échantillonnage | 10 MHz | $F_s=1$ |
| Période d'échantillonnage | 0,1 μ s | $T_s=1$ |
| Fréquence porteuse | 2 MHz | $f_c=0,2$ |
| Débit binaire | 1 Mbit/s | $D_b=0,1$ |
| Période binaire | 1 μ s | $T_b=10$ |

Partie 1 : Modulations linéaires (M-PAM, M-QAM, M-PSK)

Introduction

L'objectif de cette partie est de générer des signaux issus de modulations numériques (voir Figure 1) et de comparer les représentations en temps et en fréquence de plusieurs modulations. En se fondant sur ces représentations, les performances des modulations numériques seront ensuite étudiées en termes d'efficacité spectrale.

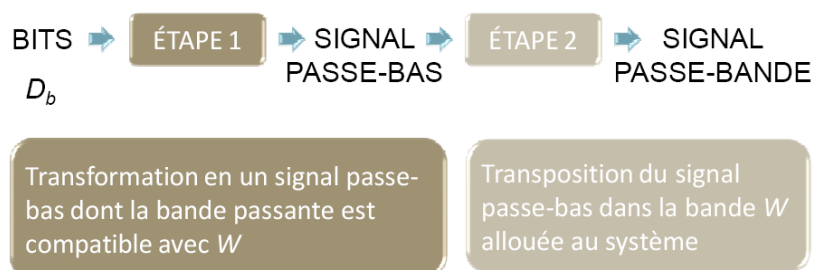


Figure 1 : génération de modulations numériques – schéma de principe

Cette partie commence par l'étude de la modulation de phase BPSK (Binary Phase Shift Keying). Puis, l'étude est généralisée aux autres modulations de phase, aux modulations d'amplitude et aux

modulations d'amplitude en quadrature, et enfin aux modulations de fréquence. La dernière section est consacrée à l'étude de l'efficacité spectrale.

I : Modulation binaire

1. Génération d'une modulation numérique de phase BPSK

(Programme MATLAB intitulé modulateur_v0.m)

La génération de la modulation se fonde sur le schéma de principe de la Figure 2.

a. Génération d'une suite aléatoire de bits α_k

Générer une suite aléatoire de N_b bits grâce à la fonction *randi()*. Pour la représentation en temps, prendre $N_b=10$ et, pour la représentation en fréquence, prendre $N_b=100$.

b. Génération de symboles BPSK $A_k \exp(j\theta_k)$

Coder la suite de bits en symboles BPSK grâce à la fonction *pskmod()*. Il s'agit d'une modulation binaire, donc le nombre de symboles, M , est égal à deux ($M=2$). Représenter les symboles avec l'instruction *scatterplot()*.

Conseil de programmation : **prévoir une partie initialisation de variables au début du code** MATLAB. L'objectif est de programmer de façon à pouvoir passer facilement aux modulations M-PSK. Par exemple, utiliser une variable M initialisée à 2 au lieu d'utiliser directement 2 dans le code.

NB : les symboles générés par la fonction *pskmod()* sont **complexes**, même pour une BPSK. Cela permet notamment de générer des symboles binaires de la forme $\exp(j\theta)$ et $\exp[j(\theta+\pi)]$.

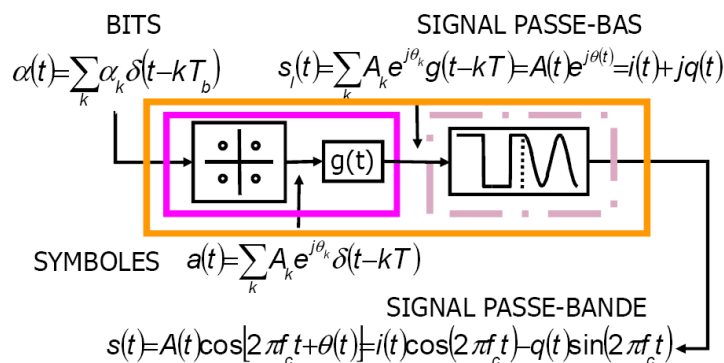


Figure 2 : génération de modulations numériques - schéma détaillé

c. Mise en forme des symboles BPSK : génération des signaux $a(t)$ et $s_l(t)$

Générer un peigne de Dirac $a(t)$ de débit D , débit symbole, dont les amplitudes correspondent aux symboles BPSK.

Rappel : $D = D_b / \log_2(M)$.

Indication : générer un vecteur nul de durée $N \cdot T$ ou N représente le nombre de symboles, avec $N = N_b / \log_2(M)$, et T désigne la période symbole ($T = 1/D$), avec l'instruction *zeros()*. Puis, placer, à intervalles de temps réguliers, les valeurs des symboles BPSK.

Filtrer le peigne de Dirac par un filtre porte d'amplitude unité et de durée T .

d. Transposition en fréquence : génération du signal $s(t)$

Transposer le signal autour de la fréquence porteuse f_c .

Remarque sur l'hypothèse de signaux à bande étroite : l'hypothèse de signaux à bande étroite (f_c très grande devant D) permet de faire un certain nombre de simplifications dans les calculs sur les signaux numériques. Ici, cette hypothèse n'est pas vérifiée. En effet, pour visualiser les signaux et leurs densités spectrales de puissance sans devoir systématiquement faire un zoom avant et pour réduire le temps de simulation, cette hypothèse a été abandonnée. Cependant, pour que les résultats établis en cours restent valables, cette hypothèse a été remplacée par une autre hypothèse, à savoir que $f_c = kD$, où k est un entier non nul.

2. Visualisation des résultats

a. Représentations en temps

Diviser une figure en trois parties et représenter les signaux suivants :

- Le signal passe-bande sur la figure du haut,
- Les parties réelles du peigne de Dirac des symboles et du signal passe-bas sur la figure centrale,
- Les parties imaginaires du peigne de Dirac des symboles et du signal passe-bas sur la figure du bas.

Indication : utiliser l'instruction `subplot()`.

Un exemple est fourni dans la Figure 3. L'axe temporel sera exprimé en multiples de la période symbole T . Pour la représentation du peigne de Dirac des symboles, seuls les échantillons non nuls seront représentés. Expliquer pourquoi le signal passe-bande n'a pas la forme exacte d'un signal sinusoïdal.

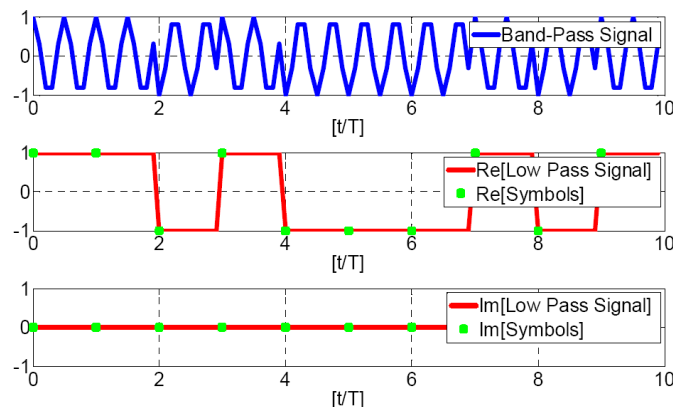


Figure 3 : représentation temporelle du signal passe-bande, des parties réelle ($\text{Re}[\text{Low Pass Signal}]$) et imaginaire ($\text{Im}[\text{Low Pass Signal}]$) du signal passe-bas, et des parties réelle ($\text{Re}[\text{Symbols}]$) et imaginaire ($\text{Im}[\text{Symbols}]$) des symboles BPSK

Consignes pour toute la suite des travaux pratiques : toujours donner une **légende explicite** aux figures. Différencier les courbes par une trace différente (en prévision d'une éventuelle impression sur une imprimante noir et blanc).

b. Représentation en fréquence

Sur une même figure, tracer les trois densités spectrales de puissance correspondant aux trois signaux suivants :

- Le peigne de Dirac des symboles,
- Le signal passe-bas à la sortie du filtre de mise en forme,
- Le signal passe-bande à la sortie du modulateur.

Les densités spectrales de puissance seront tracées entre $-F_s/2$ et $F_s/2$ en utilisant une échelle logarithme2. Un exemple est fourni Figure 4. Expliquer la forme des densités spectrales de puissance. (Le **recentrage des densités spectrales de puissance** se fait par la fonction `fftshift()`.)

Indication : la densité spectrale de puissance d'un signal donné est obtenue par le module au carré de sa transformée de Fourier (FFT pour Fast Fourier Transform), divisé par le nombre de points de la transformée de Fourier, noté $nfft$.

Rappel : le calcul de la FFT est optimisé pour un nombre de points de la FFT multiple d'une puissance de deux. Il est conseillé de choisir, comme nombre de points de la FFT, la puissance de deux immédiatement supérieure au nombre d'échantillons du signal. Pour cela, utiliser la fonction `nextpow2()`.

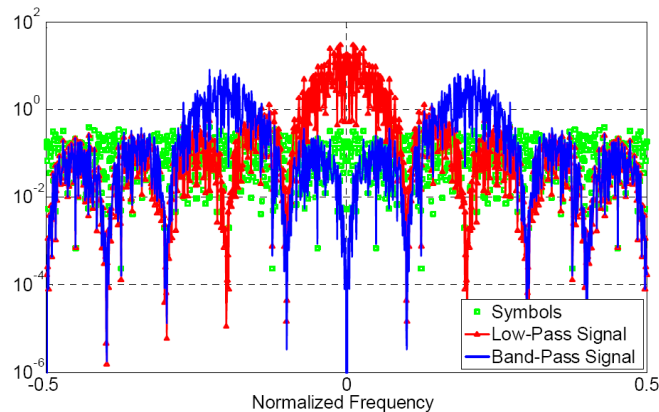


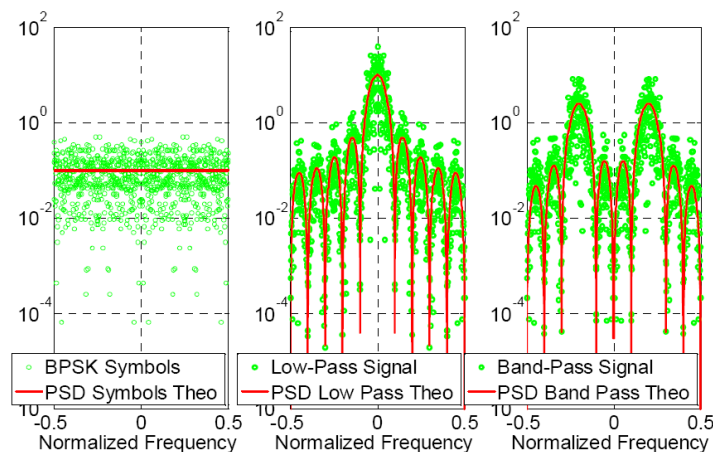
Figure 4 : représentation en fréquence des symboles (Symbols), du signal passe-bas (Low-Pass Signal) et du signal passe-bande (Band-Pass Signal) relatifs à une modulation BPSK

Compte-rendu : le programme MATLAB, intitulé **modulateur_v0.m**, les figures correspondant aux représentations en temps et en fréquence, et les explications.

3. Estimation de la densité spectrale de puissance

Moyenner 1 (Nstat=1) puis 100 (Nstat=100) estimations des densités spectrales de puissance pour les trois signaux précédents. Tracer les densités spectrales de puissance estimées et les densités spectrales de puissance théoriques. Un exemple est donné dans la figure suivante. Commenter.

Compte-rendu : le programme MATLAB, intitulé **modulateur_v1.m**, les représentations en fréquence et les commentaires. (Programme MATLAB intitulé `modulateur_v1.m`)



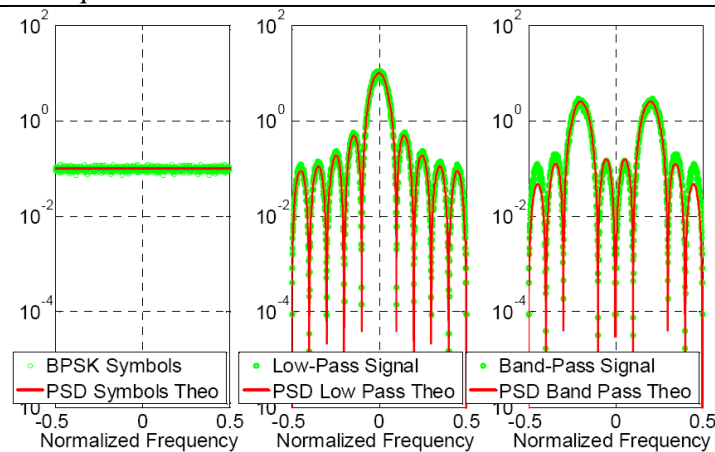


Figure 5 : densités spectrales de puissance simulées (trait continu) et théoriques (ronds) des symboles BPSK (figures de gauche), du signal passe-bas (figures centrales) et du signal passe-bande (figures de droite) pour un nombre de statistiques N_{stat} égal à 1 (figures du haut) et 100 (figures du bas)

II : Généralisation aux modulations M-aires

1. Modulations M-PSK (M-ary Phase Shift Keying)

(Programme MATLAB intitulé `modulateur_v2.m`)

Adapter le programme **modulateurv0.m** pour qu'il puisse traiter toutes les modulations de phase (cas $M > 2$). Une étape de codage supplémentaire sera rajoutée entre la génération de bits et la génération de symboles dans la mesure où les fonctions `pskmod()` n'acceptent, en entrée, que des entiers entre 0 et $M-1$.

Indication : utiliser la fonction `bi2de()`.

a. Influence de l'indice de modulation

Changer la valeur de M en 4 puis 8. Le décalage de phase initial pour le cas $M=4$ sera de $\pi/4$. Représenter les symboles avec l'instruction `scatterplot()`. Expliquer les changements observés au niveau des représentations en temps et en fréquence des signaux par rapport à la modulation BPSK.

Compte-rendu : programme MATLAB, intitulé **modulateur_v2.m**, l'ensemble des figures pour le cas $M=4$ et explications.

2. Généralisation aux cas des autres modulations linéaires

(programme MATLAB intitulé `modulateur_v3.m`)

Adapter le programme précédent afin qu'il puisse également traiter les modulations d'amplitude, par la fonction `pammod()`, et les modulations de phase en quadrature, par la fonction `qammod()`.

Compte-rendu : programme MATLAB, intitulé **modulateur_v3.m**, et l'ensemble des figures pour une modulation 16-QAM et 4-PAM. Expliquer les changements observés au niveau des représentations en temps et en fréquence des signaux par rapport à la modulation BPSK.

3. Comparaison des modulations : efficacité spectrale

Les modulations sont comparées selon leur efficacité spectrale. Pour des valeurs de M de 2, 4, 8 et 16, donner l'occupation spectrale de la modulation numérique. En déduire l'efficacité spectrale et

remplir le tableau suivant. Commenter et conclure sur l'efficacité spectrale des modulations numériques. (*Partie optionnelle : Modulation M-FSK*)

| | M-PAM | | | | M-PSK | | | | M-QAM | | | |
|----------------------|-------|---|---|----|-------|---|---|----|-------|---|---|----|
| | 2 | 4 | 8 | 16 | 2 | 4 | 8 | 16 | 2 | 4 | 8 | 16 |
| Occupation spectrale | | | | | | | | | | | | |
| Efficacité spectrale | | | | | | | | | | | | |

Compte-rendu : le tableau rempli, les commentaires et les conclusions.

Partie 2 : Récepteurs Numériques

Introduction

L'objectif de cette partie consiste à étudier la conception des récepteurs numériques (voir Figure 11). L'architecture optimale sera implantée et simulée. Des versions sous-optimales seront également implantées afin d'appréhender les éléments caractéristiques de l'architecture optimale.

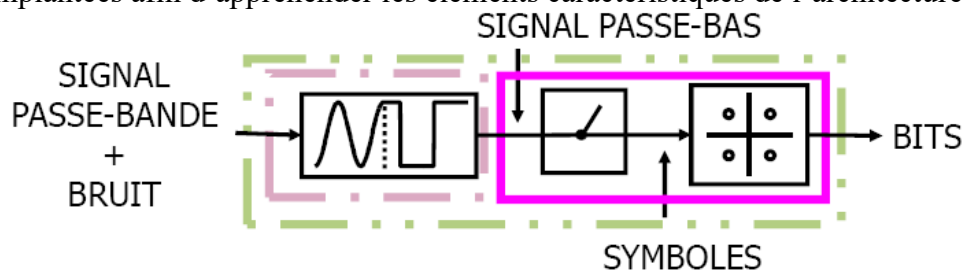


Figure 11 : réception de modulation numériques - schéma de principe

Le caractère optimal fait référence ici au fait que le récepteur proposé permet d'atteindre les meilleures performances en termes de BER (Bit Error Rate) dans le cas d'un canal AWGN (Additive White Gaussian Noise). Des lors que le canal n'est plus AWGN, le récepteur proposé n'est plus optimal.

Cette partie commence par l'étude de la modulation de phase BPSK. Puis, l'étude est généralisée aux autres modulations de phase, aux modulations d'amplitude et aux modulations d'amplitude en quadrature, et enfin aux modulations de fréquence.

1. Récepteur pour modulation BPSK (programme MATLAB intitulé demodulateur_v0.m)

e. Retour en bande de base

Le retour en bande de base se fait sur deux voies. Le signal reçu est multiplié par deux porteuses pures de la forme $\cos[2\pi f_0 t + \varphi]$ et $-\sin[2\pi f_0 t + \varphi]$ ou $f_0 = f_c$ et $\varphi = 0$. Pour les modulations BPSK, cela permet de traiter les modulations dont les symboles sont de la forme $\exp(j\theta)$ et $\exp[j(\theta + \pi)]$.

f. Filtrage adapté

La suppression des lobes secondaires en $-2f_c$ et $+2f_c$, et la maximisation du rapport signal à bruit aux instants de prise de décision, se font grâce au filtrage adapté. Filtrer le signal (après le retour en bande de base) par un filtre adapté au filtre de mise en forme.

NB : si le filtre de mise en forme est de la forme $g(t)$, alors la réponse impulsionnelle du filtre adapté est de la forme $g^*(\tau - t)$ ou τ est un retard servant à compenser les retards introduits par les différentes étapes de filtrage.

La permutation de la réponse impulsionnelle du filtre de mise en forme se fait par la fonction `fliplr()`.

g. Sous-échantillonnage et prise de décision

Sous-échantillonner le signal à la sortie du filtre adapté au rythme symbole T . Représenter les échantillons avec l'instruction `scatterplot()`. Décoder les symboles reçus par l'instruction `pskdemod()`. Comparer les bits estimés et les bits reçus. Conclure.

NB : tenir compte des retards introduits par le filtre de mise en forme et le filtre adapté (temps de propagation de groupe) dans le sous-échantillonnage. En particulier, le premier échantillon, étant donné la forme des filtres (porte de durée T), est prélevé à l'instant T .

2. Visualisation des résultats

a. Représentations en temps

Superposer, sur la même figure, les signaux suivants :

- Le peigne de Dirac représentant les symboles émis (seules les amplitudes non-nulles seront tracées),
- Le signal passe-bas à la sortie du filtre de mise en forme,
- Le signal passe-bas à la sortie du filtre adapté,
- Le peigne de Dirac correspondant aux échantillons prélevés toutes les périodes T (seules les Amplitudes non-nulles seront tracées).

L'axe temporel sera exprimé en multiples de la période symbole T . Une figure sera pour la partie réelle et une autre pour la partie imaginaire. Un exemple est fourni Figure 12. Commenter.

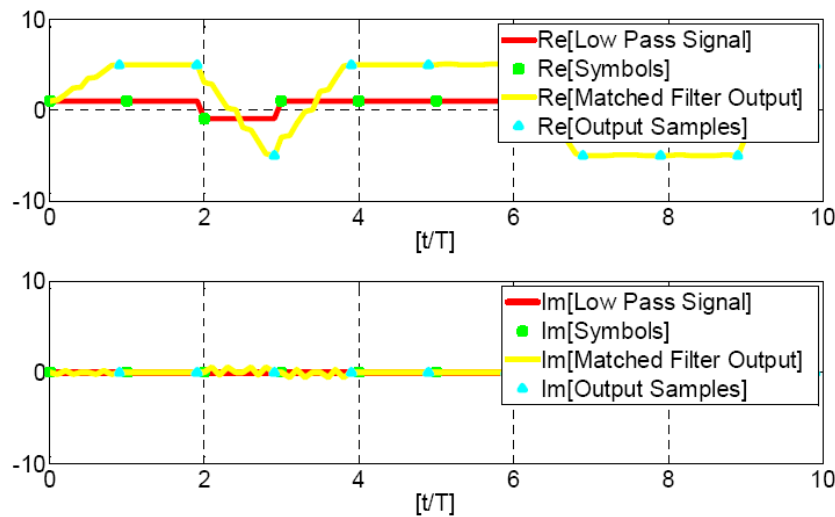


Figure 12 : représentation temporelle de la partie réelle et imaginaire du signal passe-bas (Low Pass Signal), des symboles émis (Symbols), de la sortie du filtre adapté (Matched Filter) et des échantillons prélevés à la sortie du filtre adapté (Output Samples)

b. Représentation en fréquence

Superposer, sur la même figure, les densités spectrales de puissance, entre $-F_s/2$ et $F_s/2$, du signal passe-bande émis et du signal à la sortie du filtre adapté (voir Figure 13). Commenter.

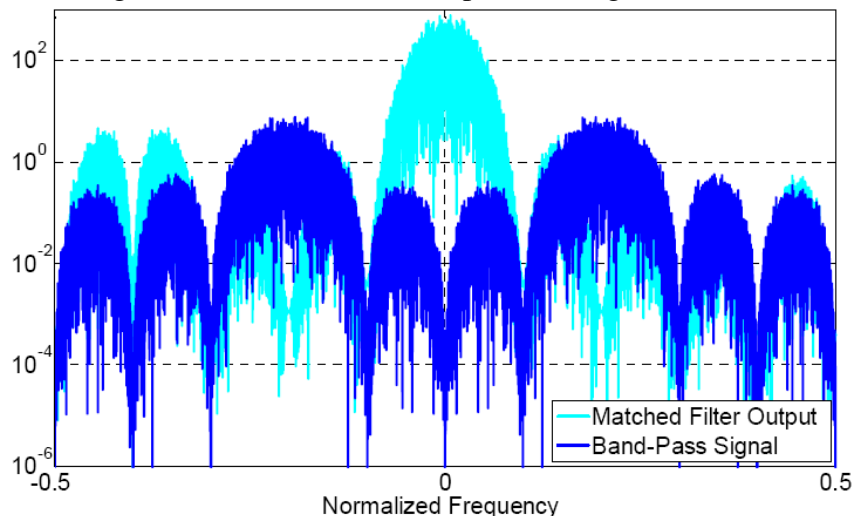


Figure 13 : représentation en fréquence du signal passe-bande émis (Band-Pass Signal) et du signal à la sortie du filtre adapté (Matched Filter Output)

Compte-rendu : le programme MATLAB, intitulé `demodulateur_v0.m`, les représentations en temps et en fréquence, ainsi que les commentaires.

3. Impact de différents paramètres sur la réception des signaux numériques

a. Impact de la synchronisation

Changer la fréquence porteuse du récepteur. Pour cela, rajouter 1% à la valeur de f_c . Sur une même figure, superposer deux signaux : l'un correspondant à la sortie du filtre adapté pour une fréquence porteuse à la réception égale à f_c , et l'autre avec la nouvelle valeur de la fréquence porteuse. Commenter et conclure.

Changer la phase de la porteuse du récepteur. Faire varier la phase de 0 à $\pi/2$ par pas de $\pi/8$ (la valeur de la fréquence porteuse doit reprendre sa valeur initiale). Sur une même figure, superposer les signaux à la sortie du filtre adapté, correspondants aux quatre valeurs de la phase. Commenter et conclure.

Compte-rendu : les figures, les commentaires et les conclusions.

| | Ancienne valeur | Nouvelle valeur |
|-----------|-----------------|----------------------------------|
| f_0 | f_c | $f_c + 1\%$ |
| φ | 0 | $0, \pi/8, \pi/4, 3\pi/8, \pi/2$ |

b. Impact du filtre adapté

Tester les combinaisons filtre de mise en forme – filtre adapté référencées dans le tableau ci-dessous. Commenter et conclure.

| | Filtre à l'émission | Filtre à la réception |
|--|--------------------------|----------------------------|
| Filtre porte de durée T et d'amplitude unité | $g = \text{ones}(1, T)$ | $h = \text{fliplr}(g)$ |
| Filtre aléatoire | $g = \text{randn}(1, T)$ | $h = \text{fliplr}(g)$ |
| Filtre non adapté | $g = \text{ones}(1, T)$ | $h = [0:T/2 \ T/2-1:-1:1]$ |
| Filtre non inversé | $g = \text{randn}(1, T)$ | $h = g$ |

Compte-rendu : les commentaires et les conclusions.

4. Réception des modulations M-PSK (programme MATLAB intitulé `démodulateur_v1.m`)

Adapter le programme précédent afin de pouvoir traiter toutes les modulations de phase. Le récepteur comportera deux voies de réception : l'une pour la voie en phase, l'autre pour la voie en quadrature. Par ailleurs, une étape supplémentaire de décodage sera rajoutée après le décodage des symboles afin de restituer les bits émis. Expliquer les différences et les similitudes avec le cas BPSK.

Indication : utiliser la fonction `de2bi()`.

Compte-rendu : programme MATLAB `demodulateur_v1.m`, les figures pour le cas $M=4$ et explications.

5. Réception des autres modulations linéaires (programme MATLAB intitulé `démodulateur_v2.m`)

Adapter le programme précédent afin qu'il puisse également traiter les modulations d'amplitude M-PAM et les modulations de phase en quadrature M-QAM.

Indication : un facteur $1/2 E_g$ a été introduit par la transmission, ou E_g désigne l'énergie du filtre de mise en forme. Ce facteur doit être pris en compte lors de la démodulation.

Compte-rendu : programme MATLAB `demodulateur_v2.m` et les explications relatives au facteur de normalisation.

Partie 3 : Performances des transmissions numériques en présence d'un canal AWGN

Introduction

L'objectif de cette partie est d'établir les performances des transmissions numériques en présence d'un canal AWGN (Additive White Gaussian Noise). Dans un premier temps, les performances d'une transmission BPSK seront établies ; puis les modulations M-PSK seront étudiées avant de passer aux modulations M-PAM, M-QAM et M-FSK.

1. Simulation d'un canal AWGN

L'hypothèse de canal AWGN consiste à supposer que l'ensemble des perturbations subies par le signal transmis par l'émetteur peut se modéliser sous la forme d'une seule et même source de perturbation aléatoire placée entre l'émetteur et le récepteur. L'ensemble des perturbations comprend notamment les imperfections des équipements électroniques d'émission et de réception (au niveau composants, connectique), les perturbations apportées par le milieu de transmission (câble, air, etc) et les perturbations ou interférences dues à la présence d'autres systèmes et/ou d'autres utilisateurs du système.

L'hypothèse de blancheur du bruit signifie que les échantillons de bruit ne sont pas corrélés les uns aux autres. Cette hypothèse est justifiée par le fait que la valeur d'un échantillon d'une source de bruit particulière n'est pas conditionnée par les valeurs des échantillons des autres sources de bruit. Par exemple, si les défauts de connectique de l'émetteur sont modélisés par une source de bruit, les valeurs aléatoires de cette source de bruit ne sont pas conditionnées par les valeurs de la source de bruit modélisant les perturbations liées au milieu de transmission, et inversement.

Test : tracer la fonction d'auto-corrélation d'un bruit blanc gaussien de moyenne nulle, de variance unité, de 1000 échantillons. Un exemple est fourni dans la figure suivante. Commenter.

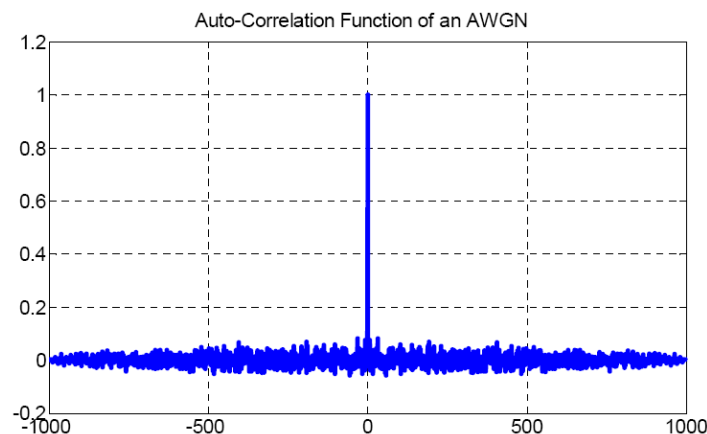


Figure 1 : fonction d'auto-corrélation d'un bruit AWGN de variance unité

L'hypothèse relative au caractère gaussien de la source de bruit vient du théorème de la limite centrale : toute combinaison de N variables aléatoires indépendantes et identiquement distribuées tend vers une loi gaussienne lorsque N tend vers l'infini. Il faut également tenir compte du fait que toute combinaison linéaire de variables aléatoires gaussiennes suit une loi gaussienne. Ainsi, lorsque toutes les sources de bruit sont référencées et classées selon leur densité de probabilité, il est possible d'obtenir, pour chaque classe, une variable aléatoire globale suivant une loi gaussienne qui, une fois combinée aux variables aléatoires globales associées aux autres classes, donne également une loi gaussienne.

Test : tracer l'histogramme d'une variable aléatoire uniformément répartie sur $[0,1[$ comprenant 1000 réalisations. Puis, tracer l'histogramme de la somme de 2, 3 et 10 variables aléatoires. Commenter. Tracer l'histogramme d'un bruit AWGN. Un exemple est fourni dans la figure suivante.

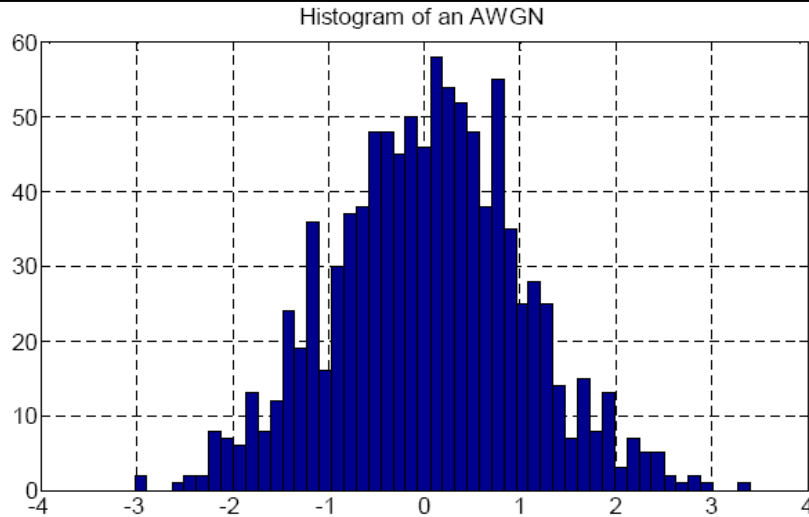


Figure 2 : histogramme d'un bruit AWGN de variance unité (1000 échantillons générés)

Compte-rendu : les courbes d'auto-corrélation, l'histogramme d'un bruit blanc et les commentaires.

2. Génération d'un bruit AWGN et caractéristiques du rapport E_b/N_0 (programme MATLAB intitulé canal.m)

Ajouter un bruit blanc gaussien à un signal QPSK de sorte que le rapport E_b/N_0 soit de 3 dB.

Indication : l'énergie moyenne transmise pour un symbole, E_s , est de la forme

$$E_s = \frac{E_g}{2} \sigma_a^2 \quad \text{avec} \quad E_g = \int_{-\infty}^{+\infty} |g(t)|^2 dt \quad \text{et} \quad \sigma_a^2 = E[|a_k|^2]$$

où E_g est l'énergie du filtre de mise en forme et σ_a^2 est la variance des symboles, à condition que ces derniers soient centrés et où les a_k sont les symboles de la constellation. L'énergie moyenne E_b d'un bit est de la forme $E_s / \log_2(M)$. Le bruit blanc gaussien doit être généré avec une variance $N_0/2$.

Représentation en temps

Sur une première figure, superposer le signal reçu sans bruit et le signal reçu avec bruit. Sur une autre figure, superposer le signal à la sortie du filtre adapté dans le cas sans bruit et dans le cas avec bruit. Commenter.

Représentation en fréquence

Sur une même figure, représenter les densités spectrales de puissance des trois signaux suivants : le signal émis, le signal reçu et le bruit. Commenter. La visualisation dépend de la réalisation des processus observés.

Pour visualiser le comportement moyen, moyenner 100 réalisations et représenter les trois densités spectrales sur la même figure. Commenter. Un exemple est fourni dans la figure suivante.

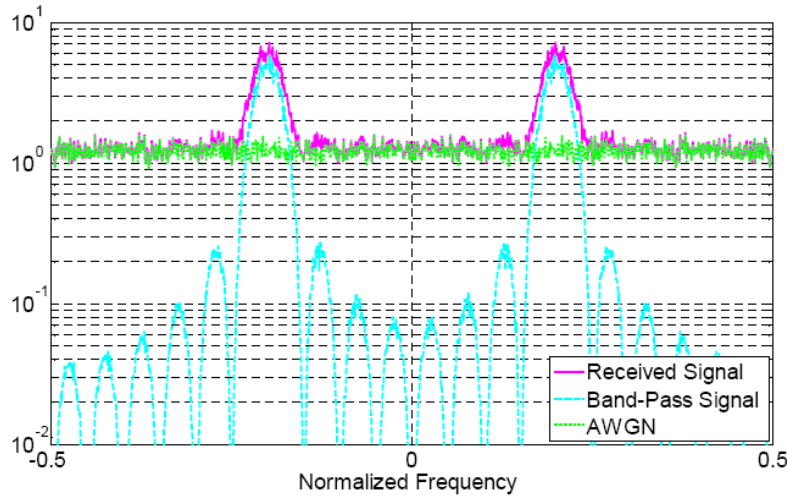


Figure 3 : représentation en fréquence du signal émis (Band-Pass Signal), du signal reçu (Received Signal) et du bruit AWGN (AWGN)

Recommencer les représentations en temps et en fréquence pour un rapport E_b/N_0 de 10 dB. Commenter.

Compte-rendu : le code MATLAB intitulé canal.m, les figures et les commentaires.

3. Performances d'une modulation BPSK dans un canal AWGN (programme MATLAB intitulé perf_BPSK.m)

Tracer la courbe donnant le BER en fonction du rapport E_b/N_0 , pour un rapport allant de 0 à 6 dB. Superposer à la courbe issue des simulations, la courbe théorique

$$P_b = \frac{1}{2} \operatorname{erfc} \sqrt{\frac{E_b}{N_0}}$$

Commenter. Modifier le code MATLAB pour obtenir des courbes complètes. Superposer aux courbes précédentes les courbes d'écart-type sur l'estimation de la probabilité d'erreur par le BER, c'est-à-dire $BER \pm \sigma_{BER}$. Commenter.

Indication : le BER est un estimateur de la probabilité d'erreur P_b avec les caractéristiques suivantes

$$m_{BER} = E[BER] = P_b$$

$$\sigma_{BER}^2 = E[BER^2] - m_{BER}^2 = \frac{P_b(1 - P_b)}{N_b}$$

La précision de la mesure de la probabilité d'erreur est quantifiée par l'erreur relative quadratique moyenne, ε^2 , définie par

$$\varepsilon^2 = \frac{\sigma_{BER}^2}{m_{BER}^2} = \frac{(1 - P_b)}{N_b P_b} \quad \text{avec} \quad \varepsilon^2 = \frac{1}{N_b P_b} \quad \text{si} \quad P_b \ll 1$$

Tracer la précision de la mesure en fonction du rapport E_b/N_0 .

Indication : la quantité $N_b P_b$ représente le nombre d'erreurs commises au cours de la simulation.

Le calcul du BER consistait jusqu'à présent à envoyer un nombre de bits arbitrairement grand pour toutes les valeurs du rapport E_b/N_0 et à observer un certain nombre d'erreurs. Ainsi, par rapport à l'expression suivante du BER

$$BER = \frac{N_e}{N_b}$$

où N_e représente le nombre d'erreurs observées, la procédure précédente consistait à fixer N_b et à compter N_e .

La procédure va maintenant être modifiée. Au lieu de fixer N_b , c'est N_e qui va être fixe. Expliquer l'impact de cette procédure sur la précision de la mesure.

Etablir une nouvelle courbe de BER avec la méthode suivante. Pour chaque valeur du rapport E_b/N_0 , envoyer des blocs de N_b bits, avec N_b assez faible (exemple : $N_b=100$), et accumuler ces erreurs dans une variable intermédiaire N_{error} . Envoyer autant de blocs de N_b bits jusqu'à ce que N_{error} soit supérieur à un seuil N_e fixé ici à 100 ($N_e=100$). Le BER estimé sera alors égale au nombre d'erreurs observées, divisé par N_b , multiplié par le nombre de blocs envoyés.

Commenter les résultats et expliquer les avantages de cette méthode par rapport à la précédente en se fondant sur la notion de précision de la mesure. Tracer également les courbes $BER \pm \sigma_{BER}$ où σ_{BER} est maintenant calculé en fonction de ε^2 . Expliquer et commenter. Un exemple est fourni dans la figure suivante.

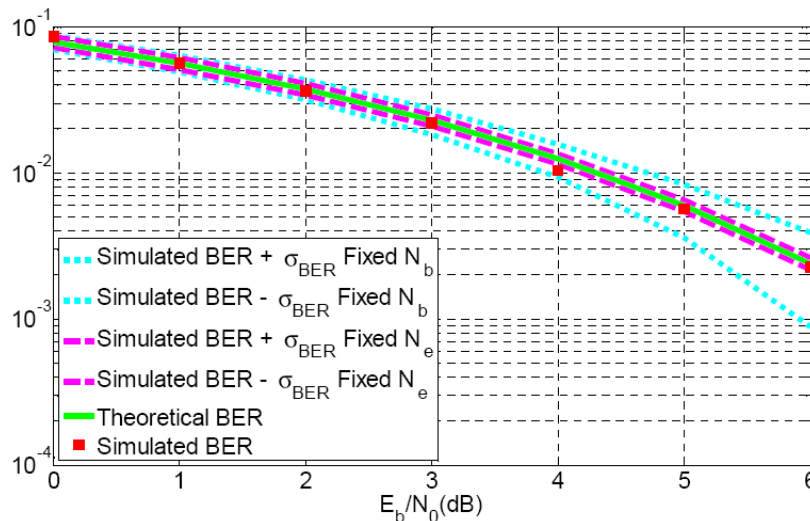


Figure 4 : courbe de BER théorique (Theoretical BER), courbe de BER simulé (Simulated BER), courbes de BER avec les écarts-type de l'estimateur suivant que le nombre de bits émis est fixe (Fixed N_b) ou que le nombre d'erreurs observées est fixe (Fixed N_e)

Compte-rendu : le code MATLAB, nommé perf_BPSK.m, les courbes de BER, les commentaires et explications.

4. Performances d'une modulation QPSK

Généraliser les résultats précédents au cas d'une modulation QPSK. Visualiser la dispersion gaussienne des échantillons prélevés à la sortie du filtre adapté. Un exemple est fourni dans la figure qui suit. Puis, tracer les courbes de BER. Tester les programmes avec et sans le codage de Gray. Commenter et conclure.

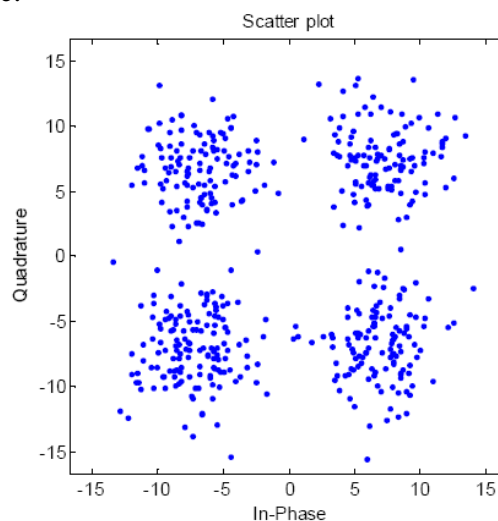


Figure 5 : dispersion des échantillons d'une modulation QPSK avant la prise de décision

Compte-rendu : les courbes de BER et les commentaires.

5. Performances d'autres modulations

Tracer les courbes de BER simulé et de BER théorique pour les modulations M-PAM avec $M=4$, M-QAM avec $M=16$ et BFSK dans la même gamme de rapport E_b/N_0 utilisée précédemment. Commenter. Pour $M=4$ superposer les courbes de BER des modulations 4-PAM, QPSK, 4-QAM et 4-FSK pour un E_b/N_0 de 0 à 10 dB et commenter les résultats.

Compte-rendu : les courbes de BER et les commentaires.

6. Comparaison de l'efficacité en puissance des modulations numériques

Envoyer un signal tel que l'énergie moyenne émise par bit soit unité. Mesurer le BER pour un rapport signal à bruit de 3 dB et remplir le tableau suivant. Commenter et Conclure.

| Modulations | M-PAM | | | | M-PSK | | | | M-QAM |
|--|-------|---|---|----|-------|---|---|----|-------|
| M | 2 | 4 | 8 | 16 | 2 | 4 | 8 | 16 | 16 |
| BER pour un E_b/N_0 de 3dB | | | | | | | | | |

Compte-rendu : le tableau rempli, les commentaires et les conclusions.