



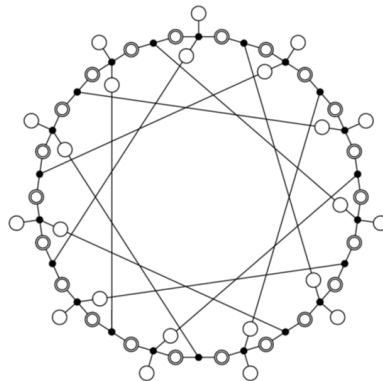
Département Sciences du Numérique

Parcours Télécommunications

Travaux Pratiques et Cours

*Critères de décodage
TP/Cours 1*

Auteur : C. Poulliat



Version 1.0 du
16 décembre 2020



1 Introduction

Nous allons étudier dans la suite le décodage ML séquence et MAP bit au sens large avec des exemples simples de codes : (a) le cas de codes en blocs simples comme le code parité et le code de répétition, (b) le code de Hamming, (c) les codes convolutifs. Le but est de comprendre les différences fondamentales entre ces deux critères.

2 Etude des codes de Répétition et de parité

Un code de répétition consiste en la répétition de N fois un bit d'information. On obtient un code $\mathcal{C}(N, 1)$ de matrice génératrice

$$G_1 = \underbrace{[1 \dots 1 \dots 1]}_N$$

Un code de vérification de parité est construit tel que $c_{N-1} = u_0 \oplus u_1 \oplus \dots \oplus u_{N-2}$ définissant un code $\mathcal{C}(N, N-1)$. Sa matrice génératrice est alors donnée par

$$G_2 = \begin{pmatrix} & 1 \\ & \vdots \\ I_{N-1} & 1 \\ & \vdots \\ & 1 \end{pmatrix}$$

On peut alors remarquer que les deux codes précédents sont duaux, ie.

$$G_1 G_2^T = \mathbf{0}$$

1. Pour chacun de ces codes donner la matrice de parité associée.
2. Pour chacun de ces codes, dériver le décodeur ML dans le domaine des probabilités pour une entrée binaire.
3. Pour chacun de ces codes, donner une écriture de ces décodeurs dans le domaine LLR.
4. Sur le modèle du démodulateur souple SISO, donner une expression du détecteur MAP fonction des vraisemblances pour chaque bits à partir de

$$\hat{c}_n = \arg \max_{c'_n} p(c'_n | \mathbf{y}). \quad (1)$$

5. Pour chacun de ces codes, dériver le décodeur MAP bit dans le domaine logarithmique. Comment cela décline -t-il pour le canal Gaussien à entrées binaires ? Que pouvez vous dire pour le cas du décodeur MAP bit du code de parité ? Comment interpréter ce décodeur par une représentation fréquentielle ?
6. Programmer des fonctions Matlab qui réalisent le décodage ML et MAP bit des codes de parité et de répétition.

Pour réaliser l'énumération des mots de codes, vous pourrez utiliser la commande suivante



```
u=de2bi((0:2^(N-1)-1)', 'left-msb');
```

L'encodage à partir d'une matrice génératrice G peut s'écrire

```
c=rem(u*G,2);
```

3 Décodage ML et MAP bit d'un code de Hamming

On considérera un code de Hamming dont la matrice de parité est donnée par

$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

1. Programmer une fonction Matlab qui réalise le décodage ML par intercorrélation.
2. Programmer une fonction Matlab qui réalise le décodage MAP bit par calcul direct.

4 Décodage ML et MAP bit d'un code convolutif

Sous Matlab, on peut utiliser des objets associés à des blocs fonctionnels d'une chaîne de communication. Parmi les objets, on peut utiliser ceux associés à l'encodage et au décodage des codes convolutifs. Pour l'encodage, la syntaxe est la suivante :

```
%Coding parameters
hConEnc = comm.ConvolutionalEncoder('TrellisStructure', poly2trellis(3, ...
    [5 7]), 'TerminationMethod', 'Terminated');
```

On pourra alors utiliser la démarche suivante pour l'application de ces objets :

```
%random data generation
bits = randi([0 1],N,1);

%encode convolutional codes
encodedData = step(hConEnc, data);

%or equivalently
encodedData = hConEnc(data);
```

Pour le décodage, on pourra alors se référer aux objets suivants dont on vérifiera la paramétrisation au préalable (!) :

```
%MAP object decoding parameters exemple
hAPPDec = comm.APPDecoder(...
    'TrellisStructure', poly2trellis(7, [171 133]), ...
    'Algorithm', 'True APP', 'CodedBitLLROutputPort', false);
```



```
%Viterbi Decoder object decoding parameters exemple
hVITDec = comm.ViterbiDecoder('InputFormat','Unquantized', 'TerminationMethod','↵
    Truncated' );
```

La détection se fera comme suit

```
%ViterbiDecoder
receivedBitsML=step(hVITDec, demodSignal);

% Decode the convolutionally encoded data. The APP decoder assumes a
% polarization of the soft inputs that may be inverse to that of the
% demodulator soft outputs, so check the sign of demodSignal
receivedSoftBits = step(hAPPDec, zeros(N,1), demodSignal);

% Convert soft decisions to hard decisions
receivedBitsMAP = double(receivedSoftBits > 0);
```

1. Comparer les performances du décodages ML pour des codes non récursifs et récursifs en utilisant les ressource Matlab pour un canal gaussien en utilisant une BPSK.
2. Pour le cas récursif, comparer au décodage MAP bit pour des codes convolutifs avec fermeture de treillis directe.
3. Comment gérer correctement une fermeture circulaire du treillis pour un code non systématique et récursif?