

Práctica 4 - Parte 1. Segmentación en símbolos OFDM: Estimación del modo, la duración del intervalo de guarda y sincronización temporal en DVB-T

Jesús del Castillo Cabrera (grjesus99@correo.ugr.es)
Sergio Fernández Morales (serfermor@correo.ugr.es)

Abstract—En este documento se realizan las tareas referentes a la primera parte de la práctica de segmentación de símbolos OFDM, para ello se nos aportan dos ficheros que contienen dos transmisiones OFDM, y a partir de los conceptos estudiados hallaremos el modo y la duración del intervalo de guarda para así conseguir la segmentación de símbolos.

I. INTRODUCCIÓN

En los archivos de datos OFDM1 y OFDM2 se han almacenado distintos instantes de una transmisión de televisión digital utilizando el estándar DVB-T. Nuestra primera preocupación será averiguar de qué modo se trata, si 8k o 2k, de esta forma sabremos el número de portadoras que se transmiten, para esto utilizaremos la función métrica de autocorrelación. Una vez obtenido el modo, podremos hallar el intervalo de guarda con un método similar. Con estos datos podremos conseguir sincronizar temporalmente nuestros símbolos y observaremos cuántos hay en cada archivo.

II. ESTIMACIÓN DEL MODO DVB-T

Como ya se ha comentado utilizamos la función métrica de autocorrelación, esta ha sido implementada en Matlab™, en los scripts 'OFDM1.M' y 'OFDM2.m' respectivos a cada uno de los archivos de datos.

En primer lugar debemos cargar los datos tal y como se comentó en la sesión de prácticas.

```
%Cargamos el fichero OFDM1
fileID = fopen('OFDM1','r');
newrow=1;
datareal=fread(fileID,'double');
Ndatareal=length(datareal);

%Agrupamos los valores devuelto por el vector real
'Ndatareal' en parejas
%de valores complejos
for row=1:2:Ndatareal
    data(newrow)=complex(datareal(row),datareal(row
+1));
    newrow=newrow+1;
end

%Cerramos el fichero fileID
fclose(fileID);
```

Para calcular la función métrica aplicamos la siguiente expresión:

$$\rho_{N_{FFT},N_W}(n) = \frac{1}{N_W} \sum_{i=0}^{N_W-1} \tilde{r}^*(n+1) \tilde{r}^*(n+i+N_{FFT}) \quad (1)$$

Para ello utilizamos la función que se encuentra en el archivo 'Max_Corr_Method.m'. Simplemente computa el bucle necesario para dicho cálculo.

```
function rho=Max_Corr_Method(Nw,r,Nd,Nfft)

rho=zeros(1,Nd);

for i=0:Nd-1
    suma=0;
    for j=0:Nw-1
        suma=conj(r(j+i+1))*r(j+i+Nfft+1)+suma;
    end
    rho(i+1)=(suma)/Nw;
end
end
```

A continuación se muestra la representación de dicha función para ambos archivos, y ambos modos.

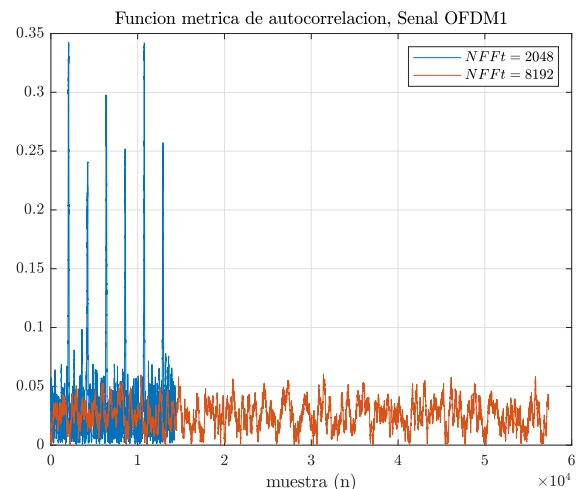


Figura 1: Función métrica de autocorrelación archivo OFDM1

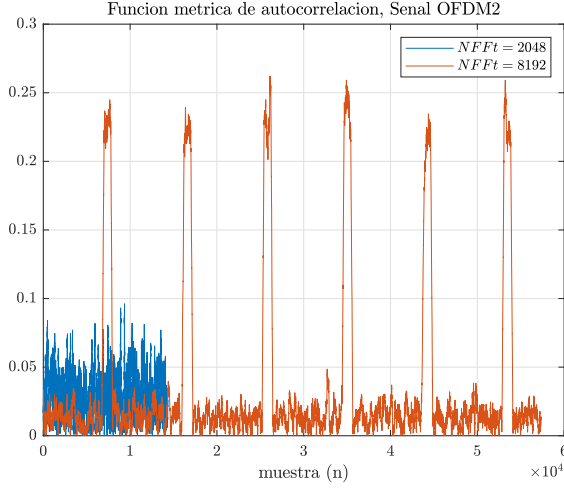


Figura 2: Función métrica de autocorrelación archivo OFDM2.

A partir de estas imágenes ya podemos comprobar que el primer archivo corresponde a modo 2k y el segundo a 8k, pues se observa como cada cierto periodo de símbolo (según el modo al que corresponda), la función de autocorrelación es periódica presentando unas zonas de mayor valor donde existe gran coincidencia.

De forma analítica se puede hallar el modo buscando aquel que maximice la 'variance to average ratio metric', para ello previamente tendremos que calcular la potencia y la media de la función.

$$\hat{N}_{FFT} = \arg \max(N_{FFT}) \frac{P_{N_{FFT}, N_W} - A_{N_{FFT}, N_W}^2}{A_{N_{FFT}, N_W}} \quad (2)$$

```
%Calculo de la potencia y la media
PNfft1=(1/Nd)*sum(abs(rho).*abs(rho));
ANfft1=(1/Nd)*sum(abs(rho));

NFFT_1=(PNfft(1)-ANfft(1)^2)/ANfft(1);
NFFT_2=(PNfft(2)-ANfft(2)^2)/ANfft(2);

max_OFDM1=max(NFFT_1,NFFT_2);
```

Como ya preveíamos a en base a la gráfica anterior, el archivo *OFDM1* opera con el modo 2k y el archivo *OFDM2* con 8k. Se han obtenido los siguientes valores de 'variance to average ratio metric'; para OFDM1 0.051 para 2k y 0.0049 para 8k; para OFDM2 0.0081 para 2k y 0.1084 para 8k. Simplemente ha bastado con seleccionar el modo que maximice dicho valor.

III. ESTIMACIÓN DE LA DURACIÓN DEL INTERVALO DE GUARDA EN DVB-T

Una vez averiguado a que modo pertenecen ambas transmisiones OFDM, debemos hallar la duración del prefijo cíclico. Como sabemos este puede ser $\frac{1}{32}$, $\frac{1}{16}$, $\frac{1}{8}$ ó $\frac{1}{4}$ de la longitud del símbolo (se construye concatenando los últimos símbolos paralelos al inicio del símbolo OFDM).

Para hallar esta duración utilizamos la función métrica de autocorrelación ya calculada, pero únicamente para el modo correspondiente. Nombramos la siguiente variable como el máximo de dicha función sobre una ventana de tamaño Nd de análisis.

$$\rho'_{N_{FFT}, N_W}(n) = \max |\rho_{N_{FFT}, N_W}(n)| \quad (3)$$

Dividimos este máximo absoluto entre tres y buscamos en la autocorrelación normalizada el número de elementos que superan este umbral. Para eso utilizamos la función $thr(a,b)$, tal que al introducirle dos números (a,b), devuelve 1 si el primero es mayor o cero si es menor.

$$\Theta_{N_{FFT}, N_W}(n) = \frac{1}{N_D} \sum_{n=0}^{N_D-1} thr(|\rho_{N_{FFT}, N_W}(n)|, \rho'_{N_{FFT}, N_W}(n)/3) \quad (4)$$

Con esto calculamos el porcentaje de elementos que superan el umbral. Esto lo realizamos con cada tamaño de intervalo de guarda, buscando minimizar la siguiente expresión.

$$\hat{N}_G = \arg \min(N_G) \left| \Theta_{N_{FFT}, N_W} - \frac{N_G}{N_G + \hat{N}_{FFT}} \right| \quad (5)$$

Este proceso se ha implementado de forma más compacta en la función 'guard_interval.m', la cual compara dicho valor para cada una de las posibles duraciones y devuelve la duración del prefijo cíclico (de las posibles candidatas) que minimiza la expresión.

```
function [Ng_est,thr,rho_max] = guard_interval(Nd,
rho,NFFT_2K)

rho_max=max(abs(rho)); %Calculo de la relacion

suma=0; %Inicializamos la sumatorio del la eq(5) a
cero

%Funcion metrica de correlacion con el valor NFFT
estimado, eq(5)
thr=zeros(1,Nd);

for i=1:Nd
    if abs(rho(i))>(rho_max/3)
        thr(i)=1;
    else
        thr(i)=0;
    end
    suma=sum(thr);
end

sigma=(1/Nd)*suma;

%Calculo de la duracion estimada en numero de
muestras del intervalo de
%guarda

Ng=NFFT_2K*[1/32 1/16 1/8 1/4]; %Posibles
duraciones del intervalo de
%guarda

Ng_prima=zeros(1,length(Ng)); %Vector que alojara
las estimaciones de
%la duracion del intervalo de guarda

for i=1:length(Ng)
    Ng_prima(i)=abs(sigma-((Ng(i))/(Ng(i)+NFFT_2K)));
end

%Estimacion del argumento que minimiza la ecuacion
[~,pos]=min(Ng_prima);
Ng_est=Ng(pos);

end
```

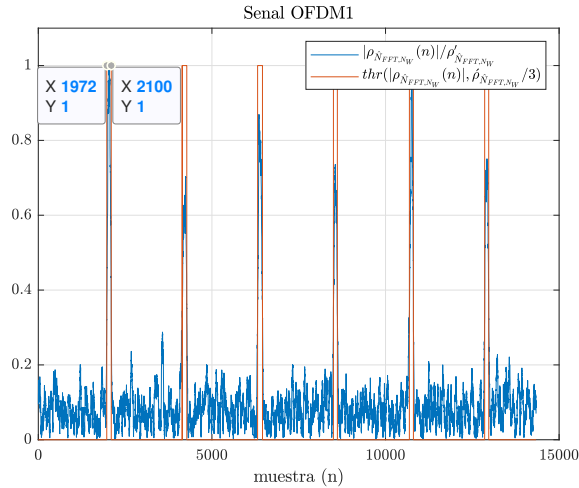


Figura 3: Funciones $|\rho_{NFFT,Nw}(n)|/\rho'_{NFFT,Nw}(n)$ y $thr(|\rho_{NFFT,Nw}(n)|, \rho'_{NFFT,Nw}(n)/3)$ para la señal OFDM1

Para el primer fichero observamos una duración de 64 símbolos de prefijo cíclico que corresponde a $\frac{1}{32}$ de la duración del símbolo paralelo.

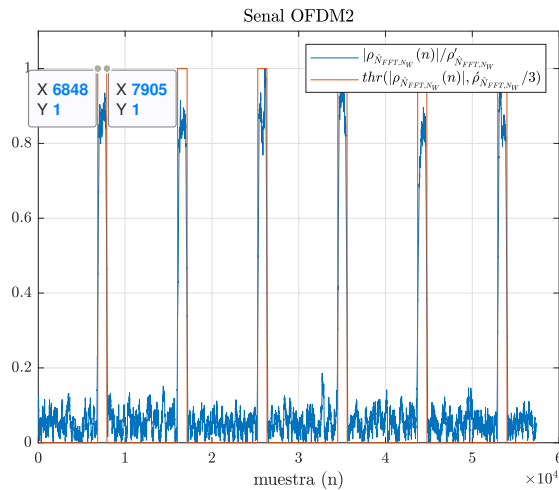


Figura 4: Funciones $|\rho_{NFFT,Nw}(n)|/\rho'_{NFFT,Nw}(n)$ y $thr(|\rho_{NFFT,Nw}(n)|, \rho'_{NFFT,Nw}(n)/3)$ para la señal OFDM2

Para el segundo fichero la duración es de 1/8 símbolo paralelo que corresponde con 1024 muestras complejas, pues vemos que la duración en muestras del pulso determinado por la función $thr(a,b)=1$ vale $7905-6848=1057$, siendo el tamaño válido de prefijo cíclico mas cercano a dicho resultado 1024.

En estas gráficas observamos como la función threshold nos devuelve un tren de pulsos, donde la duración del pulso corresponde con el intervalo de guarda y la distancia entre pulso y pulso al número de portadoras del correspondiente modo.

IV. SINCRONIZACIÓN TEMPORAL EN DVB-T

Por último debemos diseñar un algoritmo que sea capaz de sincronizar nuestros archivos OFDM, es decir, buscar donde empiezan los distintos símbolos, para así poder realizar una fragmentación.

Para esto volvemos a utilizar la función métrica de correlación, tomamos un tamaño de señal igual al intervalo de guarda previamente obtenido, y buscamos que se haga máxima para un periodo igual a la duración del símbolo paralelo para dicho modo.

```
%% Sincronizacion temporal en DVB-T

len_OFDM=NFFT_2K+Ng_est; %Longitud del simbolo OFDM
len_total=length(data); %Longitud total del vector
                        de simbolos PSK

rho=zeros(1,len_OFDM);

for i=0:len_OFDM-1
    suma=0;
    for j=0:Ng_est-1
        suma=conj(data(j+i+1))*data(j+i+NFFT_2K+1)+suma;
    end
    rho(i+1)=(suma)/Nw;
end

rho_abs=abs(rho);

%Buscamos el valor maximo asi como su indice
max_rho=find(rho_abs==max(rho_abs));
maximo=max(rho_abs);
ini_n=max_rho-2+Ng_est;
```

De esta forma conseguimos el punto en el que empieza nuestra señal, y ya solo tenemos que ir sumando la duración del símbolo OFDM, y comprobar cuantos hay. Para esto hay que tener en cuenta que el fichero empieza en la muestra 0.

```
%% Segmentamos en simbolos OFDM
n_sim=1;
l_archivo=length(data);

while ((ini_n+(n_sim-1)*(NFFT_2K+Ng_est))<=l_archivo)
    && ((ini_n+(n_sim-1)*(NFFT_2K+Ng_est)+NFFT_2K-1<=
        l_archivo))
    if n_sim==1
        sim_vec=data(ini_n:ini_n+NFFT_2K-1);
        ini_pos=ini_n;
        end_pos=ini_n+NFFT_2K-1;
    else
        sim_vec=vertcat(sim_vec,data(ini_n+(
            n_sim-1)*(NFFT_2K+Ng_est):ini_n+(n_sim-1)*(
            NFFT_2K+Ng_est)+NFFT_2K-1));
        ini_pos=horzcat(ini_pos,ini_n+(n_sim-1)*(
            NFFT_2K+Ng_est));
        end_pos=horzcat(end_pos,ini_n+(n_sim-1)*(
            NFFT_2K+Ng_est)+NFFT_2K-1);
    end
    n_sim=n_sim+1;
end
```

Para tener una idea visual de este proceso, observamos cómo varía la función métrica de correlación para distintos valores de n . El punto máximo se corresponderá con el punto sobre el que comienza el prefijo cíclico correspondiente a cada símbolo OFDM (ver figuras 5 y 6).

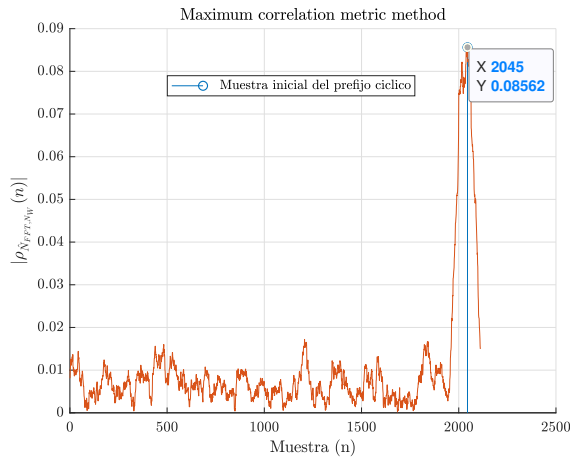


Figura 5: Muestra inicial (prefijo cíclico) para primer símbolo de la señal OFDM1.

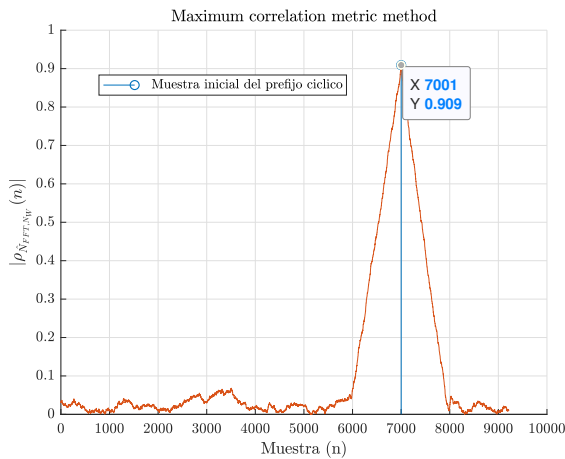


Figura 6: Muestra inicial (prefijo cíclico) primer símbolo señal OFDM2

Obtenemos que para la señal OFDM1 hay 33 símbolos completos y comienzan en la muestra 2109, y para la señal OFDM2 hay 6 símbolos OFDM completos y empiezan en la muestra compleja número 8025.

Por ejemplo, para saber la muestra inicial del primer símbolo OFDM completo en el fichero OFDM1 se procedería sumando a la muestra inicial del prefijo cíclico ($n=2045$) la duración del intervalo de guarda ($N_g=64$ muestras), indicándonos que el primer símbolo OFDM completo comienza en la muestra $n=2109$.

En las siguientes tablas se muestran el rango de muestras que abarcan cada uno de los símbolos OFDM para cada fichero analizado:

Fichero OFDM2		
Número de símbolo	Muestra inicial	Muestra final
1	8025	16216
2	17241	25432
3	26457	34648
4	35673	43864
5	44889	53080
6	54105	62296

Fichero OFDM1		
Número de símbolo	Muestra inicial	Muestra final
1	2109	4156
2	4221	6268
3	6333	8380
4	8445	10492
5	10557	12604
6	12669	14716
7	14781	16828
8	16893	18940
9	19005	21052
10	21117	23164
11	23229	25276
12	25341	27388
13	27453	29500
14	29565	31612
15	31677	33724
16	33789	35836
17	35901	37948
18	38013	40060
19	40125	42172
20	42237	44284
21	44349	46396
22	46461	48508
23	48573	50620
24	50685	52732
25	52797	54844
26	54909	56956
27	57021	59068
28	59133	61180
29	61245	63292
30	63357	65404
31	65469	67516
32	67581	69628
33	69693	71740

V. CONCLUSIÓN

En esta práctica hemos sido capaces de diseñar un algoritmo de detección automática del tipo de modo y de la duración del intervalo de guarda para el estándar DVB-T, lo cual resulta muy útil para una correcta sincronización para un posterior procesamiento (demodulación).