



SAPIENZA  
UNIVERSITÀ DI ROMA

Sapienza Università di Roma  
Facoltà di Ingegneria  
Dipartimento di Ingegneria dell'Informazione, Elettronica e Telecomunicazioni

*Dottorato di Ricerca in Telerilevamento  
XXIV ciclo*

*DVB Passive Radar: signal processing and  
experimental results*

*Dottorando  
Relatore*

*Diego Langellotti  
Prof. Pierfrancesco Lombardo*

## **INDICE**

1	Introduzione.....	5
1.1	Obiettivi della tesi .....	11
1.2	Schema di elaborazione del radar passivo basato su segnale DVB.....	11
1.3	Modello del segnale ricevuto dal sistema PBR.....	13
2	Descrizione dei segnali d'opportunità .....	15
2.1	Le caratteristiche del segnale DVB-T .....	15
2.1.1	Struttura del Frame OFDM.....	16
2.1.2	Caratteristiche dello spettro del segnale DVB-T.....	20
2.1.3	Costellazioni del segnale.....	21
2.1.4	Segnali di riferimento.....	21
2.1.5	Esempio mappatura portanti .....	25
2.2	Caratteristiche della rete digitale terrestre .....	26
2.3	Schema di acquisizione utilizzato per i test sul segnale DVB-T.....	29
2.4	Le caratteristiche del segnale DVB-SH .....	32
2.4.1	Diagramma a blocchi di un trasmettitore DVB-SH.....	33
2.4.2	Struttura del Frame SH.....	34
2.4.3	Mappatura dei bit nei casi TDM e/o OFDM .....	37
2.4.4	Struttura del Frame TDM.....	38
2.4.5	Baseband shaping and quadrature modulation .....	39
2.4.6	Struttura del Frame OFDM.....	40
2.5	Schema di acquisizione utilizzato per i test sul segnale DVB-SH.....	41
3	Controllo dei lobi della funzione di ambiguità.....	45
3.1	Stato dell'arte.....	45
3.2	Sincronizzazione del segnale DVB in un radar passivo.....	49
3.2.1	Sincronizzazione Pre-FFT: ML-Estimator.....	51
3.2.2	Sincronizzazione Post-FFT: Fine Synchronization .....	53
3.2.3	Stima della parte intera dell'Offset in frequenza .....	54
3.2.4	Stima della parte frazionaria dell'Offset in frequenza .....	55
3.2.5	Stima dell'inizio del Frame OFDM .....	57

3.3	Nuovo dispositivo di filtraggio .....	58
3.3.1	Posizione dei picchi Intra-Simbolo.....	59
3.3.2	Posizione dei picchi Inter-Simbolo.....	60
3.3.3	Descrizione del filtro .....	61
3.4	Confronto prestazionale .....	62
3.4.1	Auto-correlazione .....	62
3.4.2	Picchi residui e perdite di SNR .....	64
3.4.3	Impatto degli errori di sincronizzazione .....	67
3.5	Risultati sperimentali .....	69
3.5.1	Acquisizione dati DVB-T.....	69
3.5.2	Algoritmi di sincronizzazione .....	71
3.5.3	Filtraggio lineare.....	73
4	Equalizzazione del segnale DVB.....	75
4.1	Introduzione.....	75
4.2	Algoritmo di equalizzazione Least Mean Square (LMS) .....	76
4.2.1	Utilizzo dei piloti continui .....	77
4.2.2	Interpolazione nel dominio del tempo.....	77
4.2.3	Interpolazione nel dominio della frequenza .....	79
4.3	Analisi equalizzatore LMS .....	80
4.3.1	Valutazione dell'errore di stima commesso .....	81
4.3.2	Variazione del parametro $\mu$ e del del parametro $\gamma$ .....	82
4.3.3	Variazione dei parametro $M_1, M_2$ .....	85
4.4	Impatto del multipath sulle prestazioni del radar passivo basato su DVB-T .....	86
4.5	Miglioramento delle prestazioni utilizzando l'equalizzatore LMS .....	89
4.6	Risultati sperimentali .....	90
4.6.1	Acquisizione dati DVB-SH.....	90
4.6.2	Demodulazione del Segnale DVB-SH .....	92
4.6.3	Demodulazione del Segnale DVB-T .....	95
5	Utilizzo delle GPU per l'elaborazione del segnale .....	97
5.1	Caratteristiche generali.....	97
5.2	Libreria GPUmat.....	99
5.2.1	Requisiti di sistema e inizializzazione dell'ambiente GPU .....	102
5.2.2	Creazione di variabili GPU e conversione in variabili Matlab.....	103

5.2.3	Indicizzazione delle variabili .....	105
5.2.4	Gestione della memoria della GPU .....	106
5.2.5	Calcolo della FFT e IFFT.....	108
5.3	Confronto prestazionale .....	110
5.3.1	Implementazione effeciente della 2D-CCF .....	110
5.3.2	Implementazione sub-ottima della 2D-CCF.....	111
5.3.3	Risultati teorici e scenari di riferimento.....	113
5.3.4	Risultati sperimentali .....	115
6	Conclusioni .....	119
	Ringraziamenti.....	121
	Appendice A .....	122
	RIFERIMENTI E BIBLIOGRAFIA.....	128

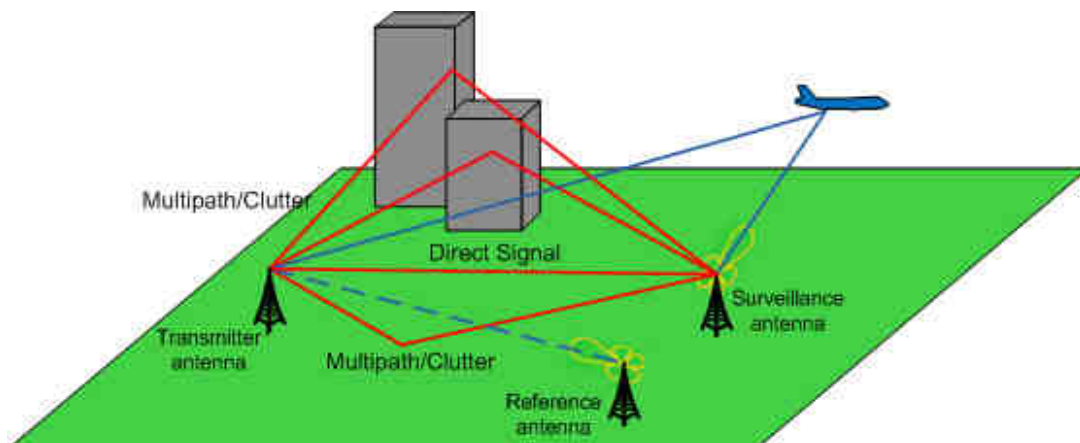
## 1 Introduzione

L'attività di ricerca principale oggetto della mia tesi riguarda i sistemi Radar Passivi Bistatici (PBR).

I PBRs hanno riscosso un forte interesse poichè permettono di rivelare e localizzare potenziali bersagli sfruttando i trasmettitori esistenti come illuminatori di opportunità con vantaggi relativi

- al basso costo di manutenzione grazie alla progettazione della sola parte ricevente;
- alla bassa vulnerabilità alle contromisure elettromagnetiche in quanto, non emettendo onde e.m., risultano difficilmente localizzabili;
- all'impatto ambientale ridotto;

Il principio base della PBR è semplice e schematizzato in Figura 1.



**Figura 1 – Principio di funzionamento del Sistema Radar Passivo**

Un ricevitore passivo raccoglie un segnale contenente i diversi ritorni provenienti da potenziali target ed effettua la correlazione con il segnale ricevuto direttamente dal trasmettitore. In questo modo è possibile estrarre le informazioni sulla posizione e velocità del bersaglio. Va sottolineato che il sistema è completamente passivo e non richiede che i potenziali bersagli siano cooperanti.

La maggiore limitazione in un tale sistema è che la forma d'onda trasmessa non è disponibile al ricevitore come nei convenzionali sistemi radar monostatici e quindi un canale del ricevitore dedicato e separato è necessario per raccogliere la forma d'onda emessa direttamente dal trasmettitore; tale segnale è poi utilizzato come segnale di riferimento ed è correlato con il segnale del canale di sorveglianza per il rilevamento degli

echi dai potenziali target. Come è evidente, il funzionamento del PBR intrinsecamente implica che la forma d'onda trasmessa non è sotto il controllo del progettista radar. Questo è in contrasto con il caso di sistemi radar convenzionali, in cui la forma d'onda trasmessa viene accuratamente progettata per fornire una funzione di ambiguità con caratteristiche adeguate (ad es picco stretto sia in Range che in Doppler e bassi lobi laterali).

Molti trasmettitori, che sono disponibili per differenti applicazioni come telecomunicazioni, sistemi di navigazione e telerilevamento, sono stati usati nel passato come sorgenti di opportunità, [2]-[3].

- I trasmettitori del segnale radio commerciale FM, nella banda 88-108 MHz, sono stati molto utilizzati grazie agli alti livelli di potenza trasmessa, alla ampia copertura che riescono a garantire ed al costo molto contenuto dei ricevitori [4]-[5];
- Trasmettitori di segnale televisivo digitale (DVB-T) e radio digitale (DAB) stanno sostituendo, in Italia e in varie parti del mondo, i corrispettivi trasmettitori di segnali analogici e sono caratterizzati da vaste coperture e bande larghe consentendo un miglioramento in termini di risoluzione in range rispetto ai segnali FM, come riportato nei riferimenti da [6] a [11];
- Trasmettitori di segnali in banda HF, includendo anche trasmettitori di segnali radio AM, Digital Radio Modulation (DRM), and illuminatori radar FMCW OTH esistenti, sono stati utilizzati per la loro potenziale capacità di rivelare bersagli a distanze molto elevate [12]-[13];
- Trasmettitori per stazioni radiobase GSM, UMTS, e WiMAX forniscono una rete molto densa di trasmettitori di opportunità che utilizzano larghezze di banda medie, da [14] a [16];
- Trasmettitori WiFi forniscono sorveglianza molto locale con elevata larghezza di banda per controllare piccole aree esterne o aree indoor, da [17] a [21];
- Sistemi di navigazione come GPS, GLONASS (e nel futuro Galileo), in orbita MEO, forniscono segnali con livelli di Potenza costante e con una qualità controllata e garantita e sono visibili da ogni parte del mondo in ogni istante, [22]-[23];
- Trasmettitori in orbita LEO per servizi di comunicazione satellitare, [24];
- Satelliti Geostazionari/Geosincroni per servizi radio e TV digitale offrono la copertura di una vasta area continentale garantendo valori di EIRP trasmessi alle medie latitudini sulla superficie terrestre, [25];

Il presente lavoro si concentra sulle forme d'onda di opportunità digitali con particolare riferimento al segnale digitale DVB-T (Digital Video Broadcast Terrestrial), legato alle trasmissioni del segnale televisivo terrestre, e DVB-SH (Digital Video Broadcast-Satellite Handheld) relativo alla trasmissione del segnale televisivo via satellite.

Le motivazioni che inducono a scegliere tali segnali come segnali d'opportunità sono legate principalmente alle caratteristiche generali di tali segnali.

L'utilizzo di segnali diversi, come ad esempio segnali radio FM, comporta una massima risoluzione bistatica in range di circa 1,5 km mentre, la larghezza di banda utilizzata dai segnali digitali in questione permette un miglioramento della risoluzione ottenibile significativo passando a risoluzioni di circa 40 metri.

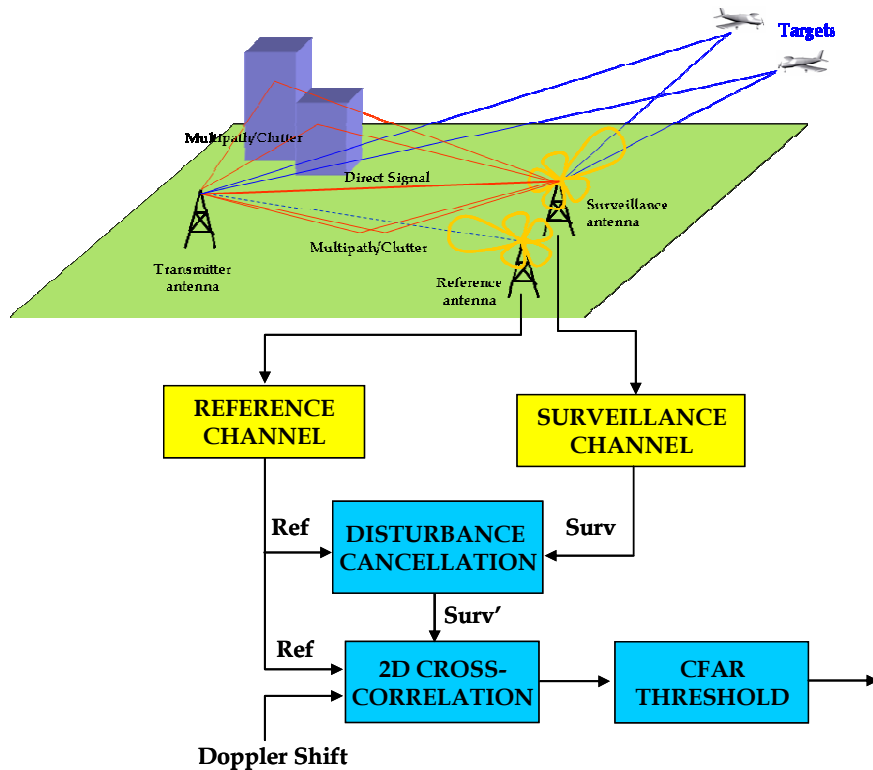
La copertura del territorio è / o sarà vasta sia nel caso di segnale DVB-T che DVB-SH; per il primo segnale, il segnale televisivo digitale terrestre, la copertura, in Italia come in tanti altri paesi europei, è garantita dalla transizione in corso (o dalla pianificazione della transizione) con le trasmissioni analogiche mentre, per il DVB-SH, la trasmissione satellitare garantisce una copertura anche maggiore dei confini nazionali.

A fronte di questi vantaggi esistono anche problemi derivanti dall'utilizzo di questi segnali come forma d'onda opportunità nei sistemi PBR come (i) Equalizzazione del canale di riferimento, (ii) Presenza di picchi spuri interni alla Mappa Range-Doppler, (iii) Costo elevato per l'implementazione della catena di processing ipotizzando l'elaborazione real time.

i. Equalizzazione del canale

Il tipico schema di processing relativo ad un sistema PBR è mostrato in Figura 2. Il sistema prevede un minimo di due canali in ricezione: un canale, denominato canale di riferimento, è dedicato alla registrazione di una copia del segnale d'opportunità, utile per la costruzione del filtro adattato bi-dimensionale, attraverso l'utilizzo di un'antenna direttiva puntata nella direzione del trasmettitore; l'altro canale, denominato canale di sorveglianza, è utilizzato per registrare il segnale riflesso (con livello di potenza molto basso) da un potenziale generico bersaglio presente nell'area di interesse. La prima operazione che viene effettuata è la rimozione dei contributi indesiderati sul canale di sorveglianza; questi contributi sono dovuti, ad esempio, al segnale diretto, trasmesso dall'illuminatore di opportunità, ricevuto dai lobi laterali dell'antenna di sorveglianza oppure, più semplicemente, da forti ritorni di clutter o multipath presenti sul canale di sorveglianza. In

letteratura sono state proposte molte tecniche, come [33], per risolvere questo problema in quanto i diversi disturbi potrebbero oscurare il livello di potenza molto basso di eventuali bersagli e/o introdurre falsi allarmi.



**Figura 2 – Schema di processing base di un PBR.**

La cancellazione quindi del segnale diretto e degli echi di clutter diventa una questione cruciale per il rilevamento di target. Tuttavia, se il segnale di riferimento è affetto da multipath, la capacità cancellazione di questi algoritmi potrebbero essere fortemente limitata impattando in modo diretto sulla rivelazione dei potenziali bersagli.

ii. Presenza dei picchi spuri interni alla Mappa Range-Doppler

Con riferimento alla Figura 2, dopo lo stadio di cancellazione, il processo di rivelazione avviene confrontando con una soglia adeguata, ottenuta con un algoritmo di rivelazione adattivo CFAR (constant false alarm rate), l'uscita del filtro adattato in grado di comprimere i segnali riflessi dai bersagli, caratterizzati dai relativi ritardi e frequenze Doppler. In particolare, la funzione di Cross-Correlazione Bi-dimensionale (2D-CCF), chiamata anche funzione d'ambiguità, può essere valutata attraverso la seguente espressione, [27],

$$\chi(\tau, f_D) = \int_{-\infty}^{\infty} s_{surv}(t) s_{ref}^*(t - \tau) e^{-j2\pi f_D t} dt \quad \text{Eq. 1}$$



Dove

- $S_{\text{surv}}(t)$  rappresenta l'involuppo complesso del segnale ottenuto dal canale di sorveglianza;
- $\tau = R_B/c$  indica il ritardo bistatico rispetto al tempo di arrivo (TDOA) di interesse, dove  $R_B$  è la distanza bistatica relativa e cioè la differenza tra la distanza due-vie e la congiungente trasmettitore-ricevitore (Baseline);
- $f_D = V_B/\lambda$  è lo spostamento in frequenza Doppler di interesse, essendo  $V_B$  la velocità bistatica;
- l'integrale è limitato rispetto al tempo di coerenza dell'elaborazione, ( $T_{\text{int}}$ );

Una funzione di ambiguità ideale dovrebbe presentare un unico lobo, con caratteristiche di risoluzione in distanza e doppler dipendenti dalla larghezza di banda e dal tempo di integrazione utilizzato, ed essere uniforme nel piano ritardo-doppler. L'assenza di lobi secondari e l'andamento uniforme della funzione ridurrebbe tutti i problemi di ambiguità minimizzando i problemi legati al mascheramento di altri bersagli presenti all'interno dell'aria sott'osservazione.

L'utilizzo di segnali DVB come forme d'onda d'opportunità comporta l'introduzione di una serie di picchi indesiderati nella 2D-CCF, che occupano ritardi e frequenze deterministiche, che potrebbe produrre un effetto di mascheramento del segnale utile e/o produrre falsi allarmi degradando le prestazioni in termini di rivelazione dei bersagli; questi picchi sono dovuti alla presenza di caratteristiche specifiche del segnale DVB, come l'intervallo di guardia e le portanti pilota (che verranno dettagliatamente descritte nel seguito). Il livello di questi picchi non può essere ridotto né aumentando il tempo di integrazione né utilizzando reti di pesatura standard (come la finestatura di Hamming, Chebyshev, ect) per il controllo dei lobi laterali.

iii. Implementazione della catena di processing real time

Con riferimento alla Figura 2, la catena di elaborazione del radar passivo base potrebbe richiedere un costo molto elevato, in termini di numero di operazioni da eseguire, soprattutto se si considera il DVB come forma d'onda d'opportunità. L'operazione più onerosa è rappresentata sicuramente dal calcolo della 2D-CCF. In particolare, la larghezza di banda elevata (8 MHz - 7 MHz - 5 MHz) del segnale utilizzato e il tempo di integrazione elevato richiesto, per ottenere il rapporto segnale a rumore accettabile, comportano la gestione e l'elaborazione di segnali composti da un numero di campioni

molto elevato (ordine di  $10^6$  campioni). Inoltre, anche le dimensioni stesse della Mappa potrebbero essere elevate, in termini di celle di risoluzione da valutare. Ad esempio se si utilizzasse un segnale DVB con una larghezza di banda di 8 MHz, e volessimo valutare una mappa Range-Doppler di dimensioni 80 km in range bistatico e  $\pm 300$  m/s in velocità bistatica, ipotizzando un tempo di integrazione pari a 350ms, dovremmo gestire segnali di dimensioni pari a 3,2 milioni di campioni e valutare una mappa con un numero di celle di risoluzione in range di 2440 e velocità di 331.

Nel seguito, dopo la definizione degli obiettivi e la descrizione del sistema radar passivo focalizzando l'attenzione sulla catena di elaborazione, verrà presentato il lavoro di tesi svolto relativo a tecniche di elaborazione del segnale digitale DVB utili per migliorare le prestazioni complessive del Radar Passivo. La prima parte (capitolo 2) è dedicata alla descrizione del segnale d'opportunità scelto; in particolare si descriverà lo standard di trasmissione DVB-T (Digital Video Broadcast Terrestrial) e quello DVB-SH (Digital Video Broadcast-Satellite Handheld). Il segnale DVB-T è legato alle trasmissioni del segnale televisivo terrestre mentre quello DVB-SH è relativo alla trasmissione del segnale televisivo via satellite; all'interno di questa sezione verranno descritti anche i diversi sistemi riceventi adottati e le diverse tecniche di processing digitale da effettuare a valle della ricezione dei segnali. In aggiunta, all'interno di ogni capitolo troviamo una sezione dedicata all'applicazione delle diverse tecniche di processing a set di dati reali registrati con i diversi schemi di acquisizione utilizzati.

Il capitolo 3 descrive, dopo una sezione dedicata alle tecniche per l'estrazione del sincronismo, una tecnica innovativa in grado di migliorare le prestazioni del PBR attraverso il controllo della struttura dei lobi laterali della funzione d'ambiguità. Il capitolo 4 descrive l'operazione di equalizzazione del segnale di riferimento. Prima delle conclusioni, il capitolo 5 descrive l'utilizzo di nuove e diverse risorse di calcolo per risolvere il problema dell'elevato costo computazionale delle diverse operazioni svolte dal sistema radar passivo basato su segnali DVB con particolare riferimento alle Graphics Processing Units (GPU).

## 1.1 Obiettivi della tesi

Con riferimento alle problematiche descritte nella parte introduttiva e dovute all'utilizzo del segnale DVB come forma d'onda d'opportunità, il presente lavoro ha lo scopo di studiare e implementare tecniche di elaborazione che consentano di migliorare le prestazioni di un Radar Passivo che utilizzi tale segnale come forme d'onda d'opportunità.

In particolare, la tesi risolve il problema

- 1- della presenza dei picchi spuri descrivendo delle tecniche per la rimozione di tali picchi presenti all'interno della funzione d'ambiguità.
- 2- della degradazione delle prestazioni a seguito della cancellazione dei disturbi studiando e implementando una tecnica che consenta l'equalizzazione del canale e contestualmente consenta di incrementare le prestazioni.
- 3- implementativo della catena di processing real time attraverso l'utilizzo di nuove e diverse risorse di calcolo.

## 1.2 Schema di elaborazione del radar passivo basato su segnale DVB

In Figura 3 si riporta lo schema di elaborazione del radar passivo proposto nel caso in cui si utilizzi un segnale DVB come trasmettitore d'opportunità.

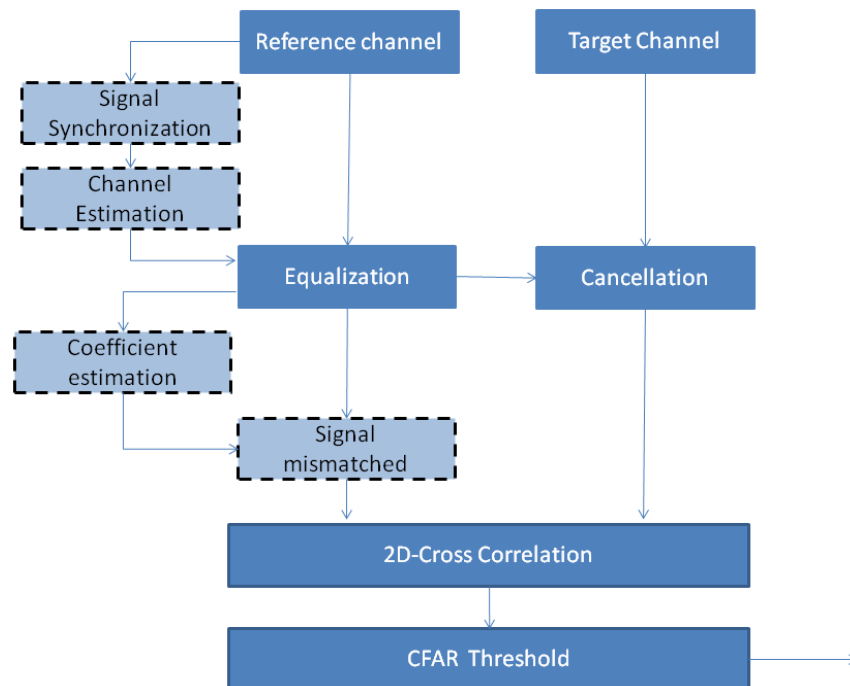


Figura 3 - Schema di processing di un PBR basato su segnale DVB

In particolare si utilizzano due canali in ricezione, riferimento e sorveglianza. La prima operazione, sul canale di sorveglianza, è la rimozione dei contributi indesiderati. Nello specifico, la cancellazione viene effettuata sottraendo al segnale di sorveglianza una copia del segnale di riferimento opportunamente scalato.

$$y(t) = s_{surv}(t) - \alpha \cdot s_{ref}(t) \quad \text{Eq. 2}$$

Il parametro  $\alpha$ , che consente di rimuovere i contributi indesiderati, è ottenuto mediante un approccio Least Square (LS), [33],

$$\min_{\alpha} \{ \|y\|^2 \} \quad \text{Eq. 3}$$

Ottenendo così il risultato desiderato.

$$\alpha = (\mathbf{S}_{ref}^H \mathbf{S}_{ref})^{-1} \mathbf{S}_{ref}^H \mathbf{s}_{surv} \quad \text{Eq. 4}$$

Per quantificare, in termini numerici, la cancellazione si utilizza il parametro CA (Clutter attenuation) definito come il rapporto tra la potenza in ingresso al filtro di cancellazione e quella in uscita.

Siccome, come accennato nell'introduzione, la cancellazione ipotizza un segnale di riferimento simile a quello realmente trasmesso, la prima operazione da effettuare sul canale di riferimento è rappresentata dalla stima del canale di trasmissione. Questa stima viene effettuata a valle della stima del sincronismo del segnale DVB rispetto alla sua struttura interna. Entrambe le stime (sincronismo ed canale) verranno dettagliatamente descritte nei capitoli 3 e 4 rispettivamente.

Dopo l'equalizzazione del segnale di riferimento e la cancellazione dei disturbi sul segnale di sorveglianza viene calcolata la mappa Range-Doppler la cui espressione è riportata in Eq. 1. In questo caso, rispetto al filtro adattato standard, viene effettuata un'operazione preliminare sul segnale di riferimento, disadattando il segnale stesso con un set di coefficienti  $\alpha$  come riportato in Eq. 5, che consente il controllo delle caratteristiche della funzione d'ambiguità.

$$s_F(t) = \alpha^* s_{ref}(t - \tau_i) \quad \text{Eq. 5}$$

dove  $\alpha$  s.t.  $\chi_F(\tau_i, 0) = 0$

Il set di coefficienti è ottenuto in modo tale da annullare i ritardi della mappa Range-Doppler dove sono presenti i picchi indesiderati.

L'ultimo step è l'applicazione di una soglia CFAR sulla mappa ottenuta per rivelare automaticamente i bersagli secondo uno schema specifico CFAR. Per ottenere la

caratteristica CFAR desiderata, può essere utilizzato uno schema standard come la media delle celle (CA) CFAR, la cella con valore massimo (GO) CFAR oppure la cella con valore minimo (SO) CFAR; questi schemi sono ben noti in letteratura per i sistemi radar standard.

### 1.3 Modello del segnale ricevuto dal sistema PBR

L'involuppo complesso del segnale totale ricevuto dall'antenna di sorveglianza può essere scritto come segue:

$$s_{surv}(t) = A_{surv} d(t) + \sum_{m=1}^{N_T} a_m d(t - \tau_m) e^{j2\pi f_{dm} t} + \sum_{i=1}^{N_C} c_i d(t - \tau_{ci}) + n_{surv}(t) \quad 0 \leq t < T_0 \quad \text{Eq. 6}$$

Dove  $T_0$  rappresenta il tempo di osservazione del segnale mentre

- $d(t)$  indica l'involuppo complesso del segnale diretto. In questo caso questa quantità rappresenta una replica ritarda della forma d'onda d'opportunità scelta;
- $A_{surv}$  indica l'ampiezza complessa del segnale diretto ricevuto dai lobi laterali dell'antenna di sorveglianza;
- $a_m$ ,  $f_{dm}$  e  $\tau_m$  rappresentano l'ampiezza complessa, la frequenza doppler e il ritardo dell' $m^{\text{th}}$  target, con  $m = 1, 2, \dots, N_T$ , presente nella scena sott'osservazione rispetto al segnale diretto;
- $c_i$  e  $\tau_{ci}$  sono l'ampiezza complessa e il ritardo, rispetto al segnale diretto, dell' $i^{\text{th}}$  scatteratore stazionario con cui viene modellato il clutter e il multipath, con  $i = 1, 2, \dots, N_C$  (i contributi di clutter e multipath possono essere modellati come un insieme discreto di piccoli scatteratori. Un ambiente in cui è presente un clutter continuo può essere modellato attraverso un gran numero di questi piccoli scatteratori).
- $n_{surv}(t)$  è il contributo di rumore termico nel canale di sorveglianza.

Inoltre, assumendo che il segnale di riferimento sia affetto da  $N_M$  repliche di multipath, il suo involuppo complesso può essere scritto come segue:

$$s_{ref}(t) = A_{ref} d(t) + \sum_{m=1}^{N_M} A_m d(t - \tau_m) + n_{ref}(t) \quad 0 \leq t < T_0 \quad \text{Eq. 7}$$

Dove  $A_{ref}$  è l'ampiezza complessa del segnale diretto,  $A_m$  e  $\tau_m$  sono l'ampiezza complessa e il ritardo rispetto al segnale diretto dell' $m^{\text{th}}$  replica multipath considerata, con  $m = 1, 2, \dots, N_M$ , e  $n_{ref}(t)$  rappresenta il contributo di rumore termico del canale di riferimento.

Ipotizzando, almeno in questo canale, un buon rapporto segnale a rumore, ipotesi verosimile perchè il segnale diretto viene ricevuto nel lobo principale dell'antenna, i contributi dovuti al clutter e ai vari target che entrano nel canale di riferimento possono essere trascurati.

## 2 Descrizione dei segnali d'opportunità

Con riferimento al discorso introduttivo, in questo capitolo si descriveranno dettagliatamente le forme d'onda digitali scelte per lo sviluppo del sistema Radar Passivo. In particolare, nel paragrafo 2.1 verranno descritte le caratteristiche del segnale DVB-T con riferimento allo standard di trasmissione, [54]. Nel paragrafo 2.2 si riportano le caratteristiche fondamentali della rete dei trasmettitori che emettono il segnale digitale televisivo descrivendo la differenza tra reti multi-frequenza e reti singola-frequenza. Nel paragrafo 2.3 viene mostrato lo schema di acquisizione del PBR che sfrutta il segnale DVB-T; le registrazioni dei dati reali sono state utilizzate per verificare il funzionamento di ogni singolo algoritmo studiato e sviluppato. Nel paragrafo 2.4 si descrive il segnale DVB-SH con riferimento allo standard di trasmissione e infine, nel paragrafo 2.5, viene descritto il sistema di acquisizione utilizzato per questo tipo di segnale con riferimento allo specifico trasmettitore satellitare utilizzato (satellite geostazionario Eutelsat W2A).

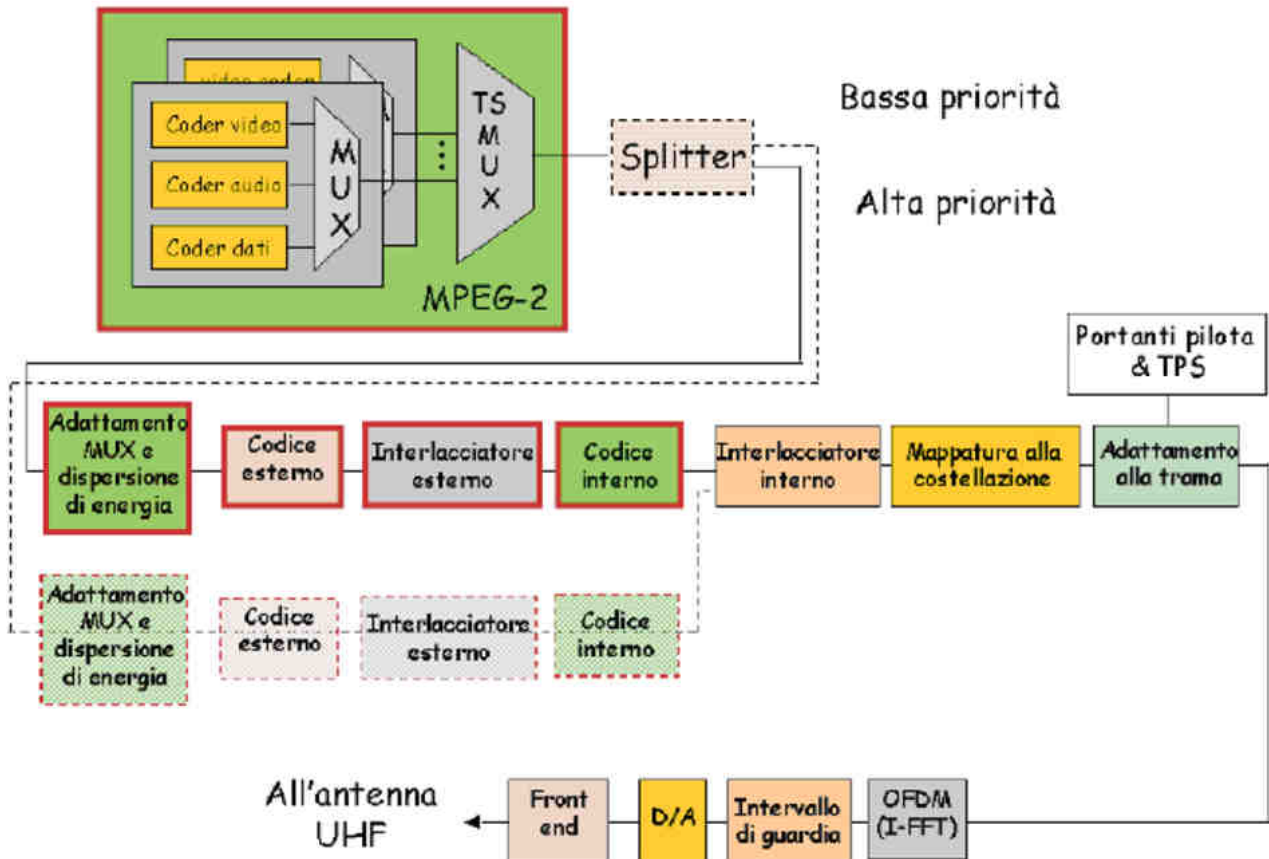
### 2.1 Le caratteristiche del segnale DVB-T

Il sistema DVB-T, la cui architettura è mostrata in Figura 4, adotta lo standard MPEG-2 per la codifica del segnale Audio/Video di sorgente e per la Multiplazione.

Gli elementi che compongono lo schema a blocchi di Figura 4 sono rispettivamente

- La struttura di trama derivata dal multiplatore di trasporto MPEG-2;
- La dispersione dell'energia del segnale per uniformare la distribuzione spettrale all'interno del canale RF;
- Una tecnica di correzione d'errore composta dalla concatenazione di un codice esterno con un codice interno a tasso di codifica variabile mediante processo di interlacciamento; il codice esterno è un Reed Salomon Accorciato ( $RS(204,188,t=8)$ ) derivato dal codice originale sistematico  $RS(255,239,t=8)$  mentre i codici interni sono convoluzionali punturati, basati su un codice convoluzionale madre a tasso 1/2 con 64 stati. Oltre al codice madre, il sistema permette i tassi punturati di 2/3, 3/4, 5/6 and 7/8.
- L'Interleaving, basato sul processo di interleaving convoluzionale di Forney, ha profondità pari a 12.
- Mapping e Modulazione in funzione del tipo di costellazione scelta;

- La trasmissione OFDM con l'inserimento dell'intervallo di guardia.



**Figura 4 – Schema a blocchi funzionale del sistema di trasmissione DVB-T**

Il sistema DVB-T è caratterizzato da due modalità operative, la prima valuta la IFFT su 2048 (2k) portanti ed è utilizzata per reti convenzionali multi-frequenza (MFN), la seconda valuta la IFFT su 8192 (8k) portanti per coprire anche reti a frequenza singola (SFN).

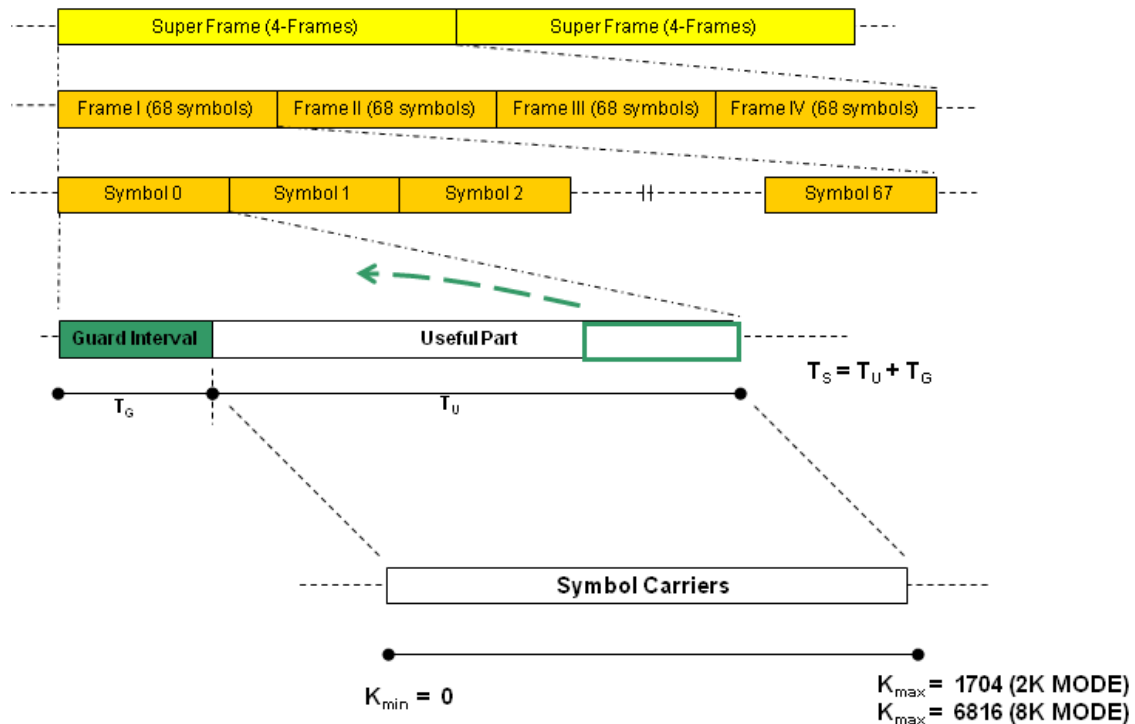
Inoltre il sistema permette diversi livelli di Modulazione QAM (QPSK, 16-QAM e 64-QAM), diversi tassi del codice interno (come accennato in precedenza) e due diverse codifiche e modulazioni di canale per garantire diverse priorità di due generici flussi binari da trasmettere.

### 2.1.1 Struttura del Frame OFDM

La Figura 5 mostra schematicamente l'organizzazione del segnale DVB-T. Il segnale trasmesso è organizzato in frames. Ogni Frame ha una durata  $T_F$ , ed è composto da 68 simboli OFDM identificati da 0 a 67. Quattro Frames costituiscono un Super-Frame.



Ogni simbolo OFDM è trasmesso con una durata  $T_S$  e si compone di due parti: una parte utile di durata  $T_U$  e un intervallo di guardia di durata  $T_G$ . La durata della parte utile differisce rispetto al tipo di modalità di trasmissione utilizzato; 896  $\mu\text{s}$  nel caso di trasmissione 8K, 224  $\mu\text{s}$  in modalità 2K. L'intervallo di guardia è ottenuto copiando parte dei campioni appartenenti alla parte utile. Possono essere utilizzati quattro diversi valori di  $T_G$  tutti proporzionali a  $T_U$  (1/4, 1/8, 1/16 e 1/32 di  $T_U$ ).



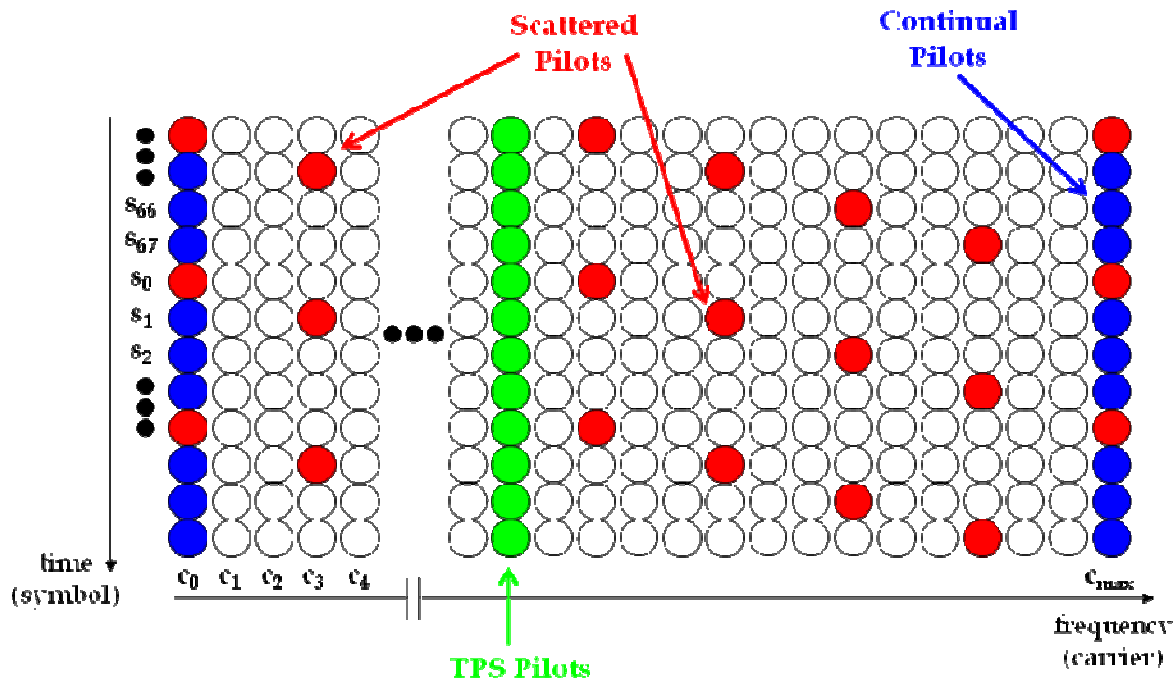
**Figura 5 – Struttura del segnale OFDM**

La parte utile di ogni simbolo OFDM è composta da un set di portanti dedicate al trasporto dei dati e, in aggiunta, contiene altre portanti, dedicate al trasporto dei piloti sparsi, piloti continui e Transmission Parameter Signalling (TPS). Queste strutture, che verranno dettagliatamente descritte nel seguito, sono segnali di riferimento noti al ricevitore e sono usate per la sincronizzazione di Frame, di frequenza e tempo e inoltre per la stima del canale e l'identificazione della modalità di trasmissione.

La Figura 6 mostra visivamente l'allocazione di tutte le portanti in accordo alla struttura del segnale DVB-T; in ascissa troviamo l'asse delle frequenze, mentre sulle ordinate abbiamo il tempo che scorre identificato dal numero del simbolo trasmesso.

In rosso sono identificati gli indici delle portanti denominate pilota sparse, mentre in blu sono rappresentate le pilota continue. In verde abbiamo gli indici riferiti alle portanti

dedicate al trasporto del messaggio TPS e, infine, in bianco abbiamo tutte le portanti dedicate al trasporto dei dati.



**Figura 6 – Struttura del Frame del segnale DVB-T**

La Figura 6 ci consente di visualizzare con facilità le periodicità legate alla posizione di tutte le portanti all'interno del simbolo OFDM. Infatti dalla figura è possibile notare che alcune portanti pilota sparse coincidono con parte delle pilota continue ogni 4 simboli OFDM e che le tutte le posizioni e i differenti contenuti delle pilota sparse, nel dominio della frequenza, si ripetono ogni 4 simboli OFDM nel tempo; inoltre, nel generico simbolo OFDM, il contenuto delle pilota sparse è randomizzata attraverso la sequenza PBRS, ma la posizione, identificata dall'indice di portante, è equidistante e pari a 12 sotto-portanti.

Il numero di portanti dedicate al trasporto dei dati, dei piloti continui, piloti sparsi e TPS è fisso da un simbolo al successivo. Questo numero varia in relazione alla modalità di trasmissione utilizzata. I valori numerici sono riportati nella Tabella 1.

I valori legati al tempo sono dati come multipli interi del periodo elementare  $T$ . Il periodo elementare rappresenta l'inverso della frequenza di campionamento e dipende dalla larghezza di banda del canale considerato.

Di seguito si riportano le diverse scelte di canale con i relativi periodi elementari considerati.

- Canale 8 MHz con periodo elementare di  $7/64 \mu\text{s}$

- Canale 7 MHz con periodo elementare di 1/8  $\mu\text{s}$
- Canale 6 MHz con periodo elementare di 7/48  $\mu\text{s}$
- Canale 5 MHz con periodo elementare di 7/40  $\mu\text{s}$
- Canale 1,7 MHz con periodo elementare di 7/12,8  $\mu\text{s}$

**Tabella 1 – Numero di portanti e relativo contenuto per ogni simbolo OFDM**

Modalità di trasmissione	Portanti totali	Portanti Piloti continui	Portanti Piloti sparsi	Portanti TPS	Portanti dati
8k	6817	177	524	68	6048
2k	1705	45	131	17	1512

La larghezza di banda teorica utile può essere calcolata moltiplicando l'inverso della durata in tempo della parte utile e il numero totale di portanti effettivamente utilizzate nel singolo simbolo OFDM.

Nello specifico, nella Tabella 2, riportiamo il calcolo della larghezza di banda del segnale DVB-T per il canale 8 MHz.

**Tabella 2 – Calcolo teorico della larghezza di banda utile per il canale 8 MHz**

Modalità di trasmissione	Portanti totali	Campioni parte utile	Durata parte utile [ $\mu\text{s}$ ]	Larghezza di banda singola portante [Hz]	Larghezza di banda [Hz]
8k	6817	8192	896	1116*	7608259*
2k	1705	2048	224	4464*	7611607*

N.B: I valori identificati con "\*" sono approssimati all' hertz.

L'espressione utilizzata per il calcolo della larghezza di banda è riportata in Eq. 8.

$$BW_{\text{utile}} = \frac{(N_{\text{carriers}} - 1)}{T_u} = \frac{(N_{\text{carriers}} - 1)}{N_u \cdot T_e} \quad \text{Eq. 8}$$

All'interno dell'espressione si ha:

- $N_{\text{carriers}}$  rappresenta il numero di portanti totali presenti all'interno del simbolo OFDM;
- $N_U$  rappresenta il numero di campioni presenti all'interno della parte utile del simbolo OFDM;
- $T_e$  rappresenta il tempo elementare suggerito dallo standard.

### 2.1.2 Caratteristiche dello spettro del segnale DVB-T

I simboli OFDM sono costituiti da una giustapposizione di portanti ortogonali equispaziate. L'ampiezza e la fase delle portanti dedicate al trasporto dei dati variano da simbolo a simbolo in accordo al processo di mapping, che verrà descritto nel seguito. Lo spettro di densità di potenza  $P_k(f)$ , di ogni singola sotto-portante è definito attraverso la seguente espressione:

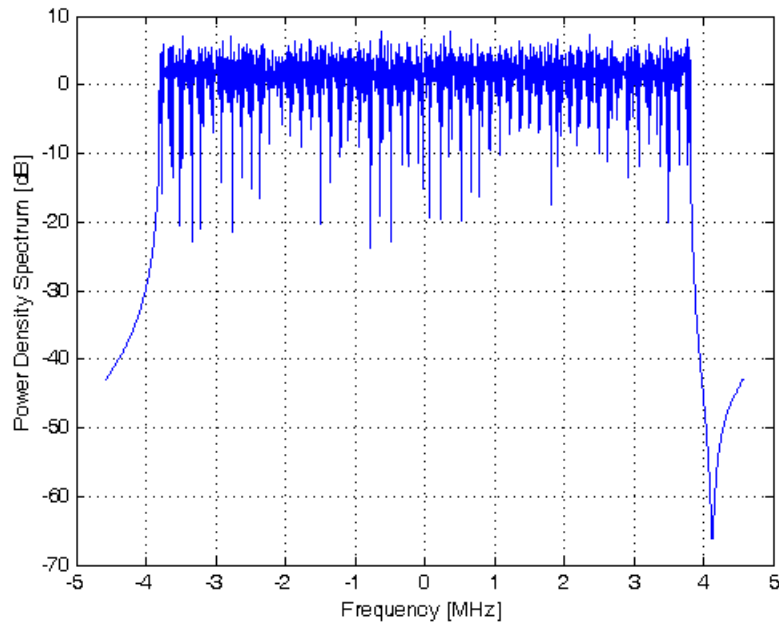
$$P_k(f) = \left[ \frac{\sin \pi(f - f_k)T_S}{\pi(f - f_k)T_S} \right]^2 \quad f_k = f_c + \frac{k}{T_U}; \left( -\frac{k-1}{2} \leq k \leq \frac{k-1}{2} \right) \quad \text{Eq. 9}$$

dove le sotto-portanti  $f_k$  sono definite rispetto alla modalità di trasmissione utilizzata,  $T_S$  rappresenta la durata del simbolo OFDM mentre  $T_U$  è la durata della parte utile del simbolo OFDM.

Lo spettro di densità di potenza delle portanti dati modulate è rappresentato dalla somma di tutti i singoli contributi. In Figura 7 è raffigurato lo spettro di densità di potenza teorico del segnale DVB-T trasmesso, considerando una canale da 8 MHz; lo spettro di densità di potenza non è costante all'interno della banda nominale, valutata attraverso la relazione Eq. 8, perché la durata del simbolo OFDM è più grande della sola parte utile che determina la spaziatura tra le portanti quindi i lobi di ogni singola sotto-portante sono più stretti.

Infine, le frequenze centrali nominali del segnale a RF, utilizzate nel canale UHF, possono essere ricavate dall'espressione:  $470 \text{ MHz} + 4 \text{ MHz} + i_1 \times 8 \text{ MHz}$ ,  $i_1 = 0, 1, 2, \dots$

In alcuni casi, questi centri di Frequenza potrebbero subire delle piccole variazioni per migliorare la condivisione della risorsa spettrale.



**Figura 7 – Spettro di Densità di Potenza del segnale DVB-T (Canale 8 MHz, intervallo di guardia 1/32)**

### 2.1.3 Costellazioni del segnale

Il sistema utilizza una trasmissione OFDM ( Orthogonal Frequency Division Multiplex). Tutte le portanti dati, contenute in una Frame OFDM possono essere modulate usando le costellazioni QPSK, 16-QAM e 64-QAM con mappatura di Gray uniformi e non uniformi. L'esatta proporzione tra le costellazioni uniformi e non uniformi dipende dal parametro alpha, che può assumere valori 1, 2 e 4, e garantisce una diversa priorità ai flussi dati.

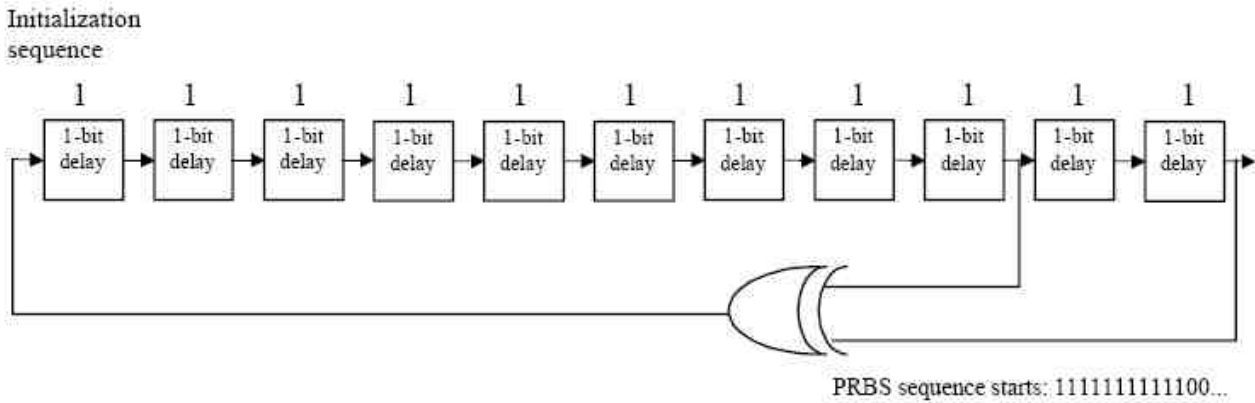
La sequenza binaria, corrispondente al flusso Audio/Video, è organizzata in gruppi da  $\eta$  bits pari a 2, 4 e 6 in base alla costellazione scelta. Ogni gruppo di bits genera un valore complesso in accordo a tale costellazione.

I diversi valori dei simboli appartenenti ai dati sono normalizzati per produrre  $E[zz^*] = 1$ , dove  $z$  rappresenta il punto della costellazione. I fattori di normalizzazione sono a  $1/\sqrt{2}$ ,  $1/\sqrt{10}$  e  $1/\sqrt{42}$  nel caso di costellazione QPSK, 16-QAM e 64-QAM rispettivamente ( oppure  $1/\sqrt{2}$ ,  $1/\sqrt{20}$  e  $1/\sqrt{60}$  nel caso di costellazione non uniforme con alpha pari a 2 oppure  $1/\sqrt{2}$ ,  $1/\sqrt{52}$  e  $1/\sqrt{108}$  con alpha pari a 4).

### 2.1.4 Segnali di riferimento

I piloti continui, i piloti sparsi e i TPS (Transmission Parameter Signalling), contenuti all'interno di un generico simbolo OFDM, sono modulati in accordo alla sequenza binaria

Pseudo Random (PRBS)  $w_k$ , dove  $k$  corrisponde all'indice di portante dedicata a tale scopo e identificata dallo standard stesso. Il polinomio generatore della suddetta sequenza è  $X^{11}+X^2+1$ . In Figura 8 si riporta la schema che consente la generazione della sequenza utilizzata come riferimento.



**Figura 8 – Generazione della sequenza PRBS**

### 2.1.4.1 Piloti continui

I piloti continui sono trasmessi in corrispondenza delle stesse sotto-portanti tra un simbolo OFDM e il successivo. Gli indici  $k$  delle posizioni sono definiti all'interno dello standard e sono riportati in Tabella 3.

**Tabella 3 – Indici relativi alle sotto-portanti dedicate al trasporto dei piloti continui**

2K MODE							8K MODE						
0	48	54	87	141	156	192	0	48	54	87	141	156	192
201	255	279	282	333	432	450	201	255	279	282	333	432	450
483	525	531	618	636	714	759	483	525	531	618	636	714	759
765	780	804	873	888	918	939	765	780	804	873	888	918	939
942	969	984	1050	1101	1107		942	969	984	1050	1101	1107	
1110	1137	1140	1146	1206	1269	1323	1110	1137	1140	1146	1206	1269	1323
1377	1491	1683	1704				1377	1491	1683	1704	1752	1758	1791
							1845	1860	1896	1905	1959	1983	1986
							2037	2136	2154	2187	2229	2235	2322
							2340	2418	2463	2469	2484	2508	2577
							2592	2622	2643	2646	2673	2688	2754
							2805	2811	2814	2841	2844	2850	2910

2973	3027	3081	3195	3387	3408	3456
3462	3495	3549	3564	3600	3609	3663
3687	3690	3741	3840	3858	3891	3933
3939	4026	4044	4122	4167	4173	4188
4212	4281	4296	4326	4347	4350	4377
4392	4458	4509	4515	4518	4545	4548
4554	4614	4677	4731	4785	4899	5091
5112	5160	5166	5199	5253	5268	5304
5313	5367	5391	5394	5445	5544	5562
5595	5637	5643	5730	5748	5826	5871
5877	5892	5916	5985	6000	6030	6051
6054	6081	6096	6162	6213	6219	6222
6249	6252	6258	6318	6381	6435	6489
6603	6795	6816				

I piloti continui sono trasmessi con un livello di potenza più alto rispetto ai dati, mentre la modulazione corrispondente è la seguente:

La parte reale del simbolo trasmesso sulla  $k$ -esima sotto-portante è ottenuta nel modo seguente

$$\operatorname{Re}\{c_{m,l,k}\} = \frac{4}{3}(1 - 2w_k) \quad \text{Eq. 10}$$

mentre la parte Immaginaria è

$$\operatorname{Im}\{c_{m,l,k}\} = 0 \quad \text{Eq. 11}$$

dove  $m$  rappresenta l'indice del frame,  $k$  rappresenta l'indice di frequenza delle sotto-portanti ed  $l$  è l'indice temporale dei simboli. L'ampiezza di questi piloti è tale da ottenere  $E[cc^*] = 16/9$ .

#### 2.1.4.2 Piloti Sparsi

I piloti sparsi sono trasmessi in corrispondenza delle stesse sotto-portanti ogni quattro simboli OFDM. Gli indici  $k$  delle posizioni possono essere ricavati dalla seguente relazione:

$$k = K_{MIN} + 3 \cdot (l \bmod 4) + 12 \cdot p \quad \text{Eq. 12}$$

dove  $p$  rappresenta un numero intero maggiore di zero tale che  $k$  appartenga all'intervallo di valori  $[K_{min}, K_{max}]$ ;  $l$  rappresenta l'indice temporale del simbolo interno ad un Frame (compreso tra 0 e 67).

I piloti sparsi, come quelli continui, sono trasmessi con livello di potenza più alto rispetto ai dati e la modulazione corrisponde a quella utilizzata per i piloti continui in Eq. 10 e Eq. 11.

### 2.1.4.3 Trasmission Parameter Signalling (TPS)

Le sotto-portanti dedicate al trasporto dei TPS sono trasmesse sulle stesse portanti da un simbolo al successivo in accordo allo standard (come riportato in Tabella 4).

**Tabella 4 – Indici delle sotto-portanti dedicate al trasporto dei TPS**

2K MODE							8K MODE						
34	50	209	346	413	569	595	34	50	209	346	413	569	595
688	790	901	1073	1219	1262		688	790	901	1073	1219	1262	
1286	1469	1594	1687				1286	1469	1594	1687	1738	1754	
							1913	2050	2117	2273	2299	2392	
							2494	2605	2777	2923	2966	2990	3173
							3298	3391	3442	3458	3617	3754	3821
							3977	4003	4096	4198	4309	4481	4627
							4670	4694	4877	5002	5095	5146	5162
							5321	5458	5525	5681	5707	5800	5902
							6013	6185	6331	6374	6398	6581	6706
							6799						

Il messaggio TPS è composto da 68 bit ed è definito su 68 simboli OFDM consecutivi (un Frame). Ogni simbolo OFDM, in parallelo su tutti gli indici delle portanti dedicati al trasporto di questo messaggio, trasporta un bit di informazione. In Tabella 5 si riporta il formato del messaggio TPS.

Il primo bit del messaggio serve all'inizializzazione per la modulazione 2-PSK. La modulazione di questo bit viene derivata dalla stessa sequenza PRBS descritta nel paragrafo 2.1.4, in accordo alle relazioni Eq. 10 e Eq. 11.



Tabella 5 – Formato del messaggio relativo ai TPS

Numero del bit	Descrizione contenuto
$S_0$	Bit di inizializzazione
Da $S_1$ a $S_{16}$	Parola di sincronizzazione (0011010111101110 oppure 1100101000010001)
Da $S_{17}$ a $S_{22}$	Indicatori di lunghezza
$S_{23}, S_{24}$	Indicatore del Frame
$S_{25}, S_{26}$	Costellazione scelta
$S_{27}, S_{28}, S_{29}$	Informazioni sulla gerarchia
$S_{30}, S_{31}, S_{32}, S_{33}$	Configurazione codici per flusso HP/LP e Interleaver
$S_{34}$	Configurazione Interleaver
$S_{35}$	Identificatore Super-Frame
$S_{36}, S_{37}$	Identificatore Intervallo di Guardia
$S_{38}, S_{39}$	Modalità di trasmissione
Da $S_{40}$ a $S_{47}$	Identificativo di cella
Da $S_{48}$ a $S_{52}$	Configurazione Interleaver
$S_{53}$	Modalità DVB-SH
Da $S_{54}$ a $S_{67}$	Errore di protezione

I bit sulle stesse portanti sono modulati DBPSK. La modulazione DBPSK è inizializzata ad ogni blocco TPS. I bit relativi a simboli OFDM diversi seguono la seguente regola:

- Se il simbolo  $S_l$  è pari a 0 allora si ha  $\text{Re}\{c_{m, l, k}\} = \text{Re}\{c_{m, l-1, k}\}$ ;  $\text{IM}\{c_{m, l, k}\} = 0$ ;
- Se il simbolo  $S_l$  è pari a 1 allora si ha  $\text{Re}\{c_{m, l, k}\} = -\text{Re}\{c_{m, l-1, k}\}$ ;  $\text{IM}\{c_{m, l, k}\} = 0$ ;

L'ampiezza di questi piloti è tale da ottenere  $E[cc^*]=1$ .

### 2.1.5 Esempio mappatura portanti

Riportiamo un esempio di mappatura dei dati al solo scopo di chiarire ulteriormente tutte le componenti del segnale. In Figura 9 si riporta un esempio di mappatura. In particolare è visualizzato il mapping dei dati con tutte le portanti pilota (continue e sparse) e le portanti dedicate al trasporto del messaggio TPS; come si nota dalla figura, i dati sono normalizzati rispetto alla costellazione di appartenenza (QPSK nello specifico) mentre i piloti e i TPS

vengono trasmessi con la parte immaginaria nulla e con diverse ampiezze in accordo a quanto detto in precedenza.

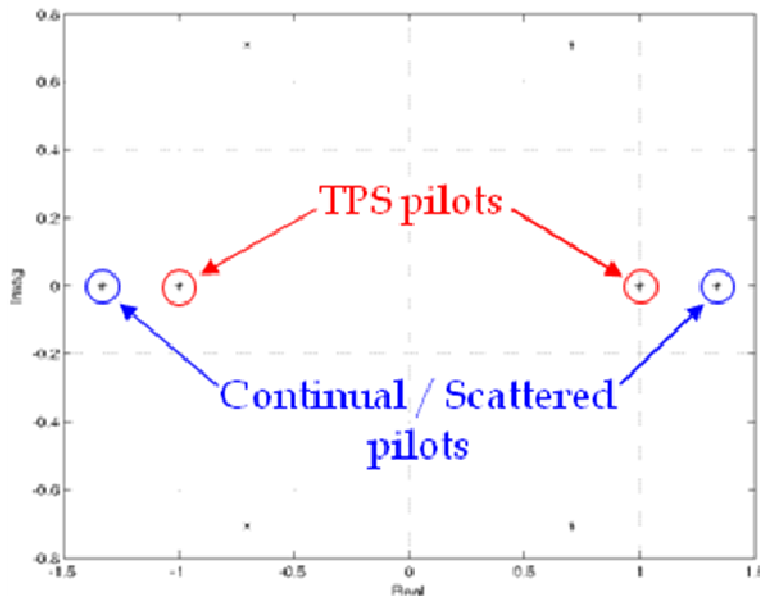


Figura 9 – Mapping dei dati e delle portanti pilota e TPS

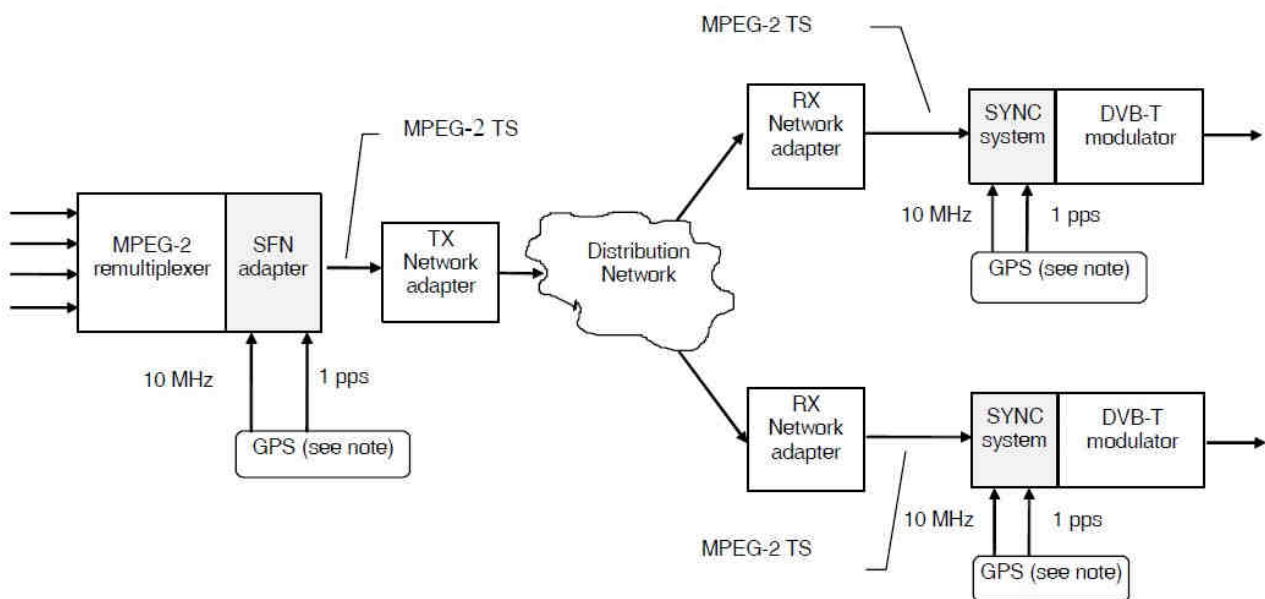
## 2.2 Caratteristiche della rete digitale terrestre

Come è noto da [53], la rete diffusiva DVB-T può essere di due tipi:

- MFN (Multi Frequency Network) prevede che ogni singolo trasmettitore operi con una frequenza diversa rispetto a quelli limitrofi. In questo caso si ha una gestione semplificata della rete e una bassa efficienza delle frequenze in sede di pianificazione;
- SFN (Single Frequency Network) prevede l'utilizzo, da parte di diversi trasmettitori della stessa rete, della stessa frequenza. In questa rete è necessario che il segnale ricevuto da qualsiasi trasmettitore assomigli il più possibile alle eco di segnale ricevuto da qualsiasi altro trasmettitore; di conseguenza, tutti i segnali devono essere sincronizzati rispetto al tempo, alla frequenza e al "bit".

La Figura 10 mostra lo schema a blocchi completo di un sistema SFN. All'interno dello schema è possibile notare due blocchi molto importanti che determinano il reale funzionamento della rete stessa:

- “SFN adapter” che forma un Mega-Frame (numero intero di pacchetti TS corrispondenti a 8 frame DVB-T in modalità 8K oppure 32 frame DVB-T in modalità 2K) inserendo un pacchetto di inizializzazione (MIP (Mega-Frame Initialization Packet)) con un valore specifico di PID (Packet Identifier); questa struttura deve garantire che i generatori di PRBS, che si occupano della dispersione di energia all'interno di ogni canale, vengano sincronizzati rispetto allo stesso istante temporale per garantire lo stesso modo deterministico di dispersione sugli stessi bit.
- “GPS”, o un qualsiasi sistema di riferimento comune, che fornisce una frequenza di riferimento e un tempo di riferimento per effettuare le operazioni necessarie.



Note: Potrebbe essere un qualsiasi sistema di riferimento comune

### Figura 10 – Distribuzione del segnale DVB-T in una rete SFN

Il vantaggio di utilizzare una rete SFN è sicuramente l'alta efficienza di frequenza rispetto alle reti MFN; infatti, con questa tecnica, è possibile servire grandi aree, con lo stesso segnale irradiato con una frequenza comune.

Oltre all'efficienza in frequenza, la tecnica SFN fornisce anche un'efficienza in potenza. Ciò può essere spiegato considerando le forti variazioni locali di intensità di campo di un dato trasmettitore. Nelle reti convenzionali un modo comune per realizzare la continuità del servizio in diversi siti è quello di includere un margine nel link-budget e quindi di aumentare la potenza del singolo trasmettitore in modo significativo. In una rete SFN, in cui il segnale utile è costituito da varie componenti del segnale di diversi trasmettitori le cui variazioni sono solo debolmente correlate, l'intensità di campo che svanisce in una area

potrebbe essere colmata con un altro trasmettitore. Questo effetto in media si risolve in più piccole variazioni dell'intensità del campo totale e quindi, di conseguenza, i trasmettitori SFN possono utilizzare livelli di potenza più bassa.

Inoltre, in una rete SFN tutti i trasmettitori sono modulati in modo sincrono con lo stesso segnale e irradiano sulla stessa frequenza. Sfruttando la capacità di un sistema di trasmissione multi-portante (COFDM), i segnali provenienti da più trasmettitori che si combinano in un ricevitore possono contribuire in modo costruttivo al segnale totale desiderato. Infatti, con tale modulazione (OFDM), il ricevitore riesce a tollerare le eco, di diversi trasmettitori o dell'ambiente propagativo, il cui ritardo è inferiore alla durata dell'intervallo di guardia, accettando un degradamento in termini del margine C/N.

L'effetto limitante della tecnica SFN è l'auto-interferenza della rete stessa; in particolare, se i segnali provenienti da trasmettitori lontani hanno un ritardo maggiore rispetto a quello consentito, pari alla durata dell'intervallo di guardia, si comportano come il rumore interferente, piuttosto che come segnale desiderato.

Una regola empirica, per ridurre con successo l'auto-interferenza a un valore accettabile, è la scelta della durata dell'intervallo di guardia che dovrebbe consentire la propagazione di segnale radio oltre la distanza tra due generici trasmettitori della rete. Un prezzo da pagare utilizzando la tecnica SFN è l'operazione sincrona di tutti i trasmettitori in una data rete.

In particolare,

- La sincronizzazione in frequenza può avvenire se tutti gli oscillatori in cascata all'interno di ogni trasmettitore sono guidati da un oscillatore di riferimento, preferibilmente accessibile a tutti i diversi siti di trasmissione, ad esempio utilizzando il segnale GPS.
- I Sistemi OFDM sono stati progettati per trarre vantaggio dalle eco, fin quando il loro ritardo è inferiore all'intervallo di guardia. Questa condizione richiede la sincronizzazione in tempo di vari e diversi trasmettitori, poiché lo stesso simbolo deve essere emesso nello stesso istante da diverse fonti, qualunque sia il tempo di ritardo introdotto dalla rete di distribuzione. Poiché l'intervallo di guardia deve essere utilizzato per compensare il ritardo del canale e non per compensare le imprecisioni della sincronizzazione di rete, è richiesta una precisione di  $\pm 1 \mu\text{s}$  della sincronizzazione dei trasmettitori.

- La sincronizzazione a livello di bit perché essi devono modulare la stessa sottoportante nello stesso istante. La tolleranza di questa operazione deve essere nulla.
- I flussi MPEG-2 TS devono essere randomizzati in modo identico in tutti i modulatori e quindi ogni generatore PRBS deve essere resettato da un meccanismo deterministico e sincrono, come affermato in precedenza.

### **2.3 Schema di acquisizione utilizzato per i test sul segnale DVB-T**

Lo schema del ricevitore prevede la demodulazione dei segnali di tipo omodina convertendo direttamente il segnale dalla banda a radio frequenza in banda base.

Uno dei principali vantaggi di tale schema, rispetto al ricevitore super-eterodina, è la possibilità di poter selezionare il canale televisivo desiderato variando la frequenza dell'oscillatore locale, senza vincoli stringenti sul filtraggio delle frequenza immagine. Inoltre, campionando il segnale in banda base, è possibile ridurre notevolmente il sample rate del digitalizzatore, anche se sarà necessario disporre di un numero di convertitori A/D doppio (per acquisire separatamente le componenti in fase e quadratura del segnale).

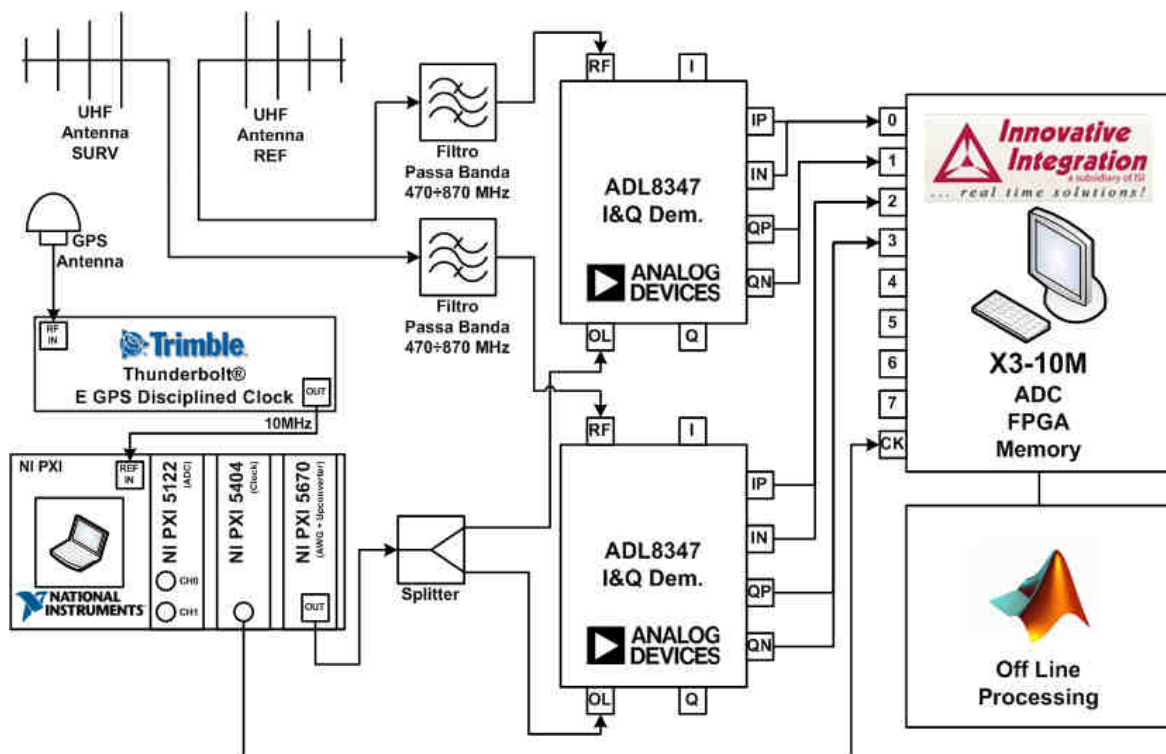
In Figura 11 è rappresentato lo schema di acquisizione dei dati di un sistema radar passivo che sfrutta i segnali DVB-T come illuminatori di opportunità. Il segnale ricevuto tramite ogni singola antenna (REF e SURV) viene opportunamente filtrato prima di essere demodolato. Il filtro in ingresso seleziona la banda UHF (470÷870 MHz) per ridurre i contributi di altri segnali interferenti (in particolare le trasmissioni FM che hanno una potenza superiore rispetto a quelle DVB-T e che causano problemi di intermodulazione nei mixer).

Dopo aver selezionato la sola banda UHF, i segnali vengono inviati in ingresso ai demodulatori. Per i primi test, di sola verifica del sistema, è stato utilizzato il demodulatore AD8347, il cui range di frequenze non copre l'intera banda UHF, tuttavia ha consentito di effettuare acquisizioni dei canali DVB-T con frequenza superiore agli 800 MHz.

Per quanto riguarda il dimostratore vero e proprio, si è scelto di utilizzare un altro demodulatore dell'Analog Devices (ADL5380) il cui range operativo sia adatto alla demodulazione di tutti i segnali della banda UHF; le caratteristiche principali sono elencate di seguito:

- Operating RF and LO frequency: 400 MHz to 6 GHz
- Input IP3: 30 dBm @ 900 MHz
- Input IP2: >65 dBm @ 900 MHz

- Input P1dB (IP1dB): 11.6 dBm @ 900 MHz
- Voltage conversion gain: ~7 dB
- Demodulation bandwidth: ~390 MHz
- Noise figure (NF): 10.9 dB @ 900 MHz
- Quadrature demodulation accuracy @ 900 MHz
- Phase noise: ~-0.07 dB
- Baseband I/Q drive: 2 V p-p into 200  $\Omega$
- Single 5 V supply.



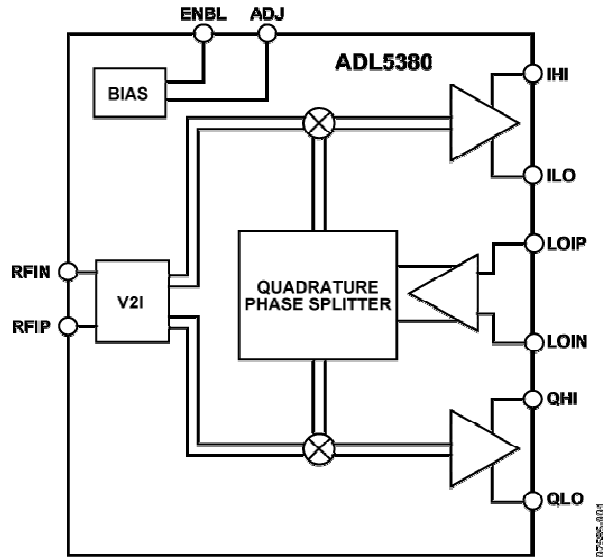
**Figura 11 – Schema di acquisizione dei dati reali**

In Figura 12 è riportato lo schema a blocchi del demodulatore ADL5380. Questo componente è fornito dal costruttore nella versione evaluation board per semplificarne i test, e nel nostro caso l'utilizzo, fornendo ingressi e uscite con connettori SMA e opportune reti di adattamento di impedenza. Per ulteriori approfondimenti, si rimanda alla documentazione tecnica [67].

Il segnale di oscillatore locale e il clock di riferimento per il campionamento, sono stati generati tramite sintetizzatori di forme d'onda della National Instruments, integrati nel sistema modulare PXI. In particolare per l'oscillatore locale è stato utilizzato il Vector Signal Generator NI PXI 5670, le cui caratteristiche sono riportate di seguito:

- 250 kHz to 2.7 GHz

- 20 MHz real-time bandwidth
- -145 to +10 dBm output power
- 16-bit resolution, 100 MS/s AWG (400 MS/s with interpolation)
- High-stability OCXO timebase
- 8, 32, 256, 512 MB memory



**Figura 12 - Schema a blocchi del demodulatore ADL5380 dell'Analog Devices**

Per quanto riguarda il clock per il campionamento, è stato invece utilizzato il Frequency Generator NI PXI 5404 le cui caratteristiche sono riportate di seguito:

- 9 kHz to 100 MHz sine wave generation
- +/- 0.2 dB flatness across sine wave passband (9 kHz to 100 MHz)
- 12-bit amplitude resolution
- DC to 100 MHz clock generation
- 1.07  $\mu$ Hz frequency resolution

Tramite il sistema PXI, è possibile generare entrambi i segnali di oscillatore locale e clock, agganciati in fase, allo stesso riferimento. Come già anticipato è necessario utilizzare come reference del sistema il segnale da 10 MHz agganciato in fase con il riferimento del sistema GPS. Per ricevere il riferimento GPS, è stato utilizzato il Thunderbolt® E GPS Disciplined Clock della Trimble (Figura 13 e dettagli in [69]), che fornisce in uscita un segnale sinusoidale alla frequenza di 10MHz.

Per quanto riguarda la digitalizzazione dei segnali, per il successivo processing (off-line), viene utilizzata la scheda di acquisizione X3-10M della Innovative Integration, con 8 canali fisici (8 ADC a 16 bit) e un processore FPGA integrato che può essere configurato per usi

futuri. Il trasferimento dei dati su disco avviene tramite un Bus PCI express. Gli ingressi sono di tipo bilanciato, adattati alla digitalizzazione di segnali in banda base, con impedenza di 1 M $\Omega$  e banda analogica di 6.5 MHz.



**Figura 13 - Trimble Thunderbolt® E GPS Disciplined Clock**

Si riportano di seguito le caratteristiche principali della scheda X3-10M:

- 8 simultaneously sampling 16-bit, 25 MSPS A/Ds
- Programmable input: +/-2V, +/-1V, +/-0.4V, +/-0.1V
- High impedance, differential inputs
- Xilinx Spartan3A DSP, 1.8/3.4M gate FPGA
- 4MB SRAM
- Programmable Low Jitter PLL timebase
- Framed, software or external triggering
- Log acquisition timing and events
- Power Management features
- PCI Express (VITA 42.3)

La selezione della frequenza di campionamento viene effettuata considerando che il clock non può avere una frequenza superiore a 25 MHz che rappresenta la massima frequenza di campionamento a cui operano gli ADC della scheda X3-10M. Pertanto i valori indicativi individuati sono di 16 MHz e 12.8 MHz; tali valori sono stati scelti perché sottomultipli della frequenza prevista dallo standard DVB-T che è di 64MHz.

## **2.4 Le caratteristiche del segnale DVB-SH**

Il DVB-SH (Digital Video Broadcasting - Satellite services to Handhelds ) è lo standard DVB per la diffusione ibrida [54], satellitare e terrestre, di servizi multimediali verso terminali mobili e portatili. Per garantire la necessaria copertura del servizio è infatti necessaria l'azione complementare di un segmento satellitare (SC), in grado di garantire

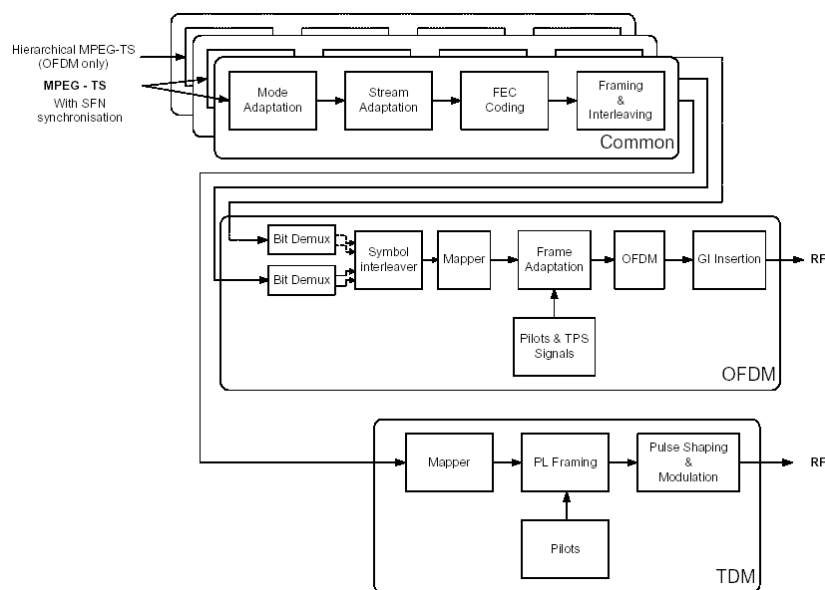


una copertura completa soprattutto nelle zone rurali, e di una rete di ripetitori terrestri (CGC) in grado di mantenere il segnale ai livelli previsti nelle zone urbane e garantire nelle medesime la copertura indoor. La scelta per la modulazione terrestre è Orthogonal Frequency Division Multiplexing (OFDM), mentre la modulazione trasmessa dal satellite può essere di due tipi in funzione delle diverse architetture esistenti.

- Architettura denominata SH-A in cui si trasmette con modulazione OFDM sia da satellite che dai ripetitori a terra.
- Architettura denominata SH-B in cui si trasmette con modulazione OFDM dai ripetitori a terra e modulazione Time Division Multiplex (TDM) da satellite.

Il segnale trasmesso dal satellite dipende dal tipo di architettura a cui tale satellite fa riferimento.

### 2.4.1 Diagramma a blocchi di un trasmettitore DVB-SH



**Figura 14 – Diagramma a blocchi funzionale di un trasmettitore DVB-SH**

La Figura 14 descrive il sistema di trasmissione. Il procedimento, che si compone di una parte comune ad entrambi i modi di trasmissioni (TDM e OFDM) e di una parte dedicata, sarà applicato a uno o più stream di dati in input (MPEG – Transport Streams).

La parte comune comprende:

- *Mode Adaptation* che consiste in una codifica Cyclic Redundant Check (CRC), per proteggere il pacchetto MPEG –TS da errori di ricezione, e nell’inserimento di bits di segnalazione.

- *Stream Adaptation* che consiste in un'aggiunta di bits (PADDING), per ottenere una lunghezza dei dati in uscita costante, e una randomizzazione dell'uscita attraverso una sequenza Pseudo Random Binary Sequence.
- Codifica Forward Error Correction (FEC) utilizzando un turbo codice nello standard 3GPP2.
- Interleaver su un blocco FEC.
- Time Interleaving and framing.

La parte dedicata al trasmissione TDM, a valle della parte comune, comprende:

- Mapping dei bits in relazione a diversi tipi di modulazione.
- Il Framing dello strato fisico del TDM
- L'inserimento dei piloti e l'applicazione del codice di scrambling
- Baseband shaping e Quadrature Modulation.

Mentre la parte dedicata alla trasmissione OFDM comprende

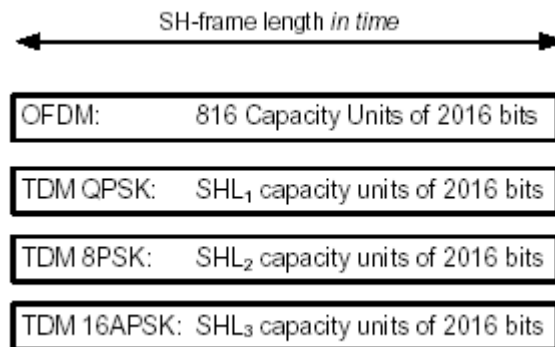
- Un simbol interleaver
- Mapping dei bits in relazione a diversi tipi di modulazione.
- Il Framing dello strato OFDM con l'inserimento dei piloti continui e sparsi, dei parametri di segnalazione della trasmissione e dell'intervallo di guardia.

## 2.4.2 Struttura del Frame SH

L'operazione combinata della singola frequenza, proveniente dal satellite, e dalla Multi Frequenza, proveniente dalla rete terrestre, ha un impatto sui parametri del Frame. Per semplificare la diversità di ricezione di entrambi i segnali in ambiente ibrido TDM/OFDM, la durata in tempo del frame della forma d'onda TDM è stata allineata a quella del OFDM. Siccome ogni singola architettura potrebbe usare larghezze di banda e rapporti di codifica (FEC) diversi, questo porta a differenti symbol rate e quindi a diverso numero di Capacity Units (CUs) nel caso TDM.

Un SH-Frame si compone di un numero intero di CUs, ognuna composta da 2016 bits. La lunghezza in tempo di un SH-Frame è derivata direttamente dai parametri della trasmissione OFDM. La lunghezza in tempo di un SH-Frame relativo a una trasmissione TDM da satellite viene allineata a quella ricavata dai parametri OFDM. In Figura 15 viene mostrato l'allineamento in tempo.

Un SH-Frame relativo alla trasmissione OFDM è composto esattamente da 816 CUs da 2016 Bits ognuna, qualunque sia la modulazione scelta. Il numero di CUs in questo caso è fisso e non dipende né dall'intervallo di guardia né dalla larghezza di banda selezionata. Nel caso di trasmissione TDM, il numero di CUs (SHL1, SHL2 e SHL3 in Figura 15) all'interno di un singolo SH-Frame dipende dall'intervallo di guardia utilizzato dalla modulazione OFDM, dal fattore di Roll off della trasmissione TDM e dalla costellazione utilizzata da entrambe le modulazioni TDM e OFDM.



**Figura 15 – Lunghezza del SH Frame in termini di CUs per TDM e OFDM**

La Tabella 6 riporta il numero di CUs per un SH-Frame in modalità di trasmissione TDM.

**Tabella 6 - Numero di Capacity Units (CUs) per un SH-FRAME in modalità TDM**

OFDM Guard Interval	TDM Rolloff	OFDM: QPSK			OFDM: 16QAM		
		TDM:QPSK	TDM:8PSK	TDM: 16APSK	TDM: PSK	TDM:8PSK	TDM: 16APSK
1/4	0,15	952	1428	1904	476	714	952
1/4	0,25	896	1344	1792	448	672	896
1/4	0,35	812	1218	1624	406	609	812
1/8	0,15	868	1302	1736	434	651	868
1/8	0,25	784	1176	1568	392	588	784
1/8	0,35	728	1092	1456	364	546	728
1/16	0,15	812	1218	1624	406	609	812
1/16	0,25	756	1134	1512	378	567	756
1/16	0,35	700	1050	1400	350	525	700
1/32	0,15	784	1176	1568	392	588	784
1/32	0,25	728	1092	1456	364	546	728
1/32	0,35	672	1008	1344	336	504	672

La lunghezza di un SH-Frame rispetto a un Frame OFDM è legata alla modulazione OFDM selezionata e alla lunghezza della FFT. In Tabella 7 si riporta la corrispondenza tra SH-frame e Frame OFDM.

**Tabella 7 – Corrispondenza tra Frame OFDM e SH-Frame**

Lunghezza FFT	Frame OFDM per ogni SH-Frame (Costellazione QPSK)	Frame OFDM per ogni SH-Frame (Costellazione 16QAM)
1k	16	8
2k	8	4
4k	4	2
8k	2	1

Poiché, come già detto, la durata del Frame TDM in tempo è vincolata alla durata del Frame OFDM si dovrà anche selezionare un appropriato Symbol Rate relativo alla trasmissione TDM, come mostrato nella Tabella 8 dove si riporta il caso in cui la larghezza di banda nominale del segnale è pari a 5 MHz.

**Tabella 8 - Symbol rate relativo alla trasmissione TDM**

Signal Bandwidth [MHz]	OFDM Sampling Frequency [MHz]	OFDM Guard Interval	TDM Symbol rate [MHz]	TDM Rolloff	TDM Symbol rate [MHz]	TDM Rolloff	TDM Symbol rate [MHz]	TDM Rolloff
5,00	40/7	1/4	17/4	0,15	4/1	0,25	29/8	0,35
		1/8	155/36		35/9		65/18	
		1/16	145/34		135/34		125/34	
		1/32	140/33		130/33		40/11	

I valori della Tabella 8 sono ottenibili risolvendo la Eq. 13;

$$S_{r\_TDM} = \text{int} \left\{ 32 \left( \frac{1+GI}{1+\alpha} \right) \right\} \frac{BW_{TDM}}{32(1+GI)} \quad \text{Eq. 13}$$

dove

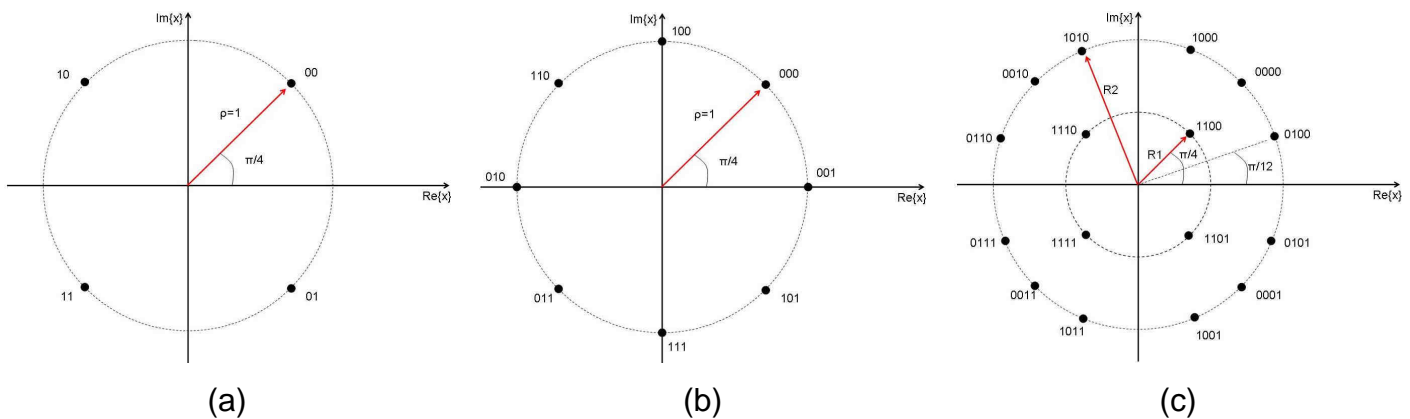
- $\alpha$  rappresenta il fattore di roll off del coseno rialzato (0.15, 0.25 e 0.35)
- GI rappresenta l'intervallo di guardia (1/4, 1/8, 1/16 e 1/32)
- $BW_{TDM}$  rappresenta la larghezza di banda del segnale TDM pari a quella del segnale OFDM (nel caso specifico 5 MHz).

La logica di questa equazione è che il Symbol Rate TDM potrebbe essere semplicemente ricavato dal periodo elementare OFDM T ( $T = 7/(8 \cdot BW_{\text{OFDM}})$ ).

## 2.4.3 Mappatura dei bit nei casi TDM e/o OFDM

### 2.4.3.1 TDM

Le costellazioni proposte nel caso TDM sono QPSK, 8PSK e 16APSK. La sequenza binaria, in uscita dalla parte comune dello schema di trasmissione, è organizzata in gruppi da  $\eta$  bits pari a 2, 3 o 4 in funzione della costellazione scelta. Ogni gruppo di bits genera un valore complesso in formato (I,Q) dove I è la componente in fase e Q la componente in quadratura.



**Figura 16 – Mappatura dei bits nelle costellazioni QPSK (a), 8PSK (b) e 16APSK (c) rispettivamente**

La Figura 16 (a) mostra la mappatura dei bit nella costellazione QPSK; in questo caso specifico 2 bit sono mappati in un simbolo QPSK e l'energia media normalizzata per simbolo sarà uguale a  $\rho^2=1$ .

La Figura 16 (b) mostra la mappatura dei bit nella costellazione 8PSK; in questo caso specifico 3 bit sono mappati in un simbolo 8PSK e l'energia media normalizzata per simbolo sarà uguale ancora a  $\rho^2=1$ .

La Figura 16 (c) mostra mappatura dei bit nella costellazione 16APSK; in questo caso specifico 4 bit sono mappati in un simbolo 16APSK. Questa costellazione è composta da due cerchi concentrici da 4 e 12 punti PSK uniformemente spazati. Il cerchio interno ha un raggio pari a  $R_1 = 1/\sqrt{7}$  mentre quello esterno pari a  $R_2 = 3/\sqrt{7}$ ; questi valori sono utili per ottenere l'energia media del segnale pari a 1.

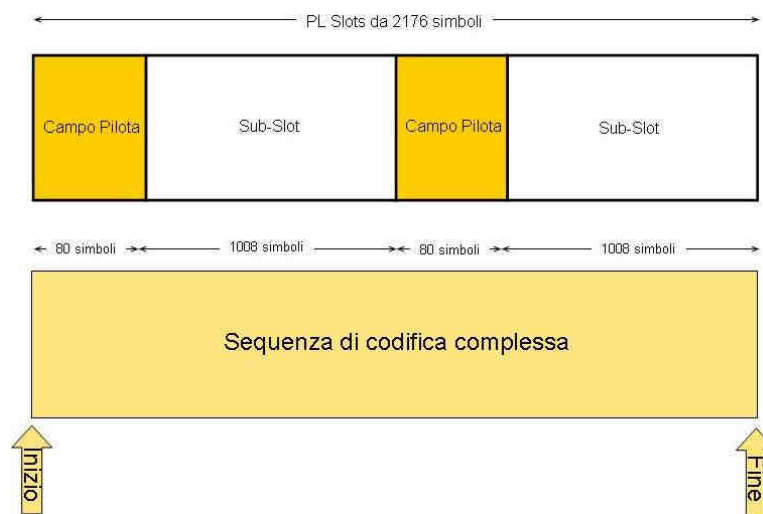
### 2.4.3.2 OFDM

In analogia a quanto accade nel caso di segnale DVB-T, descritto dettagliatamente nel paragrafo 2.1.3, le costellazioni proposte nel caso OFDM DVB-SH sono QPSK o 16QAM (uniforme e non uniforme con il parametro  $\alpha$  pari a 2 o 4). La sequenza binaria è organizzata in gruppi da  $\eta$  bits pari a 2 o 4 in base alla costellazione scelta.

### 2.4.4 Struttura del Frame TDM

Un SH-Frame, in modalità TDM, è formato da un numero di Physical Layer Slots (PL Slots) di lunghezza pari a 2176 simboli. Ogni simbolo del PL Slot può comprendere 2, 3 o 4 CUs da 2016 bits in funzione del formato di costellazione scelto, cioè QPSK, 8PSK o 16APSK rispettivamente.

L'organizzazione del PL Slot è mostrata in Figura 17. Ogni PL Slot contiene due campi pilota che hanno la stessa durata pari a 80 simboli. Ogni simbolo pilota è identificato dalle componenti in fase e in quadratura identiche e pari a  $1/\sqrt{2}$ . Il campo pilota viene inserito prima di ogni Sub-Slot di lunghezza 1008 simboli..



**Figura 17 – Struttura di un Physical Layer Slot**

Il contenuto di ogni PL Slot viene randomizzato attraverso un'unica sequenza di scrambling, di seguito denominato  $C(i)$ , di lunghezza pari a 2176 simboli.

$$C(i) = C_I(i) + jC_Q(i), \quad i = 1, 2, 3 \dots 2176;$$

$$I_{SCR} = I(i)C_I(i) - Q(i)C_Q(i), \quad Q_{SCR} = I(i)C_Q(i) + Q(i)C_I(i), \quad i = 1, 2, 3, \dots 2176;$$

In questo caso specifico è richiesta soltanto una sequenza  $C_I(i) + C_Q(i)$  che viene inizializzata di ogni PL Slot ed è ricavata attraverso la combinazione di due sequenze a massima lunghezza reali, generate attraverso due polinomi generatori da 18 stadi,  $1+x^7+x^{18}$  e  $1+y^5+y^7+y^{10}+y^{18}$  rispettivamente.

Di seguito riportiamo le principali azioni da effettuare per ricavare il codice:

1. Inizialmente poniamo  $x(0)=1$  e gli altri stadi di  $x$  uguali a zero; mentre tutti gli stadi di  $y$  vengono posti pari a 1.
2. Gli stadi delle singole sequenze vengono generati ricorsivamente considerando i polinomi generatori.
3. La sequenza Gold è definita nel modo seguente:  $Z_0(i) = [ x( (i) \bmod (2^{18}-1) ) + y(i) ] \bmod (2)$  con  $i = 0, \dots, 2^{18}-2$ ;
4. La sequenza binaria ottenuta viene convertita in un valore intero attraverso la seguente trasformazione:  $R_0(i) = 2 Z_0( (i+131072) \bmod (2^{18}-1) ) + Z_0(i)$  con  $i = 1, \dots, 2176$ ;
5. Infine, il codice di scrambling da utilizzare si ottiene nel modo seguente:  $C_I(i) + jC_Q(i) = \exp(jR_0(i)\pi/2)$ ;

### 2.4.5 Baseband shaping and quadrature modulation

Il segnale, ottenuto dopo la randomizzazione con il codice di scrambling, viene filtrato attraverso un coseno rialzato. Il fattore di Roll Off può assumere soltanto tre valori, 0.15, 0.25 e 0.35 rispettivamente.

La funzione teorica del filtro in banda base avrà le seguenti espressioni:

$$H(f) = \begin{cases} 1 & |f| < f_N(1-\alpha) \\ \left\{ \frac{1}{2} + \frac{1}{2} \sin \frac{\pi}{2f_N} \left( \frac{f_N - |f|}{\alpha} \right) \right\}^{1/2} & f_N(1-\alpha) < |f| < f_N(1+\alpha) \\ 0 & |f| < f_N(1+\alpha) \end{cases} \quad \text{Eq. 14}$$

Dove  $f_N$  rappresenta la frequenza di Nyquist e  $\alpha$  il fattore di roll off.

La quadrature modulation sarà eseguita moltiplicando i campioni in fase e quadratura attraverso il  $\sin(2\pi f_0 t)$  e  $\cos(2\pi f_0 t)$  rispettivamente (dove  $f_0$  rappresenta la frequenza portante). I due segnali risultanti saranno poi addizionati per ottenere il segnale modulato in uscita.

## 2.4.6 Struttura del Frame OFDM

Le caratteristiche tecniche del Frame OFDM sono identiche a quelle descritte in precedenza relativamente allo standard DVB-T.

La differenza sostanziale è legata alla larghezza di banda nominale utilizzata e al periodo elementare da considerare; in questo caso il canale considerato ha una larghezza di banda pari a 5 MHz con periodo elementare di  $7/40 \mu\text{s}$ .

In Tabella 9 e Tabella 10, si riportano i valori numerici dei parametri OFDM ottenuti considerando le diverse modalità di trasmissione che possono potenzialmente essere considerate.

**Tabella 9 – Valori numerici dei parametri OFDM per tutte le modalità di trasmissione considerando il canale 5 MHz**

Parametri	8k mode	4k mode	2k Mode	1k Mode
Numero di portanti	6817	3409	1705	853
Valore portante minima	0	0	0	0
Valore portante massima	6816	3408	1704	852
Durata parte utile $T_U$ ( $\mu\text{s}$ )	1433,6	716,8	358,40	179,2
Carrier spacing $1/T_U$ (Hz)	698	1395	2790	5580
Banda utile (MHz)	4,75	4,75	4,75	4,75

**Tabella 10 – Durata del simbolo OFDM per diversi intervalli di guardia considerando il canale 5 MHz.**

Intervallo di guardia	1/4	1/8	1/16	1/32
	<b>8k Mode</b>			
Durata parte utile $T_U$	8192 $\times$ T 1433,6 $\mu\text{s}$			
Durata intervallo di guardia $T_G$	2 048 $\times$ T 358,4 $\mu\text{s}$	1024 $\times$ T 179,2 $\mu\text{s}$	512 $\times$ T 89,6 $\mu\text{s}$	256 $\times$ T 44,8 $\mu\text{s}$
Durata del simbolo $T_S$	10 240 $\times$ T 1 792 $\mu\text{s}$	9 216 $\times$ T 1 612,8 $\mu\text{s}$	8 704 $\times$ T 1523,2 $\mu\text{s}$	8 448 $\times$ T 1478,4 $\mu\text{s}$



	<b>4k Mode</b>			
<b>Durata parte utile <math>T_U</math></b>	4096 × T 716,8 μs			
<b>Durata intervallo di guardia <math>T_G</math></b>	1 024 × T 179,2 μs	512 × T 89,6 μs	256 × T 44,8 μs	128 × T 22,4 μs
<b>Durata del simbolo <math>T_S</math></b>	5 120 × T 896 μs	4 608 × T 806,4 μs	4 352 × T 761,6 μs	4 224 × T 739,2 μs
	<b>2k Mode</b>			
<b>Durata parte utile <math>T_U</math></b>	2 048 × T 358.4 μs			
<b>Durata intervallo di guardia <math>T_G</math></b>	512 × T 89,6 μs	256 × T 44,8 μs	128 × T 22,4 μs	64 × T 11,2 μs
<b>Durata del simbolo <math>T_S</math></b>	2 560 × T 448 μs	2 304 × T 403,2 μs	2 176 × T 380,8 μs	2 112 × T 369,6 μs
	<b>1k Mode</b>			
<b>Durata parte utile <math>T_U</math></b>	1 024 × T 179,2 μs			
<b>Durata intervallo di guardia <math>T_G</math></b>	256 × T 44,8 μs	128 × T 22,4 μs	64 × T 11,2 μs	32 × T 5,6 μs
<b>Durata del simbolo <math>T_S</math></b>	1 280 × T 224 μs	1 152 × T 201,6 μs	1 088 × T 190,4 μs	1 056 × T 184,8 μs

## 2.5 Schema di acquisizione utilizzato per i test sul segnale DVB-SH

Dopo un'analisi estensiva delle caratteristiche degli attuali sistemi di trasmissione satellitare sulla base di determinate caratteristiche quali EIRP trasmesso per canale, Banda massima utilizzabile per l'integrazione nell'elaborazione del segnale, tempo massimo disponibile per l'integrazione del segnale e quota del satellite, è stato selezionato il segnale trasmesso dal satellite Eutelsat W2A (orbita LEO) come illuminatore di opportunità per lo sviluppo e il progetto di un radar passivo.

Eutelsat W2A è un satellite geostazionario che trasmette il segnale video broadcast in accordo allo standard DVB-SH descritto precedentemente; l'architettura di riferimento del

satellite selezionato è SH-A che implica la trasmissione del segnale modulato OFDM dal segmento spaziale.

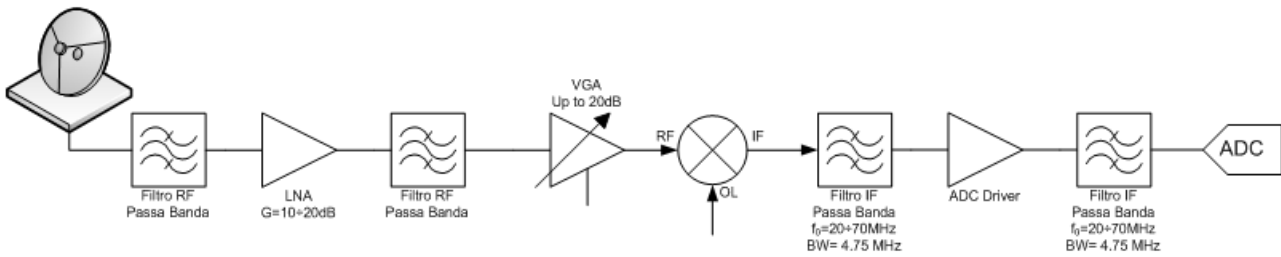
L'Eutelsat communications è uno dei maggiori operatori mondiali nel campo delle trasmissioni satellitari; con una flotta di 27 satelliti, di cui 22 di proprietà esclusiva della compagnia, ed un'ampia rete terrestre, si occupa infatti di trasmissioni video, radio, telefonia e dati in Europa, Africa ed in ampie zone del resto del globo. E' ancora oggi una delle compagnie più solide presenti sul mercato, grazie alla varietà e vastità di servizi offerti ed all'ottima reputazione che le ha garantito contratti con i maggiori network televisivi come ad esempio la News Corporation.

**Tabella 11 – Caratteristiche tecniche relative al satellite W2A**

<b>Piattaforma</b>	Thales Alenia Space Spacebus 4000-C4
<b>Quota</b>	35840 Km
<b>Latitudine</b>	0.01°N
<b>Longitudine</b>	10.05 °E
<b>Copertura</b>	lat:33.5°N - 48.1°N lon:5.3°E – 19.9°E
<b>Data di lancio</b>	3 Aprile 2009
<b>Potenza disponibile</b>	15400 W
<b>EIRP</b>	72 dBW
<b>Uplink</b>	n.a.
<b>Downlink</b>	Frequenze: 2170-2200 MHz 6 canali da 5 MHz modo SH-A in OFDM modo SH-B in TDM
<b>Modulazione</b>	TDM: Qpsk, 8psk, 16Apsk OFDM: Qpsk, 16Qam

Le trasmissioni della rete eutelsat avvengono perlopiù in banda K, ma è da poco stato avviato un progetto per la trasmissione satellitare in banda S sul continente europeo (la trasmissione in banda S è da anni usata sul continente americano).

Le principali caratteristiche tecniche del W2A sono indicate in Tabella 11.



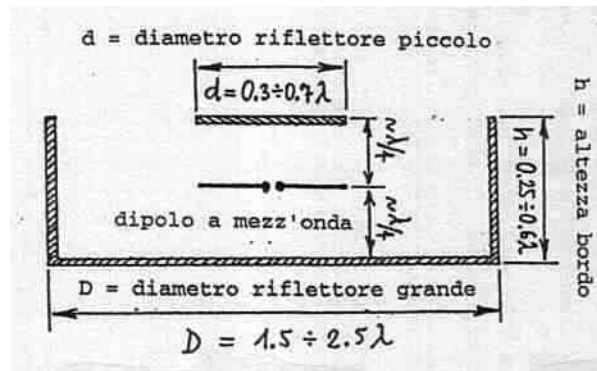
**Figura 18 – Schema a blocchi del sistema ricevente doppio canale**

In Figura 18 è mostrato lo schema a blocchi del sistema ricevente; chiaramente il ricevitore sarà dotato di due canali identici tra loro e con le stesse caratteristiche.

Le principali caratteristiche relative all'antenna utilizzata sono:

- Banda: 2200 MHz;
- Guadagno: 21 dB;
- Larghezza del fascio (-3 dB): 12°;
- Diametro (metri): 1.2;

Il Feed utilizzato è del tipo SBFA (Short Back Fire Antenna), riportato in Figura 19.



**Figura 19 –SBFA utilizzato**

Il Feed è composto da due riflettori: un riflettore piccolo, di diametro pari a  $0,3 \div 0,7\lambda$ , e uno grande di diametro pari a  $1,5 \div 2,5\lambda$ , come mostrato nella figura stessa. Oltre al feed, l'antenna ha in dotazione anche un'ibrida che consente la ricezione di segnali polarizzati sia linearmente che circolarmente.

La sezione a RF si compone di un primo filtraggio poco selettivo e con bassa perdita (per ridurre l'impatto dei segnali trasmessi nelle Bande circostanti), un LNA con guadagno compreso tra 10 e 20 dB, un secondo filtraggio con selezione ottimale della Banda RF, pari a 5 MHz, e un VGA per differenziare il guadagno dei due canali del ricevitore (canale riferimento e sorveglianza).

A valle del VGA troviamo il mixer per la traslazione del segnale intorno alla media frequenza, selezionabile tra i valori di 30 a 70 MHz. All'interno della sezione a IF troviamo due stadi di filtraggio della banda utile di 4.75 MHz e un driver ADC per definire il livello del segnale in ingresso al convertitore.

Il convertitore utilizzato per i test è il dispositivo NI PXI 5122 della National Instrument che ha le seguenti caratteristiche:

- 2 canali;
- 14 bit di risoluzione;
- 0.2V, 0.4V, 1V, 2V, 4V, Vertical Range programmabile;
- 100MHz di banda analogica in ingresso (circa 80MHz con VR 0.2V);
- 100MHz massimo Sample Rate.

L'elaborazione del segnale viene effettuata off-line attraverso l'uso del software Matlab.

### 3 Controllo dei lobi della funzione di ambiguità

Con riferimento agli obiettivi di questo lavoro di tesi descritti nel paragrafo 1.1, in questo capitolo si descrive un nuovo approccio proposto ed implementato per il controllo delle caratteristiche della funzione d'ambiguità che consente di risolvere il problema della presenza di picchi spuri all'interno della funzione d'ambiguità. Nel dettaglio, nel paragrafo 3.1 si riporta la descrizione delle tecniche presenti in letteratura che consentono di risolvere questa problematica evidenziando le peculiarità e gli svantaggi introdotti dal loro utilizzo. Il paragrafo 3.2 descrive le tecniche per stimare il sincronismo del segnale DVB utili per allineare il segnale ricevuto a quello trasmesso rispetto alla struttura definita dallo standard; queste tecniche sono indispensabili per l'implementazione delle tecniche per il controllo lobi laterali presenti in letteratura e verranno utilizzate anche per l'equalizzazione del segnale. Il paragrafo 3.3 riporta la descrizione analitica che consente di ricavare i coefficienti con cui pesare il segnale di riferimento. I paragrafi 3.4 e 3.5 riportavano il confronto prestazionale, tra l'approccio proposto e quello presente in letteratura, e i risultati ottenuti applicando le gli algoritmi di elaborazioni studiati su un set di dati reali.

#### 3.1 Stato dell'arte

È noto che un PBR opera nel piano Tempo/Frequenza attraverso la valutazione della funzione d'ambiguità tra il segnale di riferimento e quello di sorveglianza.

Questa operazione corrisponde all'implementazione di un banco di filtri adattato, ognuno sintonizzato su una differente velocità bistatica del bersaglio. Di solito, i valori delle velocità sono scelti in relazione alla risoluzione bistatica di velocità e quindi l'insieme di filtri scelto copre tutte le possibili velocità bistatiche del bersaglio. Il filtro, dove il bersaglio viene rivelato, fornisce la stima della frequenza Doppler relativa al bersaglio stesso. In modo analogo, viene effettuata la stima del range bistatico attraverso il campione in range dove viene rivelato il bersaglio.

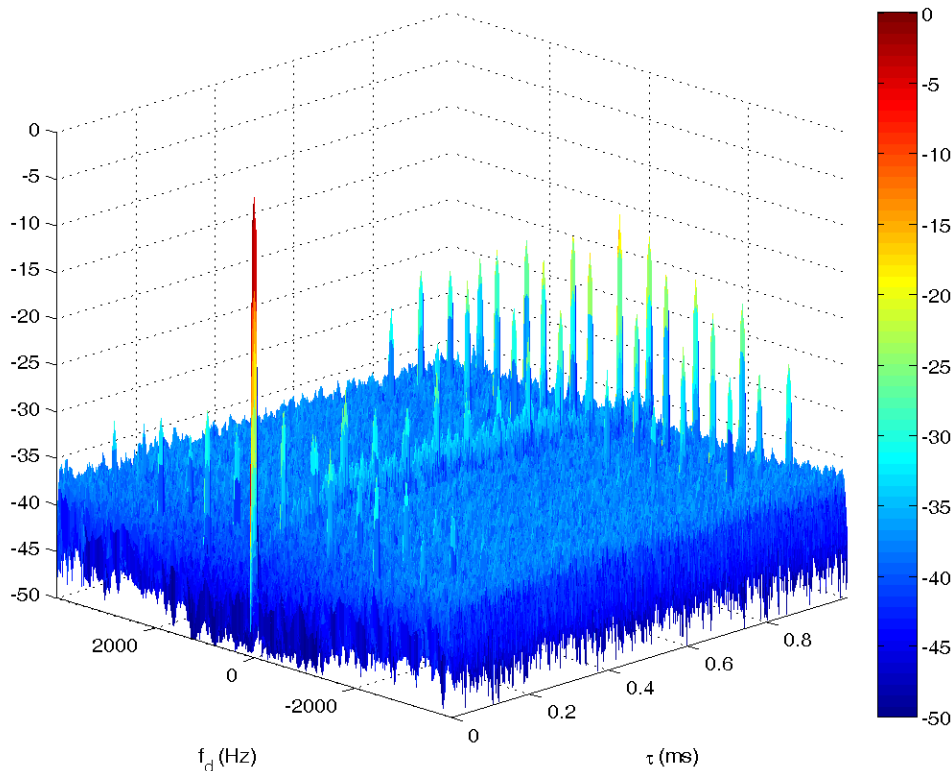
Considerando il segnale ottenuto dal canale di riferimento,  $S_{ref}(t)$ , come una copia identica del segnale trasmesso e assumendo che i segnali siano campionati con una frequenza pari a  $f_s$ , la 2D-CCF di Eq. 1 può essere espressa, nel tempo discreto, come:

$$\chi[l, m] = \sum_{n=0}^{N-1} s_{surv}[n] \cdot s_{ref}^*[n-l] \cdot e^{-j2\pi \frac{mn}{N}} \quad \text{Eq. 15}$$

Dove

- N, pari alla parte inferiore del prodotto  $T_{int}f_s$ , rappresenta il numero di campioni integrati;
- l indica il campione in tempo corrispondente al ritardo  $\tau = l/f_s$ ;
- m indica il campione in Doppler corrispondente allo spostamento Doppler  $f_D = mfs/N$ .

In aggiunta alla eco riflessa dal generico target, diverse tipologie di segnale interferente potrebbero degradare le prestazioni del sistema passivo come, ad esempio, la natura incontrollabile della forma d'onda sfruttata dal sistema oppure il segnale diretto interferente o le riflessioni del Multipath; questi contributi potrebbero sia mascherare eventuali eco di generici target che introdurre falsi allarmi nel sistema.

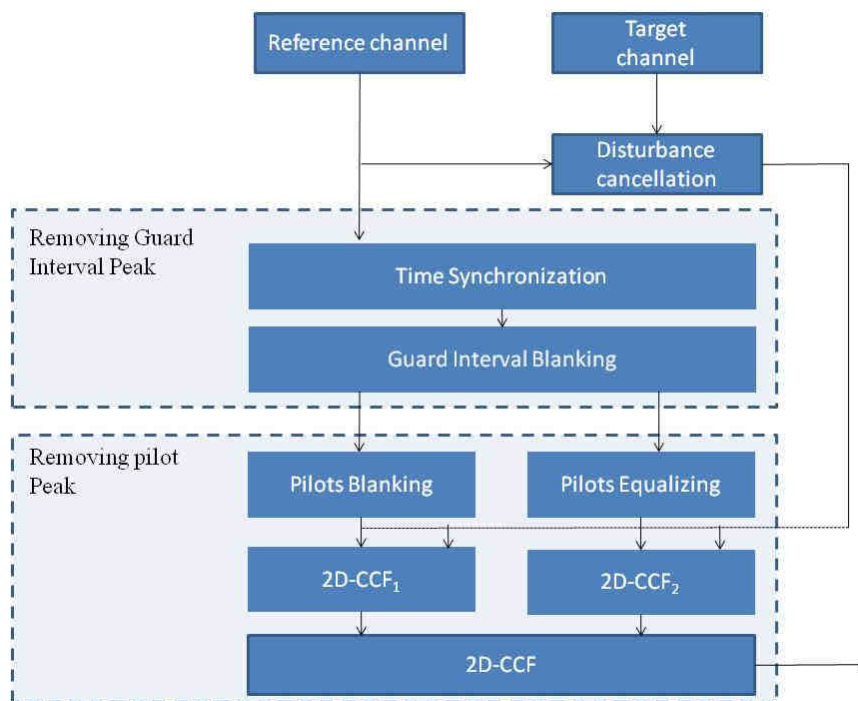


**Figura 20 – Funzione d'ambiguità ottenuta dal segnale DVB-T**

Inoltre, come si vede dalla Figura 20, la presenza di una specifica struttura deterministica e periodica interna al segnale DVB come, la presenza dell'intervallo di guardia (copia ciclica dell'ultimo pezzo della parte utile del simbolo OFDM) e quella dei piloti continui e sparsi introduce un numero di picchi indesiderati all'interno della funzione d'ambiguità che contribuisce ad aumentare il peggioramento delle prestazioni. In letteratura sono state

proposte differenti tecniche che permettono la rimozione dei picchi indesiderati all'interno della funzione d'ambiguità ottenuta utilizzando il segnale DVB-T (vedi [9], [10] e [11]). In Figura 21 è riportato lo schema a blocchi che individua i passaggi fondamentali che consentono di migliorare le prestazioni della funzione d'ambiguità.

La rimozione dei picchi indesiderati è effettuata utilizzando la conoscenza della struttura interna del generico simbolo OFDM del segnale DVB-T. In particolare, i piloti continui sono trasmessi sempre sulle stesse sottoportanti in frequenza da un simbolo OFDM al successivo mentre i piloti sparsi sono trasmessi sulle stesse sottoportanti in frequenza ogni quattro simboli OFDM. Perciò un'operazione preliminare richiesta da queste tecniche è la sincronizzazione in tempo con cui si individua l'inizio del generico simbolo OFDM e quindi può essere effettuato il blanking dell'intervallo di guardia. Inoltre queste tecniche richiedono anche la conoscenza dell'inizio del Frame e la stima dell'offset in frequenza per permettere la corretta equalizzazione e annullamento delle sottoportanti che trasportano i piloti continui e sparsi.

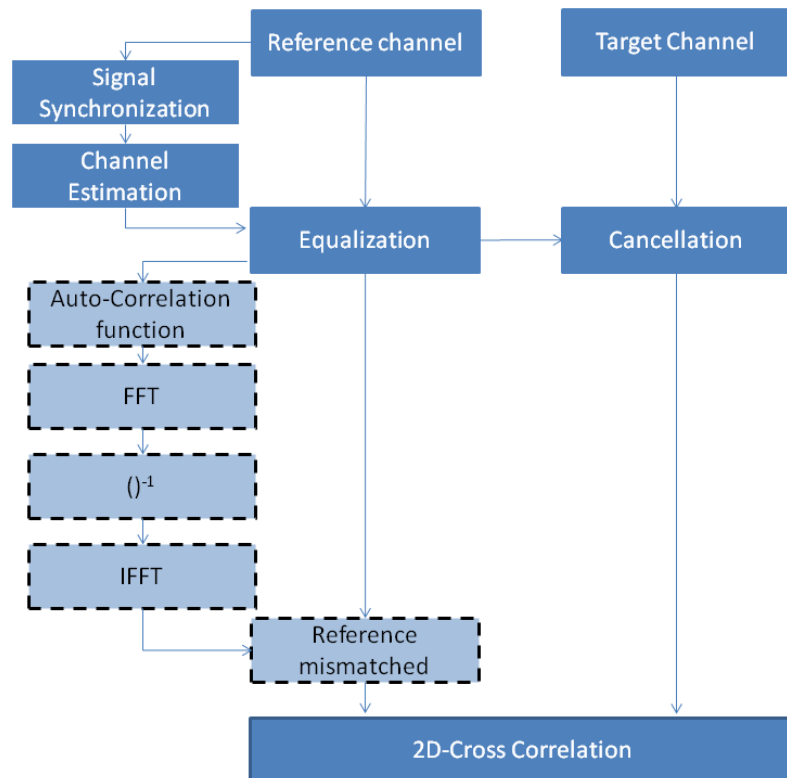


**Figura 21 – Schema a blocchi delle tecniche per il miglioramento della funzione d'ambiguità presenti in letteratura.**

Come è evidente dalla Figura 21, è indispensabile effettuare le due operazioni (annullamento ed equalizzazione dei piloti) in parallelo e separatamente per ottenere le due funzioni d'ambiguità da cui ottenere il risultato finale attraverso una combinazione non

lineare delle due (cercando il minimo valore in potenza tra i due pixel corrispondenti). Un tecnica differente da quella appena mostrata è descritta in [11] e schematizzata in Figura 22 dove le singole operazioni sono evidenziate da rettangoli tratteggiati; questa modalità è di tipo lineare ottenuta basandosi sul controllo dell'autocorrelazione del segnale. In particolare, si effettua la valutazione dei coefficienti che serviranno per pesare opportunamente il segnale di riferimento al fine di rimuovere i picchi indesiderati. Attraverso questo filtraggio si riescono a rimuovere i picchi posizionati a Doppler nulla, ma non i picchi spuri a Doppler non nulle; questa caratteristica rende il filtro utile e utilizzabile in applicazioni di sorveglianza marittima e, in generale, in applicazioni in cui le velocità dei potenziali target sono tali da produrre le frequenze doppler relative non elevate. Un vantaggio che si ottiene dall'uso di questa tecnica è legato alla possibilità di non utilizzare nessun tipo di sincronismo per l'applicazione e la valutazione del filtro lineare riducendo così la complessità del sistema.

Nel seguito, dopo la descrizione degli algoritmi di sincronizzazione e la valutazione dell'impatto che questi algoritmi hanno sulle prestazioni, descriveremo nel dettaglio la tecnica proposta e valutando le prestazioni sia in termini di perdite che in rapporto picco a lobi laterali (PSLR) ottenibile.



**Figura 22 – Schema a blocchi della tecnica innovativa proposta**



### 3.2 Sincronizzazione del segnale DVB in un radar passivo

Gli algoritmi di sincronizzazione che verranno presentati nel seguito sono legati ai principali errori presenti in un generico ricevitore. In particolare il segnale DVB è affetto da un'incertezza

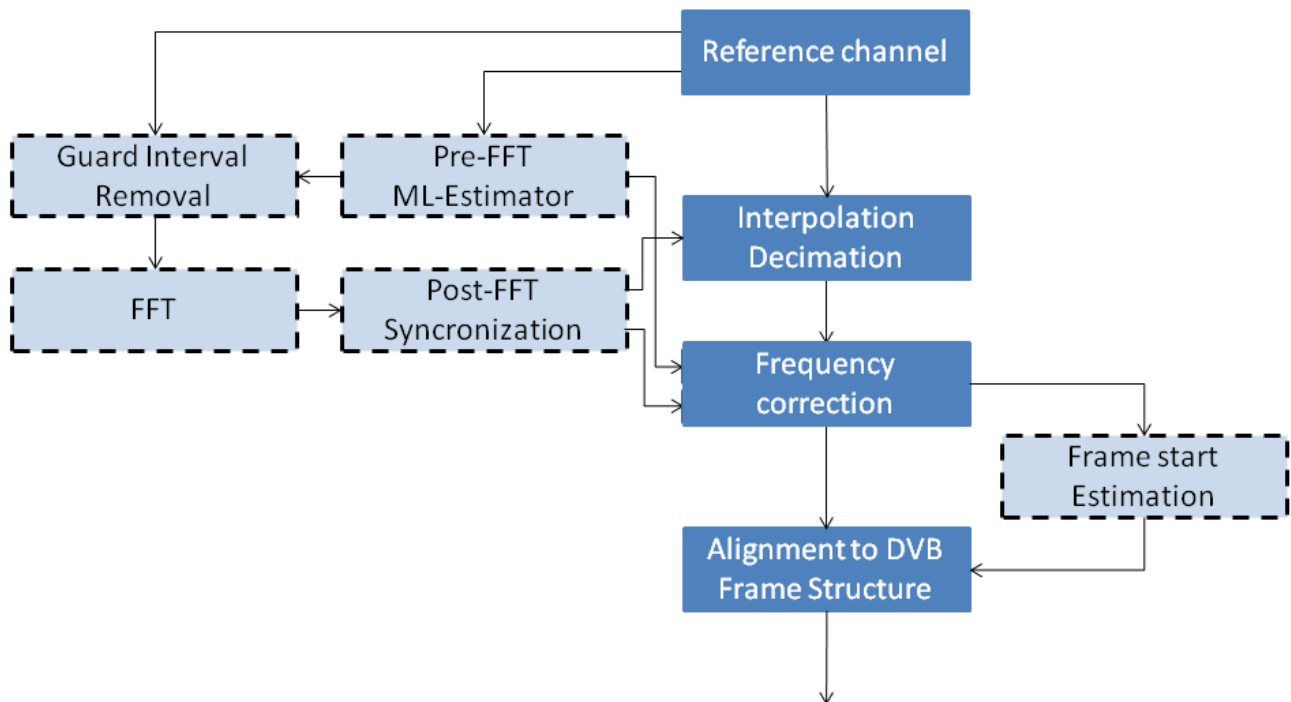
- sulla frequenza portante causata dal diverso oscillatore locale utilizzato in trasmissione e ricezione. Questo errore potrebbe introdurre uno spostamento di tutte le sottoportanti interne al generico simbolo OFDM.
- sulla frequenza di campionamento rispetto alla frequenza nominale data dallo standard. Questo errore comporterebbe un disallineamento del simbolo OFDM in tempo e uno spostamento di tutte le sottoportanti interne al simbolo OFDM.
- sul tempo di arrivo del simbolo OFDM non noto al ricevitore.

Un ricevitore quindi ha due possibilità per estrarre l'informazione necessaria al fine di sincronizzare il segnale: la prima possibilità è sfruttare una precisa sequenza di Training, prima di demodulare le sottoportanti. Poichè lo standard DVB-T e DVB-SH non prevedono l'utilizzo di una specifica sequenza di training, l'uso dell'intervallo di guardia per la sincronizzazione risulta essere di particolare interesse anche se le prestazioni dipendono dalle caratteristiche del canale in quanto l'intervallo di guardia potrebbe essere affetto da interferenza intersimbolica. La seconda possibilità è sfruttare la struttura interna del segnale DVB dopo la demodulazione delle sottoportanti. Questo approccio potrebbe aumentare il tempo di acquisizione del sistema.

In Figura 23, viene mostrato lo schema a blocchi relativo all'elaborazione del segnale di riferimento che consente di stimare il sincronismo del segnale DVB. In particolare vengono eseguiti tre passaggi fondamentali (vedi [55], [56] e [57] ) quali:

- 1- Sincronizzazione Pre-FFT: in questo caso, una copia del segnale di riferimento viene utilizzata dall'algoritmo ML-estimator per ottenere una stima del ritardo temporale relativo all'inizio del generico simbolo OFDM e una prima stima grossolana dell'offset di frequenza. La stima temporale servirà per la rimozione dell'intervallo di guardia mentre la seconda verrà utilizzata per compensare l'offset di frequenza sul segnale di riferimento;

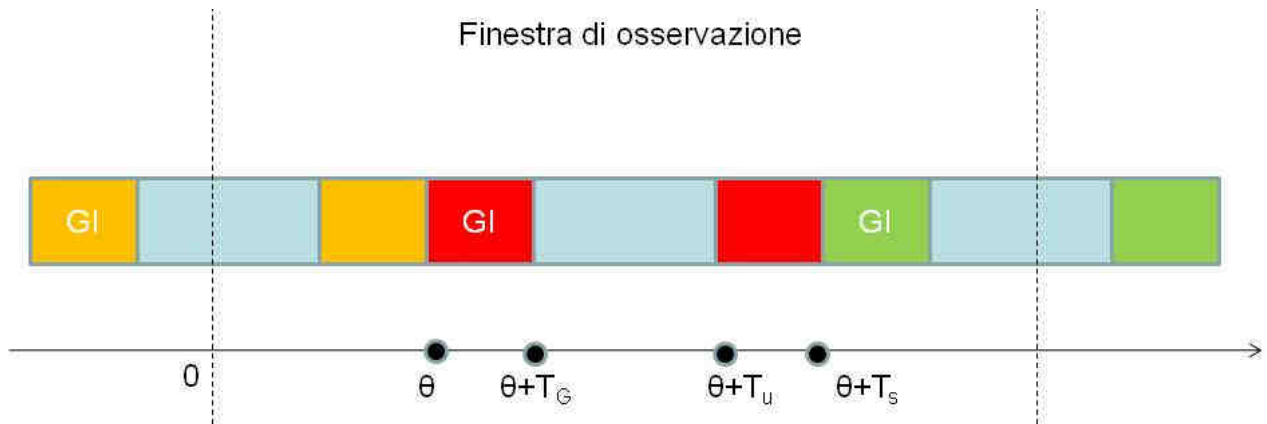
- 2- Sincronizzazione Post-FFT: in questo caso, una copia della parte utile del simbolo OFDM, ottenuta dopo l'operazione di Trasformata di Fourier, verrà utilizzata per stimare l'offset di frequenza residuo e la frequenza di campionamento. La prima stima verrà utilizzata per compensare l'offset di frequenza sul segnale di riferimento mentre la seconda per interpolare e/o decimare il segnale di riferimento stesso per compensare l'errore commesso rispetto alla frequenza di campionamento nominale.
- 3- Stima dell'inizio del Frame DVB: una copia del segnale di riferimento, ottenuto dopo la correzione della frequenza di campionamento e la compensazione dell'offset in frequenza, verrà utilizzata per stimare l'inizio del Frame OFDM al fine di allineare il segnale stesso con la struttura descritta all'interno dello standard.



**Figura 23 – Schema di processing del canale di riferimento per la stima del sincronismo**

Nel seguito verranno descritti singolarmente gli algoritmi che consentono la valutazione delle stime che consentono di ottenere il segnale sincronizzato in frequenza e allineato rispetto alla struttura del Frame DVB.

### 3.2.1 Sincronizzazione Pre-FFT: ML-Estimator



**Figura 24 – Struttura del simbolo OFDM interno al segnale DVB. GI indica l'intervallo di guardia**

In [55] è descritto lo stimatore a massima verosimiglianza congiunta del tempo di inizio del simbolo OFDM e dell'offset della frequenza portante. Il flusso dati contenuto all'interno della struttura del segnale DVB può essere considerato come una combinazione lineare di variabili aleatorie distribuite identicamente. Grazie al numero elevato di sottoportanti interne ad un singolo simbolo OFDM, per il teorema del limite centrale, è possibile approssimare il segnale DVB come un processo Gaussiano complesso con parte reale e immaginaria indipendenti.

Assumendo una finestra di osservazione del segnale DVB pari a  $2T_U+T_G$  (dove  $T_U$  indica la durata della parte utile del simbolo OFDM mentre  $T_G$  indica la durata dell'intervallo di guardia) come riportato in Figura 24, la posizione dell'inizio del simbolo all'interno della finestra d'osservazione stessa è non nota perché non è noto il ritardo introdotto dal canale al ricevitore.

La stima del tempo di inizio del simbolo OFDM e dell'offset della frequenza portante può essere valutata massimizzando la funzione di verosimiglianza, scritta nel seguente modo e ricavata in [55]:

$$\Lambda(\theta, \varepsilon) = |\gamma(\theta)| \cos(2\pi\varepsilon + \angle\gamma(\theta)) - \rho \cdot \Phi(\theta) \quad \text{Eq. 16}$$

Con

$$\Phi(\theta_i) = \frac{1}{2} \sum_{k=\theta_i}^{\theta_i+N_G-1} |s(k)|^2 + |s(k+N_U)|^2 \quad \text{Eq. 17}$$

$$\gamma(\theta_i) = \sum_{k=\theta_i}^{\theta_i+N_G-1} s(k)s^*(k+N_U) \quad \text{Eq. 18}$$

$$\rho = \left| \frac{E\{s(k)s^*(k+N_U)\}}{\sqrt{E\{|s(k)|^2\}E\{|s(k+N_U)|^2\}}} \right| \quad \text{Eq. 19}$$

Il termine  $s(k)$  presente all'interno delle relazioni Eq. 17, Eq. 18 e Eq. 19 indica il segnale DVB considerato mentre il termine  $\theta_i$  rappresenta il generico istante temporale interno alla finestra di osservazione. Il primo termine in Eq. 16 rappresenta il modulo pesato della funzione  $\gamma$ , ricavata in Eq. 18, che rappresenta la cross-correlazione tra la porzione di segnale di durata pari a  $T_G$  ( $s(k)$ ) e lo stesso segnale traslato di una quantità pari a  $T_U$  ( $s^*(k+N_U)$ ). I fattori di peso dipendono dal offset della frequenza portante che vogliamo stimare. Il secondo termine dell'equazione invece è un termine di energia che non dipende dalle quantità da stimare.

La relazione Eq. 19 rappresenta il coefficiente di correlazione i segnali  $s(k)$  e  $s(k+N_U)$ .

La massimizzazione della funzione di verosimiglianza scritta in Eq. 16 può essere ottenuta in due passi:

$$\max_{(\theta, \varepsilon)} \Lambda(\theta, \varepsilon) = \max_{(\theta)} \max_{(\varepsilon)} \Lambda(\theta, \varepsilon) = \max_{(\theta)} \Lambda(\theta, \hat{\varepsilon}_{ML}(\theta)) \quad \text{Eq. 20}$$

Il primo passo è massimizzare la funzione rispetto all'offset di frequenza  $\varepsilon$ , ponendo il termine coseno della relazione Eq. 16 uguale a uno. Questo produce la stima ML di  $\varepsilon$  scritta di seguito.

$$\hat{\varepsilon}_{ML}(\theta) = -\frac{1}{2\pi} \angle \gamma(\theta) + n \quad \text{Eq. 21}$$

Dove  $n$  rappresenta un numero intero.

La periodicità del coseno produce la stima di diversi massimi e quindi una misura ambigua della stima dell'offset di frequenza. Poiché questa ambiguità viene risolta attraverso gli algoritmi che presenteremo nel seguito, senza perdite di generalità, possiamo assumere  $n=0$  e quindi un offset di frequenza normalizzato al massimo pari a  $\pm 0,5$ .

Il secondo passo è rappresentato dalla stima dell'indice di inizio del generico simbolo OFDM. In questo caso, poiché il termine coseno è posto ad uno, possiamo scrivere

$$\Lambda(\theta, \hat{\varepsilon}_{ML}(\theta)) = |\gamma(\theta)| - \rho \cdot \Phi(\theta) \quad \text{Eq. 22}$$

Dunque, la stima congiunta di  $\theta$  e  $\varepsilon$  diviene

$$\hat{\theta}_{ML} = \arg \max_{\theta} (|\gamma(\theta)| - \rho \cdot \Phi(\theta))$$

$$\hat{\varepsilon}_{ML}(\theta) = -\frac{1}{2\pi} \angle \gamma(\theta)$$

Eq. 23

Si noti che la funzione di verosimiglianza dipende da due quantità (quindi le prestazioni dello stimatore sono legate ad esse): il numero di campioni all'interno dell'intervallo di guardia, che è noto al ricevitore, e il coefficiente di correlazione, che dipende dal SNR (può essere aumentato incrementando la durata del segnale  $s(k)$  in Eq. 18) .

In Figura 25 si mostra un esempio che riporta i segnali generati dalle stime ML in funzione del ritardo temporale (Eq. 23) in cui è stato considerato il segnale DVB trasmesso in modalità 2K con un intervallo di guardia pari a 1/8 e un rapporto segnale a rumore di 20 dB.

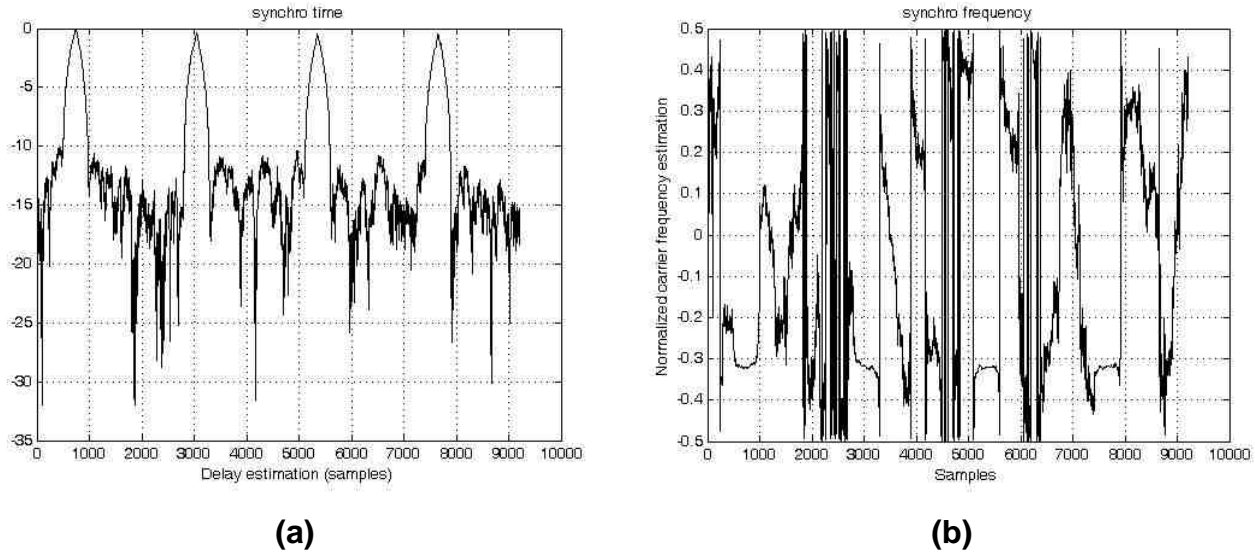


Figura 25 –Stima del ritardo temporale  $\theta$  (a) e dell'offset di frequenza  $\varepsilon$  (b)

### 3.2.2 Sincronizzazione Post-FFT: Fine Synchronization

In [56] sono presentati gli algoritmi che permettono di stimare l'offset di frequenza residuo, rappresentato dall'indice intero  $n$  interno all'Eq. 21 e dagli errori introdotti dalla frequenza di campionamento, attraverso la conoscenza della posizione dei piloti continui all'interno del simbolo OFDM.

Dopo l'applicazione dello stimatore ML, l'offset di frequenza restante ancora da stimare può essere espresso attraverso la relazione Eq. 24.

$$\Delta f' = \Delta f \cdot T_u = n_f + \Delta f'_F$$

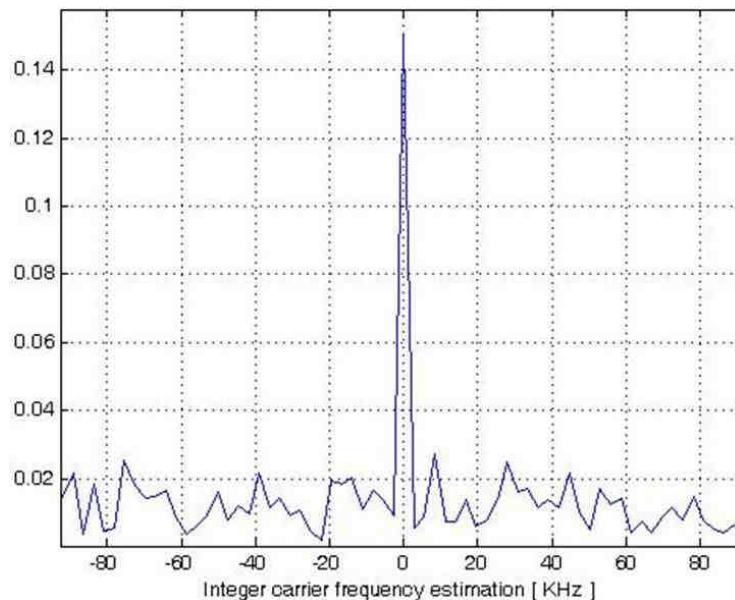
Eq. 24

La quantità  $n_l$  rappresenta un multiplo intero della quantità  $1/T_U$  che non viene stimata dallo stadio PRE FFT mentre  $\Delta f'_F$  è il residuo finale dovuto al disallineamento delle sottoportanti all'interno del simbolo OFDM.

Nel seguito verranno descritti nel dettaglio dei semplici algoritmi che permettono di stimare le componenti presenti al terzo membro dell'Eq. 24.

### 3.2.3 Stima della parte intera dell'Offset in frequenza

Dopo l'applicazione degli algoritmi Pre-FFT, l'offset di frequenza residuo  $\Delta f'_F$  è molto piccolo in quanto è dovuto essenzialmente ad errori sulla frequenza di campionamento dunque, in prima approssimazione, per la stima della parte intera dell'offset in frequenza possiamo trascurare questo termine.



**Figura 26 - Stima dell'offset in frequenza multiplo intero della quantità  $1/T_U$**

Lo spostamento spettrale  $n_l$ , di dimensione pari a multipli interi della spaziatura tra le sottoportanti, deve essere rilevato attraverso l'uso dei piloti continui, interni al simbolo OFDM, trasmessi sempre all'interno delle stesse sottoportanti in frequenza tra un simbolo e il successivo; queste sottoportanti sono trasmesse sempre con una potenza doppia rispetto al resto dei dati e sono modulate attraverso simboli tempo-invarianti. Correlando i campioni in uscita di due simboli OFDM consecutivi, in corrispondenza delle portanti continue, si ottiene il valore massimo in corrispondenza dello spostamento spettrale da stimare.

In modo del tutto simile all'algoritmo descritto in precedenza (stimatore ML), i campioni di correlazione  $s(k)$  contenuti in Eq. 25 appartengono all'insieme dei piloti continui interni al simbolo OFDM descritto nello standard.

$$\hat{n}_l = \arg \max_{m \in M} \left| \sum_{k \in I} s_l(k) s_{l+1}^*(k+m) \right| \quad \text{Eq. 25}$$

Nell'espressione riportata in Eq. 25,  $l$  indica il generico simbolo OFDM,  $I$  indica l'insieme dei piloti continui e  $M$  il set di shift considerato per la stima di  $n_l$ . In Figura 26 si riporta il segnale generato dall'algoritmo considerato nel generico istante di tempo; in questo caso specifico, la stima dell'indice è pari a zero e quindi lo spettro del simbolo OFDM ricevuto è allineato con le posizioni delle sottoportanti riportate nello standard.

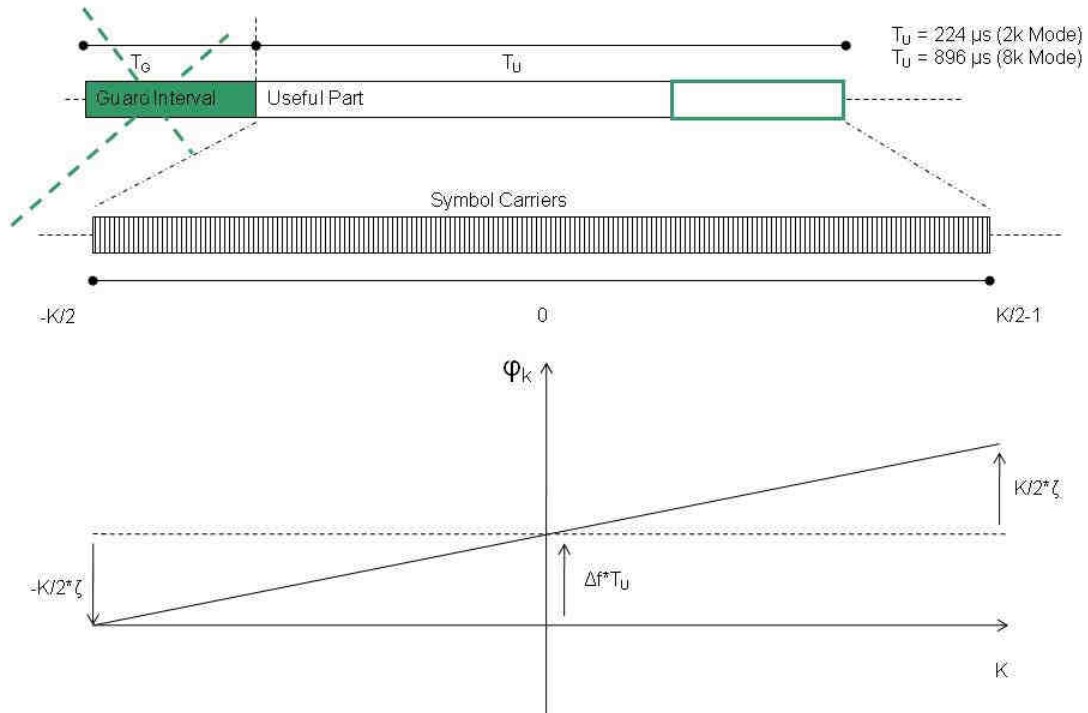
### 3.2.4 Stima della parte frazionale dell'Offset in frequenza

Dopo l'applicazione degli algoritmi precedenti (stimatore ML e stima della parte intera dell'offset in frequenza), solo un piccolo offset residuo di frequenza portante resta da stimare per sincronizzare il segnale DVB ricevuto rispetto a quello trasmesso.

Per la stima del residuo dell'offset di frequenza portante e dell'offset di frequenza di campionamento facciamo riferimento agli algoritmi descritti nel dettaglio in [56].

I rivelatori dell'offset della frequenza portante e della frequenza di campionamento sono basati ancora sull'utilizzo della conoscenza delle posizioni dei piloti continui all'interno del generico simbolo OFDM sfruttando la correlzione post-FFT come visto in precedenza.

Considerando la rotazione delle sottoportanti interne al simbolo OFDM (Figura 27), consideriamo un semplice algoritmo che permette la stima dei due offset che mancano. Come si nota dalla figura, l'effetto dell'offset di frequenza portante è quello di ruotare tutte le sottoportanti di una certa quantità proporzionale alla durata della parte utile del simbolo OFDM, mentre l'offset di frequenza di campionamento è identificato da un contributo aggiuntivo proporzionale sia all'indice della sottoportante che al valore di offset da stimare.



**Figura 27 – Rotazione delle sottoportanti del simbolo dovuta all’offset di frequenza portante e a quello della frequenza di campionamento.**

L’algoritmo utilizza separatamente il sottoinsieme di sottoportanti, che trasportano i piloti continui, destro e sinistro rispetto al centro dello spettro OFDM.

$$\Delta \tilde{f}'_F = \frac{(\varphi_{2,l} + \varphi_{1,l})}{2 \cdot 2\pi \left(1 + \frac{N_G}{N_U}\right)}$$

$$\tilde{\zeta} = \frac{(\varphi_{2,l} - \varphi_{1,l})}{2 \cdot 2\pi \left(1 + \frac{N_G}{N_U}\right)} \cdot \frac{1}{\frac{K}{2}}$$

**Eq. 26**

$$\text{Con } \varphi_{1,l} = \arg\left(\sum_{k \in C_{1,l}} s_l(k) s_{l+1}^*(k)\right), \varphi_{2,l} = \arg\left(\sum_{k \in C_{2,1}} s_l(k) s_{l+1}^*(k)\right)$$

Nello specifico, viene valutata la fase della correlazione dei piloti in uscita dall’operazione di FFT di due simboli OFDM consecutivi utilizzando separatamente i due sottoinsiemi e, attraverso le relazioni riportate in Eq. 26, si ottengono le stime desiderate. Queste stime vengono poi elaborate successivamente attraverso dei loops che consentono di inseguire la stima. Nel nostro caso, è sufficiente soltanto questa stima “one-shot” dei due parametri per sincronizzare porzioni di forma d’onda.



In Eq. 26, il parametro  $K$  indica l'indice di sottoportante,  $N_G$  il numero di campioni relativo all'intervallo di guardia mentre  $N_U$  quello relativo alla parte utile del simbolo OFDM.

### 3.2.5 Stima dell'inizio del Frame OFDM

Il risultato ottenuto dopo l'applicazione degli algoritmi precedenti è un segnale DVB sincrono rispetto all'inizio del generico simbolo OFDM e all'errore di frequenza.

In questa sezione proponiamo un algoritmo che consente di stimare l'inizio del Frame OFDM, composto da quattro simboli OFDM, ottenuto sfruttando le caratteristiche del segnale stesso. Il segnale DVB può essere ottenuto come somma di due contributi diversi, come riportato in Eq. 27

$$s(t) = s_{DATA}(t) + s_{PILOTS}(t) \quad \text{Eq. 27}$$

Dove il primo termine al secondo membro rappresenta il segnale dovuto al flusso di dati trasmesso mentre il secondo termine è dovuto alle portanti pilota continue e sparse aggiunte all'interno del simbolo OFDM.

Essendo  $S_{PILOTS}(t)$  una parte deterministica del segnale DVB, la funzione di auto-correlazione media  $\langle x \rangle$  può essere ottenuta come somma tra il valore medio della funzione di auto-correlazione ottenuta sui dati, che possono essere assunti statisticamente indipendenti, e l'auto-correlazione ottenuta dal contributo dei piloti, come riportato in Eq. 28.

$$\langle x \rangle = \langle x_{DATA} \rangle + x_{PILOTS} \quad \text{Eq. 28}$$

Se consideriamo la cross-correlazione tra il segnale opportunamente ottenuto dopo l'applicazione degli algoritmi presentati in precedenza e un segnale di riferimento sintetizzato considerando le caratteristiche statistiche dei due diversi contributi, è possibile stimare l'inizio del Frame OFDM cercando l'indice che massimizza la funzione di cross-correlazione ottenuta.

$$\hat{n}_{\tau\_Frame} = \arg \max \left( \sum s_{ref}(t) s_{sintetize}^*(t) \right) \quad \text{Eq. 29}$$

In Figura 28 è mostrata la cross-correlazione tra due diversi segnali DVB simulati. Come si può notare, esistono diversi valori massimi ottenuti quando tutte le parti comuni dei due segnali si allineano perfettamente; questo accade ogni 4 simboli OFDM (Frame) quando tutte le sottoportanti che trasportano i piloti continui e sparsi dei due flussi coincidono.

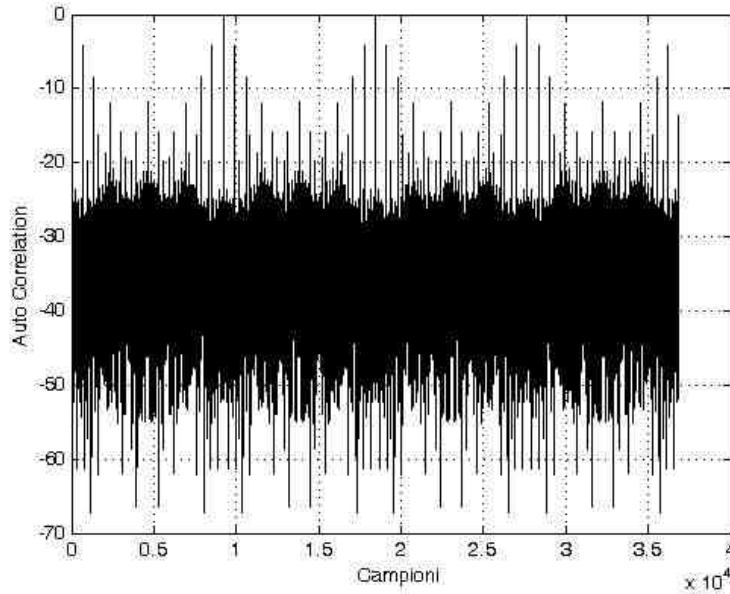


Figura 28 – Cross-correlazione del segnale DVB.

### 3.3 Nuovo dispositivo di filtraggio

Come accennato nel paragrafo 3.1, la presenza dei picchi, dovuti all'utilizzo del segnale DVB in un radar passivo, potrebbe introdurre falsi allarmi oppure mascherare un segnale riflesso da un eventuale target.

Come viene evidenziato in [51] e [52], la funzione d'ambiguità relativa a questo segnale presenta, oltre al picco principale posto a ritardo e doppler nulli, diversi picchi in posizioni differenti. La presenza di questi picchi è causata dalla struttura periodica e deterministica del segnale DVB. Nel dettaglio, il picco causato dall'esistenza dell'intervallo di guardia è posizionato ad un ritardo pari alla durata della parte utile del simbolo OFDM ( $T_U$ ) mentre, i picchi causati dalle portanti pilota, possono essere divisi in due sezioni:

1. Intra-simbolo: vengono definiti tutti i picchi posizionati ad un ritardo inferiore alla durata del simbolo OFDM ( $T_S=T_U+T_G$ ) perchè sono dovuti alla ripetizione dei piloti interni ad un singolo simbolo OFDM.
2. Inter-simbolo: vengono definiti tutti i picchi posizionati ad un ritardo superiore della durata del simbolo OFDM ( $T_S=T_U+T_G$ ) perchè sono causati dalla ripetizione dei piloti all'interno di differenti simboli OFDM.

Si noti che, come riportato anche in [9], l'aumento del tempo di integrazione con cui viene effettuata la stima della Mappa Range-Doppler, e l'utilizzo di reti di pesatura in entrambe le dimensioni, non comporta una riduzione della struttura dei lobi.

### 3.3.1 Posizione dei picchi Intra-Simbolo

L'insieme delle posizioni dei picchi intra-simbolo, denominato  $A$  in Eq. 30, può essere ricavata dall'unione delle posizioni dei picchi causati dalla ripetizione dei dati ( $A_D$ ), dei TPS ( $A_T$ ), dei piloti continui ( $A_C$ ) e di quelli sparsi ( $A_S$ ) interni ad un singolo simbolo OFDM.

$$A = A_D \cup A_T \cup A_C \cup A_S \quad \text{Eq. 30}$$

Al flusso di dati trasmesso attraverso lo standard DVB-T viene applicato l'algoritmo di compressione MPEG-2, la dispersione di Energia e l'applicazione di codici per il controllo dell'errore e l'interleaving. L'unico picco generato dai dati è causato dall'introduzione dell'intervallo di guardia che è il prefisso ciclico della parte utile del simbolo OFDM, e la cui posizione è espressa in Eq. 31.

$$A_D = \left\{ \left( T_U, \frac{i}{T_S} \right) \right\} \quad \forall i \in Z \quad \text{Eq. 31}$$

L'introduzione, all'interno del simbolo OFDM, delle portanti dedicate al trasporto del messaggio TPS non comporta nessuna nuova generazione di picchi aggiuntivi rispetto all'insieme identificato in Eq. 31. Il singolo bit del messaggio TPS viene trasportato all'interno di un singolo simbolo OFDM su diverse sotto-portanti. Il contenuto di queste sotto-portanti viene randomizzato attraverso l'utilizzo di una sequenza a massima lunghezza, come descritto in [53]. Inoltre, il contenuto del messaggio TPS non è periodico quindi la posizione dei picchi generati può essere espressa in Eq. 31.

$$A_T = A_D \quad \text{Eq. 32}$$

L'introduzione dei piloti continui comporta la nascita di picchi all'interno della funzione di ambiguità. In particolare, poiché il più alto comune divisore delle posizioni delle portanti pilota continue è pari a 3, il simbolo costituito da i piloti continui è periodico con un periodo pari a  $T_U/3$  quindi, l'insieme può assumere la seguente espressione

$$A_C = \left\{ \left( n \frac{T_U}{3}, \frac{i}{T_S} \right) \right\} \quad n = 1, 2, 3; \quad i \in Z \quad \text{Eq. 33}$$

I piloti sparsi sono inseriti all'interno del simbolo OFDM con una periodicità pari a 4 simboli con un periodo pari a  $T_U/12$  in aggiunta alla durata dell'intervallo di guardia  $T_G$ .

L'insieme contenente i picchi viene riportato in Eq. 34.

$$\begin{aligned}
A_s &= \left\{ \left( n \frac{T_U}{3}, \frac{i}{T_s} \right) \right\} \quad n = 1,2,3; \quad i \in Z \\
&\cup \left\{ \left( (4n+1) \frac{T_U}{12}, \frac{4i+1}{4T_s} \right) \right\} \quad n = 0,1,2,3; \quad i \in Z \\
&\cup \left\{ \left( (4n+2) \frac{T_U}{12}, \frac{4i+2}{4T_s} \right) \right\} \quad n = 0,1,2,3; \quad i \in Z \\
&\cup \left\{ \left( (4n+3) \frac{T_U}{12}, \frac{4i+3}{4T_s} \right) \right\} \quad n = 0,1,2; \quad i \in Z
\end{aligned}$$

Eq. 34

### 3.3.2 Posizione dei picchi Inter-Simbolo

I picchi Inter-simbolo sono generati dalla ripetizione dei piloti all'interno di differenti simboli OFDM. L'insieme contenete questi picchi è espresso in Eq. 35

$$B = B_C \cup B_S$$

Eq. 35

Dove  $B_C$  e  $B_S$  rappresentano l'insieme di picchi causato dall'insieme dei piloti continui e sparsi rispettivamente. Le posizioni dei piloti continui sono le stesse per tutti i simboli, quindi l'insieme  $B_C$  sarà

$$\begin{aligned}
B_C &= \{ (qT_s, 0) \mid q \in Z \} \\
&\cup \left\{ \left( T_G + \frac{nT_U}{3} + qT_s, \frac{i}{T_s} \right) \mid n = 1,2; q \in Z; i \in Z \right\}
\end{aligned}$$

Eq. 36

Le posizioni dei piloti sparsi sono le stesse ogni 4 simboli OFDM, quindi l'insieme  $B_S$  sarà

$$\begin{aligned}
B_S &= \{ (qT_s, 0) \mid q \in Z \} \\
&\cup \left\{ \left( \frac{nT_U}{3} + 4qT_s, \frac{i}{T_s} \right) \mid -3 < n < 3, n \neq 0; q \in Z; i \in Z \right\} \\
&\cup \left\{ \left( \frac{(4n+1)T_U}{12} + 4qT_s, \frac{4i+1}{4T_s} \right) \mid -3 < n < 3; q \in Z; i \in Z \right\} \\
&\cup \left\{ \left( \frac{(4n+2)T_U}{12} + 4qT_s, \frac{4i+2}{4T_s} \right) \mid -4 < n < 3; q \in Z; i \in Z \right\} \\
&\cup \left\{ \left( \frac{(4n+3)T_U}{12} + 4qT_s, \frac{4i+3}{4T_s} \right) \mid -4 < n < 2; q \in Z; i \in Z \right\}
\end{aligned}$$

Eq. 37

### 3.3.3 Descrizione del filtro

La rimozione dei lobi laterali della funzione d'ambiguità può essere eseguita attraverso l'elaborazione del segnale di riferimento con un filtro lineare appropriato; i coefficienti del suddetto filtro sono ricavati sulla base della conoscenza delle caratteristiche del valore atteso dell'auto-correlazione.

Il segnale DVB-T può essere ottenuto come somma di due contributi diversi, come riportato in Eq. 38.

$$s(t) = s_{DATA}(t) + s_{PILOTS}(t) \quad \text{Eq. 38}$$

Il primo termine al secondo membro rappresenta il segnale dovuto al flusso di dati trasmesso mentre il secondo termine è dovuto alle portanti pilota continue e sparse aggiunte all'interno del simbolo OFDM. Essendo  $S_{PILOTS}(t)$  una parte deterministica del segnale DVB-T, la funzione di auto-correlazione media  $\chi(\tau)$  può essere ottenuta come somma tra il valore medio della funzione di auto-correlazione ottenuta sui dati e quella ottenuta dal contributo dei piloti.

$$\langle \chi(\tau) \rangle = \langle \chi_{DATA}(\tau) \rangle + \chi_{PILOTS}(\tau) \quad \text{Eq. 39}$$

Poiché la disposizione dei piloti all'interno della struttura del segnale DVB è periodica, il valore medio dell'auto-correlazione può essere assunto periodico quando si considera la durata del segnale pari ad un multiplo intero di un super-simbolo (4 simboli OFDM).

Se consideriamo la versione campionata della funzione di auto-correlazione, è possibile ricavare i coefficienti di un filtro  $w(n)$  dalla seguente operazione:

$$\underline{\chi}(n) \otimes \underline{w}(n) = A\delta(0) \quad \text{Eq. 40}$$

Dove al primo membro dell'espressione Eq. 40 è rappresentata la convoluzione tra l'auto-correlazione e gli stessi coefficienti mentre al secondo membro è riportato il risultato desiderato cioè, a meno di un fattore di scala A, una delta di Dirac centrata nell'origine.

I coefficienti possono essere ricavati dall'espressione seguente:

$$\underline{w}(n) = \mathbf{R}^{-1} [A \ 0 \ \dots \ 0]^T = A \cdot \text{diag}(\mathbf{R}^{-1}) \quad \text{Eq. 41}$$

Dove, considerando le proprietà dell'auto-correlazione,  $\mathbf{R}$  risulta essere una matrice di Toeplitz circolante contenente sulla generica riga la funzione di auto-correlazione come specificato nel seguito.

$$\mathbf{R} = \begin{bmatrix} \chi(0) & \chi(1) & \cdots & \chi(N-1) \\ \chi(N-1) & \chi(0) & \cdots & \chi(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ \chi(N-(N-1)) & \chi(N-(N-2)) & \cdots & \chi(0) \end{bmatrix} \quad \text{Eq. 42}$$

Sfruttando le proprietà di queste matrici, in particolare le relazioni tra queste matrici e la trasformata di Fourier, l'espressione riportata in Eq. 41 può essere risolta nel modo seguente

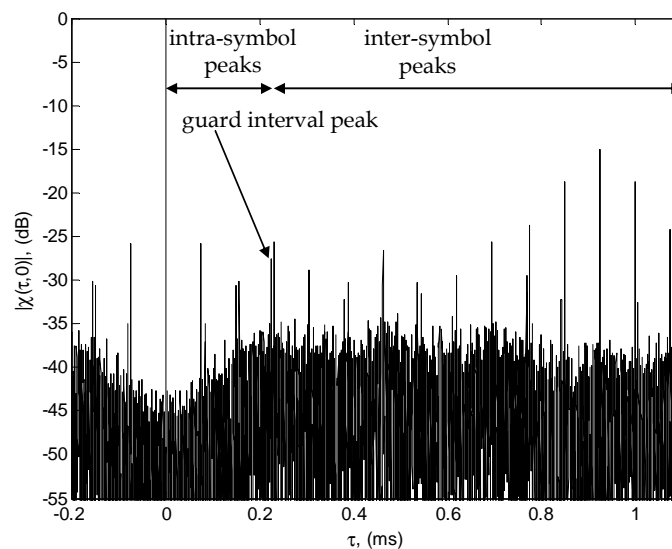
$$\underline{w}(n) = \text{IFFT} \left( \frac{1}{\text{FFT}(\underline{\chi})} \right) \quad \text{Eq. 43}$$

Ottenendo così il risultato desiderato.

### 3.4 Confronto prestazionale

#### 3.4.1 Auto-correlazione

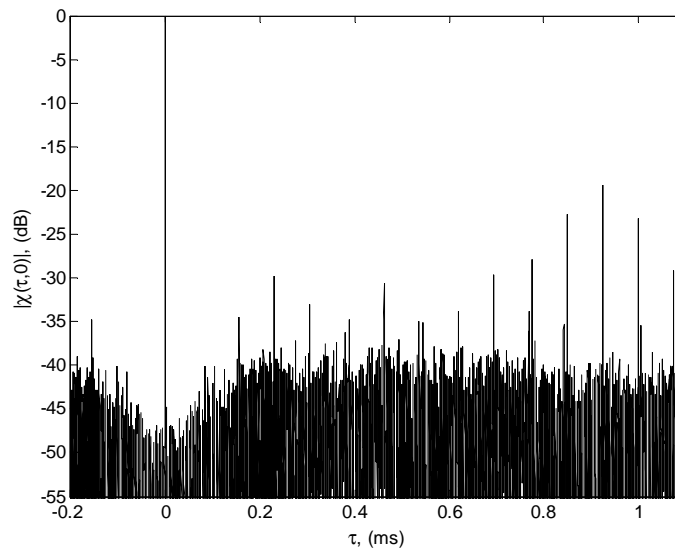
In questa sezione si mostra la funzione di auto-correlazione ottenuta utilizzando il segnale DVB-T, in modalità 2K, e confrontando le tecniche per il controllo dei lobi laterali presenti in letteratura con l'approccio proposto. In Figura 29 si riporta l'auto-correlazione del segnale DVB-T in cui sono identificati i diversi picchi indesiderati in accordo a quanto descritto nel paragrafo 3.3.



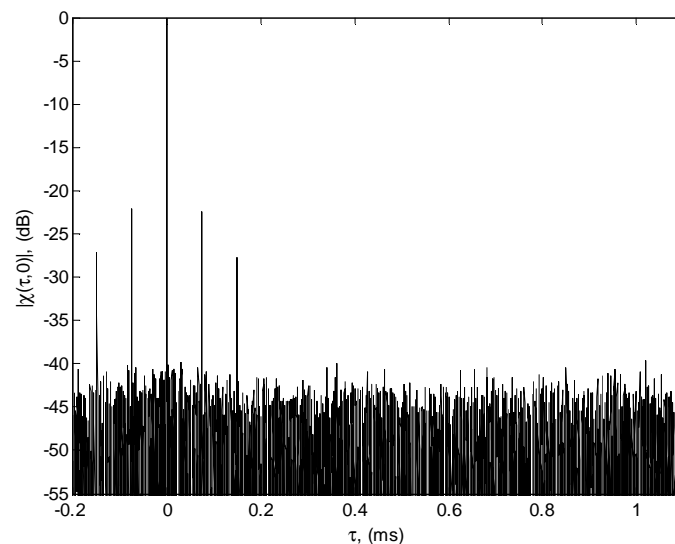
**Figura 29 – Auto-correlazione del segnale DVB-T in modalità 2K.**

In Figura 30 e Figura 31 si riportano i risultati ottenuti dopo l'applicazione separata delle due sezioni parallele, equalizzazione e annullamento rispettivamente; come si nota, le

operazioni effettuate singolarmente sono in grado di eliminare solo parte dei picchi presenti, nello specifico i picchi inter-simbolo o intra-simbolo.

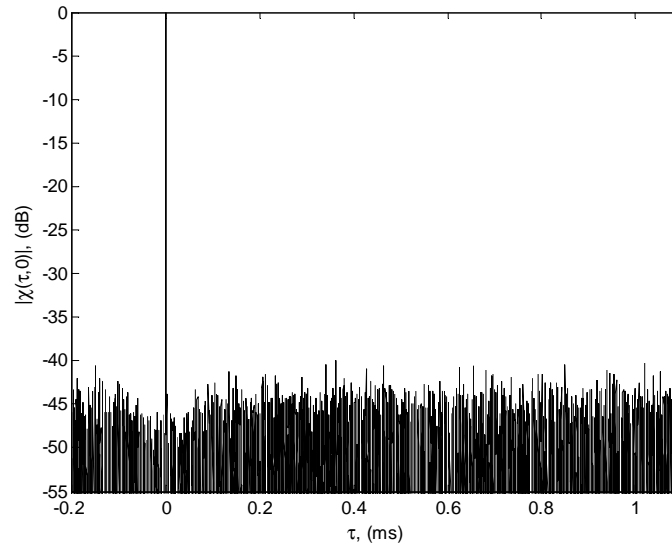


**Figura 30 – Auto-correlazione ottenuta dopo l’equalizzazione**

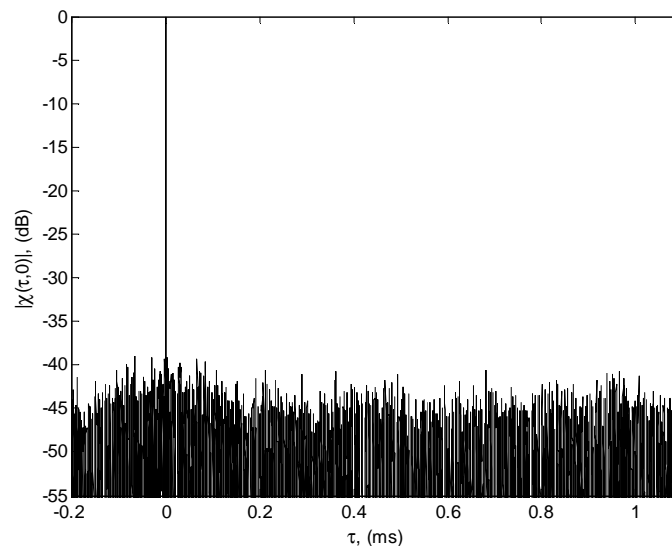


**Figura 31 – Auto-correlazione dopo l’annullamento delle portanti**

Le Figura 32 e Figura 33 riportano i risultati ottenuti dopo l’applicazione delle due tecniche; nello specifico, la Figura 32 è ottenuta minimizzando il risultato riportato nelle due figure precedenti, mentre la Figura 33 mostra l’auto-correlazione ottenuta dopo l’applicazione del filtro lineare. Da un confronto visivo, i due approcci producono gli stessi risultati in termini di PSLR in quanto sono in grado di rimuovere tutti i picchi presenti in Figura 29; tuttavia il filtraggio lineare non richiede i due step di elaborazione in parallelo separati, né tecniche di sincronizzazione per ottenere il risultato finale e quindi riduce il costo computazionale complessivo e la complessità del sistema.



**Figura 32 - Auto-correlazione ottenuta dopo la minimizzazione dei risultati in Figura 30 e Figura 31**



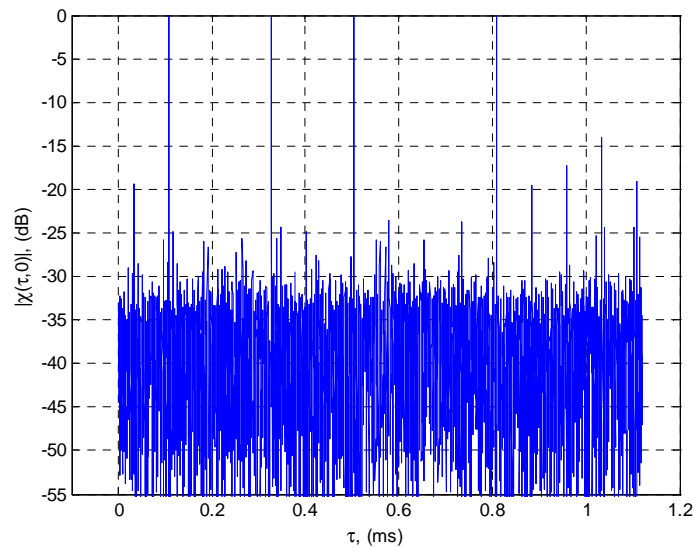
**Figura 33 – Auto-correlazione dopo l'applicazione del filtraggio lineare**

### 3.4.2 Picchi residui e perdite di SNR

Una conseguenza della combinazione non lineare (operazione di minimo), rispetto al filtraggio lineare, è la presenza di picchi residui quando più di un target, nello specifico quattro, è presente all'interno dell'area di copertura. Di seguito mostriamo un esempio in cui si considera la frequenza Doppler di tutti i target presenti nulla, l'ampiezza delle repliche relative ai vari target unitaria e i ritardi pari a 109  $\mu\text{s}$ , 328  $\mu\text{s}$ , 503  $\mu\text{s}$  e 809  $\mu\text{s}$  rispettivamente. L'unica fonte di disturbo è rappresentata dal rumore termico. I target

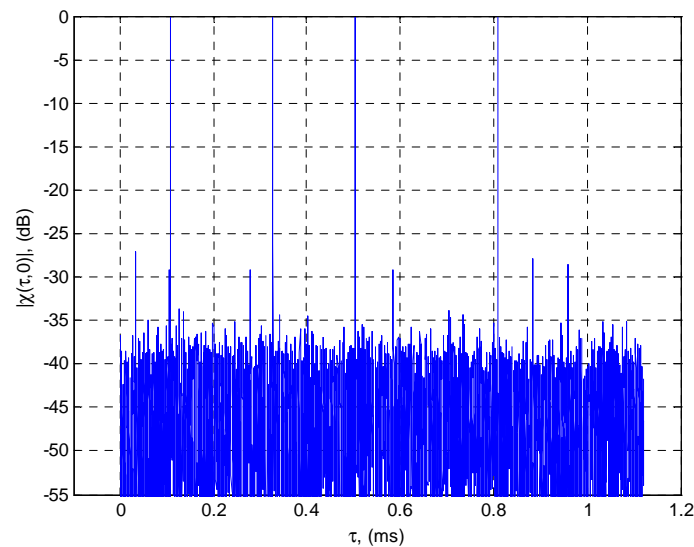


considerati vengono inseriti all'interno del segnale di sorveglianza utilizzando repliche del segnale di riferimento spostate in Doppler e ritardate.

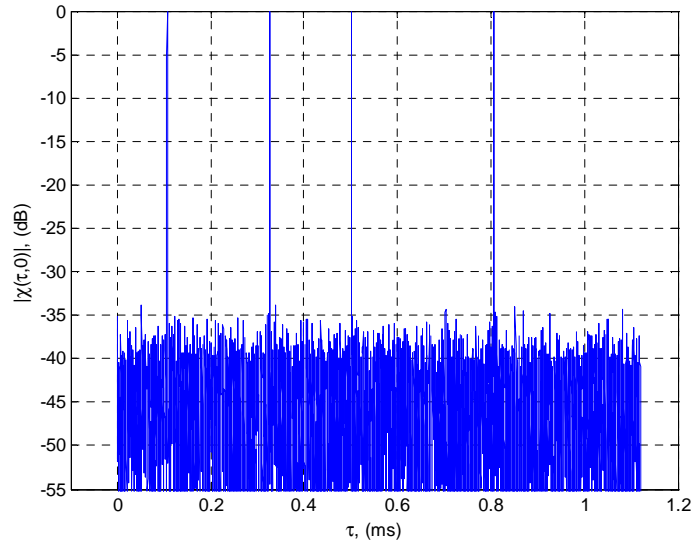


**Figura 34 – Cross-correlazione utilizzando il filtro adattato standard**

In Figura 34 si mostra il risultato ottenuto utilizzando il filtro adattato standard e normalizzando la curva al suo massimo. I target inseriti sono ben visibili in quanto sono stati inseriti con ampiezza unitaria. Oltre ai target, sono presenti molti picchi spuri dovuti alla struttura del segnale.



**Figura 35 – Cross-correlazione ottenuta utilizzando il filtro disadattato ottenuto dalla combinazione non lineare**



**Figura 36 - Cross-correlazione ottenuta utilizzando il filtro disadattato ottenuto dall'approccio lineare**

In Figura 35 e Figura 36 si riportano le cross-correlazioni, ottenute dopo il filtraggio disadattato, normalizzate al proprio massimo locale. Come atteso, il filtro ottenuto dopo la combinazione non lineare delle cross-correlazioni presenta dei picchi indesiderati nella scena rispetto alla tecnica lineare. Questo risultato è dovuto al fatto che alcuni picchi indesiderati appartenenti alle diverse repliche del segnale hanno i ritardi temporali coincidenti e quindi la minimizzazione non consente di rimuovere completamente tutte le spurie.

L'applicazione dei coefficienti al segnale di riferimento produce un disadattamento rispetto al segnale ricevuto che implica una perdita in termini di SNR; questa perdita può essere valutata considerando il rapporto tra il SNR in uscita dal filtro adattato e quello in uscita dal filtro proposto. Nello specifico si avrà

$$\text{SNR loss} = \frac{(\sum_k |rx_k|^2)(\sum_k |ref_k|^2)}{\left| \sum_k ref_k^\# rx_k \right|^2} \quad \text{Eq. 44}$$

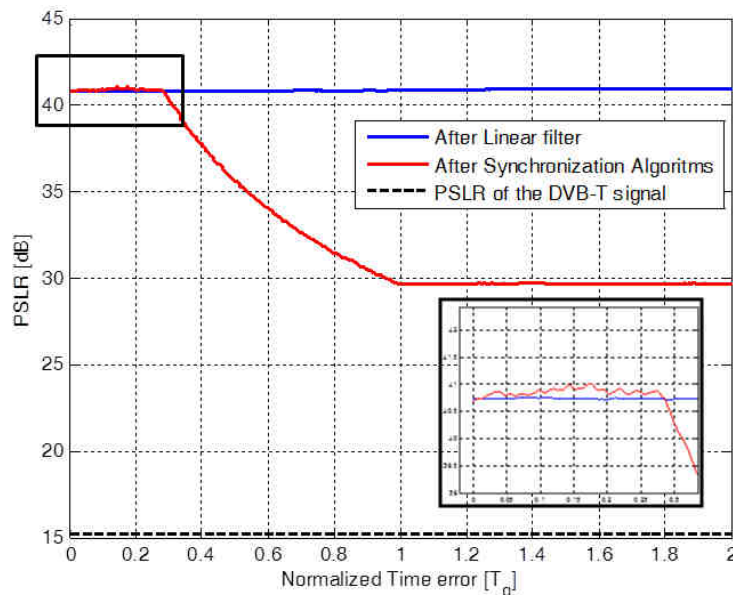
Dove  $rx_k$  rappresentano i campioni del segnale ricevuto mentre  $ref_k$  quelli del segnale di riferimento;  $ref_k^\#$  indicano i campioni del segnale di riferimento modificato in accordo ai pesi ottenuti in Eq. 43. La perdita ottenuta attraverso il filtro proposto è circa pari a 0.77 dB, confrontabile con quella ottenuta dalle tecniche presenti in letteratura.

### 3.4.3 Impatto degli errori di sincronizzazione

In questo paragrafo valutiamo l'effetto degli errori di sincronizzazione sulla valutazione della funzione d'ambiguità, ottenuta con le diverse tecniche mostrate, in termini di PSLR definito come il rapporto tra l'ampiezza del picco principale e l'ampiezza del primo lobo laterale più alto.

Tutte le curve che verranno mostrate di seguito si riferiscono a segnali DVB simulati in accordo alla modalità di trasmissione 8k. Risultati analoghi possono essere ottenuti considerando la modalità 2k.

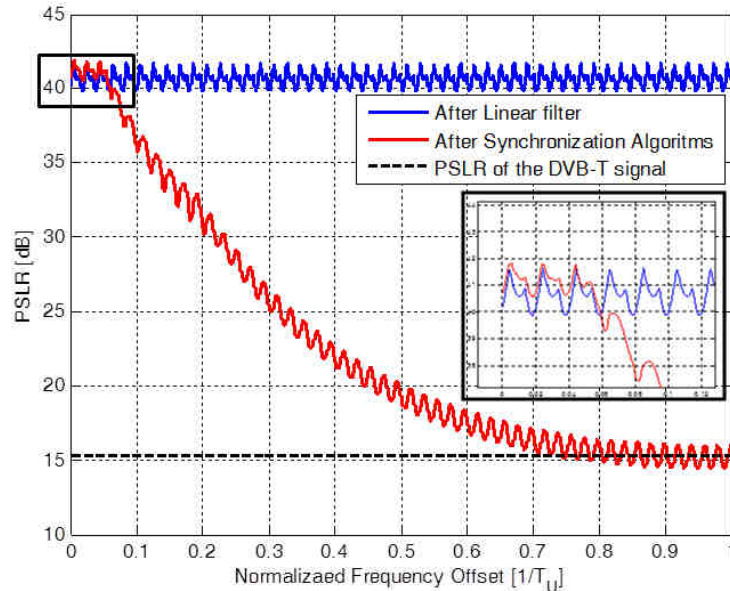
La Figura 37 mostra l'impatto dell'errore di sincronizzazione in tempo rispetto al PSLR (Peak to Side Lobe Ratio) della funzione d'ambiguità; la curva tratteggiata rappresenta il PSLR valutato sulla funzione d'ambiguità originale mentre la curva continua di colore rosso indica le prestazioni, in termini di PSLR, ottenute dopo l'applicazione della tecnica per il controllo dei lobi noti in letteratura. La curva blu è ottenuta attraverso la tecnica descritta in 3.3.3. Come atteso, applicando gli approcci presenti in letteratura che si basano sugli algoritmi di sincronizzazione, il PSLR potrebbe subire una degradazione delle prestazioni all'aumentare dell'errore in tempo rispetto alla stima dell'indice d'inizio del frame OFDM.



**Figura 37 – Effetto dell'errore di sincronizzazione in tempo sul rapporto picco a lobi laterali della funzione d'ambiguità**

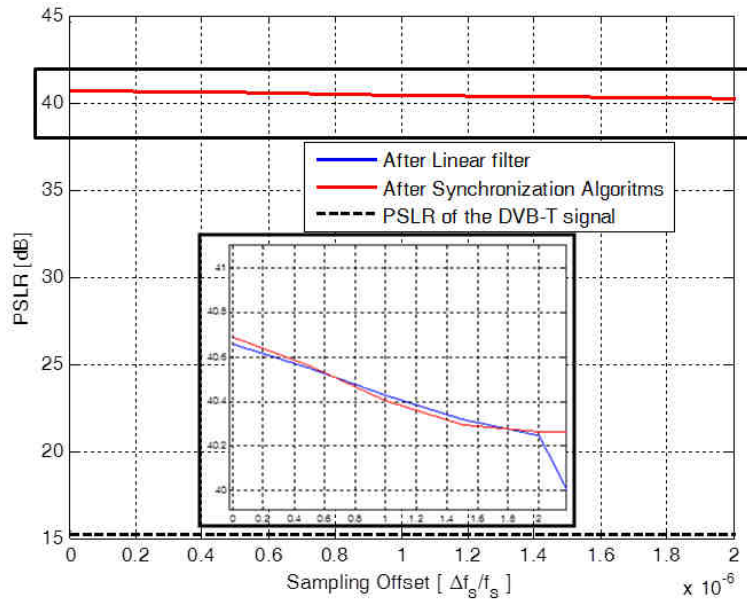
In questo caso specifico le prestazioni iniziano a peggiorare già per valori di errore inferiore all'intervallo di guardia. Nei casi pratici di interesse però lo stimatore ML garantisce un

errore tale da ottenere un PSLR sempre dell'ordine di 40 dB. Invece, le prestazioni ottenute con la tecnica lineare risultano costanti all'aumentare dell'errore in tempo grazie all'uso di segnali di durata pari ad multipli interi di un super-simbolo.



**Figura 38 – Effetto dell'errore di sincronizzazione in frequenza sul PSLR**

La Figura 38 mostra il PSLR in termini di errore di sincronizzazione in frequenza. Come atteso, la figura mostra che le prestazioni degli algoritmi noti decrescono rapidamente anche per valori molto piccoli d'errore. Inoltre, quando l'errore è uguale o maggiore della spaziatura fra le sottoportanti, il PSLR è confrontabile con il livello dei lobi della funzione d'ambiguità rappresentato dalla linea tratteggiata. Questo risultato è atteso in quanto, considerando errori in frequenza dell'ordine delle sotto-portanti, lo spettro del segnale non è allineato con gli indici definiti dallo standard e quindi tutte le sotto-portanti, e i relativi contenuti, sono spostati in frequenza proporzionalmente all'errore. Dunque, lo stimatore ML da solo non riesce a garantire delle buone prestazioni degli algoritmi per il controllo dei lobi ma è indispensabile anche l'utilizzo degli algoritmi post-FFT. Invece, le prestazioni ottenute con la tecnica lineare proposta continuano ad essere costanti rispetto all'errore in frequenza considerato.



**Figura 39 – Effetto dell’errore sulla frequenza di campionamento**

In Figura 39 è mostrato l’effetto di un piccolo errore sulla frequenza di campionamento, quantificato come  $\frac{f_s' - f_s}{f_s}$  dove  $f_s'$  rappresenta la frequenza realmente utilizzata mentre  $f_s$  quella nominale definita dallo standard, sulle prestazioni del filtro per il controllo dei lobi in termini di PSLR. Come si nota dalla figura, queste tecniche non mostrano degradazioni di prestazioni rispetto all’errore sulla frequenza di campionamento; infatti, nei casi pratici di interesse, i valori di questo errore sono dell’ordine di  $\pm 1\text{ppm}$  ( $10^{-6}$ ).

In sintesi, il confronto tra le curve proposte mostra che l’utilizzo della tecnica lineare è più robusta, rispetto agli errori di sincronizzazione, degli approcci noti in letteratura che dipendono anche dall’accuratezza con cui si valutano i valori di sincronismo.

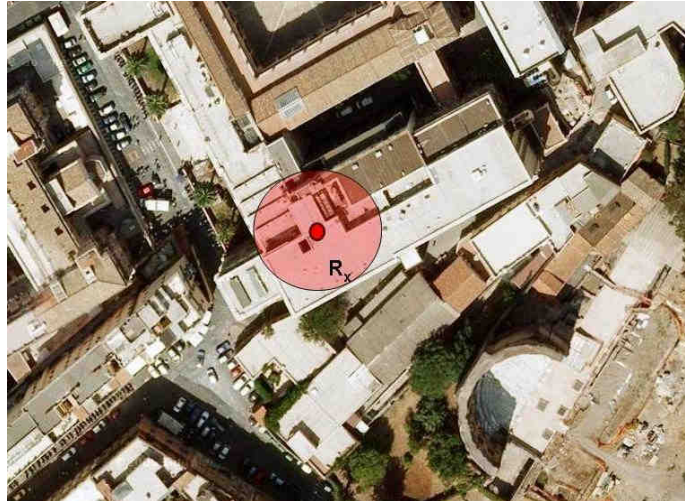
Inoltre, visto che la tecnica lineare consente di ottenere le medesime prestazioni in qualsiasi condizione non richiedendo l’uso di algoritmi di sincronizzazione, è possibile semplificare la complessità dell’intero sistema ricevente e il processing del segnale.

### 3.5 Risultati sperimentali

#### 3.5.1 Acquisizione dati DVB-T

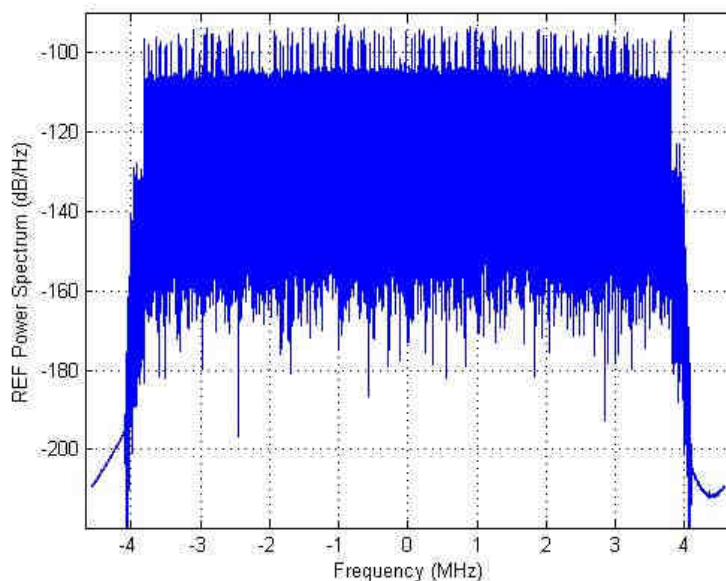
I dati utilizzati in questa sezione sono stati acquisiti con il sistema descritto nel paragrafo 2.3, in data 24 Febbraio 2011.

Il ricevitore DVB-T è posto sul tetto della nostra facoltà (via Eudossiana 18, 00184 Roma), come mostrato in Figura 40. Il canale di riferimento acquisito ha una frequenza portante pari a 754 MHz, una modalità di trasmissione 8K e un intervallo di guardia pari a 1/4 della durata della parte utile del simbolo OFDM; nel caso specifico la durata dell'intervallo di guardia è  $28\mu\text{s}$  mentre quella della parte utile è  $896\mu\text{s}$ .



**Figura 40 - Posizione del ricevitore DVB-T e DVB-SH**

I segnali acquisiti sono stati campionati con una frequenza pari a 12.8 MHz che consente di ottenere la frequenza nominale (64/7 MHz) attraverso un'operazione di upsampling con un fattore pari a 5 e, dopo l'operazione di filtraggio, un downsampling con un fattore pari a 7. Inoltre, si hanno a disposizione due versioni di dati ottenuti con e senza l'ausilio del segnale GPS (segnale di riferimento dato dal Thunderbolt® E GPS Disciplined Clock della Trimble) per generare i segnali di riferimento.



**Figura 41 – Spettro del segnale DVB-T**

In Figura 41 riportiamo lo spettro di densità di potenza del segnale DVB-T in banda base dopo il filtraggio digitale.

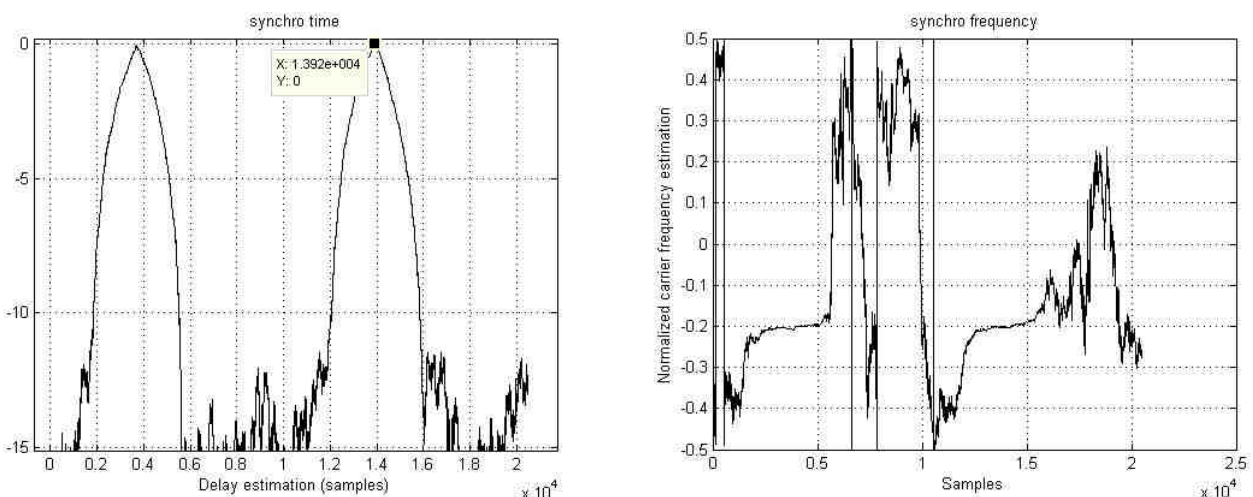
### 3.5.2 Algoritmi di sincronizzazione

Gli algoritmi di sincronizzazione descritti sono stati applicati al set di dati presentati in precedenza e, i risultati numerici, sono riportati in Tabella 12.

**Tabella 12 – Risultati numerici algoritmi di sincronizzazione**

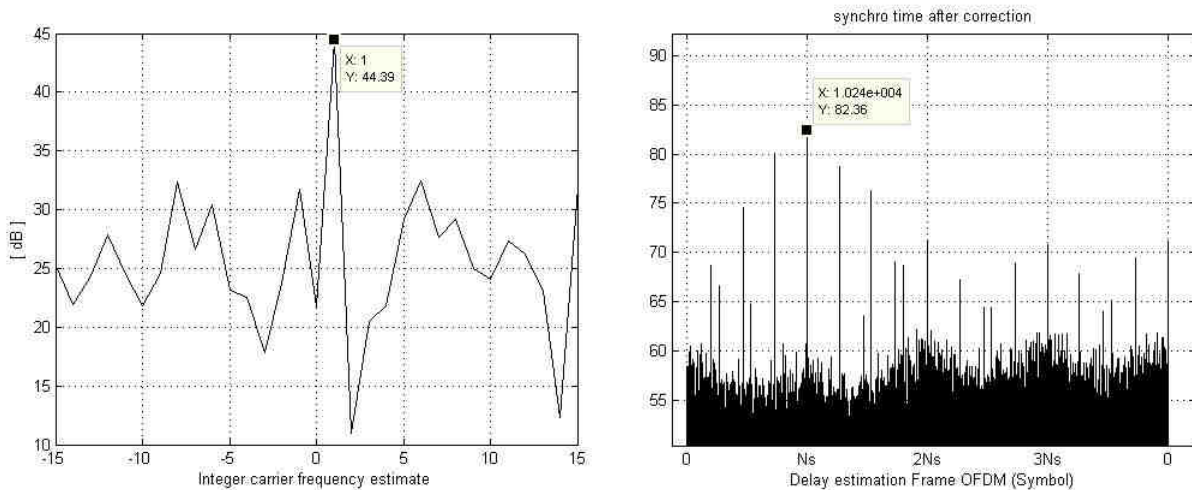
Segnale DVB-T (No Phased locked)		Value
<b>ML Estimator</b>	Stima del ritardo ( $\Delta\tau$ )	0.000403 s
	Stima dell'offset di frequenza ( $\Delta\phi$ )	-230.328 Hz
<b>Fine Synchronization</b>	Stima dell'offset di frequenza intero ( $\Delta\phi_{Int}$ )	1116.071 Hz
	Stima della frequenza di campionamento ( $\Delta\xi$ )	9143009 Hz
	Stima dell'offset di frequenza residuo ( $\Delta\phi_{Res}$ )	3.022 Hz
Segnale DVB-T (Phased locked)		Value
<b>ML Estimator</b>	Stima del ritardo ( $\Delta\tau$ )	0.001099 s
	Stima dell'offset di frequenza ( $\Delta\phi$ )	1.796 Hz
<b>Fine Synchronization</b>	Stima dell'offset di frequenza intero ( $\Delta\phi_{Int}$ )	0 Hz
	Stima della frequenza di campionamento ( $\Delta\xi$ )	9142856 Hz
	Stima dell'offset di frequenza residuo ( $\Delta\phi_{Res}$ )	-2.490 Hz

In Figura 42 si riportano i segnali generati dallo stimatore ML che utilizza il segnale DVB-T non agganciato in fase; in questo caso viene considerata una finestra di osservazione, in accordo alla Figura 24, doppia rispetto alla finestra minima necessaria e quindi l'algoritmo produce due massimi locali che indicano la stima dell'inizio di due simboli OFDM consecutivi e, in corrispondenza di ognuno, una stima grossolana dell'offset di frequenza.



**Figura 42 – Andamento dei segnali generati dallo stimatore ML che utilizza segnali non agganciati in fase**

Dopo la parziale correzione del flusso di dati rispetto alle stime effettuate, vengono applicati i successivi algoritmi (andamenti dei segnali riportati in Figura 43) con cui si stima l'offset di frequenza intero multiplo della spaziatura tra le sottoportanti, il ritardo rispetto all'inizio del Frame OFDM, l'offset di frequenza di campionamento rispetto al valore nominale. In questo caso specifico la stima dell'intero multiplo della spaziatura è pari ad 1 che significa un errore in frequenza di 1116 Hz mentre l'allineamento rispetto al frame può essere ottenuto dopo un ulteriore ritardo pari a un simbolo OFDM.

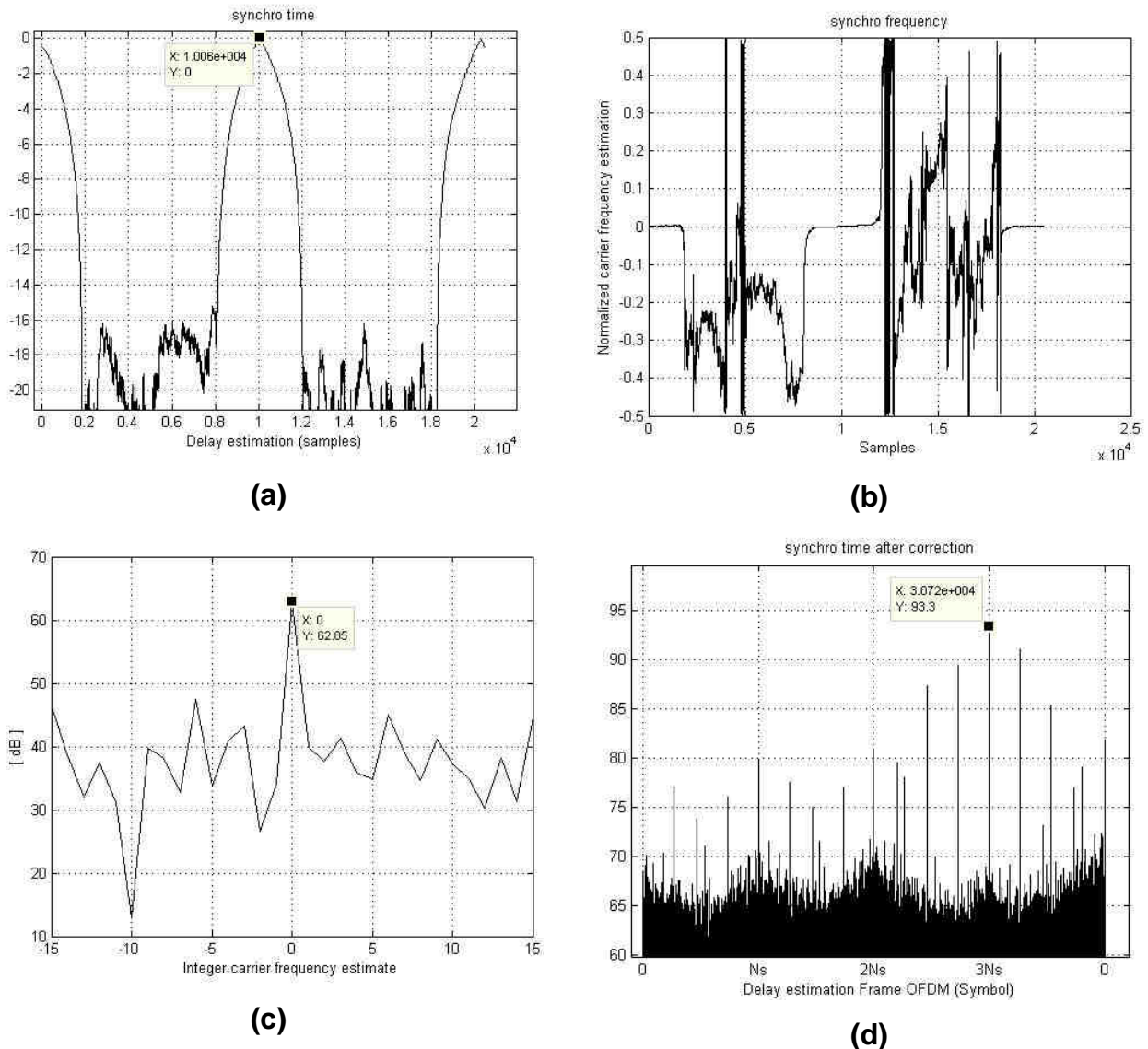


**Figura 43 - Andamento dei segnali generati dallo stimatore post-FFT che utilizza segnali non agganciati in fase**

L'utilizzo del segnale GPS come riferimento comune, con cui generare il segnale di oscillatore locale e il clock di riferimento per il campionamento, implica che l'unica incertezza è legata al tempo di arrivo del simbolo OFDM non noto al ricevitore.

Quanto detto giustifica i risultati ottenuti applicando gli algoritmi di sincronizzazione ad un segnale agganciato in fase (Figura 44 (b) e (c)); come atteso, la stima dell'offset di frequenza degli stimatori Pre-FFT e Post-FFT è pari a 0.

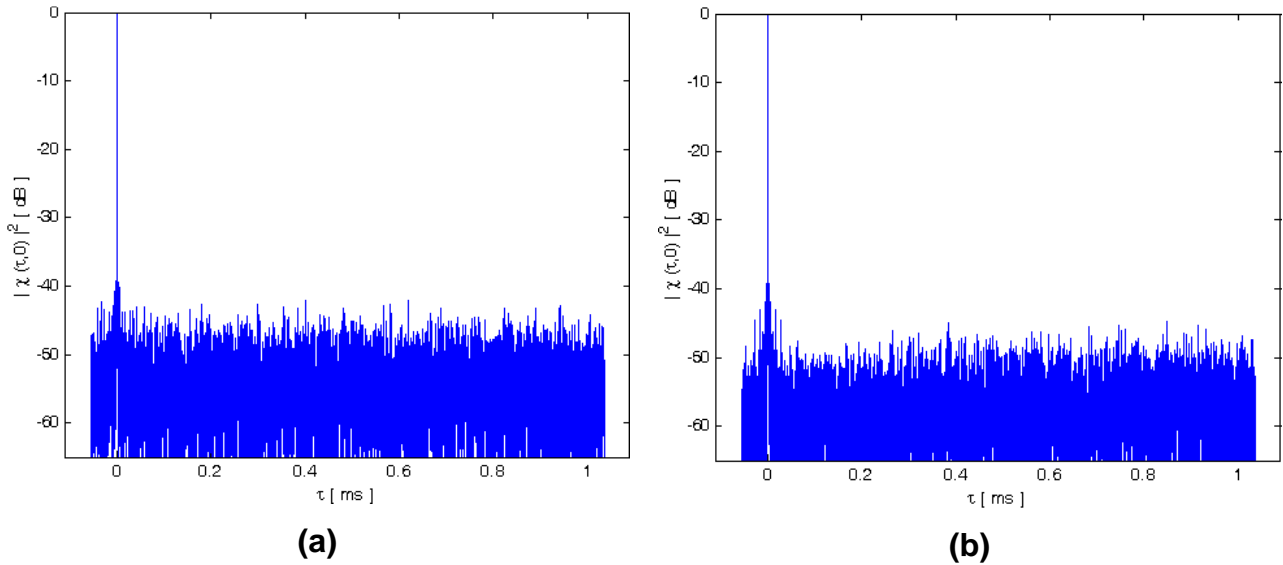




**Figura 44 - Andamento dei segnali generati dagli stimatori Pre-FFT e post-FFT che utilizza segnali agganciati in fase**

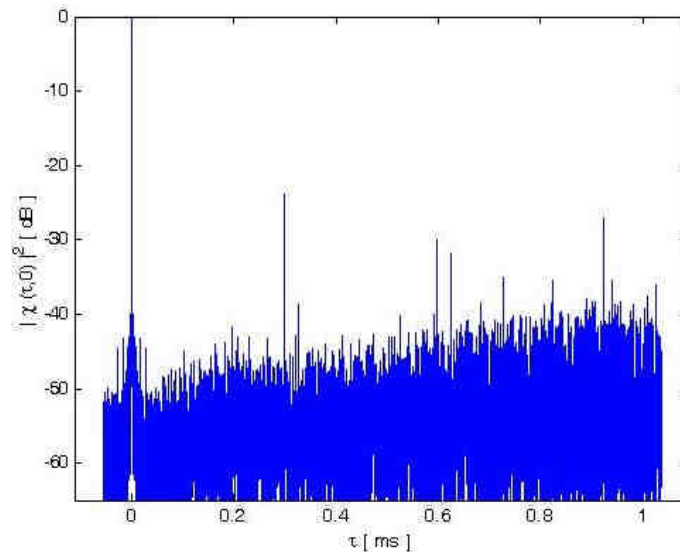
### 3.5.3 Filtraggio lineare

Nel seguito si riporta il confronto tra le tecniche per il controllo dei lobi, descritte precedentemente, applicate ai dati reali; in particolare, nella Figura 45 (a) e (b), si mostra il confronto tra l'autocorrelazione ottenuta con la tecnica riportata in Figura 21 e quella descritta nel paragrafo 3.1.



**Figura 45 – Auto-correlazione del segnale DVB-T (dati reali) dopo l'applicazione del filtraggio lineare (a) e la tecnica presente in letteratura (b)**

Come è evidente, le due tecniche hanno le stesse prestazioni in termini di PSLR ma il risultato mostrato in Figura 45 (b) è stato ottenuto dopo l'applicazione di tutti gli algoritmi di sincronizzazione (risultati riportati in Tabella 12).



**Figura 46 – Auto-correlazione utilizzando solo lo stimatore ML**

Inoltre, come si vede dalla Figura 46, poiché l'offset di frequenza è maggiore dell'intero multiplo delle sotto-portanti, se non si utilizza il segnale GPS come riferimento comune con cui generare il segnale di oscillatore locale e il clock di riferimento, anche l'applicazione del solo stimatore ML non può garantire un controllo efficiente dei lobi. Questo risultato è in linea con i risultati teorici mostrati in precedenza.

## 4 Equalizzazione del segnale DVB

Con riferimento agli obiettivi di questo lavoro di tesi descritti nel paragrafo 1.1, in questo capitolo si descrive un algoritmo che consente la stima della funzione di trasferimento del canale. In particolare, dopo una sezione dedicata all'introduzione e alla descrizione dello schema di processing, verranno descritte nel dettaglio le singole operazioni che l'algoritmo Least Mean Square (LMS) effettua per stimare il canale. L'analisi delle prestazioni del filtro sono descritte nel paragrafo 4.3 in funzione dei diversi parametri del filtro stesso. Nei paragrafi 4.4 e 4.5 si riportano l'analisi delle prestazioni di un radar passivo che utilizzi l'equalizzatore per la stima della funzione di trasferimento. Il paragrafo finale è dedicato ai risultati sperimentali ottenuti applicando l'algoritmo ai dati reali.

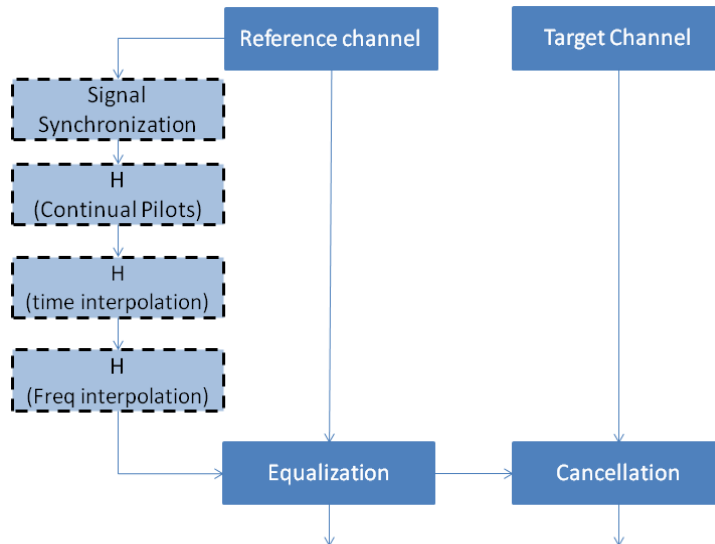
### 4.1 Introduzione

A causa della geometria del sistema passivo bistatico, gli echi dei target presenti all'interno della scena sotto osservazione potrebbero essere mascherati da diversi effetti; il contributo del segnale diretto ricevuto attraverso i lobi laterali dell'antenna ricevente (che potrebbe essere più grande dell'eco di clutter), forti echi di clutter / multipath e anche echi provenienti da altri target che si trovano vicino alla posizione del ricevitore.

Lo schema di processing di elaborazione di un PBR ipotizza la disponibilità di un segnale di riferimento "abbastanza pulito" (una copia esatta del segnale trasmesso) per essere usato sia per lo stadio di cancellazione che per il filtraggio adattato al segnale di sorveglianza. Lo stadio di cancellazione del segnale diretto e degli echi di clutter / multipath diventa una questione cruciale per il rilevamento di target. Tuttavia, se il segnale di riferimento è affetto da multipath, la capacità di cancellazione degli algoritmi presenti in letteratura potrebbe essere fortemente ridotta così come il rilevamento di target che potrebbe essere fortemente limitato.

Con riferimento ad una particolare tecnica che consente l'equalizzazione del segnale DVB (l'algoritmo Least Mean Square (LMS)), in Figura 47 si mostra lo schema di processing utilizzato per equalizzare il segnale di riferimento. In particolare, dopo l'applicazione degli algoritmi di sincronizzazione, la stima del canale può essere realizzata attraverso tre passaggi fondamentali: (i) la valutazione della funzione di trasferimento attraverso la conoscenza delle portanti pilota continue; (ii) l'interpolazione nel dominio del tempo della

funzione di trasferimento trovata precedentemente utilizzando la conoscenza delle portanti piloti scattered; (iii) la stima del canale su tutte le sotto-portanti (comprese quelle che contengono il flusso dati video) attraverso l'interpolazione nel dominio della frequenza. La stima del canale realizzata servirà per equalizzare il segnale di riferimento che verrà utilizzato sia nel processo di cancellazione che nel calcolo della 2D-CCF.



**Figura 47 – Schema di processing utilizzando l’equalizzatore**

Nel seguito verrà mostrata l’analisi relativa all’impatto del multipath sul radar passivo e il relativo miglioramento prestazionale indotto dall’uso dell’equalizzatore.

## 4.2 Algoritmo di equalizzazione Least Mean Square (LMS)

Il modello per la descrizione del canale, utilizzando un ricevitore in posizione fissa, può essere descritto dalla relazione Eq. 6 e Eq. 7.

All’interno dello standard (vedi [66]) vengono forniti i parametri che caratterizzano un canale medio; i valori di  $\rho_i$ ,  $\phi_i$  e  $\tau_i$ , che rappresentano i contributi di ampiezza, fase e ritardo relativi all’ $i$ -esimo raggio ricevuto, sono contenuti in Tabella 13 e possono essere utilizzati come base per ricavare scenari semplificati costituiti da uno o più raggi.

Di seguito descriveremo l’algoritmo utilizzato per la stima del canale considerando i segnali DVB. Sfruttando la conoscenza dei valori dei simboli noti in corrispondenza delle portanti pilota continue e sparse, l’algoritmo consente di stimare la funzione di trasferimento del canale di riferimento attraverso l’interpolazione LMS (Least Mean Square) sia in tempo che in frequenza grazie alla conoscenza del canale attraverso i piloti.

**Tabella 13 – Ampiezze, fasi e ritardi relative all descrizione del canale medio fornito dallo standard**

$i$	$\rho_i$	$\tau_i$ [ $\mu\text{s}$ ]	$\theta_i$ [rad]
1	0,057 662	1,003 019	4,855 121
2	0,176 809	5,422 091	3,419 109
3	0,407 163	0,518 650	5,864 470
4	0,303 585	2,751 772	2,215 894
5	0,258 782	0,602 895	3,758 058
6	0,061 831	1,016 585	5,430 202
7	0,150 340	0,143 556	3,952 093
8	0,051 534	0,153 832	1,093 586
9	0,185 074	3,324 866	5,775 198
10	0,400 967	1,935 570	0,154 459
11	0,295 723	0,429 948	5,928 383
12	0,350 825	3,228 872	3,053 023
13	0,262 909	0,848 831	0,628 578
14	0,225 894	0,073 883	2,128 544
15	0,170 996	0,203 952	1,099 463
16	0,149 723	0,194 207	3,462 951
17	0,240 140	0,924 450	3,664 773
18	0,116 587	1,381 320	2,833 799
19	0,221 155	0,640 512	3,334 290
20	0,259 730	1,368 671	0,393 889

#### 4.2.1 Utilizzo dei piloti continui

La prima operazione che viene effettuata alla ricezione del segnale è, dopo la stima del sincronismo, la valutazione della funzione di trasferimento in corrispondenza della posizione delle portanti contenenti i piloti continui (insieme delle portanti  $k_{PC}$  sono ottenute dallo standard e riportate in Tabella 3), in Eq. 45

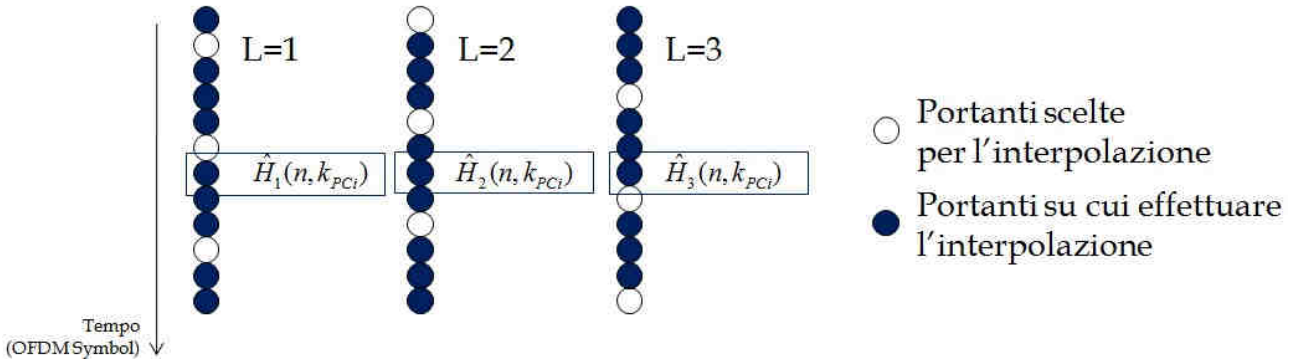
$$H(n, k_{PC}) = \frac{Y(n, k_{PC})}{X(n, k_{PC})} \quad \text{Eq. 45}$$

Dove  $Y(n, k_{PC})$  rappresenta l'insieme dei simboli complessi ricevuti all'istante  $n$  in corrispondenza delle posizioni dei piloti continui e  $X(n, k_{PC})$  l'insieme dei simboli pilota trasmessi e noti al ricevitore.

#### 4.2.2 Interpolazione nel dominio del tempo

Partendo dalla quantità  $H(n, k_{PC})$  di Eq. 45, si effettua l'interpolazione nel dominio del tempo sfruttando la conoscenza della posizione dei piloti sparsi. La distribuzione regolare dei piloti sparsi assicura che questi siano trasmessi, ogni quattro simboli OFDM (periodicità 4), nella stessa posizione. La distanza temporale tra una portante pilota e non

pilota è al massimo pari a 3, quindi la stima di canale si ottiene filtrando le portanti pilota del tono sotto analisi utilizzando un set ( $L$ ) di tre differenti coefficienti  $W_1, W_2, W_3$ , come raffigurato schematicamente in Figura 48.



**Figura 48 – Principio funzionamento dell'algoritmo di stima temporale**

Le tre differenti stime del canale possono essere valutate attraverso la relazione Eq. 46.

$$\hat{H}_L(n, k_{PCi}) = \sum_{m=-M_1+1}^{M_2} W_L(n-1, m) H(n-L+4m, k_{PCi}) \quad \text{Eq. 46}$$

La lunghezza del filtro è pari a  $4(M_1 + M_2)$  simboli OFDM, dove  $4M_1$  e  $4M_2$  rappresentano il numero di simboli utilizzati rispettivamente prima e dopo l'istante  $n$  da stimare.

Il passaggio successivo consiste nella valutazione dell'errore di stima, riportato in Eq. 47, che consentirà di effettuare l'aggiornamento dei coefficienti, come riportato in Eq. 48.

$$E_L(n, k_{PCi}) = H(n, k_{PCi}) - \hat{H}_L(n, k_{PCi}) \quad \text{Eq. 47}$$

$$W_L(n, m) = W_L(n-1, m) + \mu \sum_{i=1}^{PC} E_L^*(n, k_{PCi}) H(n-L+4m, k_{PCi}) \quad \text{Eq. 48}$$

Per l'aggiornamento dei coefficienti viene usata la variabile  $\mu$  che rappresenta il passo di adattamento del filtro interpolatore nel tempo.

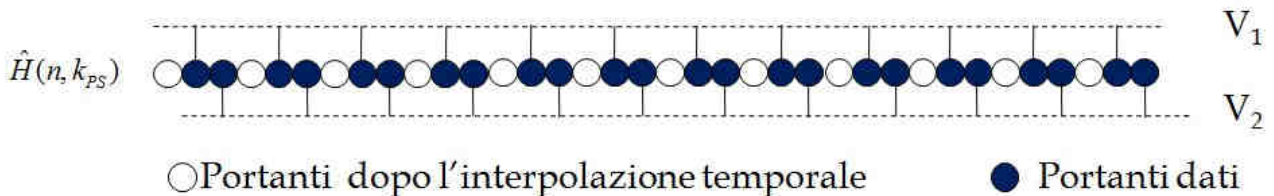
Dopo l'aggiornamento dei coefficienti si ricava la stima del canale su tutte le sotto-portanti contenenti i piloti continui e sparsi, in Eq. 49.

$$\hat{H}(n, k_{PSi}) = \sum_{m=-M_1+1}^{M_2} W_L(n, m) H(n-L+4m, k_{PSi}) \quad \text{Eq. 49}$$

Si noti che l'insieme delle portanti contenenti i piloti continui e sparsi all'istante  $n$  ( $K_{PSi}$ ) varia in funzione del simbolo OFDM sotto analisi; questo insieme può essere ricavato utilizzando la relazione Eq. 12.

### 4.2.3 Interpolazione nel dominio della frequenza

L'interpolazione nel dominio della frequenza sfrutta la conoscenza del canale, determinato in precedenza ( $H(n, k_{PSi})$ ) attraverso l'interpolazione nel dominio del tempo, per derivare la stima di canale per tutte quelle sottoportanti che trasportano informazione legata al flusso audio/video.



**Figura 49 - Principio funzionamento dell'algoritmo di stima nel dominio in frequenza**

Seguendo un approccio simile a quello utilizzato per l'interpolazione nel dominio del tempo, nel dominio della frequenza sono definiti due distinti filtri interpolatori  $V_l$  con  $l = 1, 2$ . Il filtro interpolatore in frequenza (Eq. 50) è progettato per avere una lunghezza pari a  $3(J_1 + J_2)$ , dove  $3J_1$  e  $3J_2$  rappresentano gli indici delle sotto-portanti utilizzati rispettivamente prima e dopo l'indice della sottoportante in frequenza sotto analisi.

$$\hat{H}(n, k) = \sum_{j=-J_1+1}^{J_2} V_l(n-1, j) \hat{H}(n, k - I + 3j) \quad \text{Eq. 50}$$

La quantità  $V_l(n-1, j)$  in Eq. 50 rappresenta il  $j$ -esimo coefficiente dell' $l$ -esimo filtro ( $l=1, 2$ ) aggiornato durante il precedente simbolo OFDM ( $n-1$ ).

Considerando la relazione  $k=3K+l$ , si selezionano esclusivamente le sottoportanti dati interne al simbolo OFDM sotto analisi che vengono stimate grazie alla conoscenza delle sotto-portanti calcolate durante l'interpolazione nel tempo.

In assenza di simboli pilota, il calcolo dell'errore di stima necessario per l'aggiornamento dei coefficienti si basa su un criterio di decisione diretta. Dopo aver determinato la stima della funzione di trasferimento di canale  $\hat{H}(n, k)$  è possibile ricavare la stima del simbolo trasmesso attraverso una decisione Hard fatta sullo stesso simbolo trasmesso (Eq. 51).

$$\hat{X}(n, k) = H_D \left\{ \frac{Y(n, k)}{\hat{H}(n, k)} \right\} \Rightarrow H(n, k) = \frac{Y(n, k)}{\hat{X}(n, k)} \quad \text{Eq. 51}$$

Assumendo la decisione hard affidabile, il simbolo stimato in Eq. 51 può essere usato come simbolo pilota in modo tale che la funzione di trasferimento possa essere ricalcolata.

A questo punto, come fatto in precedenza, si valuterà l'errore di stima (Eq. 52) che consentirà di effettuare l'aggiornamento dei coefficienti (Eq. 53) attraverso il parametro  $\gamma$  definito come il passo di adattamento del filtro interpolatore in frequenza.

$$D(n, k) = H(n, k) - \hat{H}(n, k) \quad \text{Eq. 52}$$

$$V_l(n, j) = V_l(n-1, j) + \gamma \sum D^*(n, k) H(n, k - L + 3j) \quad \text{Eq. 53}$$

### 4.3 Analisi equalizzatore LMS

L'analisi che segue è stata effettuata considerando:

- Lunghezza dei filtri interpolatori in tempo e frequenza fissa in particolare  $M_1=M_2=J_1=J_2=1$ ;
- Passi d'interpolazione fissi e pari a  $\mu = 5 \cdot 10^{-8}$ ;  $\gamma = 5 \cdot 10^{-8}$ ;
- I coefficienti dei filtri inizializzati a valori fissi e pari a  $W_1=W_2=W_3=V_1=V_2=0.5$ ;
- Canale composto dal raggio diretto e una replica, nel seguito 1-RAGGIO, caratterizzata dall'ampiezza pari a 0.41, ritardo pari a 1.37  $\mu\text{s}$  e fase pari a 4.86 radianti;
- Canale composto dal raggio diretto e due repliche (2-RAGGIO, caratterizzate dalle ampiezze pari a [0.41 0.25], ritardi pari a [0.37 1.00]  $\mu\text{s}$  e fasi uguali [4.86 1.62] radianti;
- Canale composto dal raggio diretto e tre repliche, nel seguito 3-RAGGIO, caratterizzate dalle ampiezze pari a [0.41 0.25 0.01], ritardi pari a [0.37 2.67 0.12]  $\mu\text{s}$  e fasi pari a [4.86 1.61 0.11] radianti.

In Figura 50 si riporta il modulo della funzione di trasferimento, utilizzata in questo esempio, relativa al canale 2-RAGGIO e la stima (in rosso) effettuata attraverso l'uso dell'algoritmo LMS.



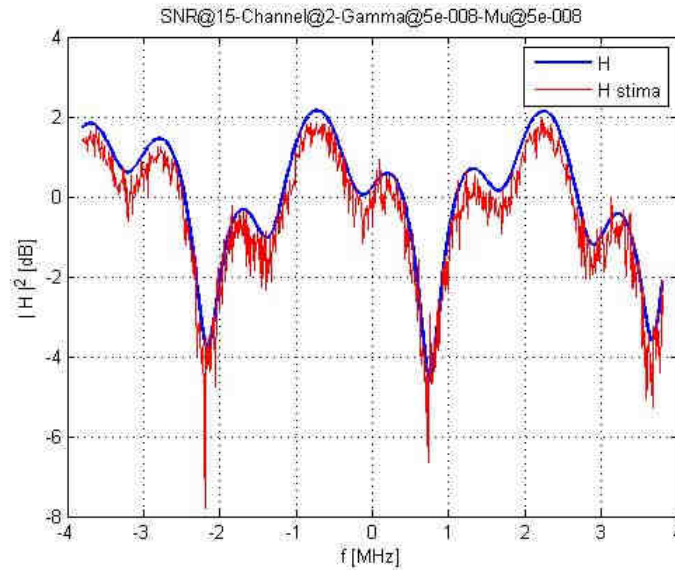


Figura 50 – Modulo della funzione di trasferimento

#### 4.3.1 Valutazione dell'errore di stima commesso

Per effettuare la valutazione dell'errore di stima commesso dall'equalizzatore si fa riferimento all'errore percentuale valutato tra la funzione di trasferimento stimata dall'algorithmo è quella del canale, come riportato in Eq. 54.

$$E = \frac{1}{f} \sum_{i=1}^f \frac{H_c(i) - \tilde{H}_s(i)}{H_c(i)} \quad \text{Eq. 54}$$

Nello specifico,  $H_c$  si riferisce alla funzione di trasferimento del canale mentre  $H_s$  indica quella stimata dall'algorithmo LMS. Come è evidente dall'espressione precedente, l'errore commesso viene valutato in media su tutte le portanti (piloti continui, piloti sparsi e dati) per quantificare la capacità di inseguimento di canale da parte dell'algorithmo di interpolazione. In Figura 51 si mostra l'errore in percentuale commesso rispetto al tempo, considerando il canale ideale e le tre diverse configurazioni viste precedentemente.

Come si nota, l'errore commesso è costante al variare del tempo e cresce, in media, all'aumentare della complessità del canale, mentre in Figura 52 si riporta l'andamento dell'errore in percentuale rispetto a diversi valori di SNR.

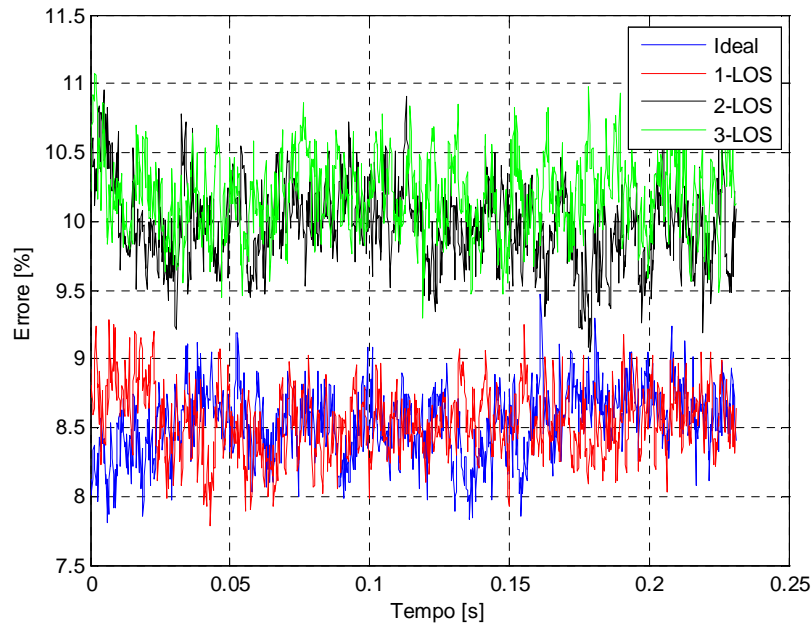


Figura 51 – Errore in percentuale al variare del tempo

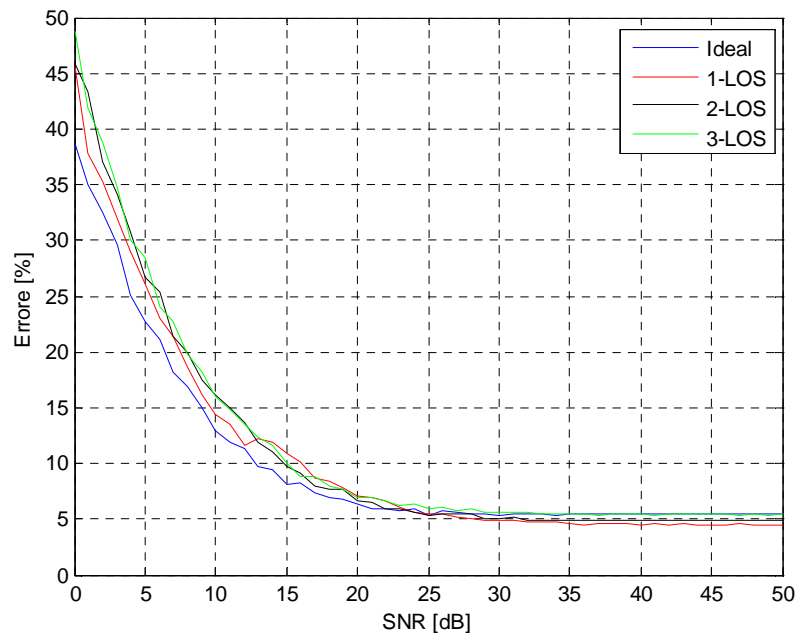
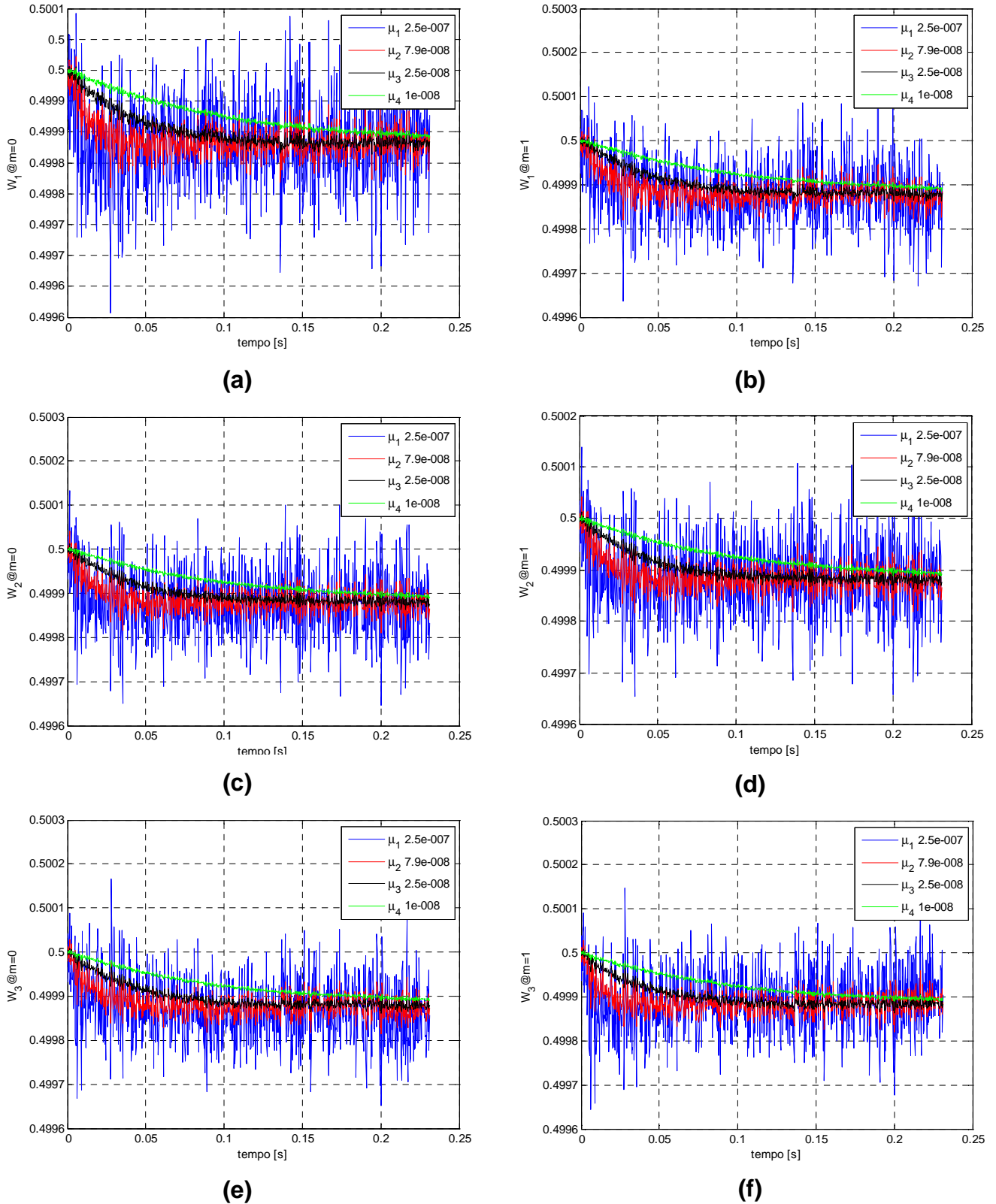


Figura 52 – Errore percentuale al variare del SNR

#### 4.3.2 Variazione del parametro $\mu$ e del del parametro $\gamma$

I valori dei parametri  $\mu$  e  $\gamma$ , in Eq. 48 e Eq. 53, non sono fissi e devono essere determinati a seconda del contesto in cui si applica la stima.

In questo paragrafo verrà studiato l'andamento dei coefficienti  $W_1, W_2$  e  $W_3$  nel tempo e dei coefficienti  $V_1$  e  $V_2$  in frequenza al variare dei passi d'interpolazione.

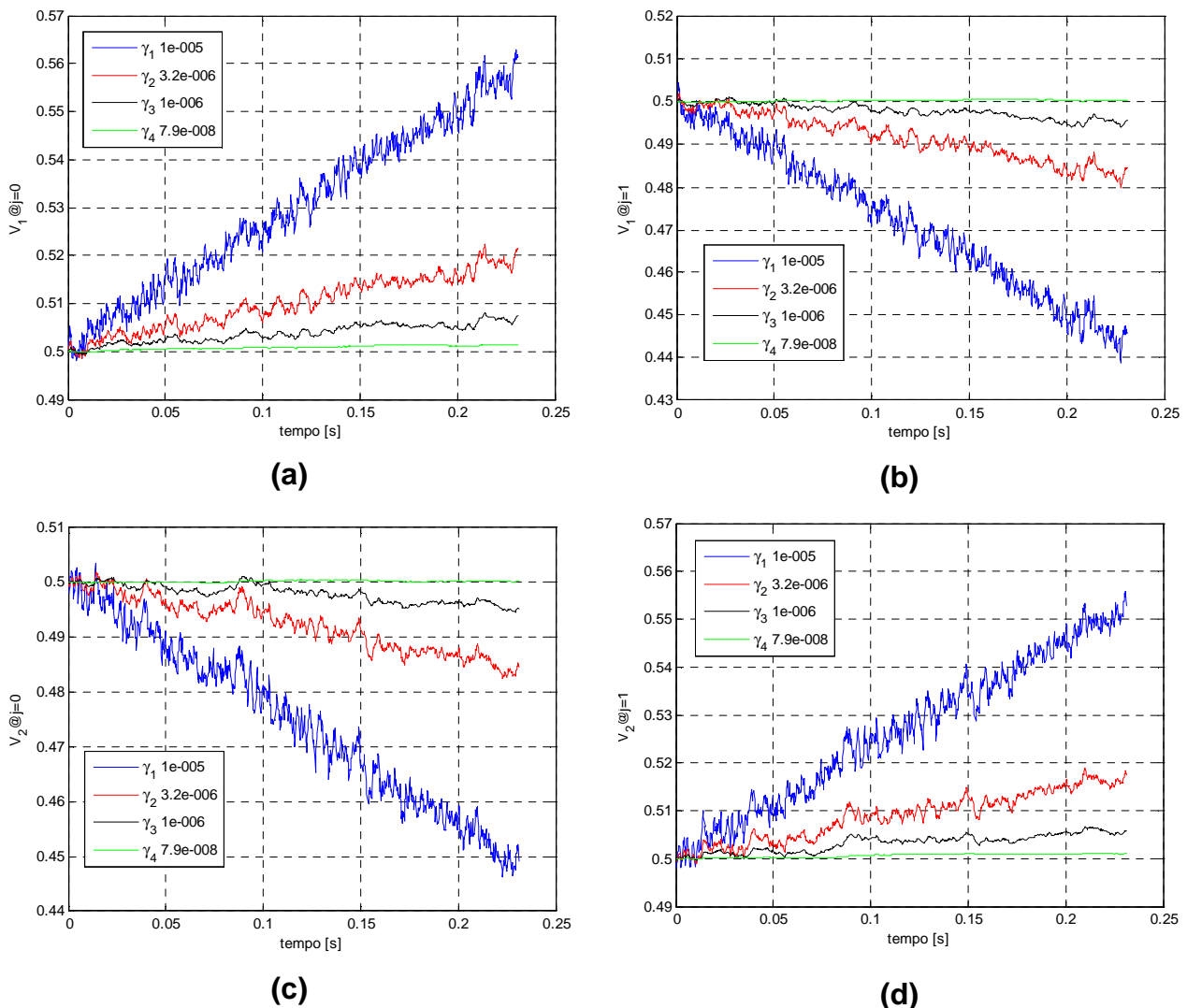


**Figura 53 – Andamenti in tempo dei coefficienti  $W_1$  (a e b),  $W_2$  (c e d) e  $W_3$  (e e f) per diversi valori del parametro  $\mu$  e fissando  $\gamma = 7.9e-008$**

Il parametro  $\mu$  è alla base della velocità di convergenza dell'algoritmo LMS. Quando  $\mu$  è basso, il tasso di convergenza è altrettanto basso e l'errore di stima è alto; al contrario, quando  $\mu$  è più alto, l'algoritmo converge più velocemente al valore ottimo.

Tuttavia, se tale parametro assume valori troppo alti, l'algoritmo diventa più instabile ed i risultati ottenuti possono essere errati.

In Figura 53 sono rappresentati gli andamenti dei coefficienti interpolatori in tempo, rispettivamente per  $L = 1,2,3$  ed  $m = 0,1$  considerando diversi valori del parametro  $\mu$  per dimostrare esattamente ciò che è stato appena detto. Per  $\mu = \mu_{\max} = 2.5 \cdot 10^{-7}$  (curva blu), il filtro interpolatore nel tempo converge dopo circa dopo 20 ms. Per  $\mu = \mu_{\min} = 1 \cdot 10^{-8}$  (curva verde) invece, il filtro impiega 200 ms.



**Figura 54 – Andamenti in tempo dei coefficienti  $V_1$ (a e b),  $V_2$ (c e d) per diversi valori del parametro  $\gamma$  e fissando  $\mu = 7.9 \cdot 10^{-008}$**

La maggiore rapidità di convergenza che si ottiene per valori grandi, si paga però con una perdita di stabilità nell'algoritmo che si traduce come una serie di oscillazioni, non trascurabili, intorno al valore di convergenza, caratteristica che si riduce gradualmente diminuendo il valore di  $\mu$ .

Un discorso analogo può essere fatto in relazione al parametro  $\gamma$  ed ai filtri  $V_1$  e  $V_2$ .

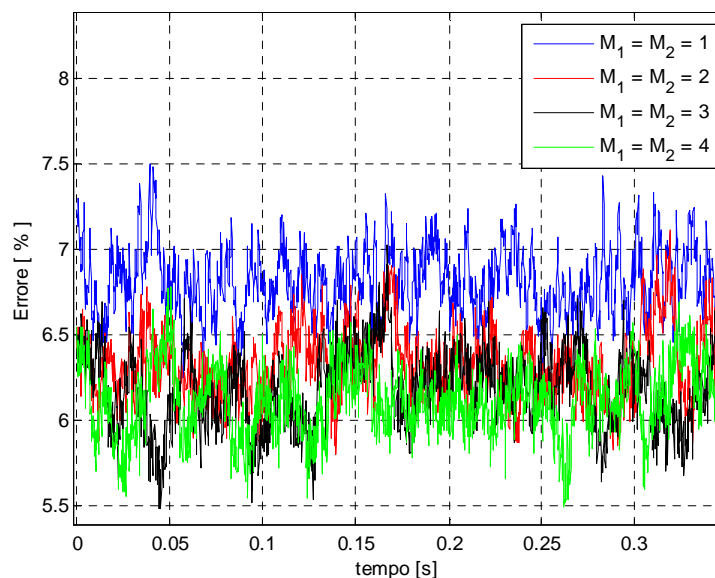
In Figura 54 sono rappresentati gli andamenti dei coefficienti interpolatori in frequenza, rispettivamente per  $l = 1, 2$  ed  $j = 0, 1$ .

I valori assunti dal suddetto filtro per  $l=1$  sono i complementi al valore ottimo (circa 0.5) rispetto a quelli calcolati per  $l = 2$ , per ogni  $\gamma$ .

Inoltre, isolando le singole curve, si osserva che per  $\gamma = \gamma_{\min} = 7.9 \cdot 10^{-8}$  la "forbice" generata dai due sottofiltri  $l = 1$  ed  $l = 2$  è decisamente più stretta di quella ottenuta per  $\gamma = \gamma_{\max} = 1 \cdot 10^{-5}$ , assumendo valori intermedi per i valori di  $\gamma$  compresi tra i due estremi.

### 4.3.3 Variazione dei parametro $M_1, M_2$

In questa sezione verranno analizzate le prestazioni dell'algoritmo al variare della "memoria", ovvero del numero di simboli utilizzati prima ( $4 \cdot M_1$ ) e dopo ( $4 \cdot M_2$ ) il simbolo da stimare nell'istante di tempo  $t$ . Ricordando l'espressione di Eq. 49, aumentando la memoria, ovvero intervallo temporale su cui effettuare l'interpolazione, le prestazioni migliorano.



**Figura 55 - Andamento dell'errore percentuale nel tempo al variare della memoria**

In Figura 55 si riporta l'andamento dell'errore in percentuale commesso rispetto al tempo per diversi valori di  $M_1$  e  $M_2$  che consente di verificare quanto appena affermato. Inoltre

valutando il valore medio dell'errore percentuale riportato nei tre casi si può apprezzare la leggera diminuzione dell'errore

- $M_1=M_2=1$  (Curva blu) 6,82%;
- $M_1=M_2=2$  (Curva rossa) 6,33%;
- $M_1=M_2=3$  (Curva nera) 6,19%;
- $M_1=M_2=4$  (Curva verde) 6,12%;

#### **4.4 Impatto del multipath sulle prestazioni del radar passivo basato su DVB-T**

In questo paragrafo verrà analizzato l'effetto negativo che il multipath, presente sul segnale di riferimento, apporta sulle prestazioni valutate in termini di CA (Clutter attenuation) ottenibile definita come la diminuzione della potenza del segnale tra l'ingresso e l'uscita del filtro cancellatore utilizzato.

La CA teorica ottenibile su dati simulati può essere valutata utilizzando la Eq. 55.

$$CA_{teo} = 10 \log_{10} \left( \frac{\sum_{i=1}^N |a_i|^2}{P_n} \right) \quad \text{con } N \text{ (numero di picchi di clutter presenti)}. \quad \text{Eq. 55}$$

L'algoritmo utilizzato per la cancellazione è denominato ECA (Extensive Cancellation Algorithm) ed è descritto dettagliatamente in [28]; tale algoritmo rimuove i contributi di disturbo dal segnale di sorveglianza dovuti al clutter ed alla percentuale di segnale diretto ricevuto sui lobi laterali dell'antenna di sorveglianza.

Il problema fondamentale dell'algoritmo ECA è che esso assume, per ipotesi, la disponibilità di un segnale diretto del canale di riferimento senza echi di multipath; Le relazioni analitiche implementate sono riportate in Eq. 2, Eq. 3 e Eq. 4 rispettivamente.

Cerchiamo, con l'esempio che segue, di valutare l'efficacia dell'algoritmo di cancellazione. Tale esempio verrà utilizzato anche per valutare il miglioramento prestazionale che si ottiene quando viene utilizzato un algoritmo di equalizzazione.

La Figura 56 (a) mostra lo scenario considerato composto da cinque diversi target (identificati dai quadrati) a diverse velocità e diversi range e il contributo di clutter (identificato con i diamanti). Nello specifico, con riferimento alle relazioni Eq. 6 e Eq. 7, il segnale di sorveglianza conterrà il contributo del segnale diretto con un CNR (Clutter Noise Ratio) di 30 dB e 4 picchi di clutter aggiuntivi che avranno un CNR pari a [10 5 10

10] dB e ritardi pari a [0.1094 1.0938 2.1875 3.2813]  $\mu$ s rispettivamente; i target invece avranno un SNR pari a [-5 -5 -15 -15 -15] e saranno identificati dalla coppia velocità [-250; 500; 100; 450; 0;] m/s e Range [7.32 6.56 32.81 34.22 33.47] km. Il segnale di riferimento invece avrà il DNR (Direct Noise Ratio) pari a 50 dB e diverse condizioni di Multipath tra quelle descritte e riportate nel paragrafo 4.3.

La Figura 56 (b) riporta la mappa delle rivelazioni (identificate con i punti bianchi) ottenuta applicando il filtro automatico CA-CFAR alla funzione 2D-CCF ottenuta considerando un segnale di riferimento non affetto da multipath e il segnale di sorveglianza non considerando né l'algoritmo di cancellazione né il controllo lobi laterali.

La Figura 56 (c) invece riporta la stessa mappa delle rivelazioni dove il segnale di riferimento non è affetto da multipath; il segnale di sorveglianza è stato ottenuto dopo l'applicazione del filtro ECA operante con 50 tappi (50 repliche del segnale di riferimento ritardate utilizzate per costruire il filtro cancellatore).

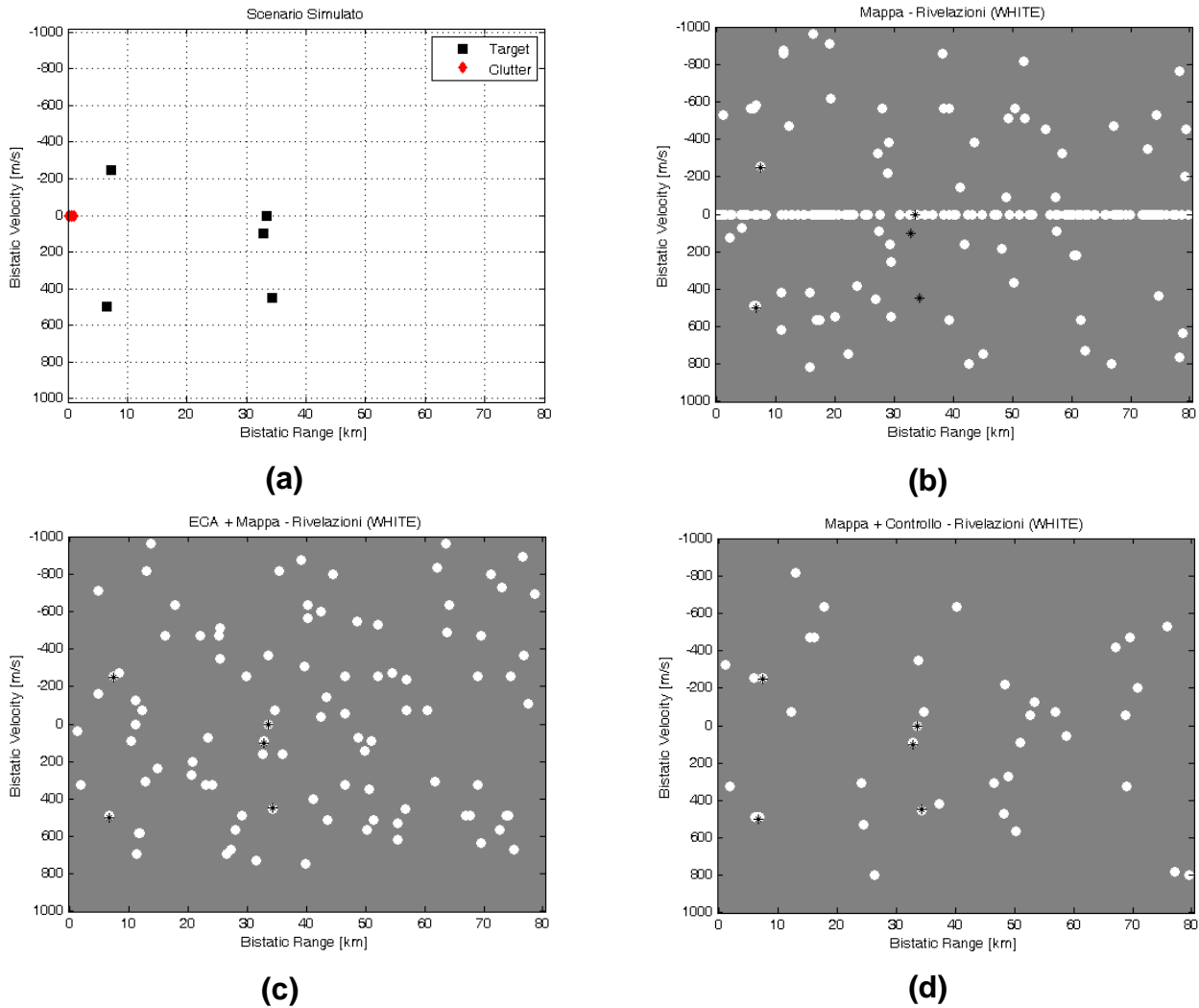
La Figura 56 (d) invece riporta la stessa mappa delle rivelazioni dove il segnale di riferimento non è affetto da multipath a cui è stato applicato il filtro per il controllo dei lobi laterali; il segnale di sorveglianza è stato ottenuto dopo l'applicazione del filtro ECA operante con 50 tappi (50 repliche del segnale di riferimento ritardate utilizzate per costruire il filtro cancellatore).

All'interno delle Figure 56 (b), (c) e (d) si riportano anche le posizioni e velocità dei target inseriti all'interno dello scenario.

Da un confronto visivo tra la Figura 56 (b) e (d) si può apprezzare la diminuzione del numero di rivelazioni che rappresentano falsi allarme. Inoltre, mentre nella mappa di Figura 56 (b) il disturbo non consente di rivelare tutti i bersagli inseriti (alcune rivelazioni (punti bianchi) non compaiono in corrispondenza dei target (asterischi)), la Figura 56 (d) mostra una corrispondenza uno a uno tra le rivelazioni e i bersagli presenti nella scena sott'osservazione.

Con riferimento all' Eq. 55, la CA valutata in corrispondenza della Figura 56 (d) è pari a 30,26dB. Questo valore rappresenta il limite superiore delle prestazioni in quanto, se il segnale di riferimento è affetto da multipath, esisterà una degradazione di prestazione che si ripercuote direttamente sul valore di CA ottenibile.

In generale, quando il segnale di riferimento è affetto da multipath, è probabile che la capacità di cancellazione risulti gravemente limitata e quindi la sua efficacia per la rivelazione di bersagli deboli, il cui livello è al di sotto dei lobi laterali del clutter.



**Figura 56 – Scenario (a), uscita del rivelatore CA-CFAR senza la cancellazione e senza controllo lobi (b), dopo la cancellazione senza controllo lobi (c) e dopo cancellazione e filtro controllo lobi (d)**

In Tabella 14 si riportano i valori di CA ottenuti considerando il segnale di sorveglianza come descritto precedentemente mentre il segnale di riferimento affetto da un canale composto dal segnale diretto con DNR pari a 50 dB e 1-RAGGIO con un CNR pari a 45dB e ritardi e fasi riportati all'interno della stessa tabella. Come si evince dai valori in tabella le prestazioni dell'algoritmo di cancellazione, applicato senza equalizzare il segnale di riferimento, peggiorano aumentando il ritardo della replica di multipath presente sul riferimento.

Inoltre, come atteso, quando tra il raggio diretto ed il raggio riflesso sul canale di riferimento esiste uno sfasamento, quindi i due segnali si trovano in opposizione di fase, le prestazioni peggiorano ulteriormente rispetto all'aumento del ritardo. L'effetto di questa degradazione si ripercuote direttamente sul processo di rivelazione dei target perchè il



livello del disturbo, non completamente cancellato, potrebbe mascherare parzialmente o completamente i target presenti nella scena sotto osservazione.

**Tabella 14 - CA in dB ottenuta non utilizzando l'equalizzatore con il segnale di sorveglianza affetto da clutter e il segnale di riferimento da multipath.**

$\phi$ Fasi [°]							
$\tau$ ritardi [μs]	1°	60°	120°	180°	240°	300°	360°
<b>0.1094</b>	30,25	30,24	30,24	30,24	30,25	30,25	30,25
<b>0.5469</b>	29,79	29,95	30,09	29,89	29,97	30,07	29,79
<b>1.6406</b>	19,36	22,95	20,78	19,16	22,90	20,89	19,33
<b>2.1875</b>	17,32	15,79	15,09	15,55	16,79	18,01	17,35
<b>3.2813</b>	12,98	11,56	10,45	10,41	11,44	12,88	12,99

#### **4.5 Miglioramento delle prestazioni utilizzando l'equalizzatore LMS**

In questa sezione mostriamo il miglioramento, in termini di CA, ottenuto utilizzando il filtro LMS descritto in precedenza.

I parametri del fitro LMS utilizzati sono riportati nel paragrafo 4.3. La Tabella 15 riporta i valori di CA ottenuti utilizzando l'equalizzatore LMS e considerando lo stesso scenario utilizzato in precedenza per ricavare i valori riportati in Tabella 14.

Dopo l'equalizzazione, sembra confermarsi lo stesso trend di degradazione sia in termini di ritardo che di fase, anche se le prestazioni sono decisamente migliorate. Questa degradazione può essere imputata all'errore commesso dall'equalizzatore stesso nella stima del canale, come descritto nel paragrafo 4.3.1.

In generale, come è evidente da un confronto tra la Tabella 14 e Tabella 15, il filtro LMS è in grado di annullare correttamente il contributo multipath del segnale di riferimento in modo da recuperare, in parte, la perdita di CA vista in Tabella 14.

**Tabella 15 - CA in dB ottenuta utilizzando l'equalizzatore con il segnale di sorveglianza affetto da clutter e il segnale di riferimento da multipath.**

$\phi$ Fasi [°]	1°	60°	120°	180°	240°	300°	360°
$\tau$ ritardi [μs]							
<b>0.1094</b>	30.26	30.22	30.22	30.22	30.23	30.24	30.27
<b>0.5469</b>	30.16	30.16	30.16	30.14	30.17	30.16	30.17
<b>1.6406</b>	29.59	29.65	29.60	29.55	29.63	29.62	29.60
<b>2.1875</b>	29.25	29.13	29.05	29.09	29.21	29.29	29.26
<b>3.2813</b>	28.29	28.11	27.95	27.95	28.12	28.30	28.30

## 4.6 Risultati sperimentali

### 4.6.1 Acquisizione dati DVB-SH

I dati utilizzati in questa sezione sono stati acquisiti con il sistema descritto nel paragrafo 2.5 (acquisizioni DVB-SH del 18 Febbraio 2010).

Il trasmettitore di opportunità, che è stato considerato in questa campagna, è il satellite Eutelsat W2A. Il ricevitore DVB-SH è posto sul tetto della nostra facoltà (via Eudossiana 18, 00184 Roma), come mostrato in Figura 40. Consideriamo il puntamento dell'antenna verso la direzione relativa alla posizione del trasmettitore, nel caso specifico il satellite geostazionario Eutelsat W2A. Partendo dalla posizione nominale del satellite il puntamento ottimo è effettuato manualmente cercando di massimizzare la potenza ricevuta.

Il segnale del satellite viene trasmesso su una frequenza centrale di 2187,5 MHz con una polarizzazione circolare destra (RHCP). I dati in uscita dal NI PXI-5660 sono posizionati intorno alla frequenza intermedia (IF) pari a 14,5 MHz con la larghezza di banda pari a 20 MHz.

Il passaggio successivo è la corretta selezione della frequenza di campionamento del convertitore A/D (l'ADC replica il segnale di ingresso in tutte le zone di Nyquist); lo

standard DVB-SH prevede una frequenza di campionamento del segnale OFDM in banda base pari a 40/7MHz.

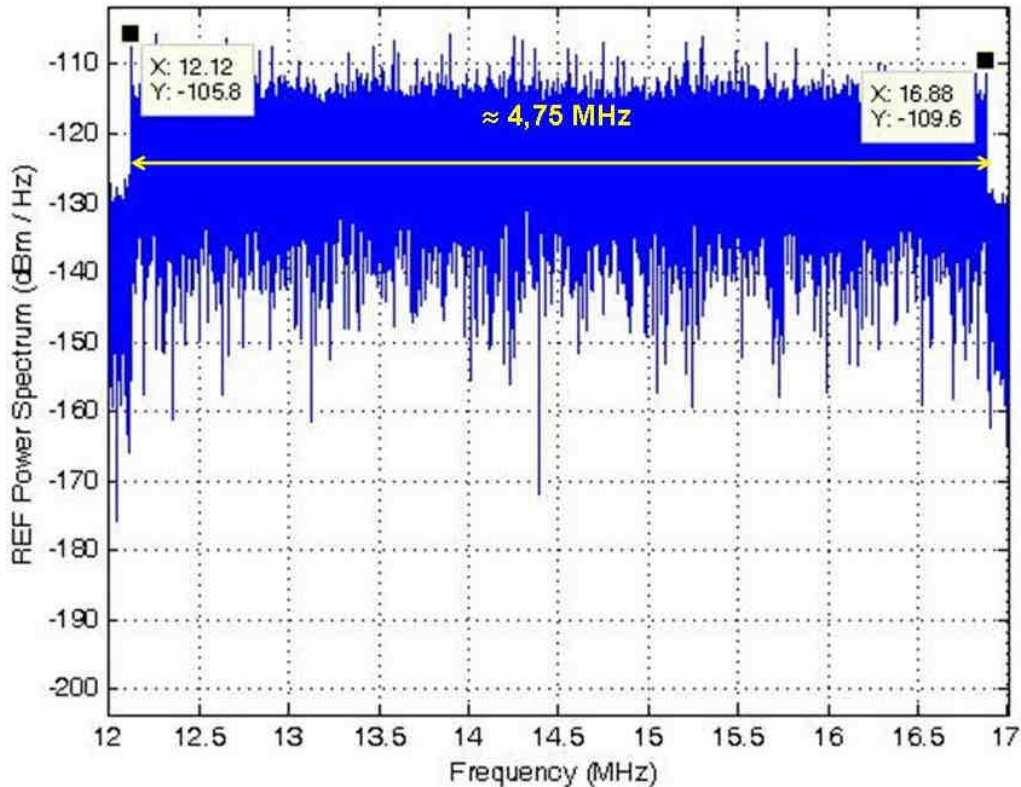
Poiché l'intera banda in uscita dal NI PXI-5660 è pari a 20 MHz centrata intorno a 14,5 MHz, per campionare correttamente i dati in uscita serve una frequenza di campionamento superiore a 49 MHz; nel nostro caso utilizziamo una frequenza di campionamento pari a 80 MHz. Questa frequenza consente di ottenere la stessa frequenza di campionamento suggerita dallo standard DVB-SH decimando con un fattore 14 il segnale dopo la traslazione in banda base dello stesso.

Nella Tabella 16 riportiamo tutti i singoli contributi per il calcolo teorico della potenza del segnale diretto, che sono in linea con le misure effettuate. Inoltre si noti che, durante le giornate di acquisizione, il satellite trasmetteva un segnale di test emettendo un EIRP pari a 59 dBW.

**Tabella 16 - Calcolo della potenza teorica del segnale diretto**

Variabili	Eutelsat W2A (Test)	Eutelsat W2A (Nominale)
EIRP <sub>TX</sub> [ dBW ]	59	72
Quota [ Km ]	35840	35840
Banda [ MHz ]	5	5
Frequenza centrale [ MHz ]	2187,5	2187,5
Guadagno antenna [ dB ]	21	21
Perdite [ dB ]	5	5
<b>Potenza segnale diretto [ dBm ]</b>	<b>-85,3</b>	<b>-72,3</b>
Guadagno Pre-Amplificazione [ dB ]	38	38
<b>Potenza segnale diretto [ dBm ]</b>	<b>-47,30</b>	<b>-34,3</b>

La larghezza di banda del segnale DVB-SH misurata è circa pari a 4.75460 MHz. L'errore commesso tra la stima della larghezza di banda sul segnale e quella teorica è circa 136 Hz; questo errore può essere attribuito al passo di campionamento utilizzato per la stima dello spettro di densità di potenza ( $df = fs/N = 80\text{MHz}/(0.5 \cdot 10^6) = 160\text{Hz}$ ).



**Figura 57 - Spettro di densità di potenza del segnale**

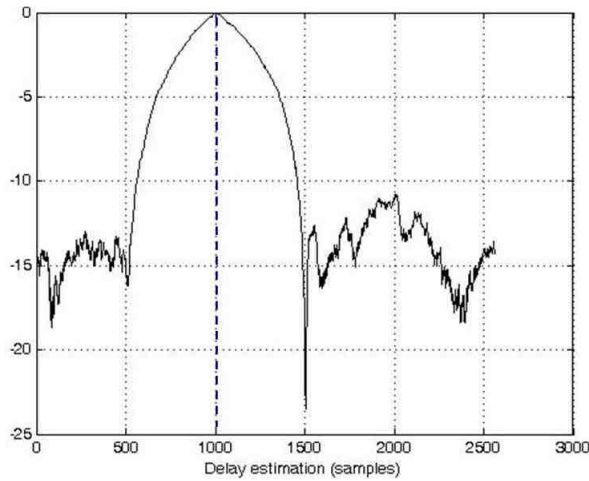
Infine, si noti che il segnale OFDM acquisito è stato trasmesso in modalità '2K', con una parte utile di durata pari a 0.35840ms e un intervallo di guardia di durata pari a 0.08960ms (1/4 della parte utile); queste informazioni sono necessarie al fine di sincronizzare ed equalizzare il segnale ricevuto.

#### 4.6.2 Demodulazione del Segnale DVB-SH

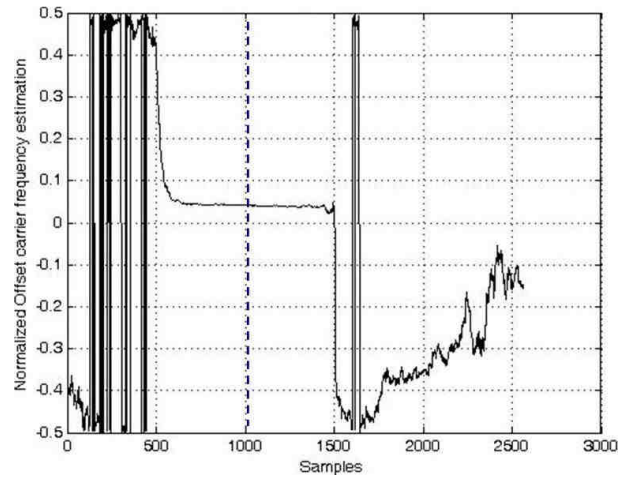
In Figura 58 si mostrano sinteticamente i risultati ottenuti dagli algoritmi di sincronizzazione su una generica acquisizione.

La stima del ritardo del segnale corrisponde a 0.1764 ms. In corrispondenza dell'indice del valore massimo ricavato della Figura 58 (a) si valuta l'offset di frequenza, mostrato in Figura 58 (b), pari a circa 95 Hz (la figura riporta la misura di offset di frequenza normalizzata rispetto alla durata della parte utile del simbolo OFDM).

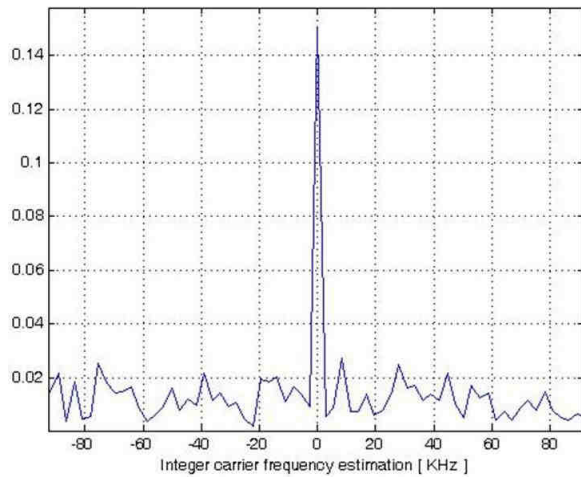
L'offset di frequenza misurato potrebbe essere affetto da un'ambiguità di misura se il valore di offset reale fosse maggiore della quantità  $1/T_U$ . In questo caso specifico, come si vede dalla Figura 58 (c), l'offset multiplo intero di  $1/T_U$  è nullo.



(a)



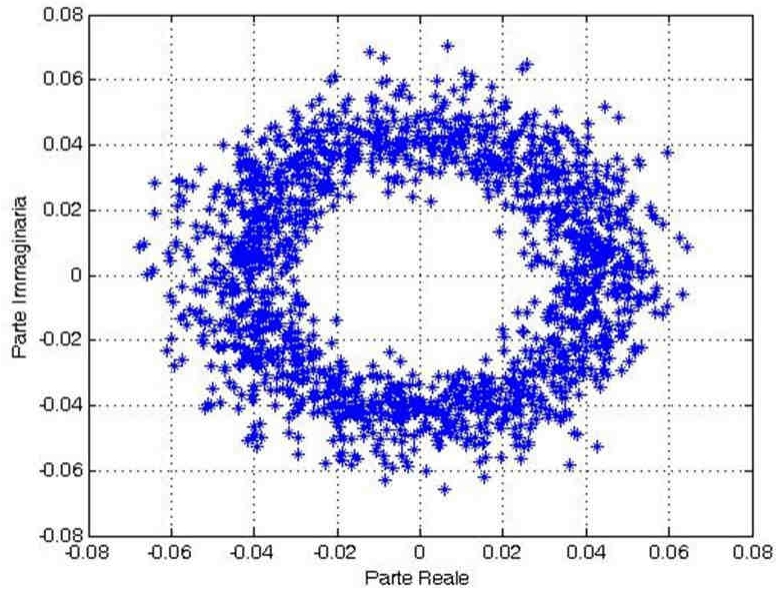
(b)



(c)

### Figura 58 – Risultati ottenuti dopo l'applicazione degli algoritmi pre-FFT e post-FFT

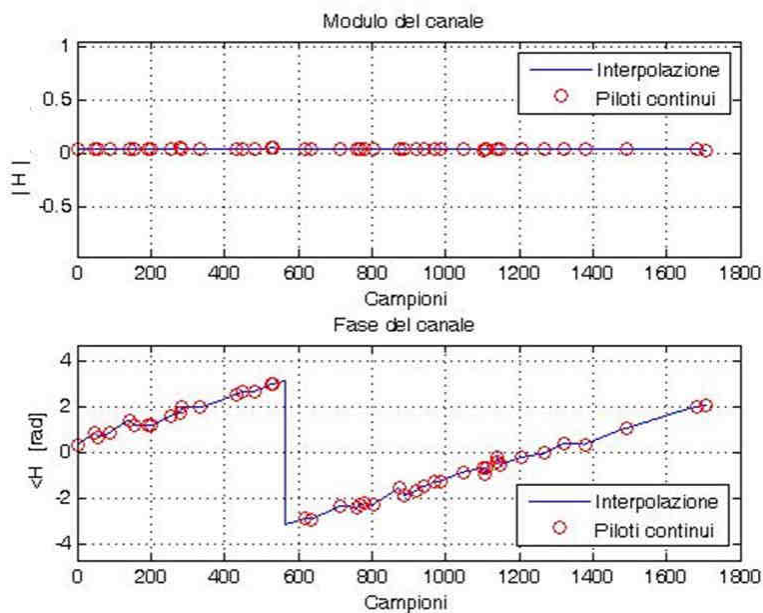
In Figura 59 si riporta il risultato ottenuto dopo l'applicazione degli algoritmi di sincronizzazione. Come si nota dalla figura, la corretta associazione dei numeri complessi ottenuti ai simboli della costellazione utilizzata è fortemente dipendente dal canale di trasmissione. In Figura 60 riportiamo la stima di canale, modulo e fase, effettuata considerando la posizione e il contenuto delle sole sottoportanti dei piloti continui, come indicato in Eq. 45;



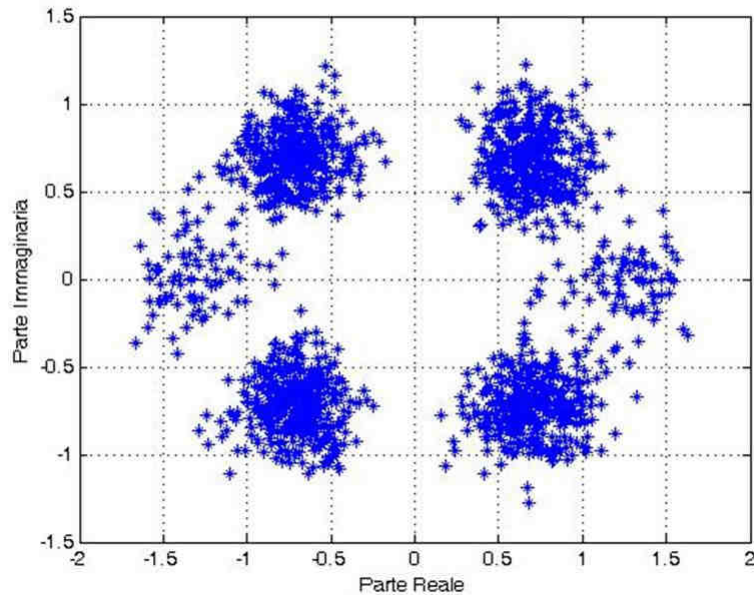
**Figura 59 - Costellazione ottenuta dopo la sincronizzazione (senza l'equalizzazione del canale)**

questa scelta è sufficiente in questo caso in quanto il segnale, proveniente dal satellite geostazionario, non ha una potenza elevata. La stima del contributo del canale sulle altre sottoportanti diverse dai Piloti Continui può essere ricavata interpolando linearmente i valori calcolati sui piloti continui stessi.

In Figura 61 si riporta il risultato da cui si riconosce la costellazione effettivamente utilizzata (QPSK).



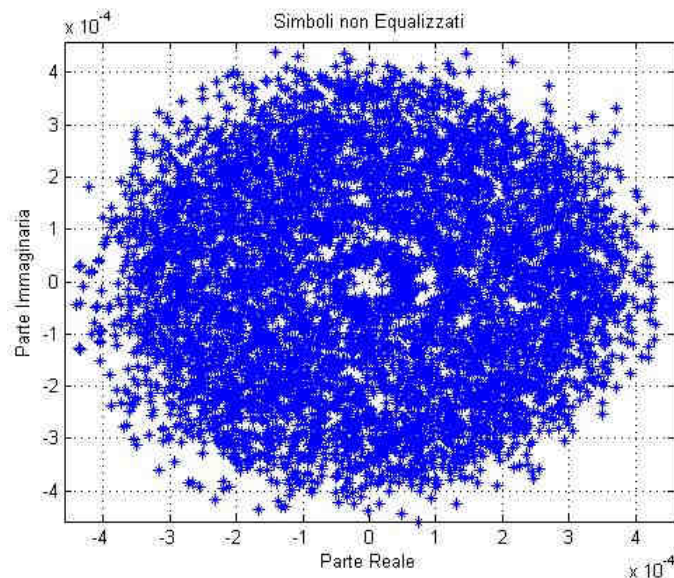
**Figura 60 - Modulo e fase della stima del canale**



**Figura 61 - Costellazione ottenuta dopo la sincronizzazione e l'equalizzazione del canale.**

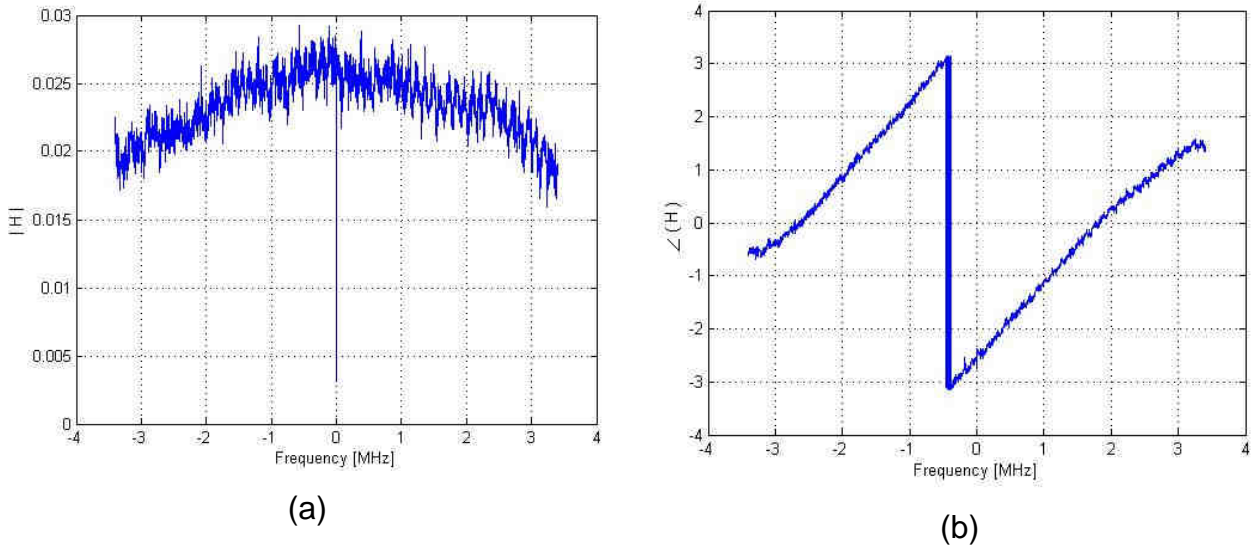
#### 4.6.3 Demodulazione del Segnale DVB-T

In analogia a quanto fatto nel caso precedente, si effettua la stessa analisi utilizzando un segnale DVB-T. Nello specifico, partendo dai risultati della sincronizzazione riportati in Tabella 12, si mostra in Figura 62 i simboli appartenenti alla costellazione.



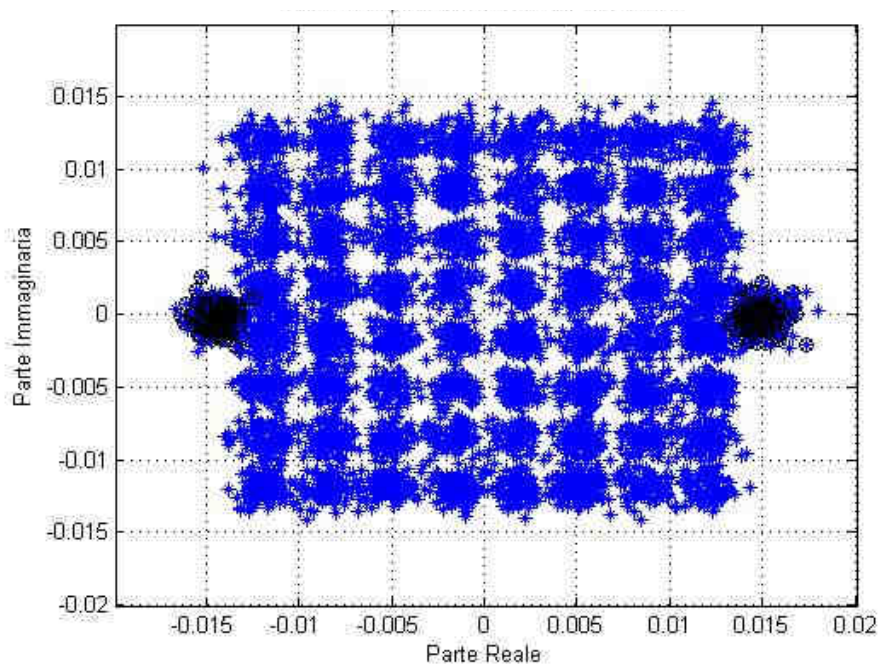
**Figura 62 - Costellazione ottenuta dopo la sincronizzazione (senza l'equalizzazione del canale)**

Anche in questo caso, come si nota dalla figura, la corretta associazione dei numeri complessi con i simboli è fortemente dipendente dal canale di trasmissione.



**Figura 63 - Modulo e fase della stima del canale**

In Figura 63 si riporta la stima del canale effettuata attraverso le relazioni riportate nel paragrafo 4.2 in corrispondenza del 150-esimo simbolo OFDM per garantire la convergenza dell'algoritmo. In Figura 64, normalizzando i simboli ricevuti attraverso la stima del canale ed compensando la fase residua, si ottiene la costellazione utilizzata (64QAM nel caso specifico).



**Figura 64 - Costellazione ottenuta dopo la sincronizzazione e l'equalizzazione del canale.**



## 5 Utilizzo delle GPU per l'elaborazione del segnale

Con riferimento agli obiettivi riportati nel paragrafo 1.1, in questo capitolo si affronta il problema legato all'implementazione real time della catena di processing.

Nel caso di utilizzo di trasmettitori DVB come illuminatori di opportunità, l'operazione di 2D-CCF, unita al resto del processing, diventa onerosa in termini di costo computazionale. Infatti, i lunghi tempi di integrazione richiesti per ottenere livelli di segnale accettabile implicano un numero elevato di moltiplicazioni complesse da eseguire. Inoltre, poichè la banda del segnale è elevata, rispetto al caso di utilizzo di segnali FM, anche le dimensioni della funzione di cross-correlazione stessa, in termini di filtri Doppler richiesti, sono elevate. In particolare la grandezza effettiva delle Mappe Range-Doppler è legata strettamente all'area di sorveglianza scelta e quindi alle dimensioni in range ( $0-R_{B\max}$ ) e Doppler ( $-|v_{bis}|_{\max}, |v_{bis}|_{\max}$ ), dove  $R_{B\max}$  e  $|v_{bis}|_{\max}$  sono la massima distanza bistatica e la massima velocità bistatica di interesse.. (vedi [4], [26] e [27]).

Nei paragrafi seguenti, mostreremo delle tecniche di programmazione che consentono di accelerare l'esecuzione del codice Matlab sia attraverso l'ottimizzazione del codice stesso che utilizzando in modo opportuno dispositivi esterni, nello specifico le schede grafiche presenti all'interno dei PC (Graphics Processing Units (GPU)), attraverso tool esterni integrabili con Matlab stesso. Infine effettueremo un confronto prestazionale in termini temporali tra codici eseguiti su CPU e GPU per il calcolo della funzione d'ambiguità riportata in Eq. 15.

### 5.1 Caratteristiche generali

CUDA ([70]) è, insieme, un'architettura di elaborazione in parallelo e una Application Programming Interface (API), realizzata da NVIDIA che permette di aumentare le prestazioni di elaborazione sfruttando la potenza di calcolo delle GPU.

L'architettura Cuda è formata da centinaia di Core che operano in parallelo per l'elaborazione dei dati. Nello specifico, come si può apprezzare dalla Figura 65, possiamo affermare che nei normali processori (Central Processing Unit - CPU), con un numero di core (ALU) che oggi giorno arriva generalmente ad 8 o poco più, la maggior parte dei transistor è spesa per la sezione di controllo e l'implementazione della cache, mentre nel caso delle GPU i core sono molti (ad oggi posso essere oltre 400 in funzione del modello

di GPU scelto) e molto più semplici e quasi tutta l'area disponibile è spesa per implementarli.



**Figura 65 – Confronto architetturale alto livello tra CPU e GPU**

Attualmente si utilizzano una CPU e una GPU insieme per ottenere un modello di elaborazione eterogeneo. La parte sequenziale dell'applicazione viene eseguita sulla CPU, mentre la parte a maggiore densità di calcolo viene eseguita sulla GPU. In questo modo l'esecuzione dell'applicazione risulta più rapida perché l'alta velocità della GPU viene sfruttata per migliorare le prestazioni.

Nello specifico, le operazioni relative alla GPU per l'elaborazione dei dati sono elencate di seguito:

- Viene effettuato il trasferimento dei dati da elaborare dalla DRAM della CPU verso la DRAM della GPU.
- La CPU istruisce la GPU, attraverso il CUDA, per l'elaborazione effettiva dei dati.
- Viene effettuata l'esecuzione parallela sui diversi core della GPU.
- Viene trasferito verso la DRAM della CPU il risultato ottenuto e memorizzato all'interno della GPU.

L'operazione di trasferimento dei dati, da e per la CPU, attraverso le memorie potrebbe limitare il tempo totale di esecuzione perché implica l'uso di un bus per il trasferimento dei dati (le GPU più moderne richiedono uno slot disponibile con un Bus PCI express 16x).

Un'alternativa rispetto alla programmazione CUDA per l'utilizzo delle GPU è fornita dalla società Matworks ([71]), proprietaria del software Matlab, che già dalla Release R2010b

supporta interamente, all'interno della pacchetto Parallel Computing Toolbox (PCT), l'utilizzo delle GPU. Il software Matlab supporta la tecnologia NVIDIA ® CUDA™ se si utilizzano GPU con una capacità di calcolo superiore a 1.3.

Tuttavia, è possibile utilizzare tool esterni come, ad esempio, GPUmat e/o Jacket che consentono, attraverso delle tecniche, costrutti e comandi appositi, di utilizzare le GPU come dispositivi esterni special purpose.

Nel caso specifico si fa riferimento alla libreria gratuita GPUmat, [72], che abilita il codice matlab per l'esecuzione sulla GPU; questa libreria permette in maniera molto semplice di accedere alla potenza di calcolo della GPU senza la conoscenza specifica della GPU stessa. Inoltre il codice matlab viene eseguito sulla GPU in modo del tutto trasparente rispetto all'utente.

## 5.2 Libreria GPUmat

Prima di iniziare la descrizione della libreria, è opportuno capire gli strumenti che consentono di ottimizzare il codice Matlab e migliorarne le prestazioni. L'ottimizzazione del codice matlab rappresenta un primo passo, molto importante, per consentire un netto miglioramento delle prestazione quando si utilizza lo stesso codice sulla GPU.

Quando si scrive un codice in Matlab (script e/o funzione) è necessario fare molta attenzione alla gestione della memoria e alla vettorizzazione.

Si dice che un codice è *memory-bound* quando le sue prestazioni sono limitate dal tempo necessario per accedere alla memoria. Alcune tecniche consentono di migliorare le prestazioni di un codice Matlab *memory-bound* come

- la preallocazione delle variabili: Matlab non richiede apriori la dichiarazione delle variabili e si possono aumentare le dimensioni di una generica variabile dinamicamente durante l'esecuzione. Questa caratteristica consente una rapida fase prototipale a discapito della lentezza e l'inefficienza.
- la memorizzazione ed l'accesso ai dati "per colonne": un codice è efficiente quando l'accesso in memoria è fatto attraverso posizioni consecutive della stessa. Poiché matlab memorizza i dati per colonne si avrà la massima efficienza quando si processano i dati nel medesimo ordine.
- la creazione di nuove variabili solo quando necessario effettuando le operazioni *in-place*: l'allocazione e copia di variabili è un processo che richiede tempo di calcolo

e quindi le operazioni in-place, modificando strutture dati già esistenti senza crearne di nuove, consentono di migliorare le prestazioni del codice.

L'altro aspetto è legato alla vettorizzazione delle operazioni. Si dice che un calcolo è *vettorizzato* quando si eseguono delle operazioni analitiche e logiche su una struttura vettoriale e/o matriciale piuttosto che elemento per elemento. In questo caso la vettorizzazione sfrutta a pieno le potenzialità di matlab perché molte funzioni di matlab (built-in) risultano essere già vettorizzate (ad esempio sqrt, sin, rand ect).

Chiaramente, queste tecniche sono quasi sempre d'aiuto ma la loro efficacia va valutata caso per caso rispetto all'architettura di calcolo utilizzata.

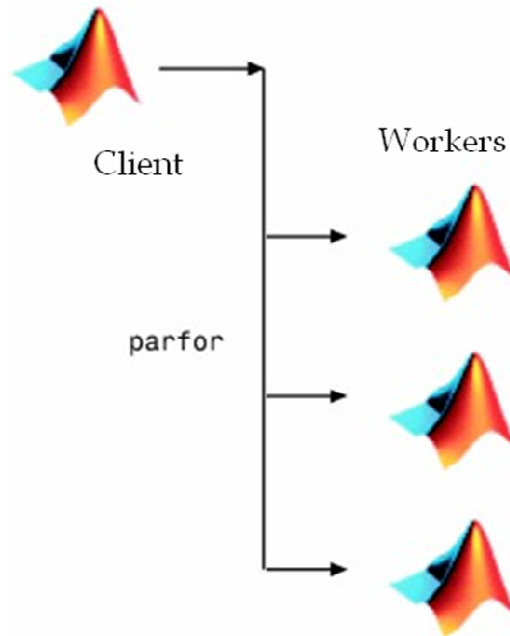
Un aspetto molto importante è la precisione del dato che si utilizza che può impattare sulle prestazioni del codice eseguito migliorando il tempo di esecuzione del codice stesso sia perché si sfrutta in modo più efficiente la larghezza di banda dei processori e coprocessori (incluse le GPU) e sia perché si dimezza l'occupazione di memoria delle singole variabili. In generale, Matlab, per default, definisce le variabili reali in doppia precisione (double) ma, attraverso il costrutto built-in *single*, è possibilità di operare anche in singola precisione perdendo cifre significative rispetto al generico valore a fronte di una minore occupazione di memoria.

```
>> a=pi
a = 3.141592653589793
>> b=single(a)
b = 3.1415927
>> whos
Name    Size    Bytes Class
a       1x1      8    double
b       1x1      4    single
```

Nello specifico, all'interno del riquadro, si riporta la command window di matlab dove sono state effettuate semplici operazioni che consentono di apprezzare le differenze sostanziali riguardo all'uso della built-in *single* sia in termini di occupazione di memoria sia in termini di precisione rispetto alle cifre significative utilizzate. L'unica accortezza, valida in generale se si usa questa built-in, è la verifica della bontà dei risultati perché gli algoritmi potrebbero dare dei risultati molto diversi da quelli attesi.

Infine va ricordato che, a partire dalla Release R2007a, Matlab beneficia anche dell'utilizzo di più processori presenti sulla propria macchina attraverso la toolbox PCT sfruttando in modo efficiente l'Hardware a disposizione. Il calcolo Multithread non è abilitato per default

(dipende dalla Release utilizzata) e deve essere richiesto attraverso appositi costrutti. Questa caratteristica può migliorare le prestazioni del codice con benefici immediati e con poco sforzo.



**Figura 66 – Costrutto parfor**

Un esempio è il costrutto *parfor* (Figura 66) che, utilizzato quando la modalità multithread è attiva, permette di parallelizzare tutte le operazioni contenute in un classico ciclo *for*. L'unica condizione che deve essere rispettata è che ogni iterazione, interna al ciclo, deve essere completamente indipendente da tutte le altre iterazioni; solo in questo caso matlab riesce a utilizzare tutti i core a disposizione.

Fatta questa premessa, introduciamo brevemente la libreria GPUmat. Le funzioni presenti nella libreria possono essere raggruppate in funzioni di basso livello e di alto livello (lista di funzioni e operatori supportati vedi Appendice A). Le funzioni di alto livello, utilizzabili in maniera del tutto analoga alle funzioni matlab esistenti, non richiedono sforzi ulteriori da parte dell'utilizzatore mentre per usufruire delle funzioni di basso livello è richiesta un minimo di competenza rispetto alla programmazione base di CUDA. A tal proposito, la libreria GPUmat rende disponibili dei costruttori Matlab (wrappers) che consentono di interagire direttamente con le librerie CUDA CUBLAS e CUFFT; questi costruttori possono essere utilizzati anche indipendentemente da GPUmat e nel seguito verranno fatti alcuni esempi. Per una lista completa delle funzioni e degli operatori supportati si fa riferimento all'Appendice A e al materiale in [72].

La libreria GPUmat può essere usata in due modalità diverse:

- come una toolbox aggiuntiva di matlab utilizzando le funzioni di alto livello (questa è la modalità più semplice).
- come un Kit di sviluppo GPU per integrare le funzioni esistenti, di basso e alto livello, con funzioni non disponibili all'interno della libreria.

Nel seguito descriveremo come utilizzare la libreria GPUmat.

### 5.2.1 Requisiti di sistema e inizializzazione dell'ambiente GPU

Il gruppo GP-you ha testato la libreria GPUmat sia in ambiente Windows che in linux con le versioni di Matlab superiori a R2007a. Oltre alle informazioni reperibili dal sito [72], sono disponibili i link diretti per scaricare le diverse librerie GPUmat che utilizzano varie e diverse versioni del CUDA; quest'ultimo deve essere installato preventivamente sul proprio sistema e per farlo si consiglia di utilizzare le istruzioni presenti sul sito [70] utilizzando la versione CUDA corretta in funzione della modello della GPU posseduta.

L'installazione della libreria è semplice e può essere eseguita in 3 passi:

- 1- Installare il pacchetto Microsoft Visual C++ 2008 (richiesto solo per sistema operativo Windows) se non già presente.
- 2- Avviare il software Matlab aggiungendo la cartella, che contiene la libreria GPUmat, all'interno dell'insieme dei Path di Matlab.
- 3- Avviare GPUmat utilizzando la funzione GPUstart.m messa a disposizione dalla libreria.

L'esecuzione della funzione GPUstart.m produrrà un risultato all'interno della propria command window di matlab contenente tutte le informazioni utili simile all'esempio riportato in Figura 67, in cui vengono fornite le informazioni relative alla versione del GPUmat utilizzata, alla versione di CUDA richiesta e utilizzata dal proprio sistema e i diversi dispositivi presenti sul proprio PC con le relative memorie DRAM disponibili.

```

>> GPUstart
Copyright gp-you.org. GPUmat is distributed as Freeware.
By using GPUmat, you accept all the terms and conditions
specified in the license.txt file.

Please send any suggestion or bug report to gp-you@gp-you.org.

Starting GPU
- GPUmat version: 0.270
- Required CUDA version: 4.0
There are 2 devices supporting CUDA
CUDA Driver Version:          4.0
CUDA Runtime Version:        4.0

Device 0: "Tesla C1060"
  CUDA Capability Major revision number:    1
  CUDA Capability Minor revision number:    3
  Total amount of global memory:           4294770688 bytes

Device 1: "Quadro FX 370"
  CUDA Capability Major revision number:    1
  CUDA Capability Minor revision number:    1
  Total amount of global memory:           268107776 bytes
- Your system has multiple GPUs installed
  -> Please specify the GPU device number to use [0-1]: 0
- CUDA compute capability 1.3
...done
- Loading module EXAMPLES_CODEOPT
- Loading module EXAMPLES_NUMERICS
  -> numerics13.cubin
- Loading module NUMERICS
  -> numerics13.cubin
>>

```

**Figura 67 – Risultato sulla Command Window dopo l'esecuzione della funzione GPUstart.m**

Se sono presenti più dispositivi installati si procede alla scelta selezionando quello desiderato e successivamente vengono caricati automaticamente diversi moduli che consentono di interfacciare Matlab con la GPU.

Oltre a GPUstart.m, il tool fornisce altre due funzioni molto utili, GPUstop.m e GPUinfo.m. GPUstop.m viene utilizzata per chiudere l'ambiente GPU mentre GPUinfo.m restituisce tutte le informazioni relative alla versione CUDA disponibile sulla macchina; inoltre, vengono mostrate anche le diverse tipologie di dispositivi installati sulla propria macchina.

### 5.2.2 Creazione di variabili GPU e conversione in variabili Matlab

Una variabile GPU può essere creata in due modi diversi. Il primo è utilizzare le due diverse classi disponibili, *GPUsingle* e *GPUdouble* rispettivamente; questi due comandi sono equivalenti alle funzioni matlab *single* e *double* che permettono di definire la

precisione di una variabile reale e/o complessa. È importante sapere che è richiesto il trasferimento di dati tra la memoria della CPU alla GPU se la variabile GPU è inizializzata con una variabile Matlab; questo trasferimento richiede tempo e potrebbe ridurre le prestazioni del codice.

La seconda modalità è creare direttamente la variabile sulla memoria della GPU attraverso l'uso di istruzioni che lo permettono (lista completa in [72]); di seguito due esempi di creazione di variabili.

```
Ahs = single (rand(100)); %%Ahs è una variabile su CPU in singola precisione  
A = GPUsingle(Ahs); %%Ahs trasferita su memoria GPU in A
```

```
Ahd = double (rand(100)); %%Ahd è una variabile su CPU in doppia precisione  
A = GPUdouble(Ahd); %% Ahd trasferita su memoria GPU in A
```

```
A = rand(1,100,GPUdouble);%%Variabile definita su memoria GPU doppia precisione  
B = rand(1,100,GPUsingle); %%Variabile definita su memoria GPU singola precisione  
C = sin(B); %% Calcolo effettuato su GPU. C è memorizzata su GPU
```

Sebbene una variabile GPU è disponibile in Matlab attraverso un puntatore e può essere visualizzata all'interno della command window, il suo contenuto è presente solo sulla memoria della stessa GPU. La conversione di una variabile GPU in una variabile Matlab implica il trasferimento del contenuto della variabile dalla DRAM della GPU alla DRAM della CPU. Di seguito viene mostrato come effettuare questo trasferimento attraverso l'uso dei comandi *single* e *double*.

```
Ahs = single (rand(100)); %%Ahs è una variabile su CPU in singola precisione  
A = GPUsingle(Ahs); %%Ahs viene trasferito su GPU in A  
C =single(A); %%A è trasferita in CPU su C
```

```
Ahd = double (rand(100)); %%Ahs è una variabile su CPU in doppia precisione  
A = GPUdouble(Ahd); %%Ahd viene trasferito su GPU in A  
C =double(A); %%A è trasferita su CPU in C
```

Non tutte le operazioni sono effettuate sulla GPU, ma solo quelle che coinvolgono le variabili GPU utilizzando funzioni definite all'interno della libreria GPUmat (lista completa in [72]). Inoltre, GPUmat implementa anche gli operatori Matlab somma, differenza, prodotto matriciale ed elemento per elemento ect. e quindi è possibile effettuare le operazioni algebriche tra variabili presenti sulla memoria della GPU.



### 5.2.3 Indicizzazione delle variabili

Un'operazione molto utile che può essere effettuata sulla GPU è la selezione di elementi di un array come qualsiasi vettore Matlab. Le classiche operazioni, riportate di seguito, sono supportate dalla libreria GPUmat che effettua la chiamata delle funzioni *subsref* e *subsasgn* per selezionare porzioni di matrici e/o vettori in analogia all'operazione effettuata da Matlab stesso.

```
A = rand(100,GPUsingle); %%Aè memorizzata sulla GPU
B = A(1,1:end);          %%Selezione della prima riga di A
C = A(1,1:10);           %%Selezione dei primi 10 elementi della prima riga di A
```

Le funzioni *slice* e *assign*, messe a disposizione dalla libreria GPUmat, posso essere utilizzate per effettuare la selezione di una porzioni di elementi di un array; queste funzioni hanno una sintassi molto simile all'indicizzazione di Matlab ma sono molto più efficienti delle funzioni *subsref* e *subsasgn*. Di seguito alcuni esempi sull'utilizzo di queste due funzioni.

```
Bh = single(rand(100));
B = rand(100,GPUsingle);

Ah = Bh(1:end);          % Sintassi Matlab
A = slice(B,[1,1,END]); % Sintassi Equivalente slice

Ah = Bh(1:10,:);        % Sintassi Matlab
A = slice(B,[1,1,10],':'); % Sintassi Equivalente slice

Ah = Bh([2 3 1],:);     % Sintassi Matlab
A = slice(B,{[2 3 1]},':'); % Sintassi Equivalente slice

Ah = Bh([2 3 1],1);     % Sintassi Matlab
A = slice(B,{[2 3 1]},1); % Sintassi Equivalente slice

A = rand(100,GPUsingle);
B = rand(10,10,GPUsingle);
Ah = single(A);
Bh = single(B);
Ah(1:10,1:10) = Bh;      % Sintassi Matlab
assign(1, A, B, [1,1,10],[1,1,10]); % Sintassi Equivalente assign

A = rand(100,GPUsingle);
B = rand(4,10,GPUsingle);
Ah = single(A);
Bh = single(B);
Ah([2 3 1 5],1:10) = Bh; % Sintassi Matlab
assign(1, A, B, {[2 3 1 5]},[1,1,10]); % Sintassi Equivalente assign
```

In Tabella 17 si riporta un confronto tra le funzioni *subsasgn* e *assign* in termini di tempo impiegato per effettuare l'operazione. La misura del tempo è realizzata utilizzando le funzioni *tic* e *toc* di Matlab che permettono di misurare il tempo trascorso; *tic* salva il tempo attuale che *toc* utilizzerà in seguito per misurare il tempo trascorso.

La sequenza di comandi è: (1) *tic*; (2) Operazioni; (3) *toc*; con cui si misura la quantità di tempo necessario per completare una o più operazioni specificate in (2) e si mostra, all'interno della command window, il tempo impiegato in secondi.

**Tabella 17 – Confronto CPU/GPU**

Operazione	CPU	GPU high level	GPU assign construct
A(1:end) = B	0.007636	0.012822	0.000382
A(1:10,:) = B	0.000613	0.000638	0.000327
A(:, :) = B	0.003462	0.000706	0.000371

### 5.2.4 Gestione della memoria della GPU

La gestione della memoria della GPU è un aspetto molto importante quando si utilizza questo dispositivo. Una situazione frequente riguarda l'allocazione di uno spazio all'interno della memoria della GPU anche se questa risulta essere insufficiente; quando questo avviene è necessario rimuovere manualmente alcune o tutte le variabili dalla memoria.

La gestione della memoria può avvenire sia attraverso funzioni alto livello che classi GPU e, in alternativa, funzioni basso livello. Un modo molto semplice per eliminare le variabili dalla memoria della GPU è utilizzare funzioni di alto livello, nello specifico il comando matlab *clear* supportato da GPUmat. Inoltre, per la verifica dello spazio disponibile, esiste un comando, *GPUmem*, che ritorna la dimensione della memoria libera sulla GPU in bytes.

Un modo alternativo per gestire la memoria è utilizzare le classi GPU come mostrato di seguito.

**Gestione memoria GPU con funzioni alto livello**

```
A = rand(100,100,GPUSingle);
clear A;
```

**Gestione memoria GPU con le classi GPU**

```
B = GPUSingle(); % crea GPUSingle vuoto
setReal(B); % Configura il tipo REAL
setSize(B,[100 100]); % Configura la dimensione del GPUSingle
GPUallocVector(B); % Alloca la memoria utile della GPU
clear B; % Pulisce la memoria della GPU dal puntatore B
```

Il costrutto GPUSingle (GPUdouble) implementa al suo interno il codice che si prende cura della compensazione delle regioni di memoria non utilizzata. Non c'è bisogno di pulire in modo esplicito la memoria della GPU. Se necessario si può comunque fare utilizzando il comando Matlab *clear*.

Infine, un metodo ulteriore di gestione della memoria è l'uso delle funzioni basso livello. Di seguito si mostra un esempio relativo all'allocazione e de allocazione di un vettore di 100 elementi in singola precisione utilizzando i wrappers matCUDA disponibili.

**Gestione memoria GPU con funzioni basso livello**

```
GPUptr = 0; % crea un nuovo puntatore
% alloca la memoria utilizzando cublasAlloc
SIZE_OF_FLOAT = 4;
NUMEL = 100;
[status GPUptr]= cublasAlloc(NUMEL,SIZE_OF_FLOAT,GPUptr);
cublasCheckStatus( status, 'Errore allocazione di memoria');
status = cublasFree(GPUptr);
cublasCheckStatus( status, '!!!! Errore memori libera (GPUptr)');
```

In particolare, nel caso specifico vengono utilizzate le seguenti funzioni:

- *cublasAlloc*:

crea un oggetto nello spazio di memoria della GPU (*GPUptr*) in grado di contenere un numero di elementi pari a *numel*, di dimensione *size\_of\_float* pari a 4 Bytes.

- *cublasFree*:

permette di eliminare l'oggetto creato e posto in *GPUptr* liberando fisicamente lo spazio di memoria allocato. Se l'utente crea un puntatore Matlab alla memoria della GPU mediante funzioni di basso livello, la memoria non viene pulita

automaticamente quando la variabile non è più utilizzata. In questo caso l'utente deve pulire manualmente la GPU memoria.

### 5.2.5 Calcolo della FFT e IFFT

Un'operazione interessante, legata all'esecuzione degli algoritmi per l'elaborazione del segnale nel PBR, è la valutazione della Trasformata di Fourier e della sua inversa. Anche in questo caso, come accade per la gestione delle memoria, è possibile utilizzare sia funzioni di basso livello che di alto livello.

Le funzioni alto livello che permettono di calcolare la trasformata sono simili alle funzioni Matlab e sono *fft* e *ifft* rispettivamente; nel seguito un esempio di utilizzo delle funzioni citate.

```
Trasformata di Fourier
A = rand(1,100,GPUSingle)+i.*rand(1,100,GPUSingle);
B = fft(A);

Trasformata inversa di Fourier
A = rand(1,100,GPUSingle)+i.*rand(1,100,GPUSingle);
B = ifft(A);
```

Un modo alternativo e più efficiente per il calcolo della trasformata di Fourier è considerare le funzioni di basso livello utilizzando matCUDA (Appendice A) come nell'esempio riportato di seguito:

```

N = 512; %% Definizione della Lunghezza dei vettori da utilizzare

h_A = single(rand(1,N)+i*rand(1,N)); %% Generico vettore su CPU

d_A = GPUsingle(h_A); %% Trasferimento di h_A su GPU in d_A
d_B = GPUsingle(h_A); %% Trasferimento di h_A su GPU in d_B

fftType = cufftType;
fftDir = cufftTransformDirections;

% Piano FFT
plan = 0;
[status, plan] = cufftPlan1d(plan, numel(d_A), fftType.CUFFT_C2C, 1);
cufftCheckStatus(status, 'Error in cufftPlan1D');

% Esecuzione FFT su GPU
[status] = cufftExecC2C(plan, getPtr(d_A), getPtr(d_B), fftDir.CUFFT_FORWARD);
cufftCheckStatus(status, 'Error in cufftExecC2C');

% Esecuzione IFFT su GPU
[status] = cufftExecC2C(plan, getPtr(d_B), getPtr(d_A), fftDir.CUFFT_INVERSE);
cufftCheckStatus(status, 'Error in cufftExecC2C');

[status] = cufftDestroy(plan);
cufftCheckStatus(status, 'Error in cuffDestroyPlan');

```

Le funzioni che vengono utilizzate sono le seguenti:

- *cufftType*:  
struttura contenente un codice che identifica il tipo di dato su cui effettuare l'operazione di trasformata supportando sia dati complessi che reali. In particolare esistono tre diversi tipi: (i) CUFFT\_R2C suppone il dato reale in ingresso e quello complesso in uscita, (ii) CUFFT\_C2R dal dato complesso al dato reale e infine (iii) CUFFT\_C2C dal dato complesso al dato complesso;
- *cufftTransformDirections*:  
struttura contenete un codice che identifica il verso della trasformata (1 fft, -1 ifft rispettivamente).
- *cufftPlan1d*:  
crea la pianificazione per il calcolo della trasformata per un specifica dimensione del segnale e del tipo di dato in ingresso. Gli ingressi di questa funzione sono:  
plan:puntatore all'oggetto che viene utilizzato per eseguire l'operazione;  
intero  $n_x$ : dimensione della trasformata, in questo caso pari a N;

type: il tipo di dato della trasformata;

batch: numero di trasformate di dimensioni  $n_x$ .

- *cufftExecC2C*:

esegue sostanzialmente la trasformata di Fourier ammettendo sia in ingresso che in uscita dati complessi. All'interno della libreria sono disponibili anche le versioni *cufftExecR2C* (ingresso dati reali, uscita dati complessi) e *cufftExecC2R* (ingresso dati complessi, uscita dati reali).

- *cufftDestroy*:

funzione che libera tutte le risorse della GPU associate ad un piano ed elimina la struttura dati pianificata internamente. Questa funzione dovrebbe essere richiamata una volta che il piano non è più necessario per evitare sprechi di memoria della GPU.

### 5.3 Confronto prestazionale

#### 5.3.1 Implementazione efficiente della 2D-CCF

L'operazione per la valutazione della 2D-CCF, riportata in Eq. 15, può essere implementata, in modo efficiente, in due distinte modalità in accordo a determinati criteri e sfruttando la conoscenza dell'algoritmo Fast Fourier Transform (FFT) per il calcolo della Trasformata Discreta di Fourier (DFT) [27].

Una prima modalità, denominata correlation-FFT, si basa sull'osservazione che, per la generica cella di risoluzione Doppler, tutti i campioni relativi alle celle di risoluzione in tempo della 2D-CCF possono essere valutati attraverso la cross-correlazione tra il segnale di riferimento e una copia spostata in Doppler del segnale di sorveglianza,  $s_m[n] = s_{surv}[n] \exp(-j2\pi mn/N)$ , come segue:

$$\chi[l, m] = \sum_{n=0}^{N-1} s_m[n] \cdot s_{ref}^*[n-l] = C_m[l] \quad \text{Eq. 56}$$

Un risparmio, in termini di costo, può essere realizzato implementando l'equazione Eq. 56 nel dominio della frequenza, come riportato dalla seguente relazione:

$$C_m[l] = IDFT \left\{ DFT \{ s_m[n] \} DFT \{ s_{ref}[n] \}^* \right\} = IDFT \{ S_m[k] S_{ref}^*[k] \} \quad \text{Eq. 57}$$

Dove, per m-th filtro Doppler, viene valutata la Trasformata Discreta Inversa di Fourier (IDFT) del prodotto degli spettri e vengono scartati i campioni di non interesse rispetto alle

dimensioni della Mappa desiderata. Inoltre, si noti che, per ogni  $m$ -th filtro Doppler di interesse, la sequenza DFT  $S_m[k]$  di  $s_m[n]$  può essere calcolata come uno spostamento circolare dei campioni della DFT del segnale di sorveglianza; di conseguenza le Trasformate di Fourier dei due segnali (riferimento e sorveglianza) possono essere valutate una sola volta.

Una seconda modalità, denominata direct-FFT, si basa sull'osservazione che, per ogni  $l$ -th cella di risoluzione in tempo, i campioni lungo frequenza doppler della 2D-CCF corrispondono ai campioni della DFT della sequenza  $x_l[n] = s_{surv}[n]s_{ref}^*[n-l]$  ottenuta come il prodotto tra il segnale di sorveglianza e una copia ritardata complessa coniugata del segnale di riferimento.

$$\chi[l, m] = X_l[m] = DFT\{x_l[n]\} \quad \text{Eq. 58}$$

In modo analogo al caso precedente, per ogni ritardo bistatico  $l$ , viene valutata la DFT e vengono scartati i campioni corrispondenti a Doppler di non interesse.

**Tabella 18 – Costo computazionale**

Algoritmi	Moltiplicazioni Complesse	Addizioni Complesse
<b>Correlation-FFT</b>	$2M\log_2(N) + N_f [N + M\log_2(N)]$	$2M\log_2(N) + N_f [M\log_2(N)]$
<b>Direct-FFT</b>	$N_f [N + M\log_2(N)]$	$N_f [M\log_2(N)]$
<b>Channelization Technique</b>	$2M\log_2(n_C) + N + M\log_2(N)$	$2M\log_2(n_C) + M\log_2(N)$

La Tabella 18 riporta separatamente, per ogni tipo di implementazione dell'operazione 2D-CCF vista, il numero di moltiplicazioni e addizioni complesse da eseguire per ottenere il risultato dove  $N$  indica il numero di campioni del segnale utilizzato corrispondenti al tempo di integrazioni desiderato,  $N_t$  ed  $N_f$  il numero di celle di risoluzione in tempo e frequenza desiderate rispettivamente.

### 5.3.2 Implementazione sub-ottima della 2D-CCF

Il costo computazionale richiesto per la valutazione della funzione 2D-CCF può essere ridotto attraverso l'utilizzo di tecniche sub-ottime, accettando perdite in termini di rapporto segnale a rumore ([27]). La riduzione del costo può essere ottenuta, oltre che evitando le operazioni non utili degli algoritmi ottimi, anche approssimando la compensazione di fase attraverso tecniche di decimazione o suddivisione in canali dei segnali in gioco in accordo all'algoritmo sub-ottimi che verrà descritto nel seguito.

L'algoritmo, denominato Channelization Technique, opera attraverso la divisione dei segnali, nel dominio della frequenza, in  $n_C$  canali (bande di frequenze) che verranno elaborati in modo separato.

Il risultato finale può essere ottenuto combinando coerentemente tutti i singoli contributi delle  $n_C$  bande producendo una Mappa Range-Doppler con un numero di operazione inferiore rispetto all'implementazione ottima ma affetta soltanto da una riduzione in SNR.

Partendo, anche in questo caso, dalla relazione Eq. 15, possiamo scrivere la cross-correlazione nel dominio del tempo tra l' $r$ -th porzione del segnale di sorveglianza e di riferimento di lunghezza pari a  $n_C$  attraverso una sua approssimazione nel dominio della frequenza:

$$\begin{aligned}
 \chi[l, m] &\cong \sum_{r=0}^{N_C-1} e^{-j2\pi \frac{mr}{N_C}} \sum_{p=0}^{n_C-1} s_{surv}[rn_C + p] \cdot s_{ref}^*[rn_C + p - l] \cong \\
 &\cong \sum_{r=0}^{N_C-1} e^{-j2\pi \frac{mr}{N_C}} \frac{1}{n_C} \sum_{k=0}^{n_C-1} S_{surv}^{(r)}[k] S_{ref}^{(r)*}[k] e^{+j2\pi \frac{kl}{n_C}} = \\
 &= \frac{1}{n_C} \sum_{k=0}^{n_C-1} e^{+j2\pi \frac{kl}{n_C}} \sum_{r=0}^{N_C-1} Y[k, r] e^{-j2\pi \frac{mr}{N_C}}
 \end{aligned} \tag{Eq. 59}$$

Si noti che, trascurando gli effetti di bordo, le  $r$ -th sotto-sequenze possono essere ottenute ciclicamente e quindi la cross-correlazione può essere stimata attraverso la IFFT del prodotto delle trasformate.

Il costo computazionale di questo algoritmo è riportato in Tabella 18, in funzione del numero di bande di frequenza  $n_C$  scelte, assumendo valida l'uguaglianza  $n_C N_C = N$ . Il vincolo che deve essere rispettato, in questo caso è  $N_C \geq N_f$ . Inoltre, poiché i campioni in Range della 2D-CCF sono ottenuti attraverso una DFT su  $n_C$  campioni, deve valere anche la seguente disuguaglianza  $n_C \geq 2N_f$ .

In generale, queste approssimazioni e questi vincoli considerati produrranno delle perdite sul bersaglio, il quale si trova all' $l$ -th campione in range e  $m$ -th campione in Doppler, che possono essere esplicitate nel modo seguente:

$$L[l, m] = 20 \log_{10} \left\{ \frac{1}{n_C} \frac{\sin[\pi m (n_C - l) / N]}{\sin(\pi m / N)} \right\} \tag{Eq. 60}$$

e dipendono dal numero di bande scelto e dalla posizione del bersaglio all'interno della Mappa.



Nei casi pratici di interesse le perdite, concentrate in alcuni punti specifici della Mappa, possono raggiungere valori molto elevati. La riduzione di queste perdite può avvenire solo se si accetta un aumento del costo computazionale. In particolare, con riferimento alla modalità ottime, poiché le operazioni in Eq. 57 e Eq. 58 sono indipendenti, è sempre possibile valutare una generica mappa Range-Doppler considerando separatamente i diversi contributi in Doppler e/o Range per poi concatenarli e produrre il risultato finale. Da questa considerazione risulta chiaro che, dividendo la mappa in blocchi  $b_l$  è sempre possibile applicare il Channelization technique su singolo blocco producendo il risultato finale concatenando i diversi risultati ottenuti da blocchi differenti. Questo permette di rilassare i vincoli sul singolo blocco producendo una minore perdita a fronte di un aumento del costo in quanto l'algoritmo deve essere applicato su un numero di blocchi maggiore di uno.

### 5.3.3 Risultati teorici e scenari di riferimento

Per poter confrontare le prestazioni, in termini di tempo impiegato per l'esecuzione su GPU dei vari algoritmi, è indispensabile identificare degli scenari di riferimento (riportati in Tabella 19). In particolare, l'identificazione dell'area di copertura, in termini di distanza e velocità bistatica massima, comporta la valutazione di una mappa Range-Doppler dalle dimensioni variabili, in funzione del numero di celle di risoluzione da valutare. La massima risoluzione radar bistatica in range è approssimativamente pari a  $c/B$ , con  $c$  velocità della luce e  $B$  la larghezza di banda, quindi un segnale di questo tipo offre una risoluzione in range potenziale pari a circa 40 metri; la risoluzione in Doppler ottenibile, invece, è definita come l'inverso del tempo di integrazione (tempo di coerenza dell'elaborazione ( $T_{int}$ )) che si sceglie di utilizzare. Fissato il tempo di integrazione, la risoluzione di velocità bistatica ottenibile all'interno della mappa Range-Doppler finale può essere ricavata dalla seguente relazione:

$$\Delta v = \frac{c}{T_{int} \cdot f_c} \quad \text{Eq. 61}$$

Per tutti gli scenari considerati in Tabella 19 si prevede la valutazione di 2440 celle di risoluzione in range.

Il numero di celle di risoluzione in velocità, fissando la frequenza centrale del canale fisico, varia rispetto ai diversi scenari considerati sia perché varia il tempo di integrazione utilizzato sia perché varia la massima velocità osservabile; nello specifico, fissando il

tempo di integrazione, l'aumento della velocità massima desiderata comporta un aumento proporzionale delle celle di risoluzione in velocità.

**Tabella 19 – Scenari di riferimento per la valutazione della funzione d'ambiguità**

Descrizione	Scenario1	Scenario 2	Scenario 3	Scenario 4
Tempo di integrazione ( $T_{int}$ )	0.035 sec	0.035 sec	0.350 sec	0.350 sec
Numero di campioni integrati (N)	320000	320000	3200000	3200000
Numero di celle di risoluzione in Range ( $N_r$ )	2440	2440	2440	2440
Massima distanza bistatica relative	80 km	80 km	80 km	80 km
Numero di celle di risoluzione in Doppler ( $N_d$ )	35	67	331	661
Frequenza portante ( $f_c$ )	470 MHz	470 MHz	470 MHz	470 MHz
Massima velocità bistatica	$\pm 300$ m/s	$\pm 600$ m/s	$\pm 300$ m/s	$\pm 600$ m/s

Utilizzando i valori riportati in Tabella 19, relativi ai diversi scenari considerati, è possibile valutare il costo computazionale teorico degli algoritmi ottimi e sub-ottimo considerati utilizzando le informazioni legate al numero di celle di risoluzione in range e velocità da valutare e il numero di campioni da integrare; in particolare, in Tabella 20 si mostrano il numero di operazioni teorico da effettuare per produrre le diverse mappe e le relative perdite. Come si può notare la tecnica ottima con un minore costo è, per i diversi scenari considerati, sempre la correlation-FFT in quanto il numero di celle di risoluzioni in velocità è sempre inferiore a quello in range. Inoltre, esiste una differenza di due ordine di grandezza tra i diversi scenari considerati dovuta essenzialmente all'incremento del tempo di integrazione considerato.

Confrontando i costi della tecnica ottima migliore (Correlation-FFT) con quelli della sub-ottima considerata si può notare come esista un miglioramento prestazionale di circa due ordini di grandezza rispetto al caso ottimo anche a fronte di un aumento del numero di blocchi considerato che consente di ridurre le perdite massime all'interno dell'area sott'osservazione ottenendo valori di circa 1 dB su tutti gli scenari considerati.

**Tabella 20 – Costi computazionali e perdite degli algoritmi in relazione ai diversi scenari**

Algorithm		Scenario1	Scenario 2	Scenario 3	Scenario 4
Correlation-FFT	Moltiplicazioni	$2.15 \cdot 10^8$	$4.19 \cdot 10^8$	$2.39 \cdot 10^{10}$	$4.77 \cdot 10^{10}$
	Addizioni	$2.04 \cdot 10^8$	$3.97 \cdot 10^8$	$2.28 \cdot 10^{10}$	$4.56 \cdot 10^{10}$
Direct-FFT	Moltiplicazioni	$1.50 \cdot 10^{10}$	$1.50 \cdot 10^{10}$	$1.76 \cdot 10^{11}$	$1.76 \cdot 10^{11}$
	Addizioni	$1.42 \cdot 10^{10}$	$1.42 \cdot 10^{10}$	$1.68 \cdot 10^{11}$	$1.68 \cdot 10^{11}$
Channelization Technique	Moltiplicazioni	$1.45 \cdot 10^7$	$1.38 \cdot 10^7$	$1.55 \cdot 10^8$	$1.48 \cdot 10^8$
	Addizioni	$1.42 \cdot 10^7$	$1.35 \cdot 10^7$	$1.52 \cdot 10^8$	$1.45 \cdot 10^8$
	Perdite [dB]	4.52	8.56	4.52	8.56
	nc	8000	4000	8000	4000
Channelization Technique Time Blocks 5	Moltiplicazioni	$4.15 \cdot 10^7$	$4.03 \cdot 10^7$	$4.46 \cdot 10^8$	$4.34 \cdot 10^8$
	Addizioni	$2.88 \cdot 10^7$	$2.76 \cdot 10^7$	$2.99 \cdot 10^8$	$2.86 \cdot 10^8$
	Perdite [dB]	1.66	2.52	1.61	2.50
	nc	2000	2560	4000	2560
Channelization Technique Time Blocks 7	Moltiplicazioni	$5.45 \cdot 10^7$	$5.35 \cdot 10^7$	$5.95 \cdot 10^8$	$5.78 \cdot 10^8$
	Addizioni	$3.56 \cdot 10^7$	$3.47 \cdot 10^7$	$3.75 \cdot 10^8$	$3.58 \cdot 10^8$
	Perdite [dB]	1.35	2.04	1.30	2.02
	nc	3200	2500	4000	2500
Channelization Technique Time Blocks 11	Moltiplicazioni	$8.17 \cdot 10^7$	$7.91 \cdot 10^7$	$8.79 \cdot 10^8$	$8.55 \cdot 10^8$
	Addizioni	$5.06 \cdot 10^7$	$4.79 \cdot 10^7$	$5.15 \cdot 10^8$	$4.90 \cdot 10^8$
	Perdite [dB]	1.00	1.52	0.96	1.50
	Nc	3200	2000	3125	2000

### 5.3.4 Risultati sperimentali

Nel seguito verranno confrontati i tempi impiegati dai diversi algoritmi implementati sia su GPU che su CPU. Nello specifico verranno utilizzati i seguenti dispositivi:

un PC con le seguenti caratteristiche,

- un Processore INTEL® core™ i7 CPU 960@3.20GHz;
- Sistema operativo 64 bit e software Matlab R2009b 64bit;
- 8-cores;
- 12 GB RAM;

una scheda grafica Tesla C1060 con le seguenti caratteristiche,

- 4 GB di RAM;
- 240 cores da 1.3 GHz;
- interfaccia PCI Express x16 per il trasferimento dati;

una scheda grafica Quadro 6000 con le seguenti caratteristiche,

- 6 GB di RAM;
- 448 cores;
- interfaccia PCI Express x16 per il trasferimento dati;

La valutazione dei tempi viene effettuata utilizzando le funzioni *tic* e *toc* di Matlab e il comando `GPUSync` che viene utilizzato per attendere la fine di tutte le operazioni sulla GPU (vedi Appendice A).

In Tabella 21 si riportano i tempi di esecuzione, in secondi, relativi all'algoritmo Correlation-FFT utilizzando l'hardware descritto per i diversi scenari identificati in Tabella 19. Le lettere *S* e *D* indicano *single* e *double* rispettivamente e rappresentano la precisione del dato utilizzato; in particolare, nel caso *single* si utilizzano 4 Bytes per rappresentare la parte reale e 4 Bytes per quella immaginaria mentre, nel caso *double* vengono utilizzati 8 bytes per la parte reale e 8 per quella immaginaria.

**Tabella 21 – Tempi [secondi] di esecuzione dell'algoritmo Correlation-FFT su CPU e GPU**

Descrizione Correlation-FFT		Scenario1		Scenario2		Scenario3		Scenario4	
		S	D	S	D	S	D	S	D
CPU	1-core, 1 Processo Matlab	0,6944	1,0741	1,3192	2,0250	67,8996	117,642	135,437	232,937
	1-core, 8 processi Matlab	1,0048	1,5377	1,6347	2,5084	69,3822	119,890	136,570	236,438
	8-core, 1 Processo Matlab	0,6223	0,9573	1,1663	1,7937	60,9496	107,402	122,412	212,869
	8-core, 8 processi Matlab	0,3618	0,6532	0,6264	1,5362	21,6278	36,5570	41,4495	70,5824
GPU	Quadro 6000	0,0331	0,0676	0,0589	0,1219	1,2454	2,973	2,2548	5,9325
	Trasferimento dati Quadro 6000	0,0041	0,0062	0,0047	0,0069	0,0227	0,0427	0,0266	0,0512
	Tesla C1060	0,0508	0,1100	0,0852	0,1965	2,2674	6,3863	4,4980	12,6728
	Trasferimento dati Tesla C1060	0,0052	0,0099	0,0066	0,0127	0,0399	0,0780	0,0506	0,0977

Il tempo migliore, considerando la CPU, è ottenuto utilizzando tutti i cores presenti sulla macchina sfruttando il calcolo multithread per l'uso efficiente di tutta la risorsa a disposizione. Nel caso in cui il calcolo multithread non fosse attivo, come si può

apprezzare dai risultati in tabella, l'utilizzo del multicore risulterebbe non efficace perché le operazioni verrebbero comunque effettuate in modalità sequenziale. Inoltre, come atteso, se si utilizzasse il calcolo multithread su una macchina che non disponga di caratteristiche multicore comporterebbe un peggioramento delle prestazioni poiché, essendo la risorsa unica, i processi andrebbero in conflitto richiedendo più tempo per l'esecuzione del codice. In definitiva, nel caso specifico, l'utilizzo della CPU non permette, in tutti i casi, di ottenere tempi di elaborazione confrontabili con il tempo di integrazione utilizzato dagli scenari considerati.

Come atteso, il tempo migliore, considerando la GPU, è ottenuto utilizzando la GPU QUADRO 6000 che ha un numero di cores a disposizione doppio rispetto alla Tesla C1060; questa caratteristica si rispecchia confrontando i tempi di elaborazione che rispettano questa proporzione. L'utilizzo della GPU permette di ridurre sensibilmente i tempi di elaborazione, che restano ancora inaccettabili (ordine dei secondi) se si considerano gli scenari 3 e 4 in cui il tempo di integrazione è pari a 0.350ms.

**Tabella 22 - Tempi [secondi] di esecuzione dell'algoritmo Channelization technique su CPU e GPU**

Descrizione Channelization Technique		Scenario1		Scenario2		Scenario3		Scenario4	
		S	D	S	D	S	D	S	D
CPU	8-cores, 8 processi	0,0094	0,0166	0,0094	0,0179	0,0707	0,1552	0,0950	0,2281
	8-cores, 8 processi Time Blocks 5	0,0376	0,0528	0,0444	0,0654	0,3724	0,5863	0,4070	0,6875
	8-cores, 8 processi Time Blocks 7	0,0479	0,0716	0,0496	0,0761	0,4863	0,7372	0,5169	0,8223
	8-cores, 8 processi Time Blocks 11	0,0693	0,1028	0,0706	0,1064	0,7075	1,0169	0,7253	1,0951
GPU	Quadro 6000	0,0068	0,0246	0,0074	0,0264	0,0360	0,2182	0,0387	0,2289
	Quadro 6000 Time Blocks 5	0,0201	0,0273	0,0211	0,0287	0,1373	0,1799	0,1375	0,2008
	Quadro 6000 Time Blocks 7	0,0264	0,0361	0,0269	0,0353	0,1585	0,2239	0,1658	0,2491
	Quadro 6000 Time Blocks 11	0,0378	0,0502	0,0366	0,0501	0,2176	0,3054	0,2247	0,3195
	Tesla c1060	0,0083	0,0444	0,0097	0,0463	0,0552	0,3022	0,0663	0,3552
	Tesla c1060 Time Blocks 5	0,0295	0,0431	0,0299	0,0454	0,2046	0,3272	0,2253	0,3824
	Tesla c1060 Time Blocks 7	0,0364	0,0550	0,0367	0,0586	0,2559	0,4005	0,2695	0,4646
	Tesla c1060 Time Blocks 11	0,0521	0,0765	0,0507	0,0783	0,3722	0,5707	0,3650	0,5946

In Tabella 22 si riportano i tempi ottenuti considerando l'algoritmo sub-ottimo, descritto precedentemente, sia sfruttando la CPU in configurazione multicore/multithread sia utilizzando le due diverse GPU. I risultati ottenuti con la GPU sono in linea con quelli teorici attesi e garantiscono, in tutte le diverse configurazioni considerate, dei tempi di calcolo paragonabili con i tempi di integrazione.

## **6 Conclusioni**

Il principio di funzionamento del sistema radar passivo è noto in letteratura e, fino ad oggi, molte tipologie di trasmissioni, disponibili per differenti applicazioni, sono state usate come sorgenti di opportunità per la rivelazione e localizzazione di potenziali target. In particolare, l'utilizzo delle forme d'onda digitali per le trasmissioni TV risultano di interesse non solo perché, in Italia, stanno progressivamente sostituendo le trasmissioni televisive analogiche ma soprattutto perché le larghezze di banda utilizzate consentono un incremento della risoluzione in range ottenibile, rispetto alle trasmissioni FM.

Il lavoro di tesi presentato è stato focalizzato sulle specifiche tecniche di elaborazione del segnale DVB aggiuntive, rispetto alle operazioni principali che vengono effettuate in un sistema radar passivo, che consentono il miglioramento complessivo delle prestazioni.

Nel corso del dottorato sono state studiate ed implementate tecniche che permettono di stimare l'offset sulla frequenza portante, frequenza di campionamento e i parametri per determinare la struttura del segnale DVB; queste tecniche consentono l'implementazione degli algoritmi per il controllo dei lobi della funzione d'ambiguità noti in letteratura. Un algoritmo diverso, per risolvere il problema dei lobi laterali, è stato proposto come soluzione alternativa alle tecniche esistenti; in particolare, la risoluzione del problema è stata ottenuta con un approccio di tipo lineare, basato sul controllo dell'autocorrelazione del segnale, che consente il controllo dei lobi laterali della funzione d'ambiguità risultando efficace per bersagli a Doppler nulla. L'utilizzo di questa tecnica permette di semplificare l'architettura del sistema ricevente e la complessità delle operazioni da svolgere perché non è richiesta la sincronizzazione rispetto alla struttura del segnale DVB ed è possibile applicare questa tecnica in modo diretto ai dati ricevuti.

Un ulteriore studio è stato condotto sul problema dell'equalizzazione del segnale di riferimento, che può limitare le prestazioni dell'intero sistema ricevente, implementando un algoritmo in grado di rimuovere i contributi di disturbo dal segnale dovuti all'ambiente. Il risultato dell'analisi è in linea con i risultati teorici attesi dunque l'equalizzazione del canale di riferimento risulta essere un'operazione indispensabile anche nel caso di trasmissione digitale.

Infine, nell'ultimo periodo, il lavoro è stato focalizzato sul problema del costo computazionale cercando nuove e diverse risorse di calcolo con cui eseguire il processing

del segnale. A riguardo sono state studiate le caratteristiche delle schede grafiche (GPU) che risultano essere delle potenziali risorse in grado di sostenere tutto il costo computazionale. In particolare è stata effettuata un'analisi su tempi di calcolo impiegati per la valutazione della funzione d'ambiguità, che rappresenta una delle operazioni più onerose da effettuare nei sistemi passivi, confrontando i tempi ottenuti dall'elaborazione dei dati su CPU e GPU. L'analisi effettuata ha messo in luce i limiti imposti dagli algoritmi ottimi per la valutazione della funzione d'ambiguità che non consentono, in alcuni scenari, di ottenere tempi di elaborazione confrontabili con il tempo di integrazione. L'utilizzo di tecniche sub-ottime per il calcolo della funzione d'ambiguità consente invece di ottenere tempi decisamente confrontabili con il tempo di integrazione accentuando delle perdite sul picco in SNR limitate a circa 1 dB. Tutte le tecniche proposte e implementate sono state applicate a diversi set di dati reali acquisiti, con i sistemi riceventi descritti, al fine di verificare l'effettiva bontà degli algoritmi studiati.

La naturale estensione di questo lavoro per il futuro è l'implementazione di tutta la catena di elaborazione ricevente considerando un modello eterogeneo di elaborazione che preveda l'utilizzo congiunto di CPU e GPU in cui le operazioni costose vengono effettuate dalle schede grafiche lasciando alla CPU il controllo e l'esecuzione della parte sequenziale.



## **Ringraziamenti**

Parte dei risultati presentati in questo lavoro sono stati ottenuti grazie alla collaborazione di Selex Sistemi Integrati e del SESM. Inoltre si ringrazia la collaborazione di Selex Galileo S.P.A.

## Appendice A

Lista delle funzioni presenti all'interno della libreria GPUmat ricavate da([72]).

Name	Description
GPUinfo	Prints information about the GPU device
GPUstart	Starts the GPU environment and loads required components
GPUstop	Stops the GPU environment
colon	Colon
double	Converts a GPU variable into a Matlab double precision variable
eye	Identity matrix
GPUdouble	GPUdouble constructor
GPUeye	Identity matrix
GPUfill	Fill a GPU variable
GPUones	GPU ones array
GPUsingle	GPUsingle constructor
GPUsync	Wait until all GPU operations are completed
GPUzeros	GPU zeros array
memCpyDtoD	Device-Device memory copy
memCpyHtoD	Host-Device memory copy
ones	GPU ones array
repmat	Replicate and tile an array
setComplex	Set a GPU variable as complex
setReal	Set a GPU variable as real
setSize	Set GPU variable size
single	Converts a GPU variable into a Matlab single precision variable
zeros	GPU zeros array
GPUallocVector	Variable allocation on GPU memory
GPUmem	Returns the free memory (bytes) on selected GPU device
rand	GPU pseudorandom generator
randn	GPU pseudorandom generator
GPUrand	GPU pseudorandom generator
GPUrandn	GPU pseudorandom generator
abs	Absolute value

acos	Inverse cosine
acosh	Inverse hyperbolic cosine
and	Logical AND
asin	Inverse sine
asinh	Inverse hyperbolic sine
assign	Indexed assignement
atan	Inverse tangent, result in radians
atanh	Inverse hyperbolic tangent
ceil	Round towards plus infinity
clone	Creates a copy of a GPUtype
conj	CONJ(X) is the complex conjugate of X
cos	Cosine of argument in radians
cosh	Hyperbolic cosine
ctranspose	Complex conjugate transpose
eq	Equal
exp	Exponential
fft	Discrete Fourier transform
fft2	Two-dimensional discrete Fourier Transform
floor	Round towards minus infinity
ge	Greater than or equal
GPUround	Round towards nearest integer
GPUsinh	Hyperbolic sine
GPUsqrt	Square root
gt	Greater than
ifft	Inverse discrete Fourier transform
ifft2	Two-dimensional inverse discrete Fourier
ldivide	Left array divide
le	Less than or equal
log	Natural logarithm
log10	Common (base 10) logarithm
log1p	Compute $\log(1+z)$ accurately
log2	Base 2 logarithm and dissect floating point number
lt	Less than
minus	Minus

mrdivide	Slash or right matrix divide
mtimes	Matrix multiply
ne	Not equal
not	Logical NOT
or	Logical OR
permute	Permute array dimensions
plus	Plus
power	Array power
rdivide	Right array divide
sin	Sine of argument in radians
sinh	Hyperbolic sine
slice	Subscripted reference
sqrt	Square root
subsref	Subscripted reference
sum	Sum of elements
tan	Tangent of argument in radians
tanh	Hyperbolic tangent
times	Array multiply
vertcat	Vertical concatenation
GPUabs	Absolute value
GPUacos	Inverse cosine
GPUacosh	Inverse hyperbolic cosine
GPUand	Logical AND
GPUasin	Inverse sine
GPUasinh	Inverse hyperbolic sine
GPUatan	Inverse tangent, result in radians
GPUatanh	Inverse hyperbolic tangent
GPUceil	Round towards plus infinity
GPUconj	GPUconj(X, R) is the complex conjugate of X
GPUcos	Cosine of argument in radians
GPUcosh	Hyperbolic cosine
GPUctranspose	Complex conjugate transpose
GPUeq	Equal
GPUexp	Exponential

GPUfloor	Round towards minus infinity
GPUge	Greater than or equal
GPUgt	Greater than
GPUldivide	Left array divide
GPUle	Less than or equal
GPUlog	Natural logarithm
GPUlog10	Common (base 10) logarithm
GPUlog1p	Compute $\log(1+z)$ accurately
GPUlog2	Base 2 logarithm and dissect floating point number
GPUlt	Less than
GPUminus	Minus
GPUmtimes	Matrix multiply
GPUne	Not equal
GPUnot	Logical NOT
GPUor	Logical OR
GPUplus	Plus
GPUpower	Array power
GPUrdivide	Right array divide
GPUsin	Sine of argument in radians
GPUtan	Tangent of argument in radians
GPUtanh	Hyperbolic tangent
GPUtimes	Array multiply
GPUtranspose	Transpose
GPUuminus	Unary minus
reshape	Reshape array
round	Round towards nearest integer
display	Display GPU variable
getPtr	Get pointer on GPU memory
getSizeOf	Get the size of the GPU datatype (similar to sizeof in C)
getType	Get the type of the GPU variable
GPUisDoublePrecision	Check if GPU is double precision
iscomplex	True for complex array
isempty	True for empty GPUsingle array
isreal	True for real array

isscalar	True if array is a scalar
length	Length of vector
ndims	Number of dimensions
numel	Number of elements in an array or sub-scripted array expression.
size	Size of array
complex	Construct complex data from real and imaginary components
GPUcomplex	Construct complex data from real and imaginary components
GPUimag	Imaginary part of complex number
GPUreal	Real part of complex number
imag	Imaginary part of complex number
real	Real part of complex number

Lista di operatori utilizzabili attraverso la libreria GPUmat ricavate ([72]).

Name	Description
$a + b$	Binary addition
$a - b$	Binary subtraction
$-a$	Unary minus
$a.*b$	Element-wise multiplication
$a*b$	Matrix multiplication
$a./b$	Right element-wise division
$a./ b$	Left element-wise division
$a.^b$	Element-wise power
$a < b$	Less than
$a > b$	Greater than
$a <= b$	Less than or equal to
$a >= b$	Greater than or equal to
$a \sim= b$	Not equal to
$a == b$	Equality
$a \& b$	Logical AND
$a   b$	Logical OR
$\sim a$	Logical NOT
$a'$	Complex conjugate transpose
$a.'$	Matrix transpose

Lista di costruttori matlab per integrare con CUDA (matCUDA wrappers in [72]).

<b>Name</b>	<b>Description</b>
cublasAlloc	Wrapper to CUBLAS cublasAlloc function
cublasCgemm	Wrapper to CUBLAS cublasCgemm function
cublasCheckStatus	Check the CUBLAS status.
cublasError	Returns a structure with CUBLAS result codes
cublasFree	Wrapper to CUBLAS cublasFree function
cublasGetError	Wrapper to CUBLAS cublasGetError function
cublasGetVector	Wrapper to CUBLAS cublasGetVector function
cublasInit	Wrapper to CUBLAS cublasInit function
cublasIsamax	Wrapper to CUBLAS cublasIsamax function
cublasIsamin	Wrapper to CUBLAS cublasIsamin function
cublasResult	Returns a structure with CUBLAS error results
cublasSasum	Wrapper to CUBLAS cublasSasum function
cublasSaxpy	Wrapper to CUBLAS cublasSaxpy function
cublasScopy	Wrapper to CUBLAS cublasScopy function
cublasSdot	Wrapper to CUBLAS cublasSdot function
cublasSetVector	Wrapper to CUBLAS cublasSetVector function
cublasSgemm	Wrapper to CUBLAS cublasSgemm function
cublasShutdown	Wrapper to CUBLAS cublasShutdown function
cublasSnrm2	Wrapper to CUBLAS cublasSnrm2 function
cublasSrot	Wrapper to CUBLAS cublasSrot function
cublasSscal	Wrapper to CUBLAS cublasSscal function
cufftCheckStatus	Checks the CUFFT status
cufftDestroy	Wrapper to CUFFT cufftDestroy
cufftExecC2C	Wrapper to CUFFT cufftExecC2C
cufftExecC2R	Wrapper to CUFFT cufftExecC2R
cufftExecR2C	Wrapper to CUFFT cufftExecR2C
cufftPlan1d	Wrapper to CUFFT cufftPlan1d
cufftPlan2d	Wrapper to CUFFT cufftPlan2d
cufftResult	Returns a structure with CUFFT result codes
cufftTransformDirections	Returns a structure with CUFFT
cufftType	Returns a structure with CUFFT

## **RIFERIMENTI E BIBLIOGRAFIA**

- [1] Special Issue on Passive Radar Systems - IEE Proceedings on Radar, Sonar and Navigation, June 2005, Vol. 152, Issue 3, pp. 106-223.
- [2] Griffiths, H.D.; Baker, C.J., "Passive coherent location radar systems. Part 1: performance prediction" IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp.153-159.
- [3] Baker, C.J.; Griffiths, H.D.; Papoutsis, I., "Passive coherent location radar systems. Part 2: waveform properties", IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp.160-168.
- [4] P.E. Howland, D. Maksimiuk, G. Reitsma, "FM radio based bistatic radar" IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp. 107-115.
- [5] C. Bongioanni, F. Colone, P. Lombardo, "Performance Analysis of a Multi-Frequency FM Based Passive Bistatic Radar", 2008 IEEE Radar Conference, Rome, Italy, May 26-30, 2008.
- [6] D. Poullin, "Passive detection using broadcasters (DAB, DVB) with CODFM modulation", IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp. 143–152.
- [7] R. Saini, M. Cherniakov, V. Lenive, "Direct path interference suppression in bistatic system: DTV based radar", Proc. of the Int. Radar Conference 2003, pp. 309-314.
- [8] C. Coleman, H. Yardley, "Passive bistatic radar based on target illuminations by digital audio broadcasting", IET Radar, Sonar & Navigation, 2008, Vol. 2, Issue 5 , pp. 366-375.
- [9] R. Saini and M. Cherniakov, "DTV signal ambiguity function analysis for radar application", IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp. 133-142.
- [10] Z. Gao, R. Tao, Y. Ma, T. Shao, "DVB-T Signal Cross-Ambiguity Functions Improvement for Passive Radar", Proc. of the 2006 CIE International Conference on Radar, Shanghai, China, October 16-19, 2006.
- [11] C. Bongioanni, F. Colone, D. Langellotti, P. Lombardo, T. Bucciarelli, "A New Approach for DVB-T Cross-Ambiguity Function Evaluation", Proceedings of



- European Radar Conference (EURAD 2009), Rome, Italy, 30 September-2 October 2009.
- [12] G. Fabrizio, F. Colone, P. Lombardo, A. Farina, "Adaptive beamforming for high-frequency over-the-horizon passive radar", *IET Radar Sonar and Navigation*, August 2009, Vol. 3, Issue 4, pp. 384-405.
- [13] J.M. Thomas, H.D. Griffiths, C.J. Baker, "Ambiguity function analysis of Digital Radio Mondiale signals for HF passive bistatic radar", *Electronic Letters*, Vol. 42, No. 25, December 2006, pp. 1482-1483.
- [14] D. Tan, H. Sun, Y. Lu, M. Lesturgie, and H. Chan, "Passive radar using global system for mobile communication signal: theory, implementation and measurements", *IEE Proceedings on Radar, Sonar and Navigation*, Vol. 152, Issue 3, June 2005, pp. 116-123.
- [15] R. Zemmari, U. Nickel, W.D. Wirth, "GSM passive radar for medium range surveillance", *Proceedings of European Radar Conference (EURAD 2009)*, Rome, Italy, 30 September-2 October 2009.
- [16] F. Colone, P. Falcone, P. Lombardo, "Ambiguity Function analysis of WiMAX transmissions for passive radar", *IEEE Int. Radar Conference 2010*, May 10-14, 2010, Washington DC, USA.
- [17] F. Colone, K. Woodbridge, H. Guo, D. Mason and C. J. Baker "Ambiguity Function Analysis of Wireless LAN transmissions for passive radar", in print on *IEEE Trans. on Aerospace and Electronic Systems*.
- [18] P. Falcone, F. Colone, P. Lombardo, T. Bucciarelli, "Range Sidelobes Reduction Filters for WiFi-Based Passive Bistatic Radar", *EURAD 2009*, Rome, Italy, 30 September-2 October 2009.
- [19] H. Guo, S. Coetzee, D. Mason, K. Woodbridge and C. Baker, "Passive Radar Detection Using Wireless Networks", in *Proc. of the Int. Conf. on Radar Systems*, Edinburgh (UK), 2007, pp. 1-4.
- [20] K. Chetty, G. Smith, H. Guo and K. Woodbridge, "Target Detection in High Clutter Using Passive Bistatic WiFi Radar" , in *Proc. of IEEE Radar Conference 2009* , 4-8 May 2009, pp. 1-5.
- [21] P. Falcone, F. Colone, C. Bongioanni, P. Lombardo, "Experimental Results for OFDM WiFi-Based Passive Bistatic Radar", *IEEE Int. Radar Conference 2010*, May 10-14, 2010, Washington DC, USA.

- [22] M. Cherniakov, T. Zeng, and E. Plakidis, "Galileo signal-based bistatic system for avalanche prediction", in Proc. IGARSS03, Toulouse, France, July 2003, pp. 784-786.
- [23] M. Cherniakov, R. Saini, R. Zuo, M. Antoniou, "Space-surface bistatic synthetic aperture radar with global navigation satellite system transmitter of opportunity - experimental results", IET Proc. on Radar, Sonar & Navigation, Vol. 1, Issue 6, November 2007, pp. 447-458.
- [24] M. Cherniakov, D. Nezhlin, K. Kubin, "Air target detection via bistatic radar based on LEOS communications system", IEE Proceedings on Radar, Sonar and Navigation, Vol. 149, Issue 1, 2002, pp. 33-38.
- [25] D. Cristallini, M. Caruso, P. Falcone, D. Langellotti, C. Bongioanni, F. Colone, S. Scafè, P. Lombardo, "Space-Based Passive Radar Enabled by the New Generation of Geostationary Broadcast Satellites", 2010 IEEE Aerospace Conference, March 6-13, 2010, Big Sky, Montana (USA).
- [26] A.N. Morabito, M.G. Meyer, J.D. Sahr, "Improved computational performance for distributed passive radar processing through channelised data", IEE Proceedings on Radar, Sonar and Navigation, Vol. 152, Issue 3, June 2005, pp.179-184.
- [27] D. Langellotti, F. Colone, C. Bongioanni, P. Lombardo, "Comparative study of ambiguity function evaluation algorithms for passive radar", International Radar Symposium 2009 – IRS 2009, Sept 9-11, 2009, Hamburg, Germany.
- [28] F. Colone, D. W. O'Hagan, P. Lombardo, C. J. Baker "A multistage processing algorithm for disturbance removal and target detection in Passive Bistatic Radar", IEEE Trans. on Aerospace and Electronic Systems, Volume 45, Issue 2, April 2009, pp. 698-721.
- [29] K.S. Kulpa and Z. Czekala, "Masking effect and its removal in PCL radar", IEE Proc. Radar Sonar Navig., Vol. 152, Issue 3, June 2005, pp. 174-178.
- [30] S. R.J. Axelsson, "Improved clutter suppression in random noise radar", URSI 2005 Commission F Symposium on Microwave Remote Sensing of the Earth, Oceans, Ice, and Atmosphere, April 2005.
- [31] Gunner, A., Temple, M.A., and Claypoole, Jr, R.J.: 'Direct-path filtering of DAB waveform from PCL receiver target channel', Electronic Letters, 2003, 39, (1), pp. 1005-1007.

- [32] R. Cardinali, F. Colone, C. Ferretti, P. Lombardo, "Comparison of clutter and multipath cancellation techniques for passive radar", IEEE 2007 Radar Conference, Boston, (MA), USA, March 2007.
- [33] F. Colone, R. Cardinali, P. Lombardo, "Cancellation of clutter and multipath in passive radar using a sequential approach", IEEE 2006 Radar Conference, Verona (NY), USA, April 24-27, 2006, pp. 393-399.
- [34] S. Haykin, "Adaptive filter theory", Prentice Hall, Upper Saddle River, NJ, USA, 2002, 4th edn.
- [35] P. Lombardo, F. Colone, C. Bongianni, A. Lauri, T. Bucciarelli, "PBR activity at INFOCOM: adaptive processing techniques and experimental results", 2008 IEEE Radar Conference, Rome, Italy, May 26-30, 2008.
- [36] J. G. Proakis, "Digital Communications", McGraw Hill International Edition, Fourth Edition, 2001.
- [37] D.N. Godard. Self-recovering equalization and carrier tracking in two-dimensional data communication systems. IEEE Trans. Commun, COMM-28:1867–1875, 1980.
- [38] J. R. Treichler, B. G. Agee, "A new approach to Multipath correction of Constant Modulus Signal", IEEE Transaction on Acoustics, Speech, and Signal Processing, vol. 31, issue 2, April 1983, pp. 459-472.
- [39] C. Zhou, J. D. Sahr, M. G. Meyer, D. M. Gidner, "Ground clutter subtraction algorithm for VHF passive radar observation of the upper atmosphere", URSI 2002, Maastricht August 2002.
- [40] F. Colone, R. Cardinali, P. Lombardo, O. Crognale, A. Cosmi, A. Lauri, T. Bucciarelli, "Space-Time CMA for multipath removal on the reference signal exploited by Passive Bistatic Radar", IET Radar Sonar and Navigation, Vol. 3, N. 3, June 2009 p. 253-264.
- [41] R. Cardinali, F. Colone, P. Lombardo, O. Crognale, A. Cosmi, A. Lauri, "Mutipath cancellation on reference antenna for passive radar which exploits FM transmission", 2007 IET International Radar Conference, Radar 2007, Edinburgh (UK), 15-18 October 2007.
- [42] J. J. Shynk, R. P. Gooch, (1996), "The constant modulus array for cochannel signal copy and direction finding", IEEE Trans. on Signal Processing, Vol. 44, Issue 3, March 1996, pp. 652-660.

- [43] S.K. Sharma, L. Shafai, "Beam focusing properties of circular monopole array antenna on a finite ground plane", *IEEE Transactions on Antennas and Propagation*, Volume 53, Issue 10, Oct. 2005, pp. 3406 – 3409.
- [44] H. Yuan, K. Hirasawa, Y. Zhang, "The mutual coupling and diffraction effects on the performance of a CMA adaptive array", *IEEE Transactions on Vehicular Technology*, Volume 47, Issue 3, Aug. 1998, pp. 728 – 736.
- [45] H. Furukawa, Y. Kamio, H. Sasaoka, "Cochannel interference reduction and path-diversity reception technique using CMA adaptive array antenna in digital land mobile communications", *IEEE Transactions on Vehicular Technology*, Volume 50, Issue 2, March 2001, pp. 605 – 616.
- [46] A. Lauri, F. Colone, R. Cardinali, P. Lombardo, "Analysis and emulation of FM radio signals for passive radar", 2007 IEEE Aerospace Conference, Big Sky (MT), USA, 3-10 March 2007.
- [47] IEEE Standard for Information technology: "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications" (IEEE Std 802.11™-1999) – and following Supplements and Amendments (IEEE Stds 802.11a™-1999, 802.11b™-1999, 802.11b™-1999/Cor 1-2001, and 802.11g™-2003)
- [48] R. Bricocchi, T. Bucciarelli, and P. T. Melacci, "Radar sensitivity and resolution in presence of range sidelobe reducing networks designed using linear programming," *The Radio and Electronic Engineer*, vol. 54, pp. 224–250, June 1984.
- [49] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming (web page and software). <http://cvxr.com/cvx>, December, 2009.
- [50] IEEE Std 802.16™-2004, IEEE Std 802.16e™-2005, IEEE Standard for Local and Metropolitan Area Network, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems
- [51] H. A. Harms, L. M. Davis, and J. Palmer, "Understanding the signal structure in DVB-T signals for passive radar detection," in *Proc. of 2010 IEEE Radar Conference*, Washington, DC, USA, May 2010, pp. 532–537.
- [52] Gao Z W, Tao R, Wang Y. Analysis and side peaks identification of Chinese DTTB signal ambiguity functions for passive radar. *Sci China Ser F-Inf Sci*, 2009, 52(8): 1409–1417, doi: 10.1007/s11432-009-0149-y

- [53] European Telecommunications Standard Institute, "Digital video broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television", EN 300 744, V1.1.2, 1997.
- [54] ETSI EN 302 583 V1.1.0 (2008-01) European Standard (Telecommunications series)
- [55] van de Beek, J.J.; Sandell, M.; Borjesson, P.O., "ML estimation of time and frequency offset in OFDM systems", *IEEE Transactions on Signal Processing*, vol.45, no.7, pp.1800-1805, Jul 1997.
- [56] Speth, M.; Fechtel, S.A.; Fock, G.; Meyr, H., "Optimum receiver design for wireless broad-band systems using OFDM. I", *IEEE Transactions on Communications*, vol.47, no.11, pp.1668-1677, Nov 1999.
- [57] D. Langellotti, C. Bongioanni, F. Colone, P. Lombardo, "Impact of Synchronization on the Ambiguity Function shape for PBR based on DVB-T signals", accepted for IRS 2010.
- [58] P. Lombardo, F. Colone, C. Bongioanni, A. Lauri, T. Bucciarelli, "PBR activity at INFOCOM: adaptive processing techniques and experimental results", 2008 IEEE Radar Conference, Rome, Italy, May 26-30, 2008.
- [59] C. Bongioanni, F. Colone, P. Lombardo, "Performance Analysis of a Multi-Frequency FM Based Passive Bistatic Radar", 2008 IEEE Radar Conference, Rome, Italy, May 26-30, 2008.
- [60] P. Lombardo, F. Colone, C. Bongioanni, "Comparison of different approaches for a Multi-Frequency FM Based Passive Bistatic Radar", *Proceedings del SET-125 Symposium on Sensors and Technology for Defence Against Terrorism*, Mannheim, Germany, April 22-25, 2008.
- [61] C. Bongioanni, F. Colone, S. Bernardini, L. Lelli, A. Stavolo, P. Lombardo, "Passive radar prototypes for multifrequency target detection", *Signal Processing Symposium 2007*, Jachranka (Poland), 24-26th May 2007.
- [62] C. Bongioanni, F. Colone, T. Martelli, R. D'Angeli, P. Lombardo, "Exploiting polarimetric diversity to mitigate the effect of interferences in FM-based passive radar", *International Radar Symposium 2010 – IRS 2010*, September 9-11, 2010, Vilnius, Germany.
- [63] M. Villano, F. Colone, P. Lombardo, "Adaptive Clutter Suppression in Passive Phased Array Radar", *International Radar Symposium IRS 2009*, Hamburg, Germany, 9-11 September 2009, pp. 343-347.

- [64] M. Malanowski, K. Kulpa, "Digital beamforming for Passive Coherent Location radar", 2008 IEEE Radar Conference, Rome (Italy), 26-30 May.
- [65] A. Farina, "Antenna-Based Signal Processing Techniques for Radar Systems", Artech House, Norwood, MA.
- [66] ETSI EN 300 744 V1.5.1 (2004-11) European Standard (Telecommunications series)
- [67] M. Skolnik, "Radar Handbook", Mc Graw Hill , Third Edition, 1984, ISBN: 978-0-07-148547-0.
- [68] Materiale consultabile dal sito web <http://www.analog.com>
- [69] Materiale consultabile dal sito web <http://www.trimble.com>.
- [70] Materiale consultabile dal sito web <http://www.nvidia.com>
- [71] Materiale consultabile dal sito web <http://www.mathworks.it/index.html>
- [72] Materiale consultabile dal sito web <http://www.gp-you.org>