



**POLITECNICO**  
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE  
E DELL'INFORMAZIONE

# Simulazione e Test OTA di un Sistema OFDM con e senza Adaptive Modulation and Coding ML-guidato

PROGETTO DI PROVA FINALE DI  
SISTEMI DI COMUNICAZIONE

Autori: **Nicola Gallucci e Matteo Malagrino**

Codici Matricola: 218452 215425  
Anno Accademico: 2024-25  
Docente: Francesco Linsalata

# Abstract

La crescente diffusione delle reti di comunicazione wireless avanzate richiede sistemi sempre più efficienti e adattabili alle condizioni variabili del canale. Questa tesi si concentra sulla simulazione e la validazione **Over-The-Air (OTA)** di un sistema di comunicazione basato su **OFDM (Orthogonal Frequency Division Multiplexing)** utilizzando **SDR (Software-Defined Radio) Adalm-Pluto**. L'obiettivo principale è implementare e testare un sistema di **Adaptive Modulation and Coding (AMC)** che ottimizza la trasmissione in base alle condizioni del canale.

In una prima fase, vengono effettuati test su un sistema **OFDM base**, sia in assenza che in presenza di **codifica di canale**, valutando le prestazioni attraverso simulazioni e test **OTA**. Successivamente, per classificare lo stato della comunicazione e adattare dinamicamente i parametri di modulazione e codifica, viene sviluppato un modello di **Machine Learning (ML)**. Tale modello analizza parametri chiave come **SNR stimato**, **BER**, permettendo di selezionare la configurazione ottimale per massimizzare l'affidabilità e l'efficienza spettrale del sistema.

La validazione sperimentale avviene tramite test **OTA (Over The Air)** in ambienti controllati, valutando le prestazioni del sistema in presenza di **interferenze**, **fading** e **effetto Doppler**, tipici di scenari dinamici. I risultati ottenuti dimostrano l'efficacia dell'approccio proposto, evidenziando il ruolo cruciale dell'**intelligenza artificiale** nell'ottimizzazione adattiva delle comunicazioni wireless.

**Parole Chiave:** OFDM, AMC, Codifica di canale, Adalm-Pluto, SDR, Machine Learning

# Indice

<b>Abstract</b>	<b>i</b>
<b>Indice</b>	<b>ii</b>
<b>Introduzione</b>	<b>iv</b>
<b>1 Sistema OFDM</b>	<b>1</b>
1.1 Introduzione . . . . .	1
1.2 Cos'è la modulazione OFDM? . . . . .	1
1.3 Vantaggi della modulazione OFDM . . . . .	2
1.4 Implementazione nel Sistema di Comunicazione . . . . .	3
<b>2 Codifica di canale</b>	<b>5</b>
2.1 Introduzione . . . . .	5
2.2 Che cos'è la codifica di canale? . . . . .	5
2.2.1 Codifiche a blocco . . . . .	6
2.2.2 Codifiche convoluzionali . . . . .	6
2.2.3 Decodifica di Viterbi . . . . .	7
2.3 Implementazione codifica e decodifica di canale nel Sistema di Comunicazione	8
<b>3 Analisi delle performance del sistema</b>	<b>10</b>
3.1 Settaggio del sistema e raccolta dei dati . . . . .	10
3.1.1 Stima dei parametri: BER ed SNR . . . . .	10
3.1.2 Analisi performance . . . . .	11
<b>4 Sistema AMC</b>	<b>16</b>
4.1 Introduzione . . . . .	16
4.2 Cos'è un sistema AMC? . . . . .	17
4.2.1 Principi di funzionamento . . . . .	18
4.2.2 Applicazioni . . . . .	19

	<b>Indice</b>	iii
4.3	Classificatore basato su Support Vector Machine per AMC . . . . .	21
4.3.1	Support Vector Machines (SVM) . . . . .	21
4.3.2	Descrizione del Modello . . . . .	25
4.3.3	Fase di generazione dei dati di training e addestramento . . . . .	25
4.3.4	Integrazione nel Sistema di Comunicazione . . . . .	27
4.3.5	Risultati sperimentali . . . . .	29
5	<b>Conclusioni</b>	<b>31</b>
	<b>Bibliografia</b>	<b>32</b>

# Introduzione

Negli ultimi anni, le reti di comunicazione wireless hanno subito un'evoluzione significativa, con l'introduzione di tecnologie avanzate come il **5G** e lo sviluppo di sistemi sempre più adattivi ed efficienti. In questo contesto, le tecniche di **Adaptive Modulation and Coding (AMC)** giocano un ruolo fondamentale nell'ottimizzazione delle trasmissioni, adattando dinamicamente i parametri del segnale in base alle condizioni variabili del canale.

Questa tesi si propone di studiare, simulare e validare un sistema di comunicazione basato su **Orthogonal Frequency Division Multiplexing (OFDM)** con capacità adattive, sfruttando **Software-Defined Radio (SDR) Adalm-Pluto** per test **Over-The-Air (OTA)** in scenari realistici. L'obiettivo principale è dimostrare l'efficacia dell'integrazione di un modello di **Machine Learning (ML)** per la classificazione dello stato della comunicazione e la selezione ottimale dei parametri di modulazione e codifica.

In una prima fase, viene analizzato un sistema **OFDM base**, valutando le sue prestazioni sia in simulazione che attraverso test OTA, con e senza codifica di canale. Successivamente, viene implementato un meccanismo adattivo basato su **ML**, il quale utilizza parametri chiave come **SNR stimato e BER** per classificare lo stato del canale e regolare dinamicamente i parametri di trasmissione.

L'ultima fase della ricerca riguarda la validazione sperimentale del sistema, eseguendo test OTA in ambienti controllati per misurare le prestazioni in presenza di fenomeni reali come **interferenze, fading e effetto Doppler**. I risultati ottenuti evidenziano il potenziale delle tecniche adattive nel migliorare la robustezza e l'efficienza spettrale del sistema rispetto a configurazioni statiche.

Questa ricerca rappresenta un passo verso l'ottimizzazione autonoma delle reti wireless, con possibili applicazioni future nelle **reti 5G** e nei sistemi di comunicazione per scenari dinamici.

# 1 | Sistema OFDM

## 1.1. Introduzione

Nel contesto delle moderne telecomunicazioni, la crescente domanda di velocità di trasmissione dati e la necessità di migliorare l'efficienza spettrale hanno portato allo sviluppo di tecniche avanzate di modulazione. Tra queste, l'**Orthogonal Frequency Division Multiplexing** (OFDM) si è affermata come una delle soluzioni più efficaci, trovando applicazione in numerosi standard di comunicazione, come il Wi-Fi (IEEE 802.11), il 4G LTE e il 5G.

## 1.2. Cos'è la modulazione OFDM?

La modulazione OFDM è una tecnica di modulazione numerica multiportante che si contraddistingue per l'ottimo utilizzo della banda del canale. Il funzionamento alla base di questa tecnica consiste nella suddivisione di una sequenza binaria su N diversi flussi, trasmessi a divisione di frequenza mediante forme d'onda ortogonali. Ciò che accade è che le diverse frequenze sono spaziate tra loro di  $\Delta$  Hz:

$$f_n = f_0 + \Delta \cdot \left(n - \frac{N}{2}\right)$$

con  $n = 0, 1, \dots, N-1$  e sono utilizzate contemporaneamente, mentre su ognuna di esse si realizza una modulazione numerica a due o più livelli con NRZ (Non Return to Zero) rettangolare di durata T. Indicando con  $\underline{a}_n^k = a_{n_c}^k + ja_{n_s}^k$  le coordinate nel piano dell'involuppo complesso di un generico punto della costellazione realizzata per la portante  $f_n$  all'istante  $t = kT$ , il segnale OFDM può essere scritto come:

$$x_{OFDM}(t) = \sum_{k=-\infty}^{\infty} \delta(t-kT) * \left( \text{rect}_T \sum_{n=0}^{N-1} \underline{a}_n^k e^{j2\pi[\Delta(n-\frac{N}{2})]t} \right)$$

in cui la prima sommatoria (su k) identifica gli istanti di simbolo, e la seconda (su n) le diverse portanti. Questa espressione non vincola la durata T di un simbolo ad un valore

particolare; deve però risultare  $T \geq T_0 = \frac{1}{\Delta}$  affinché siano rese ortogonali tra loro le frequenze  $f_n = f_0 + \Delta \cdot (n - \frac{N}{2})$  e il ricevitore sia in grado di calcolare i valori di  $\underline{a}_n^k$  per tutti gli  $n$  presenti all'istante  $t = KT$ . Nella modulazione OFDM si hanno quindi delle finestre di ricezione di intervallo  $T_0$  (anche detto *periodo principale* del simbolo OFDM) intervallate tra loro da un *tempo di guardia*  $T_g = T - T_0$ , utilizzato per evitare la corruzione del valore dei simboli a causa di **ISI** (*Interferenza Intersimbolica*) nei canali che introducono distorsione lineare.[6]

### 1.3. Vantaggi della modulazione OFDM

Si è già chiarito quanto vasto sia, nel mondo delle telecomunicazioni, l'utilizzo dell'OFDM e quale sia il suo funzionamento, ma non ci si è ancora soffermati su quali aspetti la rendano così efficiente. La modulazione a divisione ortogonale di frequenza è ampiamente utilizzata nei sistemi wireless grazie alla sua robustezza ai fenomeni di multipath, alla sua efficienza spettrale e alla facilità di implementazione su piattaforme come le Software-Defined Radio (SDR).

I vantaggi che l'OFDM garantisce sono:

- **Resistenza al multipath e all'interferenza intersimbolo (ISI):** la suddivisione del segnale in più sottoportanti ortogonali consente una maggiore tolleranza ai ritardi di propagazione e riduce la complessità dell'equalizzazione
- **Efficienza spettrale elevata:** l'uso di sottoportanti ortogonali permette una maggiore densità di trasmissione dei dati rispetto ad altre tecniche di multiplexing
- **Robustezza al fading selettivo:** grazie alla suddivisione della banda in molte sottoportanti, il sistema può compensare il deterioramento del segnale in specifiche porzioni dello spettro
- **Adattabilità alle condizioni del canale:** l'OFDM consente di implementare modulazioni adattative, regolando dinamicamente la modulazione e la codifica in base alla qualità del canale.
- **Facilità di implementazione mediante elaborazione digitale:** l'uso della trasformata di Fourier veloce (FFT/IFFT) semplifica la modulazione e la demodulazione, rendendo l'OFDM ideale per piattaforme digitali.[3]

## 1.4. Implementazione nel Sistema di Comunicazione

La modulazione OFDM è stata implementata nel sistema di comunicazione tramite l'utilizzo del software MATLAB, nel quale è stato possibile realizzare e simulare un sistema OFDM reale definendo i parametri caratteristici e programmando trasmettitore e ricevitore sulla base di essi.

I parametri, editabili direttamente sull'interfaccia del trasmettitore e del ricevitore, sono:

- **FFT Length:** definisce la lunghezza della Fast Fourier Transform
- **Cycle Prefix Length:** definisce la lunghezza del Cicle Prefix<sup>1</sup>
- **Number of Subcarriers:** definisce il numero di sottoportanti ortogonali utilizzati nella modulazione
- **Subcarriers Spacing:** indica la banda di ogni sottoportante
- **Pilot Subcarrier Spacing:** indica la banda della sottoportante pilota o principale<sup>2</sup>
- **Channel Bandwidth:** larghezza di banda del canale

In questo modo si riesce a generare una forma d'onda con modulazione OFDM da trasmettere, definita come:

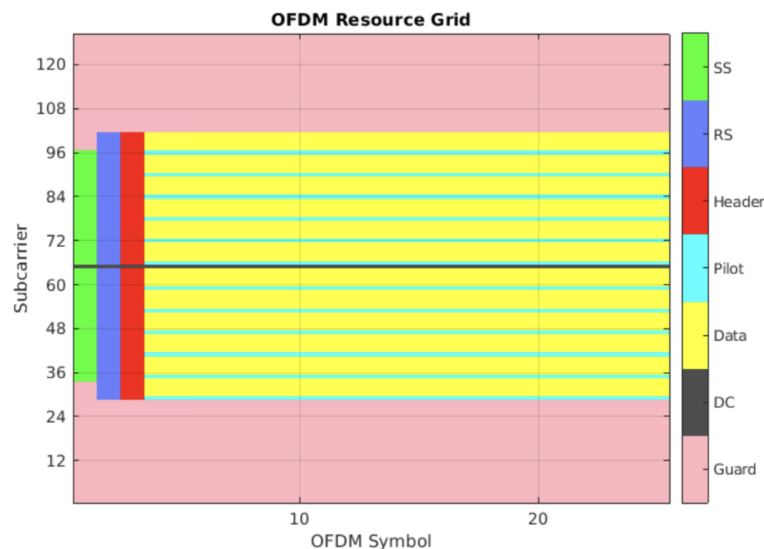


Figura 1.1: Struttura del frame trasmesso [11]

<sup>1</sup>Il Cycle Prefix determina il tempo di guardia in un sistema multiportante che utilizza FFT/IFFT [9]

<sup>2</sup>La sottoportanti pilota vengono utilizzate allo scopo di scontare gli offset di frequenza e di fase nel ricevitore



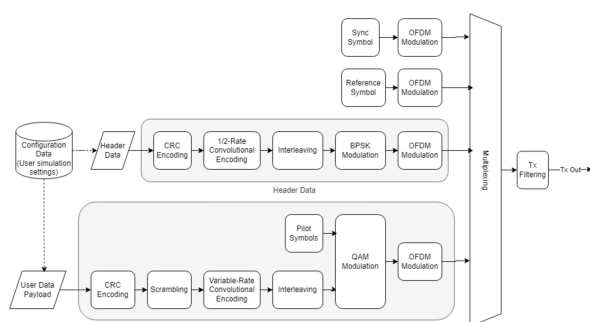


Figura 1.2: Modulo di trasmissione

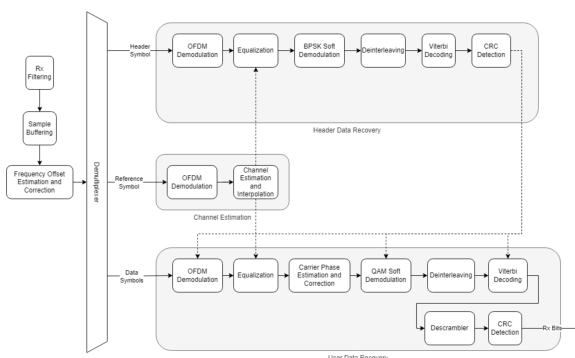


Figura 1.3: Modulo di ricezione

Sigla	Descrizione
SS: Synchronization Symbol	Un segnale di sincronizzazione (sync) viene trasmesso come primo simbolo nel frame. Il segnale di sincronizzazione è costituito da un segnale a 62 sottoportanti centrato su DC generato utilizzando uno ZadOff-Chu. Questo segnale è progettato per essere indipendente dalla larghezza di banda.
RF: Reference Symbol	Il simbolo successivo nel frame di trasmissione è il simbolo di riferimento. Noto sia al trasmettitore che al ricevitore, il simbolo di riferimento consente al ricevitore di confrontare il simbolo di riferimento ricevuto con il simbolo di riferimento noto per stimare il canale.
HS: Header Symbol	Il simbolo di header trasmette la larghezza di banda, lo schema di modulazione della sottoportante e la velocità di codifica dei simboli di dati OFDM per aiutare il ricevitore a decodificare il frame trasmesso. Le informazioni sono abbastanza importanti da essere trasmesse con ampi margini di segnalazione e codifica per massimizzare la decodifica corretta.
PS: Pilot Signals	Utilizzato per combattere il jitter di fase osservato a frequenze di trasmissione più elevate. Un pilot viene trasmesso a intervalli di sottoportante fissi all'interno dei simboli di dati per fornire un riferimento di fase al ricevitore.
DC and Guard Subcarriers	Le sottoportanti nulle al limite dello spettro di trasmissione vengono utilizzate per limitare l'energia spettrale a una larghezza di banda specificata. Anche la sottoportante in DC viene annullata per mantenere l'energia del segnale entro l'intervallo lineare dell'amplificatore di potenza.

Tabella 1.1: Legenda delle sigle utilizzate

## 2 | Codifica di canale

### 2.1. Introduzione

I sistemi di comunicazione sono generalmente affetti da interferenze e disturbi di varia natura tra cui motivi strutturali del mezzo trasmissivo, dell'ambiente circostante e di fenomeni fisici che insieme contribuiscono alla corruzione dei segnali trasmessi.

Per sopperire a questa lista di eventi spiacevoli, dal punto di vista delle telecomunicazioni, nel tempo si sono sviluppate e affinate sempre più tecniche di protezione dei dati e, più in generale, dell'intero processo di scambio di informazioni, arrivando a sistemi sempre più complessi ma maggiormente affidabili.

Tra i principali metodi utilizzati vi è quello di suddividere in apposite fasi di codifica e decodifica il segnale trasmesso, al fine di controllare, correggere e limitare le compromissioni di cui si è parlato poc'anzi.

### 2.2. Che cos'è la codifica di canale?

Una forte rilevanza nell'intero processo di comunicazione è assunta dalla cosiddetta codifica di canale, nella quale lo scopo principale è quello di aggiungere ridondanza ad un flusso binario trasmesso ad una determinata velocità  $f_b$  su un canale numerico, in modo da realizzare una protezione **FEC** (*Forward Error Correction*), capace di ridurre la probabilità di errore  $P_e$  al ricevitore.

Una categoria molto vasta è quella dei codici a blocco, che operano suddividendo il messaggio da proteggere in blocchi disgiunti, codificati in modo indipendente; una diversa classe è quella dei codici convoluzionali, che invece trattano il messaggio come una sequenza priva di suddivisioni, da cui calcolare una nuova sequenza (a velocità maggiore) che ne rappresenta la codifica.

### 2.2.1. Codifiche a blocco

Nella codifica a blocchi si ha una sequenza di  $k$  bit in ingresso (blocco) da proteggere e per ognuna di esse viene generata una *codeword* di lunghezza pari a  $n = k + q > k$  bit, in cui sono stati aggiunti  $q$  bit di protezione in funzione dei  $k$  bit di blocco. Questo algoritmo è indicato come *codice*( $n, k$ ) e l'efficienza ad esso associata è misurata attraverso il *tasso di codifica* o anche conosciuto come **Code Rate**:

$$R_c = \frac{k}{n} < 1$$

ed indica il rapporto tra i bit informativi rispetto al numero totale di bit contenuti nella codeword. In questo modo viene determinata anche la nuova velocità di trasmissione dell'informazione sul canale, che risulta essere pari a

$$f'_b = \frac{f_b}{R_c}$$

Esiste una correlazione tra  $f'_b$  e il rapporto tra energia media del bit e la densità spettrale della potenza di rumore  $\frac{E_b}{N_0}$ : all'aumentare della velocità del segnale, questo rapporto peggiora, influenzando negativamente la probabilità di errore del decisore.

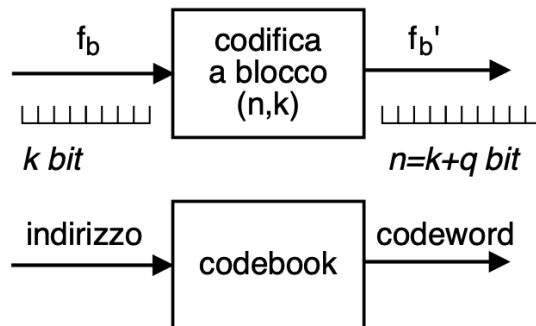


Figura 2.1: Schema di una codifica a blocchi [6]

### 2.2.2. Codifiche convoluzionali

La codifica convoluzionale si differenzia rispetto alla codifica a blocchi poiché essa produce una sequenza binaria i cui valori dipendono da gruppi di bit di ingresso temporalmente sovrapposti. La notazione utilizzata per il codice convoluzionale è  $CC(n, k, K)$ , nel quale esso è ritenuto in grado di poter generare gruppi di  $n$  bit di uscita (sequenza  $x_j$ ) in base alla conoscenza di  $K$  simboli di ingresso (sequenza  $m_j$ ), ognuno composto da  $k$  bit.

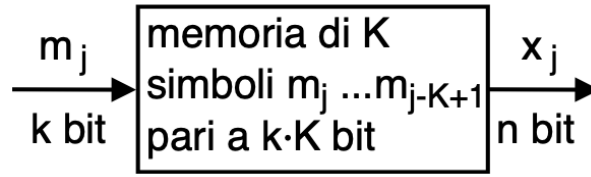


Figura 2.2: Schema a blocchi della codifica convoluzionale [6]

La CC, come si può osservare in figura, ha in ingresso una sequenza di  $k$  bit che vengono memorizzati in un registro a scorrimento in cui per ogni nuovo simbolo  $m_j$  che entra da sinistra i precedenti scorrono a destra, dimenticando il bit più vecchio. Ognuno degli  $n$  bit di uscita  $x_j(i)$  con  $i=1,2,\dots,n$  è calcolato eseguendo una somma modulo 2 tra alcuni dei  $k \cdot K$  bit di ingresso, individuati da un vettore generatore  $g_i (i = 1, 2, \dots, n)$  costituito da una parola binaria di  $k \cdot K$  bit, zero o uno a seconda se l' $i$ -esimo sommatore modulo due sia connesso o meno al corrispondente bit della finestra di ingresso.

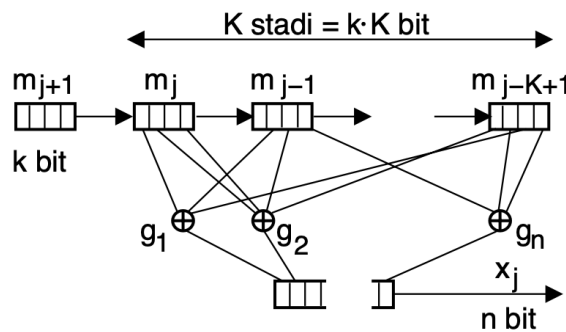


Figura 2.3: Funzionamento della codifica convoluzionale [6]

### 2.2.3. Decodifica di Viterbi

L'algoritmo di Viterbi è un metodo ricorsivo utilizzato per determinare la sequenza di stati più probabile in un processo di Markov<sup>1</sup> nascosto, basandosi su una sequenza di osservazioni.

Il Funzionamento dell'Algoritmo di Viterbi consiste nel costruire un diagramma a traliccio (trellis), che rappresenta tutti i possibili stati del sistema nel tempo e le transizioni tra di essi. Ogni nodo nel traliccio corrisponde a uno stato in un determinato istante temporale, mentre ogni arco rappresenta una possibile transizione tra stati consecutivi.

Il processo di decodifica si articola in tre fasi principali:

<sup>1</sup>È un modello matematico che descrive un sistema che evolve nel tempo in cui il futuro dipende solo dallo stato presente e non dagli stati passati

- **Inizializzazione:** Si assegnano probabilità iniziali a ciascuno stato al tempo iniziale ( $t=0$ )
- **Ricorsione:** Per ogni istante di tempo successivo, si calcola la probabilità cumulativa per ciascuno stato, considerando tutte le possibili transizioni dagli stati precedenti. Durante questo processo, si memorizza anche il percorso che ha portato alla probabilità massima per ogni stato
- **Terminazione e Ricostruzione del Percorso:** Una volta raggiunto l'ultimo istante temporale, si identifica lo stato con la probabilità massima. Risalendo indietro attraverso i percorsi memorizzati, si ricostruisce la sequenza di stati più probabile che ha generato la sequenza di osservazioni

Questo approccio consente di ridurre significativamente la complessità computazionale rispetto a una ricerca esaustiva, poiché elimina iterativamente i percorsi meno probabili [4], [7].

### 2.3. Implementazione codifica e decodifica di canale nel Sistema di Comunicazione

Il sistema di comunicazione realizzato codifica la forma d'onda attraverso codifica di canale convoluzionale nella quale si è posto un vincolo  $K$ , sia per i dati che per l'header, pari a 7 bit; ciò indica che il codificatore memorizza i 7 bit precedenti per generare gli output. Il code rate è settato a  $1/2$  di default e i **polinomi generatori**, che determinano quali bit del registro di scorrimento devono essere usati per generare ciascun bit di output, sono uguali per dati ed header e impostati, anch'essi di default, a [171 133] in **codifica ottale** (quindi 1111001 e 1011011 in binario).

Per rendere più dinamico il sistema di comunicazione e valutarne le prestazioni in ambienti e condizioni di canale differenti, si è scelto di utilizzare anche la tecnica di **puncturing** (*perforazione*) per aumentare il code rate senza intervenire direttamente sui parametri del codificatore e di conseguenza non aumentarne la complessità.

Il concetto alla base di questa tecnica di perforazione consiste nel ricavare un codice con rapporto  $\frac{k}{n}$  da un codice con rapporto "basilare"  $1/2$ . Si ottiene mediante cancellazione di alcuni bit nell'uscita del codificatore, i quali sono cancellati secondo un **vettore di perforazione** che mantiene inalterati i bit corrispondenti a '1' nel vettore di perforazione, mentre elimina quelli corrispondenti a '0'<sup>2</sup>. I vettori di perforazione utilizzati sono i

---

<sup>2</sup>Fonte: [17]

seguenti:

Vettore di perforazione	Code-Rate	Implicazione
[1 1]	1/2	Se il codificatore emette 2 bit per ogni bit in ingresso, trasmettiamo sempre entrambi.
[1 1 0 1]	2/3	Su 4 bit generati, ne trasmettiamo solo 3 poiché il terzo è eliminato.
[1 1 1 0 0 1]	3/4	Su 6 bit generati, ne trasmettiamo solo 4 poiché il quarto e il quinto sono eliminati.
[1 1 1 0 0 1 1 0 0 1]	5/6	Su 10 bit generati, ne trasmettiamo solo 6 poiché il quarto, il quinto, l'ottavo e il nono sono eliminati.

Tabella 2.1: Vettore di perforazione e relativa implicazione

La scelta di utilizzare la tecnica di puncturing permette anche di ottenere code-rate maggiori senza modificare i requisiti di banda richiesti e di conseguenza senza modificare l'efficienza spettrale del sistema.

Al ricevitore, per decodificare i bit si è scelto di utilizzare la decodifica di Viterbi e applicarla direttamente attraverso la funzione disponibile in MATLAB: "vitdec"<sup>3</sup>

<sup>3</sup>Si rimanda al codice in bibliografia per ulteriori approfondimenti [13]

## 3 | Analisi delle performance del sistema

### 3.1. Settaggio del sistema e raccolta dei dati

Per valutare il sistema in maniera completa ed efficace si è avuta la necessità di testarlo in ambienti e in situazioni differenti, così da poter confrontare i risultati e avere un quadro preciso rispetto a quelle che sono le reali prestazioni del sistema di comunicazione. I dati raccolti, come si vedrà in dettaglio successivamente, sono stati ottenuti in due modi :

- **Simulazione del sistema:** il sistema è stato simulato in maniera del tutto virtuale attraverso il software MATLAB nel quale sono stati utilizzati gli stessi parametri del sistema reale. Le simulazioni sono state eseguite variando opportunamente le caratteristiche per adattarle alle casistiche successivamente applicate nella realtà<sup>1</sup>
- **Testing del sistema reale:** attraverso l'utilizzo di *Software Defined Radio* ADALM-PLUTO e il software MATLAB si è stati in grado di programmare le SDR in modo tale da trasmettere e ricevere correttamente un segnale di testo, utilizzando una modulazione OFDM in vari ambienti e con parametri editabili direttamente dall'interfaccia utente.

Al termine della fase di testing, con i dati raccolti si è stati in grado di valutare analiticamente le prestazioni del sistema, prendendo come riferimenti i tradizionali parametri di valutazione delle performance nelle telecomunicazioni come il **BER** (*Bit Error Rate*), l'**SNR** (*Signal Noise Ratio*).

#### 3.1.1. Stima dei parametri: BER ed SNR

Per valutare il sistema è stato necessario calcolare i suddetti parametri, i quali sono stati ricavati in modo tale da approssimare al meglio il valore reale.

---

<sup>1</sup>Il riferimento al codice, simulato e non, è disponibile in versione integrale nella repository GitHub presente nella bibliografia [8]

Nel calcolo del BER, al ricevitore è nota la stringa di testo da ricevere (*stringa attesa*) e questo consente di confrontarla con la stringa ricevuta attraverso la funzione MATLAB `comm.errorRate(tx,rx)`, determinando il numero di bit effettivamente errati e ottenendo il valore del parametro.<sup>2</sup>

Per quanto riguarda invece il calcolo dell'SNR, l'approssimazione utilizzata prevede di mediare i valori di potenza ottenuti tramite la densità spettrale di potenza del segnale ricevuto, all'interno della banda di interesse (tra -100MHz e 100MHz), così da ottenere la **potenza media del segnale utile**. Successivamente si procede selezionando un punto al di fuori della banda (circa 129MHz) in cui non è presente il segnale utile così da catturare la **potenza di rumore**, necessaria per determinare il valore finale. Per ottenere il valore approssimato di SNR si procede sottraendo alla potenza di segnale utile quella di rumore (poiché i dati raccolti sono in dBm).

### 3.1.2. Analisi performance

Per comprendere al meglio le valutazioni che verranno fatte da questo momento in poi è necessario definire gli ultimi elementi del sistema che verranno presentati e richiamati di frequente. La forma d'onda oggetto della trasmissione tra dispositivi, che trasporta l'informazione di testo, è una forma d'onda a coseno rialzato del tipo:

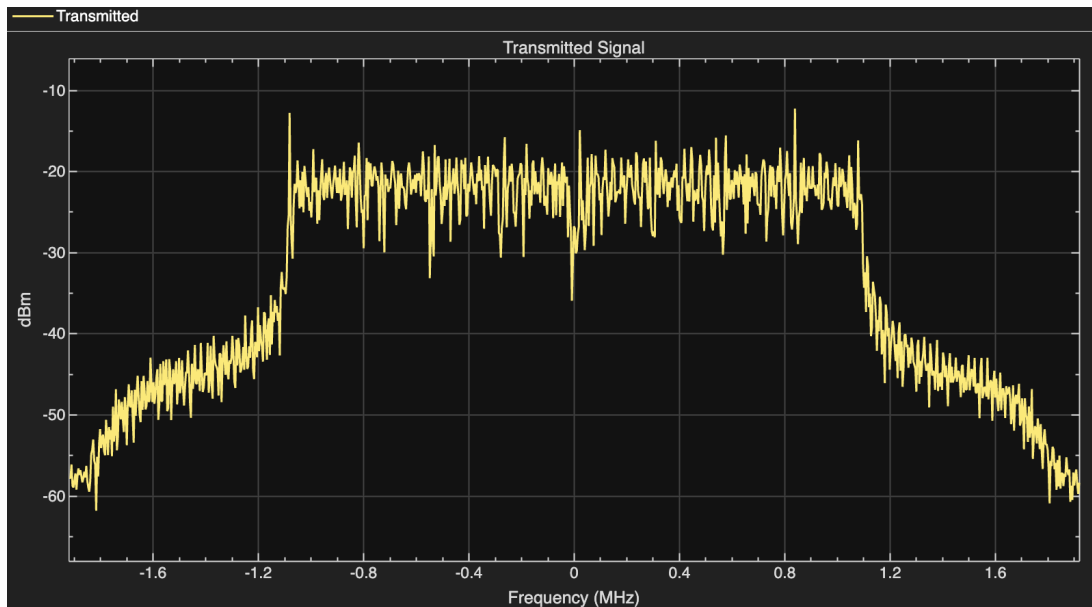


Figura 3.1: Segnale trasmesso

---

<sup>2</sup>Si consideri, come verrà mostrato successivamente, la presenza del BER teorico, ottenuto tramite simulazione software in MATLAB



Oltre a quest’ultima, restano da definire i parametri standard della trasmissione:

Parametri	Valore
FFT	128
Cycle Prefix	32
Number of Subcarriers	72
Subcarrier spacing]	30kHz
Pilot Subcarrier spacing	9
Bandwidth	3MHz
Carrier Frequency	865MHz
Gain Tx	[-10, -5, 0]
Gain Rx	[40, 60, 70]

Tabella 3.1: Parametri del sistema

La prima fase consiste nel simulare tutti i contesti di trasmissione di interesse, per avere un quadro di riferimento sul quale basare le misurazioni reali. Nelle immagini riportate di seguito (figura 3.2) sono state simulate virtualmente le curve di **BER vs SNR** e confrontate con la **curva teorica**, considerando la presenza e non della codifica di canale e dei diversi code rate:

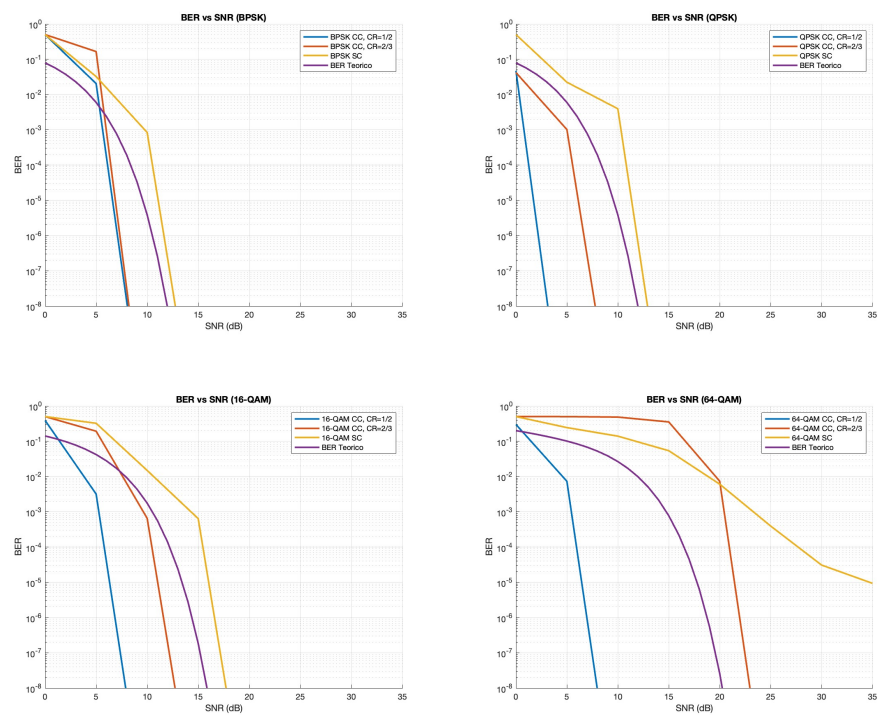


Figura 3.2: Grafici BER vs SNR simulati, con diverse modulazioni.

Da questi risultati, a priori, si può notare che in condizioni reali, la presenza delle non idealità distorce i risultati rispetto alla curva teorica, nel caso di sistema senza codifica di canale. Si noti anche che da questo grafico è possibile notare quali siano le modulazioni più robuste (si vedano la BPSK e la QPSK rispetto alle modulazioni 16 e 64-QAM) e quale sia l'influenza della correzione d'errore imposta dalla codifica di canale (massima nel caso del code-rate pari ad  $1/2$ ). Sotto la guida di questi risultati è possibile procedere con le misurazioni reali, attraverso i dispositivi SDR, per valutare concretamente quali siano gli effetti della trasmissione quando l'onda viene propagata in spazio libero, immersa in continui stimoli di rumore ed interferenze di vario genere.

Possiamo osservare quello che accade nel sistema reale guardando il segnale ricevuto, le costellazioni ottenute per ogni ambiente e i valori delle misurazioni in contesti differenti:

Modulazione	Code-Rate	BER	SNR	Codifica di canale	Distanza
BPSK	1/2	0	11	SI	5m
BPSK	2/3	0	9	SI	5m
BPSK	3/4	$1.9 \cdot 10^{-3}$	12.6	SI	5m
QPSK	1/2	0	10.3	SI	5m
QPSK	2/3	0	11.6	SI	5m
QPSK	3/4	0	12.9	SI	5m
16-QAM	1/2	$2 \cdot 10^{-2}$	10	SI	5m
16-QAM	2/3	$6.3 \cdot 10^{-3}$	11	SI	5m
16-QAM	3/4	$1.6 \cdot 10^{-2}$	12	SI	5m
64-QAM	1/2	$5 \cdot 10^{-3}$	12.6	SI	5m
64-QAM	2/3	$3 \cdot 10^{-1}$	10.8	SI	5m
64-QAM	3/4	$1.88 \cdot 10^{-1}$	11.5	SI	5m
BPSK	-	0	19.8	NO	5m
QPSK	-	0	19.9	NO	5m
16-QAM	-	$1.13 \cdot 10^{-1}$	18	NO	5m
64-QAM	-	$4.8 \cdot 10^{-1}$	20	NO	5m
QPSK	1/2	$5.2679 \cdot 10^{-5}$ (*)	11	SI	20m
QPSK	1/2	$4.587 \cdot 10^{-2}$ (*)	8	SI	20m (**)
QPSK	1/2	0	15,9	SI	60cm (**)
QPSK	1/2	0	17	SI	1.20m (**)

Tabella 3.2: Valori rilevati tramite SDR

<sup>3</sup>(\*): Massimo gain in ricezione pari a 72 e in trasmissione pari a -3

<sup>4</sup>(\*\*): Presenza di un ostacolo durante la trasmissione

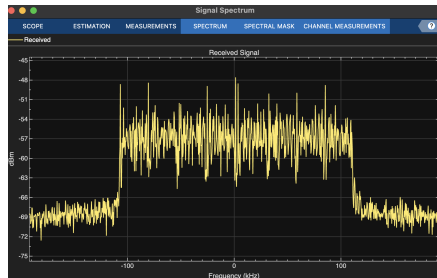


Figura 3.3: Forma d'onda ricevuta da una trasmissione in BPSK

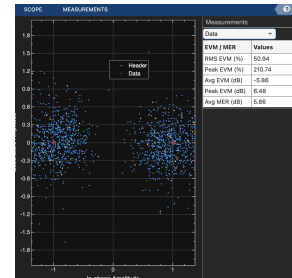


Figura 3.4: Costellazione ricevuta da una trasmissione in BPSK

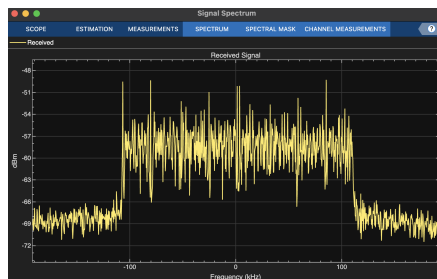


Figura 3.5: Forma d'onda ricevuta da una trasmissione in QPSK

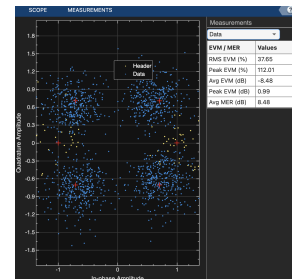


Figura 3.6: Costellazione ricevuta da una trasmissione in QPSK

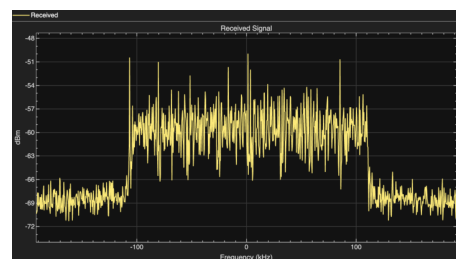


Figura 3.7: Forma d'onda ricevuta da una trasmissione in 16-QAM

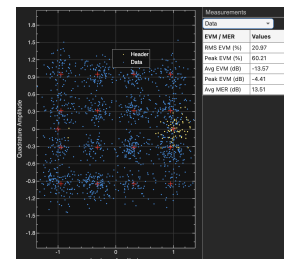


Figura 3.8: Costellazione ricevuta da una trasmissione in 16-QAM

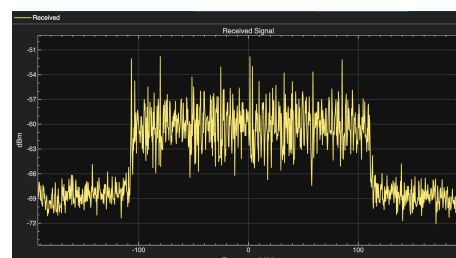


Figura 3.9: Forma d'onda ricevuta da una trasmissione in 64-QAM

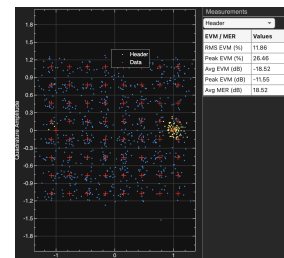


Figura 3.10: Costellazione ricevuta da una trasmissione in 64-QAM

**Nota:** per quanto riguarda le rilevazioni reali del sistema, si tenga pur sempre conto delle approssimazioni nel calcolo dei parametri di analisi che in certi casi possono dare risultati diversi da quelli attesi.

Come ultimo elemento di analisi delle prestazioni consideriamo il **Limite di Shannon**. Il limite di Shannon stabilisce un teorema fondamentale che ci dice qual è la massima quantità di informazioni che possiamo trasmettere su un canale di comunicazione, dato un certo livello di rumore e una larghezza di banda. È un concetto chiave per comprendere i limiti delle comunicazioni digitali e progettare sistemi di comunicazione efficienti. La formula che determina questo limite è la seguente:

$$R_b \leq C = B \log_2 \left( 1 + \frac{P_s}{N_0 B} \right) = B \log_2 \left( 1 + \frac{E_b R_b}{N_0 B} \right) \quad (3.1)$$

dove  $C$  è indicata come **capacità del canale** e  $B$  è la **banda del canale** (nel nostro caso  $B = 3\text{MHz}$ ).

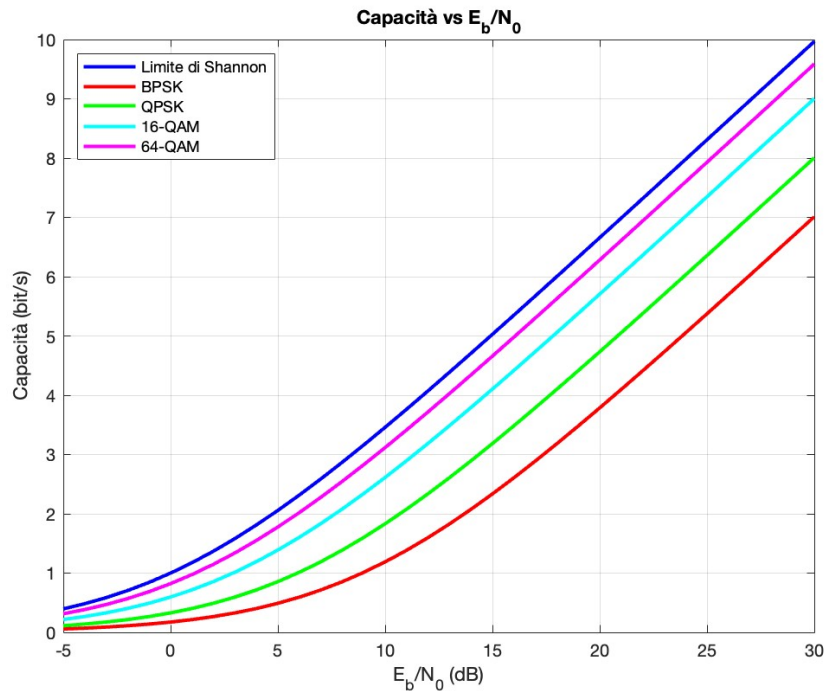


Figura 3.11: Limite di Shannon e  $C$  vs  $E_b/N_0$  nelle varie modulazioni

# 4 | Sistema AMC

## 4.1. Introduzione

La crescente esigenza di sistemi di comunicazione wireless ad alte prestazioni ha portato allo sviluppo di tecniche avanzate come l'**Adaptive Modulation and Coding (AMC)**. L'AMC è una strategia che consente di ottimizzare la trasmissione dei dati in funzione delle condizioni variabili del canale di comunicazione, migliorando così l'efficienza spettrale e la robustezza del sistema. In questa sezione, esploreremo i fondamenti teorici dell'AMC, il ruolo cruciale di un **classifier** basato su algoritmi di **Machine Learning (ML)**, e i risultati ottenuti attraverso l'implementazione pratica con le schede SDR **Adalm-Pluto**.

L'AMC si basa sulla selezione dinamica di diverse configurazioni di modulazione e codifica per adattarsi in tempo reale alle condizioni del canale, come la **SNR (Signal-to-Noise Ratio)** e il **BER (Bit Error Rate)**. A seconda della qualità del canale, vengono utilizzati schemi di modulazione ad alta efficienza, come **QAM (Quadrature Amplitude Modulation)** e **PSK (Phase Shift Keying)**, insieme a diversi tassi di codifica per ottimizzare l'affidabilità e la velocità di trasmissione.

Un aspetto fondamentale nell'implementazione dell'AMC è la capacità di monitorare e classificare lo stato del canale in tempo reale. Per questo motivo, in questo lavoro, è stato scelto di utilizzare un **Support Vector Machine (SVM)**, un algoritmo di **Machine Learning** che permette di classificare efficacemente il canale in base ai parametri osservati, come la **SNR** e il **BER**. L'SVM è un metodo di apprendimento supervisionato particolarmente adatto a scenari con elevata dimensionalità, come quelli che caratterizzano i sistemi di comunicazione.

Successivamente, verrà illustrato l'algoritmo scelto per la trasmissione dei dati, che combina la modulazione adattativa con la codifica di canale, e come l'SVM viene integrato per ottimizzare dinamicamente la configurazione di modulazione e codifica. Infine, presenteremo i risultati degli esperimenti condotti utilizzando le schede **Adalm-Pluto**, che hanno permesso di testare l'implementazione dell'AMC in scenari reali di comunicazione,

valutando le prestazioni del sistema in presenza di interferenze, fading e altre condizioni dinamiche del canale.

I risultati ottenuti dimostrano l'efficacia dell'approccio proposto, evidenziando come l'uso combinato dell'AMC e dell'apprendimento automatico possa migliorare significativamente le prestazioni dei sistemi di comunicazione wireless in ambienti complessi e variabili.

## 4.2. Cos'è un sistema AMC?

L'**Adaptive Modulation and Coding (AMC)** è una tecnologia fondamentale nei moderni sistemi di comunicazione, che migliora l'efficienza e l'affidabilità della trasmissione dei dati, adattando dinamicamente gli schemi di modulazione e i tassi di codifica in base alle condizioni del canale in tempo reale.

I canali wireless sono intrinsecamente rumorosi e soggetti a interferenze, il che può comportare errori nei dati trasmessi. Gli schemi di modulazione e codifica sono utilizzati per codificare i dati in modo che siano più robusti rispetto al rumore e alle interferenze. Tuttavia, diversi schemi di modulazione e codifica presentano compromessi tra velocità di trasmissione e affidabilità. La scelta dello schema ottimale dipende dalle condizioni del canale, e la sfida sta nel trovare il miglior equilibrio tra questi due fattori.

L'AMC affronta questo problema regolando dinamicamente lo schema di modulazione e codifica in base alla qualità istantanea del canale. L'idea di base è quella di utilizzare schemi di modulazione e codifica di ordine più elevato quando la qualità del canale è buona, aumentando così la velocità di trasmissione ma riducendo l'affidabilità. Al contrario, quando la qualità del canale è scarsa, si utilizzano schemi di modulazione e codifica di ordine inferiore, riducendo la velocità di trasmissione ma migliorando l'affidabilità. [15]

I principali vantaggi dell'AMC includono un aumento dell'**efficienza spettrale**, un miglioramento dell'**affidabilità del link** e una migliore gestione di traffico eterogeneo. Utilizzando uno schema di modulazione e codifica ad alta capacità quando la qualità del canale è buona, il sistema è in grado di raggiungere velocità di trasmissione più elevate senza compromettere l'affidabilità. Al contrario, utilizzando uno schema di modulazione e codifica a bassa capacità quando la qualità del canale è scarsa, il sistema è in grado di mantenere un link affidabile anche in condizioni difficili.

Nonostante i numerosi vantaggi, l'implementazione della tecnica **AMC** nei sistemi di comunicazione wireless presenta alcune sfide. Una delle principali difficoltà riguarda la necessità di un feedback **CQI** (Channel Quality Indicator) preciso e tempestivo. Per adattare in tempo reale lo schema di modulazione e codifica, il sistema richiede un feedback

accurato sulla qualità del link radio. Questo feedback deve essere inviato rapidamente ed efficientemente dal ricevitore al trasmettitore, minimizzando al contempo il sovraccarico del sistema.

#### 4.2.1. Principi di funzionamento

La tecnica **Adaptive Modulation and Coding (AMC)** implica generalmente due componenti principali: il meccanismo di feedback dell'**Indicatore di Qualità del Canale (CQI)** e l'algoritmo di selezione dello **Schema di Modulazione e Codifica (MCS)**. [15]

Una stima accurata della qualità del canale è fondamentale per il corretto funzionamento dell'adattamento del link. Senza una conoscenza precisa delle condizioni del canale, diventa difficile selezionare gli schemi di modulazione e codifica più appropriati. Di seguito viene descritto come la stima della qualità del canale contribuisce all'AMC:

#### Metriche di decisione

La qualità del canale è spesso stimata utilizzando metriche come il **Signal-to-Noise Ratio (SNR)**, il **Bit Error Rate (BER)** o il **Signal-to-Interference-plus-Noise Ratio (SINR)**. Queste metriche forniscono misure quantitative della qualità del canale. [14]

In particolare, il **SNR** è un indicatore critico della qualità del canale. Rappresenta il rapporto tra la potenza del segnale e la potenza del rumore ed è direttamente legato alla probabilità di errore durante la trasmissione dei dati. Valori di SNR più elevati generalmente indicano condizioni di canale migliori.[14]

#### Selezione dello Scheme di Modulazione e Codifica

Il cuore dell'AMC risiede nella selezione intelligente degli schemi di modulazione e codifica. Questo processo tiene conto della qualità del canale e del compromesso desiderato tra la velocità di trasmissione e la resilienza agli errori.

La scelta tra gli schemi di modulazione e codifica comporta un compromesso tra la velocità di trasmissione e la resilienza agli errori. Gli schemi di modulazione di ordine elevato possono trasmettere più dati per simbolo, ma sono più sensibili agli errori in canali rumorosi. Al contrario, gli schemi di modulazione di ordine inferiore offrono una maggiore resilienza agli errori, ma a costo di una minore velocità di trasmissione. [14]

In condizioni ottimali del canale, l'AMC potrebbe selezionare una modulazione di ordine

elevato (ad esempio, 64-QAM) per massimizzare la velocità di trasmissione. Questa scelta sfrutta l'affidabilità del canale per ottenere un throughput elevato.

Al contrario, in condizioni di canale sfavorevoli, con elevato rumore o interferenza, l'AMC può optare per schemi di codifica a correzione d'errore più robusti (ad esempio, Turbo Code o LDPC) per garantire l'integrità dei dati, anche a scapito della velocità di trasmissione.[14]

## Adattamento del Tasso di Codifica

Regolando il tasso di codifica, l'AMC ottimizza la capacità di correzione degli errori per adattarsi precisamente alle condizioni del canale. In buone condizioni del canale, può essere utilizzato un tasso di codifica inferiore per massimizzare la velocità di trasmissione, poiché il canale è meno incline a errori. Al contrario, in condizioni sfavorevoli, un tasso di codifica maggiore assicura una migliore resilienza agli errori.[14]

### 4.2.2. Applicazioni

#### Efficienza Energetica nelle Reti 5G

Numerosi studi hanno esplorato l'implementazione di schemi di modulazione e codifica adattiva (AMC) per migliorare l'efficienza energetica nelle reti 5G. Silvestre Malta et al. hanno proposto un sistema che utilizza l'algoritmo di apprendimento rinforzato SARSA per gestire le modalità di riposo delle stazioni base 5G, ottimizzando il consumo energetico e mantenendo la Qualità del Servizio (QoS). Il modello del sistema integra la modulazione del traffico e le politiche di consumo energetico, simulando i modelli di traffico tramite un processo di Poisson per valutare efficacemente i risparmi energetici [1].

Allo stesso modo, Fateh Elsherif et al. hanno sviluppato un approccio innovativo che controlla dinamicamente lo stato acceso/spento delle stazioni base in tempo reale, inquadrando la sfida come un Processo Decisionale di Markov (MDP). Il loro algoritmo minimizza il consumo totale di energia, considerando sia i costi operativi che quelli di commutazione. Utilizzando un algoritmo di rollout della politica con campionamento Monte Carlo, hanno ottenuto significativi risparmi energetici in vari scenari simulati, dimostrando l'adattabilità dell'approccio alle condizioni di rete reali [1].

#### Comunicazione di Tipo Macchina (MTC)

Nel contesto della comunicazione di tipo macchina (MTC), le organizzazioni di standardizzazione cellulare come 3GPP stanno avanzando lo standard Long-Term Evolution (LTE)



per supportare queste applicazioni. L'MTC è significativo poiché facilita la comunicazione tra dispositivi in un ambiente di rete, facendo grande affidamento su schemi AMC efficienti per ottimizzare la trasmissione dei dati e migliorare la capacità della rete. Implementando tecniche adattive, la rete può regolare i tassi di modulazione e codifica in base alle condizioni in tempo reale, migliorando le prestazioni complessive e l'affidabilità per le varie applicazioni MTC [5].

### Applicazione nelle comunicazioni satellitari

L'Adaptive Modulation and Coding (AMC) si rivela una tecnologia fondamentale nel panorama delle **comunicazioni satellitari**, poiché consente di superare le sfide uniche che caratterizzano questo mezzo trasmissivo.

Uno degli aspetti cruciali, evidenziato da diversi studi, è l'importanza di una **stima accurata del canale**. Infatti, l'AMC si basa sulla conoscenza delle condizioni del canale per scegliere la modulazione e la codifica più appropriate. Errori nella stima del canale possono compromettere l'efficacia dell'AMC, portando a prestazioni subottimali. Pertanto, tecniche robuste di stima del canale sono essenziali per l'implementazione efficace dell'AMC nelle comunicazioni satellitari. [2]

Un altro tema rilevante è l'applicazione dell'AMC ai sistemi di comunicazione satellitare mobile, come quelli basati su **LTE**. In questi scenari, il movimento dei terminali introduce ulteriori variazioni nella qualità del segnale, rendendo necessario un adattamento dinamico dei parametri di trasmissione. Studi in questo ambito [10] propongono schemi AMC che integrano **meccanismi di controllo della potenza e interleaving dei simboli** per mitigare gli effetti della mobilità e garantire comunicazioni affidabili anche in condizioni dinamiche.

L'ottimizzazione del **goodput, ovvero il numero di bit di dati consegnati correttamente per unità di tempo**, rappresenta un ulteriore obiettivo cruciale. Diversi lavori di ricerca si concentrano sullo sviluppo di algoritmi AMC che massimizzano il goodput, tenendo conto non solo delle condizioni del canale fisico, ma anche di altri fattori, come il ritardo di propagazione e la potenza disponibile. Questi algoritmi, spesso di tipo "**greedy**" [16], cercano di trovare il miglior compromesso tra efficienza spettrale e robustezza del collegamento, scegliendo la combinazione di modulazione e codifica più adatta per ogni pacchetto di dati.

Questi casi di studio sottolineano il ruolo critico dell'AMC nei moderni sistemi di comunicazione, in particolare nel migliorare l'efficienza energetica, supportare la comunicazione di tipo macchina e migliorare le prestazioni nelle applicazioni in tempo reale.

### 4.3. Classificatore basato su Support Vector Machine per AMC

Per ottimizzare le prestazioni del sistema di comunicazione OFDM con Adaptive Modulation and Coding (AMC), è stato implementato un classificatore basato su **Support Vector Machine (SVM)**. Questo modello fornisce una classificazione dello stato e della qualità della comunicazione sulla base del BER calcolato e del SNR stimato. Il prodotto viene utilizzato per definire i nuovi parametri di trasmissione (ordine di modulazione e tasso di codifica) che costituiscono il payload del messaggio di feedback inviato dal ricevitore al trasmettitore. Questo classificatore consente, quindi, di adattare dinamicamente la modulazione e il code rate in base alle condizioni del canale, migliorando l'affidabilità della trasmissione.

#### 4.3.1. Support Vector Machines (SVM)

Le **Support Vector Machines (SVM)**, note anche come **Support Vector Networks**, sono modelli di apprendimento supervisionato basati sul concetto di **marginе massimo**. Questi modelli utilizzano algoritmi associati per analizzare i dati e svolgere compiti di **classificazione e regressione**.

Oltre alla classificazione lineare, le SVM sono in grado di gestire **classificazioni non lineari** in modo efficiente grazie al **trucco del kernel**. Questo approccio permette di rappresentare i dati attraverso un insieme di confronti di similarità a coppie tra i punti originali, utilizzando una **funzione kernel** che li trasforma in coordinate in uno spazio delle caratteristiche di dimensione superiore. In questo spazio, la classificazione lineare diventa possibile.

Essendo modelli basati sul margine massimo, le SVM risultano **resistenti al rumore nei dati**, ad esempio in presenza di esempi classificati erroneamente.

Tuttavia, come ogni strumento potente, anche le SVM presentano alcune sfide. In particolare:

- Possono essere **computazionalmente onerose**, specialmente quando si lavora con dataset di grandi dimensioni.
- La **selezione dei parametri**, come il tipo di kernel, la regolarizzazione e i parametri specifici del kernel, richiede un'attenta ottimizzazione per garantire le migliori prestazioni del modello.

Nonostante queste sfide, quando applicate correttamente, le SVM risultano **estremamente efficaci e affidabili**, offrendo risultati precisi e robusti nella classificazione e nella regressione.[18]

### Caso lineare

L'applicazione più semplice e comune delle Support Vector Machines (SVM) si verifica nei casi in cui due classi di punti dati possono essere separate da una retta (in due dimensioni) o da un piano (in tre o più dimensioni). In questi scenari, l'SVM cerca di trovare la **migliore retta** (o **iperpiano**) che separi le classi.

In uno scenario ideale, i punti dati appartenenti a classi diverse possono essere separati da una retta in uno spazio bidimensionale o da una superficie piana in spazi di dimensione superiore. In questi casi, i dati vengono definiti **linearmente separabili**.

L'algoritmo SVM ha l'obiettivo di trovare l'**iperpiano** che **massimizza il margine** tra le due classi. Questo approccio consente al modello di essere **robusto al rumore** e di avere una migliore **capacità di generalizzazione**.

Nel modello SVM, non tutti i punti dati influenzano il confine decisionale. Solo i punti dati che si trovano più vicini all'iperpiano, appartenenti a ciascuna classe, definiscono il margine. Questi punti sono chiamati **vettori di supporto**.

Se i dati di addestramento sono **linearmente separabili**, possiamo selezionare due iperpiani paralleli che separano le due classi di dati, in modo che la distanza tra di essi sia la più grande possibile. La regione delimitata da questi due iperpiani è chiamata **margine**, e l'iperpiano del margine massimo è l'iperpiano che si trova a metà strada tra di essi.

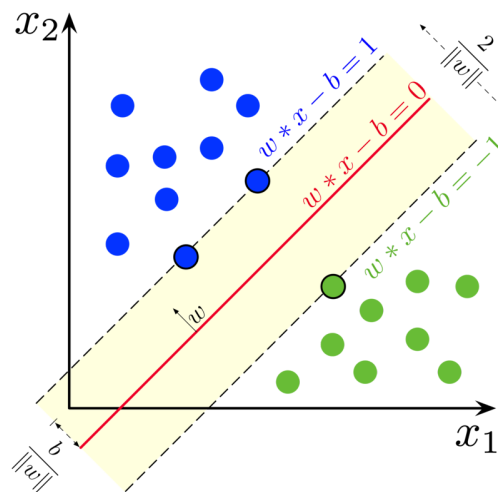


Figura 4.1: Iperpiano di massimo margine [18]

Per estendere l'SVM ai casi in cui i dati non sono separabili linearmente, si introduce la funzione di **perdita hinge** definita come

$$\max(0, 1 - y_i(w^T x_i - b)) \quad (4.1)$$

Per tutti i calcoli si rimanda a [18].

## Caso Non Lineare e Kernel Trick

In molti problemi reali, i punti dati delle diverse classi non sono distribuiti in modo semplice e separabile linearmente. Si immagini di cercare di classificare un insieme di punti dati dove le classi formano schemi complessi, come cerchi concentrici. In questi casi, tracciare una singola linea retta (o un iperpiano piatto in dimensioni superiori) per separare le due classi non è fattibile. È qui che entra in gioco il **kernel trick**: trasformando i dati in uno spazio di dimensioni superiori, dove le relazioni non lineari possono diventare lineari. Il **kernel trick** è un metodo che consente agli SVM di eseguire questa trasformazione in modo efficiente senza dover calcolare esplicitamente le coordinate dei dati in dimensioni superiori. Invece di mappare manualmente i punti dati in uno spazio di dimensioni superiori, il kernel trick ci permette di calcolare direttamente nello *spazio originale* il prodotto scalare dei punti trasformati, risparmiando così tempo e risorse computazionali. Esistono diverse funzioni di kernel utilizzate per gli SVM:

- **Kernel lineare:**

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$$

Questo è il caso più semplice, dove i dati vengono lasciati nello spazio originale senza alcuna trasformazione.

- **Kernel polinomiale:**

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + c)^d$$

Dove  $c$  è una costante e  $d$  è il grado del polinomio. Questo kernel consente di mappare i dati in uno spazio di dimensioni superiori tramite funzioni polinomiali.

- **Kernel gaussiano (RBF - Radial Basis Function):**

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$$

Questo kernel è molto utilizzato, in quanto mappa i dati in uno spazio di dimensioni infinite, permettendo la separazione di dati non lineari in modo molto flessibile.

- Kernel sigmoide:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\alpha \mathbf{x}_i^\top \mathbf{x}_j + c)$$

Dove  $\alpha$  è un parametro di scala e  $c$  è una costante. Questo kernel è simile alla funzione di attivazione in una rete neurale. [19]

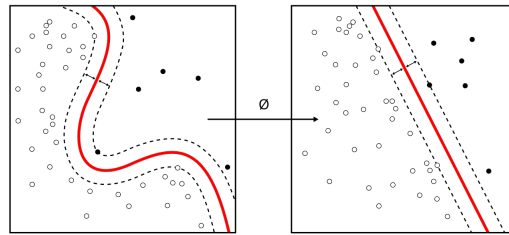


Figura 4.2: Trasformazione con Kernel non lineare [18]

### 4.3.2. Descrizione del Modello

Il classificatore SVM utilizza un insieme di caratteristiche estratte dal segnale ricevuto per determinare lo stato della comunicazione. In particolare, gli ingressi del modello includono:

- **SNR stimato:** Indice del rapporto segnale-rumore.
- **BER (Bit Error Rate):** Indicatore della qualità della decodifica.

L'uscita del classificatore è un punteggio discreto compreso tra 0 e 2, che rappresenta lo stato della comunicazione e determina la configurazione ottimale di modulazione e codifica. Di seguito sono riportati i parametri associati ai vari casi di classificazione della comunicazione (modulazione - code rate):

- **0 - Comunicazione Pessima:** QPSK, 1/2
- **1 - Comunicazione Discreta :** 16-QAM, 2/3
- **1 - Comunicazione Ottima :** 64-QAM, 3/4

### 4.3.3. Fase di generazione dei dati di training e addestramento

Per addestrare il modello SVM, è stata utilizzata una simulazione del sistema OFDM [12] in cui sono stati variati in modo casuale diversi parametri:

- **SNR ricevuto:** da 0 a 25 dB.
- **Modulazione:** BPSK, QPSK, 16-QAM, 64-QAM.
- **Code rate:** 1/2, 2/3, 3/4.
- **Velocità del ricevitore:** da 0 a 3 km/h.
- **CFO (Carrier Frequency Offset):** da 0 a 3 ppm.

I risultati ottenuti dalla simulazione sono stati salvati in un file CSV e etichettati per la classificazione in tre classi:

- **Classe 2:** se  $SNR \geq 18$  dB e  $BER < 10^{-2}$ .
- **Classe 1:** se  $8 < SNR < 18$  dB e  $10^{-2} \leq BER < 8 \times 10^{-2}$ , oppure se  $SNR \geq 18$  dB e  $BER > 10^{-2}$ .
- **Classe 0:** in tutti gli altri casi.

Il dataset finale contiene circa 13.000 campioni, suddivisi in training e test set per ottimizzare le prestazioni del modello SVM.

Il modello SVM è stato addestrato utilizzando un kernel **RBF** con parametri ottimizzati tramite *Grid Search*. Le metriche di valutazione del modello includono:

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2122
1	0.99	0.99	0.99	1207
2	0.99	0.99	0.99	1491
accuracy			1.00	4820
macro avg	0.99	0.99	0.99	4820
weighted avg	1.00	1.00	1.00	4820

Di seguito viene riportata la matrice di confusione del modello:

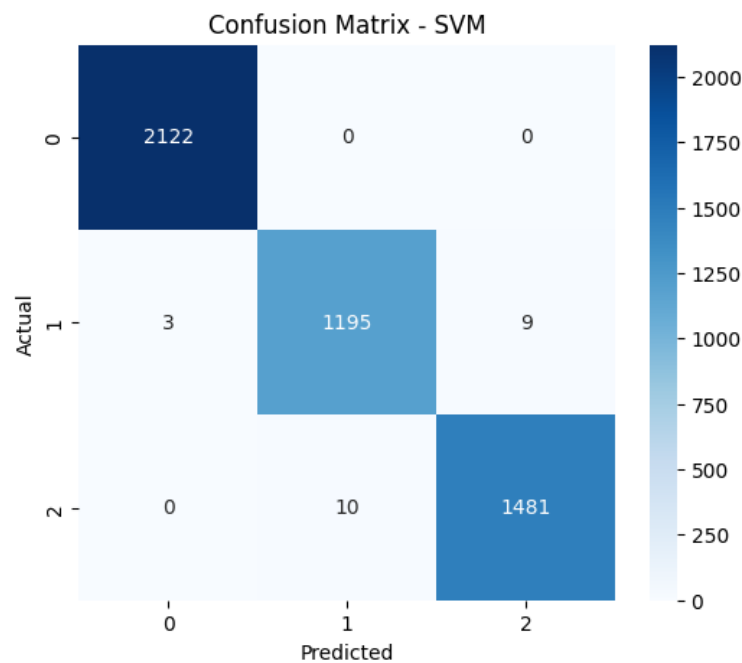


Figura 4.3: Matrice di confusione del modello SVM.

#### 4.3.4. Integrazione nel Sistema di Comunicazione

Il classificatore è stato integrato nel ricevitore OFDM, il quale, dopo aver stimato i parametri del canale, utilizza il modello SVM per determinare lo stato della comunicazione. In base alla decisione del classificatore, il trasmettitore adatta la modulazione e il code rate, garantendo un equilibrio tra efficienza spettrale e robustezza della trasmissione. Nel seguito saranno analizzati i risultati ottenuti dall'esperimento del sistema AMC attuato con l'ausilio di due SDR "Adalm-Pluto" che fungono da trasmettitore e ricevitore. Per i codici dei 2 dispositivi si rimanda alla repository GitHub del progetto. [8]

Si consideri il trasmettitore come *Dispositivo 1* e il ricevitore, con il classificatore SVM, come *Dispositivo 2*. Di seguito viene riportato l'algoritmo di trasmissione di Dispositivo 1:

---

**Algorithm 4.1** Algoritmo di trasmissione di Dispositivo 1

---

```

1: Setup iniziale dei parametri di trasmissione
2: while numAttese < numAtteseMax do
3:   Trasmissione di 500 frame
4:   ripetizioneRicezione = 0
5:   while ripetizioniRicezione < 3 do
6:     Ricevi 300 frame
7:     if rxFlag == 1 then
8:       Estrai dal messaggio ricevuto i nuovi parametri
9:       Cambia i parametri di trasmissione e attendi 8 s
10:      ripetizioneRicezione = 0
11:    else if ripetizioneRicezione == 3 then
12:      numAttese = numAttese + 1
13:      ripetizioneRicezione = ripetizioneRicezione + 1
14:    else
15:      ripetizioneRicezione = ripetizioneRicezione + 1
16:    end if
17:  end while
18: end while
19: Fine della trasmissione

```

---

Successivamente al setup dei parametri fondamentali di trasmissione (come numero di sottoportanti OFDM, frequenza portante, gain di ricezione e trasmissione), Dispositivo 1 genera 500 frame OFDM da inviare con la propria Adalm-Pluto. Una volta inviati



tutti i frame, Dispositivo 1 passa in ricezione e attende il sync signal di Dispositivo 2 e i successivi frame di feedback. Nel caso in cui si fosse raggiunto il massimo numero di ricezioni, si aumenta il numero di attese totali di Dispositivo 1. Se invece la ricezione è corretta, Dispositivo 1 estrae dai frame ricevuti il messaggio e lo associa ai nuovi parametri di trasmissione.

Si riporta ora l'algoritmo di comunicazione di Dispositivo 2:

---

**Algorithm 4.2** Algoritmo di ricezione di Dispositivo 2

---

```

1: Setup iniziale dei parametri di ricezione
2: while ripetizioniRicezione <= 4 do
3:   Ricezione di 500 frame
4:   if rxFlag == 1 then
5:     ripetizioneRicezione = 0
6:     Calcolo del BER e del SNR medi
7:     Classificazione dello stato con il modello SVM
8:     Generazione del payload di trasmissione e pausa
9:     Trasmissione di 300 frame di feedback
10:  else if headerCRCErrFlag == 1 ripetizioniRicezione! = 4 then
11:    Aggiungi il CFO stimato alla carrier
12:    ripetizioneRicezione = ripetizioneRicezione + 1
13:  else
14:    ripetizioneRicezione = ripetizioneRicezione + 1
15:  end if
16: end while
17: Fine della ricezione

```

---

Dopo aver gestito il setup dei parametri fondamentali di comunicazione, il ricevitore si mette in attesa di 500 frame per un numero massimo di 4 volte. Se non riceve nulla (ossia *rxFlag* nullo), ci potrebbe essere stato un problema di mancata sincronizzazione o di mancata demodulazione corretta del header a causa del CFO. Per ovviare tale problema, si aggiusta la portante sommando il CFO stimato dai frame ricevuti. Se invece la ricezione avviene con successo, allora il classificatore restituisce lo stato della comunicazione sulla base del BER e del SNR medi stimati. In base al prodotto ottenuto si scelgono i nuovi code rate e modulation order e si passa in trasmissione del messaggio di feedback. Se per 4 volte consecutive non viene ricevuto nulla, allora la ricezione termina.

4.3.5. Risultati sperimentali

I risultati sperimentali mostrano che l'implementazione dell'AMC basato su SVM migliora significativamente le prestazioni del sistema in scenari con variazioni rapide delle condizioni del canale. La comunicazione comincia sempre utilizzando una 64-QAM con un tasso di codifica di 3/4 per aumentare l'efficienza di trasmissione. Tuttavia, come si può osservare dai grafici, nel caso dell'esperimento la trasmissione appariva molto disturbata e le performance di trasmissione erano pessime (BER molto elevato).

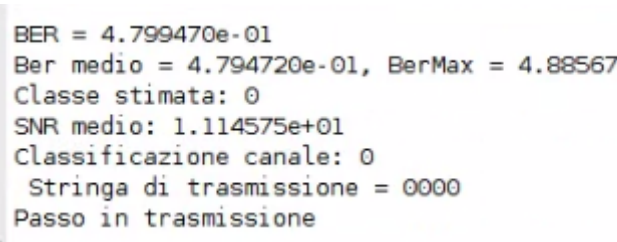


Figura 4.4: Risultati della prima trasmissione.

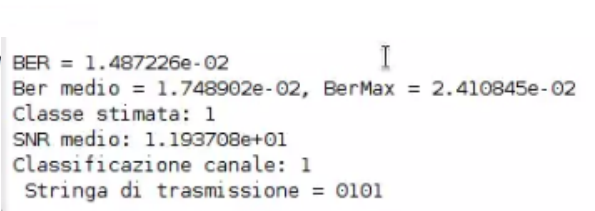


Figura 4.5: Risultati della seconda trasmissione.

Il classificatore ha restituito 0 (comunicazione pessima), quindi il messaggio di feedback inviato sarà 0000. Per l'invio del feedback si scelgono code rate e modulazioni robuste quali BPSK e 1/2. Il trasmettitore, una volta ricevuto il feedback, cambia i propri parametri di trasmissione. Nelle figure sottostanti è possibile osservare le modulazioni ricevute nei due casi.

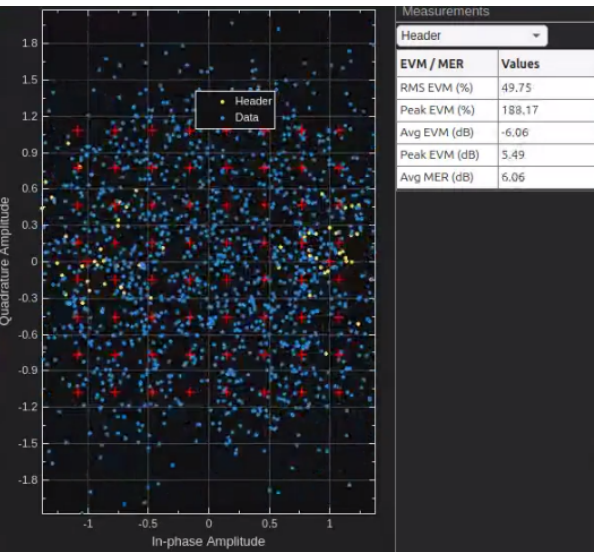


Figura 4.6: Costellazione 64QAM.

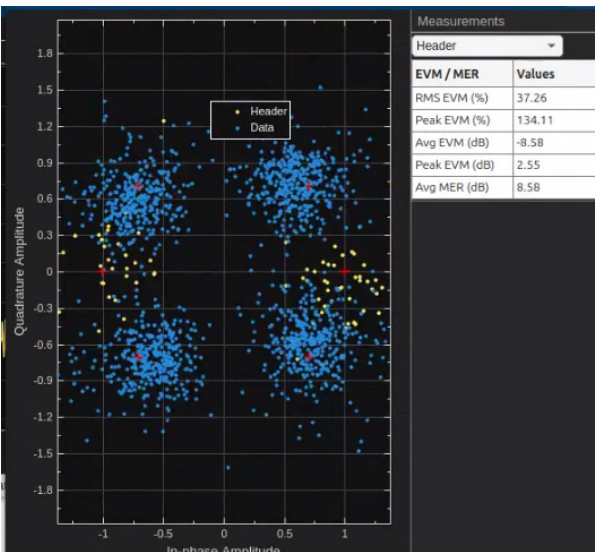


Figura 4.7: Costellazione QPSK.

Il risultato della ricezione nella seconda trasmissione è indicato nella figura 4.5. Il trasmettitore quindi sceglie la coppia 16-QAM  $2/3$  come nuovi parametri di trasmissione. In figura 4.8 è possibile osservare l'andamento dei parametri di trasmissione di Dispositivo 1 ai vari step di trasmissione.

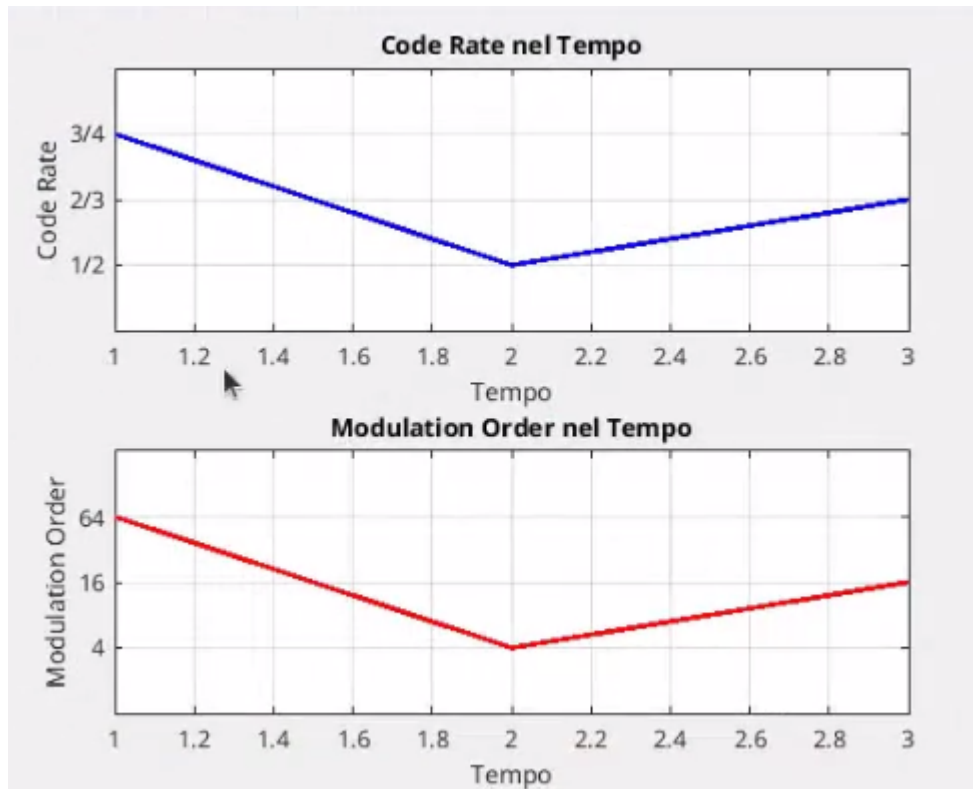


Figura 4.8: Code Rate e Modulation Order nel tempo

## 5 | Conclusioni

Il lavoro svolto in questa tesi ha dimostrato l'efficacia di un sistema di comunicazione basato su **OFDM** con **Adaptive Modulation and Coding (AMC)**, ottimizzato tramite tecniche di **Machine Learning** e validato in condizioni reali tramite test **Over-The-Air (OTA)** con **SDR Adalm-Pluto**. L'integrazione di un modello ML per la classificazione dello stato del canale ha permesso di adattare dinamicamente i parametri di trasmissione, migliorando l'affidabilità e l'efficienza spettrale del sistema.

Prima di implementare il sistema AMC, sono stati condotti test approfonditi sul sistema **OFDM base**, sia in simulazione che in ambiente reale tramite prove OTA, valutando le prestazioni in assenza e in presenza di codifica di canale. I risultati hanno evidenziato come la **codifica di canale** migliori significativamente la resilienza alle interferenze e alle degradazioni del canale, riducendo il **BER (Bit Error Rate)** rispetto alla configurazione senza codifica. Tuttavia, senza un meccanismo adattivo, il sistema non è in grado di rispondere dinamicamente alle variazioni delle condizioni di propagazione, portando a inefficienze o degrado delle prestazioni in scenari complessi.

I risultati sperimentali hanno confermato che l'uso di **tecniche adattive** consente di mitigare gli effetti delle variazioni del canale, come interferenze, fading ed effetto Doppler, rendendo il sistema più robusto rispetto a una configurazione statica. In particolare, la capacità del modello di apprendere le condizioni del canale e selezionare la modulazione e il code rate più appropriati rappresenta un passo avanti verso l'ottimizzazione autonoma delle reti wireless.

Questa ricerca apre la strada a future applicazioni in scenari reali, come le **reti 5G**, dove l'adattabilità del sistema alle condizioni variabili del canale è fondamentale. Ulteriori sviluppi potrebbero includere l'integrazione con tecniche avanzate di **Deep Learning** e l'estensione del sistema a scenari multi-utente e multi-antenna (**MIMO**), migliorando ulteriormente la capacità di adattamento alle dinamiche del canale.

## Bibliografia

- [1] I. A., N. P., and et al. Energy efficiency for 5g and beyond 5g: Potential, limitations, and future directions. *Sensors*, 2024.
- [2] M. F. Abaei, S. A. Vorobyov, and J. A. Nossek. Adaptive coding and modulation for satellite communication links in the presence of channel estimation errors. *ResearchGate*, 2013.
- [3] M. Araújo and M. Melo. Ofdm: Theory and applications, 2010. URL [https://www.researchgate.net/publication/290774058\\_OFDM\\_Theory\\_and\\_applications](https://www.researchgate.net/publication/290774058_OFDM_Theory_and_applications).
- [4] D. Benvenuti. Cos'è l'algoritmo di viterbi, 2005. URL <https://www.vialattea.net/content/2261/>.
- [5] Dao, Park, Kim, and Cho. Adaptive mcs selection and resource planning for energy-efficient communication in lte-m based iot sensing platform. *PLOS one*, 2017.
- [6] A. Falaschi. *Trasmissione dei Segnali e Sistemi di Telecomunicazione*. Auto-published, 2023.
- [7] R. V. Federico Campeotto. Cos'è l'algoritmo di viterbi, 2010. URL <https://users.dimi.uniud.it/~agostino.dovier/DID/TI/CampeDominiLAB.pdf>.
- [8] N. Gallucci and M. Malagrino. Simulazione e test ota di un sistema ofdm con e senza adaptive modulation and coding ml-guidato, 2025. URL <https://github.com/nico-net/ProgettoSistemiDiComunicazione/>.
- [9] K. M. Z. Godini. Ofdm con prefisso ciclico parziale, 2011. URL [https://thesis.unipd.it/retrieve/4e4347ee-d181-4d06-9264-c6ac9741a9da/tesi\\_x\\_seg..pdf](https://thesis.unipd.it/retrieve/4e4347ee-d181-4d06-9264-c6ac9741a9da/tesi_x_seg..pdf).
- [10] J. H. Lee, Y. H. Lee, and B. J. Kwak. Adaptive modulation and coding for long term evolution-based mobile satellite communication system. *ResearchGate*, 2012.

- [11] MathWorks. Ofdm transmitter using software-defined radio. URL <https://it.mathworks.com/help/comm/plutoradio/ug/ofdm-transmitter-using-pluto-sdr.html>.
- [12] MathWorks. Ofdm transmitter and receiver, 2023. URL <https://it.mathworks.com/help/comm/ug/ofdm-transmitter-and-receiver.html>.
- [13] MathWorks. Ofdm receiver using software-defined radio, 2024. URL <https://it.mathworks.com/help/comm/plutoradio/ug/ofdm-receiver-using-pluto-sdr.html>.
- [14] D. E. Rodriguez. Exploring the principles and techniques of adaptive modulation and coding (amc), 2023. URL <https://www.matlabassignmentexperts.com/blog/adaptive-modulation-and-coding-principles-techniques.html>.
- [15] T. Trainer. Amc (adaptive modulation and coding), 2023. URL <https://www.telecomtrainer.com/amc-adaptivemodulation-and-coding/>.
- [16] L. Vandendorpe. A greedy algorithm for goodput-based adaptive modulation and coding in bic-ofdm systems. *European Wireless 2010*, 2010.
- [17] Wikipedia. Codice convoluzionale. URL [https://it.wikipedia.org/wiki/Codice\\_convoluzionale](https://it.wikipedia.org/wiki/Codice_convoluzionale).
- [18] Wikipedia. Support vector machines, 2025. URL [https://en.wikipedia.org/wiki/Support\\_vector\\_machine](https://en.wikipedia.org/wiki/Support_vector_machine).
- [19] A. Wrld. Support vector machines (svm): A comprehensive guide to one of machine learning's most powerful algorithms, 2024. URL <http://bit.ly/3EI701I>.