

Codice per la simulazione di trasmissioni digitali: Documentazione e Test

Mario Pellegrino Ambrosone

December 2024

Abstract

In merito ai sistemi di trasmissione digitale, è chiaro quanto e come le tecniche di modulazione adottate abbiano ripercussioni sulle prestazioni generali del sistema, qualora le prestazioni siano valutate con riferimento a probabilità di errore, sfruttamento del SNR ed efficienza spettrale.

A tal proposito, col fine di analizzare il comportamento dei sistemi di trasmissione digitale con utenti mobile e in reti wireless, si è redatto il codice MATLAB. oggetto della seguente trattazione. Nello specifico, testeremo il codice utilizzando come espediente la scelta della migliore tecnica di modulazione tra quelle fondamentali (PAM, PPM, PSK, QAM) mediante un confronto tra risultati teorici e simulati.

La documentazione, sviluppata in tre sezioni, ha l'obiettivo di garantire l'affidabilità del codice, non solo fornendo i vettori di test e i risultati di simulazione ad essi associati, ma anche un'analisi di detti risultati con l'intento di avvalorare l'espediente di progettazione.

Contents

1	Test Cases	3
1.1	Parametri Invarianti	3
1.2	Formalizzazione dei Test Vectors	4
2	Simulazione	5
2.1	Previsioni	5
2.2	M-PAM	5
2.3	M-PPM	7
2.4	M-PSK	8
2.5	M-QAM	10
3	Conclusioni	12

1 Test Cases

Sia di esempio il caso della Rel. 5 del 3GPP, release caratterizzata dall'introduzione della tecnologia HSDPA (High Speed Downlink Packet Access).

Commercializzata nel 2002, la Rel.5 vide sostituita la modulazione QPSK (Quadratic Phase Shifting Key) in downlink con le modulazioni 16 e 64-QAM, migliorando la capacità di trasmissione e l'efficienza spettrale delle reti 3G.

Le tecniche di modulazione più diffuse in quegli anni, per reti wireless, senza considerare le varianti di QAM, includono:

- QPSK, con 2 bit per simbolo, utilizzata già nella Rel.99;
- 8-PSK, con 3 bit per simbolo, utilizzata in EDGE (reti 2G);
- 4/8-PPM, con 2/3 bit per simbolo, utilizzate in IrDA per comunicazioni a corto raggio;
- 2/4-PAM, con 2/3 bit per simbolo, utilizzate in HomeRF (wireless in ambito industriale);

Già da questa piccola panoramica si può vedere come, per Mobile Networks, si è preferito utilizzare, perlomeno fino agli inizi del 2000, tecniche di modulazione di tipo Phase Shifting Key o QAM piuttosto che modulazioni in Ampiezza, di tipo PAM, o Position, di tipo PPM.

È chiaro che, la necessità di rendere più prestanti le reti porta ad un aumento di simboli per singola trasmissione ma, come vedremo grazie ai risultati in simulazione, tale scelta ha risvolti più negativi che positivi, influenzando la probabilità di errore media per simbolo.

Utilizzeremo pertanto il codice MATLAB redatto, oggetto del documento, al fine di valutare le prestazioni delle suddette tecniche di modulazione, a meno del QPSK, mediante la definizione di alcuni test cases.

I test cases avranno come obiettivo non solo quello di dare robustezza a quelli che sono i nostri prospetti, di natura teorica, per la comprensione delle scelte progettuali del 3GPP, ma anche e soprattutto quello di valutare, mediante la strumentalizzazione e il confronto con detti risultati, come il codice risponda in fase di simulazione.

1.1 Parametri Invarianti

Si tenga in considerazione nello sviluppo dei test cases (ogni test case per semplicità di trattazione prevede un singolo test vector), che i test vectors in input al codice definiscono, in ordine di elencazione:

- M , numero di simboli trasmissibili;
- SNR_{dB} , intervallo di rapporto Segnale Rumore espresso in decibel;
- *Montecarlo*, numero di realizzazioni di simulazione;
- *Model*, nome del tipo di modulazione utilizzata per il test corrente;
- L , numero di rami per implementare tecnica di diversità nel tempo.

Siano dunque fissati i parametri, e siano essi detti parametri invarianti, M , SNR_{dB} , Montecarlo ed L , tali che:

- $M = 16$;
fissare un alto numero di simboli trasmissibili in maniera equiprobabile, in virtù del data load caratterizzante le reti mobile a tecnologia 3G, permette di paragonare le prestazioni delle diverse tecniche di modulazione confrontandole in maniera più efficace.
- $SNR_{dB} = \{0, 20\} dB$;
avere un SNR che varia tra 1 e 100, in numeri reali, permette di coprire con una certa affidabilità nei risultati, sia casi pessimi con segnali appena percettibili a basso rapporto Segnale-Rumore ($SNR = \{0, 10\} dB$), sia casi a SNR più moderato.

- $Montecarlo = 10^4$;
avere un alto numero di simulazioni di trasmissione garantisce risultati apprezzabilmente validi per Pe (Probabilità di errore per simbolo) almeno pari a $Pe \approx 10^{-6}$.
- $L = 3$;
fissare il valore L , tenendo conto del fatto che mediamente nella Rel.5 di 3GPP i valori di L oscillavano tra $L=[2, \dots, 4]$, permette di standardizzare la velocità di fading e la larghezza di banda del canale utilizzato, avvalorando quindi l'idea di simulazione in ambienti suburbani con qualità del canale media e utenti a mobilità contenuta.

Si tenga infine a mente che i valori associati ai parametri invarianti mirano a definire le condizioni, in average case, di un ambiente urbano/sub-urbano in cui gli utenti mobile sfruttano un canale wireless a rumore gaussiano, affetto da fading e, per semplificare le simulazioni, senza interferenze. Per essere più precisi, quando parleremo di fading faremo sempre riferimento ad un Frequency-Flat Slow Fading i cui coefficienti vanno come variabili aleatorie di Rayleigh; inoltre, per la gestione del fading mediante tecnica di diversità nel tempo, si è considerato ed implementato solo Selection Combining come regola di combining lato ricevente.

1.2 Formalizzazione dei Test Vectors

A questo punto procediamo con la definizione dei 4 vettori di test, ognuno rappresentativo di ogni test case. Sia detta forma base la struttura di un qualsiasi input al codice, e sia essa definita come segue:

$$simulation(M, SNRdB, Montecarlo, Model, L);$$

Allora i nostri test vectors saranno del tipo:

1. 16-PAM
`simulation(16, 0:20, 1e4, 'PAM', 3);`
2. 16-PPM
`simulation(16, 0:20, 1e4, 'PPM', 3);`
3. 16-PSK
`simulation(16, 0:20, 1e4, 'PSK', 3);`
4. 16-QAM
`simulation(16, 0:20, 1e4, 'QAM', 3);`

2 Simulazione

Il codice redatto, qualunque sia l'input, a patto che esso rispetti la forma base (vedi Sez. 1.2), fornisce come output:

1. La matrice A dei simboli trasmissibili;
2. Due Grafici che confrontano l'andamento delle probabilità di errore medie del sistema, simulate per condizioni differenti, con l'andamento della probabilità di errore teorica ($Pe_{Teorica}$); nello specifico i grafici avranno, oltre $Pe_{Teorica}$:
 - (a) Pe media simulata su canale senza fading ($Pe_{Simulata} - nonFading$) e Pe media simulata su canale con fading ($Pe_{Simulata} - Fading$);
 - (b) Pe media simulata su canale con fading ($Pe_{Simulata} - Fading$) e Pe media simulata, su canale con fading, con implementazione di tecnica di diversità nel tempo in invio e selection combining in ricezione ($Pe_{Simulata} - Diversity$).

Si tenga in oltre a mente che ogni grafico di simulazione è dotato di una legenda, ove, per semplicità e coerenza tra output, l'andamento della $Pe_{Simulata} - Fading$ è sempre descritta da una linea continua di colore blu, $Pe_{Teorica}$ è sempre descritta da una linea continua di colore nero mentre gli altri parametri ($Pe_{Simulata} - nonFading$ e $Pe_{Simulata} - Diversity$) sono descritti da linee rosse tratteggiate e segnate per ogni valore di SNR_{dB} rilevante.

2.1 Previsioni

Ciò che ci aspettiamo, anche in funzione del nostro “espediente” di trattazione, ovvero la scelta del 3GPP di optare per 16-QAM, è che sia proprio il 16-QAM la tecnica di modulazione che mostrerà le probabilità di errore più ridotte.

Sicuramente non è da escludere la possibilità che qualora tale previsione risulti errata, e dunque che altre tecniche mostrino andamenti delle probabilità di errore medie migliori, la scelta potrebbe comunque essere giustificata da un miglior utilizzo della banda e quindi una maggiore efficienza spettrale.

2.2 M-PAM

Sia il test vector : `simulation(16, 0:20, 1e4, 'PAM', 3);`

Allora il codice avrà come output i grafici in Fig. 2.2.1 e Fig. 2.2.2.

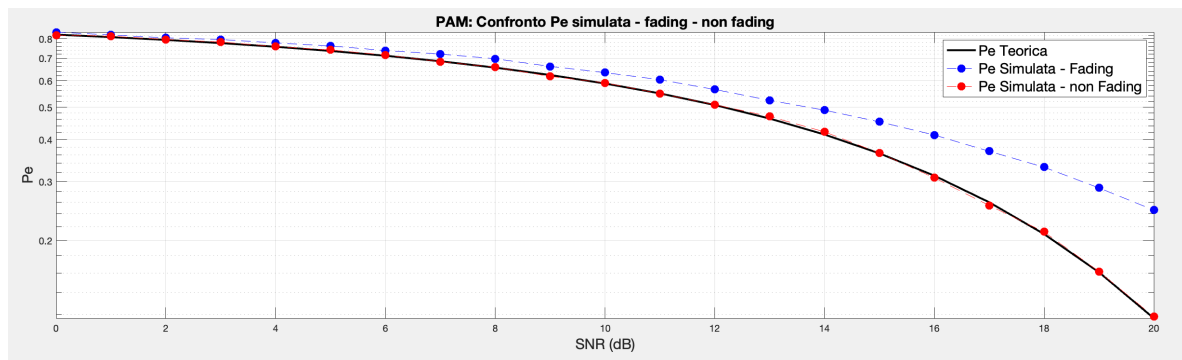


Fig: 2.2.1

Le modulazioni M-PAM sono, in condizioni ideali, abbastanza robuste. Qualitativamente, la robustezza di cui si parla è strettamente legata all'SNR, se per condizioni ideali si fa riferimento ad un canale con rumore AWGN ma senza interferenze ed in assenza di fading.

Ciò che ci aspettiamo in simulazione è quindi una certa coerenza tra gli output simulati ($Pe_{Simulata} -$

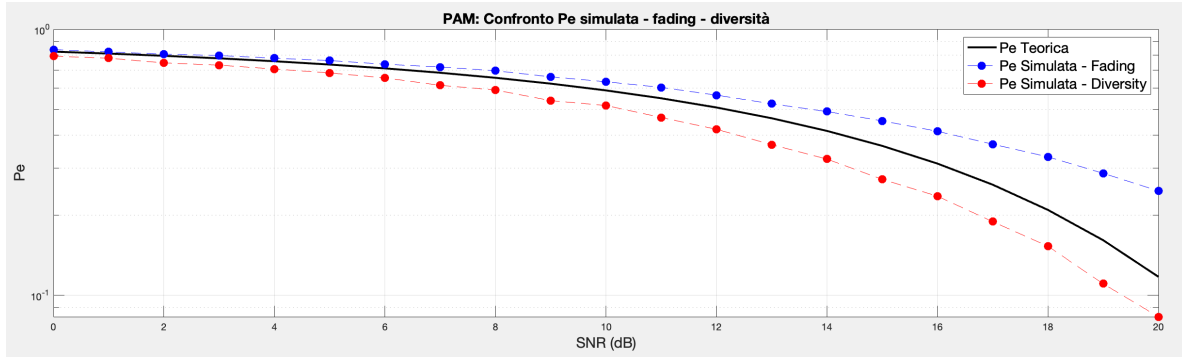


Fig: 2.2.2

nonFading) e la curva teorica ($Pe_{Teorica}$). Osservando la Fig 2.2.1 possiamo notare come infatti la curva simulata sia molto vicina alla curva teorica, con una probabilità di errore che diminuisce progressivamente all'aumentare del SNR; questo ci mostra, in buona sostanza, come in assenza di fading il rumore si distribuisca in maniera uniforme, senza eccessive variazioni temporali o spaziali, o perlomeno senza effetti imprevedibili mediante modelli teorici.

Data la natura del fading, ci aspettiamo che, in presenza di Frequency-Flat Slow Fading, quindi in trasmissioni che subiscano gli effetti combinati del multipath e del Doppler, ci siano importanti variazioni, sia in modulo che in fase, del segnale ricevuto rispetto a quello trasmesso.

Uno dei principali effetti dei canali con fading è infatti l'attenuazione dei segnali, ovvero una riduzione della potenza dei segnali trasmessi, data dal prodotto tra il segnale trasmesso ad un tempo t ed un coefficiente di Rayleigh, caratteristico dello stato del canale al momento della trasmissione t (la risposta del canale non sarà tempo variante negli intervalli di simbolo, data l'assunzione di Slow Fading).

Questo ci fa pensare che, soprattutto senza implementazioni atte al contrasto del fading, le Pe vertiginosamente. Le nostre aspettative sono confermate dal plot, il quale mostra come la curva simulata si discosti dalla curva teorica, mostrando una probabilità di errore più elevata per ogni valore di SNR.

La Fig. 2.2.2 è il secondo grafico ottenuto in output; possiamo vedere come le tecniche di diversità giochino a nostro vantaggio nel miglioramento delle prestazioni, anche per una tecnica così poco affidabile come il PAM, soprattutto con un numero così alto di simboli.

Ciò che possiamo constatare è che, seppur la curva rossa mostri una riduzione della probabilità di errore rispetto alla curva di $Pe_{Simulata} - Fading$, tali riduzioni non sono poi così tanto significative. I miglioramenti sono così relativi che porre $L = 3$ può essere quasi considerato uno spreco di risorse temporali.

È bene notare, che potremmo ottenere un risultato piuttosto simile, e comunque prossimo alle prestazioni teoriche, quindi con andamenti esponenziali delle probabilità, ponendo anche $L = 2$ (Fig. 2.2.3), guadagnandone in termini di energia spesa e latenza.

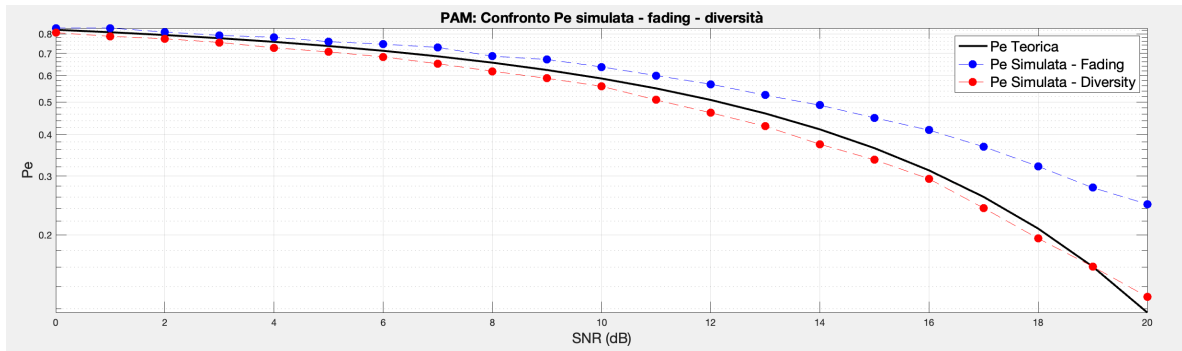


Fig: 2.2.3

2.3 M-PPM

Sia il test vector : `simulation(16, 0:20, 1e4, 'PPM', 3);`

Allora il codice avrà come output i grafici in Fig. 2.3.1 e Fig. 2.3.2.

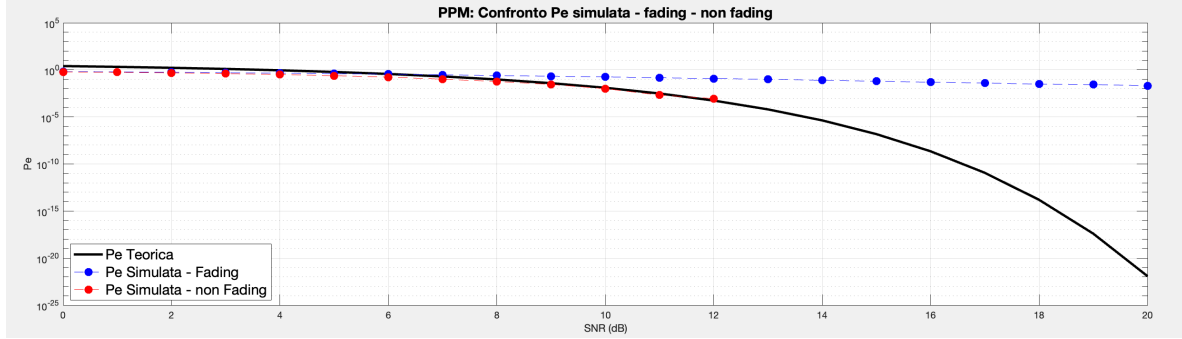


Fig: 2.3.1

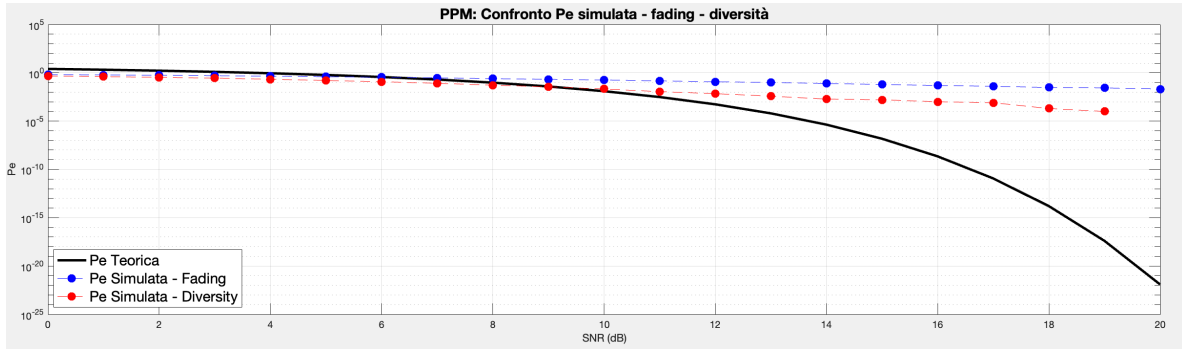


Fig: 2.3.2

Osservando il primo dei due output (Fig 2.3.1) notiamo che la probabilità di errore *Pe Simulata - non Fading*, all'aumentare del SNR, decresce molto più rapidamente rispetto a quanto osservato con il PAM (Fig. 2.2.1); a parità di rapporto segnale rumore, ad esempio $SNR_{dB} = 10$, quindi con un segnale ≈ 100 volte più forte del rumore, abbiamo per 16-PPM un valore di *PeSimulata - nonFading* $\approx 10^{-3}$, contro un valore, con 16-PAM, di *PeSimulata - nonFading* ≈ 0.6 .

Questa riduzione così apprezzabile della probabilità di errore non ci stupisce affatto; con una tecnica M-PPM infatti è molto più “difficile” avere errori in ricezione, dato che i simboli vengono modulati garantendo sempre ortogonalità. Come se ciò non bastasse, all'aumentare del numero di simboli, perlomeno secondo le curve teoriche, la probabilità di errore a parità di SNR decade molto più rapidamente.

Tali risultati teorici, per quanto interessanti, pongono dei limiti importanti alla progettazione dei test vectors, poiché per poter osservare probabilità di errore sempre più ridotte si necessita di un quantitativo di simulazioni sempre maggiore. Questo dunque spiega come mai la curva rossa in Fig. 2.3.1 assume valori fino a *PeSimulata - nonFading* $\approx 10^{-6}$ per un numero di simulazioni *MonteCarlo* = 10^4 per singolo valore di SNR.

Aumentare MonteCarlo di qualche ordine di grandezza permetterebbe sicuramente di ottenere risultati di *Pe* anche per valori maggiori di SNR, ma per semplicità di trattazione si eviteranno test con valori differenti di MonteCarlo.

Chiaramente in assenza di fading l'unico fattore che influenza i risultati simulati, rispetto agli andamenti teorici, è il rumore; di conseguenza all'aumentare del SNR il segnale diventa progressivamente più forte rispetto al rumore, rendendo la distinzione tra simboli più accurata, restituendo un andamento della curva praticamente identico a quello teorico.

In presenza di fading, la curva resta sempre vicina a $\approx 10^0$, ovvero ad una probabilità d'errore di quasi pari ad 1, anche per valori crescenti di SNR, arrivando al più a $\approx 10^{-1}$, indice del fatto che non solo l'M-PPM richiederebbe livelli estremamente elevati di SNR per ottenere una probabilità di errore

accettabile, ma anche che l'andamento delle Pe non è più esponenziale ma iperbolico.

Proviamo a vedere, analizzando il secondo output, Fig 2.3.2, come l'implementazione della tecnica di diversità nel tempo possa giovare al sistema.

È chiaramente visibile come, nonostante ci siano sostanzialmente 3 rami per ogni MonteCarlo-esima trasmissione, la probabilità di errore, seppur si riduca di qualche ordine di grandezza, con un minimo di $\approx 10^{-4}$, continua ad avere un andamento iperbolico, molto più simile a quello di $Pe_{Simulata} - Fading$ che a quello di $Pe - Teorica$.

Sicuramente il risultato non è pessimo come per il PAM, anche se il “trend” è dovuto più alla natura della tecnica di modulazione, intrinsecamente più stabile, che all'implementazione della tecnica di diversità.

Per ottenere risultati più soddisfacenti potremmo ad esempio aumentare il valore di L , e ri-simulare con $L = 10$.

In queste condizioni, come si vede dalla Fig. 2.3.3, la probabilità di errore si abbassa notevolmente raggiungendo valori accettabili, ma soprattutto un andamento paragonabile a quello teorico.

Il trade-off è che, in generale con tecniche di ritrasmissioni su L rami, e nello specifico con così tanti rami, ogni scambio di informazione, dove per informazione si intenda il simbolo modulato che si vuole comunicare, con la funzione d'onda ad esso associata, richiede il decuplo dell'energia, del tempo e ci porta a sfruttare $\frac{1}{10}$ della banda a disposizione.

Concludiamo in merito al PPM M-ario che, seppur esso rappresenti un metodo di modulazione più stabile e “sicuro” di un PAM, oltre ad essere estremamente poco efficiente da un punto di vista spettrale, dato che ortonormalizzare un numero crescente di segnali richiede una banda sempre maggiore, ha anche prestazioni piuttosto mediocri, con tecnica di diversità nel tempo.

Possiamo dire dunque che, data la “dispendiosità” di implementazione con così tanti simboli, la poca efficienza spettrale e i risultati non soddisfacenti di $Pe_{Simulata} - Diversity$, ottenuti per $L = 3$, scegliere una modulazione di tipo PPM per un sistema di telecomunicazione mobile sarebbe piuttosto sconsigliato.

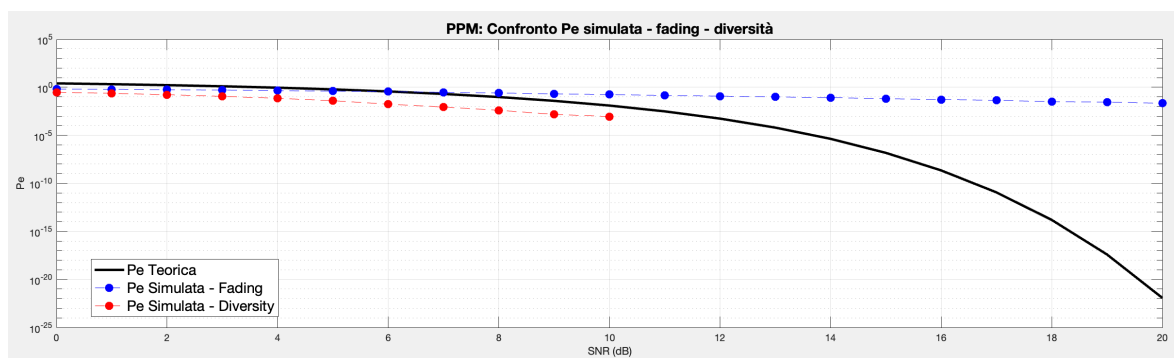


Fig: 2.3.3

2.4 M-PSK

Sia il test vector : `simulation(16, 0:20, 1e4, 'PSK', 3);`

Allora il codice avrà come output i grafici in Fig. 2.4.1 e Fig. 2.4.2.

La Fig 2.4.1 ci mostra il primo dei due output. In assenza di fading, la curva simulata, segue con accuratezza l'andamento teorico, rappresentando così una riduzione esponenziale della probabilità di errore, in accordo alla QFUNC delle Pe media per un generico Union Bound, all'aumentare del SNR. Essendo le modulazioni di un PSK basate sullo shift in fase dei simboli in maniera uniforme tra $(0, 2\pi)$, ci si aspetta che in condizioni ideali, all'aumento dell'energia del segnale rispetto al rumore, il ricevitore possa distinguere le fasi con maggiore precisione, riducendo la probabilità di interpretare erroneamente un simbolo.

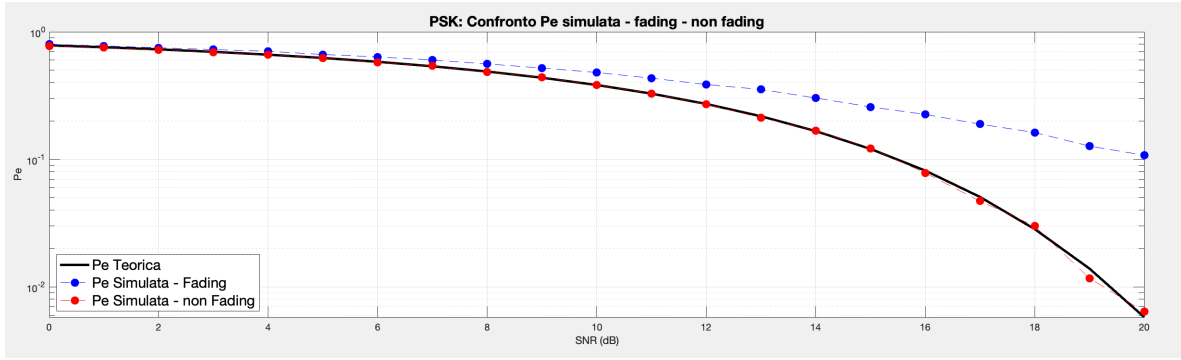


Fig: 2.4.1

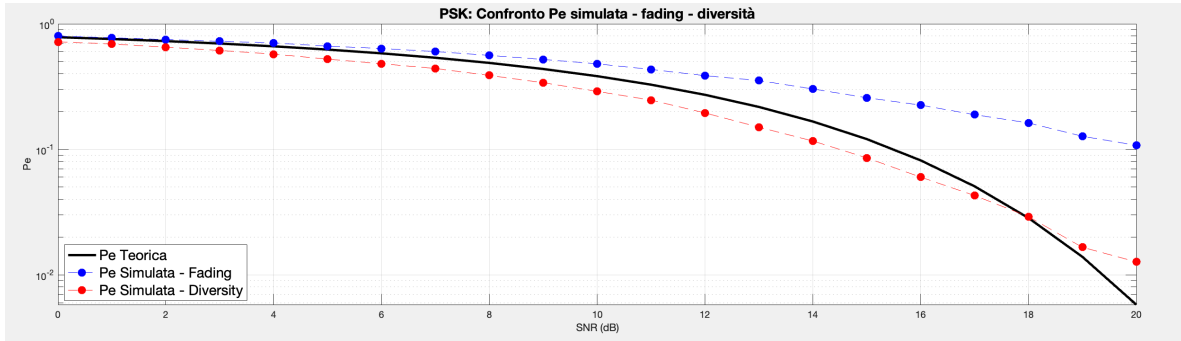


Fig: 2.4.2

In presenza di fading la curva simulata, sicuramente mostra un miglioramento della probabilità di errore con l'aumento dell'SNR, ma comunque, come abbiamo imparato a riconoscere anche dai precedenti risultati, ha un andamento iperbolico, con un conseguente peggioramento delle prestazioni. Questo risultato, oltre ad essere totalmente prevedibile in funzione degli effetti del fading visti sulle altre modulazioni, è anche piuttosto facile da decodificare. Ciò che è davvero interessante da valutare è che questa volta, a differenza dalle precedenti simulazioni, le prestazioni non subiscono un crollo così vertiginoso; senza alcuna contromisura al fading, la probabilità di errore per Signal-to-Noise ratio mediocri, quindi $SNR_{dB} = [10, \dots, 20]$, ha una riduzione, a parità di SNR, di al più un ordine di grandezza rispetto alla $Pe_{Simulata - nonFading}$.

Il nostro codice infatti utilizza come decisore, in ricezione, la minima distanza, decisore ottimo per un sistema con modulazioni eterogenee ma basilari, valutando quindi il modulo dei segnali ricevuti piuttosto che la fase, ragion per cui la maggior parte degli “effetti negativi” del fading in fase si perdono e gli errori non aumentano poi così tanto.

Con l'ausilio della tecnica di diversità nel tempo, analizzando il secondo output in Fig. 2.4.2, $Pe_{Simulata - Diversity}$ diminuisce nettamente rispetto a $Pe_{Simulata - Fading}$, ritornando quasi agli andamenti su canale AWGN; si noti che, nonostante quanto appena detto, soprattutto ad alti SNR ad esempio $SNR_{dB} = 20$, l'andamento della Pe con diversità seppur esponenziale decresca in maniera sensibilmente più lenta di quello teorico. Questa particolare efficacia, che mostra addirittura per alcuni SNR prestazioni migliori di quelle nella curva $Pe_{Simulata - nonFading}$, è garantita dai risultati sopra discussi; la ritrasmissione su L rami permette infatti di “evitare” condizioni di deep fade e riequilibrare i fattori di attenuazione del Fading annullandone gli effetti sul modulo, ma non sulla fase, dei segnali.

2.5 M-QAM

Sia il test vector : `simulation(16, 0:20, 1e4, 'QAM', 3);`

Allora il codice avrà come output i grafici in Fig. 2.5.1 e Fig. 2.5.2.

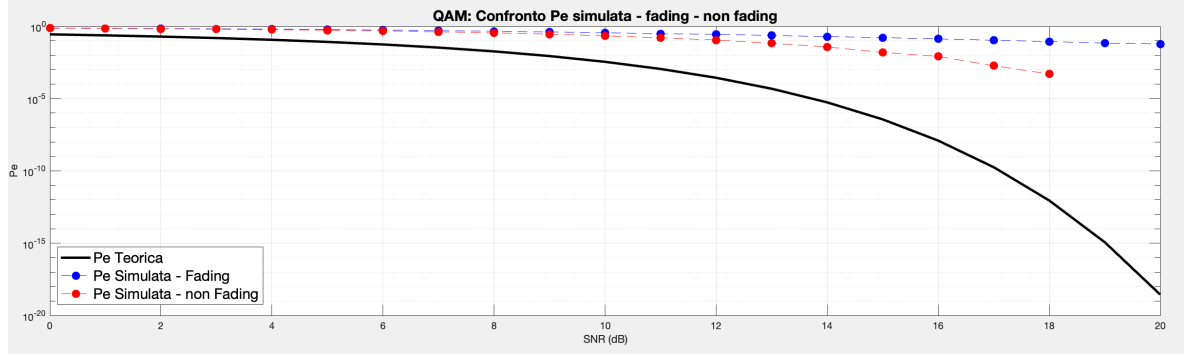


Fig: 2.5.1

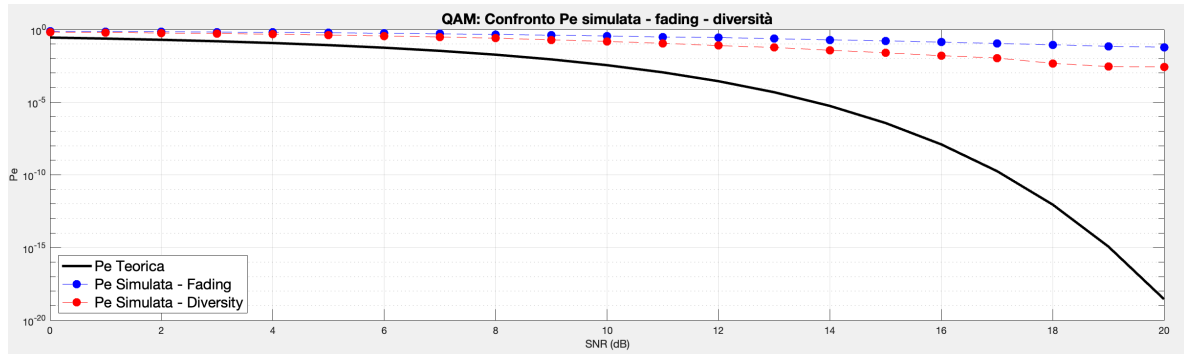


Fig: 2.5.2

Analizzando il primo degli output, rappresentato in Fig. 2.5.1, la prima cosa che potremmo pensare è che un QAM con così tanti simboli sia assolutamente pessimo.

L'idea potrebbe essere giustificata dal fatto che, perlomeno osservando la curva *Pe Simulata - non Fading*, l'andamento delle probabilità simulate sembra essere un esponenziale molto lento, sinonimo di pessime prestazioni. In realtà i singoli valori di *Pe Simulata - non Fading* nell'intervallo $SNR_{dB} = [10, \dots, 20]$, mostrano una probabilità di errore che va da $\approx 10^{-1}$ fino ad un minimo di $\approx 10^{-5}$, che in effetti sono i valori più bassi incontrati fino ad ora, a meno chiaramente del PPM.

Il fatto che tra le modulazioni analizzate il PPM e il QAM siano quelle con *Pe* più basse non è casuale, difatti, seppur in maniera e in “senso” diverso, entrambe le tecniche modulano i simboli con un principio di ortogonalità; se il PPM garantisce l'ortogonalità da un punto di vista temporale rendendo gli impulsi che rappresentano i simboli mutuamente esclusivi separandoli nel tempo, il QAM lo fa mediante due portanti, una in fase e una in quadratura (da qui il nome Quadrature Amplitude Modulation).

Questo significa sostanzialmente che tutti gli svantaggi in termini di prestazioni in spettro, tipici del PPM, nel QAM non ci sono.

Ciò che ci aspettiamo di vedere con il fading poi è il solito andamento iperbolico, non diverso da quanto ottenuto con ogni altra tecnica di modulazione, ma con risultati più simili a quelli del PPM, dove per alti SNR si sono osservati *Pe* minime di $\approx 10^{-2}$. In realtà il minimo raggiunto dalla curva blu, quindi il valore di *Pe Simulata - Fading* per $SNR_{dB} = 20$ è di $\approx 10^{-1}$, sintomo della robustezza del PPM e di come il QAM soffra le variazioni sulle portanti impresse dal Fading.

La Fig. 2.5.2 mostra il secondo output al QAM, ultimo dei test case; seppur da un lato la nostra aspettativa sia quella di osservare un netto miglioramento delle prestazioni, soprattutto in virtù di quanto constatato nelle precedenti analisi, ci aspettiamo comunque un andamento simile a quanto ottenuto con il PPM. In effetti osservando le curve rosse di Fig. 2.5.2 e Fig. 2.3.2 possiamo notare una certa “somiglianza” dei risultati. Così come per il PPM, quando ci sono così tanti simboli, anche con tecnica di diversità nel tempo con $L = 3$ rami, abbiamo al più una leggera riduzione della probabilità di errore, specialmente ad SNR più alti.

La ragione per cui non bastano così pochi rami, oltre al fatto che potrebbe essere legata alla regola di combining scelta dal lato ricevitore, che ricordiamo essere Selection Combining nel nostro caso, è da ricercare negli effetti che il fading ha ad ogni trasmissione; i coefficienti di attenuazione infatti, in tecniche come PAM, esercitano la maggior parte dei loro effetti negativi sulla potenza del segnale ricevuto ma non creano così tanti problemi sulla fase. Nel QAM invece, dove avere shift in fase vuol dire far variare le portanti su cui i fasori trasmessi modulano i simboli, il fading ha effetti sia in modulo che in fase e un volta ricevuto il segnale, andando al decisore a distanza minima, la scelta diventa quasi sempre impossibile da effettuare correttamente.

Avere dei rami chiaramente aiuta, però la probabilità di errore resta troppo superiore alla curva teorica e troppo lentamente variabile per avere un andamento esponenziale, evidenziando quanto la diversità mitighi ma non elimini gli effetti del fading.

Per ottenere un risultato quantomeno vicino alla probabilità di errore teorica bisognerebbe aumentare enormemente il numero di ritrasmissioni, ponendo almeno $L = 70$ (il risultato di una simulazione con questo parametro è mostrato in Fig. 2.5.3).

Purtroppo a sostegno di tale affermazione non riusciamo a ottenere risultati eleganti che possano descrivere come mai proprio per $L = 70$ si ha la maggiore somiglianza con la curva teorica, ma possiamo sicuramente dire che all’aumentare del numero di rami le curve avranno andamenti esponenziali decrescenti sempre più veloci, fino ad un netto miglioramento per $L \approx 150$.

Ciò che deduciamo è che un quantitativo così grande di rami permetterebbe di ottenere risultati ideali e prestazioni ottime in termini di probabilità di errore, ma ovviamente non sarebbe applicabile poiché l’energia spesa tenderebbe ad infinito e la banda a disposizione sarebbe sfruttata sempre di meno.

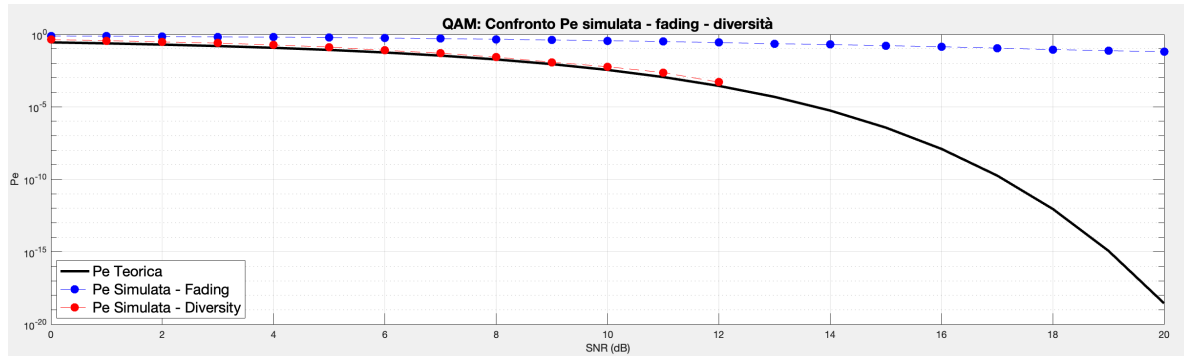


Fig: 2.5.3

3 Conclusioni

Ora che abbiamo valutato, a parità di numero di simboli, le prestazioni di ognuna delle tecniche di modulazione in analisi, possiamo dare la nostra interpretazione dei risultati ottenuti.

In ognuno dei test case abbiamo osservato come il fading causi un sensibile deterioramento delle prestazioni, rispetto alle probabilità di errore medie, su tutto l'intervallo SNRdB.

È interessante però vedere, nonostante l'andamento della curva *Pe Simulata - Fading* sia sempre lo stesso e i singoli valori delle probabilità di errore siano circa uguali, come la stabilità di alcune tecniche generi delle variazioni di prestazioni uniche.

Per PAM e PSK le variazioni ΔPAM , ΔPSK tra *Pe Simulata - Fading* e *Pe Simulata - nonFading* sono infatti, per $SNRdB = 20$:

$$\Delta PAM = \frac{((PeSimulata-Fading)PAM - (PeSimulata-nonFading)PAM)}{(PeSimulata-nonFading)PAM} \approx 1,50 = 150\%,$$

$$\Delta PSK = \frac{((PeSimulata-Fading)PSK - (PeSimulata-nonFading)PSK)}{(PeSimulata-nonFading)PSK} \approx 9,00 = 900\%,$$

mentre per QAM, sempre ad $SNRdB = 20$:

$$\Delta QAM = \frac{((PeSimulata-Fading)QAM - (PeSimulata-nonFading)QAM)}{(PeSimulata-nonFading)QAM} \approx 9999 = 999900\%.$$

Questi valori percentuali ci danno un indice quantitativo del peggioramento delle prestazioni causato dal fading, almeno localmente al fissato valore $SNRdB = 20$. Ne ricaviamo che, nonostante la modulazione “meno affetta” sembri essere il PAM c'è comunque da considerare il fatto che i valori minimi di *Pe Simulata - nonFading* sono fin troppo alti, con *Pe Simulata - nonFading* $\approx 10^{-1}$, e quindi il vantaggio risulta piuttosto relativo.

Purtroppo per PPM, non avendo un adeguato numero di simulazioni per ottenere valori di *Pe Simulata - nonFading* così ridotti come quelli che mostrerebbe ad SNR così elevati, non riusciamo ad avere una stima accurata delle variazioni. Osservando l'andamento della curva in Fig. 2.3.1, quello che ci aspettiamo è una differenza tra i due valori di probabilità almeno uguale ai tre ordini di grandezza mostrati dal QAM in Fig. 2.5.1 poiché già ad $SNRdB = 10$ c'è una variazione di circa un ordine di grandezza.

Sicuramente, a prescindere dalla modulazione, i danni del fading sono enormi e assolutamente non trascurabili in nessun sistema che ne sia affetto. Purtroppo la nostra prima previsione, ovvero quella secondo cui il 16-QAM avrebbe mostrato le probabilità di errore minori su tutto l'intervallo, è errata. Non solo l'andamento del QAM sembra essere, a meno del fading, quello di un esponenziale estremamente lento, ma le sue probabilità di errore non sono minimamente vicine a quelle del PPM, soprattutto per SNR medio-bassi come quelli da noi testati. In effetti, con un numero di simboli così alto, il PPM è avvantaggiato per via del metodo attraverso cui implementa l'ortogonalità tra i segnali modulati. Come già spiegato però, tale tecnica genera un deterioramento delle prestazioni in termini di efficienza spettrale così importante da rendere il PPM inutilizzabile per trasmissioni in ambienti urbani e con così tanti utenti quanti quelli di destinazione per le reti mobile.

La seconda delle previsioni era dunque corretta, il QAM infatti non solo mostra le migliori *Pe*, subito dopo il PPM, ma soprattutto non ha gli stessi “problemi” di efficienza spettrale.

Gli ultimi parametri interessanti da analizzare sono le differenze relative tra le *Pe Simulata - Diversity* e le *Pe Simulata - Fading*. Questi parametri, sulla falsariga dei precedenti, possono fornire degli indici percentuali di miglioramento delle prestazioni grazie all'implementazione della tecnica di diversità nel tempo e del selection combining. Sia sempre $SNRdB = 20$ avremo:

$$\Delta PAM = \frac{((PeSimulata-Diversity)PAM - (PeSimulata-Fading)PAM)}{(PeSimulata-Fading)PAM} \approx -0,6 = -60\%,$$

$$\Delta PSK = \frac{((PeSimulata-Diversity)PSK - (PeSimulata-Fading)PSK)}{(PeSimulata-Fading)PSK} \approx -0,9 = -90\%,$$

$$\Delta QAM = \frac{((PeSimulata-Diversity)QAM - (PeSimulata-Fading)QAM)}{(PeSimulata-Fading)QAM} \approx -0,99 = -99\%.$$

e per il PPM:

$$\Delta PPM = \frac{((PeSimulata-Diversity)PPM - (PeSimulata-Fading)PPM)}{(PeSimulata-Fading)PPM} \approx -0,99 = -99\%.$$

Come vediamo dai risultati, interpretando i valori percentuali come indici di riduzione delle probabilità di *PeSimulata – Diversity* rispetto a *PeSimulata – Fading*, anche in questo caso gli unici due “competitor” sono PPM e QAM. Le tecniche di diversità infatti riducono le probabilità di errore circa del 99%, ovvero circa due ordini di grandezza, migliorando così le prestazioni generali del sistema.

Non è poi da trascurare il PSK, che sia nelle precedenti che nelle correnti valutazioni prestazionali sembra aver avuto risultati accettabili, con peggioramenti del 900% in presenza di fading ma miglioramenti del 90% con tecnica di diversità temporale. Nonostante ciò la sua efficienza spettrale è più ridotta del 16-QAM poiché questi utilizza sia ampiezza che fase per la trasmissione, a differenza del PSK che modula in fase.

Conveniamo quindi che, al netto di probabilità di errore che possano essere considerate accettabili ed eventualmente valori di SNR migliori di 20dB, il 16-QAM risulta essere la migliore tra le modulazioni considerate in termini di “equilibrio” tra *Pe*, miglioramenti con tecniche di diversità ed efficienza spettrale con un numero così elevato di simboli.